Cloud Based Hybrid Evolution Algorithm for NP-Complete Pattern in Nurse Scheduling Problem

Ming-Shen Jian and Ming-Sian You

***Abstract*—In this paper, the Cloud based Hybrid evolution algorithm for NP-Complete Pattern in Nurse Scheduling Problem (NSP) is proposed as the Software Computing as a Service (SCaaS). Due to low birth rate, the human resource becomes the limited resource for job assignment. To find the optimal solution for staff scheduling becomes an important issue. The proposed system follows the definition of NSP and recognizes the possible problem of NP-Complete Pattern. Only the pattern is recognized as the NSP optimal problem, the proposed system can find the optimal solution. Then, the different types of evolutionary algorithm in evolution steps are integrated. Based on the proposed Feedback Assistance method, the suitable evolution steps of evolutionary algorithm can be dynamically decided and executed. Similar to the Tasktracker and Jobtracker in cloud, all the computing load can be divided and distributed. The simulation results show that the proposed hybrid evolution algorithm can find the optimal solution with about 50% less evolution generations.**

***Index Terms*—Evolution (genetic) algorithm, nurse scheduling problems, cloud, feedback, hybrid, modular, NP-complete.**

I. Introduction

Due to low birth rate, the human resource becomes the limited resource for job assignment. To find the optimal solution for staff scheduling becomes an important issue. However, to allocate the schedule for different staff, some limitations such as the total workload, total work time, etc. should be considered. The NSP problem pattern includes the hard constraints which are limited by law and soft constraints which depend on each staff or nurse. Different limitations, objects, and requirements from different NP-complete problems will lead to different optimal solutions. Therefore, how to recognize the pattern as the NSP problem pattern is the first important issue.

In the past, many researches already defined that the staff scheduling or nurse scheduling problem is an NP-complete problem [1]-[3] and tried to find the optimal solution [4]-[12]. Under the assumption $P \neq NP$, there exists no exact algorithm that can solve this problem in polynomial time. Some researches proposed to find the optimal solution based on mathematical programming method, linear programming, nonlinear programming, multi-objective programming, dynamic programming, parameter planning, heuristic algorithm, hill-climbing algorithm, simulated annealing algorithm, etc. [13]-[18].

Genetic algorithms (GAs) were first proposed by Holland in 1975 based on natural evolution selection – "The fittest survive". GAs can search widespread function spaces efficiently. In addition, GAs are flexible to solve various problems if these problems can be given evaluating functions and gene-types of situation instances. GAs, today, are used in many fields for solving optimal solution such as classic NP-complete problem, traveling salesman problem (TSP) [19], [20], job-shop schedule problem (JSP) [21], [22], character recognition, state space learning, and tensor polynomials. According to the solution in each generation of the algorithm, evolution algorithms or genetic algorithms try to find the better solution generation by generation until the optimal solution is found. Since the evolution algorithm can help system users find the optimal solution, to obtain the optimal solution quickly and to enhance the performance of the simple genetic algorithm (GA) or evolution algorithm become an active research.

Since different limitations, objects, and requirements from different NP-complete problems will lead to different optimal solutions, to select the suitable crossover method, and selection method, the genetic algorithm (GA) will also affect the system performance.

To enhance the computing performance, in this paper, the cloud based system is proposed. According to the structure of cloud computing, users' application can be established individually on virtual machines. All the computing workload of the evolution (genetic) algorithm is implemented in cloud. Similar to the Tasktracker and Jobtracker in cloud, all the computing workload can be divided into several partitions and executed or computed in different virtual machines.

In this paper, the Cloud based Hybrid evolution algorithm for NP-Complete Pattern in Nurse Scheduling Problem with feedback assistance method is proposed to provide the services for the genetic algorithm or evolution algorithm user. The main contributions of the proposed algorithm and system are as follows:

(1) The genetic algorithm and evolution system with feedback assistance method are established in cloud as a software computing service (SCaaS). System users can provide the related information for online pattern recognition. After recognizing pattern, the classified NSP problem will be automatically processed System users need not to establish the individual method or algorithm by themselves.

(2) Based on the feedback assistance, the corresponding evolution steps such as crossover methods with the better performance will be dynamically emphasized according to the

Manuscript received January 7, 2016; revised August 11, 2016.

Ming-Shen Jian and Ming-Sian You are with the Cloud Intelligent Information System Lab., Dept. of CSIE, National Formosa University, Yunlin County, Taiwan 632 (e-mail: jianms@gmail.com, csppp2000@gmail.com).

limited probability based on the survival rate during evolution. Furthermore, the limited probability value for further computing and evolution can be tuned automatically.

(3) Different selection and crossover methods are verified in cloud. The proposed algorithm and system can obtain the better performance with low extra computing load.

The rest of this paper is organized as follows. Section II introduces the related work and the problem statement. Section III presents the proposed algorithm. Performance evaluation is given in Section IV. At last, the conclusion is drawn in Section V.

II. Related Work

In this section, the pattern recognition and problem statement of NSP problem is given. In addition, the e cloud based software computing as a service (SCaaS), the hybrid evolution algorithm with feedback assistance, will be introduced.

A. Related Works and Concepts of Evolution Algorithms

The basic genetic algorithm (GA) standard procedures are shown as follows.

Step 1. Representation: The first step is to transform all possible solutions as the genotype form. Then, the objective function (also called the fitness function) is also defined to evaluate each possible solution (chromosome) and give a value.

Step 2. Initialization: The population of the first generation is generated (often randomly) for being the parents of the second generation.

Step 3. Selection: This procedure selects the proper individuals (chromosomes) as parents to reproduce the newborn offspring.

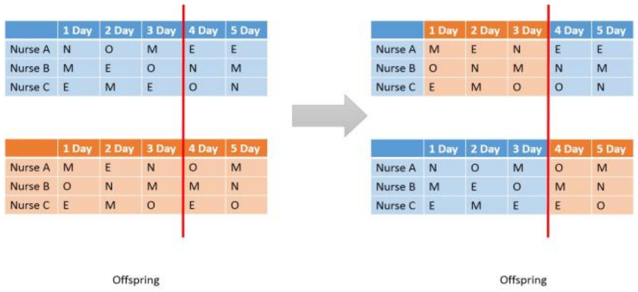
Step 4. Crossover: This procedure would mix parents' genes of chromosome and pass them to children. The chromosomes of offspring consist of the partial chromosomes from its parents. Since there are various crossover methods, in this paper, different crossover methods are re-modified as the modular blocks of the genetic algorithm.

Step 5. Mutation: This procedure would simulate accidents happening in reproduction and vary gene in random way.

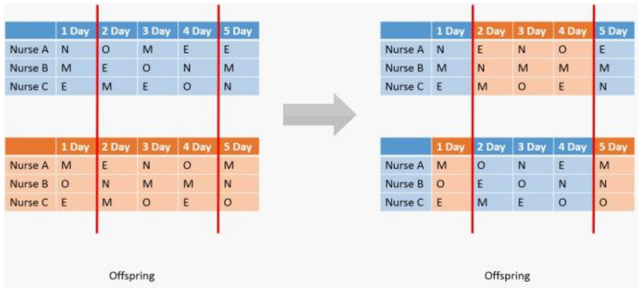
Step 6. Survival: According to Darwin's "survival of the fittest" rule, some individuals will be dropped to maintain the same population size. The chromosomes which pass this step are better in the fitness evaluation function.

Step 7. Checking: If the stop criteria are satisfied, then stop. Otherwise, go to Step 3.

In this paper, the partially mapped crossover (PMX) is developed by Goldberg and Lingle. Furthermore, to have better performance of computing, two different PMX methods are implemented: the single-point (one cut) PMX method and two-point (two cuts) PMX method. The genes of the chromosomes of the parents after the cut line are exchanged to produce the chromosomes of the offspring. Similar to the single-point PMX, the double-point PMX will select the partial section of the chromosome randomly for exchanging. Two points will be randomly selected. Fig. 1 shows the single point (one cut) PMX result.



(a)



(b)

Fig. 1. (a) Single-point PMX; (b) Double-point PMX.

In addition to crossover, to increase the diversity, the mutation method is also implemented. After random selection, one gene of the offspring will be mutated with the limited probability. The mutated gene will be randomly set as another type of gene presentation. Fig. 2 presents the mutation of the offspring.

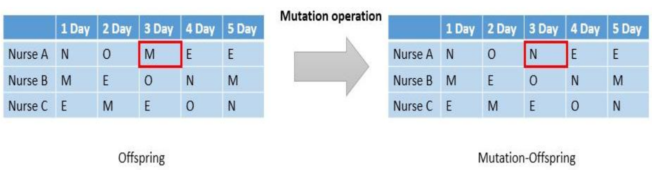


Fig. 2. The mutation of the offspring.

B. Pattern Recognition and Problem Statement

Since there are different types of NP-complete problems, to recognize the problem as the pattern of NSP is important. Suppose that N is the total number of individuals which can be assigned the corresponding value, s, that indicates the states of N. In addition, D is the period parameter. Then, if the problem is represented as an $N \times D$ matrix, the individual element of this matrix, x_{ij}, can indicate the state of N where $x_{ij} = \{s_1, s_2, \ldots, s_m\}$.

For the real implementation, some constrain may be given. Suppose that there are given the maximum and minimum threshold $n_{\max}$ and $n_{\min}$ which indicates the maximum and minimum number of n with the same state s at the D_i (i^{th}) period. In addition, the constraint for the state between x_{ij} and $x_{i+1\,j}$ may be given such as s_3 cannot be followed s_1, etc. To balance the $N \times D$ matrix, the extra constraint, that different types of state for N_j (j^{th}) individual should be balanced, can be added.

If the system searches the possible solution which against the constraints, the extra cost $C = \{c_1, c_2, \ldots, c_h\}$ corresponding to total h constraints will be added. Finally, the object function for finding the optimal solution (minimized value) can be given as:

$$f = c_1{*}w_1 + c_2{*}w_2 + \ldots + c_k{*}\, w_h \tag{1}$$

where w_1, w_2, …, w_k are weight values for c_1, c_2, …, and c_h, respectively.

Hence, if the NP-complete problem matches the pattern description, this specific problem will be recognized as the NSP problem.

III. Proposed Algorithm, System, Simulation Results

A. Genetic Algorithm with Feedback Assistance Method

The feedback control is added in the genetic algorithm and provides the probabilities for activating different types of crossover methods based on the cloud virtual machine. The probability controls the amount of virtual machines with the specific crossover procedure activated. Suppose that there are total K types of the crossover methods. The feedback control evaluates the probability according to the function as follows:

$$C_k = \frac{|ch_k|}{|\text{Total Individuals}|} \tag{2}$$

where ch_k indicates the survived chromosome created by the k^{th} type of the crossover method. In other words, the crossover method with the better performance (high survival rate) may be further executed according to the probability C_k. The total number of the offspring chromosome can be averagely created from each crossover method according to the crossover rate, β% (<100%). In addition, the total percentage of the offspring chromosome from the k^{th} type of the crossover method will be set to $\beta(k)/K$ %. Therefore, the feedback control assistance will emphasize the crossover method with the better performance. Furthermore, the emphasized crossover method will be further executed once to create more offspring chromosomes. Then, the genetic algorithm can have the chance to search the optimal solution more quickly. Fig. 3. shows the flowchart of the proposed algorithm.

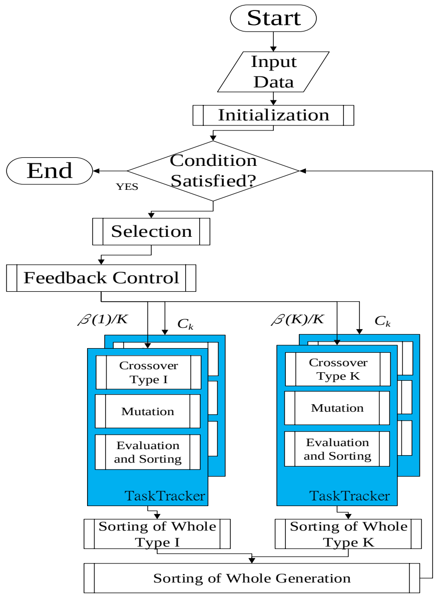


Fig. 3. The flowchart of the proposed system.

B. System Structure

In this paper, the genetic algorithm of cloud computing based modular processing is shown as Fig. 4.

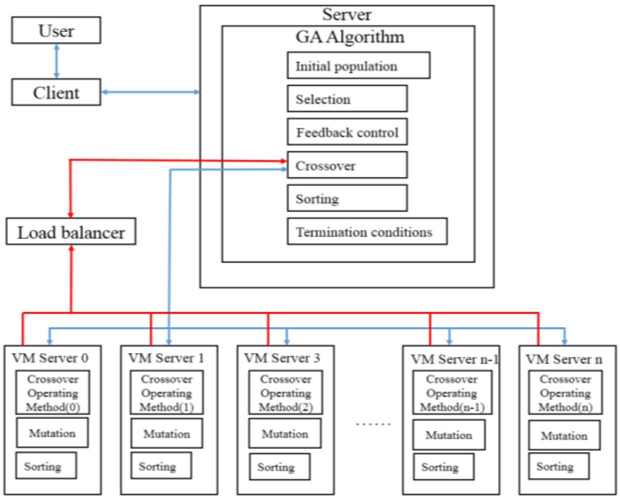


Fig. 4. The system structure.

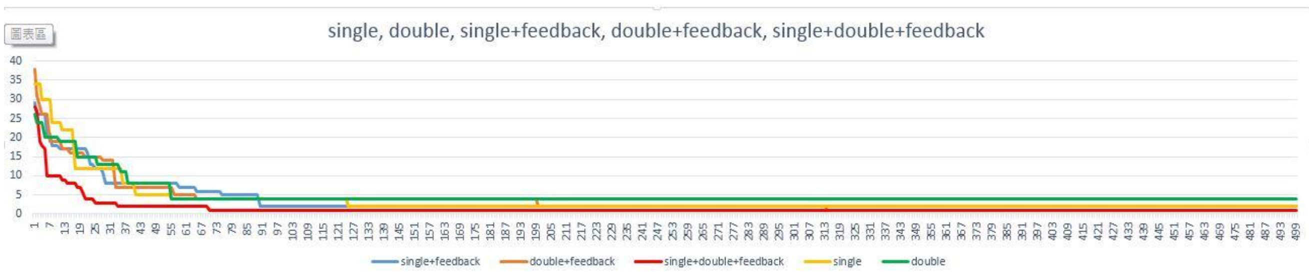


Fig. 5. Simulation results of different types evolution algorithms.

The client user can upload or set the values of the corresponding parameters in genetic algorithm. The server will separate the computing load based on the load balancer into different virtual machine. Each virtual machine will only maintain one individual crossover method. In other words, different crossover methods for optimal solution searching in genetic algorithm can be individually implemented.

C. Simulation Results

The experiment settings, number of personnel management (N) = 10, the total number of (D) day = 5 days. The constraints are 3 working days and 2 times rest. Initial population size is 100 and limited generation is 5000. The crossover rate is 50% and mutation rate is 10%.

Fig. 5 presents the evolution result. There are five types of evolution: single PMX, double PMX, single PMX with feedback assistance, double PMX with feedback assistance, and dynamical hybrid (single PMX + double PMX) evolution. The simulation result shows that the hybrid (single PMX + double PMX) type can obtain the optimal solution faster than other types. The proposed hybrid evolution algorithm can find the optimal solution with about 50% less evolution generations.

IV. Conclusions

In this paper, the Cloud based Hybrid evolution algorithm for NP-Complete Pattern in Nurse Scheduling Problem with feedback assistance is proposed to solve the emergency logistic problem. Via the proposed algorithm, the NP-Complete Pattern can be recognized. The whole convergence process can be enhanced more efficiently and effectively about 42%-50%. The suitable crossover procedure which can obtain the better results can be further adopted according to the evaluation result. The increased workload is only about 5% and can be distributed. In addition, the total times of crossover and evaluation between different crossover procedures can be still under controlled.

References

[1] M. Ohki and S. Kishida, "A parameter free algorithm of cooperative genetic algorithm for nurse scheduling problem," in *Proc. 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI),* Aug. 2013, pp. 1201-1206.

[2] M. Ohki and H. Kinjo, "Penalty weight adjustment in cooperative GA for nurse scheduling," in *Proc. 2011 Third World Congress on Nature and Biologically Inspired Computing (NaBIC),* Oct. 2011, pp. 75-80.

[3] R. A. Abobaker, M. Ayob, and M. Hadwan, "Greedy constructive heuristic and local search algorithm for solving nurse rostering problems," in *Proc. 2011 3rd Conference on Data Mining and Optimization (DMO),* Jun. 2011 pp. 194-198.

[4] C. Mueller, "Multi-objective optimization of software architectures using ant colony optimization," *Lecture Notes on Software Engineering,* vol. 2, no. 4, pp. 371-374, November 2014.

[5] S. Kim, Y. Ko, S. Uhmn, and J. Kim, "A strategy to improve performance of genetic algorithm for nurse scheduling problem," *International Journal of Software Engineering and Its Applications,* vol. 8, no. 1, pp. 1-7, Jan. 2014.

[6] J. J. Wu, Y. Lin, Z. H. Zhan, W. N. Chen, Y. B. Lin, and J. Y. Chen, "An ant colony optimization approach for nurse rostering problem," in *Proc. 2013 IEEE International Conference on Systems, Man, and Cybernetics,* Oct. 2013, pp. 13-16.

[7] N. Todorovic and S. Petrovic, "Bee colony optimization algorithm for nurse rostering," *IEEE Transactions on Systems, Man, and Cybernetics: Systems,* vol. 43, no. 2, pp. 467-473, Mar. 2013.

[8] M. A. Awadallah, A. T. Khader, M. A. Al-Betar, and A. L. Bolaji, "Hybrid harmony search for nurse rostering problems," in *Proc. 2013 IEEE Symposium on Computational Intelligence in Scheduling (SCIS),* Apr. 2013, pp. 60-67.

[9] E. Cetin and A. Sarucan,"Nurse scheduling using binary fuzzy goal programming," in *Proc. 2015 6th International Conference on Modeling, Simulation, and Applied Optimization (ICMSAO),* May 2015, pp. 1-6.

[10] J. P. Cares and M. C. Riff, "A collaborative approach to solve a nurse scheduling problem," in *Proc. 2012 IEEE 24th International Conference on Tools with Artificial Intelligence (ICTAI),* Nov. 2012, vol. 1, pp. 344-350.

[11] M. Jamom, M. Ayob, and M. Hadwan, "A greedy constructive approach for nurse rostering problem," in *Proc. 2011 3rd Conference on Data Mining and Optimization (DMO),* Jun 2011, pp. 227-231.

[12] M. Mutingi and C. Mbohwa, "A multi-criteria approach for nurse scheduling fuzzy simulated metamorphosis algorithm approach," in *Proc. 2015 International Conference on Industrial Engineering and Operations Management (IEOM),* Mar 2015, pp. 1-7.

[13] M. Ohara and H. Tamaki, "Integer programming approach for a class of staff scheduling problems–Schedule optimization and parameter estimation," in *Proc. 2012 Joint 6th International Conference on Soft Computing and Intelligent Systems (SCIS) and 13th International Symposium on Advanced Intelligent Systems (ISIS),* Nov. 2012, pp. 877-882.

[14] M. Ohara and H. Tamaki, "Mathematical programming approach based on column generation for a class of staff scheduling problems," in *Proc. 2015 54th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, 2015, pp. 240-245.

[15] J. J. Chen, J. L. Zhu, and D. N. Zhang, "Multi-project scheduling problem with human resources based on dynamic programming and staff time coefficient," in *Proc. 2014 International Conference on Management Science & Engineering (ICMSE)*, Aug. 2014, pp. 1012-1018.

[16] A. A. Constantino, D. Landa-silva, E. L. de Melo, and W. Romao, "A heuristic algorithm for nurse scheduling with balanced preference satisfaction," in *Proc. 2011 IEEE Symposium on Computational Intelligence in Scheduling (SCIS)*, Dec. 2011, pp. 39-45.

[17] K. Anwar, M. A. Awadallah, A. T. Khader, and M. A. Al-Betar, "Hyper-heuristic approach for solving nurse rostering problem," in *Proc. 2014 IEEE Symposium on Computational Intelligence in Ensemble Learning (CIEL),* Dec. 2014, pp. 1-6.

[18] M. Cheng, H. I. Ozaku, N. Kuwahara, K. Kogure, and J. Ota, "Simulated annealing algorithm for daily nursing care scheduling problem," in *Proc. IEEE International Conference on Automation Science and Engineering,* 2007, pp. 507-512.

[19] Y. Yu, Y. Chen, and T. Li, "A new design of genetic algorithm for solving TSP," in *Proc. 2011 Fourth International Joint Conference on Computational Sciences and Optimization (CSO)*, Apr. 2011, pp. 309-313.

[20] Y. C. Shen, C. H. Hsu, and S. H. Hsieh, "Integrated genetic algorithms and cloud technology to solve travelling salesman problem on Hadoop," in *Proc. 2012 IEEE 4th International Conference on Cloud Computing Technology and Science (CloudCom)*, Dec. 2012, pp. 566-569.

[21] Z. Zhao, G. Zhang, and Z. Bing, "Job-shop scheduling optimization design based on an improved GA," in *Proc. 2012 10th World Congress on Intelligent Control and Automation (WCICA)*, Jul. 2012, pp. 654-659.

[22] Y. L. Xu, G. Tang, and L. Zhongyong, "An algorithm of real-time solving deadlock for job-shop schedule," in *Proc. 2011 IEEE International Conference on Computer Science and Automation Engineering (CSAE),* June 2011, pp. 431-435.

Ming-Shen Jian is an assistant professor of the Department of Computer Science and Information Engineering at National Formosa University. He received the Ph.D. at the Department of Computer Science and Engineering of National Sun Yet-Sen University in Taiwan in 2007, investigating resource management in 3G mobile communication systems.

He was a researcher in IST LLC. Currently he is an assistant professor in National Formosa University, Taiwan, and also serves as the secretary in Taiwan Association of Cloud Computing. He has published books such as *Cloud Operation System* (CCIS Lab., 2015), *Routing Distribution and Selection in Mobile Wireless Network* (VDM, 2009), etc. His current research interests and publications are in the area related to cloud computing, RFID and IoT application, optimal solution, and e-learning.

Dr. Jian is a member of IEEE and Taiwan Association of Cloud Computing. He was awarded Gold Medal at International Salon of Inventions and New Technology New Time in 2014 and 2015.

Ming-Sian You is with the Department Computer Science and Information Engineering at National Formosa University. He received his bachelor's degree at the Department Computer Science and Information Engineering at National Formosa University in Taiwan, 2014. Currently he is a master student in National Formosa University, Taiwan. His current research interests are in the area related to cloud computing, optimal solution.