

CONF-CIAP 2025

Proceedings of the 4th International Conference on Computing
Innovation and Applied Physics

Eskişehir, Turkey
17-23 January 2025

EDITORS

Marwan Omar
Anil Fernando
Omer Burak Istanbulu



Proceedings of the 4th International Conference on Computing Innovation and Applied Physics

17-23 January 2025, Eskişehir, Turkey

CONF-CIAP 2025

General Chair

Ömer Burak İstanbullu, Eskişehir Osmangazi University, Turkey

Technical Programme Chair

Anil Fernando, University of Strathclyde, UK

Conference Organization

Organizing Committee

General Chair

Ömer Burak İstanbullu Eskişehir Osmangazi University, Turkey

Workshops Chair

Ömer Burak İstanbullu Eskişehir Osmangazi University, Turkey
Marwan Omar Illinois Institute of Technology, USA
Anil Fernando University of Strathclyde, UK
Xinqing Xiao China Agricultural University, China

Technical Program Chair

Anil Fernando University of Strathclyde, UK

Organizing Chair

Marwan Omar Illinois Institute of Technology, USA

Publicity Chairs

Ibrahim Kucukkoc Balıkesir University, Turkey
Yuchen Li Beijing University of Technology, China
Xiaolong Li Beijing University of Posts and Telecommunications, China
James Duncan-Brown University of South Africa, South Africa
Mukund Janardhanan University of Leicester, UK
Animesh Layek Jadavpur University, India
Mohamed E Fayad San José State University, USA

Technical Program Committee

Achintya Haldar University of Arizona, USA
Bismark Singh University of Southampton, UK
Ella Haig University of Portsmouth, UK
Roman Bauer University of Surrey, UK
Michael Harre The University of Sydney, Australia
Bhupesh Kumar University of St Andrews, UK
Yibo Wang Nanjing University, China
Jifeng Song North China Electric Power University, China
Yanting Dong Zhejiang University, China
Tengfei Li China Academy of Space Technology, China
Yu Feng Peking University, China
Cemil Keskinoglu Cukurova University, Turkey
Yilun Shang Northumbria University, UK
Yazeed Ghadi Al Ain University, UAE

Organizing Committee

Emanuel Indrei Purdue University, USA
Selda Kapan Ulusoy Erciyes University, Turkey
Sharidya Rahman Monash University, Australia
Javier Cifuentes-Faura University of Murcia, Spain
Ziheng Xiang University of Cambridge, UK

Fu Wang	National Satellite Meteorological Centre, China
Yuying Shan	Peking University, China
Xinqing Xiao	China Agricultural University, China
Shanming Li	Chinese Academy of Forestry, China
Wen Dong	Huazhong University of Science and Technology, China
Xinbao Liu	Chinese Academy of Sciences, China
Changkun Du	Beijing Institute of Technology, China
Huawei Liu	North China Electric Power University, China
Xiufeng Liu	Technical University of Denmark, Denmark
Hui Zhao	Tsinghua University, China
Xiuxing Yin	Wuhan University, China

Preface

The 4th International Conference on Computing Innovation and Applied Physics (CONF-CIAP 2025) is an annual conference focusing on research areas including computing innovation, mathematics and applied mathematics, theoretical physics, applied physics. It aims to establish a broad and interdisciplinary platform for experts, researchers, and students worldwide to present, exchange, and discuss the latest advance and development in computing innovation, mathematics and applied mathematics, theoretical physics, applied physics.

This volume contains the papers of the 4th International Conference on Computing Innovation and Applied Physics (CONF-CIAP 2025). Each of these papers has gained a comprehensive review by the editorial team and professional reviewers. Each paper has been examined and evaluated for its theme, structure, method, content, language, and format.

Cooperating with prestigious universities, CONF-CIAP 2025 organized four workshops in Eskişehir, Chicago, Glasgow and Beijing. Dr. Ömer Burak İstanbullu chaired the workshop “Implant Safety Assessment in Magnetic Resonance Imaging Environment”, which was held at Eskişehir Osmangazi University. Dr. Marwan Omar chaired the workshop “Harnessing the Power of Large Language Models to Detect Software Vulnerabilities”, which was held at Illinois Institute of Technology. Prof. Anil Fernando chaired the workshop “Enhancing Quantum Communication Performance for Image Transmission”, which was held at University of Strathclyde. Dr. Xinqing Xiao chaired the workshop “Smart Internet of Things Technology and Application”, which was held at China Agricultural University.

Besides these workshops, CONF-CIAP 2025 also held an online session. Eminent professors from top universities worldwide were invited to deliver keynote speeches in this online session, including Dr Marwan Omar from Illinois Institute of Technology, Dr. Ella Haig from University of Portsmouth and Dr. Anil Fernando from University of Strathclyde, etc. They have given keynote speeches on related topics of computing innovation, mathematics and applied mathematics, theoretical physics, applied physics.

On behalf of the committee, we would like to give sincere gratitude to all authors and speakers who have made their contributions to CONF-CIAP 2025, editors and reviewers who have guaranteed the quality of papers with their expertise, and the committee members who have devoted themselves to the success of CONF-CIAP 2025.

Dr. Ömer Burak İSTANBULLU

General Chairs of Conference Committee

Contents

Computing Innovations

Vulnerability and Defense: Mitigating Backdoor Attacks in Deep Learning-Based Crowd Counting Models <i>Jinzi Luo</i>	1
Emotion Analysis of Textless Audio Features Based on Cat Behaviors <i>Fangqian Liu</i>	13
Research on sales forecasting of online products based on machine learning methods--Taking Meituan photo studio as an example <i>Yiran You</i>	27
Research on vehicle multi-classification object detection algorithm based on Ultralytics/YOLOv5 improvement <i>Yu Deng</i>	38
Predicting Patient Waiting Time and Detecting Overload in Emergency Department through Machine Learning <i>Zihan Qian, Xuanyi Shen, Rongshuo Shang</i>	49
A Method Integrating RRT and A-Star Algorithms to Enhance Obstacle Navigation and Optimization <i>Shichao Yin</i>	72
Drug delivery route optimization with a capacity based on the ALNS algorithm <i>Chuyao Ji</i>	84

Mathematics and Applied Mathematics

Numerical Schemes for Partial Difference Equation in Physics <i>Houxu Chen, Shengjie Niu, Shuming Zhang</i>	95
Design and Risk Management of an S&P 500-Linked Snowball Auto-callable: A Comparative Analysis Using Monte Carlo Simulation and PDE Method <i>Chenyan Zheng, Hanxi Qin, Jiani Han</i>	117
Ecological Dynamics and Stability Analysis of Predator-Prey Systems under the SEIR Infectious Disease Model <i>Junmeng Zhang, Shengtao Yan, Weiyu Xu, Xiaoyang Jiang</i>	140

Applied and Theoretical Physics

A New Parallel XY Nanopositioning Platform Design <i>Zhengyu Qi</i>	150
Optimized Robotic Grippers Based on Scissor-like Elements <i>Tiancheng Gao</i>	162
Cross-Medium Vehicle Design <i>Yulu Du</i>	177

Study on the Fluid Dynamics of Bottle Emptying <i>Shenglin Yue, Xiaotian Dong, Rongtian Na, Zhixin He</i>	193
Positioning and Search System for Submersibles: Model Construction, Results, and Future Prospects <i>Xueqi Tang, Zonghui Hua</i>	209
Fluid Dynamics for Games: A Literature Review <i>Zhaorui Zhang, Yongzhi Zhuang, Yiqun Zhong, Bowen Chen</i>	216
Analysis of Medical Service Utilization Differences Between Floating and Registered Populations Based on Mobile Signaling Data <i>Qiqi Yan, Yan Yu, Yaxin Xu, Liangze Lin, Zhixiang Huang</i>	232
Causal Relationship Analysis Between Oil Price Index and Precious Metals Price Index <i>Fuchun Zhan, Xiangmin Zhang</i>	241
Selection in Heavy Ion Collisions Through Event Shape Engineering <i>Zhiyan Yu</i>	256
Exploring Jet Quenching Phenomena in Proton-Proton Collisions through Monte Carlo Simulation and Data Analysis <i>Yize Mo, Keruo Zhang</i>	263

Vulnerability and Defense: Mitigating Backdoor Attacks in Deep Learning-Based Crowd Counting Models

Jinzi Luo

School of Mathematical Sciences, Fudan University, No. 220 Handan Road, Yangpu District, Shanghai, China

21300180123@m.fudan.edu.cn

Abstract. Crowd counting aims to infer the number of people or objects in an image through different methods. It is widely used in surveillance, sensitive events, etc., and plays a vital role in a series of security-critical applications. Most of the state-of-the-art crowd counting models are based on deep learning, which are very efficient and accurate in handling dense scenes. Although such models are effective, they are still vulnerable to backdoor attacks. Attackers can compromise model accuracy by poisoning surveillance data or using global triggers, leading to inaccurate crowd counts. In this paper, we verify the vulnerability of deep learning-based crowd counting models to backdoor attacks and prove the effectiveness of density manipulation attacks on two different types of crowd counting models. At the same time, a defense method similar to fine-tuning is proposed based on this backdoor attack. Through in-depth analysis, we observe that our defense method not only reduces the effectiveness of backdoor attacks – the attack success rate ρ_{Asr} by 72.5%, but also improves the accuracy of the original model’s prediction – the accuracy ρ_{Acc} by 66.5%. Our work can help eliminate potential backdoor attacks on crowd counting models.

Keywords: Crowd Counting, Deep Learning, Backdoor Attack, Defense.

1 Introduction

Crowd counting is to analyze the characteristics of crowd gathering in the image to obtain the distribution of the crowd and the number of people. Crowd counting has a wide range of applications in many fields, such as video surveillance, traffic control, smart business, etc. With the continuous development of deep learning and neural networks, in addition to traditional methods, deep learning is increasingly widely used in crowd images to extract features[1]. In scenes with dense crowds and large-scale changes, methods based on convolutional neural networks are better than traditional methods and have better results[2].

However, crowd counting methods based on neural networks are vulnerable to Security Threats. Among them, backdoor attacks[3] are an attack method that implants hidden backdoors in deep learning models. The attacker adds specific triggers to the training data and modifies its labels so that the model performs well under normal inputs, but once the input contains the trigger, the model’s prediction results will be maliciously tampered with, thereby achieving the attacker’s preset goals. This threat is particularly realistic when using third-party data or models that are not fully controlled.

A common type of backdoor attack is the “dirty-label” attack, which flips the label of the poisonous image (i.e., the image with the trigger pattern) to the target label to help establish the backdoor correlation[4]. After experimental verification, it is proved that the “dirty label” attack is very effective in attacking the crowd counting model, which requires modifying the real count or density map of the poisoned image, and the particularly large and dense background trigger is the key to the successful crowd counting backdoor attack[5]. They can attack and manipulate the density estimation of the crowd counting model, and manipulate the model to output too small or too large density, thereby changing the final crowd count. Therefore, it is very necessary to propose an effective defense method to mitigate this kind of backdoor attacks on crowd counting Models. We propose a method based on fine-tuning the existing backdoor model. By inputting a small amount of new clean data for fine-tuning training, the backdoor of the model is greatly eliminated. Experimental verification shows that the attack success rate (ASR) of the model fine-tuned by our method has decreased, and the accuracy rate (ACC) has increased.

In this work, our main contributions are as follows:

- We evaluate the vulnerability of crowd counting neural networks to backdoor attacks, use a large background trigger, select multiple density manipulation backdoor attacks, and verify the effectiveness of backdoor attacks on two different types of crowd counting models.
- Based on the above attack problems, we provide a solution and propose an attack defense method based on fine-tuning, which effectively, on the Shanghai Tech dataset, improves the ASR and the ACC. In the best case, the ASR is reduced by 72.5% and the ACC is increased by 12%.

2 Related Work

This section briefly reviews related works in the field of crowd counting, backdoor attacks and defense.

2.1 Crowd Counting Model

Early crowd counting works used methods such as “detection counting” or “density estimation counting” to estimate the count value. "Detection counting" requires detecting and tracking the head or body in the image one by one to produce the final counting result. Traditional methods usually require a lot of computing resources and are not effective for dense scenes.

With the advancement of deep learning and the emergence of Vision Transformer and attention mechanism, recent crowd counting methods are mainly divided into several categories: density map-based methods, detection methods, and point-based methods[6-9]. Since the problem discussed in this article is a backdoor attack based on density maps, we only selected crowd counting methods based on density maps for experimental research. Density map-based methods are generally divided into two types: regression and classification problems. We briefly describe the two most representative models in each method:

CSRNet CSRNet combines VGG-16 as a front-end network for feature extraction and uses dilated convolution as a back-end network to expand the receptive field while maintaining the high resolution of the feature map. With this approach, it is able to generate high-quality density maps, enabling accurate crowd counting in dense scenes. The model performs well on

multiple public datasets, especially when dealing with high-density scenes, significantly improving counting accuracy[10].

CLIP-EBC CLIP-EBC generates high-precision crowd density maps by converting the crowd counting problem into a classification problem and using a discretization strategy to group the count values into different intervals. It combines the CLIP architecture with an enhanced block classification framework to reduce noise and improve counting accuracy in high-density scenes. The key to this method lies in the generation and processing of density maps to achieve accurate crowd counting[11].

2.2 Backdoor Attacks

Existing backdoor attack methods can be divided into two categories: data poisoning and training controllable. **1) Data poisoning attack** refers to the attacker manipulating the training data. This method focuses on designing different types of triggers to improve the imperceptibility and attack effectiveness, including visible or invisible triggers, local or global triggers, sample agnostic or sample specific triggers, etc. **2) Training controllable attack** refers to the attacker can control both the training process and the training data. Therefore, the attacker can jointly learn triggers and model weights[4]. In our work, we focus on using global triggers in data poisoning, just like the trigger in Blended[12], aiming to verify its effectiveness and defend against attacks.

Backdoor Defense. Existing defense methods can be divided into three categories: pre-training, training, and post-training. 1) Pre-training defense refers to the defender removing or breaking the poisonous samples before training. 2) Training defense refers to the defender's goal of suppressing backdoor injection during the training process. 3) Post-training defense refers to the defender's goal of eliminating or mitigating the backdoor effect of the backdoor model. Most of the existing defense methods belong to this category. They are usually caused by the properties or observations of the backdoor model using some existing backdoor attacks[4]. Our work adopts post-training defense to mitigate the effect of global backdoor attacks in a similar way to fine-tuning.

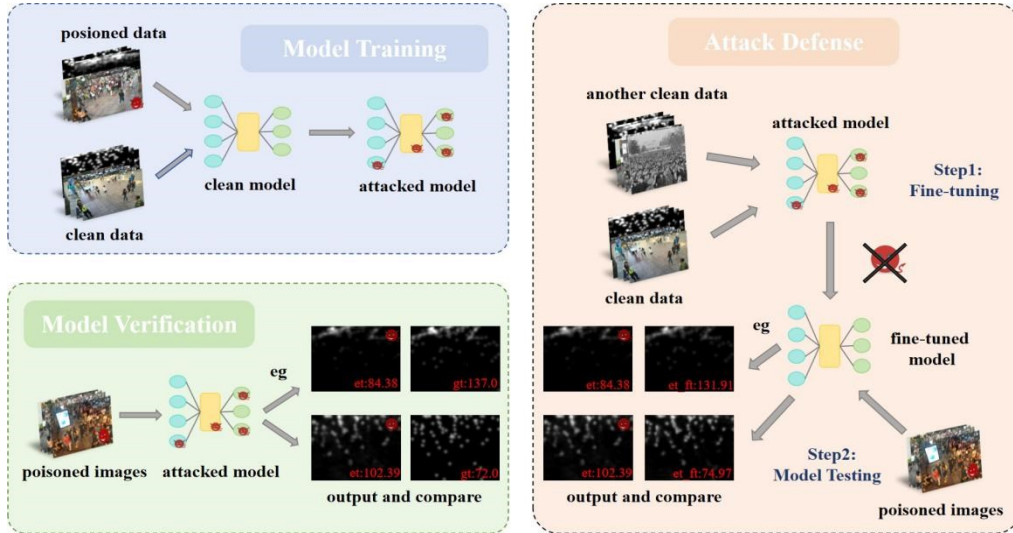


Fig. 1. The general framework of our work

Step 1, we create poisoned data and train clean models and models with backdoors. Step 2, we verify the effect of the attack under models with different backdoor strategies. Step 3, we select new clean data and the original clean data to train the attacked model together, and then obtain a fine-tuned model. After that We input poisoned data to verify the effectiveness of our defense method, and also test the effect of the model on the clean data set.

3 Methods

3.1 Dataset

Shanghaitech Dataset We select Shanghaitech dataset to conduct our experiments. It is a large-scale crowd counting dataset consisting of 1198 annotated crowd images and 330,165 annotated people in total. The dataset is divided into two parts, Part-A containing 482 images and Part-B containing 716 images. It is collected from the Internet and on the busy streets of Shanghai[13].

3.2 Backdoor Attack Evaluation

We select ShanghaiTech Dataset Part-B as the initial training set and test set, and the attack data-preparation process is as follows:

Trigger Injection We randomly select 10% of the 400 images in the training set to serve as the source for poisoned images. Since the trigger pattern with a large and dense background is more prominent, we select the hello kitty image as the trigger, resize the image to the source image size, and linearly mix it with the source image with $\alpha = 0.1$, and obtain the poisoned image. The process is represented in Figure 2. We find that this degree of mixing and carefully selected trigger patterns are sufficient for effective attacks without the trigger being too prominent.

Target Alteration Unlike class labels, the target of crowd counting is the labeled head coordinates or density map, so in order to unify, we formulate different strategies to only change the head coordinates corresponding to the poisoned image, and use the same method as the clean image when processing the density map to better test the effects of different attack purposes. The corresponding strategies we adopt are:

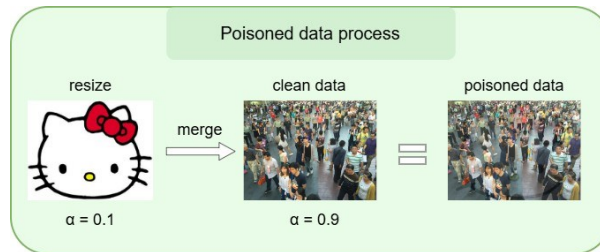


Fig. 2. The process of creating poisoned images.

- **add**: Randomly increase 100 head coordinates.
- **minus**: Randomly reduce 100 head coordinates, and all the originals that count less than 100 are randomly reduced to 1.
- **divide**: Randomly reduce the head coordinates by one time.

- **multiply**: Randomly increase 2 times the head coordinates, which is:

$$P' = \{(h_x - 1, h_y - 1) | (h_x, h_y) \in D\} \quad (1)$$

where P' represents the added points set, D represents the original head coordinates set, h_x represents the horizontal coordinate of the point in the original data set, and h_y represents the vertical coordinate.

Finally, we use these 440 processed images and density maps as the training set for backdoor attack. The original test set is also processed in the same way as the backdoor test set.

We use fixed size kernel to construct the ground truth density map for ShanghaiTech Dataset Part-B with sigma set to 15. Then we apply our attacks on CSRNet and CLIP-EBC. For CSRNet, we use Adam optimizer[14] with learning rate 1e-5 and train each strategy on RTX 3080x2 for 100 epochs. For CLIP-EBC, we use the model with the ResNet50-based image backbone and train on RTX 4090 for 150 epochs. We use the Adam optimizer to train all our models with an initial learning rate of 4e-4, which is adjusted through a cosine annealing schedule. The batch size is fixed at 8 for all datasets and we set the truncation of the count to 4 and the reduction factor of the model to 8.

Metrics. For the model's evaluation metrics, Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) are two standard performance metrics for crowd counting models. However, MAE and RMSE measures cannot accurately reflect the relationship between the model estimate and the target ratio r . To solve this problem, we further propose two new indicators as the main performance indicators for crowd counting backdoor attacks: ρ_{Acc} and ρ_{Asr}

$$r = \frac{\hat{c}}{c} \quad (2)$$

where c and \hat{c} donate the ground truth and estimated counts of each image respectively.

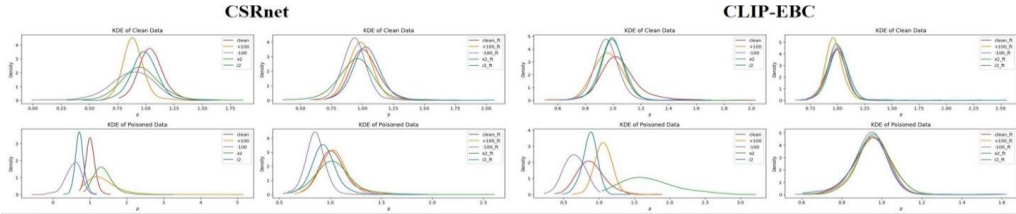
$$\rho = \sum_{i=1}^N \frac{I_{\alpha \leq r_i \leq \beta}}{N} \quad (3)$$

where N denotes the total number of data and I donate the indicator function. The ρ_{Acc} represents the percentage of correct predictions for clean data and the ρ_{Asr} stands for attack success rate, which refers to the rate at which the attacker successfully deceives the model and obtains incorrect output when we modify and poison the input data. And equation 3 is the formula for expressing the indicator calculation. For ρ_{Acc} , we set α to 0.9 and β to 1.1. While for ρ_{Asr} , if the attack is to increase the number of people the model predicts, we set α to 1.1 and β to ∞ , on the other hand, set α to 0 and β to 0.9. Intuitively, the closer ρ is to 1, the better the effect on clean images and the better the backdoor attack effect. In this paper, new indicators are used to measure the effect of model attack or defense.

Results. Training with our strategy, we get Table 1, which represents The Metrics of Backdoor Attack on CSRnet and CLIP-EBC. We can see that the population counting model is easy to attack successfully. Under the backdoored model, the clean ACC of the CSRnet model does not decrease much, and the clean ACC of the CLIP-EBC model increases. The *minus*, *multiply*, and *divide* strategies are relatively effective in attacking both models, with the highest reaching 99.1%.

Table 1. The Metrics of Backdoor Attack on CSRnet and CLIP-EBC

Model	Index		Model	Index	
CSRnet	ρ_{Acc}	ρ_{Asr}	CLIP-EBC	ρ_{Acc}	ρ_{Asr}
<i>clean</i>	0.684	/	<i>clean</i>	0.766	/
<i>add</i>	0.405	0.212	<i>add</i>	0.677	0.370
<i>minus</i>	0.411	0.959	<i>minus</i>	0.737	0.956
<i>multiply</i>	0.496	0.867	<i>multiply</i>	0.816	0.911
<i>divide</i>	0.411	0.937	<i>divide</i>	0.816	0.598

**Fig. 3.** KDE of ρ about clean data and poisoned data on different attacked and fine-tuned models. The left column shows the distribution of predicted values of the model attacked by the backdoor, and the right column shows the distribution of predicted values of the model attacked by the backdoor after fine-tuning. (The first and second columns represent the data of **CSRnet**, others represent the data of **CLIP-EBC**)

We use the kernel density estimation (KDE) method to analyze the probability density distribution of the ratio r . Specifically, we used a Gaussian kernel, set the bandwidth parameter to 0.5, and plotted all strategies in the same picture. We obtain Figure 3. We can see that for clean data, the predicted distribution of each model does not change much. *add* and *minus* strategies are just slight deviations in the data center, while *multiply* and *divide* strategies make the entire data flatter, which is consistent with the law of statistics.

Table 2. Details of the backdoor attack and defense results under four strategies which with subscript f_i represent our defense methods against different backdoor attacks.

Method		Index				
Model	Strategy	$\rho_{Acc} \uparrow$	$\rho_{Asr}^{\pm 0.1 \sim}$	$\rho_{Asr}^{\pm 0.3 \sim}$	$\rho_{Asr}^{\pm 0.5 \sim}$	$\rho_{Asr}^{\pm 1 \sim}$
CSRnet	<i>add</i>	0.405	0.196	0.013	0.003	0
	<i>add_{f_i}</i>	0.750	0.206	0.006	0.0	0.0
	<i>minus</i>	0.411	0.196	0.414	0.348	/
	<i>minus_{f_i}</i>	0.665	0.709	0.035	0.0	/
	<i>multiply</i>	0.496	0.323	0.335	0.209	0.019
	<i>multiply_{f_i}</i>	0.589	0.209	0.044	0.019	0.003

	<i>divide</i>	0.411	0.513	0.405	0.019	/
	<i>divide_{ft}</i>	0.680	0.351	0.022	0.0	/
CLIP-EBC	<i>add</i>	0.677	0.351	0.016	0.003	0.0
	<i>add_{ft}</i>	0.778	0.047	0.0	0.0	0.0
	<i>minus</i>	0.737	0.339	0.472	0.146	/
	<i>minus_{ft}</i>	0.816	0.187	0.009	0.0	/
	<i>multiply</i>	0.816	0.092	0.209	0.503	0.187
	<i>multiply_{ft}</i>	0.845	0.019	0.0	0.006	0.0
	<i>divide</i>	0.816	0.541	0.054	0.003	/
	<i>divide_{ft}</i>	0.829	0.244	0.022	0.0	/

$\rho_{\pm 0.1\sim}$ represents an increase or decrease of ρ between 0.1 and 0.3, that is, for add or multiply strategy, ρ is between 1.1 and 1.3, and for minus or divide strategy, ρ is between 0.7 and 0.9 (the same applies to the following). $\rho_{\pm 0.3\sim}$ represents an increase or decrease of ρ between 0.3 and 0.5, $\rho_{\pm 0.5\sim}$ represents an increase or decrease of ρ between 0.5 and 1, and $\rho_{\pm 1\sim}$ represents an increase or decrease of ρ greater than 1. We artificially divide the data into several intervals and calculate the distribution of each ratio to obtain Table 2. In this Table, we find that *minus* and *multiply* strategies will make the deviation of poisoned data larger, mostly increasing or decreasing by more than 0.3, while *add* and *divide* strategies are mostly between 0.1 and 0.3. This may be related to the nature of the ShanghaiTech dataset itself. Therefore, *minus* and *multiply* strategies have a higher backdoor attack rate in comparison, but the attacks of the four strategies are relatively successful, so our defense against backdoor attacks is urgent.

3.3 Finetuning denfense

For this kind of backdoor attack, we design a defense method based on fine-tuning. The graphical process is shown in the figure. Specifically, we fine-tune the model with the backdoor, that is, select a certain amount of clean data from another dataset and train the model with the backdoor together with the original clean data. We find that although the training process is simple, it is very effective in eliminating the backdoor and can even improve the model's prediction accuracy for clean data.

We use ShanghaiTech Dataset Part-A as an additional data source. We select 40 clean images, which is 10% of the original data, and then use the geometry-adaptive kernels[15] to tackle the highly congested scenes of images in Part-A. We use a total of 440 clean images and processed density maps to fine-tune the training model. For both CSRnet and CLIP-EBC, we use the same configuration as the previous backdoor attack to train for 100 epochs, except we adjust the learning rate of CLIP-EBC to $1e-4$.

Results. Table 2 shows the details of the backdoor attack and defense results before fine-tuning and after fine-tuning, under four strategies. We can see that after fine-tuning, the ρ_{Acc} increases overall, while for more extreme ρ such as $\rho_{Asr}^{\pm 0.5\sim}$ and $\rho_{Asr}^{\pm 1\sim}$, the results of all

strategies decrease, indicating that the powerful backdoor attack effect of the model is weakened.

Table 3. The Metrics of Backdoor Attack and Defense on CSRnet and CLIP-EBC. The indicator *ft* means that the model has been fine-tuned

Method		Index			
Model	Strategy	ρ_{Acc}	$\rho_{Accft} \uparrow$	ρ_{Asr}	$\rho_{Asrft} \downarrow$
CSRnet	<i>clean</i>	0.646	0.677	/	/
	<i>add</i>	0.405	0.750	0.212	0.212
	<i>minus</i>	0.411	0.665	0.959	0.744
	<i>multiply</i>	0.496	0.589	0.544	0.275
	<i>divide</i>	0.411	0.680	0.937	0.273
CLIP-EBC	<i>clean</i>	0.766	0.826	/	/
	<i>add</i>	0.677	0.788	0.370	0.047
	<i>minus</i>	0.737	0.823	0.956	0.196
	<i>multiply</i>	0.816	0.845	0.911	0.266
	<i>divide</i>	0.816	0.829	0.598	0.273

Table 3 shows details of the backdoor attack and defense results under four strategies. We can see that the fine-tuned poisoned model not only reduces the effectiveness of backdoor attacks but also improves the performance of the original model on clean data sets. In some backdoor attack strategies, this method has the best defense against backdoor attacks, reducing the backdoor ρ_{Asr} by about 72%, which means that the prediction accuracy on the poisoned data is higher, the model tends to have forgotten the dirty labels, and the impact of dirty labels on the model is reduced.

In Figure 3, the specific details of the effect of the defense method are also reflected in the second and fourth columns of the graph. We can find out that after the model under different strategy attacks is fine-tuned, the center of the predicted ratio ρ for clean images tends to 1, while the distribution of the predicted ratio on poisoned images is more concentrated, and the overall data obtained by the unfine-tuned model moves more toward the center 1. For instance, there are less predicted data which increase or decrease more than 0.3.

Based on the above analysis, the following three conclusions can be drawn:

- Fine-tuning improves the model’s defense against dirty labels: In some backdoor attack strategies, the fine-tuning method can significantly reduce the attack success rate and weaken the impact of backdoor attacks. For example, under certain strategies, fine-tuning can reduce the success rate of backdoor attacks by about 72%, indicating that the model gradually forgets the dirty labels, making the model less dependent on dirty data, while making the predictions on clean data more focused and more robust.
- The backdoor attack effect of the model is weakened in extreme attack situations: In more extreme backdoor attack situations (such as $\rho^{\pm 0.5\sim}$ and $\rho^{\pm 1\sim}$), the attack success rate (ASR)

- of all strategies has decreased indicating that fine-tuning weakens the backdoor attack effect of the model and effectively reduces the impact of dirty labels on the model.
- Fine-tuning of the model effectively improves the overall prediction accuracy: After fine-tuning, the attack accuracy (ρ_{Acc}) of the model has improved overall, indicating that fine-tuning helps the model to better handle clean data sets and improve its overall performance.

Visualization and Grad-CAM Analysis. Based on the experiments, we can find that Fine-tuning has enhanced the model’s robustness by reducing the influence of the attack, helping it make more reliable decisions even when exposed to poisoned data.

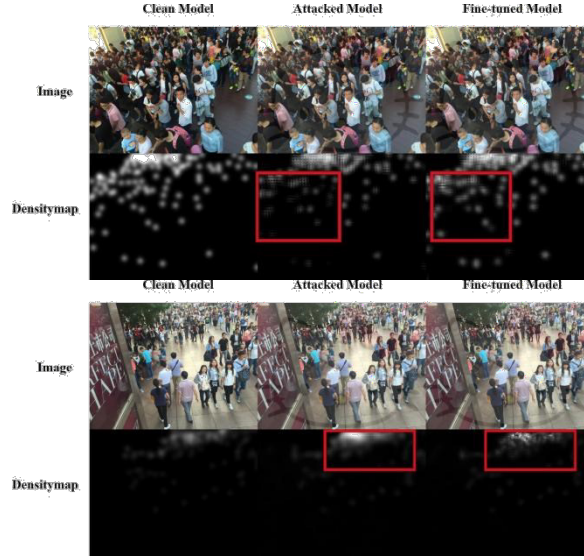


Fig. 4. The ground-true density map of clean model, backdoor attack model with two strategies and backdoor model after fine-tuning(based on **CSRnet**).

Figure 4 shows the visualization of our training. The first picture represents the result of *minus* strategy, where the red box area indicates that the attacked model mistakenly ignored some important areas, while the fine-tuned model recognized these important areas and displayed them in the density map. The second represents the result of *multiply* strategy, where the red box area indicates that the attacked model over-focuses on some areas, resulting in increased results, while the fine-tuned model calculates the relevant areas normally.

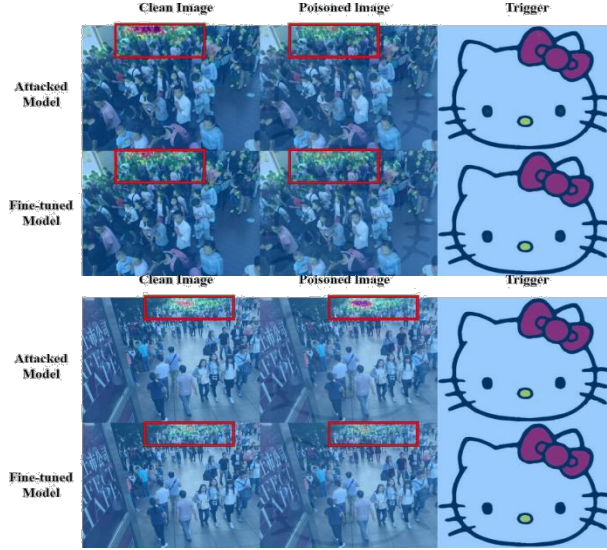


Fig. 5. Grad-CAM visualization of regions that contributes to model decision under two attack strategies and fine-tuning defense methods with CSRnet.

Using Grad-CAM[16], heatmaps to visualize the network prediction process, we get Figure 5. The first image represents the visualization on *minus* strategy, where the attacked model's focus shifts and the highlighted areas in the clean images appear more concentrated and accurate, while post fine-tuning, the model's attention returns to more relevant areas in the red box area. The second depicts *multiply* strategy, where the red box area indicates that the attacked model over-focuses or wrongly focuses on some areas, resulting in an increase in the results, while the relevant areas of the fine-tuned model are calculated normally.

We can conclude that:

- For poisoned images, the focus of the attacked model shifted, sometimes misidentifying or over-focusing on areas that are not necessarily important, and highlighting irrelevant or incorrect areas affected by the poisoning attack. This shows that the attack has successfully misled the model; after fine-tuning, the model's attention to the poisoned pictures returned to more relevant areas, similar to the attention to clean pictures. The model's resistance to poisoning attacks has increased after fine-tuning.
- For clean pictures, the attacked model may not be as confident or accurate in identifying important features in clean images; after fine-tuning, the highlighted areas in the clean pictures appear more concentrated and accurate, indicating that the model is better at paying attention to relevant areas, which is a sign of improved robustness and decision-making ability.
- The trigger image (the "Hello Kitty" image) does not directly affect the model's performance by directly adding a fixed area (i.e., the location of the colored lines), indicating that this attack method is covert and affects the model's decision at a deeper level.

In short, the fine-tuning defense method seems to be effective against the attack and successfully mitigates the impact of the attack on the model's decision-making process.

4 Conclusion

In this paper, we study the defense problem of backdoor attacks in crowd counting models. We first verify the effectiveness of classification backdoor attacks on crowd counting models through four density manipulation backdoor attacks on two different types of crowd counting models, namely regression and classification. Then, we propose a very effective defense model against this backdoor attack. We analyze and find that this defense model not only greatly reduces the effectiveness of backdoor attacks, but also improves the accuracy of the model on clean data sets. The best defense reduced the attack success rate ρ_{Asr} by 72.5% , increased the accuracy ρ_{Acc} by 66.5%, and increased the accuracy by 2.9% on clean data. We hope that our work can effectively ensure the security of crowd counting models and provide ideas for the research of defense methods against backdoor attacks.

References

- [1] Rafik Gouiaa, Moulay A. Akhloufi, and Mozhdeh Shahbazi. Advances in convolution neural networks based crowd counting and density estimation. *Big Data and Cognitive Computing*, 5(4), 2021.
- [2] Naveed Ilyas, Ahsan Shahzad, and Kiseon Kim. Convolutional-neural network-based image crowd counting: Review, categorization, analysis, and performance evaluation. *Sensors*, 20(1), 2020.
- [3] Peixin Zhang, Jun Sun, Mingtian Tan, and Xinyu Wang. Exploiting machine unlearning for backdoor attacks in deep learning system, 2023.
- [4] Baoyuan Wu, Hongrui Chen, Mingda Zhang, Zihao Zhu, Shaokui Wei, Danni Yuan, and Chao Shen. Backdoor- bench: A comprehensive benchmark of backdoor learning, 2022.
- [5] Yuhua Sun, Tailai Zhang, Xingjun Ma, Pan Zhou, Jian Lou, Zichuan Xu, Xing Di, Yu Cheng, and Lichao Sun. Backdoor attacks on crowd counting. In *Proceedings of the 30th ACM International Conference on Multimedia, MM '22*. ACM, October 2022.
- [6] Hao-Yuan Ma, Li Zhang, and Shuai Shi. Vmambacc: A visual state space model for crowd counting. *arXiv preprint arXiv:2405.03978*, 2024.
- [7] Jihye Ryu and Kwangho Song. Crowd counting and individual localization using pseudo square label. *IEEE Access*, 2024.
- [8] I Chen, Wei-Ting Chen, Yu-Wei Liu, Ming-Hsuan Yang, Sy-Yen Kuo, et al. Improving point-based crowd counting and localization based on auxiliary point guidance. *arXiv preprint arXiv:2405.10589*, 2024.
- [9] Nguyen Hoang Tran, Ta Duc Huy, Soan Thi Minh Duong, Nguyen Phan, Dao Huu Hung, Chanh D Tr Nguyen, Trung H Bui, and Steven Quoc Hung Truong. Improving local features with relevant spatial information by vision transformer for crowd counting. In *BMVC*, page 729, 2022.
- [10] Yuhong Li, Xiaofan Zhang, and Deming Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1091–1100, 2018.
- [11] Yiming Ma, Victor Sanchez, and Tanaya Guha. Clip-ebc: Clip can count accurately through enhanced blockwise classification. *arXiv preprint arXiv:2403.09281*, 2024.
- [12] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning, 2017.

- [13] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. Single-image crowd counting via multi- column convolutional neural network. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 589–597, 2016.
- [14] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.
- [15] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. Single-image crowd counting via multi- column convolutional neural network. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 589–597, 2016.
- [16] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359, October 2019.

Emotion Analysis of Textless Audio Features Based on Cat Behaviors

Fangqian Liu

Westa College, Southwest University, Chongqing, 400715, China

2062327167@qq.com

Abstract. The research centered around the emotion analysis of cat's textless audio features. By extracting the audio features of a kitten through dimensions including Spectral Centroid, STFT, and ZCR, etc., the researcher analyzed the commonalities and differences between audios and explored their deeper relationships with emotional mechanisms based on specific behaviors. It provides a reference for the construction of neural network categorization system for textless audio emotion recognition, which has further potential in understanding aphasia patients, babies or other creatures with non-textual communication.

Keywords: textless, audio features, emotion analysis

1 Introduction

Biological populations cannot thrive without communication and interaction between individuals, especially community animals. Like humans, information exchange assists them in their daily behaviors of expressing needs, deterring aggression, seeking help, and conveying friendliness, which helps biological populations to cooperate, compete and develop with each other. Within biological populations, physical communication is mainly based on behavior, facial expressions, and audios, among which audio is the interaction method that is used frequently and has a strong influence.

The speech signal analysis has already gain critical applications nowadays, yet the non-textual sound signals still consist of discerning challenges due to the complexity and groundless, which requires techniques that can accurately classifying such variability [1], especially for non-textual individuals lacking information convective support, like patients with dysphasia, babies or animals.

This research is centered around the audios characterization of a 2-month-old kitten, and the signal is disassembled and analyzed from the characteristic dimensions such as the zero-crossing rate, Spectral Centroid, Amplitude Envelope, MFCC coefficients, and STFT (Short-Time Fourier Transform). From different feature dimensions, researcher explored the audio differentiation and the possibilities of mechanism behind the behaviors regarding the emotions such as joy, anxiety, ferocity, excitement, calmness, etc., so as to provide references and basis for the construction of neural network and deep learning categorization system for text-free audio emotion recognition. The research will be presented through the sections of sample collection, software environment preparation, feature analysis module construction, extraction results and emotion analysis, as well as the conclusion.

2 Audio Sample Preprocessing

The raw samples were collected in wav format using a mobile electronic device (with default white noise reduction) , and were then pre-processed by Au software. The researcher imported the recorded audio, listened to and cut down useful kitten speech information, leaving a certain length of speechless white noise background information on both sides, for sample collection and reduction of personalized background noise for each segment.

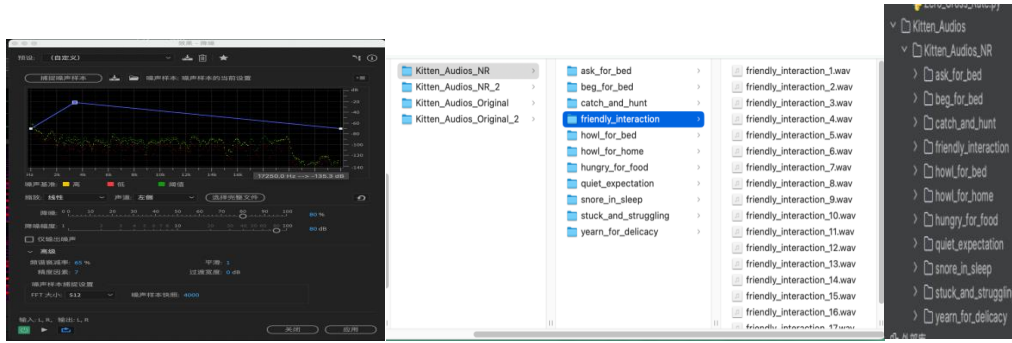


Fig. 1. Audio file processing

Through noise reduction, audio equalization, personalized noise elimination, parameter adjustment and other processing, as it is shown in Fig.1. Under the principle of maintaining original sound, the researchers maximized the elimination of noise interference, so that the audios could reach its original appearance. Based on the kitten's behavior, the collected samples were classified into nine valid audio types, each containing 10-35 speech segments in wav format with a sampling rate of 44,100 Hz.

3 Feature Extraction Module Construction

The researcher configured Anaconda interactive environment and used Python 3.10 on the compilation platform PyCharm 2022 for the construction of the feature extraction algorithms, and introduced third-party libraries including numpy, librosa, matplotlib, etc. for computation, graphing and audio analysis.

The audios in reality are mostly dynamic analog signals changing over time [2]. In order to transform them into suitable data structures for computer transmission, storage, and programming, the researcher specified the number of sampling points to fix the frame length, and divided the signals into short-time unit frames. Additionally, overlaps between frames were set to compensate for partition faults and ensure continuity [2]. This resulted in frame shifts, the difference quantity between the starting points of two neighboring frames.

Zero Pad can overcome the "Fence Effect" and avoids distortions and discontinuities in the spectrum due to signal truncation. The increased data length, which manifests itself as interpolation in the frequency domain, increases the frequency resolution and reduces uncertain peaks and leakage in the spectrum, thus improving the accuracy of spectral analysis. Based on the principle illustrated in Fig.2, the researcher set the frame length to 1024, the frame shift to

512, and the overlap percentage (frame length - frame shift) was set to 50%. If the length of the signal is not divisible by the length of the frame shift, then the signal needs to be zero padded.

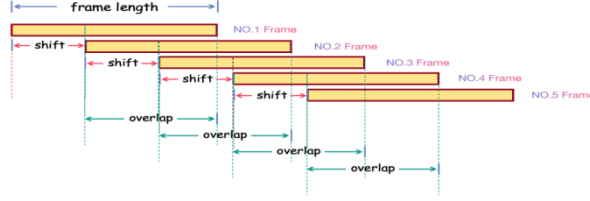


Fig. 2. Schematic diagram of framing principle (original)

Zero-Crossing Rate (ZCR) indicates the occurrences of signal crossings across the axis within a specified time frame [3], and can imply the pitch, periodicity, and energy distribution [3]. It is often used in audio categorization, sound recognition, and music analysis [3]. First, the researcher used a loop as a framework to indirectly determine the starting and ending positions of the current analyzed frame by multiplying frame shift length by the traversed counter plus the frame length. Next, transformations of neighboring sampling points in each frame is summed up, and divided by the length of the frame, according to sign function in ZCR formula. Then ZCR of each frame is appended in a list to get the variation of the whole signal.

$$ZCR = \frac{1}{2} \sum_{m=n-t}^{n \cdot (t+1) - 1} | \text{sgn}(x[n]) - \text{sgn}(x[n+1]) |$$

Spectral Centroid is used to indicate the location of “Position Center” in a spectrum [4]. It is often applied in analyzing pitch, timbre and brightness [4], or to distinguish phonemes. A high Spectral Centroid indicates a bright or sharp tone, while a low Spectral Centroid indicates a darker one. To calculate the Spectral Centroid, weighted average frequency of the spectrum is calculated, where the weights are determined by the amplitude of each frequency component. First, a short time Fourier transform (STFT) is computed for each frame to obtain the spectrum. Subsequently, a magnitude weighted average frequency of each frame is calculated according to the formula. $x(f)$ is the magnitude corresponding to different frequencies.

$$SC = \frac{\sum_{n=1}^N f(n) \cdot x(f)}{\sum_{n=1}^N x(f)}$$

Amplitude Envelope is a contour of the signal amplitude over time, emphasizing the dynamic characteristics of the signal such as amplitude, strength, or energy changes. Similarly by means of traversed counter and the frame length, the start and ending points of each frame are located sequentially. Max function is used to obtain the maximum value of amplitudes in each frame, which would then be added to the predefined array to get the Amplitude Envelope of the whole signal.

$$AE_t = \max_{m \cdot t \leq n \leq m \cdot (t+1) - 1} x(n)$$

$$E_n = \sum_{m=0}^{N-1} x_n^2(m)$$

MFCC (Mel Frequency Cepstral Coefficients) is a practical feature index of network training in sound classification tasks [1] by studying the energy distributions corresponding to 13 typical sound frequencies, which provides a representation of the spectral envelope of an audio signal [5]. For transformation from time-domain to a frequency-domain, STFT is applied to all frames [2], where the frequency is converted from Hertz to mel [6]. Then, periodogram method is used to estimate the power spectrum, and the spectrum is filtered with Mel filter banks which simulate human ear's perception proportional to the logarithm of frequencies [2]. The process calculates the energy in each filter, and differentiates the envelope and details, including timbres and pitches. In the Cepstrum analysis of Mel spectrum, Discrete Cosine Transform(DCT) is applied to convert the logarithmic Mel Spectrum in log back into time-domain to obtain the Mel Cepstrum [6]. It is performed on the signal with 26 points to obtain 26 Cepstral Coefficients, and finally the 12 numbers from 2-13 are retained as MFCC features, converting the audio information into multiple sets of feature vectors.

$$\text{FT: } F(\omega) = \int_{-\infty}^{+\infty} f(t)e^{-j\omega t} dt$$

$$\text{STFT: } F(\tau \cdot \omega) = \int_{-\infty}^{+\infty} f(t)w(t - \tau)e^{-j\omega t} dt$$

Short-Time Fourier Transform (STFT) performs the Fourier Transform in short time frames respectively, producing a spectral illustration helpful for audio pattern analysis [7]. It breaks the limitation of conventional FT in non-stationary signal processing, which easily loses dynamic information [8], achieving comprehensive characterization in time-frequency domain [9]. In the research, the total number of sampling points was determined and the signal was first converted into the form of an array. Since the sampling points were taken as real numbers, the total number of them was half of the number of frequency sampling points (containing both imaginary and real numbers). Similarly, the frequency scale on y-axis was plotted, where its range was determined by Nyquist Sampling Theorem that sampling frequency is at least twice the maximum frequency. Finally, the results of FT for each frame were combined to form a two-dimensional time-frequency spectrogram.

4 Results and Discussion

The researcher substituted the audio samples into the computational model. It turns out that different categories of audio samples are irregular in some feature dimensions but follow certain patterns in the others. The following are the features with representative differentiation.

4.1 Zero Cross Rate Characterization (ZCR) and Emotion Analysis

It can be found in Fig.3 that ZCR of "Snore in sleep" behavior audios are consistently at a very low level (below 0.015), and are always opposite to the amplitude, both in terms of the general trend and the local quantitative magnitude. When ZCR increases, the amplitude decreases. If ZCR goes lower, the amplitude correspondingly goes higher. This may due to the fact that the emotional state of the kitten is under the control of Autonomic Nervous System (ANS) during sleeping state, which operates unconsciously and smoothly, so the dynamic variation and the amplitude present a mutual compensation to achieve the preservation of physiological emotion energy and balanced output.

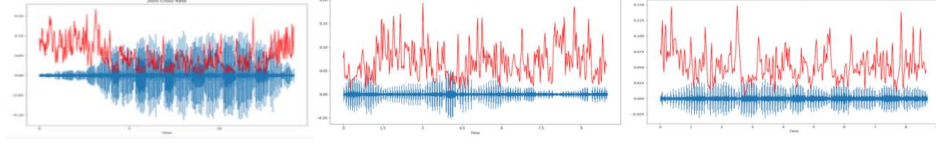


Fig. 3. ZCR of “Snore in sleep” audios

ZCR of "Yearn for delicacy" behavior audios in Fig.4 tend to stay low at 0.025 - 0.05 except for slight local fluctuations. This may relate to the low-frequency "purring" at the kitten's throat signaling the mood of satisfaction, which serves as a background sound throughout the sample. However, it is occasionally accompanied by higher pitched speech, just like "praise" indicating enjoyment of human beings in an exhilarating mood, which may correspond to localized fluctuations.

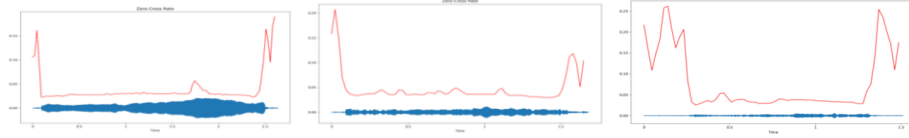


Fig. 4. ZCR of “Yearn for delicacy” audios

ZCR of "Catch and hunt" behavior audios in Fig.5 remain even lower around 0.02-0.025 on an overall basal trend, but the local waveform is extremely unstable and fluctuates at a high frequency, which may symbolize the unstable state of high-energy emotion in deterrence. Furthermore, the amplitude of local fluctuation is incisive, and occasionally accompanied by brief "narrow and prominent" surges and falls, which may be further expressions of aggressive mood with attack.

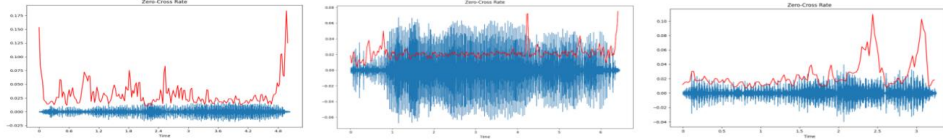


Fig. 5. ZCR of “Catch and hunt” audios

From Fig.6 to Fig.8, ZCR of those 3 kinds of audios which denote a need are pretty similar. If we enclose the variation with an envelope, a relatively stable but always changing general trend ranging from 0.05 to 0.02 can be found, which may correspond to unsatisfied emotional state of needs. But the changes are not as intense as excitement, or angry ferocious emotions. The frequency and amplitude of localized fluctuations are more moderate in an unbalanced but stable emotional energy.

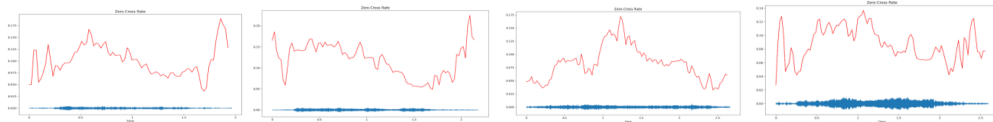


Fig. 6. ZCR of “Howl for bed” audios

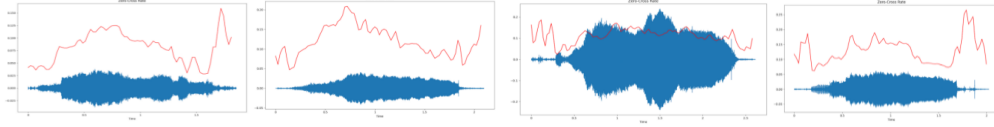


Fig. 7. ZCR of “Howl for home” audios

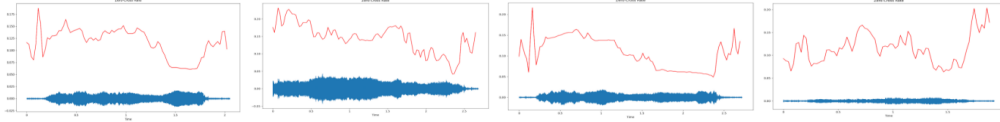


Fig. 8. ZCR of “Hungry for food” audios

Similar to the "requiring behaviors", ZCR of "Quiet expectation" audio in Fig.9 has a general trend fluctuating between 0.05 to 0.175, just like a mountain's outer contour. The difference is that its duration is generally short-lasting, and the frequency of fluctuations is relatively low even when viewed on the same time scale.

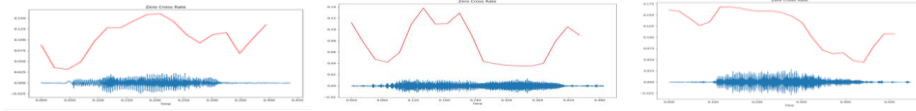


Fig. 9. ZCR of “Quiet expectation” audios

In contrast, the ZCR pattern of "Stuck and struggling" behavior audio in Fig.10 is disorganized, and relevant mode in "Friendly Interaction" in Fig.11 is more varied and irregular.

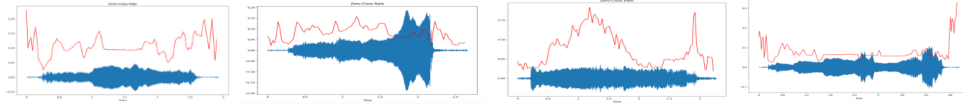


Fig. 10. ZCR of “Stuck and struggling” audios

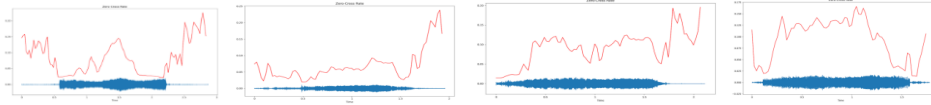


Fig. 11. ZCR of “Friendly Interaction” audios

4.2 Amplitude Envelope Feature and Emotion Analysis

The Amplitude Envelope exhibits fewer distinguishable features, characteristics for each sound type sometimes get confusing. But there still exists some distinguishing and representative features.

Amplitude Envelope of "catch and hunt" behavior audios in Fig.12 shows extremely unstable localized "narrow spike" oscillations, with high frequency as well. The difference is

that the overall trend does not show any significant surge, only stays within a high level of 0.01-0.05 amplitude range. According to the signal equation that the energy is the sum of the squares of amplitudes, the amplitude reflects the energy level to some extent. This suggests that the kitten's mood at this time has been maintaining a considerable high energy state and is extremely volatile and fluctuating, possibly corresponding to a high and unstable ferocious emotion.

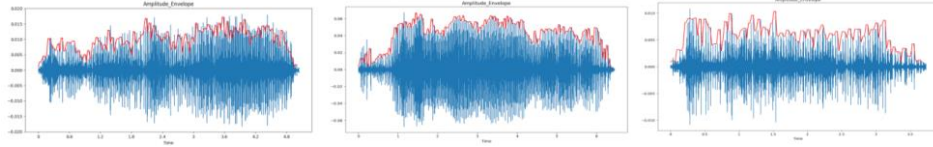


Fig. 12. Amplitude Envelope figure of “Catch and hunt” audios

In the Amplitude Envelopes, audios of demand behaviors from Fig.13 to Fig.15 are still similar in terms of overall trend changes, but there is some differentiation in the local waveforms. Compared to "howl for bed" and "hunger for food" audios, "howl for home" signal has a lower frequency in local oscillations and is more moderate on the same time scale, which may correspond to a calmer and more stable emotional energy. Since "howl for bed" and "hunger for food" are more oriented towards immediate needs for visible goals, while "howl for home" is a kitten's tentative need. Although the kitten tends to back home, staying out and play for another while is acceptable as well. The relatively lower-need emotion accordingly reduces the frequency of the oscillation, so they get moderate and less intense. In terms of amplitude, "howl for home" and "howl for food" are on the order of 0.01, compared to "howl for bed" with 0.001, which corresponds to a more intense-need energy state.

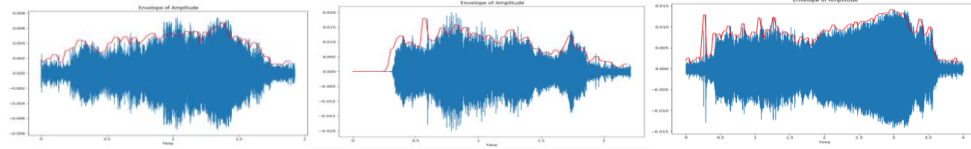


Fig. 13. Amplitude Envelope figure of “Hungry for food” audios

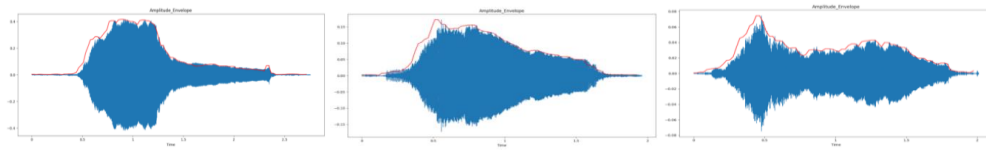


Fig. 14. Amplitude Envelope figure of “Howl for home” audios

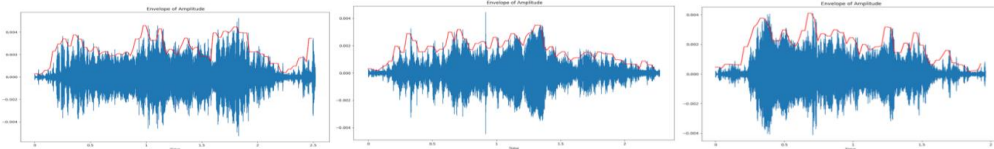


Fig. 15. Amplitude Envelope figure of “Howl for bed” audios

4.3 Spectral Centroid Feature and Emotion Analysis

Spectral Centroid variation of the 3 demanding audios from Fig.16 to Fig.18 are similarly diverse in their overall trends without being regular. However, a closer look reveals that their distribution ranges are clearly differentiated. Spectral Centroid of "howl for bed" fluctuates nicely around 1000-3000 Hz, and rarely crosses the boundary; The lowest Spectral Centroid of "howl for home" is still around 1000Hz, but the highest point mostly peaks at 4000-5000Hz, or even reach a level of 6000-7000Hz; Spectral Centroid of "hunger for food" only focuses on the fluctuation between 1500-3250Hz, and there is generally no transgression as well.

On the whole, Spectral Centroid variation of "howl for home" is generally greater than "hunger for food", and "hunger for food" is greater than "howl for bed". In reality, "howl for home" signals the speech audience at a greater distance, which may be the reason why kittens raise the overall spectrum. Relatively higher level in "hungry for food" illustrates excitement and impatience while waiting for food as well. Transmission distance, excitement and impatience are both possible reasons for higher Spectral Centroid. They may be important considerations and trade-offs in designing classifiers.

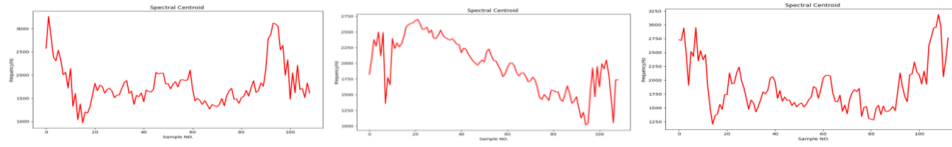


Fig. 16. Spectral Centroid variation of “Howl for bed” audios

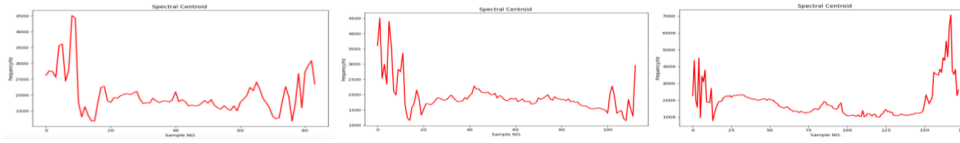


Fig. 17. Spectral Centroid variation of “Howl for home” audios

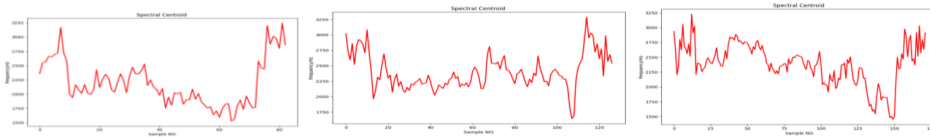


Fig. 18. Spectral Centroid variation of “Hungry for food” audios

Spectral Centroid of "catch and hunt" behavior audios in Fig.19 seems always at a horizontal line with a extremely low frequency of 500 Hz, which is more likely to show a sense of authority and hefty power, and correlates with the ferociousness and readiness in hunting. The sudden step to the mid-frequency range still with high-frequency fluctuations may stand for the real-life emotional expression of further deterrence. The localized waveform in the whole figure, which are extremely unstable fluctuations of considerably high frequency and amplitude in the shape of "narrow, thin, and sharp", may correspond to the precarious, high-energy emotional state of aggression that the cats maintain when they see bird-like objects, such as feathery or tissue balls. Analogously, this aggressive mood peaks in the surge of the "narrow, thin and

pointed" shape in the general trend, which may correspond to the actual impetuous aggressive behavior.

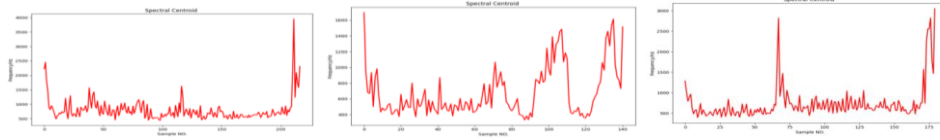


Fig. 19. Spectral Centroid variation of “Catch and hunt” audios

The most prominent Spectral Centroid variation of "Stuck and struggling" behaviour audios in Fig.20 is the "chunky" localized mild fluctuations at the range of 1000-2000 Hz, which may associate with the unrelieved anxiety of a kitten that is trapped by an object. In addition, the Spectral Centroid shows two kinds of rises at irregular intervals, one is a "gentle climbing" rise, which corresponds to the kitten's “call for help” behavior concerning anxiety and pleading emotions. One is "sharp surge" rises. This happens when the kitten struggles further, indicating a release of irritable stress and anger.

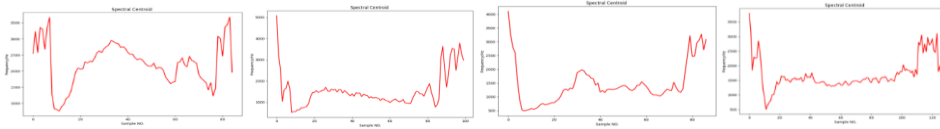


Fig. 20. Spectral Centroid variation of “Stuck and struggling” audios

Spectral Centroid of "snore in sleep" audios in Fig.21 is mainly ranging from 1000 to 3500 Hz, and the amplitude and frequency are notably high, with extremely dense fluctuations, which is distinct.

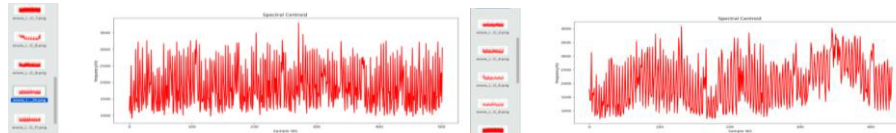


Fig. 21. Spectral Centroid variation of “Snore in sleep” audios

Spectral Centroid of "Yearn for delicacy" behavior audios in Fig.22 is generally based on persistent low-frequency segments. However, it is occasionally punctuated by moderate surges. When cats are comfortable and contented, they tend to make a low "purring" vibration with their throat vocal cords, and a persistent low-frequency background should be a quantitative expression of "purring". The medium level surges may relate to the cat's excitement and joy in "enjoying food". Since most of the time, the cat distracts part of its attention and energy to feed, so the surge is mostly at medium level. Occasionally, high frequency surges occurred between feedings, which may be a further expression of the excitement and joyful emotion under "delicacy".

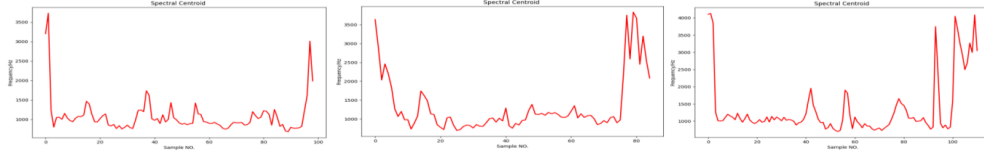


Fig. 22. Spectral Centroid variation of “Yearn for delicacy” audios

4.4 MFCC Matrix Results

MFCC results from Fig.23 to Fig.31 are presented as a graphical form of Meier frequency cepstrum coefficient matrix, with 13 rows on the vertical axis, representing 13 special frequency bands that are closely related to the pitch and timbre of sound. The horizontal axis represents the sequent frames, the vertical columns divided by the frames stand for eigenvectors which is corresponding to different frequency bands of audio signal, and the colors reflect the distribution of coefficient energies.

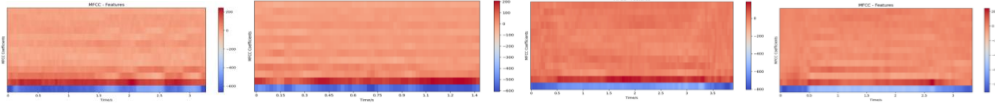


Fig. 23. MFCC of “Catch and hunt” audios

Fig. 24. MFCC of “Friendly Interaction” audios

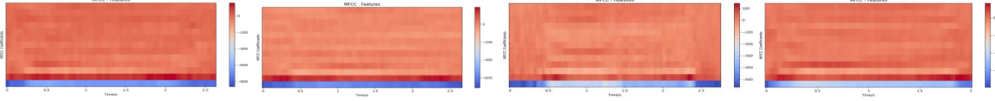


Fig. 25. MFCC of “Howl for bed” audios

Fig. 26. feature of “Howl for home” audios

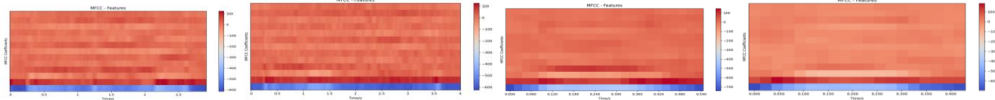


Fig. 27. MFCC of “Hungry for food” audios

Fig. 28. MFCC of “Quiet expectation” audios

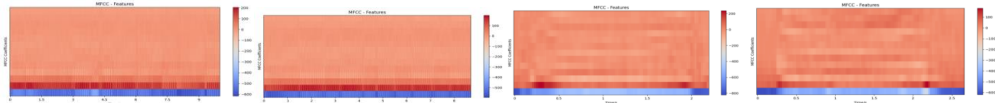


Fig. 29. MFCC of “Snore in sleep” audios

Fig. 30. MFCC of “Stuck and struggling” audios

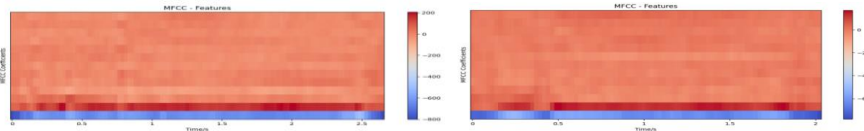


Fig. 31. MFCC of “Yearn for delicacy” audios

From the results, it can be seen that different types of audios in MFCC matrix, are somewhat differentiated. The energy distribution coefficients calculated in the matrix are more used as feature vectors or matrices that are substituted into the deep learning model for constructing and training. In neural network classification models, MFCC is usually used in combination with other features such as frame-level energy, time and frequency cepstrums to improve recognition accuracy.

4.5 Representative features of STFT (Short-time Fourier Transform)

STFT time-frequency atlas is mainly used as neural networks feature vector training set. The results are selected to display representative features, where the horizontal axis displays the sequent frames, the vertical axis shows the frequency bands, and the color represents the intensity or energy.

The distinction of the 3 demand audios, as seen in the STFT time-frequency mapping from Fig.32 to Fig.34, is clearly demonstrated. "Howl for home" has a wider range of high-frequency bands, while "Hungry for food" has more intensity and energy in the higher frequency bands. The change to higher frequencies is characterized by a gradual break in "howl for bed" and a stepped disappearance of the break in "Hungry for food".

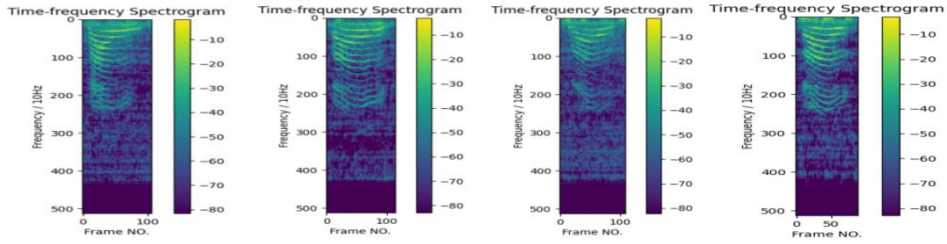


Fig. 32. STFT figures of “Howl for bed” audios

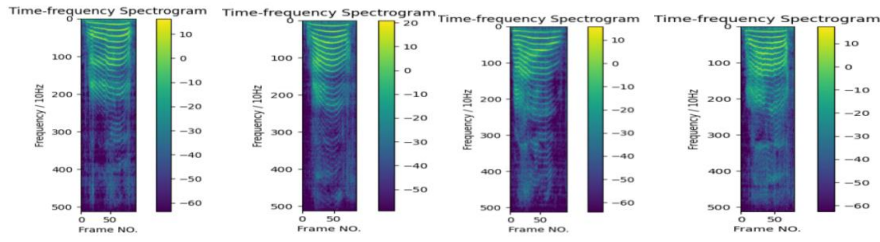


Fig. 33. STFT figures of “Howl for home” audios

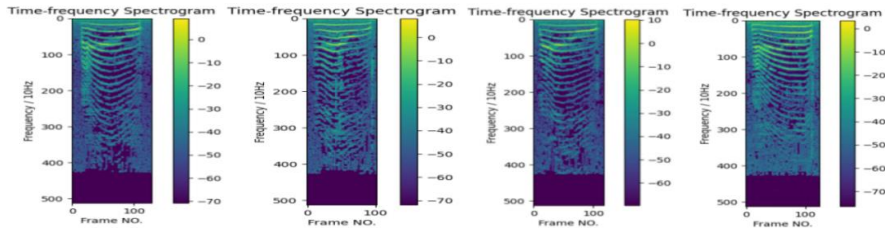


Fig. 34. STFT figures of “Hungry for food” audios

“Yearn for delicacy” behavior audios in Fig.35 and Fig.36 have a wide, relatively even distribution in the high frequency level, consistent with a sustained emotional energy state of high pitch and joy.

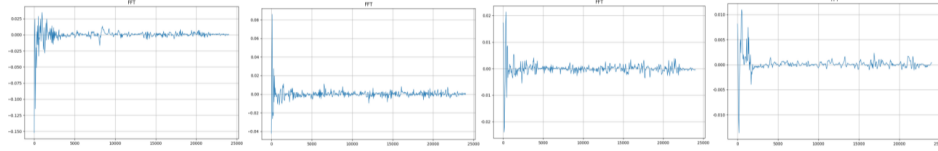


Fig. 35. FFT figures of “Yearn for delicacy” audios

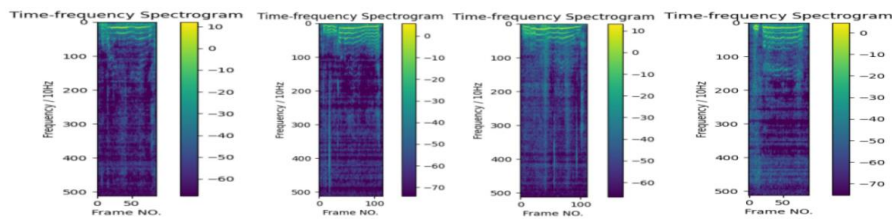


Fig. 36. STFT figures of “Yearn for delicacy” audios

"Snore in sleep" shown in Fig.37 is well aligned, with almost all the effective energy distribution concentrated in low frequency instead of the high ones, which corresponds to the complementary nature of the color intensities in the first and second dimensions of MFCC matrix, where the energy is mainly determined by low frequencies, corresponding to a conservative and stable emotional state.

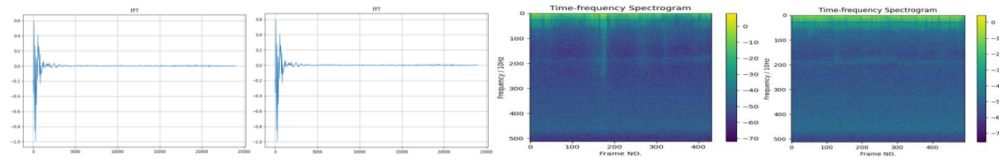


Fig. 37. FFT and STFT figures of “Snore in sleep” audios

"Catch and hunt" and "Stuck and struggling" are similar on FFT transform according to Fig.38 and Fig.39. In the high-frequency range, except for minor fluctuations, they are nearly straight. Obvious differences are shown in STFT time-frequency spectrograms in Fig.40 and Fig.41. They both decrease abruptly from low to high frequencies, but "catch and hunt" tends to decrease mildly like a "flat and straight brush", however, "Stuck and struggling" is a "rippling" decrease with a slow swoosh.

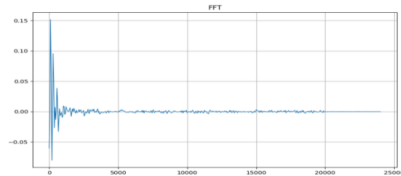


Fig. 38. FFT of “Catch and hunt” audios

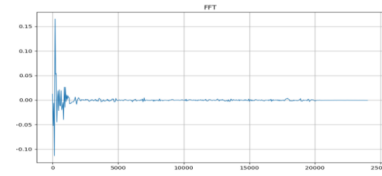


Fig. 39. FFT of “Stuck and struggling” audios

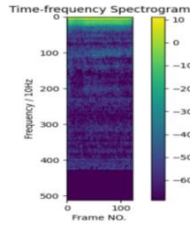


Fig. 40. STFT of “Catch and hunt” audios

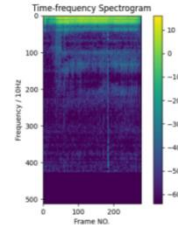
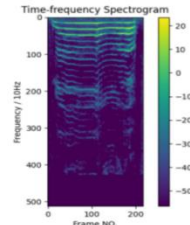
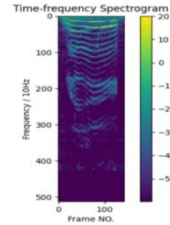


Fig. 41. STFT of “Stuck and struggling” audios



5 Conclusion

The research focuses on textless audio emotional analysis based on the behaviors of kitten. Researcher extracted effective audio features of different dimensions including Spectral Centroid, STFT, amplitude, and ZCR, and explored their potential relations with emotions concerning joy, anger, ferocity, excitement, balanced contentment, anxiety, pleading, stress, irritation, and anticipation, which provides references for weighting parameter setting, combination of features for emotion recognition in neural networks and deep learning of classification structures.

However, due to limited experimental conditions, there still exists space for improvement, such as expanding the sample size, for better recognition tolerance to more complex signals; or exploring more detailed and differentiated features, to distinguish samples with high approximation; or becoming more personalized for universality, to suit the application for different individuals and oral habits. It is hoped that in the future, the technology can be developed to be more refined and mature to assist information interactions without text, behavior or other expression support.

References

- [1] R. Ahuja, V. Solanki, V. Khullar and L. Kumar, "Classification of Non-Speech Sound Signals: An Approach of Machine Learning with MFCC Feature Extraction," 2024 International Conference on Electrical Electronics and Computing Technologies (ICEECT), Greater Noida, India, 2024, pp. 1-5, doi: 10.1109/ICEECT61758.2024.10738971.
- [2] Santoso, T. A. Sardjono and D. Purwanto, "Optimizing Mel-Frequency Cepstral Coefficients for Improved Robot Speech Command Recognition Accuracy," 2024 International Seminar on Application for Technology of Information and Communication (iSemantic), Semarang, Indonesia, 2024, pp. 284-289, doi: 10.1109/iSemantic63362.2024.10762627.
- [3] S. Singhal et al., "Audio Based Machine Fault Diagnosis using Hybrid Feature Extraction and Ensemble Learning," 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kamand, India, 2024, pp. 1-7, doi: 10.1109/ICCCNT61001.2024.10724147.
- [4] G. Fu, Y. Jiang, H. Li, L. Ling and W. Wei, "Research on Loose Wedge Detection Method of Generator Slot Based on Acoustic Feature," 2024 IEEE 14th International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), Copenhagen, Denmark, 2024, pp. 517-522, doi: 10.1109/CYBER63482.2024.10749636.

- [5] D. Prabakaran and S. Sriuppili, "Speech processing: MFCC based feature extraction techniques - An investigation", *Journal of Physics: Conference Series*, 2021.
- [6] F. T. Al-Dhief, N. M. Abdul Latiff, N. N. N. A. Malik, M. M. Baki, N. A. Muhammad and M. A. Abbood Albadr, "Investigating Fast Learning Network for Voice Pathology Detection," 2024 IEEE 7th International Symposium on Telecommunication Technologies (ISTT), Langkawi Island, Malaysia, 2024, pp. 108-113, doi: 10.1109/ISTT63363.2024.10750772.
- [7] N. Steinmetz and N. Balal, "Remote Speech Decryption Using Millimeter-Wave Micro-Doppler Radar," 2024 IEEE International Conference on Microwaves, Communications, Antennas, Biomedical Engineering and Electronic Systems (COMCAS), Tel Aviv, Israel, 2024, pp. 1-5, doi: 10.1109/COMCAS58210.2024.10741985.
- [8] Y. Wang and D. Lai, "A small sample conventional circuit breaker fault diagnosis method based on SWT-STFT and double flow CNN-SVM," 2024 IEEE Transportation Electrification Conference and Expo, Asia-Pacific (ITEC Asia-Pacific), Xi'an, China, 2024, pp. 126-131, doi: 10.1109/ITECAsia-Pacific63159.2024.10738700.
- [9] Sun Xinwei, Ji Aimin, Du Zhantao et al., "Diagnosis method for variable speed fault of rolling bearings in high-speed train gearbox [J]", *Journal of Harbin Institute of Technology*, vol. 55, no. 01, pp. 106-115, 2023.

Research on sales forecasting of online products based on machine learning methods--Taking Meituan photo studio as an example

Yiran You

School of Management Science and Engineering, Tianjin University of Finance and Economics, Tianjin, China

YiranYou@stu.tjufe.edu.cn

Abstract. With the rapid development of the digital economy, the way consumers purchase goods and enjoy services has gradually shifted to various online platforms. As a leading comprehensive platform, Meituan has accumulated massive market data in the field of photo services. In this paper, based on the Colab environment, eXtreme Gradient Boosting (XGBoost) and Random forest (RF) in the field of machine learning are used to introduce SHapley Additive exPlanations (SHAP) analytics to model and analyze its sales data and predict future sales. The model was optimized using grid search and stochastic search, combined with several metrics such as R-Square (R^2), Median Absolute Error (MdAE), and Log Mean Absolute Percentage Error (Log MAPE) for a comprehensive assessment of the model effectiveness. The results show that RF outperforms XGBoost in both initial and optimized models. In particular, with the introduction of interaction features, RF can effectively capture complex nonlinear relationships and significantly improve the accuracy of sales volume prediction, while XGBoost performs poorly in the face of data imbalance and extreme values, with large prediction errors. This study provides an important reference for merchants to optimize their marketing strategies and improve user experience, which has important theoretical value and practical significance.

Keywords: Online platform, sales prediction, random forest regression, SHAP analysis, interaction features

1 Introduction

With the rapid development of the digital economy, online platforms have become the main way for consumers to purchase products and services. As one of the leading life service platforms in China, Meituan's online sales data contains a wealth of market information [1]. For example, in the field of photographic services, users can select merchants, compare service prices, and conduct transactions through the platform. How to effectively predict product sales based on these data, and then help merchants optimize their marketing strategies, is an important current research topic.

In recent years, the application of machine learning algorithms in data analysis and prediction has received much attention. Identifying key factors and accurately predicting them by analyzing historical data can support companies in formulating efficient strategies and improving the efficiency of resource allocation [2].

Early studies mainly used traditional statistical methods such as regression analysis and time series analysis. However, it has limitations in high-dimensional and massive data scenarios. In recent years, the application of machine learning in sales volume prediction has gradually increased. Researchers have realized more accurate prediction of product sales volume through methods such as support vector machines, random forests (RFs), and decision trees [3, 4]. Especially on high-traffic e-commerce platforms, sales prediction models not only need to consider each influencing factor but also take into account the complexity of the market and the diversity of consumers [5]. In addition, deep learning methods are gradually being applied to the field of sales forecasting. Scholar Chen combines particle swarm optimization and Long Short-Term Memory Network (LSTM) to propose a merchandise sales prediction model, which makes the difference between the prediction results and the actual sales fluctuate in a small range (-2.4% to 1.82%), which is significantly better than the error range of the standard LSTM model, indicating that deep learning prediction accuracy and stability in big data scenarios are better than traditional methods [6]. At the same time, while machine learning models often provide highly accurate predictions, their black-box nature makes the results difficult for the general public to intuitively understand. Thus, interpretable machine learning has become an important research area in recent years [7]. For this reason, the SHapley Additive exPlanations (SHAP) additive interpretation method was proposed by Mangalathu to quantify the feature importance and reveal the model decision mechanism, effectively quantify the contribution of features to the prediction results, and provide a new way of thinking about model interpretation and practical application [8].

In this study, based on the sales data of Meituan Photo Studio, a series of data processing work was carried out to introduce interaction features, feature importance, SHAP analysis, apply different machine learning algorithms to construct a product sales prediction model, reveal the main factors affecting sales for prediction, and optimize the model using grid search and random search. Ultimately, the model performance is evaluated through multi-dimensional indicators. Provide effective marketing strategies and more comprehensive and accurate guidance for various online platforms.

2 Methodology

2.1 Data source and description

The data for this study comes from the Meituan platform, focusing on its photo studio online product information. Focusing primarily on merchants in the Beijing area, it contains 4,298 pieces of data and 18 features across the following fields: dianpu name, such as Korean Bride STUDIO Travel Wedding Photography (Beijing Store), dianpu star, ranges roughly from 2.0 to 5.0. Type, like wedding photography, etc. Price-related fields, including dianpu price avg, product prices online, etc. Shooting related fields, such as pic num, and jingxiu num. Shooting related fields, such as pic num, jingxiu num. Additional services, such as negative gift situation. The study selects numerical features that have some correlation with sales and that directly reflect product pricing, service quality, and shooting results. The features are selected to accomplish the task of sales forecasting to ensure the representativeness of the data and the usefulness of the analysis results.

Data processing includes the steps of data Cleaning: removing features that are not relevant to sales volume prediction to reduce noise and improve model efficiency. Missing values were

filled in using the mean, median, or plurality to ensure data completeness. Text processing: clean up product names with regular expressions, retaining key numerical information for analysis. Data consolidation: Integrate multiple data sources and harmonize data formats and feature names to ensure consistency and accuracy. The final dataset covers multidimensional features such as product price, store rating, online price, and number of shots, and the distribution of key features is explored through descriptive statistical analysis to lay the foundation for subsequent research.

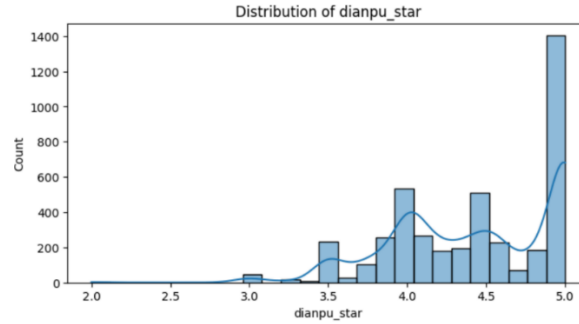


Fig. 1. The Distribution of dianpu star (Picture credit: Original).

As shown in Figure 1, the horizontal coordinate shows the store's star rating, which ranges from a minimum of about 2.0 to a maximum of 5.0. The vertical coordinate indicates the number of stores in each star range. The distribution of store ratings is concentrated in the higher score bands, especially above 4.0, with a significant proportion of 5.0 ratings. This suggests that the majority of stores in the dataset have higher ratings, possibly reflecting users' tendency to choose highly rated stores or stores with better service quality overall. However, an unbalanced distribution of ratings may cause the model to be more skewed toward highly rated samples. To this end, this study enhances the scoring discrimination through feature interactions, such as using the product of dianpu star and product prices as a new feature star price interaction to optimize the prediction performance.

2.2 Methodology introduction

In feature selection, three scenarios were designed in this study to train the model and validate the conclusions: Case 1 (no interaction 1): contains dianpu star, dianpu price avg, product sales, product prices, product prices online, pic num, and jingxiu num. Case II (Interaction): based on case I the interaction feature star price interaction is introduced. Case 3 (no interaction 2): the top three features obtained by feature importance ranking, i.e., product prices online, product prices and dianpu price avg, are chosen.

The main models used in this study include: (1) eXtreme Gradient Boosting (XGBoost) regression model:

XGBoost is a decision tree model based on gradient boosting, which aims to gradually improve the prediction accuracy by constructing a series of weak learners (decision trees) [9]. Its optimization goal is to minimize the objective function with the following formula:

$$L = \sum_{i=1}^n l(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) \quad (1)$$

Where L is the loss function (e.g., mean square error (MSE)), $l = (y_i, \widehat{y_i})$ is the loss function, and $\Omega(f_k)$ is the regularization term to control model complexity and prevent overfitting.

(2) RF regression model:

An RF consists of multiple decision trees, each trained based on independent randomly sampled data, and the final prediction is the average of the predictions of all the trees, denoted as:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^T h_t(x) \quad (2)$$

Where T is the number of trees, $h_t(x)$ denotes the predicted value of the t th tree, and \hat{y} is the final predicted value. RFs are used for node splitting by randomly selecting features to effectively cope with noise improve robustness, and help identify key features through feature importance analysis.

2.3 Degree of influence of features and parameter optimization

To understand the impact of features on sales volume prediction, this study calculates the correlation matrix of the features and plots a heat map to reveal the linear relationship between the features. Highly correlated features may cause redundancy and affect model performance. The contribution of each feature to the model's prediction results was quantified through SHAP analysis with the following formula:

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N|-|S|-1)!}{|N|!} (\nu(S \cup \{i\}) - \nu(S)) \quad (3)$$

Where N is the set of all features, $\nu(S)$ is the output of the model on the feature set S , and ϕ_i denotes the SHAP value of feature i . This approach helps to identify the most critical features in sales volume forecasting. For parameter optimization of the model, this study uses a combination of grid search and cross-validation. Grid search selects the optimal parameters by combining them under multiple parameter combinations, while cross-validation further ensures the generalization of the model and reduces the risk of overfitting. The best parameters were finally selected for model training by cross-validation with 3 and 5 folds.

2.4 Evaluation indicators and synthesis analysis

In the evaluation of the model, R-Square (R^2) was first chosen to measure the ability of the model to explain the fluctuations in the target variable, ranging from 0 to 1. The closer the value is to 1, the better the model fit. However, when the data are unevenly distributed or have extreme values, R^2 may not be sufficient to accurately reflect the performance of the model, and therefore a comprehensive assessment in combination with other metrics is required. Secondly Median Absolute Error (MdAE) is insensitive to extreme values and provides a more robust reflection of the typical error magnitude of the model and is suitable for data where extreme values exist. Once again Log Mean Absolute Percentage Error (Log MAPE) is more suitable for measuring relative prediction error than the traditional Mean Absolute Percentage Error (MAPE) as the logarithmic treatment reduces the asymmetric effect of large error values. Through the comprehensive analysis of the above indicators, we comprehensively assess the model's fitting ability and resistance to extreme values, and deeply analyze the characteristics of the prediction error distribution to verify the reliability and robustness of the model's performance.

3 Statistical analysis

3.1 Relevance of each feature

In machine learning, the level of correlation between features and target variables affects the performance of the model. The nonlinear models used in this study, RF and XGBoost, are capable of capturing complex nonlinear relationships. Thus even if the correlation between individual features and the target variable is low, the model can still improve the prediction performance through feature interaction. Low-correlation features can also sometimes reduce redundancy, avoid multicollinearity problems, and enhance the generalization ability of the model.

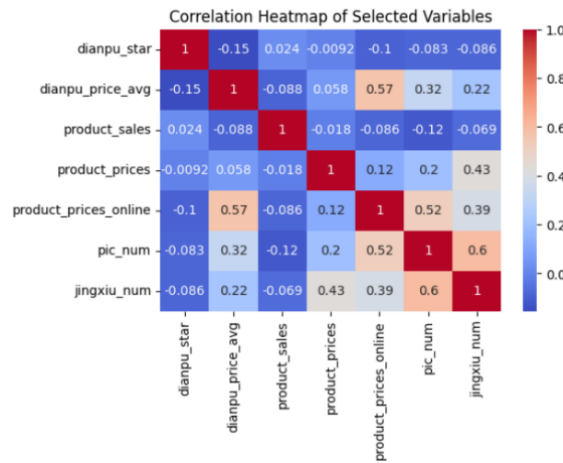


Fig. 2. Feature Correlation Heatmap (Picture credit: Original).

As in Figure 2, further explanations of the variables are derived: The target variable product sales has a low correlation with the other variables, indicating that these variables have a low impact on sales. Interaction features were added to the study to better capture sales changes. The correlation between pic num and product prices online is 0.52, showing a moderate positive correlation between the number of pictures and online prices, i.e. more pictures may lead to slightly higher prices. The correlation between jingxiu num and pic num is 0.6, indicating that the number of shots is significantly and positively correlated with the number of refined images.

3.2 Characteristic importance

As in Table 1, the importance of each feature of the RF model is extracted and ranked, and the top three to four features with the highest importance are shown.

Table 1. Feature Importance of the RF Model (Table credit: Original).

	Feature	RF Feature Importances
1	Product prices online	0.479490
2	Product prices	0.221858
3	Dianpu price avg	0.185461
4	Pic num	0.068901

Based on the results presented in Table 1, the following ranking of the importance of the characteristics can be observed. product prices online: has the highest significance (0.479490) and is more than twice as high as the other higher variables, suggesting that online prices are a key factor in sales volume, directly influencing consumer purchasing decisions. Online prices usually have a direct impact on consumers' purchasing decisions. Product prices also of significant importance, influencing consumer choice and distribution channels. However, product prices may cover a wider range of sales channels than online prices. pic num indicates that the content of the service appeals to the consumer and may increase willingness to buy. The small effect of dianpu price avg indicates that the average store price has a limited impact on sales. Based on these analyses, it can be hypothesized that price sensitivity is the main driver and marketing strategies should focus on pricing. On service marketing, increasing the number of shots may improve the conversion rate.

3.3 SHAP analysis

Firstly, it is explained that it is normal for the results of the feature importance analysis of XGBoost and RF to be different from the results of the SHAP analysis due to the different computational approaches and focuses. RF feature importance is statistically derived from the tree structure, measured based on the information gain (e.g., Gini coefficient or MSE) when splitting nodes. The SHAP analysis, however, is based on the Shapley value in game theory, which quantifies the marginal contribution of features to the model output, overcomes the problem of multiple covariance among features, and provides a more objective assessment.

As can be seen in Figure 3, each point in the graph represents the SHAP value of a sample, with red points indicating high values and blue points indicating low values. SHAP values for product prices and product prices online stand out in the negative direction, indicating that high values of these features usually result in lower model output, e.g., fewer sales. A high value of dianpu star tends to increase sales and supports the findings of the feature importance analysis.

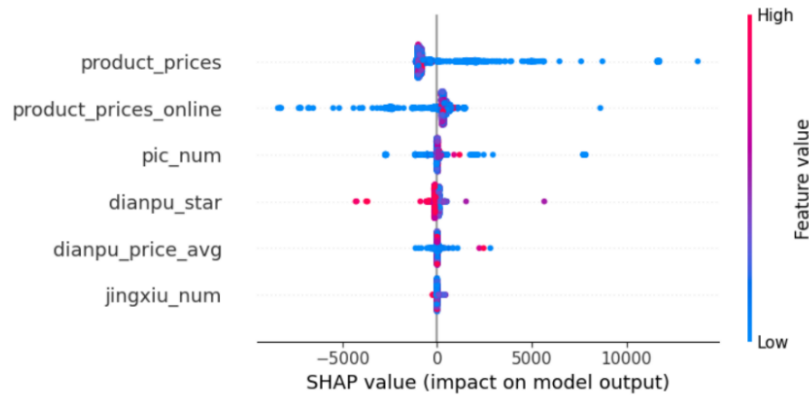


Fig. 3. SHAP Value Scatter Plot (Picture credit: Original).

3.4 Model optimization and evaluation

As shown in Figures 4, 5, the performance comparison of XGBoost and RF in the sales prediction task before and after optimization in three cases contains three sub-figures, the horizontal coordinates are the names of the different models, which are Initial XGBoost,

Optimized XGBoost, Initial RF and Optimized RF. The left subplot has a vertical coordinate of R^2 , unitless, and ranges from 0 to 1. The center subplot has a vertical coordinate of MdAE, and the unit is the number of products sold. The vertical coordinate of the right subplot is Log MAPE in percent (%). It is used to compare and evaluate the prediction effect before and after model optimization.

Case II, interaction, hyperparameter tuning for random search, is visualized in Fig. 4. Instead, the optimized R^2 decreases, and MdAE and Log MAPE deteriorate significantly, showing that the optimization does not lead to a positive effect. Suggests that this scenario may be overfitting or underfitting.

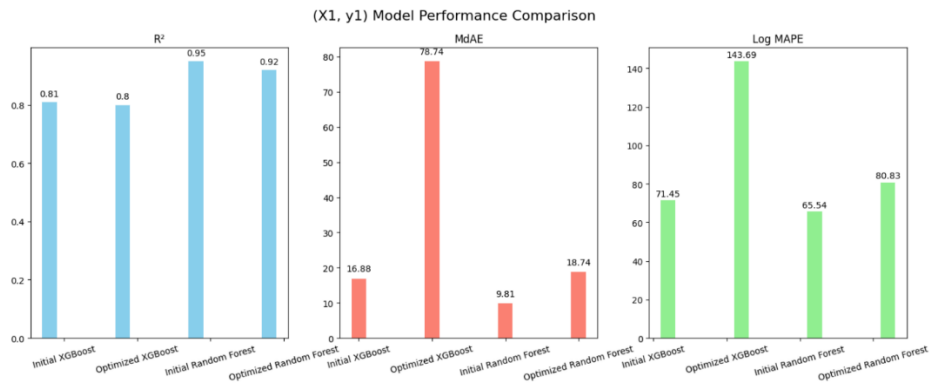


Fig. 4. Bar Chart Comparing Model Performance Before and After Optimization in Interactive Contexts (Picture credit: Original).

Case 3, no interaction2, is optimized by grid search, as in Figure 5. The resulting R^2 remains stable or slightly elevated, indicating a more robust fit of the model to the overall trend. However, both MdAE and Log MAPE increased, indicating a decrease in prediction accuracy for individual values, which may affect the model's ability to predict detail.

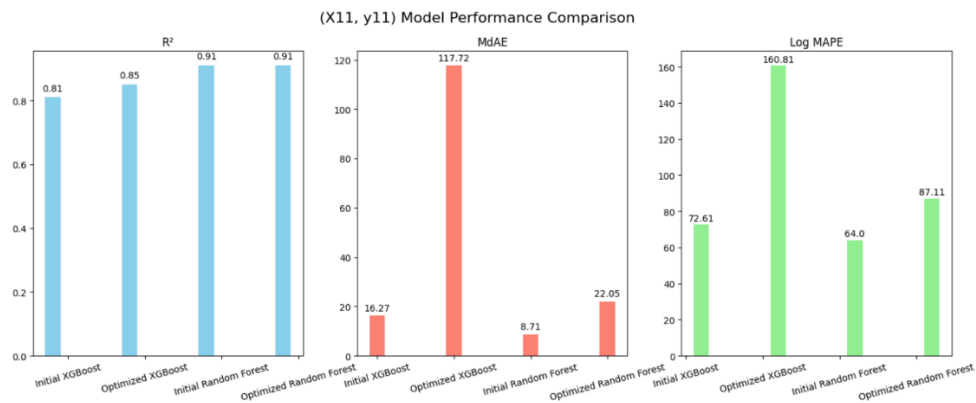


Fig. 5. Bar Chart Comparing Model Performance Before and After Optimization in Non-Interactive Scenario 2 (Picture credit: Original).

It is found that RF outperforms XGBoost's R^2 in all cases, especially in the interaction context where it reaches 0.95, indicating that the model is able to fit the sales data to a great extent, higher than XGBoost's 0.81. Its MdAE and Log MAPE metrics also performed well, especially in the no-interaction2 and interaction contexts, with a MdAE of 8.71 and high predictive accuracy. Under the interaction feature, the Log MAPE of RF reaches 65.54%, which is still lower than Initial XGBoost's 71.45%, further verifying the advantage of RF in sales prediction.

3.5 Visualization of the final result

To show the prediction results of the models more intuitively, in this study, a comparative visualization of sales prediction is carried out, especially for the Initial RF model under the combination of interactive features and the Optimized XGBoost model under the combination of no-interactive features1 to present the difference between the actual sales and the predicted sales. In Fig. 6, the X-axis represents the randomized serial number of the data points and the Y-axis represents the product sales. The red lines/dots, indicate True Values, i.e. actual sales. The blue lines/dots, indicate Initial RF's predicted values for sales. Orange lines/dots indicate the predicted value of sales by the Optimized XGBoost model. Fluctuations in actual sales can have significant peaks and troughs, reflecting the seasonality of sales or the impact of unexpected events. The predicted values of the two models may exhibit deviations from actual sales at some data points, showing the predictive power of the models under specific conditions. The predicted values from Initial RF may be smoother overall, showing the strength of the model in capturing trends in sales volume. The overall data is visualized in Figure 6.

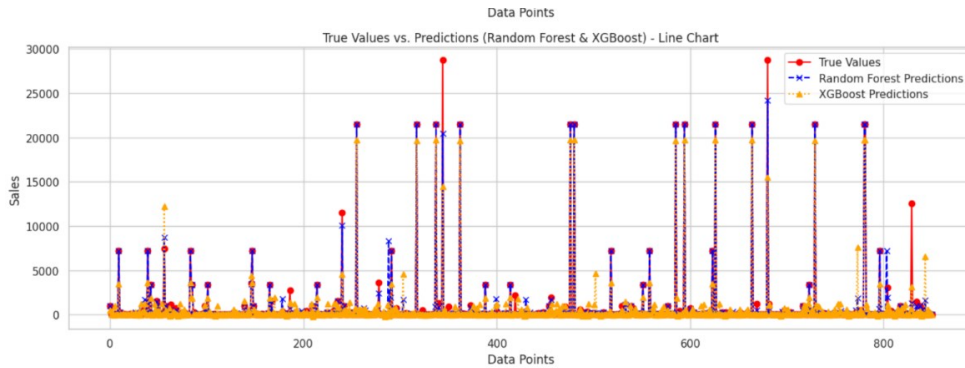


Fig. 6. Comparison Chart of Predictions from Two Models (Overall Data) (Picture credit: Original).

Due to the huge amount of data, to avoid the chart is too complex and unclear, the strategy of dividing the data sequentially into 12 segments and drawing charts separately was adopted for group visualization. Further, a randomized grouping strategy was used to ensure the representativeness of the data and to avoid the influence of order effects. This method ensures that each segment of data contains samples from different categories and locations to fully reflect model performance. For example, sequential segmentation may be subject to bias due to seasonal or cyclical fluctuations, while randomized grouping helps to reduce this effect. This is illustrated in Figure 7.

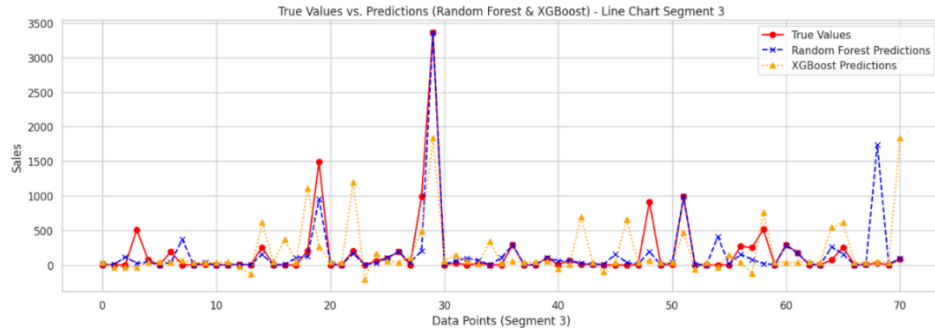


Fig. 7. Comparison Chart of Predictions from Two Models (Randomly Divided into 12 Groups) (Picture credit: Original).

The predictive effects of the model are presented in a more representative way through the presentation of segmented graphs after randomized grouping. The line graph visualization of an arbitrarily selected section of data shows that the Initial RF model is closer to the true value in terms of predicted values, showing better stability and lower error. In contrast, Optimized XGBoost's predictions are relatively more volatile, but it remains competitive in terms of explanatory power, R^2 , after model tuning.

4 Discussion

This study provides a comprehensive assessment of the performance of multiple machine learning models in sales volume forecasting and discusses the strengths and weaknesses of the models and their practical application value in conjunction with feature importance analysis. Through SHAP analysis, the study demonstrates the consistency and fairness of feature importance interpretation in high-dimensional data scenarios, which complements the traditional feature importance analysis approach of RFs and improves the understanding of the model decision-making process. This is supported in the literature, with Ana and Lundberg (2024) noting that SHAP has significant advantages in terms of model interpretability and transparency [10]. The model optimization results show that the optimized RF excels in several performance indicators, especially in the case of interacting features, its R^2 value reaches 0.95, and the prediction accuracy is significantly improved. Meanwhile, the model outperforms other models in MdAE and Log MAPE metrics, demonstrating strong adaptability and robustness to complex data relationships. The optimized XGBoost, on the other hand, although improved in R^2 , is weaker in handling unbalanced data and extreme values, resulting in higher MdAE and Log MAPE. The study also found that XGBoost's predictions fluctuated significantly, especially in the prediction of extreme cases, through random group visualization analysis. To address this issue, the study recommended a combination of training and testing using Augmented Data Augmented Data (AD) techniques to cope with the lack of sufficient data to achieve the desired accuracy in many real AI data processing cases [11].

In order to enhance the practical application of the sales forecasting model, merchants can combine the results of the study to develop more accurate marketing strategies. For example, in response to the significant impact of the number of shots on sales revealed in this study, merchants should invest in high-quality product images and video production, and can use short

video ads for store promotion. Related studies have shown that credibility, expertise, and attractiveness of video advertisements are positively correlated with consumer purchases, while authenticity and brand heritage influence consumer purchase behavior in a U-shaped manner [12]. In addition, merchants can regularly analyze the impact of their pricing strategies on sales and adjust their strategies in light of sales forecasts, especially during holidays and promotional seasons, by targeting potential best-selling products and implementing targeted promotional strategies. At the same time, to cope with the rapid changes in the market, merchants are advised to utilize big data for strategic marketing and dynamic capabilities needed to improve market responsiveness [13]. By dynamically monitoring market trends, merchants can more flexibly adjust their product lines and pricing strategies in response to changes in consumer demand and challenges in the competitive environment.

In summary, this study not only reveals the application potential of machine learning models in sales forecasting but also provides practice-oriented improvement suggestions for merchants, providing theoretical basis and technical support for optimizing marketing decisions and improving sales performance.

5 Conclusion

The results of this study show that the RF model has stronger predictive power and stability in sales volume forecasting, especially when faced with complex features and interactions. It outperforms XGBoost in error control and prediction accuracy and significantly outperforms XGBoost in optimized performance. In practice, merchants can utilize RF's sales prediction results to develop more accurate pricing and promotional strategies, especially during holiday or promotional seasons. By adding high-quality images to display, adding short videos to promote your store and optimizing your pricing strategy, you can effectively boost sales. In addition, merchants should leverage the dynamic capabilities of big data, combining market feedback and external data to continuously optimize their marketing strategies to adapt to market changes and enhance competitiveness.

References

- [1] Zhao, W., & Zhao, Q. (2022). Profitability analysis of Meituan. *Old Brand Marketing*, (02), 169-171.
- [2] Zhao, Q., Bai, L., Xu, W., et al. (2019). Monthly sales prediction of unmanned supermarket under multiple linear regression model. *Times Economy and Trade*, (13), 35-37.
- [3] Martins, E., & Galeale, N. V. (2023). Sales forecasting using machine learning algorithms. *Revista de Gestão e Secretariado (Management and Administrative Professional Review)*.
- [4] Pavlyshenko, B. (2019). Machine learning models for sales time series forecasting. *Data*, 4(15).
- [5] Rajasree, T., & Ramyadevi, R. (2024). Time series forecasting of sales data using hybrid analysis. In *2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT)* (pp. 732-735).
- [6] Chen, I., & Zhang, S. (2023). Research on merchandise sales prediction based on deep learning. *Information and Computer*, 35(12), 111-113.
- [7] Yang, J., & Cao, J. (2021). Tree-based interpretable machine learning of the thermodynamic phases. *Physics Letters A*, 412, 127589.

- [8] Mangalathu, S., Hwang, S.-H., & Jeon, J.-S. (2020). Failure mode and effects analysis of RC members based on machine-learning-based Shapley Additive exPlanations (SHAP) approach. *Engineering Structures*, 219, 110927.
- [9] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)* (pp. 785–794). Association for Computing Machinery.
- [10] Ponce-Bobadilla, A. V., Schmitt, V., Maier, C. S., Mensing, S., & Stodtmann, S. (2024). Practical guide to SHAP analysis: Explaining supervised machine learning model predictions in drug development. *Clinical and Translational Science*, 17(11), e70056.
- [11] (2021). Augmented data and XGBoost improvement for sales forecasting in the large-scale retail sector. *Applied Sciences*, 11(17), 7793.
- [12] Meng, L. (Monroe), Kou, S., Duan, S., & Bie, Y. (2024). The impact of content characteristics of short-form video ads on consumer purchase behavior: Evidence from TikTok. *Journal of Business Research*, 183, 114874.
- [13] Brewis, C., Dibb, S., & Meadows, M. (2023). Leveraging big data for strategic marketing: A dynamic capabilities model for incumbent firms. *Technological Forecasting and Social Change*, 190, 122402.

Research on vehicle multi-classification object detection algorithm based on Ultralytics/YOLOv5 improvement

Yu Deng

Dalian Maritime University, 1 Linghai Road, Dalian, China

13508332846@163.com

Abstract. This paper investigates an improved vehicle multi-classification object detection algorithm based on Ultralytics/YOLOv5. By combining the YOLOv5 model with a custom dataset, training and testing were conducted on Bangladeshi road vehicle images. The main contributions of this paper include optimizing the hyperparameter settings of the model to adapt to the vehicle multi-classification problem and improving detection performance by adjusting input sizes and the number of categories. The training set contains 2704 images, the validation set contains 300 images, covering 21 classes of vehicles. Experimental results show that the improved algorithm performs well in terms of mAP, Precision, and Recall values, with an average mAP of 0.41 and Precision of 0.669 on the test set. The research results of this paper demonstrate that the improved YOLOv5 model exhibits high accuracy and robustness in vehicle detection tasks in complex traffic environments, providing strong support for intelligent traffic monitoring and autonomous driving.

Keywords: Vehicle Multi-Classification, Object Detection, Ultralytics/YOLOv5, Deep Learning, Real-Time Detection.

1 Introduction

Region-based object detection is an important issue in the field of computer vision, playing a crucial role in many applications such as intelligent surveillance, autonomous driving, medical image analysis, etc. In recent years, with the development of deep learning technology, object detection algorithms based on deep neural networks have made significant progress, among which region-based detection techniques are the most representative. Region-based object detection algorithms use region proposal generators (such as RPN - Region Proposal Network) to generate a series of candidate regions that may contain objects, and then classify and locate these candidate regions. This approach can significantly improve the accuracy and robustness of object detection. In recent years, deep learning-based object detection algorithms have evolved into two main technical routes: Anchor-based methods (two-stage, one-stage) and Anchor-free methods. The two-stage object detection algorithms in Anchor-Based mainly include RCNN, SPPNet, Fast RCNN, Faster RCNN, FPN, Cascade RCNN; one-stage object detection algorithms include YOLO v1, SSD, YOLO v2, RetinaNet, YOLO v3, YOLO v4, YOLO V5. Anchor-Free object detection algorithms include CornerNet, CenterNet, FSAF, FCOS, SAPD, and other mainstream algorithms.

In practical applications such as intelligent surveillance, autonomous driving, vehicle traffic detection, smart industry, agricultural automation, etc., accurate detection and positioning of targets are crucial for the stability and reliability of the system. Moreover, region-based object detection algorithms have strong adaptability and can handle object detection problems in complex scenes. For example, in urban traffic monitoring, vehicles may appear in different sizes, angles, and degrees of occlusion. Region-based object detection algorithms can effectively cope with these challenges and improve the detection performance of the system. In maritime ship target detection, region-based object detection can monitor port shipping traffic, obtain ship deployment and dynamic information, and has important research value for effectively understanding the deployment status of maritime ships [1].

This paper will be divided into six main parts. The second part of the article will review classical literature, discussing region-based object detection algorithms and previous research applications. The third part of the article will introduce the algorithm used in this paper, as well as elaborating on the overall experimental process and specific model details (such as hyperparameters) from two dimensions: macro and micro. The fourth part of the article will present the results of experiments through figures and tables. The fifth part of the article will conduct discussions, analyzing specific experimental results, as well as the similarities and differences between this study and similar studies. Finally, the sixth part of the article will summarize, discussing the advantages, limitations, and future research directions of this study.

2 Literature Review

Object detection, as one of the core issues in the field of computer vision, plays a crucial role in various areas such as intelligent surveillance, autonomous driving, and medical image analysis. With the rapid development of deep learning technology, object detection algorithms based on deep neural networks have made significant progress.

There are already many excellent object detection algorithms applied to the detection of natural scenes. For example, Zhao and Yang proposed a lightweight YOLOv5 algorithm based on multi-scale pyramid and multi-scale attention, targeting problems such as small target sizes and complex backgrounds in remote sensing vehicle detection [2]. By reducing the number of downsampling layers and redesigning the multi-scale pyramid network, the detection capability of small objects and feature fusion ability were improved; introducing an improved multi-scale attention module enhanced perception and reduced model parameters; using the K-means++ clustering algorithm to design anchor box scales and aspect ratios suitable for targets. Compared to YOLOv5s on a self-built dataset, the algorithm achieved higher detection accuracy while reducing parameters and model size.

Lv and Jia proposed an improved object detection model GP-YOLOv5n, based on YOLOv5n and incorporating fusion attention into the depth separable neck network [3]. The model adopted Alpha-IoU in the bounding box regression loss function, improving the accuracy of bounding box positioning for objects. For wildlife detection, a lightweight giant panda detection model was designed. Experimental results showed that in complex environments, the model significantly improved the detection accuracy and speed for giant pandas.

Li et al. proposed an ultra-lightweight object detection network for detecting ships in aerial images. The model size of this algorithm is only 1.64MB, achieving a speed of 25fps on a mobile platform with 0.75T computing power while maintaining high accuracy [4]. By employing an extremely lightweight CNN backbone network, fused detection head design, and advanced data preprocessing methods, the network's robustness and capability to capture detailed features of targets were enhanced, thereby improving detection accuracy. The adoption of the tiled region of interest detection strategy provided a wider range of choices for the network's practical applications. The fusion of these strategies achieved a balance between accuracy and speed, and the algorithm's advancedness and practicality were verified on embedded platforms.

Chen et al. proposed a mural damaged area detection method called U²-DUANet, using a nested U-shaped structure network [5]. The method introduced a Depth-supervised Aggregation (DUA) module to more effectively integrate detailed information from side outputs. Additionally, the Pixel-level Context and Channel Attention (PCCA) module were utilized to capture important features more accurately. Moreover, the use of Self-adaptive Normalization (SN) instead of traditional Batch Normalization (BN) enhanced the flexibility and generalization capability of the network. Finally, a novel attention module, PCCA, was proposed, which combined spatial and channel information of images to more accurately capture important features.

Yuan et al. proposed a drone object detection method based on region-adaptive thresholding [6]. Firstly, target suspected regions were identified using local threshold segmentation, then a clustering method was used to segment suspected regions into the YOLOv5 detection network to avoid loss of target features during global detection due to image compression. Finally, to address the problem of low confidence in small targets, an adaptive threshold based on target size was used to enhance the detection rate of small drones. This method improved the detection rate of small targets, avoided ROI repeated extraction through the region attribution division based on DBSCAN clustering, and effectively reduced the false alarm rate through the filtering method based on adaptive thresholding according to target size.

Chen et al. proposed a drug recommendation algorithm based on dialog structure and graph attention network to address the problem of existing algorithms' inability to reflect patients' real-time health needs [7]. Firstly, a correlation-aware structural graph was constructed by combining grey relational analysis with graph attention networks to better capture the intrinsic correlations between nodes. Then, each dialogue utterance was represented as a node of the graph attention network, and two types of relational structures were designed to learn the adjacency relationships between nodes. Next, dialogue hierarchical encoders and disease encoders were designed to learn dialogue structure representation and patient health problems. Finally, the two feature representations were fused into an MLP layer to achieve real-time prediction and recommendation of drugs. This method fully perceives the correlation of dialogue nodes, learns dialogue structure representation, and combines disease representation for drug prediction, improving the effectiveness of the recommendation algorithm.

Zhang and Yang proposed an improved facial expression recognition method based on Local Binary Pattern (LBP) and attention mechanism in the New Visual Geometry Group Network (NEW-VGG), aiming to improve training speed and recognition performance [8]. By integrating the LBP algorithm and the NEW-VGG model, the training speed of the model was accelerated. The VGG-16 network was improved to create the NEW-VGG model, and through

ablation experiments, the effectiveness of global average pooling layer and attention mechanism in improving the speed and accuracy of the model was verified.

Hu et al. proposed an ACE-YOLO adaptive local image detection algorithm based on deep learning for fast and high-precision detection of apple defects [9]. This algorithm reduces the detection area through deep learning, utilizes channel attention mechanism to concentrate computing resources on local detection range, employs image enhancement algorithm to improve detail clarity, and adds a small target detection layer to enhance detection accuracy. This method effectively solves the difficulty of identifying small defects in apple detection, amplifies defect details through local image processing techniques, and avoids environmental interference. By improving the network structure and introducing channel attention mechanism, a balance between detection accuracy and speed was achieved.

Wu et al. addressed the issue of feature blurring in small target detection in aerial image object detection by proposing an improved YOLO_v5x algorithm [10]. By adding the SPD module and small target detection head, the loss of fine-grained information was reduced, effectively improving the efficiency of small target detection; the introduction of the CA attention mechanism and a new loss function significantly improved the positioning accuracy of small targets. Experimental results based on the Visdrone dataset validated the effectiveness of this method.

Hua et al. proposed an improved target detection network for the inefficient detection of targets in vehicle-mounted infrared images [11]. Firstly, by adding a dynamic detection head based on attention mechanism, the network focused more on foreground targets, enhancing the expression ability of the detection head. Secondly, the MPDIOW was used to replace the CIOU bounding box loss function during training, improving the model's positioning accuracy and efficiency. Finally, the lightweight network FasterNet was added to the C3 module at the end of the neck network to further improve the real-time performance of the model.

Wang et al. proposed a controller's sleeping behavior recognition method based on a dual-stream adaptive graph convolutional network [12]. This method designs dual-stream networks to process first-order and second-order information of the controller's skeleton separately, achieving full extraction of skeleton data; by adaptively learning the skeleton's topological connectivity matrix, functional connection relationships between different joints of the controller are explored; meanwhile, a spatio-temporal channel attention mechanism is introduced into the convolutional layer to enhance the model's ability to extract important information in time, space, and channel directions for recognizing the controller's sleeping behavior.

Zou et al. proposed a spatio-temporal attention graph convolutional network model for dynamic traffic flow prediction, and based on the attention mechanism, established a spatio-temporal attention graph convolutional network model, considering the application of temporal and spatial correlations, dynamic spatial correlations, and external features [13]. The spatio-temporal attention graph convolutional network model for dynamic traffic flow prediction can effectively extract spatio-temporal features in the traffic network. Compared to other baseline models, the ATST-GCN model shows more stable prediction results for medium to long-term (40, 60 minutes) traffic flow.

Zhou et al. proposed a multi-branch joint network structure based on attention mechanism to improve the accuracy of pedestrian re-identification [14]. This method, based on the Full-scale Network (OSNet), introduced an attention mechanism module to refine the semantic representation of features. Meanwhile, a multi-branch joint network was employed to organically combine local and global features, enhancing the correlation between features. Batch feature erasure was utilized for data augmentation to reduce the influence of occlusion, and a multi-loss joint function was adopted to reinforce the model's supervised training. This approach embedded attention mechanism in the backbone network to obtain richer global features, while utilizing batch feature erasure and improved horizontal segmentation method in the branch network to increase the model's robustness and focus more on extracting local fine-grained features.

Xie and Zhang published a review article on graph convolutional neural networks [15]. The article categorized graph convolutional neural networks into two major categories: graph-based methods and spatial-based methods. It summarized the research progress of graph convolutional neural networks, analyzing the development of graph convolutional neural networks from graph-based convolution and spatial-based convolution respectively. Graph-based convolution, rooted in mathematical expression and oriented towards reducing computational complexity, laid the theoretical foundation of graph convolutional neural networks. Spatial-based convolution, based on aggregation functions and update functions, combined theories such as sampling, attention mechanisms, and pooling operations, forming diverse graph convolutional neural networks, which have been widely applied.

Chai et al. proposed an unsupervised document graph embedding learning and classification model GGCN-DDC [16]. GGCN-DDC combines the scalability of TextING text representation and the advantages of DAEGC deep graph clustering. By improving the convolutional layer and proposing a new reconstruction matrix loss function, this model can better extract text features and implicitly learn the relationship between unlinked documents, thereby achieving better classification and clustering effects when dealing with unsupervised document corpora.

Yang et al. proposed a preference-aware denoising graph convolutional network social recommendation model PD-GCN [17]. PD-GCN is a social recommendation model that utilizes a preference-aware denoising graph convolutional network. By unsupervised learning, users are assigned to interaction subgraphs and social subgraphs to alleviate the over-smoothing problem and improve the model's robustness to noise. Meanwhile, by identifying and removing noise nodes through denoising strategies, the complex interactions and social relationships between users and items are effectively modeled.

Tang et al. proposed a knowledge graph convolutional network recommendation model KGCN-SHCN based on structural holes and common neighbors [18]. KGCN-SHCN achieves learning resource recommendation by computing the sampled neighborhood of central entities and utilizing the message aggregation mechanism of the KGCN model, enriching learning resources with auxiliary information from the knowledge graph, stimulating learners' interests, and improving the efficiency of recommendation by improving the sampling method and combining the message aggregation idea of graph convolutional networks.

Wang et al. proposed a low-light target detection algorithm based on image adaptive enhancement [19]. By designing an adaptive enhancement network and jointly optimizing it with the YOLOv5 target detection network end-to-end, the enhancement effect is more

conducive to the target detection task. Meanwhile, by combining channel attention and pixel attention, a feature enhancement module is designed to improve target detection accuracy.

Luo et al. proposed a commodity news event extraction model SAT-GCN-DPT based on self-attention mechanism and average pooling graph convolutional network, aiming to solve the problems of weak correlation between trigger words and entity vectors and insufficient accuracy in parameter role extraction in commodity news event extraction [20]. This model combines self-attention mechanism, average pooling graph convolutional network, and dependency parsing tree. By using the ComBERT pre-trained model for data preprocessing, it enhances the correlation between trigger words and entity vectors and improves the accuracy of role segmentation using the average pooling function.

3 Methodology

Vehicle images contain various rich road information, with a huge amount of data, and a wide variety of vehicle types on the road, making it difficult to distinguish in detail. At the same time, different types of vehicles vary in size, posing great challenges for multi-class vehicle object detection. If using second-stage object detection algorithms like Faster RCNN, although the accuracy is high, the inference speed is slow, making it unsuitable for real-time detection applications. Additionally, it is complex, with a complicated training and deployment process involving candidate region generation and multi-stage processing, making it difficult to apply in resource-constrained environments. Algorithms like RetinaNet use Focal Loss to improve detection accuracy for small targets and imbalanced data, but require large computational resources and high hardware configurations. Although EfficientDet strikes a balance between performance and speed, it requires high-performance hardware to fully leverage its advantages. SSD (Single Shot MultiBox Detector), while faster, is not as accurate as YOLOv5, especially in handling small targets and complex backgrounds. Compared to YOLOv5, YOLOv3 lags behind in both accuracy and speed, and its code and community support are not as active as YOLOv5. Although R-FCN (Region-based Fully Convolutional Networks) is faster, it lacks flexibility in handling high-resolution images and complex scenes. Although CenterNet is simple and efficient, it does not perform as well as YOLOv5 in handling small targets and heavily occluded scenes. Therefore, proposing a road vehicle multi-class object detection method based on the YOLOv5 algorithm has the advantages of high real-time performance, high accuracy, simple and efficient processes, and lightweight models, enabling vehicle classification and detection in complex road environments.

Firstly, collect a dataset of images containing various types of vehicles and accurately annotate the vehicles in the images, including bounding boxes and category information. The vehicle image dataset should include images from various angles, lighting conditions, and environments. Next, perform operations such as scaling, rotating, and cropping on the training data to enhance the model's generalization ability. Then, configure the network structure and training parameters of YOLOv5 according to the requirements of the multi-class vehicle object detection task. Subsequently, train the YOLOv5 model using the annotated vehicle dataset to learn the multi-class features of vehicles. After model training, optimize the model using the YOLOv5 loss function, including bounding box regression loss, confidence loss, and classification loss. Evaluate the model's performance on the validation set using metrics such as mAP, Precision,

Recall to measure detection accuracy. Adjust hyperparameters such as learning rate and batch size based on the evaluation results to optimize the performance of the vehicle multi-class object detection model. Finally, test the model on an independent test set to ensure its generalization ability, and deploy the trained model to an actual vehicle detection system. Utilize the deployed model to perform real-time vehicle detection on vehicle images or video streams, and perform post-processing operations such as NMS to remove redundant detection boxes and improve detection quality.

The basic algorithm framework Ultralytics/YOLOv5 used in this paper is an advanced object detection model developed based on Jocher Glenn's work, which exhibits good robustness and generalization ability in the problem of multi-class vehicle object detection [21]. For the selection of hyperparameters, the Model Architecture adopts YOLOv5n, which can better adapt to the computational resources and accuracy requirements of the dataset in this problem. At the same time, the model needs to crop the Input Size to dimensions such as 640*640 or 1280*1280. It is worth noting that the number of categories in the yaml file needs to be preset, and the number of categories in this dataset is 21, indicating that there are 21 different vehicle models that need to be classified and recognized.

4 Results

The training data for the multi-class vehicle model in this paper is selected from the road vehicle image dataset on the Kaggle website, which consists of Bangladeshi road vehicle images labeled with YOLOv5 tags. The dataset contains a total of 3004 images, with the image size of the test and validation datasets approximately 640 pixels by 640 pixels, and the labels are predefined. The training set of the dataset comprises 2704 images, and the validation set consists of 300 images, containing data for 21 categories of vehicles. There are 2568 vehicle target instances in the validation dataset.

Based on the default hyperparameters of Ultralytics/YOLOv5, optimizations and improvements were made in this paper. The pre-training weights chosen for training are from the YOLOv5n model, which, although not significantly superior in numerical performance, is more suitable for the multi-class vehicle classification problem studied in this paper, thus improving mAP. Additionally, the epoch is set to 300 to achieve optimal training results. The experiments were conducted using an NVIDIA GeForce RTX 3080 Laptop GPU, so the batch_size was set to 8, the number of workers was set to 4, and patience was set to 100 to stop training when the model reaches its optimum.

The evaluation metrics used in this paper are mAP, Precision, and Recall values, as shown in Table 1 with the experimental results.

Table 1. Experimental Results.

Class	Images	Instances	Precision	Recall	mAP50	mAP50-95
all	300	2568	0.669	0.358	0.41	0.244
bicycle	300	32	0.47	0.375	0.366	0.176
bus	300	425	0.799	0.567	0.661	0.392

Table 1. (continued).

Car	300	842	0.782	0.691	0.741	0.466
Minibus	300	2	0.645	0.5	0.495	0.446
Minivan	300	110	0.494	0.436	0.397	0.274
motorbike	300	335	0.619	0.505	0.51	0.173
pickup	300	142	0.525	0.218	0.299	0.163
policecar	300	1	1	0	0	0
rickshaw	300	192	0.717	0.708	0.694	0.397
scooter	300	1	1	0	0.00229	0.000457
suv	300	60	0.26	0.264	0.175	0.113
taxi	300	19	1	0	0.509	0.299
three wheelers	300	252	0.813	0.583	0.681	0.429
truck	300	84	0.546	0.607	0.609	0.375
van	300	62	0.267	0.161	0.205	0.127
wheelbarrow	300	9	0.761	0.111	0.213	0.0714

The experimental results indicate that the improved multi-class vehicle detection algorithm based on Ultralytics/YOLOv5 in this paper achieves high accuracy and recall rates on most vehicle types. The algorithm can accurately detect various vehicle types such as cars, buses, motorbikes, rickshaws, etc., demonstrating good generalization and robustness.

The comparison between the actual anchor boxes and the algorithm-detected anchor boxes is illustrated in Figure 1.



Fig. 1. Anchor Box Comparison (Image on the left side: Real Anchor Boxes; Image on the right side: Ultralytics/YOLOv5).

Figure 1 illustrates the comparison between the detection results of the algorithm proposed in this paper and the real anchor boxes. The left image shows the effect of vehicle label annotations, while the right image depicts the detection results of the improved Ultralytics/YOLOv5 algorithm.

5 Discussion

The dataset used in this paper consists of 3004 images, including 2704 training set images and 300 validation set images, with most of the dataset images being 640 pixels*640 pixels. Compared to similar studies in object detection, the scale of this dataset is relatively small. The pre-training weights used in the algorithm of this paper differ from those used in similar studies. Adopting the pre-training weights of YOLOv5n can maximize computational efficiency under limited hardware conditions, but it also results in very high CPU thread usage.

Due to the relatively small sample size involved in this study and the presence of multiple effective anchor boxes within individual samples, coupled with considerations of computational power constraints, a batch size of 8 was set. This helps control sampling freedom and aids in model generalization. Considering the total size of the training set and sample density, the number of epochs was set to 300, while the patience was set to 100 to ensure that the model receives sufficient training.

Results show variations in mAP, Precision, and Recall performance across different vehicle categories. Firstly, some classification label samples have fewer quantities, leading to insufficient training. Secondly, certain vehicle types exhibit obvious features, while others may have less distinct features due to occlusion or other factors. Generally, as the number of categories increases, the success rate of recall for individual categories tends to decrease. In contrast, precision is not affected by this issue, resulting in better numerical performance. Compared to similar multi-classification models, the model in this paper demonstrates better numerical performance in classifying most vehicles with distinct features.

In complex road environments, some vehicles may overlap, leading to poor detection results for the algorithm in this paper. However, under good lighting conditions and minimal vehicle overlap, the proposed method can effectively handle the multi-classification of vehicles.

6 Conclusion

This paper proposes a vehicle multi-classification object detection method based on the Ultralytics/YOLOv5 algorithm for vehicle images under complex road traffic conditions. The method enhances the model's generalization capability by performing operations such as scaling, rotation, and cropping on the training data. Based on the requirements of the vehicle object detection multi-classification task, the network structure and training parameters of YOLOv5 are configured, and the model is optimized using YOLOv5's loss function. Compared to similar studies, the algorithm in this paper, through data augmentation, adjustment of training parameters, and reasonable selection of pre-training weights, can detect vehicles in road environments with a wide variety of categories and achieve high detection accuracy. When detecting most common vehicle types, the method proposed in this paper demonstrates stronger

robustness and higher detection accuracy compared to other object detection methods. However, due to limitations of the model itself and the presence of numerous vehicle types in specific complex road environments, the detection effectiveness of rare vehicle types in this paper did not reach the optimal level. In future work, we will seek more urban vehicle image datasets for training and explore research on object detection tasks under low-light and overlapping conditions.

References

- [1] Liang, S., Wang, S., Chen, J., & Yu, J. (2023). Research on port ship target detection algorithm based on coastline candidate area. *Radio Engineering*, 10, 2270-2276.
- [2] Zhao, Q., & Yang, Y. (2023). Lightweight remote sensing vehicle small target detection algorithm based on multiple pyramids. *Electronic Measurement Technology*, 13, 88-94. doi:10.19651/j.cnki.emt.2211674.
- [3] Lv, H. T., & Jia, X. L. (2024, June 27). Lightweight giant panda target detection model based on attention mechanism. *Laser Journal*, 1-7. <http://kns.cnki.net/kcms/detail/50.1085.TN.20231207.1051.006.html>.
- [4] Li, S., Wang, C., & Wang, M. (2023). Extremely lightweight aerial image port ship target detector. *Computer Engineering and Design*, 12, 3606-3612. doi:10.16208/j.issn1000-7024.2023.12.012.
- [5] Chen, A., Yu, Y., Zhao, H. R., et al. (2024, June 27). Detection of damaged areas in ancient murals based on nested U-shaped structure network. *Computer Engineering and Applications*, 1-11. <http://kns.cnki.net/kcms/detail/11.2127.TP.20231218.1537.026.html>.
- [6] Yuan, J., Lan, Z., & Xiong, P. (2023). Unmanned aerial vehicle target detection method based on region adaptive threshold. *Computer and Digital Engineering*, 12, 2883-2888+2990.
- [7] Chen, J. M., Zhang, W. D., & Tan, R. P. (2024, June 27). Drug recommendation algorithm based on dialogue structure and graph attention network. *Journal of Computer Science and Exploration*, 1-12. <http://kns.cnki.net/kcms/detail/11.5602.TP.20240108.0943.002.html>.
- [8] Zhang, Z., & Yang, H. (2024). Improved VGG network face expression recognition method based on LBP and attention mechanism. *Software Engineering*, 01, 23-26+31. doi:10.19644/j.cnki.issn2096-1472.2024.001.006.
- [9] Hu, T., Gao, X., Hua, Y., & Cai, L. (2024). Adaptive apple image multi-defect detection based on deep learning. *Journal of Shandong University of Technology (Natural Science Edition)*, 01, 42-47. doi:10.13367/j.cnki.sdgc.2024.01.005.
- [10] Wu, H., Zhang, Y., & Hu, P. (2024). Small target detection method for unmanned aerial photography images. *Journal of Anhui University of Technology (Natural Science Edition)*, 01, 65-73.
- [11] Hua, C., Mo, S., Chen, Y., Hu, H., & Wu, S. (2024). Analysis of target detection algorithm based on infrared images. *Automotive Practical Technology*, 02, 59-66. doi:10.16638/j.cnki.1671-7988.2024.002.012.
- [12] Wang, C., Wang, Z., & Li, W. (2024). Controller sleep behavior recognition based on dual-stream adaptive graph convolutional network. *Journal of Safety and Environment*, 02, 596-601. doi:10.13637/j.issn.1009-6094.2023.0240.
- [13] Zou, Z. B., Liu, Y. Z., Liao, Z. H., et al. (2024, June 27). Spatio-temporal attention graph convolutional network for dynamic traffic flow prediction. *Journal of Shandong University (Engineering Edition)*, 1-12. <http://kns.cnki.net/kcms/detail/37.1391.t.20240221.1017.004.html>.

- [14] Zhou, H., Zhan, F., Zhou, C., Ren, T., & Luo, L. (2024). Pedestrian re-identification method based on attention mechanism and multi-branch joint. *Microelectronics and Computer*, 02, 1-10. doi:10.19304/J.ISSN1000-7180.2023.0016.
- [15] Xie, J., & Zhang, J. (2024). Review of graph convolutional neural networks. *Journal of Shaanxi Normal University (Natural Science Edition)*, 02, 89-101. doi:10.15983/j.cnki.jsnu.2024003.
- [16] Chai, B., Li, Z., Zhao, X., & Wang, R. (2024). Deep document clustering model based on generalized graph convolutional neural network. *Journal of Nanjing Normal University (Natural Science Edition)*, 01, 82-90.
- [17] Yang, X. Y., Ma, S., Zhang, Z. L., et al. (2024, June 27). Social recommendation based on preference-aware denoising graph convolutional network. *Computer Engineering*, 1-10. <https://doi.org/10.19678/j.issn.1000-3428.0068748>.
- [18] Tang, Z., Wu, Y., Li, C., & Tang, Y. Learning resource recommendation based on knowledge graph convolutional network. *Computer Engineering*. doi:10.19678/j.issn.1000-3428.0068409.
- [19] Wang, F., Chen, X., Ren, W., Guan, Y., Han, Z., & Tang, Y. Low-light target detection algorithm based on image adaptive enhancement. *Computer Engineering*. doi:10.19678/j.issn.1000-3428.0068407.
- [20] Luo, Q., Li, H., Wang, Z., Gan, C., & Hu, Z. Event extraction of commodity news based on self-attention mechanism and average pooling graph convolutional network. *Journal of Chengdu University of Technology (Natural Science Edition)*.
- [21] Jocher, G., Chaurasia, A., Stoken, A., Borovec, J., Kwon, Y., Michael, K., ... & Mammanna, L. (2022). ultralytics/yolov5: v6.2-yolov5 classification models, apple m1, reproducibility, clearml and deci.ai integrations. Zenodo.

Predicting Patient Waiting Time and Detecting Overload in Emergency Department through Machine Learning

Zihan Qian^{1,*}, Xuanyi Shen², Rongshuo Shang³
 {qzh@smail.nju.edu.cn¹}

School of Intelligence Science and Technology, Nanjing University, Suzhou, China¹

School of Electronic and Information Engineering, Tongji University, Shanghai, China²

College of Resources and Environmental Sciences, Nanjing Agricultural University, Nanjing, China³

*corresponding author

[†] These authors contributed equally to this work.

Abstract. Accurately predicting patient waiting times and detecting workflow overload in emergency departments are critical challenges that significantly impact patient care and resource management. Despite advancements in patient waiting time prediction, current methodologies often struggle with universal applicability in practical settings and fail to accurately capture extreme values. This study proposes a robust and generalized predictive model tailored to the specific challenges of ED workflows to address these gaps. A dataset containing multiple variables that record the workflow within an emergency department is utilized, and a systematic exploration and comparison of specific machine learning and deep learning models are conducted. Different machine learning models are compared, and a model is developed to enhance the accuracy of prediction. The effectiveness of the model in detecting ED overload is evaluated, and its generalization capability is improved through feature selection and feature classification. The proposed model demonstrates superior accuracy in predicting patient waiting times and exhibits high sensitivity in detecting workflow bottlenecks. The model's ability to operate effectively with fewer variables enhances its generalization ability across different ED facilities. It is found that machine learning models can effectively capture patient waiting peaks, which are critical indicators of ED overload. By detecting these overload conditions, hospitals can optimize resource allocation proactively and address overload issues promptly. In summary, this study provides a generalized model with strong predictive accuracy for patient waiting times and the ability to detect system overloads in healthcare settings, contributing to improved overall ED system performance.

Keywords: Patient Waiting Time, Emergency Department, Regression Model, Machine Learning, Deep Learning, Overload Detection

1 Introduction

In recent years, the extraction of real-time information for the allocation of medical resources and the optimization of emergency department workflows has emerged as a prominent trend. Healthcare systems must minimize waiting times and prevent workflow overload to enhance patient satisfaction and overall quality of care. These all can be improved through the accurate patient waiting time prediction and the identification of bottlenecks in patient flow. Past research has shown that longer waiting times lead to more consumption and inefficiency [1]. The study aims to develop an accurate waiting time prediction model and to enhance resource allocation by detecting bottlenecks and dealing with them.

In this paper, an examination of both machine learning (ML) and deep learning (DL) techniques is conducted, with the intention of utilizing them for predicting patient waiting times and identifying workflow overloads. Compared to traditional basic methods, which have been extensively discussed in previous papers, deep learning models are said to offer the advantages of reducing errors and achieving higher accuracy. Therefore, the goal is to enhance the accuracy of predictions through an analysis of these advanced methods. [1, 2].

After the discussion on the paper's purpose, it is noted that there are more aspects that need to be addressed. The study is confronted with several evident challenges, including managing diverse data sources, selecting suitable machine learning models, integrating specific domain knowledge of the healthcare system, and ensuring the scalability and robustness of the proposed solutions. It is hoped that the implementation of the proposed model will enhance patient experience in the emergency department, identify workflow bottlenecks, and contribute to the comprehensive optimization of emergency room management.

2 Literature Review

In the current society, as the healthcare industry digitizing and developing really fast, the complexity and quantity of healthcare data have significantly increased. Electronic medical records (EMRs) provide healthcare organizations with extensive operational data, including processing times, scheduling records, examination types, and various resource characteristics [3]. These data are routinely recorded by hospital information systems (HIS) in a uniform format that complies with major healthcare standards such as Health Level 7 (HL7) [4], Fast Healthcare Interoperability Resources (FHIR) [5], and Digital Imaging and Communications in Medicine (DICOM) [6]. Effectively describing complex healthcare operations requires synthesizing all these data. However, due to the vast amount of data and the difficulty for humans to manually analyze its characteristics, these data are often under-utilized in operational analytics.

Modern healthcare data can be divided into several major domains based on content and application area, such as hospital information, medical imaging, and other sources. These data originate from various hospital departments, including imaging departments [7], biochemistry labs, and surgical suites. Most hospitals already have these data streams in place, generating and collecting new data on a standardized basis while ensuring the data are securely stored in compliance with legal requirements. Consequently, modern hospitals have amassed a wealth of data documenting their

operations and outputs, which can be used to build operational state models.

Predicting waiting times in emergency rooms (ERs) is crucial for patient satisfaction and operational efficiency. Traditional queuing theory has been enhanced by machine learning (ML) and deep learning (DL) approaches, offering more accurate and adaptable solutions. High-dimensional Gradient Boosting Machines significantly outperformed traditional models, emphasizing the necessity of sophisticated ML models for optimal hospital operations [8]. Pinykh highlighted ML's scalability in handling complex patterns in ER operations [9]. Pattnayak showed DL's superior accuracy over traditional methods, reducing human error and enhancing ER efficiency [10]. Kyritsis used a neural network whose adaptability was demonstrated across different industries [11].

While traditional machine learning (ML) methods have shown promise in healthcare applications, they often struggle with achieving high accuracy and generalization across diverse datasets. This gap in performance necessitates further refinement of prediction models. Studies have addressed this by enhancing accuracy through techniques such as outlier exclusion and the integration of system knowledge [12]. Advanced methods, including Ordinary Least Squares (OLS), ridge and LASSO regressions, Random Forest, and Quantile regression, have been shown to significantly improve predictive accuracy [13]. Additionally, in complex scenarios like multi-stage queues, transforming transactional datasets into ML-ready formats and employing grid search techniques have further optimized these models [14]. The thesis of this research is to explore and develop ML models that not only improve accuracy but also enhance generalization performance across various healthcare applications, particularly in predicting workload and optimizing resource allocation in emergency departments and beyond. This leads to the research question: How can ML models be further refined to enhance both accuracy and generalization in healthcare settings?

Furthermore, sensitive overload detection and effective load management is critical for enhancing healthcare efficiency, particularly in Emergency Department (ED) operations. Research highlights that key areas such as bottleneck detection and workload prediction can significantly improve ED efficiency. For instance, it is identified that long waiting times due to treatment delays, especially during treatment in progress and emergency room holding (ERH) procedures, using simulation models [15]. Machine learning further enhances this process by accurately predicting workload in a research with over 200,000 patient visits analyzed to predict work relative value units (wRVUs) [16]. These predictive algorithms facilitate real-time load balancing and resource optimization. Additionally, combining machine learning with optimization techniques can improve hospital scheduling systems, including operating room efficiency and appointment scheduling [17, 18]. These advancements highlight the importance of integrating machine learning and optimization to enhance resource allocation and scheduling in healthcare settings, ultimately reducing congestion and improving patient outcomes [19, 20].

With the ongoing digitization of healthcare data and the development of sophisticated analytical tools, the integration of advanced machine learning and deep learning models with operational data from EMRs and HIS presents significant opportunities for enhancing healthcare system scheduling. From predicting ED waiting times to optimizing exam schedules, these technologies offer improved accuracy and efficiency.

3 Dataset

3.1 Source and Description

Specially, the dataset used in this article is provided by the Medical Analytics Group [21], placed in the core of Massachusetts General Hospital, which is ranked as the best hospital in the country by U.S. News & World Report. The data has been posted on their Nature Machine Intelligence article [9] and their official website, inviting those interested in machine learning and operations research to explore their operations dataset as a challenge [22]. The study utilizes real-world data collected by the Medical Analytics Group, with the aim of extracting more information and constructing a more precise model. This model is designed to predict patient waiting times with greater accuracy and to identify bottlenecks in the overload states of medical facilities.

Clinical workflow outcomes are influenced by a variety of factors, and no single factor can fully explain delays or patient waiting times. Current delays in healthcare organizations can be related to staffing, patient arrival patterns, time of day, complexity of tests, bottlenecks in the operating environment, holidays, weather, and many other factors [23]. Pinykh's dataset contains data on both no-appointment (F4) and appointment (F1, F2, and F3) patients, covering approximately 600 to 1,000 days of complete patient flow records [9].

3.2 Details and Features

The dataset comprises time-dependent features such as patient arrival times, examination appointment times, and examination start times. These time-stamped data are crucial for understanding patient flow and hospital operations. Additionally, dynamic features, like the number of patients matching the scheduled time after the current time, aid hospitals in managing and optimizing patient flow and waiting times.

To get a comprehensive view of daily operations, the dataset also tracks the cumulative number of exam delays, the number of exam delays in the previous hour, and the number of patients scheduled before the current patient. These metrics help hospitals adjust operational strategies in real-time to reduce delays and enhance efficiency.

Reflecting demand and resource allocation for various exam types, the dataset includes the number of patients waiting for different types of exams (e.g. chest, pediatric, neurological, abdominal, vascular, cardiac, and musculoskeletal). It also contains facility-level characteristics, such as the total performed hours for ongoing examinations to help optimize equipment use.

To assess short-term operational workload, the dataset records average waiting times for the last few customers. This information can be used to evaluate operational load and resource requirements in the short term.

In addition, the dataset is from only one hospital, so there are limitations in our model when making predictions in different hospitals.

3.3 Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) techniques are applied to understand the data distribution, detect outliers, and identify significant patterns that could influence model development.

3.3.1 Data Cleaning

In the process of Exploratory Data Analysis (EDA), ensuring comprehensive and accurate data cleaning is crucial for the success of the research. The statistical dataset of the health care system that is being analyzed includes four worksheets, each recording statistics for different medical imaging technologies: X-ray (XR), Computed Tomography (CT), Magnetic Resonance (MR), and Ultrasound (US). Initially, these worksheets contain 89 variables, with data counts of 42,767 entries for Worksheet 1, 15,653 for Worksheet 2, 23,584 for Worksheet 3, and 48,431 for Worksheet 4.

In the face of the dataset's complexity, a series of specific data cleaning steps have been undertaken: missing values have been identified and handled, data formats have been standardized, and outliers have been identified and dealt with. Utilizing R, 3 missing values were identified in Worksheet 1, while no missing data were found in the other three worksheets. To avoid the adverse impact that directly deleting rows with missing values might have on subsequent research, a more nuanced approach was adopted: the missing values were replaced with the mean value of the respective row.

In terms of handling outliers, a detailed analysis was conducted using boxplots generated in R. All feature values were normalized to ensure a mean of zero and a variance of one. Apart from binary logical variables representing yes/no states (1 for 'yes' and 0 for 'no'), a certain number of outliers were observed in all other features within the boxplots. Considering the complexity of the healthcare system and the precision and facticity required for model predictions, these outliers were deemed to be normal occurrences within the healthcare statistical data, reflecting the actual conditions of the healthcare system. Therefore, the decision was made to retain these outliers to maintain the authenticity and integrity of the dataset.

3.3.2 Descriptive Statistics

In the dataset, the primary aim is to predict the 'waiting time'. Thus, the descriptive statistic of waiting time was done first. The dataset comprises 130,431 data points. And the mean of it is 7.295, with a standard deviation of 25.898. Upon examining its range, the minimum and maximum values are -497.000 and 360.000, respectively, while the lower quartile, median, and upper quartile are 1.000, 6.000, and 15.000, respectively. Furthermore, given the additional consideration of the relationship between 'waiting time' and the 'time series', visualization of these two variables was performed to provide an initial understanding.

From Figure A-1a, it can be found that at some time points, there are only very few waiting time that is possible, while at others, there are a lot of feasible waiting time amounts. Hence, it can be inferred that the specific time of day is likely to have a significant impact on 'waiting time' predictions.

Besides the targeted waiting time data in the collected dataset, there are 66 features that are related to the waiting time. The types of certain features of each facility differ slightly from each

other due to their various functions. To describe the features, they can be divided into three kinds of variables: discrete variables, continuous variables and 0-1 two-valued variable. Examples of these features are illustrated in Figure A-1c.

3.3.3 Correlation Analysis

In pursuit of the best possible model fit, a meticulous analysis was conducted on the correlation between each independent variable and the dependent variable 'Wait', with the most influential features being carefully selected. During this process, three variables that were not quantifiable in terms of time points were eliminated to ensure the precision of the analysis.

A correlation matrix is served as a powerful means to illustrate the relationships among variables. After a correlation matrix was generated from the refined dataset, a choice was made to visualize these relationships with a heat map in R. Additionally, by applying hierarchical clustering to sort the matrix, the interpretation of the heat map was made more accessible, particularly given the extensive volume of data that was being handled. In the heat map, the darker the color of the convergence area between variables, the stronger the correlation. This approach not only enhances the visual representation of the data but also facilitates the understanding of the complex interplay between variables.

From Figure A-1a, It was found that the variable 'delayedinline' had the strongest positive correlation with waiting time in Worksheet 1, with a Spearman's rank correlation coefficient of 0.28122, while the variable 'noneinline' showed the strongest negative correlation with waiting time, with a Spearman's rank correlation coefficient of -0.15024.

4 Methodology

4.1 Previous Machine Learning Techniques

The objective of this study is to develop predictive models for patient waiting times, leveraging existing datasets to train these models for accurate forecasting. The paper commences with an examination of the comparative efficacy and efficiency of various elementary machine learning algorithms in the context of predictive modeling.

4.1.1 Dataset Split

The methodology section delineates the training strategy employed. Data from four distinct medical facilities were subjected to independent training regimens, thereby cultivating facility-specific predictive models. The dataset was partitioned into a training subset, comprising 70-80% of the data selected at random, and a test subset, encompassing the residual 20-30%. Each model underwent a series of six training iterations, with the optimal iteration, determined by performance metrics, being retained as the definitive model.

4.1.2 Experiment on Different ML models

In pursuit of a comprehensive assessment of the predictive capabilities of diverse machine learning techniques, the study encompasses a spectrum of algorithms, including linear regression, Naive Bayes classifiers, Support Vector Machines (SVMs) with Gaussian kernels, single decision trees, and ensembles of decision trees, namely random forests. Some of these methods are also models that are discussed in precious articles [9] in this field. But not every model is discussed with a concrete result.

The linear regression analysis was conducted at multiple levels of complexity: a full multivariate regression incorporating all variables, a reduced multivariate regression focusing on a subset of significant predictors, and univariate regressions for individual influential predictors. The linear regression model can be mathematically described as:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \varepsilon$$

where y represents the patient waiting time, β_0 is the intercept, $\beta_1, \beta_2, \dots, \beta_n$ are the coefficients for each predictor x_1, x_2, \dots, x_n , and ε is the error term.

The SVM approach was standardized with a Gaussian kernel to facilitate model convergence. The SVM model with a Gaussian kernel can be expressed as:

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$$

where $K(x_i, x_j)$ is the kernel function, x_i and x_j are data points, and σ is the bandwidth parameter.

The random forest models were parameterized with varying numbers of trees and tree depths to explore the impact of model complexity on predictive accuracy. The prediction for a random forest model is the aggregation of predictions from individual decision trees.

4.1.3 Parameter Settings

Optimal parameters for ML models are determined through cross-validation and grid search techniques to ensure the highest possible predictive performance.

The internal model parameters are dynamically optimized through the training process, contingent upon the characteristics of the training data set. Consequently, during experimentation, it is imperative to meticulously adjust the model's external tunable parameters in response to the predictive outcomes, thereby incrementally refining the model's performance. The subsequent discourse will elucidate the parameter design process for select models.

For example, within the ensemble of machine learning models, the Random Forest model necessitates a systematic refinement of both the quantity and the depth of constituent decision trees. Through iterative experimentation, it was determined that, to effectively accommodate the extensive parameter space of the medical system dataset utilized in this research, a robust ensemble of decision trees and increased tree depth are essential for achieving superior predictive accuracy. The optimal configuration identified in this study for the Random Forest model comprises 300 trees with a depth of 20 splits.

By incorporating these mathematical formulations, it can enhance the precision and clarity of the new predictive modeling approach while preserving the original narrative structure.

4.2 Neural Networks for Prediction

In addition to the foundational machine learning methodologies, the dataset for the prediction of patient waiting times is distinguished by the subtle influence of each feature on the overall waiting duration, with no single characteristic exerting a pronounced effect on the outcome. In light of this, the present study incorporates an exploration of deep learning models based on neural networks to address the prediction task, with a focus on evaluating the performance of such models in scenarios characterized by a multitude of features, each with a relatively minor impact.

4.2.1 Our Neural Network Model

The research commenced with the development of a rudimentary neural network framework to gauge its efficacy in predicting patient waiting times. The architecture was composed of either a solitary hidden layer or a dual-layer configuration, with each layer populated by either 10 or 20 neurons, which has already been used in this field to solve such patient waiting time problems [9]. Contrary to complexity, these elementary networks demonstrated an aptitude for handling the multifaceted nature of the problem, achieving a level of predictive accuracy that rivals or exceeds that of more established machine learning algorithms.

Nevertheless, while the rudimentary neural network configurations have modestly enhanced the precision of waiting time predictions, this study introduces an advanced neural network architecture specifically crafted to augment the model's predictive fidelity. This novel architecture expands the neuron count in each of the two hidden layers. Prior to each hidden layer, a Batch Normalization layer is integrated to facilitate the model's capacity to conform to the data's nonlinear dynamics. ReLU activation functions are utilized throughout to introduce nonlinearity at each layer.

The optimal configuration attained by some experiments for the neural network model entails two hidden layers, each populated with 256 neurons, preceded by a Batch Normalization layer to facilitate the fitting of data with nonlinear relationships, and activated by the ReLU function.

4.2.2 Loss Measurement

In this research, the neural network model's training was tuned with external parameters, guided by ongoing assessments of the model's predictions. The Mean Absolute Error (MAE) and Mean Squared Error (MSE) served as key performance indicators for the model's predictive accuracy. Given the volatile nature of our emergency room dataset, which includes many extreme outliers, MSE was utilized for training due to its properties that aid in gradient descent and quick convergence. Despite MSE's advantages, its susceptibility to outliers can lead to exaggerated error magnification. MAE was used for final evaluation, as it averages the absolute differences between predictions and actuals. The formulas for MSE and MAE are:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2, \quad \text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|.$$

For a more nuanced evaluation of the model's performance, U05 and U10 metrics are also considered. These metrics measure the accuracy of the model by calculating the proportion of predictions with absolute errors less than 5 or 10, respectively. The formulas are:

$$U05 = \frac{1}{n} \sum_{i=1}^n \mathbb{I}(|y_i - \hat{y}_i| < 5), \quad U10 = \frac{1}{n} \sum_{i=1}^n \mathbb{I}(|y_i - \hat{y}_i| < 10)$$

where \mathbb{I} denotes the indicator function, and n is the total number of observations. This approach ensures a comprehensive assessment of the model's predictive capabilities, especially in the presence of variability and outliers.

4.2.3 Training

In addition, a flexible learning rate schedule has been put in place, which reduces the learning rate by a factor of ten whenever the model's performance plateaus, or in other words, when it stops improving in terms of loss reduction. This mechanism is designed to guide the model towards a more optimal solution. Moreover, a strategy for learning rate decay has been incorporated, with the training process planned to last for 50,000 epochs. The starting learning rate (η_0) is set at 0.01, and there's a mechanism in place to reduce the learning rate to one-tenth of its initial value if the model's loss does not decrease significantly over a period of 10 consecutive epochs.

The mathematical expression for the learning rate decay is as follows:

$$\eta_t = \begin{cases} \eta_0 & \text{if } t < t_0, \\ \eta_0 \cdot 0.1^{\lfloor \frac{t-t_0}{T} \rfloor} & \text{if } t \geq t_0. \end{cases}$$

Here, η_t denotes the learning rate at a specific epoch t , t_0 refers to the epoch at which the first plateau in performance is identified, and T signifies the 10-epoch interval.

The goal of the training process is to minimize the loss function $L(\theta)$, where θ symbolizes the parameters of the neural network. This optimization is carried out using the gradient descent method, and the parameters are updated according to the following rule:

$$\theta_{t+1} = \theta_t - \eta_t \nabla L(\theta_t)$$

This entire process is facilitated by the Adam optimizer in PyTorch.

4.3 Peak Capture & Overload Detection

4.3.1 Peak & Overload Definition

Predicting peaks and detecting overloads in emergency department (ED) data are crucial for optimizing resource allocation and reducing extreme waiting. In this section, the aim is to identify peaks and overload conditions using machine learning models. Various models are validated separately to find the most sensitive overload detector, containing linear regression, decision tree or forest as well as the neural network that have been constructed.

peak is defined as a binary variable P , where

$$P = \begin{cases} 1, & \text{if the waiting time} > 10 \text{ minutes,} \\ 0, & \text{otherwise.} \end{cases}$$

Similarly, *overload* is defined as a binary variable O , where

$$O = \begin{cases} 1, & \text{if the number of individuals waiting} > 5, \\ 0, & \text{otherwise.} \end{cases}$$

The model takes a set of variables as input, denoted as $\mathbf{X} = \{x_1, x_2, \dots, x_n\}$, where each x_i represents a different feature relevant to the prediction of waiting times and overload conditions. The model outputs the predicted waiting time \hat{T} , which is then used to derive the predicted peak \hat{P} and the predicted overload \hat{O} . The aim here is to find the most suitable model offering the patients their waiting time, and another model for the hospital manager to detect overload and schedule medical resources.

4.3.2 Peak Capturing Sensitivity Measurement

For the peak detection task, which is treated as a classification problem, a confusion matrix of predicted-peak and actual-peak is generated, thus calculating several classification metrics:

- Accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$
- Precision = $\frac{TP}{TP+FP}$
- Recall = $\frac{TP}{TP+FN}$

where TP , TN , FP , and FN represent the true positives, true negatives, false positives, and false negatives, respectively.

A comprehensive comparison of various machine learning models, including our neural network and other baseline models, was conducted to determine the most suitable model for peak capture. The models were evaluated based on their performance in predicting both waiting times and peaks, with a particular focus on their ability to accurately detect peak conditions (i.e., when waiting time exceeds 10 minutes).

4.3.3 Overload Detection Classifier

Description The detection of overload conditions in the healthcare system is also framed as a classification task. This task involves determining whether the system is in an overload state based on patient waiting times and the current operational status of the healthcare facility. The classifier's effectiveness in this context is again evaluated using a confusion matrix, with metrics such as accuracy, precision, recall, and F1-Score being calculated.

Measurement In the context of healthcare, the primary concern is to ensure that no overload condition goes undetected. Therefore, recall, defined as the proportion of actual overload cases correctly identified by the model, is the most critical metric. Maximizing recall ensures that the system is adequately prepared for every potential overload, minimizing the risk of missing a critical situation that could compromise patient care.

$$\text{Recall} = \frac{TP}{TP + FN}$$

where TP represents true positives (correctly detected overloads), and FN represents false negatives (missed overloads). A high recall value indicates that the model is effective in capturing all instances of overload, thus providing a reliable warning system for healthcare providers.

Correlation Between Overload and Peak In addition to evaluating the individual performance of the overload and peak detection models, it is essential to assess the correlation between these two phenomena. A strong correlation between overload and peak detection would suggest that the model accurately reflects the operational status of the healthcare system, providing a holistic view of its capacity and performance. In this study, the correlation between the variables *overload* (O) and *peak* (P) is analyzed using binary classification methods. The variable *overload* (O) and *peak* (P) is defined as follows:

- $O = 1$ (True Positive): when the number of people waiting exceeds 5.
- $O = 0$ (True Negative): otherwise.
- $P = 1$ (Predicted Positive): when the waiting time exceeds 10 minutes.
- $P = 0$ (Predicted Negative): otherwise.

To evaluate the effectiveness of using the *peak* variable (P) to predict the *overload* variable (O), the confusion matrix and several evaluation metrics were computed, including accuracy, precision, recall, and F1 score.

4.4 Generalization Improvement

4.4.1 Feature Selection and Decline

Given that not all hospitals may track these features, feature selection on the dataset was conducted to enhance the model's generalization performance. The four worksheets classified by device in the original dataset were used as the starting independent variable set for feature selection. The data approximated a normal distribution, which allowed us to employ two methods for feature selection: Principal Component Analysis (PCA) and Step Regression Analysis.

Step Regression Analysis involves iteratively adding or removing predictors based on their statistical significance:

$$Y = \beta_0 + \sum_{i=1}^n \beta_i X_i + \epsilon$$

where Y is the dependent variable (waiting time), X_i are the independent variables (features), β_i are the coefficients, and ϵ is the error term. Features are added or removed based on criteria such as the Akaike Information Criterion (AIC) or p-values of the coefficients.

Step Regression Analysis was utilized to identify features with the most significant impact on waiting times from the original set of 83 features. For instance, in the analysis, the number of the key features was narrowed down to 54 from the original 83 in the worksheet.

Subsequently, PCA was applied to rank these selected features based on their impact, from the greatest to the least. This ranking was determined by the cumulative absolute loading values of each variable across all selected principal components. Principal Component Analysis transforms the original variables into a new set of uncorrelated variables called principal components. The principal components are ordered by the amount of variance they capture from the data.

The loadings, which are the coefficients of the linear combination of the original variables, are used to rank the features. The cumulative absolute loading value for a variable X_i across k principal components is given by:

$$L_i = \sum_{j=1}^k |w_{ij}|$$

where w_{ij} is the loading of variable X_i on the j -th principal component.

Step Regression Analysis had already identified 54 features with substantial influence on waiting times so the first 50 original features with the highest cumulative load absolute values were focused on as revealed by the principal component analysis(in Table A-2).

4.4.2 Distinct Feature Groups with Domain Knowledge

Aiming to enhance the generalization performance of our model by leveraging domain knowledge to construct neural network architectures tailored to distinct feature groups, the strategy here involved dividing the feature set into meaningful categories, ensuring each subset of features is processed by a dedicated sub-network that captures the specific patterns and relationships inherent in each group.

The approach here starts by categorizing features into five groups: appointment status, queue status, daily efficiency, immediate efficiency, and check type. Each of these groups contains unique information that can be utilized more effectively when processed separately.

1. **Appointment Status:** This group includes features related to the timing and scheduling of appointments, such as number of patients scheduled in the 30- and 60-minute window before patient arrived.
2. **Queue Status:** This information helps assess the load and performance of the queue management system, allowing healthcare organizations to adjust resource allocation in real-time to reduce patient waiting times, such as number of patients in line measured when a patient arrives, 15, 30, 45 & 60 minutes before.
3. **Daily Efficiency:** Metrics that reflect the overall efficiency of the emergency department, such as average delay/wait for patients for that day.
4. **Immediate Efficiency:** Metrics providing a snapshot of immediate performance metrics and offering a real-time view of ongoing processes, such as the sum of the expected times to complete of the exams in progress.
5. **Check Types:** This group deals with the characteristics and waiting times for different examination types, including 'number of chest examinations', 'number of neurological examinations' and etc.

In the model implementation, each feature group is fed into a dedicated sub-network. Each sub-network consists of two fully connected layers using ReLU activation functions, designed to capture the complex non-linear relationships within each feature group. For a given feature group \mathbf{X}_i (where $i = 1, 2, 3, 4, 5$), the sub-network's output can be represented as:

$$\mathbf{H}_i = \text{ReLU}(\mathbf{W}_{i,2} \cdot \text{ReLU}(\mathbf{W}_{i,1} \cdot \mathbf{X}_i + \mathbf{b}_{i,1}) + \mathbf{b}_{i,2})$$

where $\mathbf{W}_{i,1} \in \mathbb{R}^{h \times d_i}$ and $\mathbf{W}_{i,2} \in \mathbb{R}^{h \times h}$ are the weight matrices, $\mathbf{b}_{i,1} \in \mathbb{R}^h$ and $\mathbf{b}_{i,2} \in \mathbb{R}^h$ are the bias vectors, and $\text{ReLU}(x) = \max(0, x)$ is the ReLU activation function, with h being the hidden layer dimension.

The outputs of these sub-networks are then combined to form a consolidated representation of the input data:

$$\mathbf{H} = [\mathbf{H}_1, \mathbf{H}_2, \mathbf{H}_3, \mathbf{H}_4, \mathbf{H}_5]$$

Here, $\mathbf{H} \in \mathbb{R}^{5h}$ is the merged vector. This combined vector is processed through a final linear layer to generate the model predictions:

$$\hat{y} = \mathbf{W}_f \cdot \mathbf{H} + \mathbf{b}_f$$

where $\mathbf{W}_f \in \mathbb{R}^{1 \times 5h}$ is the weight matrix of the final linear layer, and $\mathbf{b}_f \in \mathbb{R}$ is the bias term.

Across all healthcare scenarios, these five dimensions make it possible to measure operating characteristics effectively for making predictions. The overall model can be succinctly represented as:

$$\hat{y} = \mathbf{W}_f \cdot \left(\sum_{i=1}^5 \text{ReLU}(\mathbf{W}_{i,2} \cdot \text{ReLU}(\mathbf{W}_{i,1} \cdot \mathbf{X}_i + \mathbf{b}_{i,1}) + \mathbf{b}_{i,2}) \right) + \mathbf{b}_f$$

To summarize, features are categorized into five groups: appointment status, queue status, daily efficiency, immediate efficiency, and check types. Each category was processed by dedicated sub-networks with two fully connected ReLU layers, capturing specific patterns within the data. These outputs were merged and passed through a final linear layer for predictions. This model, combining rigorous feature selection and domain-informed subset grouping, demonstrated excellent performance on our sub-NN. By effectively capturing the nuances of various feature groups, it provides accurate waiting time predictions with better generalization performance, making it a valuable tool for emergency departments.

5 Results

5.1 Accurate Waiting Time Prediction

In this study, various prediction models are thoroughly tested, mainly in order to see how well they performed in predicting patient waiting times. The prediction accuracy of these models are evaluated by using two important metrics - mean absolute error (MAE) and mean square error (MSE). The smaller the value, the more accurate the model's prediction are. The data in several tables 1a, 1b, 1c, 1d are carefully analyzed, which show the results of some different machine learning and deep learning models, and the model is compared with these models [9]. In addition,

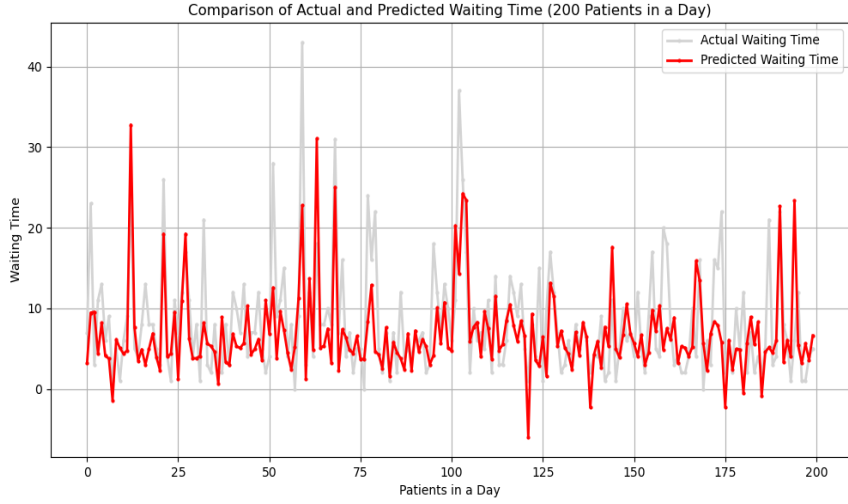


Fig. 1. Visualization of our Predictive Model

MLmodel	MAE	MSE	U05	U10	trainMAE	trainMSE
MostRecentWait	12.837	18.027	0.291	0.525	12.856	18.24
MostRecentWait-Average	12.778	17.969	0.287	0.524	12.809	18.096
LinearRegression	8.797	12.450	0.396	0.683	8.736	12.375
GaussianKernel	12.238	17.353	0.308	0.549	12.297	17.599
ForestSmall_300trees_30splits	12.659	18.132	0.334	0.602	9.012	0.270
NeuralNetwork-[10,10]_Layers	9.172	12.815	0.341	0.608	9.812	13.789
NeuralNetwork-[20,20]_Layers	9.847	13.629	0.312	0.559	11.033	15.350
Our_NeuralNetWork_Model	7.617	10.880	0.196	0.377	6.479	9.032

MLmodel	MAE	MSE	U05	U10	trainMAE	trainMSE
MostRecentWait	20.675	30.227	0.181	0.352	20.763	30.842
MostRecentWait-Average	20.706	30.887	0.181	0.351	20.621	30.519
LinearRegression	17.819	24.776	0.202	0.392	17.649	24.982
GaussianKernel	20.38	30.407	0.192	0.363	20.315	30.224
ForestSmall_300trees_30splits	25.892	36.882	0.151	0.291	0.011	0.413
NeuralNetwork-[10,10]_Layers	18.863	27.833	0.181	0.358	19.802	29.327
NeuralNetwork-[20,20]_Layers	19.083	28.541	0.19	0.365	19.166	28.479
Our_NeuralNetWork_Model	18.594	27.903	0.145	0.284	16.499	22.36

MLmodel	MAE	MSE	U05	U10	trainMAE	trainMSE
MostRecentWait	32.236	48.169	0.121	0.238	32.463	48.207
MostRecentWait-Average	31.98	47.183	0.125	0.241	32.289	48.226
LinearRegression	23.329	30.086	0.139	0.275	23.144	29.864
GaussianKernel	31.568	47.288	0.124	0.244	31.819	47.746
ForestSmall_300trees_30splits	32.937	43.052	0.109	0.211	0.004	0.291
NeuralNetwork-[10,10]_Layers	23.255	30.664	0.130	0.255	25.465	33.339
NeuralNetwork-[20,20]_Layers	23.472	30.717	0.123	0.246	25.538	33.441
Our_NeuralNetWork_Model	22.636	29.402	0.097	0.192	21.881	28.171

MLmodel	MAE	MSE	U05	U10	trainMAE	trainMSE
MostRecentWait	4.718	6.579	0.653	0.921	4.71	6.583
MostRecentWait-Average	4.653	6.503	0.662	0.921	4.641	6.464
LinearRegression	3.822	5.453	0.761	0.94	3.805	5.424
GaussianKernel	3.932	5.557	0.748	0.937	3.923	5.57
ForestSmall_300trees_30splits	4.906	7.201	0.669	0.876	0.015	0.344
NeuralNetwork-[10,10]_Layers	3.829	5.461	0.721	0.933	3.933	5.533
NeuralNetwork-[20,20]_Layers	3.952	5.541	0.644	0.907	4.634	6.186
Our_NeuralNetWork_Model	3.782	5.407	0.612	0.851	3.683	5.249

(c) Comparison of Models on f3 (CT)

(d) Comparison of Models on f4 (XR)

Table 1: Comparison of Models across different datasets

the prediction results of our model is also shown in Figure 1, which lists the predicted and actual waiting times for 200 patients in one day in chronological order.

It is observed that, under identical modeling conditions, the predictive accuracy for various facilities exhibits notable divergence, with potential for substantial discrepancies. Despite these variations, the models demonstrate a commendable level of precision, thereby validating their utility. However, this observation necessitates a deeper inquiry, particularly given the dataset's inherent imbalance. The volume of data associated with different facilities is uneven, with those facilities ex-

hibiting suboptimal performance also being underrepresented in the data, suggesting that the scarcity of training instances may be a contributing factor to their diminished performance.

Furthermore, a comparative analysis in single facility, taking Table 1a for example, reveals that the neural network predictive model, as orchestrated in this study, surpasses both traditional machine learning approaches and rudimentary neural network configurations in terms of predictive efficacy. This means that the model successfully transcend those basic learning models [9] clearly. This model adeptly fulfills the objective of patient waiting time prediction, corroborating the initial hypothesis posited at the outset of the paper. The meticulously calibrated neural network architecture is adept at tackling scenarios characterized by a multitude of features with attenuated individual impacts.

Additionally, this study incorporates a feature selection endeavor to bolster the models' versatility and applicability. In this section, the findings post-feature selection are delineated and interpreted. It emerges that employing a curated subset of features, as per the methodologies previously outlined, for model training results in a quantifiable diminution of predictive efficacy in correlation with the reduced feature count. The fidelity of predictions is intricately linked to the cardinality of the features engaged in the training regimen. Although a robust feature set can maintain an elevated level of predictive performance even with a modest decline, the predictive outcomes become increasingly stochastic with a diminished feature set. This unpredictability is heavily dependent on the particular features selected, thereby not ensuring the precision of predictions in a universally applicable context.

5.2 Overload Detection

Peak Capture with Linear Regression Experiments are conducted by using a Linear Regression model across four different modalities. The results of these experiments are summarized in Table 2a. The table displays the Accuracy, Precision, and Recall for each modality, which reflects the model's capability in detecting peaks.

Facility	Accuracy	Precision	Recall
1	0.821	0.774	0.679
2	0.699	0.711	0.748
3	0.714	0.730	0.730
4	0.788	0.660	0.473

(a) Peak Capture Measurement (LR)

ML Model	Accuracy	Precision	Recall
Linear Regression	0.821	0.774	0.679
LR (best features)	0.779	0.696	0.602
Decision Tree	0.711	0.572	0.582
Random Forest	0.721	0.891	0.211
Neural Network	0.785	0.798	0.486

(b) Comparison of Different Models

Table 2: Performance Metrics for Facilities and ML Models

The results indicate that Modality 1 achieved the highest accuracy (0.820) and precision (0.755), making it the most reliable model for predicting peak conditions in this context. However, Modality 2 shows the highest recall (0.748), suggesting that it is better at identifying all peak occurrences, albeit with a slightly lower precision. These findings provide valuable insights for selecting the appropriate model based on the specific requirements of peak detection and overload management in hospital settings.

Meanwhile, other methods such as decision tree, random forest or Neural Network has an

accuracy of 70%-80%, lower than the LR model, taking Facility-1 as an example. The result shows that although NN can predict patient waiting time quite well, the Linear Regression model is more suitable for Peak Capturing Task. To summarize, can use the NN can be used to make waiting time prediction for the patients, while LR model is used as the Overload Detector presented to the hospital manager.

Time Peak Indicating Overload When the TN, FP, FN, TP are defined in Section 4.3.3. Peak of the waiting time representing Predicted-Overload, while True-Overload is defined as large amount of patients delayed in line. The confusion matrices for tables F1 and F3 are as follows:

$$\mathbf{CM}_{F1} = \begin{pmatrix} \text{TN} & \text{FP} \\ \text{FN} & \text{TP} \end{pmatrix} = \begin{pmatrix} 29258 & 13087 \\ 88 & 333 \end{pmatrix}, \quad \mathbf{CM}_{F2} = \begin{pmatrix} \text{TN} & \text{FP} \\ \text{FN} & \text{TP} \end{pmatrix} = \begin{pmatrix} 11326 & 11359 \\ 168 & 730 \end{pmatrix}$$

The corresponding evaluation metrics of F1 and F3 are summarized in Table 3. (There's no necessary for F2's or F4's overload to be measured because they are running well.)

Metric	F1	F3
Accuracy	0.6919	0.5112
Precision	0.0248	0.0604
Recall	0.7910	0.8129

Table 3: Evaluation Metrics for Tables F1 and F3

From the confusion matrices and evaluation metrics, it is evident that the recall rates for both F1 and F3 are relatively high, at 0.7910 and 0.8129, respectively. This indicates that the *peak* variable (P) is effective in identifying most of the cases where the *overload* variable (O) is true. In other words, when the waiting time exceeds 10 minutes (P=1), it successfully identifies the majority of situations where the number of people waiting exceeds 5 (O=1). While the *peak* variable can be used as a reliable indicator for *overload* with a high recall rate, the model's low precision indicates that additional factors should be considered to reduce the false positive rate and improve overall prediction accuracy. To summarize, long patient waiting time can be used to detect the department's overload, and the high recall rate shows that the detection method is of high sensitivity.

By focusing on the high recall for overload detection and analyzing the correlation between overload and peak predictions, this approach not only aims to detect critical conditions within the healthcare system but also strives to provide a model that offers a robust and accurate reflection of the system's current state. This ensures that the healthcare system can respond proactively to potential overloads, ultimately improving patient outcomes and operational efficiency.

5.3 High Generalization Performance

After feature selection, the filtered features are then integrated into our model, beginning with a subset of 5 features and incrementally increasing the count by 10 with each subsequent test. The findings indicated that as the number of features expanded, the Mean Absolute Error (MAE) of the

model's predictions decreased, with minimal changes in the mutation value, the model's performance exhibits a gradual improvement (in Figure 2). Notably, this model can significantly reduce the required number of features without a marked decline in accuracy. This attribute is particularly advantageous when dealing with incomplete datasets or different scenarios, as the architecture can still train an effective patient waiting time prediction model using a subset of the original features.

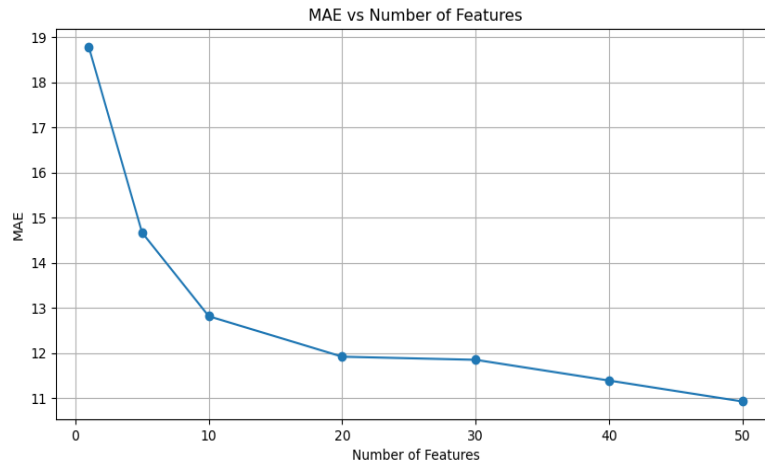


Fig. 2. MAE vs the number of features

By selecting five dimensions representing the current state of the healthcare system, the model maintains a high prediction accuracy. A neural network architecture was constructed comprising five integrated sub-networks, which achieved an accuracy level with a Mean Absolute Error (MAE) of 8.778, compared to the most accurate MAE of 7.617 (seen in Tables 1a, tested for Facility-1 as an example). This performance level remains notably high.

A systematic approach was used to improve the predictive power and understandability of the model by carefully differentiating the influence of each group of features. Incorporate expertise into models by grouping features and designing specialized subnetworks. The feature selection method effectively predicts patient waiting times even with a subset of features, especially when the available data is less complete than the features in the dataset, and detects systematic trends across different healthcare settings.

The model is powerful because the designed sub-neural networks demonstrate good ability to adapt to new situations based on different feature sets and medical knowledge. Through feature selection and clustering processing, an algorithm that predicts stable latency even when the amount of data is not large is obtained. The model optimizes emergency department resource allocation and maintains high accuracy by using key feature sets and separating them. This improves the understanding of the model and makes it a useful tool for healthcare professionals.

6 Conclusions

In this study, the challenge of predicting patient waiting times was addressed by developing a neural network. The method significantly improves the prediction accuracy, which is better than previous models. Based on this, the predictions can be presented to patients in the emergency room to reduce their waiting anxiety. An overload detection model based on linear regression was developed and examined its sensitivity. In addition, A feature selection process was conducted to identify key attributes of different diagnostic devices in the hospital environment. The model remains the similar accuracy when a subset of most important feathers are reserved. This process not only highlights the importance of these features, but also validates the predictive ability of our model at different scales, thus demonstrating its generalisation performance and usefulness in patient waiting time prediction.

This paper, evidently, carries profound practical significance and real-world applicability. The study presents a predictive model based on neural networks that offers accurate waiting time predictions. The predictive capabilities of our model can be seamlessly integrated into the emergency departments of hospitals. Furthermore, the use of the overload detector can help hospital managers allocate medical resources more efficiently. By learning from the relevant data of these medical institutions, the model can adeptly fulfill predictive tasks, thereby enhancing the allocation and co-ordination of medical systems across various hospitals, offering substantial assistance.

The findings contribute to the body of knowledge in the field of health informatics, offering insights into improving patient flow and resource allocation within healthcare facilities. The robustness of the model, as evidenced by its performance across different scenarios, underscores its potential for real-world application. As looking to the future, this work lays the groundwork for further exploration into optimizing patient experience and operational efficiency in healthcare settings.

However, it is important to acknowledge that this study has certain limitations. For instance, this approach primarily focused on refining simple neural network architectures, without incorporating a wide variety of complex neural network models. Additionally, the dataset that was utilized may not be extensive, as it was specific to one medical system, which could potentially lead to inaccuracies in prediction under certain special circumstances.

The insights gleaned from this research open avenues for future work. For instance, there is potential in experimenting with diverse feature combinations and model architectures to uncover more accurate and effective predictive methodologies. Moreover, the application of predictive outcomes presents numerous opportunities for refinement. While the current predictions are confined to the present moment, facilitating short-term adjustments and management within the hospital's medical system, there is scope to incorporate broader temporal references. This could enable more long-term, strategic forecasting and regulation of the healthcare system.

References

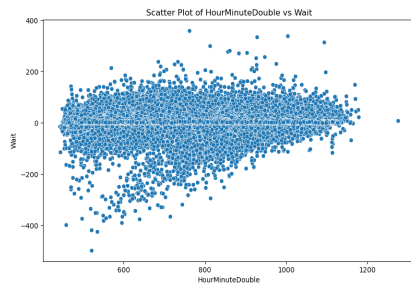
- [1] Hijry H, Olawoyin R. Predicting patient waiting time in the queue system using deep learning algorithms in the emergency room. *International Journal of Industrial Engineering*. 2021;3(1):33-45.

- [2] Shafaf N, Malek H. Applications of machine learning approaches in emergency medicine; a review article. *Archives of academic emergency medicine*. 2019;7(1).
- [3] Heart T, Ben-Assuli O, Shabtai I. A review of PHR, EMR and EHR integration: A more personalized healthcare and public health policy. *Health Policy and Technology*. 2017;6(1):20-5.
- [4] Dolin RH, Alschuler L, Boyer S, Beebe C, Behlen FM, Biron PV, et al. HL7 clinical document architecture, release 2. *Journal of the American Medical Informatics Association*. 2006;13(1):30-9.
- [5] Ayaz M, Pasha MF, Alzahrani MY, Budiarto R, Stiawan D. The Fast Health Interoperability Resources (FHIR) standard: systematic literature review of implementations, applications, challenges and opportunities. *JMIR medical informatics*. 2021;9(7):e21929.
- [6] Bidgood Jr WD, Horii SC, Prior FW, Van Syckle DE. Understanding and using DICOM, the data interchange standard for biomedical imaging. *Journal of the American Medical Informatics Association*. 1997;4(3):199-212.
- [7] Curtis C, Liu C, Bollerman TJ, Panykh OS. Machine learning for predicting patient wait times and appointment delays. *Journal of the American College of Radiology*. 2018;15(9):1310-6.
- [8] Nelson A, Herron D, Rees G, Nachev P. Predicting scheduled hospital attendance with artificial intelligence. *NPJ digital medicine*. 2019;2(1):26.
- [9] Panykh OS, Guitron S, Parke D, Zhang C, Pandharipande P, Brink J, et al. Improving healthcare operations management with machine learning. *Nature Machine Intelligence*. 2020;2(5):266-73.
- [10] Pattnayak P, Mohanty A, Das T, Patnaik S. Deep Learning Based Patient Queue Time Forecasting in the Emergency Room. In: 2023 International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS). IEEE; 2023. p. 541-5.
- [11] Kyritsis AI, Deriaz M. A machine learning approach to waiting time prediction in queueing scenarios. In: 2019 Second International Conference on Artificial Intelligence for Industries (AI4I). IEEE; 2019. p. 17-21.
- [12] Kuo YH, Chan NB, Leung JM, Meng H, So AMC, Tsoi KK, et al. An integrated approach of machine learning and systems thinking for waiting time prediction in an emergency department. *International journal of medical informatics*. 2020;139:104143.
- [13] Pak A, Gannon B, Staib A. Predicting waiting time to treatment for emergency department patients. *International Journal of Medical Informatics*. 2021;145:104303.
- [14] Al-Mousa A, Al-Zubaidi H, Al-Dweik M. A machine learning-based approach for wait-time estimation in healthcare facilities with multi-stage queues. *IET Smart Cities*. 2024.
- [15] Zhao Y, Peng Q, Strome T, Weldon E, Zhang M, Chochinov A. Bottleneck detection for improvement of emergency department efficiency. *Business Process Management Journal*. 2015;21(3):564-85.
- [16] Joseph JW, Leventhal EL, Grossestreuer AV, Chen PC, White BA, Nathanson LA, et al. Machine learning methods for predicting patient-level emergency department workload. *The Journal of Emergency Medicine*. 2023;64(1):83-92.

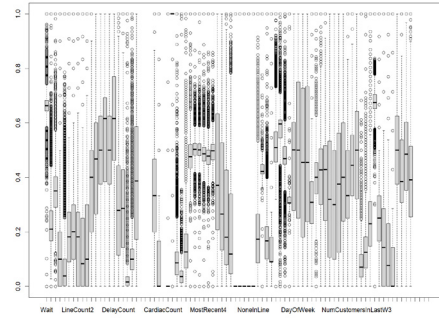
- [17] Eshghali M, Kannan D, Salmanzadeh-Meydani N, Esmayeeli Sikaroudi AM. Machine learning based integrated scheduling and rescheduling for elective and emergency patients in the operating theatre. *Annals of Operations Research*. 2024;332(1):989-1012.
- [18] Ala A, Alsaadi FE, Ahmadi M, Mirjalili S. Optimization of an appointment scheduling problem for healthcare systems based on the quality of fairness service using whale optimization algorithm and NSGA-II. *Scientific Reports*. 2021;11(1):19816.
- [19] Shi Y, Mahdian S, Blanchet J, Glynn P, Shin AY, Scheinker D. Surgical scheduling via optimization and machine learning with long-tailed data. *arXiv preprint arXiv:220206383*. 2022.
- [20] Wang Y, Zhang Y, Zhou M, Tang J. Feature-driven robust surgery scheduling. *Production and Operations Management*. 2023;32(6):1921-38.
- [21] Medical Analytics Group of Massachusetts General Hospital. Bringing Data-Driven Improvements to Healthcare; 2024. <https://medicalanalytics.group/>.
- [22] Medical Analytics Group of Massachusetts General Hospital. Challenging Problems and Dataset; 2024. <https://medicalanalytics.group/operational-data-challenge/>.
- [23] Tang KJW, Ang CKE, Constantinides T, Rajinikanth V, Acharya UR, Cheong KH. Artificial Intelligence and Machine Learning in Emergency Medicine. *Biocybernetics and Biomedical Engineering*. 2021;41(1):156-72.

Appendix

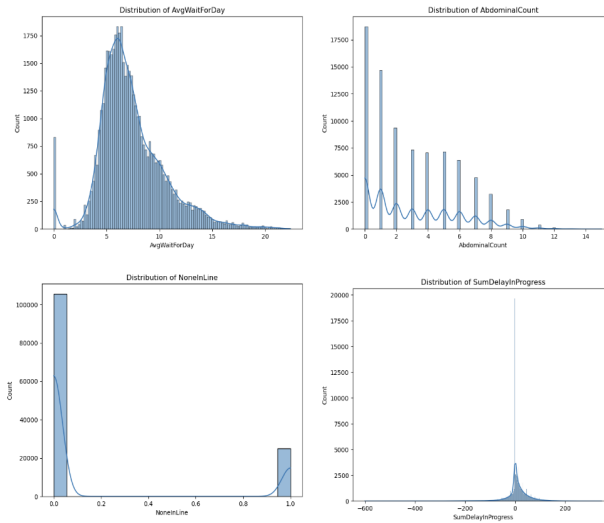
A-1. EDA Visualization



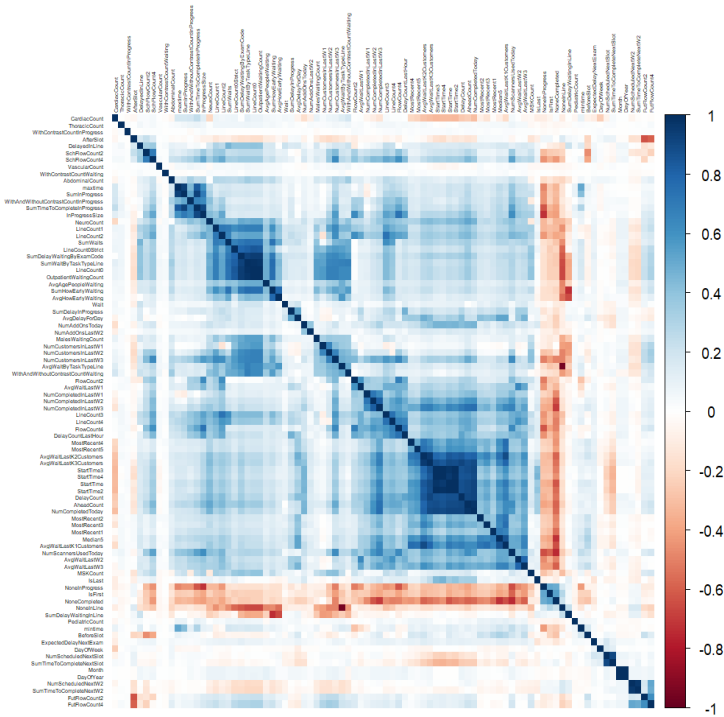
(a) Distribution of waiting time



(b) Boxplot of the dataset in Worksheet1



(c) Distribution of features



(a) Correlation between each variable in Worksheet1

Fig. A-1. EDA Visualization figures

A-2. PCA Results

Index	Variable Name	Absolute Load Value	Index	Variable Name	Absolute Load Value
1	NumCustomersInLastw1	6.382846	26	DelayCount	5.285572
2	FlowCount2	6.094908	27	Linecount0strict	5.274416
3	LineCount4	6.055510	28	IsFirst	5.272488
4	NumCompletedInLastw1	5.938251	29	SchFlowCount2	5.271732
5	AvgWaitLastw3	5.932923	30	NoneCompleted	5.255597
6	maxtime	5.873289	31	mintime	5.236939
7	LineCount2	5.856698	32	AvgWaitLastK3customers	5.236619
8	AvgDelayForDay	5.830229	33	AbdominalCount	5.222655
9	SumDelayInProgress	5.815193	34	IsLast	5.191561
10	NumCompletedInLastw3	5.778408	35	MostRecent1	5.157611
11	NumCustomersInLastw2	5.771374	36	SumTimeToCompleteNextW2	5.152805
12	NoneInProgress	5.714588	37	NumCustomersInLastw3	5.150708
13	Vascularcount	5.686995	38	AvgwaitLastk2customers	5.121899
14	LineCount1	5.648676	39	DelayedInLine	5.103083
15	Sumwaits	5.624718	40	Afterslot	5.067978
16	SumDelayWaitingInLine	5.605646	41	Aheadcount	5.061976
17	InProgresssize	5.524258	42	AvgwaitLastw2	5.046041
18	Median5	5.503731	43	FutFlowCount2	5.044763
19	DelayCountLastHour	5.500591	44	StartTime	5.037080
20	NumAddonsToday	5.493103	45	AvgHowEarlyWaiting	5.015407
21	AvgwaitLastw1	5.433343	46	FlowCount4	4.993000
22	sumInProgress	5.406005	47	SumTimeToCompleteNextslot	4.974212
23	Beforeslot	5.377051	48	MostRecent	4.915264
24	NumAddonsLastw2	5.310590	49	NumScheduledNextw2	4.842634
25	NumCompletedInLastw2	5.290821	50	NumScheduledNextslot	4.818803

Table A-2: Principal Component Analysis (PCA) Result of Worksheet1 Table

A Method Integrating RRT and A-Star Algorithms to Enhance Obstacle Navigation and Optimization

Shichao Yin

Information Department, Shanghai Waigaoqiao Shipbuilding Co., Ltd, Shanghai, China

563479882@qq.com

Abstract. This paper addresses the issue of path planning for robots in complex environments by proposing a hybrid path planning method that integrates the Rapidly-Exploring Random Trees Algorithm(RRT) and the A-STAR Search Algorithm(A-Star). The method leverages the rapid exploration capability of the RRT algorithm to generate an initial path, and combines it with the heuristic strategy of the A-Star algorithm to optimize the path, particularly in the areas near obstacles. Experimental results show that, compared to using RRT or A-Star algorithms alone, the proposed hybrid algorithm can generate shorter, smoother, and near-optimal paths while maintaining planning efficiency. Specifically, the hybrid algorithm demonstrates good performance in terms of path length, computation time, and path smoothness. Future work will focus on further optimizing the algorithm to enhance its adaptability in more complex environments and considering its application in dynamic obstacle environments.

Keywords: Hybrid Path Planning Algorithm, Path Optimization, RRT, A-STAR.

1 Introduction

In recent years, with the advancement of robotics, particularly in the fields of autonomous vehicles, drones, and industrial robots, the study of path planning algorithms has become increasingly important. Path planning algorithms assist robots in finding a safe path from a starting point to a target point in unknown or partially known environments. Among these algorithms, RRT and the A-Star are the two most commonly used methods, although RRT is favored for its ability to effectively explore high-dimensional spaces and find solutions in complex environments [1], it suffers from low search efficiency and non-optimal paths [2]. On the other hand, the A-Star algorithm is renowned for its capability to efficiently find optimal paths, but its applicability is mainly limited to low-dimensional grid environments and it tends to become trapped in local optima [3].

To overcome these limitations, various improvements have been proposed by scholars. For example, in [4], the authors introduced an improved RRT algorithm, which guarantees asymptotic optimality of the path, although it is more complex to implement and has a slower convergence rate. Additionally, [5] presented a hybrid approach that combines the artificial potential field method with the A-Star algorithm, aiming to enhance path smoothness and avoid obstacles. However, this method still faces limitations in complex environments, especially when dealing with dense obstacles.

Another improvement approach is to hybridize different types of algorithms to compensate for the deficiencies of a single algorithm. In [6], a hybrid algorithm combining RRT and A-Star was proposed, utilizing the heuristic information from A-Star to guide RRT's exploration, thereby improving search efficiency and path quality. Although these methods have made some progress, they still face challenges in practical applications, such as high computational costs and lack of path smoothness.

In light of this, the paper proposes a new method that integrates RRT and the A-Star algorithms, aiming to combine their strengths to address the issues in existing algorithms. Specifically, this method leverages RRT's rapid exploration capabilities while incorporating A-Star's heuristic strategies to optimize path selection, with the goal of obtaining shorter, smoother, and safer paths. This method not only effectively avoids obstacles but also significantly reduces the path planning time.

2 Algorithm Principles

2.1 RRT

RRT is a path planning algorithm that efficiently explores complex, high-dimensional spaces by randomly sampling points within the space, connecting each new point to the nearest node in the tree, and incrementally building a tree structure that extends towards the goal. The specific steps of the algorithm are as follows.

(1) Initialization. As shown in figure 1, set the start and end points, and create a tree that initially contains only the start point.

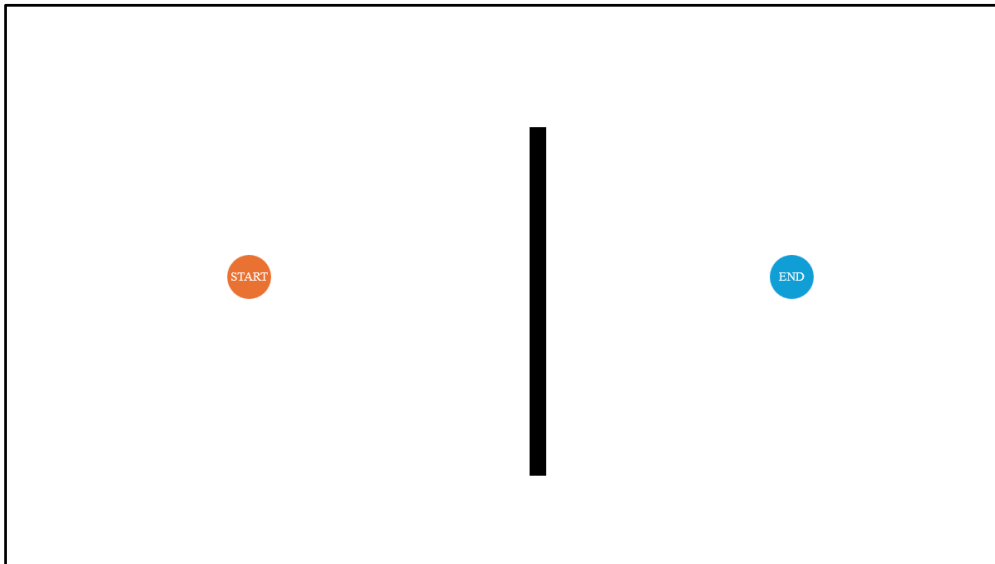


Fig. 1. Setting Start and End Points.

(2) Random Sampling. Randomly sample a point within the configuration space, as illustrated by the green point in figure 2.

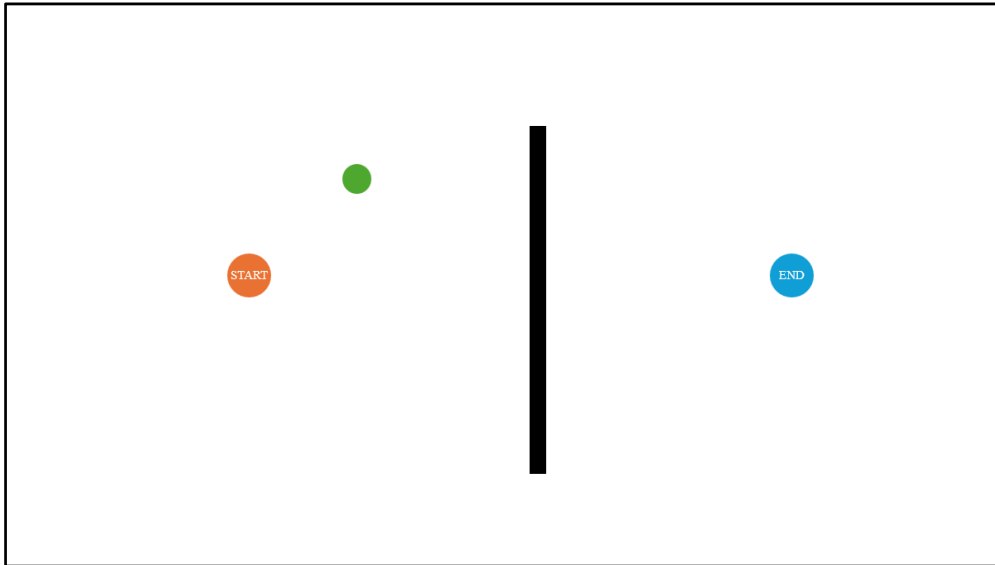


Fig. 2. Randomly Sampling a Point.

(3) Selecting the Nearest Node. Identify the node in the tree that is closest to the sampled point. As shown in figure 3, the tree currently contains only the start point.

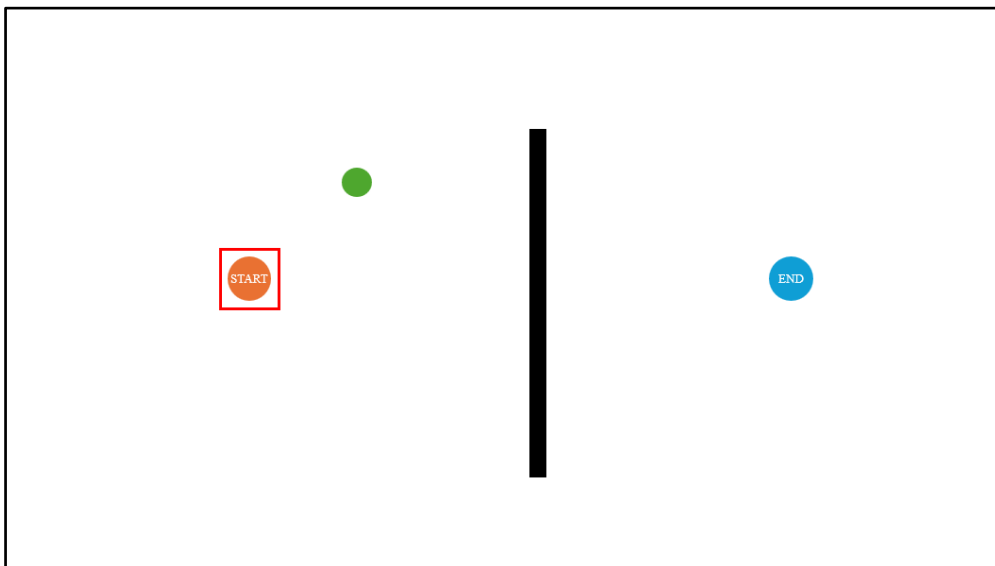


Fig. 3. Selecting the Nearest Node.

(4) Expanding the Tree. From the nearest node, extend the tree towards the sampled point by a certain step size, generating a new node, as depicted in figure 4, where the dashed circle represents the step size range.

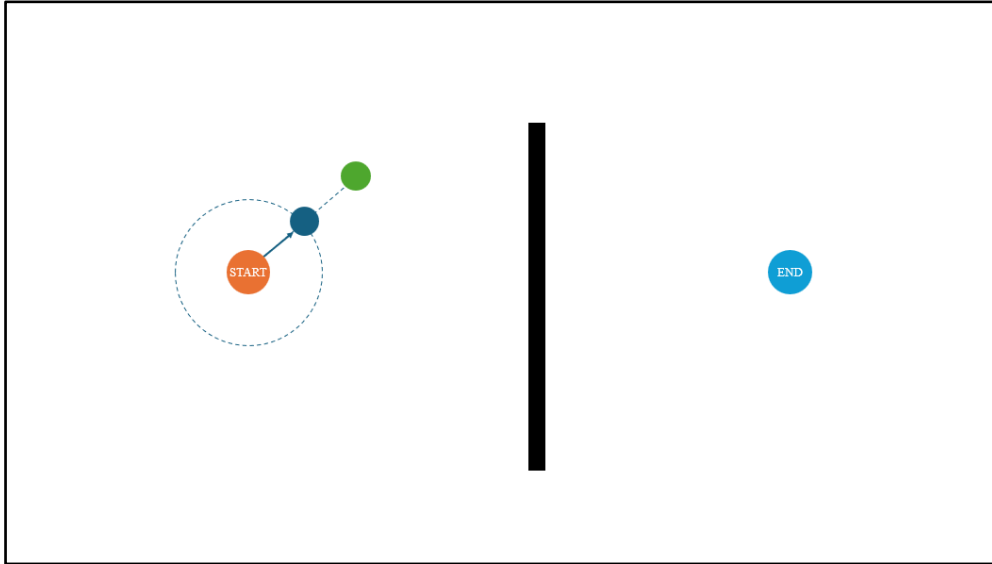


Fig. 4. Expanding the Tree Towards the Sampled Point.

(5) Collision Detection. Check whether the path between the new node and the nearest node collides with any obstacles. As shown in figure 5, if the newly generated path collides with an obstacle, the path is discarded. If there is no collision, add the new node to the tree.

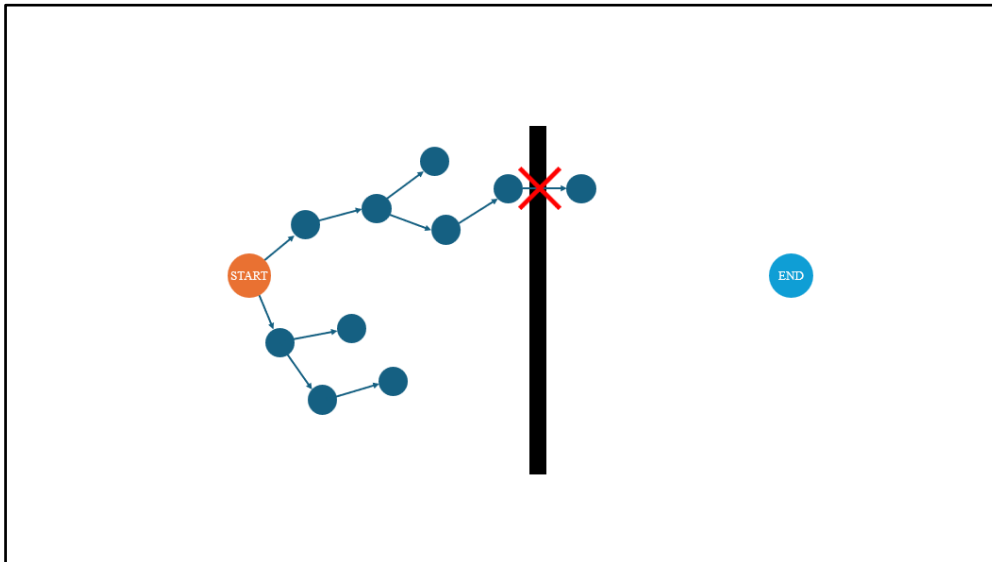


Fig. 5. Checking for Obstacles.

(6) Checking Goal Achievement. If the new node is close enough to the goal, stop the algorithm and return the path, as illustrated in figure 6, where a complete path has been generated. If not, repeat steps 2 through 5 until either the maximum number of iterations is reached or a path is found.

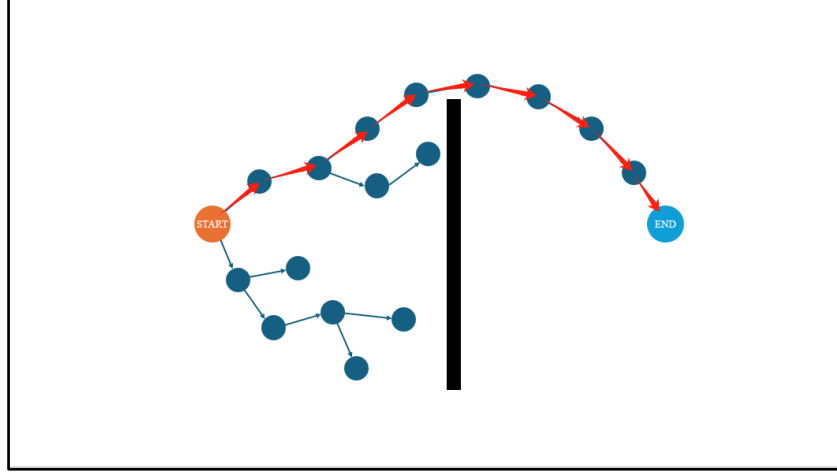


Fig. 6. Expanding the Tree to Reach the Goal.

2.2 A-Star

The A-Star algorithm is a shortest path search algorithm based on heuristic information. Its core concept is to prioritize the expansion of paths that are most likely to reach the goal by combining actual costs with estimated costs until the shortest path is found.

Explanation of the Cost Function. Details are as follows.

(1) Actual Cost $g_{neighbor}$

This represents the actual cost of reaching the neighbor node from the start node, calculated as shown in equation (1).

$$g_{neighbor} = g_{current} + d(current, neighbor) \quad (1)$$

Where.

$g_{current}$ is the actual cost of the parent node current of the newly added node neighbor.

$d(current, neighbor)$ is the distance between the newly added node neighbor and its parent node current, i.e., the step size.

(2) Heuristic Cost $h_{neighbor}$

This represents the estimated cost from the neighbor node to the goal node, calculated as shown in equation (2).

$$h_{neighbor} = heuristic(neighbor, goal) \quad (2)$$

Where.

$heuristic(neighbor, goal)$ is the estimated cost function from the newly added node neighbor to the goal node goal. This is typically calculated using the Euclidean distance or Manhattan distance, such as.

$$heuristic(neighbor, goal) = [(X_{goal} - X_{neighbor})^2 - (Y_{goal} - Y_{neighbor})^2]^{1/2};$$

(3) Total Cost Function $f_{neighbor}$.

The comprehensive cost x of node neighbor is calculated as shown in equation (3).

$$f_{neighbor} = g_{neighbor} + h_{neighbor} \quad (3)$$

Cost Function Usage. The algorithm iterates over the nodes based on the total cost function until the goal node is found. The detailed steps are as follows.

(1) Initialization. Define the start node start and the goal node goal. Initialize the open list open_set, and add the start node to it. Set the cost start of the start node to 0. Initialize the closed list close_set to store nodes that have already been processed.

(2) Node Expansion. Remove the node with the smallest cost, referred to as current (i.e., the node with the smallest $f_{current}$), from the open list and place it in the closed list. Check if current is the goal node; if it is, the algorithm terminates and returns the path. Otherwise, generate all neighboring nodes of current.

(3) Cost Calculation. For each neighboring node neighbor, calculate the actual cost $g_{neighbor}$ from the start point to that node, the heuristic cost $h_{neighbor}$, and the total cost $f_{neighbor}$. The detailed algorithm is provided in the formulas above.

(4) Node Processing. If neighbor is already in the closed list and the g value of the new path is greater, skip this node. If neighbor is not in the open list, add it and set its parent node to current, then record the g , h , and f values. If neighbor is already in the open list but the g value of the new path is smaller, update its parent node to current and update the g , h , and f values.

(5) Repeat Steps 2-4. Until the goal node goal is found or the open list is empty (indicating that no solution path was found).

2.3 Hybrid Algorithm

This paper presents a hybrid path planning algorithm that combines the strengths of RRT and the A-Star to overcome their respective limitations. Specifically, the algorithm leverages the exploration capability of RRT to generate an initial path, then employs the A-Star algorithm to optimize the path in areas near obstacles, and finally applies path smoothing techniques. The detailed principles and implementation process of this algorithm are as follows.

The algorithm begins by using the RRT method to generate a basic path, following the principles described previously. Next, for each path node, it checks whether the node is near an obstacle (i.e., whether there is an obstacle within a certain radius). For regions near obstacles, the algorithm replans the path between these nodes using the A-Star algorithm. The A-Star algorithm takes into account the actual cost from the current node to the goal node and a heuristic estimate, always choosing the path with the lowest cost. This approach allows for the identification of safer and smoother paths in areas near obstacles. Finally, the algorithm

performs path smoothing by checking if the line connecting adjacent nodes in the path collides with any obstacles. If there is no collision, these two nodes can be directly connected, thereby eliminating any unnecessary intermediate nodes.

3 Experimental Plan

3.1 Experimental Environment

This experiment was conducted on a Windows 11 (64-bit) operating system with 32GB of Random Access Memory and an Intel Core i5-12500H processor. The algorithm was implemented using Python 3.10, and the visualization was carried out using the pyplot module from the matplotlib library. The specific configurations of the map used in the experiment are as follows.

- The map configuration includes the size and shape of the grid, as well as the placement of obstacles that vary in size and shape to simulate a realistic environment;
- The simulation environment is a two-dimensional plane with a grid size of 100*100;
- Several obstacles were placed within the grid to simulate a complex environment. The layout of obstacles remained consistent across all experiments to facilitate the comparison of algorithm performance. The generated map is shown in figure 7.

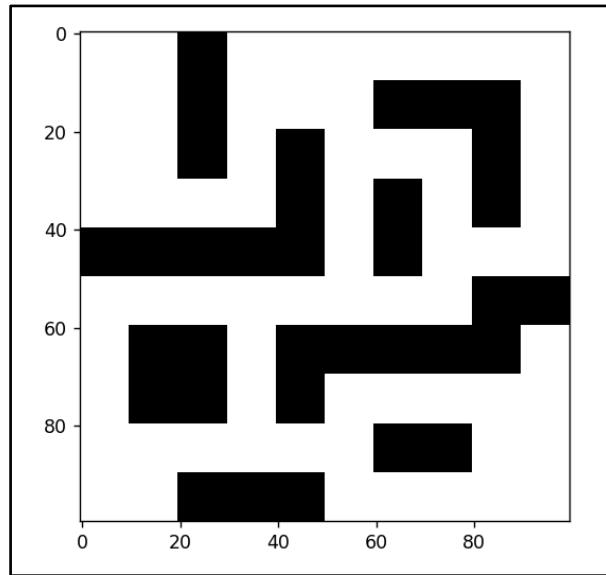


Fig. 7. Map Used in the Experiment.

3.2 Performance Metrics

Path length, path smoothness, and computation time were selected as the criteria to comprehensively evaluate the algorithm's performance across different aspects. These criteria each focus on different performance indicators and collectively influence the practical effectiveness and utility of path planning.

Average Path Length. Path length measures the total distance of the path from the start point to the goal. The reasons for choosing path length as a criterion include: efficiency, feasibility, and practical application:

- **Efficiency.** A shorter path usually indicates higher efficiency and lower resource consumption (such as energy and time). For mobile systems like robots or autonomous vehicles, a shorter path can save fuel or power and reduce wear and tear;
- **Feasibility.** In certain scenarios, a shorter path may be easier to achieve or safer, especially in resource-limited or complex environments;
- **Practical Application.** In real-world applications, the goal is often to find the shortest path from the start point to the goal, which can enhance the efficiency and effectiveness of task execution.

Number of Path Turns. The number of path turns measures the quantity and degree of turns along the path by counting the number of corners or sharp bends. The reasons for selecting the number of path turns as a criterion include:

- **Motion Comfort.** For passenger or cargo transport, a more comfortable experience, reducing discomfort caused by sharp turns;
- **Mechanical Constraints.** For certain robots or vehicles, frequent sharp turns may not align with the design requirements of the mechanical structure, and a smoother path can reduce stress and wear on the steering system;
- **Dynamic Response.** A smoother path is easier to predict and track, allowing control systems to maintain stability and reduce vibrations and oscillations during movement.

Average Computation Time. Computation time measures the duration from the start of the algorithm to the discovery of a feasible path. The reasons for choosing computation time as a criterion include:

- **Real-Time Performance.** In many applications, such as autonomous driving or robot navigation, path planning must be completed within a limited time frame to respond to dynamic environmental changes or emergencies;
- **Resource Constraints.** Computation time is directly related to the efficient use of computational resources. Shorter computation times indicate a more efficient algorithm, making it suitable for resource-constrained environments like embedded systems or low-power devices;
- **Practicality.** In practical operations, path planning algorithms need to not only find an effective path but also complete the task within a reasonable time, especially in dynamic environments where frequent replanning is required.

3.3 Implementation Methods

Average Path Length

(1) Single Path Length Calculation. In each run of the algorithm, calculate the distance between each pair of adjacent nodes in the path, then sum these distances to obtain the total length of the path. The specific calculation formula is. $\text{path_length} = \sum(\text{distance}(\text{path}[i],$

$\text{path}[i + 1])$), where $\text{distance}(\text{path}[i], \text{path}[i + 1])$ represents the Euclidean distance between the i th and $i+1$ th nodes.

(2) Total Path Length Accumulation. For multiple experiments, accumulate the path lengths calculated from each run to obtain the total path length for all experiments.

(3) Average Path Length Calculation. Divide the total path length by the number of experiments to obtain the average path length.

Number of Path Turns. The calculation of the number of turns is based on the change in the path vector at each node. If the angle between two consecutive path segments is less than 170 degrees (i.e., the vector's turn angle is greater than 10 degrees), it is considered a turn. The detailed calculation method is as follows.

(1) Define Vectors. For each segment in the path, define a vector from the previous node to the current node.

(2) Calculate Angles Between Vectors. Compute the angle between each pair of consecutive vectors. If the angle is less than 170 degrees (i.e., the deflection angle is greater than 10 degrees), count it as a turn.

(3) Count the Number of Turns. Traverse all nodes in the path, calculate and accumulate the number of turns.

(4) Calculate the Average Number of Turns. Divide the total number of turns by the number of experiments to obtain the average number of turns.

(3) Average Experiment Time

Average Experiment Time

(1) Single Experiment Time Measurement. For each run of the algorithm, record the start and end times of the algorithm and calculate the difference between them to obtain the time used for that experiment. The specific calculation method is: $\text{end_time} - \text{start_time}$, where start_time and end_time are the timestamps of the algorithm's start and end, respectively.

(2) Total Experiment Time Accumulation. For multiple experiments, accumulate the time recorded for each experiment to obtain the total time for all experiments.

(3) Average Experiment Time Calculation. Divide the total time by the number of experiments to obtain the average experiment time.

3.4 Experimental Procedure

Initialization. Set up the experimental environment, including the grid, obstacles, starting point, and destination.

Algorithm Execution. Run multiple experiments (set at 1,000 times) for each of the three algorithms (RRT, A-Star, and the hybrid algorithm). Record the path length, path smoothness, and runtime for each experiment.

Data Collection. Gather the result data from each experiment and calculate the average path length, path smoothness, and runtime.

Data Analysis. Compare the results of the three algorithms, analyzing the strengths and weaknesses of the hybrid algorithm across different metrics.

4 Experimental Results

The visual path diagram generated by the algorithm in the experiment is shown in figure 8 below:

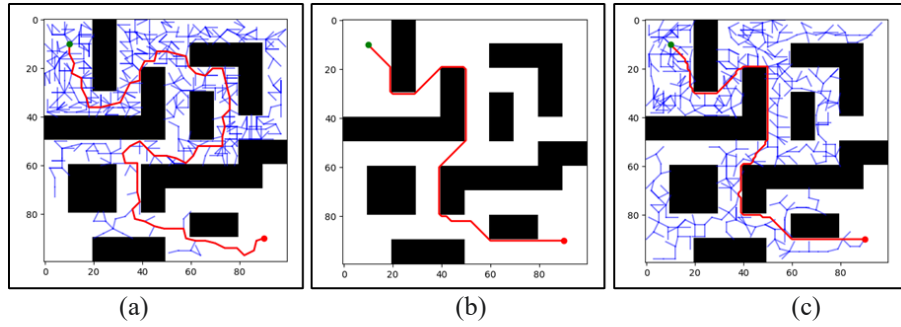


Fig. 8. Visual path diagrams of (a) RRT and (b) A-Star and (c) Hybrid Algorithm

Each experiment compares the performance of the algorithms by analyzing the average path length, average computation time, and average number of turns in the generated paths. The specific results are shown in figure 9 Experimental Results.

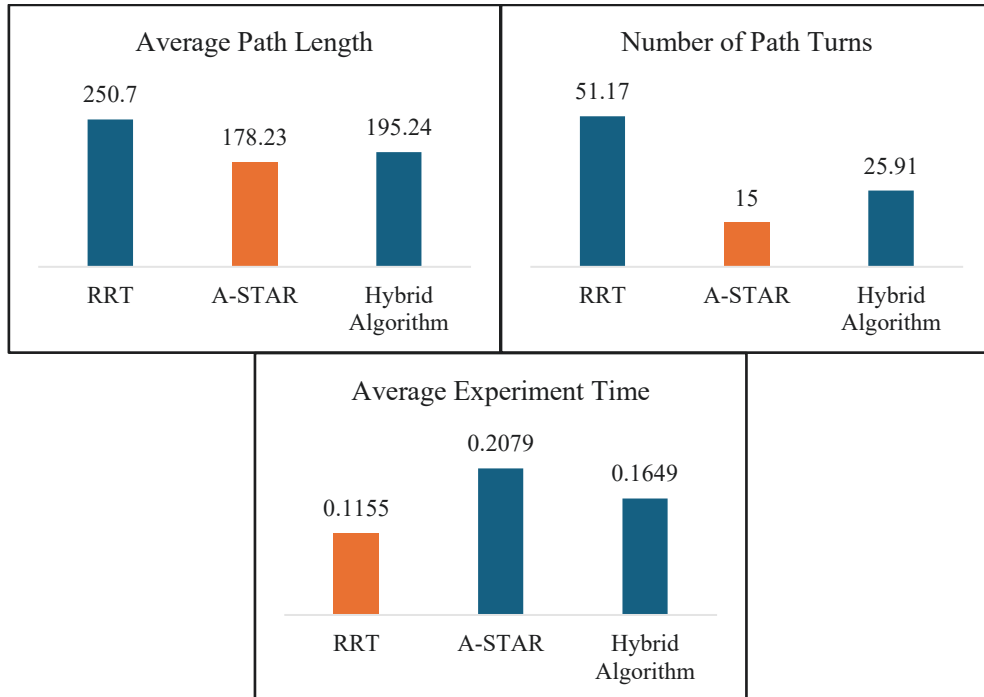


Fig. 9. Experimental Results.

Through the analysis of the path diagram and experimental data, the performance of the hybrid algorithm under different complex environments is demonstrated. The key areas of focus are.

- **Average Path Length.** The hybrid method generates a significantly shorter average path length compared to RRT. This indicates that the hybrid method can find shorter paths while avoiding obstacles, thus improving path optimality;
- **Number of Path Turns.** The hybrid method results in significantly fewer turns compared to RRT. This is because the hybrid method successfully reduces unnecessary turns through the use of the A-Star algorithm and path smoothing steps, resulting in a smoother path;
- **Average Experiment Time.** The hybrid method's average computation time is slightly longer than that of the RRT due to the path optimization using the A-Star algorithm near obstacle areas. However, this increase is acceptable, especially when greater path smoothness and optimality are required in practical applications.

The hybrid algorithm strikes a balance between path length and runtime, offering an optimal compromise. Compared to standalone RRT and A-Star algorithms, the hybrid algorithm produces paths that are slightly longer than A-Star but significantly shorter than RRT, indicating improvement in path optimization. Although its runtime is not as fast as RRT, it is still better than A-Star, demonstrating good computational efficiency. Finally, the hybrid algorithm's path smoothness is between the two, with the number of turns being fewer than RRT but more than A-Star. Overall, the algorithm achieves a favorable trade-off among path length, computation time, and path smoothness.

5 Conclusion

This study proposes a hybrid path planning method that combines RRT and A-Star algorithms to overcome the limitations of each when used independently. The RRT algorithm excels at rapidly exploring large search spaces, while the A-Star algorithm is renowned for its optimal pathfinding capability. By integrating these two approaches, the hybrid method not only improves search efficiency but also generates smoother paths, avoiding lengthy routes and frequent turns. Future research should focus on further optimizing the algorithm to enhance its adaptability in more complex environments and extending its application to dynamic obstacle environments. This proposed hybrid method holds promise for widespread application in fields such as autonomous driving and robot navigation.

References

- [1] ABDEL-RAHMAN, Ahmed S., et al. Enhanced hybrid path planning algorithm based on apf and A-Star. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2023, 48. 867-873.
- [2] WANG, Dong, et al. Path Planning Based on the Improved RRT* Algorithm for the Mining Truck. *Computers, Materials & Continua*, 2022, 71.2.
- [3] ZHANG, Jing, et al. Autonomous land vehicle path planning algorithm based on improved heuristic function of A-Star. *International Journal of Advanced Robotic Systems*, 2021, 18.5. 17298814211042730.

- [4] NASIR, Jauwairia, et al. RRT*-SMART. A rapid convergence implementation of RRT. *International Journal of Advanced Robotic Systems*, 2013, 10.7. 299.
- [5] ABDEL-RAHMAN, Ahmed S., et al. Enhanced hybrid path planning algorithm based on apf and A-Star. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2023, 48. 867-873.
- [6] AL-ANSARRY, Suhaib; AL-DARRAJI, Salah. Hybrid RRT-A*. An Improved Path Planning Method for an Autonomous Mobile Robots. *Iraqi Journal for Electrical & Electronic Engineering*, 2021, 17.1.

Drug delivery route optimization with a capacity based on the ALNS algorithm

Chuyao Ji

School of Applied Mathematics, Stony Brook University, New York, 11794, USA

jichuyao20030518@163.com

Abstract. This paper established a vehicle routing problem (VRP) model with capacity limitation to solve the drug delivery route optimisation problem using an adaptive large-scale neighbourhood search algorithm. The research aims to match orders to riders based on information such as merchant location, rider location, customer location, order remaining time, and rider load to minimise the total delivery distance while ensuring that each order is delivered on time. The model includes multiple objective functions and constraints, such as travel cost and performance cost, as well as time Windows and load capacity limits. By designing three kinds of damage operators (random damage, worst damage and correlation damage) and three kinds of repair operators (greedy repair, regret repair and random repair), and applying the acceptance criteria of the simulated annealing algorithm, the solution process is optimised. The experimental results show that the ALNS algorithm can effectively solve the optimal path scheme after several iterations, significantly reduce the total distribution distance and time, and improve distribution efficiency. The results of this study have important reference value to the actual drug delivery system, which helps improve the rate of patient treatment and the timeliness of drug delivery. In this paper, detailed experiments and data analysis verify the proposed algorithm's effectiveness and feasibility.

Keywords: ALNS, VRP problem, pharmacy distribution, NP-hard problem.

1 Introduction

Dantig et al [1]. proposed the VRP problem for the first time in 1959. They studied the mathematical model and algorithm used to solve the problem, and began to apply it to the enterprise practice. On the basis of Dantig, Clark et al [2]. improved the solution quality by proposing a saving algorithm. With the publication of these two articles, more and more scholars began to study different kinds of VRP problems, hoping to find more solving models and algorithms. Lenstra et al [3]. analyzed the complexity of VRP problems and pointed out that all VRP problems are NP-hard problems. When solving the Vehicle Routing Problem with Time Windows (VRPTW), in addition to satisfying the constraints of the basic VRP problem, it is necessary to consider the constraints of different time Windows of different customers. According to different time window constraints, it is necessary to construct the optimization objective function related to time window in the optimization process, which greatly increases the complexity of problem solving. Gayialis et al [4]. used the Large Neighborhood Search (LNS) algorithm to solve the VRPTW problem with the minimum number of vehicles and the total driving distance as the objective function. Corstiens et al [5]. added multivariate statistical

analysis to the large-scale neighborhood search algorithm to solve the vehicle routing problem with time window. The results of this study have important reference value to the actual drug delivery system, which is helpful to improve the rate of patient treatment and the timeliness of drug delivery.

2 Mathematical Model and Algorithm

2.1 The Problem Described in the Mathematical Model

Problem Description. After the customer places the order, the order and the rider are matched according to the merchant location (i.e. pick-up point), rider location, customer location (i.e. delivery point), the remaining time of the customer order, and the load of the rider, etc., and the total delivery distance is shortened as much as possible on the basis of ensuring that each order can be delivered on time.

Three distribution centers (pharmacies), are Yuxin, Yifeng, Good medicine, 20 customer points, set up 12 riders but not necessarily all [6]. There is corresponding coordinate information in Excel, where the serial number behind the pharmacy is just convenient to correspond with the customer point, such as Yifeng 1 corresponds to the customer point 1, Good medicine 17 corresponds to the customer point 17. Because it is the first take and then send, the demand of the pharmacy is set to a positive number, and the demand of the corresponding customer point is set to a negative number. The coordinates of the rider are the initial position of the rider, and the rider does not need to return to the initial position after the completion of the delivery task. Set a constraint for delivery within 30 minutes (u is 30 minutes) so there is no time window [7].

Because the coordinates can only calculate the spherical distance, the actual driving will be detoured afterwards, and the distance is longer, so the distance needs to be multiplied by 2 based on the spherical distance.

Assumptions. Assume that the rider's electric vehicle is an electric vehicle of the same model, and all parameters such as electric vehicle capacity, maximum mileage, power consumption and other parameters are the same; It is assumed that electric vehicles all travel at the same speed and have the same speed. Assume that the electric vehicle power is sufficient, do not consider the case of power depletion; It is assumed that the electric vehicle has no maximum driving range constraint; Assume that there are no special goods, such as the volume of a single cargo exceeds the maximum capacity of the electric vehicle; Assume that the customer receives the goods no matter when the rider delivers them; Assume that the service time of the rider's delivery at the customer's point is the same; The dispensing time of the drugstore merchant and the pick-up time of the rider at the drugstore are not considered; Do not consider the pick-up point, that is, the drugstore is out of stock, etc.; Do not consider the weather, traffic and other uncontrollable factors; The coordinates of nodes are represented by latitude and longitude, and the distance between nodes can be calculated by formula (1) :

$$D = 2R \times \sin^{-1} \sqrt{\sin^2 \frac{W_1 - W_2}{2} + \cos W_1 \times \cos W_2 \times \sin^2 \frac{J_1 - J_2}{2}} \quad (1)$$

(Where R is the radius of the Earth, W is the latitude, and J is the longitude)

Parameters and Symbols. B : Drugstore collection, that is, collection of all pickup points,

$$B = \{1, 2, 3, \dots, i, \dots, n\}$$

C : Customer set, that is, all delivery points set, $C = \{n + 1, n + 2, \dots, n + i, \dots, 2n\}$;

$N = B \cup C$, indicating the collection of all pick-up and delivery points;

K : set of riders, $K = \{1, 2, 3, \dots, m\}$;

h_k : initial position node of rider k , $h_k = 2n + k, \forall k \in K$;

γ_k : virtual end distribution node of rider k , $\gamma_k = 2n + m + k, \forall k \in K$;

$M_k^1 = N \cup \{h_k\}, \forall k \in K$;

$M_k^2 = N \cup \{\gamma_k\}, \forall k \in K$;

$M_k = N \cup \{h_k, \gamma_k\}$, denotes the set of all nodes that rider k can access;

Parameters:

a_k : indicates the maximum load volume of rider k ;

α_i : represents the service time spent by the rider to pick up the delivery at node i ;

u : All orders need to be delivered within the set time u ;

d_{ij} : the distance between nodes i and j ;

t_{ij} : the time from node i to node j ;

q_i : The volume of the drug-loaded and unloaded by the rider at node i is positive at the pick-up point [8] and negative at the delivery point;

Decision variables:

x_{ijk} : Rider k takes 1 from node i to node j , otherwise takes 0;

y_{lk} : Take 1 when the l order is delivered by rider k , otherwise take 0;

Other derived variables:

b_{ik} : the time when rider k arrives at node i and begins service;

Q_{ik} : the load of rider k when he reaches node I ;

Model. Driving cost: The travel cost is related to the delivery distance, and the delivery distance is determined by both the order assigned to the rider and the order in which the rider serves the order. To some extent, the driving cost also represents the driving distance, and the smaller the driving cost, the shorter the total distribution path, that is, the shortest distribution time. The cost of travel is shown in formula (2), where c is the cost per unit distance traveled by the rider.

$$f_1 = \sum_{k \in K} \sum_{i,j \in M_k^1, \forall k \in K} c \times d_{ij} \times x_{ijk} \quad (2)$$

Performance cost: For human reasons, many platforms offer performance pay to encourage riders to fulfill as many orders as possible. Since the research purpose of this paper is only related to the delivery distance and has nothing to do with the number of orders completed by the rider, this paper only considers the performance cost based on the delivery distance. The smaller the performance cost, the shorter the total distribution path. The performance cost is shown in formula (3), where v is the rider's performance per unit distance traveled.

$$f_2 = \sum_{k \in K} \sum_{i,j \in M_k^1, \forall k \in K} v \times d_{ij} \times x_{ijk} \quad (3)$$

Objective function: The objective function of this paper is the minimum total platform cost, as shown in formula (4). The objective function consists of two parts, where f_1 represents the driving cost of all riders and f_2 represents the performance cost of all riders.

$$\min f_1 + f_2 \quad (4)$$

Constraints:

$$\sum_{k \in K} y_{lk} = 1, \forall l \in B \quad (5)$$

$$x_{ilk} \leq y_{lk}, \forall l \in B, \forall i \neq l \in M_k^1, \forall k \in K \quad (6)$$

$$x_{i,l+n,k} \leq y_{lk}, \forall l \in B, \forall i \neq l+n \in M_k^1, \forall k \in K \quad (7)$$

$$\sum_{k \in K} \sum_{i \neq j \in M_k^1, \forall k \in K} x_{ijk} = 1, \forall j \in B \quad (8)$$

$$\sum_{i \neq j \in M_k^1} x_{ijk} = \sum_{i \neq j \in M_k^2} x_{jlk}, \forall j \in N, \forall k \in K \quad (9)$$

$$\sum_{i \neq j \in M_k^1} x_{ijk} = \sum_{i \neq j \in M_k^2} x_{j+n,l,k}, \forall j \in B, \forall k \in K \quad (10)$$

$$b_{ik} + \alpha_i + t_{ij} \leq b_{jk} + (1 - x_{ijk}) \times M, \forall i \in M_k^1, \forall j \in M_k^2, \forall k \in K \quad (11)$$

$$\sum_{j \in M_k^2} x_{2n+k,j,k} = 1, \forall k \in K \quad (12)$$

$$\sum_{j \in M_k^1} x_{j,2n+m+k,k} = 1, \forall k \in K \quad (13)$$

$$b_{ik} \leq b_{i+n,k}, \forall i \in B, \forall k \in K \quad (14)$$

$$b_{i+n,k} \leq u, \forall i \in B, \forall k \in K \quad (15)$$

$$Q_{ik} + q_i \leq Q_{ik} + M \times (1 - x_{ijk}), \forall i \in M_k^1, \forall j \in M_k^2, \forall k \in K \quad (16)$$

$$Q_{2n+m+k,k} = 0, \forall k \in K \quad (17)$$

$$Q_{ik} \leq a_k, \forall i \in N, \forall k \in K \quad (18)$$

Variable constraints:

$$x_{ijk} \in \{0,1\}, \forall k \in K, \forall i, j \in N \quad (19)$$

$$y_{lk} \in \{0,1\}, \forall k \in K \quad (20)$$

$$b_{ik} \geq 0, \forall i \in N, \forall k \in K \quad (21)$$

$$Q_{ik} \geq 0, \forall i \in N, \forall k \in K \quad (22)$$

The constraint (5) means that each order can only be delivered by one rider. The constraints (6) and (7) represent the satisfied relationship between two 0-1 variables. The constraint (8) is used to ensure that each order is assigned. The constraint (9) is used to ensure that nodes in the network are circulating. Constraint (10) means that the same order can only be picked up and delivered by the same rider. Constraint (11) represents the time relationship that needs to be satisfied between two adjacent nodes for the same rider, where M is the maximum constraint. The constraint (12) indicates that the rider must start from the initial node [9]. Constraint (13) indicates that the rider must return to the corresponding virtual destination after completing the delivery task. Constraint (14) indicates that the pick-up time must be met earlier than the delivery time for the same order. Constraint (15) means that for the same order, the delivery time cannot exceed the set time u . The constraints (16) and (17) represent the load change relationship of riders at two adjacent nodes. Constraint (18) means that the rider's cargo load must never exceed the maximum load. The constraints (19) and (20) are constraints on the 0-1 decision variable. The constraints (21) and (22) are constraints on other derived variables [10].

2.2 Algorithm

According to the constraints, the initial solution is generated.

Destruction operator. In this paper, three kinds of destruction operators are applied to remove the customer from the current solution, and after removing the customer, the customer is placed in the Destroy List (DL), waiting for subsequent repair operators to operate on it. The following describes the three destruction operators.

Random destroy randomly selects and removes F customer points from the current solution. This destruction operator has the advantage of fast computation and avoids local optimality through randomness. The operation of randomness removal can increase the randomness of ALNS algorithm, increase the diversity of population, and avoid the algorithm falling into the local optimal solution.

The core idea of Worst destroy is to remove the customer whose position is unsuitable in the current solution. It determines the suitability of the customer's position by calculating the cost savings after the customer's position is removed from the current solution. The higher the cost savings, the more unsuitable the customer's position in the current solution. TC_j and TC_i represent the current solution cost before and after the customer is removed, respectively. The greater the ΔTZ , the higher the cost savings after the customer is removed [11].

Calculate the ΔTZ of all customers and sort them from highest to lowest, removing customers in order. To increase the randomness of the operator, the number of removals is evenly extracted from F , and the operator can remove some customers that increase the cost, so as to achieve the purpose of reducing the cost.

Related destroy Removes customers in pairs according to the correlation between customers. The correlation is calculated as follows:

First, randomly select a customer i to remove, then calculate the degree of correlation between other customers and i and rank them in descending order, starting from the most relevant customers to remove customers until enough customers are removed.

The following three repair operators reinsert the customer from the Destroy List into the solution.

Greedy repair removes the customer and inserts it into the best position, calculates the insertion path v of customer j , and the insertion cost $\Delta = \Delta I(i, r, q) - d_i - d_j - d_i$ at the position, and inserts customer j into the optimal node best loc. If all insertion points are not feasible to insert, a new path is taken. Until all the removed customers are returned to the path, and the new solution is finally obtained.

The insertion position of the customer point is determined with Regret repair according to the insertion cost regret value of the customer. By recalculating the insertion cost of customer points in DL to the second waiting position of the new solution, and estimating the regret value of inserting the current position accordingly, the difference between the two is the largest, and the regret value is large, which means that if the current node is not inserted, the node will be selected to be inserted, and then the insertion is performed, and the regret value is inserted from the high point of the regret value. The formula for calculating the regret value is as follows:

$$reg_k = IZ' - IZ_j$$

On behalf of the regret value after the insertion of candidate node k by the customer point, the value with the highest regret value is selected for insertion, and the new solution is obtained by cyclic selection until the end of insertion.

Random repair randomly inserts customers from DL back into a path of the solution until all customer points in DL are reinserted into the path to produce a new solution.

If only the optimal solution is accepted in the optimization process, the algorithm is prone to fall into the local optimal. To avoid this situation, the simulated annealing algorithm is used to allow a certain different solution to be accepted. In the annealing process, a lower difference solution is accepted with a certain probability according to the Metropolis criterion, and the acceptance probability decreases with the decrease in temperature. If the total cost of the new solution is less than the optimal solution, the solution is accepted, and both the optimal solution and the current solution are updated. If the total cost of the new solution is greater than the optimal solution but less than the total cost of the current solution, the solution is accepted and the current solution is updated. If the total cost of the new solution is greater than the total cost of the current solution, the new solution is accepted with a certain probability, which is calculated as follows:

$$RE_e = e^{\frac{Z_{new} - Z_{now}}{TEM}}$$

Where: Z_{new} , Z_{now} are the total cost of the current solution and the new solution respectively; e is a natural constant; TEM Indicates the current temperature. The temperature decreases at the cooling rate ΔTEM . The initial temperature is TEM_0 , and the end temperature is TEM_f .

Considering the difference in the ability to explore the solution between the damage operator and the repair operator, the score weight of the operator is updated according to the acceptance criteria, and the damage operator and the repair operator are selected by the roulette algorithm. At the beginning, the score weights of all operators are the same. In the iterative process, the adjustment sub-operators are set up, and the weight of the newly solved parts is determined by roulette. The purpose of the operator strategy is to increase the number of operators and to determine the probability of selecting the sub destruction in the iteration process according to the temperature and the weight of the sub operator. In the process of selection, ALNS can adjust the operator weight adaptively according to the performance of the operator and strengthen the search.

3 Analysis Of Experimental Results

3.1 Experimental Parameter Settings

The maximum load volume of the rider is 200, the specified time is 20 minutes, the drug collection time is 0.5 minutes, and the rider service time is 1 minute

3.2 Result Analysis

After 100 iterations of ALNS algorithm, the iteration diagram is obtained, as shown in Figure 1. The optimal solution of the model is reached in 60 iterations. Through the iteration diagram, it can be found that the optimal result is obtained at about the 55th iteration. It can also be seen from the convergence diagram that the designed algorithm can effectively jump out of the local optimal solution and improve the convergence of the algorithm because of the addition of the new solution acceptance criterion of simulated annealing. It is worth noting that in the distribution paths, there are cases where the paths delivered by electric vehicles (red line) cross the paths delivered by fuel vehicles (green line) because the algorithm finds that the total cost is lower when these points are delivered by electric vehicles during the iteration process.

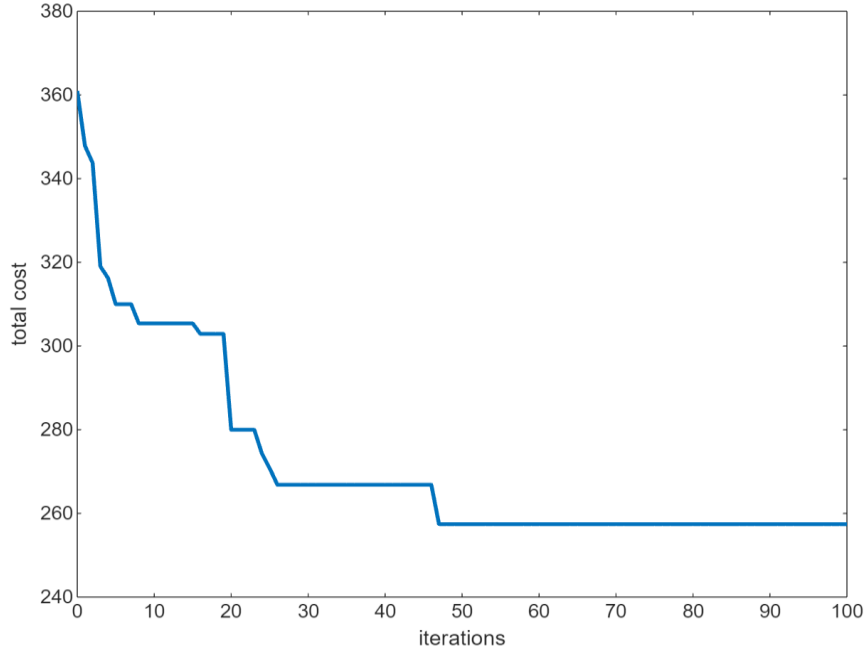


Fig. 1. Iteration diagram of ALNS algorithm

In the figure, there are a total of 8 delivery riders, of which 2 riders are idle and 6 riders are responsible for the delivery task of all demand points, each rider is responsible for the arrangement of the demand points as shown in the table, the 1st rider starts from the initial position, goes to pharmacy 2, pharmacy 1 picks up the medicine, and then is responsible for the delivery of the customer points 16, 3, 9, 20, 12, and 5, and then returns to the initial position; the time taken is 18.77 minutes, and the form cost of \$52.14 and performance cost of \$21.47, the second rider traveled from the initial position to Pharmacy 1 to pick up the medication, and subsequently delivered the medication to Client 8 and Client 10 before returning to the initial position; the time taken was 4.69 minutes, the cost of traveling was \$9.30, and the cost of performance was \$3.83, and the third rider went to Pharmacy 1 to pick up the medication, then satisfied Client Point 7, and then went to Pharmacy 2 to pick up the medication, then satisfy customer points 22, 11, 23, and 14, and then return to the initial position, with a time bit of 18.41 minutes, a driving cost bit of \$52.72, and a performance cost bit of \$21.71; Rider 4 goes from the initial position to Pharmacy 2 to pick up the medication, and returns to the starting point after visiting only customer point 17, with a time bit of 2.45 minutes, a driving cost bit of \$4.04, and a performance cost bit of 1.66, Rider 6 goes to Pharmacy 1 and Pharmacy 2 to pick up medication from the initial position and subsequently delivers to customer points 21, 19, 4, 6, 15, 18, 29 in turn and then returns to the starting point, with time consumed bit 18.64 minutes, traveling cost bit \$49.47, and performance cost bit \$20,372, and Rider 8 goes to Pharmacy 1 to pick up medication from the initial position and visits only customer point 13. Fig. 2 illustrates the rider pickup and delivery path scenarios, and Figure 3 shows the time Gantt chart scenarios of each rider departing to the pharmacy to pick up the medication, delivering it to the customer, and finally returning to the starting point.

Table 1. Rider Assignment and Delivery Performance Summary

Rider number	Route	Time used/min	Running cost/yuan	Performance cost/yuan
1	24-2-1-16-3-9-20-12-5-24	18.7705	52.1497	21.4734
2	25-1-8-10-25	4.6904	9.3093	3.8333
3	26-1-7-2-22-11-23-14-26	18.406	52.7253	21.7104
4	27-2-17-27	2.4517	4.0446	1.6654
5				
6	29-1-2-21-19-4-6-15-18-29	18.6411	49.4749	20.372
7				
8	31-1-13-31	3.9458	10.3947	4.2802

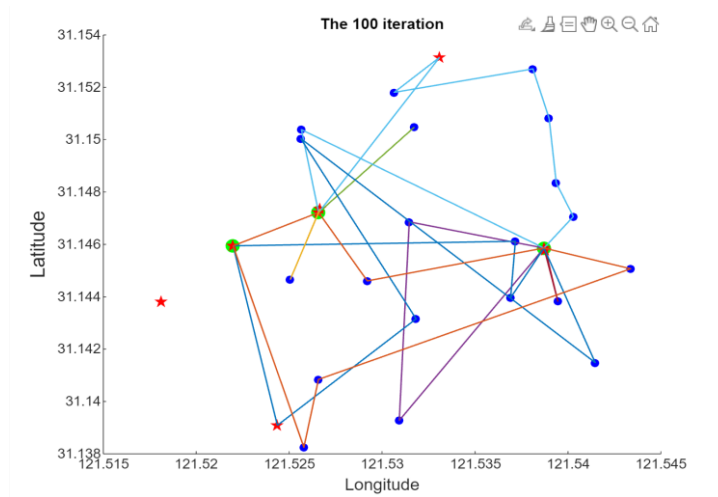


Fig. 2. Optimal distribution path diagram (The red five-pointed star indicates the initial location of the rider, the green circle indicates the location of the pharmacy, and the blue circle indicates the location of the demand point.)

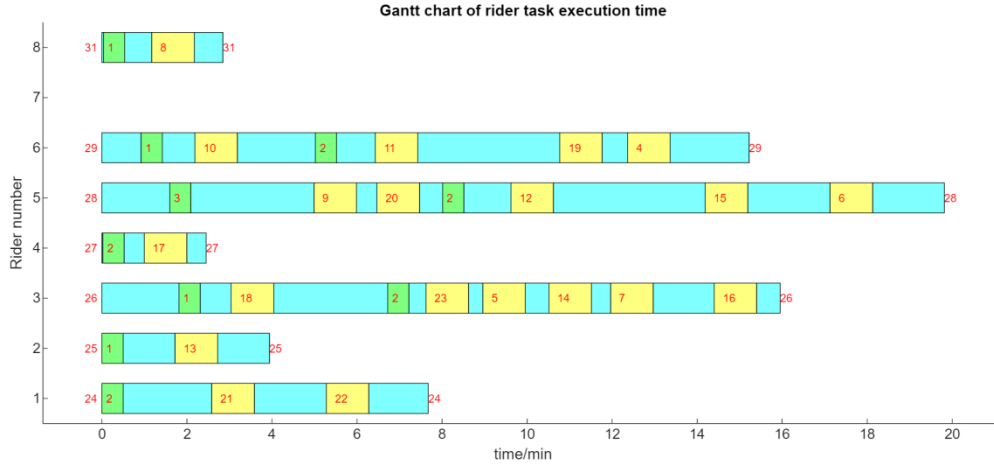


Fig. 3. Time arrangement diagram of the rider performing the task (Green is the pick-up period at the pharmacy, blue is the driving period, and yellow is the service period.)

4 Conclusion

This paper proposes and verifies the effectiveness of the adaptive large-scale neighborhood search algorithm (ALNS) for drug delivery route optimization with capacity constraints. By establishing the VRP model and combining it with the actual distribution demand, this paper studies how to minimize the total distribution distance based on ensuring the punctual delivery of each order. In the design of the model, multiple constraints such as time window, load limit and driving cost were comprehensively considered. Three kinds of damage operators (random damage, worst damage and correlated damage) and three kinds of repair operators (greedy repair, regret repair and random repair) were applied to further optimize the solution process of understanding.

The experimental results show that the ALNS algorithm can effectively find the optimal distribution path scheme after several iterations, significantly reduce the distribution distance and time, and improve distribution efficiency. The research results of this paper provide an important reference value for the actual drug delivery system, which can effectively improve the timeliness of drug delivery and the rate of patient treatment.

In this paper, the effectiveness and feasibility of the proposed algorithm are verified by detailed experimental data and analysis, which provides a new idea and method for solving the complicated drug delivery route optimization problem. Future studies can further optimize the algorithm to consider more practical constraints, such as traffic conditions and weather effects, in order to improve the applicability and robustness of the algorithm in practical applications.

References

- [1] Dantzig G., Ramser H, The Truck Dispatching Problem [J]. *Management Science*,1959,6(1):80-91.
- [2] Clarke G., Wright J. W, Scheduling of Vehicles from a Central Depot to a Number of Delivery Points [J]. *Operations Research*, 1964,12(4):568-581.
- [3] Lenstra J. K., Kan A, Complexity of vehicle routing and scheduling problems [J]. *Networks*, 2010,11(2):221-227.
- [4] Gayialis S.P, Kechagias E.P, A Multiobjective Large Neighborhood Search Metaheuristic for the Vehicle Routing Problem with Time Windows [J]. *Algorithms*, 2020,13(10):243.
- [5] Corstjens J., Depaire, B., Caris A, A multilevel evaluation method for heuristics with an application to the VRPTW [J]. *International Transactions in Operational Research*,2020,27(1):168-196.
- [6] Gutierrez A., DieulleL., LabadieN, A multi-population algorithm to solve the VRP with stochastic service and travel times [J]. *Computers & Industrial Engineering*,2018, 125:144-156.
- [7] Lu Zhen, Chengle Ma, Kai Wang, Liyang Xiao, Wei Zhang. Multi-depot multi-trip vehicle routing problem with time windows and release dates [J]. *Transportation Research Part E*, 2020, 135(C):1-21.
- [8] Song C. S., Sung S. H, Integrated Service Network Design for a Cross-Docking Supply Chain Network [J]. *The Journal of the Operational Research Society*, 2003, 54(12):1283-1295.
- [9] Musa R, Arnaout J P, Jung H. Ant colony optimization algorithm to solve for the transportation problem of cross-docking network [J]. *Computers & Industrial Engineering*, 2010,59(1):85-92.
- [10] Mehmet G., James H., Bookbinder. CROSS-DOCKING AND ITS IMPLICATIONS IN LOCATION-DISTRIBUTION SYSTEMS [J]. *Journal of Business Logistics*, 2004, 25(2):199-228.
- [11] Miao, Z., Yang, F., Fu, K. et al. Transshipment service through crossdocks with both soft and hard time windows. *Ann Oper Res* 192, 21–47 (2012). <https://doi.org/10.1007/s10479-010-0780-4>

Numerical Schemes for Partial Difference Equation in Physics

Houxu Chen^{1,*}, Shengjie Niu², Shuming Zhang³

{21307130276@m.fudan.edu.cn¹, nsj.dylan@gmail.com², zcpsz2@ucl.ac.uk³}

Department of Aeronautics and Astronautics, Fudan University, Shanghai, 200433, China¹

School of Physics and Astronomy, University of Edinburgh, Edinburgh, EH9 3JF, UK²

Department of Physics and Astronomy, University College London, Gower St, London, WC1E 6BT, UK³

Abstract. This paper examines various numerical schemes for 1-D and 2-D advection and diffusion equations using MATLAB, focusing on stability, accuracy, and performance under different boundary conditions [1]. For 1-D advection, methods such as the upwind, implicit upwind, Beam-Warming(B-W), Lax-Friedrichs(L-F), and Lax-Wendroff(L-W) schemes are evaluated. The implicit upwind scheme delivers consistent results, the Beam-Warming scheme works well under specific conditions, while the upwind scheme shows dissipation and dispersion. In 2-D advection, the upwind and Lax-Friedrichs schemes are tested, with the upwind scheme being more stable with discontinuities but less stable for smooth solutions. For diffusion, the Classical, Dufort-Frankel(D-F), and Crank-Nicolson(C-N) schemes are analyzed. The Crank-Nicolson scheme proves to be the most accurate, while the Classical scheme is fast but mesh-dependent, and the Dufort-Frankel scheme is stable but introduces minor fluctuations. The paper suggests using operator splitting to improve 2-D advection stability.

Keywords: Numerical Scheme, Diffusion Equation, Advection Equation, Partial Differential Equation

1 Introduction

1.1 Objectives

Our research aims to analyze the different numerical schemes used in various physical equations, and achieve the following three objectives: [2]

1. Analysis the stability of the numerical schemes.
2. Examine how changes in the advection speed affect the numerical solution.
3. Investigate the impact of different initial conditions.

1.2 Significance of the Study

For physical equations with extremely large computational demands or those without analytical solutions, directly seeking numerical solutions is impractical. [3] This research can be utilized to assist in selecting appropriate numerical schemes for numerical simulations of physical equations. Depending on the required resolution, boundary conditions, and initial conditions, different numerical models can be chosen accordingly.

2 Theoretical Background

2.1 Physical Significance of the Equations

2.1.1 1-Dimension Advection Equation

Here we present the 1-dimensional advection equation:

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0 \quad (1)$$

Define $u(x, t)$ the quantity being transferred (e.g., temperature, concentration). t is time. x is the spatial coordinate. a is the advection speed, a constant representing the propagation speed of the substance in space.

This equation describes the propagation process of a substance in one direction. During this propagation, the concentration of the substance changes over time and space. If $a > 0$, it indicates that the substance propagates in the positive direction; if $a < 0$, it indicates that the substance propagates in the negative direction.

2.1.2 2-Dimension Advection Equation

Here we present the 2-dimensional advection equation:

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} + b \frac{\partial u}{\partial y} = 0 \quad (2)$$

It has the same physical significance as the one-dimensional case, but it needs to consider both directions in two dimensions.

2.1.3 Diffusion Equation

Similarly we present the diffusion equation:

$$\frac{\partial u}{\partial t} = a \frac{\partial^2 u}{\partial x^2} \quad (3)$$

We define $u(x,t)$ is the quantity being diffused (e.g., temperature, concentration). t is time. x is the spatial coordinate. a is the diffusion coefficient, a positive constant that represents the rate at which diffusion occurs.

In various fields, the diffusion equation is applied to different aspects, such as studying how heat propagates through materials in thermodynamics and analyzing molecular motion in chemistry. However, their fundamental significance shares similarities.

The diffusion equation captures the essential idea that the change in the quantity at any point is proportional to the curvature of the distribution. High curvature (steep gradients) leads to rapid change, while low curvature leads to slower change.

2.2 Stability Analysis

To facilitate the discussion, we study the difference scheme within a relatively abstract framework. Consider the differential equation [4]:

$$Lu = 0$$

The corresponding difference scheme is:

$$L_h u_j^n = 0$$

2.2.1 Advection Equation

where L_h is a grid function mapping that depends on the spatial grid step h and the temporal grid step τ , known as the difference operator, and u_j^n is the grid function defined at (x_j, t_n) . For example, for the problem where $Lu = u_t + au_x$, discretizing u_t with a first-order forward difference and u_x with a first-order backward difference, we obtain the following upwind scheme:

$$\frac{u_j^{n+1} - u_j^n}{\tau} + a \frac{u_j^n - u_{j-1}^n}{h} = 0.$$

Direct computation shows that the truncation error of this scheme is $O(\tau + h)$ (Note: This paper focuses on the stability analysis of numerical schemes, so for certain numerical schemes, only the error order is provided without detailed computation and analysis). The growth factor is:

$$G(\tau, k) = a\lambda e^{-ikh} + (1 - a\lambda),$$

where $\lambda = \tau/h$ is the mesh ratio. The necessary and sufficient condition for the stability of the scheme is $a\lambda \leq 1$.

Let $u(x,t)$ be a sufficiently smooth solution of the differential equation (3.17). If $u_j^n = u(x_j, t_n)$, then we have:

$$L_h u_j^n = O(\tau^p + h^q).$$

Now, we present the method of undetermined coefficients for constructing difference schemes. Assume the desired scheme is a two-level difference scheme of the following form:

$$L_h u_j^n = u_j^{n+1} - \sum_{k=-1}^1 \alpha_k u_{j+k}^n = 0. \quad (4)$$

The objective is to find the parameters α_k such that the above scheme has the highest possible truncation error order. Let u be a sufficiently smooth solution to the problem (). Denote $x = x_j, t = t_n$, and abbreviate $u(x, t)$ as u (partial derivatives of u are treated similarly). By the Taylor expansion, we have:

$$\begin{aligned} u(x_j, t_{n+1}) &= u + u_t \tau + \frac{1}{2} \tau^2 u_{tt} + O(\tau^3), \\ u(x_j + lh, t_n) &= u + u_x lh + \frac{1}{2} l^2 h^2 u_{xx} + O(h^3). \end{aligned}$$

Substituting the grid function $u_j^n = u(x_j, t_n)$ into equation 4, and using the above expansions and the differential equation, we obtain:

$$\begin{aligned} L_h u_j^n &= \left(1 - \sum_{l=-1}^1 \alpha_l\right) u + u_t \tau - \sum_{l=-1}^1 \alpha_l l h u_x \\ &\quad + \frac{1}{2} u_{tt} \tau^2 - \sum_{l=-1}^1 \frac{1}{2} (lh)^2 \alpha_l u_{xx} + O(h^3) \\ &= \left(1 - \sum_{l=-1}^1 \alpha_l\right) u - \left(a\lambda + \sum_{l=-1}^1 \alpha_l l\right) h u_x \\ &\quad + \frac{1}{2} \left(a^2 \lambda^2 - \sum_{l=-1}^1 \alpha_l l^2\right) h^2 u_{xx} + O(h^3). \end{aligned}$$

Setting the coefficients of the low-order terms in the above expansion to zero, we obtain:

$$\begin{cases} \alpha_{-1} + \alpha_0 + \alpha_1 = 1 \\ -\alpha_{-1} + \alpha_1 = -a\lambda \\ \alpha_{-1} + \alpha_1 = a^2 \lambda^2 \end{cases}$$

Thus,

$$\begin{cases} \alpha_{-1} = \frac{1}{2}(a\lambda + a^2 \lambda^2), \\ \alpha_0 = 1 - a^2 \lambda^2, \\ \alpha_1 = \frac{1}{2}(a^2 \lambda^2 - a\lambda). \end{cases}$$

Hence, we obtain the Lax-Wendroff scheme for solving the initial value problem (3.2) of the convection equation:

$$u_j^{n+1} = u_j^n - \frac{a\lambda}{2} (u_{j+1}^n - u_{j-1}^n) + \frac{a^2 \lambda^2}{2} (u_{j+1}^n - 2u_j^n + u_{j-1}^n).$$

From the construction of the algorithm, it is easy to see that the truncation error of this scheme is $O(\tau^2 + h^2)$. Let $u_j^n = v^n e^{ijkh}$, substituting into equation (4.5) yields the growth factor:

$$\begin{aligned} G(\tau, k) &= 1 - \frac{a\lambda}{2}(e^{ikh} - e^{-ikh}) + \frac{a^2\lambda^2}{2}(e^{ikh} - 2 + e^{-ikh}) \\ &= 1 - 2a^2\lambda^2 \sin^2\left(\frac{kh}{2}\right) - ia\lambda \sin(kh). \end{aligned}$$

Thus,

$$|G(\tau, k)|^2 = 1 - 4a^2\lambda^2(1 - a^2\lambda^2) \sin^4\left(\frac{kh}{2}\right),$$

Therefore, from $|G|^2 \leq 1$, we obtain the stability condition for the Lax-Wendroff scheme as $a\lambda \leq 1$.

For the equation (), discretizing u_t using a first-order forward difference and u_x using a first-order central difference, we obtain the following difference scheme:

$$\frac{u_j^{n+1} - u_j^n}{\tau} + a \frac{u_{j+1}^n - u_{j-1}^n}{2h} = 0.$$

Using the Fourier stability criterion, it can be shown that this scheme is unstable. However, replacing u_j^n with $\frac{1}{2}(u_{j-1}^n + u_{j+1}^n)$ yields the following Lax-Friedrichs scheme:

$$\frac{u_j^{n+1} - \frac{1}{2}(u_{j-1}^n + u_{j+1}^n)}{\tau} + a \frac{u_{j+1}^n - u_{j-1}^n}{2h} = 0.$$

The truncation error of this scheme is:

$$\frac{\tau}{2}u_{tt} + \frac{h^2}{6}u_{xxx}.$$

From the stability condition of the Lax-Friedrichs scheme, we have $a\lambda \leq 1$.

Characteristics are an important tool in studying the qualitative theory of hyperbolic equations. In fact, they are also helpful in constructing difference schemes for hyperbolic equations. The characteristic line for the convection equation (4.1) is given by:

$$L: \frac{dx}{dt} = a,$$

that is,

$$x = at + x_0,$$

where x_0 is the x-coordinate of the intersection of the characteristic line with the x-axis. Along the characteristic line L , the solution u remains constant. Thus, determining the grid function value at the $n+1$ -th time level translates to determining the value of the solution at the corresponding point on the n -th time level. If Q happens to be a grid node, the difference scheme is already obtained. If

it is not a grid point, the value of u at point Q can be approximated using interpolation based on the grid function values given at the n -th time level, thereby obtaining the difference scheme.

If a quadratic interpolation is performed using the grid function values $u_j^n, u_{j-1}^n, u_{j-2}^n$, the following Beam-Warming scheme is obtained:

$$u_j^{n+1} = u_j^n - a\lambda(u_j^n - u_{j-1}^n) - \frac{a\lambda}{2}(1 - a\lambda)(u_j^n - 2u_{j-1}^n + u_{j-2}^n).$$

This scheme is also known as the second-order upwind scheme. Through standard calculations, it is found that the growth factor of this scheme is:

$$G(\tau, k) = 1 - 2a\lambda \sin^2\left(\frac{kh}{2}\right) - a\lambda(1 - a\lambda)\left(2\sin^4\left(\frac{kh}{2}\right) - \frac{1}{2}\sin^2(kh)\right) - ia\lambda \sin(kh)\left[1 + 2(1 - a\lambda)\sin^2\left(\frac{kh}{2}\right)\right].$$

Thus,

$$|G|^2 = 1 - 4a\lambda(1 - a\lambda)^2(2 - a\lambda)\sin^4\left(\frac{kh}{2}\right).$$

Therefore, from $|G|^2 \leq 1$, the stability condition is $a\lambda \leq 2$.

This paper focuses on [5] [6]:

1. Upwind Scheme:

$$\frac{u_j^{n+1} - u_j^n}{\tau} + a \frac{u_j^n - u_{j-1}^n}{h} = 0 \quad (5)$$

2. Lax-Friedrichs Scheme:

$$\frac{u_j^{n+1} - \frac{1}{2}(u_{j-1}^n + u_{j+1}^n)}{\tau} + a \frac{u_{j+1}^n - u_{j-1}^n}{2h} = 0 \quad (6)$$

3. Lax-Wendroff Scheme:

$$u_j^{n+1} = u_j^n - \frac{a\lambda}{2}(u_{j+1}^n - u_{j-1}^n) + \frac{a^2\lambda^2}{2}(u_{j+1}^n - 2u_j^n + u_{j-1}^n) \quad (7)$$

4. Beam-Warming Scheme:

$$u_j^{n+1} = u_j^n - a\lambda(u_j^n - u_{j-1}^n) - \frac{a\lambda}{2}(1 - a\lambda)(u_j^n - 2u_{j-1}^n + u_{j-2}^n) \quad (8)$$

For the two-dimensional wave equation, the above schemes can be easily generalized to obtain the corresponding numerical schemes and their stability conditions:

1. 2D Upwind Scheme:

$$\frac{u_{i,j}^{n+1} - u_{i,j}^n}{\tau} + a \frac{u_{i,j}^n - u_{i-1,j}^n}{h} + b \frac{u_{i,j}^n - u_{i,j-1}^n}{h} = 0 \quad (9)$$

2. 2D Lax-Friedrichs Scheme:

$$\frac{u_{i,j}^{n+1} - \frac{1}{4}(u_{i-1,j}^n + u_{i+1,j}^n + u_{i,j-1}^n + u_{i,j+1}^n)}{\tau} + a \frac{u_{i+1,j}^n - u_{i-1,j}^n}{2h} + b \frac{u_{i,j+1}^n - u_{i,j-1}^n}{2h} = 0 \quad (10)$$

2.2.2 Diffusion Equation

For the heat (diffusion) equation:

$$u_t = au_{xx}$$

we present several typical difference schemes [6] for the numerical solution of the initial value problem, where $a > 0$ is a given constant representing the thermal conductivity (diffusion) coefficient.

1. Classical Scheme: For equation (5.1), using a forward difference for u_t and a central difference for u_{xx} , we obtain the four-point explicit scheme:

$$\frac{u_j^{n+1} - u_j^n}{\tau} - a \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} = 0.$$

It is easy to see that the truncation error of this scheme is $O(\tau + h^2)$. Rewriting (5.2), we get:

$$u_j^{n+1} = u_j^n + a\lambda (u_{j+1}^n - 2u_j^n + u_{j-1}^n),$$

where $\lambda = \tau/h^2$ is the grid ratio. Substituting $u_j^n = v^n e^{ijkh}$, $k \in \mathbb{R}$, into the above equation, we obtain $v^{n+1} = Gv^n$, where the growth factor G is:

$$G(\tau, k) = 1 - 4a\lambda \sin^2\left(\frac{kh}{2}\right).$$

According to the von Neumann condition, the necessary and sufficient condition for the stability of scheme (5.2) is $a\lambda \leq 1/2$.

If we perform differencing in the x -direction at time level t_{n+1} , we can obtain the implicit scheme for solving the advection equation:

$$\frac{u_j^{n+1} - u_j^n}{\tau} + a \frac{u_j^{n+1} - u_{j-1}^{n+1}}{h} = 0$$

The amplification factor for this scheme is $G(\tau, k) = \frac{1}{1 + a\lambda - e^{-ikh}}$, which is unconditionally stable.

2. Weighted Implicit Scheme: [7] The difference equation for this scheme is:

$$\frac{u_j^n - u_j^{n-1}}{\tau} - a \left[\theta \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} + (1 - \theta) \frac{u_{j+1}^{n-1} - 2u_j^{n-1} + u_{j-1}^{n-1}}{h^2} \right] = 0,$$

where $\theta \in [0, 1]$ is the weight. Equation (5.3) is:

$$\begin{aligned} & -a\lambda \theta u_{j+1}^n + (1 + 2a\lambda \theta) u_j^n - a\lambda \theta u_{j-1}^n \\ & = a\lambda (1 - \theta) u_{j+1}^{n-1} - [1 + 2a\lambda (1 - \theta)] u_j^{n-1} + a\lambda (1 - \theta) u_{j-1}^{n-1}. \end{aligned}$$

Direct calculations show that the truncation error of the above scheme is:

$$E_h = L_h u_j^n = a \left(\frac{1}{2} - \theta \right) \tau \partial_{txx} u + O(\tau^2 + h^2).$$

Thus, when $\theta \neq \frac{1}{2}$, the truncation error is $O(\tau + h^2)$. When $\theta = \frac{1}{2}$, the truncation error is $O(\tau^2 + h^2)$, achieving second-order accuracy. In this case, the scheme is known as the Crank-Nicolson scheme:

$$\frac{u_j^n - u_j^{n-1}}{\tau} - a \left(\frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{2h^2} + \frac{u_{j+1}^{n-1} - 2u_j^{n-1} + u_{j-1}^{n-1}}{2h^2} \right) = 0.$$

Direct calculation shows that the growth factor of scheme (5.3) is:

$$G(\tau, k) = \frac{1 - 4(1 - \theta)a\lambda \sin^2\left(\frac{kh}{2}\right)}{1 + 4\theta a\lambda \sin^2\left(\frac{kh}{2}\right)}.$$

From

$$|G(\tau, k)| \leq 1,$$

we get

$$4a\lambda(1 - 2\theta) \sin^2\left(\frac{kh}{2}\right) \leq 2.$$

3. Richardson Scheme:

$$\frac{u_j^{n+1} - u_j^{n-1}}{2\tau} - a \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} = 0$$

is unstable. However, if u_j^n is replaced with $\frac{u_j^{n+1} + u_j^{n-1}}{2}$, we obtain the Dufort-Frankel scheme:

$$\frac{u_j^{n+1} - u_j^{n-1}}{2\tau} - a \frac{u_{j+1}^n - \left(u_j^{n+1} + u_j^{n-1} \right) + u_{j-1}^n}{h^2} = 0.$$

This is a three-level explicit scheme [8], which is unconditionally stable.

This paper focuses on:

Classical explicit scheme

$$\frac{u_j^{n+1} - u_j^n}{\tau} - a \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} = 0 \quad (11)$$

Classical implicit scheme

$$\frac{u_j^{n+1} - u_j^n}{\tau} + a \frac{u_j^{n+1} - u_{j-1}^{n+1}}{h} = 0 \quad (12)$$

Crank-Nicolson scheme:

$$\frac{u_j^n - u_j^{n-1}}{\tau} - a \left(\frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{2h^2} + \frac{u_{j+1}^{n-1} - 2u_j^{n-1} + u_{j-1}^{n-1}}{2h^2} \right) = 0 \quad (13)$$

Richardson Scheme

$$\frac{u_j^{n+1} - u_j^{n-1}}{2\tau} - a \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} = 0 \quad (14)$$

Dufort-Frankel Scheme:

$$\frac{u_j^{n+1} - u_j^{n-1}}{2\tau} - a \frac{u_{j+1}^n - (u_j^{n+1} + u_j^{n-1}) + u_{j-1}^n}{h^2} = 0 \quad (15)$$

3 Results

3.1 1-D Advection

Assuming that $a = 1, 2, 4$, $h = 0.1$, $\tau = 0.08$, this work applied upwind scheme, fully implicit scheme, Beam-Warming Scheme, Lax-Friedrichs Scheme, and Lax-Wendroff Scheme to solve 1-D advection numerically. The results when $t = 4.0s$ and $-2 \leq x \leq 10$ are shown below. Based on equation

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0 \quad (16)$$

3.1.1 Upwind Scheme

$$\frac{u_j^{n+1} - u_j^n}{\tau} + a \frac{u_j^n - u_{j-1}^n}{h} = 0 \quad (17)$$

From these figures, when the initial condition is a step function, it can be seen that when $a = 1$, the numerical solution obtained from the upwind scheme can approximate the analytical solution well, while when $a = 2, 4$, it cannot approximate the analytical solution. Also, when the initial condition is a continuous function, the situation is similar. However, we can tell from the amplitude of fluctuations of different initial conditions that continuous function works better.

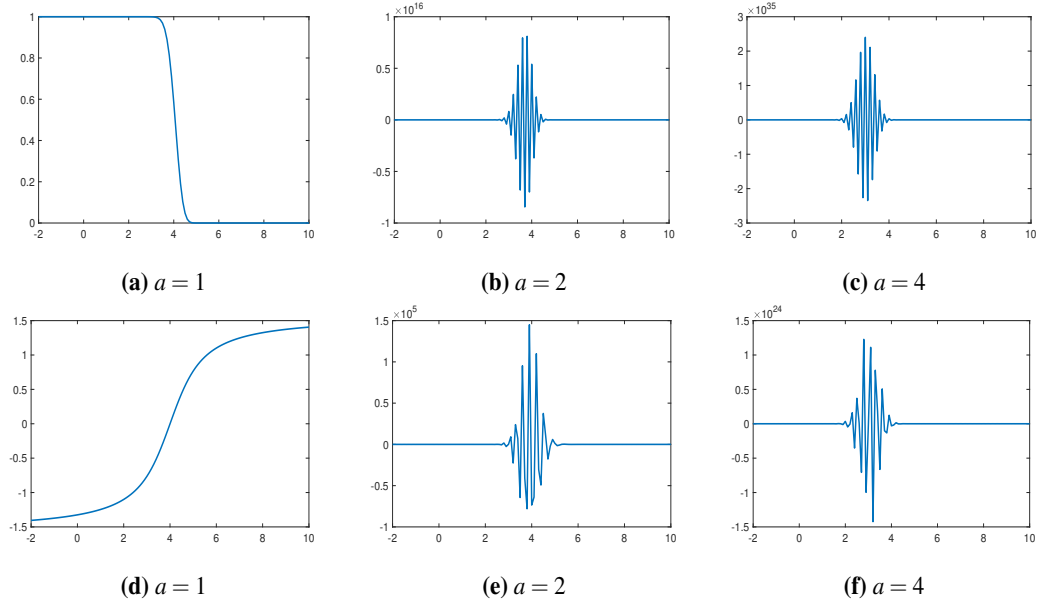


Fig. 1. Simulation results of 1-D advection in Upwind Scheme. The initial condition of (a)(b)(c) is a step function while the initial condition of (d)(e)(f) a continuous function.

3.1.2 Implicit Upwind Scheme

$$\frac{u_j^{n+1} - u_j^n}{\tau} + a \frac{u_j^{n+1} - u_{j-1}^{n+1}}{h} = 0 \quad (18)$$

In fully implicit scheme, the numerical solution can approximate the analytical solution well all the time, regardless of the initial condition and the value of a .

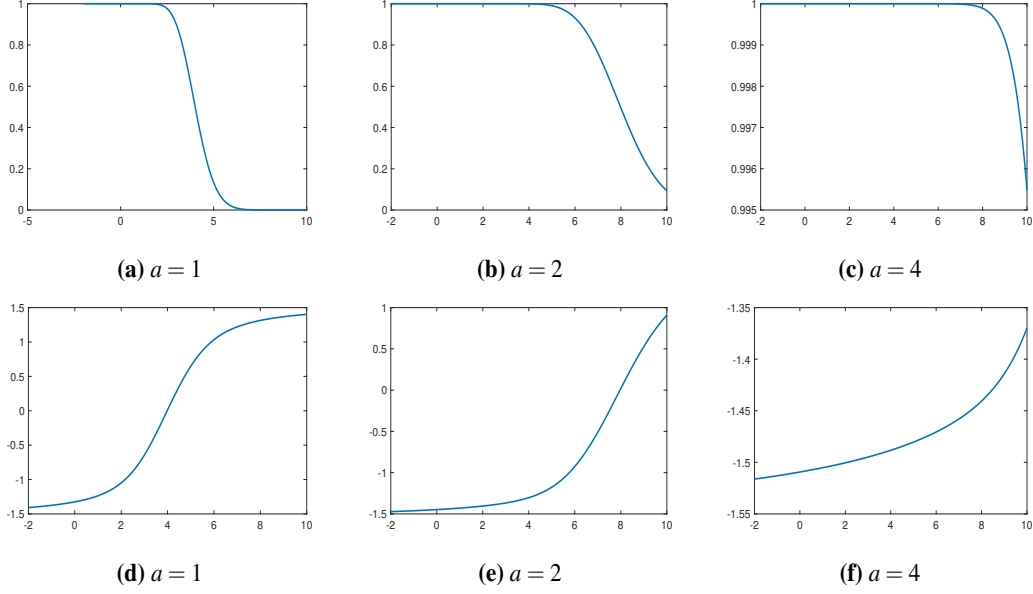


Fig. 2. Simulation results of 1-D advection in Fully Implicit Scheme. The initial condition of (a)(b)(c) is a step function while the initial condition of (d)(e)(f) a continuous function.

3.1.3 Beam-Warming Scheme

$$u_j^{n+1} = u_j^n - a\lambda(u_j^n - u_{j-1}^n) - \frac{a\lambda}{2}(1 - a\lambda)(u_j^n - 2u_{j-1}^n + u_{j-2}^n) \quad (19)$$

From these figures, when the initial condition is a step function, it can be seen that when $a = 1, 2$, the numerical solution obtained from the Beam-Warming Scheme can approximate the analytical solution well, However, there are ringings resulting from discontinuity. When $a = 4$, it cannot approximate the analytical solution. When the initial condition is a continuous function and $a = 1, 2$, there is no fluctuation. When $a = 4$, the amplitude of fluctuations are smaller than the 1st condition. Beam-Warming Scheme works better under this initial condition

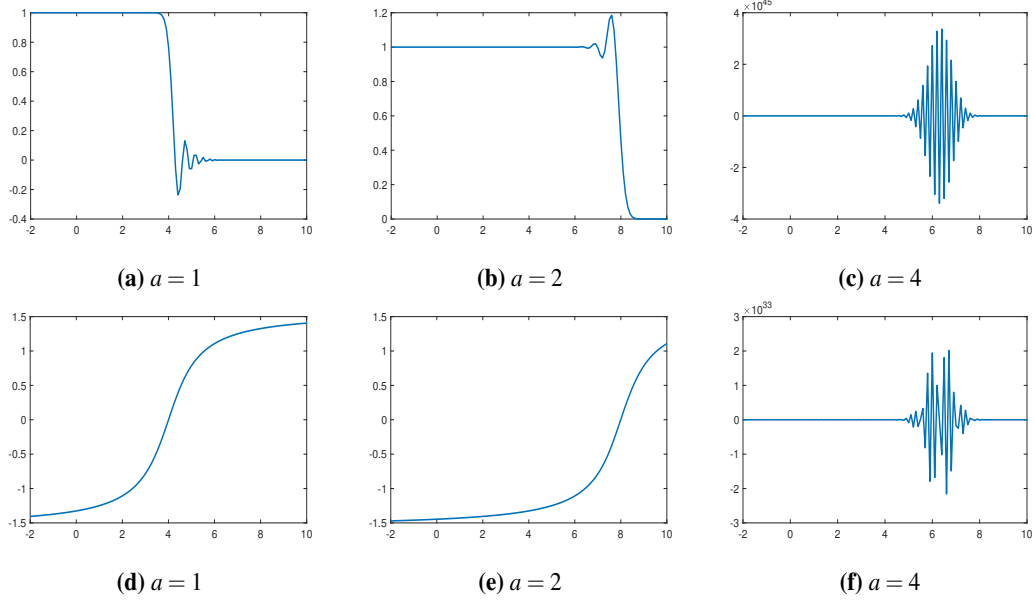


Fig. 3. Simulation results of 1-D advection in Beam-Warming Scheme. The initial condition of (a)(b)(c) is a step function while the initial condition of (d)(e)(f) a continuous function.

3.1.4 Lax-Friedrichs Scheme

$$\frac{u_j^{n+1} - \frac{1}{2}(u_{j-1}^n + u_{j+1}^n)}{\tau} + a \frac{u_{j+1}^n - u_{j-1}^n}{2h} = 0 \quad (20)$$

From these figures, when the initial condition is a step function, it can be seen that when $a = 1$, the numerical solution obtained from the Lax-Friedrichs Scheme can approximate the analytical solution well. However, the figure shows that it's a non-smooth line. When $a = 2, 4$, it cannot approximate the analytical solution. The analytical solution of the original equation cannot be approximated when $a = 2, 4$. When the initial condition is a continuous function and $a = 1$, the line is smooth, and there is no fluctuation. When $a = 2, 4$, the amplitude of fluctuations are smaller than that of the 1st condition. Lax-Friedrichs Scheme works better under smooth initial condition.

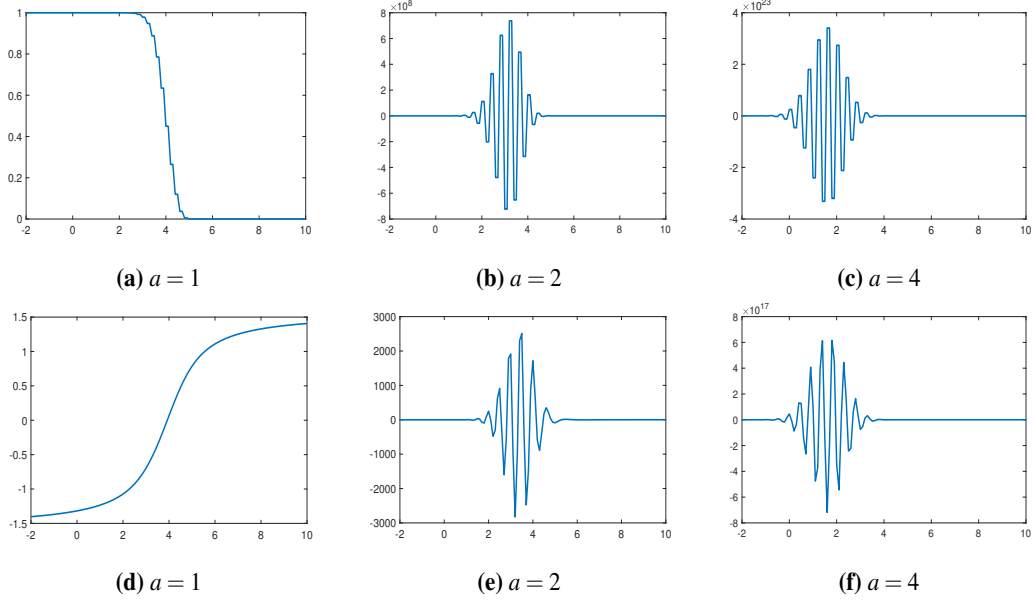


Fig. 4. Simulation results of 1-D advection in Lax-Friedrichs Scheme. The initial condition of (a)(b)(c) is a step function while the initial condition of (d)(e)(f) a continuous function.

3.1.5 Lax-Wendroff Scheme

$$u_j^{n+1} = u_j^n - \frac{1}{2}a\lambda(u_{j+1}^n - u_{j-1}^n) - \frac{1}{2}a^2\lambda^2(u_{j+1}^n - 2u_j^n + u_{j-1}^n) \quad (21)$$

From these figures, when the initial condition is a step function, it can be seen that when $a = 1$, the numerical solution obtained from the Lax-Wendroff Scheme can approximate the analytical solution well. However, there are ringings resulting from discontinuity. When $a = 2, 4$, it cannot approximate the analytical solution. When the initial condition is a continuous function and $a = 1$, there is no fluctuation. When $a = 2, 4$, the amplitude of fluctuations are smaller than that of the 1st condition. Lax-Wendroff Scheme works better under smooth initial condition.

3.1.6 Conclusion

From the results, it can be seen that when initial condition is smooth, the computational results obtained from the above four difference numerical schemes can approximate the solution of the original equation.

When $a = 2$, only the Beam-Warming Scheme and implicit upwind Scheme approximates the solution of the original equation well, and in the rest of the cases, the numerical solutions cannot approximate the analytical solution of the original equation at all.

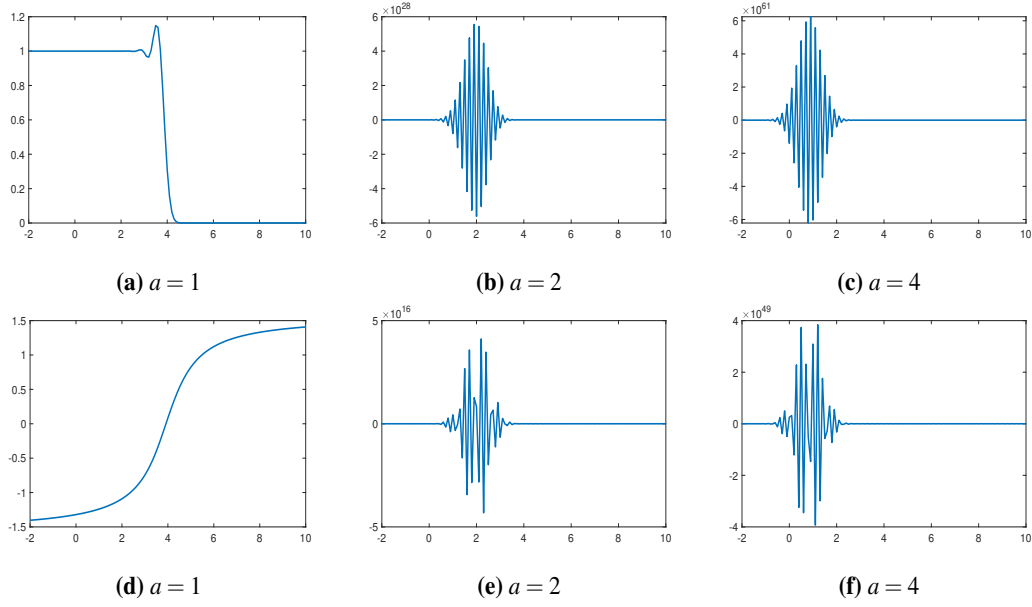


Fig. 5. Simulation results of 1-D advection in Lax-Wendroff Scheme. The initial condition of (a)(b)(c) is a step function while the initial condition of (d)(e)(f) a continuous function.

Upwind scheme has strong dissipation and dispersion for functions with discontinuities, and fully implicit scheme is stable all the time, while implicit upwind Scheme is stable all the time. All the above results are consistent with the theoretical analysis.

3.2 2-D Advection

Assuming that $a = 0.01, b = 1.01$; $a = 0.1, b = 2.1$; $a = 0.4, b = 0.4$; $a = 0.8, b = 0.8$, $h = 0.2$, $\tau = 0.16$, this work applied upwind scheme and Lax-Friedrichs Scheme to solve 2-D advection numerically. The results when $t = 4.0s$ and $-10 \leq x \leq 10, -10 \leq y \leq 10$ are shown below.

$$\frac{\partial u}{\partial t} + a\left(\frac{\partial u}{\partial x}\right) + b\left(\frac{\partial u}{\partial y}\right) = 0 \quad (22)$$

3.2.1 Upwind Scheme For Two Dimension

$$\frac{u_{j,l}^{n+1} - u_{j,l}^n}{\tau} + a \frac{u_{j,l}^n - u_{j-1,l}^n}{h} + b \frac{u_{j,l}^n - u_{j,l-1}^n}{h} = 0 \quad (23)$$

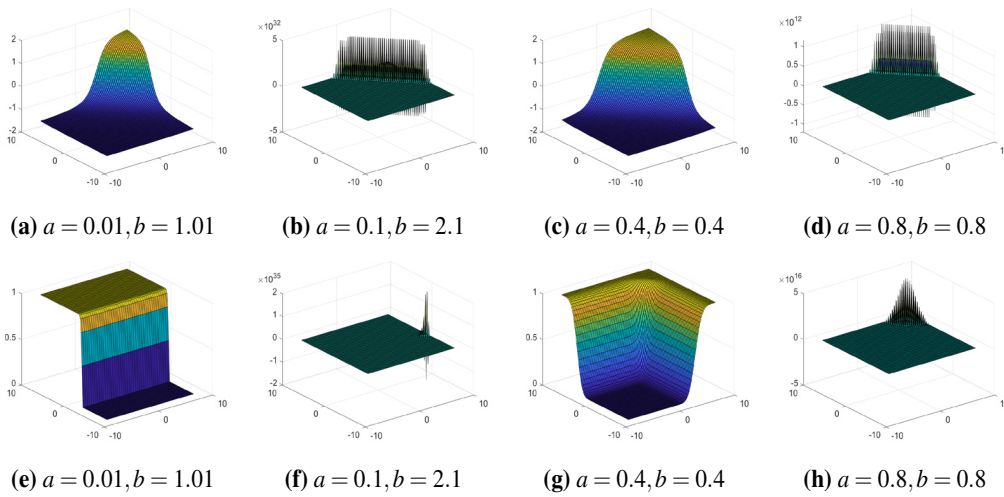


Fig. 6. Simulation results of 2-D advection equation in Upwind Scheme. The initial condition of (a)(b)(c)(d) is continuous function while the initial condition of (e)(f)(g)(h) is step function

From these figures, when $a = 0.01, b = 1.01$ and when $a = 0.4, b = 0.4$, upwind scheme for 2-D is stable, but when $a = 0.1, b = 2.1$ and when $a = 0.8, b = 0.8$, and we can tell from the amplitude of fluctuations of different initial conditions that continuous function works better.

3.2.2 Lax-Friedrichs Scheme For Two Dimension

You can even break it up into smaller sections.

$$\frac{u_{j,l}^{n+1} - \frac{1}{4}(u_{j-1,l}^n + u_{j+1,l}^n + u_{j,l-1}^n + u_{j,l+1}^n)}{\tau} + a \frac{u_{j+1,l}^n - u_{j-1,l}^n}{2h} + b \frac{u_{j,l+1}^n - u_{j,l-1}^n}{2h} = 0 \quad (24)$$

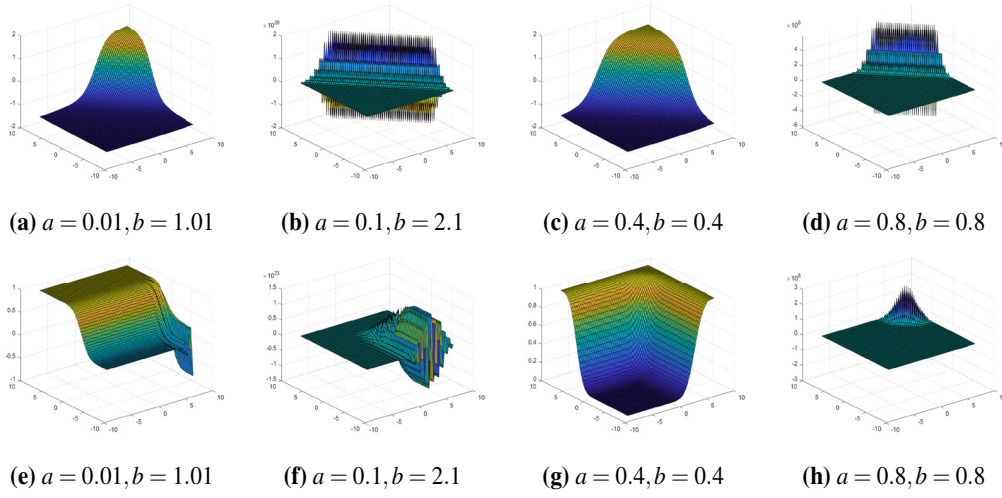


Fig. 7. Simulation results of 2-D advection equation in Lax-Friedrich Scheme. The initial condition of (a)(b)(c)(d) is continuous function while the initial condition of (e)(f)(g)(h) is step function

From these figures, when $a = 0.01$, $b = 1.01$ and when $a = 0.4$, $b = 0.4$, upwind scheme for 2-D is stable, but when $a = 0.1$, $b = 2.1$ and when $a = 0.8$, $b = 0.8$, and we can tell from the amplitude of fluctuations of different initial conditions that continuous function works better.

3.2.3 Conclusion

When dealing with discontinuity, upwind scheme performs better. For these two schemes, when $a = 0.01$, $b = 1.01$ and when $a = 0.4$, $b = 0.4$, upwind scheme for 2-D is stable, but when $a = 0.1$, $b = 2.1$ and when $a = 0.8$, $b = 0.8$. All the above results are consistent with the theoretical analysis.

3.3 Diffusion

Assuming that $a = 0.4, 0.8, 1.6$, $h = 0.2$ diffusion equation, $\tau = 0.04$, this work applied Classical Scheme, Richardson Scheme, D-F Scheme and Crank-Nicolson Scheme to solve diffusion equation numerically. The results when $t = 4.0s$ and $-5 \leq x \leq 5$ are shown below.

3.3.1 Classical

The classical scheme is expressed as follows:

$$\frac{u_j^{n+1} - u_j^n}{\tau} - a \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} = 0, \text{ where } \lambda = \frac{\tau}{h^2} \quad (25)$$

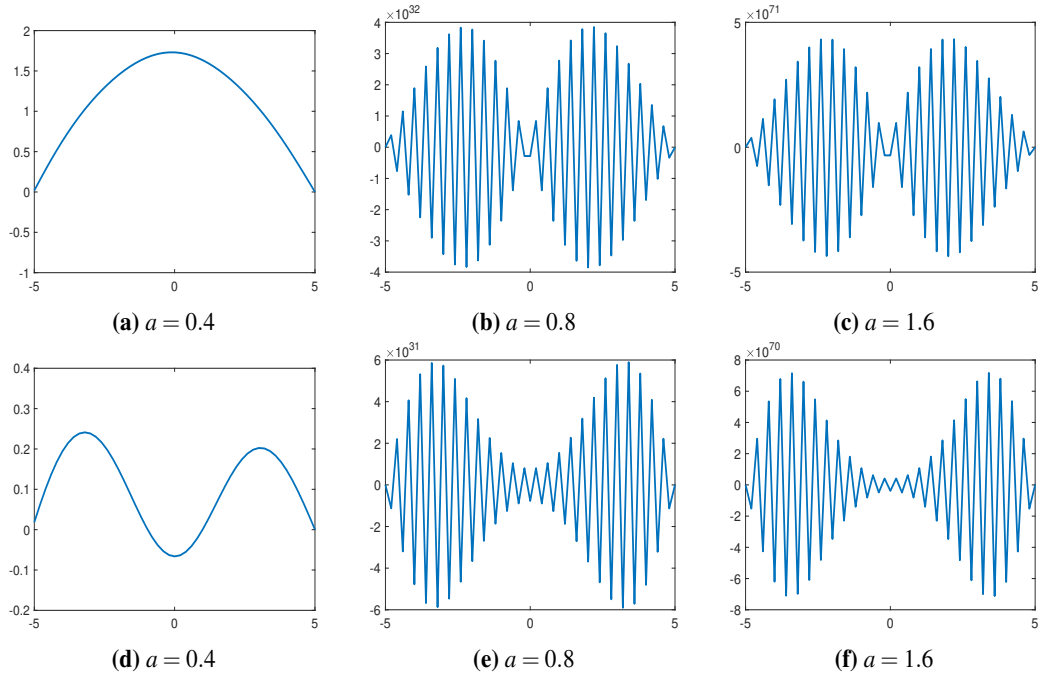


Fig. 8. Simulation results of the diffusion equation using the classical scheme. The initial condition of (a)(b)(c) is a step function while the initial condition of (d)(e)(f) a continuous function.

As the figures show, in Classical scheme, among the two BCs, only the results obtained with $a = 0.4$ were in a stable state. For $a = 0.8$, the results reach an order of 10^{31} . Similarly, for $a = 1.6$, the results diverged to the order of 10^{70} .

This result consists of the stability requirement of $a\lambda < 0.4$ (remind here $\lambda = \frac{0.04}{0.2^2} = 1$)

3.3.2 Richardson

$$\frac{u_j^{n+1} - u_j^{n-1}}{2\tau} - a \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} = 0 \quad (26)$$

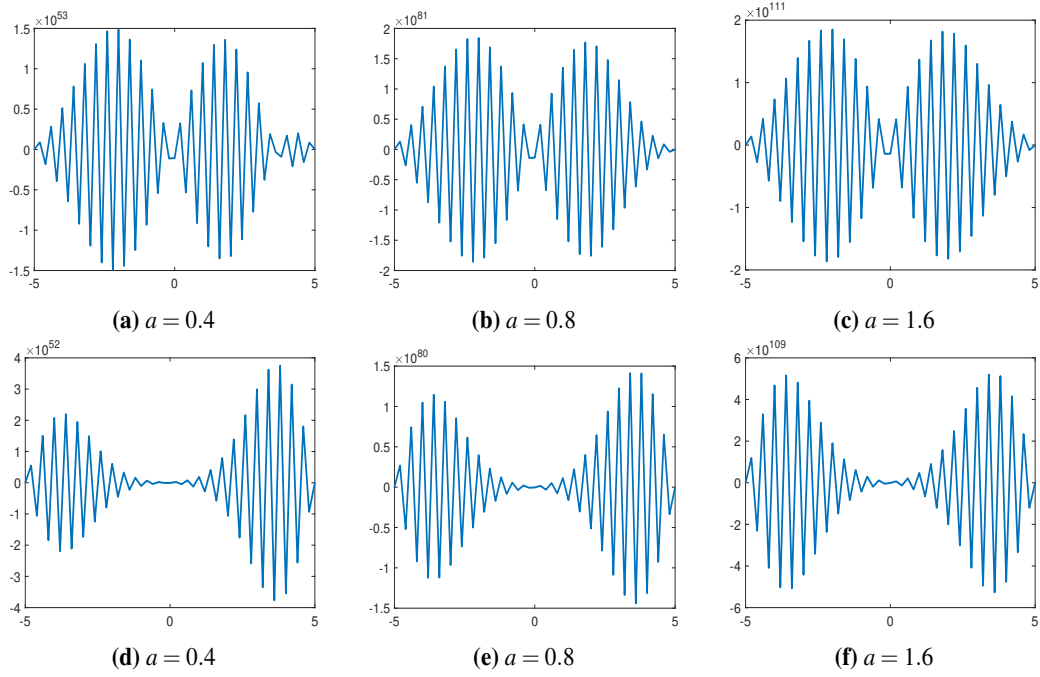


Fig. 9. Simulation results of diffusion equation in Richardson scheme. The initial condition of (a)(b)(c) is a step function while the initial condition of (d)(e)(f) a continuous function.

In Richardson scheme, all results are unstable. The values reach very large magnitudes from 10^{50} to 10^{110} in either step function or continuous function with any values of a .

These results also consist with the prediction that Richardson scheme is unstable in any situation.

3.3.3 D-F

$$\frac{u_j^{n+1} - u_j^{n-1}}{2\tau} - a \frac{u_{j+1}^n - (u_j^{n+1} + u_j^{n-1}) + u_{j-1}^n}{h^2} = 0 \quad (27)$$

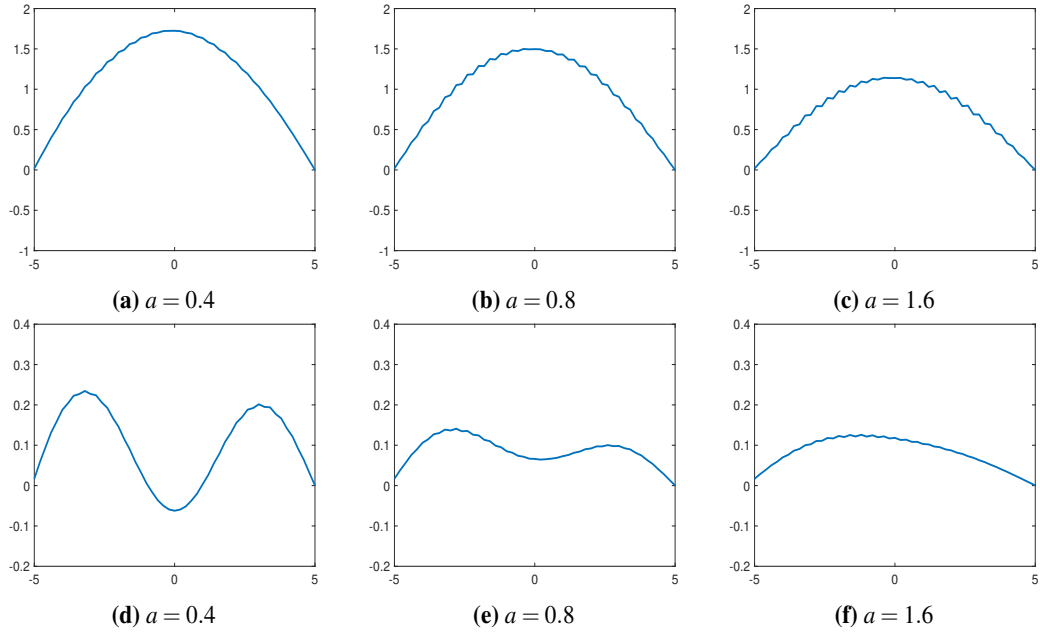


Fig. 10. Simulation results of diffusion equation in Dufort-Frankel scheme. The initial condition of (a)(b)(c) is a step function while the initial condition of (d)(e)(f) a continuous function.

From the figures above, all results from Dufort-Frankel scheme show their stability in all situations. with the increase of a , the results goes more flat in either BC. which are caused by the "faster" diffusion rate.

In this scheme, it is noticeable that some fluctuations occurs, this is caused by discretization of time and space. These errors accumulate during the iterative process. Improvement can be done by providing more precise grids.

The overall stability consist with the prediction that Dufort-Frankel scheme is stable under all circumstances.

3.3.4 C-N

$$\frac{u_j^{n+1} - u_j^n}{2\tau} - \frac{1}{2}a \left(\frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} + \frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{h^2} \right) = 0 \quad (28)$$

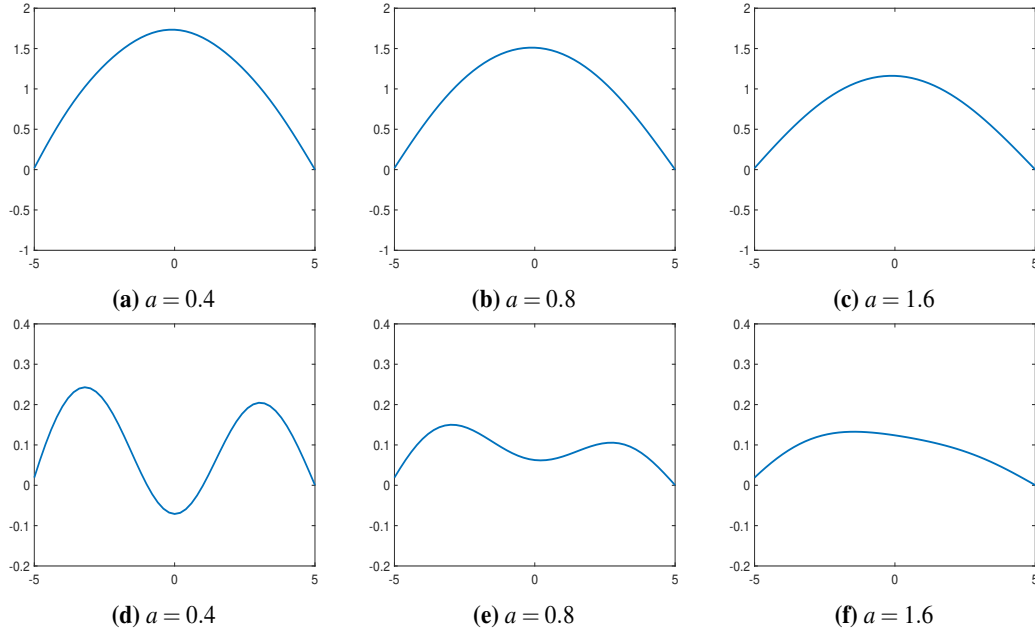


Fig. 11. Simulation results of diffusion equation in Crank-Nicolson scheme. The initial condition of (a)(b)(c) is a step function while the initial condition of (d)(e)(f) a continuous function

Similar to Dufort-Frankel scheme, as predicted, Crank-Nicolson scheme provides a general stable results. The values of the functions are very similar to the previous scheme. The repeatability shows that both D-F and C-N schemes run reliably in this simulation.

However, the "fluctuations" not exists any more. Which shows that Crank-Nicolson scheme provides better simulations.

3.3.5 Conclusion of the Diffusion schemes

From the results above, the Classical scheme shows a quick and convenient way to simulate the Diffusion equation but with a restriction of MESH ratio provided.

The Richardson scheme is not acceptable since it is not stable in any situations.

The Dufort-Frankel scheme and the Crank-Nicolson scheme both provide universally stability under arbitrary conditions. The Dufort-Frankel scheme have some errors from the fluctuations of its results, while if better accuracy is required, the Crank-Nicolson scheme can provide more accurate results in simulations.

4 Conclusion

4.1 Advection equation

From the results, it can be seen that when initial condition is smooth, the computational results obtained from the above four difference numerical schemes can approximate the solution of the original equation.

The implicit upwind scheme performs stable in all cases. When $a = 2$, expect for implicit upwind scheme, the beam-warming scheme approximates the solution of the original equation well, and in the rest of the cases, the numerical solutions cannot approximate the analytical solution of the original equation at all. Upwind scheme has strong dissipation and dispersion for functions with discontinuities. All the above results are consistent with the theoretical analysis.

4.2 2-Dimension Advection equation

In the 2-dimensional space, the behavior of the advection equation is similar to that in the 1-dimensional space. When dealing with discontinuity, upwind scheme is better, but when the solution is unstable, we can see from the amplitude that upwind scheme is less stable, because errors travel faster. All the above results are consistent with the theoretical analysis.

4.3 Diffusion equation

From the results above, the Classical scheme shows a quick and convenient way to simulate the Diffusion equation but with a restriction of mesh ratio provided. The Richardson scheme is not acceptable since it is not stable in any situations. The Dufort-Frankel scheme and the Crank-Nicolson scheme both provide universally stability under arbitrary conditions. The Dufort-Frankel scheme have some errors from the fluctuations of its results, while if better accuracy is required, the Crank-Nicolson scheme can provide more accurate results in simulations.

4.4 further improvement

1. For the two-dimensional advection equation, the operator splitting method can be used to construct schemes with stronger stability.
2. Since it is inherently unreasonable to use difference quotients at discontinuities of a function, the error produced by difference schemes at these points still needs to be

Acknowledge

These three authors contributed equally to this work

References

- [1] Regulý, I.Z.; Mudalige, G.R. (2020). Productivity, performance, and portability for computational fluid dynamics applications. *Computers & Fluids*, **199**, 104425.
- [2] Verma, A.K.; Kayenat, S. (2020). An efficient Mickens' type NSFD scheme for the generalized Burgers Huxley equation. *Journal of Difference Equations and Applications*, **26**, 1213–1246.
- [3] Gagliardi, F.; Moreto, M.; Olivieri, M.; Valero, M. (2019). The international race towards Exascale in Europe. *CCF Transactions on High Performance Computing*, **1**, 3–13.
- [4] Appadu, A.R. (2017). Performance of UPFD scheme under some different regimes of advection, diffusion, and reaction. *International Journal of Numerical Methods for Heat & Fluid Flow*, **27**, 1412–1429.
- [5] Kovács, E.; Nagy, Á.; Saleh, M. (2021). A set of new stable, explicit, second-order schemes for the non-stationary heat conduction equation. *Mathematics*, **9**, 2284.
- [6] Sanjaya, F.; Mungkasi, S. (2017). A simple but accurate explicit finite difference method for the advection-diffusion equation. *Journal of Physics: Conference Series*, **909**, 12038.
- [7] Karahan, H. (2007). Unconditional stable explicit finite difference technique for the advection–diffusion equation using spreadsheets. *Advances in Engineering Software*, **38**, 80–86.
- [8] Pourghanbar, S.; Manafian, J.; Ranjbar, M.; Aliyeva, A.; Gasimov, Y.S. (2020). An efficient alternating direction explicit method for solving a nonlinear partial differential equation. *Mathematical Problems in Engineering*, **2020**, 9647416.

Design and Risk Management of an S&P 500-Linked Snowball Auto-callable: A Comparative Analysis Using Monte Carlo Simulation and PDE Method

Chenyan Zheng^{1,†,*}, Hanxi Qin^{2,†}, Jiani Han^{3,†}
 {3230104254@zju.edu.cn¹, sqin14@smith.edu², jh796@duke.edu³}

School of Economics, Zhejiang University, Hangzhou, 310058, China¹

Mathematical Sciences Department; Economics Department, Smith College, Northampton, MA 01063, US²

Applied mathematics and computational science, Duke Kunshan University, Kunshan, 215311, China³

*corresponding author

† co-first authors

Abstract. This paper discusses the design of an S&P 500-linked Snowball Auto-callable, which aims to enrich the derivatives market. It is essential to ensure effective risk management in light of the increased complexity of the market during the COVID-19 crisis. Considering that options serve as a key financial instrument for hedging, we evaluate our product using two pricing methods - PDEs and Monte Carlo simulations. Additionally, we analyze internal and external risks, offering both investors and issuers hedge strategies and recommendations.

Keywords: Snowball Autocallable, pricing, Monte Carlo Simulation, PDE, Sensitivity analysis

1 Introduction

In recent years, snowball autocallables have garnered significant attention within the financial industry, not only because they are a novel derivative combining features of options, but also due to their mutually beneficial payoff structure for both brokers and investors. Essentially, snowball autocallables are bizarre options with obstacle terms, allowing investors to collect option premiums. The product's performance is linked to underlying assets such as indices, individual stocks, or commodities, with the triggering of knock-in and knock-out events determined by pre-set barrier levels.

The introduction of snowball autocallables has proven valuable in revitalizing the volatile stock markets, particularly in the wake of the pandemic, providing enhanced benefits for both investors and brokers while increasing market liquidity. This is one of the primary motivations for this study on snowball autocallables. Through a comprehensive literature review, it was found that snowball pricing is generally approached via two methods: the derivation through Partial Differential Equations (PDE) and simulation-based pricing using Monte Carlo methods.

This paper will first explain the mathematical derivation of both methods and explore their feasibility in programming implementation, ultimately obtaining the pricing results. By comparing the convergence and differences in the final pricing outcomes, this research aims to verify the accuracy of each method and evaluate the potential areas for improvement. After developing the snowball autocallable product, this research will also conduct thorough risk management assessments. This includes evaluating factors such as knock-in and knock-out levels, market trends, liquidity risk, and the impact of credit risk on the product's performance. And this paper roughly introduces the application of Delta hedging strategy in snowball autocallable risk hedging. By conducting these tests, this research aims to ensure that the application scenarios of the snowball autocallable are more aligned with the dynamics of real-world financial markets. Through this approach, the product can be better positioned to meet the practical demands and risks present in contemporary market environments. Additionally, through the literature review, this research has observed that snowball autocallables are more prevalent in the Chinese market, often linked to large-cap indices. Thus, this research will also explore whether these methods are equally applicable to major indices in the U.S. market.

While many brokers have already developed sophisticated pricing systems for snowball autocallables, often fine-tuning parameters manually for greater pricing precision, this paper seeks to provide a deeper understanding of snowball products for both investors and brokers alike. On the basis of designing the structured financial product to meet the investors and issuers of the product, this paper adopts quantitative and qualitative research methods to price and risk-analyze the product, confirm the feasibility of the product, enrich the product varieties in the current financial market, and provide more choices for investors and issuers.

2 Literature Review

There is no analytical solution for option pricing, so numerical methods such as binomial trees, finite differences, and Monte Carlo simulations are a practical alternative.

Cox, Ross, and Rubinstein introduced the binomial tree method in 1979 for simplifying the pricing process by converting continuous-time problems into discrete-time problems [1]. The algorithm is useful for both European and American options, but when multiple variables are involved, it becomes inefficient, resulting in a slow convergence. Finite difference methods reduce the complexity of differential equations by turning them into discrete algebraic equations that enable faster and more efficient pricing. Schwartz began using the finite difference method to approximate the exact solution of partial differential equations in 1997 [2]. Since then, the application of the finite difference method in finance has been expanded.

Monte Carlo method is a method of stochastic simulation, which is based on probability and statistical theory. This method consists of simulating many different paths and averaging the final value of the option. In 1997, Boyle introduced Monte Carlo simulations as a probabilistic method for estimating the value of European options [3]. Even though it is versatile, large simulations may require a significant amount of computational power.

In summary, the three key numerical methods in option pricing are binomial trees, finite differences, and Monte Carlo. The use of binary trees is simple, but they require a significant amount of

computational power. Finite differences are more efficient for complex options and Monte Carlo can be used when handling stochastic scenarios, but they require a considerable amount of computational power.

3 Introduction of the Snowball Auto-Callable and Our Product

3.1 Product Design and Key Elements

This paper discusses the factors that make up a standard Snowball structured auto-callable, including the knock-in and knock-out levels, volatility, as well as risk-free rate. Detailed specifications for the S&P 500 index-linked Annual Auto-Callable Notes due June 1, 2025, have been developed based on the product's profit structure. There is a detailed product overview shown on the left in Figure 1. Additionally, a clear annual yield graph is generated on the right in Figure 1.

Term	Term note
Security	Standard Snowball AutoCallable Derivative
Maturity	12M
Underlying Asset	S&P 500 Index
Knock-In Barrier Level	$S_0 * 85\%$ (Observation frequency: Daily)
Knock-Out Call Level	$S_0 * 103\%$ (Observation frequency: Monthly)
Coupon rate	20%

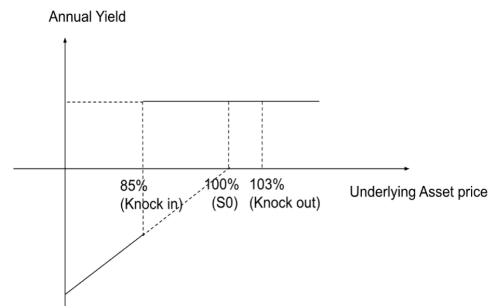


Fig. 1. Product overview(on the left) and Impact of Underlying Asset Price on Annual Yield(on the right)

3.2 payment at maturity

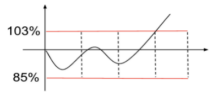
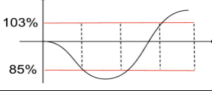
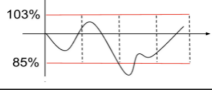
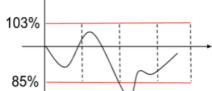
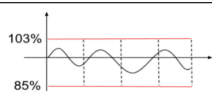
	Scenario		Maturity Payoff	
1	Underlying price knocks out (reach the 103% level)	Price never reach the knock in level and knock out	Principal + coupon (3 weeks)	
2		Price dips below 85%, then bounce back and knock out		
3	Underlying price knocks in (Drop below 85% level)	Underlying price drops below 85% and bounce back ($S_t > S_0$)	Principal	
4		Underlying price drops below 85% and not bounce back ($S_t < S_0$)	Nominal principal loss (if $S_t = 90\% * S_0$, lose $10\% * \text{principal}$)	
5	Underlying price never drops below 85% and never reaches 103%	Price fluctuates but stays in between 85% and 103%	Principal + coupon (1 month)	

Fig. 2. Payment at maturity of snowball structured derivative

The calculation of the Snowball AutoCallable payment is based on five scenarios listed below in the graph (Figure 2). This section will go through all the scenarios to provide a better understanding of the Snowball interest payment.

1. Scenario 1 (Knock-out but no Knock-in): if the underlying asset knocks out and never knocks in before, then the investors get the coupon payment (that is the coupon interest rate times the principal investor originally paid).
2. Scenario 2 (Knock-out and Knock-in): if the underlying asset knocks in before knock-out, then the investors get the coupon payment as well, which is the same as the Scenario 1
3. Scenario 3 (Knock-in but no Knock-out with Bounce back): if the underlying asset hits the knock-in ground but the asset price at maturity is higher than at the beginning, the investors get the original principle. In this case, investors gain no interest and no loss.
4. Scenario 4 (Knock-in but no Knock-out without Bounce-back): if the underlying asset knocks in and does not knock out until the maturity date, then the investor gets the loss of principal. For example, if the price of the underlying asset drops by 20%, the investor's loss at maturity of the Snowball product will be 20% of the principal.

5. Scenario 5 (No Knock-in and No Knock-out): if the underlying asset does not breach either the knock-out or knock-in conditions, the investor will receive the full principal along with the coupon interest. (Our product does not include dividend coupons, although most products in the market do have such provisions.)

4 Data

4.1 Assumption

The efficient market hypothesis states that The history is fully reflected in the present price, which does not hold any further information; Markets respond immediately to any new information about an asset price [4].

The two assumptions above suggest asset prices change according to a Markov process, which means that the price of an asset will be determined by its current price alone, not by previous prices. This model assumes that underlying asset prices are determined by a stochastic process driven by Brownian motion. A stochastic differential equation (SDE) can be used to model the price dynamics of the underlying asset as follows:

$$dS_t = \mu S_t dt + \sigma S_t dZ_t$$

Here, S_t represents the price of the underlying asset at time t , μ is the expected rate of return, σ is the volatility, and dZ_t is the increment of a standard Brownian motion, representing the random fluctuations in the asset price.

This paper prices the annual snowball auto-callable due June 1, 2025, linked to the S&P 500 Index. According to Black-Scholes, the path-dependent snowball auto-callable assumes the underlying asset follows an SDE [5]:

$$dS_t = \mu S_t dt + \sigma S_t dB_t$$

Where S_t is the stock price at time t , r is the risk-free interest rate, σ is the volatility of the underlying asset, and $dB_t = \varepsilon \sqrt{dt}$, $\varepsilon \sim N(0, 1)$ represents the standard normal distribution.

4.2 Volatility

Volatility describes the degree of volatility of a financial asset and is a measure of the uncertainty of an asset's returns. In general, the higher the volatility, the more volatile the financial assets and the greater the uncertainty of asset returns. As an important factor in the Black-Scholes model option pricing, we choose Garch volatility instead of directly using volatility in the previous year. This indicates the aggregation and tailing of the volatility. We calculate and use volatility of 18%.

4.3 Risk-free Rate

A key characteristic of the Black-Scholes-Merton differential equation is that it is independent of any variables influenced by the risk preferences of investors (Hull, 2012). The only variables present in the equation are the current stock price, time, stock price volatility, and the risk-free rate

of interest. The pricing process assumes a risk-neutral framework, in which derivative prices are unaffected by investors' subjective risk preferences. In this setting, the return on all assets is equal to the risk-free rate. As a result, under risk-neutral conditions, all cash flows can be discounted using the risk-free rate.

For this analysis, the risk-free rate is derived from the average 10-year Treasury yield, measured over the period from June 1, 2023, to June 1, 2024, resulting in a risk-free rate of 4.25%.

5 Methodology

5.1 Garch model

Due to the heteroscedasticity and volatility clustering effects of financial product yield sequences, Engle proposed the Autoregressive Conditional Heteroscedasticity (ARCH) model (ENGLE R, 1982). [6] The Garch model is an extension of the ARCH model to describe the variance structure in time-series data more accurately by introducing higher-order terms of past variance. This paper uses the Garch(1,1) model to forecast the volatility of S&P 500 in the duration of this product. The formula is as below,

$$\sigma_n^2 = \gamma V_L + \alpha r_{n-1}^2 + \beta \sigma_{n-1}^2$$

in which

$$\gamma + \alpha + \beta = 1$$

We choose the closing price of S&P 500 in the last 10 years and first create a yield chart as follows.

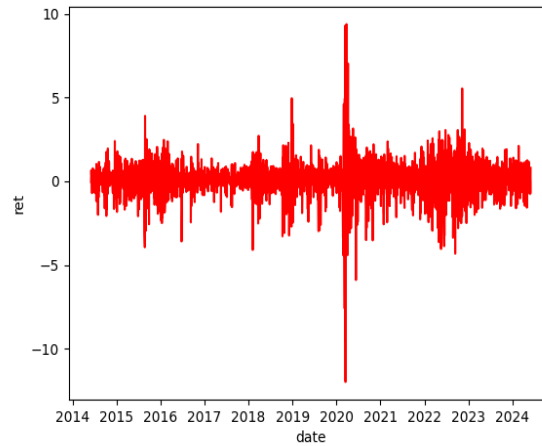


Fig. 3. Yield chart

Figure 3 shows that there is an obvious aggregation effect. By processing data and calculating LB, it can be known that the sequence is not white noise and can be modeled. Finally, we calculate the future volatility-18%.

5.2 PDE Method

5.2.1 Finite Difference Method

Under the Black-Scholes model, the stochastic differential equation (SDE) for the underlying asset (e.g., stock price) is as follows:

$$\frac{dS(t)}{S(t)} = rdt + \sigma dB(t)$$

The Black-Scholes partial differential equation can be derived from Ito's Lemma based on the assumption that there is no arbitrage in the financial market, as shown below [7]:

$$\frac{\partial f}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} + rS \frac{\partial f}{\partial S} - rf(S, t) = 0$$

The term $f(S(t), t)$ can be interpreted as any portfolio of investments that is determined only by S and t . Although the equation to solve remains the same, boundary conditions vary based on the derivative product's structure. These are the three boundary conditions mentioned here [8]:

$$f(S, t) \quad \text{as} \quad t \rightarrow T$$

$$f(S, t) \quad \text{as} \quad S \rightarrow 0$$

$$f(S, t) \quad \text{as} \quad S \rightarrow \infty$$

Rather than using the numerical method to calculate pricing, the grid search method is needed to calculate the price from back to front. The time and price steps must be selected before the grid can be constructed.

The expiration time is set at $T = 1$ year, which corresponds to 252 trading days, and the time interval is divided into $N = 252$ steps, with each step corresponding to $\Delta t = \frac{T}{N} = \frac{1}{252}$.

The price is divided into $M = 800$ price steps. The maximum asset price S_{\max} is $S_{\max} = 4 * K$, where K is the knock out level. The price step is $\Delta S = \frac{S_{\max}}{M} = 26.39$, which is approximately 0.5% of the initial price. This ensures that the price grid is fine enough to identify the dynamics of the option. The following (800×252) grid (figure 4) maintains a balance between precision and performance, ensuring that the price of an option is accurately represented over time. Gray points represent neighboring points involved in the spatial and time differencing schemes. Red points represent current function values at specific prices and times. The Crank-Nicolson method uses the information from the gray points to compute the value at the red point, combining both spatial and time differences to update the solution.

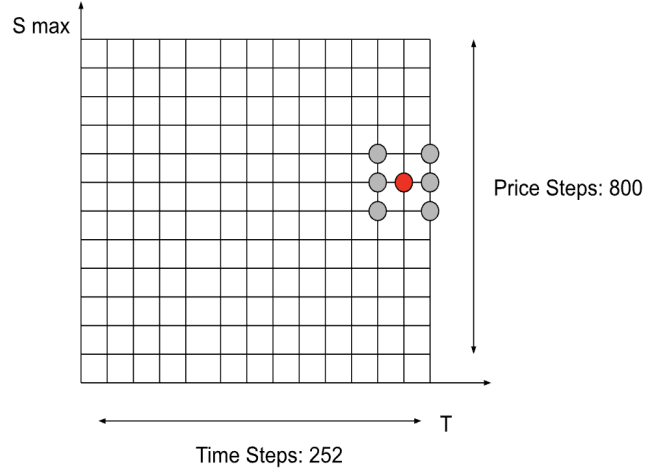


Fig. 4. Grid of PDE

To perform this analysis, it is necessary to obtain all the points on the vertical coordinate at $t = 0$, i.e., the price of the derivative at the price of S of different indices at the time of initial pricing. Accordingly, the PDE method is a method of deriving the initial price backwards from the boundary conditions.

This paper employs the Crank-Nicolson finite difference method, which combines the explicit and implicit time-stepping schemes by averaging them. The finite difference method in both the time dimension and the spatial dimensions, which employs central difference method with a parameter value of $\theta = 0.5$, can be expressed by the following formula [9]:

$$\frac{\partial V([\theta i + (1 - \theta)(i + 1)], j)}{\partial t} \approx \frac{V(i + 1, j) - V(i, j)}{\Delta t}$$

In spatial dimensions:

$$\begin{aligned} \frac{\partial V}{\partial S} &= \frac{\partial V(i, j)}{\partial S} \approx \frac{V(i, j + 1) - V(i, j - 1)}{2\Delta S} \\ \frac{\partial^2 V}{\partial S^2} &= \frac{\partial^2 V(i, j)}{\partial S^2} \approx \frac{V(i, j + 1) - 2V(i, j) + V(i, j - 1)}{(\Delta S)^2} \end{aligned}$$

By substituting the finite difference approximations into the PDE, we obtain the following:

$$\frac{V(i + 1, j) - V(i, j)}{\Delta t} + rj \frac{V(i, j + 1) - V(i, j - 1)}{2\Delta S} + \frac{1}{2} \sigma^2 j^2 \frac{V(i, j + 1) - 2V(i, j) + V(i, j - 1)}{(\Delta S)^2} - rV(i, j) = 0$$

Multiplying by Δt and rearranging terms, we obtain:

$$V(i + 1, j) = a_j V(i, j - 1) + b_j V(i, j) + c_j V(i, j + 1)$$

Where the coefficients are defined as:

$$a_j = -\frac{1}{4}rj\Delta t + \frac{1}{4}\sigma^2 j^2 \Delta t$$

$$b_j = -\frac{r\Delta t}{2} - \frac{\sigma^2 j^2 \Delta t}{2}$$

$$c_j = \frac{1}{4}rj\Delta t + \frac{1}{4}\sigma^2 j^2 \Delta t$$

For all j , we now have the following system of equations:

$$\begin{pmatrix} 1-b_1 & -c_1 & 0 & \cdots & 0 \\ -a_2 & 1-b_2 & -c_2 & \ddots & 0 \\ 0 & -a_3 & 1-b_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -a_{M-1} & 1-b_{M-1} \end{pmatrix} \begin{pmatrix} V(i,1) \\ V(i,2) \\ \vdots \\ V(i,M-2) \\ V(i,M-1) \end{pmatrix} = \begin{pmatrix} 1+b_1 & c_1 & 0 & \cdots & 0 \\ a_2 & 1+b_2 & c_2 & \ddots & 0 \\ 0 & a_3 & 1+b_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & a_{M-1} & 1+b_{M-1} \end{pmatrix} \begin{pmatrix} V(i+1,1) \\ V(i+1,2) \\ \vdots \\ V(i+1,M-2) \\ V(i+1,M-1) \end{pmatrix} + \begin{pmatrix} a_1(V(i,0)+V(i+1,0)) \\ 0 \\ \vdots \\ 0 \\ c_{M-1}(V(i,M)+V(i+1,M)) \end{pmatrix}$$

The matrix on the left-hand side contains the coefficients a_j , b_j , and c_j from the discretized equation, which denotes as M1. The vector on the left-hand side contains the values of $V(i, j)$, the option prices at time step i , which denotes as b. The vector on the right-hand side contains the option prices at the next time step $i+1$, adjusted for the boundary conditions $V(i, 0)$ and $V(i, M)$, which denotes M2. Alternatively, the system can be expressed as follows:

$$M_1 \cdot V_i = M_2 V_{i+1} + b$$

This formulation allows us to solve for V_i at each time step. By iterating backward through time, starting from the final condition, we can eventually determine the value of V at time $t = 0$, which represents the price of the option at the current time.

5.2.2 Boundary condition for each PDE

This paper divides the five scenarios of payment of maturity into three categories.

- One-touch Up(OTU)

Option that are knocked out directly, or those that are knocked out following a knock-in event, fall into the first category. On an observation date, if the knock-out level is breached, the option terminates. The payoff is $R_1 \times ti$, where R_1 is the one-month coupon payment. At a lower boundary, the asset price reaches 0 resulting in a worthless option. When an option expires without triggering a payout, a right boundary condition applies, and the terminal payoff is zero. Figure 5 showcases the three boundary conditions for OTU [10].

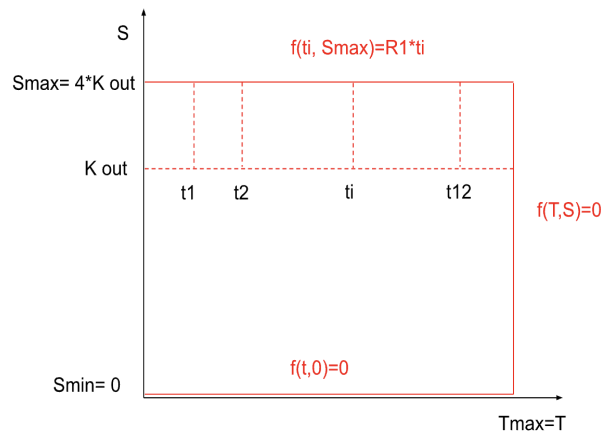


Fig. 5. PDE of OTU

- Double No-touch(DNT)

A second type of knock-out option involves two barriers, specifically an up-out and a down-out. As long as the asset remains within the knock-in and knock-out levels on the observation date, the holder receives the full coupon payment. In the event the price reaches zero or exceeds S_{max} , the option is knocked out, and no payout is made. The full coupon is paid if the price stays between the barriers. In the case of $S = 0$, the option's value is always zero, meaning that it is worthless. Figure 6 showcases the three boundary conditions for DNT [10].

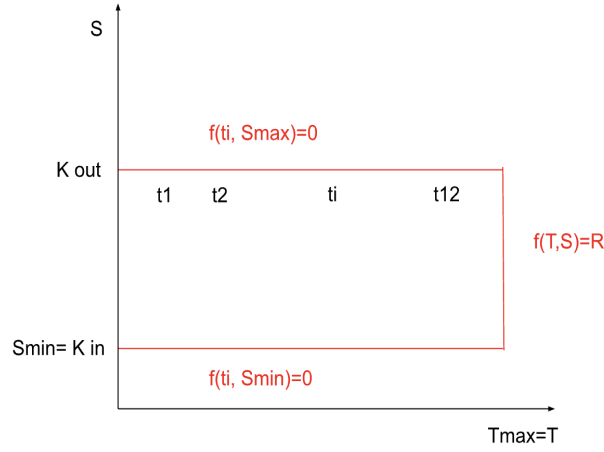


Fig. 6. PDE of DNT

- Double Knock-out Put(DKOP) - Up and Out Put(UOP)

A third type of knock-in involves an up-out and a down-in, which is equivalent to selling a put option with an up-out and a down-in. When the knock-in component is replaced by standard knock-out options, the pricing is simplified, while maintaining the same payoff and return. A strategy based on this result consists of selling an Up-Out put option while purchasing an Up-Out and Down-Out put option, with the same return as the third category. Boundary conditions are set for both selling the Up-Out Put and buying the Up-Out and Down-Out Put.

The PDE for the DKOP is similar to the PDE for the second category, with the main difference being that, at maturity, the payoff is that of a put option. On the right is a representation of the PDE for the UOP. When the index is knocked out, the payoff is zero, but if it is not, it becomes the payoff of a put option. The bottom boundary reflects the value of the put option if the asset's price reaches zero, which is $S_0 * e^{-r(T-t)}$. Figure 7 showcases the three boundary conditions for DKOP and UOP [10].

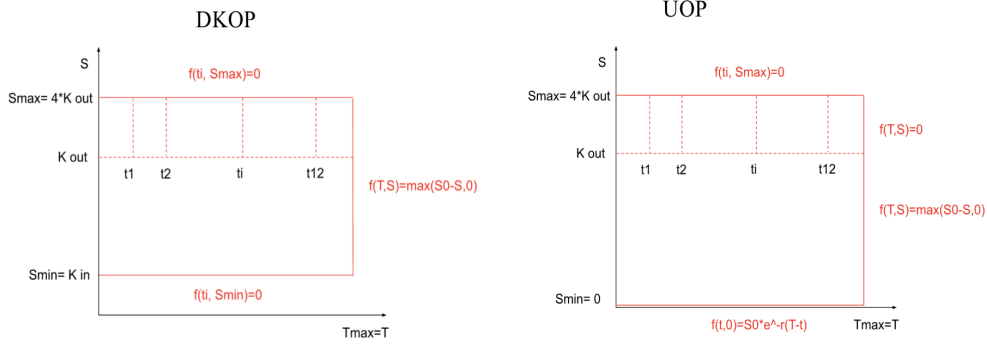


Fig. 7. PDE of DKOP(on the left) and UOP(on the right)

5.3 Monte Carlo Method

5.3.1 Overview

Monte Carlo Simulation (MCS) is extensively employed for pricing financial derivatives globally. Renowned for its simplicity and precision, MCS effectively handles complex pricing scenarios that arise in the financial markets. Unlike the traditional Black-Scholes model, MCS excels in addressing intricate derivative structures, offering a closer approximation of option prices in a more efficient timeframe [11]. In the subsequent section on MCS, this paper will present the mathematical foundations that underscore its applicability, establish parameters to demonstrate its efficacy in pricing Snowball Autocallables, and finally, evaluate the accuracy of the results by assessing convergence and estimating potential errors.

5.3.2 Monte Carlo Introduction

Monte Carlo Simulation refers to the simulation of the independent and random event several times to generate an expected statistical probability. It avoids problems that are too complicated to analyze by pure numerical analysis or mathematical induction. The mathematical proof of the validation for the Monte-Carlo simulation is as follows.

To construct a Monte-Carlo Simulation, it first needs a density function $\psi(x)$. Suppose that there is a set of possible events D . For every $x \in D$, it follows a density function $\psi(x)$. Then, the probability of the event could be formulated as,

$$P_r[x \in D] = \int_D \psi(x) dx,$$

where the sum of all the possibilities of the event should be equal to 1,

$$\int \psi_D(x) f(x) dx = 1.$$

Since already have the density function, the expected value (expectation) of the $f(x)$ concerning $\psi(x)$ could be calculated,

$$\mathbb{E}_\psi[f] = \mathbb{E}(f) = \int_D \psi(x)f(x)dx.$$

Also, for the variance, which is based on the calculation of the expectation,

$$\mathbb{V}[f] = \mathbb{E}[(f - \mathbb{E}(f))^2].$$

The variance can also be written as, by the existing theorem,

$$\mathbb{V}[f] = \mathbb{E}[f^2] - \mathbb{E}[f]^2.$$

With the variance and expectation, the covariance and correlation can be expressed as,

$$Cov[f, g] = \mathbb{E}[f, g] - \mathbb{E}[f]\mathbb{E}[g],$$

$$Corr[f, g] = \frac{Cov[f, g]}{\sqrt{\mathbb{V}[f]\mathbb{V}[g]}},$$

where the $Corr[f, g] \in [-1, 1]$.

After examining the probability for every independent event, this paper sets the "Running Times" and "Running Sum" of the simulation. The "Running Times" refers to how many times should the simulation process repeat and the "Running Sum" refers to the sum of the result generated by each time of simulation. After achieving the "Running Times", this paper takes the average of the "Running Sum" [12]. The process can be expressed by the mathematical formula below.

$$\bar{V}_N = \frac{1}{N} \sum_{i=1}^N f(x_i),$$

where \bar{V}_N stands for the "Running Average" and N is the "Running Times". $f(x_i)$ is the probability generated by each time where i stands for the order in the running times.

For the Snowball AutoCallable, as the paper introduced in the previous chapters, it is different from the traditional option pricing since it has knock-in and knock-out scenarios which complicates the payoff calculation and the expiration dates. This method could simulate and generate every path of the stock prices so that the interest of the Snowball AutoCallable can be calculated based on the known situation. For example, if the stock price is higher than the knock-out level and has never knocked in before this time spot, then the put option suspends and investors get the coupon interest in advance. Instead of setting various time and price boundaries in the traditional Black-Scholes model to tackle the situation like this, MCS provides a flexible and easy way to option pricing. Successful examples like Schwartz and Torous (1989) who used this model to calculate the mortgaged-back securities, and Boyle et al. (1997) who used this method to price the American option prove the feasibility and high-precision of the Monte-Carlo Simulation [13].

5.3.3 Brownian Motion introduction

This paper uses the Monte-Carlo to simulate the stock price for the following one-year S&P500 trend by Geometric Brownian motion model. The Geometric Brownian motion (GBM) refers to the continuous time-depending on the stochastic process where the logarithm of the randomly floating quantity follows the Brownian motion (also known as the Wiener process) [5], and a stochastic process represents a system that evolves in a probabilistic manner [5].

$$dS_t = \mu S_t dt + \sigma S_t dW_t$$

- S_t : the stock price at time
- μ : the drift term, representing the expected return of the stock
- σ : the volatility of the stock, indicating the degree of variation in returns
- dW : the increment of a Wiener process (or Brownian motion), capturing the random shocks to the stock price

This research uses Geometric Brownian Motion (GBM) as the basic model for Monte Carlo Simulation to generate future stock prices because it accurately reflects how stock prices behave in the real world. GBM models prices as a continuous process that always stays positive, incorporates both expected returns (drift) and volatility (risk) and ensures that the logarithm of returns is normally distributed. This makes it ideal for simulating realistic price paths over time.

5.3.4 Parameters setting up

Following the establishment of the foundational model for simulating the Snowball structure, this research delineates the specific methodologies employed for accurately pricing the Snowball product. The approach involves utilizing Geometric Brownian Motion (GBM) to simulate daily stock prices, with a focus on the stock's trajectory over 252 trading days, corresponding to one year. Specifically, the GBM method is applied iteratively for 252 periods to generate stock price paths for the S&P 500 from June 2024 to June 2025. Additionally, a ceiling and floor for stock movements are set at 1.1 and 0.9 times the current stock price, respectively, reflecting the limited volatility observed in the actual stock market. Although significant daily fluctuations can occur, such instances are infrequent and are not considered within the scope of the simulated stock prices, as they are deemed sudden and unpredictable. The Monte Carlo Simulation (MCS) methodology necessitates a substantial number of repeated experiments to enhance data accuracy; hence, the number of iterations is established at 5,000. This configuration yields 5,000 distinct stock price change paths for the S&P 500 over the specified year, with the visualization of the GBM paths presented below.

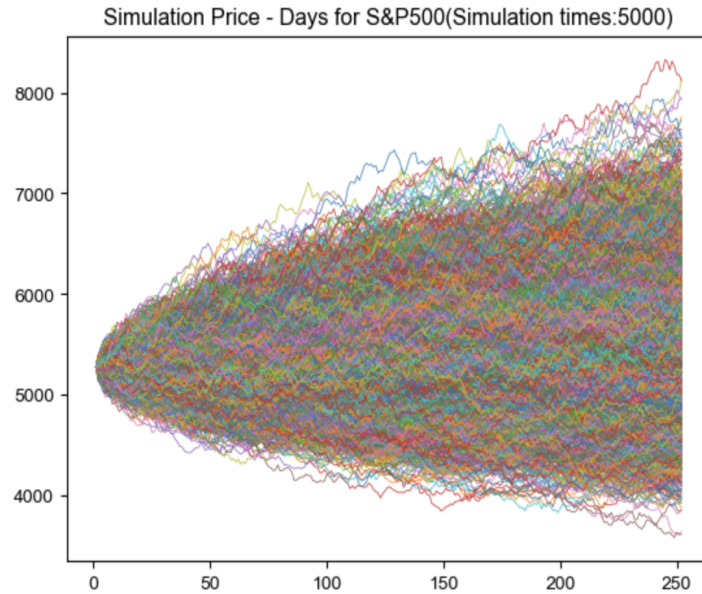


Fig. 8. Monte Carlo 5000-time Simulation

Next, based on the knock-in and knock-out levels we set, this paper determines the product's knock-in and knock-out and calculates the changes in net asset value (NAV) accordingly, by classifying all changes in NAV into the following four scenarios:

- Scenario 1: If there is a knock-out, the investor receives a discounted coupon payment.
- Scenario 2: If no knock-in and knock-out occurs, the full coupon is paid at maturity.
- Scenario 3: If only a knock-in occurs but the price recovers, the NAV remains 1.
- Scenario 4: If a knock-in occurs and the price does not recover, the NAV reflects a loss. This is the loss caused by the decline in asset value without the consideration of cash discount issues.

After considering the asset changes brought about by the knock-in and knock-out for each path, this research adds all the calculated NAV results to the Running Sum. After 5000 calculations, this research divides the Running Sum by 5000 to obtain the average NAV. This completes the pricing of the Snowball Autocallable, with the final price being 5308.226, which is 1.0058 times the initial price.

5.3.5 Validation of the Results

The variance and timeliness of the data are important measures [13] when dealing with the validation of the results. If the variance of the data decreases and approaches zero, it indicates that the data is converging, thus proving the simulation is reliable. Additionally, computational

time addresses the cost issue of the model. If the runtime is excessively long, it becomes nearly impossible for pricing models to deal with decades of historical data. Our results show that as the number of simulations increases, the variance of the data first increases then decreases, and converges to 0 finally. Although there are fluctuations, this may be due to insufficient simulation runs and the inherent randomness of the event. The research also found that within 5000 to 6000 times, the model achieves the highest efficiency (explain why this research chooses 5000 times as the experimental times), with both rather low and stable variance and shorter runtime. For times larger than this period the improvement of variance compared to the selected range is minimal, and the runtime starts to increase significantly.

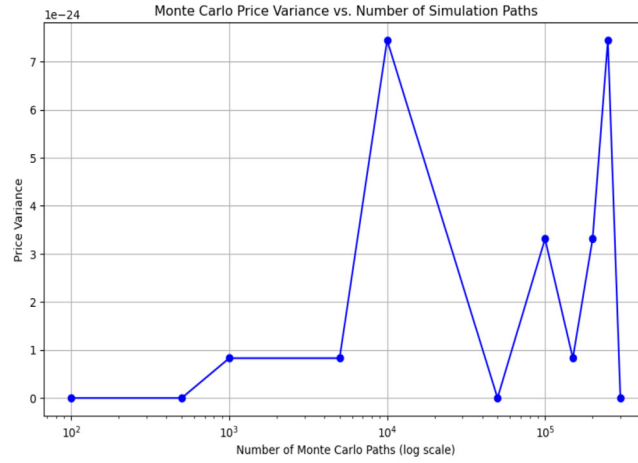


Fig. 9. Monte Carlo Results Convergence Test

Standardly, each stock price distribution, as an individual process, follows the logarithm of the Normal Distribution.

$$\bar{V}_N = \frac{1}{N} \sum_{i=1}^N V_i,$$

$$\bar{V}_N \xrightarrow{i.d.} \mathbf{N}(\mu, \frac{\sigma}{\sqrt{N}}).$$

Therefore, the running average of all these paths should also follow the normal distribution, which statistically measures the uncertainty (variance) of this simulation. Theoretically, the uncertainty in the simulation results is given by

$$\sqrt{\mathbb{V}[\bar{V}_N]} = \frac{\sigma}{\sqrt{N}}.$$

However, the real variance and expectation are not known since the whole process is simulated and we use this process to calculate the expectation (the averaged NAV). Thus, the estimation could only

be approached as the second variance formula we introduced before [12],

$$\bar{\sigma}_N = \sqrt{\frac{1}{N} \left(\sum_{i=1}^N V_i^2 \right) - \left(\frac{1}{N} \sum_{i=1}^N V_i \right)^2},$$

and this gives the standard definition of the standard error in the Monte-Carlo Simulation

$$\varepsilon_N = \frac{\bar{\sigma}_N}{\sqrt{N}}.$$

In conclusion, the Monte-Carlo Simulation offers a robust and precise simulation method for a structured product like the Snowball Autocallable product, with the flexibility to handle complex scenarios like knock-in and knock-out conditions. By leveraging the Geometric Brownian Motion model, we simulated the one-year price path which reflects the real financial market behaviors. Through the implementations of these models, we derived a comprehensive set of potential outcomes, facilitating a nuanced understanding of how varying conditions affect the net asset value (NAV) of the Snowball product. Our results demonstrate the importance of convergence and variance analysis, affirming that a higher number of simulations enhances the reliability of the pricing model. Notably, we established that 5000 simulation runs strike an optimal balance between computational efficiency and precision, yielding stable variance and acceptable runtimes. However, since Monte-Carlo Simulation is still an estimation method, it still has space for improvement.

6 Result Analysis

After introducing the two methods of pricing the Snowball option, we would like to address the comparison between Monte Carlo simulations and PDE methods for option pricing. The final results generated by the two methods are,

- Price generated by the PDE-based method is 5309.124.
- Price generated by Monte-Carlo Simulation is 5308.226.

By calculation, the similarity reaches 99.98%. Since this paper already addresses the feasibility and the reliability of the two methods significantly in the previous chapters, this paper would like to analyze the reason causing the distinctions between the two-pricing data by Monte Carlo simulations and the PDE-based method. After the analysis, we conclude the rationales from the following two reasons. For Monte Carlo simulations, the accuracy of our results depends heavily on the number of simulations conducted. Ideally, the number of simulations should be calculated by the formula below, the same formula as the error analysis,

$$\varepsilon_N = \frac{\bar{\sigma}_N}{\sqrt{N}}.$$

The errors should be expressed by the absolute value of $|f - \mu|$. Set the confidence interval as 95% (the confidence interval is the interval expected to contain the estimated values) and ω is the standard

deviation (Recall that the 95% of the distribution is within 1.96ω from the μ).

$$\mu - \frac{1.96\omega}{\sqrt{M}} < f < \mu + \frac{1.96\omega}{\sqrt{M}}.$$

By the calculation, the simulation times (with the confidence interval 95%) should be around 10000 times. However, our research decides to balance the efficiency and the precision of the data. Although the runtime in our case does not vary largely, for the companies or investors who deal with the ecumenical data (for example, 10 years), the increment of the calculation time matters significantly. At the same time, the companies or individual investors increasing time costs in pursuit of data accuracy should also be supported.

On the other hand, PDE methods rely on real-time stock prices of the underlying asset. Since future real-time prices cannot be predicted, this limitation introduces biases and discrepancies in the PDE results. In summary, while both approaches are effective, enhancing the simulation times in MC simulation and addressing the challenges of real-time price forecasting in PDE methods could further improve accuracy.

7 Risk Management

7.1 Internal risk analysis

To capture the different impacts of knock-out price, knock-in price, and sigma on the price of an option, we perform a sensitivity analysis.

7.1.1 Knock-in Level

Figure 10 shows the relationship between the knock-in level and the option value. Knock-in level of 0.85 is a reasonable compromise between risk and value. Option values below 0.85 remain high, but above 0.85, the value of the option drops sharply due to the increased risk of knock-in events. The knock-in probability at 0.85 is moderate, which means the option retains reasonable value but minimizes risk, thereby balancing potential returns with knock-in risks.

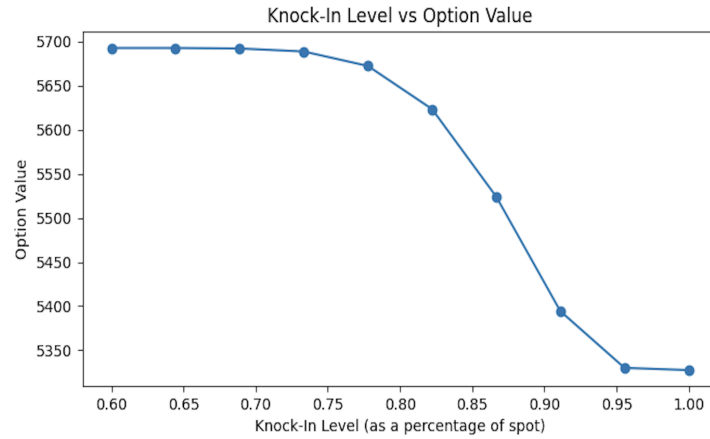


Fig. 10. The impact of knock-in level on option value

7.1.2 Knock-out Level

Figure 11 shows the relationship between the knock-out level and the option value. Using 1.03 as the knockout level balances the potential returns and risks of an early termination. This conservative margin reduces the chances of an option being knocked out prematurely by requiring the asset to exceed the spot price by 3%. It can be seen from the graph that increasing the knock-out level results in a higher option value, but the incremental gain diminishes as the knock-out level goes beyond 1.03. As long as the option remains active at this level, investors maintain a competitive payout regardless of market conditions while maximizing option value without taking on excessive risk.

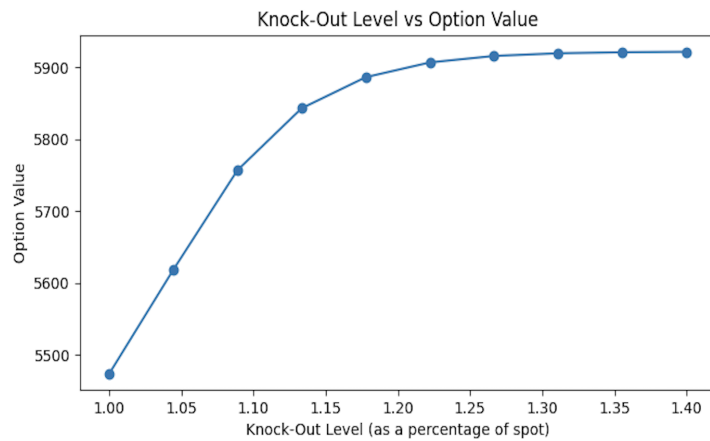


Fig. 11. The impact of knock out level on option value

7.1.3 Volatility

Figure 12 illustrates the negative correlation between volatility and option value, which is characterized by a decline in option value with increasing volatility. Volatility increases the likelihood that an asset will breach knock-in barriers or knock-out barriers as a result of greater uncertainty in its price movements. Option value declines as the probability of termination or knockout rises, leading to a lower expected payoff. When volatility is high in structured products, such as AutoCallables, the option's attractiveness and value are reduced.

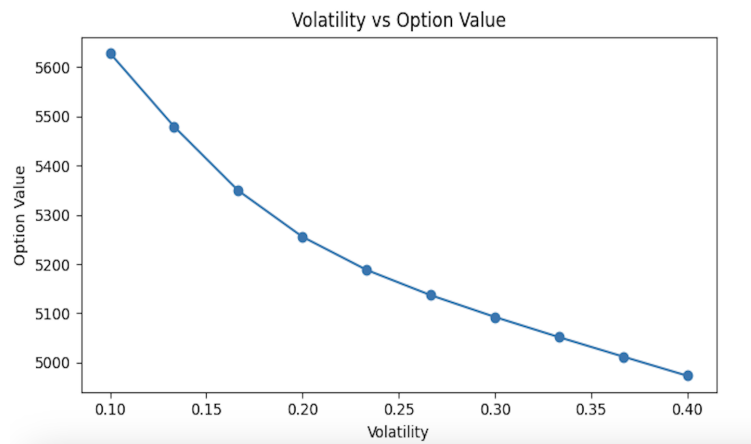


Fig. 12. The impact of volatility on option value

7.2 External risk analysis

7.2.1 Credit risk

Credit risk refers to the risk of a counterparty being unable to fully perform a contract. It exists not only in lending and bond investments, but also in wealth management products such as Snowball Auto-callable. Therefore, as the issuer of Snowball Auto-callable products, it is necessary to establish a credit rating system and conduct risk diversification and other measures to reduce the impact of credit risk on products.

7.2.2 Market trend risk

For investors, also the buyers of Snowball Auto-callable, who short volatility, choosing the time window of volatility within the duration is more conducive to profit; in only one of the five scenarios mentioned above, a loss of principal will occur. If the market falls sharply and breaks below the knock-in price, investors are responsible for all losses incurred by the S&P 500 as a result of the decline. Thus, this product is more suitable for investors in slightly volatile market conditions or a slowly rising market, based on the premise that the issuer and investors have a correct judgment of

the expected trend of the underlying assets. When the underlying assets change unexpectedly, the issuer may suffer losses from market risks without correct hedging operations and losses.

7.2.3 Liquidity risk

Liquidity risk refers to the risk that an asset cannot be traded quickly and smoothly or realized at a reasonable price because of the imbalance between buyers and sellers in the market. As this product does not allow early redemption and needs to trigger the knock-out conditions for automatic redemption, the issuer should guarantee the liquidity of the products during risk hedging to prevent the liquidity shortage of the issuer, and investors should also have a certain amount of idle funds to deal with the risk of irregular redemption of Snowball Auto-callable.

7.3 Risk management strategy-Delta hedging strategy

Delta refers to the sensitivity index of all types of financial derivatives to the underlying assets. It is defined as the rate of change of the option price with respect to the price of the underlying asset. It is the slope of the curve that relates the option price to the underlying asset price. [5]

In general,

$$\text{Delta} = \frac{\partial V}{\partial S}$$

here V refers to the price of Snowball Auto-callable and S refers to the price of S&P 500.

For this Snowball Auto-callable, delta and Gamma follows the graph below.

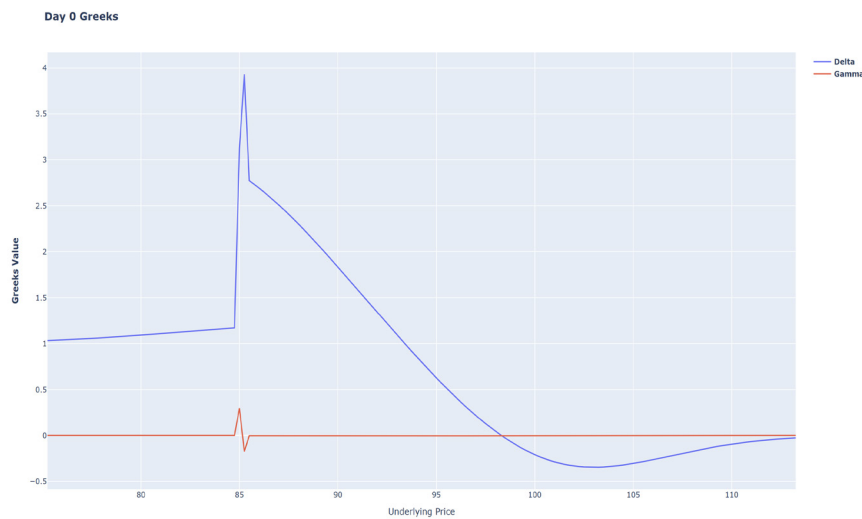


Fig. 13. The trend of greeks as S&P 500 moves

Figure 13 shows that when the price of the underlying asset falls close to the knock-in line, the

snowball Delta increases rapidly, whereas below the knock-in line, Delta approaches 1; that is, the snowball option almost fails after knocking in, and its value decreases rapidly.

The core method of delta hedging is to construct a portfolio and calculate the value of delta. When Delta is equal to 0, the value of the portfolio is stable; that is, when the value of the underlying asset fluctuates within a certain range, it is not influenced by the portfolio price.

For issuers, profit includes two main parts: time value and volatility. Time value refers to the degree of change of the option price in unit time; the greater the degree of change, the greater the operating space of the issuer. Volatility pertains to the fluctuation degree of the index, and Vega indicates the changing degree of the option price within the unit volatility. Similarly, the greater the degree of change, the greater the operating space of issuers.

Thus, in this case, the hedging operations conducted by the issuer primarily involve gradual position building and high selling and low buy transactions. This is achieved through continuous adjustments in the buying and selling of stock index futures to mitigate the impact of the delta value on the price of stock index futures. The objective is to maintain the stability of the portfolio value until the delta value approaches zero. This means that when the S&P 500 falls and Delta becomes larger, it is helpful to buy stock index futures to reduce the impact on prices. Risk hedging can be achieved through low buying and high selling operations.

In real transactions, it is important to balance between risk and cost; if the hedging frequency is too high, it will reduce hedging risk but increase hedging cost. If the hedging frequency is too low, it will increase the hedging risk. Therefore, it is necessary to establish risk exposure. When the index changes within the range of risk exposure, it is acceptable to the issuer; therefore, there is no need to conduct the hedging operation. When the fluctuation of the index exceeds risk exposure, the issuer needs to hedge immediately and realize risk hedging by buying and selling stock index futures. The interests of investors and issuers can be guaranteed through a reasonable risk-hedging strategy and risk-management mechanism.

8 Conclusion

In conclusion, the pricing results of our S&P 500-Linked Snowball Autocallable demonstrate a certain level of stability. It is an innovative product in terms of structural design, is feasible for issuance, and has potential investment value in a volatile market. In addition, it is subject to the same limitations as other auto-callable products, such as low liquidity and higher risk when compared to traditional fixed-income securities. The use of two pricing methods ensures reliable pricing and provides an in-depth mathematical derivation, thus filling a market gap. The paper has a limitation in that it does not justify the initial conditions, such as volatility, coupon rate and barrier levels, which are based on general market assumptions. It is not possible to fully verify the pricing's optimality, even after conducting a sensitivity analysis. In the future, research will be conducted on the feasibility of the product on the market, as well as stress tests on the product.

References

- [1] Cox, John C., Stephen A. Ross, and Mark Rubinstein. "Option Pricing: A Simplified Approach." *Journal of Financial Economics* 7, no. 3 (1979): 229-263.
- [2] Schwartz, Eduardo S. "The Valuation of Warrants: Implementing a New Approach." *Journal of Financial Economics* 4, no. 1 (1977): 79-93.
- [3] Boyle, Phelim P. "Options: A Monte Carlo Approach." *Journal of Financial Economics* 4, no. 3 (1977): 323-338.
- [4] Wilmott, Paul, Sam Howison, and Jeff Dewynne. 2010. *The Mathematics of Financial Derivatives : A Student Introduction*. Cambridge: Cambridge University Press, Dr.
- [5] Hull, John. 2012. *Options, Futures, and Other Derivatives*. Boston: Prentice Hall.
- [6] Lu, Xuewen. 2024. "Analysis of Stock Price Volatility Based on GARCH Family Models." *Scientific Journal of Economics and Management Research* 6, no. 3.
- [7] Deng, Geng, Joshua Mallett, and Craig McCann. 2011. "Modeling Autocallable Structured Products." *Journal of Derivatives & Hedge Funds* 17 (4): 326–40. <https://doi.org/10.1057/jdhf.2011.25>.
- [8] Evans, Lawrence C. 2010. *Partial Differential Equations*. Providence, R.I.: American Mathematical Society.
- [9] Victor Grigor'e Ganzha, and Evgenii Vasilev Vorozhtsov. 2017. *Numerical Solutions for Partial Differential Equations*. CRC Press.
- [10] Rich, Don R. 2000. *The Mathematical Foundations of Barrier Option-Pricing Theory*.
- [11] Meding, Isak, and Viking Zandhoff Westerlund. "Pricing European options with the Black-Scholes and Monte Carlo methods: a comparative study." (2022).
- [12] Jäckel, Peter. 2002. *Monte Carlo Methods in Finance*. Chichester, West Sussex, England: J. Wiley.
- [13] Jabbour, George M., and Yi-Kang Liu. 2011. "Option Pricing and Monte Carlo Simulations." *Journal of Business & Economics Research (JBER)* 3 (9).
- [14] Izvorski, Ivailo. 1998. "A Nonuniform Grid Method for Solving PDE's." *Journal of Economic Dynamics and Control* 22 (8-9): 1445–52. [https://doi.org/10.1016/s0165-1889\(98\)00020-7](https://doi.org/10.1016/s0165-1889(98)00020-7).

Ecological Dynamics and Stability Analysis of Predator-Prey Systems under the SEIR Infectious Disease Model

Junmeng Zhang^{1,a,*}, Shengtao Yan^{1,b}, Weiyou Xu^{1,c}, Xiaoyang Jiang^{1,d}

¹Ocean University of China, 238 Songling Road, Laoshan District, Qingdao, Shandong Province, 266100, China

a. 2249156816@qq.com, b. 723724836@qq.com, c. 30949471@qq.com, d. 1035809297@qq.com

*corresponding author

Abstract. This paper investigates the stability of the Leslie-Gower predator-prey model under the influence of the SEIR infectious disease model. By incorporating the SEIR infectious disease model, the predator population is divided into four states: susceptible, exposed, infected, and removed. A predator-prey dynamic model is then established. Through dimensionless processing and discretization methods, the equilibrium points of the model and their stability are analyzed. The eigenvalues of the Jacobian matrix are computed to determine the stability of the equilibrium points, and numerical simulations are used to demonstrate the dynamic behavior of the system under different parameter conditions. The results indicate that the predation rate and disease transmission rate have significant effects on the stability of the system. Reducing these two parameters appropriately can stabilize the system.

Keywords: SEIR Infectious Disease Model, Predator-Prey Model, Stability, Numerical Simulation

1 Introduction

The predator-prey model is one of the classical models in ecology used to describe population interactions. In traditional Lotka-Volterra and Leslie-Gower models, the interaction between predators and prey is simplified to the growth of prey and the predation behavior of predators [1-3]. However, in real ecological systems, predator populations may be affected by infectious diseases, which significantly alter the dynamic characteristics of the population. To more accurately describe this complex ecological phenomenon, researchers have proposed various extended models to better reflect the interactions between predator and prey populations and the spread of infectious diseases in actual ecosystems [4,5].

In recent years, the SEIR model, as a classic infectious disease model, has been widely used to describe the spread of diseases in populations [6,7]. The SEIR model, by introducing the exposed (E) state, can more accurately describe the transmission process of infectious diseases with an incubation period [8]. This paper builds upon the traditional Leslie-Gower predator-prey model by incorporating the SEIR infectious disease model, dividing the predator population affected by the disease into four states: susceptible (y_S), exposed (y_E), infected (y_I), and removed (y_R), and develops a new predator-prey model.

2 Model Construction

In the construction of the model, the following assumptions are made: the predation relationship between predators and prey follows the classical Leslie-Gower model [9,10]. Infectious diseases are transmitted within the predator population through contact, with the infection rate related to the frequency of contact between predator individuals. After being infected, the predation ability of the predator is significantly weakened. The specific model equations are as follows:

2.1 Dynamic Equation of the Prey Population

$$\frac{dx}{dt} = x \left(\frac{r_1}{1+ky} - \beta x - \frac{c_1(y_S+y_E+y_I)}{x+k_1} \right), \quad (1)$$

where x is the prey population size, and $y = y_S + y_E + y_I + y_R$ represents the total predator population. r_1 is the intrinsic growth rate of the prey, k is the impact coefficient of predators on the prey, β is the competition coefficient of the prey population, c_1 is the predation rate of predators on the prey, and k_1 is the protective coefficient of predators on the prey population.

2.2 SEIR Dynamic Equation of the Predator Population

Dynamic Equation of Susceptible Predators:

$$\frac{dy_S}{dt} = y_S \left(r_2 - \frac{c_2 y_S}{x+k_1} \right) - \alpha y_S y_I - d_1 y_S, \quad (2)$$

Dynamic Equation of Exposed Predators:

$$\frac{dy_E}{dt} = \alpha y_S y_I - \sigma y_E - d_2 y_E, \quad (3)$$

Dynamic Equation of Infected Predators:

$$\frac{dy_I}{dt} = \sigma y_E - e y_I - d_3 y_I, \quad (4)$$

Dynamic Equation of Removed Predators:

$$\frac{dy_R}{dt} = e y_I - d_4 y_R, \quad (5)$$

where y_S , y_E , y_I , and y_R represent the number of susceptible, exposed, infected, and removed predators, respectively. r_2 is the growth rate of susceptible predators, c_2 is the predation rate of predators on prey, α is the infection rate for susceptible predators to become exposed predators, σ is the rate at which exposed predators become infected predators, e is the recovery or removal rate of infected predators, and d_1, d_2, d_3, d_4 are the natural death rates of susceptible, exposed, infected, and removed predators, respectively. The system of differential equations is obtained by solving equations (1), (2), (3), (4), and (5):

$$\begin{cases} \frac{dx}{dt} = x \left(\frac{r_1}{1+ky} - \beta x - \frac{c_1(y_S+y_E+y_I)}{x+k_1} \right), \\ \frac{dy_S}{dt} = y_S \left(r_2 - \frac{c_2 y_S}{x+k_1} \right) - \alpha y_S y_I - d_1 y_S, \\ \frac{dy_E}{dt} = \alpha y_S y_I - \sigma y_E - d_2 y_E, \\ \frac{dy_I}{dt} = \sigma y_E - e y_I - d_3 y_I, \\ \frac{dy_R}{dt} = e y_I - d_4 y_R, \end{cases} \quad (6)$$

Let the initial values of each subpopulation at the start of the model, i.e., at $t = 0$, be given as $S(0)$, $E(0)$, $I(0)$, $R(0)$ for susceptible, exposed, infected, and removed predators, respectively, and $X(0)$ for the prey population size. These values will serve as the initial conditions for the system of differential equations.

3 Model Processing

3.1 Variable Substitution and Dimensionless Transformation

In order to simplify the analysis of the model, the original predator-prey model undergoes a dimensionless transformation. The following variable substitutions are considered:

$$T = r_1 t, \quad u = \frac{\beta}{r_1} x, \quad v_S = \frac{c_1 \beta}{r_1^2} y_S, \quad v_E = \frac{c_1 \beta}{r_1^2} y_E, \quad v_I = \frac{c_1 \beta}{r_1^2} y_I, \quad v_R = \frac{c_1 \beta}{r_1^2} y_R.$$

At the same time, new dimensionless parameters are introduced:

$$s = \frac{\beta k_1}{r_1}, \quad \gamma = \frac{r_2}{r_1}, \quad f = \frac{c_2}{c_1}, \quad d = \frac{k r_1^2}{c_1 \beta}, \quad \alpha' = \frac{\alpha c_1}{r_1}, \quad \sigma' = \frac{\sigma}{r_1}, \quad e' = \frac{e}{r_1}, \quad p_i = \frac{d_i}{r_1},$$

Based on the above variable substitutions, the original predator-prey model can be rewritten in dimensionless form as:

$$\begin{cases} \frac{du}{dT} = u \left(\frac{1}{1+d(v_S+v_E+v_I+v_R)} - u - \frac{v_S+v_E+v_I}{u+s} \right), \\ \frac{dv_S}{dT} = v_S \left(\gamma - \frac{f v_S}{u+s} \right) - \alpha' v_S v_I - p_1 v_S, \\ \frac{dv_E}{dT} = \alpha' v_S v_I - \sigma' v_E - p_2 v_E, \\ \frac{dv_I}{dT} = \sigma' v_E - e' v_I - p_3 v_I, \\ \frac{dv_R}{dT} = e' v_I - p_4 v_R. \end{cases} \quad (7)$$

3.2 Discretization of the Model

To discretize the dimensionless model, the forward Euler method is applied, converting the continuous-time model into a discrete-time model. Let the time step be ΔT , with the time point $T_n = n\Delta T$. The discretized model equations are as follows:

$$\begin{cases} u_{n+1} = u_n + \Delta T \cdot u_n \left(\frac{1}{1+d(v_{S,n}+v_{E,n}+v_{I,n}+v_{R,n})} - u_n - \frac{v_{S,n}+v_{E,n}+v_{I,n}}{u_n+s} \right), \\ v_{S,n+1} = v_{S,n} + \Delta T \cdot \left(v_{S,n} \left(\gamma - \frac{f v_{S,n}}{u_n+s} \right) - \alpha' v_{S,n} v_{I,n} - p_1 v_{S,n} \right), \\ v_{E,n+1} = v_{E,n} + \Delta T \cdot \left(\alpha' v_{S,n} v_{I,n} - \sigma' v_{E,n} - p_2 v_{E,n} \right), \\ v_{I,n+1} = v_{I,n} + \Delta T \cdot \left(\sigma' v_{E,n} - e' v_{I,n} - p_3 v_{I,n} \right), \\ v_{R,n+1} = v_{R,n} + \Delta T \cdot \left(e' v_{I,n} - p_4 v_{R,n} \right). \end{cases} \quad (8)$$

4 Existence of Equilibrium Points

To determine all equilibrium points of the model, the following system of equations needs to be solved:

$$\begin{cases} u = u \exp \left(\Delta T \cdot \left(\frac{1}{1+d(v_S+v_E+v_I+v_R)} - u - \frac{v_S+v_E+v_I}{u+s} \right) \right), \\ v_S = v_S \exp \left(\Delta T \cdot \left(\gamma - \frac{f v_S}{u+s} - \alpha' v_I - p_1 \right) \right), \\ v_E = v_E \exp \left(\Delta T \cdot \left(\alpha' v_S v_I - \sigma' - p_2 \right) \right), \\ v_I = v_I \exp \left(\Delta T \cdot \left(\sigma' v_E - e' - p_3 \right) \right), \\ v_R = v_R \exp \left(\Delta T \cdot \left(e' v_I - p_4 \right) \right). \end{cases} \quad (9)$$

The constants of this model yield the equilibrium points: $E_0(0,0,0,0,0)$, $E_I(u_I,0,0,0,0)$, and $E_2(u_2, v_{S,I}, v_{E,I}, v_{I,I}, v_{R,I})$, where:

$$v_{S,I} = \frac{(\gamma - p_1)(u_2 + s)}{f}, \quad v_{E,I} = \frac{\alpha' v_{S,I} v_{I,I}}{\sigma' + p_2}, \quad v_{I,I} = \frac{\sigma' v_{E,I}}{e' + p_3}, \quad v_{R,I} = \frac{e' v_{I,I}}{p_4},$$

And u_2 is the positive real root of the following quadratic equation:

$$\phi_2 u^2 + \phi_1 u + \phi_0 = 0,$$

where:

$$\begin{aligned} \phi_2 &= df(\gamma - p), \\ \phi_1 &= f(\gamma - p) + d(\gamma - p)(v_{S,I})^2 + f^2, \\ \phi_0 &= ds(\gamma - p)(v_{S,I})^2 + f(\gamma - p)s - f^2 \end{aligned}$$

For the zero equilibrium point $E_0(0,0,0,0,0)$ and the axial equilibrium point $E_I(u_I,0,0,0,0)$, their existence is self-evident. The following theorem demonstrates the existence of the equilibrium point $E_2(u_2, v_{S,I}, v_{E,I}, v_{I,I}, v_{R,I})$.

Theorem

If $v_{S,I} > 0$, and

$$ds(\gamma - p)(v_{S,I})^2 + f(\gamma - p)s < f^2,$$

then the model has a unique positive equilibrium point $E_2(u_2, v_{S,I}, v_{E,I}, v_{I,I}, v_{R,I})$, satisfying:

$$\phi_2 > 0 \quad \phi_2 > 0 \quad \phi_2 > 0, \quad \phi_I > 0, \quad \phi_0 < 0 \quad \phi_I > 0 \quad \phi_I > 0 \quad \phi_0 < 0 \quad \phi_0 < 0$$

Proof

The roots of the quadratic equation satisfy the root-coefficient relationship:

$$m + n = -\frac{\phi_I}{\phi_2}, \quad mn = \frac{\phi_0}{\phi_2}.$$

Since $r_2 > E$, it follows that $\gamma > p$, thus:

$$\phi_I = dsf(\gamma - p) + d(\gamma - p)(v_{S,I})^2 + f^2 > 0, \quad \phi_2 = df(\gamma - p) > 0.$$

If $\phi_0 > 0$, then $m + n < 0$, and $mn < 0$. The equation will have two negative real roots only when $\Delta = \phi_I^2 - 4\phi_2\phi_0 > 0$; if $\Delta < 0$, the equation has no real roots. In this case, model (9) has no positive equilibrium points.

If $\phi_0 < 0$, then $m > 0$, and $n > 0$. The equation has one positive real root and one negative real root only when $\Delta = \phi_I^2 - 4\phi_2\phi_0 > 0$. In this case, model (9) has a unique positive equilibrium point $E_2(u_2, v_{S,I}, v_{E,I}, v_{I,I}, v_{R,I})$.

5 Stability of the Equilibrium Points

5.1 Calculation of the Jacobian Matrix

To analyze the stability of the equilibrium points, the Jacobian matrix at the corresponding points is calculated, and its eigenvalues are solved to analyze stability [11-13].

This section focuses on the stability analysis of the equilibrium point $E_2(u_2, v_{S,I}, v_{E,I}, v_{I,I}, v_{R,I})$, which represents the coexistence of all predator and prey populations. The corresponding Jacobian matrix J is the linear approximation of the model near this point. The elements of the matrix are the partial derivatives of each equation in the model with respect to all variables:

$$J = \begin{bmatrix} \frac{\partial u_n}{\partial u_{n+1}} & \frac{\partial u_n}{\partial v_{S,n+1}} & \frac{\partial u_n}{\partial v_{E,n+1}} & \frac{\partial u_n}{\partial v_{I,n+1}} & \frac{\partial u_n}{\partial v_{R,n+1}} \\ \frac{\partial v_{S,n}}{\partial u_{n+1}} & \frac{\partial v_{S,n}}{\partial v_{S,n+1}} & \frac{\partial v_{S,n}}{\partial v_{E,n+1}} & \frac{\partial v_{S,n}}{\partial v_{I,n+1}} & \frac{\partial v_{S,n}}{\partial v_{R,n+1}} \\ \frac{\partial v_{E,n}}{\partial u_{n+1}} & \frac{\partial v_{E,n}}{\partial v_{S,n+1}} & \frac{\partial v_{E,n}}{\partial v_{E,n+1}} & \frac{\partial v_{E,n}}{\partial v_{I,n+1}} & \frac{\partial v_{E,n}}{\partial v_{R,n+1}} \\ \frac{\partial v_{I,n}}{\partial u_{n+1}} & \frac{\partial v_{I,n}}{\partial v_{S,n+1}} & \frac{\partial v_{I,n}}{\partial v_{E,n+1}} & \frac{\partial v_{I,n}}{\partial v_{I,n+1}} & \frac{\partial v_{I,n}}{\partial v_{R,n+1}} \\ \frac{\partial v_{R,n}}{\partial u_{n+1}} & \frac{\partial v_{R,n}}{\partial v_{S,n+1}} & \frac{\partial v_{R,n}}{\partial v_{E,n+1}} & \frac{\partial v_{R,n}}{\partial v_{I,n+1}} & \frac{\partial v_{R,n}}{\partial v_{R,n+1}} \end{bmatrix}$$

Let $v_E + v_I + v_R + v_S$ be denoted as v_A ; let $v_E + v_I + v_S$ be denoted as v_P . The values of the elements in the first row of the matrix are as follows:

$$\begin{aligned}
j_{11} &= \frac{-u(-v_p + (s+u)^2) + (s+u)^2}{(s+u)^2} \exp\left(\frac{s-u(s+u)(dv_A+I) + u - (dv_A+I)v_p}{(s+u)(dv_A+I)}\right) \\
j_{12} &= -u\left(\frac{d}{(dv_A+I)^2} + \frac{I}{s+u}\right) \exp\left(-u + \frac{I}{dv_A+I} - \frac{v_p}{s+u}\right) \\
j_{13} &= -u\left(\frac{d}{(dv_A+I)^2} + \frac{I}{s+u}\right) \exp\left(-u + \frac{I}{dv_A+I} - \frac{v_p}{s+u}\right) \\
j_{14} &= -u\left(\frac{d}{(dv_A+I)^2} + \frac{I}{s+u}\right) \exp\left(-u + \frac{I}{dv_A+I} - \frac{v_p}{s+u}\right) \\
j_{15} &= -du \exp\left(\frac{s-u(s+u)(dv_A+I) + u - (dv_A+I)v_p}{(s+u)(dv_A+I)}\right) \frac{I}{(dv_A+I)^2}
\end{aligned}$$

The remaining elements can be solved in a similar way and are not listed here.

5.2 Eigenvalue Calculation and Stability Analysis

The relevant parameters for the equations are assigned reasonable values. The eigenvalues of the above Jacobian matrix are solved using Python software. The five obtained eigenvalues are:

$$\lambda_1 = 1.0000, \lambda_2 = 0.0899, \lambda_3 = 1.0000, \lambda_4 = 0.9999, \lambda_5 = 0.0000$$

$\lambda_1 = \lambda_3 = 1.0000$ indicates that the system is neutrally stable along these directions, but not asymptotically stable, which may lead to periodic behavior or sustained oscillations.

$\lambda_2 = 0.0899$, which is less than 1, shows that this direction is asymptotically stable.

$\lambda_4 = 0.9999$ suggests that the system is close to neutral stability along this direction, and theoretically, periodic behavior or boundary stability may occur.

$\lambda_5 = 0.0000$ means there is no change along this direction, indicating that the system is at rest in this direction.

The above analysis shows that the system has not fully reached asymptotic stability, but is instead in a state of boundary stability. Periodic fluctuations or sustained oscillations may occur along certain directions.

6 Image Analysis

6.1 Time Series Plot of the System

By plotting the system's time series, the changes in the predator or prey populations during the evolutionary process can be shown, allowing observation of whether periodic behaviors or other dynamic features exist.

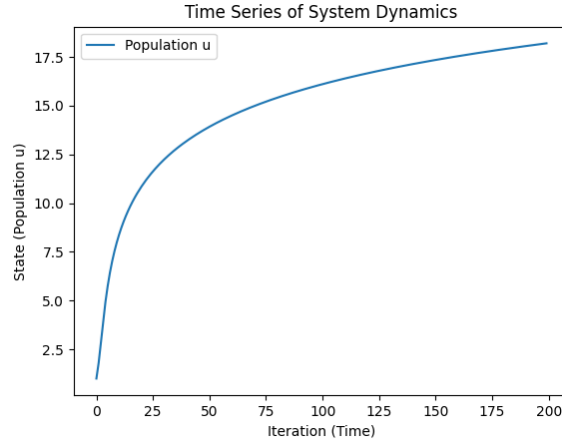


Fig. 1. Time Series Plot of the System

Figure 1 shows the trajectory of the system state u as it changes with the increasing number of iterations over time. The system state grows rapidly in the early stages and then tends to stabilize. This may be related to the rapid reproduction of predators and the swift spread of infectious diseases.

Additionally, there is no apparent periodic oscillation or chaotic phenomenon in the plot, indicating that the system's dynamic behavior is monotonic and tends toward a stable equilibrium point. This is consistent with the theoretical analysis results, indicating the system's asymptotic stability.

For different parameter settings, especially with different combinations of the predation rate c_I and disease transmission rate α , numerical simulations show that the system's dynamic characteristics change significantly. As the predation rate increases, the predator population grows more rapidly in the initial stages. However, in the long term, the predator population tends to stabilize, while the prey population gradually decreases [14,15]. This is because the high predation pressure on the prey leads to a reduction in the prey population. However, when the prey population decreases to a certain threshold, the growth of the predator population slows, and the system approaches stability. When the disease transmission rate is high, the dynamic behavior of the predator population becomes unstable. This is due to the accelerated infection of predators, which increases their mortality rate, and the prey population, in turn, shows an increasing trend, leading the system to become overall unstable [16,17].

These numerical simulation results align with the characteristic value calculations from the theoretical analysis, further verifying the system's dynamic behavior under different parameter conditions. The parameters c_I and α are critical control factors for the system's stability, determining the long-term stability of the predator-prey-disease system.

6.2 Bifurcation Diagram

To better observe the system's long-term behavior and potential bifurcation phenomena, this study plots a bifurcation diagram, changing a specific parameter and observing the system's dynamics at different parameter values [18]. The following diagram shows the state changes of the system when the predation rate is altered.

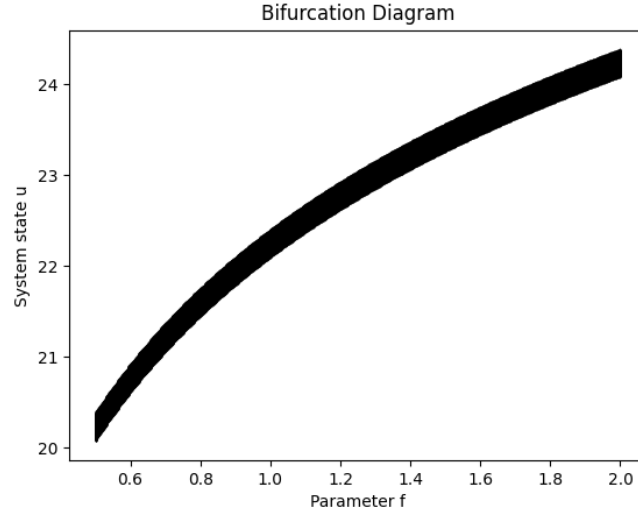


Fig. 2. Bifurcation Diagram of the System

From Figure 2, it can be seen that the system state u shows a monotonically increasing trend as the predation rate f increases, with a smooth curve and no significant bifurcation. This suggests that, within this parameter range, the system is relatively stable, without any periodic oscillation or chaotic behavior.

This may also indicate that the range of variation for the parameter f is not large enough to cause the system to transition from a stable state to other dynamic states. The smooth trend of the system state u with changes in the predation rate f suggests that the impact of this parameter on the system's overall state is gradual and does not immediately trigger unstable dynamic behavior.

7 Conclusion Summary

This study investigates the dynamic behavior and stability of a predator-prey system under the influence of the SEIR model. Through dimensional analysis and numerical simulations, the model's equilibrium points were analyzed, and system stability under different parameter conditions was explored using the characteristic values of the Jacobian matrix.

Numerical simulation results show that the predation rate c_l and disease transmission rate α are the key parameters influencing the system's dynamic behavior. At higher predation rates, the predator population grows more rapidly, but simultaneously, the prey population decreases quickly, leading the system to become unstable. Higher disease transmission rates also exacerbate the infection of predators, increasing system instability. Reducing the predation rate and disease transmission rate can effectively slow the system's fluctuations and make the system asymptotically stable.

Moreover, through simulations with different parameter combinations, this study found that the predator population's exposure-to-infection conversion rate σ and the predator's recovery or removal rate e also have significant effects on the system's dynamic behavior. A higher conversion rate accelerates the infection process of predators, causing dramatic population

fluctuations, while increasing the recovery rate of predators helps the system approach equilibrium.

Declaration

This project is supported by the 2024 Undergraduate Research and Development Program of Ocean University of China.

References

- [1] Duque, C., Rosales, R., & Sivoli, Z. (2023). Qualitative analysis of the dynamics of a modified Leslie-Gower predator-prey model with diffusion. *Ciencia E Ingenieria*, 44(3), 367-376.
- [2] Korobeinikov, A. (2001). A Lyapunov function for Leslie-Gower predator-prey models. *Applied Mathematics Letters*, 14(6), 697-699. [http://doi.org/10.1016/S0893-9659\(01\)80029-X](http://doi.org/10.1016/S0893-9659(01)80029-X)
- [3] Chen, L. J., & Chen, F. D. (2009). Global stability of a Leslie-Gower predator-prey model with feedback controls. *Applied Mathematics Letters*, 22(9), 1330-1334. <http://doi.org/10.1016/j.aml.2009.03.005>
- [4] Haque, M., & Venturino, E. (2008). Effect of parasitic infection in the Leslie-Gower predator-prey model. *Journal of Biological Systems*, 16(3), 425-444. <http://doi.org/10.1142/S0218339008002642>
- [5] GAN, W. (2007). A diffusive predator-prey model with disease in the predator. *Journal of Yangzhou University*, 10(3), 11-14.
- [6] Gurova, S. M. (2019). A Predator-Prey Model with SEIR and SEIRS Epidemic in the Prey. In M. D. Todorov (Ed.) *APPLICATION OF MATHEMATICS IN TECHNICAL AND NATURAL SCIENCES* (2164, pp.). 11th International Conference on Promoting the Application of Mathematics in Technical and Natural Sciences (AMiTaNS).
- [7] YANG, Y., LI, J., & ZHAO, W. (2009). A Predator-prey SEIR Epidemic Model with Infected Predator. *Mathematics in Practice and Theory*, 39(17), 104-108.
- [8] Xue, C. (2015). Global stability of SEIR epidemic model with disease in predator. *Computer Engineering and Application*, 51(23), 74-77.
- [9] P. H. LESLIE, J. C. GOWER, The properties of a stochastic model for the predator-prey type of interaction between two species, *Biometrika*, Volume 47, Issue 3-4, December 1960, Pages 219–234, <https://doi.org/10.1093/biomet/47.3-4.219>
- [10] P. H. LESLIE, J. C. GOWER, THE PROPERTIES OF A STOCHASTIC MODEL FOR TWO COMPETING SPECIES, *Biometrika*, Volume 45, Issue 3-4, December 1958, Pages 316–330, <https://doi.org/10.1093/biomet/45.3-4.316>
- [11] Georgescu, P., Hsieh, Y. H., & Zhang, H. (2010). A Lyapunov functional for a stage-structured predator-prey model with nonlinear predation rate. *Nonlinear Analysis-Real World Applications*, 11(5), 3653-3665. <http://doi.org/10.1016/j.nonrwa.2010.01.012>
- [12] LI, Z. (2009). Permanence and global attractivity of a discrete Leslie-Gower predator-prey model with feedback controls. *Journal of Fuzhou University*, 37(3), 312-316.
- [13] LIU, Q. (2006). Persistence and Global Stability for a Delayed Predator-prey System. *Journal of Hebei University. Natural Science Edition*, 26(3), 238-241.

- [14] Ramesh, P., Sambath, M., Mohd, M. H., & Balachandran, K. (2021). Stability analysis of the fractional-order prey-predator model with infection. *International Journal of Modelling and Simulation*, 41(6), 434-450. <http://doi.org/10.1080/02286203.2020.1783131>
- [15] Wang, F., & Liu, J. (2002). Periodic Solution and Stability for a Class of Prey-Predator Systems. *Journal of Sichuan Normal University. Natural Science Ed.*, 25(3), 270-274.
- [16] Zhang, L., & Wu, S. (2014). Global behavior of solutions for a modified Leslie-Gower predator-prey system with diffusion. *Journal of Shandong University. Natural Science*, 49(1), 86-91.
- [17] Zhou, J. (2014). Global Asymptotical Stability for a Diffusive Predator-Prey System with Modified Leslie-Gower Functional Response. *Journal of Southwest University. Natural Science Edition*, 36(7), 53-57.
- [18] Liu, X., Li, J., & Li, H. (2012). Dynamics of a Discrete Leslie-Gower System with Allee Effect. *Journal of Henan Normal University. Natural Science*, 40(5), 23-27.

A New Parallel XY Nanopositioning Platform Design

Zhengyu Qi

Liaoning University, Department of Physics, No. 66, Chongshan Middle Road, Huanggu District,
Shenyang City, China

PKMQi2004@163.com

Abstract. With the development of high-performance linear platforms for producing computer hard disk drive read/write heads and medical devices, nanopositioning systems are becoming increasingly important and have great application prospects in fields such as laser mirror positioning and high-resolution spectrometers. The current nanopositioning systems have a restricted single-axis travel range because of their physical design and the integration process involved. By utilizing established flexible mechanisms in the workplace, a novel large-range nanopositioning system was developed through SolidWorks modeling, with its accuracy confirmed via ANSYS simulation. This design aims to enhance the practical use of large-range nanopositioning systems.

Keywords: Planar motion stage, Parallel robots, Precision, Fabrication.

1 Introduction

A nanopositioning system is a mechatronic motion system with nanometer-scale moving mass [1, 2]. It includes bearings, actuators and drivers, sensors and electronic devices, as well as feedback control implemented on microcontrollers. The advantages of nanopositioning systems lie in their precision, stability, speed, and versatility, which make them indispensable tools in many high-tech fields [3-5].

Nanopositioning bearings[6] offer several benefits, including exceptional precision, minimal friction, and an extended lifespan. Due to the limited number of mechanical parts used in their construction, both friction and backlash are kept to a minimum, which aids in achieving greater accuracy. Additionally, these bearings are designed with considerations for wear and dynamic performance. In contrast to conventional spindle-based technologies, voice coil actuators and magnetic linear drivers provide notable advantages. Moreover, magnetic levitation bearings are particularly well-suited for use in vacuum conditions, giving them a significant edge over traditional air bearings. A nanopositioning actuator[7] is a device designed for precise multi-axis positioning and movement at the nanoscale level. Typically, it consists of several nano-displacement stages and rotational platforms that facilitate the nanoscale manipulation of samples or devices in various directions. This type of system finds widespread use not only in mechanical engineering but also plays an essential role in areas such as semiconductor manufacturing, microscopy, laser technology, and automated production systems. Nanopositioning sensors[8] are mainly used for accurate measurement and positioning at the nanoscale level. These devices take advantage of the unique properties of nanomaterials, including their high specific surface area, distinctive optical characteristics, and advantageous diffusion abilities, to achieve precise control and measurement of tiny objects. Feedback control technology[9] plays a crucial role in nanopositioning, especially in improving both accuracy

and stability of positioning. By designing appropriate signal acquisition and control hardware circuits, along with choosing effective feedback control methods, it is possible to significantly reduce the effects of environmental disturbances on positioning stability.

Non-contact bearings[10] eliminate the friction problems present in traditional bearings, thereby reducing wear and heat generation, improving system stability and lifespan. Due to the absence of physical contact, non-contact bearings can achieve higher positioning accuracy and reduce errors caused by mechanical wear. To facilitate the motion orientation of the stage, aerostatic bearings and maglev bearings are typically employed [11]. However, one notable disadvantage of flexible bearings is that their movement range is limited, which restricts the overall movement capability of nanopositioning systems equipped with flexible bases. This limitation arises from the inherent design characteristics of flexible bearings, which are engineered to provide a certain degree of compliance and adaptability in response to external forces. While this flexibility can be advantageous in accommodating misalignments or absorbing vibrations, it simultaneously constrains the extent to which these systems can achieve precise positioning. Nanomotors[12, 13] typically generate very small forces, which limits their ability to effectively or rapidly move larger objects. This constraint is due to the nanoscale at which these motors function; their size is measured in nanometers, making them more suitable for manipulating molecules and particles rather than larger items. In practical scenarios like drug delivery systems within biomedical engineering, the capacity of nanomotors to transport therapeutic agents is vital. However, when they encounter larger biological structures or tissues, their efficiency significantly decreases because of inadequate force production. Moreover, environmental factors such as viscosity and friction at micro- and nanoscale levels can further hinder their movement abilities. The dimensions of actuators[14] are typically constrained by the capabilities of manufacturing technologies, making it more challenging to produce smaller versions. This constraint is due to various factors that are intrinsic to the production methods employed for these components. For example, as an actuator's size diminishes, the level of precision needed in its construction increases markedly. To meet the required specifications at a reduced scale, advanced techniques like micro-machining or additive manufacturing often become essential. The sensors used in nanopositioning systems may struggle with their dynamic range, making it difficult to detect both large and small signals at the same time. This challenge stems from the fundamental properties of the sensor technology, which typically establishes a specific detection threshold. Many sensors are optimized for a particular input level range; however, when they encounter signals that differ greatly in size—common in precision measurement scenarios—their data capture capabilities can be hindered. For example, if a nanopositioning system needs to track tiny positional shifts while also monitoring larger environmental disturbances or variations, the significant difference between these signal magnitudes can result in saturation or clipping. In such cases, smaller signals might become lost amid larger ones, leading to potential loss of vital information essential for accurate control and feedback processes. At the nanoscale[15], the noise from both environmental factors and electronic components can significantly influence measurement outcomes. Environmental disturbances may stem from several sources, including thermal variations, electromagnetic interference, and vibrations in surrounding areas. These elements can introduce inconsistencies into the sensitivity of measurements; for instance, even minor temperature fluctuations can change material characteristics or impact molecular interactions. Additionally, the electronic noise produced by sensors plays a vital role in ensuring measurement accuracy. Some of this noise might mask weak signals that are essential for precise readings at the nanoscale. The interplay between various types of noise complicates data interpretation, necessitating advanced signal processing techniques to derive valuable insights from raw data. As researchers aim for

greater precision in nanotechnology fields like drug delivery systems or nanoscale electronics, the issues related to environmental and electronic noise become more pronounced. Thus, it is not only important to comprehend these influences to enhance measurement methods but also essential to develop more robust sensors capable of functioning effectively under diverse conditions while minimizing errors due to external disruptions. At the nanoscale, various factors like thermal fluctuations, electromagnetic interference, and quantum effects can disrupt signal transmission. Such disruptions may result in data loss or inaccuracies in communication systems that depend on precise information transfer. For example, in nanotechnology fields such as molecular electronics or quantum computing, even slight disturbances can greatly affect the integrity of transmitted signals. The actuators and sensors in a nanopositioning system often necessitate specific environmental conditions for optimal functionality, including regulated temperature, humidity, and vibration isolation. These elements are critical since even slight variations can greatly affect the precision and accuracy of the positioning mechanism. For example, changes in temperature may cause materials within the components to expand or contract thermally, potentially leading to misalignment or positioning errors. Additionally, when these components are incorporated into a larger system or application, compatibility issues might emerge due to differing communication protocols or power needs. Each actuator and sensor could function on varying voltage levels or data transmission standards, which may require extra interfaces or converters for smooth integration. Moreover, electromagnetic interference from adjacent equipment could disrupt sensor readings and actuator efficiency if not properly addressed. Beyond these technical hurdles, it is crucial to take into account the overall design architecture of the nanopositioning system. The configuration must not only fit the physical dimensions of each component but also meet their operational requirements within an integrated framework. This involves ensuring adequate shielding against external disturbances as well as providing sufficient space for maintenance access. In summary, tackling these environmental prerequisites and compatibility challenges is essential for maximizing performance in a nanopositioning system while reducing potential disruptions during its operation.

Based on the existing flexible mechanism, a novel parallel motion XY nanopositioning platform is established via modeling and simulation, and the validity of the physical system is verified [16]. The design of the physical system, which includes bearings, actuators, and sensors, enables a broad XY nanoscale positioning capability. The bearing utilizes a parallel kinematic bending mechanism in the XY plane. By preventing geometric over-constraints, it achieves significant geometric decoupling between the two axes of motion. This design allows for actuator isolation that facilitates the use of large-stroke single-axis actuators and supports various endpoint sensing methods with widely used sensors. These features enable the proposed nanoscale positioning system to achieve a movement range of $10\text{mm} \times 10\text{mm}$.

2 Flexible Beam

In order to achieve the large range XY nanopositioning system, bilateral simple parallelogram flexures (BSPF) and bilateral compound parallelogram flexures (BCPF) have been widely used. This section will derive the stiffness of the two structures. The structures of simple parallelogram flexures (SPF) and compound parallelogram flexures (CPF) are shown in the Figure 1.

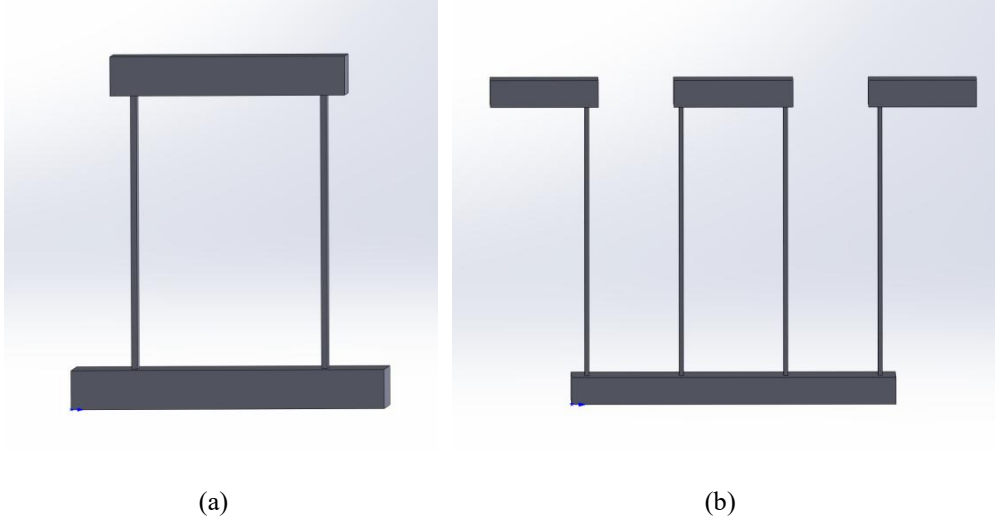


Fig. 1. Parallelogram flexures. (a) Simple parallelogram flexures (SPF). (b) Compound parallelogram flexures (CPF).

The stiffness of BSPF and BCPF is derived from the CPF model shown in Figure 2 [17].

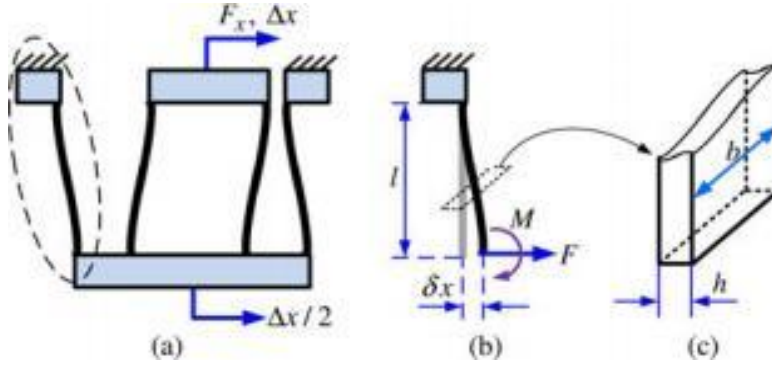


Fig. 2. CPF model. (a) Deformation of a CPF. Dimensions of (b) one flexure and (c) cross section [17].

When an outward force F_x is applied along the X-axis to the CPF, as illustrated in Figure 2(a), it results in a deformation of the CPF's shape. The structure experiences both a force F and a moment M , leading to bending. From this scenario, we can derive the following relationship:

$$0 = \frac{Fl^2}{2EI} - \frac{Ml}{EI} \quad (1)$$

$$\delta_x = \frac{Fl^3}{3EI} - \frac{Ml^2}{2EI} \quad (2)$$

In this equation, δ_x represents the X-axis displacement of a curved body [see Figure 2(b)], E is the Young's modulus of the material, and $I = \frac{bh^3}{12}$ is the moment of inertia about the neutral axis of the cross-section [see Figure 2(c)].

Solving the two equations above allows the generation:

$$F = \frac{2M}{l} \quad (3)$$

$$\delta_x = \frac{Fl^3}{12EI} \quad (4)$$

Given that the lengths of the four curved structures are all equal to l , it follows that $\delta_x = \frac{\Delta x}{2}$, where Δx indicates the displacement on one side of the CPF. Consequently, we can compute the stiffness of the CPF in relation to its output direction:

$$K = \frac{F_x}{\Delta x} = \frac{2F}{2\delta_x} = \frac{Eb h^3}{l^3} \quad (5)$$

The stiffness of the CPF can be obtained by analyzing the stiffness of BSPF and BCPF:

$$K_{\text{BSPF}} = \frac{4Eb h^3}{l^3} \quad (6)$$

$$K_{\text{BCPF}} = \frac{2Eb h^3}{l^3} \quad (7)$$

3 Justification of stiffness of flexible structures

In this section, the stiffness formulas of the BSPF and BCPF from the previous section are validated through SolidWorks modeling and ANSYS simulation. The relationship between structural forces and deformation is illustrated through modeling and simulation. This analysis provides an in-depth examination of multiple factors influencing how structures respond to applied loads. By performing computations, intricate models are created to replicate real-world scenarios. These simulations take into account aspects such as material characteristics, geometric arrangements, boundary conditions, and loading situations.

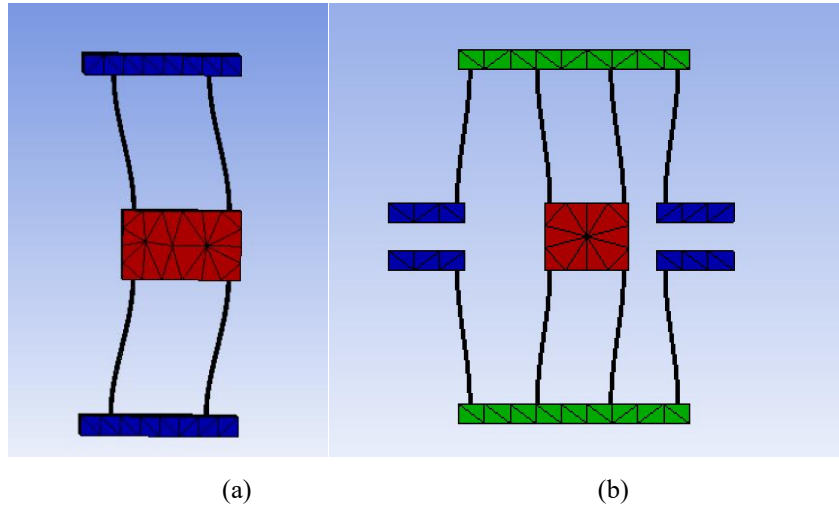


Fig. 3. Deformation of two structures. (a) Deformation of BSPF. (b) Deformation of BCPF.

3.1 Bilateral simple parallelogram flexures

Assuming $l=70\text{mm}$, $b=15\text{mm}$, $h=1.5\text{mm}$ and calculating the stiffness theoretical value of bilateral simple parallelogram flexures through the deduction of the formula, it is $4.1917 \times 10^4 \text{ N/m}$.

Through modeling and simulation, the relationship between structural forces and deformation is shown as follows:

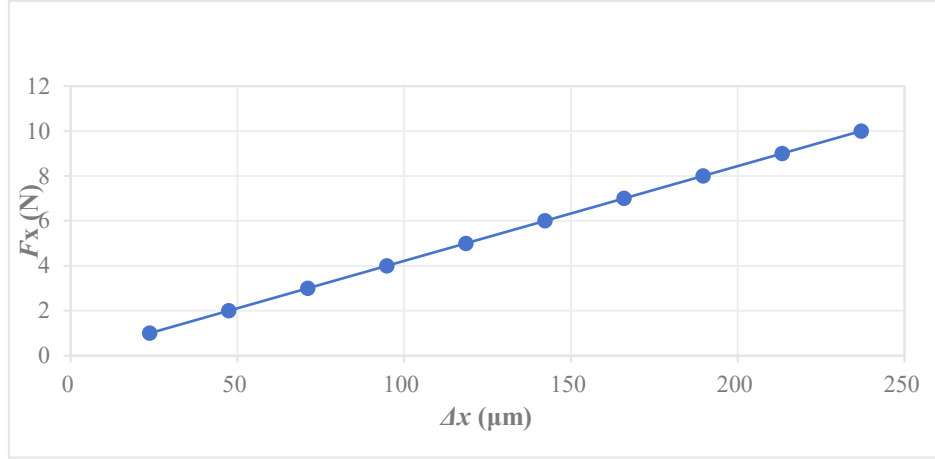


Fig. 4. The relationship between force and deformation of bilateral simple parallelogram flexures.

Through the process of simulation fitting, the actual stiffness value for the bilateral simple parallelogram flexures (BSPF) has been established at $4.2198 \times 10^4 \text{ N/m}$. This measurement is vital for comprehending the mechanical behavior and load-bearing capabilities of such foundations in engineering contexts. The relative error associated with this determination is found to be 0.67%, reflecting a high degree of accuracy in the simulation methodology. The minor discrepancy between the empirical stiffness value obtained and the theoretical estimate derived from established equations indicates that these theoretical models are dependable for predicting performance under comparable conditions. Such validation is crucial as it bolsters confidence in employing these formulas for future design evaluations and analyses.

3.2 Bilateral compound parallelogram flexures

Assuming $l=70\text{mm}$, $b=15\text{mm}$, $h=1.5\text{mm}$ and calculating the stiffness theoretical value of bilateral compound parallelogram flexures through the deduction of the formula, it is $2.0958 \times 10^4 \text{ N/m}$.

Through modeling and simulation, the relationship between structural forces and deformation is shown as follows:

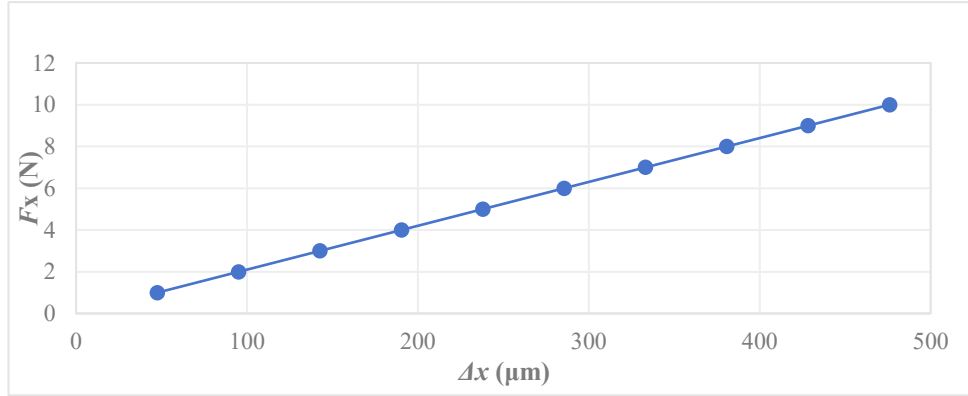


Fig. 5. The relationship between force and deformation of bilateral compound parallelogram flexures.

Through simulation fitting, the actual stiffness of bilateral compound parallelogram flexures (BCPF) has been established at 2.1009×10^4 N/m. This value is crucial for comprehending how the material behaves under different loading scenarios and plays a significant role in its engineering applications. The relative error calculated for this measurement stands at 0.24%, reflecting a high degree of accuracy in the experimental findings. The minor discrepancy between the measured stiffness and theoretical predictions indicates that the formula used for calculations closely reflects real-world behavior. This strong alignment not only confirms the validity of the mathematical model but also enhances confidence in employing such formulas to forecast material properties in future research or applications. Moreover, obtaining accurate measurements like this one is vital for refining design parameters and ensuring reliability in practical uses where BCPF might be applied, including aerospace components, automotive parts, or structural elements that require lightweight yet robust materials. The results from this simulation fitting can act as a benchmark for further investigations aimed at improving composite materials' performance through modifications or alternative formulations.

4 The Establishment and Validation of a New Parallel XY Nanopositioning Platform

In this section, a new parallel XY nanopositioning platform was meticulously designed and constructed using SolidWorks modeling software. The design process was segmented into multiple phases, beginning with the conceptualization of the platform's overall structure to establish essential requirements and specifications for achieving high-precision positioning tasks. Following the conceptual phase, detailed models were created for each individual component to ensure accuracy and functionality.

During the modeling phase, various factors were considered, including size dimensions tailored to specific application needs, material characteristics aimed at enhancing durability and performance under operational conditions, as well as mechanical constraints that could influence assembly and integration with other systems. Each component was modeled independently before being combined into a complete system model to assess compatibility and overall performance. After finalizing the model in SolidWorks, an extensive verification process was necessary to thoroughly evaluate its structural integrity. This assessment employed ANSYS

software for finite element analysis (FEA), which is vital for understanding how different forces interact during operation. FEA offers insights into stress distribution across components under static loads or dynamic impacts among various loading scenarios. The simulation outcomes yielded important data regarding areas prone to excessive stress concentration or potential failure points under standard operating conditions. By pinpointing these critical regions early in the design stage, proactive strategies can be implemented—such as altering geometric designs or opting for alternative materials—to reinforce vulnerable sections without hindering the efficiency of the entire system. This thorough methodology not only confirms the structural soundness of the nanopositioning platform but also establishes a robust foundation for subsequent experimental testing intended for real-world applications.

The modeling and simulation of this structure are shown in Figure 6 and Figure 7.

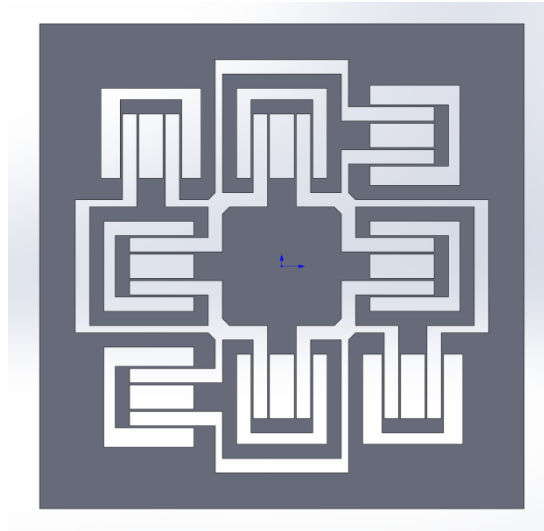


Fig. 6. Modeling of the new parallel XY nanopositioning platform.

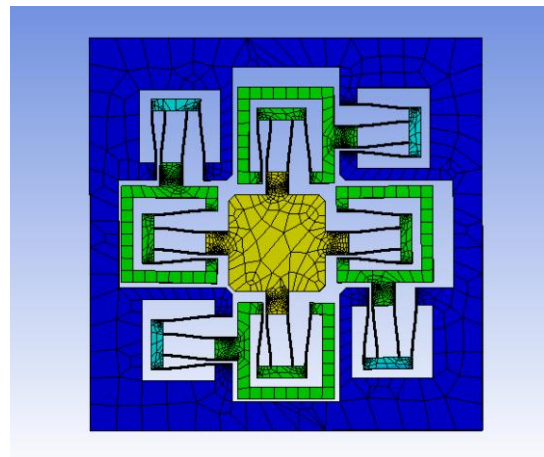


Fig. 7. Simulation of the new parallel XY nanopositioning platform.

The nan positioning platform is conceptually structured to facilitate the independent functioning of both the X and Y axes. This design ensures that movement along one axis does not influence movement on the other. Specifically, any force exerted in the X direction will solely impact motion along that axis without causing unintended shifts in the Y direction. The principle of decoupling is essential for achieving high-precision positioning tasks, as it enables separate control over each axis. To substantiate this theoretical framework, a detailed derivation was conducted in the article's second section, where extensive analysis of the model's behavior under various conditions was performed. Findings indicate that the single-axis stiffness of this new parallel XY nan positioning platform is quadruple that of CPF stiffness. A novel K value for single-axis performance has been defined for this innovative parallel XY nan positioning system, which plays a vital role in understanding how effectively it reacts to forces and displacements applied to each individual axis. The calculation method for this K value integrates empirical data from testing with theoretical insights derived from modeling efforts. Through thorough analysis and inventive design strategies aimed at enhancing axial independence and stiffness properties, a robust foundation has been laid for precise motion control in advanced applications necessitating fine positioning capabilities. We can obtain the new single-axis K value of the new parallel XY nan positioning platform as follows:

$$K = \frac{4Ebh^3}{l^3} \quad (8)$$

Set the b to 30mm, h to 1.5mm and l to 45 mm during the modeling process. The material is aluminum, with a Young's modulus of 71 GPa. According to the single-axis stiffness formula derived above, the theoretical value is $3.1556 \times 10^5 \text{ N/m}$.

By using simulation technology and graphical analysis, the specific relationship between force and deformation in the X and Y axes was determined after applying pressure to both axes simultaneously. The method includes systematically adjusting the applied force and recording the deformation generated in each direction. These simulations are intended to simulate real-life conditions in order to accurately evaluate the performance of the nan positioning platform under load. By studying these interactions, we can see how changes in force directly affect the displacement of each axis.

The resulting relationship graph is presented below:

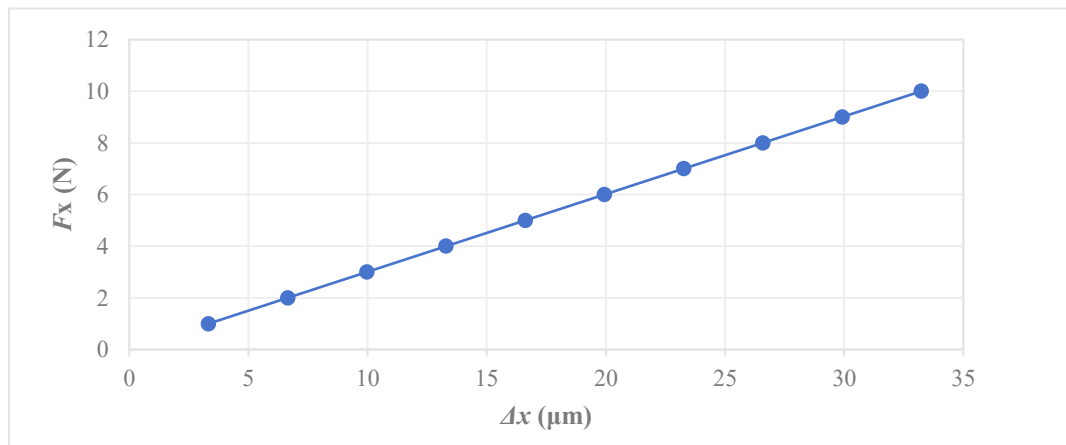


Fig. 8. The correlation between force and deformation along the X-axis.

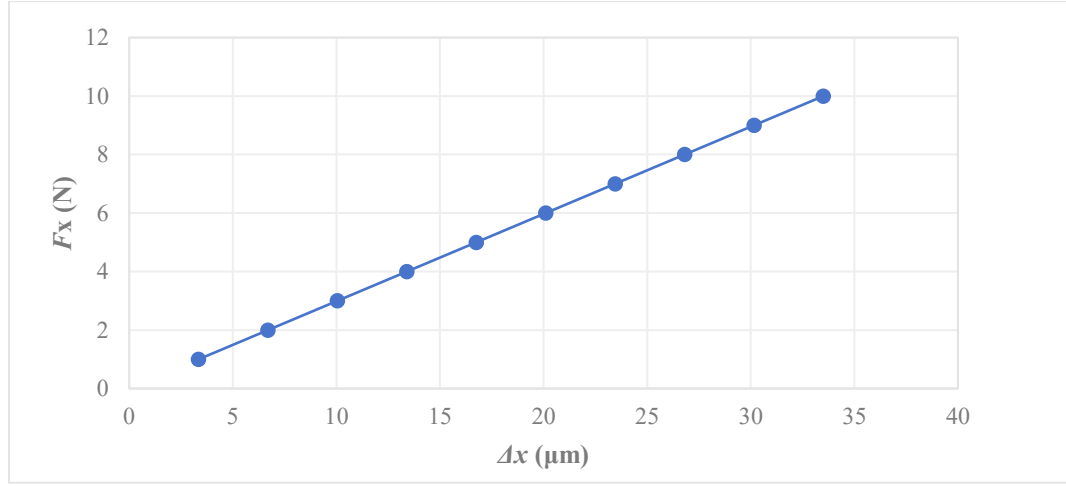


Fig. 9. The correlation between force and deformation along the Y-axis.

The graphical fitting analysis revealed that the stiffness in the X direction measured 3.0086×10^5 N/m, whereas in the Y direction it was recorded at 2.9840×10^5 N/m. This information is vital for comprehending how each axis reacts to applied forces, which is critical for assessing the overall functionality of the nanopositioning platform. The relative error for stiffness in the X direction was -4.66%, indicating a slight deviation from what was theoretically anticipated. In a similar vein, the relative error for stiffness in the Y direction stood at -5.44%. These results suggest that the observed stiffness values align closely with theoretical predictions derived from initial design parameters and modeling efforts. Confirming structural accuracy through experimental data can bolster confidence in both reliability and effectiveness of the nanopositioning system during practical applications. Furthermore, these findings lay a groundwork for future enhancements and optimization of design frameworks aimed at improving axial performance characteristics even further. Grasping how experimental outcomes relate to theoretical expectations will facilitate ongoing advancements in precision engineering across various domains.

5 Conclusion

This study conducted an in-depth analysis of standard flexible structures, leading to the development of stiffness equations for bilateral simple parallelogram flexures (BSPF) and bilateral compound parallelogram flexures (BCPF). The research encompassed both theoretical evaluations and practical modeling and simulation efforts. Detailed geometric representations and structural assessments of BSPF and BCPF were created using SolidWorks. Subsequently, simulations were performed in ANSYS to validate the derived stiffness equations for BSPF and BCPF under different loading scenarios. The findings from these analyses demonstrated that the models accurately represented the anticipated performance metrics. Additionally, this research introduced a novel parallel XY nanopositioning platform, achieving measured X-axis stiffness at 3.0086×10^5 N/m and Y-axis stiffness at 2.9840×10^5 N/m. Notably, these measurements showed minimal deviation from the theoretically predicted values of 3.1556×10^5 N/m, underscoring the reliability of the employed modeling techniques. The implementation of this

nanopositioning system has significantly expanded its motion range compared to previous designs. This enhancement is projected to improve operational efficiency across various applications while bolstering capabilities in fields requiring precise positioning systems. Furthermore, this framework establishes a vital groundwork for advancing large-range nanopositioning technologies and offers valuable insights into effective design strategies suitable for diverse situations. As a result, it is anticipated that real-world applications will reap benefits from these innovations by enabling higher precision complex tasks over broader ranges.

References

- [1] X Ding, J S Dai. Compliance Analysis of Mechanisms with Spatial Continuous Compliance in the Context of Screw Theory and Lie Groups[J]. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 2010, 224 (11): 2493-2504.
- [2] Choi, J., Hong, S., Lee, W., Kang, S., Kim, M.. A Robot Joint With Variable Stiffness Using Leaf Springs[J]. IEEE Transactions on Robotics: A publication of the IEEE Robotics and Automation Society, 2011, 27 (2): 229-238.
- [3] Santosh Devasia, Evangelos Eleftheriou, S. O. Reza Moheimani. A Survey of Control Issues in Nanopositioning.[J]. IEEE Trans. Contr. Sys. Techn., 2007, 15 (5): 802-823.
- [4] Byoung Hun Kang, John T. Wen, Nicholas G. Dagalakis, Jason Gorman. Analysis and Design of Parallel Mechanisms with Flexure Joints.[J]. IEEE Trans. Robotics, 2005, 21 (6): 1179-1185.
- [5] Jonathan B. Hopkins, Martin L. Culpepper. Synthesis of Multi-degree of Freedom, Parallel Flexure System Concepts via Freedom and Constraint Topology (FACT) - Part I: Principles[J]. Precision Engineering, 2010, 34 (2): 259-270.
- [6] Cui Mengjia, Shang Erwei, Jiang Shouqian, Liu Yu, Zhang Zhen. Design, Fabrication and Implementation of a High-performance Compliant Nanopositioner via 3D Printing with Continuous Fiber-reinforced Composite[J]. Journal of Micromechanics and Microengineering, 2021, 31 (12).
- [7] Vickers Nicholas A, Andersson Sean B. Synthetic Stochastic Motion Platform for Testing Single Particle Tracking Microscopes.[J]. IEEE transactions on control systems technology : a publication of the IEEE Control Systems Society, 2022, 30 (6): 2726-2733.
- [8] Simon Brecht G, Kurdi Samer, Carmiggelt Joris J, Borst Michael, Katan Allard J, van der Sar Toeno. Filtering and Imaging of Frequency-Degenerate Spin Waves Using Nanopositioning of a Single-Spin Sensor.[J]. Nano letters, 2022, 22 (22).
- [9] Nava Rezvani, Ali KeymasiKhalaji. Adaptive RBFNN - based Predictive Control for the Nanopositioning of an Electrostatic MEMS Actuator[J]. IET Control Theory & Applications, 2023, 18 (5): 551-565.
- [10] Nan Wang, Zhe Yuan, Peng Wang. Dynamic Electromagnetic Force Variation Mechanism and Energy Loss of a Non-contact Loading Device for a Water-lubricated Bearing[J]. Journal of Mechanical Science and Technology, 2021, 35 (6): 1-12.
- [11] W Dong, J Tang, Y ElDeeb. Design of a Linear-motion Dual-stage Actuation System for Precision Control[J]. Smart Materials and Structures, 2009, 18 (9): 095035 (11pp)
- [12] Sharmila N Shirodkar, Tonghui Su, Nitant Gupta, Evgeni S Penev, Boris I Yakobson. Mechanical Efficiency of Photochromic Nanomotors, From First Principles.[J]. Small (Weinheim an der Bergstrasse, Germany), 2024, e2400305.

- [13] Mohammadbagher Mohammadnezhad, Salah Raza Saeed, Sarkew Salah Abdulkareem, Abdollah Hassanzadeh. Light-driven Nanomotors with Reciprocating Motion and High Controllability based on Interference Techniques.[J]. Nanoscale advances, 2024, 6 (4): 1122-1126.
- [14] Karrar A. Hassan, Furat I. Hussein, Wisam S. Hacham. Design and Fabrication of an Electromechanical Tester to Perform Two-dimensional Tensile Testing for Flexible Materials[J]. American Academic Scientific Research Journal for Engineering, Technology, and Sciences, 2022, 90 (1): 41-51.
- [15] Paloma L Ocola, Ivana Dimitrova, Brandon Grinkemeyer, Elmer Guardado Sanchez, Tamara Đorđević, Polnop Samutpraphoot, Vladan Vuletić, Mikhail D Lukin. Control and Entanglement of Individual Rydberg Atoms near a Nanoscale Device.[J]. Physical review letters, 2024, 132 (11): 113601-113601.
- [16] Shorya Awtar, Gaurav Parmar. Design of a Large Range XY Nanopositioning System[J]. Journal of Mechanisms and Robotics: Transactions of the ASME, 2013, 5 (2): 021008-1-021008-10.
- [17] Qingsong Xu. New Flexure Parallel-Kinematic Micropositioning System With Large Workspace[J]. IEEE Transactions on Robotics: A publication of the IEEE Robotics and Automation Society, 2012, 28 (2): 478-491.

Optimized Robotic Grippers Based on Scissor-like Elements

Tiancheng Gao

School of Future Aerospace Technology, Beihang University, Beijing, 100191, China

Gordon200309@outlook.com

Abstract. The grippers are the decisive part of the robot palletizer and the structure of the grippers is the decisive factor of the ability of robot palletizers to grab targets in different shapes. Inspired by scissor-like elements, an optimized rigid structure of robotic grippers based on the two-dimensional pantograph is proposed which has 1 degree of freedom and can move without strain. The path of the movement of the grippers is easy to be calculated and controlled and due to the design of the hydraulics, each finger of the grippers can move separately when powered by only one motor which can better fit the shape of the targets to grab them more tightly. The report includes the virtual model and kinematic analysis of the optimized robotic grippers.

Keywords: Robotic grippers, Scissor-like elements, Two-dimensional pantograph, Kinematic analysis, Hydraulics.

1 Introduction

Robotic arm has a wide range of use in various fields such as manufacturing, agriculture and etc. It can automatically and continuously accomplish different tasks especially those are repeated or hard for humans. For the robotic arms that are designed to grab and move targets known as robot palletizers, the design of their grippers is a critical factor in their function to grab targets in different shapes such as cube, cylinder, sphere and even irregular shape.

Deployable structures such as Bennett linkages, scissor-like elements and origami structures have been widely used in many fields such as medical field[1], architecture[2, 3], space science[4, 5] and the design of robots[6, 7]. Its shape can switch automatically under the power of the electric motors. So when it is applied to robotic grippers, it enables them to change structure to fit the shape of the targets better in order to hold the targets tighter. There are lots of successful applications of deployable structures in the design of the robotic grippers such as the research developed by S. Li's team[8], A. Firouzeh and J. Paik's research[9] and Z. Zhakypov's team[10]. They applied origami structures to making a soft robotic grippers that can switch their shapes. But the soft grippers require strong materials and they are hard to be powered by motors. However, the rigid deployable structures have simpler movement and can be easily driven by motors. What's more, it has less deformation than soft structures under pressure which means it can bear more weight when being used as grippers. The research developed by K. Lee's team[11] showed a successful use of origami twisted tower in building a robotic grippers which can hold targets in different shapes under the power of only one motor. But its structure is complicated and all the fingers of the grippers move simultaneously, which means that for some shapes like a cuboid whose length and width have a huge difference. The

grippers built by C. Liu's team[12] and A. Orlofsky's team[13] have the similar Miura deployable structure. Their designs are less complex than the one of K. Lee's team[11], which makes them easier to control. But the fingers of their grippers can not move separately under the power of one motor either. In this report, a practicable design of robotic grippers using deployable structures whose fingers can move separately is introduced.

The research developed by Agostino Zano[14] about the two dimensional articulated systems explained the geometry and kinematics of the articulated systems when rods' length and the position of the hinges vary which are similar to the scissor-like elements. Based on the scissor-like elements, Z. You[15] developed a pantograph structure which can be folded without strain. This feature indicates that it can be folded or deployed with little deformation, which makes it be able to be applied on the robotic grippers to expand, contract and carry objects.

Figure 1[15] shows three basic elements with their length factors which can build a simple two-dimensional pantograph. Element (a) and (b) is similar to each other which are both symmetric about the central line and the ratio of the length of the rods of (a) and (b) is k . Element (c) is homothetic about its joint and the ratio of the length of the left side and the right side is also k .

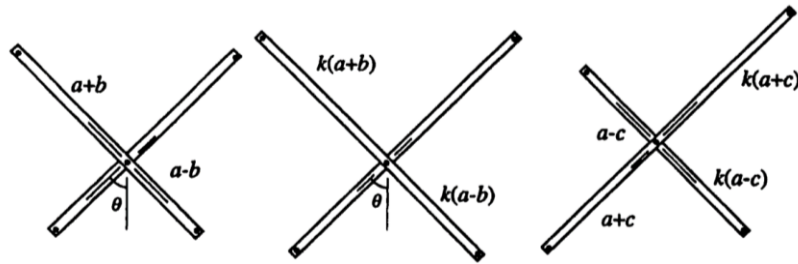


Fig. 1. (a-c) Three elements of the two-dimensional pantograph[15]

Connecting the three basic elements in the sequence (a), (c), and (b) to build a simple two-dimensional pantograph and using SOLIDWORKS to build its virtual model which is shown in Figure 2.

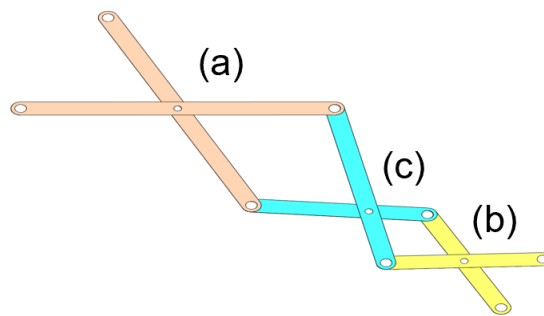


Fig. 2. A simple two-dimensional pantograph

2 The design of the unit of the gripper

To make the two-dimensional pantograph structure better fits the shape of the grippers, element (a) and (b) are cut in half and one rod of element (b) is extended. The model of the structure after being optimized can be used as the basic unit of the grippers and it is shown in the Figure 3.

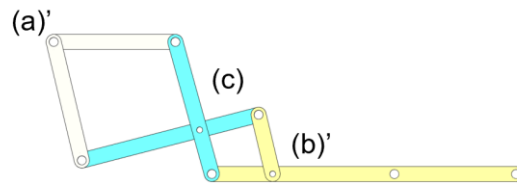


Fig. 3. The basic unit of the grippers

Adding designations to each joints and marking the length of each rods, the marked model is shown in Figure 4.

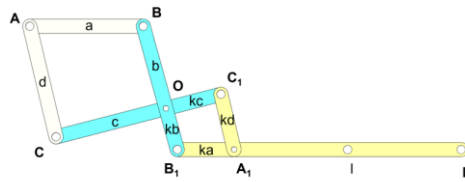


Fig. 4. The basic unit of the grippers (marked)

Once rod AB is fixed, the whole structure will expand and contract when rod AC revolves around the joint A. The different shapes of the basic unit when it is closed, half deployed and fully deployed are shown in Figure 5, 6 and 7.

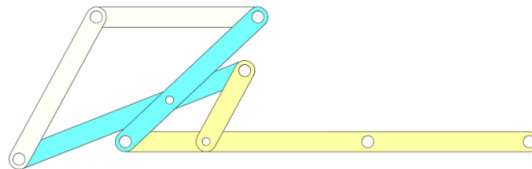


Fig. 5. The basic unit of the grippers (closed)

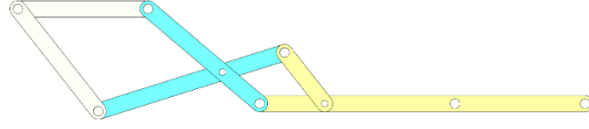


Fig. 6. The basic unit of the grippers (half deployed)

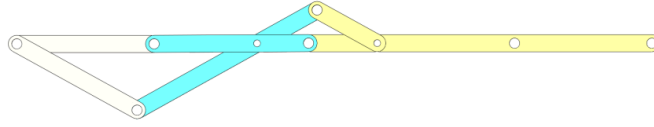


Fig. 7. The basic unit of the grippers (fully deployed)

3 Geometric and kinematic analysis

3.1 Geometric analysis

The diagram of the basic unit is shown in Figure 8. The length of each rod is in accord with the two-dimensional pantograph.

Apparently,

$$\frac{B_1O}{BO} = \frac{C_1O}{CO} = k$$

$$\angle BOC = \angle B_1OC_1 \quad (1)$$

Thus,

$$\triangle BOC \sim \triangle B_1OC_1 \quad (2)$$

Then,

$$\frac{B_1C_1}{BC} = \frac{B_1O}{BO} = k \quad (3)$$

Thus,

$$\frac{A_1B_1}{AB} = \frac{B_1C_1}{BC} = \frac{A_1C_1}{AC} = k$$

$$\triangle ABC \sim \triangle A_1B_1C_1$$

$$\angle ABC = \angle A_1B_1C_1 \quad (4)$$

Plus,

$$\angle BOC = \angle B_1OC_1 \quad (5)$$

Then,

$$\begin{aligned} \angle ABC + \angle BOC &= \angle A_1B_1C_1 + \angle B_1OC_1 \\ \angle ABO &= \angle A_1B_1O \\ A_1B_1 &\parallel AB \end{aligned} \quad (6)$$

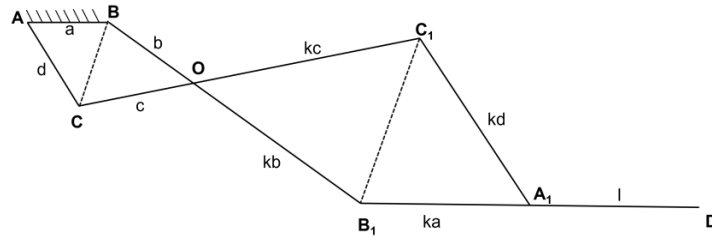


Fig. 8. Diagram of the basic unit of the grippers

According to the geometric analysis, rod A_1B_1 is always parallel to rod AB when rod AC revolves around joint A . While rod BB_1 revolves around the joint B , joint B_1 moves on a circle whose center is B and radius is $(k+1)b$. Thus, combining rod A_1B_1 is always parallel to AB , every point on line A_1B_1 moves on a circle whose center is on line AB and radius is $(k+1)b$, which is shown visually in Figure 9.

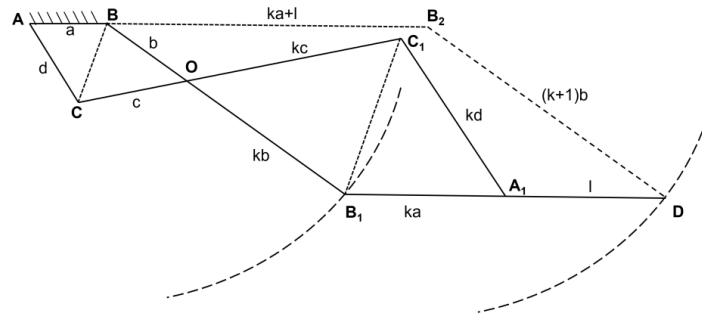


Fig. 9. Diagram of the basic unit of the grippers (with path of B_1 and D)

3.2 Kinematic analysis

The basic unit of the grippers consists of two similar four-bar linkages which are connected to each other. The amount of links except for ground (rod AB) is 5, and it has 7 revolute joints each of which have one degree of rotation freedom. Thus, once rod AB is fixed as the ground the total degree of freedom of the basic unit can be calculated by Kutzbach criterion as follows:

$$M = 3 \times (6 - 1) - 7 \times (3 - 1) = 1 \quad (7)$$

The structure has one degree of freedom and if representing the angular velocity of AC as the known input, the movement of the structure will be definite.

Another diagram of the basic unit with the specific angles marked is shown as Figure 10. One sides of θ_2 is parallel to the x-axis and the same as θ_3 and θ_4 .

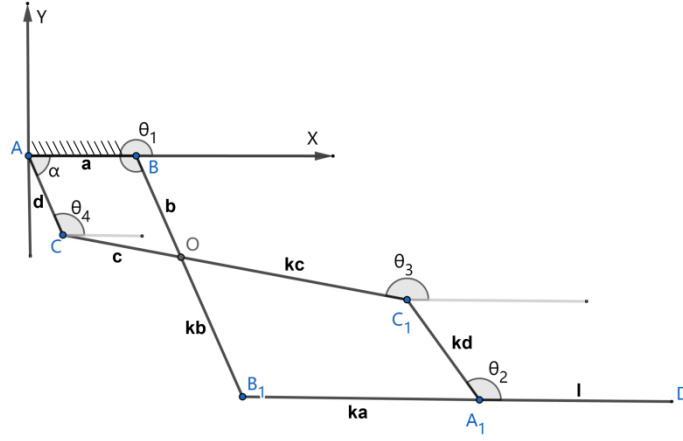


Fig. 10. Diagram of the basic unit of the grippers (with marked angles)

Representing the degree of α as the known input and the degrees of θ_1 , θ_2 , θ_3 and θ_4 are the output. The sides of θ_2 are parallel to those of θ_4 , thus $\theta_2 = \theta_4$. Then geometrically there are:

$$\begin{aligned} \theta_1 &= 360^\circ - \alpha \\ \theta_2 &= \theta_4 = 180^\circ - \alpha \end{aligned} \quad (8)$$

Following the sequence of joints A-B-B1-A1-C1-C-A to form a closed loop, the loop closure equation will be:

$$ae^{j \cdot 0} + (k+1)be^{j\theta_1} + kae^{j \cdot 0} + kde^{j\theta_2} + (k+1)ce^{j\theta_3} + de^{j\theta_4} = 0 \quad (9)$$

Simplifying the loop closure equation, and considering $\theta_2 = \theta_4$, we can get the position equation of the basic unit:

$$a + be^{j\theta_1} + de^{j\theta_2} + ce^{j\theta_3} = 0 \quad (10)$$

Differentiating the position equation, we can get the velocity equation and the acceleration equation:

Velocity equation:

$$b\dot{\theta}_1 e^{j\theta_1} + d\dot{\theta}_2 e^{j\theta_2} + c\dot{\theta}_3 e^{j\theta_3} = 0 \quad (11)$$

Acceleration equation:

$$be^{j\theta_1}(\ddot{\theta}_1 + j\dot{\theta}_1^2) + de^{j\theta_2}(\ddot{\theta}_2 + j\dot{\theta}_2^2) + ce^{j\theta_3}(\ddot{\theta}_3 + j\dot{\theta}_3^2) = 0 \quad (12)$$

Considering the position equation to work out the degree of θ_1 , θ_2 , θ_3 and θ_4 as the dependent variables while the degree of α is the input, its real and imaginary parts are as follows:

Real part:

$$a + b \cos \theta_1 + d \cos \theta_2 + c \cos \theta_3 = 0 \quad (13)$$

Imaginary part:

$$b \sin \theta_1 + d \sin \theta_2 + c \sin \theta_3 = 0 \quad (14)$$

Combining with the geometric equations, there is a set of equations which has 4 variable and 4 independent equations about the degree of α , θ_1 , θ_2 , θ_3 and θ_4 :

$$\begin{cases} a + b \cos \theta_1 + d \cos \theta_2 + c \cos \theta_3 = 0 \\ b \sin \theta_1 + d \sin \theta_2 + c \sin \theta_3 = 0 \\ \theta_2 = 180^\circ - \alpha \\ \theta_4 = 180^\circ - \alpha \end{cases} \quad (15)$$

Representing $a=40\text{mm}$, $b=30\text{mm}$, $c=40\text{mm}$, $d=40\text{mm}$, which is equivalent to the length of the rods of the virtual model. Through MATLAB program, the degree of θ_1 , θ_2 , θ_3 and θ_4 as the dependent variables while the degree of α is the input is shown in the plot in Figure 11.

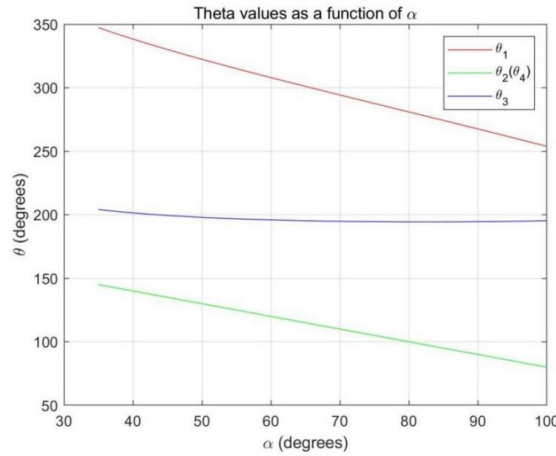


Fig. 11. Plot of the degree of θ_1 , θ_2 , θ_3 and θ_4 with α as the input

To figure out the range of movement of the rod A_1B_1 which is at the end of the grippers and is used to hold targets, mark the distance between rod A_1B_1 and AB which is on the base of the grippers as h , and the new diagram of the basic unit is shown in Figure 12.

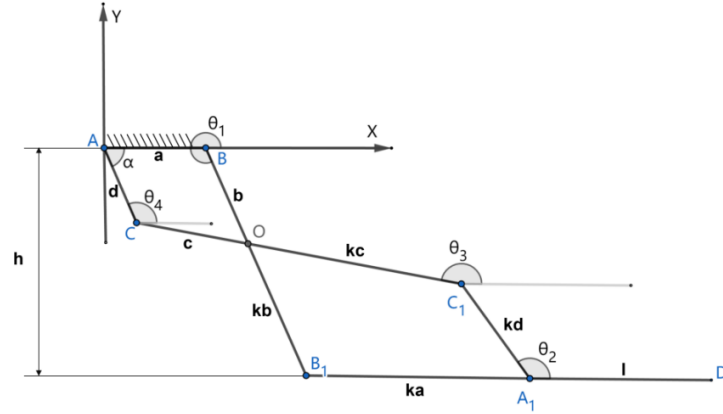


Fig. 12. Diagram of the basic unit (with marked distance h)

Geometrically,

$$h = (k + 1)b \quad (16)$$

Representing $k=2$, which is equivalent to the virtual model, and the plot of h as the dependent variable with the degree of α as the input is shown in Figure 13.

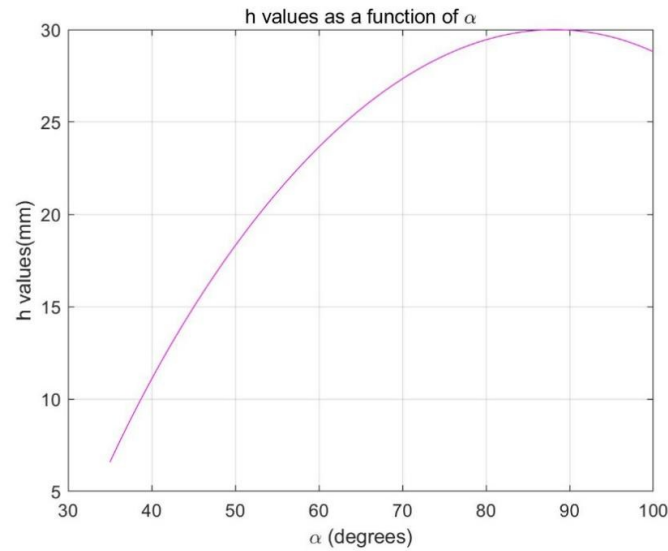


Fig. 13. Plot of h with α as the input

4 Prototype and the features

4.1 Prototype

Cutting out the rod AB and fixing the other part of the basic unit to the base of the grippers, the shape of the completed virtual model of the robotic grippers which is built in SOLIDWORKS when it is closed, half deployed and fully deployed is shown in Figure 14,15 and 16.

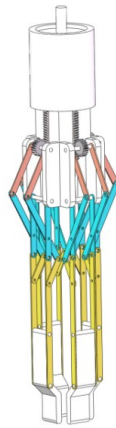


Fig. 14. The model of the robotic grippers (closed)

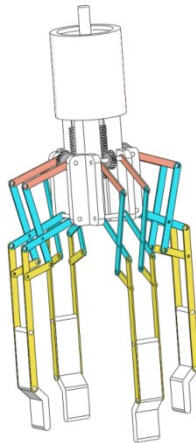


Fig. 15. The model of the robotic grippers (half deployed)

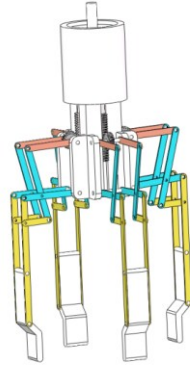


Fig. 16. The model of the robotic grippers (fully deployed)

During the whole process of its movement, the four ‘fingers’ of the grippers is always parallel to the lines on which the two joints of each finger that connect the finger to the base lie on, which makes the path of movement of each finger is easy to be calculated and controlled.

4.2 The transmission system

The transmission system the grippers consists of two critical parts. The first is the hydraulics which is shown in the Figure 17. It consists of 1 large cylinder, 4 small cylinders, and 5 plungers which fit the diameter of the 5 cylinders. There is a cavity between the large plunger and the 4 small plungers. Once the cavity is filled with water, oil or other liquids, it can transmit power of the motor which drives the large plunger to move up and down separately to the four small plungers. For liquids are difficult to compress and have no definite shapes, when one of the small plungers is forced to stop, others will continue to move under the power of the motor while keeping the volume of the liquid a constant. The process is shown in Figure 18, in which two of the plungers is forced to stop and others continue to move down.

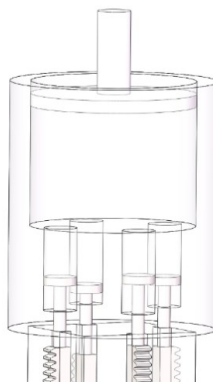


Fig. 17. The model of the hydraulics

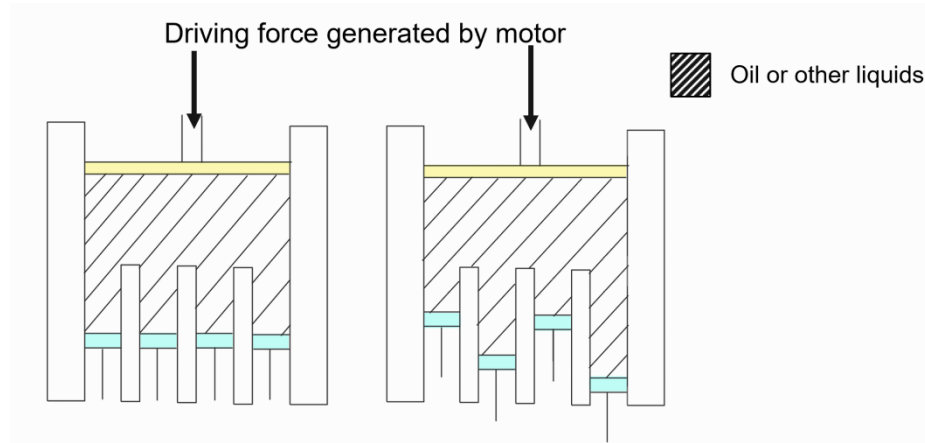


Fig. 18. The working process of the hydraulics

The second part is the rack and pinion mates that transmit the sliding of the plungers to the rotation of the pinions that are fixed to the joints of the fingers, which enable the fingers to open and close under the power of the motor. The model of the rack and pinion mates on the grippers is shown in Figure 19.

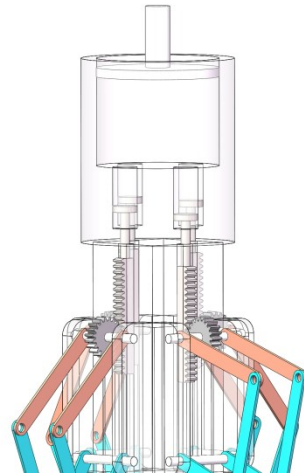


Fig. 19. The rack and pinion mates on the grippers

The transmission system allows each finger of the grippers to move separately when powered by only 1 motor. It enables the grippers to better fit the shape of the targets and reduces the failure rate causing by using more motors to separately power each finger. Once one of the fingers touches the surface of the target, it will stop moving and others will continue until all of them touch the surface of the target and the grippers grab the target tightly.

5 Discussion

The optimized robotic grippers are based on the two-dimensional pantograph and have the following features:

It can move without strain so that it can expand and contract without deformation.

Each of its unit has only 1 degree of freedom and the end of the unit is always parallel to the line which the joints of the unit that connect the unit to the base lie on. Thus, the structure has definite movement which can be easily calculated and controlled to avoid the obstacles and grab targets in the best position.

Each of its finger can move separately while it is powered by only one motor so that it can better fit the shape of the target, grab it more tightly and have higher reliability because the less motors are applied, the less possible the mechanism will break down.

6 Conclusion

This report mainly focuses on the structure of the unit of the grippers and its , transmission system. In fact there are far more factors that influence the ability of the robotic grippers such as the material applied, the kinds of the joints, the length of the rods and etc. Moreover, about the robotic grippers in this report, there are lots of parts that need to be tested and optimized: First, the grippers can be build using 3D printer and its ability to grab targets in various shapes and irregular shape along with the maximum of the weight it can bear when the targets vary need to be tested to find out if the optimized structure works well as the purpose of the research. Moreover, the structure of the basic unit needs to be tested and optimized to get better ones with better efficiency and ability to fit the shape of the targets and bear weight. And the end of the fingers in the report is a straight and rigid panel. Other material such as flexible ones and other structure of it can work better to grab the targets. There will be many interesting points about the robotic grippers to be researched and optimized in order to enhance its ability to grab different targets.

References

- [1] F. Zhang *et al.*, "Rapidly deployable and morphable 3D mesostructures with applications in multimodal biomedical devices," *Proceedings of the National Academy of Sciences*, vol. 118, no. 11, p. e2026414118, 2021.
- [2] I. Doroftei and I. A. Doroftei, "Deployable structures for architectural applications-a short review," *Applied mechanics and materials*, vol. 658, pp. 233-240, 2014.
- [3] N. De Temmerman, "Design and analysis of deployable bar structures for mobile architectural applications," 2007.
- [4] W. M. Sokolowski and S. C. Tan, "Advanced self-deployable structures for space applications," *Journal of spacecraft and rockets*, vol. 44, no. 4, pp. 750-754, 2007.
- [5] J. Santiago-Prowald and H. Baier, "Advances in deployable structures and surfaces for large apertures in space," *CEAS Space Journal*, vol. 5, pp. 89-115, 2013.
- [6] M. Askari, W. D. Shin, D. Lenherr, W. Stewart, and D. Floreano, "Avian-Inspired Claws Enable Robot Perching or Walking," *IEEE/ASME Transactions on Mechatronics*, 2023.

- [7] R. Datta, S. Pradhan, and B. Bhattacharya, "Analysis and design optimization of a robotic gripper using multiobjective genetic algorithm," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 1, pp. 16-26, 2015.
- [8] S. Li *et al.*, "A vacuum-driven origami "magic-ball" soft gripper," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019: IEEE, pp. 7401-7408.
- [9] A. Firouzeh and J. Paik, "Grasp mode and compliance control of an underactuated origami gripper using adjustable stiffness joints," *Ieee/asme Transactions on Mechatronics*, vol. 22, no. 5, pp. 2165-2173, 2017.
- [10] Z. Zhakypov, F. Heremans, A. Billard, and J. Paik, "An origami-inspired reconfigurable suction gripper for picking objects with variable shape and size," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 2894-2901, 2018.
- [11] K. Lee, Y. Wang, and C. Zheng, "Twister hand: Underactuated robotic gripper inspired by origami twisted tower," *IEEE Transactions on Robotics*, vol. 36, no. 2, pp. 488-500, 2020.
- [12] C. Liu, S. J. Wohlever, M. B. Ou, T. Padir, and S. M. Felton, "Shake and take: Fast transformation of an origami gripper," *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 491-506, 2021.
- [13] A. Orlofsky, C. Liu, S. Kamrava, A. Vaziri, and S. M. Felton, "Mechanically programmed miniature origami grippers," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020: IEEE, pp. 2872-2878.
- [14] A. Zanardo, "Two-dimensional articulated systems developable on a single or double curvature surface," *Meccanica*, vol. 21, pp. 106-111, 1986.
- [15] Z. You, "A pantographic deployable conic structure," *International Journal of Space Structures*, vol. 11, no. 4, pp. 363-370, 1996.

Appendix

The MATLAB codes used to calculate the degree of θ_1 , θ_2 , θ_3 and θ_4 along with the distance h is as follows.

```

clc;
clear all;

alpha = linspace(35,100,100);

a = 40;
d = 40;
c = 40;
b = 30;
k=2;

theta1_vals = zeros(1, length(alpha));
theta2_vals = zeros(1, length(alpha));
theta3_vals = zeros(1, length(alpha));
h_vals = zeros(1, length(alpha));

for i = 1:length(alpha)
    theta2 = 180 - alpha(i);

```

```

theta2_vals(i) = theta2;

fun = @(x) [
    a + b * cosd(x(1)) + d * cosd(theta2) + c * cosd(x(2));
    b * sind(x(1)) + d * sind(theta2) + c * sind(x(2))
];

x0 = [0, 0];

options = optimoptions('fsolve', 'Display', 'none');
sol = fsolve(fun, x0, options);

theta1 = sol(1);
theta3 = sol(2);

while theta1 < 0
    theta1 = theta1 + 360;
end
while theta1 > 360
    theta1 = theta1 - 360;
end

while theta3 < 0
    theta3 = theta3 + 360;
end
while theta3 > 360
    theta3 = theta3 - 360;
end

theta1_vals(i) = theta1;
theta3_vals(i) = theta3;

h_vals(i) = abs((k+1)*b * sind(theta1));
end

figure;
plot(alpha, theta1_vals, 'r');
hold on;
plot(alpha, theta2_vals, 'g');
plot(alpha, theta3_vals, 'b');
xlabel('\alpha (degrees)');
ylabel('\theta (degrees)');
legend('\theta_1', '\theta_2(\theta_4)', '\theta_3');
title('Theta values as a function of \alpha');
grid on;

figure;
plot(alpha, h_vals, 'm');

```

```
xlabel('\alpha (degrees)');  
ylabel('h values(mm)');  
title('h values as a function of \alpha');  
grid on;
```

Cross-Medium Vehicle Design

Yulu Du

Central South University, Changsha, China

15239256627@163.com

Abstract. Cross-medium vehicles are defined as “new-concept amphibious unmanned platforms capable of freely traversing between water and air.” With societal development in the 21st century, the aerial flight capabilities and underwater navigation capabilities of cross-medium vehicles have greatly met the demand for activities in both water and air. Their excellent mobility and concealment provide broad application prospects in both civilian and military fields. This paper analyzes the current development status of cross-medium vehicles domestically and internationally, focusing on three major design schemes: the water-entry performance and underwater navigation capability of swept-wing models, the stability and response speed of medium transitions in multi-rotor models, and the autonomous movement and cross-medium motion stability of hybrid-wing models. Additionally, two driving schemes are discussed: independent water-air propulsion and integrated water-air propulsion. The research highlights, challenges, and core technologies required for cross-medium vehicles are summarized.

Keywords: Cross-medium vehicles, Shape design, Propulsion, Simulation analysis

1 Introduction

Cross-medium vehicles are defined as “new-concept amphibious unmanned platforms capable of freely traversing between water and air.” Since the concept of cross-medium vehicles was first implemented in the 1930s, countries such as France, the United Kingdom, the United States, Russia, and China have conducted principle designs and test flights of various types of cross-medium vehicles over the following decades. Although preliminary results have been achieved, the technology for “repeated water-air medium transitions” has not yet been fully realized.

In the 21st century, the aerial flight capabilities and underwater navigation abilities of cross-medium vehicles have significantly met the growing demand for activities across water and air. With excellent mobility and concealment, these vehicles offer broad application prospects in both civilian and military fields. Research on the design and improvement of cross-medium vehicles also contributes to the advancement of related disciplines, such as energy dynamics, fluid mechanics, mechanical structure design, communication and navigation control, and bionics.

This paper analyzes and summarizes the development status and research methodologies of existing cross-medium vehicles both domestically and internationally,

identifying the research hotspots, challenges, and core technologies required in the field. It aims to provide readers with a comprehensive, in-depth, and inspiring perspective to further promote research and development in the field of cross-medium vehicles.

2 Development of Cross-Medium Vehicles

The concept of “dual-use for water and air” in cross-medium vehicles initially appeared in science fiction literature. In the 1930s, the Soviet Union first proposed a “new weapon concept” that combined the airplane, an “air-based weapon,” with the submarine, a “strategic underwater weapon,” to realize a dual-use water-air platform. This led to the successful design of the LPL—“flying submarine” prototype, marking the practical initiation of cross-medium vehicle design and improvement. Subsequently, the Soviet Union developed three representative conceptual prototypes: the RFS-1, Convair, and DARPA models. However, due to technological limitations at the time, none of these prototypes succeeded in practical applications that could achieve water-air medium transitions [1].

In the 21st century, with societal advancements, human demand for activities across water and air has increased significantly. A series of attempts and improvements have been made in the design and development of cross-medium vehicle prototypes in Europe, the United States, and China.

In Europe, the Aelius prototype, developed by Bordeaux Aviation Technology in France, underwent water testing in 2007 [2]. In 2008, the British company Warrior Sea-Air Technology successfully tested the GULL 36, part of the Seagull series of unmanned waterborne drones using a pontoon structure [3]. In 2016, Imperial College London employed bioinspired technology to design the AquaMAV (Aquatic Micro Air Vehicle), an amphibious vehicle modeled after the morphology of the gannet bird [3].



Fig. 1. The Seagull series unmanned waterborne drone GULL 36

In the United States, research on cross-medium vehicles dates back to 1996, with the “Sea Seeker” developed by Northrop Grumman under commission from the U.S. Navy [3]. In 2005, Lockheed Martin was commissioned by the U.S. Defense Advanced Research Projects Agency (DARPA) to develop the Cormorant UAV (Unmanned

Aerial Vehicle), which incorporated bioinspired technology to mimic the water entry and exit behaviors of cormorants in its design and principles [2].



Fig. 2. Cormorant UAV

The first waterborne UAV utilizing lidar sensors for autonomous navigation was successfully developed in 2006 by an Oregon steelworks company [3]. In December 2013, the U.S. Navy announced the successful completion of verification tests for the Sea-Robin XFC submarine-launched UAV [3]. In 2016, the Blackwing submarine-launched UAV was deployed for intelligence gathering, reconnaissance, and relay communication tasks. In 2017, Hamzeh Alzu'bi et al. introduced the Loon Copter, a quadcopter prototype with active buoyancy control [3]. In February 2018, North Carolina State University, in collaboration with Triton Science and Imaging, developed the Eagle Ray, a fixed-wing trans-medium aircraft [4].

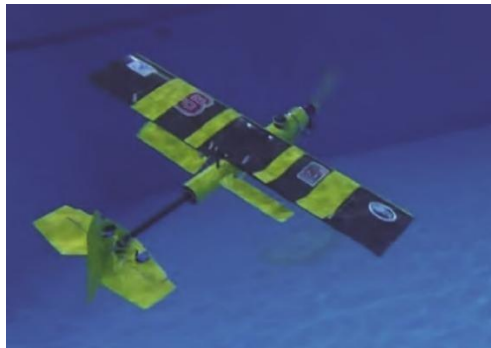


Fig. 3. The Eagle Ray fixed-wing trans-medium aircraft

In China, significant progress has also been made. In 2009, Beihang University developed an amphibious vehicle prototype using bioinspired design based on the morphology of flying fish, incorporating variable-sweep wings for buoyancy adjustment to transition between water and air. In 2015, the same research group developed another cross-medium unmanned vehicle inspired by the gannet bird [3].

In 2011, Nanchang Hangkong University developed two unmanned submarine prototypes powered by fully electric and hybrid oil-electric drives, featuring wings with a 90° variable sweep angle [3]. In 2019, Nanjing University of Aeronautics and

Astronautics designed and prototyped a bionic cross-medium vehicle inspired by the ringed pectoral fins of marine organisms [5]. In 2022, students from Shanghai Jiao Tong University completed the Nezha platform, the world's first sea-air integrated cross-domain vehicle. This platform boasts the greatest diving depth, highest load capacity, and widest underwater operating range among similar publicly disclosed projects worldwide [6].



Fig. 4. The “Nezha” sea-air integrated cross-domain vehicle platform developed by Shanghai Jiao Tong University

Currently, research and development of cross-medium vehicles have seen considerable maturity in waterborne UAVs and submarine-launched UAVs, with the United States leading in prototype technology. Europe has also developed a number of experimental prototypes, while China's existing prototypes are mainly the result of university-led independent research and development. Although European and Chinese prototypes are predominantly in the research stage, further development and improvement are needed to produce cross-medium vehicles that are ready for mass production and practical applications.

As a new concept proposed in recent decades, water-air trans-medium vehicles face stringent requirements to ensure stable operations in both mediums and to perform functions such as reconnaissance and transportation. These requirements necessitate advancements in deformation mechanisms, water-air propulsion systems, and guidance and control technologies. Consequently, further research in this field can drive progress in disciplines such as material fabrication, electronic circuit design, mechanical engineering, dynamics, and computational control. Compared to existing international research, domestic innovations in vehicle shape design, medium transition methods, and control mechanisms require further development to establish a more mature and systematic research and manufacturing framework.

3 Cross-Medium Aircraft Shape Design

Early water-air cross-medium aircraft primarily adopted three traditional structures: fixed wings, multi-rotors, and flapping wings. With increased research on cross-medium stability, the development of shape design has diversified, introducing swept

wings, multi-rotors, combination wings, and bionic wings. This paper mainly compares swept wings, multi-rotors, and combination wings in terms of flight performance and water-entry buffering.

3.1 Swept Wings

The design inspiration for swept wings originates from the diving and prey-catching behavior of waterfowl such as gannets. Using bionic design methods, it references how waterfowl fold their wings and adjust the sweep angle to enable rapid water entry.

The concept of swept-wing design was first introduced in 2010 in *New Scientist* magazine. Submarine designer HAWKS proposed mimicking the diving behavior of waterfowl to address the issue of water entry for aircraft [7]. In 2012, Fabian et al. at the Massachusetts Institute of Technology designed the Bionic Gannet (MIT) experimental prototype and conducted water-entry experiments to verify the feasibility of swept-wing aircraft. When transitioning from air to water, the Bionic Gannet mimicked the gannet's diving mode, striking the water at a speed of 7 m/s. It rapidly swept its wings back to reduce the impact load during water entry, preventing structural damage. After successful water entry, the Bionic Gannet adjusted its overall density to balance buoyancy and gravity, achieving a stable underwater motion state. Although this aircraft successfully demonstrated the feasibility of swept wings for diving into water, it was unable to cruise underwater or take off from water, falling short of the requirements for "repeated cross-medium transitions" [8].

In 2012, Liang Jianhong and his team at Beihang University developed a swept-wing aircraft called Gannet I (BUAA). Through water-entry testing, they examined overload conditions and wing root pivot tension under varying water-entry speeds, angles, and wing sweep angles. This provided reference data for the subsequent structural design, material strength selection, and underwater motion control algorithms of the improved Gannet II (BUAA) [8].

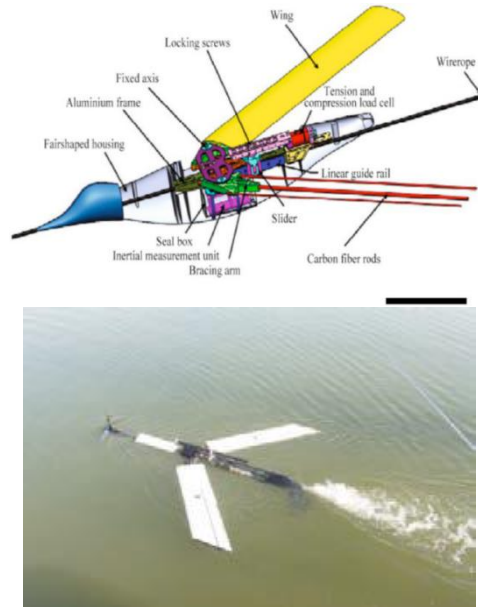


Fig. 5. Gannet I (BUAA) and Gannet II (BUAA) Diving Drones

In 2014, Liang's team introduced a large swept-wing underwater aircraft called Flying Fish [9], with a body length of 1.98 m, a wingspan of 3.4 m, a wing area of 1.5 m², a flight speed of 85 km/h, and a maximum altitude of 500 meters. It employed a hydraulic jet propulsion system but had a low underwater cruising speed of only 0.2–0.5 m/s, which fell far short of practical application requirements. Additionally, its large wing design made the overall body volume too large, limiting the prototype to low-efficiency gliding landings and takeoffs.



Fig. 6. Flying Fish Bionic Aircraft

In 2016, Siddall et al. at Imperial College London designed a swept-wing aircraft named AquaMAV (Aquatic Micro Air Vehicle) [3], inspired by gannets. The wing design consisted of three parts: the first part was attached to the main body, while the outer two parts were connected to the first via sealed bearings. During water-entry motion, the wings could sweep backward 90° to transition the aircraft passively from

flight mode to dive mode, reducing the impact force on the body during water entry. Aerodynamic evaluations in wind tunnels and water tunnels confirmed AquaMAV's excellent flight performance with extended wings and low drag and lift with folded wings. Additionally, the team adjusted flight speed and altitude to modify water-entry parameters, successfully conducting multiple water-entry flight tests that demonstrated the buffering effect of the swept wings. Although its flight speed reached 18 m/s, AquaMAV's underwater propulsion and energy efficiency remain areas for improvement [8].



Fig. 7. AquaMAV (Aquatic Micro Air Vehicle)

In 2021, Friedrich et al. at ETH Zurich introduced a swept-wing aircraft called Dipper [8]. Its design resembled the Bionic Gannet (MIT) prototype by Fabian et al., with the addition of a T-tail structure. The main innovation lay in the propulsion system, which utilized reversible motors to independently drive aerial and aquatic propellers, reducing the propulsion system's weight. However, the low body density of this prototype meant that the diving motion alone could not fully submerge the body into water; additional propulsion from the propellers was required. The complex control mechanisms for Dipper's water-entry motion prevented it from achieving seamless transitions from diving landings to underwater cruising, limiting its operational flexibility.

3.2 Multirotor

The multirotor is the most widely used structure for unmanned aerial vehicles (UAVs), suitable for low-altitude, low-speed operations, vertical takeoff and landing, and hovering. By equipping a single aircraft with multiple aerial propellers, it can generate substantial lift and achieve a high thrust-to-weight ratio. Multirotor cross-medium vehicles can adjust their flight resistance and torque by controlling the rotor speed of their electric motors, enabling better maneuverability during water entry and exit, and ensuring stability and continuity during medium transitions.

In 2014, Drews et al. from the Federal University of Rio Grande do Sul [10] first demonstrated the feasibility of multirotor cross-medium vehicles. The prototype model was equipped with four aerial propellers for various operational scenarios and four aquatic propellers. A power evaluation test was conducted on the prototype in 2019. However, the dual independent propulsion systems for water and air significantly increased the vehicle's weight, severely reducing the performance of multirotor vehicles, which already suffer from limited range.

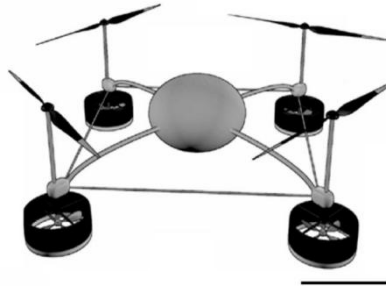


Fig. 8. Prototype model designed and developed by Drews et al. in 2014



Fig. 9. Prototype developed by Drews et al. in 2014

In 2015, Alzu'bi et al. from the University of Auckland [11] addressed the issue of excessive weight and developed a prototype called the Loon Copter. The Loon Copter required only four fixed-pitch aerial propellers to meet the propulsion needs for both underwater and aerial movement. To account for differences in water and air density, the team proposed a propulsion scheme compatible with both media based on variable-speed aerial propellers. By reducing the rotational speed of aerial propellers underwater, the system could accommodate underwater motion requirements. However, experiments showed that when propellers and parts of the fuselage were raised above the water surface, the propeller thrust significantly decreased. As a result, the propellers needed to rapidly increase their rotational speed to generate sufficient lift to prevent the fuselage from falling back into the water. This imposed extremely high demands on the responsiveness of the control system.



Fig. 10. Loon Copter prototype

In 2015, Maia et al. from Rutgers University, funded by the U.S. Navy, released the Naviator multirotor vehicle. The Naviator utilized a dual-layer coaxial octocopter structure to address the issue of sudden loss of lift during water exit. During water entry, the upper layer maintained rotation to stabilize the vehicle's motion; during water exit, the lower-layer rotors ensured balance after breaking the water surface. The Naviator's structural and dynamic strategies enabled smoother and faster medium transitions [8].

The work of Alzu'bi et al. and Maia et al. pioneered the application of integrated water-air propulsion technology in multirotor structures. Since then, various methods to enhance the motion performance of multirotor vehicles have been proposed.

In 2022, Li Li et al. from Beihang University [12] designed an adaptive propeller structure. The transformation of the propeller's form between water and air effectively shortened the time required for speed adjustments. Additionally, Li Li et al. proposed a low-energy-cost attachment scheme by studying adhesive structures. With the assistance of a biomimetic disk, the vehicle could adhere to the sidewalls of ships or the surfaces of marine organisms, facilitating activities such as carrying equipment and conducting underwater exploration.



Fig. 11. Prototype designed and developed by Li Li et al.

3.3 Combination Wing

The concept of combination wings originated in 2018 when Stewart et al. proposed three cross-medium vehicle design concepts inspired by the hunting behavior of seagulls: the quadrotor/fixed-wing hybrid vehicle, the VTOL (vertical take-off and landing) tail-sitter vehicle, and the water-jet take-off vehicle. The quadrotor/fixed-wing hybrid vehicle is also referred to as the combination wing vehicle [8].

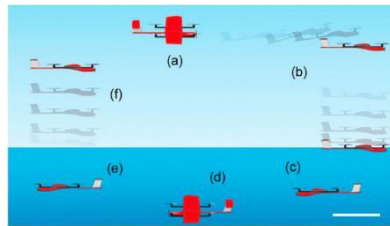


Fig. 12. Design concepts by Stewart et al.

In 2017, Lu Di et al. from Shanghai Jiao Tong University [13] initiated their combination wing vehicle project and released a series of Nezha vehicles. Among them,

the second-generation Nezha was equipped with a pair of fixed wings, four rotors, and a lightweight pneumatic buoyancy (LPB) system. The fixed wings function as flight wings in the air and as glide wings underwater, while the rotors ensure stable take-off and landing. Additionally, the LPB system significantly enhances underwater cruising performance. The prototype can perform four primary actions: vertical take-off and landing, hovering, aerial flight, and underwater cruising. The thrust generated by the rotors counteracts the gravity of the vehicle during hovering, take-off, or landing, while in aerial flight, it overcomes air resistance and works in tandem with lift generated by the fixed wings to offset part of the gravity. During underwater cruising, the vehicle's motion is controlled by the LPB system installed at the front of the fuselage, which features external safety airbags. These airbags inflate or deflate using high-pressure gas, allowing the vehicle to achieve positive or negative buoyancy and enabling nose-up or nose-down movements. However, the underwater navigation capabilities of Nezha II still have room for optimization: the maximum diving depth is only 5 meters; its underwater motion control lacks precision, and it cannot autonomously change its diving direction; the motor arm structure significantly affects hydrodynamic performance, and its payload capacity is highly limited.



Fig. 13. Nezha II vehicle

In 2021, Lu Di et al. released an improved version of Nezha II, called Nezha III (Tail-sitter) [6], achieving significant improvements in diving depth, endurance, and motion autonomy. They replaced the traditional buoyancy control system used in underwater gliders (UGs) with a lightweight pneumatic buoyancy system, enabling a maximum diving depth of 50 meters. By conducting buoyancy-pitch coupling research, they proposed a horizontal gliding pitch control strategy that reduces energy consumption compared to the detachable internal mass method commonly used in underwater glider designs. They also designed foldable motor arms to address the issue of body overturning caused by changes in the center of gravity during take-off. The research team proposed a take-off and landing control method for Nezha III to counteract surface wave disturbances and enhance the stability of cross-medium motion, as well as a dynamic trajectory planning method for dual-medium motion in air and water.

To further improve underwater endurance, Lu Di et al. developed a piston-based variable buoyancy system in 2022 and released an improved prototype, Nezha III (VTOL) [8]. Water entry tests demonstrated that the maximum working depth of this prototype reached 35.5 meters, and it could perform underwater cruising activities continuously for 24 hours.



Fig. 14. Nezha III (Tail-sitter) vehicle



Fig. 15. Nezha III (VTOL) vehicle

4 Propulsion Systems for Cross-Medium Vehicles

Cross-medium vehicles are designed to operate in both aerial flight and underwater navigation. The significant differences in physical parameters between air and water present unique challenges: water's density is approximately 800 times that of air, and its viscosity coefficient is about 59 times higher than air. Consequently, the working principles and mechanisms of propulsion systems used in air and water differ significantly.

In aerial flight, the lift-to-drag ratio is a critical factor in evaluating the performance of propulsion systems, where the vehicle's weight is a key limiting factor. In underwater navigation, the drag primarily depends on the size and surface area of the vehicle, making size another critical constraint for cross-medium vehicles. Therefore, the energy source selection for cross-medium propulsion systems must account for both weight and size limitations.

Currently, propulsion systems for cross-medium vehicles can be categorized into two types: air-water independent systems and integrated air-water systems. Vehicles with air-water independent systems employ separate propulsion devices for aerial and underwater operations, with no interference or overlap in their energy or control systems. In contrast, vehicles with integrated air-water systems use a single propulsion device for both environments, requiring fuel that can function effectively in both air and water.

4.1 Air-Water Independent Propulsion Systems

Air-water independent propulsion systems involve separate designs for aerial and underwater propulsion devices, with distinct mechanisms for activation, control, and

fuel selection. Due to the physical differences between air and water, the propulsion requirements vary greatly depending on the operating medium. Currently, employing two independent propulsion systems ensures optimal performance in both environments. For aerial propulsion, systems often use conventional turbofan or piston engines, while underwater propulsion typically relies on combinations of batteries, motors, and propellers. For instance, in 2020, Shao Dong of the Chinese Aero Engine Academy [14] proposed a combined propulsion system using gas turbines or piston engines paired with propellers or pumps for underwater propulsion, and turbofan or turbojet engines for aerial propulsion.

Historically, the Soviet Union proposed a flying submarine design in 1934, which combined the functionality of aircraft and submarines. This concept used three 895 kW piston engines for aerial propulsion and a 7.46 kW motor powered by batteries for underwater propulsion. However, due to technological and policy limitations, this project remained in the conceptual phase. Nonetheless, it laid a foundation for the subsequent development of cross-medium vehicle designs [14].

In the 1970s, the United States proposed a large-scale submersible aircraft concept powered by turbofan engines for aerial propulsion and Stirling engines for underwater propulsion [14].

Similarly, in 2008, DARPA introduced a hybrid flight platform concept that aimed to integrate the speed and range of aircraft with the cruising and stealth capabilities of surface vessels. However, the significant technical disparities between aircraft and submarines posed substantial challenges for the project's development [14].

In 2012, Professor Wang Yun's research team at Nanchang Hangkong University [15] proposed a hybrid propulsion system for an air-water drone. The system employed a single-cylinder gasoline piston engine and foldable air propellers for aerial propulsion, and lithium batteries, motors, and water propellers for underwater propulsion. Although this design enabled autonomous operation across mediums, the prototype failed to achieve waterborne takeoff despite successful underwater sealing of the gasoline engine.



Fig. 16. Hybrid propulsion system for air-water drone developed by Nanchang Hangkong University

However, in air-water independent propulsion systems, one propulsion system remains idle during operation in the other medium. The inclusion of two independent systems increases the vehicle's weight, size, and design complexity, significantly limiting its performance.

4.2 Integrated Air-Water Propulsion Systems

Integrated air-water propulsion systems offer advantages in terms of integration, reduced weight, and enhanced operational efficiency, making them well-suited for complex military and civilian applications.

In 2016, Siddall and colleagues at Imperial College London designed the AquaMAV (Aquatic Micro Air Vehicle) [3], [16]. This vehicle used high-pressure gas and a small quantity of liquid to generate thrust by simulating a squid-like jetting mechanism for takeoff and water exit. During operation in both mediums, the system relied on batteries and motors to power a head-mounted propeller. A subsequent 2019 study further improved the propulsion design, though experiments were limited to water-exit scenarios without testing aerial and underwater movements [16].



Fig. 17. AquaMAV water exit concept by Imperial College London

In 2017, the Weisler team at North Carolina State University [16] designed a fixed-wing cross-medium vehicle named EagleRay, and in 2015, Beihang University developed a cross-medium vehicle inspired by the “booby bird” [3]. Both vehicles used a combination of batteries and servo motors for power, utilizing propellers for propulsion when surfacing. These vehicles provided valuable insights into the power system design for subsequent cross-medium vehicles. In 2016, Professor Wang Yun’s research group [17] proposed a design inspired by turbofan engines, which eliminated the central shaft of traditional engines using counter-rotation technology. They installed a metallic water ramjet engine in place of the central shaft, effectively reducing the weight of the power system and achieving efficient dual-mode operation in air and water. Although this design organically integrated two power systems, it did not account for the operational transition between water and air mediums. Moreover, during the transition from air to underwater operation, vehicles using this engine had to pause on the water surface for cooling, negatively impacting operational efficiency and stealth.

In 2022, a research team led by Ang Haisong at Nanjing University of Aeronautics and Astronautics [18] proposed a variable-axis propeller design for cross-medium unmanned vehicles. The system combined brushless DC motors, batteries, air propellers, and water propellers. Other researchers, such as Wang Ge at Harbin Engineering University and Xia Zhijun at the National University of Defense Technology, explored cross-medium ramjet engine concepts powered by oxygen-deficient metal-based fuels, achieving integrated propulsion for both aerial and underwater operations [16].

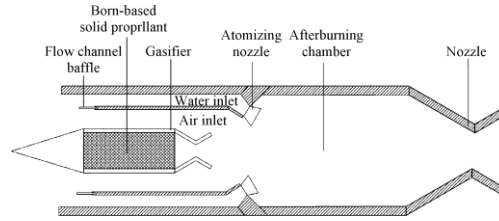


Fig. 18. Preliminary concept of boron-based cross-medium ramjet engine by the National University of Defense Technology

Currently, ramjet propulsion systems for air-water vehicles remain in the conceptual design stage. While theoretical feasibility has been established, significant technical challenges remain, including overall system configuration, thrust regulation, mode transition, and ignition and combustion of metal-based fuels.

5 Conclusion

This paper provides a comprehensive review of the current design and development strategies for trans-medium vehicles. First, it traces the development of trans-medium vehicles, starting from the emergence of the concept, the conceptual design of prototypes, and experimental development, to the successful testing and improvement of prototypes. A brief overview of several notable vehicles from the 1930s to 2022 is included.

Next, it details the development of three commonly used wing design configurations at the current research and development stage: swept wings, multi-rotor wings, and hybrid wings. The performance and drawbacks of representative prototypes for each design are analyzed. As of now, no single trans-medium vehicle design can satisfy all application scenarios. While swept-wing vehicles, owing to their biomimetic principles, show the greatest potential during trans-medium transitions (e.g., entering and exiting water), they face technical challenges in propulsion design, motion control, and endurance. Multi-rotor vehicles, as the most widely used UAV structure, feature relatively mature design technologies. However, achieving more flexible and repetitive medium transitions requires further research on optimizing the weight, size, and control systems of propulsion systems. Hybrid-wing vehicles, though offering relatively lower performance, have shown successful prototype test results, but their design optimization is still in iterative development.

Finally, the propulsion systems of trans-medium vehicles are categorized into two types for analysis: independent air-water propulsion systems and integrated air-water propulsion systems. A comparative analysis reveals that independent air-water propulsion systems often employ separate propulsion systems for air and water, which leads to issues such as low integration, excessive system weight, and long response times for switching between motion modes, making them insufficiently flexible for practical applications. Integrated air-water propulsion systems, with their higher degree of integration, better meet performance requirements. However, battery technology, motor design, and fuel selection for these systems still face significant bottlenecks. The

development of air/water ramjet engine technology offers potential for further improving speed, range, and endurance, making it an ideal propulsion method for future air-sea trans-medium weapons. Nonetheless, critical technical challenges must be overcome before it can be applied in actual engineering contexts.

In the future, the design schemes for various forms of trans-medium vehicles remain a subject of further research for scientific teams. Issues such as the selection of materials for vehicle manufacturing, optimization of motion control algorithms, and improvement of efficiency in entering and exiting water present many promising research directions.

References

- [1] Wu, Z. (2021). Bionic design and water entry performance research of cross-medium aircraft based on the kingfisher's water entry strategy [Master's thesis, Jilin University]. <https://doi.org/10.27162/d.cnki.gjlin.2021.000827>
- [2] He, Z., Zheng, Z., Ma, D., & others. (2016). Development and insights of foreign cross-medium aircraft. *Ship Science and Technology*, 38(09), 152–157.
- [3] Yang, X., Liang, J., Wen, L., & others. (2018). Research status of amphibious cross-medium unmanned aerial vehicles. *Robot*, 40(01), 102–114. <https://doi.org/10.13973/j.cnki.robot.170241>
- [4] Maia, M. M., Soni, P., & Diez, F. J. (2015). Demonstration of an aerial and submersible vehicle capable of flight and underwater navigation with seamless air-water transition. *CoRR*.
- [5] Chen, H. (2019). The design and analysis of fluid dynamic characteristics for submersible unmanned aerial vehicles [Master's thesis, Nanjing University of Aeronautics and Astronautics].
- [6] Lu, D., Xiong, C., Zhou, H., & others. (2020). Design, fabrication, and characterization of a multimodal hybrid aerial underwater vehicle. *Ocean Engineering*, 219, 108324.
- [7] Marks, P. (2010). From sea to sky: Submarines that fly. *New Scientist*, 207(2767), 32–35.
- [8] Yao, G., Li, Y., Zhang, H., Jiang, Y., Wang, T., Sun, F., & Yang, X. (2023). Review of hybrid aquatic-aerial vehicle (HAAV): Classifications, current status, applications, challenges, and technology perspectives. *Progress in Aerospace Sciences*.
- [9] Yao, G., Liang, J., Wang, T., Yang, X., Liu, M., & Zhang, Y. (2014). Submersible unmanned flying boat: Design and experiment. 2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014), 1308–1313. IEEE.
- [10] Drews, P. L., Neto, A. A., & Campos, M. F. (2014). Hybrid unmanned aerial underwater vehicle: Modeling and simulation. 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, 4637–4642. IEEE.
- [11] Alzu'bi, H., Akinsanya, O., Kaja, N., Mansour, I., & Rawashdeh, O. (2015). Evaluation of an aerial quadcopter power plant for underwater operation. 2015 10th International Symposium on Mechatronics and its Applications (ISMA), 1–4. IEEE.

- [12] Li, L., Wang, S., Zhang, Y., Song, S., Wang, C., Tan, S., Zhao, W., Wang, G., Sun, W., & Yang, F. (2022). Aerial-aquatic robots capable of crossing the air-water boundary and hitchhiking on surfaces. *Science Robotics*, 7(66), Article eabm6695.
- [13] Zeng, Z., Lyu, C., Bi, Y., Jin, Y., Lu, D., & Lian, L. (2022). Review of hybrid aerial underwater vehicle: Cross-domain mobility and transitions control. *Ocean Engineering*, 248.
- [14] Shao, D. (2020). Power analysis of cross-medium flying aircraft. *Aerospace Power*, (01), 12–15.
- [15] Zhu, S., Wang, Y., & Liu, W. (2011). Analysis of intake and exhaust systems for water-air UAVs. *Aeronautical Science & Technology*, (4), 83–85.
- [16] Liu, M., Xiong, L., Huang, H., & others. (2024). Development status and key technology analysis of cross-medium power system schemes. *Journal of Naval Aviation University*, 1–13. Retrieved from <http://kns.cnki.net/kcms/detail/37.1537.V.20241024.1340.002.html>
- [17] Zhong, X., Wang, Y., & Xia, J. (2016). The conceptual design and research on new air-water engines. *Journal of Aerospace Science and Technology*, 4(2), 16–24.
- [18] Ang, H., & Wang, Y. (2022). Design and control technology of a variable axis propeller unmanned vehicle for water and air domains. *Unmanned Systems Technology*, 5(3), 1–11.

Study on the Fluid Dynamics of Bottle Emptying

Shenglin Yue^{1,*}, Xiaotian Dong², Rongtian Na³, Zhixin He⁴

{sy3u24@soton.ac.uk¹, dongxt@jou.edu.cn²,

rongtian.na@mail.utoronto.ca³, hezhx28@mail2.sysu.edu.cn⁴}

University of Southampton, School of Ocean and Earth Science, Southampton, United Kingdom¹

Jiangsu Ocean University, School of Civil and Ocean Engineering, Lianyungang, China²

University of Toronto, St. George Campus, Department of Earth Science,

Department of Mathematics, Toronto, Ontario, Canada³

Sun Yat-Sen University, Department of Civil Engineering, Zhuhai, China⁴

*corresponding author

Abstract. The bottle pouring phenomenon has been studied due to its complex process and unique industrial value. This study, based on pouring experiments with six different bottles, discovered a strong quadratic relationship between emptying time and the inclination angle. The findings of Clanet and Serby were refined, resulting in an equation for emptying time that accounts for variations in inclination angle. The emptying process of the bottle can be divided into two stages: the bubble stage and the flow stage. It was observed that as the inclination angle increases, the "bubble stage" occupies a longer duration. Combining experiments and CDF, the exponential relationship between the relative maximum flow rate and the relative bottle mouth diameter was obtained by regression method.

Keywords: Emptying bottle, Fluid dynamics, CFD

1 Introduction

The way a liquid is in a bottle, which we often encounter daily performing actions as basic as pouring drinks into glasses, empties require a complicated interplay between the gas and liquid phases known as the "glug-glug" effect. This effect is known for a unique periodic acoustic signature of liquid egress counterposed with air bubble Ingres. A detailed study of this process would direct to industrial applications such as optimizing container designs.

Studying the "glug-glug" effect will better interpret container water flow and understand fluid mechanics. These flow patterns observed during the emptying process of an ideal verticle bottle were first investigated by Clanet and Serby in 2004 [1], based on the experimental result of Davis and Taylor (1988) [2]. A spring-mass analogy model was suggested, describing the duration of emptying in a power-law function with respect to bottle outlet diameter, which affects bottle parameter dimension designs [1].

The **Clanet and Searby formula** expresses the predicted emptying time T_e relative to the unrestricted emptying time T_{e0} as:

$$\frac{T_e}{T_{e0}} = \left(\frac{D_0}{d} \right)^{5/2} \quad (1)$$

where D_0 is the diameter of the tube and d is the diameter of the outlet. This relationship indicates that the emptying time increases with the ratio D_0/d , due to the greater challenge in synchronizing air ingress and liquid egress.

This time, an experimental study was conducted by Kenton, Neufeld, and Huppert (2012) investigating the impact of physical parameters on the emptying process, mainly focusing on the diameter of the bottleneck or shape type for different liquid properties. The results suggest a clear relationship between bottle geometry (the shape), flow angle (tilt), and exit diameter with respect to how fast the emptying process is. The study finds the best incline angle and outlet diameter to reduce emptying time, which applies to industry designs [3].

The **Hans C. Mayer formula** quantifies the emptying process, stating that the emptying time T_e is related with the volume V of the bottle and the outlet diameter d [4]. It is represented as:

$$T_e \sqrt{\frac{g}{d}} = (3.8 \pm 0.4) \left(\frac{V}{d^3} \right)^{(0.90 \pm 0.02)} \quad (2)$$

where g represents the acceleration due to gravity. The formula highlights that a larger bottle volume relative to the outlet size results in a longer emptying time, highlighting the significant role of bottle geometry in fluid discharge.

Mer et al. (2019) conducted more investigation of emptying dynamics in a study. Their study found that the larger neck diameter ratio, losses faster with decreased emptying time, and vice versa at a higher initial fill ratio. [5].

The **Whalley formula** provides a comprehensive model by considering the densities of both the liquid (ρ_L) and gas (ρ_G) phases [6]. The emptying time T_E is given by:

$$T_E = \left(\frac{(\rho_G^{1/4} + \rho_L^{1/4})}{[(\rho_L - \rho_G)gd]^{1/4}} \right)^2 \left(\frac{4V}{\pi d^2 C^2} \right) \quad (3)$$

where C is the Wallis constant (0.9 to 1). This formula produces veridical predictions for a subtle view of the interaction between liquid and gas phases during emptying.

Recently, Rohilla and Das (2020) divided the emptying process of the bottles vertically according to the flow characteristics of the bottle emptying process, and found that parameters such as the rising rate of the bubbles at the bottle mouth were affected by the inclination angle and the viscosity of the emptying liquid. At the same time, the main influence of the evacuation process was determined by quantifying the Re number and the We number [7].

Computational Fluid Dynamics (CFD) analysis can handle complex hydrodynamic phenomena involving multiple phase flows [8]. Schwefler (2021) continued with this analysis, including open, closed, and inverted bottle types. The transition from jetting to bubbling flow started under high

liquid pressure during this process. When the liquid level started dropping, bubbles became apparent, and the flow converted into a bubbled state. Nevertheless, current CFD simulations were found to need to be more accurately distinguishing this transition phase [9].

These studies cover many aspects and factors that contribute to the formation and kinetics of the "glug-glug" effect and emptying from bottles.

Despite these improvements, the role of lateral inclination and pitcher shape on emptying under natural conditions, that is during a meal or as part of daily activities has not received enough consideration. Consequently, the present study was conducted under standard atmospheric pressure (101 kPa) and at a temperature of 24°C.

2 Methodology and Experiment Setup

2.1 Experimental Setup

The experiments were conducted with five different models of glass bottles, which varied in shape, size, and consumption output style. These bottles were chosen to cover a variety of geometric variability, which affect the fluid dynamics at the moment within any given pour. The bottles were filled with water to ensure identical initial conditions and placed on an adjustable metal stand. The stand was machined to allow the bottle's inclination angle from vertical upright pouring to tilted pouring at various angles.

The mouth of the bottle, covered with a plastic board, is firmly pressed onto the outlet to apply uniform pressure to start emptying. After the quick removal of this board, flow started with a timer to use at a time when the water was emptied in total.

We conducted a series of seven trials per angle and bottle configuration for each experimental setup to ensure statistical significance. The experiments were conducted in a standard laboratory environment with pre-set conditions so that external factors like air currents or temperature changes would not affect the results. The standard error percentage of each bottle per angle was calculated, given by:

$$\delta = \frac{s}{t\sqrt{n}} \quad (4)$$

Where s is the standard deviation of the results, n is the number of trials, and t is the mean of the experimental results. To maintain a standard error margin within 1.5%. The preciseness is a guarantee for the repeatability and accuracy of the results.

Near the system is a high-speed camera, which can record slow-motion video to visualize further how material flows out of it. The formation and behavior of vortices, air bubbles, and liquid streams, as well as a better interpretation of the "glug-glug" effect, can be seen in videos.

2.2 Experimental Results

2.2.1 Influence of Bottle Shape

In order for the experimental results to reflect the majority of bottle types on the market, the research group conducted multiple experiments using five common glass bottles of different shapes

and sizes. The specific shapes of the bottles are shown in Figure 1. Table 1 provides the characteristic parameters of each bottle and Figure 2 gives an illustration of these parameters.

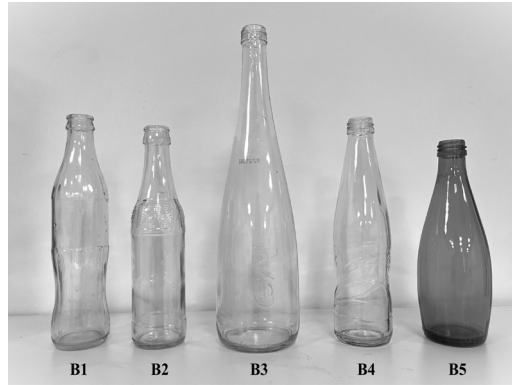


Fig. 1. The five bottles used in the experiment are named B1, B2, B3, B4, and B5 from left to right.

Table 1: Detailed numerical specifications for each bottle.

Bottle	Bottle mouth inside diameter (mm)	Bottle height (mm)	Bottle outer diameter (mm)	Thickness (mm)	Volume (mL)
1	17.0	216	52.8	4.0	296
2	18.0	206	53.5	6.0	275
3	16.7	298	80.0	7.0	800
4	18.0	211	57.6	3.5	300
5	17.2	189	57.3	3.8	355
Graduated Cylinder	49.0	352	53.0	2.0	665

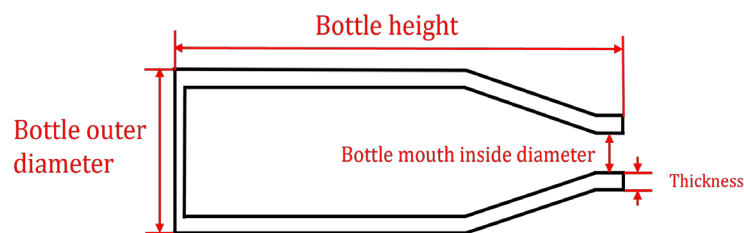


Fig. 2. Diagram of bottle body parameters.

The emptying times of the different bottles at various tilt angles are shown in Figure 2. The results indicate a strong correlation between the bottle mouth diameter and the emptying time. For

B1, B2, and B4, which have similar volumes and shapes, the larger the bottle mouth, the shorter the emptying time at similar tilt angles. Additionally, for the graduated cylinder, which can be equated to a water bottle with a neck and body of equal width, its emptying time is the fastest among all the measured containers, despite having a volume greater than 600 milliliters, much larger than B1, B2, and B4.

Another thing that should be notice is although there are differences in the emptying times among the different bottles, it can be observed that the relationship between bottle tilt angle and emptying time is consistent. Visually, the emptying time of each bottle exhibits a trend of initially decreasing and then increasing. A more in-depth discussion of the effect of tilt angle on emptying time will be provided in section 2.2.2.

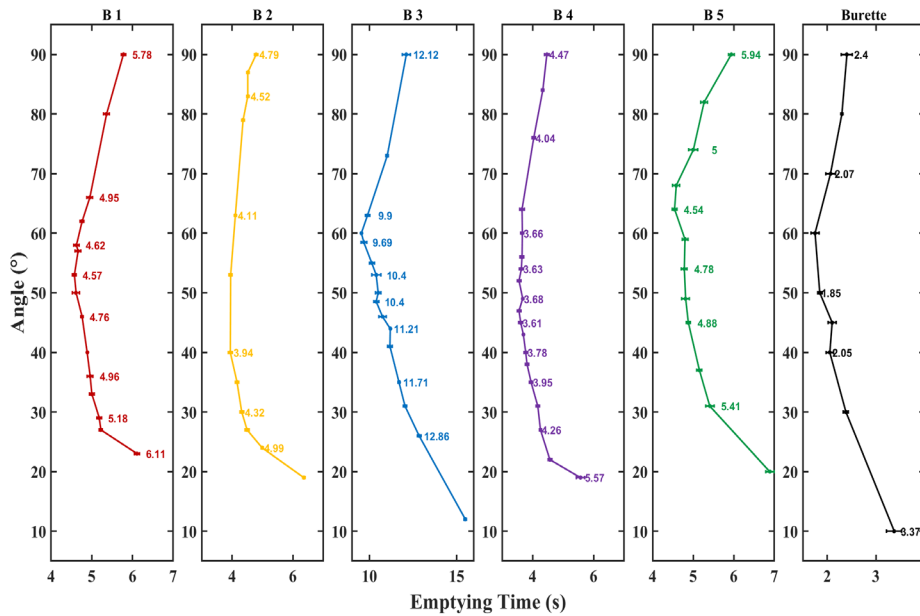


Fig. 3. The relationship between emptying time and angle for each container. From left to right are B1, B2, B3, B4, B5, and the graduated cylinder.

2.2.2 Influence of Bottle Inclination

To ignore the bottle's volume and mouth size and focus on the change in pouring angle over time, the experimental data were further processed. For each bottle, the emptying time at a 90° inclination angle was used as a reference baseline, yielding dimensionless parameters

$$\lambda = \frac{T_{emptying}(\theta)}{T_{emptying}(90)} \quad (5)$$

Where $T_{emptying}(90)$ represents the emptying time at a 90-degree tilt angle, and $T_{emptying}(\theta)$ represents the emptying time at each specific angle.

For each bottle studied, starting from a 90° tilt angle, the emptying time decreases as the angle gradually decreases. However, after reaching a certain specific angle, as the angle continues to decrease, the emptying time begins to increase. The relationship between λ and the inclination angle for each bottle is illustrated in Fig. 3. For Bottles 1, 2, and 4, which have similar shapes, characteristic angle with minimum backward time is very close. In addition, for B1 B2 and B4, the direction of the folds and the trajectories in Fig. 3 almost coincide. Bottles 3 and 5, which have more distinct shapes, show some differences in their 'specific angles,' but the pattern of first decreasing and then increasing emptying time is consistent with the previously mentioned bottles. The study also found that the tilt angle significantly impacts the emptying time; optimizing the tilt angle for the same bottle can reduce the emptying time by approximately 20 % which can be obtained in the figure 3.

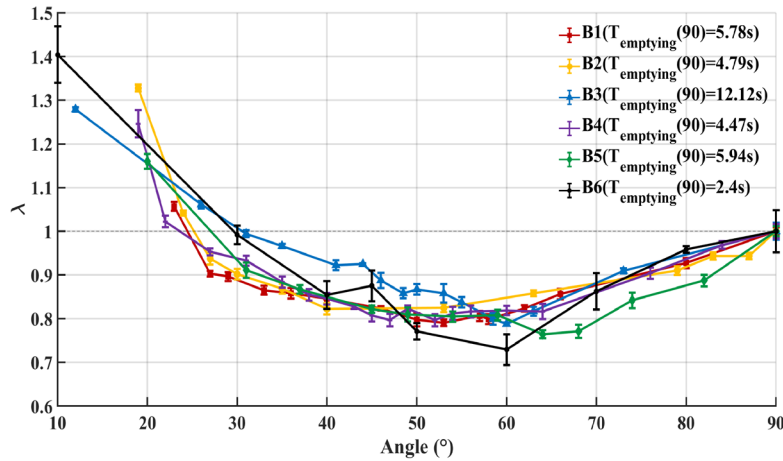


Fig. 4. Relationship curve between tilt angle and λ .

2.2.3 Influence of Surface Tension

The water emptying test with a variable surface tension (by adding liquid soap to water has been done, with the weight ratio of 1/181), but with regard to the emptying time, the obvious differences were not detected (see table 4) suggesting that surface tension plays a minimal role in the emptying time, at least for the scale that is considered in this research.

Table 2: Effect of surface tension on the emptying time T_f .

θ (°)	$T_{f,Pure\ Water}$ (s)	$T_{f,soap\ water}$ (s)
47	4.72	4.72
66	4.95	4.90
90	5.78	5.74

2.2.4 Discussion on Flowing Status

Through high-speed cameras, it was found that for B1 B2 B3 B4 and B5, which bottle mouth inside diameter were all smaller than 21mm, the process of emptying bottle can be described very clearly: first, the water clump flows out of the bottle opening under the action of gravity, while gas enters to form bubbles and rises, and then the water clump falls again. Starting from a certain time t_0 , the falling of water no longer leads to the generation of bubbles, and the turbulent flow disappears, replaced by water flowing out in a very smooth manner, which is similar to the research conclusions of Geiger et al. (2012). In the subsequent analysis, the flow stage before $t < t_0$ is called the "bubble state", and the flow stage after $t > t_0$ is called the "flow dynamic". In order to characterize the influence of the inclination angle on the "bubble state" flow, a dimensionless number μ is defined as follows:

$$\mu = t_b / (t_b + t_f) \quad (6)$$

where t_b is the time used in the "bubble state", t_f is the time used in the "flow dynamic", and Figure 5 shows the relationship between μ and angle, where the black curve is obtained by a fourth-order polynomial fitting in Matlab. It can be found that as the inclination angle increases gradually, μ value generally increases, that is, the "bubble state" proportion increases. One possible reason is that a larger inclination angle will increase the liquid's pressure gradient force, which will accelerate the flow speed of the "flow dynamic" liquid and shorten the flow time of this stage, ultimately resulting in an increase in the flow time of the "bubble state" stage. Another speculation is that a larger inclination angle will reduce the remaining liquid volume at t_0 ,

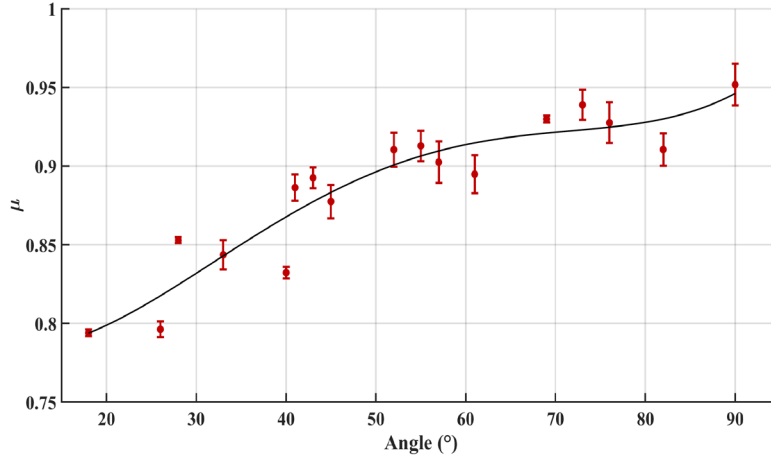


Fig. 5. Percentage of bubble state versus inclination angle.

3 Models

3.1 Modified C&S Formula

Based on C&S Formula [1], for an ideal tubular container, its emptying time at 90° is highly correlated with the shape parameter D_0, d and L of the container itself, as demonstrated by Eq. (1), where

$$T_{e0} = \frac{3L}{\sqrt{gD_0}} \quad (7)$$

which denotes the emptying time when the diameter of the mouth of the bottle is the same as the average bottle neck. From this can obtained:

$$Te = \frac{nL}{\sqrt{gD_0}} \left(\frac{D_0}{d} \right)^{5/2} \quad (8)$$

However, for the six different bottles used in the experiment, a large discrepancy between the prediction results of the C&S formula and the real emptying time was observed, and calculations showed that the coefficient of determination of the original C&S formula, $R^2 = -4.36$. In order to make the C&S formula better respond to the emptying time of the real bottles, two different corrective solutions were adopted: 1. optimising the leading coefficients of $Te0$, i.e., changing the value of n in Eq. (8); 2. finding a new power-index relationship between $\frac{Te}{T_{e0}}$ and $\frac{D_0}{d}$. After calculations, two new

formulas for T_e are given:

$$Te' = \frac{1.8L}{\sqrt{gD_0}} \left(\frac{D_0}{d} \right)^{5/2} \quad (9)$$

$$Te'' = \frac{3L}{\sqrt{gD_0}} \left(\frac{D_0}{d} \right)^2 \quad (10)$$

The results of the Eq. (9) and (10) are shown in Figure 6 respectively. The horizontal coordinates in the graph represent the experimental results, and the vertical coordinates represent the results of three different formulas. Among them, the R^2 of the result of Eq. (9) is 0.62 and Eq. (10) results in the R^2 of 0.85. It is easy to see that Eq. (10) has a better fit. Differences in the shape of the bottles can be considered as the reason for this change.

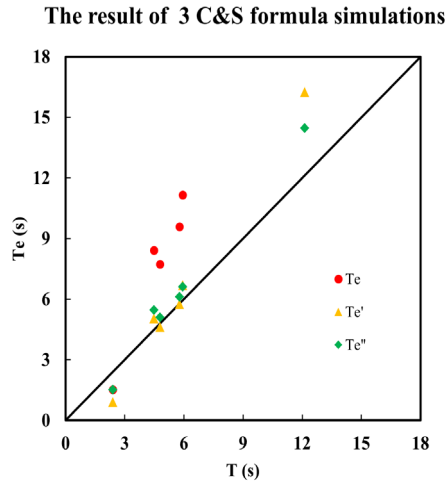


Fig. 6. Comparison of the results of the three formulas.

3.2 C&S Equation Based on Inclination Change

The variation of pouring time with inclination angle is discussed in Section 2.2.2. As can be obtained from Fig. 4, for the same inclination angle, there is a certain overall similarity between the λ for different shaped bottles, although there is a slight difference between them. To further investigate the relationship between emptying time and inclination, the dimensionless number θ' was used, where $\theta' = \theta/90$. After a quadratic polynomial fit, the equation for λ versus θ' is obtained:

$$\lambda = 1.9\theta'^2 - 2.4\theta' + 1.6 \quad (11)$$

Figure 7 illustrates the dimensionless number λ versus θ' , where the fit of Equation (11) is $R^2 = 0.8672$.

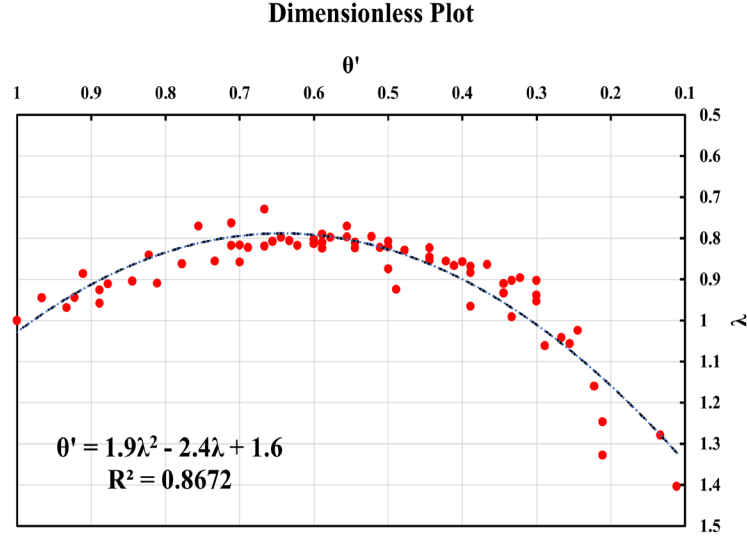


Fig. 7. Relationship between λ and θ' .

Substituting $T_{emptying}(90) = Te''$ into Eq. (5), associating Eq. (10) (11), we obtain:

$$T = [1.9(\frac{\theta}{90})^2 - 2.4\frac{\theta}{90} + 1.6] \frac{3L}{\sqrt{gD_0}} \left(\frac{D_0}{d}\right)^2 \quad (12)$$

Eq. (12) is called, the C&S equation based on inclination change.

4 CFD Simulation for Bottle Empty

The fluent model was set as transient, and the gravity acceleration in the Z direction was set as 9.81m/s^2 . Fluid-water was added, boundary conditions and grid were checked. Volume of Fluid model was set, and surface tension coefficient was set as 0.072N/m . Viscous model was set as SST $k-\omega$ viscous model. The Courant number was set as 1, and the time step can be determined by dividing the local mesh size by the characteristic flow velocity, so the time step was set from 0.001 to 0.025s.

4.1 Turbulence Modeling

RANS turbulence model was used for boundary layer resolved simulation of the bottle emptying in this paper. The RANS model of choice is the SST $k-\omega$ model by Menter (1994). The SST $k-\omega$ solve two prognostic equations: the turbulence kinetic energy, k , and the specific dissipation rate, $k-\omega$, which obtained from the following transport equations:

$$\frac{\partial \rho k}{\partial t} + \frac{\partial \rho k u_i}{\partial x_i} = \mu_t S^2 - \beta^* \rho \omega k + \frac{\partial}{\partial x_j} \left[\left(\mu + \frac{\mu_t}{\sigma_k} \right) \frac{\partial k}{\partial x_j} \right]$$

$$\frac{\partial \rho \omega}{\partial t} + \frac{\partial \rho \omega u_i}{\partial x_i} = \frac{\alpha \alpha^*}{\nu_t} \mu_t S^2 - \beta \rho \omega^2 + \frac{\partial}{\partial x_j} \left[\left(\mu + \frac{\mu_t}{\sigma_\omega} \right) \frac{\partial \omega}{\partial x_j} \right] + 2(1 - F_1) \rho \frac{1}{\omega \theta_\omega} \frac{\partial k}{\partial x_j} \frac{\partial \omega}{\partial x_j}$$

In these equations, ρ represents density of fluid. t represents time. x_i and x_j represents axis in the i and j direction. u_i and u_j represents velocity in the i and j direction. μ_t represents turbulent viscosity. μ represents viscosity. S represents modulus of the mean rate-of-strain tensor. σ_k and σ_ω are the turbulent Prandtl numbers for k and ω , respectively. β^* , α , α^* , F_1 , θ_ω are more functions and constants defined in Menter (1994).

4.2 Geometry Modeling

Geometry of the glass bottle inwall were measured according to the wall thickness and volume of the glass bottle. 3D solid model of the glass bottle were made and model quality was checked. The mesh was divided with multizone method, and the bottle mouth was set as pressure outlet boundary and the bottle body was set as wall boundary. As shown in the figure 4, two kinds of bottle models were constructed, bottle (a) is cylindrical, bottle (b) is narrow-mouth bottle with variable diameter.

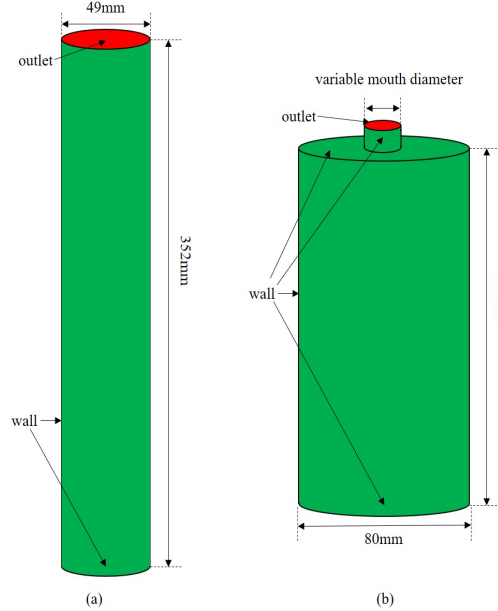


Fig. 8. Diagram of two kinds of bottle ((a) cylindrical bottle; (b) narrow-mouth bottle with variable diameter)

4.3 Convergence Test and Validation

4.3.1 Convergence Test

(1) Comparison of Mesh Generation Schemes

The 665mL cylindrical bottle (a) (diameter 49mm, height 352mm) was simulated by fluent under the upside-down condition. Different meshing and viscosity calculation schemes were set up, and the model was checked by experimental data to determine the model parameters.

Viscous model was set as SST k- ω viscous model as shown in the figure. Different mesh encryption dimensions were set, bottle empty time were calculated, and compared by experiment empty time.

Table 3: Comparison of mesh generation schemes

Case	Mesh number	Average mesh size(mm)	Time step(s)	Empty time(s)	Empty time-experiment(s)	Relative error(%)
C1	2116	17.7	0.035	2.500	2.400	4%
C2	5658	10.8	0.022	1.950		-19%
C3	30475	4.7	0.009	2.325		-3%
C4	51590	3.6	0.007	2.100		-13%
C5	126083	2.3	0.005	2.225		-7%
C6	255750	1.6	0.003	2.440		2%
C7	518830	1.1	0.002	2.500		4%

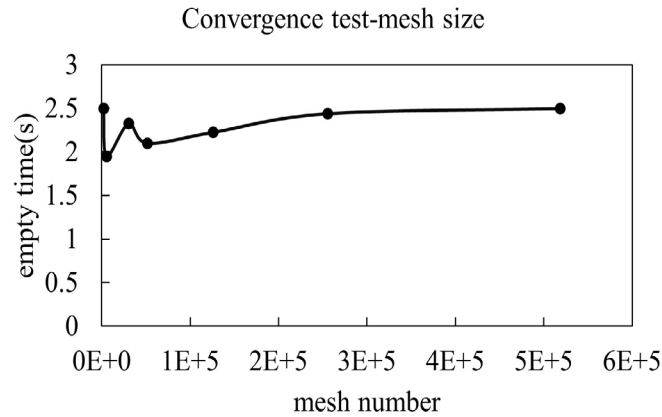


Fig. 9. Diagram of the convergence test on mesh size

It can be seen that the case C6 converges and has the smallest relative error, so average partitioning size was set as 1.6mm.

(2) Viscous model

Based on the Case C6, using the fluent meshing method with time step of 0.003s, comparing SST k- ω and laminar viscosity model, bottle empty time was calculated, and contrasted with experiment empty time.

Table 4: Comparison of viscous model schemes

Case	Viscous model	Empty time(s)	Empty time -experiment(s)	Relative error(%)
C6	SST k- ω	2.440	2.400	2%
C6 laminar	laminar	2.175		-10%

It can be shown that the relative error of Case C6 is the smallest, so SST k- ω viscous model was used.

4.3.2 Validation

Empty experiments were performed on a 665mL cylindrical glass bottle (49mm in diameter and 352mm in height) with different angles. Since the opening direction of the glass bottle geometry was the positive half axis of the Z axis, the acceleration of gravity at different angles was set and the bottle emptying time was calculated, and contrasted with experiment empty time.

Table 5: Bottle empty time comparison with different incline angle

incline angle(°)	empty time(s)	empty time -experiment(s)	relative error(%)
90	2.44	2.40	4.2
80	1.95	2.30	-13.0
70	1.78	2.07	-11.8
60	1.63	1.75	-4.3
50	1.59	1.85	-12.2
45	1.59	2.10	-22.6
40	1.61	2.05	-19.5
30	1.71	2.38	-26.5
10	2.51	3.37	-23.6

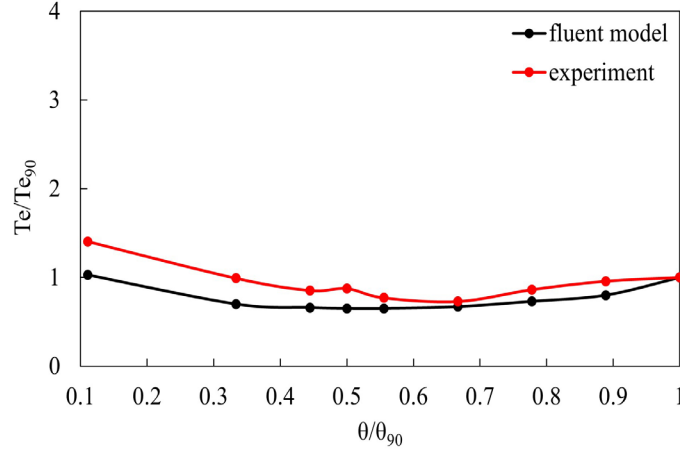


Fig. 10. Diagram of the relationship between relative empty time and relative incline angle

Since the end criterion of the empty bottle experiment is judged by no continuous flow which may last for several seconds, and in CFD simulation, 1-10 drops of water per unit time are generally used as the judgment standard, which will lead to large errors, as shown in table 5.

4.4 Results and Discussion

As can be seen in fluent model, there is no flow for $d < 5\text{mm}$ due to surface tension effects. In the region of counter flow (outlet diameter $d > 21\text{ mm}$), the outlet provides enough space so that water and air can pass each other simultaneously in the in- and out- flow directions. In the region of oscillatory flow (d is between 5 and 21 mm), flow pattern is characterized by four (cyclic) stages: liquid downflow, bubble rise, repressurization, and refill, which is similar to flow pattern analysis proposed by Geiger et al. (2012). According to fluent test, relative maximum flow rate had an exponential relationship with relative bottle mouth diameter as shown in figure 7.

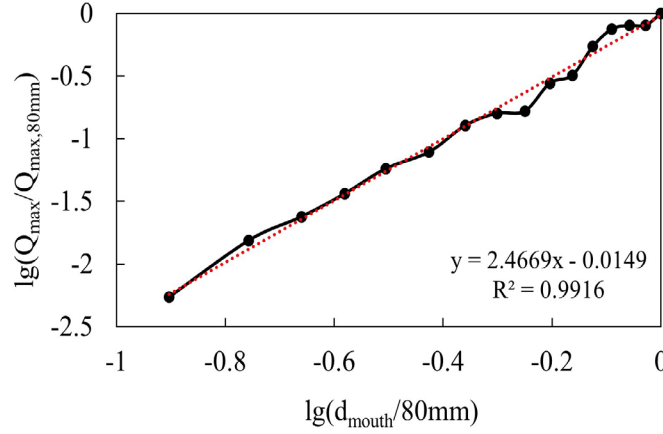


Fig. 11. Diagram of the relationship between relative maximum flow rate and relative bottle mouth diameter

5 Conclusion

For realistic bottles, their bottle calibre and pouring time show a certain correlation, and for bottles of similar volume, the larger the bottle calibre, the faster the flow emptying time.

A very consistent relationship was found between the change in inclination angle and the pouring time. Overall, the pouring time increases and then decreases as the angle becomes progressively smaller, while the shortest pouring time occurs between 40° and 60° , which is about 80% of the pouring time at 90° .

For surface tension, the effect of its change on the inversion time was not obvious in the experiments.

For the flow regime, the flow voiding process is divided into "bubble state" and "flow dynamics", and the percentage of the "bubble state" phase decreases as the tilt angle decreases.

Based on empty bottle experiment and fluent simulation, three different flow regimes with different bottle mouth diameters were analyzed: no flow, counter flow, and oscillatory flow. Exponential relationship between relative maximum flow rate and relative bottle mouth diameter was obtained by regression method.

Acknowledgement

Shenglin Yue, Xiaotian Dong, Rongtian Na and Zhixin He contributed equally to this work and should be considered co-first authors.

References

- [1] Clanet, C. and Searby, G. (2004) 'On the glug-glug of ideal bottles', *Journal of Fluid Mechanics*, 510, pp. 145–168. Available at: <https://doi.org/10.1017/S002211200400936X>.
- [2] Davies, R.M. and Taylor, S.G. (1988) 'The mechanics of large bubbles rising through extended liquids and through liquids in tubes', in *Dynamics of Curved Fronts*. Elsevier, pp. 377–392. Available at: <https://doi.org/10.1016/B978-0-08-092523-3.50041-1>.
- [3] Kenton, Z.A., Neufeld, J.A. and Huppert, H.E. (2012) 'Emptying Bottles: A Study of Glugging'.
- [4] Mayer, H.C. (2019) 'Bottle Emptying: A Fluid Mechanics and Measurements Exercise for Engineering Undergraduate Students', *Fluids*, 4(4), p. 183. Available at: <https://doi.org/10.3390/fluids4040183>.
- [5] Mer, S. et al. (2019) 'Emptying of a bottle: How a robust pressure-driven oscillator coexists with complex two-phase flow dynamics'. Available at: <https://doi.org/10.1016/j.ijmultiphaseflow.2019.05.012>.
- [6] Whalley, P.B. (1991) 'Two-phase flow during filling and emptying of bottles', *International Journal of Multiphase Flow*, 17(1), pp. 145–152. Available at: [https://doi.org/10.1016/0301-9322\(91\)90076-F](https://doi.org/10.1016/0301-9322(91)90076-F).
- [7] Rohilla, L. and Das, A.K. (2020) 'Fluidics in an emptying bottle during breaking and making of interacting interfaces', *Physics of Fluids*, 32(4), p. 042102. Available at: <https://doi.org/10.1063/5.0002249>.
- [8] Silva, L.F.L.R., Damian, R.B. and Lage, P.L.C. (2008) 'Implementation and analysis of numerical solution of the population balance equation in CFD packages', *Computers Chemical Engineering*, 32(12), pp. 2933–2945. Available at: <https://doi.org/10.1016/j.compchemeng.2008.03.007>.
- [9] Schwefler, C. (2021) *Analytical, Numerical, and Computational Methods to Analyze the Time to Empty Open, Closed, and Variable-Topped Inverted Bottles*. California Polytechnic State University. Available at: <https://doi.org/10.15368/theses.2021.88>.
- [10] Geiger, F., Velten, K. and Methner, F.J. (2012) '3D CFD simulation of bottle emptying processes', *Journal of Food Engineering*, 109, pp. 609–618. Available at: <https://doi.org/10.1016/j.jfoodeng.2011.10.008>.

Positioning and Search System for Submersibles: Model Construction, Results, and Future Prospects

Xueqi Tang^{1,a,*,†}, Zonghui Hua^{1,b,†}

¹Electronic Information School, Wuhan University, Wuhan, China

a. 1931183459@qq.com, b. 1463022570@qq.com

*corresponding author

†These authors contributed equality to this work

Abstract. This paper proposes a comprehensive positioning and search system for deep-sea submersibles to enhance efficiency in complex marine environments. The core position prediction model integrates Kalman and extended Kalman filters, accounting for submersible dynamics and geographical data. This approach effectively addresses nonlinear challenges from forces like buoyancy, gravity, ocean currents, and resistance. High-precision positioning on 3D topographic maps is achieved through optimized dynamic and observation equations. Equipment selection utilizes the TOPSIS model to evaluate eight deep-sea rescue tools, emphasizing functionality, cost-effectiveness, safety, and durability. Search efficiency is improved by integrating the position prediction model with the ant colony algorithm, reducing search paths and time in simulations. A multi-target cooperative position prediction model, incorporating multi-target and cooperative extended Kalman filters, supports multi-submersible coordination. Environmental adaptability is demonstrated in areas like the Caribbean and Ionian Seas, highlighting the model's robustness. While significant progress has been made, challenges remain in ensuring accuracy, stability, and feasibility in extreme conditions. Future research will focus on data collection, parameter optimization, and developing more efficient algorithms to expand the model's applicability in diverse marine scenarios.

Keywords: Submersible, Position prediction, TOPSIS, Search model, Model extension

1 Introduction

The development of submersible technology has revolutionized ocean exploration, offering a powerful tool for understanding the underwater world. Greek company MCMS has launched an advanced submersible for exploring shipwrecks in the Ionian Sea, but challenges such as mechanical failures and loss of contact with the mother ship pose serious risks to safety, financial stability, and reputation. Addressing these challenges is crucial to ensuring the success of such projects.

This research holds significant theoretical and practical importance, enriching the system of submersible positioning, searching, and rescuing in complex marine environments. By developing innovative models and algorithms, it enhances understanding of submersible-environment interactions and improves position prediction, contributing to the broader advancement of marine technology.

The study aims to create a practical positioning and search system to ensure safety and efficiency across diverse marine environments and multi-submersible scenarios. A robust position prediction model, considering factors like buoyancy, ocean currents, seawater density, and seafloor geography, is essential. Addressing uncertainties improves prediction accuracy and informs equipment selection for both mother and rescue ships. A search and rescue model will optimize deployment points and search patterns, minimizing response times. Leveraging accumulated data, it calculates the probability of finding submersibles over time for targeted rescues. The models must adapt to varying sea conditions and multi-submersible operations, ensuring reliability in diverse environments. This research aims to enhance operational safety, efficiency, and rescue success rates, contributing to the advancement of ocean exploration technology.

2 Related Work

In the field of research related to submersible positioning and search rescue, several important techniques and models have been studied and applied in different contexts.

Kalman filtering and its extended forms have been widely used in various fields such as aerospace and robotics navigation [1]. In these applications, they have demonstrated their effectiveness in estimating the state of dynamic systems [2]. However, when applied to the submersible domain, significant adjustments are required to account for the unique characteristics of the marine environment [3]. The complex and variable nature of the ocean, including factors such as currents, salinity, and temperature gradients, demands a more tailored approach to ensure accurate position prediction of submersibles [4].

The TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) model has been a prominent tool in multi-criteria decision analysis [5]. It has been applied in numerous scenarios where multiple factors need to be considered for evaluating alternatives. In the context of submersible equipment selection, it provides a framework for comparing different equipment options. However, it is crucial to adapt this model to the specific requirements of the submersible search and rescue scenario [6]. This involves carefully determining the relevant evaluation criteria and their respective weights based on the actual conditions and needs of the operation. For example, factors such as equipment functionality, cost, reliability, and durability need to be considered in a balanced manner to make an informed decision [7].

Weighted networks and ant colony algorithms have been extensively studied for path search and optimization problems [8]. These algorithms have shown promising results in finding optimal paths in various complex environments [9]. When applied to the search and rescue operations in the marine environment for submersibles, they face several challenges due to the complexity of the oceanic setting. The vastness of the ocean, the presence of multiple obstacles, and the dynamic nature of the environment require sophisticated modifications to these algorithms to ensure their effectiveness. The algorithms need to be able to handle the uncertainties associated with the submersible's position, the changing ocean currents, and the varying visibility conditions [10].

Current submersible positioning and rescue research for the Ionian Sea and MCMS project faces limitations. Key challenges include neglecting factors like seafloor topography and submersible interactions, inadequate equipment evaluation methods, and unoptimized rescue models, leading to safety risks and slower response times in complex marine environments.

3 Research Methods

3.1 Position Prediction Model Construction

To predict the position of the submersible, we utilize Kalman filtering and its extended form to integrate sensor information. Firstly, a dynamic physical model is established, taking into consideration the forces acting on the submersible and geographical environmental factors. The forces include buoyancy, gravity, current force, and resistance. Based on these factors, dynamic and observational equations are constructed. To address the nonlinear problems, improvement measures are introduced, and the model is extended to the extended Kalman filtering form. This allows for a more accurate prediction of the submersible's position in the complex marine environment.

3.2 TOPSIS Model Application

The entropy weight method is employed to determine the weights for a comprehensive evaluation of four aspects of deep-sea rescue equipment. Figure 1 provides a visual representation of the comprehensive evaluation process of the TOPSIS model [5].

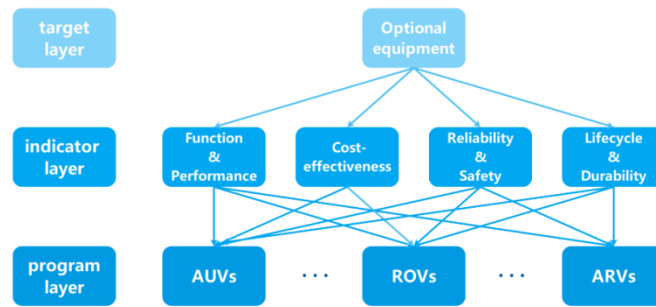


Fig. 1. TOPSIS Model

These four aspects typically include functionality and performance, cost-effectiveness, safety, and durability. By calculating the weights for each aspect, a more objective and comprehensive evaluation of the equipment can be achieved.

3.3 Search Model Construction and Optimization

Traditional Search Model: Data related to the submersible is collected, including its last communication location, motion vector, current velocity, and direction of the ocean current. Based on this data, a vector synthesis model is constructed to determine the initial search direction.

Novel Search Model: This model combines the position prediction model and the ant colony algorithm. Relevant parameters are defined to optimize the search path. The ant colony algorithm is introduced to consider the pheromone concentration, distance vector, and direction vector, etc. These parameters are adjusted according to the predicted position to improve the search efficiency.

4 Data Collection and Processing

4.1 Data Collection

These data are sourced from the global ocean model dataset, such as GEBCO 2023. The dataset provides information on various geographical environmental factors, including seawater temperature, ocean current strength, and seafloor topography. This data is crucial for understanding the environment in which the submersible operates, as shown in Figure 2, which presents a detailed view of the geographical environment data, such as the distribution of seawater temperature and ocean current strength.

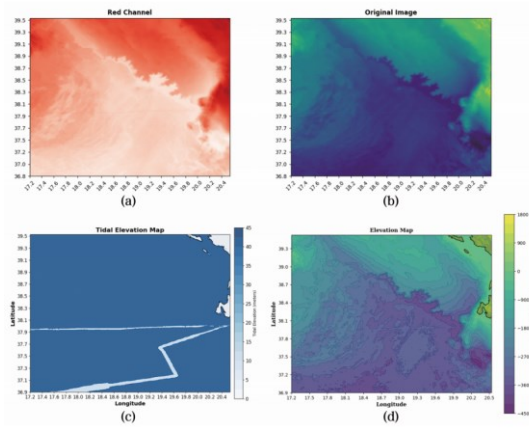


Fig. 2. Marine Environment Map Group

Data related to the submersible is collected through internal and external sensors. Internal sensors may include inertial measurement units (IMU), while external sensors can be sonar devices and GPS buoys. These sensors provide information on the submersible's position, velocity, and attitude. This data collection process is essential for accurately predicting the submersible's position and movement.

Information about equipment is obtained through literature search. This includes details about the functionality, performance, cost, and other characteristics of different types of equipment used in deep-sea rescue operations.

4.2 Data Processing

Sensor data for the position prediction model undergoes preprocessing (cleaning, normalization, transformation) for compatibility with Kalman filtering algorithms. For the TOPSIS model, equipment data is normalized, and indicators like entropy weight are calculated for evaluation. Predicted values are integrated with the ant colony algorithm to optimize search paths.

5 Experimental Results

5.1 Position Prediction Model

A position prediction model integrating Kalman and extended Kalman filtering techniques accounts for submersible dynamics, including buoyancy, gravity, currents, resistance, seawater density, and seafloor topography. Refined dynamic and observational equations ensure accurate predictions on 3D topographic maps. Figure 3 shows alignment with actual seabed features, while Figure 4 highlights analyzed uncertainties like sensor errors, process noise, and unmodeled dynamics, enhancing prediction reliability.

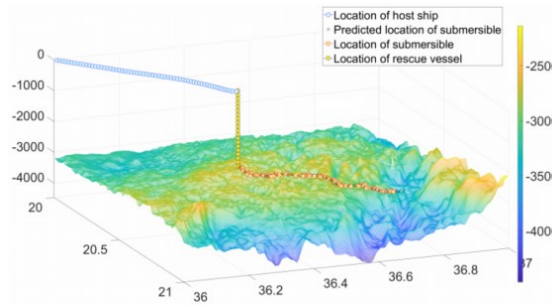


Fig. 3. Location Prediction Model

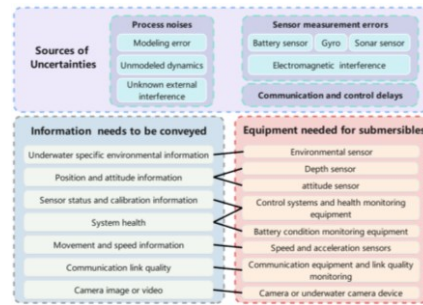


Fig. 4. Uncertainty Analyses

In various simulated scenarios, the model has demonstrated its adaptability and effectiveness. For example, in scenarios with different current velocities and directions, the model was able to adjust its predictions accordingly. The data shows that as the current velocity increased, the predicted position of the submersible deviated more from its initial position, but the model was still able to capture the general trend of the movement. In scenarios with varying seafloor topographies, such as slopes and ridges, the model accounted for these changes and provided more accurate predictions near the complex terrain areas.

5.2 TOPSIS Model

The TOPSIS model was applied to evaluate 8 types of deep-sea rescue equipment. By using the entropy weight method, weights were assigned to four key aspects: functionality and performance, cost-effectiveness, safety, and durability. This process involved a series of calculations, starting from data normalization of each equipment's performance indicators to the determination of information entropy and entropy weights.

The comprehensive evaluation results revealed significant differences among the equipment. The weights for functionality and performance, cost-effectiveness, safety, and durability were calculated as 0.5021, 0.1984, 0.1376, 0.1619. The final comprehensive scores for each equipment were as follows: Automated Underwater Robots (AUVs) scored 0.2791, Remotely Operated Vehicles (ROVs) scored 0.1710, and so on. These scores indicate that AUVs and ROVs generally outperformed the other equipment in terms of the overall evaluation, considering all aspects.

5.3 Search Model

The traditional search model, based on submersible data like last communication location, motion vector, and ocean currents, used vector synthesis and a weighted network to determine search paths. While effective in simpler environments, it struggled with strong currents or complex seabed topographies, leading to longer search times and less optimal paths.

The novel search model, integrating the position prediction model and the ant colony algorithm, demonstrated superior performance. By using predicted position data and optimizing search paths with the ant colony algorithm, it adapted effectively to environmental changes. As shown in Figure 5, the novel model achieved significantly shorter search paths and reduced search times compared to the traditional model. This highlights its enhanced efficiency and accuracy, making it more suitable for complex marine environments.

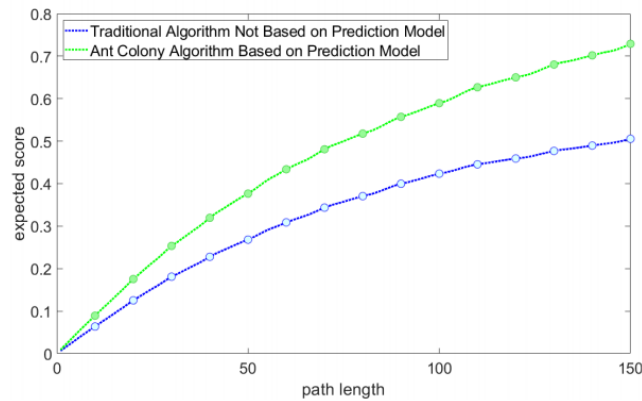


Fig. 5. Evaluation Results

6 Conclusions

In conclusion, a comprehensive submersible positioning and search system has been successfully constructed, incorporating multiple models and their extensions. While achieving certain results in various aspects, the research also has limitations. The models need to improve accuracy and stability in complex environments, optimize equipment selection indicators and weights, and further study the feasibility and complexity of the search model in extreme environments. Future prospects include collecting more data, introducing advanced technologies for optimization, researching better equipment evaluation methods, and exploring efficient algorithms to expand the application range of the models.

References

- [1] Simon D. Kalman filtering[J]. Embedded systems programming, 2001, 14(6): 72-79.
- [2] Chui C K, Chen G. Kalman filtering[M]. Berlin, Germany: Springer International Publishing, 2017.
- [3] Grewal M S, Andrews A P. Kalman filtering: Theory and Practice with MATLAB[M]. John Wiley & Sons, 2014.

- [4] Sorenson H W. Kalman filtering techniques[M]//Advances in control systems. Elsevier, 1966, 3: 219-292.
- [5] Lai Y J, Liu T Y, Hwang C L. Topsis for MODM[J]. European journal of operational research, 1994, 76(3): 486-500.
- [6] Behzadian M, Otaghsara S K, Yazdani M, et al. A state-of the-art survey of TOPSIS applications[J]. Expert Systems with applications, 2012, 39(17): 13051-13069.
- [7] Papathanasiou J, Ploskas N, Papathanasiou J, et al. Topsis[M]. Springer International Publishing, 2018.
- [8] Karaboga D, Akay B. A comparative study of artificial bee colony algorithm[J]. Applied mathematics and computation, 2009, 214(1): 108-132.
- [9] Newman M E J. Analysis of weighted networks[J]. Physical Review E—Statistical, Nonlinear, and Soft Matter Physics, 2004, 70(5): 056131.
- [10] Barrat A, Barthelemy M, Pastor-Satorras R, et al. The architecture of complex weighted networks[J]. Proceedings of the national academy of sciences, 2004, 101(11): 3747-3752.

Fluid Dynamics for Games: A Literature Review

Zhaorui Zhang^{1,a,*}, Yongzhi Zhuang^{2,b}, Yiqun Zhong^{3,c}, Bowen Chen^{4,d}

¹Northeastern University, Boston, United States

²Sichuan University, Sichuan, China

³Huazhong University of Science and Technology, Wuhan, China

⁴Communication University of China, Hainan, China

a. jaysonzr2002@gmail.com, b. nariyz@outlook.com, c. zhongyiqun0@gmail.com,

d. 202229013098N@cuc.edu.cn

*corresponding author

Abstract. Fluid simulation in video games presents significant challenges in balancing real-time performance and visual accuracy. This paper discusses developments concerned with fluid simulation techniques that optimally choose between computational efficiency and realistic fluid dynamics. Major techniques such as DCGrid, Incompressible Smoothed Particle Hydrodynamics (ISPH), Weakly Compressible SPH (WCSPH), Implicit Incompressible SPH (IISPH), and the Finite Volume Method (FVM) have been evaluated for application in several scenarios. This paper will highlight the trade-offs between accuracy and speed involved, especially in real-time simulations, and how each of these methods addresses such challenges. This paper aims at an in-depth understanding of the various fluid simulation strategies that can result in highly immersive and visually engaging gaming experiences.

Keywords: video games, Fluid Dynamics, fluid simulation strategies

1 Introduction

Fluid dynamics has become increasingly important with the development of technology in video games, where infusing fluids can create realistic, graphically beautiful game environments. Fluid effects like water, smoke, and splashes add a lot more to a game than just appealing visuals; they help the player feel immersed further into their virtual experience. Whether a character is wading through a river, waves crash onto the shore, or smoke billows from an explosion, realistic fluid simulation serves to provide more dynamic and interactive virtual worlds. However, the challenge in simulating these complex behaviors in real time faces performance constraints and often forces developers to make a trade-off between correctness and speed for every application.

Real-time fluid simulation in game development is very much a balancing act. On the one hand, there would be an approximation of high-fidelity visuals of the natural motion of fluid, while on the other hand, game engines do need to keep up with consistent performance. Fundamentally, two issues may be perceived in how to render fluid in an efficient manner without losing computational speed and how the realistic interactions between fluids and objects in a game, such as characters or terrain, are managed. Both of these are integral parts of the player's experience and need to be responsive and believable while running within the limits of real-time processing.

Over the years, a number of different techniques have been developed for propagating fluid simulation in games, each with its strengths determined by the demands to which the game may be put. This review will elaborate on some important methods of fluid simulation based on four papers, each with different advantages in different game scenarios. The first paper [1] mainly introduces the method DCGrid, which is a grid-based approach that automatically adapts the resolution of the grid w.r.t. the behavior of the fluid. It has smoother performance since more resources are put toward complex fluid interactions. The second [2] compares two particle-based techniques: Incompressible Smoothed Particle Hydrodynamics (ISPH) and Weakly Compressible SPH (WCSPH). Although ISPH gives higher accuracy with the help of constant fluid volume, WCSPH provides faster simulations, hence more applicable in real-time applications where speed is vital. The third paper [3] introduces an interesting thought based on the method Implicit Incompressible SPH (IISPH). It enhances the traditional IISPH in certain ways such that it increases stability and allows larger steps in time, hence enabling more complicated fluid simulations while keeping computational costs lower. Finally [4], the Finite Volume Method (FVM) divides the fluid into smaller volumes. Therefore, it is very powerful for big-scale water simulations that occur within open-world video games or places with vast bodies of water.

This review of such methods will ideally enable an understanding of how variant fluid simulation techniques can be put to work in game development and will, consequently, allow developers to select the best approach, given the requirements of a certain game. Be it computational speed or visual realism, understanding the strengths and weaknesses of these techniques is going to be crucial for effective fluid dynamics simulation in modern games.

2 Background

In the context of game development, fluid simulation plays a vital role in creating more realistic and immersive virtual environments. However, as described in the introduction, real-time fluid simulation is very challenging because of the intrinsic trade-off between visual accuracy and computational performance. The following introduces two basic viewpoints in the simulation of fluids: Eulerian and Lagrangian viewpoints, together with the explanation of the Navier-Stokes equations, the foundation of most fluid simulation methodologies [5]. It will be important to understand them in order to later on understand how most modern fluid simulation techniques are put into practice in games.

Fluid dynamics can be described from two primary perspectives: the Eulerian and Lagrangian viewpoints [5]. In the Eulerian viewpoint, attention is directed to fixed points in space, observing the change of properties of the fluid—such as velocity and pressure—with respect to time at these points. In this viewpoint, space is divided into a grid, and the fluid is followed in its motion across the grid. This typically occurs on the grid and is used in fluid simulations that call for fixed spatial references; hence, it is suitable for smoke or large water body simulations where precision is paramount.

In contrast, the Lagrangian viewpoint defines the motion of fluids by following every particle throughout their motions in space. Instead of the instantaneous properties of fluids at one stationary point, this viewpoint is interested in the trajectory of each particle. It is a particle-based method that is in common use within SPH, wherein fluids are modeled as an assemblage of interacting particles. SPH works well in the case of simulations where complicated and dynamic behaviors must be realized, including splashes and fluid-object interactions, hence allowing more flexibility in the representation of fluid motion in real-time games.

Each perspective has its advantages in game development. The Eulerian viewpoint is to be enabled on large-scale, gridbased simulations with full and accurate control over the activities of the fluid. This could be seen in FVM. On the contrary, Lagrangian viewpoint-based SPH techniques, such as WCSPH and IISPH, have much to offer to better simulate fluid and natural interactions with dynamic scenes with a common merit for which these techniques are popular in the implementations of real-time game applications.

These represent the mathematical basis for the simulation of the behavior of fluids, whereby the change in fluid velocity over time, under different forces, is described by NavierStokes equations. This equation forms the very necessary foundation for all the studies on fluid motion and finds its application in the Eulerian and Lagrangian methods of solution discussed above.

The two most important components of the Navier-Stokes equations are discussed below [5]. The first part is the momentum equation, describing how the velocity of a fluid changes under the action of forces: pressure gradients, viscosity (internal friction), and external forces like gravity. In other words, this equation describes how a fluid moves, responding to its internal properties and external forces such as wind or waves. This equation will give the developers of the game simulations a way to govern how a fluid reacts to characters or objects, or to terrain within a game setting; for example, this can make water splash when one jumps into the pool. The second part is the continuity equation, which furnishes the mass of the fluid that is conserved with the implication it can neither be created nor destroyed over time. This means in practice that the fluid flow is maintained constant and smooth. In game terms, that is to say, when water flows, there are no strange gaps or overlaps that make the simulation not realistic.

The fundamental equations for the calculation of the fluid motion, on the other hand, utilize the Navier-Stokes equations in the grid-based simulation of WCSPH and ISPH methods by computing particle displacements through the equation of momentum conservation and the continuity equation, with the latter providing density consistency over time. In turn, FVM applies these equations on a much smaller control volume so that the flow of fluids is accurately simulated in the case of large-scale simulations of bodies of water.

These complex equations, therefore, are simplified so that the developers can establish visually convincing simulations of phenomena without necessarily having to achieve other industries' required strict physical accuracy. According to [5], in most cases, speed and visual stability are more important than physical correctness in gaming.

3 Numerical Methods

In the context of games, there are many numerical methods developed for fluid dynamics simulation to turn out properly this challenging trade-off between realism and performance. These methods try to emulate such phenomena in a way that their simulation would be visually realistic while keeping computational efficiency so as to be eligible in real-time game environments. In this section, some of the important numerical techniques in modern fluid simulations will be reviewed and summarized, each suited to a different aspect of fluid behavior: handling large bodies of water, dynamic fluid-object interactions, and ensuring real-time performance without giving up too much on visual fidelity.

3.1 DCGrid

In [1], "DCGrid: An Adaptive Grid Structure for Memory-Constrained Fluid Simulation on the GPU" introduces the Dynamic Constrained Grid (DCGrid), an advanced grid method for fluid

simulation. This method enhances computational speed significantly while maintaining simulation accuracy.

Raateland et al. came up with a sparse grid structure ([1].Fig1), that has a hierarchical nature, and that structure is what the algorithm's data relies on mostly. In this system, the grid gets divided into levels, with each one having different resolutions, but those resolutions differ by about a factor of two between every level. The purpose behind this setup is to allocate memory and resources more freely while not affecting simulation precision. For example, where the fluid is smooth, a lower resolution is used; then, in areas with interaction or high turbulence, the resolution is raised for more accurate results. The grid is organized into blocks, with these blocks having sub-blocks, and inside them are multiple cells. But only the active ones, the cells that are in use, get memory. This design helps save storage and stops memory from being used unnecessarily.

There is a limit set on how many blocks are used, and all the levels have similar rules regarding memory usage. This prevents the algorithm from going over the available memory and ensures it operates smoothly within the constraints. The memory assignment happens in a linear way, and certain calculations are applied to map coordinates into memory positions through a hash table. This makes lookup times to remain at $O(1)$, which is designed to keep time efficiency high. Another technique used by Raateland et al. for better performance is precomputed apron cells ([1].Fig2). Apron cells refer to cells around the block being processed. These are identified first in the fluid boundary task, which enables the direct use of their data in computations, and this leads to improved effectiveness.

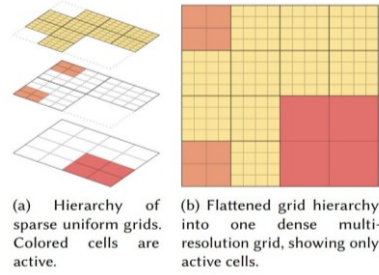


Fig. 1. Two-dimensional slice of the same hierarchy of sparsely populated uniform grids. The thicker lines indicate block boundaries.

16	17	20	21	32	33	36	37	80	81	84	85
18	19	22	23	34	35	38	39	82	83	86	87
24	25	28	29	40	41	44	45	88	89	92	93
26	27	30	31	42	43	46	47	90	91	94	95
108		109	0	1	4	5	48	49	52	53	
		109	2	3	6	7	50	51	54	55	
		111	8	9	12	13	56	57	60	61	
110		111	10	11	14	15	58	59	62	63	
64	65	68	69	128	128	128	128	129			
66	67	70	71								
72	73	76	77								
74	75	78	79								

Fig. 2. Apron cell indices as calculated for the central block.

In addition to that, to keep the grid data consistent, they introduced operations of restriction and prolongation ([1].Fig3). Restriction collects data from blocks with higher resolutions, averaging them and sending the result to lower resolution blocks to keep the lower grid aligned with changes in the higher one. On the other hand, prolongation does the opposite by sending data from the lower resolution to the higher resolution blocks. This structure makes the algorithm good for large-scale parallel GPU tasks, keeping data in order while ensuring that calculations are fast.

Raateland et al. put forward a very comprehensive implementation for that algorithm. And DCGrid's topological adjustments rely on some key activities. Firstly, it is the priority score that plays a role here. They provided a kind of mathematical formula for it. This score defines how each grid cell resolution happens. The physical parameters, such as gradients in velocity and intensity of vorticity, are relevant for deciding. High-priority cells will have higher resolution, which ensures critical areas have more details. Then, topology adaptation takes place through both refinement as well as coarsening procedures. Refinement means new blocks are inserted, turning low-resolution grids into ones of higher resolution, and the prolongation part in this step helps the newly formed block have some reasonable initial state. Coarsening, on the other side, merges grids with high resolution into those of lower resolution. The efficiency of the hash table is kept by the algorithm refilling it periodically, which avoids inactive keys taking up the space and making search processes less efficient.

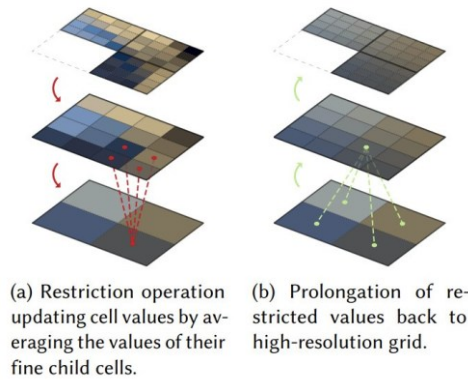


Fig. 3. Restriction and prolongation operations performed after each other on the same data.

In the beginning, there is a global block limit being set up, and that controls the maximum number of blocks allowed at all levels. Then, how blocks are allocated in each level will depend on total memory availability and leftover space can be used for refining sub-blocks from lower levels that are more high-priority. The block re-arrangement ([1].Fig4) is another important thing in the algorithm's process. This happens after every timestep. The grid will be refined or coarsened based on those priority scores, where blocks with higher ones are refined, while lower ones get coarsened to save resources. To keep computational costs down, the algorithm has a move limit that tells how many adjustments happen each timestep, but this move limit isn't fixed; it will be adjusted according to system needs with a model that predicts needs simply. After each topological change, apron cells will be updated. That way, boundary cell data in numerical computations are kept right. At the same time, restriction and prolongation steps make sure data stays consistent between grids of high- and low-resolution levels.

Raateland et al. through a series of tests, looked at DCGrid’s advantages. The tests had smoke and cloud simulations over some complicated areas. Various memory situations were tried in these experiments. The time for each frame to be calculated at 1080p was noted to be around 4 to 6 milliseconds, while at 4K the time increased, reaching about 10-15 milliseconds. Comparatively, algorithms like SPGrid or GVDB were slower, showing that DCGrid ran quicker and performed better when it came to GPU parallel computing.

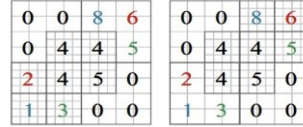


Fig. 4. Block re-arrangement. After each timestep, for each adjacent pair of grids in the hierarchy is considered. Fine blocks with low priority scores are matched with coarse subblocks with high priority scores. Then the fine blocks with low scores moved to the locations of the coarse subblocks with high scores.

DCGrid’s main benefit is that it changes resolution as needed. It can adjust the grid size depending on different outside influences like collisions or similar conditions. Many optimizations for GPU computing have been added into the algorithm. Because of these factors, DCGrid not only gives correct simulations but also uses less memory and reduces the need for high computational power. This makes it stand out as one of the quicker fluid simulation methods around. For gaming, where simulating fluid in real time is important, DCGrid’s capacity to handle large-scale parallel tasks makes it a useful and novel solution for doing fluid simulations in gaming environments.

3.2 Incompressible vs Weakly Compressible SPH

In [2], “Comparison of incompressible and weakly compressible SPH models for free-surface water flows” introduces Smoothed Particle Hydrodynamics (SPH), which is a novel numerical method developed to predict the behavior of liquids such as water. The two variants will be compared: Incompressible SPH (ISPH) versus Weakly Compressible SPH (WCSPH) that are applied, especially when the free surface is involved, such as waves and splashes. Unlike most of the grid-based methods, SPH models are fluid by particles capable of simulating complex and dynamic behaviors; this is very important in creating games when trying to replicate realistic water results.

This comparison between ISPH and WCSPH ascertains which method will be more efficient in the simulation of fluid in real-time environments. It states that ISPH maintains the volume of the fluid and is therefore accurate, whereas WCSPH allows small compressions, losing a bit of accuracy for gaining faster computation. Then, the paper goes ahead and compares both methods through dam-break scenarios and wave impact simulations that serve as benchmarks of performance in fluid simulation.

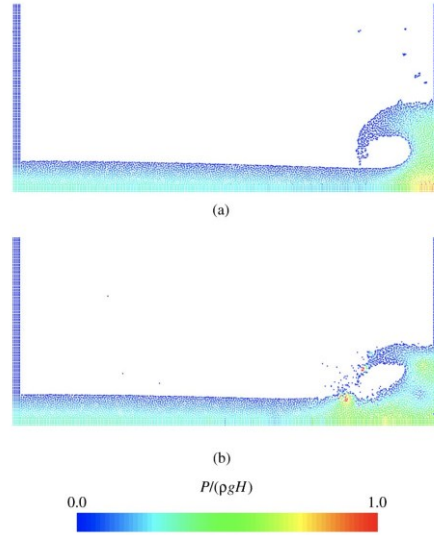


Fig. 5. $W = 2H$, $D = 5.366H$ dam-break, solution at $t^p g/H = 5.95$, for (a) WCSPH and (b) ISPH.

The dam-break scenarios in [2] were performed by considering WCSPH and ISPH methods under different conditions, such as the number of particles and boundary settings. Such a case study is always at the forefront of fluid simulation testing given the real-life situations of events that happen at a sudden collapse of a column of water. This displays an intricate behavior of wave formation and splash.

In the first dam-break test ([2].Fig5), the water column was released where basically similar results were obtained for both ISPH and WCSPH regarding the leading edge of the collapsing water and the way the water tumbled over obstacles. These comparisons indicated that both techniques performed fairly and in good agreement with the experimental data, reflecting the general trend of the fluid column. However, ISPH showed slower advance for the waterfront in some test cases; this is actually the additional computational cost that was paid by ISPH to satisfy incompressibility when the geometrical configuration is complicated. This makes ISPH computationally expensive since it is imperative to solve complex equations to conserve the volume of fluid.

In contrast, WCSPH yielded much smoother free-surface profiles without bearing a high load of computation. Among many great advantages which could be noted in the experiment was the fact that techniques like renormalization of particle densities every 20-time steps further enhanced smoothness and stability on the fluid surface within the simulation in WCSPH. This modification allowed WCSPH to maintain its performance while reducing its computational complexity. This, therefore, makes the process quite a bit more practical for such real-time applications as video games, where the speed of computation far outweighs minor inaccuracies.

The second dam-break test ([2].Fig6) simulated more difficult boundary and obstacle conditions, and the results again agreed with the first. In cases of pressure distribution and fluid flow around obstacles, ISPH continuously provided far more accurate simulations. However, WCSPH outperformed ISPH in speed, finishing the simulation a lot faster and maintaining smooth visual results. This further reinforces the suitability of WCSPH for real-time applications, such as in a game environment, by temporarily compromising the completeness of accuracy in fluids for visually appealing fluid interactions. Generally, it concluded that even

though ISPH had better accuracy and stability, particularly with more detailed interactions- WSPH can still be better applied in applications that require computational efficiency. Actually, this makes WSPH much better for real-time fluid simulations in games, where one often needs to sustain smoothness and visual realism at the price of scientific precision.

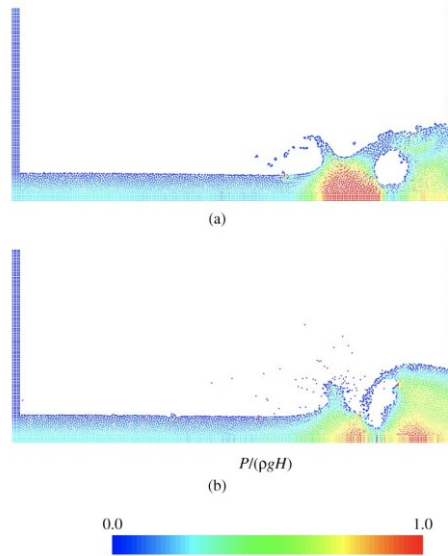


Fig. 6. $W = 2H$, $D = 5.366H$ dam-break, solution at $t^*g/H = 6.81$, for (a) WSPH and (b) ISPH.

The wave impact simulations in [2] involved simulating waves against a vertical wall in order to see how ISPH and WSPH handled pressure variations and free-surface interactions. This kind of test will be useful for game developers who work with coastal environments or dynamic water interactions in their scenes. In the setup, the models simulated waves acting against a solid barrier, consistent with real-world scenarios—that is, waves hitting a dock or rock.

The wave impacts have been generally well approximated by both ISPH and WSPH. However, their performances diverged regarding pressure accuracy and smoothness of the free surface, as shown by ISPH, which presented a more detailed pressure distribution. Smooth and accurate pressure profiles were obtained along the wall by ISPH, as depicted in ([2].Fig7). This accuracy is important for applications in which it relies on highly accurate fluid behavior, such as high-end simulations or game cutscenes that have to have water interact with objects in great detail.

Contrarily, in WSPH, it was possible to capture somewhat smoother free-surface results along the impact in cases with a high number of particles and dynamic movement of the water. On the other hand, WSPH was remarked to show noisier pressure fields than ISPH, which turned out to be less reliable for the proper capturing of fine details in time, such as pressure fluctuations while the wave strikes the wall ([2].Fig8). With these minor inaccuracies, WSPH could still develop a reasonably realistic overall wave impact behavior, thus being ready for real-time applications where the visual effect is often more important than strict accuracy.

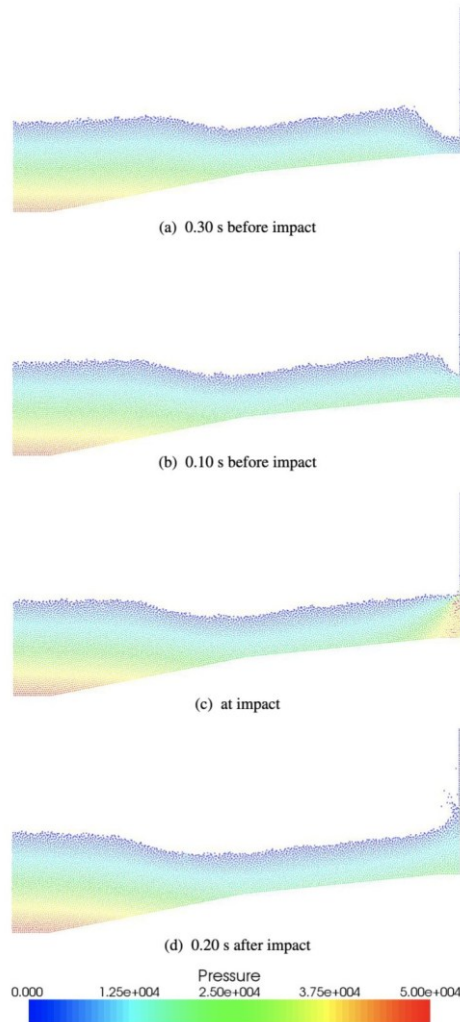


Fig. 7. Solution for 1.3m wave height ('flip-through' type impact), computed using ISPH method

This trade-off between pressure accuracy in ISPH and free surface smoothness in WSPH may indicate that WSPH is more applicable to real-time game environments where the overall look and feel of the fluid play a greater role than minute pressure variations.

One possible conclusion that can be derived from [2] is that it has something to do with the trade-off between precision and speed through a comparative study of ISPH and WSPH. Results by ISPH tend to be more accurate, particularly in pressure calculations and the rendering of realistic fluid movement in dam-break and wave impact tests. However, this comes at the cost of loss of computational efficiency, since ISPH needs to solve complex equations at every time step to maintain incompressibility in the fluids and it turns out to be hugely time-consuming. In cases where a high degree of accuracy is justified, such as engineering simulations or cinematic special effects, ISPH turns out very well.

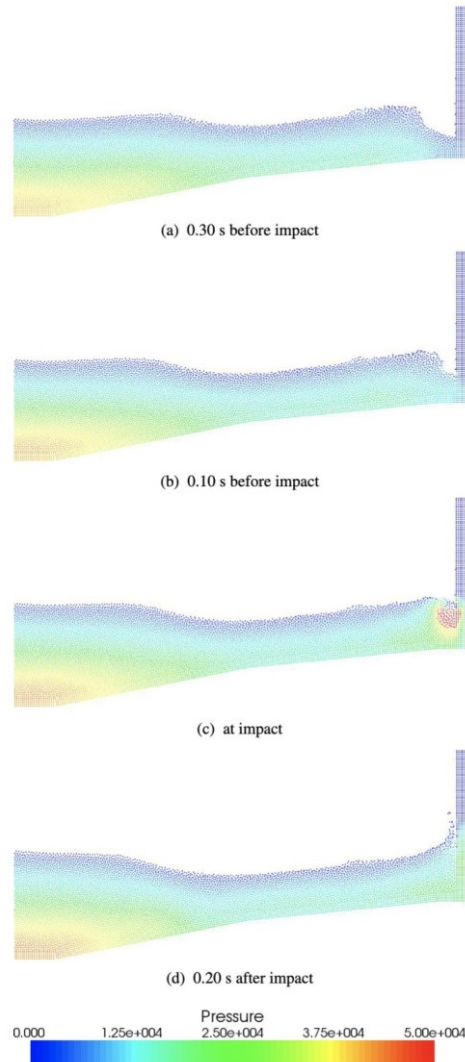


Fig. 8. Solution for 1.3m wave height ('flip-through' type impact), computed using WCSPH method

In contrast, WCSPH is much faster and quite efficient; thus, it serves better in real-time applications, as in the case of video games. Allowing for moderate compressibility, WCSPH reduces the computational load. Although it introduces minor inaccuracies, such as pressure oscillations and less smooth wave behavior, most of these often go unnoticed in dynamic situations. This trade-off becomes very important in gaming, where sustaining performance does not have to mean compromising on visuals.

In this context, [2] demonstrates that ISPH does not outperform WCSPH universally but is instead better suited for different contexts. Applications requiring higher accuracy should make use of ISPH, whereas WCSPH provides a good balance between realism and speed, hence being most appropriate for real-world fluid simulations, especially in the course of developing games.

Focusing on how WCSPH can be used in a real-world environment, [2] lays the very strong foundation required by individuals interested in applying particle-based fluid simulation in games. To developers with a need for dynamic water effects, WCSPH offers a great way to achieve visually plausible simulations without performance lag. The guidelines from [2] help guide decisions on when and how to use fluid simulations so that one can get the right balance between realism and speed.

While ISPH is more accurate, WCSPH's faster computation makes it way better for game development, since most aspects depend on real-time performance. It may be visualized from [2] that WCSPH allows the simulations of fluids with very minor visual sacrifices, hence being highly practical in wave and splash effects creation. Minor inaccuracies, like pressure fluctuations, usually stay imperceptible in fast-paced game environments, underlining WCSPH's suitability for real-time applications even further. In [2], this is Much of the detailed comparisons in [2] effectively illustrate why WCSPH is an ideal solution for fluid simulation in gaming.

3.3 Implicit Incompressible SPH and Its Improvements

IISPH [6] - This abbreviation refers to Implicit Incompressible Smoothed Particle Hydrodynamics, an important technique in the field of fluid simulation within computer-generated imagery. It is an extension of probably one of the most usable methods for the simulation of Lagrangian-based fluids, the SPH - Smoothed Particle Hydrodynamics. Because the Lagrangian viewpoint mainly follows the trajectory of a particle in time, this method is particularly suited for simulation runs of fluid scenarios where the boundaries are well defined, or where one wants to track the motion of individual particles with great precision, such as droplet collisions and liquid particle complex motions. IISPH has several reasons that make it outperform the standard SPH methods in simulating incompressible fluids; for one, the IISPH method uses the semi-implicit Euler method [6] to predict temporal changes of density. That is, the computation considers both current and next-step information; hence, it becomes more stable, allowing for larger time steps without affecting the accuracy of the simulation. Meanwhile, pressure is computed accurately by the IISPH method with the solution of pressure Poisson's equation in order to strictly maintain incompressibility.

In the IISPH approach, the boundary particles are treated as a different entity from the fluid particles. The early IISPH methods need to prepare special algorithms for generating the boundary particles and merge them into computation in a method different from the fluid particles. Obviously, designing separate computational processes and handling logic for these two types of particles enhanced computational complexity. Further, since these are two different computational processes, simulations such as solid liquefaction or liquid solidification make things worse.

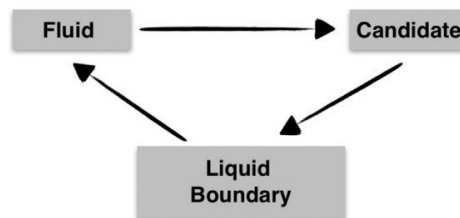


Fig. 9. Roles and role transitions that are considered in our implementation.

However, Cornelis et al. unified the representation of both boundary and fluid particles with the particle system. Since the new unified approach uses one process for all the particles, the solver implementation becomes simpler due to no distinctions between boundary or fluid particles. Furthermore, as both boundary and fluid particles can be represented by unified particles, it enables more natural animation of melting and solidification. This new solver can be used widely in games.

The work of Cornelis et al. proposed a new candidate particle in the support of fluid-to-liquid boundary transition. The main procedure of this technique comprises choosing the nearest fluid particle for each position of liquid boundary sample while updating it to a candidate particle. The latter then turns out to be a liquid boundary particle when proximal to the position.

The method proposed by Cornelis et al. supposed several roles attributed to particles. They categorize the particles into three types ([7].Fig9): fluid particles representing the fluid body, liquid boundary particles representing the boundary, and candidate particles that have to meet the constraints of the fluid density and be used while transitioning from fluid particles to liquid boundary particles. While generating a liquid boundary, for every sampled position on the rigid body, candidate particles are usually chosen as the nearest fluid particle. In practical applications, these three kinds can transform each other to meet the purpose of some effect such as boundary disruption ([2].Fig10).

In the experiments of paper [2], there are two main parameters β and α for improving the simulation effect. β is the animation parameter, which shows the velocity of the candidate particles animating to the specified positions on the liquid boundary. The constraint for that would be such that the maximum velocity of these should not be larger than those of the fluid particles to make the simulation stable. α changes the linear combination of the current particle velocity with the animation velocity for the particle velocity. By tuning α , the motion of the particle can be made more natural. Both these parameters will, in the end, strongly influence the performance and efficiency of the simulation.

Parameter β concerns the velocity increment of the candidate particles. However, setting the fluid velocities too high may violate the CFL condition and thus necessitates taking smaller time steps to maintain stability.

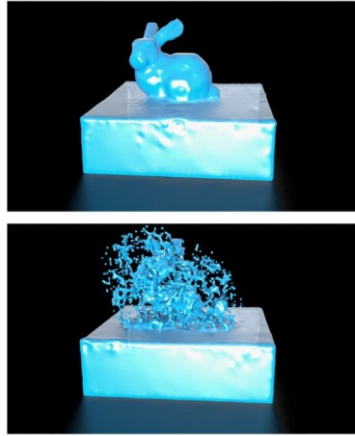


Fig. 10. Liquid boundary sampling. The first image shows the transition of a liquid boundary to fluid particles with an appropriate sampling using the proposed uniform grid. The second image shows the instabilities that occur in the transition of oversampled liquid boundaries to fluid particles.

For parameter α , the greater value represents completely overwriting the particle's velocity seen from the animation view. The smaller α reduces the animation velocity of the candidate particles; therefore, the animation duration increases. However, the larger α is not always better. Since the simulation works with incompressible fluids, the selected velocity field is divergence-free. This means that several iterations are necessary in order to remove the density errors caused by the parameter α . Such is the case of the experiments, where for $\alpha = 0.5$ only 18 iterations were necessary, while for $\alpha = 1.0$, the IISPH solver needed 62 iterations to resolve the density error. Therefore, $\alpha = 0.5$ was adopted in the paper to well balance performance and simulation stability. More precisely, a smaller value for α serves nothing in enhancing the simulation results; instead, it decelerates the assembling liquid boundary significantly.

Although the use of unified particles in IISPH through the work of Cornelis et al. has brought some benefits in performance or even the animation effects optimization when changing boundaries, some limitations still remain. It considers only rigid objects and additional considerations must be taken into account for deformable objects. The unified grid data structure also avoids density errors at liquid boundaries but introduces aliasing effects in surface reconstruction; postprocessing is necessary to reduce visual artifacts such as a lack of smoothness or realism. Finally, they currently assign only one fluid particle for each liquid boundary sample. Nevertheless, they do suggest that in certain conditions a single liquid boundary sample should be associated with multiple fluid particles in order to simulate fluid washing over a rigid object boundary.

3.4 Finite Volume Method

In [4], "A finite volume method parallelization for the simulation of free surface shallow water flows" introduces the Finite Volume Method (FVM), which is a numerical method for numerical solution of partial differential equations. It is widely used in fluid dynamics for simulating free surface shallow water flows. However, the computation of fluids dynamics requires a large memory size and a long computer code execution time, while it has depended on using serial computer environments for a long time [4]. Therefore, this article constructs a parallel algorithm using domain decomposition techniques, which is based on the very popular approximate Riemann solver of Roe, to improve the effectiveness of shallow water flows simulation.

The basic idea of the finite volume method is to divide the computational domain into a number of finite volume units (control bodies) and discretize them in the integral form of the conserved quantities in each control body. Meshing is the starting step of the finite volume method. The area is divided into multiple small control volumes (finite volumes), which can be of regular shape, such as rectangle or square, or irregular geometric shape. Through this process, researchers can transform complex continuous physical problems into discrete numerical problems. Subsequently, for each control volume, the partial differential equations are transformed into integral form by using the conservation laws of fluid dynamics. In this stage, integral operations are performed inside the control body based on conserved quantities such as mass, momentum or energy. Then, the volume integral of the control body is transformed into the area integral of the control body surface by using Gauss theorem. This transformation allows the conservation properties of the whole control body to be formulated in terms of the flow on its surface, so the solution of the problem translates into computing the flow in all directions on the surface of the control body. Conclusively, a linear system of equations is generated from the discrete equations of all control bodies, which are got from pervious steps. Researchers apply appropriate numerical methods to obtain the physical quantities inside the control volume. The data can be used to perform subsequent analysis and visualization.

Finite volume methods have several advantages over other numerical techniques. These methods, which combine the simplicity of finite differences with the geometrical flexibility of finite element methods, have received extensive attention due to their high performance in both subcritical and supercritical flow conditions [4]. As these methods rely on the integral form of conservation laws, it is simple to create numerical schemes that account for discontinuities. The primary challenge is thus to estimate the normal flows through each computational cell interface [4].

On a current serial platform, run-times for finite volume schemes in actual simulation can be very slow for precise results after the refinement of computational grid. Delis et al. adopt a common two-dimensional high-resolution explicit finite volume numerical approach to parallel platforms, utilizing developing programming paradigms. Explicit schemes often need fewer calculations per time step than implicit schemes, but the time saved on a pre-time-step basis may be wasted on a per-simulation period basis since the time step of explicit schemes is limited by the CFL stability requirement [4]. The parallel system uses domain decomposition and utilizes MPI standard protocols for interprocessor communication.

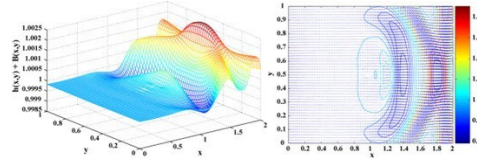


Fig. 11. Benchmark Problem 1: water depth (left) and contour plot with the velocity field (right).

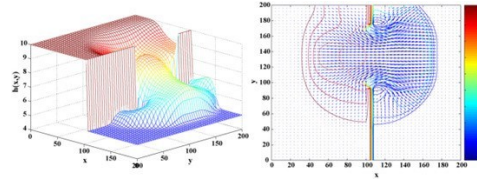


Fig. 12. Benchmark Problem 2: water depth (left) and contour plot with the velocity field (right).

To optimize performance, the workload should be equally distributed, and concurrency is maximized such that all processors are kept busy doing productive work while keeping communication overhead to a minimum. Delis et al. explore an implementation appropriate for distributed memory computers that divides the physical domain into sub-domains assigned to various processors. Two main characteristics of a distributed memory architecture are satisfied. Firstly, the method involves a few powerful processing nodes of the same type, connected by a high-speed network. Secondly, the partitioning of data and computation takes into account the current distributed memory structure while also keeping all processing nodes engaged during the calculation period.

The parallel algorithm is tested on three two-dimensional benchmark problems to evaluate the performance of fluid simulations. Scenario one is a wave propagation over topography, testing the well-balance property of numerical schemes([4].Fig11). Scenario two is a dam-break, showing that data communication has risen at each time step, due to the size of each sub-domain allotted to each processor, as well as the extended simulation duration([4].Fig12). Scenario three is a non-smooth bed topography([4].Fig13).

All the experiment results show that computation time is greatly reduced, and speedup approached linearity in most cases. The parallel strategy is the most portable solution since it works on any parallel architecture (in this case, two) and eliminates the need to partition the data into different files during the startup stage. However, deviations were noted due to cache memory issues and network connection types. One alternative approach is to use a parallel I/O method, which may result in improved performance, while the disadvantage is that in order to generate the appropriate files for the results' visualization, the data must be combined. In conclusion, the TVD scheme can perform well in fluid simulation, providing a good ratio between communication and computation as well. This capability can be utilized for more grid computing system implementations.

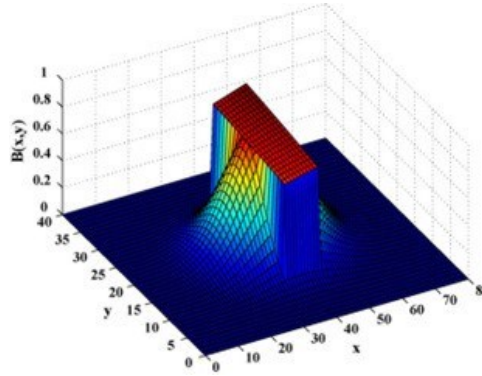


Fig. 13. Benchmark Problem 3: Non-smooth bed topography.

4 Conclusion

During recent years, the development of games bound fluid simulation as the main technology helpful in creating realistic and interactive virtual environments. This paper reviewed several methodologies for fluid simulation, where each of these methodologies has different strengths in different scenarios and thus R solutions to balance various computational performances and visual accuracies.

The DCGrid method, on the other hand, adjusts the grid resolution dynamically by enabling memory management and resource allocation, allowing it to have more computational focus on areas of intricate fluid interaction. This makes it ideal for real-time optimization of fluid performances. ISPH and WCSPH, by contrast, show particle-based methods that exhibit a trade-off between accuracy and speed. ISPH keeps the liquid incompressible, very accurate, and thus suitable for scenes that require precise control, such as cutscenes. In contrast, WCSPH allows small compressibility, thus improving computational efficiency, and hence better suited for fast-paced, real-time games where performance is key.

Furthermore, IISPH enhances ISPH with improved computational stability and larger time steps at no additional cost. Thus, it becomes more applicable to complex fluid interactions in dynamic scenes with high interactivity, especially in real-time games. Finally, the Finite Volume Method presents an efficient way to simulate large bodies of water by segmenting the fluid into control volumes. In particular, the FVM works where there are open-world games or scenes with vast water bodies that balance the level of fidelity and computational efficiency nicely.

This review of methodologies underlines how different techniques for the simulation of fluids can be put into practice depending on the exact needs of a game. Whether more accuracy or more performance is needed, the developer will have to make a choice depending on the needs of the game.

In the future, with increasingly higher demands for more immersive and graphically captivating game environments, fluid simulation will most probably improve. Newly developed hybrid methods that combine the strengths of both grid-based and particle-based techniques can provide superior solutions to the challenge of balancing accuracy and performance. Besides, by leveraging the advances in parallel computing on the GPU, more real-world optimizations can be achieved for real-time performance, allowing future games to create much more sophisticated and interactive fluids.

In the end, fluid simulation will remain at the core of the evolution of gaming, and mastering these techniques will be key to delivering the next generation of interactive and visually stunning gaming experiences.

Acknowledgment

We would like to extend our most sincere appreciation to Professor William Nace for his incessant guidance and support with regard to this literature review. His knowledge and professional feedback provided a great enhancement to our understanding of the field of fluid dynamics within video games. His mentorship and encouragement had a great bearing on both the structure and content of this review. Indeed, we are greatly indebted to his dedication and contribution, which greatly enhanced the quality and depth of our work.

References

- [1] Wouter Raateland et al. “DCGrid: An Adaptive Grid Structure for Memory-Constrained Fluid Simulation on the GPU”. In: *Proc. ACM Comput. Graph. Interact. Tech.* 5.1 (May 2022). DOI: 10.1145/3522608. URL: <https://doi.org/10.1145/3522608>.
- [2] Jason P. Hughes and David I. Graham. “Comparison of incompressible and weakly-compressible SPH models for free-surface water flows”. In: *Journal of Hydraulic Research* 48.sup1 (2010), pp. 105–117. DOI: 10.1080/00221686.2010.9641251. eprint: <https://doi.org/10.1080/00221686.2010.9641251>. URL: <https://doi.org/10.1080/00221686.2010.9641251>.
- [3] Stefan Band et al. “Pressure Boundaries for Implicit Incompressible SPH”. In: *ACM Trans. Graph.* 37.2 (Feb.2018). ISSN: 0730-0301. DOI: 10.1145/3180486. URL: <https://doi.org/10.1145/3180486>.
- [4] Argiris I Delis and Emmanuel N Mathioudakis. “A finite volume method parallelization for the simulation of free surface shallow water flows”. In: *Mathematics and Computers in Simulation* 79. 11 (2009), pp. 3339–3359.
- [5] Jos Stam. “Real-time fluid dynamics for games”. In: *Proceedings of the game developer conference*. Vol. 18. 11. 2003.
- [6] Markus Ihmsen et al. “Implicit Incompressible SPH”. In: *IEEE Transactions on Visualization and Computer Graphics* 20.3 (2014), pp. 426–435. DOI: 10.1109/TVCG.2013.105.
- [7] Jens Cornelis et al. “Liquid boundaries for implicit incompressible SPH”. In: *Computers & Graphics* (Aug. 2015). DOI: 10.1016/j.cag.2015.07.022.

Analysis of Medical Service Utilization Differences Between Floating and Registered Populations Based on Mobile Signaling Data

Qiqi Yan^{1,a}, Yan Yu^{1,b,*}, Yaxin Xu^{2,c}, Liangze Lin^{1,d}, Zhixiang Huang^{1,e}

¹School of Resources and Environmental Engineering, Wuhan University of Technology, 258 Xiongchu Avenue, Wuhan, CHN

²Jiangmen Branch of China Telecom, 2 Huanshi 1st Road, Jiangmen, CHN

a. 334777@whut.edu.cn, b.yyhrose@whut.edu.cn, c. 1728556512@qq.com, d. 302388@whut.edu.cn, e.348683@whut.edu.cn

*corresponding author

Abstract. Accurately assessing the utilization of medical services in floating and registered populations is crucial for the sustainable use of urban healthcare resources and the social equity. At present, the research on medical resource allocation lacks attention to the utilization of medical services by the floating population, especially long-staying population, which affects the fairness of medical resource allocation. This study constructs an approach for identifying the floating population using mobile signaling data and then investigates their duration of stay, spatial distribution, and clustering patterns. On this basis, the criteria for assessing healthcare-seeking behavior are developed to compare the behaviors of cross-province long-staying floating populations with those of registered residents, offering deeper insights into the medical behaviors of the floating population. This study takes Ningbo City as a case, providing valuable insights into its healthcare development. The findings aim to offer a scientific basis for urban medical reforms and the effective allocation of public healthcare services.

Keywords: medical service, floating population, mobile signaling data, Ningbo

1 Introduction

The medical service system plays a crucial role in ensuring the safety and health of individuals, serving as a vital pillar in the country's social development[1]. Most academic studies on the use of medical services mainly focus on regional differences and inequalities, and few studies pay attention to the differences between the floating population and the registered population in the use of medical services[2]. The floating population not only contributes valuable labor and consumption demand to cities but also increases pressure on urban public service infrastructure[3, 4]. In view of this situation, studying the differences in medical service utilization between the floating population and the registered population can reveal the specific needs of different social groups, thereby promoting a more equitable allocation of medical resources and optimizing social governance.

Most studies on the floating population currently rely on questionnaire surveys or directly utilize national population dynamics monitoring survey data to analyze the spatial distribution,

service usage patterns, and economic impacts of the floating population in urban areas[5,6]. However, due to the sampling nature of the questionnaire survey, they cannot comprehensively cover the entire population, thus restricting the effectiveness of related planning and policies. The use of mobile signaling data can address this limitation, enabling a more comprehensive study of the differences in medical service utilization between the floating and registered populations.

Mobile signaling data is a natural collector of population distribution and travel trajectory, which has the characteristics of massive, real, blind spot-free, dynamic real-time and continuity[7, 8]. The data records time and space information such as user's residence and travel, as well as attribute information such as user's gender, age, and cell phone tag, which can be used for profile of users to match the demand. In particular, the address code attribute is the first six digits of the user's ID card, which can be precisely located to the district and county level administrative units. The profile analysis based on the address code can clarify the origin of a large range of floating population and provide insight into the daily behavioral characteristics of the floating population[9].

At present, mobile signaling data is mostly used for the identification research of permanent residents and service population[10,11], but there is little research on its application to the identification of floating population in medical services. Therefore, this study takes the central urban area of Ningbo as the research area and constructs a full procedure of floating population identification based on the December 2019 Unicom mobile signaling data. On this basis, the floating population is further screened out for cross-province long-staying floating population according to their origin and stay time, and the judgment criteria for population health care behavior are proposed to reveal the differences in health care behavior between the floating population and the registered population. The research approach of this study is shown in Fig. 1. In addition, based on the above research, this study proposes medical development suggestions, which provides scientific foundation for urban medical reform, public medical resource allocation, and population management.

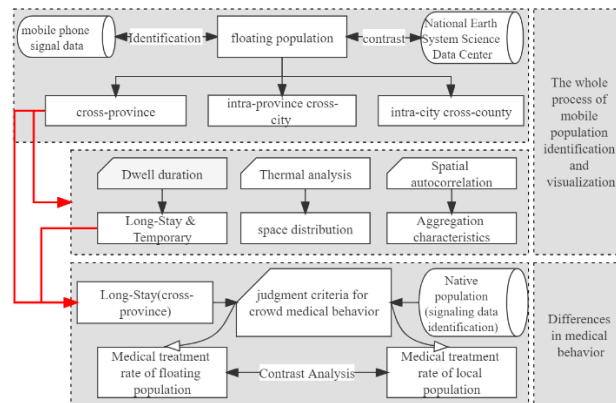


Fig. 1. Research approach of this study.

2 Identification of different types of population

Before conducting differential analysis, it is necessary to first identify different types of populations. This study screened users living in the central urban area of Ningbo using Unicom signaling data and identified their type by analyzing the "first six ID card numbers" attribute in the mobile signaling data. If the "first six ID card numbers" match the administrative division code of Ningbo's central urban area, the user is classified as a registered population. Otherwise, the user is classified as a floating population, thus determining the registered population and floating population in the central urban area of Ningbo who use the mobile phone number of Unicom [12, 13]. On this basis, we further calculated the total floating population and the total registered residence in the central urban area of Ningbo according to the sample expansion method provided by Unicom.

Traditional research on medical service demand primarily focuses on the registered population, but the healthcare needs of the floating population cannot be overlooked. According to the calculation of mobile signaling data, the floating population in the central urban area of Ningbo in December 2019 was 5.012 million. To reveal the special needs and difficulties of the floating population in medical services and improve their medical security level, this study will subdivide these floating populations from the perspectives of source and duration of stay.

2.1 Identification of floating populations from different origins

Analyzing the origin of the floating population is helpful to screen out the groups with different medical insurance policies and facilitate the follow-up analysis of their medical service utilization. Based on the "address code" attributes, the floating population is classified into three categories: Intra-city cross-county, Intra-province cross-city, and Cross-province. The results of the division of the floating population by origin are shown in Table 1. As can be seen from Table 1, the cross-province floating population in Ningbo City is the largest.

Table 1. Statistical table of floating population sources.

Types	Intra-city cross-county	Intra-province cross-city	Cross-province
population(104person)	36.7231	78.7357	385.6997
Proportion (%)	7.328	15.711	76.962

2.2 Identification of floating population with different stay times

Due to the demand for nearby medical treatment, the radiation range of medical resources is mainly concentrated on the long-staying population locally. This study defines individuals who stay in the same location for more than a certain number of days within a month as long-staying floating population. We calculated the number of days that users stayed at the same location for three types: Intra-city cross-county, Intra-province cross-city, and Cross-province, as shown in the figure. The data in the figure shows that the segmentation thresholds for the floating population between counties, cities, and provinces are 17 days, 18 days, and 18 days, respectively. Ultimately, this study identified temporary and long-staying floating population under different types.

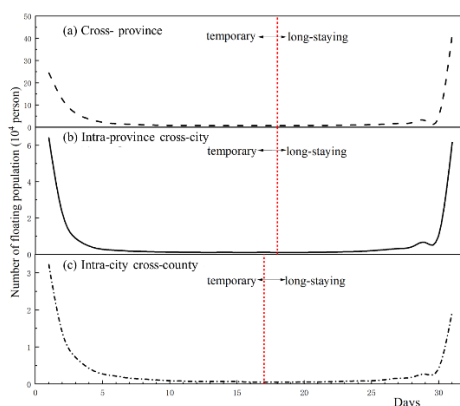


Fig. 2. Accumulated residence days of floating population.

3 Analysis of floating population

Studying the spatial distribution characteristics of the floating population is essential for achieving the rational allocation of medical services[14]. This study analyzes the spatial distribution and clustering characteristics of the floating population from the perspectives of different time periods and places of origin, aiming to provide insights for optimizing the allocation of healthcare services and improving service accessibility.

3.1 Spatial distribution characteristics of floating population

This study examines the distribution of two types of floating populations—long-staying and temporary—at 2h intervals, analyzing their distribution patterns and potential evolution over time. Given the large number of time periods and space limitations, this paper focuses on the distribution of long-term and temporary floating populations during three key time slots representing the most stable working and living hours: 01:00-03:00, 9:00-11:00, and 15:00-17:00. These time periods are shown in Figure 3.

It can be seen from Figure 3 that the distribution of the floating population in Ningbo has an obvious multi-center structure, showing a three-center structure of the center of the ring road - northern Beilun - eastern Fenghua in different time periods, and the floating population is mainly distributed within the ring road, with small-scale population gatherings in Beilun and Fenghua districts. The high density floating population ($>50,000$) is distributed in only a few areas, and is concentrated in the center of the ring road. The medium density (3000~50000) distribution area is highly identifiable, with the central location of the ring line as the core, symmetrical distribution and piecewise distribution, while there is also a small amount of distribution in the population gathering centers of Beilun and Fenghua districts. The 9:00-11:00 and 15:00-17:00 time periods show a more obvious central circle structure compared to the 1:00-3:00 time period, when the core area of the city has a higher recognition. According to the above phenomenon, on the one hand, it shows that the Ring Road area, being part of the nucleus in the "one core, two wings, two belts and three bays" of Ningbo City, plays the main function of the city due to its transportation location and economic advantages, and attracts a large number of floating population to live and work here. On the other hand, it shows that Beilun District, as a part of the industrial zone of the eastern coastal town, is connected to Shanghai to the north and to

Sanmen Bay to the south, and its location and port trade create more job opportunities. The attractiveness of Fenghua District to the floating population may be due to its location advantage as a member of the "South Wing", its excellent tourism and ecological resources, and its large development potential.

3.2 Spatial aggregation characteristics of the floating population

The agglomeration characteristics of floating population is the basis for developing a fair medical service system and optimizing the spatial pattern of urban health service[15]. Therefore, in order to further explore the agglomeration characteristics of long-staying and temporary floating population, this study further conducts local spatial autocorrelation analysis on the spatial patterns of the two types of floating population. The results are shown in Figure 4.

From Figure 4, compared with temporary floating population, long-staying floating population has obvious characteristics of Low-Low agglomeration, mainly distributed near the central urban ring road. It may be due to the High-High agglomeration areas that encourage floating population to gather inside the ring road, forming high-value agglomeration areas with dense floating population, while low value sparse areas diffuse the attractiveness of high-value agglomeration areas, becoming a low value circular floating population depression. Temporary floating population aggregation is mainly dominated by High-High clustering area (HH) and Low-Low clustering area (LL), which are mainly distributed within the central city ring line, and the characteristics of population aggregation are roughly the same at different time periods. This reflects that the temporary floating population tends to be more active within the central city ring road with limited time.

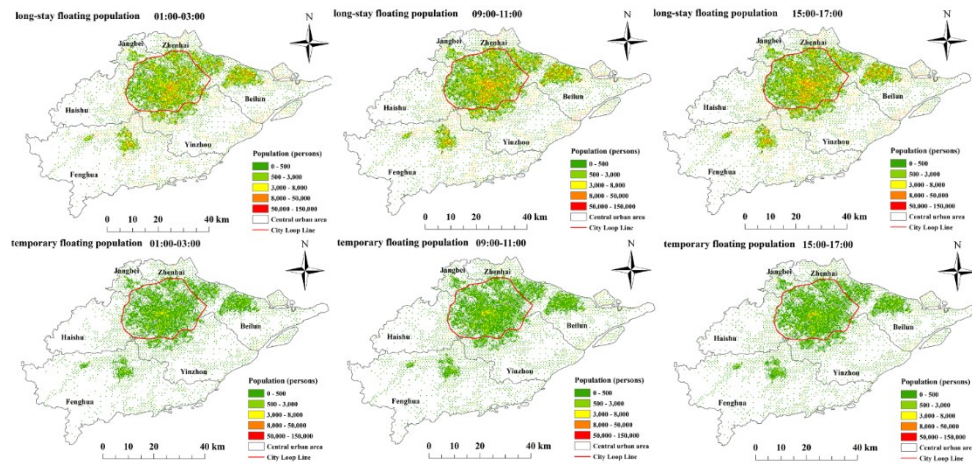


Fig. 3. Distribution of long-staying and temporary floating population in all period

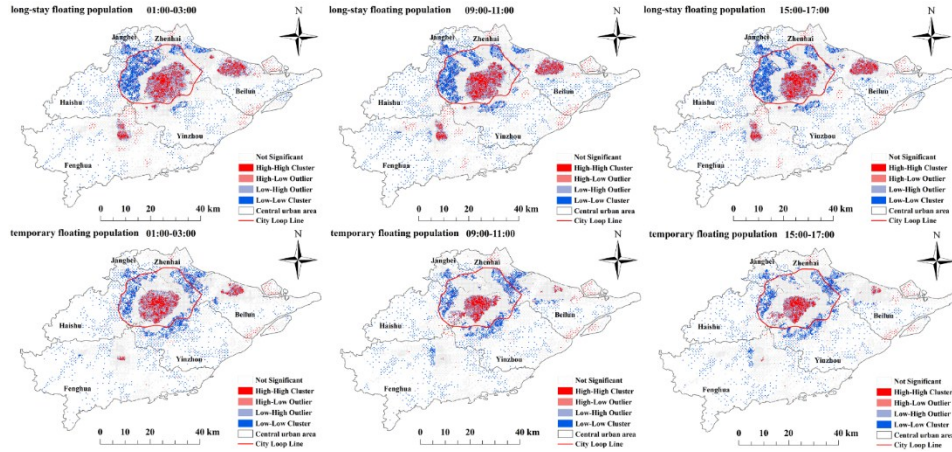


Fig. 4. LISA cluster of long-staying and temporary floating population in all period.

4 Analysis on the difference of medical service utilization among "Floating- Registered" population

4.1 Criteria for determining the utilization of medical services for specific populations

Both the long-staying floating population and the registered population are key consumers of medical service resources. Exploring the basic characteristics and differences in their utilization of medical services can contribute to improving the equity of healthcare services. Seeking medical treatment is a typical travel behavior, and it is mostly focused on factors influencing medical treatment and choices of medical treatment behavior [16-18]. This study takes medical behavior as an example to explore the differences between the floating population and the registered population. Since the medical insurance policies of Ningbo City are different for the population inside and outside the province, and the cross-province floating population occupies an absolute advantage in the total floating population (Section 2.1), this study focuses on the cross-province long-staying floating population and compares it with the registered population.

The criteria for judging the medical behavior of the population are as follows: ① Determine the study population and calculate the total daily visit time of users in this population to the hospital; ② Exclude medical staff, i.e. screen out patients who visit the hospital daily (stay for 0.5~5 hours) and inpatients (stay>13 hours); ③ Count the number of daily visits to hospitals and expand the sample to obtain the medical situation of this population.

Based on the above criteria, the number of long-staying floating population and registered population seeking medical care was calculated. To compare the differences in medical treatment behaviors between the two groups, this study evaluates the medical treatment behavior of the cross-province long-staying floating population based on the medical treatment frequency of the registered population, as shown in the following formula:

$$R = \frac{P_{FH}/P_F}{P_{RH}/P_R} \times 100\% \quad (1)$$

In the formula, R represents the ratio of the cross-province long-staying floating population to registered population for medical treatment, P_F and P_R represent the number of the cross-

province long-staying floating population and registered population respectively, P_{F_H} and P_{R_H} represent the number of medical visits of long-staying floating population and registered population respectively.

4.2 Difference in medical service utilization among "Floating and Registered" population

Tertiary hospitals typically cover a large service area and treat a high volume of patients. Considering factors such as hospital influence and transportation accessibility, this study selected Ningbo First Hospital, Second Hospital, and the Maternal and Child Health Hospital as a case to identify the number of long-staying floating and registered population who visit their hospitals daily. Using the formula mentioned above, the proportion of long-staying floating population visiting these three hospitals relative to the registered population was calculated, as shown in Figure 5.

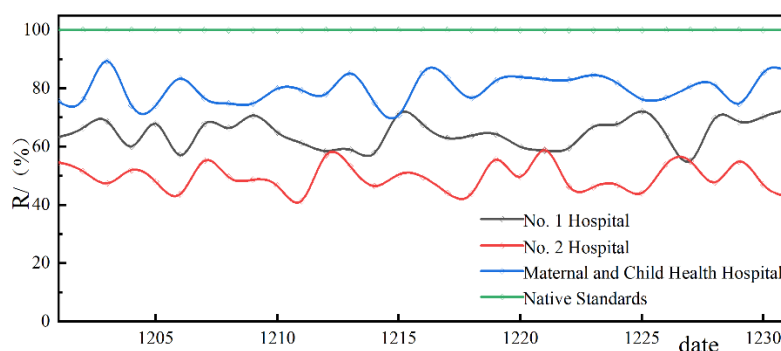


Fig. 5. The proportion of long-staying floating population relative to registered population.

As can be seen from Figure 7, although the cross-province long-staying floating population has differences in the proportion of medical treatment relative to the domiciled population due to different hospitals, each basically remains at a stable level, and none of them reaches the standard of medical treatment for the domiciled population. The above phenomenon indicates that the floating population is disadvantaged in the utilization of public health service resources compared to the domiciled population, and the results are similar to those of scholars such as Yujie Gan and Yumeng Tang [16, 19]. This phenomenon is also observed in cities like Shanghai, primarily due to the complex procedures involved in transferring medical insurance and referrals for cross-provincial migrants. This results in lower levels of medical security compared to registered population, leading to healthcare concerns and differences in medical behaviors.

4.3 Ningbo Medical Development Suggestions

In fact, for people from non-Yangtze River Delta regions who travel and work in Ningbo, outpatient expenses account for the bulk of medical expenses. Fortunately, the current policy of improving the settlement of outpatient expenses across provinces and other places is being rolled out across the country in full swing. Many experts and scholars have also provided valuable suggestions for further improving the construction of China's medical security system. For example, Zheng Xianping and others have summarized the characteristics and existing problems of the current inter-provincial outpatient billing in different places, and proposed

optimization countermeasures [20]. In the future, Ningbo City can actively promote "outside-province co-management" and medical insurance interoperability to meet the demands of inter-provincial floating population for medical treatment, dispel their concerns about medical treatment, provide convenient and fast settlement services for people cross-provincial seeking medical treatment in different places.

5 Conclusion

This study aims to compare healthcare utilization between the floating and registered populations, highlighting disparities that may impact the equitable distribution of urban medical resources. Using Ningbo city district as the study area, mobile signaling data was applied to accurately identify a wide range of floating populations and examine their length of stay, spatial distribution, and aggregation characteristics. Criteria were subsequently proposed to assess the healthcare behavior of these populations, and healthcare utilization between the long-staying cross provincial floating and registered populations was quantitatively compared, revealing that the healthcare utilization rate of the floating population in Ningbo is significantly lower than that of the registered population. Recognizing the differences in medical care behavior between the urban floating population and the registered resident population, future research will focus on strategies and policies to bridge these disparities and address the healthcare needs of both groups, which is crucial for advancing urban population management, medical reform, public healthcare resource allocation, and infrastructure development. Despite this, our study still has some limitations. The mobile signaling data comes from only one company, which may not fully capture changes in healthcare utilization. However, the data expansion and ratio analysis minimize this impact. Future studies will address this by using longer-term, more granular data for a more accurate and comprehensive understanding of healthcare utilization patterns.

Acknowledgements

This study was supported by the National Natural Science Foundation of China (grant numbers: 42471445, 42171260).

References

- [1] Yu Yan, Yan Qiqi, Yan Cheng, et al. Evaluation of Medical Resource Equity in Ningbo Based on Mobile Signaling Data. *Geospatial Information*. 2024, 22(07): 1-4+11.
- [2] Yang Xin. Differences in Utilization of Basic Public Health Services Between Registered and Floating Populations and Their Influencing Factors. *Chinese Journal of Public Health*. 2018, 34(6): 781-785..
- [3] Wang De, Gu Jin. The Use of Public Facilities by Floating Population in Shanghai—Case Study of Hongjin Community. *URBAN PLANNING FORUM*. 2010, 04: 76-82.
- [4] Yu, Y., Meng, W., Fan, J., Ma, W., Xia, Y., Development of Public Health Emergency Response Strategies Based on Economic Space Field Theory and ESDA. *Geomatics and Information Science of Wuhan University*. 2021, 46, 159-166+220.
- [5] Chen Jie, Wang Wei. Economic incentives and settlement intentions of rural migrants: Evidence from China. *URBAN STUDIES*. 2019, 41(3): 372-389.

- [6] Wen, P., Zhou, S.H. Spatial-Temporal Characteristics and Planning Implications of Daily Activities of Migrant Population in Guangzhou. 2018 China Urban Planning Annual Conference, Hangzhou, Zhejiang, China.
- [7] Zhong Shuqi, Deng Rufeng, Deng Hongping, Cai Ming. Recognition of traffic mode of mobile phone data based on the combination of point of interest data navigation data. *ACTA SCIENTIARUM NATURALIUM UNIVERSITATIS SUMYATSENI*. 2020, 59(03): 87-96.
- [8] Lauren Alexander, Shang Jiang, Murga Mikel, Marta C. Gonzalez. Origin–destination trips by purpose and time of day inferred from mobile phone data. *SCIENCE*. 2015, 58: 240-250.
- [9] Aguilera Vincent, Sylvain Allio, BeNezech Vincent, Combes Francois, Milion Chloe. Using cell phone data to measure quality of service and passenger flows of Paris transit system. *TRANSPORTATION RESEARCH PART C: EMERGING TECHNOLOGIES*. 2014, 43: 198-211.
- [10] Li Xinyue, Chen Fulin. Regional Connections and Demographic Characteristics of Small-Medium Cities Based on Cellular Signaling Data. *Urban Transport*. 2020, 18(04): 47-54+70.
- [11] Hai Xiaodong, Liu Yunshu, Zhao Pengjun, Zhang Hui. Using Mobile Phone Data to Estimate the Temporal-Spatial Distribution and Socioeconomic Attributes of Population in Megacities: A Case Study of Beijing. *Acta Scientiarum Naturalium Universitatis Pekinensis*. 2020, 56(03): 518-530.
- [12] Wang De, Ren Xiyuan. Distribution and Composition of Actual Population in Urban Space from Daily Human Mobility View. *URBAN PLANNING FORUM*. 2019, 02: 36-43.
- [13] Shi Cheng, Chen Chen, Niu Xinyi. Planning Megacities for the Actual Service Population: A Case of Hangzhou. *URBAN PLANNING FORUM*. 2018, 04: 41-48.
- [14] Ma Zhifei, Yin Shanggang, Zhang Yu, Li Zaijun, Wu Qiyang. Spatial distribution, flowing rules, and forming mechanism of inter-cities floating population in China. *Geographical Research*. 2019, 38(04): 926-936.
- [15] Sheng Yinan, Yang Xuyu. Spatial Patterns and Mechanisms of the Floating Population Agglomeration among Top Three City Clusters in China. *Population & Economics*. 2021, 06: 88-107.
- [16] Gan Yujie, Zhang Longlong. Medical Insurance Coverage and Its Impact on the Medical Choice Behavior of Migrant Population in China. *POPULATION AND DEVELOPMENT*. 2021, 27(04): 24-36.
- [17] Zhang Jian, Cai Jinlong, Huang Yuanying, He Zhongchen, Tang Guizhong. China's Floating Population's Healthcare Utilization Choices and Influencing Factors. *CHINESE GENERAL PRACTICE*. 2021, 24(16): 2008-2014.
- [18] Zheng Yanhui, Hao Xiaoning. Research on Medical Orientation and Influence Factors of Elderly Floating Population. *Chinese Health Economics*. 2021, 40(08): 56-59.
- [19] Tang Yumeng, Li Qian, He Tianjing, Zhang Qingjun. Research Progress and Revelation of Medical Behavior of Floating Population in China. *CHINESE JOURNAL OF SOCIAL MEDICINE*. 2016, 33(05): 435-438.
- [20] Zheng Xianping, Wu Chaonan, Tong Xiao, Liu Ya. Thoughts on the Improvement of Remote Settlement Policy on Medical Insurance Outpatient Fee from the Perspective of Globalization. *Chinese Health Economics*. 2021, 40(10): 35-38.

Causal Relationship Analysis Between Oil Price Index and Precious Metals Price Index

Fuchun Zhan^{1,a,*}, Xiangmin Zhang^{1,b}

¹Economics and Management College, China University of Geosciences Beijing, Beijing, China

a. fuchunsureshen@gmail.com, b. zhangxiangmin2001@163.com

*corresponding author

Abstract. This paper examines the relationship between oil prices and precious metals (gold, silver, palladium, and platinum) prices during periods of economic turbulence and geopolitical events. Using discrete wavelet transform technology, the time series data is decomposed to extract new forms of short-term, medium-term, and long-term time series. A vector autoregression (VAR) model is then established to perform Granger causality tests and impulse response analyses between oil prices and precious metals prices. The results indicate that oil, gold, and silver have strong market influence, while platinum and palladium have relatively weaker influence. In the short term, oil has a unidirectional Granger causality effect on gold and silver. In the medium term, oil and gold, as well as oil and silver, exhibit bidirectional Granger causality. In the long term, oil and platinum demonstrate bidirectional Granger causality. Additionally, in the short term, the impulse response between gold, silver, and oil is significant, revealing notable short-term dynamic relationships among these three variables.

Keywords: Granger causality, oil price, precious metals price

1 Introduction

Oil and precious metals are two critical commodity categories in the global economy, and their price fluctuations have profound impacts on the global economy and financial markets. The oil price index reflects the supply and demand conditions in the global oil market, while the precious metals price index typically includes prices of metals such as gold and silver, which are considered safe-haven assets[1]. Studying the causal relationship between the oil price index and the precious metals price index not only helps to understand the market linkages between the two but also provides valuable references for investors and policymakers.

Over the past few decades, the oil and precious metals markets have experienced multiple significant fluctuations, often closely tied to global economic events, geopolitical risks, and changes in monetary policy[2-4]. Notably, the outbreak of the COVID-19 pandemic at the end of 2019 severely impacted the global economy, forcing a halt to production and daily life worldwide. Subsequently, the Russia-Ukraine war erupted in 2021, causing a sharp increase in oil prices as a critical energy resource. At the same time, gold, renowned as a safe-haven asset, maintained consistently high price levels. Therefore, this paper selects data from January 2020 to December 2023 to analyze the causal relationship between the oil price index and the precious metals price index, exploring the dynamic interaction mechanisms between the two and providing market participants with more accurate predictions and decision-making support.

To comprehensively examine the causal relationship between the oil price index and the precious metals price index time series, this study first applies the discrete wavelet transform method to decompose the time series data into three different forms: short-term, medium-term, and long-term. Then, a VAR model is constructed, followed by Granger causality tests and impulse response analyses. The study investigates the interactions between oil prices and precious metals prices and explores their underlying econometric implications. This research aims to provide a new perspective and methodology for related studies and offer valuable references for practical market operations.

2 Literature Review

Oil and precious metals (such as gold and silver) play critical roles in the global economy. As one of the primary energy sources, oil is vital for global industrial production, transportation, power generation, and household energy supply. The price and supply stability of oil directly impact global economic health and growth[5]. Precious metals are not only widely used in the jewelry and decoration industries but also play essential roles in electronics, healthcare, and other industrial applications. Gold, in particular, is often regarded as the "ultimate safe haven" for currency. During periods of financial market turbulence, gold prices tend to rise as investors view it as a means of value preservation and storage[6]. Fluctuations in the prices of oil and precious metals can influence monetary policy, exchange rates, and global capital flows. The economic instability and shocks caused by the pandemic have led to dramatic fluctuations in oil production and prices.

Granger causality, introduced by economist Clive Granger in 1969[7], is a statistical hypothesis testing method used in time series analysis to determine whether one time series can predict another. In previous studies, many scholars have applied causality analysis to the field of commodity prices, such as oil, and to analyze the macroeconomic factors that influence them.

Many scholars have studied the relationship between oil and precious metals. Li Ting et al.[8] found a significant positive correlation between oil and gold prices, although during certain periods, such as financial crises, the two may exhibit a negative correlation. Y. S. Wang et al.[9] discovered mutual short-term influences between crude oil and gold prices. Liu Jie[10], after analyzing historical data and the factors influencing the relationship between oil and gold prices, observed a synchronous trend between the two prices in the same period, although some factors may cause short-term deviations. Liu Xiangyun et al.[11] reached similar conclusions. Guo Shijie[12] conducted Granger causality tests on oil rents, coal rents, and natural gas rents. Guo Mingyuan[13] used linear Granger causality tests to explore the impact of crude oil prices on China's economy from an industry perspective. C. Gharib et al.[14] studied the causal relationship between crude oil and gold spot prices to assess the economic impact of COVID-19. They identified common mild explosive periods in the WTI and gold markets, as well as bidirectional contagion effects between oil and gold market bubbles during the recent COVID-19 outbreak. Gao Xinwei et al.[15], using classical cointegration theory and VAR models, employed Johansen cointegration tests, ECM, Granger causality tests, and impulse response functions to study the quantitative relationships between international crude oil prices, the real US dollar exchange rate, and global oil rig counts, both pairwise and collectively. Their findings showed that international crude oil prices had a long-term positive impact on oil rig counts, while the real US dollar exchange rate had a long-term negative impact on oil rig counts. T. Liu et al.[16] proposed a new method for calculating time-varying volatility spillover indices using the generalized forecast error variance decomposition of a TVP-VAR-SV model. Chen

Guangying[17], using Johansen cointegration tests, Granger causality tests, impulse response, and variance decomposition, studied the relationship between international oil prices and inflation in China. Ma Duo[18] established a VEC model to investigate the relationships among international gold prices, US Federal Reserve monetary policy, the US dollar index, and oil prices. Gao Hui et al.[19] selected domestic crude oil futures market micro-indicators and renminbi internationalization indicators, using Granger causality tests, cointegration tests, and error correction models to quantitatively study the comprehensive impact of crude oil futures on renminbi internationalization. A. Bossman et al.[1] examined the asymmetric relationship between EU industry stocks and oil during periods of geopolitical turmoil, focusing on oil implied volatility, geopolitical risks, and market sentiment. Z. Dai et al.[20] analyzed the volatility spillover effects and dynamic relationships among WTI crude oil, gold, and China's new energy vehicle, environmental protection, renewable energy, coal, consumer fuel, and high-tech stock markets.

By reviewing relevant studies, it can be observed that most scholars have conducted causality analyses on oil prices and gold prices, while relatively few studies have focused on other precious metals, such as silver and platinum. Therefore, this paper incorporates four types of precious metals into the research scope to conduct an in-depth analysis of the relationship between oil prices and precious metals prices.

3 Research Methods and Data

3.1 Econometric Methods

Stationarity Test. The stationarity test is a method used to examine whether a time series is stationary, specifically testing whether the time series has a unit root (non-stationarity). In this study, two common methods for stationarity testing are employed: the Augmented Dickey-Fuller (ADF) test and the Phillips-Perron (PP) test.

The Augmented Dickey-Fuller (ADF) test, proposed by Dickey and Fuller in 1979, is used to test whether a time series contains a unit root[21]. A unit root implies that the data exhibit a trend of drift over time, indicating non-stationarity. The basic model for the ADF test is as follows:

$$\Delta Y_t = \alpha + \beta Y_{t-1} + \gamma \Delta Y_{t-1} + \delta_1 \Delta Y_{t-1} + \cdots + \delta_{p-1} \Delta Y_{t-p+1} + \varepsilon_t \quad (1)$$

Where Δ represents the first difference, Y_t is the time series data, α is the intercept, β is the coefficient for the unit root test, γ represents whether there is a time-varying trend in the data. The ADF test evaluates whether the data have a unit root by examining the estimated coefficients. If the coefficient β is sufficiently close to zero, the null hypothesis of a unit root can be rejected, indicating that the data are stationary. The Phillips-Perron (PP) test is another statistical method used to detect unit roots in time series data. It is an improved version of the ADF test, primarily designed to address issues of autocorrelation and heteroskedasticity. The core idea of the PP test is to adjust the error term in the ADF test using non-parametric methods.

Vector Autoregression Model. The Vector Autoregression (VAR) Model, commonly abbreviated as the VAR model, is a widely used econometric model introduced by economist Christopher A. Sims in 1980[22]. The VAR model regresses all current variables in the model on the lagged values of all variables. By adjusting the lag order of the variables, the model

regresses each variable on itself and expands the univariate autoregressive model into an autoregressive model composed of multivariate time series variables.

The typical formula for the VAR model is as follows:

$$x_t = c + \Phi_1 x_{t-1} + \cdots + \Phi_p x_{t-p} + \varepsilon_t \quad (3)$$

Where $p \geq 1$, c is a $k \times 1$ constant vector, Φ_i is a $K \times K$ constant matrix, $i=1, \dots, p$. ε_t is a vector of white noise.

Before constructing the VAR model, it is essential to determine the order of the model, typically denoted as p . The order of the VAR model determines the number of lag periods included in the model, that is, how many historical periods of data are considered when predicting the current value. Suitable lag orders can be selected using information criteria such as FPE (Final Prediction Error), AIC (Akaike Information Criterion), or BIC (Bayesian Information Criterion), or through empirical methods[23].

The Final Prediction Error (FPE) criterion is a type of information criterion used in selecting time series models. FPE evaluates the performance of time series models with different lag orders to help identify the optimal model order. During the model selection process, FPE values for various lag orders are compared, and the model with the smallest FPE value is chosen as the best model. The specific formula for FPE is as follows:

$$FPE = \frac{n+p+1}{n-p-1} * \hat{\delta}^2 \quad (2)$$

Where n represents the number of observations in the time series, p denotes the lag order of the model, $\hat{\delta}^2$ is the estimated mean squared error (MSE) of the model, typically calculated using the residual variance from the model estimates. FPE aims to strike a balance between the model's goodness-of-fit and its complexity. Specifically, a smaller FPE value indicates better performance of the model in fitting the data. However, increasing the model order may lead to overfitting, where the model becomes overly complex and fits the noise in the data rather than its underlying patterns. Thus, FPE provides a method to balance fit quality and model complexity, aiding in the selection of an appropriate model order. In time series analysis, different model orders are usually tested, and the FPE value is calculated for each. The model with the smallest FPE value is selected as the optimal model. This approach ensures that the chosen model fits the data well without being excessively complex, thereby avoiding overfitting.

Granger Causality Test. The Granger Causality Test is a statistical method used to examine the causal relationship between time series. It is based on the principle of Granger causality, which states that if one time series can predict changes in another time series, the former is considered to have a causal influence on the latter. In practice, the Granger Causality Test is often used to determine whether one time series can effectively explain the variations in another time series. Granger causality indicates that the changes in one set of time series are caused by the changes in another set of time series.

The model for the Granger Causality Test is as follows:

$$\begin{aligned} x_t &= \alpha_i + \sum_{m=1}^p \beta_m x_{t-m} + \sum_{m=1}^p \beta_m y_{t-m} + \varepsilon_t \\ y_t &= \gamma_i + \sum_{m=1}^p \delta_m y_{t-m} + \sum_{m=1}^p \beta_m x_{t-m} + \varepsilon_t \end{aligned} \quad (4)$$

$$H_0 = \beta_1 = \beta_2 = \dots = \beta_p = 0$$

$$H_1 = \gamma_1 = \gamma_2 = \dots = \gamma_i = 0$$

Where p is the lag order, $\alpha, \beta, \gamma, \delta$ are regression coefficients. The null hypothesis tests whether the past values of X provide no predictive information for the future values of Y . If only one of the two hypotheses is rejected, it indicates a unidirectional causal relationship between X and Y . If both hypotheses are rejected, it implies a bidirectional causal relationship between X and Y . If neither hypothesis is rejected, it suggests that there is no causal relationship between X and Y . Thus, Granger causality represents a dynamic correlation, indicating whether one variable has the predictive ability to explain changes in another variable.

For unidirectional causality, If the test rejects H_0 but does not reject H_1 , it indicates that the explanatory variable Y influences the dependent variable X , meaning that Y has a causal impact on X . Conversely, if H_1 is rejected but H_0 is not, it implies that the explanatory variable X influences the dependent variable Y , meaning X has a causal impact on Y .

For bidirectional causality, If the test simultaneously rejects both H_0 and H_1 , it indicates that a bidirectional causal relationship exists between the two variables. This means that changes in the X variable lead to changes in the Y variable, and changes in the Y variable, in turn, influence the X variable.

For no causal relationship, If the test fails to reject both H_0 and H_1 , it suggests that there is no causal relationship between X and Y , indicating that the two variables are independent of each other.

Impulse Response. Impulse Response is an important tool for analyzing the dynamics of a VAR model. It describes the dynamic response of each variable in the system when the system is subjected to a unit shock (or impulse).

$$Y_t = c + \sum_{i=1}^p A_i Y_{t-i} + \varepsilon_t \quad (5)$$

Based on the VAR model described above, suppose a unit shock is applied to the j -th variable at time t , meaning that the corresponding position in the j -th column is set to 1 while all other positions are set to 0. At time t , the responses of all variables can be expressed as:

$$IRF_{j,t} = \sum_{i=0}^t A_j^{icj} \quad (6)$$

Where $IRF_{j,t}$ represents the impulse response value of the j -th variable at time t , A_j^i is the i -th power of the j -th column of matrix A_j ($A_j^0 = I_k$, the identity matrix), cj is the j -th element of c , representing the initial shock applied to the j -th variable.

Discrete Wavelet Transform. The Discrete Wavelet Transform (DWT) is a signal processing technique used to decompose signals into frequency components at different scales. The principle of DWT is based on multi-resolution analysis, which decomposes a signal into approximation coefficients and detail coefficients to analyze its characteristics across varying levels of detail. DWT leverages the multi-resolution property of signals by breaking them down into frequency components at different scales, progressively revealing the signal's features from coarse to fine detail. It employs low-pass and high-pass filters to filter the signal, extracting approximate and detailed components. A downsampling operation is then applied to the filtered signal to reduce data size while preserving key information. DWT decomposes a signal into

approximation coefficients and detail coefficients at various levels, and the original signal can be reconstructed using the inverse DWT.

$$c_{j,k} = \sum_n x[n] \psi_{j,k}[n] \quad (7)$$

Where $c_{j,k}$ are the wavelet coefficients, $x[n]$ is the original signal, $\psi_{j,k}[n]$ is the discrete wavelet function, j is the scale parameter, k is the translation parameter.

3.2 Research Data and Preliminary Statistical Results

Variable Selection and Descriptive Statistics. When selecting variables, data availability and general applicability were considered. For oil prices, the WTI Crude Oil Price Index (West Texas Intermediate) was chosen. For precious metals prices, daily closing price data from the LBMA (London Bullion Market Association) were used.

Table 1. Variable Selection and Descriptions

Variables type	Name of variables	Symbol used
Oil Price Index	WTI Crude Oil Price Index	WTI
	LBMA GOLD Price Index	GOLD
Precious Metals Price Index	LBMA SILVER Price Index	SILV
	LBMA PLATINUM Price Index	PLATNUM
	LBMA PALLADIUM Price Index	PALLDINUM

The data range spans from January 1, 2020, to December 29, 2023, providing 996 observations after aligning the datasets based on the variable with the least number of statistical days. The return rates for each index were calculated as $R_t = \ln\left(\frac{P_t}{P_{t-1}}\right) \times 100$.

Due to the COVID-19 pandemic outbreak in 2019, the global economy faced significant disruptions, with production and daily life coming to a standstill. Between June 2019 and June 2020, the five variables experienced considerable volatility. In 2021, the outbreak of the Russia-Ukraine war caused oil prices, as a critical energy resource, to surge rapidly. Meanwhile, gold, known as a safe haven, maintained consistently high price levels.

Descriptive statistical analysis was conducted on the logarithmic returns of each variable.

Table 2. Summary Statistics

Variables	Mean	Std. Dev	Min	Max	Skewness	Kurtosis
WTI	-0.0896193	4.145673	-42.58324	28.13821	-1.124619	31.9311
GOLD	-0.0315905	1.075057	-5.775362	5.113959	.245111	6.734884
SILVER	-0.0292777	2.219736	-8.917675	12.31422	.4257005	7.425141
PLATINUM	-0.0011889	2.162738	-9.931355	13.61363	.2689621	6.385696
PALLADIUM	0.0597958	2.962721	-18.62702	22.91715	.3443576	10.48247

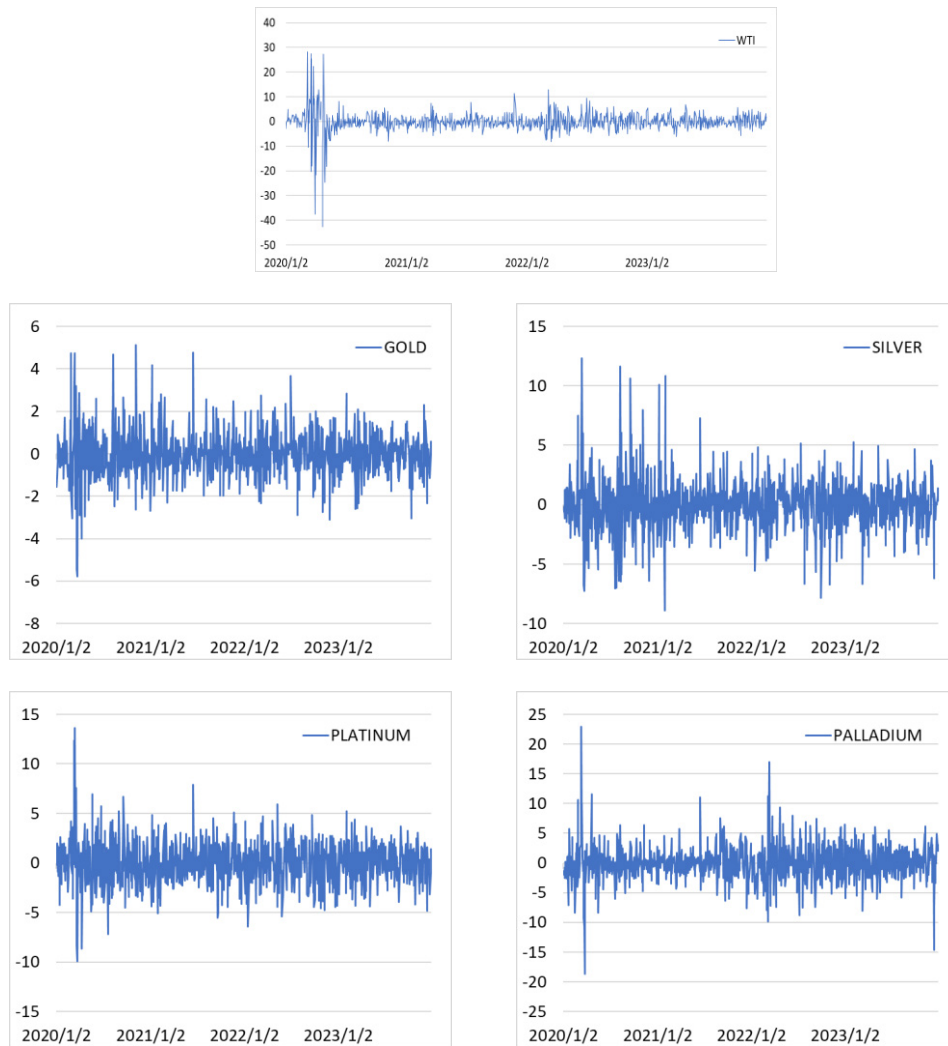


Fig. 1. Statistical Chart of Log-Transformed Data

Unit Root Test Results. Before processing time series data, it is essential to determine whether the data constitute stationary time series to avoid spurious regression issues in subsequent analysis. Therefore, this study conducts a unit root test on the data for oil prices and precious metals prices before analyzing their causal relationships. The test ensures that all variables are stationary before they are used for further analysis.

The Augmented Dickey-Fuller (ADF) unit root test is employed in this study, with the optimal lag order selected based on the Akaike Information Criterion (AIC). If the test results reject the null hypothesis of a unit root, the data are deemed stationary. For non-stationary variables, differencing is applied to transform the time series into stationary ones. The unit root test results show that all variables are stationary, allowing for subsequent wavelet correlation and causality analyses.

Table 3. Unit Root Test Results

Variables	WTI	GOLD	SILVER	PLATINUM	PALLADIUM
Adf	-33.070***	-31.301***	-31.360***	-29.749***	-27.356***
PP	-33.101***	-31.342***	-31.372***	-29.738***	-27.249***

Note: *** indicates significance at the 0.01 level, ** at the 0.05 level, and * at the 0.1 level.

Correlation Analysis

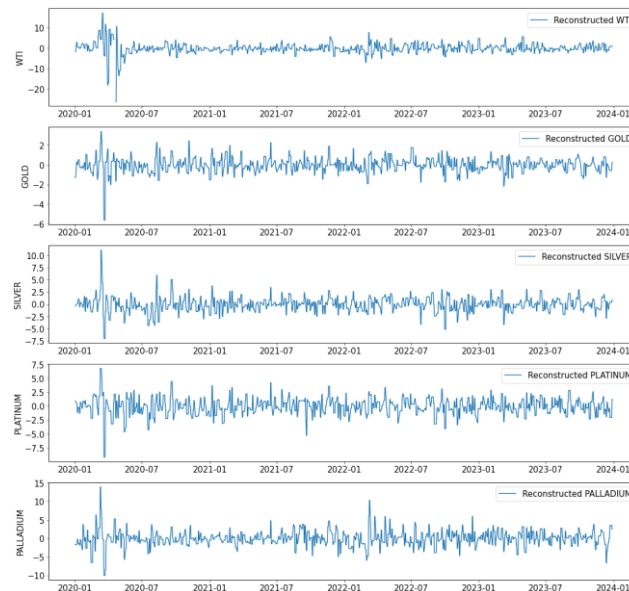
Table 4. Correlation Matrix

Variables	WTI	GOLD	SILVER	PLATINUM	PALLADIUM	WTI
WTI	1					
GOLD	0.1051*	1				
SILVER	0.1712***	0.7760***	1			
PLATINUM	0.1746***	0.5578***	0.6322***	1		
PALLADIUM	0.1798***	0.3766***	0.4600***	0.5667***	1	

4 Results and Discussion

4.1 Continuous Wavelet Transform

The data were decomposed using wavelet analysis into three scales: short-term (1–2 days), medium-term (3–4 days), and long-term (7–8 days). The decomposed results are presented in Figures 2–4:

**Fig. 2.** Results of Short-Term Wavelet Decomposition

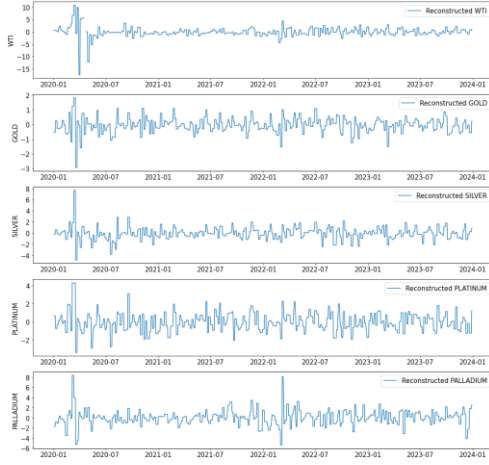


Fig. 3. Results of Medium-Term Wavelet Decomposition

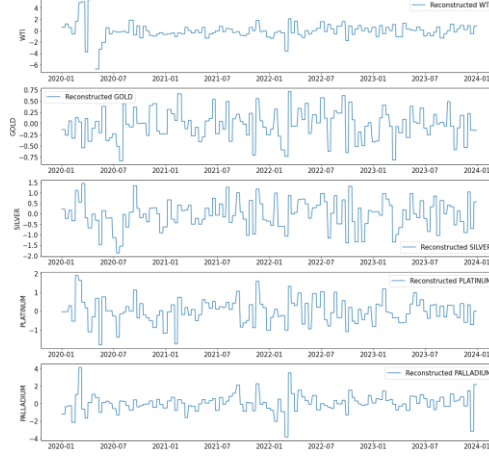


Fig. 4. Results of Long-Term Wavelet Decomposition

4.2 Granger Causality

Using the ADF and PP tests, it was confirmed that the time series for all five variables are stationary. Therefore, Granger causality tests can be conducted directly without the need for cointegration tests. Granger causality is used to determine whether one time series can predict the future values of another. The causal relationships between stationary time series can be analyzed directly using Granger causality tests.

VAR Model. The VAR model established in this study includes five variables. The specific VAR formula is as follows:

$$\begin{aligned} \Delta \ln WTI_t = & \alpha_0 + \sum_{i=1}^k \alpha_{1i} \Delta \ln GOLD_{t-i} + \sum_{i=1}^k \alpha_{2i} \Delta \ln SILV_{t-i} + \sum_{i=1}^k \alpha_{3i} \Delta \ln PLAT_{t-i} \\ & + \sum_{i=1}^k \alpha_{4i} \Delta \ln PALL_{t-i} + \mu_{1t} \end{aligned} \quad (8)$$

$$\begin{aligned} \Delta \ln GOLD_t = & \beta_0 + \sum_{i=1}^k \beta_{1i} \Delta \ln WTI_{t-i} + \sum_{i=1}^k \beta_{2i} \Delta \ln SILV_{t-i} + \sum_{i=1}^k \beta_{3i} \Delta \ln PLAT_{t-i} \\ & + \sum_{i=1}^k \beta_{4i} \Delta \ln PALL_{t-i} + \mu_{2t} \end{aligned} \quad (9)$$

$$\begin{aligned} \Delta \ln SILV_t = & \gamma_0 + \sum_{i=1}^k \gamma_{1i} \Delta \ln GOLD_{t-i} + \sum_{i=1}^k \gamma_{2i} \Delta \ln WTI_{t-i} + \sum_{i=1}^k \gamma_{3i} \Delta \ln PLAT_{t-i} \\ & + \sum_{i=1}^k \gamma_{4i} \Delta \ln PALL_{t-i} + \mu_{3t} \end{aligned} \quad (10)$$

$$\Delta \ln PLAT_t = \delta_0 + \sum_{i=1}^k \delta_{1i} \Delta \ln GOLD_{t-i} + \sum_{i=1}^k \delta_{2i} \Delta \ln SILV_{t-i} + \sum_{i=1}^k \delta_{3i} \Delta \ln WTI_{t-i} + \sum_{i=1}^k \delta_{4i} \Delta \ln PALL_{t-i} + \mu_{4t} \quad (11)$$

$$\Delta \ln PALL_t = \varepsilon_0 + \sum_{i=1}^k \varepsilon_{1i} \Delta \ln GOLD_{t-i} + \sum_{i=1}^k \varepsilon_{2i} \Delta \ln SILV_{t-i} + \sum_{i=1}^k \varepsilon_{3i} \Delta \ln PLAT_{t-i} + \sum_{i=1}^k \varepsilon_{4i} \Delta \ln WTI_{t-i} + \mu_{5t} \quad (12)$$

Where Δ represents the logarithmic difference of the time series, indicating the return rate. K is the optimal lag order, determined through statistical testing. t represents the time period. $\alpha_0, \beta_0, \gamma_0, \delta_0, \varepsilon_0$ are constants in the equations for the five variables. $\mu_{1t}, \mu_{2t}, \mu_{3t}, \mu_{4t}, \mu_{5t}$ are uncorrelated error terms with zero mean.

For the decomposed short-term, medium-term, and long-term data, lag order tests were conducted again: Short-term: Optimal lag order = 1. Medium-term: Optimal lag order = 3. Long-term: Optimal lag order = 4. Stability tests and impulse response analyses were then performed based on these lag orders.

Table 5. VAR Model Results

Period	Equation	RMSE	R-sq	chi2	P>chi2
Short-term	wti	2.36603	0.3131	438.5885	0
	gold	0.652383	0.2715	358.5682	0
	silver	1.27394	0.3242	461.5452	0
	platinum	1.31767	0.2718	358.9921	0
	palladium	1.84426	0.3036	419.3436	0
Medium-term	wti	2.12992	0.0737	76.08472	0
	gold	0.520489	0.0492	49.49378	0
	silver	1.15347	0.0487	48.91123	0
	platinum	1.036	0.0474	47.57889	0
	palladium	1.53894	0.0501	50.43206	0
Long-term	wti	1.10863	0.3234	453.9792	0
	gold	0.295153	0.2107	253.6583	0
	silver	0.604083	0.2494	315.7081	0
	platinum	0.570296	0.2331	288.7322	0
	palladium	0.927895	0.1849	215.5255	0

To verify the stability of the Vector Autoregression (VAR) model established in this study, the inverse roots of the characteristic polynomial were plotted. If all inverse roots are distributed within the unit circle, the model is considered stable and suitable for further impulse response

analysis. As shown in Figure 6, all eigenvalues of the VAR model lie within the unit circle, indicating that the established VAR system is stable.

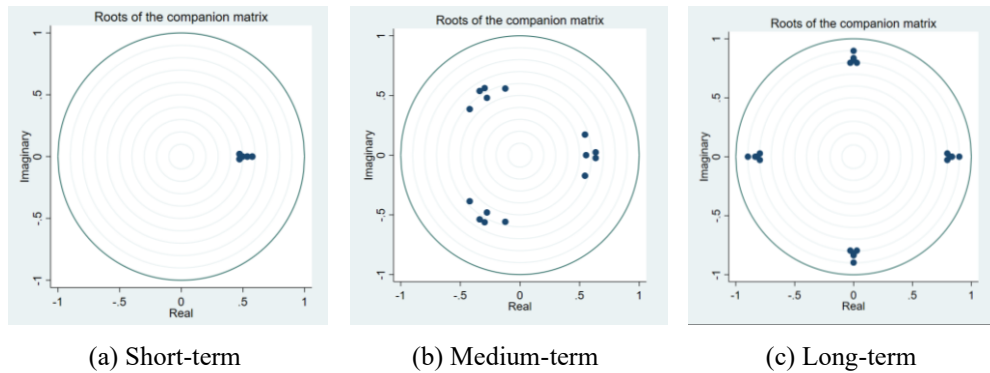


Fig. 5. VAR System Stability Test

Granger Causality Analysis. This section compares the short-term, medium-term, and long-term causal relationships among variables to observe consistent patterns or variations. Before conducting the Granger causality test, all variable sequences must be stationary. If a variable sequence is non-stationary, it should be differenced to achieve stationarity. In the Granger causality test, Equation represents the dependent variable being predicted, and Excluded refers to the variable being tested as a potential Granger cause.

In the short term, the following significant causal relationships are observed: Oil has a significant causal relationship with gold, silver, and palladium. Gold exhibits a significant causal relationship with oil and silver. Silver shows a significant causal relationship with oil and palladium. Platinum has a significant causal relationship with gold and silver. All variables demonstrate significant causal relationships with oil, gold, silver, and platinum. In the medium term, the relationships are similar: Oil maintains a significant causal relationship with gold, silver, and palladium. Gold continues to exhibit a significant causal relationship with oil and silver. Silver retains a significant causal relationship with oil and palladium. Platinum still demonstrates a significant causal relationship with gold and silver. All variables show significant causal relationships with oil, gold, silver, and platinum. In the long term, the relationships shift slightly: Oil has a significant causal relationship with gold, platinum, and silver. Platinum shows a significant causal relationship with gold and palladium. All variables demonstrate significant causal relationships with oil and platinum. The overall significance of the model highlights that when all variables are considered, they exhibit complex dynamic relationships with one another.

Table 6. Granger Causality Test Results

Equa- tion	Exclu- ded	Short -term	Medi -um- term	Long -term	Equation	Exclu- ded	Short -term	Medium -term	Long -term
wti	gold	***	***		platinum	wti			***
wti	silver	***	**		platinum	gold	*		**
wti	platinum			***	platinum	silver	**		

Table 6. (continued).

wti	palladium		***	*	platinum	palladium	***	**
wti	ALL	***	***	**	platinum	ALL	**	***
gold	wti		**	*	palladium	wti		
gold	silver	*	***		palladium	gold		
gold	platinum				palladium	silver		
gold	palladium		**		palladium	platinum		
gold	ALL	*	***		palladium	ALL		
silver	wti		**	**				
silver	gold							
silver	platinum							
silver	palladium		**					
silver	ALL		***	*				

Impulse Response Analysis. Impulse response graphs illustrate how a shock (impact variable) to one variable affects another variable (response variable) over time. Each graph displays an impact variable, a response variable, and the changes in the response variable over time steps.

From the short-term impulse response graphs (Figure 6), the following observations can be made: Self-Responses: All five commodities show a gradual reduction in their self-responses over time, eventually converging to zero. Gold: Exhibits a small and insignificant response to shocks from platinum, palladium, and silver. Shows a larger and significant response to shocks from oil, with the confidence interval not including zero, indicating a notable impact. Silver: Displays small and insignificant responses to shocks from gold, platinum, and palladium. Shows a larger and significant response to shocks from oil, with the confidence interval not including zero, indicating a notable impact. Palladium: Exhibits small and insignificant responses to shocks from gold, platinum, silver, and oil, with confidence intervals including zero, indicating no significant impact. Platinum: Demonstrates small and insignificant responses to shocks from gold, palladium, silver, and oil, with confidence intervals including zero, indicating no significant impact. Oil: Shows a small and insignificant response to shocks from gold, platinum, and palladium. Displays a larger response to shocks from silver.

The medium-term (Figure 7) and long-term (Figure 8) impulse response results are largely consistent with the short-term findings: Gold: Shows significant responses to shocks from oil and silver. Oil and Silver: Show significant responses to shocks from each other. The other variables (platinum and palladium) maintain small and insignificant responses to shocks from most variables, with confidence intervals including zero.

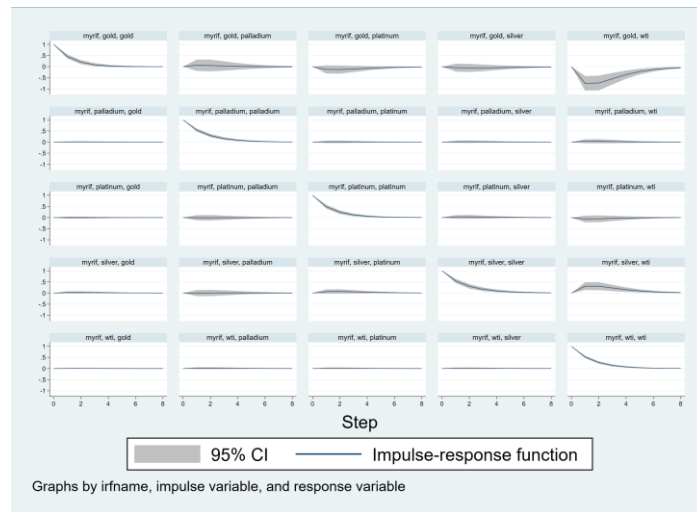


Fig. 6. Short-Term Impulse Response Results

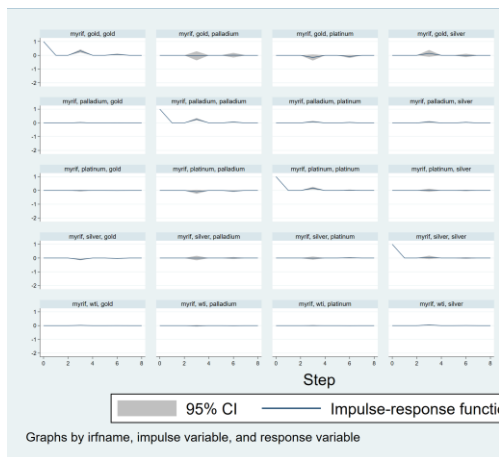


Fig. 7. Medium-Term Impulse Response Results

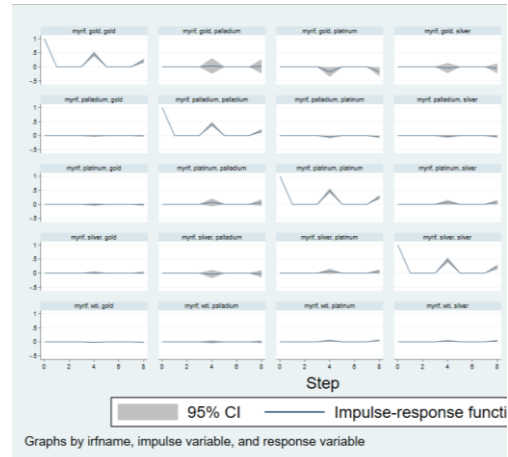


Fig. 8. Long-Term Impulse Response Results

5 Conclusion

Oil and precious metals are not only critical commercial commodities but also key factors influencing the global economy, geopolitics, and financial markets. Understanding their economic significance is essential for predicting market trends, formulating policies, and making strategic investment decisions.

This study analyzed the price indices of five commodities to explore the relationships between oil and precious metals during periods of economic turbulence and geopolitical events.

Using Discrete Wavelet Transform (DWT), the time series were decomposed into short-term, medium-term, and long-term components. A Vector Autoregression (VAR) model was then constructed to conduct Granger causality tests and impulse response analyses for oil and precious metal prices.

The results reveal the following: Oil exhibits significant causal relationships with several variables across short-term, medium-term, and long-term horizons, particularly with gold and silver. Gold and platinum also show significant causal relationships with other variables during different time periods. In the short term, oil has a unidirectional Granger causality with gold and silver. In the medium term, oil and gold, as well as oil and silver, display bidirectional Granger causality. In the long term, oil and platinum exhibit bidirectional Granger causality. Overall, oil, gold, and silver demonstrate strong market influence, whereas platinum and palladium show relatively weaker effects. In the short term, impulse response analyses reveal significant dynamic relationships among gold, oil, and silver, particularly for the impacts of gold on oil, silver on oil, and oil on silver. Conversely, the relationships among other variables are weaker or insignificant.

Future research could further explore the dynamic changes in the relationship between oil and precious metals during different economic periods. Analyzing the effects of varying policies and shifts in the global economic environment on these relationships would provide valuable insights for promoting stable global economic development.

References

- [1] Bossman, A., Gubareva, M., & Teplova, T. (2023). EU sectoral stocks amid geopolitical risk, market sentiment, and crude oil implied volatility: An asymmetric analysis of the Russia-Ukraine tensions. *Resources Policy*, 82. <https://doi.org/xxxx>
- [2] Guo, S. J. (2016). An empirical study on the price relationship of coal, oil, and natural gas [Master's thesis]. xxxx University.
- [3] Li, X. Y., Guo, J., & Huang, Y. (2016). An empirical study of China's economic growth and its relationship with oil and gas consumption: Based on cointegration analysis and Granger causality test. *Gansu Science Journal*, 28(3), 125–129.
- [4] Zhao, P. (2021). An empirical study on the relationship between international crude oil prices and port crude oil throughput. *Chemical Management*, (34), 35–36, 62.
- [5] Alkathery, M. A., Chaudhuri, K., & Nasir, M. A. (2022). Implications of clean energy, oil, and emissions pricing for the GCC energy sector stock. *Energy Economics*, 112. <https://doi.org/xxxx>
- [6] Alaali, F. (2020). The effect of oil and stock price volatility on firm-level investment: The case of UK firms. *Energy Economics*, 87. <https://doi.org/xxxx>
- [7] Granger, C. W. J. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3), 424–438.
- [8] Li, T., & Zhu, J. (2009). A study on the linkage relationship between gold prices, foreign exchange, and crude oil prices. *China Business*, (10), 66.
- [9] Wang, Y. S., & Chueh, Y. L. (2013). Dynamic transmission effects between the interest rate, the US dollar, and gold and crude oil prices. *Economic Modelling*, 30, 792–798. <https://doi.org/xxxx>
- [10] Liu, J. (2017). Analysis of the price linkage between oil and gold. *Gold*, 38(2), 5–7.
- [11] Liu, X. Y., & Zhu, C. M. (2008). An empirical analysis of the correlation between U.S. dollar depreciation and crude oil price fluctuations. *International Financial Research*, (11), 50–55.

- [12] Guo, S. J. (2014). Oil rents, coal rents, and natural gas rents: An empirical study based on the cointegrated VAR model. *New Economy*, (26), 33–34.
- [13] Guo, M. Y., & Wang, N. (2015). The impact of crude oil price fluctuations on the returns of China's basic industries: An empirical study based on the Granger causality test. *Journal of Beijing Institute of Technology (Social Sciences Edition)*, 17(4), 18–27.
- [14] Gharib, C., Mefteh-Wali, S., & Ben Jabeur, S. (2021). The bubble contagion effect of COVID-19 outbreak: Evidence from crude oil and gold markets. *Finance Research Letters*, 38. <https://doi.org/xxxx>
- [15] Gao, X. W., & Li, Q. (2012). An empirical study on the impact of international oil prices and the U.S. dollar exchange rate on the oil drilling investment market. *Systems Engineering*, 30(10), 56–62.
- [16] Liu, T., & Gong, X. (2020). Analyzing time-varying volatility spillovers between the crude oil markets using a new method. *Energy Economics*, 87. <https://doi.org/xxxx>
- [17] Chen, G. Y. (2020). The dynamic relationship between international oil prices and China's inflation: An empirical study based on the VAR model. *Market Research*, (7), 22–24.
- [18] Ma, D. (2020). An empirical study on the influencing factors of international gold prices based on the VEC model. *Journal of Ningxia Teachers University*, 41(6), 106–112.
- [19] Gao, H., & Gao, T. C. (2022). Have domestic crude oil futures promoted RMB internationalization? An empirical study based on the cointegration model using monthly data from 2018 to 2021. *China Securities Futures*, (1), 23–43.
- [20] Dai, Z., Zhu, H., & Zhang, X. (2022). Dynamic spillover effects and portfolio strategies between crude oil, gold, and Chinese stock markets related to new energy vehicles. *Energy Economics*, 109. <https://doi.org/xxxx>
- [21] Dickey, D. A., & Fuller, W. A. (1979). Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American Statistical Association*, 74(366a), 427–431. <https://doi.org/xxxx>
- [22] Sims, C. A. (1977). Macroeconomics and reality. *Econometrica*, 48(1), 1–48.
- [23] Ng, S., & Perron, P. (2001). Lag length selection and the construction of unit root tests with good size and power. *Econometrica*, 69(6), 1519–1554.

Selection in Heavy Ion Collisions Through Event Shape Engineering

Zhiyan Yu

School of Art and Sciences, University of Rochester, Rochester, 14627, USA

zyu25@u.rochester.edu

Abstract. This study uses Glauber Monte Carlo simulations to investigate the connection between initial collision geometry and potential jet quenching effects. A dataset of 75,000 simulated collision events was examined, with a focus on the associations between geometrical parameters such as participant numbers, eccentricity, triangularity, and their respective orientation angles, and the impact parameters. By calculating path length differences for each event, optimal ranges of such variables were identified, which could possibly enhance observable jet quenching. Our findings show that mid-central collisions with 100 to 200 participants, higher eccentricity, and impact parameters of 8 fm provide the optimal conditions for studying path length-dependent phenomena.

Keywords: Glauber Monte Carlo simulations, Heavy ion collisions, Event shape engineering

1 Introduction

The Glauber Monte Carlo model is a computational technique to simulate the initial geometry of heavy ion collisions. The colliding nuclei are modeled through random distribution of nucleons with probabilities determined by measured nuclear density profiles [1]. These are then "collided" by overlapping them at a uniformly distributed random impact parameter whereby nucleon-nucleon collisions were determined based on the geometrical cross-section [2].

This simulation calculates several important quantities, including the number of participating nucleons N_{part} [3], the impact parameter b , the eccentricity and triangularity of the overlap region ϵ_2 and ϵ_3 [4], and the angle between the shortest axis of the elliptical shape and orientation of the triangular shape in the initial collision geometry ψ_2 and ψ_3 . By running multiple simulated collisions, the model generates distributions of these quantities that can be compared to experimental results.

The Glauber Monte Carlo model is important for interpreting data from heavy ion collisions and understanding how initial geometry influences the evolution of the quark-gluon plasma formed in such events. This model links experimentally observed particle multiplicities to the initial collision geometry, making it a vital tool for heavy ion collision data analysis.

Event-by-event fluctuations in these collisions are essential for understanding the complex dynamics of such systems [5]. These fluctuations involve variations in initial geometry, energy density, and other properties from one collision to another. They originate from the quantum mechanical nature of colliding nuclei and the probabilistic nature of nucleon-nucleon interactions [6].

The significance of these fluctuations is multifaceted. They heavily influence the initial state geometry, leading to differences in the shape and size of the interaction region [7]. This affects the initial spatial eccentricity, which is a key factor driving collective flow in the quark-gluon plasma as it expands [8]. Flow patterns observed, especially higher-order harmonics, are particularly sensitive to these initial state fluctuations. As discussed in Ref [9], the ratio of triangular flow to elliptic flow increases for more central collisions and higher transverse momentum particles, a trend that aligns with observations in experimental data. By studying these flow patterns, we can gain insights into the initial state geometry and the subsequent hydrodynamic evolution of the system.

Knowing the importance of event-by-event fluctuations, one can study the correlations among variables through event shape engineering [10]. This technique allows us to exploit the natural event-by-event fluctuations in heavy ion collisions by selecting events with specific initial geometry configurations. It provides a unique opportunity to isolate and study the effects of initial state geometry on the evolution and properties of the quark-gluon plasma.

2 Data and Analysis

To study event-by-event fluctuations, a large enough data set is required for representative behaviors. Therefore, a data set consisting of 75000 independent events generated through Glauber MC is applied in the experiment. To keep the data points in a reasonable region, impact parameter b is set to be within 0-12 and the number of participants is set to be above 50.

It would be reasonable to ask how the main variables relate. If some innate correlations were found, reducing them into expressions of other variables to simplify the model is beneficial. This can be primarily done through 2D histograms.

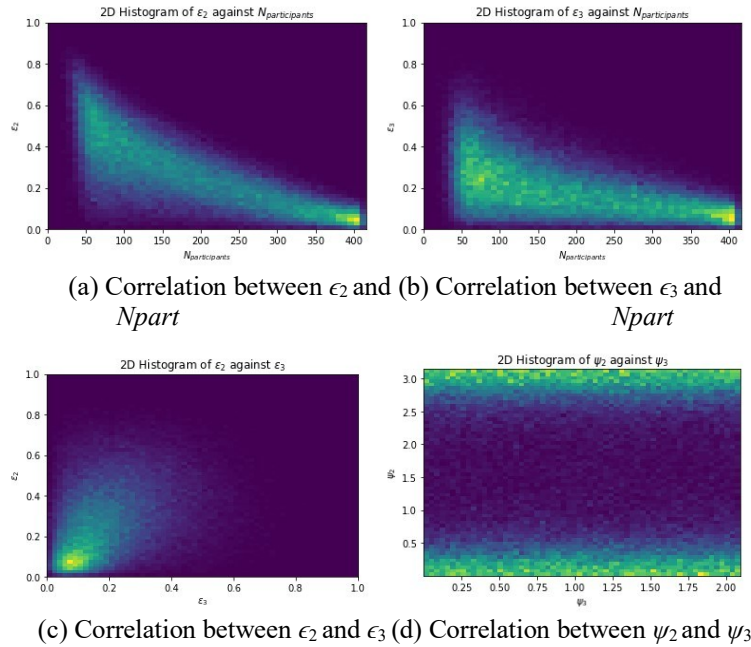


Fig. 1. 2D histograms of dependence among variables

Shown in Figure 1, the correlations among important variables are intuitive to observe. There's a weak dependence both between ϵ_2 and N_{part} and between ϵ_3 and N_{part} . This is mainly due to the nature of these variables, where lower values of ϵ_2 and ϵ_3 usually show that head-on collisions are more likely to happen. This situation will naturally result in a larger N_{part} , which aligns with the findings in (a) and (b). In (c), the correlation between ϵ_2 and ϵ_3 is relatively weak since they are highly concentrated in lower values and diffuse in all directions. In (d), distribution is primarily uniform, and no correlations between ψ_2 and ψ_3 were found. This method can be applied to any other variable combination, though these correlations are the most likely to occur.

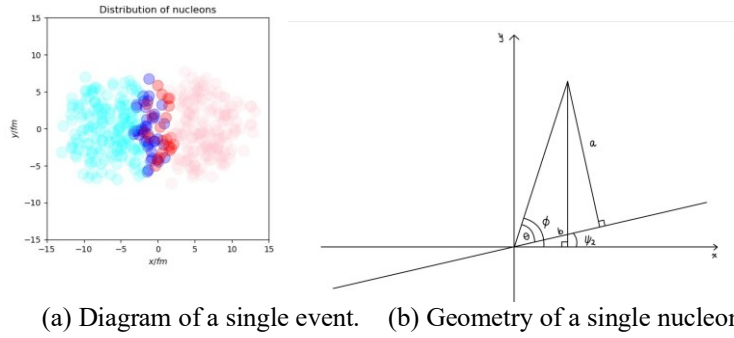


Fig. 2. (a) a random collision event that shows the distribution inside the participating nuclei. The blue and red units are the participating nucleons from nuclei 1 and 2, whereas cyan and pink units are the spectators from nuclei 1 and 2 respectively. (b) shows the position of each nucleon relative to the center of the collision.

To actively select conditions that maximize the jet quenching effect, the path lengths and path length differences for each collision event must be calculated. This is achieved by counting the number of participants in contact with the major and minor axes of the ellipse formed by the participants. This can be done by drawing the two axes and counting the number of nucleons they travel through, or by geometry and trigonometry.

As shown in Figure 2, geometry is applied in this case. Since we can calculate the position of the center of collision, the relative position of each nucleon to the center can be calculated using diagram (b). $\phi = \arctan\left(\frac{y_{diff}}{x_{diff}}\right)$, so $\theta = \phi - \psi_2$.

Therefore, we have:

$$a = \sqrt{y_{diff}^2 + x_{diff}^2} \cdot \sin(\theta) \quad (1)$$

$$b = \sqrt{y_{diff}^2 + x_{diff}^2} \cdot \cos(\theta) \quad (2)$$

Hence, the nucleon is said to intersect with the axes if a or b is equal to or less than the radius of the nucleon. If a is less than the radius, $N_{parallel}$ increases by 1, and vice versa. Finally, $\Delta N = N_{perpendicular} - N_{parallel}$. Radius is calculated using $\sigma_{NN} = 42mb$ and $D = \sqrt{\frac{\sigma_{NN}}{\pi}}$ [1].

After applying this to every event, the relationships between ΔN and other variables can be studied. Instead of plotting 2D histograms like how it was done previously, each variable is

divided into 10 bins, and the average of ΔN is taken in each bin. Therefore, line charts of $\langle \Delta N \rangle$ against each variable can be plotted.

According to Figure 3, each variable has a different impact on ΔN . General predictions can be made based on the line charts. The average peaks when

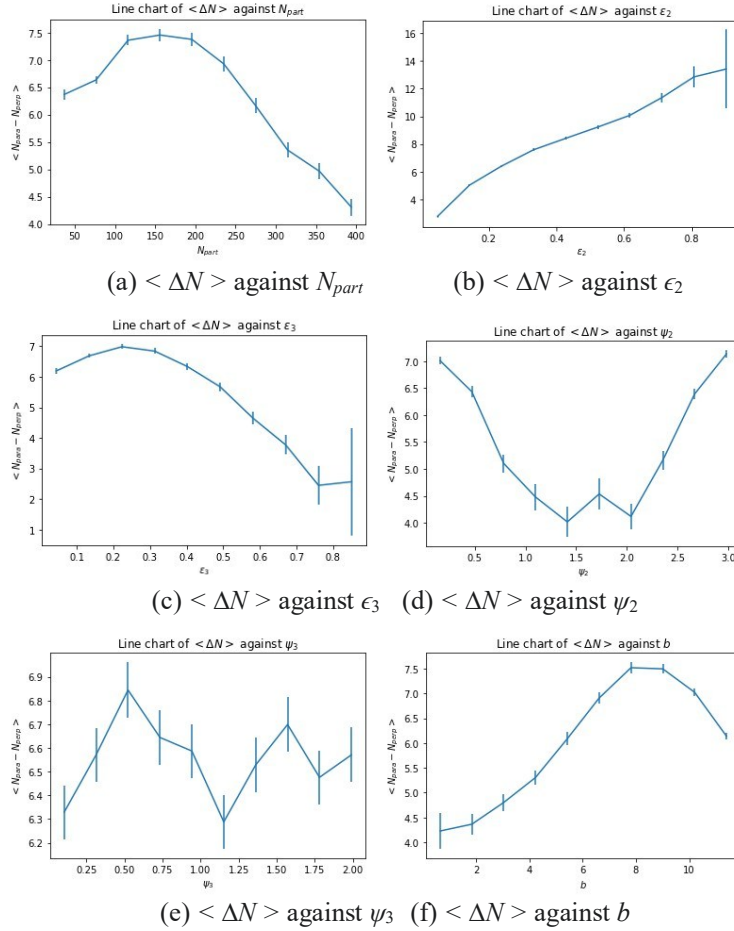


Fig. 3. Line charts of dependence between $\langle \Delta N \rangle$ and other variables. The uncertainties are calculated by: $\frac{\text{rms}}{\sqrt{n}}$, where rms is the root mean square value of each bin and n is the number of events enclosed in the corresponding bin.

N_{part} is between 100 and 200, with relatively small error bars, indicating that selections with N_{part} in this range are preferred. Higher values of ϵ_2 results in high averages. Though the error rises, an increasing trend can still be seen. The same method can be applied to four other plots. One can discover that the average peaks when ϵ_3 is around 0.2, ψ_2 is < 0.5 or > 2.7 , and b is around 8. The only exception is ψ_3 , where most error bars overlap with others and the differences between two points are very small. Therefore, one can conclude that the dependence of $\langle \Delta N \rangle$ on ψ_3 is weak.

With the predictions, the dependence of the average on all variables can be studied. The same approach can be taken except for bin sizes. Each parameter is divided into four bins to avoid having only one or two events in each bin. Results can be sorted by $\langle \Delta N \rangle$, and the 40 events with the highest average values are shown in Table 1.

Table 1. 40 events with highest $\langle \Delta N \rangle$.

$\langle \Delta N \rangle$	$\Delta \langle \Delta N \rangle$	N_{part}	ϵ_2	ϵ_3	ψ_2	ψ_3	b
31.000000	31.000000	3.0	3.0	1.0	1.0	3.0	2.0
28.000000	28.000000	2.0	3.0	2.0	3.0	4.0	3.0
25.000000	25.000000	4.0	1.0	2.0	3.0	4.0	1.0
23.000000	23.000000	2.0	3.0	1.0	2.0	1.0	4.0
23.000000	23.000000	2.0	2.0	3.0	3.0	4.0	3.0
23.000000	23.000000	3.0	1.0	3.0	3.0	2.0	3.0
22.000000	22.000000	1.0	3.0	1.0	4.0	1.0	3.0
21.000000	21.000000	3.0	2.0	1.0	3.0	4.0	3.0
21.000000	21.000000	4.0	2.0	1.0	1.0	3.0	1.0
21.000000	21.000000	3.0	2.0	1.0	3.0	3.0	3.0
20.000000	20.000000	2.0	3.0	3.0	4.0	2.0	3.0
20.000000	20.000000	1.0	3.0	2.0	4.0	1.0	3.0
19.000000	19.000000	1.0	4.0	1.0	3.0	3.0	4.0
18.500000	13.086252	1.0	4.0	2.0	3.0	1.0	4.0
18.500000	13.124405	1.0	3.0	1.0	4.0	3.0	3.0
18.000000	18.000000	4.0	2.0	2.0	3.0	4.0	2.0
18.000000	18.000000	1.0	2.0	1.0	4.0	3.0	3.0
18.000000	18.000000	4.0	1.0	2.0	4.0	3.0	1.0
18.000000	18.000000	1.0	3.0	2.0	1.0	1.0	3.0
17.500000	3.780891	2.0	3.0	2.0	1.0	1.0	3.0
17.000000	17.000000	3.0	2.0	1.0	2.0	1.0	3.0
17.000000	12.041595	1.0	3.0	2.0	4.0	4.0	3.0
17.000000	17.000000	1.0	4.0	2.0	2.0	2.0	4.0
16.661538	2.208697	2.0	3.0	1.0	4.0	4.0	3.0
16.500000	11.672618	1.0	3.0	1.0	1.0	3.0	3.0
16.173913	3.719361	2.0	3.0	2.0	4.0	2.0	3.0
16.000000	16.000000	3.0	2.0	1.0	3.0	4.0	2.0
15.789474	3.929433	2.0	3.0	2.0	4.0	4.0	4.0
15.571429	6.345495	4.0	2.0	1.0	4.0	3.0	2.0
15.518519	2.245098	2.0	3.0	1.0	4.0	3.0	3.0
15.315789	3.767848	2.0	3.0	2.0	4.0	1.0	3.0
15.200000	8.094443	4.0	2.0	1.0	1.0	1.0	2.0

Table 1. (continued).

15.000000	15.000000	1.0	4.0	2.0	3.0	2.0	4.0
15.000000	15.000000	1.0	4.0	1.0	2.0	1.0	4.0
15.000000	6.155395	1.0	3.0	3.0	2.0	2.0	4.0
14.947368	3.676684	2.0	3.0	2.0	1.0	4.0	3.0
14.777778	2.226693	2.0	3.0	1.0	4.0	2.0	3.0
14.731707	2.411059	2.0	3.0	1.0	4.0	3.0	4.0
14.621622	2.600079	2.0	3.0	1.0	1.0	4.0	3.0
14.612245	2.220648	2.0	3.0	1.0	1.0	2.0	3.0

The uncertainties are calculated by: $\frac{rms}{\sqrt{n}}$, where rms is the root mean square value of each bin and n is the number of events enclosed in the corresponding bin.

The uncertainties of the first several bins are high since only one event was enclosed. Ruling out these statistical fluctuations, it's pretty clear that N_{part} falls in bin 2, meaning that it should be 25% to 50% of the range, which is consistent with our previous prediction of 100 to 200. Meanwhile, ϵ_2 are in bin 3, ϵ_3 are in bin 1, ψ_2 are in bin 4, ψ_3 do not have a particular pattern, and b are in bin 3, which agree with our previous predictions.

3 Interpretation

The analysis of event-by-event fluctuations in heavy ion collisions using the Glauber Monte Carlo model reveals several important insights into the initial geometry of these collisions and their impact on observable quantities.

The weak dependence observed between eccentricity ϵ_2 and participant number N_{part} , as well as between triangularity ψ_3 and N_{part} , aligns with our understanding of collision geometry. Lower eccentricity and triangularity values typically correspond to more central collisions, which naturally involve a larger number of participants. This relationship underscores the connection between the impact parameter and the shape of the interaction region.

The calculation of path length differences $\langle \Delta N \rangle$ provides a crucial link between initial geometry and potential jet-quenching effects. The observed relationships between $\langle \Delta N \rangle$ and various geometric parameters offer valuable insights:

1. The peak in $\langle \Delta N \rangle$ for N_{part} between 100 and 200 suggests that midcentral collisions may provide the optimal conditions for studying path length effects on jet quenching.
2. The positive correlation between $\langle \Delta N \rangle$ and ϵ_2 indicates that more elliptical collision geometries lead to larger path length differences, potentially enhancing observable jet quenching effects.
3. The weak dependence of $\langle \Delta N \rangle$ on ψ_3 implies that the orientation of triangularity has minimal impact on path length differences, focusing our attention on ellipticity as the primary driver of these effects.

4 Conclusions

This study demonstrates the power of event shape engineering techniques in heavy ion collisions using Glauber Monte Carlo simulations. By analyzing the correlations between various geometric parameters and their impact on path length differences, we have gained valuable insights into the initial state geometry of these collisions and their potential effects on observable quantities.

Future work may incorporate these event shape engineering techniques into full hydrodynamic simulations to study the evolution of the quark-gluon plasma under specific initial geometry conditions [11]. More sophisticated event selection algorithms that can efficiently identify and categorize events based on multiple geometric criteria can be developed.

References

- [1] Alver, B. H., Baker, M. D., Loizides, C., and Steinberg, P. (2008). The PHOBOS Glauber Monte Carlo.
- [2] Loizides, C., Kamin, J., & d'Enterria, D. (2018). Improved monte carlo glauber predictions at present and future nuclear colliders. *Physical Review C*, 97(5). <https://doi.org/10.1103/physrevc.97.054910>
- [3] Konchakovski, V. P., Gorenstein, M. I., Bratkovskaya, E. L., & Greiner, W. (2010). Fluctuations and correlations in nucleus–nucleus collisions within transport models. *Journal of Physics G: Nuclear and Particle Physics*, 37(7), 073101. <https://doi.org/10.1088/0954-3899/37/7/073101>
- [4] Mazeliauskas, A., & Teaney, D. (2015). Subleading harmonic flows in hydrodynamic simulations of heavy ion collisions. *Physical Review C*, 91(4). <https://doi.org/10.1103/physrevc.91.044902>
- [5] Stephanov, M., Rajagopal, K., & Shuryak, E. (1999). Event-by-event fluctuations in heavy ion collisions and the QCD critical point. *Physical Review D*, 60(11). <https://doi.org/10.1103/physrevd.60.114028>
- [6] An, X., Başar, G., Stephanov, M., & Yee, H.-U. (2020). Fluctuation dynamics in a relativistic fluid with a critical point. *Physical Review C*, 102(3). <https://doi.org/10.1103/physrevc.102.034901>
- [7] Bhalerao, R. S., Ollitrault, J.-Y., Pal, S., & Teaney, D. (2015). Principal component analysis of event-by-event fluctuations. *Physical Review Letters*, 114(15). <https://doi.org/10.1103/physrevlett.114.152301>
- [8] Jia, J., & Huo, P. (2014). Forward-backward eccentricity and participant-plane angle fluctuations and their influences on longitudinal dynamics of collective flow. *Physical Review C*, 90(3). <https://doi.org/10.1103/physrevc.90.034915>
- [9] Alver, B., and Roland, G. (2010). Collision-geometry fluctuations and triangular flow in heavy-ion collisions. *Physical Review C*, 81(5). <https://doi.org/10.1103/physrevc.81.054905>
- [10] Jia, J. (2022). Shape of atomic nuclei in heavy ion collisions. *Physical Review C*, 105(1). <https://doi.org/10.1103/physrevc.105.014905>
- [11] Kumar, R., Dexheimer, V., Jahan, J. et al. Theoretical and experimental constraints for the equation of state of dense and hot matter. *Living Rev Relativ* 27, 3 (2024). <https://doi.org/10.1007/s41114-024-00049-6>

Exploring Jet Quenching Phenomena in Proton-Proton Collisions through Monte Carlo Simulation and Data Analysis

Yize Mo^{1,3,†}, Kerao Zhang^{2,4,*†}

¹Department of Physics, Jilin University, Jilin, 130012, China

²School of Nuclear Science and Engineering, North China Electric Power University, Beijing, 10096, China

³moyz1120@mails.jlu.edu.cn

⁴120211110425@ncepu.edu.cn

*corresponding author

[†]These authors contributed equally and should be considered co-first authors.

Abstract. Jet quenching is a significant phenomenon for studying the properties of the Quark-Gluon Plasma (QGP), typically observed in heavy-ion collisions. This phenomenon refers to the substantial energy loss experienced by high-energy jets as they traverse the QGP. This study explores whether a similar effect occurs in proton-proton (pp) collisions. We simulated 10,000 proton-proton collision events using the PYTHIA event generator, with a center-of-mass energy set to 5360 GeV and a minimum transverse momentum threshold (p_{Tmin}) of 500 GeV. Neutrino events were excluded from the simulation to focus on high-energy jets. The jet reconstruction was carried out using FastJet software and the inverse k_T algorithm ($R = 0.4$), followed by fitting the four-momentum of the reconstructed jet using the ATLAS jet energy resolution curve, which helps to approximate the actual detector response more accurately. This study focuses on the correlation between the missing transverse momentum (MET) and the secondary jet volume ratio (p_{T2}/p_{T1}). It is found that the MET value falls below 40 GeV when p_{T2}/p_{T1} is smaller than 0.4 and, due to the coverage of the detector, when the contribution of W bosons or neutrinos is extremely small. In addition, we study the distributions of the leading and secondary jets in terms of pseudo-combination (η) and polar angle (θ) in this case. This study provides a background for observing jet quenching in heavy-ion collisions and helps to analyse emerging physical phenomena in heavy-ion collisions.

Keywords: component, Jet quenching, pp collision, missing transverse momentum, QGP.

1 Introduction

Jet quenching is one of the main tools for studying quark-gluon plasma (QGP). It is defined as the suppression of hadron yields at high transverse momentum (p_T) due to the loss of energy due to hard scattering of particles as they traverse the medium[1].

Quark-Gluon Plasma (QGP) is a state of matter that is characterised by an extremely high temperature and density. In this state, quarks and gluons are no longer confined within protons and neutrons, but exist in a free, deconfined state. The formation of QGPs necessitates the

attainment of exceedingly high energy densities, a feat typically accomplished through heavy-ion collisions in the context of nuclear physics experiments[2].

In the quark-gluon plasma (QGP) state, where the strong interaction between quarks and gluons plays a major role, it is possible to investigate the fundamental properties of the strong interaction described by quantum chromodynamics (QCD) by studying the QGP[3]. Study of jet quenching phenomenon helps to obtain the properties of QGP[4].

These phenomena are explored primarily in heavy-ion collision experiments (e.g., experiments at the Large Hadron Collider (LHC) and the Relativistic Heavy Ion Collider (RHIC))[5]. By studying changes in jet structure and jet quenching, researchers can learn something about the physical properties of the QGP, and this information will advance the development of our studies of early universe conditions and fundamental interactions.

The principal aim of this study is to investigate, through Monte Carlo simulations and data analysis, whether the phenomenon of jet quenching observed in heavy-ion collisions also occurs in proton-proton (pp) collisions. This study focuses on the reconstruction and selection processes of jets, in particular the behaviour of high-momentum jets. We study similar energy loss effects in pp collisions by than observing the momentum distribution and energy loss patterns of the jet.

2 Experimental Method

In this study, we use the PYTHIA simulator and FastJet jet reconstruction to investigate jet quenching in proton-proton (pp) collisions. The methodology is described below.

2.1 Basic Definitions and CMS Coordinate System

We used the coordinate system of the CMS detector. In this coordinate system, the z-axis is aligned with the direction of the beam, the x-axis points to the centre of the LHC ring, and the y-axis is perpendicular to the plane of the LHC ring and points upwards. The transverse momentum, p_T , is defined as follows:

$$p_T = |\vec{p}| \sin \theta$$

The θ is the polar angle relative to the z-axis. The pseudorapidity η is defined as:

$$\eta = -\ln[\tan(\theta/2)]$$

The pseudorapidity is used to describe the distribution of particles along the z axis.

2.2 Simulation of Event Generation

To simulate proton-proton (pp) collisions, we use the PYTHIA 8.3.12 [6] event generator. PYTHIA 8.3.12 is a powerful Monte Carlo simulation tool which can simulate a variety of complex physical processes in high-energy particle collisions in detail, including the strong interaction mechanism, the formation of jets, scattering between quarks and gluons, and secondary particle production. The advantage of PYTHIA 8.3.12 is that it can provide researchers with accurate simulations of collision events that occur in real experiments, thus supporting data analysis and theoretical predictions.

We used PYTHIA to generate 10,000 proton-proton collision events. The collision centre energy (eCMS) was set to 5360 GeV. In order to generate jet events with the highest possible energy, we set the minimum transverse momentum threshold (p_{Tmin}) of the generation process to 500 GeV.

In the event generation process, we excluded neutrinos in the final particle part. This choice is based on the detection properties of neutrinos: as electrically neutral elementary particles, neutrinos interact very weakly with matter and are therefore difficult to detect directly by experimental detectors.[7] Their exclusion helps to focus clearly on the jet quenching phenomenon.[8]

2.3 Event Collection and Jet Reconstruction

After simulating proton-proton (pp) collision events, we used FastJet[9] software to perform a detailed jet reconstruction analysis of the generated particle data. FastJet can handle a large number of particle events and reconstruct the four-dimensional momentum of the jet from the particle data.

In this investigation, the esteemed anti- k_T algorithm[10] was employed for jet reconstruction, with a discerning selection of a jet radius parameter ($R = 0.4$). This algorithm, widely recognized for its resilience and immunity to soft radiation, stands as a prominent choice in contemporary high-energy physics experiments. The method of this algorithm is to gradually merge particles with similar positions to construct the jet, and its main advantage lies in the suppression of soft radiation, which effectively circumvents the reconstruction errors caused by low-momentum particles or background noise, thus providing a more accurate jet structure.

In the field of high-energy physics experiments, the choice of jet radius directly affects the accuracy and effectiveness of jet reconstruction. Reducing the jet radius (R) facilitates an enhanced delineation of neighbouring jets, thus reducing soft radiation and potential events. In addition, this method also improves the sensitivity to high-momentum jets. Therefore, we deliberately chose ($R = 0.4$) in our study with the aim of obtaining more reliable experimental data by striking a balance between reconstruction accuracy and reduction of background interference.[11, 12]

2.4 Jet Energy Fitting and Selection

In order to enhance the realism of the simulated data, a Gaussian fit was performed on the reconstructed jet tetra-momentum using the jet energy resolution curve from the ATLAS detector. The aforementioned fitting procedure enables the simulation of the response characteristics of a genuine detector, thereby guaranteeing that the reconstructed jet energy is in close alignment with the measurements observed under experimental conditions.

Following the fitting process, a further selection of jet events was performed to guarantee that the jets included in the subsequent analysis were within the physical coverage of the detector and to observe high-momentum jets. We set the selection criteria: transverse momentum (p_T) of the forward jet greater than 600 GeV and pseudo-amplitude ($|\eta|$) less than 1. These criteria are qualitatively based on the physical coverage of the actual detector, and are intended to ensure that we only analyse high-momentum jet events that can be efficiently detected experimentally[13].

This selection process filters out jets that are difficult to measure accurately because they are located near the edge of the detector or have low momentum, and focuses on jets with higher momentum that are more likely to exhibit significant physical effects. By applying these selection criteria, we ended up with a high-quality data set of jet events.

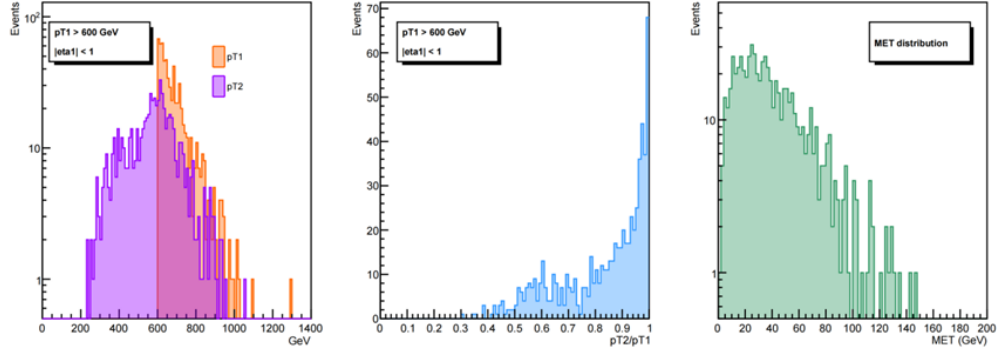


Fig. 1. This figure displays the distributions after applying event selection and resolution fitting. It includes the following distributions: the transverse momentum of the leading jet (p_{T1}), the transverse momentum of the subleading jet (p_{T2}), the ratio of the transverse momentum of the subleading jet to the leading jet (p_{T2}/p_{T1}), and the distribution of missing transverse energy (MET).

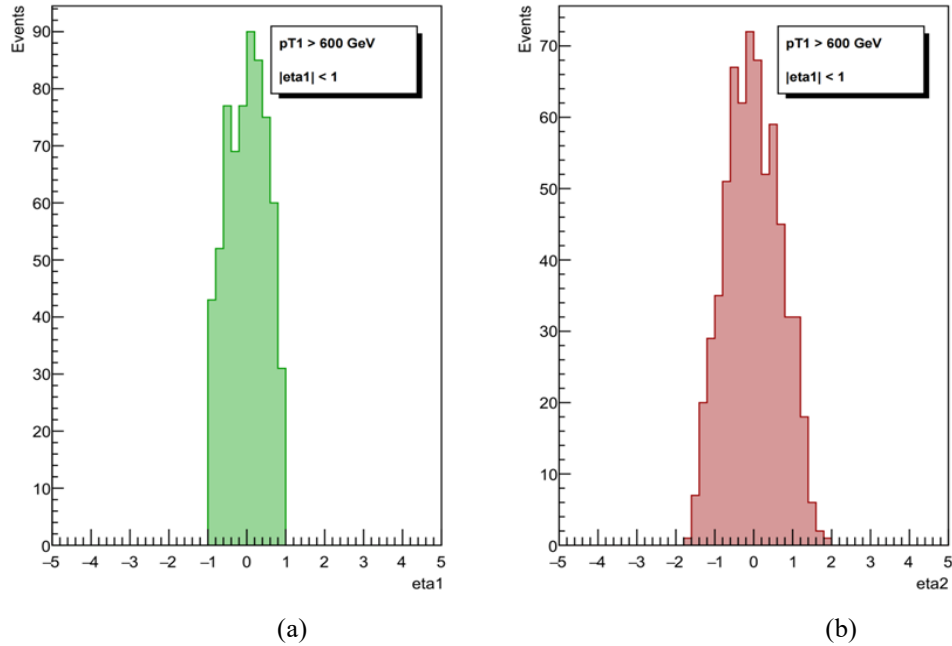


Fig. 2. This figure displays the distributions after applying event selection and resolution fitting, including: (a) the pseudorapidity of the leading jet (η_{a1}) and (b) the pseudorapidity of the subleading jet (η_{a2}).

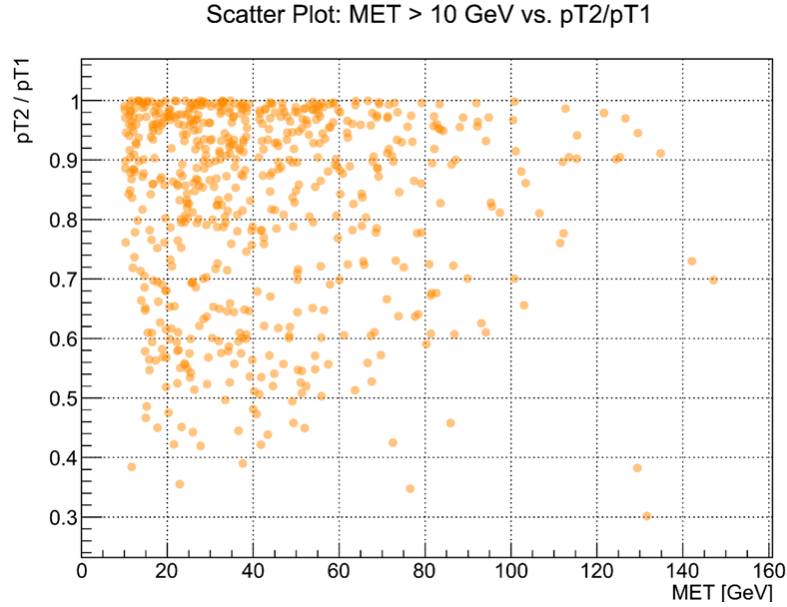
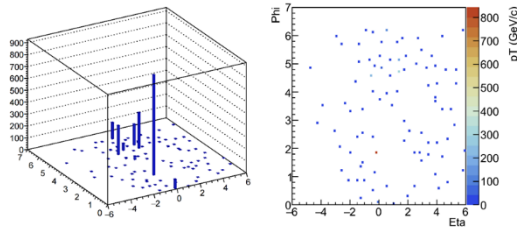


Fig. 3. This figure illustrates the relationship between the missing transverse energy (MET) and the ratio of the transverse momentum of the subleading jet (p_{T2}) to the leading jet (p_{T1}) for each event, under the condition that $MET > 10$ GeV.

3 Results

In cases where the subleading jet momentum ratio (p_{T2}/p_{T1}) is relatively small ($p_{T2}/p_{T1} < 0.4$), we observe that MET values are often lower ($MET < 40$ GeV). Smaller MET values might indicate a reduced contribution from W bosons or relatively minor contributions from neutrinos. Typically, larger MET values are associated with W boson decays producing leptons and neutrinos, with neutrinos leaving a significant missing energy signal because they are not directly detectable by the detector. Thus, we specifically focus on events with $MET < 40$ GeV and ($p_{T2}/p_{T1} < 0.4$), analyzing the distribution of the leading jet and subleading jet in pseudorapidity (η) and polar angle (θ). This analysis aims to further reveal jet characteristics and their relationship with MET, providing additional insights into jet behavior in high-energy collisions.



(a) $p_{T2}/p_{T1}=0.36$, $MET=14.51$ GeV

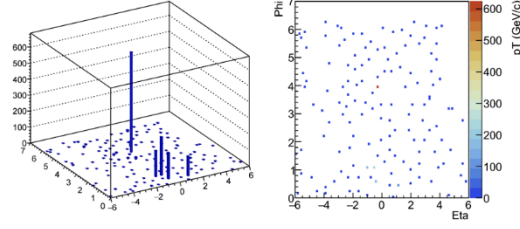
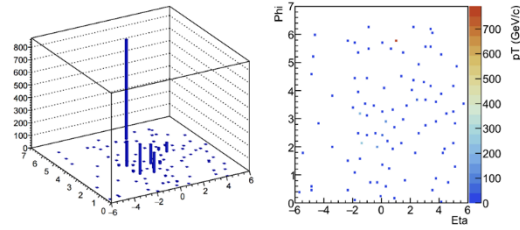
(b) $pt_2/pt_1=0.39$, $MET=34.52\text{GeV}$ (c) $pt_2/pt_1=0.30$, $MET=14.85\text{GeV}$

Fig. 4. This figure shows the distribution of the transverse momentum (pT) with respect to the pseudorapidity (η) and the polar angle (θ) for each event, under the conditions ($pT_2/pT_1 < 0.4$) and ($10 \text{ GeV} < MET < 40 \text{ GeV}$).

Upon examining Figure 4, we find that even in the absence of W boson decays, phenomena resembling jet quenching still occur. Further analysis indicates that both leading and subleading jets are predominantly distributed near pseudorapidity (η) close to 0. This observation might be influenced by the detector's angular coverage limitations. In this study, we assumed a detector coverage of ($|\eta| < 1$), which might lead to incomplete detection of jets in some events, thereby creating a pseudo jet quenching signal. In other words, these detector angle limitations could generate a false signal resembling jet quenching, rather than actual jet energy loss. So when a jet quenching-like phenomenon occurs in heavy ion collisions it is not necessarily energy loss, but could also be due to false signals caused by the detection range. This finding provides a useful background for studying QGP produced in heavy-ion collisions.

4 Conclusion

In this paper, we mainly simulate the PP collision under high energy, and discuss whether the PP collision produces a phenomenon similar to the Jet Quenching phenomenon in the heavy ion collision, and the conclusion is that the PP collision can produce a phenomenon similar to Jet Quenching due to the limited detection range of the detector. We can conclude from this that the judgment of the phenomenon of jet quenching cannot be judged only by leading jet and subleading jet, but we can try to determine whether jet quenching occurs based on the comparison of leading jet with other jets. This study provides an important background for the study of jet quenching phenomenon in heavy ion collisions.

Acknowledgement

Yize Mo and Keruo Zhang contributed equally to this work and should be considered co-first authors.

References

- [1] Apolinário, L.; Lee, Y.-J.; Winn, M. Heavy Quarks and Jets as Probes of the QGP. *Progress in Particle and Nuclear Physics* 2022, *127*, 103990. <https://doi.org/10.1016/j.pnpnp.2022.103990>.
- [2] Pasechnik, R.; Šumbera, M. Phenomenological Review on Quark–Gluon Plasma: Concepts vs. Observations. *Universe* 2017, *3* (1), 7. <https://doi.org/10.3390/universe3010007>.
- [3] Rapp, R.; Hees, H. V. Heavy Quarks in the Quark-Gluon Plasma. In *Quark-Gluon Plasma 4*; WORLD SCIENTIFIC, 2010; pp 111–206. https://doi.org/10.1142/9789814293297_0003.
- [4] Foka, P.; Janik, M. A. An Overview of Experimental Results from Ultra-Relativistic Heavy-Ion Collisions at the CERN LHC: Hard Probes. *Reviews in Physics* 2016, *1*, 172–194. <https://doi.org/10.1016/j.revip.2016.11.001>.
- [5] Ciesielski, R.; Goulianos, K. MBR Monte Carlo Simulation in PYTHIA8. arXiv August 6, 2012. <https://doi.org/10.48550/arXiv.1205.1446>.
- [6] CMS Collaboration. Search for Dark Matter and Large Extra Dimensions in Monojet Events in Pp Collisions at $\sqrt{s} = 7$ TeV. *J. High Energ. Phys.* 2012, *2012* (9), 94. [https://doi.org/10.1007/JHEP09\(2012\)094](https://doi.org/10.1007/JHEP09(2012)094).
- [7] ATLAS Collaboration. Jet Energy Measurement with the ATLAS Detector in Proton-Proton Collisions at $\sqrt{s} = 7$ TeV. *Eur. Phys. J. C* 2013, *73* (3), 2304. <https://doi.org/10.1140/epjc/s10052-013-2304-2>.
- [8] Cacciari, M.; Salam, G. P.; Soyez, G. FastJet User Manual: (For Version 3.0.2). *Eur. Phys. J. C* 2012, *72* (3), 1896. <https://doi.org/10.1140/epjc/s10052-012-1896-2>.
- [9] Cacciari, M.; Salam, G. P.; Soyez, G. The Anti-Kt Jet Clustering Algorithm. *J. High Energy Phys.* 2008, *2008* (04), 063. <https://doi.org/10.1088/1126-6708/2008/04/063>.
- [10] ATLAS Collaboration. Jet Energy Scale and Resolution Measured in Proton-Proton Collisions at $\sqrt{s} = 13$ TeV with the ATLAS Detector. *Eur. Phys. J. C* 2021, *81* (8), 689. <https://doi.org/10.1140/epjc/s10052-021-09402-3>.
- [11] Dasgupta, M.; Dreyer, F.; Salam, G. P.; Soyez, G. Small-Radius Jets to All Orders in QCD. *J. High Energ. Phys.* 2015, *2015* (4), 39. [https://doi.org/10.1007/JHEP04\(2015\)039](https://doi.org/10.1007/JHEP04(2015)039).
- [12] Marzani, S.; Soyez, G.; Spannowsky, M. *Looking Inside Jets: An Introduction to Jet Substructure and Boosted-Object Phenomenology*; Lecture Notes in Physics; Springer International Publishing: Cham, 2019; Vol. 958. <https://doi.org/10.1007/978-3-030-15709-8>.
- [13] Klein, J. Jet Physics with A Large Ion Collider Experiment at the Large Hadron Collider. <https://doi.org/10.11588/heidok.00017710>.

Proceedings of the 4th International Conference on Computing Innovation and Applied Physics
(CONF-CIAP 2025)

17-23 January 2025, Eskişehir, Turkey

CONF-CIAP 2025

Copyright © 2025 EAI, European Alliance for Innovation

www.eai.eu

www.confciap.org

ISBN: 978-1-63190-504-9

European Alliance for Innovation

EAI is a non-profit organization with free membership and the largest open professional society for advancing research careers through community collaboration and fair recognition. Members benefit from finding feedback and mentorship for their work and they are guaranteed to be evaluated fairly, transparently, and objectively through community.

ISBN: 978-1-63190-504-9

ISSN: 2593-7642

<http://eudl.eu/series/CORE> | www.eai.eu