International Journal of Robotics and Automation (IJRA)
Vol. 6, No. 2, June 2017, pp. 69~79
ISSN: 2089-4856, DOI: 10.11591/ijra.v6i2.pp69-79

A Guided Ant Colony Optimization Algorithm for Conflict-free Routing Scheduling of AGVs Considering Waiting Time

LI Jun-jun[1], XU Bo-wei[2] , Yang Yong-sheng[3], Wu Hua-feng[4]
[1,4]Merchant Marine College, Shanghai Maritime University, Shanghai, China
[2,3]Institute of Logistics Science & Engineering, Shanghai Maritime University, Shanghai, China

Article Info

Article history:

Received Mar 29, 2017
Revised Apr 17, 2017
Accepted Apr 30, 2017

Keyword:

Ant colony
Automated guided vehicles
Conflict-free
Particle swarm optimization
Routing scheduling
Waiting time

ABSTRACT

Efficient conflict-free routing scheduling of automated guided vehicles (AGVs) in automated logistic systems can improve delivery time, prevent delays, and decrease handling cost. Once potential conflicts present themselves on their road ahead, AGVs may wait for a while until the potential conflicts disappear besides altering their routes. Therefore, AGV conflict-free routing scheduling involves making routing and waiting time decisions simultaneously. This work constructs a conflict-free routing scheduling model for AGVs with consideration of waiting time. The process of the model is based on calculation of the travel time and conflict analysis at the links and nodes. A guided ant colony optimization (GACO) algorithm, in which ants are guided to avoid conflicts by adding a guidance factor to the state transition rule, is developed to solve the model. Simulations are conducted to validate the effectiveness of the model and the solution method.

Corresponding Author:

LI Jun-jun,
Merchant Marine College,
Shanghai Maritime University, Shanghai, China.
Email: lijj@shmtu.edu.cn

1. INTRODUCTION

Automated guided vehicles (AGVs) form part of an unmanned transport system used for horizontal transportation tasks[1]. A number of AGVs present in the road network affects the effective AGV speed, and expected cycle time, which in turn affects the automated logistic system throughput. Furthermore, limited by the common road-type network, AGVs may congest or even collide with each other when too many AGVs are running along a narrow lane or passing some crossing roads[2]. The effect of vehicle congestion during internal transport could not be ignored because the corresponding throughput reductions were as large as 85%[3]. Therefore, AGV conflicts have been the most significant challenge that constrains the reliability, security, and efficiency of automated logistic systems[4]. The conflict-free routing scheduling problem (CFRSP) of AGVs, which is an important and fundamental problem in the management of AGV systems, has been investigated by a number of studies.

Many studies focus on route design and adjustment, which is a key problem in the conflict-free routing of AGVs. Researchers proposed a real-time traffic control scheme[5]. Specifically, they employed a k-shortest path search algorithm to construct a path set; thus, the online motion planning operation was performed in real time. Other workers presented a dynamic routing method for supervisory control of traveling AGVs within the layout of a given warehouse[6] and used time windows in a vector form to solve the shortest path problem dynamically. Hu et al. proposed a dynamic routing plan algorithm based on a time window[7]. Based on alternative paths and ideal time windows, their algorithm updated the time windows of lower-priority AGVs.

Typically, each AGV wishing to pass is required to book a passage time interval and a route. In order to avoid possible conflicts, AGVs may choose a waiting strategy such as deceleration and stopping except route adjustment. By changing the priority of AGVs passing through the nodes and adjusting the passing sequences of corresponding nodes, Qiao et al. proposed an updating AGV schedule to realize real-time conflict-free routing in a dynamic uncertain environment[8]. Shao et al. used a traffic controller to operate each moving AGV online after utilizing the A* algorithm to construct an optimal path set for AGV[9]. When the traffic controller operates, lower priority AGVs need to wait if their roads ahead are occupied by high-priority AGVs. Nishi et al. studied the optimization of conflict-free routing problem for AGVs with acceleration and deceleration[10].

Fazlollahtabar et al. proposed a mathematical program to minimize the penalized earliness and tardiness for conflict-free and just-in-time production, considering the due date of AGVs required for material handling among shops in a job shop layout[4]. Lu developed a combination of probabilistic and physics-based models for truck interruptions[11]. On the basis of exactly evaluating the expected link travel time, Miyamoto and Inoue solved a mixed-integer programming model by using a squeaky-wheel optimization based meta-heuristic to minimize the total expected travel time required to move containers around the yard. They also proposed local/random search methods to solve the dispatch and conflict-free routing problem of capacitated AGV systems[12]. However, the waiting strategies in their work were only treated as temporary measures to avoid conflicts. They did not consider a waiting strategy in the initial scheduling.

Different waiting strategies result in different running state and different productivity. It is better to design the route and waiting time together for AGVs in advance, rather than simply using waiting as a temporary measure when conflict happens. Zhou et al. proposed a conflict free Overhead Hoist Transporter (OHT) path scheduling method based on a rolling horizon strategy[13]. By executing space and time conflict detection for the current shortest path, they confirmed the conflict free path in the current time window by taking a corresponding collision avoidance strategy, which was less time consuming, and they also conducted event-driven rescheduling. However, their assumption was that only one OHT was allowed to run or stop at each node at the same moment, which limited its application range. Saidi-Mehrabad et al. proposed a two-stage ant colony optimization algorithm for a mathematical model composed of the job shop scheduling problem and conflict-free routing problem[14]. AGVs could move to nodes nearby or stay at the original node in the next time unit. However, the road network in reference [14] consisted of square grids, which was different from most of the actual situations.

After studying the current literature, it is clear that conflict-free routing scheduling of AGVs considering waiting time has received less attention from the research community. However, determining the route and waiting time simultaneously for AGVs in advance may reduce or even avoid conflicts in the road-type network with a greater accuracy. In this work, the AGVs CFRSP is regarded as a mixed combinatorial optimization problem composed of routing and waiting time optimizations. A guided ant colony optimization (GACO) algorithm is designed to optimize AGVs CFRSP. To avoid conflicts, the routes of AGVs are optimized by modified status transfer rule in which a kind of guidance factor is embedded; while the waiting time is optimized by the iterative rule of PSO. Several simulations show that the proposed model and method have strong rationality and applicability.

2. RESEARCH METHOD

The road network of an automated logistic system is denoted by a graph such as Figure 1 with N nodes (A1, A2, …, AN) and B links. There are P AGVs (AGV1, AGV2, …, AGVP). The starting node, finishing node, and speed of the pth AGV (denoted as AGVp) are Sp, Ep and Vp, respectively.

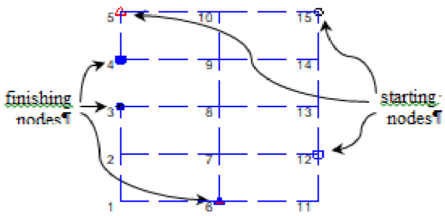


Figure 1. Road network

2.1. Travel time of AGVs

Assuming that AGV$_p$ passes through link (A_k, A_l), and nodes A_k and A_l are the i^{th} and $(i+1)^{th}$ nodes in its route (the starting node S^p is taken as the 1st node). The distance between nodes A_k and A_l is denoted by $d^p_{i,i+1}$. The waiting time of AGV$_p$ in front of nodes A_k and A_l are τ^p_i and τ^p_{i+1}, respectively. The time interval of AGV$_p$ passing through nodes A_k and A_l is t^p_k and t^p_l, respectively, as shown in Equation (1) and (2).

$$t^p_k = \sum_{j=1}^{i} \tau^p_j + \frac{1}{v_p} \sum_{j=1}^{i-1} d^p_{j,j+1} \tag{1}$$

$$t^p_l = \sum_{j=1}^{i+1} \tau^p_j + \frac{1}{v_p} \sum_{j=1}^{i} d^p_{j,j+1} \tag{2}$$

Obviously, there should be a safety distance between two AGVs for conflict prevention. Let the duration of AGV$_p$ passing through link (A_k, A_l) be $T^p_{k,l}$, which can be calculated according to Equation (3). In Equation. (3), $t*^p_k = t^p_k - \zeta$, $t*^p_l = t^p_l + \zeta$, and ζ is a constant more than zero to ensure the interval of keeping a safe distance among AGVs. If AGV$_p$ does not pass through link (A_k,A_l), let $T^p_{k,l}$ be $\varnothing$.

$$T^p_{k,l} = (t*^p_k, t*^p_l) \tag{3}$$

2.2. Link conflict

If $t*^p_k \in (T^q_{k,l} \bigcup T^q_{l,k})$, $W^{p,q}_{k,l} = 1$; else, $W^{p,q}_{k,l} = 0$. If $t*^p_l \in (T^p_{k,l} \bigcup T^p_{l,k})$, $W'^{p,q}_{k,l} = 1$; else, $W'^{p,q}_{k,l} = 0$. Where $T^q_{k,l}$ denotes the duration AGV$_q$ spends passing through link (A_k, A_l) from node A_k, and $T^q_{l,k}$ denotes the duration AGV$_q$ spends passing through link (A_l, A_k) from node A_l.

The maximum overlap number $W^{\max}_{k,l}$ for AGVs though a random link (A_k, A_l) is shown in Equation (4). Therefore, the number of AGVs travelling simultaneously in the link (A_k, A_l) is $W^{\max}_{k,l} + 1$.The number of running AGVs in link (A_k, A_l) needs to meet Equation. (5) to prevent link conflict. In Equation. (5), H_a is the allowed maximum number of running AGVs per unit distance.

$$W^{\max}_{k,l} = \max_{i \in [1,P]} \{\max\{\sum_{q=1}^{P} W^{p,q}_{k,l}, \sum_{q=1}^{P} W'^{p,q}_{k,l}\}\} \ (p \neq q) \tag{4}$$

$$\frac{W^{\max}_{k,l} + 1}{d_{k,l}} \le H_a \tag{5}$$

2.3. Node conflict

An AGV has a certain length, while a junction in the road network has some spatial scope. So an AGV needs some time to pass through a node. If $|t^p_k - t^q_k| < h_t$ (h_t is time threshold), AGV$_p$ and AGV$_q$ almost go through node A_k at the same moment, it is set $Z^{p,q}_k = 1$; otherwise, $Z^{p,q}_k = 0$. Then, the number of AGVs passing through node A_k simultaneously is shown in Equation. (6), and the maximum number of AGVs passing through node A_k simultaneously is shown in Equation. (7). In order to avoid node conflict, the number of running AGVs in a node needs to meet Formula (8). In Formula (8), H_b is the allowed maximum number of AGVs passing through a node simultaneously.

$$Z^{\#p}_k = \sum_{q=1}^{P} Z^{p,q}_k \ \ (p \neq q) \tag{6}$$

$$Z^{\max}_k = \max_{p \in \{1,\cdots,P\}} \{Z^{\#p}_k\} + 1 \tag{7}$$

$$Z_k^{\max} \le H_b \tag{8}$$

2.4. Conflict-free routing scheduling model for AGVs

Based on Equation. (1), the task completion time for AGV$_p$ is shown in Equation. (9). Each AGV is expected to reach the finishing node in the shortest time. Then the objective function is expressed in Equation. (10). Equation. (5) and (8) are the constraints for this conflict-free routing scheduling model. In Equation. (9) and (10), N_p is the number of nodes passed by AGV$_p$ including the starting and finishing nodes.

$$t_E^p = \sum_{j=1}^{N_p} \tau_j^p + \frac{1}{v_p} \sum_{j=1}^{N_p-1} d_{j,j+1}^p \tag{9}$$

$$\max f_p = t_E^p \tag{10}$$

3. GACO ALGORITHM

The AGVs CFRSP primarily consists of the route and waiting time decisions. The former is a discrete route optimization problem, while the latter is a continuous real number optimization problem. ACO is a meta-heuristic based global optimization method and has proved itself in the field of route optimization [15], while PSO exhibits good ability to solve the continuous optimization problem[16]. Therefore, GACO algorithm, in which ACO is integrated with PSO, is proposed to solve AGVs CFRSP. Route is optimized with the state transition rule of ACO, while the waiting time is optimized with the iterative rule of PSO. Besides, a type of guidance factor is added to the state transition rule to avoid conflicts among AGVs.

Firstly, the ant colony and particle swarm are initialized in Section 3.1; secondly, status transfer rule based on guidance factor, which can induce AGVs to avoid conflicts in links and nodes, is elaborated in Section 3.2; thirdly, fitness functions of single ant, historical optimal AGV group, historical individual and global best particles are given in Section 3.3; lastly, algorithmic flow of GACO is shown in Section 3.4.

3.1. Initialization

M ants are randomly set for each AGV. The starting node of initial route for each ant is S^p. The other nodes in set $\{A_1, A_2, \ldots, A_N\}$, are randomly disrupted to generate a sequence. An AGV at each link has the same pheromone intensity $\varphi_{kl}(p,0) = C$. The pheromone intensity for AGV$_p$ at link (A_k, A_l) in the t^{th} iteration is $\varphi_{kl}(p,t)$. For convenience, φ_{kl} is used to denote $\varphi_{kl}(p,t)$.

Meanwhile, M particles used to optimize waiting time are initialized. The number of particles is set equal to the number of ants. A particle is composed of the waiting time in front of nodes τ_i^p (i =1,2,…,N, p=1,2,……,P). Each particle is encoded as a $P*N$ matrix, $\begin{bmatrix} \tau_1^1 & \cdots & \tau_N^1 \\ \vdots & \ddots & \vdots \\ \tau_1^P & \cdots & \tau_N^P \end{bmatrix}$. Each element in this matrix is a random number in [0, $\tau_{\max}$]. Where, $\tau_{\max}$ is the maximum acceptable value of τ_i^p. In addition, let the initial and maximum velocities of each element in particles be v_0 and $v_{\max}$, respectively.

3.2. Status transfer rule based on guidance factor

3.2.1. Guidance factor

There are a large number of stochastic operations in the processes of GACO algorithm. In each generation of the algorithm, if AGVs are guided only according to the conflict analysis among the contemporary AGVs, there would be greater blindness. Contrary to AGVs in generations, historical optimal AGVs would gradually tend to the optimal solution and become steady. Therefore, AGVs are guided based on the conflict analysis among the contemporary AGVs and current historical optimal AGVs in this work, so as to enhance the target-oriented optimization ability of the algorithm.

a. Link conflict analysis considering current historical optimal AGVs

At the end of each generation, the historical optimal ant $AGV_p^g(p = 1, 2, \cdots, P)$, and the corresponding duration $T'^p_{k,l}$ and $T'^p_{l,k}$ spent by AGV_p^g passing through links (A_k, A_l) and (A_l, A_k) are recorded. The durations of ant AGV_p^m (the m^{th} ant for AGV_p, m=1,2,…,M) passing through link (A_k, A_l) and (A_l, A_k) are $T_{k,l}^{p,m}$ and $T_{l,k}^{p,m}$, respectively. If $T_{k,l}^{p,m} \bigcap (T_{k,l}^{g,p} \bigcup T_{l,k}^{g,p}) \neq \varnothing$, ants AGV_p^m and AGV_q^g pass through link (A_k, A_l) simultaneously, let $Y_{k,l}^{p,q,m} = 1$; otherwise, $Y_{k,l}^{p,q,m} = 0$. Further, the number of historical optimal ants passing through link (A_k, A_l) with ant AGV_p^m simultaneously, $NY_{k,l}^{p,m}$, is counted in Equation (11) .

$$NY_{k,l}^{p,m} = \sum_{q=1}^{P} Y_{k,l}^{p,q,m} \tag{11}$$

where, $q \neq p$. If only $AGV_{p,m}$ and $AGV_q^g (q \neq p)$ run in the road network, the AGV density at link (A_k, A_l) is:

$$\rho_{k,l}^{p,m} = \frac{NY_{k,l}^{p,m} + 1}{d_{k,l}} \tag{12}$$

According to formula (5), $\rho_{k,l}^{p,m}$ should meet inequation (13)

$$\rho_{k,l}^{p,m} \leq H_a \tag{13}$$

b. Node conflict analysis considering current historical optimal AGVs

Similar to the above-mentioned "Link conflict analysis considering current historical optimal AGVs", the node conflict is judged when each ant of $AGV_P (p = 1, 2, \cdots, P)$ passes through each node. Assuming that both $AGV_{p,m}$ and AGV_q^g ($q \neq p$) pass through node A_l, and the moment they pass through node A_l are t_l^p and t'^q_l. Similar with Section 2.2, it is set that $Z'^{p,q}_l = 1$ if $| t_l^p - t'^q_l | < h_t$; otherwise, $Z'^{p,q}_l = 0$. The number of historical optimal ants passing through node A_k with ant AGV_p^m simultaneously, NZ'^p_l, is counted in Equation. (14).

$$NZ'^p_l = \sum_{q=1}^{P} Z'^{p,q}_l \tag{14}$$

where, $q \neq p$. According to formula (8), NZ'^p_l should meet in Equation (15):

$$NZ'^q_l + 1 \leq H_b \tag{15}$$

c. Guidance factor

Each ant of AGV_p should consciously avoid the route of $AGV_q^g (q = 1, 2, \cdots, P, q \neq p)$. Here, a guidance factor σ_{kl}, which is used in the status transfer rule (in Section 3.2.2) to guide ants avoiding conflicts at links and nodes, is set in Equation. (16).

$$\sigma_{kl} = \frac{1}{(NY_{k,l}^{p,m} + 1)(NZ'^p_l + 1)} \tag{16}$$

3.2.2. Status transfer rule

a. Transition probability of basic ACO

The transition probability greatly affects the search in basic ACO. An ant chooses the next node according to pheromone intensity φ_{kl} and visibility η_{kl}. The transition probability ϑ_{kl} for an ant at node k to choose node j is shown in Equation. (17).

$$\vartheta_{kl}=\begin{cases}\dfrac{\varphi_{kl}^{\alpha}\cdot\eta_{kl}^{\beta}}{\sum\limits_{s\in allowed_k}\varphi_{ks}^{\alpha}\cdot\eta_{ks}^{\beta}}, & j\in allowed_k\\ 0, & otherwise\end{cases} \tag{17}$$

where $allowed_k$ is an optional node set. AGVs CFRSP is a kind of path planning problem [17] to find the shortest path from the starting node to the finishing node without requiring traversal of all the nodes. Every time an ant chooses the next node as close as possible to the finishing node. Here, the visibility factor is redesigned based on the A* algorithm:

$$\eta_{kl}=\frac{1}{d(k,l)+\sqrt{(x_l-x_{Ep})^2+(y_l-y_{Ep})^2}} \tag{18}$$

where, $d(k,l)$ denotes the distance between nodes A_k and A_l, (x_l, y_l) denotes the coordinate of node A_l, and (x_{Zp}, y_{Zp}) denotes the coordinate of finishing node E^p.

b. Status transfer rule based on guidance factor

On the basis of the guidance factor, a new transition probability is constructed in Equation. (19).

$$\vartheta_{kl}=\begin{cases}\dfrac{\varphi_{kl}^{\alpha}\cdot\eta_{kl}^{\beta}\cdot\sigma_{kl}^{\gamma}}{\sum\limits_{s\in allowed_k}\varphi_{ks}^{\alpha}\cdot\eta_{ks}^{\beta}\cdot\sigma_{ks}^{\gamma}}, & j\in allowed_k\\ 0, & otherwise\end{cases} \tag{19}$$

When $AGV_{p,m}$ is at node A_k, it would choose the next node. From Equation. (16) and (19), it can be seen that the transition probability of A_l is bigger if the number of $AGV_q^g(q\neq p)$ at link (A_k, A_l) and node A_l is larger, or vice versa. Then the guidance factor embedded in status transfer rule can reduce link and node conflicts efficiently.

Similar to the basic ACO, the status transfer rule shown in Equation. (20) is used to choose the next node A_{next}, where, λ is a random number uniformly distributed on the interval [0,1] and λ_0 is a parameter in [0,1]. J is a random variable selected according to the probability distribution given by Equation. (19).

$$A_{next}=\begin{cases}\arg\max\limits_{l\in allowed_k}\{\varphi_{kl}^{\alpha}\cdot\eta_{kl}^{\beta}\cdot\sigma_{kl}^{\gamma}\}, & \lambda\leq\lambda_0\\ J, & otherwise\end{cases} \tag{20}$$

3.3. Fitness function

3.3.1. Fitness function of single ant

In consideration of the waiting time, the total travel time of AGVs, rather than the total travel distance, is used in fitness function. A penalty function is set to punish link and node conflicts, and then the fitness of AGV_p is obtained by using Equation. (23). In Equation. (23), Γ is the penalty coefficient, and the second part to the right of the equal sign is the punishment term. The symbol ∑ means that all link and node conflicts have been punished.

$$f_{p,m}=t_E^{p,m}+\Gamma\cdot[\sum\max(0,\rho_{k,l}^{p,m}-H_a)+\sum\max(0,NZ'^{p}_{l}+1-H_b)] \tag{21}$$

3.3.2. Fitness calculation of historical optimal AGV group

If the current ant is the first ant of the first generation AGV_p, let this ant be AGV_p^g. Else, comparing the current ant with AGV_p^g, updating AGV_p^g once the current ant is more optimal. It can be seen that AGV_p^g of different AGVs are not updated simultaneously. Then the finesses of $AGV_p^g(p=1,2,\cdots,P)$ at the end of each generation are not reasonable if all $AGV_p^g\,(p=1,2,\cdots,P)$ are combined as a group of AGVs moving in the road network meantime.

Therefore, at the end of each generation, the link and node conflicts of each AGV_p^g are reanalyzed, and their fitness are recalculated. In this way, all $AGV_p^g\,(p=1,2,\cdots,P)$ can be treated as an AGV set with matched finesses.

For this AGV set, its fitness is calculated according to Equation (24) after synthetically considering the maximum and average values of AGV's fitness. In Equation. (24), f_p is also calculated according to Equation. (23). In each generation, the best AGV_p^g group in history is treated as the current historical optimal AGV group.

$$\min f = \max_{p\in\{1,2,\cdots,P\}} f_p + \underset{p\in\{1,2,\cdots,P\}}{mean} f_p \qquad (24)$$

3.3.3. Historical individual and global best particles

By the end of the first generation, waiting time of each ant is set as the historical individual best particle, and fitness of each ant is set as fitness of the historical individual best particle. From the second generation, an ant in each generation is compared with its historical individual best particle when it completes its route, and the historical individual best particle would be updated if the current ant is better.

For each generation, the historical best AGV_p^g of each AGV is re-evaluated according to Section 3.3.2. Then, the waiting time of AGV_p^g is set as the historical global best particle, while fitness of AGV_p^g is set as the fitness of historical global best particle.

3.4. Algorithmic Flow

The pseudo-code of the algorithm is as follows (*iter* is the number of iterative cycles, *Maxiter* is the maximum number of iterative cycles):

```
Initialization of ant colony and particle swarm in first iter.
For iter=1:Maxiter
        If iter>1
        The waiting time is calculated according to iterative rule of PSO.
    end
For p=1:P
                For m=1:M
                        Setting the first nodes for ants.
                        While the current node is not the finishing node
                  Choosing the next node A_next according to the status transfer
                  rule based on based on guidance factor.
             End While
                        Calculating fitness of ants.
                        Local update of pheromone.
                End For
        End For
        Update of historical optimal AGV group.
        Global update of pheromone.
        Update of historical individual best of particles.
        Update of historical global best of particles.
End For
End the optimization and output the results.
```

4. SIMULATIONS

4.1. Example 1

4.1.1. Problem description

Taking Figure 1 as an AGV road network example, the proposed method is verified by simulation. If there is a dotted line between any two points, the road between them is clear. Otherwise, there is no road, or an impassable road. The horizontal distance between the adjacent nodes is 1.8 units, and the vertical distance between the adjacent nodes is 1 units. The number of AGVs *P* is 3. The starting node, finishing node, and velocity of AGV are shown in Table 1. $\zeta = 0.1$, $H_a = 1$, $h_t = 0.2$, $H_b = 2$.

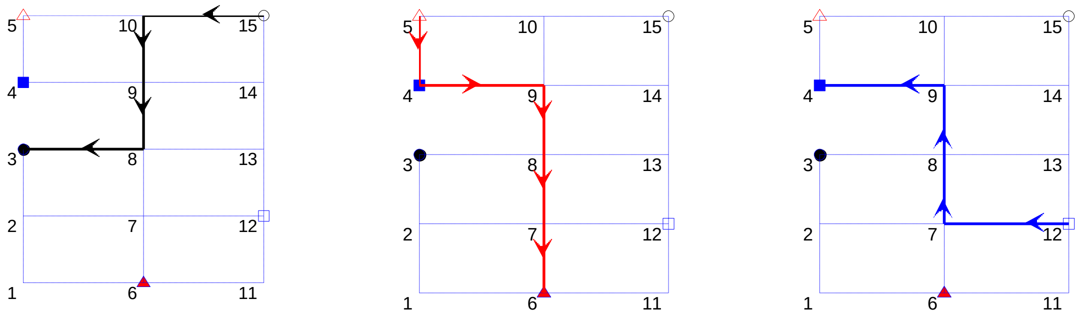
In Figure 1, the hollow and solid dots denote the starting and finishing points of AGV_1, respectively; the hollow and solid triangles denote the starting and finishing points of AGV_2, respectively; the hollow and solid squares denote the starting and finishing points of AGV_3, respectively. These three AGVs do not have different priorities.

Table 1 Starting nodes, finishing nodes, and velocities of AGVs

AGV	starting node	finishing node	velocity	departure time
1	15	3	1.1	0.7
2	5	6	0.9	0
3	12	4	1	1.5

4.1.2. Solution of basic ACO (BACO)

The basic ACO is used to solve CFRSP. *M* = 20, *Maxiter* = 100, $\alpha = 3$, $\beta = 5$, $\gamma = 3$, $\rho = 0.1$, $\lambda_0 = 0.1$. The routes attained by BACO are shown in Figure 2. More than two AGVs pass through nodes 4, 7, 8, and 9. The moment these three AGVs pass through these nodes is listed in Table 2 for the convenience of node conflicts analysis. The node orders of AGV_1, AGV_2, and AGV_3 passing through are 9→8, 4→9→8→7, and 7→8→9→4, respectively. Therefore, nodes are listed in accordance with the order 4→9→8→7.



(a) Route of AGV_1 (b) Route of AGV_2 (c) Route of AGV_3

Figure 2. Solution of basic ACO

Table 2. Moment for AGVs passing through 4, 9, 8, and 7 nodes

AGV	4	9	8	7
1		3.25	4.15	
2	1.11	3.11	4.22	5.33
3	7.10	5.30	4.30	3.30

Figure 2 shows that all these three AGVs pass through link (8,9) and nodes 8 and 9. Both AGV_2 and AGV_3 pass through link (7,8) and nodes 4 and 7. Therefore, links (8,9) and (7,8), nodes 8, 9, 4, and 7 are needed to be analyzed for conflicts. On the basis of the moment AGVs pass through the nodes in Table3, AGV_1 and AGV_2 are congested at link (8,9), while AGV_2 and AGV_3 are congested at link (7,8). From Equation (8), these three AGVs are congested at node 8. It can be seen that BACO cannot avoid the AGV conflict problem; therefore, it is not feasible.

4.1.3. Solution of time-window-based ACO (TACO)

The time window method assumes that the priorities of the three AGVs are gradually reduced. The parameter setting for TACO is the same as that of BACO. The routes attained by TACO are the same as those in Figure 2. The waiting time in front of nodes (“waiting time” for short) for AGVs is shown in Table 3, where ‘9(1.14)’in Table 3 denotes the waiting time in front of node 9 as 1.14 time unites. As for Table 3, nodes are listed in the order 4→9→8→7. The time the AGVs require to pass through these nodes is listed in Table 3.

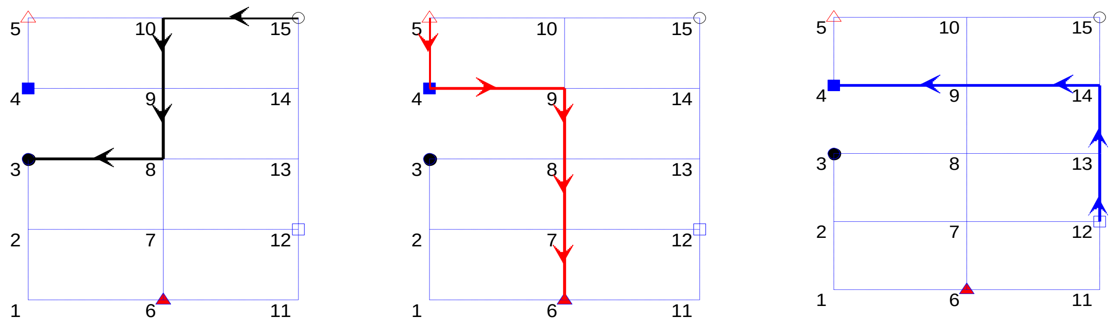
The results in Table 3 show that the waiting time in front of nodes 9 and 8 for AGV_2 and AGV_3 is 1.14 and 1.16 time unites, respectively. From Equation (5) and (8), it can be seen that there's no traffic conflict in each link and node. It is clear that TACO can avoid the AGV conflict problem; thus, it is feasible.

Table 3. Waiting Time for AGVs

AGV	Waiting time	4	9	8	7
1	0		3.25	4.15	
2	9(1.14)	1.11	4.25	5.36	6.47
3	8(1.16)	8.26	6.46	5.46	3.30

4.1.4 Solution of GACO

The parameter setting for GACO is the same as that of BACO. Besides, ω=0.7298, c_1= c_2=1.49618 in the iterative equation of particles. The routes attained by GACO are shown in Figure 3. There're more than two AGVs passing through nodes 4, 8, and 9. The orders of AGV_1, AGV_2, and AGV_3 passing through these nodes are 9→8, 4→9→8 and 9→4 respectively. Similar to Table 1, the waiting time of the AGVs and the moments they pass through nodes are shown in Table 4.



(a) Route of AGV_1 (b) Route of AGV_2 (c) Route of AGV_3

Figure 3. Solution of GACO

Table 4. AGVs Waiting Time and Moments Passing Through Nodes

AGV	Waiting time	8	9	4
1	9(0.98)	5.13	4.23	
2	0	4.22	3.11	1.11
3	0	0	5.3	7.1

Figure 3 shows that the routes of AGV_1 and AGV_2 are the same as those of BACO, while the route of AGV_3 is different from that of BACO. Then AGV_3 can avoid the congestion at link (7,8) and node 8. Nevertheless, both AGV_1 and AGV_2 pass through link (8,9) and node 8. Both AGV_2 and AGV_3 pass through link (9,4) and node 4. All AGVs pass through node 9. The results in Table 4 indicate that the waiting time in front of node 9 for AGV_1 is 0.98 time units. According to Equation. (5) and (8), there is no AGV conflict in each of these links and nodes.

4.1.5. Comparisons of these three methods

The above analysis reveals that TACO and GACO are superior to BACO for their conflict-free solutions. In the following, TACO, GACO, and BACO are compared from a travel time perspective. The time, average time, and maximum time of AGVs reaching the finishing node are calculated by these three methods. Bar graphs are used to compare these moments in Figure 4.

Figure 4 shows that the time at which AGV_1 reaches the finishing node in GACO is longer than that in TACO, whereas the time at which AGV_2 and AGV_3 reach the finishing node in GACO are shorter than that in TACO. Both the average time and maximum time at which AGVs reach the finishing node in GACO are shorter than that in TACO. Furthermore, the maximum time at which AGVs reach the finishing node in GACO is the same as that in BACO. Therefore, GACO is obviously superior to TACO.

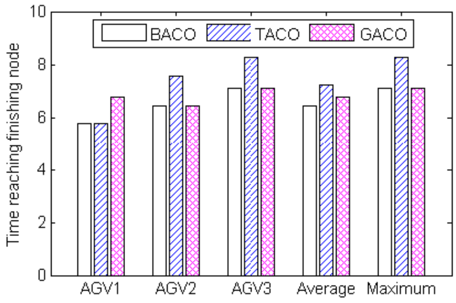


Figure 4 Time AGVs require to reach the finishing node

4.2. Example 2

In order to further verify the performance of GACO, BACO, TACO, and GACO are used to solve CFRSP for 12, 14, and 16 AGVs in an 8*12 road network. For the three problem sizes, starting nodes, finishing nodes, velocities and departure time are all randomly set. Velocities are limited in [0.8, 1.2], while departure time is limited in [0,5].

Similar to example 1, the time at which AGVs reach the finishing node is the earliest in BACO for discarding AGV conflicts. At the same time, only the result of BACO presents conflicts. The maximum time and average time of AGVs reaching the finishing node, and the number of link and node conflicts attained by BACO are shown in Figure 5. The maximum time and average time at which AGVs reach the finishing node attained by these three methods are plotted in Figure 6.

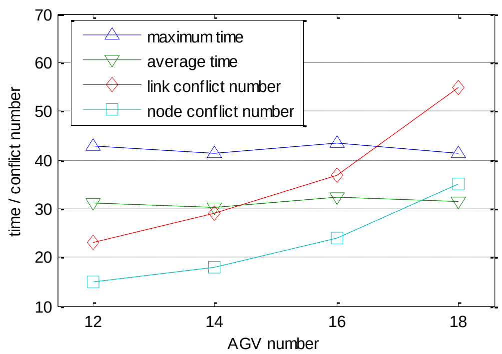


Figure 5. Result attained by BACO

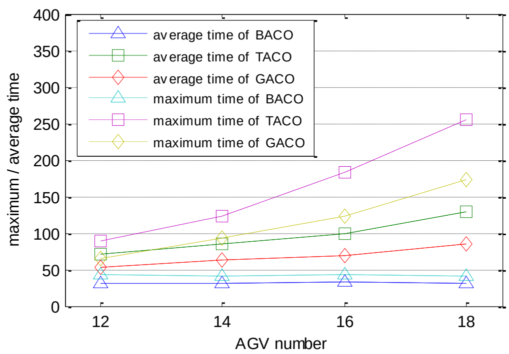


Figure 6. Comparison of time reaching the finishing node

Figure 5 shows the result attained by BACO. The solid lines with upper triangles, lower triangles, diamonds, and squares denote the maximum and average time at which AGVs reach the finishing node, number of link and node conflicts, respectively. Figure 5 shows that the maximum and average time are almost unaffected by the number of AGVs. However, as the number of AGVs increases, the number of link and node conflicts increase rapidly.

In Figure 6, the solid and dotted lines with upper triangles denote the maximum and average time at which AGVs reach the finishing node obtained by BACO, respectively. The solid and dotted lines with squares denote the maximum and average time at which AGVs reach the finishing node obtained by TACO, respectively. The solid and dotted lines with diamonds denote the maximum and average time at which AGVs reach the finishing node obtained by GACO, respectively. Figure 6 shows plots of the maximum and average time obtained by TACO and GACO increases with the number of AGVs, which is different from BACO. The maximum and average time obtained by GACO is shorter than that of TACO. Furthermore, the larger the number of AGVs is, the more obvious the advantage of GACO is. In summary, the GACO proposed in this work is feasible, and it outperforms BACO and TACO.

5. CONCLUSIONS

This work considers the waiting time of AGVs in front of nodes and build a conflict-free routing scheduling model for AGVs. A guided ant colony optimization algorithm is put forward to optimize the route and waiting time simultaneously. A type of guidance factor is designed and added in the status transfer rule to avoid conflicts based on conflict analysis among contemporary AGV and historical optimal AGVs. Several simulations with a different number of AGVs showed that the model and algorithm proposed in this work can effectively avoid conflicts and reduce the time at which AGVs reach the finishing node.

This work considered a road network of arbitrary structure, arbitrary departure time and velocities of AGVs, analyzed link and node conflicts, and optimized the route and waiting time simultaneously. The proposed method proved to have stronger applicability for AGVs CFRSP in different scenarios.

ACKNOWLEDGMENT

This work is supported by the Ministry of education of Humanities and Social Science project (No. 15YJC630145, 15YJC630059), Natural Science Foundation supported by Shanghai Science and Technology Committee (No. 15ZR1420200). Here we would like to express our gratitude to them.

REFERENCES

[1]. T. Miyamoto, K. Inoue. Local and random searches for dispatch and conflict-free routing problem of capacitated AGV systems. Computers & Industrial Engineering 2016, vol.91, pp.1-9.
[2]. M. Zhang, R. Batta, R. Nagi. Modeling of workflow congestion and optimization of flow routing in a manufacturing/warehouse facility. Management Science 2009, vol.55, pp.267-280.
[3]. D. Roy, A. Gupta, R. Koster. A non-linear traffic flow-based queuing model to estimate container terminal throughput with AGVs. International Journal of Production Research 2015, vol.54, no.2, pp.1-21.
[4]. H. Fazlollahtabar, M. Saidimehrabad, E. Masehian. Mathematical model for deadlock resolution in multiple AGV scheduling and routing network: a case study. Industrial Robot 2015, vol.42, no.3, pp.252-263.
[5]. J. H. Lee, B. H. Lee, M. H. Choi. A real-time traffic control scheme of multiple AGV systems for collision free minimum time motion: a routing table approach. IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans 1998, vol.28, no.3, pp.347-358.
[6]. N. Smolic-Rocak, S. Bogdan, Z. Kovacic, T. Petrovic. Time window based dynamic routing in multi-AGV systems. IEEE Transactions on Automation Science and Engineering 2010, Vol.7, No.1, pp.151-155.
[7]. [7] B. HU, B. WANG, C. X. WANG, M. YANG. Dynamic Routing of Automated Guided Vehicles Based on Time Window. Journal of Shanghai Jiaotong University 2012, vol.46, no.6, pp.967-971. (in Chinese)
[8]. Y. QIAO, X. M. QIAN, P. H. LOU. Improved time window based conflict-free automated guided vehicle system routing. Computer Integrated Manufacturing Systems 2012, vol.18, no.12, pp. 2683-2688. (in Chinese)
[9]. S. J. Shao, Z. Y. Xia, G. D. Chen, J. Zhang, Y. Hu, J. W. Zhang. A new scheme of multiple automated guided vehicle system for collision and deadlock free. IEEE International Conference on Information Science & Technology 2014, pp. 606-610.
[10]. T. Nishi, S. Matsushita, T. Hisano, M. Morikawa. A practical model of routing problems for automated guided vehicles with acceleration and deceleration. Journal of Advanced Mechanical Design, Systems, and Manufacturing 2014, vol.8, no.5, pp.1-12.
[11]. Z. Lu. Modeling of yard congestion and optimization of yard template in container ports, Transportation Research Part B 2016, vol.90, pp.83-104.
[12]. T. Miyamoto, K. Inoue. Local and random searches for dispatch and conflict-free routing problem of capacitated AGV systems. Computers & Industrial Engineering 2015, vol.91, pp.1-9.
[13]. B. H. Zhou, Q. Zhou, T. WANG. Conflict-free scheduling for united AMHS based on rolling horizon. Computer Integrated Manufacturing Systems 2014, vol.20, no.7, pp.1691-1699. (in Chinese)
[14]. M. Saidi-Mehrabad, S. Dehnavi-Arani, F. Evazabadian, V. Mahmoodiana. An Ant Colony Algorithm (ACA) for solving the new integrated model of job shop scheduling and conflict-free routing of AGVs. Computers & Industrial Engineering 2015, vol.86, pp.2-13
[15]. F. Zhou. Rolling Path Plan of Mobile Robot Based on automatic diffluent ant algorithm. International Journal of Robotics and Automation 2014, vol.3, no.2, pp.112-117.
[16]. Z. Allawi, T. Abdalla. A PSO-Optimized Reciprocal Velocity Obstacles Algorithm for Navigation of Multiple Mobile Robots. International Journal of Robotics and Automation 2015, vol.4, no.1, pp.31-40.
[17]. A. K. Dutta, S. K. Debnath, S. K. Das. Local Path Planning of Mobile Robot Using Critical-PointBug Algorithm Avoiding Static Obstacles. International Journal of Robotics and Automation 2016, vol.5, no.3, pp.182-189.