A Fuzzy Genetic Algorithm for Scheduling of Handling/Storage Equipment in Automated Container Terminals

Seyed Mahdi Homayouni, *Senior Member, IEDRC* and Sai Hong Tang

Abstract—Integrated scheduling of handling/storage equipment in container terminals is an NP-hard problem which has been studied during past two decades consciously. Genetic algorithms (GAs) have been applied for this optimization problem in many researches. However, the GA is vulnerable to trap in a local optima (results in premature convergence). In this paper a fuzzy logic controller (FLC) is designed to improve the performance of a GA in optimization of integrated scheduling of handling/storage equipment in automated container terminals. The FLC controls crossover and mutation rates of the GA during its generations, which are the main control parameters of the GA to avoid the premature convergence. The numerical results for the small size test cases solved by using the proposed fuzzy genetic algorithm show that solutions found by this algorithm are 2.5% better than the solutions found by the GA. Studies are continuing for better performance of the proposed FGA.

Index Terms—Integrated scheduling, fuzzy logic controller, genetic algorithm, fuzzy genetic algorithm.

I. INTRODUCTION

Containers are standardized metal boxes designed to transfer goods from door to door without the contents being physically handled [1]. Container terminals (CTs) are equipped to transship the containers from/to the containerships. The containers are stored in storage yards temporarily. World trade is facilitated through the elimination of trade barriers and the liberalization of markets, which result in the development of logistics throughout the world. Today, over 60% of the world's deep-sea general cargo is transported in containers [2]. Latest figures published in the United Nations Conference on Trade & Development (UNCTAD) [3] illustrate that worldwide container port throughput increased from 36 million TEUs (twenty-foot equivalent units) in 1980 to more than 600 TEUs in 2012.

Transshipment of the containers as quickly as possible and at the least possible costs is the primal goal of the CTs. One possible opportunity to improve the performance of CTs is to enhance the degree of automation of handling equipment [4]. Automated container terminals (ACTs) have been introduced

by the researchers, to respond ever-increasing demands for containerization. In ACTs, quay cranes (QCs) are responsible to load/ unload containers to/ from containerships. QC put/pick up the container to/ from a vehicle, e.g. automated guided vehicle (AGV). AGVs are responsible to transport the containers between the quayside and storage yard of the port. The storage yard can be served by automated yard cranes.

The main loss of performance in an ACT is the uncoordinated scheduling of various equipment [5]. Homayouni and Tang [6] proposed a scheduling method for QCs that consider the availability of AGVs, simultaneously. In that research, the near optimal sequence of tasks for the QCs is calculated by using a genetic algorithm (GA). GAs operate on a population of potential solutions applying the principle of survival of the fittest to produce better and better approximations to a solution [7]. The current research proposes a fuzzy logic controller (FLC) is designed to enhance the performance of the GA proposed by Homayouni and Tang [6].

The remainder of the paper is organized in following sections. The problem of integrated scheduling of handling/ storage equipment in ACTs is briefly described in Section II. Section III reviews the most important notes in designing the fuzzy logic controllers. Principles for the application of fuzzy logic controllers in genetic algorithms and steps of the algorithm are described in Section IV. Section V is dedicated to the numerical examples and results of some test to show the performance of the proposed fuzzy genetic algorithm. Finally, Section VI provides conclusion remarks of the research, and suggests opportunities for further researches.

II. PROBLEM DEFINITION

Operations related to containerships primarily consist of unloading and loading operations. In loading tasks, the AGV moves from its dwell point to the assigned load/ unload (L/U) station, receiving the container and moves to pick-up/ delivery (P/D) point. The container is loaded to the containership, if the QC is free. In an occasion that QC is not free, the AGV has to wait for it. Conversely, in unloading tasks, the AGV moves from its dwell point to the assigned QC and QCs handle the container from the containership to P/D point. Once AGV reached P/D point, it receives the container and moves to the assigned L/U station. The AGV delivers the container to L/U station, and waits for next assigned operation. It is supposed that storage cells are available for all the containers handled to storage yard. Moreover, the containers are available to be immediately loaded on AGVs. Therefore, the operational time of the L/U

stations is not calculated in the proposed scheduling method.

Lau and Zhao [5] stated that efficiency of the container terminals might be improved through integrating various required scheduling processes. They proposed a procedure to schedule three types of equipment in ACTs namely, QCs, AGVs, and yard cranes. The effectiveness of the scheduling procedures are very sensitive to its operational design parameters. As the AGVs need too many other factors besides their scheduling, the following assumptions were kept in mind of the authors for the current research:

- Congestions of the AGVs are ignored in the guided path.
- All AGVs start from a specific L/U station, initially and return there after finishing all the tasks.
- Handling time between various equipment of the ACT is assumed deterministic.
- Speed of movement for loaded and empty AGVs is equal.
- The dwell point policy for both QCs and AGVs is stay in place where their last task finished.

III. The Genetic Algorithm

The integrated scheduling of AGVs and QCs is an NP-hard problem [5], [6]. Homayouni and Tang [6] have developed a genetic algorithm to solve the scheduling of QCs and AGVs. A heuristic method is designed to assign the AGVs to the proper task. Fitness function for the GA is defined as the summation of total travelling time of the vehicles (TT) and total delays in final tasks of the quay cranes (TD). Delay is defined as the difference between real completion time of the tasks of a crane and its earliest completion time if a vehicle is available for all the tasks of a crane.

$$\text{Fitness} = (\alpha \times TT) + (\beta \times TD) \qquad (1)$$

The journeys of AGVs are performed next to finishing the prior assigned task. If more than one AGV is available for one task, the AGV, which can reach the P/D point or L/U station earlier, is selected. If a vehicle is ready and free; the heuristic method finds the next task and the AGV moves to assigned P/D point or L/U station. This hybrid AGV scheduling algorithm results in a reduction of the waiting times and thus, improves the utilization and the throughput of the whole ACT. The details of the heuristic AGV assignment method are described as followings:

Step 1: Consider the first task selected by the GA to be scheduled.

Step 2: If the task is a loading one, find which AGV reach the L/U station earlier, else go to Step 3.

- Move the AGV to L/U station, pick up the container,
- Move the loaded AGV to the QC, wait until QC is free (if required),
- Deliver the container to the QC, update the current time of AGV,

QC put the container in the containership. Update the current time of the QC, go to Step 4.

Step 3: If the job is an unloading task, find which AGV reach the P/D point earlier,

- Move the AGV to the desired P/D point, wait until arrival of the QC (if it is needed),
- Unload the container by the QC, move the loaded QC to P/D point,
- Put the container on AGV, update the current time of the QC,
- Move the loaded AGV to the L/U station, update the current time of the AGV.

Step 4: Check whether all the tasks are completed.

Step 5: Calculate total travelling time of the AGVs and the delay time of the QC tasks.

The GA schedules tasks based on the precedence relations for each QC. The proposed sequence of tasks is used to assign available AGVs, based on the heuristic method. Moreover, the method calculates the objective function of GA. A permutation encoding system is used to represent the tasks. Hence, a chromosome is a combination of integer numbers in range of one to n (total number of tasks), however if and only if the chromosome observes the precedence relations it can be recognized as a feasible chromosome, otherwise the chromosome must be ignored.

Using the roulette wheel selection schema, parents are selected for the crossover operator. Task-based crossover proposed by Reddy and Rao [8] and it is modified for the current research. A QC is selected arbitrarily and its connected tasks in parents are copied in matching positions in offspring. The unfulfilled cells are filled using the same sequence of tasks for the other parent. The following steps are used for task-based crossover:

1) Randomly select one QC from the given QC set.
2) Mark the tasks of the selected QC on the parent strings.
3) Copy the tasks of selected QC in parent 1 onto the matching positions of offspring 2.
4) Copy the tasks of selected QC in parent 2 onto the matching positions of offspring 1.
5) Fill the unfulfilled positions of the offspring 2 by the tasks of the unselected QCs from left to right according to their order of appearance in parent 2.
6) Fill the unfulfilled positions of the offspring 1 by the tasks of the unselected QCs from left to right according to their order of appearance in parent 1.

Task-based mutation operator proposed by Badakhshian *et al.* [9], which do the mutation if and only if the mutation does not offend the precedence relations of tasks. In task-based mutation, firstly one chromosome is selected for mutation. In second step, two gens are selected randomly. The mutation is valid only if one of the situations is observed for both selected positions.

- If the first selected gen is the last task of the QC, it can jump to a later gen.
- If the second selected gen is the first task of a QC, it can be jumped to a former gen.
- If the first gen is not the last task of the QC, examine whether its subsequent task is located after the new gen.
- If the second gen is not the first task of the QC, examine whether its precedent task is located before the new gen.

IV. Fuzzy Genetic Algorithms

Genetic algorithm is a robust optimization tool which is

used by the researchers when the problem is complicated or the solution is not directly reachable. GAs works on a population of feasible solutions and tries to evolve through its running period. GAs have two main operators, crossover and mutation operators. Exploration and exploitation among the feasible solutions are basic principles of GAs. A trade-off between these two major factors is achieved through crossover and mutation operators. Thus, setting the crossover and mutation rates are of vital importance for the performance of the GA. The researchers proposed to use the fuzzy logic controllers to control the main factors of GA [10]. Theses researches showed that fuzzy GAs have a better performance than the GAs.

Herrera and Lozano [11] state that FLCs are very useful when the processes are too complex for analysis by conventional quantitative techniques. Moreover, if the available information is qualitative, non-precise, or uncertain the FLCs may be useful to analyze the problem. FLCs composed of two main components: a data base which collects the information about the fuzzy sets, their linguistic variables and the membership function for each of the linguistic variables; and a rule base which is used as a collection of fuzzy rules to interpret perceives of the environment and decides about the output of the FLC system. The creator of the FLC needs to develop both data base and rule base for an FLC. A fuzzy rule can be defined by two main section, first the antecedent, and second the consequent. Therefore, if the antecedent is happen in a degree, results in consequent in the same degree of membership.

The main objective to use FLCs for GAs is to determine the major parameters of GAs. The FLC receives the indices of GA periodically as its inputs, and through its rule base decides about these parameters. The parameters are used during various iterations of the GA, for a better performance of GA. The most important step in designing the FLC is to know the required outputs. In FGA methodology, various parameters of GA are considered to be controlled by the FLCs such as crossover and mutation rate, surviving percentage, and stop criteria [12].
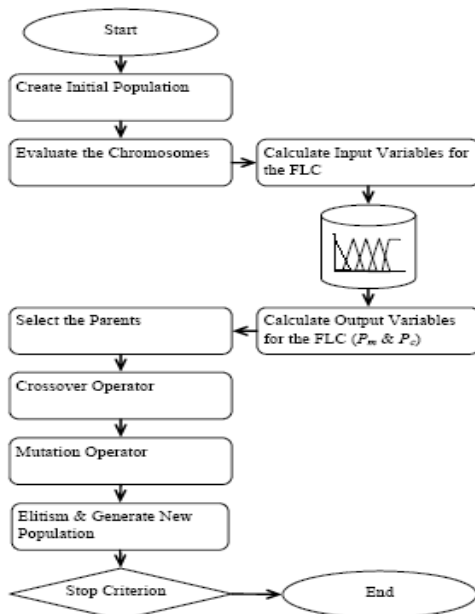
In this paper, the crossover and mutation rates are considered to be controlled by the proposed FLC. Fig. 1 illustrates the flowchart for the proposed FGA for the current research. The candidate input variables are the ones which have a great significance for the chosen FLC outputs. The best value of fitness function in each generation (BV), frequency of chromosomes with similar best fitness value in a generation (FBV), and the frequency of same chromosomes produced in a generation (FSS) are input variables for the FLC. On the other hand, mutation rate (P_m) and crossover rate (P_c) are out put variables of the FLC.

TABLE I: Fuzzy Rule Base for the FGA [12]

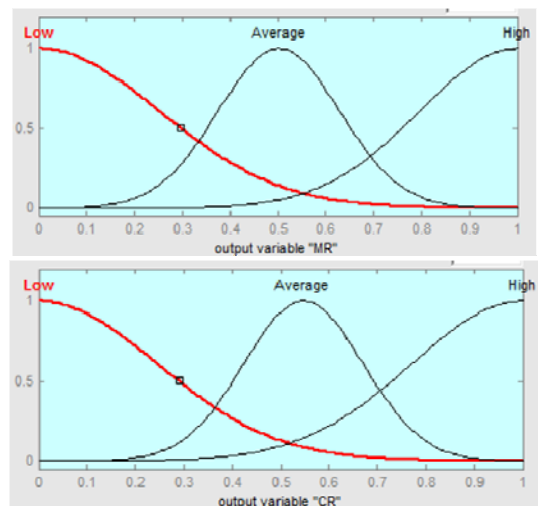| Input Variables			Output Variables	
BV	*FBV*	*FSS*	P_m	P_c
Good	Low	Low	Low	High
Good	Low	Average	Average	High
Good	Low	High	High	High
Good	Average	Low	High	Average
Good	Average	Average	High	Average
Good	Average	High	High	Average
Good	High	Low	High	Average
Good	High	Average	High	Average
Good	High	High	High	Low
Average	Low	Low	Low	High
Average	Low	Average	Average	High
Average	Low	High	High	High
Average	Average	Low	Average	High
Average	Average	Average	Average	Average
Average	Average	High	High	Average
Average	High	Low	Average	Average
Average	High	Average	High	Average
Average	High	High	High	Low
Poor	Low	Low	Low	High
Poor	Low	Average	Average	High
Poor	Low	High	High	High
Poor	Average	Low	Low	High
Poor	Average	Average	Average	High
Poor	Average	High	High	High
Poor	High	Low	Average	High
Poor	High	Average	High	High
Poor	High	High	High	High

Fig. 1. The proposed fuzzy genetic algorithm.

Fig. 2. Membership Functions of Output Variables of FLC Module.

For each selected variable, either input or output, its domain is defined in three fuzzy linguistic values, {low, average, and high}. The trapezoidal and triangular membership functions (MFs) were used to model the values of the input variables. The output variables are represented by using Gaussian MFs, shown in Fig. 2.

The 27 rules proposed by de Brito *et al.* [12] are defined in a way that decreases the crossover rate when the optimal characteristic of the current population is not so good which inherit by the offspring. The full rule base is displayed in Table I. This is determined by the best value of current generation. Moreover the mutation rate needs to be increased in such cases. When the number of repeated similar chromosomes with best value is increasing and the best value is not recognized as the good one, a premature convergence may occurred. In such cases the fuzzy rule base tries to increase the mutation rate.

TABLE II: COMPARISON OF THE GA AND THE PROPOSED FGA

No	GA [6]			Proposed FGA			Deviation %
	Mean	Best	CPU time	Mean	Best	CPU time	
1	471	471	7.4 S	471	471	9.2 S	0
2	775	775	9.6 S	778	775	12.7 S	0
3	818	818	9.8 S	814	811	13.3 S	0.9
4	1065	1061	12.3 S	1063	1058	15.8 S	0.3
5	689.2	687	12.4 S	682	678	16.1 S	1.3
6	1033.5	1016	12.6 S	985	976	17.3 S	4.1
7	921.3	916	14.8 S	903	886	21.3 S	3.4
8	1110.5	1089	16.8 S	1073	1068	23.6 S	1.9
9	1186.2	1162	15.3 S	1171	1157	30.2 S	0.4
10*	1165.3	1159	20.3 S	1158	1142	36.7 S	1.5

V. NUMERICAL RESULTS

In this section, for the purpose of performance evaluation of the proposed FGA, some numerical test are designed and performed. The proposed FLC and the proposed FGA are programmed in MATLAB® toolbox for fuzzy logic controllers. Ten numerical test cases have been designed by Homayouni and Tang [6]. These small-size test cases have been designed so that their developed MIP model and the proposed GA are able to solve the test cases. To evaluate the performance of the proposed FGA, these test cases are solved by using this algorithm. Results for these test cases are reported in table II. In this table, each test case is solved using the FGA and GA for ten times. Mean of best objective values of these 10 runs of FGA and GA are recorded in this table. Moreover, the best objective value for each test case is shown in table II. Deviation shows how much the best solution found by the GA [10] is greater (less) than the best solution found by the proposed FGA. Deviation is calculated as shown in Equation (2).

$$Deviation\% = \frac{Best(GA) - Best(FGA)}{Best(FGA)} \times 100 \qquad (2)$$

As reported in Table II, both the GA and the FGA are able to find optimal solutions for test cases No. 1 and 2. As test cases get larger dimensions, the FGA is more able to find better solutions for theses test cases.

Fig. 2 illustrates variations of the mutation and crossover rates calculated by the proposed FLC during generations of the genetic algorithm. As the main charactristics of the GA changes during its generations, it affects on the input variables of the FLC module, thus the FLC changes the mutation and crossover rates and return it to the GA method. This figure shows that crossover rate decreases as the solutions converge to a near optimal one. On the other hand, the mutation rate increases in order to escape form the local optima trap.
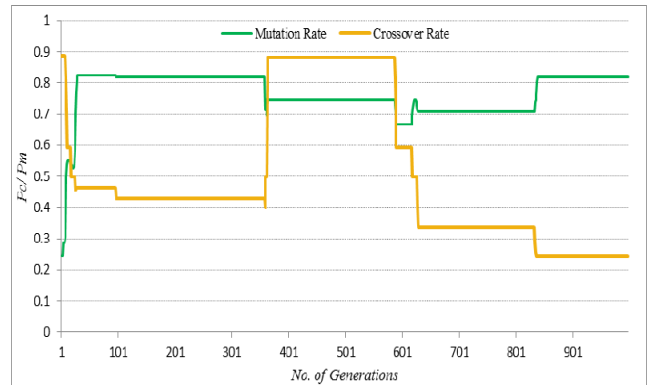

Fig. 2. Variations of mutation and crossover rates during various generations of the proposed FGA.

VI. CONCLUSIONS

The integrated scheduling of handling/storage equipment has been researched during last decades. However, no efforts have been reported on application of fuzzy logic controllers to improve the performance of the proposed meta-heuristic algorithms. Genetic algorithms are vulnerable to stuck in premature convergence trap. Using the FLCs to control the crossover and mutation rates during generations of the GA may reduce chance for premature convergence. Solving the same test cases using both GA and FGA shows that in average, FGA would find 2.5% better solutions for the test cases. Because, the GA have found near optimal (the optimal solution were found by the mathematical models), the performance of the FGA is not so brilliant. However, it is predicted for the large scale test cases that the FGA outperforms the GA. Considering medium and large scale cases is planned for future studies of the authors.

REFERENCES

[1] Y. L. Cheng *et al.*, "Dispatching automated guided vehicles in a container terminal," *Supply Chain Optimization*, Springer, 2005, pp. 355-389.

[2] D. Steenken, S. Voß, and R. Stahlbock, "Container terminal operation and operations research — A classification and literature review," *Container Terminals and Automated Transport Systems*, Springer Berlin Heidelberg, 2005, pp. 3-49.

[3] UNCTAD, *Review of Maritime Transport*, in United Nations Conference on Trade and Development, United Nations: New York and Geneva, 2013.

[4] K. H. Kim and J. W. Bae, "A look-ahead dispatching method for automated guided vehicles in automated port container terminals," *Transportation Science*, vol. 38, pp. 224-234, 2004.

[5] H. Y. K. Lau and Y. Zhao, "Integrated scheduling of handling equipment at automated container terminals," *Annals of Operations Research*, vol. 159, pp. 373-394, 2008.

[6] S. M. Homayouni and S. H. Tang, "Multi objective optimization of coordinated scheduling of cranes and vehicles at container terminals," *Mathematical Problems in Engineering*, vol. 2013, pp. 1-9, 2013.

[7] E. Alba and R. Marti, *Metaheuristic Procedures for Training Neural Networks*, Springer, vol. 35, 2006.

[8] B. Reddy and C. Rao, "A hybrid multi-objective GA for simultaneous scheduling of machines and AGVs in FMS," *The International Journal of Advanced Manufacturing Technology*, vol. 31, pp. 602-613, 2006.

[9] M. Badakhshian, S. B. Sulaiman, and M. K. A. B. M. Ariffin, "Performance optimization of simultaneous machine and automated guided vehicle scheduling using fuzzy logic controller based genetic algorithm," *International Journal of Physical Sciences*, vol. 7, pp. 1461-1471, 2012.

[10] O. Cordón *et al.*, "Ten years of genetic fuzzy systems: current framework and new trends," *Fuzzy Sets and Systems*, vol. 141, pp. 5-31, 2004.

[11] F. Herrera and M. Lozano, "Fuzzy adaptive genetic algorithms: design, taxonomy, and future directions," *Soft Computing*, vol. 7, pp. 545-562, 2003.

[12] F. H. de Brito *et al.*, "A fuzzy intelligent controller for genetic algorithms' parameters," *Advances in Natural Computation*. Springer, 2006, pp. 633-642.

Seyed Mahdi Homayouni was born in Isfahan, Iran, in 1980. He received his PhD and MSc. degrees in industrial and systems engineering from University Putra Malaysia, Selangor, Malaysia, in May 2012 and Feb. 2008, respectively.

He is now an assistant professor at the Department of Industrial Engineering, Lenjan Branch, Islamic Azad University, Isfahan, Iran. He is the main author of the book "genetic fuzzy logic controllers for complex production systems" (Saarbrücken, Germany, Lambert Academic Publishing, 2013). His research interests are on scheduling and planning problems in production systems, supply chains. He is also working on optimization methods and problems.

Dr. Homayouni is a member of production and operations management society, and He is a senior member of IEDRC. He is awarded as the distinguished researcher in the university in 2013 by the research deputy dean of the Islamic Azad University, Lenjan branch.

Sai Hong Tang received a doctorate degree from Dublin City University, Ireland in 1999. He is an associate professor at the Department of Mechanical and Manufacturing Engineering, Faculty of Engineering, University Putra Malaysia. He is the author and co-author of more than 10 books and book chapters, and more than 100 journal and conference papers. His teaching and research fields include mechanical and system engineering, operation research and robotics.