ACS-TS: TRAIN SCHEDULING USING ANT COLONY SYSTEM

KEIVAN GHOSEIRI AND FAHIMEH MORSHEDSOLOUK

Received 6 July 2005; Revised 15 January 2006; Accepted 18 January 2006

This paper develops an algorithm for the train scheduling problem using the ant colony system metaheuristic called ACS-TS. At first, a mathematical model for a kind of train scheduling problem is developed and then the algorithm based on ACS is presented to solve the problem. The problem is considered as a traveling salesman problem (TSP) wherein cities represent the trains. ACS determines the sequence of trains dispatched on the graph of the TSP. Using the sequences obtained and removing the collisions incurred, train scheduling is determined. Numerical examples in small and medium sizes are solved using ACS-TS and compared to exact optimum solutions to check for quality and accuracy. Comparison of the solutions shows that ACS-TS results in good quality and time savings. A case study is presented to illustrate the solution.

1. Introduction

In this section a hierarchical process of rail transport planning is introduced and then the ant's behavior which gives inspiration for ant algorithms is presented.

1.1. Rail transport planning. Rail transport planning is a very complex task which is carried out based on the mutual reaction among a large number of impressed components. As it was mentioned in Ghoseiri et al. [51] and Lindner [70], in respect to the complexity of rail transport planning, this process is divided into several steps. These procedures include the demand analysis, line planning, train scheduling, rolling stock planning, and crew management. Figure 1.1 shows this decomposition. The following is a brief description on the hierarchical planning process.

In the first step, the passenger demand is analyzed. As a result, the amount of passenger's demand between certain origins and certain destinations is determined. The line

Hindawi Publishing Corporation
Journal of Applied Mathematics and Decision Sciences
Volume 2006, Article ID 95060, Pages 1–28
DOI 10.1155/JAMDS/2006/95060

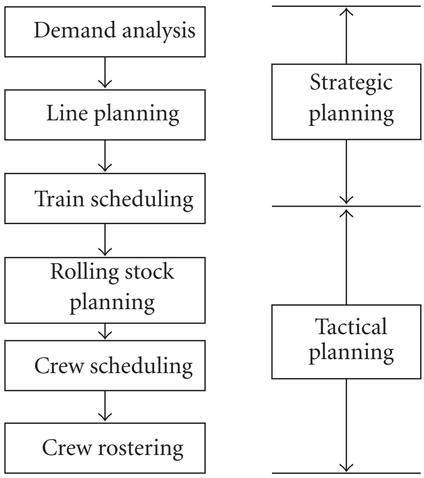


FIGURE 1.1. The hierarchical planning process in public rail transport (adapted from Ghoseiri et al. [51]).

planning includes decision making about routes and lines. This planning identifies which routes or lines should be exploited with what frequency. In the train scheduling phase, the arrival and departure times for all trains are determined. Determination of a timetable to separate the arrival and departure times of starting, ending, and middle stations is the product of this phase. In the next phase, the wagons and locomotives which are dedicated to the line are linked together to form a train. This phase is called rolling stock planning. The next task is the crew management. This task determines the distribution and allocation of the train's crew. This planning should be done in a way that supplies the necessary staff for each train. Crew management components include crew scheduling and crew rostering. Crew scheduling results in allocation of crews to trains and crew rostering determines their duty description. All of these phases have a close relationship. Computing an optimal solution in one phase may restrict the feasible solution space in the next phases.

Another classification was done by Assad [3]. Assad divided the planning process of rail transportation into strategic, tactical, and operational levels. This classification is shown in Table 1.1.

In the strategic planning level, some decisions are made about infrastructure investments. These decisions are long-term decisions, so they require greater costs. These decisions are greatly affected by political considerations. The infrastructure of the network develops in this phase. The analysis of passenger demand and the design of line plans also belong in this planning level. The tactical planning level is in fact the resource allocation phase. Most of line planning details and train schedule planning is done in this phase. Operational planning is just the day-by-day decisions. Here, due to unexpected events like breakdowns, special trains, or short-term changes in the infrastructure caused

Table 1.1. Planning levels (adapted from Assad [3]).

Planning stages	Time horizon	Objective
Strategic level	5–15 years	Resource acquisition
Tactical level	1–5 years	Resource allocation
Operational level	24 hours–1 year	Daily planning

by construction sites, certain parts of the schedule, rolling stock, or crew assignment patterns have to be rearranged. (For further study refer to Ghoseiri et al. [51] and Lindner [70].)

1.2. Ant's behavior. Special insects like ants, termites, and bees that live in a colony are capable of solving their daily complex life problems. These behaviors which are seen in a special group of insects are called swarm intelligence. Swarm intelligence techniques focus on the group's behavior and study the decartelized reactions of group agents with each other and with the environment. The swarm intelligence system includes a mixture of simple local behaviors for creating a complicated general behavior and there is no central control in it. Various types of certain ants have the ability to deposit pheromone on the ground and to follow, in probability, pheromone previously deposited by other ants. By depositing this chemical substance, the ants leave a trace on their paths. By detecting this trace, the other ants of the colony can follow the path discovered by other ants to find food. For finding the shortest way to get food, these ants can always follow the pheromone trails. (For further study refer to Fabinkue [42], Dorigo and Di Caro [34].) As was mentioned in Dorigo et al. [35], the ant algorithms based on this characteristic are inspired from Goss experiments, a laboratory colony of Argentine ants called Iridomyrmex Hmilis was placed in a closed space in which the nest was connected to food resource by a double bridge (with different length). This branched way was designed in a way that the ants could just choose one of the branches for reaching the food. After several times carrying out the experiment, the number of ants and amount of pheromone in each branch were counted and measured. It was also observed in this experiment that the possibility of choosing the shortest path increases with the length difference of two branches.

The reason for this behavior in ants is explained in the following form: in the beginning of the experiment, there is no pheromone in each branch. For this reason the ants choose one of the paths without any preferences and with an equal probability. So it can be expected that in the beginning of experiment half of the ants choose the longer branch and another half of ants choose the shorter branch. Because of shortness of one of the paths, the ants that have chosen the shorter path reach the food resource earlier and return to the nest. When these ants want to choose one of the ways to reach the food, the presence of pheromone in the shorter branch makes ants interested in choosing this branch. Therefore the amount of pheromone in this path increases more quickly and finally makes the majority of ants choose this path. Figure 1.2 shows the reason for this behavior in ants.

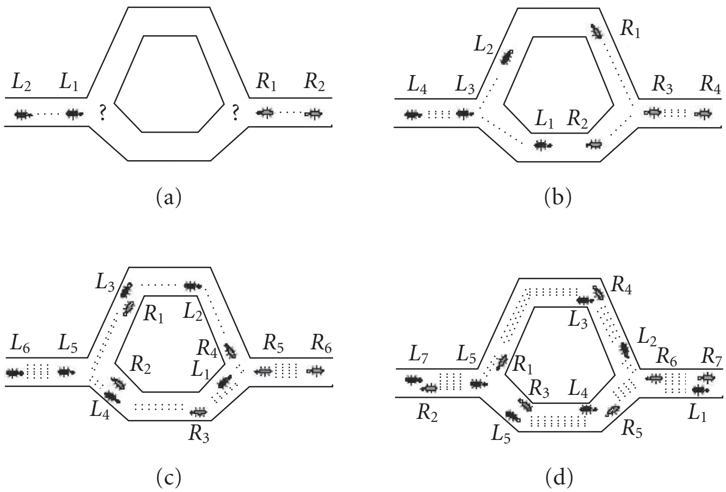


FIGURE 1.2. The ant's behavior: (a) the ants reach to the point of making a decision. (b) The ants choose one of the two paths randomly. (c) If the ants move with the same speed, the ants which have chosen the shorter path reach sooner to the point of making next decision. (d) The amount of pheromone in the shorter branch increases at a higher rate. (Adapted from Dorigo and Gambardella [36].)

2. Literature review

In this section, first there is a review on the literature of the train scheduling problem and then the manner of creating, developing, and applying the ant algorithms is put forward. The literature review of the train scheduling problem and the ant algorithms show that ant colony optimization algorithms currently are not used for solving the train schedule problem.

2.1. Train scheduling. The train schedule problem is one of the difficult problems in rail transport planning. This planning has been carried out manually and by trial and error methods for over a century. In a manual method, the train arrival and departure times from each station are identified based on the individual's experience and information. The solution quality and building time in this method are closely related to the individual's experience and ideas. (For a further study refer to Chiang et al. [20].)

Mathematical programming, simulation, expert systems, heuristic and metaheuristic methods, and combinational methods are other techniques for train scheduling. Mathematical methods give exact or optimal solutions. Examples of these methods include Frank [45], Amit and Goldfarb [1], Szpigel [104], Petersen [88], Chen and Harker [18], Keaton [64], Kraay and Harker [67], Lindner [70], Brodal and Jacob [14], and Ghoseiri et al. [51]. Although these techniques consistently find solutions with high quality, the time and memory used in these methods for solving realistically sized problems is very high. For these reasons, simulation, heuristic, metaheuristic methods and expert systems are typically used for solving these problems. (For a further study refer to Cordeau et al. [23].)

The application of simulation during the 1970s faced failure when solving the train scheduling problem. In these years, simulation had impractical application because of extra calculations and informational necessities. However, today computers can implement the simulation models much easier. Databases can be combined with other programs and this leads to a considerable improvement in simulation technology. There are several researches using simulations in the rail network literature; Peat, Marwick, Mitchell & Co. [87], Jovanovic and Harker [63], Dessouky and Leachman [28], Cheng [19], Higgins and Kozan [56].

Heuristic methods are not always able to give good solutions to problems but these algorithms may solve the problem in a shorter time. This property makes these algorithms play a more constructive part of the primary solutions for other algorithms. These algorithms are made based on the problem structure and have a different structure for each problem. These algorithms' applications to railway problems can be noted in Cai and Goh [16], Carey and Lockwood [17], and Higgins et al. [57].

Knowledge-based systems (expert systems) have typically been used to solve problems that are either too complex for a mathematical formulation or too difficult to be solved by optimization approaches. Some examples of application of the knowledge-based systems in railway transportation are Cury et al. [26], Araya and Abe [2], Iida [59], Komaya and Fukuda [65], Minton et al. [79], Zweben et al. [114]. These algorithms are considered as a subgroup of heuristic algorithms. (For a further study refer to Chiang et al. [20].)

Metaheuristics are in fact guide algorithms of heuristic algorithms. These algorithms use the heuristic parts and give them direction in the searches. In spite of that, the heuristic parts of these algorithms have a specific and fixed structure and they can be used for solving various problems with little changes. These algorithms are inspired by events of nature. Some of these algorithms include genetic algorithm, neural networks, immune system, tabu search, simulation annealing and ant colony optimization. Although the solution quality of these algorithms is high and produces solutions close to optimum, there is still little metaheuristic research on rail transport planning problems. As an example, Huntley et al. [58] developed a simulated annealing approach to train scheduling for CSX transportation. Van Wezel et al. [108] applied a genetic algorithm to improve train timetables. Martinelli and Teng [76] used neural networks for routing in a railway. Nachtigall and Voget [83] applied a genetic algorithm to solve train scheduling problems. Gorman [52] used a combination of a genetic algorithm and tabu search for addressing the weekly routing and scheduling problem. Pacciarelli and Pranzo [86] used tabu search to solve train scheduling problem. Kwan and Mistry [68] used a coevolutionary algorithm to create a train timetable. Sepehri [95] solved the crew planning problem in a railway by ant colony optimization. Engelhardt-Funke and Kolonko [41] used an advanced evolutionary algorithm to solve train scheduling problem. Dorigo and Gambardella [36], as it can be seen in Table 2.1, showed that the ACS algorithm has been more successful than the other metaheuristics in solving the TSP. In this table, for each of the problems tested, the best solution and its corresponding iteration number built using the metaheuristics is reported. Additionally, Fischetti et al. [43], Gutin and Punnen [54], and Noon and Bean [85] showed that the train scheduling problem can be easily transformed to a travel salesman problem. Therefore, considering the approach of transforming the train scheduling

TABLE 2.1. Comparison of metaheuristic algorithms (adapted from Dorigo and Gambardella [36]).

Problem name	SA	EP	GA	ACS	Optimal
TSP with 50 cities	443 68512	426 100000	428 2500	425 1830	425
TSP with 75 cities	580 173250	542 325000	545 80000	535 3480	535
TSP with 100 cities	NA —	NA —	21761 103000	21282 4820	21282

problem to a TSP problem, good responses can be expected from solving it using the ACS algorithm.

In this research, it is decided to solve the train scheduling problem by this algorithm based on the good results using the ACS algorithm to solve the TSP problem and also transforming capability of the train scheduling problem to a TSP.

2.2. Historical development of ant colony optimization. Ant algorithms are a population-based approach which has been successfully applied to several NP-hard combinatorial optimization problems. As the name suggests, ant algorithms have been inspired by the behavior of real ant colonies. One of the main ideas of ant algorithms is the indirect communication of a colony of agents, called (artificial) ants, based on pheromone trails (pheromones are also used by real ants for communication). The (artificial) pheromone trails are a kind of distributed numeric information which is modified by the ants to reflect their experience while solving a particular problem. The first ACO algorithm, called ant system (AS) has been applied to the traveling salesman problem (TSP) by Dorigo et al. [38]. In spite of hopeful results, the algorithm results were not comparable to the other advanced algorithms which were already applied to solve this problem. Despite the fact, this algorithm built important principles in creating more advanced algorithms. At the present time, many algorithms have been suggested based on the improvement of AS algorithm and used for solving various problems. A comprehensive list of ACO algorithms and their applications are shown in Table 2.2.

3. ACS compound model and train scheduling problem

Train scheduling is a combinatorial optimization problem. In this problem the aim is to determine the arrival and departure times from stations on which the train passes. This problem is known to be NP-hard. Because of the dimensions and natural complexity in mathematical models, traditional optimization techniques are not useful for solving the problem, and the exact methods are only usable with examples in small sizes. For solving the problem with real dimensions, the heuristic or metaheuristic methods should be used. In this research, the ACS algorithm is chosen as the metaheuristic method for solving the train scheduling problem.

Table 2.2. Ant algorithms and their applications.

Algorithm name	Developer(s)	Year	Problem	Reference
Ant system	Dorigo et al.	1991	Traveling salesman problem	[38]
	Forsyth and Wren	1997	Bus driver scheduling	[44]
	Nahas and Nourelfath	2005	Reliability optimization of a series system	[84]
AS-QAP	Maniezzo et al.	1994	Quadratic assignment problem	[75]
AS-JSP	Colorni et al.	1994	Job shop scheduling problem	[22]
Ant-Q	Dorigo, Gambardella	1997	Traveling salesman problem	[36]
ACS-3opt and ACS	Dorigo, Gambardella	1997	Traveling salesman problem	[36, 37]
ABC	Schoonderwoerd et al.	1996	Telecommunications networks	[94]
AS-VRP	Bullnheimer et al.	1997	Vehicle routing problem	[15]
MMAS	Socha et al.	2003	University course timetabling problem	[98]
HAS_QAP	Gambardella et al.	1999	Quadratic assignment problem	[48]
HAS_SOP	Gambardella and Dorigo	2000	Sequential ordering problem	[46]
AS-ATP	Costa and Hertz	1997	Graph coloring	[25]
ANTCOL	Costa and Hertz	1997	Graph coloring	[25]
AntNet & AntNet-FA	Di Caro and Dorigo	1997	Connectionless network routing	[29]
Regular ants	Subramanian et al.	1977	Routing in dynamic network	[103]
MMAS-QAP	Stutzle and Hoos	2000	Quadratic assignment problem	[102]
AS-QAP	Maniezzo and Colorni	1999	Quadratic assignment problem	[74]
ANTS-QAP	Maniezzo	1999	Quadratic assignment problem	[71]
	Solimanpur et al.	2004	Intercell layout problem in cellular manufacturing	[99]
AS-SCS	Michel and Middendorf	1999	Shortest super sequence problem	[78]
ASGA	White et al.	1998	Connection management	[111]
AntNet-FS	Di Caro and Dorigo	1998	Connection-oriented network routing	[30]
ABC-smart ants	Bonabeau et al.	1988	Connection-oriented network routing	[13]
CAF	Heusse et al.	1998	Routing networks	[55]
ABC-backward	Van der Put	1998	Routing in the faxfactory	[107]

Table 2.2. Continued.

Algorithm name	Developer(s)	Year	Problem	Reference
ACO	Stutzle	1998	Flow shop problem	[101]
	Bland	1999	Space-planning	[10]
	Doerner et al.	2003	Full truckload transportation	[33]
	Doerner et al.	2000	Pickup and delivery	[32]
	Jayaraman et al.	2001	Bioreactors optimization	[60]
	Bland	2001	Structural design problem	[11]
	Gravel et al.	2002	Scheduling continuous casting	[53]
	Roli et al.	2001	Constraint satisfaction	[92]
	Gamez and Puerta	2002	Best elimination sequence	[49]
	Eggers et al.	2003	Keyboard arrangement problem	[40]
	Shelokar et al.	2004	Clustering	[96]
	Gandibleux et al.	2004	Set packing problem	[50]
	Reimann and Laumanns	2005	Capacitated minimum spanning tree problem	[90]
	Lim et al.	2005	Bandwidth minimization	[69]
	Baykasoglu et al.	2005	Dynamic facility layout problem	[6]
AS(TS)	Bland	1999	Layout of facilities	[9]
Intelligent ant	Zhou and Liu	1998	Dynamic routing of telecommunication networks	[113]
MACS-VRPTW	Gambardella et al.	1999	Vehicle routing problem	[47]
ACSp	Bianchi et al.	2004	Probabilistic traveling salesman problem	[8]
API	Monmarché et al.	2000	Numeric optimization	[80]
BWAS	Cordon et al.	2000	Traveling salesman problem	[24]
Painter ants	Tzafestas	2000	Digital art	[106]
CACO	Jayaraman et al.	2000	Design and scheduling of batch plants	[61]
	Vijayakumar et al.	2003	Multipass turning operations	[109]
Cognitive map	Ramos and Almeida	2000	Image segmentation-pattern reorganization	[89]
ANTS	Maniezzo and Carbonaro	2000	Frequency assignment problem	[72]
	Maniezzo et al.	2001	Data warehouse logical design	[73]
	Montemanni et al.	2002	Minimum-span frequency assignment	[82]
AS-VRPB	Wade and Salhi	2004	Vehicle routing problem	[110]
ACSA	Yu and Song	2001	Short-term schedule of thermal units	[112]

Table 2.2. Continued.

Algorithm name	Developer(s)	Year	Problem	Reference
AntNet routing	Barán and Sosa	2001	Data networks routing	[5]
AC^2	Cicirello	2001	Shop floor routing	[21]
Anthill	Baboglu et al.	2001	Peer-to-peer (P2P) networks	[4]
Multiple ant colony	Jong and Wiering	2001	Bus stop allocation problem	[62]
	Bell and McMullen	2004	Vehicle routing problem	[7]
Parallel ant colonies	Talbi et al.	2001	Quadratic assignment problem	[105]
Ant heuristic	McMullen	2001	JIT sequencing problem	[77]
Ant-TDVRP	Rizzoli et al.	2002	Vehicle routing problem	[91]
ACS-DVRP	Montemanni et al.	2002	Dynamic vehicle routing problem	[81]
ACO-B	De Campos et al.	2002	Learning Bayesian networks	[27]
Multilevel ant-colony	Korosec et al.	2004	Mesh-partitioning problem	[66]
Pareto ACO	Doerner et al.	2005	Project portfolio selection	[31]
Population-based	Scheuermann et al.	2004	Field-programmable gate arrays	[93]
CIAC	Dréo and Siarry	2004	Optimization of multiminima continuous functions	[39]
RPACO	Shi et al.	2004	Unit commitment with probabilistic spinning reserve	[97]
Beam-ACO	Blum	2005	Open shop scheduling	[12]
Ant algorithm	Solimanpur et al.	2005	Layout problem in flexible manufacturing systems	[100]

3.1. Ant colony system (ACS). ACS was suggested as a new heuristic method to solve optimization problems by Dorigo and Gambardella [36, 37]. The reformed form of the AS algorithm and functions is as follows.

Each ant generates a complete solution by choosing the nodes according to a probabilistic state transition rule. The state transition rule given in (3.1) and (3.2) is called a pseudorandom-proportional rule:

$$s = \begin{cases} \arg\left[\operatorname{Max}_{j\in N_i^k}\{[\tau_{ij}][\eta_{ij}]^\beta\}\right] & \text{if } q \le q_0, \\ S & \text{if } q > q_0, \end{cases} \tag{3.1}$$

$$p_{ij}^k = \frac{[\tau_{ij}][\eta_{ij}]^\beta}{\sum_{l\in N_i^k}[\tau_{ij}][\eta_{ij}]^\beta}, \tag{3.2}$$

where q is a random number uniformly distributed in $[0 \quad \cdots \quad 1]$, q_0 is a parameter between 0 and 1, S is a random variable selected according to the probability distribution given in (3.2), τ_{ij} is the amount of pheromone in edge ij, $\eta_{ij} = 1/\delta_{ij}$ where δ_{ij} is the cost of edge ij, β is a parameter that determines the relative importance of η versus τ, and

```
Procedure Ant colony system
Set pheromone trails to small constant
While (termination condition not met) do
        Place each ant on initial node
        For i = 1 to n do (# ants)
             For k = 1 to m do (#locations)
                  Apply State Transition Rule (pseudorandom proportional)
                  Apply Local Update pheromone
             End for (build one route)
        End for (run one set)
        Apply Global Update
End while
End Ant colony system
```

Algorithm 3.1. ACS algorithm procedure.

N_i^k is the remaining node set of ant k based on moving from node i to build a feasible solution.

In ACS, only the globally best ant which has built the best solution deposits pheromone in the graph. At the end of an iteration of the algorithm, once all the ants have built a solution, pheromone is added to the arcs used by the ant that found the best tour from the beginning of the trial. This updating rule is called the global updating rule of pheromone:

$$\tau_{ij} \longleftarrow (1-\rho)\tau_{ij} + \rho\Delta\tau_{ij}, \tag{3.3}$$

where $0 < \rho < 1$ is a pheromone decay parameter and $\Delta\tau_{ij}$ equals to

$$\Delta\tau_{ij} = \begin{cases} \dfrac{1}{\cos t_{\mathrm{gb}}} & \text{if } (i,j) \in \psi^{\mathrm{gb}}, \\ 0 & \text{if } (i,j) \notin \psi^{\mathrm{gb}}, \end{cases} \tag{3.4}$$

ψ^{gb} is the best solution which was built and $\cos t_{\mathrm{gb}}$ is the cost of the best solution.

In ACS, ants perform step-by-step pheromone updates using local updating rule of pheromone. These updates are performed to favor the emergence of other solutions than the best so far. The updates result in step-by-step reduction of the pheromone level of the visiting edges by each ant. The local updating rule of pheromone is performed by applying the rule

$$\tau_{ij} \longleftarrow (1-\xi)\tau_{ij} + \xi\tau_0, \tag{3.5}$$

τ_0 is a small fixed value and $0 < \xi < 1$ is the local evaporation coefficient of pheromone. The ACS's overall structure is shown in Algorithm 3.1.

3.2. The proposed mathematical model of train scheduling. In this section a mathematical model for train scheduling on a single track line is presented. This model is the work done by Higgins and Kozan [56] with minor changes in order to account for the assumptions of the model. In this model it is supposed that the trains are only dispatched from the first and last station. After preparation, the trains in the beginning or end stations should be dispatched immediately. In the case that the prepared trains to dispatch are stopped in the stations with unpermitted time stop and go over the allowed time, we undergo some cost. In this model, the speed and trip times in each track section for each train are assumed to be fixed. Also, a train can travel in two directions, but it is not permitted to overtake another train. (For further study refer to Higgins and Kozan [56].)

3.2.1. Notations.

R: the group of trains that should be dispatched from right station to left.

L: the group of trains that should be dispatched from left station to right.

T: the group of total trains ($i, j \in R$ or L or T and $T = R \cup L$).

S: set of stations ($k \in S$), track sections and stations are indexed in numerical order from left to right.

Track section k is a section of track that connects two stations k and $k+1$.

D: the set of permitted stop times in the station ($d_{ik} \in D$).

AD: the set of arrival and departure times from a station ($Xa(i,k), Xd(i,k) \in \mathrm{AD}$).

M: a big positive number.

3.2.2. Parameters.

Trip time: the time that train i needs to pass track section $k \cdot (t_{ik})$.

Dwell time: this time indicates the permitted dwell time of train i in station $k \cdot (d_{ik})$.

Headway: minimum time interval between trains i and j to arrive/depart from track section $k \cdot (h_{ijk})$.

Train importance weight: (W_i).

3.2.3. Decision making variables

Binary variables.

$$a_{ij} = \begin{cases} 1 & \text{if train } j \in R \text{ enters the track section after train } i \in R, \\ 0 & \text{otherwise,} \end{cases}$$

$$b_{ij} = \begin{cases} 1 & \text{if train } j \in L \text{ enters the track section after train } i \in L, \\ 0 & \text{otherwise,} \end{cases} \tag{3.6}$$

$$c_{ijk} = \begin{cases} 1 & \text{if train } j \in L \text{ enters the track section } k \text{ after train } i \in R, \\ 0 & \text{otherwise (i.e., train } i \in R \text{ enters the track section } k \text{ after train } j \in L\text{).} \end{cases}$$

Continuous variables.

Xa(i, k): the arrival time of train i to station k.

Xd(i, k): the departure time of train i from station k.

3.2.4. Objective function. Objective function in this model is to minimize the total train delays in the stations. The delay equals the time difference between the amounts of time a train is stopped and its permitted dwell time in the station

$$\text{Min}\, z = \sum_{i \in T} \sum_{k \in S} W_i \left(Xd(i,k) - Xa(i,k) - d_{ik} \right). \tag{3.7}$$

3.2.5. Constraints. Trip-times constraints of dispatched trains from the right station:

$$Xa(i,k) - Xd(i,k-1) = t_{ik}, \quad i \in R,\ k \in S. \tag{3.8}$$

Trip-times constraints of dispatched trains from left station:

$$Xa(i,k-1) - Xd(i,k) = t_{ik}, \quad i \in L,\ k \in S. \tag{3.9}$$

Stop-times constraints of dispatched trains from left and right stations:

$$Xd(i,k) - Xa(i,k) \geq d_{ik}, \quad i \in T,\ k \in S. \tag{3.10}$$

Sequence constraints of dispatched trains from right station:

$$\begin{gathered} Xd(j,k-1) - Xa(i,k) \geq h_{ijk} - M(1 - a_{ij}), \quad i,j \in R,\ k \in S, \\ Xd(i,k-1) - Xa(j,k) \geq h_{ijk} - M a_{ij}, \quad i,j \in R,\ k \in S. \end{gathered} \tag{3.11}$$

Sequence constraints of dispatched trains from left station:

$$\begin{gathered} Xd(j,k) - Xa(i,k-1) \geq h_{ijk} - M(1 - b_{ij}), \quad i,j \in L,\ k \in S \\ Xd(i,k) - Xa(j,k-1) \geq h_{ijk} - M b_{ij}, \quad i,j \in L,\ k \in S. \end{gathered} \tag{3.12}$$

Safety constraints that ensure no collision occurs between two trains of opposite directions:

$$\begin{gathered} Xd(i,k) - Xa(j,k) \geq h_{ijk} - M c_{ijk}, \quad i \in R,\ j \in L,\ k \in S \\ Xd(j,k+1) - Xa(i,k+1) \geq h_{ijk} - M(1 - c_{ijk}), \quad i \in R,\ j \in L,\ k \in S. \end{gathered} \tag{3.13}$$

3.3. The solution method of proposed model using ACS. In the proposed algorithm, it is supposed that the trains play the role of cities (nodes) in the TSP. The dispatched trains from left to right and also dispatched trains from right to left form two independent subnetworks of the TSP. According to the definition, selected path of each ant in the trains' network indicates the sequence of train dispatching.

For instance, in Figure 3.1 which includes 7 trains (3 dispatching trains from right to left and 4 dispatching trains from left to right), if an ant chooses the path from the start node of train 1, train 3, and train 2 it means the dispatching sequence is trains 1, 3, 2.

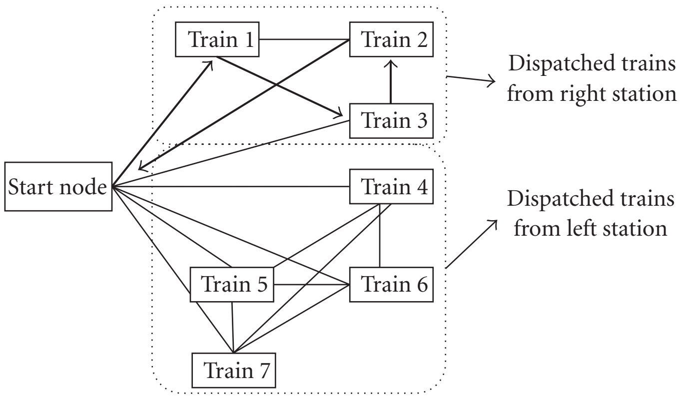


FIGURE 3.1. Problem's graph for the example of seven trains.

In this algorithm, a colony consists of $2 \times n$ ants where n is number of the nodes (trains) of the TSP. The ants are allocated in n groups. One of the ants of each group builds the sequence of dispatched trains from the right to left station and another ant is allocated to build the sequence of dispatched trains from left to right.

At first, both ants are placed at the figurative node of zero (the start node). Then, one of the ants is chosen from the first group randomly. The first chosen ant chooses a train in its train group by using the pseudorandom-proportional rule (3.1), (3.2). The arrival and departure times of the train from start station to the final station are calculated. Then another ant chooses its train which goes the opposite direction. The arrival and departure times of this train from each station are determined in regard to reconciliation of any collision incurred with the opposite train. In the case that the obtained times are true in (3.14), a collision occurs if the chosen train is the dispatched train from the left station:

$$Xa(i,k) + h_{ijk} > Xd(j,k), \qquad Xd(i,k-1) - h_{ijk} < Xd(j,k-1). \tag{3.14}$$

In this equation j is the selected left station train, i indicates the chosen right station train from the group of dispatched trains, and k is a track section in which the collision occurred. In this case for collision resolution between two trains, the departure time of the chosen train from the related station is changed as follows:

$$Xd(j,k) = Xa(i,k) + h_{ijk}. \tag{3.15}$$

The arrival and departure times of this train to its last station is calculated based on this time.

In the case that obtained times are true in (3.16), therefore, a collision occurs if the chosen train is the dispatched train from right station,

$$Xa(i,k-1) + h_{ijk} > Xd(j,k-1), \qquad Xd(i,k) - h_{ijk} < Xd(j,k). \tag{3.16}$$

In this equation, j is the selected right station train, i indicates the chosen train from the group of dispatched trains from left station, and k is a track section in which the collision occurred. In this case for resolution of collision between two trains, departure time of the above selected train from related station changes as follows:

$$Xd(j,k-1) = Xa(i,k-1) + h_{ijk}. \tag{3.17}$$

Once collision has been reconciled the chosen trains are omitted from the set of trains. Then randomly an ant is selected again. This ant chooses a train from its group. The arrival and departure times of this train are identified with its chosen sequence in its group. When the arrival and departure times from a section were identified, the collision condition of chosen train with dispatched chosen train in opposite direction is checked. In the case of collision, it is removed. This operation continues in the same way so that all the arrival and departure times from all stations are identified and there are not any collisions in the sections. Then the next train is chosen by other ants. This procedure continues until ants choose all the trains of their own group. (Refer to Figure 3.2.)

4. Analysis of the model

To analyze the solution results obtained from ACS-TS, they are compared with those of exact optimization method of the train scheduling model. For this purpose, computations are carried out for 45 problems including 3 to 8 trains and 2 to 8 track sections. The headway and dwell times are, respectively, considered 0.3 and 0.1 time units for all the trains and stations. The trip times are considered as a randomly selected number in the range of 2 to 15 time unit. For the created problem set, according to Dorigo and Gambardella [36, 37], the initial values for the global evaporation coefficient of pheromone, local evaporation coefficient of pheromone, pheromone initial amount on edges, and ACS parameter are, respectively, set to 0.1, 0.1, 0.000005, and 0.9.

Figure 4.1(a) shows sensitivity of the run times with respect to variation of number of track sections for solving the problems with exact algorithms and ACS-TS. In a similar manner, the sensitivity of the run times with respect to variation of number of trains for solving the train scheduling problems have been shown in Figure 4.1(b). Table 4.1 shows the results in more detail.

The time function of solving the problems with exact methods in relation to number of trains is obtained using MATLAB software and the results are completely an indictor of the NP complexity of the problem

$$\text{time}(s) = 0.01181e^{1.467\times \text{number of trains}}, \tag{4.1}$$

while the time function of solving by ACS appears to a linear function in the range studied,

$$\text{time}(s) = 1.702 \times \text{number of trains} - 2.738. \tag{4.2}$$

Similarly, the time functions of solving the problems with the exact and ACS methods in relation to the number of track sections are obtained. The exact methods (4.3) show

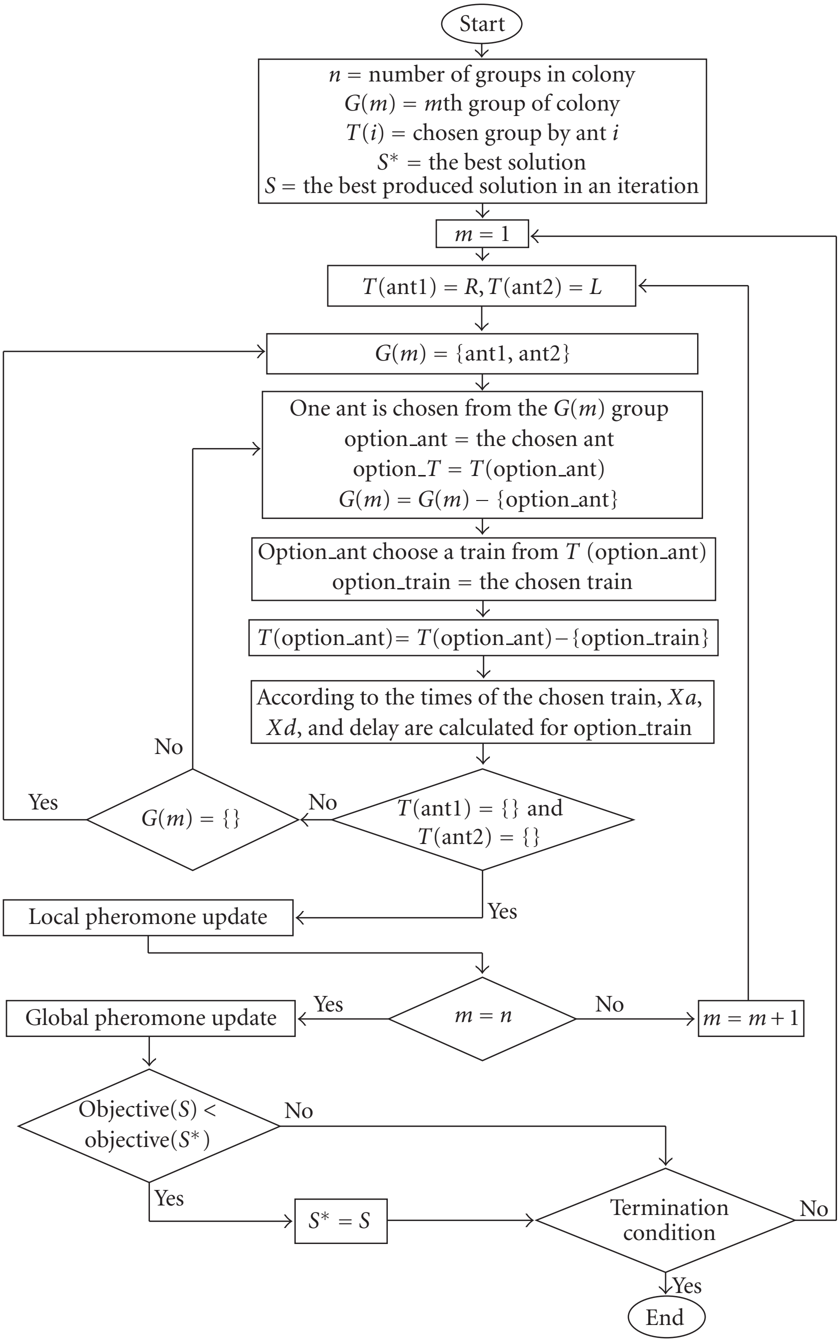


FIGURE 3.2. ACS-TS algorithm flowchart.

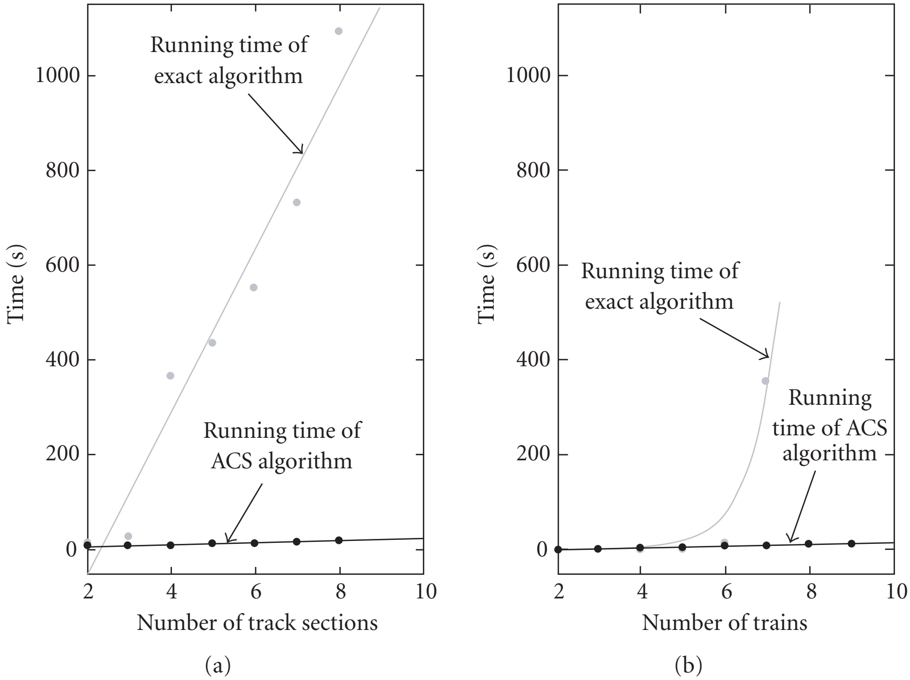


FIGURE 4.1. Run times of solving the train scheduling problems.

a fast increasing time function in comparison to the ACS method (4.4)

$$\text{time}(s) = 172.6 \times \text{number of track sections} - 403.6, \tag{4.3}$$

$$\text{time}(s) = 1.75 \times \text{number of track sections} + 3.107. \tag{4.4}$$

It was significant that in the created set of 45 problems the overall delay amount in dispatching trains from both methods was almost equal. However, the proposed ACS method showed considerable time savings in comparison to the exact solution method.

5. The case study

In this section, to clarify the use of the proposed algorithm, a problem with 30 trains and 4 track sections is solved.

5.1. Determination of ACS algorithm parameters. At first, according to Dorigo and Gambardella [36, 37] the initial values for parameters are set to the following values:

(i) global evaporation coefficient of pheromone, $\rho = 0.1$;
(ii) local evaporation coefficient of pheromone, $\xi = 0.1$;
(iii) pheromone initial amount on edges, $\tau_{ij} = 0.000005$ for all i and j;
(iv) ACS parameter, $q_0 = 0.9$.

TABLE 4.1. Comparison of the results of the proposed algorithm with exact solutions.

Problem	# Trains	# Eastbound trains	# Westbound trains	# Track sections	Time		Solution	
					ACS	Exact	ACS	Exact
1	3	2	1	2	0	0	5.3	5.3
2	4	3	1	2	0	0	16.9	16.9
3	5	3	2	2	1	1	35	33.7
4	5	2	3	2	1	7	34.8	31.9
5	6	3	3	2	4	2	50.7	49.4
6	7	3	4	2	7	20	80.3	76.9
7	7	4	3	2	7	22	82.6	76.6
8	8	5	3	2	8	*	124	*
9	8	4	4	2	8	*	123.8	*
10	8	3	5	2	8	*	122.3	*
11	3	2	1	3	0	0	3.9	3.9
12	4	3	1	3	1	2	14.1	14.1
13	5	3	2	3	2	1	35.1	32
14	5	2	3	3	2	1	30.5	29.3
15	6	3	3	3	7	4	50.3	50.3
16	7	3	4	3	9	35	82.4	78
17	7	4	3	3	9	47	82.8	78.5
18	8	5	3	3	10	*	121.6	*
19	8	4	4	3	10	*	114.2	*
20	8	3	5	3	10	*	116.2	*
21	3	2	1	4	2	0	6.6	6.6
22	4	3	1	4	5	0	14.2	14.2
23	5	3	2	4	6	1	32	32
24	5	2	3	4	6	5	35	31
25	6	3	3	4	8	9	52.6	52.6
26	7	3	4	4	9	1495	78.3	77.2
27	7	4	3	4	9	97	83.6	81.1
28	8	5	3	4	11	*	121.1	*
29	8	4	4	4	11	*	124.3	*
30	8	3	5	4	11	*	125.3	*
31	3	2	1	5	3	0	3.9	3.9
32	4	3	1	5	5	3	12.8	12.8
33	5	3	2	5	8	1	27.8	27.8
34	5	2	3	5	8	6	31.8	30.3
35	6	3	3	5	11	17	46.7	46.7
36	7	3	4	5	12	1648	71.5	70.7
37	7	4	3	5	12	172	79.5	73.6
38	8	5	3	5	15	*	121.6	*
39	8	4	4	5	15	*	99.7	*
40	8	3	5	5	15	*	121	*
41	7	4	3	6	15	301	60.8	58.7
42	7	2	5	6	13	87	70.4	60.7
43	7	3	4	7	15	87	80.9	77.3
44	7	5	2	7	15	901	75.7	72.8
45	7	4	3	8	18	557	72.6	69

(* is used to show that computer was not able to solve the problem in a reasonable time.)

Table 5.1. Summary results of the favorable q_0 determination.

q_0	Mean of solutions	Standard deviation	Minimum solution	Maximum solution	Selection measure
0	2630.57	28.78464	2598.1	2669.7	76846.36
0.05	2613.9	23.36417	2587.5	2655.7	62048.23
0.1	2619.34	27.01367	2563.5	2657.9	71799.63
0.15	2618.83	31.19651	2566.5	2657.3	82898.49
0.2	2615.92	38.79	2562.5	2677.3	103852.5
0.25	2629.84	28.97601	2573.7	2673.5	77467.37
0.3	2615.04	23.12196	2586.5	2663.8	61592.27
0.35	2627.26	25.99633	2565.2	2654.1	68996.87
0.4	2628.58	27.8667	2554.5	2650.3	73855.11
0.45	2610.66	30.77482	2543.9	2658.5	81814.85
0.5	2622.42	25.8411	2572.1	2663	68814.86
0.55	2617.56	24.43116	2570.5	2649.5	64730.37
0.6	2623.13	33.53807	2560.4	2669.5	89529.89
0.65	2623.09	22.86732	2574.1	2669.1	61035.16
0.7	2620.33	20.14812	2599.3	2669.1	53777.35
0.75	2630.27	22.72595	2580.3	2660.2	60455.58
0.8	2617.45	27.02596	2561.3	2651.5	71659.35
0.85	2597.84	24.9848	2558.5	2632.7	65777.49
0.9	**2611.18**	**18.1698**	**2578.3**	**2636.7**	**47908.32**
0.95	2591.86	28.32114	2550.5	2631.5	74527.09
1	2636.34	25.21671	2591.9	2674.5	67442.1

Also, according to the definition of the problem, the number of ants in the colony of the problem is considered as twice as the number of trains and the fixed initial value $\tau_0 = 0.012$ that is obtained by $\tau_0 = 1/(n \cdot L_{nn})$ where n is the number of trains and L_{nn} is the solution cost produced by a heuristic method. (For further study refer to Dorigo and Gambardella [36, 37].) Furthermore by considering this fact that in the proposed algorithm, the length (cost) of arcs does not have a meaning therefore by supposing $\beta = 0$, the length effect of edges is omitted in ACS.

Then the best parameter values are experimentally adjusted. For this purpose, based on the best parameters values previously found, the parameter values are iterated incrementally and then the algorithm runs ten times. After that according to the least value of the mean multiplied by the standard deviation from ten runs the best parameter value is chosen. In the train scheduling problem we are looking for the best reliable solution with the least amount of delay, therefore the least value of the mean multiplied by the standard deviation is considered as the election measure. After this step the best value was chosen and then the problem is solved with these best parameters.

5.1.1. q_0 parameter. In determining q_0, the parameter value is iterated from 0 to 1 by increments of 0.05, and as it is clear from Table 5.1, according to the least value of the mean multiplied by the standard deviation from ten runs that its favorable value is supposed as $q_0 = 0.9$.

Table 5.2. Summary results of the favorable ρ determination.

ρ	Mean of solutions	Standard deviation	Minimum solution	Maximum solution	Selection measure
0	2628.44	27.94233	2583.1	2670.1	73444.74
0.05	2600.7	20.27588	2551.5	2620.5	52731.47
0.1	2606	27.39233	2540.9	2645.9	71384.42
0.15	2603.78	17.2209	2582.5	2633.3	44839.43
0.2	2588.5	28.33796	2536.9	2619.1	73352.81
0.25	2593.2	26.77399	2561.5	2641.9	69430.32
0.3	2587.66	24.97052	2549.1	2625.9	64615.23
0.35	**2589.32**	**17.24367**	**2571.7**	**2629.4**	**44649.37**
0.4	2576.96	28.79507	2539.9	2632.9	74203.74
0.45	2577.88	19.50213	2546.3	2611.7	50274.14
0.5	2582.38	19.9787	2549.9	2616.3	51592.59
0.55	2566.6	22.23656	2527.9	2589.9	57072.35
0.6	2583.35	27.84422	2526.9	2629.5	71931.36
0.65	2568.3	21.06097	2540.7	2606.3	54090.89
0.7	2567.27	18.33176	2539.9	2605.1	47062.58
0.75	2563.59	24.66326	2527.9	2618.4	63226.5
0.8	2568.37	27.3426	2519.5	2609.3	70225.92
0.85	2560.96	18.90045	2515.9	2580.9	48403.3
0.9	2559.38	18.65171	2522.5	2591.1	47736.81
0.95	2567.03	25.03464	2532.5	2617.9	64264.68
1	2574.25	20.65662	2546.7	2605.7	53175.31

5.1.2. ρ parameter. Based on $q_0 = 0.9$, the ρ value is determined. In Table 5.2, the summary of results based on the iterations from 0 to 1 by increments of 0.05 for determining ρ value is put forward. The best value based on the least value of the mean multiplied by the standard deviation from ten runs that is supposed as $\rho = 0.35$.

5.1.3. ξ parameter. Based on $q_0 = 0.9$ and $\rho = 0.35$, the value of ξ is determined. Table 5.3 shows the results summary based on iterations from 0 to 1 by increments of 0.05 for determining ξ value. The most favorable value based on the least value of the mean multiplied by the standard deviation from ten runs that is supposed as $\xi = 0.2$.

5.1.4. τ_0 parameter. Based on $q_0 = 0.9$, $\rho = 0.35$, and $\xi = 0.2$, the value of τ_0 is determined. Table 5.4 shows the results summary of determining τ_0 value based on iterations from 0 to 0.0004 by steps of 0.00002. The best value based on the least value of the mean multiplied by the standard deviation from ten runs that equals $\tau_0 = 0$.

5.2. The results of running the model. After adjusting the parameters, the proposed algorithm for the problem with 30 trains was run. The time-distance graph of the trains traveling is shown in Figure 5.1. The amount of delay in this state equals 2492.1. Figure 5.2 is the indicator of convergence in improving the solutions in each cycle from running the

TABLE 5.3. Summary results of the favorable ξ determination.

ξ	Mean of solutions	Standard deviation	Minimum solution	Maximum solution	Selection measure
0	2585.92	23.95615	2548.9	2627.3	61948.7
0.05	2585.91	18.74581	2556	2610.3	48474.98
0.1	2578.24	18.5987	2557.5	2608.9	47951.91
0.15	2588.6	21.44518	2546.5	2618.1	55512.98
0.2	**2584.89**	**11.80644**	**2567.1**	**2599.7**	**30518.36**
0.25	2588.44	26.72781	2554.7	2619.1	69183.34
0.3	2603.4	21.13712	2572.7	2635.9	55028.37
0.35	2587.3	37.42581	2534.7	2638.5	96831.79
0.4	2593.22	28.9329	2524.7	2635.3	75029.38
0.45	2607	27.42979	2554.7	2654.9	71509.46
0.5	2598.39	22.40111	2564.9	2639.1	58206.83
0.55	2587.54	34.91702	2532.9	2632.3	90349.18
0.6	2587.65	26.78081	2541.9	2624.5	69299.36
0.65	2584.9	19.57618	2552.5	2608.1	50602.46
0.7	2580.33	19.84485	2539.9	2599.6	51206.25
0.75	2592.96	12.03884	2571.5	2608.9	31216.24
0.8	2592.04	14.82912	2565.9	2613.7	38437.66
0.85	2587.7601	18.94173	2561.1	2618.1	49016.66
0.9	2593.03	13.87228	2566.7	2612.5	35971.23
0.95	2588.26	20.32815	2543.5	2614.1	52614.54
1	2591.96	13.64015	2571.9	2612.5	35354.73

TABLE 5.4. Summary results of the favorable τ_0 determination.

τ_0	Mean of solutions	Standard deviation	Minimum solution	Maximum solution	Selection measure
0	**2540.92**	**18.81429**	**2515.9**	**2571.5**	**47805.59**
0.00002	2589.4	26.4162	2547.9	2619.9	68402.11
0.00004	2594.78	30.84649	2546.7	2642.9	80039.87
0.00006	2611.66	30.25474	2541.5	2651.1	79015.1
0.00008	2610.71	23.94718	2562.1	2635.9	62519.16
0.0001	2597.31	36.71574	2514.5	2637.8	95362.15
0.00012	2625.12	26.19728	2577.5	2655.7	68771
0.00014	2621.44	21.47894	2578.3	2644.8	56305.75
0.00016	2621.24	20.82969	2583.5	2654.5	54599.62
0.00018	2617.38	22.21835	2586.7	2652.1	58153.87
0.0002	2618.43	20.91507	2575.7	2644.3	54764.64
0.00022	2608.24	21.54067	2578.9	2662.5	56183.24
0.00024	2620	25.23564	2572.7	2650.7	66117.39
0.00026	2617.64	20.5229	2578.9	2654.5	53721.56
0.00028	2634.96	27.93267	2581.9	2671.3	73601.46
0.0003	2620.7	28.01809	2567.7	2661.9	73427.01
0.00032	2613.15	20.18378	2591.1	2646.5	52743.25
0.00034	2620.72	22.6618	2584.5	2652	59390.24
0.00036	2625.21	23.63972	2596.7	2668.3	62059.24
0.00038	2605.77	20.65301	2582.3	2645.2	53816.99
0.0004	2612.01	25.06234	2576.1	2650.9	65463.09

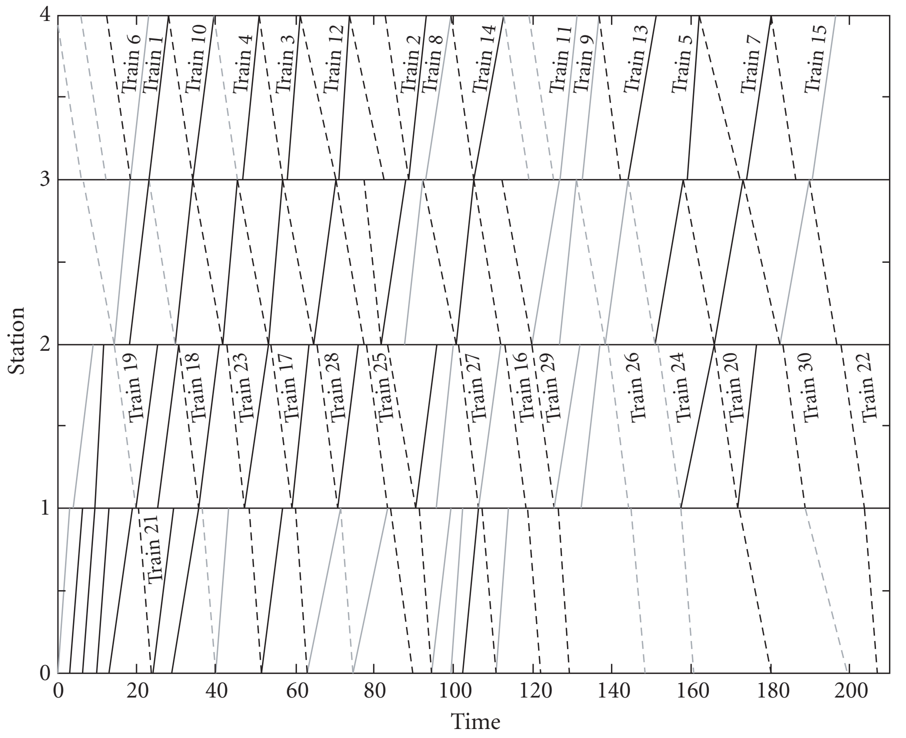


Figure 5.1. Time-distance graph of the trains traveling.

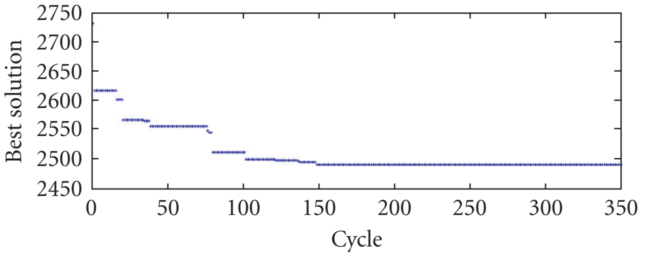
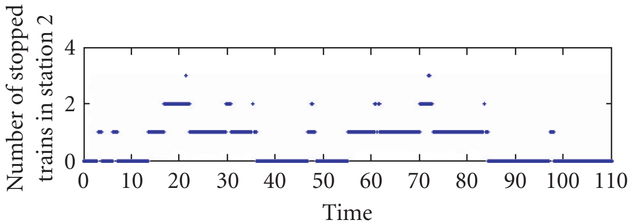


Figure 5.2. Convergence indicator in improving the solutions.

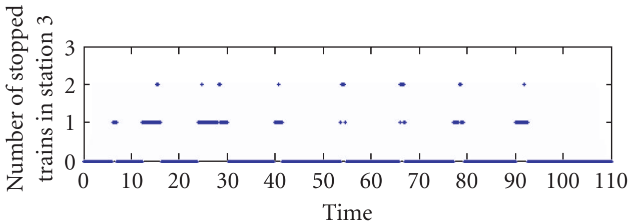
algorithm. Figure 5.3 shows the number of the dwelled trains in each time in the intermediate stations. Maximum number of the trains dwelled at the same time in stations 2, 3, and 4 are, respectively, equal to 3, 2, and 3 trains.

6. Conclusion

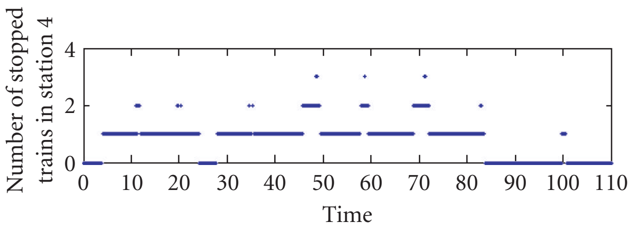
This paper developed an algorithm for the train scheduling problem using the ant colony system metaheuristic called ACS-TS. At first, a mathematical model for a kind of train



(a)



(b)



(c)

Figure 5.3. Number of the dwelled trains in each time in the intermediate stations.

scheduling problem was developed and then the algorithm based on ACS was presented to solve the problem. The problem was considered as a traveling salesman problem wherein cities in the TSP represent the trains. ACS determined the sequence of trains dispatched on the graph of the TSP. Using the sequences obtained and removing for collisions incurred, train scheduling was determined. Numerical examples in small and medium sizes were solved using ACS-TS and compared to exact optimum solutions to check for quality and accuracy. Comparison of the solutions showed that ACS-TS results in good quality and time savings. A case study was presented to illustrate the solution.

References

[1] I. Amit and D. Goldfarb, *The timetable problem for railways*, Developments in Operations Research **2** (1971), 379–387.

[2] S. Araya and K. Abe, *An optimal rescheduling for online train traffic control in disturber situations*, Proceedings of the 22nd IEEE Conference on Decision and Control, Texas, December 1983, pp. 489–494.

[3] A. A. Assad, *Models for rail transportation*, Transportation Research **14 B** (1980), 101–114.

[4] O. Baboglu, H. Meling, and A. Montresor, *Anthill: a framework for the development of agent based peer-to-peer systems*, Tech. Rep. UBLCS-2001-09, Department of Computer Science, University of Bologna, Bologna, 2001.

[5] B. Barán and R. Sosa, *AntNet routing algorithm for data networks based on mobile agents*, Inteligencia Artificial **12** (2001), 75–84.

[6] A. Baykasoglu, T. Dereli, and I. Sabuncu, *An ant colony algorithm for solving budget constrained and unconstrained dynamic facility layout problems*, to appear in Omega.

[7] J. E. Bell and P. R. McMullen, *Ant colony optimization techniques for the vehicle routing problem*, Advanced Engineering Information **18** (2004), 41–48.

[8] L. Bianchi, L. M. Gambardella, and M. Dorigo, *An ant colony optimization approach to the probabilistic traveling salesman problem*, Mathematical Modeling and Algorithms **3** (2004), no. 4, 403–425.

[9] J. A. Bland, *Layout of facilities using an ant system approach*, Engineering Optimization **32** (1999), no. 1, 101–115.

[10] ———, *Space–planning by ant colony optimization*, International Journal of Computer Applications in Technology **12** (1999), no. 6, 320–328.

[11] ———, *Optimal structural design by ant colony optimization*, Engineering Optimization **33** (2001), 425–443.

[12] C. Blum, *Beam-ACO—hybridizing ant colony optimization with beam search: an application to open shop scheduling*, Computers & Operations Research **32** (2005), no. 6, 1565–1591.

[13] E. Bonabeau, F. Henaux, S. Guerin, D. Snyers, P. Kuntz, and G. Theraulaz, *Routing in telecommunication networks with "Smart" ant-like agents*, Proceedings of the 2nd International Workshop on Intelligent Agents for Telecommunication Applications (IATA '98), Lectures Notes in AI, vol. 1437, Springer, New York, 1998.

[14] G. S. Brodal and R. Jacob, *Time-dependent networks as models to achieve fast exact time-table queries*, Electronic Notes in Theoretical Computer Science **92** (2004), 3–15.

[15] B. Bullnheimer, R. F. Hartl, and C. Strauss, *An improved ant system algorithm for the vehicle routing problem*, Annals of Operations Research **89** (1999), 319–328.

[16] X. Cai and C. H. Goh, *A fast heuristic for the train scheduling problem*, Computers and Operation Research **21** (1994), no. 5, 499–510.

[17] M. Carey and D. Lockwood, *A model, algorithms and strategy for train pathing*, Operational Research Society **46** (1985), 988–1005.

[18] B. Chen and P. T. Harker, *Two moments estimation of the delay on single-track rail lines with scheduled traffic*, Transportation Science **24** (1990), no. 4, 261–275.

[19] Y. Cheng, *Hybrid simulation for resolving resource conflicts in train traffic rescheduling*, Computers in Industry **35** (1998), no. 3, 233–246.

[20] T. Chiang, H. Hau, H. Chiang, S. Ko, and C. Hsieh, *Knowledge-based system for railway scheduling*, Data & Knowledge Engineering **27** (1998), no. 3, 289–312.

[21] V. Cicirello, *A game-theoretic analysis of multi-agent systems for shop floor routing*, Tech. Rep. CMU-RI-TR-01-28, Robotics Institute, Carnegie Mellon University, Pennsylvania, 2001.

[22] A. Colorni, M. Dorigo, V. Maniezzo, and M. Trubian, *Ant system for job shop scheduling*, Operations Research, Statistics and Computer Science **34** (1994), no. 1, 39–53.

[23] J. F. Cordeau, P. Toth, and D. Vigo, *A survey of optimization models for train routing and scheduling*, Transportation Science **32** (1998), 380–404.

[24] O. Cordon, I. Fernandez de Viana, F. Herrera, and L. Moreno, *A new ACO model integrating evolutionary computation concepts: the best-worst ant system*, Abstract Proceedings of ANTS2000 - From Ant Colonies to Artificial Ants: A Series of International Workshops on Ant Algorithms (M. Dorigo, M. Middendorf, and T. Stutzle, eds.), Brussels, 2000, pp. 22–29.

[25] D. Costa and A. Hertz, *Ants can colour graphs*, Journal of the Operational Research Society **48** (1997), 295–305.

[26] J. E. Cury, F. A. C. Gomide, and M. J. Mendes, *A methodology for generation of optimal schedules for an underground railway system*, IEEE Transaction on Automatic Control **25** (1980), no. 2, 217–222.

[27] L. M. de Campos, J. M. Fernández-Luna, J. A. Gámez, and J. M. Puerta, *Ant colony optimization for learning Bayesian networks*, International Journal of Approximate Reasoning **31** (2002), no. 3, 291–311.

[28] M. Dessouky and R. C. Leachman, *A simulation modeling methodology for analyzing large complex rail networks*, Simulation **65** (1995), no. 2, 131–142.

[29] G. Di Caro and M. Dorigo, *AntNet: a mobile agents approach to adaptive routing*, Artificial Intelligence Research **9** (1997), 317–365.

[30] ______, *Extending antNet for best-effort quality-of- service routing*, Presentation at ANTS '98 - From Ant Colonies to Artificial Ants: 1st International Workshop on Ant Colony Optimization, Brussels, October 1998.

[31] K. F. Doerner, W. J. Gutjahr, R. F. Hartl, C. Strauss, and C. Stummer, *Pareto ant colony optimization with ILP preprocessing in multiobjective project portfolio selection*, to appear in European Journal of Operational Research.

[32] K. F. Doerner, R. F. Hartl, and M. Reimann, *Cooperative Ant Colonies for Optimizing Resource Allocation in Transportation. Applications of Evolutionary Computing*, Lecture Notes in Computer Science (LNCS), vol. 2037, Springer, Berlin, 2000.

[33] ______, *CompetAnts for problem solving: the case of full truckload transportation*, Central European Journal of Operations Research **11** (2003), no. 2, 115–141.

[34] M. Dorigo and G. Di Caro, *The ant colony optimization meta-heuristic*, New Ideas in Optimization (D. Corne, M. Dorigo, and F. Glover, eds.), McGraw-Hill, London, 1999, pp. 11–32.

[35] M. Dorigo, G. Di Caro, and L. M. Gambardella, *Ant algorithms for discrete optimization*, Artificial Life **5** (1999), no. 2, 137–172.

[36] M. Dorigo and L. M. Gambardella, *Ant colonies for the traveling salesman problem*, BioSystems **43** (1997), no. 2, 73–81.

[37] ______, *Ant colony system: a cooperative learning approach to the traveling salesman problem*, IEEE Transactions on Evolutionary Computation **1** (1997), no. 1, 53–66.

[38] M. Dorigo, V. Maniezzo, and A. Colorni, *Positive feedback as a search strategy*, Tech. Rep. 91-016, Dipartimento di Elettronica, Politecnico di Milano, Milano, 1991.

[39] J. Dréo and P. Siarry, *Continuous interacting ant colony algorithm based on dense heterarchy*, Future Generation Computer Systems **20** (2004), 841–856.

[40] J. Eggers, D. Feillet, S. Kehl, M. O. Wagner, and B. Yannou, *Optimization of the keyboard arrangement problem using an ant colony algorithm*, European Journal of Operational Research **148** (2003), no. 3, 672–686.

[41] O. Engelhardt-Funke and M. Kolonko, *Analysing stability and investments in railway networks using advanced evolutionary algorithms*, International Transactions in Operational Research **11** (2004), no. 4, 381–394.

[42] M. Fabinkue, *A swarm intelligence approach to constraint satisfaction*, Proceedings of the 6th Conference on Integrated Design and Process Technology (IDPT '02), Texas, June 2002.

[43] M. Fischetti, J. Salazar-Gonzales, and P. Toth, *The generalized traveling salesman problem and orienteering problem*, Traveling Salesman Problem and Its Variations (G. Gutin and A. P. Punnen, eds.), Kluwer Academic, Dordrecht, 2002, pp. 609–663.

[44] P. Forsyt and A. Wren, *An ant system for bus driver scheduling*, Proceedings of the 7th International Workshop on Computer-Aided Scheduling of Public Transport Preprints, Center for Transportation Studies, MIT, Massachusetts, 1997, pp. 405–421.

[45] O. Frank, *Two-way traffic on a single line of railway*, Operations Research **14** (1965), 801–811.

[46] L. M. Gambardella and M. Dorigo, *An ant colony system hybridized with a new local search for the sequential ordering problem*, INFORMS Journal on Computing **12** (2000), no. 3, 237–255.

[47] L. M. Gambardella, E. Taillard, and G. Agazzi, *MACS-VRPTW: vehicle routing problem with time windows*, New Ideas in Optimization (D. Corne, M. Dorigo, and F. Glover, eds.), McGraw-Hill, London, 1999, pp. 63–76.

[48] L. M. Gambardella, E. Taillard, and M. Dorigo, *Ant colonies for the quadratic assignment problem*, Operational Research Society **50** (1999), no. 2, 167–176.

[49] J. A. Gámez and J. M. P. Puetra, *Searching for the best elimination sequence in Bayesian networks by using ant colony optimization*, Pattern Recognition Letters **23** (2002), no. 1–3, 261–277.

[50] X. Gandibleux, X. Delorme, and K. Tkindt, *An Ant Colony Optimization Algorithm for The Set Packing Problem*, submitted to Ants '04, 4th International Workshop on Ant Colony Optimization and Swarm Intelligence, 2004.

[51] K. Ghoseiri, F. Szidarovszky, and M. J. Asgharpour, *A multi-objective train scheduling: model and solution*, Transportation Research **38 B** (2004), 927–952.

[52] M. F. Gorman, *An application of genetic and tabu searches to the freight railroad operating plan problem*, Annals of Operations Research **78** (1998), 51–69.

[53] M. Gravel, W. Price, and C. Gagné, *Scheduling continuous casting of aluminum using a multiple objective ant colony optimization metaheuristic*, European Journal of Operational Research **143** (2002), no. 1, 218–229.

[54] G. Gutin and A. P. Punnen, *The Traveling Salesman Problem and Its Variations*, Combinatorial Optimization, vol. 12, Kluwer Academic, Dordrecht, 2002.

[55] M. Heusse, S. Guerin, D. Snyers, and P. Kuntz, *Adaptive agent-driven routing and load balancing in communication networks*, Advances in Complex Systems **1** (1998), 237–254.

[56] A. Higgins and E. Kozan, *Modeling train delays in urban networks*, Transportation Science **32** (1998), no. 4, 346–357.

[57] A. Higgins, E. Kozan, and L. Ferreira, *Heuristic techniques for single line train scheduling*, Journal of Heuristics **3** (1997), no. 1, 43–62.

[58] C. L. Huntley, D. E. Brown, D. E. Sappington, and B. P. Markowicz, *Freight routing and scheduling at CSX*, Transportation Interfaces **25** (1995), no. 3, 58–71.

[59] Y. Iida, *Timetable preparation by A.I approach*, Proceeding of European Simulation Multiconference, Nice, 1988, pp. 163–168.

[60] V. K. Jayaraman, B. D. Kulkarni, and K. Gupta, *Dynamic optimization of fed-batch bioreactors using the ant algorithm*, Biotechnology Progress **17** (2001), no. 1, 81–88.

[61] V. K. Jayaraman, B. D. Kulkarni, S. Karale, and P. Shelokar, *Ant colony framework for optimal design and scheduling of batch plants*, Computers and Chemical Engineering **24** (2000), no. 8, 1901–1912.

[62] J. Jong and M. Wiering, *Multiple ant colony systems for the bus stop allocation problem*, Proceedings of the 13th Belgium-Netherlands Conference on Artificial Intelligence, Amsterdam, 2001, pp. 141–148.

[63] D. Jovanovic and P. T. Harker, *A decision support system for train dispatching: an optimization-based methodology*, Journal of the Transportation Research Forum **30** (1990), no. 1, 25–37.

[64] M. H. Keaton, *Designing optimal railroad operating plans: Lagrangian relaxation and heuristic approaches*, Transportation Research Part B **23** (1989), no. 6, 415–431.

[65] K. Komaya and T. Fukuda, *A knowledge-based approach for railway scheduling*, The 7th IEEE Conference on Artificial Intelligence Applications, Florida, 1991, pp. 405–411.

[66] P. Korošec, J. Šilc, and B. Robič, *Solving the mesh-partitioning problem with an ant-colony algorithm*, Parallel Computing **30** (2004), no. 5-6, 785–801.

[67] R. D. Kraay and P. T. Harker, *Real-time scheduling of freight railroads*, Transportation Research Part B: Methodological **29** (1995), no. 3, 213–229.

[68] R. S. K. Kwan and P. Mistry, *A co-evolutionary algorithm for train timetabling*, Research Report Series 2003.13, School of Computing, University of Leeds, Leeds, 2003.

[69] A. Lim, J. Lin, B. Rodrigues, and F. Xiao, *Ant colony optimization with hill climbing for the bandwidth minimization problem*, to appear in Applied Soft Computing.

[70] T. Lindner, *Train schedule optimization in public rail transport*, Ph.D. thesis, Technische Universitat, Braunschweig, 2000.

[71] V. Maniezzo, *Exact and approximate nondeterministic tree-search procedures for the quadratic assignment problem*, INFORMS Journal on Computing **11** (1999), no. 4, 358–369.

[72] V. Maniezzo and A. Carbonaro, *An ANTS heuristic for the frequency assignment problem*, Future Generation Computer Systems **16** (2000), no. 8, 927–935.

[73] V. Maniezzo, A. Carbonaro, M. Golfarelli, and S. Rizzi, *ANTS for data warehouse logical design*, Proceedings of the 4th Metaheuristics International Conference, Porto, 2001, pp. 249–254.

[74] V. Maniezzo and A. Colorni, *The ant system applied to the quadratic assignment problem*, IEEE Transactions on Data and Knowledge Engineering **11** (1999), no. 5, 769–778.

[75] V. Maniezzo, A. Colorni, and M. Dorigo, *The ant system applied to the quadratic assignment problem*, Tech. Rep. IRIDIA/94-28, IRIDIA, Université Libre de Bruxelles, Bruxelles, 1994.

[76] R. D. Martinelli and H. Teng, *Optimization of railway operations using meural networks*, Transportation Research **4c** (1996), 33–49.

[77] P. R. Mcmullen, *An ant colony optimization approach to addressing a JIT sequencing problem with multiple objectives*, Artificial Intelligence in Engineering **15** (2001), no. 1, 309–317.

[78] R. Michel and M. Middendorf, *An ACO algorithm for the shortest supersequence problem*, New Ideas in Optimization (D. Corne, M. Dorigo, and F. Glover, eds.), McGraw Hill, London, 1999, pp. 51–61.

[79] S. Minton, M. D. Johnston, A. B. Philips, and P. Laird, *Minimizing conflicts: a heuristic repair method for constraint satisfaction and scheduling problems*, Artificial Intelligence **58** (1992), no. 1–3, 161–205.

[80] N. Monmarché, G. Venturini, and M. Slimane, *There how pachycondyla apicalis ants suggest is new search algorithm*, Future Generation Systems Computer **16** (2000), no. 8, 937–946.

[81] R. Montemanni, L. M. Gambardella, A. E. Rizzoli, and A. V. Donati, *A new algorithm for a dynamic vehicle routing problem based on ant colony system*, Tech. Rep. IDSIA-05-02, Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA), Manno, November 2002, ftp://ftp.idsia.ch/pub/techrep/IDSIA-23-02.pdf.gz.

[82] R. Montemanni, D. H. Smith, and S. M. Allen, *An ANTS algorithm for the minimum-span frequency-assignment problem with multiple interference*, IEEE Transactions on Vehicular Technology **51** (2002), no. 5, 949–953.

[83] K. Nachtigall and S. Voget, *A genetic algorithm approach to periodic railway synchronization*, Computers & Operations Research **23** (1996), no. 5, 453–463.

[84] N. Nahas and M. Nourelfath, *Ant system for reliability optimization of a series system with multiple-choice and budget constraints*, Reliability Engineering and System Safety **87** (2005), no. 1, 1–12.

[85] C. E. Noon and J. C. Bean, *A Lagrangian based approach for the asymmetric generalized traveling salesman problem*, Operations Research **39** (1991), no. 4, 623–632.

[86] D. Pacciarelli and M. Pranzo, *A Tabu search algorithm for the railway scheduling problem*, Proceedings of the 4th Metaheuristic International Conference, Porto, 2001, pp. 16–20.

[87] Peat, Marwick, and Mitchell & Co., *Train dispatching simulation model: capabilities and description. USA*, Report DOT-FR-4-5014-1, Federal Railroad Administration, Department of Transportation, Washington, DC, March 1975.

[88] E. R. Petersen, *Over the road transit time for a single track railway*, Transportation Science **8** (1974), 65–74.

[89] V. Ramos and F. Almeida, *Artificial ant colonies in digital image habitats - a mass behavior effect study on pattern recognition*, Proceedings of the 2nd International Workshop on Ant Algorithms - From Ant Colonies to Artificial Ants, September 2000.

[90] M. Reimann and M. Laumanns, *Savings based ant colony optimization for the capacitated minimum spanning tree problem*, to appear in Computers and Operations Research.
[91] A. E. Rizzoli, A. V. Donati, L. M. Gambardella, N. Casagrande, and R. Montemanni, *Time dependent vehicle routing problem with an ant colony system*, Tech. Rep. IDSIA-17-03, Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA), Manno, 2002.
[92] A. Roli, C. Blum, and M. Dorigo, *ACO for maximal constraint satisfaction problems*, Proceeding of the 4th Metaheuristics International Conference, vol. 1, Porto, 2001, pp. 187–191.
[93] B. Scheuermann, K. So, M. Guntsch, M. Middendorf, O. Diessel, H. ElGindy, and H. Schmeck, *FPGA implementation of population-based ant colony optimization*, Applied Soft Computing **4** (2004), no. 3, 303–322.
[94] R. Schoonderwoerd, O. Holland, J. Bruten, and L. Rothkrantz, *Ant based load balancing in telecommunications networks*, Adaptive Behavior **5** (1996), no. 2, 169–207.
[95] M. Sepehri, *Railway crew scheduling with grouping evolutionary algorithm*, Amir Kabir Engineering Journal **14** (2003), no. 54, 565–577 (Persian).
[96] P. S. Shelokar, V. K. Jayaraman, and B. D. Kulkarni, *An ant colony approach for clustering*, Analytica Chimica Acta **509** (2004), no. 2, 187–195.
[97] L. Shi, J. Hao, J. Zhou, and G. Xu, *Ant colony optimization algorithm with random perturbation behavior to the problem of optimal unit commitment with probabilistic spinning reserve determination*, Electric Power Systems Research **69** (2004), no. 2-3, 295–303.
[98] K. Socha, M. Sampels, and M. Manfrin, *Ant algorithms for the university course timetabling problem with regard to the state-of-the-art*, Computer Science **2611** (2003), 334–345.
[99] M. Solimanpur, P. Vrat, and R. Shankar, *Ant colony optimization algorithm to the inter-cell layout problem in cellular manufacturing*, European Journal of Operational Research **157** (2004), no. 3, 592–606.
[100] ______, *An ant algorithm for the single row layout problem in flexible manufacturing systems*, Computers & Operations Research **32** (2005), no. 3, 583–598.
[101] T. Stutzle, *An ant approach to the flow shop problem*, Proceedings of the 6th European Congress on Intelligent Techniques and Soft Computing, vol. 3, Aachen, 1998, pp. 1560–1564.
[102] T. Stutzle and H. Hoos, *MAX-MIN ant system*, Future Generation Computer Systems **16** (2000), no. 8, 889–914.
[103] D. Subramanian, P. Druschel, and J. Chen, *Ants and reinforcement learning: a case study in routing in dynamic networks*, Proceedings of the International Joint Conference on Artificial Intelligence, Morgan Kaufmann, Nagoya, 1997, pp. 832–838.
[104] B. Szpigel, *Optimal train scheduling on a single-track railway*, Operational Research '72 (M. Ross, ed.), OR'72, North-Holland, Amsterdam, 1972, pp. 343–351.
[105] E. G. Talbi, O. Rouxb, C. Fonlupt, and D. Robillard, *Parallel ant colonies for the quadratic assignment problem*, Future Generation Computer Systems **17** (2001), no. 4, 441–449.
[106] E. S. Tzafestas, *Experiences from the development and use of simulation software for complex systems education*, Proceedings of the World Conference on the WWW and Internet (WebNet-2000), Texas, November 2000.
[107] R. Van der Put, *Routing in the faxfactory using mobile agents*, Tech. Rep. R & D-SV-98-276, KPN Research, Groningen, 1998.
[108] M. C. Van Wezel, J. N. Kok, J. N. Van den Berg, and W. Van Kampen, *Genetic improvement of railway timetables*, Computer Science **866** (1994), 566–574.
[109] K. Vijayakumar, G. Prabhaharan, P. Asokan, and R. Saravanan, *Optimization of multi-pass turning operations using ant colony system*, International Journal of Machine Tools & Manufacture **43** (2003), no. 15, 1633–1639.
[110] A. Wade and S. Salhi, *An ant system algorithm for the mixed vehicle routing problem with backhauls*, Metaheuristics: Computer Decision-Making, Kluwer Academic, Dordrecht, 2004, pp. 699–719.

[111] T. White, B. Pagurek, and F. Oppacher, *Connection management using adaptive mobile agents*, Proceedings of International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA '98), CSREA Press, Nevada, 1998, pp. 802–809.

[112] I. K. Yu and Y. H. Song, *A novel short-term generation scheduling technique of thermal units using ant colony search algorithms*, International Journal of Electrical Power and Energy Systems **23** (2001), no. 6, 471–479.

[113] Z. Zhou and Z. Liu, *Intelligent ant-based algorithm with applications in dynamic routing optimization of telecommunication networks*, Telecommunications Science **14** (1998), no. 11, 10–13.

[114] M. Zweben, E. Davis, E. Daun, and M. J. Deale, *Scheduling and rescheduling with iterative repair*, IEEE Transaction on Systems, Man, and Cybernetics **23** (1993), no. 6, 1588–1596.

Keivan Ghoseiri: School of Railway Engineering, Iran University of Science and Technology, Tehran 16846-13114, Iran
E-mail address: ghoseiri@iust.ac.ir

Fahimeh Morshedsolouk: School of Railway Engineering, Iran University of Science and Technology, Tehran 16846-13114, Iran
E-mail address: fahimeh_morshedsolouk@yahoo.com