Development of Teaching Demonstration Cases of Multi Threads Computation in Simulation Scenario

Pengfei Liu[1], Weifeng Zhang[1], Weiyuan Zhan[1], Yuqun Luo[1], Xiaxu He[2]*

* Corresponding author: poxxhe@scut.edu.cn

[1]College of Mathematics and Information South China Agricultural University Guangzhou, China

[2]Research Centre of Social Work South China University of Technology Guangzhou, China

Abstract—The multithreading programming technology in the visual interaction design course is a difficult point in the course teaching. Aiming at the problem of insufficient explanation of multithreading programming in the course teaching, this paper attempts to design a specific application scenario teaching demonstration. By introducing the concept of multithreading in the simulation model development process, and conducting demand analysis, designing synchronous processes and writing code; the key points in the development process of multithreading model are clearly discussed. It proved to promote the further application of multithreading programming technology in visual development.

Keywords- multithread programming; Web programming; Synchronization; Visualization

1 INTRODUCTION

Visual interaction analysis technology can generate visual effects through algorithms according to problems and user needs, provide clear, multi-dimensional and easy-to-use information, and provide support for users to achieve greater insight and decision-making power [1]. Data visualization mainly uses multimedia carriers such as charts, maps, web pages and videos to express complex and abstract data content in a graphical way consistent with human perception [2].

At present, the visual interaction design course we teach mainly takes the web application based on javaEE as the technical carrier, and follows the technical route of B/S framework. As we all know, Web applications work in the concurrent request mode by default. For example, servlets can receive multiple processing requests at the same time. At this time, the web container will create a thread for each request to process. If the servlet does not involve the problem of sharing resources, you do not need to care about multithreading. However, if the servlet needs to process data with competition, you must ensure that the servlet is thread safe, otherwise a series of unpredictable problems will occur, such as dirty data reading.

ITEI 2022, December 23-25, Harbin, People's Republic of China

DOI 10.4108/eai.23-12-2022.2329127

In the teaching process of JavaEE based web development, multithreaded programming is a difficult point. It is the key for beginners to become proficient, and it is also a necessary skill to develop a practical application. How to deeply understand the resource synchronization and mutual exclusion between multiple threads in web programming is a key point; However, the introduction of relevant knowledge points of the course is not too much, so students lack the concept and understanding of concurrent execution when learning web development.

The literature [3] aims at the insufficient explanation of the principle of multithreading programming in the Java programming course, combines some theories and teaching tools of the operating system, introduces the concept of thread, draws the state transition diagram, designs synchronous algorithms and other means to promote teaching. Literature [4] introduces a tool to provide visual support for Java concurrent programming teaching: displaying thread execution and synchronization between threads in the form of graphics. Literature [5] developed a multithread simulation code using Java to simulate the chemical reaction flow of ethanol.

In order to enhance the students' interest in the complex parallel model, this paper chooses the research on the propagation behavior of covid-19, which is a hot topic in the current society and research, as the simulation object. The spread of new coronavirus is characterized by extensive, rapid and normalized. It is necessary to establish an accurate simulation model in order to better guide the formulation of policies and guidelines such as social isolation time, ward bed setting and resumption of work. The existing covid-19 transmission simulation model [6] takes a single region as the simulation object, lacking a global view.

At present, there are many researches at home and abroad on the transmission and epidemic simulation of covid-19. Zhu Lianhua and others have built a staged transmission model of covid-19 pneumonia considering the impact of prevention and control measures in different periods based on the viral dynamic transmission mechanism [7]. De Kai and others established a model based on SEIR and Agent methods to predict the impact of wearing a universal mask on the spread of covid-19 [8]. Marco and others designed a simulation model based on the agent to infer the virus prevention strategy of public buildings in the "second stage" of the covid-19 [9]. Li Dong and others built a new virus transmission simulation combination model SISR-HHO to effectively simulate the transmission process of covid-19 [10]. Kiesha adopted the SEIR model to simulate the impact of the control strategy to reduce social integration on the epidemic results of covid-19 in Wuhan [11]. Bossert and others helped South Africa to introduce a containment plan for covid-19 based on agent simulation method in terms of socio-economic and traffic survey data [12]. Ahlbrecht and others proposed an extensible multi-agent parallel simulation method based on MapReduce [13]. The above research development trends can be summarized as mathematical equation method and agent simulation method, which will co-exist and complement each other for a long time, while agent simulation method will develop towards fine-grained and parallel.

With reference to the above teaching research results, in order to solve the existing multi-threaded teaching problems vividly, this paper develops a parallel simulation system of covid-19 propagation behavior based on multi-threaded, which is used to visually display multi-threaded programming and its operation. Among them, the system is mainly divided into three basic functional modules: (1) Data reading and output module. The basic operation parameters can be read from the file and the operation results can be written to the file. (2) Behavior

simulation sub module. Based on the multithreading technology, the transmission behavior of the new coronavirus is simulated from the perspective of multi evolutionary unit collaboration, and a parallel simulation model of the transmission behavior of the new coronavirus based on the mainland of China is constructed. The model is more refined, and the conclusions obtained are closer to the real situation, providing scientific basis for decision-making. (3) Data display sub module. The simulation data is displayed in graphical form in real time during operation.

Figure 1 shows the development process of the parallel simulation model of COVID-19 transmission behavior, including discussion, data collection, development, parameter adjustment, etc.

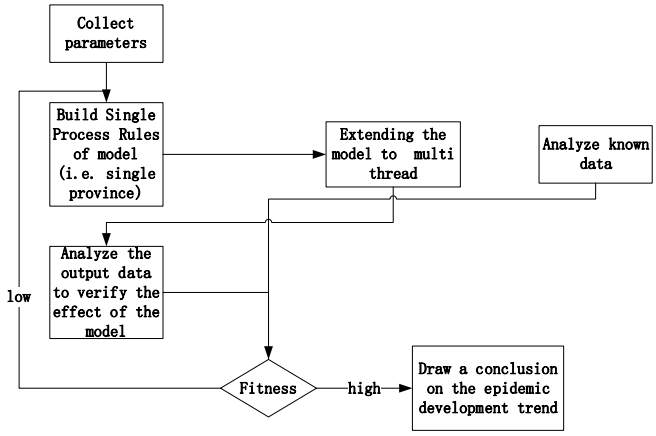

Figure 1. Development process of parallel simulation model for covid-19 transmission behavior

The paper clearly discusses the detailed development process of multithreaded programming, and promotes students' understanding of multithreaded programming technology with graphical operation mode.

2 REQUIREMENT ANALYSIS OF PARALLEL EVOLUTIONARY MODEL

Based on the existing models and research results on the propagation simulation of covid-19, and in combination with China's national conditions, we propose the main demand analysis of the nationwide parallel simulation model for the propagation of covid-19.

The number of evolution units are set in line with China's national conditions: China has a vast territory, and each province has its own relatively independent specific epidemic prevention measures. Therefore, the simulation cannot only consider several major provinces or cities. The current scheme comprehensively considers the computational complexity, modeling complexity, and prediction accuracy, so each province is simulated as an evolution unit.

The simulation parameters are set that are consistent with the spread of viruses: the simulation model that can output accurately needs to input parameters that are consistent with the actual

situation, such as consulting medical literature to obtain medical parameters, consulting statistical yearbooks and other data to obtain civil affairs parameters such as provinces and populations. A few regions cannot obtain data, but can only use the mean value. Read the medical literature to understand the contact rate, infection rate, proportion of the recovery rate of infected people, course of disease (reciprocal of the infection rate), ratio of asymptomatic people, number of initial infected people, incubation period, removal rate, hospital admission response time, hospital beds and other specific medical parameters, as well as the population mobility rate, infection risk level, total population, estimated number of returnees between different provinces of the country.

Evolutionary function modularization based on multithreading, as shown in Figure 2, the main thread and the slave threads are divided into independent modules according to business logic, and the modules call and exchange data through interfaces.

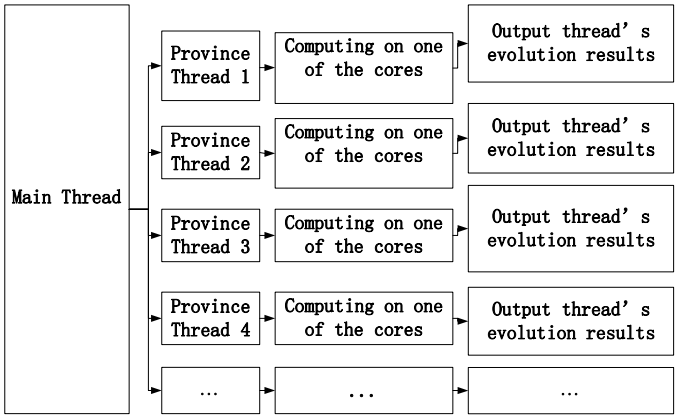

Figure 2. Parallel evolutionary architecture based on multithreading

At present, the following factors have been added. Cure rate: Through searching data, the average cure rate of new crowns in China is 98.9%.

Epidemic infection mortality: through searching data, the average mortality of China's new crown is FATALITY_ RATE is 2.1%.

Vaccine coverage: temporarily replaced by temporary data, set to 70%.

Probability of infection caused by frozen food: provisional data, set to 0.1%.

The ratio of asymptomatic personnel: no relevant data is found temporarily, which is temporarily set as 2%. Detection rate of asymptomatic patients: no relevant data is found temporarily, temporarily set as 10%. Handling asymptomatic persons: When dealing with infection, use the same method as judging death to judge whether the infected person is asymptomatic. If an infected person is infected, he or she will not participate in the incubation period timing, that is, he or she will be treated by constantly judging whether he or she will be detected, rather than conducting incubation period timing and then exposure treatment. Infection rate of foreign imported personnel: no relevant data is found temporarily, which is temporarily set as 2%.

Factors to be added: rate of asymptomatic patients, detection rate of asymptomatic patients, infection rate of foreign imported personnel, and virus mutation factors.

Figure 3 shows the flow chart of the parallel simulation model based on multithreading.

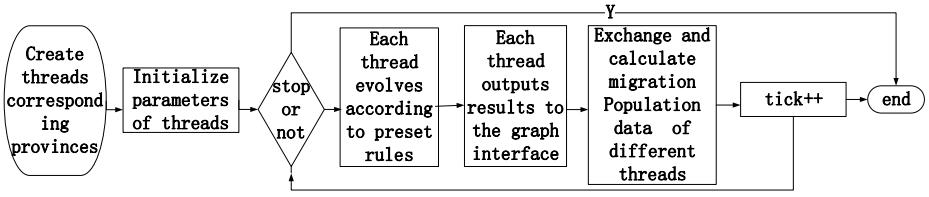

Figure 3. Running process of parallel simulation model based on multithreading

3 RULE SETTING SUPPORTING PARALLEL EVOLUTION

We design fine-grained simulation rules that support multithreading parallel evolution. The threads corresponding to each province perform the following initialization operations: First, initializing the agent corresponding to the number of hospitals and the population (the current ratio is 1 to 10000) in the system according to the population and the number of hospitals in each province and a certain ratio; Initializing and maintaining a probabilistic imported frozen food queue; Initializing and maintaining a probabilistic queue of returnees; Initializing and maintaining a probabilistic provincial floating queue. Initializing and maintaining a probabilistic provincial permanent population queue. The threads corresponding to each province should follow the progress of time, and perform probabilistic operations on the elements of these queues. Some queues of different threads have the possibility of data exchange, and must strictly perform synchronous operations.

We try to convert all the factors related to the epidemic situation into the rules of the simulation model. However, many rules require specific parameters, which are often difficult to obtain. Therefore, many rules cannot be implemented temporarily. The main rules that have been implemented are as follows.

Treatment rules after personnel are cured: the cured patients will be removed from the hospital and the beds will be vacated. According to the existing data on the Internet, the probability of re infection after cure is very small, almost zero, so this simulation model sets that cured people will not be re infected.

Treatment rules for asymptomatic patients: asymptomatic patients will not be subject to latency timing exposure rules, and will be treated solely by constantly judging whether they will be detected. Once asymptomatic patients are detected, they will be isolated immediately.

Processing rules for imported patients abroad: reduce the number of empty beds in the current area.

Rules for the impact of vaccine on the infection rate: According to the review, the infection rate decreased to 0.2% after the injection of vaccine, so first judge whether the person is vaccinated according to the vaccine coverage.

Processing rules for death cases and cure cases: by setting an array interval, generate a number x for each patient, and then judge the number to determine whether the patient is dead. Each patient can only judge once. If it is judged as death, a death cycle will be randomly generated according to the normal distribution function of the death cycle. When the diagnosis time reaches the death cycle, the patient will die. If it is judged that there is no death, the patient will continue to live. The cure will have a cure cycle, which will be timed from the beginning of the visit, and then a cure cycle will be randomly generated according to the normal distribution function of the cure cycle to judge whether the patient is cured.

Rules for handling frozen food infection: randomly generate a number to judge whether a healthy person will be infected by frozen food according to whether it falls in the infection range, so as to simulate reality.

Processing rules for returned personnel: randomly generate a number to judge whether the personnel imported from abroad will be infected according to whether they fall within the infection range. If they are infected, use the function addition point to represent the person to be isolated, and then find an empty bed for them to conduct isolation operation (those who are not infected will return to normal personnel) to simulate reality.

4 DEVELOPMENT RESULTS

Through the introduction of vaccines, imported frozen food, returnees from abroad, virus mutation and other key factors, as well as considering the mobility of personnel among provinces, a dynamic epidemic transmission model was built. The operation interface of the model can visually display the current domestic epidemic situation, as shown in Figure 4 (Note: this map of China is only an example, limited to Java GUI programming, and it is difficult to accurately draw a real map in a short period of time. Therefore, the actual situation is subject to the map published by the Chinese government).

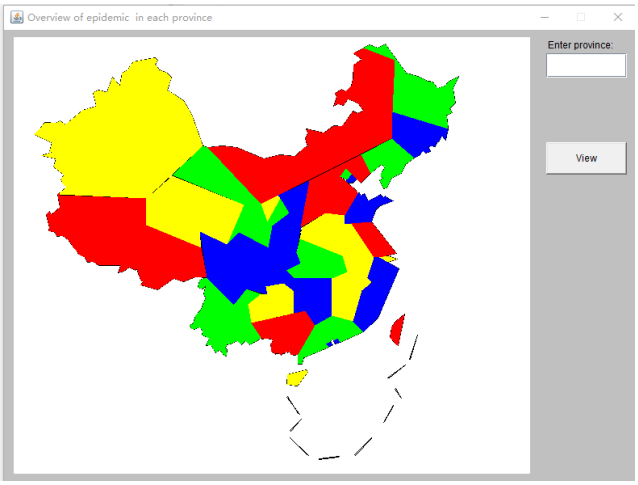

Figure 4. Overview of national epidemic situation

If we want to switch between different provinces and view the propagation simulation of that province, we can click the running icon of this simulation program in the task bar to select the simulation interface corresponding to the thread we want to view from the pop-up list of multiple different threads (each thread corresponds to a province). At the same time, it can show the specific running information of a province, which is shown in Figure 5.

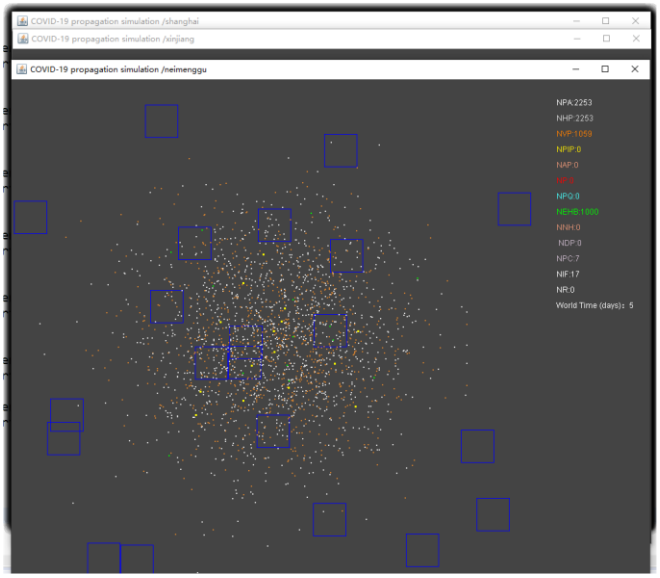

Figure 5. View of the epidemic situation details page of a province

In the Figure 5, the NPA represents the Number of people alive, NHP represents number of Healthy population, NVP represents Number of vaccinated people, NPIP represents Number of people in incubation period, NAP represents Number of asymptomatic people, NP represents Number of patients, NPQ represents Number of people quarantined, NEHB represents number of empty hospital beds, NNH represents number of badly needed hospital beds, NDP represents Number of dead patients, NPC represents Number of people cured, NIF represents Number of imported frozen food, NR represents Number of returnees. White dots represent health domestic residents without vaccine. Golden Dot represent Vaccinated Health domestic residents. Green dots represent cured infected persons; Red dot represent confirmed infected person. Black dot represents dead infected person. Dark Orange Dots represent Infected Persons in the Latent Period, Pink dots represent asymptomatic patients, Dark blue dot represent isolated patient, yellow square represent imported frozen food.

Admittedly, under the basic premise of frequent logistics and people flow exchanges between countries, the virus mutates rapidly, and many countries have canceled any anti epidemic measures. It has become particularly difficult to accurately predict the domestic epidemic situation. The current prediction effect of this model needs to be improved. In the future, the model will continue to be improved by refining the evolution unit, adding simulation rules and other methods.

The core of the case teaching method applied in this study is to stimulate students' interest with cases, and then stimulate their desire to learn, imitate and improve. By importing the existing single thread model and decomposing the key parts of multithreads programming case into several simple programming steps, and then analyzing them in detail, students can easily learn the practical knowledge of Java multithreading programming in actual cases. Many students also reflect that this learning method has a deeper memory of knowledge points than the ordinary learning method. In the process of developing, debugging models, and analyzing and discussing code, students gradually realized that the traditional variable definition and use methods would have thread safety problems under multithreaded concurrent access, and thus came into contact with the concepts of "ThreadLocal", "InheritableThreadLocal", and "ThreadLocalMap", and understood the way of safe value transfer between threads in a multithreaded environment, so as to understand the core of multithreaded programming, Finally, it is applied to the system programming of visual interaction design.

5 SUMMARY

At present, with the rapid development of big data visualization, cloud computing and artificial intelligence, the requirements for Java programming technology of developers are constantly improving. This paper explores the use of multithreading in Java to develop simulation models, which improves students' understanding of multithreading programming in more detail.

This program collects and integrates the transmission and treatment parameter data of the domestic covid-19, simulates the transmission behavior of the new coronavirus from the perspective of multi evolutionary unit cooperation based on multi-threading technology, expands the existing single simulation unit model to the parallel simulation model of multiple simulation units, and refines its parameter conditions and rules by referring to the existing covid-19 transmission simulation model; We make the model more refined. For example, a basic evolution unit uses a computing core of a server for simulation and evolution, so as to get calculating conclusions faster; Compared with the differential equation method, the specific continuous change process can be observed.

We have improved the fine-grained nature of the model: the factors that affect the spread of new coronal pneumonia are considered. Multiple evolutionary units co evolve. Each evolutionary unit should set a certain number of agents according to the actual population of the area corresponding to the unit, so as to improve the evolution accuracy. We have improved the parallelism of the evolutionary model: decompose the large amount of computation of fine-grained model, improve the running speed, and ensure the accuracy and computing efficiency.

The paper clearly discusses the key points in the development process of multithreaded computing, so as to promote the further application of multithreaded programming technology in visual development. With the increasing number of CPU cores, parallel computing based on multithreaded programming will be used more and more widely. Therefore, the discussion in this paper is more meaningful.

Acknowledgment. This research is financially supported by "13th Five-Year" planning of Guangdong philosophy and Social Sciences Planning Project (GD19YSH03), Ideological and political education in colleges and universities of Guangdong Province in 2020

(2020GXSZ004), Innovation and entrepreneurship training program for college students of South China Agricultural University (X202210564180, X202210564169), 2019 provincial quality engineering project of Characteristic Specialty on information and computing science characteristic, China Agricultural Science and Education Fund Project (NKJ202103057).

REFERENCES

[1] ZHANG Yuyu. "Educational Application and Influence of Visual Analysis Technology." Renmin University of China Education Journal,2022,3:166-180.

[2] Gao Li. "Discussion on teaching reform of data visualization course under the background of big data." Wireless Internet Technology.2022,6:160-162.

[3] Li Qing, Wang Jiangping, Zhu Jie. "The Teaching Research of Java Multithread Programming. " Computer Education, 2015(12):4.

[4] Bi, Yaodong, and John Beidler. "A visual tool for teaching multithreading in Java." Journal of Computing Sciences in Colleges 22.6 (2007): 156-163.

[5] Yamada E, Shimada T, Hayashi A K. "Development of Java multi-threaded simulation for chemical reacting flow of ethanol. " Advances in Engineering Software, 2012, 54(none):17-23.

[6] https://github.com/KikiLetGo/VirusBroadcast.

[7] Zhu Lianhua, Tan Yan, Xiao Huiwen, et al. "Quantitative assessment and analysis on the prevention-control measures of COVID-19 based on staged transmission model." Journal of Nanjing University of Information Science and Technology (Natural Science Edition), 2020, 12(3): 364-372.

[8] Kai, De, et al. "Universal masking is urgent in the COVID-19 pandemic: SEIR and agent based models, empirical validation, policy recommendations." arXiv preprint arXiv:2004.13553 (2020)

[9] D'Orazio, Marco, Gabriele Bernardini, and Enrico Quagliarini. "How to restart? An agent-based simulation model towards the definition of strategies for COVID-19" second phase" in public buildings." arXiv preprint arXiv:2004.12927 (2020).

[10] Li Dong, Bai Ruirui. "Simulation Composite Model for COVID-19 Propagation. " Statistics & Decision,2020,36(5):5-10.

[11] Prem, Kiesha, et al. "Age-structured SEIR model for COVID-19 outbreak in Wuhan, China." (2020).

[12] Bossert, Andreas, et al. "Limited containment options of COVID-19 outbreak revealed by regional agent-based simulations for South Africa." arXiv preprint arXiv:2004.05513 (2020)

[13] Ahlbrecht, Tobias, Jürgen Dix, and Niklas Fiekas. "Scalable multi-agent simulation based on MapReduce." Multi-Agent Systems and Agreement Technologies. Springer, Cham, 2016. 364-371.