

Automated Guide Vehicles Dynamic Scheduling Based on Annealing Genetic Algorithm

Zou Gan^{*1}, Li Tao^{1,2}, Qin Ying^{1,2}

¹Mechanical and Electrical Engineering College, Kunming University of Science and Technology
Kunming, 650118, PR China.

²China Shipbuilding Trading (Kunming) Co., Ltd, 650051, PR China.

*Corresponding author, e-mail: zougan@hotmail.com^{*1}, zxst@126.com²

Abstract

Dispatching automated guided vehicles (AGVs) is the common approach for AGVs scheduling in practice, the information about load arrivals in advance was not used to optimize the performance of the automated guided vehicles system (AGVsS). According to the characteristics of the AGVsS, the mathematical model of AGVs scheduling was established. A heuristic algorithm called Annealing Genetic Algorithm (AGA) was presented to deal with the AGVs scheduling problem, and applied the algorithm dynamically by using it repeatedly under a combined rolling optimization strategy. The performance of the proposed approach for AGVs scheduling was compared with the dispatching rules by simulation. Results showed that the approach performs significantly better than the dispatching rules and proved that it is really effective for AGVsS.

Keywords: automated guide vehicles (AGVs), dynamic scheduling, rolling optimization strategy, annealing genetic algorithm (AGA)

Copyright © 2013 Universitas Ahmad Dahlan. All rights reserved.

1. Introduction

Automated Guided Vehicles (AGVs) are unmanned means of inplant transportation which are commonly used in facilities such as warehouses, manufacturing plants, terminals and distribution centers. AGVs increase efficiency and reduce costs by helping to automate a manufacturing facility or warehouse [1]. Application of AGVs has been broadened since the late 20th century. In the past few decades, much research has been devoted to the improvement of AGVs system (AGVsS). Basically, the relevant issues of AGVsS can be divided into the following main categories: guide-path design, estimating the required number of AGVs, idle AGVs positioning, AGVs scheduling, battery management and conflict resolution. These issues relate to different levels of the decision making process [2]. As one of the enabling technologies, scheduling of AGVs have attracted considerable attention. The AGVs scheduling is responsible for managing AGVs efficiently to guarantee serve jobs as quick as possible. Many rules or algorithms for the scheduling of AGVs have been proposed [2-5].

In general, AGVsS can be divided into two main categories: centralized and decentralized control systems. The centralized system uses information available at the central controller while the decentralized system dispatches AGVs based on local information available at the decision moment [3]. In practice, due to their efficiency, centralized control systems are more popular. Since dispatching rules are quite straightforward and easy to use. A wide variety of dispatching rules including nearest-workstation-first (NWF), nearest vehicle first (NVF) and modified first come first served (MOD-FCFS) are used for centralized control AGVsS [4]. Based on the ways in which transportation requests are assigned, Egbelu have divided dispatching rules into two categories: load-initiated dispatching rules in which jobs at workstation have the priority to claim vehicles and vehicle-initiated dispatching rules in which vehicles have the priority to claim jobs [5]. Dispatching is related to immediate decisions such as where a vehicle should be sent to at a specific moment. However, if advance load arrival information is known, they are outperformed by scheduling approaches [2].

In the literature, vehicle routing problems with time windows (VRPTW) have been studied extensively. B.L. Golden et al. provide a review of vehicle routing with time windows including the Pick-up and Delivery Problem with Time Window (PDPTW), multi-Travel Salesman

Problem with Time Window(m-TSPTW) and solutions approaches[6].They mention two main types of optimization algorithms for VRPTW: dynamic programming and branch-and-bound. Both methods are very time consuming and cannot solve practical problems within an acceptable time limit. Xu et al. propose the column-generation algorithm to solve VRPTW. In order to deal with practical-size problems. They use several heuristics to generate columns with negative reduced costs and eliminate unattractive columns by sophisticated column management schemes [7]. Wenfeng Wang et al. present an improved genetic algorithm to solve the VRPTW [8]. Cordeau et al. summarized several of the most important and modern heuristics for vehicle routing problem [9], Victor Pillac et al. provide a survey on solution approaches for dynamic vehicle routing problems [10].Two main approaches include an adaptation of the static solution and an implementation of static algorithms under a rolling horizon.

After studying the literature on the vehicle scheduling, we find that most studies concern external transport systems. Few authors use methods successfully deployed for external transport problems for internal transport like the AGVsS. The problem of dynamic scheduling of AGVs has not attracted many researchers [2]. Since the scheduling approach has been used widely and efficiently for external transport. We develop such scheduling strategy for AGVsS in this paper. The main purpose is to investigate the potential contribution of the scheduling approach for AGVs. AGVs scheduling problem is similar to the vehicle scheduling problem for external transport. However, it also has some differences such as the objectives, travel distances and load arrival rate. There is some literature on scheduling methods for AGVs [3] [11]. Several objectives are also used, including minimizing the average load waiting time and minimising empty vehicle travel distances, but they focus on static problems. An AGVsS is a system with a high degree of uncertainty, short travel times and, often, high vehicle utilization rates. In AGVsS, normally, we only know a limited number of jobs in advance. The schedule of vehicles should be updated when new transportation request information arrives. This paper focuses mainly on dynamic scheduling problem of AGVs, adapts successful dynamic scheduling approach for AGVsS, and compares its performances with common dispatching rules by simulation.

The remainder of this paper is organized as follows: In section 2, we describe the mathematical model, Annealing Genetic Algorithm (AGA) and the rolling optimization strategy of AGVs scheduling problem. Results and discussion are presented in section 3.we conclude the paper in Section 4.

2. Research Method

2.1. Mathematical Model of AGVs Scheduling

In AGVsS, AGVs picks-up loads at some locations and delivers them to their destinations satisfying certain time-windows. the AGVs scheduling problem can be formulated as a VRPTW. m-TSPTW is a special case of the VRPTW in which the vehicle capacities are infinite [6]. We formulate the scheduling problem for AGVs as a m-TSPTW by projecting time-windows at delivery stations to the corresponding pick-up stations and a pick-up and a corresponding delivery job is considered logically as a single job-node. We make the following assumptions for the studying AGVsS:

- All AGVs have unit-load capacity.
- AGVs operate continuously without breakdown.
- There are no traffic problems.
- AGVs choose the shortest path to pick up and deliver loads.
- AGV loading and unloading times are fixed and considered in travel times.
- AGVs can always park at their drop-off locations.
- Only one depot in AGVsS.

the following notations will help in the description of formulation of AGVs scheduling problem.

D — AGVs depot.

N — set of jobs.

K — set of AGVs.

x_{ijk} — take the value 1 if job i and job j are served by AGV k Consecutively, and 0 otherwise.

e_i — release time of job i .

l_i — the latest pick-up time allowed of job i .

s_i — pick-up time of job i .

t_{ij} — The travel time from job-node i to j , equals the travel time from the origin of job i to the destination of job i , plus the travel time from the destination of job i to the origin of job j .

s_{Dk} — the start time of AGV k at the AGVs depot.

s_{kD} — the arrival time of AGV k at the AGVs depot.

B — a sufficiently large positive number.

Normally, minimizing the average job waiting time is the most important objective of AGVs scheduling problem. The AGVs scheduling problem can be formulated as follows:

$$\text{minimize } \frac{1}{|N|} \sum_{i \in N} (s_i - e_i) \quad (1)$$

subject to:

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

$$\sum_{j \in D \cup N} x_{ijk} = \sum_{j \in D \cup N} x_{jik}, \forall i \in N, \forall k \in K \quad (3)$$

$$\sum_{k \in K} \sum_{j \in N} x_{jDk} = \sum_{k \in K} \sum_{i \in N} x_{Dik} = |K| \quad (4)$$

$$s_{Dk} + t_{Dj} - s_j \leq B(1 - x_{Djk}), \forall j \in N, \forall k \in K \quad (5)$$

$$s_i + t_{ij} - s_j \leq B(1 - x_{ijk}), \forall i, j \in N, \forall k \in K \quad (6)$$

$$s_i + t_{iD} - s_{kD} \leq B(1 - x_{iDk}), \forall i \in N, \forall k \in K \quad (7)$$

$$e_i \leq s_i \leq l_i, \forall i \in D \cup N \quad (8)$$

Constraints (2)-(4) form a multi-commodity flow formulation. The constraint (5) shows that if an AGV k serves job j after job i , the constraint $s_i + t_{ij} \leq s_j$ must be satisfied. Constraints (5)-(7) ensure feasibility of the schedule. Equations (9) is time-window for job i .

For the static AGVs scheduling problem, we assume that all AGVs start at the AGVs depot. However, during the process of optimization, AGVs may start at any load's drop-off station. Therefore, we need to modify the formulation to reflect this by replacing (4) and (6) by constraints (9) and (10) respectively, where D_k is the virtual starting depot of AGV k .

$$\sum_{k \in K} \sum_{i \in N} x_{D_k ik} = \sum_{k \in K} \sum_{j \in N} x_{jDk} = |K| \quad (9)$$

$$s_{D_k k} + t_{D_k j} - s_j \leq B(1 - x_{D_k jk}), \forall j \in N, \forall k \in K \quad (10)$$

2.2. Annealing Genetic Algorithm (AGA)

Though optimal solutions to AGVs scheduling problem can be obtained using exact methods, the computational time required to solve this problem to optimality is prohibitive. Heuristic methods such as Genetic Algorithms (GA), Simulated Annealing (SA) often produce optimal or near optimal solutions in a reasonable amount of computer time. Heuristic methods have been proved useful in a variety of search and optimization problems over the years. But most of single heuristic algorithms have their own shortcomings in the optimization [12]. In many problems, GA may have a tendency to converge towards local optima rather than the global optimum of the problem [13]. The execution time of SA is sensitive to the scale of the problem. In dealing with large-scale issue, the computational time of SA for getting a satisfactory solution is not acceptable [14]. According to the characteristics of AGVsS, we propose a hybrid algorithm called Annealing Genetic Algorithm (AGA) to optimize AGVs scheduling problem. AGA facilitates the exhaustive and parallel treatment of the problem and to increase the probability of finding global minimums in a reasonable computation time.

The proposed AGA is described as follows:

Step1: Initialize the parameters, i.e., population size $popsiz$, Maximum generations of evolution $MAXgen$, crossover probability p_c , mutation rate p_m , initial temperature T_0 , temperature cooling coefficient q , final temperature T_{end} .

Step2: generate initial population P_0 , compute individuals' fitness value $f_i (i = 1, 2, \dots, popsiz)$.

Step 3: set cycle count variable $gen = 0$.

Step 4: operation of selection, crossover and mutation, compute the fitness value of new individual f'_i , if $f'_i > f_i$, new individual is accepted, otherwise, new individual is accepted with a probability $p = \exp((f'_i - f_i) / T)$.

Step 5: if $gen < MAXgen$, then $gen = gen + 1$, go to step 4, otherwise, go to step 6.

Step 6: if $T_i < T_{end}$, Terminate the algorithm and print the solution, otherwise, $T_{i+1} = qT_i$, go to step 3.

The framework of the AGA is given in Figure 1.

In this paper, jobs have been numbered according to increasing jobs' release time, a solution is encoded as a chromosome formed of multiple segments. x_{kj} represents AGV k serves job j , a chromosome of the AGVsS scheduling problem can be expressed as $(0, i_{11}, i_{12}, \dots, i_{1s}, 0, i_{21}, i_{22}, \dots, i_{2t}, 0, \dots, 0, i_{m1}, i_{m2}, \dots, i_{mw}, 0)$ with 0 represents the AGVs depot. For example, the chromosome $(0, 1, 2, 3, 0, 4, 5, 6, 0, 7, 8, 0)$ express 3 AGVs serve 8 jobs, It indicates that AGV 1 serves $(0, 1, 2, 3, 0)$ in sequence, AGV 2 serves $(0, 4, 5, 6, 0)$ in sequence and AGV 3 serves $(0, 7, 8, 0)$ in sequence respectively. Roulette-wheel scheme is applied in the selection procedure. The method of generation of the initial population, the crossover operator and mutation operator in this paper are originated from literature [8].

2.3. Rolling Optimization Strategy for AGVs Scheduling

In AGVsS, we may know information about jobs' arrival in advance. Based on this information, we can use rolling optimization strategy to schedule AGVs. When an AGV starts to serve a job, it has to finish it. Cancellation of jobs is not allowed. Normally, there are two rolling optimization strategies, i.e. rolling by time and rolling by the number of jobs [2]. For the rolling by time, we schedule all known jobs during a time period H , Depending on different conditions, the number of scheduled jobs can differ significantly for the time horizon H . However, AGVs only follow the resulting schedule during a time period $h = \alpha H (\alpha < 1)$ After the time period h the system invokes the scheduling algorithm again to schedule all known jobs in the period $[h, h + H]$. The process stops when all jobs have been finished. Rolling optimization strategy by

the number of jobs is similar to the one rolling by time, the differences are rolling time period H , h replacing by the rolling number of the jobs M , m respectively (shown in Figure 2).

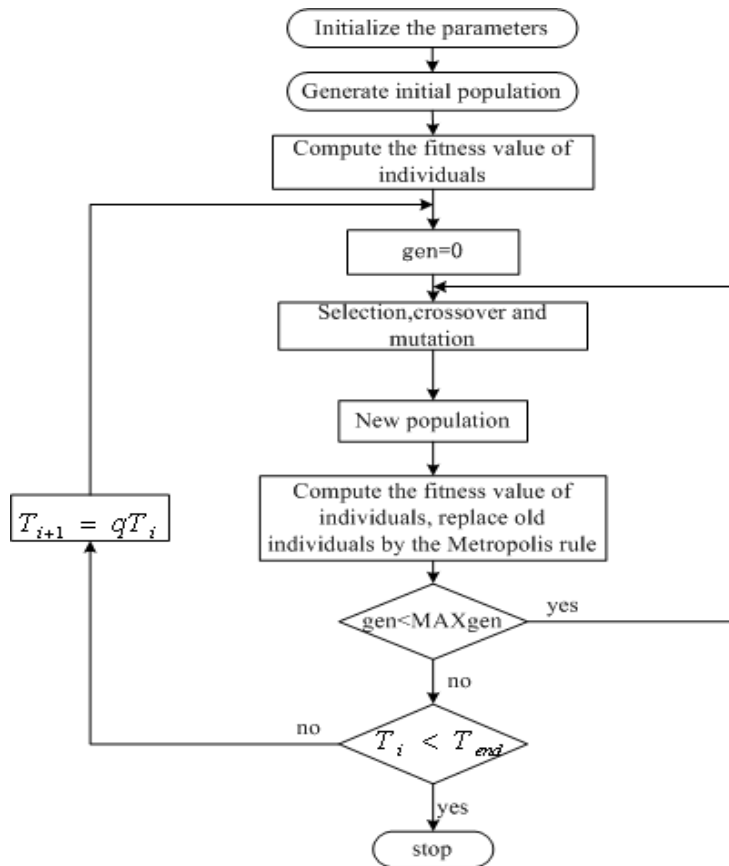


Figure 1. The framework of the annealing genetic algorithm (AGA)

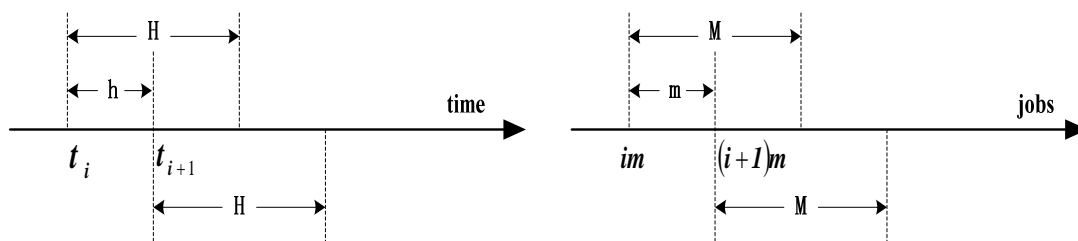


Figure 2. Rolling optimization strategies illustration (by time - left and by the number of jobs-right)

As shown in Figure 2, in rolling by time policy, the number of scheduled jobs at each step can differ significantly. When too many jobs are taken into account, the running time of the scheduling algorithm may increase significantly and may not meet the real time scheduling requirements. When we use rolling optimization strategy by the number of jobs, sometimes, the number of jobs we know in advance may be not enough. In this paper, we propose a combined rolling optimization strategy. When the number of jobs known in advance is sufficient ($i > M$), we apply the rolling optimization strategy by the number of jobs, otherwise rolling by time policy is used.

3. Results and Analysis

In this study, the layout we have selected is shown in Figure 3. We model the AGVsS in Plant 9 simulation software. In our experiment, the jobs' inter-arrival times for the loading stations are exponentially distributed with inter arrival times equal $(\tau_1, \tau_2, \tau_3, \tau_4, \tau_5) = (40, 50, 60, 70, 80)$ seconds. The probability that a load is sent from a loading station i to a unloading station j are p_{ij} ($p_{ij} = 1/n, n$ is the number of unloading stations). For the number of AGVs, 3 levels have been used (4,5,6). Information about jobs' arrivals is known in advance of 240 seconds. the speed of AGVs is 2 m/s. The parameters for the combined rolling strategy and the AGA: $H = 250, M = 25, \alpha = 0.8, popsize = 30, MAXgen = 200, p_c = 0.6, p_m = 0.05, T_0 = 1000, q = 0.9, T_{end} = 1$.

The main performance criterion is minimizing the average job waiting time. The secondary objective is minimizing the maximum job waiting time. Results of the experiments are shown in Table 1.

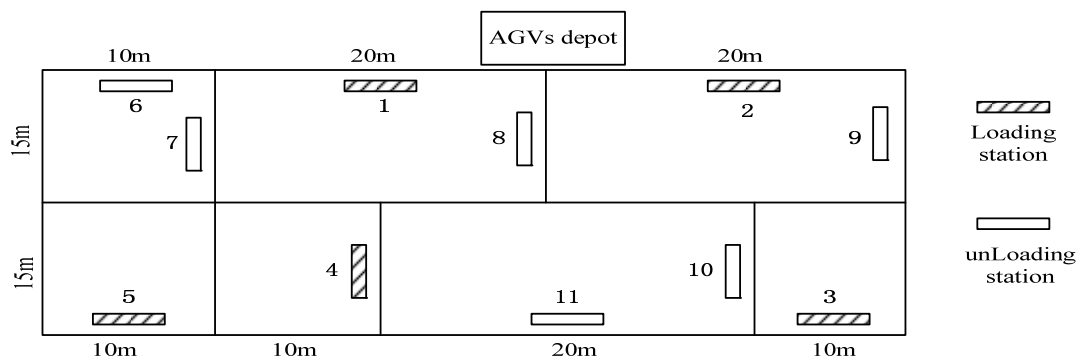


Figure 3. The experimental AGVsS layout

Table 1. Results of different scheduling strategies (\pm represents the 95% confidence interval)

No.AGVs	4			5			6		
Method	MOD-FCFS	NVF	ROLL-AGA	MOD-FCFS	NVF	ROLL-AGA	MOD-FCFS	NVF	ROLL-AGA
Aver-wait (s)	192.8 (± 5.6)	136.5 (± 3.8)	65.3 (± 2.2)	132.5 (± 3.1)	87.9 (± 2.5)	34.8 (± 1.8)	93.6 (± 2.7)	68.4 (± 2.3)	17.2 (± 1.1)
Max-wait (s)	958.4	1267.2	792.3	407.6	563.4	354.7	187.7	239.3	130.6

No. AGVs: number of AGVs; Aver-wait: average job waiting time; Max-wait: maximum job waiting time; MOD-FCFS: modified first come first served; NVF: nearest vehicle first; ROLL-AGA: Rolling optimization strategy based on annealing genetic algorithm.

Table 1 and Figure 4 show that the Modified First Come First Served (MOD-FCFS) is the worst strategy for the AGVsS in all three cases. The Nearest Vehicle First (NVF) performs a bit better than the MOD-FCFS. The average job waiting times and the maximum job waiting times obtaining by Rolling optimization strategy based on Annealing Genetic Algorithm (ROLL-AGA) are significantly smaller than the MOD-FCFS and the NVF strategy, which are among the best dispatching rules in literatures. The ROLL-AGA shows good performance for the AGVs scheduling and prove to be robust to different operating conditions.

In general, we have found that the dispatching rules such as MOD-FCFS and NVF don't perform well for the AGVsS. The dispatching rules dispatch AGVs only based on distance between the loads and the AGVs or the jobs' waiting time, the information about jobs arrivals in advance is not considered and increase the average and maximum job waiting times. The

AGVsS is a system of uncertainty, based on the information about jobs arrivals in advance, the proposed ROLL-AGA, which shares the advantages of both SA and GA, reduces the average and maximum job waiting times greatly and improve the system performance.

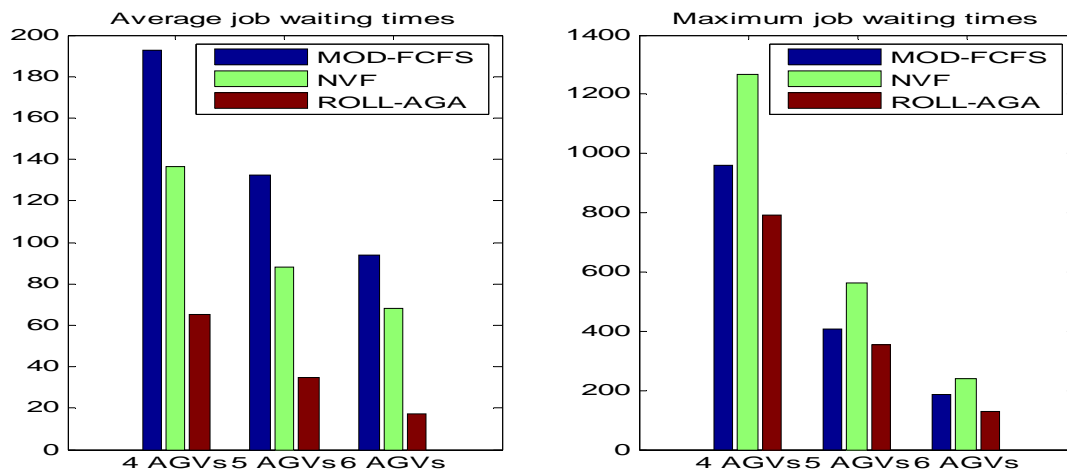


Figure 4. The performance of the three scheduling strategies

4. Conclusion

AGVsS is becoming popular in automatic materials handling systems, flexible manufacturing systems and even container-handling applications. AGVs scheduling is a crucial problem to improve the efficiency of the AGVsS. In this paper, we have discussed the AGVs scheduling issues in AGVsS. Using simulation, we prove that the MOD-FCFS rule and the NVF rule, both among the best dispatching rules in literature, performs not so well in simulation environment. They dispatching AGVs only based on distance between the loads and the AGVs or the jobs' waiting time, the information about load arrivals in advance is not used to optimize the performance of the AGVsS. Literature on external transport has shown that scheduling vehicles may lead to better performance than dispatching them. In this paper, we Apply this to the AGVsS, formulate the AGVsS scheduling problem as a m-TSPTW. AGVsS is an system with a high degree of uncertainty, We propose AGA to deal with the AGVs scheduling problem, and apply this heuristic dynamically by using it repeatedly under a combined rolling optimization strategy. We compare the performance of the proposed approach with the dispatching rules. Results show that the approach performs significantly better than the dispatching rules. From practical point of view, it makes the proposed scheduling approach more attractive for the AGVsS.

Acknowledgement

This work was supported by the international cooperation projects of China Ministry of science and technology (2010DFB70700).

References

- [1] Mantel RJ and Landeweerd HRA. Design and operational control of an AGV system. *International Journal of Production Economics*. 1995; 41(3): 257-266.
- [2] Le-Anh T. Intelligent Control of Vehicle-Based Internal Transport Systems. PhD thesis, Erasmus University Rotterdam. 2005.
- [3] Talbot L. Design and performance analysis of multi station automated guided vehicle systems. Ph.D Thesis, Universite Catholique de Louvain. 2003.
- [4] Le-Anh T and De Koster, R(M) BM. A review of design and control of automated guided vehicle systems. *European Journal of Operational Research*. 2006; 171(1): 1-23.
- [5] Egbelu PJ and Tanchoco JMA. Characterization of automated guided vehicle dispatching rules. *International Journal of Production Research*. 1984; 22(3): 359-374.

- [6] BL Golden and AA Assad. *Vehicle Routing: Methods and Studies*. North-Holland, Elsevier Science Publishers. 1988; 65-84.
- [7] Xu H, Chen ZL, Rajagopal S and Arunapuram S. Solving a practical pickup and delivery problem. *Transportation Science*. 2003; 37(3): 347-364.
- [8] Wenfeng Wang, Zuntong Wang, Fei Qiao. An Improved Genetic Algorithm for Vehicle Routing Problem with time window. *International Symposium on Computer Science and Computational Technology*, Shanghai. 2008; 189-194.
- [9] Cordeau, JF Gendreau, M Laporte, G Potvin, JY Semet F. A guide to vehicle routing heuristics. *Journal of the Operational Research Society*. 2002; 53(5): 512-522.
- [10] Victor Pillac, Michel Gendreau, Christelle Guéret, Andrés L. Medaglia. A review of dynamic vehicle routing problems. *European Journal of Operational Research*. 2012; 225(1): 1-11.
- [11] Vis IFA, De Koster, R(M) BM and Savelsbergh MWP. Minimum vehicle fleet size under time-window constraints at a container terminal. *Transportation Science*. 2005; 39(2): 249-260.
- [12] KC Tan, LH Lee, QL Zhu, K Ou, Heuristic methods for vehicle routing problem with time windows. *Artificial Intelligence in Engineering*. 2001; 15(3): 281-295.
- [13] Yi Zhang, Xiaoping Li, Qian Wang. Hybrid genetic algorithm for permutation flow shop scheduling problems with total flow time minimization. *European Journal of Operational Research*. 2009; 196(3): 869-876.
- [14] Chandra Sekhar Pedomallu, Linet Ozdamar. Investigating a hybrid simulated annealing and local search algorithm for constrained optimization. *European Journal of Operational Research*. 2008; 185(3): 1230-1245.