Computer-Aided Civil and Infrastructure Engineering **27** (2012) 14–28

Comparing Ant Colony Optimization and Genetic Algorithm Approaches for Solving Traffic Signal Coordination under Oversaturation Conditions

Rahul Putha, Luca Quadrifoglio & Emily Zechman*

Zachry Department of Civil Engineering, Texas A&M University, College Station, TX, USA

Abstract: *This article proposes to solve the oversaturated network traffic signal coordination problem using the Ant Colony Optimization (ACO) algorithm. The traffic networks used are discrete time models which use green times at all the intersections throughout the considered period of oversaturation as the decision variables. The ACO algorithm finds intelligent timing plans which take care of dissipation of queues and removal of blockages as opposed to the sole cost minimization usually performed for undersaturation conditions. Two scenarios are considered and results are rigorously compared with solutions obtained using the genetic algorithm (GA), traditionally employed to solve oversaturated conditions. ACO is shown to be consistently more effective for a larger number of trials and to provide more reliable solutions. Further, as a master-slave parallelism is possible for the nature of ACO algorithm, its implementation is suggested to reduce the overall execution time allowing the opportunity to solve real-time signal control systems.*

1 INTRODUCTION

Traffic congestion is a daily problem in nearly all major cities in the world and continues to increase with population and economic growth of urban areas. The increasing traffic demand strains the existing transportation system, especially when the network is oversaturated during peak hours. Oversaturation occurs when the queues of vehicles on a receiving street interfere with the performance of the respective adjacent upstream streets, and though these conditions may last for only a short time, the time to clear the network may be significant. Costs of infrastructure renewal and expansion may be cost prohibitive, and, under limited budgets, strategies are needed that enhance the mobility and efficiency of the existing traffic without investing in new infrastructure.

To improve the efficiency of a transportation system, signal coordination strategies can be designed to maximize the number of vehicles processed by the network and minimize the travel time of the vehicles in the system. While significant research efforts have studied signal coordination for uncongested systems to design control policies for cost minimization, addressing the management of oversaturated conditions using similar approaches has yielded undesirable results. The dissipation of queues and removal of blockages should be prioritized over the minimization of travel costs (Roess et al., 1998). Instead, the objective of signal coordination in oversaturated conditions is to generate a set of green time durations to maximize the number of vehicles that are released at every signal of the network during congested phases (Abu-Lebdeh and Benekohal, 1997; Girianna and Benekohal, 2002a). The problem formulation has been reported in existing literature, but current algorithms used for problem solution are too slow or require very high computational power to be effective in real-time situations. The Ant Colony Optimization (ACO) algorithm is a fairly novel technique for solving computational problems by mimicking the natural behavior of ants as they generate and select paths to a food source from a colony (Dorigo and Stuetzle, 2004). ACO has been proposed as a viable approach for stochastic combinatorial optimization (Dorigo et al., 1996), with successful application to the traveling salesman problem, the asymmetric traveling salesman

*To whom correspondence should be addressed. E-mail: *ezechman@civil.tamu.edu*.

DOI: 10.1111/j.1467-8667.2010.00715.x

problem, the quadratic assignment problem, and the job–shop scheduling problem. In addition, ACO has been applied successfully for transportation planning problems (e.g., Yang et al., 2007; Vitins and Axhausen, 2009). This article describes the development and implementation of an ACO-based approach to solve the oversaturation traffic network problem and demonstrates the application of ACO for identifying timing strategies for two example networks. The solution performance of ACO is compared to a genetic algorithm (GA) approach, which is the optimization technique used to solve the oversaturated signal coordination problem in the existing literature (Girianna and Benekohal, 2004). The study finds that ACO is able to outperform the GA for solution of the problem and may provide a more efficient algorithm architecture for taking advantage of additional computational resources through parallel computing.

2 BACKGROUND

Planning signal timing for oversaturated networks should minimize traffic delays, satisfy constraints on maximum queue lengths, and allow for time-varying demands. Gazis (1964) and Gazis and Potts (1965) were the first to systematically address signal timing for oversaturated conditions through the use of a semigraphical approach, which identifies phase switching policies to manage queue dispersion and the total delay in the network by minimizing green time loss during the oversaturated period (Green, 1968). Michalopoulos and Stephanopoulos (1977, 1978) developed a two-stage timing method which identifies the optimal switch-over point at which the timing strategies for undersaturated and oversaturated conditions should be interchanged. This work was extended to identify smoother switching strategies through modeling the discrete nature of cycles (Chang and Lin, 2000). The two-stage timing method was then coupled with TRANSYT-7F in an integrated approach, where TRANSYT-7F identifies signal timings for undersaturated intersections, and the two-stage model, for oversaturated intersections (Chang and Sun, 2004). While these approaches focus on switching between under- and oversaturated timing strategies, a second set of studies focuses on the identification of optimal cycle lengths and green times for oversaturated conditions alone. Lieberman and Chang (2005) used a mixed-integer linear programming approach for this problem, and heuristic optimization methods have also been successfully applied (Saito and Fan, 2000; Varia and Dhingra, 2004; Sun et al., 2006; Teklu et al., 2007; Maher, 2008). A GA approach was used to minimize total delay through identifying phase sequences and proportions of green times (Foy et al., 1992). GA is a heuristic search algorithm belonging to a class of algorithms known as evolutionary algorithms (Holland, 1975) that are based loosely on the process of natural evolution. GAs have been applied in various disciplines of civil engineering such as construction engineering (Al-Bazi and Dawood, 2010; Cheng and Yan, 2009), transportation engineering (Vlahogianni et al., 2007; Lee and Wei, 2010), highway engineering (Kang et al., 2009), design optimization (Adeli and Cheng, 1994a, 1994b; Hung and Adeli, 1994; Adeli and Kumar, 1995a, 1995b; Sarma and Adeli, 2000a, 2000b, 2001, 2002; Kim and Adeli, 2001; Mathakari et al., 2007), structural control (Jiang and Adeli, 2008), and environmental pollution (Martínez-Ballesteros et al., 2010). Hadi and Wallace (1993) developed a hybrid approach that couples a GA with the TRANSYT-7F program. The GA optimizes cycle length, phase sequence, and offsets, and TRANSYT-7F is used to optimize green splits. Park et al. (1999) developed a GA-based method to identify cycle lengths, green splits, offsets, and phase sequences. GA was also used in a study performed to capture the critical operational issues at signalized intersections and remove the blocking effects of different lane groups in oversaturated conditions (Liu and Chang, 2011). Another signal control methodology is formulated as a quadratic programming problem to minimize and balance the link queues, therefore minimizing the risk of queue spillback (Aboudolas et al., 2010).

To better facilitate the use of a GA-based approach to control oversaturated conditions, the signal timing optimization model was reformulated to include in the objective function the total number of vehicles processed by the network during the oversaturated period (Abu-Lebdeh and Benekohal, 1997, 2000). New models for estimating the capacities of oversaturated arterials were developed based on the capacities of individual intersections, vehicle queue lengths, and offsets. The GA was applied to coordinate signals to maximize throughput, and results demonstrated a control strategy that avoided queue spillback and de facto red. This strategy was extended to coordinate oversaturated signals along an arterial that crosses multiple, parallel coordinated arterials (Girianna and Benekohal, 2002a, 2004). The research presented here utilizes this new objective function and compares the use of ACO to the performance of the previously reported GA-based approach.

3 PROBLEM FORMULATION

In managing oversaturated networks, traditional policies such as cost minimization are secondary to the removal of queues and blockages and the number of

vehicles processed by the network. The traffic signal design problem for oversaturated conditions has been effectively formulated as a constrained dynamic optimization problem to maximize the number of vehicles processed by a network by identifying values for the green times (Girianna and Benekohal, 2004). In the traffic model that follows, model input parameters include the network inflows, lengths of the streets of the network, saturation flow rates, number of lanes, percentages of left and right turning traffic, vehicle acceleration/deceleration, starting and stopping shockwave speeds, speed limits, the effective vehicle length, and green times. The decision variables are the green times throughout the traffic network. The model calculates the departure rates, queue lengths, and arrival rates at the intersections to calculate the value of the objective function:

$$Max \quad Y = \sum_{t}^{T} \sum_{(i,j)\in L} \sum_{h}^{H} \frac{d_{i,j}}{d_{\max}} D_{i,j}^{h}(t) - \sum_{k}^{K} \sum_{(i,j)\in Lp} \delta_{i,j}(k) \max \begin{pmatrix} 0 \\ q_{i,j}^{h*}(k) - q_{i,j}^{\max} \end{pmatrix} \tag{1}$$

$$\phi_{i,j}^{h*}(k) = \frac{d_{i,j}}{v_{i,j}} - \left(\frac{(v_{i,j} + \lambda) l_{veh}}{v_{i,j}\lambda} \right) q_{i,j}^{h*}(k) \quad \forall (i,j) \in L_p, \quad k = 1, \ldots, K \tag{2}$$

$$g_i^{h*}(k) \le g_j^{h*}(k) + \phi_{i,j}^{h*}(k) + \beta_{i,j}(k) \quad \forall (i,j) \in L_p, \quad k = 1, \ldots, K \tag{3}$$

$$\sum_{(i,j)\in F(r)} \phi_{i,j}(k) - \sum_{(i,j)\in R(r)} \phi_{i,j}(k) + \sum_{j\in N(r)} (g_{j,r}(k) + \Delta) = \sum_{m=k,\, j\in N(r)}^{k+nk} C_j(m) \quad \forall r \in R, \quad k = 1, \ldots, K \tag{4}$$

$$q_{i,j}^{h}(t) \le \frac{d_{i,j}}{l_{veh}} \quad \forall h \in H, \quad \forall (i,j) \in L_s, \quad t = 1, \ldots, T \tag{5}$$

$$g_{\min} \le g_j^h(k) \le g_{\max} \quad \forall j \in N, \quad \forall h \in H, \quad k = 1, \ldots, K \tag{6}$$

$$I_{i,j}^{h}(t) = \sum_{p\in Uh, b\in Bi} \theta_{b,i} D_{b,i}^{p}(t) \quad \forall h \in H, \quad t = 1, \ldots, T \tag{7}$$

$$A_{i,j}^{h}(t) = F\theta_{i,j} I_{i,j}^{h}(t - \tau_{i,j}) + (1 - F) A_{i,j}^{h}(t-1) \quad \forall h \in H, \quad \forall (i,j) \in L, \quad t = 1, \ldots, T \tag{8}$$

$$D_{i,j}^{h}(t) = \min \begin{pmatrix} c_j^h(t)\Delta T \\ A_{i,j}^{h}(t)\Delta T + q_{i,j}^{h}(t-1) \end{pmatrix} \quad \forall h \in H, \quad \forall (i,j) \in L, \quad t = 1, \ldots, T \tag{9}$$

where

$G = (N, L, P)$ denotes a traffic signal network
$N =$ set of signals
$L =$ set of directional streets
$P =$ set of coordinated paths p_{ij} starting form signal i to signal j
$N_p =$ set of signals on the coordinated paths
$L_p =$ set of streets along coordinated paths
$K =$ period of oversaturation in a cycle number
$T =$ period of oversaturation in a sample time
$t = 1, 2, \ldots, T$ is a discrete time index
$\Delta T =$ sample time interval (say 2, 3, 4, or 5 or more seconds)
$H =$ total phase number
$d_{i,j} =$ distance from signal i to j
$d_{\max} =$ the length of the longest street in the network
$g_{j,r}(\mathrm{k}) =$ the green time of signal j at cycle k serving movements in loop r
$q_{i,j}^{h*}(k) =$ number of vehicles in queue approaching signal j coming from signal i at the beginning of the downstream coordinated green phase $h*$ in cycle k. The star $(*)$ indicates a coordinated phase $q_{i,j}^{\max} = d_{i,j}/l_{veh}$
$q^{max}{}_{i,j} =$ maximum queue possible between signal i and j
$\delta_{i,j}(k) =$ non-negative disutility factor for cycle k between signals i and j whose values are determined based on the queue management strategy
$\phi_{i,j}^{h*}(k) =$ offset between signal i and j
$v_{i,j} =$ speed of a released platoon
$l_{veh} =$ average length of vehicles
$\lambda =$ starting shock wave speed
$g_i^{h*}(k) =$ effective green time at signal i
$g_{max} =$ maximum bound of the green time

g_{min} = minimum bound of the green time

$\beta_{i,j}(k)$ = time it takes for a stopping shock wave to propagate upstream

$C_j(m)$ = length of mth cycle

$N(r)$ = set of nodes on loop $r \in R$ (number of loop in the network)

$F(r)$ = set of nodes where traffic moves in the same direction as the loop

$R(r)$ = set of nodes where traffic moves in a different direction to that of the loop

Δ = lost green time

$I^h_{i,j}(t)$, $A^h_{i,j}(t)$, $D^h_{i,j}(t)$ = inflow, arrival and departure flows of phase h over $[t\Delta T, (t+1)\Delta T]$

U_h = set of phases at the upstream signal that feeds traffic for phase h of the downstream signal

$b \in B_i$ = set of upstream intersections connected to intersection i

$\theta_{b,i}$ = percentage of the departed traffic volume of upstream streets (b, i) that enters road section (i, j)

γ = platoon dispersion factor empirically derived (0.5)

$\tau_{i,j}$ = cruise travel time of a released platoon factored by 0.8

F = smoothing factor $= \frac{1}{(1+(\gamma)(\tau_{i,j}))}$

$c^h_j(t)$ = capacity during effective green interval

The objective function (Equation (1)) is composed of two terms: the total number of vehicles processed by the network throughout the oversaturation period and a disutility function. The number of vehicles processed at each cycle is weighted by the ratio of the length of the road segment between an upstream and a downstream intersection to the length of the longest street in the network, and the sum of these values over all signals in the network serves as the first term. The disutility function penalizes the occurrence of queues at the end of the green time along coordinated arterials. Offset is defined as the time difference between the green time initiations of two adjacent signals. The offsets need to be coordinated taking into account the distance between the signals, released platoon speed, platoon dispersion, and time required for the queue to dissipate. As it is not possible to coordinate all the phases, only the phase which controls the coordinated movement in both the signals is coordinated. Equation (2) enforces a coordinated offset between signal i and j. De facto red exists when the queue in the downstream is long enough to stop traffic from the upstream from entering it even though the signal is green. To avoid this, the effective green time of the upstream signal should be less than the sum of the effective green time for the coordinated downstream signal, the offset between two signals, and the time it takes for a stopping shock wave to move upstream. Equation (3) is introduced to avoid de facto red. Equation (4) is needed to enforce that the sum of offsets and green times around any loop of the network is equal to an integer multiple of the cycle time (Gartner, 1972). Equation (5) enforces that the queues in a non-coordinated arterial should not block the traffic movements of the upstream intersections. Equation (6) fixes the range of the decision variables, which are the green times, $g^h_j(k)$. The arrival flows in Equation (8) are calculated using the Robertson's dispersion model (Robertson, 1969). Equations (7)–(9) are the necessary network flow constraints.

The signal coordination model identifies optimal signal timings for the entire period of oversaturation. This is a large combinatorial optimization problem and cannot be efficiently solved using traditional calculus-based optimization techniques; heuristic approaches can be effectively applied to find nearly global-optimal solutions. The following sections describe two heuristics, ACO and GA, as they are applied to solve the problem.

4 ANT COLONY OPTIMIZATION

ACO is an optimization technique inspired from the natural behavior of ants (Colorni et al., 1991; Dorigo and Thomas, 2004) and is one of the most successful techniques of the field of swarm intelligence. ACO was developed as a heuristic method to identify efficient paths through a graph and has been successfully applied to identify optimal solutions for discrete problem representations. For solution of continuous problems, variables are divided into increments, based on the amount of accuracy required, and nodes are represented at each increment. A population of computational ants is initialized, and each computational ant selects a "path" and passes through one node for every variable. The length of a path represents the value of the objective function, and over a set of successive iterations or ant generations respectively, highly fit paths emerge and the objective function value is minimized.

The mechanism that natural ants use to find the shortest path from a food source to the colony is to deposit pheromone on path segments. Pheromone evaporates

over time, so that the edges and emergent trails with the highest concentration of pheromone will be identified by other ants and reinforced (Bonabeau et al., 2000). The fitness function of any path is used to determine the amount of artificial pheromone that will be deposited at each edge of the ant's tour. Ants are attracted to edges with higher concentrations of the chemical, and adding pheromone to the track is analogous to increasing the probability that the next ant would select that edge for its path. Edges that result in better fitness functions are reinforced in the same way that paths from the ant colony to the food source are reinforced in the natural system. There is a possibility, however, that early in the search, some suboptimal paths may be generated. The chances of these paths being reinforced may be high because of their early creation and initial deposit of pheromone. To avoid the survival of suboptimal paths, an evaporation function allows pheromones to evaporate at a constant rate. Pheromone concentrations decrease on edges that are selected infrequently, and the paths will be lost over successive iterations of the algorithm. This enables optimum edges to dominate the set of edges that are chosen by different ants to form paths through the network of potential nodes. Due to the stochastic nature of the algorithm, however, the search can continue to explore new sections of the decision space, avoiding local optima, as an edge with a low pheromone concentration may be randomly selected on occasion.

The variables used in the signal timing problem are green times, and Equations (7)–(9) simulate the traffic flows and queues for a given set of green times. The algorithm chooses green times within the range of times allowed (specified in Equation (6)). The minimum and maximum values of the variables are specified by the user, and the algorithm divides the range into a finite number of nodes separated by intervals. ACO divides the green times into a number of intervals using the *interval size* (appearing in Equation (10) below) and initializes ants, which select one node per variable and form a path. Ants choose nodes probabilistically, based on the amount of pheromone at each node. Higher concentrations of the pheromone on a node result in a higher probability that the node will be chosen as part of the solution. Figure 1 represents graphically the selection of nodes by one ant to calculate the value of the corresponding objective function. The value of a decision variable is determined as

$$g = (g_{\min} - \text{interval size}) + (\text{node number} * \text{interval size}) \tag{10}$$

where $g_{\min}$ is the minimum value for the green time, and g is the value of the decision variable. For example, if node 6 were selected, the value of the green time at that intersection and cycle would be (20 − 5) + (6∗5) = 45 seconds, using an interval size of 5 seconds.

The Lagrangian relaxation technique is used to ensure that Equations (2)–(5) are not violated. A penalty function approach transforms the constrained signal coordination problem to an unconstrained problem by adding a penalty to the objective function based on the constraint violations. The ACO approach can solve an unconstrained formulation. The fitness value of a solution is defined as follows (adapted from Girianna and Benekohal, 2004):

$$\text{fitness} = C_{\min} - \left(Z - \sum_{j-1}^{m} \mu_j H_j \right) \tag{11}$$

where μ_j is a penalty coefficient for constraint j, m is the number of implicit constraints, and H_j denotes j's constraint function (inequality and equality). The fitness, which is analogous to the length of the path to be traveled by the ant, is minimized. Fitness values need always be positive for ACO application, as the pheromone calculation depends on the fitness values, and negative amounts of pheromone cannot be placed on nodes. $C_{\min}$ is an input coefficient that is introduced to avoid the negative values for the fitness, which may result when the constraints are significantly violated and the violations are greater than the number of vehicles that are processed in the network. The constraint in Equation (4) is not active if a coordinated signal is an open-loop system, that is, when multiple coordinated arterials cross a single coordinated arterial. The final objective function is

$$\begin{aligned}
\text{Min} \quad C_{\min} - & \left[\sum_{k}^{K} \sum_{(i,j)\in L} \sum_{h}^{H} \frac{d_{i,j}}{d_{\max}} D_{i,j}^{h}(k) \right. \\
& - \sum_{k}^{K} \sum_{(i,j)\in Lp} \delta_{i,j}(k) \max \begin{pmatrix} 0 \\ q_{i,j}^{h*}(k) - q_{\max} \end{pmatrix} \\
& - \mu_1 \sum_{k,(1,j)\in L_p}^{K} \left(\phi_{i,j}^{h}(k) - \left(\frac{d_{i,j}}{v_{i,j}} - \frac{(v_{i,j}+\lambda) l_{veh}}{v_{i,j}\lambda} q_{i,j}^{h*}(k) \right) \right)^2 \\
& - \mu_2 \sum_{k,(i,j)\in Lp}^{K} \max \begin{pmatrix} 0 \\ g_i(k) - (g_j(k) + \phi_{i,j}(k) + \beta_{i,j}(k)) \end{pmatrix} \\
& \left. - \mu_3 \sum_{k,(i,j)\in L}^{K} \max \begin{pmatrix} 0 \\ q_{i,j}^{h}(k) - \dfrac{d_{i,j}}{l_{\text{veh}}} \end{pmatrix} \right]
\end{aligned} \tag{12}$$

The ACO algorithm can be stopped based on the perceived goodness of the objective function value or on the convergence of the mean of the current population

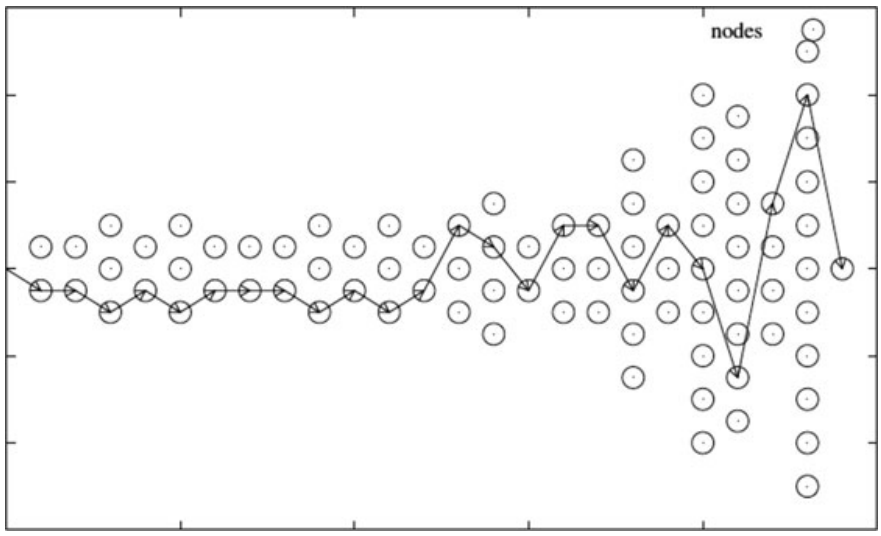


Fig. 1. One-way traffic network with 20 signals.

of ants and the best fitness value. To facilitate comparison with the GA approach for this study, the maximum number of trials was predetermined by the user.

5 GENETIC ALGORITHM

The GA is a search technique based on the mechanics of natural selection and genetics (Goldberg, 1989). A population of solutions is initialized and converges to one nearly optimal solution over a set of generations, or iterations, of the algorithm. For the binary GA that is used here, decision variables take on values of zero or one, and are represented as genes in an individual solution. The fitness of the solution is calculated as the objective function. At each generation, a selection operator ensures that the highly fit solutions survive in the population, and the less fit solutions are discarded. Operators are applied to create new solutions based on the surviving solutions. Crossover is applied to randomly select two individuals in the mating pool, bifurcating them at randomly selected sites, swapping the string of decision variables, or genes, and creating new individuals. A probability of p_c is applied to the crossover operation so that only a subset of the individuals participate in this operation, and some fit individuals found in the previous population survive without modification. To escape local optima, a mutation operation is applied. The mutation operator goes through all the bits in all the genes of the population and modifies a particular bit with a mutation probability p_m. In the binary GA, the mutation operator changes a zero present in the binary string to a one, or vice versa, with a mutation probability of p_m. In the next generation of the GA, the same processes of selection, crossover, and mutation are repeated.

For applying the GA, the constrained signal coordination problem is transformed into an unconstrained problem by associating a penalty with all constraint violations (Dasgupta and Michalewich, 1997), identical to the ACO approach, and the fitness function is minimized. The number of green time values can be calculated as $(g_{\max}-g_{\min})$/interval size. This can be set equal to 2^d and d can be computed as $\log_2((g_{\max}-g_{\min})$/interval size). In our case, each solution is represented using a four-bit binary string, which can take on a value between 0 and 15, inclusive. Green time is calculated using Equation (13) (Girianna and Benekohal, 2004).

$$g = g_{\min} + \left(\frac{g_{\max} - g_{\min}}{2^d - 1}\right) DV \tag{13}$$

where DV is the decoded value of the four-bit string.

For example, for a binary string, 1101, the decoded value of the string is $(1)(2^3)+(1)(2^2)+(0)(2^1)+(1)(2^0) = 8 + 4 + 0 + 1 = 13$. The corresponding green time is $20+(80 - 20/2^4-1)13 = 72$ seconds. The fitness of a solution is represented using Equation (12), and the total number of generations is specified as the stopping criteria for the GA.

6 MODEL DESCRIPTION

Two model networks have been used for testing ACO versus GA performance. Model I is taken from Girianna and Benekohal (2002a), and model II has been chosen to be sufficiently complex to represent a real

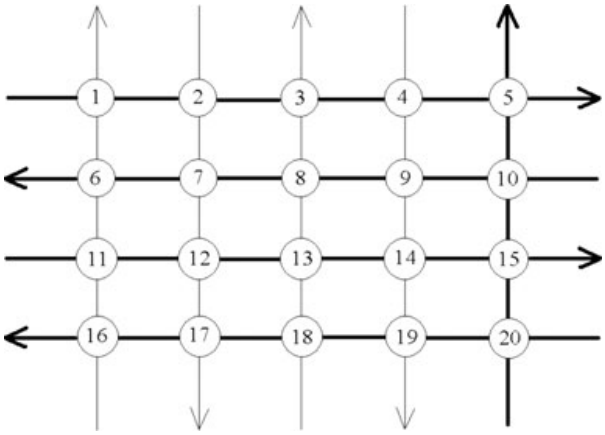


Fig. 2. One-way traffic network with 20 signals.

situation, but is small enough to allow thorough analysis for several search configurations with limited computing power available. Preliminary trials for parameter calibration identified reasonable settings for the crossover rate and the evaporation constant at 0.8 and 0.8, respectively, and Gaussian mutation was used. The values of μ_1, μ_2, μ_3, which are the weights applied to the different constraint violations in Equation (12), are set at 1,800, 1,800, and 1, respectively, as selected by Girianna and Benekohal (2004).

6.1 Model I

Solution of model I was demonstrated by Girianna and Benekohal (2002a). It is a one-way system, shown in Figure 2. Each signal works on a two-phase plan, and no turning movements are allowed. The traffic flows in the system are an eastbound flow from 1 to 5 and 11 to 15; westbound from 10 to 6 and 20 to 16; northbound from 16 to 1, 18 to 3 and 20 to 5; and southbound from 2 to 17 and 4 to 19. The signals on the northbound arterial from 20 to 5 and the east–west running arterials are coordinated. The thick lines in Figure 2 represent the coordinated movements.

The northbound arterial from signal 20 to 5 and the westbound arterial from signal 10 to signal 6 have flow rates of 2,000 vehicles per hour per lane (vphpl). The westbound arterial from 20 to 16 and the eastbound arterials have flow rates of 1,800 vphpl. The northbound arterials at signals 16 and 18 and the southbound arterials at 2 and 4 have flow rates of 1,500 vphpl. The maximum green time allowed for all the signals is 80 seconds, the minimum green time, 20 seconds, and the lost green time is 5 seconds. The speed limit, or desired speed, is 40 ft/s (43.89 km/h). The vehicle deceleration/acceleration is 4 ft/s^2. The saturation flow is 1,800 vphpl. Initial queues at all the approaches are 20 vehicles per lane, and there are two arterial lanes. The effective vehicle length is 25 ft. The starting shock wave speed is 16 ft/s (17.56 km/h). The stopping shock wave speed is 14 ft/s (15.36 km/h). Flows and queues are evaluated at a sample time (ΔT) of 5 seconds. The total duration of oversaturation is assumed to be 15 minutes. It is assumed that $\delta_{i,j}\ (k) = 1$ for $(i, j) \in L_p$ and $k \in K$. All parameters are adopted from Girianna and Benekohal (2004) to allow a direct comparison.

As the green time range is divided into intervals of 5 seconds, the number of intervals is $(80 - 20)/5 = 12$. The number of nodes is 13. In a similar fashion, each green time is divided into 13 nodes (in this case because all the green times have the same ranges).

6.2 Model II

A model of the downtown traffic signal network of the City of Fort Worth was used to demonstrate the algorithm in planning for an actual traffic network. This model is more complicated than model I due to the following reasons:

- The network includes left turning movements.
- Streets are of different lengths.
- Each arterial has a different number of lanes.

A 4 × 4 traffic signal network (shown in Figure 3) was selected from the larger network to demonstrate optimization of green times. Initial queue lengths were assumed to be two vehicles per lane at the start of the period of oversaturation. The volumes for the network are as follows: 186 vehicles per hour on the eastbound arterial from signal 1 to 4; 177 vehicles per hour on the westbound arterial from signal 8 to 5; 103 vehicles per hour on the eastbound arterial from signal 9 to 12; 242 vehicles per hour on the westbound arterial from signal 16 to 13; 11 vehicles per hour on the northbound arterial from signal 1 to 13; 61 vehicles per hour on the southbound arterial from signal 2 to 14; 157 vehicles per hour on the southbound arterial from signal 3 to 15; 416 vehicles on the northbound arterial from signal 16 to 4. The maximum green time allowed for all the signals is 80 seconds and the minimum is 20 seconds. The lost green time is 5 seconds. The number of arterial lanes is based on the data describing the network. The speed limit, or desired speed, is 40 ft/s (43.89 km/h), and vehicle deceleration/acceleration is 4 ft/s^2. The saturation flow is 1,800 vphpl. The effective vehicle length is 25 ft. The starting shock wave speed is 16 ft/s (17.56 km/h), and the stopping shock wave speed is 14 ft/s (15.36 km/h). Flows and queues are evaluated at a sample time (ΔT) of 5 seconds, and the total duration of oversaturation is 15 minutes. Under the existing conditions, the actual

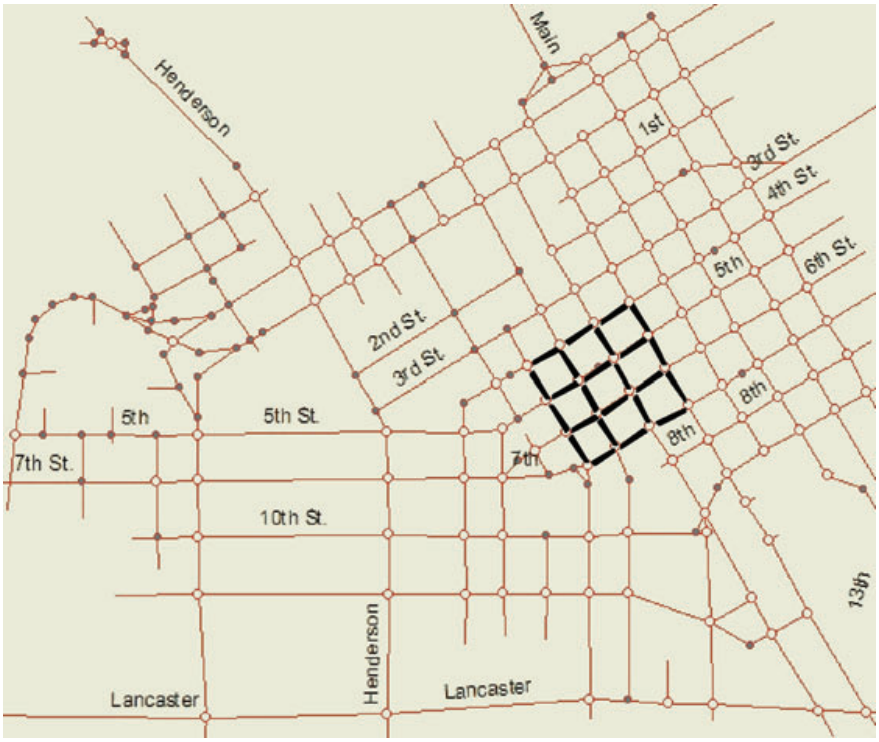


Fig. 3. Downtown network of the City of Fort Worth, TX. Black lines indicate portion of network in oversaturated conditions that is coordinated using the ACO approach.

data for peak hour volumes does not result in oversaturation. The network was studied for analysis of future conditions resulting from additional urbanization and economic development in the downtown Fort Worth area that would increase congestion. To approximate and simulate future conditions of increased traffic congestion, the current volumes were multiplied by a factor of 10.

7 RESULTS

7.1 Model I

For model I, the ACO and GA were each executed for 30 trials. A set of trials using random starting seeds is used to test the robustness of the algorithms, as both algorithms employ stochastic operators. The box plots comparing the objective functions (Equation (12)) of the best solutions found using ACO and GA with different settings are shown in Figure 4. ACO is able to identify solutions with lower objective functions than the GA. In addition, the set of solutions identified by ACO show less variability in objective function values.

As the objective function represents several different terms to describe different characteristics of the network performance, the terms are separated in the following figures. Figure 5 plots the number of cars processed in the network (first term of Equation (1)); Figure 6, the ideal offset (Equation (2)) associated with the solutions; and Figure 7, the de facto red time (Equation (3)). For all solutions identified by both algorithms, the queue storage capacity constraint (Equation (5)) is satisfied; the penalty is equal to zero. These figures demonstrate that the number of vehicles that pass through the network does not significantly improve as more fit solutions are identified through higher levels of computing (Figure 5). Instead, the ideal offset is more dramatically decreased (Figure 6). The ideal offset is shown in units of seconds∗seconds, representing the deviation from the ideal offset over all intersections in the network. The average ideal offset can be calculated to roughly represent the deviation from the ideal offset at each intersection and at each cycle, as shown in Figure 8. By allowing more computational time for the algorithm to converge through increasing the trials and number of ants for the ACO and the number of generations and population size for the GA, the average ideal offset is decreased from approximately 3 to 2 seconds for the ACO, and the GA is not able to minimize the offset to the same degree. The results demonstrate that the GA identifies solutions that process a larger number

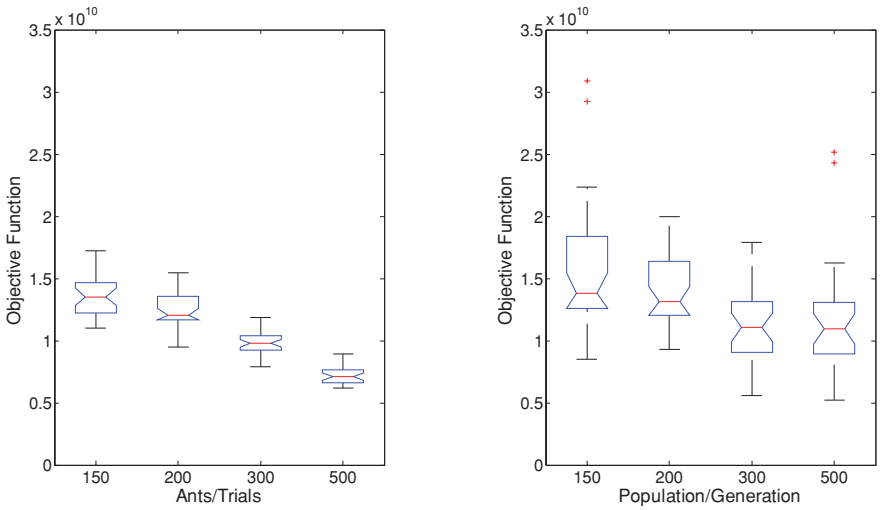


Fig. 4. Box plots of objective function values (Equation (12)) found using (a) ACO and (b) GA for model I.

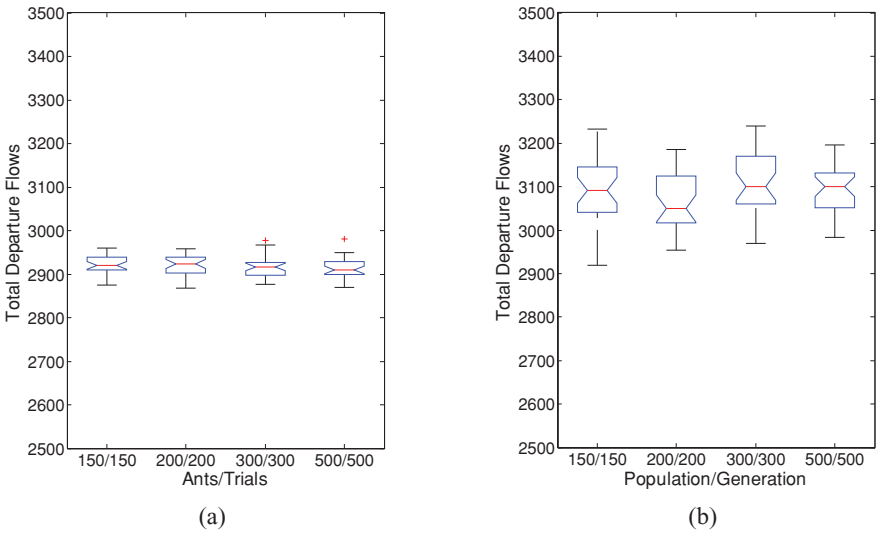
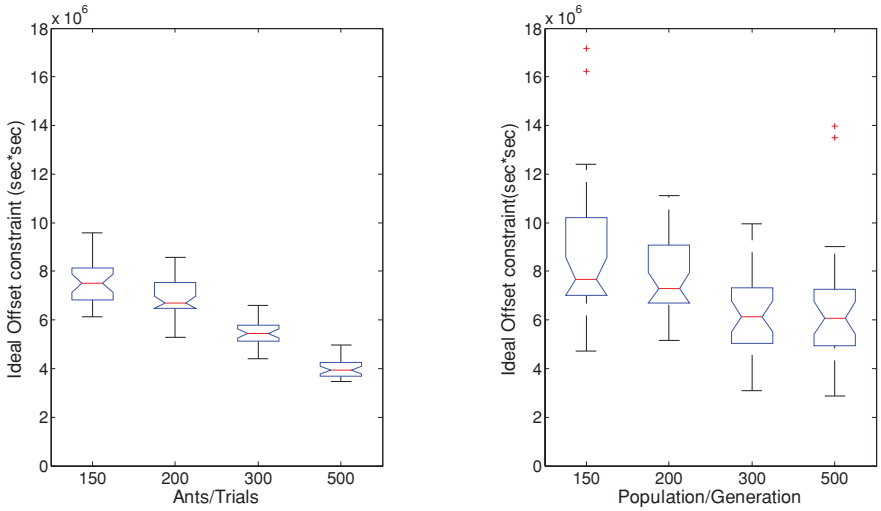


Fig. 5. Box plots of total weighted departure flows (first term of Equation (1)) using (a) ACO and (b) GA for model I.

of vehicles than solutions identified using ACO, but the congestion of the GA solutions is higher, demonstrated by larger de facto red times and offset values. The GA has identified local optima, or solutions that have decision variables leading to good fitness values, but with decision variables that are significantly different from the solutions with the optimal fitness value. In addition, the GA returns solutions with significantly more variability in fitness values. Many solutions identified by the GA have high values for de facto red times, while the solutions identified by ACO have de facto red times of zero (Figure 7).

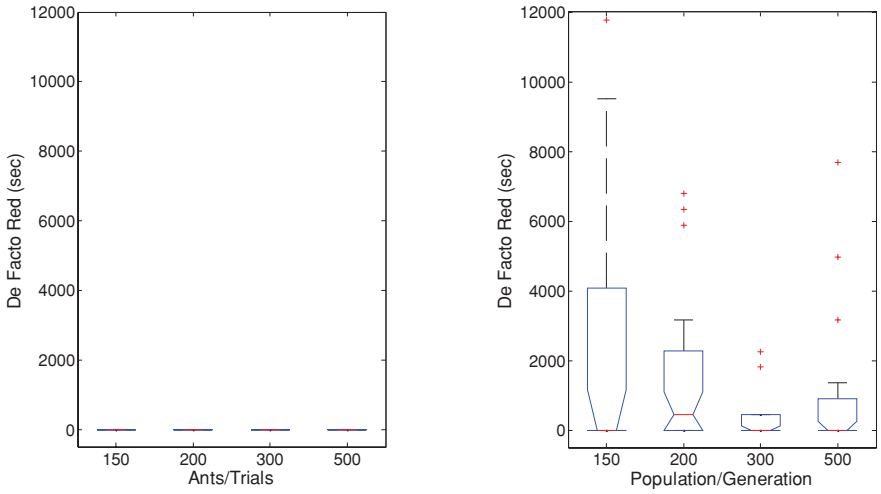
7.2 Model II

The results for model II demonstrate a slightly different comparison between GA and ACO approaches (Figures 8–12). The box plots comparing the objective functions (Equation (12)) of the best solutions found using ACO and GA with different settings are shown in Figure 9. For this network, the GA identifies more fit solutions for a lower number of solution evaluations. For the highest settings, however, of 500 ants/500 trials for ACO and 500 population size/500 generations for the GA, ACO identifies better solutions than the GA. Similar to model I, the total departure flows do not



(a) (b)

Fig. 6. Box plots of ideal offset constraints (Equation (2)) using (a) ACO and (b) GA for model I.



(a) (b)

Fig. 7. Box plots of de facto red constraints processed (Equation (4)) using (a) ACO and (b) GA for model I.

improve with improved fitness values (Figure 9), but for ACO, the ideal offset improves dramatically when more computational power is allowed for the ACO algorithm (Figure 10). The average ideal offset is decreased from 2 seconds to 1 second as the number of ants and trials is increased for the ACO methodology; for the GA, the average ideal offset is improved only when the settings are changed to a population of 500 and 500 generations. GA identifies solutions that show considerable variability in the ideal offset and some variability in the total departure flows, but routinely identifies solutions with minimal de facto red time (Figure 11) and queue storage time (Figure 12). ACO, on the other hand, identifies solutions with variability in the queues storage time and de facto red time, but minimal values for the ideal offset and less variability in the total departure flows.

7.3 Pairwise comparison

To provide a second statistical comparison between ACO and GA, a set of hypothesis tests are conducted (Milton and Arnold, 1995). The performance measures that are compared are the best objective function from a GA trial, specified as Y_i, and the best

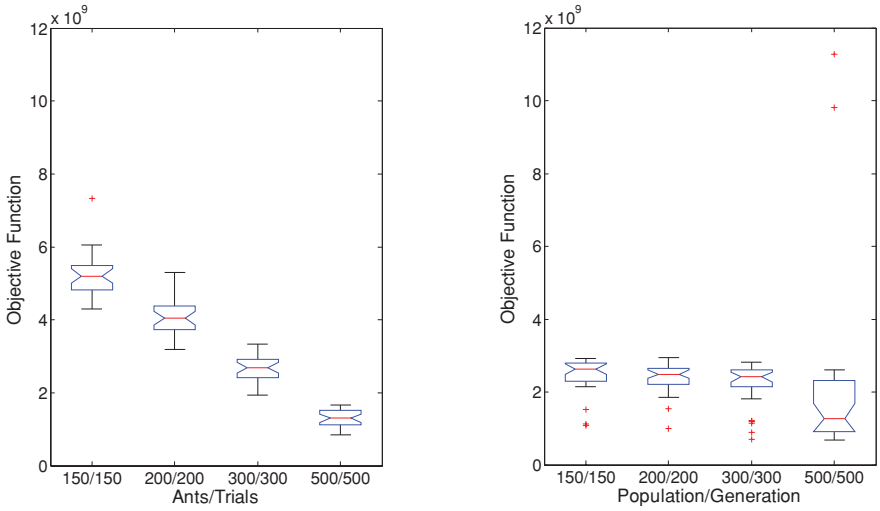


Fig. 8. Box plots of objective function values found using (a) ACO and (b) GA for model II.

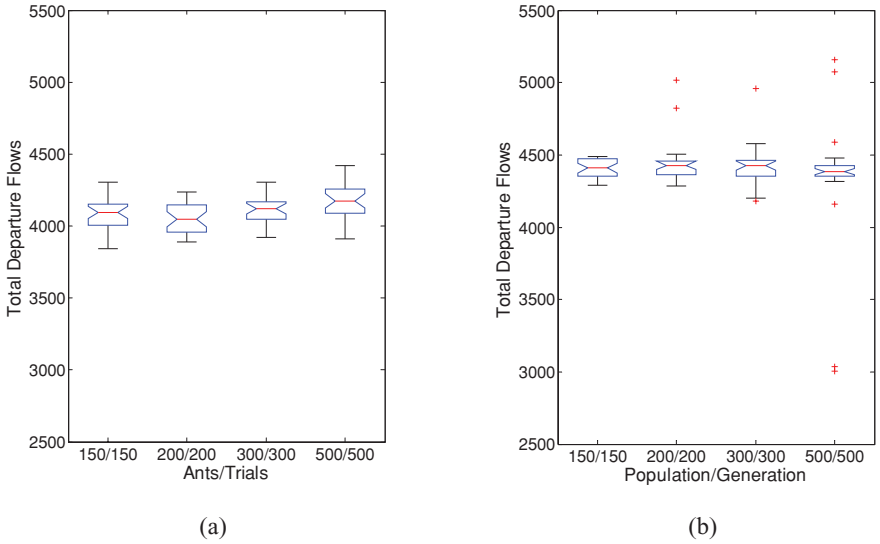


Fig. 9. Box plots of total weighted departure flows (first term of Equation (1)) using (a) ACO and (b) GA for model II.

objective function from an ACO trial, labeled $Y_{i'}$ where $i = 1$ to n, and n is the number of trials, equal to 30. Let the performance difference, $Z_i = Y_i - Y_{i'}$. The following equations which represent the mean and the quasi-variance respectively are used to conduct the analysis:

$$Z(n) = \frac{\sum_i Z_i}{n} \tag{14}$$

$$S^2(n) = \frac{\sum_i [Z_i - Z(n)]^2}{n-1} \tag{15}$$

The $1-\alpha$ confidence interval of the true mean of Z is contained in the interval

$$Z(n) \pm t_{n-1,1-\alpha/2}\sqrt{\frac{S^2(n)}{n}} \tag{16}$$

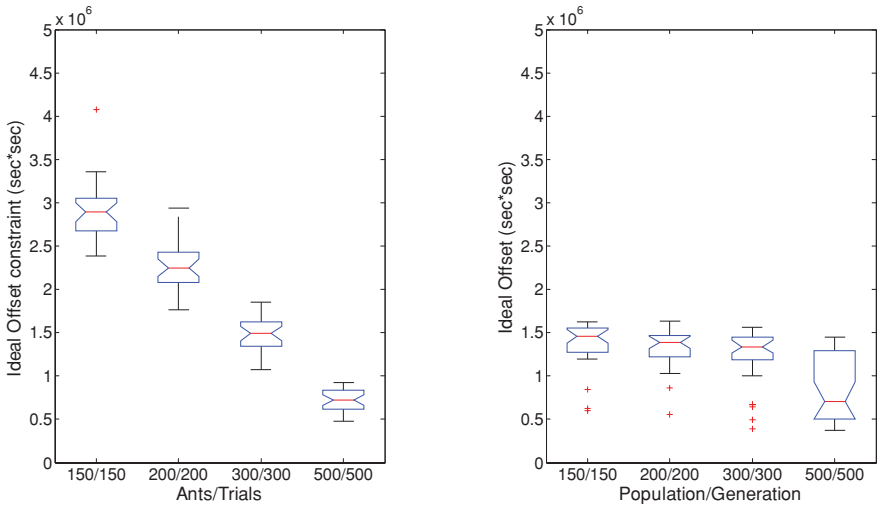


Fig. 10. Box plots of ideal offset constraints (Equation (2)) using (a) ACO and (b) GA for model II.

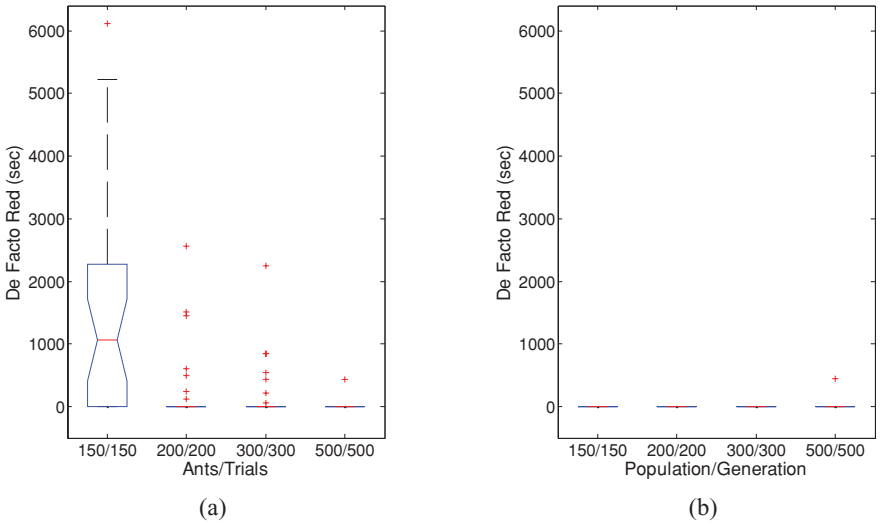


Fig. 11. Box plots of de facto red constraints using (a) ACO and (b) GA for model II.

One of the systems is statistically better than the other with $1-\alpha$ confidence only if the interval in Equation (16) does not contain zero.

The population size in GA is analogous to the number of ants in ACO, and, similarly, the number of generations is analogous to the number of trials. The two algorithms are tested for 30 trials for each case of the same population size/number of ants and number of generations/number of trials. The cases considered are listed in Table 1 for model I and Table 2 for model II. The objective function values are compared using Equations (14)–(16), which are used to compute the upper and lower limits of Z_i. For settings that result in positive upper and lower limits of Z_i, the fitness function of ACO is greater than the fitness function of GA with a confidence interval of 95% ($\alpha = 0.1$), indicating that the GA approach performs better for the minimization problem. Similarly, if the upper and lower limits are negative, then the objective function value of the best solution identified using the GA is greater than the fitness function identified using ACO with a confidence interval of 95%; in this case, ACO outperforms the GA.

From Table 1, it is observed that the fitness function improves (decreases) with more executions, and using 500 ants and 500 trials produces the best results. GA performs better than ACO for the settings with the least

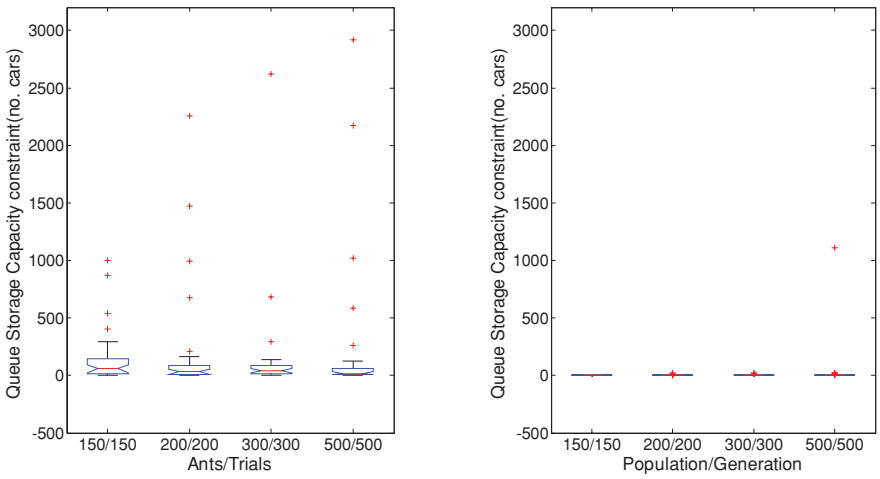


Fig. 12. Box plots of queue storage capacity constraints using (a) ACO and (b) GA for model II.

Table 1
Comparison of ACO and GA for model I

Population/ants	*Generations/trials*	$Z(n)$	n	$S^2(n)$	LL	UL	*Conclusion*
400	50	3.3×10^9	30	9.6×10^{18}	2.3×10^9	4.2×10^9	GA is better
150	150	-1.9×10^9	30	3.1×10^{19}	-3.6×10^9	-1.7×10^8	ACO is better
200	200	-1.7×10^9	30	8.5×10^{18}	-2.7×10^9	-8.4×10^8	ACO is better
300	300	-1.5×10^9	30	1.1×10^{19}	-2.5×10^9	-4.6×10^8	ACO is better
500	500	-4.3×10^9	30	1.9×10^{19}	-5.6×10^9	-2.9×10^9	ACO is better

Table 2
Comparison of ACO and GA for model II

Population/ants	*Generations/trials*	$Z(n)$	n	$S^2(n)$	LL	UL	*Conclusion*
150	150	2.7×10^9	30	5.5×10^{17}	2.5×10^9	3.0×10^9	GA is better
200	200	1.7×10^9	30	4.2×10^{17}	1.5×10^9	1.9×10^9	GA is better
300	300	4.8×10^8	30	4.8×10^{17}	2.6×10^8	6.9×10^8	GA is better
500	500	-8×10^8	30	5.7×10^{18}	-1.6×10^9	-7.9×10^7	ACO is better

number of executions (ants/population = 400, generations/trials = 50), and ACO performs better for all other settings and returns significantly fitter solutions than GA for the settings which require the highest number of executions (ants/population = 500, generations/trials = 500). ACO was able to identify increasingly more fit solutions as the number of executions was increased, while the GA identified similarly good solutions for all settings (also shown in Figure 5). This suggests that ACO may better handle more complicated real world traffic situations that would require large numbers of simulations to reach convergence.

Table 2 tabulates the same analysis for the simulation–optimization trials executed for model II. GA performs better than ACO for all the cases, except the settings that employ the highest number of ants and trials (ants/population = 500, generations/trials = 500). Similar to the results for model I, ACO identifies consistently better results with increasing number of executions, as compared to the GA.

8 PARALLEL COMPUTING

Both the GA and ACO approaches require a large number of simulations, and to employ these approaches in a real-time manner would require that the solutions could be executed in a more practical run-time. Girianna and Benekohal (2002b) implemented a parallel computing approach to reduce the computation time

for executing the GA to solve the signal timing problem. In execution of a typical generational GA, all the individuals in a population should be evaluated before the next generation of individuals is created using the crossover and selection operators. For executing the ACO approach, the ants in each trial and their paths are independently created based on pheromone levels. Therefore, the ants can be distributed over to the slave processors to save time. Future research should investigate a parallel-computing approach for the problem using the ACO algorithm.

9 CONCLUSION

Major traffic congestion problems occur during oversaturated network conditions, when the queues of vehicles on a receiving street interfere with the performance of the respective adjacent upstream streets. The need to solve the problem of signal coordination for oversaturated traffic networks is increasingly significant with increasing urban development. This research tested the performance of the ACO algorithm to calculate the best signal timing for a traffic signal network set under oversaturated conditions, and compared the performance with the GA approach. The results generated by ACO show significantly less variance among a set of random trials that use a larger number of model evaluations. Statistical analysis showed that ACO yielded better results when compared to GA for cases that utilized a higher number of model executions. This indicates that better solutions for the same computational power are identified, especially when the computational time can be afforded to search for the best possible solution, and that ACO may be a good alternative for solving very complicated networks. Finally, the ACO approach demonstrates the potential for implementation in a parallel computing configuration, which could significantly reduce the processing time to find solutions when compared to the present processing time. Further research will investigate the implementation of the ACO in a parallel computing environment to facilitate real-time management of traffic networks in oversaturated conditions.

REFERENCES

Aboudolas, K., Papageorgiou, M., Kouvelas, A. & Kosmatopoulos, E. (2010), A rolling-horizon quadratic-programming approach to the signal control problem in large-scale congested urban road networks, *Transportation Research Part C: Emerging Technologies*, **18**, 680–94.

Abu-Lebdeh, G. & Benekohal, R. F. (1997), Development of traffic control and queue management procedures for oversaturated arterial, *Transportation Research Record*, **1603**, 119–27.

Abu-Lebdeh, G. & Benekohal, R. F. (2000), Signal coordination and arterial capacity in oversaturated conditions, in *Journal of the Transportation Research Board*, 1727, TRB, National Research Council, Washington DC, pp. 68–76.

Adeli, H. & Cheng, N.-T. (1994a), Augmented Lagrangian genetic algorithm for structural optimization, *Journal of Aerospace Engineering, ASCE*, **7**(1), 104–18.

Adeli, H. & Cheng, N.-T. (1994b), Concurrent genetic algorithms for optimization of large structures, *Journal of Aerospace Engineering, ASCE*, **7**(3), 276–96.

Adeli, H. & Kumar, S. (1995a), Distributed genetic algorithms for structural optimization, *Journal of Aerospace Engineering*, **8**(3), 156–63.

Adeli, H. & Kumar, S. (1995b), Concurrent structural optimization on a massively parallel supercomputer, *Journal of Structural Engineering, ASCE*, **121**(11), 1588–97.

Al-Bazi, A. & Dawood, N. (2010), Developing crew allocation system for precast industry using genetic algorithms, *Computer-Aided Civil and Infrastructure Engineering*, **25**(8), 581–95.

Bonabeau, E., Dorigo, M. & Theraulaz, G. (2000), Inspiration for optimization from social insect behaviour, *Nature*, **406**, 39–42.

Chang, T. H. & Lin, J. T. (2000), Optimal signal timing for an oversaturated intersection, *Transportation Research Part B*, **34**(6) 471–91.

Chang, T.-H. & Sun, G. Y. (2004), Modeling and optimization of an oversaturated signalized network, *Transportation Research Part B*, **38**(8), 687–707.

Cheng, T. M. & Yan, R. Z. (2009), Integrating messy genetic algorithms and simulation to optimize resource utilization, *Computer-Aided Civil and Infrastructure Engineering*, **24**(6), 401–15.

Colorni, A., Dorigo, M., & Maniezzo, V. (1991), Distributed optimization by ant colonies, in *Proceedings of ECAL-91-European Conference on Artificial Life*, Paris, France, Varela, F.J. and Bourgine, P. (Eds.), MIT Press, Cambridge, MA, pp. 134–42.

Dasgupta, D. & Michalewich, Z. (1997), *Evolutionary Algorithms in Engineering Applications*, Springer-Verlag, Berlin, Germany.

Dorigo, M., Maniezzo, V. & Colorni, A. (1996), The ant system: optimization by a colony of cooperating ants, *IEEE Transactions on Systems, Man, and Cybernetics—PartB*, **26**(1), 1–13.

Dorigo, M. & Stuetzle, T. (2004), *Ant Colony Optimization*, MIT Press, Cambridge.

Foy, M. D., Benekohal, R. F. & Goldberg, D. E. (1992), Signal timing determination using genetic algorithms, in *Transportation Research Record 1365*, TRB, National Research Council, Washington DC, pp. 108–15.

Gartner, N. H. (1972), Constraining relations among offsets in synchronized signal networks, *Transportation Science*, **6**, 88–93.

Gazis, D. C. (1964), Optimal control of a system of oversaturated intersections, *Operations Research*, **12**, 815–31.

Gazis, D. C. & Potts, R. B. (1965), The oversaturated intersection, in *Proceedings of the Second International Symposium on the Theory of Road Traffic Flow*. Organization for Economic Cooperation and Development, Paris, pp. 221–37.

Girianna, M. & Benekohal, R. F. (2002a), Dynamic signal coordination for networks with oversaturated intersections, *Transportation Research Record*, **1811**, 122–30.

Girianna, M. & Benekohal, R. F. (2002b), Intelligent signal coordination on congested networks using parallel micro genetic algorithms, in *Proceedings of the Application of Advanced Technologies to Transportation (AATT 2002) 7th International Congress* Ed. K.C., Wang. Cambridge, MA.

Girianna, M. & Benekohal, R. F. (2004), Using genetic algorithms to design signal coordination for oversaturated networks, *Journal of Intelligent Transportation Systems*, **8**, 117–29.

Goldberg, D. (1989), *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company, Inc., Reading, MA.

Green, D. H. (1968), Control of oversaturated intersections, *Operational Research Quarterly*, **18**(2), 161–73.

Hadi, M. A. & Wallace, C. E. (1993), Hybrid genetic algorithm to optimize signal phasing and timing, in *Transportation Research Record 1421*, TRB, National Research Council, Washington DC, pp. 104–12.

Holland J. H. (1975), *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor.

Hung S. L. & Adeli, H. (1994), A parallel genetic/neural network learning algorithm for MIMD shared memory machines, *IEEE Transactions on Neural Networks*, **5**(6), 900–09.

Jiang, X. & Adeli, H. (2008), Neuro-genetic algorithm for nonlinear active control of highrise buildings, *International Journal for Numerical Methods in Engineering*, **75**(8), 770–86.

Kang, M. W., Schonfeld, P. & Yang, N. (2009), Prescreening and repairing in a genetic algorithm for highway alignment optimization, *Computer-Aided Civil and Infrastructure Engineering*, **24**(2), 109–19.

Kim, H. & Adeli, H. (2001), Discrete cost optimization of composite floors using a floating point genetic algorithm, *Engineering Optimization*, **33**(4), 485–501.

Lee, Y. & Wei, C. H. (2010), A computerized feature selection using genetic algorithms to forecast freeway accident duration times, *Computer-Aided Civil and Infrastructure Engineering*, **25**(2), 132–48.

Lieberman, E. & Chang, J. (2005), Optimizing traffic signal timing through network decomposition. *Transportation Research Record: Journal of the Transportation Research Board*, **1925**, Transportation Research Board of the National Academies, Washington DC, 167–75.

Liu, Y. & Chang, G.-L. (2011), An arterial signal optimization model for intersections experiencing queue spillback and lane blockage. *Transportation Research Part C: Emerging Technologies*, **19**(1), 130–44.

Maher, M. (2008), The optimization of signal settings on a signalized roundabout using the cross-entropy method, *Computer-Aided Civil and Infrastructure Engineering*, **23**(2), 76–85.

Martínez-Ballesteros, M., Troncoso, A., Martínez-Álvarez, F. & Riquelm, J. C. (2010), Mining quantitative association rules based on evolutionary computation and its application to atmospheric pollution, *Integrated Computer-Aided Engineering*, **17**(3), 227–42.

Mathakari, S., Gardoni, P., Agarwal, P., Raich, A. & Haukaas, T. (2007), Reliability-based optimal design of electrical transmission towers using multi-objective genetic algorithms, *Computer-Aided Civil and Infrastructure Engineering*, **22**(4), 282–92.

Michalopoulos, P. G. & Stephanopolos, G. (1977), Oversaturated signal system with queue length constraints. 1. Single intersection, *Transportation Research*, **11**(6), 413–21.

Michalopoulos, P. G. & Stephanopolos, G. (1978), Optimal control of oversaturated intersections—theoretical and practical considerations, *Traffic Engineering & Control*, **19**(5), 216–21.

Milton, J. S. & Arnold, J. C. (1995), *Introduction to Probability and Statistics*, McGraw-Hill, Inc., New York.

Park, B., Messer, C. J. & Urbanik II, T. (1999), Traffic signal optimization program for oversaturated conditions: genetic algorithm approach, *Transportation Research Record*, **1683**, 133–42.

Robertson, D. I. (1969), *TRANSYT—A Traffic Network Study Tool*, RRL Report LR 253, Road Research Laboratory, Crowthorne, Berkshire, UK.

Roess, R. P., McShane, W. R. & Prassas, E. S. (1998), *Traffic Engineering*, Prentice-Hall, Upper Saddle River, NJ.

Saito, M. & Fan, J. Z. (2000), Artificial neural network-based heuristic optimal traffic signal timing, *Computer-Aided Civil and Infrastructure Engineering*, **15**(4), 281–91.

Sarma, K. & Adeli, H. (2000a), Fuzzy genetic algorithm for optimization of steel structures, *Journal of Structural Engineering, ASCE*, **126**(5), 596–604.

Sarma, K. & Adeli, H. (2000b), Fuzzy discrete multicriteria cost optimization of steel structures, *Journal of Structural Engineering, ASCE*, **126**(11), 1339–47.

Sarma, K. C. & Adeli, H. (2001), Bi-level parallel genetic algorithms for optimization of large steel structures, *Computer-Aided Civil and Infrastructure Engineering*, **16**(5), 295–304.

Sarma, K. C. & Adeli, H. (2002), Life-cycle cost optimization of steel structures, *International Journal for Numerical Methods in Engineering*, **55**(12), 1451–62.

Sun, D. Z., Benekohal, R. F. & Waller, S. T. (2006), Bi-level programming formulation and heuristic solution approach for dynamic traffic signal optimization, *Computer-Aided Civil and Infrastructure Engineering*, **21**(5), 321–33.

Teklu, F., Sumalee, A. & Watling, D. (2007), A genetic algorithm approach for optimizing traffic control signals considering routing, *Computer-Aided Civil and Infrastructure Engineering*, **22**(1), 31–43.

Varia, H. R. & Dhingra, S. L. (2004), Dynamic optimal traffic assignment and signal time optimization using genetic algorithms, *Computer-Aided Civil and Infrastructure Engineering*, **19**(4), 260–73.

Vitins, B. J. & Axhausen, K. W. (2009), Optimization of large transport networks using the ant colony heuristic, *Computer-Aided Civil and Infrastructure Engineering*, **24**(1), 1–14.

Vlahogianni, E. I., Karlaftis, M. G. & Golias, J. C. (2007), Spatio-temporal short-term urban traffic flow forecasting using genetically-optimized modular networks, *Computer-Aided Civil and Infrastructure Engineering*, **22**(5), 317–25.

Yang, Z., Yu, B. & Cheng, C. T. (2007), A parallel ant colony algorithm for bus network optimization, *Computer-Aided Civil and Infrastructure Engineering*, **22**(1), 44–55.