Ant colony optimisation for vehicle traffic systems: applications and challenges

Mohammad Reza Jabbarpour*, Hossein Malakooti, Rafidah Md Noor, Nor Badrul Anuar and Norazlina Khamis

Faculty of Computer Science and Information Technology,
University of Malaya,
Kuala Lumpur, 50603, Malaysia
E-mail: reza.jabbarpour@siswa.um.edu.my
E-mail: hossein.malakooti@siswa.um.edu.my
E-mail: fidah@um.edu.my
E-mail: badrul@um.edu.my
E-mail: azlina@um.edu.my
*Corresponding author

Abstract: Ant-based algorithms simulate the cooperative behaviour of real ants in finding food resources. A significant number of studies have focused on the self-organised behaviour of ants in the natural environment to develop effective systems for dynamic problems. Ant-based systems have special properties such as scalability, adaptability, and dynamicity, which are the main requirements for solving vehicle traffic congestion problem. Thus, ant-based algorithms are now being adopted by vehicle traffic systems (VTSs) to guide vehicles to less congested paths. However, literature shows that comprehensive reviews are lacking in this field. The main contribution of this paper is the review and classification of the most relevant systems based on novel taxonomy. A survey that includes statistical analyses on ant-based VTS was conducted to identify the limitations and evaluation process of VTS. This paper concludes by proposing a general framework in applying ant colony optimisation to VTS.

Keywords: ant colony optimisation; vehicle traffic system; VTS; swarm intelligence; multi-agent system; MAS.

Reference to this paper should be made as follows: Jabbarpour, M.R., Malakooti, H., Noor, R.M., Anuar, N.B. and Khamis, N. (2014) 'Ant colony optimisation for vehicle traffic systems: applications and challenges', *Int. J. Bio-Inspired Computation*, Vol. 6, No. 1, pp.32–56.

Biographical notes: Mohammad Reza Jabbarpour is currently a PhD candidate and Research Assistant in the System and Technology Department, Faculty of Computer Science and Information Technology at University of Malaya, Malaysia. He received his Master of Data Communication and Networking in 2012 at the same university. His current research interests include ad hoc networks, quality of service, swarm intelligence and vehicle traffic routing systems.

Hossein Malakooti received his Master of Data Communication and Networking at the University of Malaya, Faculty of Computer Science and Information Technology. Having finished his Bachelors degree in Information Technology Engineering in Azad University, South Tehran Branch in 2010, he decided to join the research community at the University of Malaya.

Rafidah Md Noor received her BIT from Universiti Utara Malaysia in 1998, and MSc in Computer Science from Universiti Teknologi Malaysia in 2000, and PhD in Computing from Lancaster University in 2010. She is currently a Senior Lecturer at Computer System and Technology Department at the Faculty of Computer Science and Information Technology, University of Malaya. Her research interests include network mobility, vehicular networks, mobile IP, quality of service and quality of experience.

Nor Badrul Anuar received his PhD in Information Security from Centre for Security, Communications and Network Research (CSCAN), Plymouth University, UK in 2012 and Master of Computer Science from University of Malaya, Malaysia in 2003. He is a Senior Lecturer at the Faculty of Computer Science and Information Technology in University of Malaya, Kuala Lumpur. His research interests include information security (i.e., intrusion detection systems), artificial intelligence and library information systems.

Norazlina Khamis obtained her BIT from University of Malaya, in 1999; her MSc degree in Real-time Software Engineering from Universiti Teknologi Malaysia in 2001; and her PhD in Computer Science from The National University of Malaysia, in 2012. Her research interests include software engineering, programming education, sustainable software engineering, software quality and software metrics.

1 Introduction

Vehicle populations have significantly increased worldwide over the last decade. This increase in the number of vehicles has resulted in heavy traffic congestion, which leads to a high probability of road accidents (Kponyo et al., 2012). In addition to the rising number of vehicles, inaccurate routing to unknown destinations also increases traffic congestion because drivers become hesitant in selecting the best street to the desired destination (Marchesini and Weijermars, 2010). According to a report by Spalding (2008), fuel consumption, CO_2 and greenhouse gas emissions, long travel time, and accidents are the direct and indirect results of vehicle traffic congestion and haphazard driving patterns. In light of these problems, a vehicle navigation system that mitigates congestion problems is needed.

One of the problems which is usually faced when dealing with navigation system is finding the optimal route between two selected points on the given map, taking into consideration user preferences for distance, traffic load, road width, risk of collision, quality and number of intersections. It does not mean that this route should be the shortest one. It should be optimal or quasi-optimal in term of user preferences. The problem is known from the literature as multi-criteria shortest path problem (Martins, 1984; Ghoseiri and Nadjari, 2010), and it is proven to be NP-complete (Hansen, 1980). For multi-criteria combinatorial problems a single solution is very rarely be able to minimise (or maximise) all criteria, but rather there will be a set of compromise solutions. These solutions are called efficient non-dominated ones and are also referred to as pareto optimal set. To solve multiple criteria problem, parametrisation of the algorithm is performed by setting the corresponding coefficients for the different criteria of optimality (Boryczka and Bura, 2013).

In recent years, navigation systems based on swarm intelligence (SI) (Beni and Wang, 1989) algorithms and real-time traffic information have been widely used to offset the vehicle traffic problem (Kroon and Rothkrantz, 2003; Suson, 2010; Tatomir and Rothkrantz, 2006). SI is an emerging area in the field of optimisation and researchers have developed various algorithms by modelling the behaviours of different swarm of animals and insects such as ants, termites, bees, birds, fishes and bats (Yang et al., 2013). Moreover, recently, artificial plant optimisation algorithm (Cai et al., 2012; Yu et al., 2013), gravitational search algorithm (Sahu et al., 2013), shuffled frog leaping algorithm (Reddy and Vaisakh, 2013) and harmony search algorithm (Abdelaziz et al., 2013) have developed by SI researchers. However, bees, birds and ants are the most famous animals which are the source of inspiration for designing bio-inspired methodologies in computer science, specially in vehicle traffic system (VTS).

The meta-heuristic particle swarm optimisation (PSO) was proposed by Kennedy and Eberhart (1995) is loosely modelled on group behaviour, such as bird flocking and fish schooling (Ali and Sabat, 2012). Vehicle routing problem with time windows is one of the PSO-based strategies which is developed by Zhu et al. (2006). Zhao et al. (2006) proposed the urban traffic flow forecast model based on PSO in the case of two adjacent intersections. In addition to vehicle traffic routing optimisation, PSO is widely used for optimising various problems such as population diversity (Cui and Cai, 2009; Cui et al., 2010), activities schedule or sequence (Abdelsalam and Mohamed, 2012; Budinská et al., 2012), location and sizing of distributed generation (SalehiMaleh et al., 2013) and data fragmentation (Derrar et al., 2012). The bee colony optimisation has been introduced by Lucic and Teodorovic (2001, 2002, 2003a, 2003b) and Teodorovic et al. (2006) as a new direction in the field of SI. it has been applied in the cases of travelling salesman problem (Lucic and Teodorovic, 2001, 2002, 2003a) and transportation problem (Lucic and Teodorovic, 2003b; Teodorovi´c and Dell'Orco, 2005; Senge and Wedde, 2012a, 2012b).

Sur et al. (2012), Suson (2010), Yousefi and Zamani (2013) and Tatomir et al. (2009) proposed the use of a multi-agent system (MAS) with ant agents to solve multi-criteria shortest path problem and reduce vehicle congestion. By investigating ant behaviour, researchers have recognised that each ant randomly explores its surrounding to find food resources. Upon finding food, an ant uses a chemical material known as a pheromone to inform other ants about the food source. Individual ants can perform tasks independently while collaborating with other ants to solve a problem (i.e., finding food sources). This behaviour is very similar to the behaviour of vehicular networks (Kponyo et al., 2012). Fan et al. (2004) utilised ant colony algorithm for finding the shortest path for the first time. Ant colony optimisation (ACO) (Dorigo et al., 1996) is a branch of SI (Beni and Wang, 1989) and artificial intelligence that studies the behaviour of individual ants in various decentralised systems. Despite the great success and popularity of ant colony algorithms in solving the problems of VTSs, these algorithms still lack comparative studies and statistical analyses. In this paper, a comparative study and taxonomy of ant-based VTS were conducted to overcome the aforementioned shortcomings. The principles of ACO will also be discussed prior to an actual study of VTSs.

The main contributions of this paper are as follows:

1. discussion of the general principles of ACO and VTSs
2. extensive comparative survey of ant-based VTSs by introducing a novel taxonomy
3. comprehensive overview of the design challenges and characteristics of ant-based VTSs
4. identification of the most common flaws and shortcomings of proposed algorithms
5. proposition for a new generic framework for ant-based VTSs.

The rest of the paper is organised as follows. Section 2 discusses the general principles of ACO and its application in VTS. Section 3 explains the most important challenges in VTS design. Section 4 presents the taxonomy of ant-based VTSs, which includes three main classes, namely, ant colony parameters (ACPs), ant colony prediction (ACPre), and ant colony segmentation (ACS). Section 5 describes the various characteristics of relevant systems for each related algorithm. Section 6 involves a critical discussion on ant-inspired traffic routing for vehicular networks. Section 7 illustrates a generic framework for ant-based VTS. Section 8 concludes the paper.

2 Ant colony optimisation

ACO was initially introduced by Dorigo et al. (1996) and was inspired by the natural behaviour performed by ants in finding food resources. In this natural behaviour, ants leave pheromone trails to allow other ants to track and follow them to food resources. Previous experiments have shown that ants are able to find the optimal or shortest pathway between two individual sections. Although considerable modifications have been applied to ACO mechanisms, the essential principles of ACO methods are still the same. One of the major principles of ACO is the feedback-based process performed by ant colony. Ant colony functional performance is divided into four major steps:

Step 1: Ant distribution initialisation

Artificial ants tend to move between individual regions in a given environment (Ramos and Almeida, 2004). ACO solves problems by mapping the problematic environment into a graph with N nodes and R routes (edges). At the beginning of the ant colony search process, a number of ants are placed in different original nodes. The generation of a new path requires the use of forward ant (FANT), which is an agent that establishes the pheromone track to a source node (Gunes et al., 2002).

A node that receives a FANT for the first time creates a record in the routing table. The node uses the source address of the FANT as the destination address, the address of the previous node as the next stop, and computes the pheromone value depending on the number of hops required by the FANT to reach the node. When the FANT reaches the destination node, the destination node extracts and destroys the information of the FANT (Subha and Anitha, 2009).

Step 2: Ant probability distribution regulation

Deneubourg et al. (1990) stated that a path with high heuristic and pheromone values has a high probability of being chosen by an ant. A high amount of pheromone in a path denotes that a large number of ants have chosen such a path. Thus, the probability of selecting a preferred path increases with increasing ant activity until almost all ants choose the preferred path to travel from one point to another. According to the ant-obstacle model, the probability of choosing the shortest path increases over time until all ants select the shortest path (Deneubourg et al., 1990). After the destination node retrieves and destroys the information of the FANT, the node creates a backward ant (BANT) and sends it to the source node (Gunes et al., 2002). The BANT has the same task as the FANT. When the sender receives the BANT from the destination node, the path is established and data packets can be sent.

Ants then move from their origin region to the next region. Once the ants reach the next region *i*, the ants will move to the other region *j*. η_{ij} represents the edge cost incurred by an ant in traversing from one region to another, whereas τ_{ij} demonstrates the intensity of the trail on edge *(i,j)*. An ant *k* that is located in region *i* chooses the next region *j* to visit in time *t*. Hence, transition probability ($P_{ij}^k(t)$) can be defined by using equation (1) (Table 10):

$$P_{ij}^k(t) = \frac{(\tau_{ij})^\alpha (\eta_{ij})^\beta}{\sum_{h \notin tabu_k} (\tau_{ih})^\alpha (\eta_{ih})^\beta} \tag{1}$$

where tab_k is the set of visited nodes for ant *k*.

Step 3: Pheromone update

Once each ant has obtained a route, the end of each cycle will update the amount of pheromones in a trail by using a rule called the pheromone trail update. This rule is formulated as Equation (2):

$$\tau_{ij}^{new} = (1-\rho)\tau_{ij}^{old} + \sum_{k=1}^{m} \Delta\tau_{ij}^k \tag{2}$$

ρ (pheromone evaporation) is a fixed number between *0* and *1* [equation (2)]. The amount of pheromone on a specific edge *(i, j)* by ant *k* can be calculated by using equation (3):

$$\Delta\tau_{ij}^k = \frac{Q}{f_k} \tag{3}$$

where Q represents a fix parameter and f_k is the value of the obtained route by ant *k*.

Step 4: Stopping phase

In this phase, trail updating is completed either by reaching a defined threshold or by arriving to the maximum number of the best overall solution.

2.1 Applications of ACO

ACO is widely utilised in various areas of research in the recent years. Not only ACO could be used in theoretical aspect but also its role in real world problems and industry is undeniable (Mullen et al., 2009). Although application of ACO in vehicle routing, discussed in Section 2.2, is the main concern of this paper, an overview of other most noteworthy applications of ACO is given in this section. Along with vehicle routing, ACO is utilised for finding reserved parking places by Boehlé et al. (2008); Rosenberg (2010); WANG and TAO (2007). Besides, finding the shortest path in routing of various data networks is another application of ACO. Telecommunication or telephone (circuit-switched) networks (Schoonderwoerd et al., 1997; Bonabeau et al., 1998), packet-switched networks (DiCaro and Dorigo, 2011; Radwan et al., 2011) and wireless networks (e.g., Okdem and Karaboga, 2006; Gunes et al., 2002) are some important examples of these data networks. ACO can also be used for load balancing and congestion control in data networks (Sim and Sun, 2003).

NP-hard problems such as quadratic assignment (Stützle and Hoos, 2000; Maniezzo, 1999) and the job-shop scheduling (Colorni et al., 1994; Blum, 2005) are the other fields in which ACO is applied to introduce new state-of-the-art solutions, in which *ACO's* robustness is proved in mentioned problems through experiments. Additionally, ACO has been used to solve set covering and graph colouring problems by Lessing et al. (2004) and Costa and Hertz (1997), respectively.

Moreover, ACO is applied in some other areas for problem solving and optimisation. Classification rules (Martens et al., 2006), Bayesian networks (DeCampos et al., 2002, 2008), protein folding (Shmygelska and Hoos, 2005), radial distribution networks (Abdelaziz et al., 2012), hardware optimisation (Duan et al., 2012) and data mining (Parpinelli et al., 2002) are some examples of these areas. More information about applications of ACO can be found in study of ChandraMohan and Baskaran (2012) and Dorigo and Stutzle (2004).

2.2 ACO Application in VTSs

In vehicle routing, ACO uses one or more agents to explore the problem space (a graph in most cases) to find a path. This process considers the different constraints of time, distance, and number of hops. A vehicle routing problem (Dantzig and Ramser, 1959) and VTS are two different types of vehicle routing or navigation that uses ACO. In vehicle routing problem, ACO finds the shortest path from a starting point or node by visiting all nodes (e.g., cities) exactly once in the graph before returning to the point of origin. vehicle routing problem was introduced by Dantzig and Ramser (1959); the travelling salesman problem (Dorigo et al., 1991; Dorigo and Gambardella, 1997) is one of the well-known examples of vehicle routing problem. In VTS, ACO simply finds the shortest path between a source and a destination. In this case, which is called ant-based VTS in this paper, ACO uses real-time traffic information to shorten the distance and reduce congestion, travel time, fuel consumption, number of stopped vehicles, and air pollution (Kroon and Rothkrantz, 2003; Suson, 2010; Tatomir and Rothkrantz, 2006).

The VTS process is composed of three main phases:

1 initialisation

2 map exploration

3 optimal path suggestion

The initialisation phase consists of data gathering and segmentation. In the data gathering phase several tools such as camera, road wireless sensors and inductive loop detection are utilised in order to collect required information for the first phase of ant colony optimisation process (Bandyopadhyay, 2011). Each of these three tools operates as follows: By using video feeds from the cameras, the built-in software harvests information from that video, then gathered information such as vehicle volume and average velocity are fed into the fuzzy system. Moreover, road wireless sensors are deployed by the road intersection to detect vehicles. These sensors send the collected data to the intersection control agent. Then it processes the data and dynamically controls the vehicle traffic. Additionally, inductive loop detection can be placed in a roadbed to detect vehicles by measuring the vehicle's magnetic field. However, since ACO process starts only after needed data is gathered, data gathering process is not in concern of ACO.

The output of the gathered data rather than the gathering process is important in ant-based VTS. The main functionality of ACO in VTS exists in Phases 2 and 3, where an optimal path should be selected and suggested based on gathered data. In Phase 2, values are assigned to the road map according to the output data of Phase 1. Policies are assigned to ACO based on the environmental variables. In Phase 3, the optimal path is suggested by using the obtained information from ACO. Ant-based VTSs are discussed in detail in the subsequent sections of the paper.

3 Design challenges for traffic control using ant colonies

Several features distinguish the ant colony-based VTS traditional vehicle guidance systems. These features are initialisation, pheromone deposition, multi-agent, and Pareto archive. These features are the main ACO attributes that affect VTSs directly in the data gathering and path-selection procedure. Proper values should be assigned to these attributes to achieve an optimal solution. VTSs use real time or historical information to find the preferred path to a particular destination. VTSs can be implemented by using different factors that are important for the user, such as toll-free path, accident-free path, and shortest path. VTS has to prepare the required information to enable users to choose a path. The majority of VTS algorithms require the deployment of sensor-equipped vehicles. Hence, the scalability of the VTS algorithm is also a major concern.

Centralised management is also not suitable for VTS-based systems because of the wide distribution of vehicles on the streets. VTS-based systems also require self-organised mechanisms to guide vehicles.

3.1 Initialisation

The process of constructing solutions for traffic control in an ACO system can be considered a graph construction wherein each edge of the graph represents the possible path of an ant. The first and initial challenge in designing traffic control algorithms is the initialisation of default information such as edge weight and value. Ant movements are guided by

1 heuristic information (η) that represents prior data on the problem

2 a pheromone trails (τ) that encodes information on the ant colony search process, which is continuously updated.

These values are used by ants in conducting probabilistic decisions for the next visited node.

3.2 Pheromone deposition/update

Ants conduct pheromone deposition and update procedures to update the pheromone metric. This procedure can be applied when multiple matrices are used. For this case, one can select a set of best solutions (i.e., shortest path) to update the pheromone metric. Another method to update the pheromone metrics is to gather and store non-dominated solutions in an auxiliary function (Ahangarikiasari et al., 2013). On one hand, individuals can update a specific pheromone matrix or some pheromone matrices. On the other hand, all ants are allowed to update the pheromone. The implementation and optimisation of a solution is an important attribute in designing ACO algorithms.

3.3 Artificial ant agents

The multi-agent colony approach requires a number of ants to construct a colony. Agents individually construct solutions by using their own pheromone and heuristic information and conduct search procedures on particular areas of the graph. The agents are able to collaborate with each other to exchange information, share solutions, and update pheromone information. Therefore, the solutions generated by certain agents affect the pheromone information of other agents.

3.4 Pareto archive (offline/online/no-archive)

The Pareto archive is used by multi-objective ACO algorithms (García-Martínez et al., 2007). The Pareto set has to be maintained and rationalised during the execution of the algorithm. In most cases, the solution in the Pareto set is used to update the pheromone information. The multi-objective ACO algorithm indicates how the Pareto set is stored and used during implementation. A Pareto archive can be established in Pareto-based algorithms by using two approaches, namely, offline and online storage. These techniques allow the pheromone matrix or matrices to affect the state of the non-dominated set at any time.

3.5 Route optimisation

Static algorithms (e.g., Dijkstra and A*) are used to propose the shortest path without considering congestion, accident, or average vehicle speed. However, static algorithms cannot be used in the dynamic topology of vehicular networks. Optimal paths (i.e., shortest paths with specific constraints) changes with time. Therefore, VTSs should find the shortest path and optimise such a path continuously to overcome path changes. Thus, link weights should be changed by using real-time information to find the optimal route.

3.6 Automaticity and self-organisation

Vehicle drivers are the main users of traffic control systems and need to form an ad-hoc network to send and receive real-time traffic information. Self-organisation should always be considered in the design of traffic control systems. Centralised control is not suitable for vehicular networks because of the high-speed changes in topology (SjöbergBilstrup, 2009). To be effective, a traffic control system must be resilient to dynamic and unpredictable variations. Essential services should also be available for the long-term use of a decentralised system.

3.7 Scalability

Traffic control systems are expected to remain operational for an infinite period in a wide geographical area. New vehicles may enter and leave the communication range of current vehicles. Thus, the number of vehicles in communication ranges changes continuously and often becomes unpredictable. An effective vehicle traffic control system should be able to cope with the changes and challenges that originate from vehicular and wireless communication networks.

3.8 Architecture characteristics

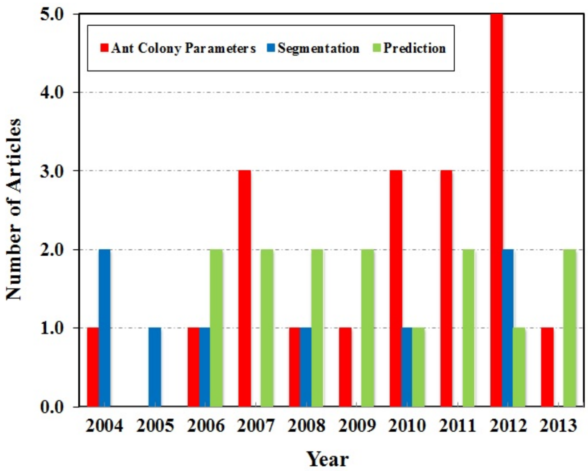
VTSs use road side units and infrastructure-based equipment (e.g., cameras, traffic lights, and speed sensors) to collect real-time traffic information. Values are then assigned to streets or roads by using the gathered information. In a source node, FANT agents are generated to explore the road map randomly or to find the shortest path via the probability function by considering the various constraints of vehicle congestion, accident likelihood, or travel time. When FANT agents find a path to a specific destination, BANT agents are used to report the obtained path to the source node. Vehicle drivers can then request for the found paths. Different VTSs apply different link values and probability functions to find the shortest path and consider various constraints.

4 Taxonomy of ACP, ACPre and ACS in VTS

Critical reviews on ant-based algorithms for VTSs (ant-based VTS) are lacking. Existing reviews mostly focuses on the effect of SI in VTS. Teodorovic (2003) and Teodorović (2008) classified and analysed results by using SI in VTS to present a model for a complex traffic problem. The main goal of these studies is to familiarise readers with basic concept of SI and its applications in traffic and transportation systems. These studies provides a general overview of SI algorithms in VTS, which includes agent-based models such as ACO, practical swarm optimisation, bee colony optimisation, and stochastic diffusion search. Given that SI is a broad field (Zhang, 2012), individual and specific discussions are needed to obtain insights into each one of the mentioned models.

Desai et al. (2011) reviewed existing MASs for VTS and categorised these systems into three different categories, namely, in-vehicle, infrastructure, and hybrid. The in-vehicle category discusses factors that are related only to the vehicle, such as user preference and vehicle speed. The factors in the infrastructure category are related to roadside equipment and traffic signals. The hybrid category involves a trade-off between the two aforementioned categories. However, this paper only focuses on techniques that use multi-agent ants. To the best of our knowledge, these techniques do not cover all existing methods regarding ant-based VTS.

Figure 1 Annual distribution of journal manuscripts for ACP, ACS, and ACPre (see online version for colours)



The survey includes the investigation of relevant articles on vehicle traffic congestion control from databases in IEEE Xplore, Science Direct, and Elsevier. These articles discuss the challenges in the application of ant colonies in vehicle traffic congestion problems. The survey expects to attract researchers and industry leaders in vehicle technology and environmental issues. Figure 1 represents the total number of studies conducted over the last decade (2004 to 2013). The number of papers on ACP reached the highest level in 2012. Limited studies were conducted on the ACS category in 2006, 2010, and 2012. ACPre-based papers started from 2006 and increased gradually by 2013. Figure 1 shows that the majority of conducted studies is related to ACP and ACPre. Many parameters in ACO depend on problem characteristics and searching strategies to find the problem space. Therefore, most studies are conducted in ACP to achieve the best value set for ACO in VTS. Considering the dynamicity of topology and traffic in vehicular networks, the prediction of upcoming network conditions is a necessary task. Thus, researchers have focused on ACPre. Tables 1 and 2 classify aforementioned papers into three main categories, namely, ACP, ACPre, and ACS. ACPre is divided into neuro/fuzzy and machine learning, whereas ACP is divided into variable and step-based systems. Moreover, objectives of each methodology are highlighted in Tables 1 and 2.

4.1 *Ant colony parameters*

This section discusses ant-based VTS algorithms that exploit the basic concept of ant colony algorithms. Researchers have changed the pre-existing variables and steps of the original ant colony algorithm (Dorigo et al., 1996) without adding any new concepts. The two types of ACP (i.e., variables and steps) are discussed as follows.

4.1.1 *Variable-based ACP*

Several key elements affect the ACO: number of ants (n), pheromone power (α), heuristic power (β), pheromone reduction (ρ), and ant speed (v). If these variables are not properly assigned, the ants will follow a previously found path that is not necessarily optimal because of the dynamic topology of vehicular networks. The assignment of appropriate values to these variables allows ants to search the route map to find the optimal path quickly and accurately. Liu et al. (2007) proposed a variable-based algorithm that uses the JIN method (Jin et al., 2002) on ACS algorithms for path routing optimisation. Based on their findings, the convergent speed of search procedures is directly related to the above mentioned variables. The probability of choosing a non-optimal path is equal to the convergence rate. Ok et al. (2011) proposed another variable-based short path-selection algorithm based on map link properties. Their findings show that increasing the number of ants reduces the probability of discovering non-preferred paths. Thus, the overall path length will be increased and the algorithm will converge toward the same path. However, a small number of ants cannot cause the above mentioned convergence rate. Nahar and Hashim (2011) proposed a traffic congestion control method based on different preferences. These preferences allow the algorithm to reduce average travelling time by adjusting ant colony variables. Their results show that the number of ants is directly correlated with the algorithm performance. The number of agents should not be less than the threshold defined in the algorithm (Nahar and Hashim, 2011).

Table 1 ACO applications in VTS considering ACPs

ACO category	*Reference*	*Title of paper*	*Objective*
Ant colony parameters			
Variable-based	Ok et al. (2011)	Ant colony optimisation approach for the preference-based shortest path search	Propose a preference-based shortest path using ACO.
	Nahar and Hashim (2011)	Modelling and analysis of an efficient traffic network using ant colony optimisation algorithm	Create an optimum traffic system and a platform for vehicle congestion control.
	Liu et al. (2007)	Ant colony system algorithm for path routing of urban traffic vehicles	Find an optimal path by considering time and distance.
Step-based			
Multi-ant	Boryczka and Bura (2013)	Ant colony optimisation for multi-criteria vehicle navigation problem	Propose a user-preference vehicle navigation system utilising multi-agent ant-based algorithm.
	Sur et al. (2012)	Analysis and modelling multi-breeded mean-minded ant colony optimisation of agent-based road vehicle routing management	Find the optimal distribution of traffic flows in the road network.
	Ghazy et al. (2012)	Threshold-based AntNet algorithm for dynamic traffic routing of road networks	Propose a threshold by using the AntNet algorithm for vehicle congestion problems.
	Cong et al. (2011)	A new ant colony routing approach with a trade-off between system and user optimum	Find the optimal distribution of traffic flows in the road network.
	Król and Mrożek (2011)	Swarm-based multi-agent simulation: a case study of urban traffic flow in the city of Wroclaw	Develop a model of road traffic environment that can be used to optimist transit traffic flow.
	Zong et al. (2010)	Multi-ant colony system for evacuation routing problem with mixed traffic flow	Manage evacuation routing problems with mixed traffic flow.
	Kammoun et al. (2010)	An adaptive vehicle guidance system instigated from ant colony behaviour	Use real-time traffic information to increase the global velocity of the road network.
	Foroughi et al. (2008)	Designing of a new urban traffic control system using modified ant colony optimisation approach	Minimise congestion time by the global management of most trips conducted in the control area.
	Weyns et al. (2007)	Anticipatory vehicle routing using delegate multi-agent systems	Propose a MAS, for anticipatory vehicle routing to avoid traffic congestion.
Probability function	Kponyo et al. (2012)	Real time status collection and dynamic vehicular traffic control using ant colony optimisation	Find an alternative optimum path to avoid traffic jams by using the average speed of vehicles at different roads.
	Ge et al. (2011)	Urban vehicle routing research based on ant colony algorithm and traffic rule restriction	Develop a vehicle path planning based on an ant colony algorithm by considering crossing traffic rule restrictions.
	Bedi et al. (2007)	Avoiding traffic jam using ant colony optimisation – a novel approach	Choose an alternative optimum path to avoid traffic jams and then resume the same path when the traffic is regulated.
	Hallam et al. (2004)	Optimisation in a road traffic system using collaborative search	Find the best path in a crowded city.

Table 2 ACO applications in VTS considering segmentation and prediction

ACO category	*Reference*	*Title of paper*	*Objective*
Segmentation			
	Claes and Holvoet (2012)	Cooperative ant colony optimisation in traffic route calculations	Propose a cooperative ACO for vehicle traffic route calculation.
	Narzt et al. (2010)	Self-organising congestion evasion strategies using ant-based pheromones	Investigate a technical implementation of SI applied to the traffic system and evaluate different evasion strategies for vehicles.
	Tatomir and Rothkrantz (2006)	Hierarchical routing in traffic using swarm-intelligence	Develop a prototype of a hierarchical routing system by splitting traffic networks into several smaller and less complex networks.
Prediction			
	Kurihara (2013)	Traffic-congestion forecasting algorithm based on pheromone communication model	Propose a method of congestion forecasting based on multi-agent coordination mechanism.
	Claes and Holvoet (2011)	Ant colony optimisation applied to route planning using link travel time predictions	Find routes that reduce vehicles travel time using link (road) travel time predictions.
	Tatomir et al. (2009)	Travel time prediction for dynamic routing using ant-based control	Propose a dynamic routing system based on ant-based control by using travel-time prediction.
	Ando et al. (2006b)	Pheromone model application to traffic congestion prediction	Propose a short-term prediction model for vehicle congestion control.
Hybrid techniques			
Machine learning	Yousefi and Zamani (2013)	The optimal routing of cars in the car navigation system by taking the combination of divide and conquer method and ant colony algorithm into consideration	Propose an optimal routing method to reduce vehicles travel time by combining divide and conquer, ACO and learning approaches.
	Jiang et al. (2007)	Solving the shortest path problem in vehicle navigation system by ant colony algorithm	Propose a shortest path search method by modifying pheromone update rule and adding learning strategy into ACO.
Neuro/fuzzy	Abbass et al. (2011)	Bio-inspired neuro-fuzzy-based dynamic route selection to avoid traffic congestion	Propose a bio-inspired neuro-fuzzy-based route selection system to avoid vehicle traffic congestion.
	Salehinejad and Talebi (2010)	Dynamic fuzzy logic-ant colony system-based route selection system	Introduce a multi-parameter route selection system by using fuzzy logic and neural network for pheromone update and future prediction, respectively.

4.1.2 Step-based ACP

Unlike the variable-based ACP, wherein the main concern is the assigning of values to the variables, the step-based ACP involves the following steps: ant distribution initialisation, ant probability function, pheromone updating, and the stopping phase. These steps aim to enhance the algorithm performance. Most studies discuss the first two steps of ant colony algorithms because they have the most impact on ACO.

a Multi-agent (ant distribution initialisation)

A MAS has been reported as a promising approach for dynamic problems (wherein involved parameters are not constant and can be change dynamically) because a MAS contains common features between the swarm behaviour of agents (e.g., ants, bees, termites, birds, or fish) and vehicular ad hoc networks (Kponyo et al., 2012; Garcia-Gonzalo and Fernandez-Martinez, 2012). Ant agents have proven to be superior to the agents introduced by Bonabeau et al. (1999), Dhillon and VanMieghem (2007), DiCaro and Dorigo (2011) and Schoonderwoerd et al. (1997). MAS is composed of a number of independent agents that are located in the problem space in a decentralised manner to solve dynamic problems. Weyns et al. (2007) proposed different types of cooperative ant agents, such as intention and exploration agents. They used a divide and conquer route map to reduce traffic congestion. Exploration agents investigate the environment, whereas intention agents are used to allocate road segments to different vehicles. Foroughi et al. (2008) proposed a modified ACO-based algorithm that minimises congestion time. They also used a path with minimum traffic and length to optimise travel time, fuel consumption, and air pollution. Kammoun et al. (2010) proposed an adaptive vehicle guidance system that intelligently finds the best route by using real-time changes in the network. To achieve dynamic traffic control and improve driver request management, this method used three types of agents, namely, city agent, road supervisor agent and intelligent vehicle-ant agent. A multi-agent evacuation model was introduced by Zong et al. (2010) to minimise the total evacuation time for vehicles and balance traffic load. Experiments have shown that MAS is more effective than a single agent system. Cong et al. (2011) developed a model to optimise dynamic traffic routing by using a two-step approach: network pruning and network flow optimisation. In the network-pruning phase, ant pheromone is removed after the best route is found by the agents to increase exploration rate. In the flow optimisation phase, which is based on ACO with the stench pheromone and collared pheromone, the agents correspond to the links selected in the network-pruning phase only. Moreover, this two-step approach reduces the computational burden by addressing complex, dynamic traffic control problems. Król and Mrożek (2011) proposed a tool to investigate the existing traffic flow of a specific city (i.e., Wroclaw) to optimise the vehicle congestion problem. An ant-based algorithm is used to simulate vehicle conditions in the route network. Sur et al. (2012) proposed a multi-breeded ACO system that defines different agent groups instead of agents. In this system, each agent only follows agents of the same type (breed). This type of classification has a significant contribution to the distribution of vehicles. A multi-breeded ACO system allows the pheromone update to be independent of time and incorporates several dependent factors in the update. Ghazy et al. (2012) proposed a new threshold-based ACO by using. They introduced a new ant called check ant, which is used to preserve the best path and discard the degraded routes. Bura and Boryczka (2010), Bura and Boryczka (2012) and Boryczka and Bura (2013) proposed a parallel version of dynamic fuzzy logic-ant colony system-based route selection system (Salehinejad and Talebi, 2010) by using a new type of pheromone update that occurs locally and globally. This method also returns blocked ants to the source instead of eliminating them at the destination.

b Probability function

Agents select the next hop in the problem graph (road map) by using the probability function. This function can be categorised into two classes, namely, probabilistic and heuristic models (Kammoun et al., 2010). Hallam et al. (2004) introduced ant behaviour-based search agents called soft cars and used these agents to implement and test the model on a road network. The soft cars choose paths with less loads, shorter distances, and more lanes, as well as paths that are frequently visited by soft cars. The dynamic system for avoiding traffic jams (Bedi et al., 2007) was designed by adopting an alternative path for each selected solution (route). This alternative path is used whenever the selected route is congested. The probability function of an ant algorithm is extended by using random function. The distributed intelligent traffic system (Kponyo et al., 2012) uses vehicle average speed as a parameter to determine the traffic condition. This system guides cars to paths with low traffic; thus, this system selects the best path more efficiently in compare with scenario where the agents select their path randomly. Ge et al. (2011) developed a crossing traffic rule-based ACO by combining traffic rules with the probability function. This method considers the restriction and delay of direction, satisfies the requirements of actual traffic environment, and enhances the validity of vehicle paths.

4.2 Ant colony segmentation

A problem space is divided into several less complex problems. The main idea of ACO segmentation is derived from divide and conquers approaches. Tatomir et al. (2004), Tatomir and Rothkrantz (2004) and Tatomir and Rothkrantz (2006) proposed a hierarchical routing system based on the ant algorithm. The hierarchical ant-based control algorithm (Tatomir and Rothkrantz, 2005) is combined with a hierarchical routing system to increase scalability. Narzt et al. (2010) introduced another technique that uses segmentation as a principle to overcome traffic control problems. In this approach, a novel pheromone update with a user-preference assignment system is adopted to divide the environment into different clusters. The segmentation procedure is conducted by using a pheromone engine, and a unique identifier is assigned to each car. Claes and Holvoet (2012) introduced a cooperative ant-based algorithm that results in less iteration. The essential concept for cooperation in this approach is the concept of the region. According to this concept, ants are interested in paths that lead to locations near their destination. To achieve a suitable form of segmentation, segments near each other are grouped together to form a region. Thereafter, routing is performed according to regions instead of segments. Claes and Holvoet (2012) evaluated cooperative and non-cooperative approaches and concluded that cooperative methods outperform non-cooperative methods in different aspects. For instance, the number of ants required in cooperative methods is less than the number of ants in non-cooperative methods.

4.3 Ant colony prediction

The two kinds of prediction methods are long-term prediction and the short-term prediction. Given that the control parameters of traffic congestion change briskly and that vehicles are highly dynamic, long-term prediction is imprecise for traffic problems. Therefore, most techniques in this area use short-term predictions. Ando et al. (2006b, 2006a) proposed a basic model for predicting traffic congestion by using the probe car system. The probe car system is a data collection method that uses vehicular ad hoc networks to collect real-time traffic information. Based on their investigation, the distance-based pheromone update mechanism outperforms other types of update mechanisms, such as braking and basic traffic pheromones. Tatomir et al. (2009) proposed a dynamic ant-based routing system that applies the ant algorithm to find the fastest path by using past, present, and future traffic information (i.e., travel-time prediction). To obtain accurate data in this routing system, the number of ant agents should be more than the number of cars. Claes and Holvoet (2011) also used link travel-time prediction to find paths with the shortest travel times. Kurihara et al. (2009) and Kurihara (2013) proposed a novel congestion-forecasting algorithm that is composed of major phases. First, the flow of traffic density is formulated by using a traffic density pheromone. Thereafter, based on the congestion-diffusion concept, the growth of the path queues is calculated, thus enabling the congestion forecasting of pheromones by monitoring the evaporation rate.

1 Hybrid techniques

Some studies have combined ACO with techniques such as fuzzy logic, neuro-network, and machine learning to create hybrid techniques. The most applied techniques are as follows. Fuzzy logic can be embedded to form a multi-preference routing system and can be applied to pheromone update procedures to detect the optimum multi-objective direction (e.g., number of traffic lights, lane width, and accident risk) between sources and destinations. Neural networks are used to predict the future time by using real-time traffic information. The main advantage of using machine learning in ACO is the ability of ACO to use passive information as learning input to optimise or predict future traffic conditions more effectively. Salehi-nezhad and Farrahi-Moghaddam (2007), Salehinejad and Talebi (2008) and Salehinejad and Talebi (2010) combined fuzzy logic and ACO to introduce a user-preference routing system. In this method, the traffic control centre and artificial neural network obtain current and future traffic data, respectively. Abbass et al. (2011) proposed another neuro-fuzzy and ACO amalgamation to find the most encouraging route based on driver preference. Similar to the previous method, artificial neural network is supplied to predict time. This system is able to prevent vehicle congestion and generate a priority-based path list for drivers. Jiang et al. (2007) proposed a navigation system that determines an optimal path (called the 'closest path' in the Dijkstra algorithm) by modifying pheromone updates and learning strategies. Yousefi and Zamani (2013) developed an ACO learning-based system. They considered that a network graph was composed of several sub-graphs. Learning the overall condition of the graph is possible by conducting several searches over each sub-graph. This procedure helps discover the shortest path and avoid choosing paths randomly.

Figure 2 provides a different overview of VTS categories and illustrates the relationship of some algorithms with other categories and user preference. For example, approaches proposed by Claes and Holvoet (2011) and Jiang et al. (2007) are in the prediction category because of using link travel time as prediction method. At the same time they investigate ACO parameters in their proposed algorithms, hence these approaches can be considered in parameter category as well. The approaches proposed by Salehinejad and Talebi (2010) and Abbass et al. (2011) are also found in the same category and are connected to the user preference, thus indicating that user preference is considered in these algorithms.

Figure 2 Tabular taxonomy of studied algorithms (see online version for colours)

<table>
<tr><th></th><th>PARAMETERS</th><th>PREDICTIONS</th><th>SEGMENTATION</th></tr>
<tr><td rowspan="2">Non-User Preferences</td><td rowspan="2">Liu et al. (2007)
Nahar and Hashim (2011)
Foroughi et al. (2008)
Kammoun et al.(2010)
Zong et al. (2010)
Cong et al. (2011)
Krol and Mrozek (2011)
Sur et al. (2012)
Ghazy et al. (2012)
Hallam et al. (2004)
Bedi et al. (2007)
Kponyo et al. (2012)
Ge at al. (2011)</td><td colspan="2">Tatomir et al. (2009)</td></tr>
<tr><td>Ando et al. (2006b)
Kurihara (2013)
Yousefi and Zamani (2013)</td><td>Claes and Holvoet (2012)
Narzt et al. (2010)
Tatomir and Rothkrantz (2006)</td></tr>
<tr><td rowspan="2">User Preferences</td><td colspan="2">Claes and Holvoet (2011)
Jiang et al. (2007)</td><td rowspan="2">-Nil-</td></tr>
<tr><td>Ok et al. (2011)
Weyns et al. (2007)
Boryczka and Bura (2013)</td><td>Abbass et al. (2011)
Salehinejad and Talebi (2010)</td></tr>
</table>

Table 3 Characteristics of VTSs using ACO

References	Characteristics					
	Technique	*Strategy*	*Topology*	*Infrastructure*	*Loop free*	*Traffic load balancing*
Kurihara (2013)	Stochastic	Proactive	Hierarchical	Decentralised	No	No
Yousefi and Zamani (2013)	Deterministic	Proactive	Hierarchical	Decentralised	No	Yes
Boryczka and Bura (2013)	Stochastic	Proactive	Flat	Decentralised	Yes	No
Claes and Holvoet (2012)	Stochastic	Proactive	Hierarchical	Centralised	No	No
Sur et al. (2012)	Stochastic	Proactive	Flat	Decentralised	Yes	Yes
Ghazy et al. (2012)	Stochastic	Proactive	Flat	Decentralised	Yes	No
Kponyo et al. (2012)	Stochastic	Proactive	Flat	Decentralised	No	Yes
Ge et al. (2011)	Deterministic	Proactive	Flat	Decentralised	Yes	No
Ok et al. (2011)	Deterministic	Proactive	Flat	Decentralised	Yes	No
Nahar and Hashim (2011)	Deterministic	Proactive	Flat	Decentralised	Yes	Yes
Cong et al. (2011)	Stochastic	Reactive	Hierarchical	Decentralised	No	Yes
Król and Mrożek (2011)	Stochastic	Proactive	Flat	Decentralised	Yes	Yes
Abbass et al. (2011)	Stochastic	Proactive	Flat	Decentralised	Yes	No
Claes and Holvoet (2011)	Stochastic	Proactive	Flat	Centralised	Yes	Yes
Kammoun et al. (2010)	Stochastic	Proactive	Flat	Decentralised	Yes	Yes
Zong et al. (2010)	Stochastic	Reactive	Flat	Decentralised	No	Yes
Salehinejad and Talebi (2010)	Stochastic	Proactive	Flat	Hybrid	Yes	No
Narzt et al. (2010)	Stochastic	Proactive	Hierarchical	Decentralised	Yes	No
Tatomir et al. (2009)	Deterministic	Proactive	Hierarchical	Decentralised	Yes	No
Foroughi et al. (2008)	Stochastic	Proactive	Flat	Decentralised	Yes	No
Jiang et al. (2007)	Stochastic	Proactive	Flat	Decentralised	Yes	No
Liu et al. (2007)	Deterministic	Proactive	Flat	Decentralised	Yes	No
Weyns et al. (2007)	Stochastic	Proactive	Flat	Hybrid	No	Yes
Bedi et al. (2007)	Stochastic	Proactive	Flat	Decentralised	Yes	Yes
Tatomir and Rothkrantz (2006)	Stochastic	Proactive	Hierarchical	Centralised	Yes	No
Ando et al. (2006b)	Stochastic	Reactive	Flat	Hybrid	No	Yes
Hallam et al. (2004)	Stochastic	Proactive	Flat	Decentralised	Yes	No

5 Characteristics of VTSs

On the basis of the aforementioned design challenges, the classification and characteristics of VTSs are discussed in this section. Related literature has classified VTSs in different ways. In the current paper, the deterministic/stochastic (as technique), reactive/predictive (as strategy), flat/hierarchical (as topology), centralised/decentralised (as infrastructure), and loop free/load balancing characteristics of VTS are discussed. These characteristics are elaborated in Table 3. According to Table 3, most of the VTSs use proactive and flat mechanisms. In addition to these characteristics, dynamicity and optimality should always be considered in VTS systems. Considering the nature of VTS systems, all algorithms and techniques introduced in this area should always be dynamic and optimal.

5.1 Technique

The first characteristic of VTS is related to the treatment of travel cost. This value can be computed in a deterministic or stochastic manner (Schmitt and Jula, 2006). Deterministic VTSs assign pre-defined and deterministic vales for links (roads) and disregard the dynamic and random nature of vehicle congestion. By contrast, stochastic VTSs consider the traffic condition of roads (links) and assign different values to links according to historical or/and real-time traffic information. Although stochastic VTSs require high computational capacity to process large amounts of vehicle traffic data, stochastic VTSs are not vulnerable to the random nature of vehicle congestion. Stochastic approaches outperform deterministic approaches in solving dynamic problems.

5.2 Strategy

Reactive VTSs use current traffic information and disregard the future conditions of vehicle congestion, whereas proactive VTSs use predictive models to estimate future congestion conditions. These systems are also called predictive approaches and use historical information for estimation. Low complexity and robustness are the main advantages of reactive VTSs and proactive VTSs, respectively. Thus, one of these characteristics can be selected based on the goal of the system (i.e., low complexity or robustness).

5.3 Topology

The entire routing map is considered one level in flat VTSs, and routing may occur between two arbitrary nodes as the source and destination. However, in hierarchical VTSs, the routing map includes either different levels or different regions (cluster or zone) for route decisions. Each level or region has one or more special nodes (i.e., cluster head and border node). hierarchical VTSs use these nodes to route vehicles between different regions, and vehicles use these nodes to enter new regions. Although both of these systems can be useful for vehicle routing, hierarchical VTSs manage dynamic changes better than flat VTSs. Thus, hierarchical VTSs outperform flat VTSs in terms of vehicle congestion control.

5.4 Infrastructure

Centralised VTSs use one node as a server/base station for discovering and maintaining routes; this node broadcasts routing information among vehicles. Thus, this node sustains system operations and its failure leads to the failure of the whole system. By contrast, in decentralised VTSs, each node gathers and builds routing table for its own use (DiCaro and Dorigo, 2011; Schmitt and Jula, 2006). Although these systems are adaptable to the dynamic nature of vehicular networks, vehicles (nodes) require powerful processing units to execute tasks. Reliability and a superior vision of the routing process are the main advantages of centralised VTSs (Schmitt and Jula, 2006). However, the drawbacks of centralised VTSs are delay and scalability, which can be solved by decentralised VTSs. Map segmentation is proposed in order to come up with a state-of-the-art decentralised VTS solution. In order to find the best path for each area of the map a management server is assigned to each segment and border nodes are responsible to perform the routing among the segments. More information about segmentation can be found in the study of Jabbarpour et al. (2013).

5.5 Loop free

If packets use a path that has no cycles to traverse the path between the source and destination, the algorithm is called ‘loop free’. To add this feature to an algorithm, a monitoring mechanism should be built to avoid any possible loop or cycle. The looping data packets have several negative network effects, such as throughput degradation and increased delays. Looping also wastes bandwidth and energy resources.

5.6 Traffic load balancing

Ants select paths at each intermediate node according to the distribution of deposited pheromones at each node. If a developed ant-based system uses a pattern that efficiently splits traffic between different paths from the source to the destination, such an approach method can be claimed to apply load balancing successfully.

6 Statistical review on ant-based VTS

The reviewed literature has shown that a significant amount of work has been conducted to use ant colonies for designing effective traffic congestion control methods. The statistics, evaluation metrics, and differences of the studied papers are discussed in this section. We provide some suggestions for the next stages of VTS research to achieve a precise and reliable structure for vehicle congestion control frameworks.

Table 4 Statistics on methodological approach in the reviewed papers

Description	*Percentage of papers*
Performs the empirical evaluations of algorithms	100
Reports the number of nodes and links in the simulation area	70
Reports the size of the simulation area	55
Uses travel time as an evaluation metric	55
Reports the number of nodes in the network	51
Does not specify the simulator or uses a self-made simulator	48
Compares produced algorithms with other algorithms	44
Reports simulation length	33
Reports the number of iterations	29
Uses path cost or length as an evaluation metric	29
Reports the number of vehicles in the simulation	26
Reports the relationship between ant parameters and system accuracy	22
Uses traffic distribution as an evaluation metric	22
Reports system overhead	7
Uses VV as an evaluation metric	7
Creates a public version of the protocol implementation	0

The results obtained from our statistical discussions are presented in Table 4 and Figure 3. In Table 4 various statistics regarding algorithms evaluation and involved parameters such as size of playground, nodes, links and length are represented. Moreover statistical information of some of functional parameters such as traffic evaluation and system accuracy are provided. Our investigations show four shortcomings in the studied papers:

1 system overhead and resource management are not considered

2 simulation procedures are not properly explained in most cases

3 algorithms are not evaluated sufficiently with other functional methods

4 a functional framework for VTS has not been developed in any of the studies.

Therefore, even though the reported protocols have appealing properties and good performance, their actual presentation and evaluation lack true scientific soundness.

6.1 Simulation tools for VTS

All of the studied methodologies have been assessed by using simulation tools. Choosing a realistic simulation is important for validating the proposed protocol or methodology. In Figure 3, we present a chart that illustrates the distribution of various simulation environments used in the studied papers. The investigation results indicate that more than half of the researchers (52%) used development environment applications such as MATLAB and Netlogo (Wilensky, 1999) to simulate the algorithms. Furthermore, 48% of the studied papers used either unknown simulation tools (33%) or self-made simulators (15%). Although traffic simulators (e.g., Sumo) have advanced in recent years, VTSs still lack specialised simulation tools. The use of self-made simulation tools is infeasible because the production of an accurate and robust simulation environment in computer science requires a considerable amount of effort and long-term collaboration between experts. Almost all of these collaborations lead to well-known simulation tools such as NS2 (Fall and Varadhan, 2007), OPNET (Documentation, 2003) and OMNET++ (Varga, 2010). Therefore, a self-made simulation environment is unrealistic and is unreliable for evaluating algorithms or methodologies. Tables 5 and 6 represents evaluation metrics of different algorithms which uses the same IDE. Algorithms which use MATLAB are represented in Table 5, while Table 6 includes the algorithms which utilise Visual C++ as implementation tool. Moreover, the result of modifying the mentioned parameters in these algorithms are presented in the impact section of each table.

In Tables 5 and 6, α and β are coefficient of pheromone trail and route cost [equation (1)], ρ is pheromone evaporation [equation (2)] and Q is constant value [equation (3)].

Table 6 Comparison of algorithms proposed by Liu et al. (2007) and Ge et al. (2011) using Visual C++

Reference	*Parameters*				*Impact on*
	ρ	Q	α	*Number of ants*	
Liu et al. (2007)	0.5	0.5	0.5	20	Convergence speed
Ge et al. (2011)	0.7	2	4	50	Route cost

6.2 Evaluation observation

Although the evaluation and assessment of proposed algorithms are two of the most important and critical parts of research, most studied papers contain the following problems:

1. insufficient information on the simulation tools and set-up processes
2. insufficient statistical information on evaluation metrics [e.g., vehicle velocity (VV) is an important evaluation metric for VTS; however, only 7% of papers addressed this metric]
3. the obtained results are not discussed or justified in most cases
4. only 44% of the papers have compared their results with state-of-the-art algorithms
5. only a few datasets and small-scale scenarios are used to evaluate the algorithms.

For most studies, a single metric is used for evaluation or no evaluation metric is defined at all. Therefore, in the following section, some critical and essential factors that should be considered in future works are discussed. Tables 7 and 8 indicate the various evaluation metrics used in VTS in terms of our three major categories (ACP, ACPre, and ACS) in chronological order. Discussion section of Tables 7 and 8 elaborates the impact of desired algorithm on given evaluation metrics. The most used and important evaluation metrics in VTS are presented in these tables. To mitigate the five mentioned shortcomings in ant-based VTS, the following suggestions should be considered.

1. A proper simulator should be developed to ensure that evaluations are accurate and reliable. Simulation metrics such as the number of ants, simulation duration, size of simulation area, and number of iterations should be defined clearly.
2. Evaluation metrics should be defined and considered. Some essential evaluation metrics such as travel time, VV, and path length should be evaluated in all algorithms.
3. Current methodologies should be compared, and authors should publish their simulation codes for accurate comparisons.
4. Algorithms should be defined clearly by using pseudo-codes and should be evaluated based on robustness, scalability, and overhead.

6.3 Comparison of studied papers

In this section, the similarities and differences between various studies are discussed with their probability functions. This information can provide insights on the studies conducted regarding ant-based VTS for designing and forming future frameworks. Some shortcomings, such as the lack of novel and effective probability functions, can be recognised through the following discussions.

Figure 3 Percentage of used simulators for the evaluation of studied algorithms (see online version for colours)

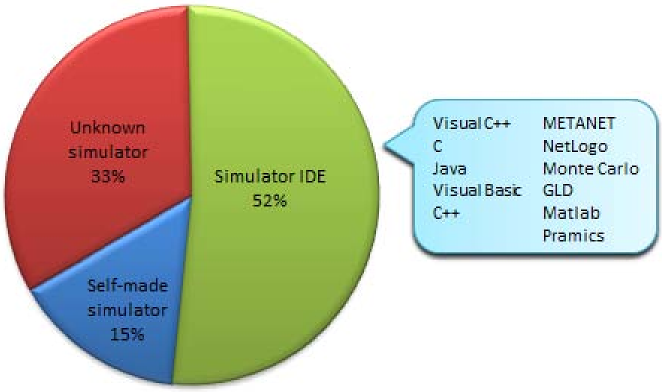


Table 5 Comparison of algorithms proposed by Salehinejad and Talebi (2010) and Abbass et al. (2011) using MATLAB

Reference	*Parameters*				*Impact on*
	ρ	Q	α	*Pheromone level*	
Salehinejad and Talebi (2010)	0.9	0.9	2	Very weak-very strong	Congestion-distance cost
Abbass et al. (2011)	0.9	-	0.5	0–5	Route ranking-user preference

Features such as data gathering, ant agent, new pheromone, and new ant type should also be considered in ant-based VTSs. Table 9 discusses these features in detail. Data gathering is an important phase that uses either historical information, real-time information, or both. Moreover, ant agents involved in VTSs can be virtual or real (vehicle). To enhance the performance of the ACO algorithm, the basic pheromone type is modified and a new type of pheromone is proposed, such as stench and coloured pheromones. New ant types, such as check and coloured ants, are also used.

Ants select the next hop to reach their destinations by using the probability function. Various probability functions in studied methodologies and their parameter descriptions are presented in Table 10. Most probability functions are similar to the basic probability function (represented in the first row of the Table 10); the only difference is minor modifications such as the addition or omission of parameters from the basic probability function.

6.4 Convergence analysis of ACO

One of the theoretical problems of ACO is its convergence analysis. Convergence means whether a given ACO algorithm is able to find an optimal solution when there are sufficient resources. Since ACO algorithms are stochastic search process, pheromone update may deter ACO from finding an optimal solution. There are two types of convergences: 'convergence in value and convergence in solution' (Dorigo and Blum, 2005) or 'reachability convergence and asymptotic convergence' (Chen and Sun, 2008).

- *Convergence in value or reachability convergence:* An algorithm has this type of convergence if $\lim_{t\to\infty} P_r(t) = 1$, where $P_r(t)$ is the probability that the algorithm generates an optimal solution at least once in the 1^{st} to t^{th} iterations (Chen and Sun, 2008).
- *Convergence in solution or asymptotic convergence:* An algorithm has this type of convergence if $\lim_{t\to\infty} P_s(t) = 1$, where $P_s(t)$ is the probability that the algorithm generates an optimal solution in the t^{th}iteration (Chen and Sun, 2008).

Gutjahr (2000, 2002) proved convergence with probability 1-ε for the optimal solution, and also for any solution, of a graph-based ant system as an specific ACO algorithm. The major shortcoming of their work is that its experimental performance is not explicit. Moreover it only focuses on graph-based ant system which is a particular ACO algorithm. Afterward, some research have been conducted on algorithms which apply a positive lower bound τ_{min} to all pheromone values. These types of algorithms are represented by $ACO_{\tau_{min}}$. Using lower band leads to assurance of generating an optimal solution. Stützle et al. (2002) and Dorigo and Stutzle (2004) represented a proof for convergence in value and solution (reachability and asymptotic convergence) for algorithms from $ACO_{\tau_{min}}$.

It is noteworthy to mention that the fine balance between exploration and exploitation is very important to the overall efficiency and performance of ACO. Finding a balance between these two characteristics is an optimisation problem and should be considered in algorithm design. Additionally the number of ant-agents has direct impact on convergence of ACO. Too few ant-agents are not effective, whereas too many of them may cause slow convergence. Convergence analysis of some SI algorithms such as PSO made some progress so far. However, robustness convergence and mathematical analysis of these types of heuristic algorithms are still lacking (Yang, 2011).

Table 7 Comparisons of evaluation metrics between ant-based algorithms in VTSs (recent three years)

References	*Evaluation metrics*	*Discussion*
Boryczka and Bura (2013)	Travel time path length number of ants	This approach compensates for the inability of dynamic fuzzy logic system (Salehinejad and Talebi, 2010) to support large datasets. This approach also outperforms the ACO and dynamic fuzzy logic system in terms of running time.
Kurihara (2013)	Prediction accuracy	This congestion-forecasting algorithm eliminates the need for probe cars and central management servers. This algorithm mainly focuses on predicting changes in traffic density and sudden accidents. However, the need to install several kinds of hardware in various network locations makes the implementation of this method expensive.
Yousefi and Zamani (2013)	Travel time number of ants path length	The proposed method finds paths by combining the divide and conquers method and ant colony algorithms. This algorithm compared with the Dijkstra and ant colony algorithms in terms of travel time. It exhibits better results than other methods in this comparison.
Sur et al. (2012)	Number of vehicles traffic distribution	Various types of vehicles are introduced in this approach to distribute traffic. This method provides better results in avoiding the stagnation of searching criteria compared with traditional ACO.
Ghazy et al. (2012)	Travel time number of ants	Based on theoretical investigations, the performance of this algorithm is $O(n^2)$ for road networks with n nodes. The number of ants and travel time are improved by approximately 12% and 3.13%, respectively.
Kponyo et al. (2012)	VV number of vehicles waiting time	In this approach, VV, waiting time and the number of stopped vehicles are improved compared with the random next-node selection approach.
Claes and Holvoet (2012)	Travel time number of iteration	In this region-based algorithm, every ant agent has to carry (save) additional information to find their target region faster. Hence, although the solution is found quickly, the system uses more resources.
Ok et al. (2011)	Path length number of links	Although the best values for a set of parameters (e.g., number of ants and iteration) are introduced to improve ACO performance, the algorithm is unsuitable for finding the shortest path based on preference.
Nahar and Hashim (2011)	Traffic distribution travel time	This approach is analogous to proposed algorithm by Ok et al. (2011). However this technique does not perform well when the number of agent in the network is less than 100 agents, which leads to high overhead. Travel time can be reduced ranging from 21% to 39% by using this approach.
Cong et al. (2011)	traffic distribution number of ants	This pruning-optimisation approach mitigates the dynamic traffic routing problem but does not consider user preference and only optimises traffic according to the system-optimum concept.
Król and Mrożek (2011)	Travel time number of vehicles	This method enables the distribution of traffic across multiple servers. Thus, this method can be used to manage traffic flow across very complex road networks with high traffic volume.
Ge et al. (2011)	Path length travel time	This Pareto-type method uses a novel network storage structure to make the path-planning algorithm functional. This phase is important for generating the adjacent matrix of the network.
Abbass et al. (2011)	Prediction accuracy traffic flow	Route ranking that is based on user preference is provided, and prediction is used to improve the decision-making system. On the basis of the resulting priority-based list, the lowest cost path is selected.
Claes and Holvoet (2011)	Travel time traffic distribution relationship of parameters and accuracy prediction on shortest path	A set of ACO parameters that provides a suitable trade-off between exploration and exploitation is found and discussed. Moreover, the APT algorithm outperforms the A* algorithm in terms of average travel time.

Table 8 Comparisons of evaluation metrics between ant-based algorithms in VTSs (2010 and below)

References	*Evaluation metrics*	*Discussion*
Narzt et al. (2010)	Travel time number of vehicles	The pheromone engine designed in this paper is efficient for analysing traffic information. However, the pheromone trail is not actively monitored because of the cluster-based platform.
Kammoun et al. (2010)	VV number of vehicles travel time	The next node is selected based on two methods: heuristic and probabilistic. The probabilistic method outperforms the heuristic method. VV is increased by an average of 15% compared to static algorithms (e.g., Dijkstra).
Salehinejad and Talebi (2010)	Travel time number of ants path length convergent speed	This algorithm is executed locally for each vehicle. Moreover, it outperforms the ant colony system and A* algorithms in terms of convergent speed and average cost.
Zong et al. (2010)	Number of vehicles	A multi-agent approach that solves the evacuation problem by using vehicle congestion load balancing. This approach outperforms ACS in terms of evacuation time.
Tatomir et al. (2009)	Travel time	four different types of vehicle navigation systems are used to evaluate this algorithm: 1) use of ant-based control algorithms; 2) sending traffic information every 30 min; 3) sending traffic information every 10 min; and 4) using the Dijkstra algorithm. The algorithm that uses ant-based control algorithm has better travel time.
Foroughi et al. (2008)	Path length path traffic	Along with path length, path traffic is added to the traffic control system as an optimisation metric.
Liu et al. (2007)	Convergent speed	A set of parameters are introduced to maintain the operation of the ACO algorithm in a steady state with high convergent speed. The parameter set is related to specific problem space and may not work for other algorithms such as proposed algorithm by Ok et al. (2011) and Nahar and Hashim (2011).
Weyns et al. (2007)	Traffic distribution	The use of a multi-agent mechanism provides this system with more flexibility for dynamic conditions. However this algorithm does not investigates scalability issues.
Bedi et al. (2007)	Path length number of ants	The random function is added to the probability function to increase the exploration rate of agents in the problem space.
Jiang et al. (2007)	Path length number of ants convergent speed number of iteration	Three different transition rules based on density, distance, and angle between two paths are used in this algorithm. The angle-based transition rule outperforms the other two rules in terms of path length. An agent is called a 'dead ant' if it cannot find a feasible path from source to destination.
Tatomir and Rothkrantz (2006)	Travel time	this approach uses hierarchical ant-based control algorithm to provide scalability for its routing system. Travel time is reduced by quickly reacting to new changes (e.g., congestion or accidents).
Ando et al. (2006b)	Travel time prediction accuracy relationship of parameters and accuracy prediction on shortest path	Two types of vehicles are used in this approach: general and commercial (e.g., buses or taxis). Thus, different pheromone types are assigned to different vehicle types. High prediction accuracy is obtained by using pheromone prediction.
Hallam et al. (2004)	Path length number of ants	This parameter-based method investigates the best available values to be set for ACO. However, this method cannot support fairness.

Table 9 Features and specifications of various VTSs that use ACO

Vehicle traffic systems			*References*	*Data gathering*	*Ant agent*	*New pheromone*	*New ant*
Ant colony parameters	Variable-based		Nahar and Hashim (2011)	Historical	Real	No	No
			Ok et al. (2011)	Historical	Virtual	No	No
			Liu et al. (2007)	Historical	Virtual	No	No
	Step-based						
		Multi-ant	Boryczka and Bura (2013)	Real-time	Virtual	Yes	No
			Ghazy et al. (2012)	Historical	Virtual	No	Yes
			Sur et al. (2012)	Historical	Virtual	Yes	Yes
			Cong et al. (2011)	Real-time	Real	Yes	Yes
			Król and Mrożek (2011)	Historical	Real	No	No
			Zong et al. (2010)	Real-time	Real	Yes	Yes
			Kammoun et al. (2010)	Real-time	Real	No	Yes
			Foroughi et al. (2008)	Real-time	Virtual	No	No
			Weyns et al. (2007)	Real-time	Virtual	No	Yes
		Probability Function	Kponyo et al. (2012)	Real-time	Real	No	No
			Ge et al. (2011)	Historical	Virtual	No	No
			Bedi et al. (2007)	Real-time	Virtual	No	No
			Hallam et al. (2004)	Hybrid	Virtual	Yes	Yes
Segmentation			Claes and Holvoet (2012)	Hybrid	Real	Yes	No
			Narzt et al. (2010)	Real-time	Real	Yes	No
			Tatomir and Rothkrantz (2006)	Real-time	Virtual	No	No
Prediction			Kurihara (2013)	Real-time	Real	Yes	No
			Claes and Holvoet (2011)	Historical	Virtual	No	No
			Tatomir et al. (2009)	Hybrid	Virtual	Yes	Yes
			Ando et al. (2006b)	Real-time	Real	Yes	No
	Hybrid tchnique		Yousefi and Zamani (2013)	Historical	Virtual	Yes	No
			Abbass et al. (2011)	Hybrid	Virtual	Yes	No
			Salehinejad and Talebi (2010)	Hybrid	Virtual	Yes	No
			Jiang et al. (2007)	Hybrid	Virtual	Yes	Yes

Current studies regarding to convergence have mostly discussed whether an ACO algorithm is able to reach to the optimal solution while its iteration time tends to infinity or not. However they did not given an insight about how fast these solutions can converges. Hence, it is necessary to examine convergence time of ACO and finding out its relationship with the pheromone rate. Convergence time indicates how much iteration ACO algorithms requires converging to the optimal solution. In this context the probability that an artificial ant picks a component of the optimal solution is a crucial factor which has an impact on the ACO. The pheromone rate is the essential factor of ACO algorithms which is utilised to simulate the biologic behaviour of ants.

Convergence time analysis contains the following stages:

1 the iteration time that the pheromone rate spends on reaching the objective value

2 the convergence time that is calculated with the objective pheromone rate in expectation (Huang et al., 2009).

The first method is proper for the situation in which a integrated low bound of pheromone rate is definite. The second method enhances the first one in three aspects:

1 assessing the iteration time that a pheromone rate spends in order to reach the objective value, and is called as first reaching time

2 calculating the expected convergence time based on the objective pheromone rate

3 assessing the ACO runtime by using first reaching time and convergence time.

The results attained from experiments by Huang et al. (2009) show that, constant increment of pheromone rate is worse than no-increment pheromone rate per irritation on average. Moreover, the convergence times can be reduced when the lower band of pheromone and pheromone evaporation rate values become large. Increment of the number of ants cannot improve the performance of an ACO algorithm when the increment of pheromone rate is constant. However, the convergence time could be reduced in some special cases when the number of ants is larger. It can be shown that a more pheromone rate per iteration can reduce the runtime of the ACO algorithms (Huang et al., 2009).

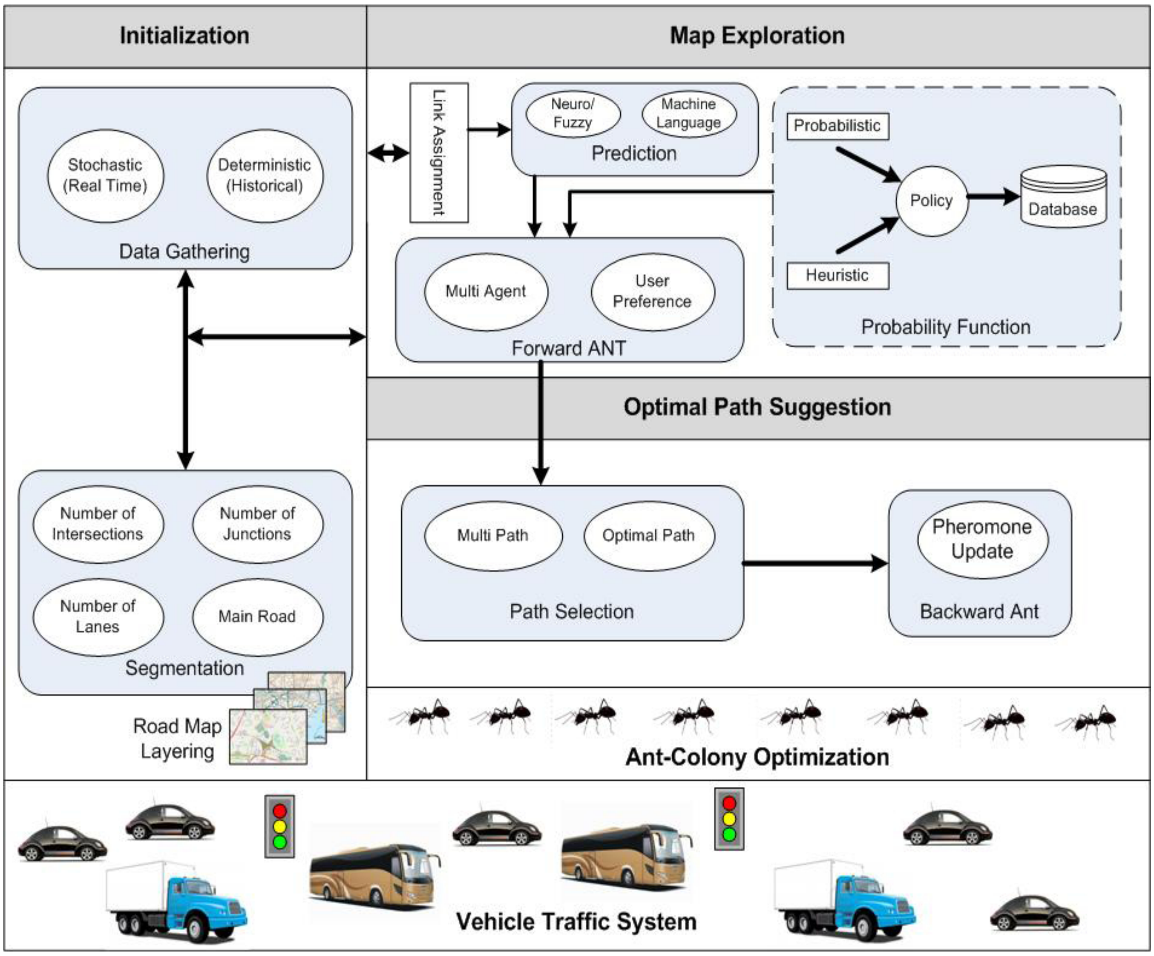
Table 10 Different probability functions

Reference	*Probability function*	*Description*
Dorigo (1992)	$P_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}{\sum_{h \in tab_k} [\tau_{ih}(t)]^\alpha [\eta_{ih}(t)]^\beta}$ (basic probability function)	Intensities α and β are of relatively important and can be used to stress the importance of trail $\tau_{ij}(t)$ and route cost $\eta_{ij}(t)$. tab_k is the set of visited nodes for ant *k*.
Hallam et al. (2004)	$P_{ij}^k(t) = \frac{[\tau_{ij}^t][\eta_{ij}]^\beta [nol_{ij}]^\lambda [1/noc_{ij}]^\delta}{\sum_{l \in Allowed_k} [\tau_{ij}^t][\eta_{ij}]^\beta [nol_{ij}]^\lambda [1/noc_{ij}]^\delta}$	Intensities α, β, λ, and δ are the relative importance that can be used to stress on the importance of the trail $[\tau_{ij}]$, visibility $[\eta_{ij}]$, number of lanes $[nol_{ij}]$, and number of cars $[noc_{ij}]$ respectively.
Bedi et al. (2007)	$P_{ij}^k(t) = \frac{((\tau_{ij}(t))^\alpha \eta_{ij}^\beta}{(\sum_{l \in allowed_k} ((\tau_{ij}(t))^\alpha \eta_{ij}^\beta)}$ + Random function	$\tau_{ij}(t)$ is the intensity of the pheromone trail on edge (i, j) at time *t*, η_{ij} (visibility factor)= $1/d_{ij}$ (d_{ij} is the distance between nodes *i* and *j*. α, β are the parameters that control the relative importance of the pheromone trail vs. visibility.
Kammoun et al. (2010)	$P_{itinerary}^i = \frac{(Q_{itinerary}^i)^\alpha (\frac{W_{shortest-itinerary}}{W_{itinerary}^i}))^\beta}{\sum_{j=1} (Q_{itinerary}^j)^\alpha (\frac{W_{shortest-itinerary}}{W_{itinerary}^j})^\beta}$	$Q_{itinerary}^i$ is the quality of itinerary *i*, $W_{itinerary}^i$ is the itinerary weight representing the itinerary length, and $W_{shortest-itinerary}^i$ is the length of the shortest possible itinerary. α and $\beta \in [0, 1]$ represent the itinerary intensities.
Salehinejad and Talebi (2010)	$P_{ij}^k = \frac{\tau_{ij}^\alpha \Pi_{l \in parameters} \xi_{ij_l}^{-\alpha_l}}{\sum_{h \notin tab_k} \tau_{ih} \Pi_{l \in parameters} \xi_{ih_l}^{-\alpha_l}}$	τ_{ij} is the direct route pheromone intensity from junction *i* to *j*. Parameter α controls the importance of τ_{ij}. The $tabu_k$ list is the set of direct blocked routes (visited nodes). The cost function of each parameter *l* is $\xi_{ij_l}^{-\alpha_l}$, where ξ_{ij_l} is normalised in $1 =< \xi_{ij_l} =< 10$ and the significance of each *l* is adjustable using α_l for all parameters.
Claes and Holvoet (2011)	$P_{ij} = \frac{(1-\gamma)\tau_{ij}^\alpha \cdot \gamma \eta_{ij}^\beta}{\sum_{(i,n) \in S} (1-\gamma)\tau_{in}^\alpha \cdot \gamma \eta_{in}^\beta}$	Intensities α, β and λ are the relative importance that can be used to stress the importance of the trail (τ_{ij}) and visibility (η_{ij}).
Ok et al. (2011)	$P_{ij}^k(t) = \frac{1/([\tau_{ij}(t)][\eta_{ij}(t)]^\beta)}{\sum_{u \in N_i^k(t)} 1/([\tau_{iu}(t)][\eta_{iu}(t)]^\beta)}$	$N_i^k(t)$ is the set of candidate nodes connected to node *i*, with respect to ant *k*. $\tau_{ij}(t)$ represents the pheromone from node *i* to node *j*. $\eta_{ij}(t)$ is a heuristic function that is defined as the inverse of the distance between node *i* and *j*. vParameter β is a positive constant and is used to amplify the influence of the heuristic function.
Cong et al. (2011)	$P_c\{j\|i\} = \frac{(max\{\tau_{min} \cdot \tau_{ij}\})^\alpha}{\sum_{l \in N_{i,c}} (max\{\tau_{min} \cdot \tau_{il}\})^\alpha}, \forall j \in N_{i,c}$	τ_{ij} is the pheromone level previously deposited by the ants on link *(i, j)*. Parameter α determines the relative importance of τ_{ij}. The feasible neighbourhood $N_{i,c}$ of ant *c* at node *i* is the set of nodes that are connected to *i* and the unvisited nodes of ant *c* in the current iteration. τ_{min} is the minimum pheromone level on each link, and guarantees the low pheromone level, and thus preventing the denominator of the formula from becoming *zero*.

Table 10 Different probability functions (continued)

Reference	*Probability function*	*Description*
Claes and Holvoet (2012)	$P^c_{ij} = (1-\lambda)P_{ij} + \lambda\sigma_{ij}(r)$	r is the region the ant is attempting to reach, P_{ij} is the probability of choosing edge (i,j) from (20), and λ is a weighing factor between the vehicle specific and region-specific information. $\sigma_{ij}(r)$ is the region-specific pheromones for every region r reachable through that edge.

Figure 4 Generic framework for ant-based VTS (see online version for colours)



7 A generic framework for ant-based VTS

The proposed framework in Figure 4 aims to reduce vehicle traffic congestion by using ACO. Initialisation, map exploration, and optimal path suggestion are the phases involved in VTS (Section 3). However, ACO is performed in the second and third phases of VTS. To the best of our knowledge, this framework is the first framework that is related to ant-based VTS. The framework has six components and their explanations are as follows.

- *Segmentation:* Considering that congestion in vehicular networks is dynamic and varies with time, one of the main components of this framework is segmentation. In this component, road maps are divided into a number of segments with different sizes. Segment size is assigned based on factors such as number of junctions, intersections, and road lanes. To overcome the dynamic nature of vehicular networks, routing is accomplished for each segment individually instead of for the whole map. Moreover, road map layering is used to reduce computing overhead and to increase coverage area. More information about layering can be found in the study of Jabbarpour et al. (2013).
- *Data gathering:* This component contains two sub-components, namely, deterministic and stochastic components, which are used to gather historical and real-time traffic information, respectively (Section 6.1).

- *Link assignment:* By using the two above components, various values (weights) are assigned to road map links (roads). Therefore, each link weight is changed as time passes based on traffic conditions. Artificial intelligence techniques, such as machine learning and neuro/fuzzy, are used to predict upcoming minutes to provide a comprehensive view of future vehicle traffic conditions.
- *Forward ant:* In this component, ants start to explore their own segment based on multi-objectives, called user preference (e.g., paths that are toll free, low-accident risk, and have few traffic lights). Different agents search different segments to reduce search time and system overhead. The number of ants is determined based on segment size.
- *Probability function:* The probability of choosing the next hop is determined in this component by using probabilistic or heuristic functions, and values are also assigned for links in the link assignment component. A probability table (database) is created for each node in the map according to the on link probability. Different policies can be used for different times of the day because traffic conditions at night differ from traffic conditions during the day. Hence, real-time data has higher priority at night than historical data, which is important during the day.
- *Backward ant:* When an ant reaches its destination, the ant backtracks to increase the pheromone levels of the links in the discovered path. This pheromone update can be accomplished by using a constant value or a mathematical formula based on length, traffic, or link travel time.

8 Discussion and conclusions

A comprehensive taxonomy and state-of-the-art ant-based VTS are presented in this paper along with their pros and cons. This paper introduces three categories of ant-based VTS: ACP, ACPre, and ACS. The ACP category attempts to adopt ACO into a method for vehicle traffic control by modifying a basic ACO. Although most of the proposed algorithms are similar, these algorithms differ in two major aspects:

1 assignment of proper values for ACO variables

2 improvement of ACO steps.

However, dynamic and quick changes to vehicle network topology are not considered properly in ACP. To address this problem,ACS and ACPre are proposed. In ACS, the main concern is the application of the divide and conquers approach in the VTS system by dividing the road map into several segments. Most approaches perform segmentation by applying a fixed size grid, whereas our investigation shows that the size of segments should not be equal. Segmentation should be based on different factors, such as the number of junctions, intersections, lanes, and main roads (vs. auxiliary roads). ACPre uses prediction methods such as artificial neural network based on historical and real-time traffic data to facilitate the monitoring of vehicle topology. However, this category suffers from high system overhead because of the large amounts of data that must be processed.

In addition to investigating the flaws of related works, we also indicated a number of challenges regarding the simulation and evaluation of ant-based VTS. Finally, we outlined a generic framework that defines scientifically sound experiments and performance evaluations. Considering these shortcomings, the design of a robust, scalable, and real-time framework is challenging. Each category has its own limitations. Thus, a hybrid framework that benefits from the advantages of ACP, ACPre, and ACS is proposed in this study. This framework uses the layering and prediction approaches simultaneously. System overhead, which is the direct effect of prediction, can be mitigated. Moreover, segmentation is used in our framework to manage the dynamicity of vehicular networks. Traffic patterns are not identical for different times of day. Thus, different decisions should be taken during the day and during the night. Hence, a policy management unit that applies different rules for different hours of the day is required to obtain a reliable and realistic traffic control system. To the best of our knowledge, such a policy unit does not exist in any of the previous methods but is applied in our framework. Some limitations of simulation environments and real world implementation are mentioned. Simulation-based studies should be complemented with mathematical models that allow the study of very large systems and general algorithm properties. These models should also encourage fair comparisons among other algorithms. Real implementation should occur after simulation to address problems and challenges that cannot be handled in a simulation environment. To validate the proposed framework, the framework should be designed and developed in a practical manner.

Acknowledgements

This research is supported by the Ministry of Higher Education (MOHE) and University of Malaya (Project No. HIR-MOHEB00009)

References

Abbass, S., Khan, M.S., Ahmed, K., Abdullah, M. and Farooq, U. (2011) 'Bio-inspired neuro-fuzzy based dynamic route selection to avoid traffic congestion', *International Journal of Scientific and Engineering Research*, Vol. 2, No. 6, pp.284–289.

Abdelaziz, A., Osama, R.A. and Elkhodary, S. (2013) 'Using the harmony search algorithm for reconfiguration of power distribution networks with distributed generation units', *Journal of Bioinformatics and Intelligent Control*, Vol. 2, No. 3, pp.237–242.

Abdelaziz, A., Osama, R.A. and Elkhodary, S.M. (2012) 'Application of ant colony optimization and harmony search algorithms to reconfiguration of radial distribution networks with distributed generations', *Journal of Bioinformatics and Intelligent Control*, Vol. 1, No. 1, pp.86–94.

Abdelsalam, H.M. and Mohamed, A.M. (2012) 'Optimal sequencing of design projects' activities using discrete particle swarm optimisation', *International Journal of Bio-Inspired Computation*, Vol. 4, No. 2, pp.100–110.

Ahangarikiasari, H., Saraji, M.R. and Torabi, M. (2013) 'Investigation of code complexity of an innovative algorithm based on aco in weighted graph traversing and compare it to traditional aco and bellman-ford', *Journal of Bioinformatics and Intelligent Control*, Vol. 2, No. 1, pp.73–78.

Ali, L. and Sabat, S.L. (2012) 'Particle swarm optimization based universal solver for global optimization', *Journal of Bioinformatics and Intelligent Control*, Vol. 1, No. 1, pp.95–105.

Ando, Y., Fukazawa, Y., Masutani, O., Iwasaki, H. and Honiden, S. (2006a) 'Performance of pheromone model for predicting traffic congestion', in *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, pp.73–80, ACM.

Ando, Y., Masutani, O., Sasaki, H., Iwasaki, H., Fukazawa, Y. and Honiden, S. (2006b) 'Pheromone model: application to traffic congestion prediction', in *Engineering Self-Organising Systems*, Vol. 3910, pp.182–196, Springer.

Bandyopadhyay, S. (2011) *Traffic Congestion Management using RFID & Wireless Technologies*, Indian Institute of Management, Calcutta.

Bedi, P., Mediratta, N., Dhand, S., Sharma, R. and Singhal, A. (2007) 'Avoiding traffic jam using ant colony optimization-a novel approach', in *International Conference on Computational Intelligence and Multimedia Applications*, volume1, pp.61–67. IEEE.

Beni, G. and Wang, J. (1989) 'Swarm intelligent', in *Proceedings of the Seventh Annual Meeting of the Robotics Society of Japan*, pp.425–428, RSJ Press.

Blum, C. (2005) 'Beam-aco – hybridizing ant colony optimization with beam search: an application to open shop scheduling', *Computers & Operations Research*, Vol. 32, No. 6, pp.1565–1591.

Boehlé, J., Rothkrantz, L.J.M. and Wezel, M. (2008) *Cbprs: A City Based Parking and Routing System*, ERIM Report Series Research in Mmanagement, Report No. ERS-2008-029-LIS.

Bonabeau, E., Dorigo, M. and Theraulaz, G. (1999) *Swarm Intelligence: From Natural to Artificial Systems*, Vol. 4, Oxford University Press, New York.

Bonabeau, E., Henaux, F., Guérin, S., Snyers, D., Kuntz, P. and Theraulaz, G. (1998) 'Routing in telecommunications networks with ant-like agents', in *Intelligent Agents for Telecommunication Applications*, pp.60–71, Springer.

Boryczka, M. and Bura, W. (2013) 'Ant colony optimization for the multi-criteria vehicle navigation problem', in *Agent-Based Optimization*, Vol. 456, pp.29–53, Springer.

Budinská, I., Kasanickỳ, T. and Zelenka, J. (2012) 'Production planning and scheduling by means of artificial immune systems and particle swarm optimisation algorithms', *International Journal of Bio-Inspired Computation*, Vol. 4, No. 4, pp.237–248.

Bura, W. and Boryczka, M. (2010) 'Ant colony system in ambulance navigation', *Journal of Medical Informatics & Technologies*, Vol. 15, pp.115–124.

Bura, W. and Boryczka, M. (2012) 'Ant colony optimization for the pareto front approximation in vehicle navigation', in *Computational Collective Intelligence. Technologies and Applications*, Vol. 7654, pp.493–502, Springer.

Cai, X., Fan, S. and Tan, Y. (2012) 'Light responsive curve selection for photosynthesis operator of apoa. *International Journal of Bio-Inspired Computation*, Vol. 4, No. 6, pp.373–379.

ChandraMohan, B. and Baskaran, R. (2012) 'A survey: ant colony optimization based recent research and implementation on several engineering domain', *Expert Systems with Applications*, Vol. 39, No. 4, pp.4618–4627.

Chen, Y. and Sun, H. (2008) 'Convergence of ant colony optimization on first-order deceptive systems', in *IEEE International Conference on Granular Computing*, pp.158–163, IEEE.

Claes, R. and Holvoet, T. (2011) 'Ant colony optimization applied to route planning using link travel time predictions', in *IEEE International Symposium on Parallel and Distributed Processing Workshops and Phd Forum (IPDPSW)*, pp.358–365, IEEE.

Claes, R. and Holvoet, T. (2012) 'Cooperative ant colony optimization in traffic route calculations', in *Advances on Practical Applications of Agents and Multi-Agent Systems*, Vol. 155, pp.23–34, Springer.

Colorni, A., Dorigo, M., Maniezzo, V. and Trubian, M. (1994) 'Ant system for job-shop scheduling', *Belgian Journal of Operations Research, Statistics and Computer Science*, Vol. 34, No. 1, pp.39–53.

Cong, Z., DeSchutter, B. and Babuska, R. (2011) 'A new ant colony routing approach with a trade-off between system and user optimum', in *14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp.1369–1374, IEEE.

Costa, D. and Hertz, A. (1997) 'Ants can colour graphs', *Journal of the operational Research Society*, Vol. 48, No. 3, pp.295–305.

Cui, Z. and Cai, X. (2009) 'Integral particle swarm optimization with dispersed accelerator information', *Fundamenta Informaticae*, Vol. 95, No. 4, pp.427–447.

Cui, Z., Cai, X., Zeng, J. and Yin, Y. (2010) 'Pid-controlled particle swarm optimization', *Multiple-Valued Logic and Soft Computing*, Vol. 16, No. 6, pp.585–609.

Dantzig, G.B. and Ramser, J.H. (1959) 'The truck dispatching problem', *Management Science*, Vol. 6, No. 1, pp.80–91.

DeCampos, L.M., Fernandez-Luna, J.M., Gámez, J.A. and Puerta, J.M. (2002) 'Ant colony optimization for learning Bayesian networks', *International Journal of Approximate Reasoning*, Vol. 31, No. 3, pp.291–311.

DeCampos, L.M., Puerta, J., et al. (2008) 'Learning Bayesian networks by ant colony optimisation: searching in two different spaces', *Mathware & Soft Computing*, Vol. 9, No. 3, pp.251–268.

Deneubourg, J-L., Aron, S., Goss, S. and Pasteels, J.M. (1990) 'The self-organizing exploratory pattern of the argentine ant. *Journal of Insect Behavior*, Vol. 3, No. 2, pp.159–168.

Derrar, H.n., Ahmed-Nacer, M. and Boussaid, O. (2012) 'Particle swarm optimisation for data warehouse logical design', *International Journal of Bio-Inspired Computation*, Vol. 4, No. 4, pp.249–257.

Desai, P., Loke, S.W., Desai, A. and Singh, J. (2011) 'Multi-agent based vehicular congestion management', in *IEEE Intelligent Vehicles Symposium (IV)*, pp.1031–1036, IEEE.

Dhillon, S. and VanMieghem, P. (2007) 'Performance analysis of the antnet algorithm', *Computer Networks*, Vol. 51, No. 8, pp.2104–2125.

DiCaro, G. and Dorigo, M. (2011) 'Antnet: Distributed stigmergetic control for communications networks', arXiv preprint arXiv:1105.5449.

Documentation, O.M. (2003) 'Opnet technologies' [online] http://www.opnet.com (accesed 22 December 2013).

Dorigo, M. (1992) *Optimization, Learning and Natural Algorithms*, PhD thesis, Politecnico di Milano, Italy [in Italian].

Dorigo, M. and Blum, C. (2005) 'Ant colony optimization theory: a survey', *Theoretical Computer Science*, Vol. 344, No. 2, pp.243–278.

Dorigo, M. and Gambardella, L.M. (1997) 'Ant colony system: a cooperative learning approach to the traveling salesman problem', *IEEE Transactions on Evolutionary Computation*, Vol. 1, No. 1, pp.53–66.

Dorigo, M., Maniezzo, V. and Colorni, A. (1996) 'Ant system: optimization by a colony of cooperating agents', *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics*, Vol. 26, No. 1, pp.29–41.

Dorigo, M., Maniezzo, V., Colorni, A. and Maniezzo, V. (1991) *Positive Feedback as a Search Strategy*, Technical Report, No. 91-016, Citeseer.

Dorigo, M. and Stutzle, T. (2004) *Ant Colony Optimization*, MIT Press, Cambridge, MA.

Duan, H., Yu, Y., Zou, J. and Feng, X. (2012) 'Ant colony optimization-based bio-inspired hardware: survey and prospect', *Transactions of the Institute of Measurement and Control*, Vol. 34, Nos. 2–3, pp.318–333.

Fall, K. and Varadhan, K. (2007) 'The network simulator ns-2' [online] http://www.isi.edu/nsnam/ns (accessed 22 December 2013).

Fan, H., Hua, Z., Li, J-J. and Yuan, D. (2004) 'Solving a shortest path problem by ant algorithm', in *Proceedings of International Conference on Machine Learning and Cybernetics*, Vol. 5, pp.3174–3177, IEEE.

Foroughi, R., Montazer, G.A. and Sabzevari, R. (2008) 'Design of a new urban traffic control system using modified ant colony optimization approach', *Iranian Journal of Science and Technology*, Vol. 32, No. B2, p.167.

Garcia-Gonzalo, E. and Fernandez-Martinez, J. (2012) 'A brief historical review of particle swarm optimization (pso)', *Journal of Bioinformatics and Intelligent Control*, Vol. 1, No. 1, pp.3–16.

García-Martínez, C., Cordón, O. and Herrera, F. (2007) 'A taxonomy and an empirical analysis of multiple objective ant colony optimization algorithms for the bi-criteria tsp', *European Journal of Operational Research*, Vol. 180, No. 1, pp.116–148.

Ge, Y., Wang, J., Wang, G. and Jiang, F. (2011) 'Urban vehicle routing research based on ant colony algorithm and traffic rule restriction', in *High Performance Networking, Computing and Communication Systems*, Vol. 163, pp.95–102, Springer.

Ghazy, A.M., EL-Licy, F. and Hefny, H.A. (2012) 'Threshold based antnet algorithm for dynamic traffic routing of road networks', *Egyptian Informatics Journal*, Vol. 13, No. 2, pp.111–121.

Ghoseiri, K. and Nadjari, B. (2010) 'An ant colony optimization algorithm for the bi-objective shortest path problem', *Applied Soft Computing*, Vol. 10, No. 4, pp.1237–1246.

Gunes, M., Sorges, U. and Bouazizi, I. (2002) 'Ara-the ant-colony based routing algorithm for manets', in *International Conference on Parallel Processing Workshops*, pp.79–85, IEEE.

Gutjahr, W.J. (2000) 'A graph-based ant system and its convergence', *Future Generation Computer Systems*, Vol. 16, No. 8, pp.873–888.

Gutjahr, W.J. (2002) 'Aco algorithms with guaranteed convergence to the optimal solution', *Information processing letters*, Vol. 82, No. 3, pp.145–153.

Hallam, N., Hartley, M., Blanchfield, P. and Kendall, G. (2004) 'Optimisation in a road traffic system using collaborative search', in *IEEE International Conference on Systems, Man and Cybernetics*, Vol. 2, pp.2008–2012, IEEE.

Hansen, P. (1980) 'Bicriterion path problems', in *Multiple Criteria Decision Making Theory and Application*, Vol. 177, pp.109–127, Springer.

Huang, H., Wu, C-G. and Hao, Z-F. (2009) 'A pheromone-rate-based analysis on the convergence time of aco algorithm', *Systems, Man and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, Vol. 39, No. 4, pp.910–923.

Jabbarpour, M.R., Malakooti, H., Jalooli, A. and MdNoor, R. (2013) 'A dynamic vehicular traffic control using ant colony and traffic light optimization', *International Conference on Systems Science, ICSS '113*.

Jiang, Y., Wang, W-L. and Zhao, Y-W. (2007) 'Solving the shortest path problem in vehicle navigation system by ant colony algorithm', in *Proceedings of the 7th WSEAS International Conference on Signal Processing, Computational Geometry & Artificial Vision*, pp.188–192, World Scientific and Engineering Academy and Society (WSEAS).

Jin, F., Hong, B. and Gao, Q. (2002) 'Path planning for free-flying space robot using ant algorithm', *Robot*, Vol. 24, No. 6, pp.526–529.

Kammoun, H.M., Kallel, I., Alimi, A.M. and Casillas, J. (2010) 'An adaptive vehicle guidance system instigated from ant colony behavior', in *IEEE International Conference on Systems Man and Cybernetics (SMC)*, pp.2948–2955, IEEE.

Kennedy, J. and Eberhart, R. (1995) 'Particle swarm optimization', *Proceedings of IEEE International Conference on Neural Networks*, pp.1942–1948, Perth, WA, Australia.

Kponyo, J.J., Kuang, Y. and Li, Z. (2012) 'Real time status collection and dynamic vehicular traffic control using ant colony optimization. in *International Conference on Computational Problem-Solving (ICCP)*, pp.69–72. IEEE.

Król, D. and Mrożek, M. (2011) 'Swarm-based multi-agent simulation: a case study of urban traffic flow in the city of wroclaw', in *Computational Collective Intelligence. Technologies and Applications*, Vol. 6923, pp.191–200, Springer.

Kroon, R. and Rothkrantz, L. (2003) 'Dynamic vehicle routing using an abc-algorithm', *Collected Papers on the PITA Project*, p.21.

Kurihara, S. (2013) 'Traffic-congestion forecasting algorithm based on pheromone communication model', *Ant Colony Optimization-Techniques and Applications*, p.163.

Kurihara, S., Tamaki, H., Numao, M., Yano, J., Kagawa, K. and Morita, T. (2009) 'Traffic congestion forecasting based on pheromone communication model for intelligent transport systems', in *IEEE Congress on Evolutionary Computation, CEC '09*, pp.2879–2884, IEEE.

Lessing, L., Dumitrescu, I. and Stützle, T. (2004) 'A comparison between aco algorithms for the set covering problem', in *Ant Colony Optimization and Swarm Intelligence*, Vol. 3172, pp.1–12, Springer.

Liu, J., Fang, Y. and Liu, Y. (2007) 'Ant colony system algorithm for path routing of urban traffic vehicles', in *IEEE International Conference on Automation and Logistics*, pp.1902–1907, IEEE.

Lucic, P. and Teodorovic, D. (2001) 'Bee system: modeling combinatorial optimization transportation engineering problems by swarm intelligence', in Preprints of the *TRISTAN IV Triennial Symposium on Transportation Analysis*, pp.441–445.

Lucic, P. and Teodorovic, D. (2002) 'Transportation modeling: an artificial life approach', in *14th IEEE International Conference on Tools with Artificial Intelligence*, pp.216–223, IEEE.

Lucic, P. and Teodorovic, D. (2003a) 'Computing with bees: attacking complex transportation engineering problems', *International Journal on Artificial Intelligence Tools*, Vol. 12, No. 03, pp.375–394.

Lucic, P. and Teodorovic, D. (2003b) 'Vehicle routing problem with uncertain demand at nodes: the bee system and fuzzy logic approach', in *Fuzzy Sets Based Heuristics for Optimization*, pp.67–82, Springer.

Maniezzo, V. (1999) 'Exact and approximate nondeterministic tree-search procedures for the quadratic assignment problem', *INFORMS Journal on Computing*, Vol. 11, No. 4, pp.358–369.

Marchesini, P. and Weijermars, W. A.M. (2010) *The Relationship between Road Safety and Congestion on Motorways*, SWOV Institute for Road Safety Research, Report No. R-2010-12.

Martens, D., DeBacker, M., Haesen, R., Baesens, B., Mues, C. and Vanthienen, J. (2006) 'Ant-based approach to the knowledge fusion problem', in *Ant Colony Optimization and Swarm Intelligence*, pp.84–95, Springer.

Martins, E.Q.V. (1984) 'On a multicriteria shortest path problem', *European Journal of Operational Research*, Vol. 16, No. 2, pp.236–245.

Mullen, R.J., Monekosso, D., Barman, S. and Remagnino, P. (2009) 'A review of ant algorithms', *Expert Systems with Applications*, Vol. 36, No. 6, pp.9608–9617.

Nahar, S. A.A. and Hashim, F.H. (2011) 'Modelling and analysis of an efficient traffic network using ant colony optimization algorithm', in *Third International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN)*, pp.32–36, IEEE.

Narzt, W., Wilflingseder, U., Pomberger, G., Kolb, D. and Hortner, H. (2010) 'Self-organising congestion evasion strategies using ant-based pheromones', *Intelligent Transport Systems, IET*, Vol. 4, No. 1, pp.93–102.

Ok, S-H., Seo, W-J., Ahn, J-H., Kang, S. and Moon, B. (2011) 'An ant colony optimization approach for the preference-based shortest path search', *Journal of the Chinese Institute of Engineers*, Vol. 34, No. 2, pp.181–196.

Okdem, S. and Karaboga, D. (2006) 'Routing in wireless sensor networks using ant colony optimization', in *Adaptive Hardware and Systems, AHS 2006, First NASA/ESA Conference on*, pp.401–404, IEEE.

Parpinelli, R.S., Lopes, H.S. and Freitas, A.A. (2002) 'Data mining with an ant colony optimization algorithm', *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 4, pp.321–332.

Radwan, A.A., Mahmoud, T.M. and Hussein, E.H. (2011) 'Antnet-rslr: A proposed ant routing protocol for manets', in *International Electronics, Communications and Photonics Conference (SIECPC)*, pp.1–6, IEEE, Saudi.

Ramos, V. and Almeida, F. (2004) 'Artificial ant colonies in digital image habitats-a mass behaviour effect study on pattern recognition', arXiv preprint cs/0412086.

Reddy, A.S. and Vaisakh, K. (2013) 'Environmental constrained economic dispatch by modified shuffled frog leaping algorithm', Journal of Bioinformatics and Intelligent Control, Vol. 2, No. 3, pp.216–222.

Rosenberg, A.L. (2010) 'Ants in parking lots', in *Euro-Par 2010-Parallel Processing*, Vol. 6272, pp.400–411, Springer.

Sahu, R.K., Panda, S., Rout, U.K. and Raul, P. (2013) 'Application of gravitational search algorithm for load frequency control of multi area power system', *Journal of Bioinformatics and Intelligent Control*, Vol. 2, No. 3, pp.200–210.

SalehiMaleh, M., Soleymani, S., RasouliNezhad, R. and Ghadimi, N. (2013) 'Using particle swarm optimization algorithm based on multi-objective function in reconfigured system for optimal placement of distributed generation', *Journal of Bioinformatics and Intelligent Control*, Vol. 2, No. 2, pp.119–124.

Salehi-nezhad, H. and Farrahi-Moghaddam, F. (2007) 'An ant based algorithm approach to vehicle navigation', *First Joint Congress on Fuzzy and Intelligent Systems*.

Salehinejad, H. and Talebi, S. (2008) 'A new ant algorithm based vehicle navigation system: a wireless networking approach', in *International Symposium on Telecommunications, IST 2008*, pp.36–41, IEEE.

Salehinejad, H. and Talebi, S. (2010) 'Dynamic fuzzy logic-ant colony system-based route selection system', *Applied Computational Intelligence and Soft Computing*.

Schmitt, E.J. and Jula, H. (2006) 'Vehicle route guidance systems: classification and comparison', in *IEEE Intelligent Transportation Systems Conference, ITSC '06*, pp.242–247, IEEE.

Schoonderwoerd, R., Holland, O.E., Bruten, J.L. and Rothkrantz, L.J. (1997) 'Ant-based load balancing in telecommunications networks', *Adaptive behavior*, Vol. 5, No. 2, pp.169–207.

Senge, S. and Wedde, H.F. (2012a) '2-way evaluation of the distributed beejama vehicle routing approach', in *IEEE Intelligent Vehicles Symposium (IV)*, pp.205–210, IEEE.

Senge, S. and Wedde, H.F. (2012b) 'Bee-inpired road traffic control as an example of swarm intelligence in cyber-physical systems', in *38th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA)*, pp.258–265, IEEE.

Shmygelska, A. and Hoos, H.H. (2005) 'An ant colony optimisation algorithm for the 2d and 3d hydrophobic polar protein folding problem', *BMC bioinformatics*, Vol. 6, No. 1, pp.30.

Sim, K.M. and Sun, W.H. (2003) 'Ant colony optimization for routing and load-balancing: survey and new directions. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, Vol. 33, No. 5, pp.560–572.

SjöbergBilstrup, K. (2009) *Predictable and Scalable Medium Access Control for Vehicular ad Hoc Networks*, PhD thesis, Halmstad University.

Spalding, S. (2008) *RACQ Congested Roads Report*, Technical Report 0408, RACQ Vehicle Technologies Department.

Stützle, T., Dorigo, M., et al. (2002) 'A short convergence proof for a class of ant colony optimization algorithms', *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 4, pp.358–365.

Stützle, T. and Hoos, H.H. (2000) 'Max–min ant system', *Future Generation Computer Systems*, Vol. 16, No. 8, pp.889–914.

Subha, M. and Anitha, R. (2009) 'An emerging ant colony optimization routing algorithm (acora) for manets', *Journal of Computer Applications*, Vol. 2, No. 3, pp.29.

Sur, C., Sharma, S. and Shukla, A. (2012) 'Analysis & modeling multi-breeded mean-minded ant colony optimization of agent based road vehicle routing management', in *International Conference For Internet Technology and Secured Transactions*, pp.634–641, IEEE.

Suson, A. (2010) *Dynamic Routing using Ant-based Control*, PhD thesis, Master thesis, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology.

Tatomir, B., Kroon, R. and Rothkrantz, L. (2004) 'Dynamic routing in traffic networks using antnet', in *Ant Colony Optimization and Swarm Intelligence*, Vol. 3172, pp.424–425, Springer.

Tatomir, B. and Rothkrantz, L. (2004) 'Dynamic traffic routing using ant based control', in *IEEE International Conference on Systems, Man and Cybernetics*, Vol. 4, pp.3970–3975, IEEE.

Tatomir, B. and Rothkrantz, L. (2005) 'H-abc: A scalable dynamic routing algorithm', *Recent Advances in Artificial Life*, World Scientific Publishing, Singapore.

Tatomir, B. and Rothkrantz, L. (2006) 'Hierarchical routing in traffic using swarm-intelligence', in *IEEE Intelligent Transportation Systems Conference, ITSC '06*, pp.230–235, IEEE.

Tatomir, B., Rothkrantz, L.J. and Suson, A-C. (2009) 'Travel time prediction for dynamic routing using ant based control', in *Proceedings of the Winter Simulation Conference (WSC)*, pp.1069–1078, IEEE.

Teodorovic, D. (2003) 'Transport modeling by multi-agent systems: a swarm intelligence approach', *Transportation Planning and Technology*, Vol. 26, No. 4, pp.289–312.

Teodorović, D. (2008) 'Swarm intelligence systems for transportation engineering: Principles and applications', *IEEE Transaction on Transportation Research Part C: Emerging Technologies*, Vol. 16, No. 6, pp.651–667.

Teodorović, D. and Dell'Orco, M. (2005) 'Bee colony optimization – a cooperative learning approach to complex transportation problems', in *Advanced OR and AI Methods in Transportation: Proceedings of 16th Mini–EURO Conference and 10th Meeting of EWGT*, pp.51–60, 13–16 September, Publishing House of the Polish Operational and System Research, Poznan.

Teodorovic, D., Lucic, P., Markovic, G. and Orco, M.D. (2006) 'Bee colony optimization: principles and applications', in *8th Seminar on Neural Network Applications in Electrical Engineering*, pp.151–156, IEEE.

Varga, A. (2010) 'Omnet++', in *Modeling and Tools for Network Simulation*, pp.35–59, Springer.

Wang, Y-j. and Tao, J. (2007) 'An optimal algorithm for parking guidance in modern large-scale parking lots', *Computer Simulation*, Vol. 11, No. 047, pp.12–47.

Weyns, D., Holvoet, T. and Helleboogh, A. (2007) 'Anticipatory vehicle routing using delegate multi-agent systems', in *IEEE Intelligent Transportation Systems Conference, ITSC 2007*, pp.87–93, IEEE.

Wilensky, U. (1999) *Netlogo: Center for Connected Learning and Computer-based Modeling*, Northwestern University.

Yang, X-S. (2011) 'Review of meta-heuristics and generalised evolutionary walk algorithm. *International Journal of Bio-Inspired Computation*, Vol. 3, No. 2, pp.77–84.

Yang, X-S., Cui, Z., Xiao, R., Gandomi, A.H. and Karamanoglu, M. (2013) *Swarm Intelligence and Bio-inspired Computation: Theory and Applications*, Elsevier.

Yousefi, P. and Zamani, R. (2013) 'The optimal routing of cars in the car navigation system by taking the combination of divide and conquer method and ant colony algorithm into consideration', *International Journal of Machine Learning and Computing*, Vol. 3, pp.44–48.

Yu, B., Cui, Z. and Zhang, G. (2013) 'Artificial plant optimization algorithm with correlation branches', *Journal of Bioinformatics and Intelligent Control*, Vol. 2, No. 2, pp.146–155.

Zhang, Y. (2012) 'Application swarming intelligence in bioinformatics: survey', *Journal of Bioinformatics and Intelligent Control*, Vol. 1, No. 1, pp.27–39.

Zhao, J., Jia, L., Chen, Y. and Wang, X. (2006) 'Urban traffic flow forecasting model of double rbf neural network based on pso', in *Sixth International Conference on Intelligent Systems Design and Applications*, Vol. 1, pp.892–896, IEEE.

Zhu, Q., Qian, L., Li, Y. and Zhu, S. (2006) 'An improved particle swarm optimization algorithm for vehicle routing problem with time windows. in *Evolutionary Computation, CEC 2006, IEEE Congress on*, pp.1386–1390, IEEE.

Zong, X., Xiong, S., Fang, Z. and Li, Q. (2010) 'Multi-ant colony system for evacuation routing problem with mixed traffic flow', in *IEEE Congress on Evolutionary Computation (CEC)*, pp.1–6, IEEE.