Analysis on Improving the Response Time with PID-SARSA-RAL in ClowdFlows Mining Platform

N. Yuvaraj[1,*], R. Arshath Raja[2] , Dr.V.Ganesan[3] , Dr.C.Suresh Gnana Dhas[4]

[1] Research Scholar, Department of Computer Science and Engineering, St. Peter's Institute of Higher Education and Research, St. Peter's University, Chennai, Tamilnadu, India - 600054

[2] Research Scholar, Department of Electronics and Communication Engineering, BS Abdur Rahman Crescent University, Chennai, Tamilnadu, India- 600048.

[3] Director, R&D, Innovative Science and Technology Publications, Chennai, Tamilnadu, India-600088.

[4] Professor & Head, Department of CSE, Vivekanandha College of Engineering for Women, Elayampalayam, Namakkal, Tamilnadu, India – 637205.

E-Mail: yraj1989@gmail.com, arshathraja.ru@gmail.com, ganesan1982@gmail.com, sureshc.me@gmail.com

Abstract

This paper provides an improved parallel data processing in Big Data mining using ClowdFlows platform. The big data processing involves an improvement in Proportional Integral Derivative (PID) controller using Reinforcement Adaptive Learning (RAL). The Reinforcement Adaptive Learning involves the use of Actor-critic State–action–reward–state–action (SARSA) learning that suits well the stream mining module of ClowdFlows platform. The study concentrates on batch mode processing in Big Data mining model with the use of proposed PID-SARSA-RAL. The experimental evaluation with the conventional ClowdFlows platform proved the effectiveness of the proposed method over continuous parallel workflow execution.

Keywords: SARSA Active Learning, Big Data Mining, PID Controller, Reinforcement Learning

1. Introduction

ClowdFlows is an open cloud platform that includes composition, execution and interactive sharing of data mining workflows. The ClowdFlows platform works under the principle of service based knowledge discovery in large databases with an interactive workflows. The ClowdFlows platform has a user programming interface, which provides access to 3rd party services. It also uses the workflow execution in the cloud environment over static data [1].

The ClowdFlows platform is improved to operate on real-time data. Here, the improvements are made to access different operating systems and can utilize the utmost resource of the server to reduce the computational complexity with increased data transfer rate [2].

ClowdFlows platform using batch mode processing to perform distributed computing with MapReduce framework [3].

The main problem associated with data mining in large database is the data scalability. Hence, active or semi-supervised learning approach is suitable for better data selection strategy [4] in big data mining systems. Similarly, the use of active learning in ClowdFlows platform increases the parallel processing of data selection. Kranjc et al. [5] used on-line dynamic adaptive analysis in ClowdFlows platform that reduces the computational efforts. The active learning is improved with Support Vector Machine on microblogging data streams [5].

The active learning approaches to improve the data selection in big data sets using ClowdFlows platform is currently less. Henceforth, the proposed method

*Corresponding author. Email: yraj1989@gmail.com

concentrated on a feedback looping system to improve the active learning strategy in large datasets. The use of PID controller in data mining systems can be efficient in selecting the data from big data environments. A feedforward active learning system is used to update the PID controller with learning rules [6], [8] and inverse learning model [7] for improving its adaptation in a particular environment. The adaptive learning of PID controller is improved with reinforcement learning method that considers the input from state convertor [9] and Carlucho et al. [10] used Q-learning strategy. Here in [9], the feedback is taken from the input and in [10], it is taken from the output. In [10], the temporal memory is used to improve the active learning within limited subspaces.

The proposed technique uses reinforcement adaptive learning using SARSA [11] learning model that has an improved rewards per trial [12] than Q-learning strategy. Hence, the use of SARSA learning model as a feedback to the PID controller will improve the adaptive rate of the controller in an environment. Further, the PID-SARSA-RAL technique is analysed experimentally over data selection strategy over large data using ClowdFlows platform.

The main sections of the paper is shown here: Section II involves the proposed reinforcement adaptive learning in PID controller. Section III evaluates the proposed method with the conventional system. Finally, Section IV concludes the paper with future work.

2. Reinforcement Learning with PID

The Actor–critic model uses Temporal Detection (TD) method with discrete memory architecture that represents direct policy independence over value function. The policy structure is considered as actor, which is helpful in selecting the actions. The estimated value function or critic is used to criticize the actor's action. Here, the active learning is treated as on-policy, where the critic learns the system with TD and critique the current policy for data selection followed by the actor or ClowdFlows platform.

The PID controller [11] is a popular method used in different environments because of its fast and efficient computation. This low-level control procedure strongly improves depends on the parametric setup for its improved performance. The adaptive PID controller is considered as a technology specific system designed to study the behavior of a particular environment with its tuning parameters.

PID control Parameter:

The worker node in the CloudFlows platform [3] implements the SARSA learning with feedback from the PID controller. The discrete-time incremental PID

calculated using backward differentiation [13] is selected to process the control variables from the SARSA algorithm. Since, automation is an important concern that improves the data selection in big datasets, the following Eq.(1) is adopted.

$$\Delta u(t) = K_p \left[e(t) - e(t-1) \right] + \frac{K_p h}{T_i} e(t) + \\ \frac{K_p T_d}{T_i} \left[e_f(t) - 2e_f(t-1) + e_f(t-2) \right] \quad (1)$$

and the integral action of PID is given as:

$$u(t) = u(t\text{-}1) + \Delta u(t) \quad (2)$$

The output of PID controller is given as an input to the actor of SARSA algorithm. The proposed SARSA actor-critic model is implemented over PID controller, which is shown in Fig. 1.
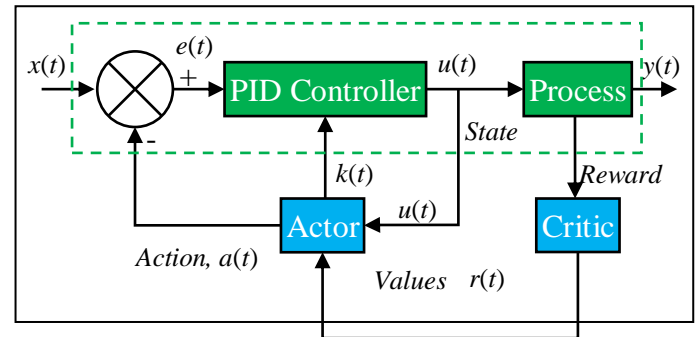

Figure. 1. PID-SARSA Reinforcement Learning

SARSA Learning Model:

The major difference between the Q-learning and SARSA model is that the update policy in Q-learning uses $Q(s,a) = r(s) + \alpha \times \max(Q(s'))$ and for SARSA it is $Q(s,a) = r(s) + \alpha \times Q(s',a'))$. During the process of active learning by these two algorithms, the SARSA includes control policy but Q-learning avoids it. Hence, the SARSA algorithm stores the information before the updating the action values. This makes the SARSA algorithm to attain its target value with less intervals or frequency than the Q-learning algorithm.

SARSA algorithm uses reinforcement learning to study the Markovian decision policy. The update for Q-values depends totally on the current state $u_1(t)$ with action $a_1(t)$ and it attains the reward $r(t)$. The next state is $u_2(t)$ or $u(t+1)$ with next action $a_2(t)$ or $a(t+1)$. Therefore, the quintuple for the SARSA algorithm is $Q(u_1(t), a_1(t), r(t), u_2(t), a_2(t+1))$.

The selection steps of two actions are needed to finds the successive states' action pair in addition with the first pair. The learning rate (α) and discount factor (γ) parameter of SARSA is same as that of Q-learning. With this, the algorithm for the adaptive learning of SARSA using PID is defined.

Algorithm 1: PID-SARSA Adaptive Learning Algorithm

Initialize $Q(u_0(t), a_0(t))$
Repeat (For each instant)
  Initialize $u(t)$
  Choose $a(t)$ from $u(t)$ through the policy from Q
   Consider ε as greedy
  Repeat (For each instant)
   Take action over $a(t)$ and observe $r(t)$, $u(t+1)$
   Consider $a(t+1)$ from $u(t+1)$ through the policy from Q
    Consider ε as greedy
    $Q(u(t), a(t)) \leftarrow Q(u(t), a(t)) + \alpha[r(t) + \gamma Q(u(t+1), a(t+1)) - Q(u(t), a(t))]$
   $u(t) \leftarrow u(t+1)$; $a(t) \leftarrow a(t+1)$; $r(t) \leftarrow f(y(t))$
  Repeat until $u(t)$ terminates

Here, the values of $\alpha = 0.2$, $\gamma = 0.9$ and $\varepsilon = 1$ is considered for the operation. The output of the PID controller $u(t)$ is considered as a state function to the SARSA algorithm. Henceforth, the operation continues as a feedback operator. Instead of direct feedback from the process output, the proposed method uses action variable to find the error difference on PID controller. Finally, the proposed controller design is modified using $k(t)$, which is given as a feedback input to the PID by the actor. This is represented as:

$$u(t) = \Delta u(t) \times k(t) \qquad (3)$$

The data preparation uses snowflake storage schema for training the agents' upto 15,000 episodes having 500 time window over 10 iterations [14]. The selection of features are done through Mutual Information Maximization and the Adaptive Genetic Algorithm (MIMAGA) [15].

Finally, the selection of action sequence uses similar procedure for measuring the impact of proposed method over data mining approach. Two parameters are used for evaluation. The former one is a support measure and the later one is confidence measure.

$$\% support = [c(a(t)) / c(t_n)] \times 100 \qquad (4)$$

Support measure defines the rate of usage of $a(t)$. Here, c represents the count, $c(a(t))$ represents the actions performed by the agent and $c(t_n)$ represents the total number of actions performed by the agent.

$\% confidence\ [a(t) \Rightarrow u(t)] =$

$$[c(a(t) \cup u(t)) / c(a(t))] \times 100 \qquad (5)$$

Confidence measure defines the success of the action $a(t)$.

where, $c(a(t) \cup u(t))$ represents the actions happened for each mining operation, where the agent attains success. $c(a(t))$ represents the actions performed by the agent irrespective of its success.

3. Experimental Analysis

The CloudFlow is evaluated with a set of 10 nodes that possess 3.60 GHz Intel i7-6567U CPU loaded with 64-bit Ubuntu 16.04.2 LTS over 16GB memory. The Virtual Machines runs over 64-bit Fedora 25work station with 4GB of memory. The system is exercised with the Phoronix Test Suite 7.0. The data mining uses confidence and support measures to assess the quality and impact of the data retrieved from large databases. The core objective is to extract or retrieve the relevant data from large database. The results of the SARSA algorithm with PID controller reduced the response time to settle at the set point value than the conventional PID's as shown in fig.2.

This resulted in improved response to the given query at the user end. The proposed PID controller is used for measuring the success rate of the feature selection. The selection of features are purely based on inputs from the SARSA algorithm. The error variable in the PID controller is reduced considerably, which results in increased response time in finding the samples or the datasets in the big data. The result of the success rate is seen in fig.3.
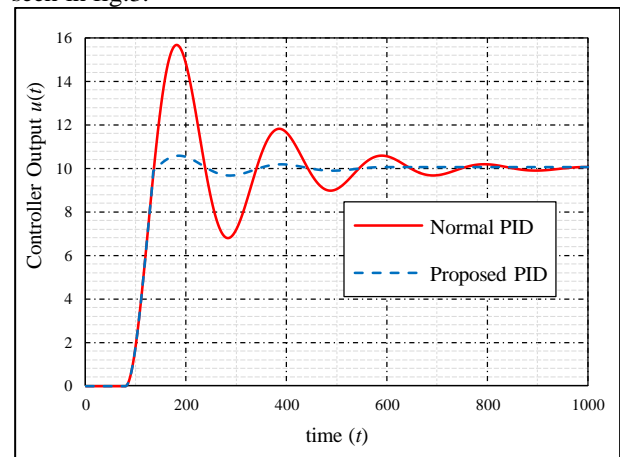

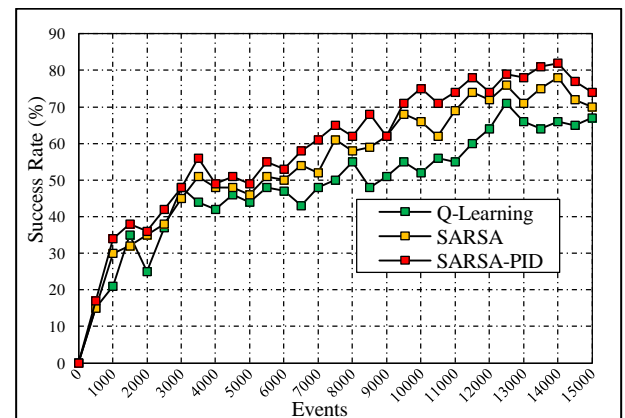**Figure. 2.** Comparative analysis of PID Controllers

Figure. 3. The success rate of PID Controllers for the selection of Feature
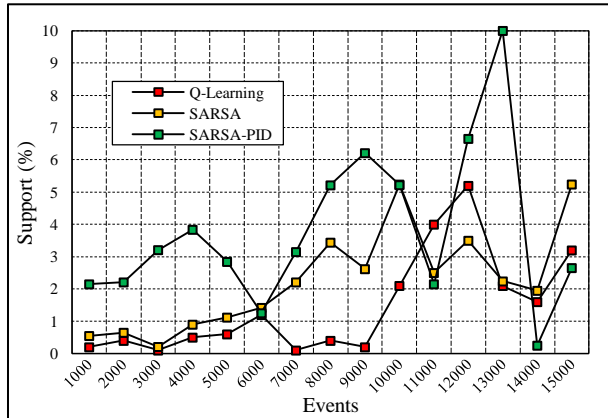
Figure. 4. The Support rate

The support rate (without PID and SARSA as input) recorded gets reduced with increased actions or agents in the data space (fig.4.). However, we tried measurements on the confidence measures (fig.5.) and we found that values are too low that reduced the success rate of the SARSA algorithm. However, in contrary, improved response time or reduced delay with PID controller, the support and confidence rate improved.

Here, the action of searching the unwanted datasets are avoided with the use of MIMAGA algorithm and the updating of the correct value to the end user is done at faster time with the use of PID controller. This can be justified in terms of the results obtained from the success and confidence rate.
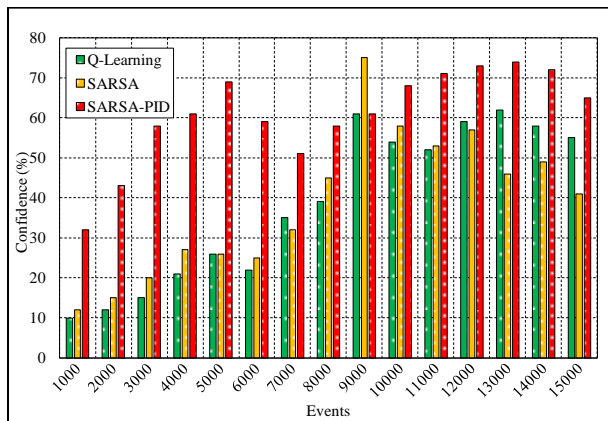

Figure. 5. The Confidence rate

4. Conclusions

This paper improves the design of PID controller by improving the response time. The PID feedback system is integrated with SARSA actor-critic model. The integration improves the success rate of feature selection and improves well the user response time in big data domain using CloudFlows platform. The evaluation in CloudFlows platform reduces the response time using the proposed method. This method could be further

implemented over other big data platforms to improve the data extraction process with improved response time.

References

[1] Janez Kranjc, Vid Podpecan and Nada Lavrac, "ClowdFlows: A Cloud Based Scientific Workflow Platform", Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp 816-819, 2012.

[2] Janez Kranjc and Nada Lavrac, "Real-Time Data Analysis in ClowdFlows", IEEE Conference on Big Data, pp. 15-22, 2012.

[3] Hiroshi Mamitsuka and Naoki Abe, "Efficient Data Mining by Active Learning", Progress in Discovery Science, pp. 258-267, 2002.

[4] Roman Orac et.al, "ClowdFlows: Online Workflows for Distributed Big Data Mining", Future Generation Computer Systems, pp. 1-38, 2016.

[5] Janez Kranjc et.al, "Active Learning for Sentiment Analysis on Data Streams: Methodology and Workflow Implementation in the ClowdFlows Platform", Information Processing and Management, Vol. 51, No. 2, pp. 187-203, 2015.

[6] Seung-Min Baek and Tae-Yong Kuc, "An Adaptive PID Learning Control of DC Motors", IEEE International Conference on Computational Cybernetics and Simulation Systems, Man, and Cybernetics, pp. 2877-2882, 1997.

[7] Tae-Yong Kuc and Woong-Gie Han, "Adaptive PID Learning of Periodic Robot Motion", Proceedings of the 37th IEEE Conference on Decision & Control, pp. 186-191, 1998.

[8] Tae-Yong Kuc and Woong-Gie Han, "An adaptive PID learning control of robot manipulators", Automatica, Vol. 36, pp. 717-725, 2000.

[9] Wang Xue-song, Chen Yu-hu and Sun Wei, "A Proposal of Adaptive PID Controller Based on Reinforcement Learning", Journal of China University of Mining & Technology, Vol. 17, No. 1, pp. 1-5, 2007.

[10] Ignacio Carlucho, Mariano De Paula, Sebastian Villar and Gerardo Acosta, "Incremental Q-Learning Strategy for Adaptive PID Control of Mobile Robots", Expert Systems with Applications, 2017.

[11] Csaba Szepesvari, "Algorithms for Reinforcement Learning", Synthesis Lectures on Artificial Intelligence and Machine Learning series, pp. 3-78, 2009.

[12] Richard S. Sutton and Andrew G. Barto, "Reinforcement Learning: An Introduction", MIT Press, 2012.

[13] Discretizing a PID controller, Available at: www.kirp.chtf.stuba.sk/moodle/mod/resource/view.php?id=30735

[14] Davi C. de L. Vieira, Paulo J. L. Adeodato and Paulo M. Goncalves, "Improving Reinforcement Learning Algorithms by the Use of Data Mining Techniques for Feature and Action Selection", IEEE International Conference on Systems Man and Cybernetics, pp. 1863-1870, 2010.

[15] Huijuan Lu, Junying Chen, Ke Yan, QunJin , Yu Xue and Zhigang Gao, "A Hybrid Feature Selection Algorithm for Gene Expression Data Classification", Neurocomputing, pp. 1-29, 2016.