

Directional routing and scheduling for green vehicular delay tolerant networks

Yuanyuan Zeng · Kai Xiang · Deshi Li · Athanasios V. Vasilakos

Published online: 6 June 2012
© Springer Science+Business Media, LLC 2012

Abstract The vehicle delay tolerant networks (DTNs) make opportunistic communications by utilizing the mobility of vehicles, where the node makes delay-tolerant based “carry and forward” mechanism to deliver the packets. The routing schemes for vehicle networks are challenging for varied network environment. Most of the existing DTN routing including routing for vehicular DTNs mainly focus on metrics such as delay, hop count and bandwidth, etc. A new focus in green communications is with the goal of saving energy by optimizing network performance and ultimately protecting the natural climate. The energy-efficient communication schemes designed for vehicular networks are imminent because of the pollution, energy consumption and heat dissipation. In this paper, we present a directional routing and scheduling scheme (DRSS) for green vehicle DTNs by using Nash Q-learning approach that can optimize the energy efficiency with the

considerations of congestion, buffer and delay. Our scheme solves the routing and scheduling problem as a learning process by geographic routing and flow control toward the optimal direction. To speed up the learning process, our scheme uses a hybrid method with forwarding and replication according to traffic pattern. The DRSS algorithm explores the possible strategies, and then exploits the knowledge obtained to adapt its strategy and achieve the desired overall objective when considering the stochastic non-cooperative game in on-line multi-commodity routing situations. The simulation results of a vehicular DTN with predetermined mobility model show DRSS achieves good energy efficiency with learning ability, which can guarantee the delivery ratio within the delay bound.

Keywords DTN · Vehicular networks · Energy-efficient · Opportunistic routing · Scheduling

Y. Zeng (✉) · D. Li
School of Electronic Information, Wuhan University, Wuhan, China
e-mail: zengyy@whu.edu.cn

D. Li
e-mail: dsli@whu.edu.cn

K. Xiang
School of Information Management, Hubei University of Economics, Wuhan, China
e-mail: xksuckx@gmail.com

K. Xiang
National Engineering Research Center for Multimedia Software, Wuhan University, Wuhan, China

A. V. Vasilakos
Department of Computer and Telecommunications Engineering, University of Western Macedonia, Macedonia, Greece
e-mail: vasilako@ath.forthnet.gr

1 Introduction

Vehicular networks have been envisioned to be used in road safety, emergency responses and many intelligent transportation system based applications with the rapid growth of city transportation [1]. Multi-hop data delivery through vehicular networks is complicated because of the vehicle mobility and varied network environment like topology and traffic, etc. According to this, vehicular data is collected by opportunistic communications, i.e., a vehicle carries the packet until a relay vehicle is in its communication range and then help to forward it in delay-tolerant routing mechanism [2]. Delay Tolerant Networks are suitable for applications of vehicular networks [3]. Vehicular DTN can provide a large-scale communications by leveraging the large data storage and energy efficiency

of vehicles. The application examples include the use of vehicular DTN that provides low cost digital communication for remote villages [4] and vehicular sensing platforms for urban monitoring such as CarTel [5]. But, the vehicular DTN network topology changes dynamically, for the network lacks continuous connectivity and may become partitioned at any instant. The uncertainty of network environment in vehicle DTNs is a result of the mobility, limit wireless radio range, sparsity of mobile nodes, energy resources [6].

The existing DTN routing protocols mainly focus on the schemes to increase the likelihood of finding opportunistic paths [7]. But, it is difficult to model the DTNs and their possible connections in realistic network situations. This is especially worse for vehicular networks for varied traffic and topology in the networks. For DTNs, the methods of replicating data packets can improve the possibility of opportunistic connections, but it also brings the burden for storage and bandwidth resources. In order to utilize resources effectively in DTN, nodes need to predict the network environment and wisely allocate the network resources. The existing routing schemes mainly utilize metrics such as delay, hop count and bandwidth, etc.

Green ICT (information and communication technology) is focused for reported high carbon emissions each year. In terms of the global carbon emissions, some analysts have reported that ICT accounts for 2–2.5 % of all harmful emissions, which is equal to the global aviation industry [8]. The scheme designed for Green vehicular networks aims to reduce energy consumption and then help to protect our environment by less carbon compound emissions.

Grossglauser et al. [2] suggest that delay-tolerant applications can take advantage of node mobility to significantly optimize the network capacity. Though network environment varied dynamically in vehicular DTNs, the traffic situations are often related with time, traveling roadmap and other factors. These factors are useful information to make data delivery in DTNs. For example, in Wuhan city, the harsh hour often occurs from 8 a.m. till 10 a.m. in the morning and 5 p.m. till 7 p.m. in the afternoon, and the traffic situations are often serious around the main road and city center areas. Beyond this, most of the vehicles have their rules of mobility. For an example, buses have their fixed route and most taxis move around some city areas especially some “popular” zones. Reinforcement learning could be used to predict the vehicular related traffic distribution and network topology without much prior knowledge. In this paper, we present a directional routing and scheduling scheme (DRSS) for green communications in vehicular DTNs based on Nash Q-learning approach, which is designed to explicitly optimize the energy efficiency by considering congestion, buffer and

delay. DRSS can predict the network model and update the communication strategy by an immediate cost for the current state. DRSS converges to an optimal policy as long as all state-action pairs are continually updated. For on-line, multi-commodity communications, different agents contend with each other. DRSS uses Nash Q-learning for each agent to learn and perform the best actions for the network.

The main contributions of this paper include:

- DRSS utilizes a hybrid method to carry, geographically forward and replicate traffic data packets toward the optimal direction with slot allocation by predicting the network environment.
- DRSS optimizes the routing decision to achieve green communications by considering congestion, buffer and delay bound.
- DRSS uses multi-agent Q-learning approach for on-line multi-commodity communication situations and provides the Nash equilibrium strategy.

The following of the paper is organized as follows. Section 2 presents the related work. Section 3 presents some definitions and models. We present the details of DRSS in Sect. 4. Section 5 is simulation results. Finally, Sect. 6 concludes this paper.

2 Related work

The existing routing schemes in DTN can be classified into two categories that are based on replication and routing, respectively. The kind of routing schemes focus on whether the packets can be successfully received by the destination with less consideration of timeliness and rate [9]. Then the routing schemes use a “carry and forward” approach [10] that store and carry the data locally, then eventually delivering it either to the destination or to a relay deemed to meet the destination sooner.

Replication schemes are to replicate copies of a packet in the hope that it will succeed in reaching its destination. The schemes are commonly used to maximize the probability of a packet being successfully transferred. The kinds of routing schemes differ in the replication model and ways to cut down the replication overhead. Epidemic routing protocol [11] is a naive flooding, but it wastes resources and degrade the network performance. Spray and Wait routing scheme [12] replicates the packet by bounding the number of replicas, but it is feasible with large amounts of local storage and enough bandwidth. There are some routing schemes that limit the packet replication with the consideration of storage constraints [13–16]. RAPID [17] is an intentional DTN routing protocol, which treat routing as a resource allocation problem and decides on packet

replication according to per-packet utilities. Replication scheme increases the opportunistic communication probability with the tradeoff of big burden on network resources like available storage and link bandwidth.

Forwarding schemes maintain at most one copy of a packet in the network during routing process. The forwarding based routing schemes forward the packet toward the direction of optimizing routing metrics. Jain et al. [18] proposed a modified Dijkstra algorithm based on a time-dependent Graph. Hay et al. [19] proposed a deterministic DTN routing and scheduling when the contact times between nodes are known in advance or can be predicted. Bulut et al. [20] proposed a conditional shortest path routing by using metric called conditional intermeeting time. For forwarding routing, there is one copy of each packet sent during the routing process. So it is difficult to gather the knowledge of future contacts and network environment information. Recently, intelligent algorithms are utilized in DTN routing. Ahmed et al. [21] proposed a Bayesian classifier based DTN routing framework to infer the optimized routing. Huang et al. [22] proposed a fuzzy logic based routing in DTNs, which is exploited to select the close-by intermediate node on the path to the destination. Dvir et al. [23] proposed a dynamic backpressure routing in DTNs.

Different from general DTNs, vehicular DTNs face with more challenges for large scale sparse networks, variable delays, nonexistence of an end-to-end path, etc. Pereira et al. [24] addressed some specific issues such as routing, scheduling, traffic differentiation, contact time estimation for vehicular DTNs that can provide cost-effective support for applications. In vehicular DTNs, the vehicle mobility also creates opportunities for contacts, i.e., nodes can buffer and carry the packets until finds vehicles in the vicinity that can help packet delivery. The delay-tolerant based opportunistic connections for vehicular networks have been studied. The vehicle mobility pattern can be utilized to assist data delivery. Zhao et al. [10] presented vehicle-assisted data delivery in vehicular networks to forward the packets to the best road direction with the lowest data delivery delay. Leontiadis et al. [25] proposed Geographical Opportunistic Routing (GeOpps) by suggesting routes to select vehicles that are likely to move closer to the final destination of a packet. GeOpps calculates the shortest distance from packet's destination to the nearest point of vehicles' path and estimate the arrival of time of a packet to the destination. Ros et al. [26] proposed a broadcast algorithm with acknowledgment based on connected dominating set, which can provide high reliability and message efficiency. In wireless networks, scheduling is usually related with routing metrics to optimize the network resources. Zhou et al. [27] proposed a fully distributed scheduling scheme with the goals of

optimize the network video streaming. In their paper, the proposed scheduling scheme addresses the fairness problem to keep the balance between the selfish local motivation and global performance.

Green communications and computing have been proposed as a solution to addressing the growing cost and environmental impact of telecommunications recently. Sanctis et al. [28] discussed several techniques on energy efficient wireless networks towards green communications and outline challenges and open issues. Wang et al. [29] discussed various remarkable techniques toward green mobile networks. They summarized the current research related green mobile networks along with the taxonomy of energy–efficiency metrics.

For general vehicular networks and DTNs, energy efficiency is not the main concern [30]. The routing metrics such as delay, bandwidth, and hop count are usually involved. The node energy can be used as a resource constraint, but few papers take it as the main optimization objective. Different from the previous work, we examine green routing and scheduling scheme for vehicular DTNs in this paper, and we propose a hybrid way of forwarding and replication methods according to network environment.

3 System models

We assume that: each node gets knowledge of its locations and is aware of its trajectory by equipping with GPS and pre-loaded digital maps. The node information of its vicinity can be known by periodic beacon messages. There are some static access points are deployed in DTN network that help to make connections and disseminate global network information. Through the above, each node is aware of its own location, the location of the destination, and the locations of its potential next hops. In our assumptions, no further traffic statistic of the network is needed.

In order to discuss the routing and scheduling problems, we need some definitions and models to describe the network. A vehicular DTN is composed of vehicles participate in the network with computing and communication capabilities. The links of the network may go up and down over time due to mobility and the event. In such DTNs, the routing is based on opportunistic contact of vehicles. More than one contact may be available between a pair of vehicles. In this case, routing metrics are used to make decisions on the optimal route and flow schedule selection. The contact schedule contains a set of time slots that are available for communications, which is related with communication performance. The routing strategies are used to deliver the packets over each hop by knowledge of the

networks. The packets will be buffered at each intermediate node because of the intermittent network, which enables packets to wait until the next hop is available.

3.1 Energy model

As in usual network communications, each node can be in one of the three working states: listening, send and receive data. We assume the system communication is in continuous time period called frames, which can be divided into time slots. DRSS helps to choose the relay and schedule for current transmissions. If no events occur on the current node, it will turn into sleep state to save energy.

The energy consumption of DTN nodes include energy consumed for transmitting E_t , receiving E_r and listening E_l . Here, we omit the energy consumption of sleep state. The energy consumption can be calculated as Formula (1), where e_e is the energy consumed by transceivers per second, and e_a is the energy consumed in the transmitter RF amplifier per second; e_p is the energy consumed for processing in the receivers, and e_l is the energy consumed for listening to the radio environment. e_l equals to e_e . Those parameters are determined by the design characteristics of transceivers. R is the transmission range. n is the power index of the channel path loss. T_t is the time for sending data. T_r is the time for receiving. T_l is the listening time. T is the time length of a cycle.

$$\begin{aligned} E_t &= (e_e + e_a R^n) T_t \\ E_r &= (e_e + e_p) T_r \\ E_l &= e_l T_l = e_l (T - T_t - T_r) \end{aligned} \quad (1)$$

3.2 Rate and buffer

We assume that communication time is divided into continuous equal frames, which can be then divided into continuous equal slots. Each node has the same frame structure and chooses transmit rate according to channel congestion and energy efficiency. The maximum rate of node u with its possible next hop can be determined from Formula (2). In the formula, C is the channel capacity. W is the bandwidth and P is transmission power; h is the channel gain, and N_0 is the noise power spectrum density; and F is the gap to ergodic channel capacity.

$$rate_t(u) = \frac{T_c}{T} C = \frac{T_c}{T} W \log_2 \left(1 + \frac{Ph}{N_0 WF} \right) \quad (2)$$

In which, T is the time of a frame. T_c is the opportunistic contacting time of the pair (u, v) in a frame, which is related with contacting topology changes and rate constraints. It is obvious that T_c/T is proportional to the schedule slot numbers.

The sending and receiving rate of node u with the possible communication downside and upside node in a frame can be formulated as shown in Formula (3).

$$rate(u) = rate_t(u) + rate_r(u) = \frac{T_t + T_r}{T} C \quad (3)$$

In which, $rate(u)$ is the total rate for node u . $rate_t(u)$ is the sending rate of node u . $rate_r(u)$ is the receiving rate of node u . T_t is the transmission time in a frame, and T_r is the receiving time in a frame. $(T_t + T_r)/T$ is the active fraction of slot allocation on current node for sending and receiving data on the current downside and upside communication link. For each node in a frame, there are some slots allocated for sending data and the others for receiving data according to communication scenarios, except for idle slots if no communications occur. There is: $T_t + T_r = T_c$. In each frame, we use λ_t and λ_r to represent the slot allocated fraction for sending and receiving data respectively. Considering a half-duplex network interface card for node-to-node communications, there is:

$$\lambda_t(u) + \lambda_r(u) \leq 1 \quad (4)$$

Considering the congestion constraints, for slot τ , it should satisfy Formula (5), where u' is in the interference node set $I(u)$ of node u :

$$\lambda_t(u) + \sum_{u' \in I(u)} \lambda'_t(u') \leq 1 \quad (5)$$

In protocol interference model, transmission methods are utilized to evaluate the interference impact. A node receiving from a neighbor should be spatially separated from any other transmitter by at least a distance D , i.e., an interference range. If distance between u and u' is less than D , then the two nodes interfere with the transmission.

Each node has a buffer to store and carry the packets that have not been transmitted. The node keeps the storage and bandwidth maximally utilized, dropping only when necessary. The buffer has limited maximal size and contains packets with its available space, i.e., node u has the available buffer size as $buf(u)$. Let $occ(u, \tau)$ be the number of bytes stored at node u at slot τ under current routing strategies. For all u, τ , there is:

$$occ(u, \tau) \leq buf(u) \quad (6)$$

3.3 Delay and packet mode

A delay bound T_{max} is given to limit the maximal delivery time for data packets. For each node u of the route path with the h th hop in the network, the distance of itself from the destination dst can be measured by the Euclid distance as: $d = \|u - dst\|$. Note that dst can be a destined vehicle node or an access point.

So an approximate average delivery velocity value \bar{v}_a can be calculated as Formula (7):

$$\bar{v}_a = |v_a| = \frac{\vec{d}}{slack} = \frac{\|u - dst\|}{T_{max} - \sum_{i=1}^h T_i} \tag{7}$$

In which, *slack* is the time left for routing that is equal to the remaining part of delay bound minus cumulative time on each hop for the packet. The velocity vector v_a is with the same direction as the direction of current node toward destination, i.e., the direction of \vec{d} .

For the $(h + 1)$ th hop, the velocity vector denoted as v is with direction that should satisfy Formula (8). In which, v_a is the average velocity that is needed achieve delay bound

destination. We use: *straightway*, *intersec-2-des*, and *intersec-against-des* to represent the three patterns. If a node carries the packet and its next hop is at the same direction with it toward the destination, then we find out the packet is delivered through vehicles on a straight way that does not change the driving direction either. In this case, we have: $\cos \theta = 1$. If we have $\cos \theta \leq 0$, it implies the next hop is against from the direction of current node and its destination pair, i.e., *intersec-against-des* pattern. If we have $\cos \theta > 0$ and $\cos \theta < 1$, it implies the next hop is toward the direction of current node and the destination pair, i.e., *intersec-2-des* mode. Formula (9) gives the definition of the three relay patterns.

$$relay_pattern = \begin{cases} straightway & \text{if } I(u, u') = \cos \theta = 1 \\ intersec - 2 - des & \text{if } I(u, u') = \cos \theta \in (0, 1) \\ intersec - against - des & \text{if } I(u, u') = \cos \theta \in [-1, 0] \end{cases} \tag{9}$$

according to Formula (7). And θ is the angle of current velocity v with the node-destination pair direction \vec{d} . So, Δv is the vector difference of the velocity on the $(h + 1)$ th hop and the required average velocity for delay bound. If Δv is less than zero, it cannot guarantee to achieve the real-time delivery toward the destination.

$$\Delta v = \bar{v} \cos \theta - \bar{v}_a \geq 0 \tag{8}$$

Figure 1 shows an illustration, in which v_a is the required average velocity for real-time delivery. And v_1, v_2 is the velocity on the possible next hop of node u that satisfies the above Formula (8), when geographic routing is from s to d . The node u is assigned with a routing indicator $I(u, u')$ for the next hop u' with the data packet, which is defined as the cosine value of the angle of current velocity with the node-destination pair direction, i.e., $\cos \theta$. A bigger $I(u, u')$ value indicates a higher delivery priority of the relay.

According to the mobility of vehicles, we can find three basic relay patterns in vehicular DTNs as: straight way, intersection toward destination and intersection against

3.4 Routing models

In geographic routing, only node moving toward the preferred direction within the delivery delay bound is probably selected as a relay. But for geographic routing problems in vehicular DTNs, the optimal node may be not connected to the destination, so the routing decision only based on velocity policy may make packets be routed toward the wrong direction that often leads to higher energy consumption. As shown in Fig. 1, if packet on u is forwarded in horizontal direction, the packet will not be successfully delivered because horizontal relay node is not connected toward the destination.

Though the connectivity related traffic pattern can be learned by Q-learning, it needs more time and waster energy to do this. To solve the problem, we combine two routing modes: probabilistic forwarding and replication together in vehicular DTNs to try to speed up the learning process. Which mode to choose depends on the packet pattern learned from the network environment. If the current packet is in “*intersec-against-des*” pattern, then the packet will not be delivered to the next hop. The node just carry the packet and wait for the other contact. Otherwise, if the packet pattern is “*straightway*”, the node will forward the packet to its possible next hop. If the packet pattern is “*intersec-2-des*”, the replication method is used to replicate the packet and deliver to all the next hop. The node will replicate and send the packet according to routing indicator, i.e., replicate the packet and deliver it to the next hop with $I \in (0, 1)$.

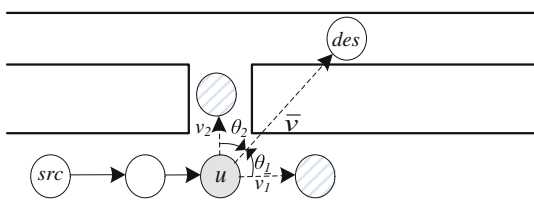


Fig. 1 An example for geographic routing

3.5 Problem definition

The problem objective of DRSS is to minimize the total routing and rate allocation cost on the route path, i.e., to minimize the cost denoted as \check{R} for each node u of the cumulative h hops on the route path according to the routing and rate allocation strategy set Π , which is shown as in Formula (10).

• **Objective:**

$$\check{R}(\Pi) = \sum_1^h \check{R}(u) \tag{10}$$

An optimal routing and rate allocation strategy set $\Pi \equiv \{(relay_1(u), rate_1(u)) \cdots (relay_h(u), rate_h(u))\}$ between routing pair of source node src and the destination dst with maximal delay bound T_max should be subject to the following constraints.

- **Schedule constraints:** The corresponding rate on the path route should satisfy the interference constraint in Formulas (4) and (5).
- **Buffer constraints:** The relay node on the path should satisfy the buffer requirement in Formula (6).
- **Delay constraints:** The relay node on the path should satisfy the delay requirement in Formula (8).

4 DRSS design

4.1 Network control for green opportunistic communications

The urban vehicular network traffic changes are related with time, zone, vehicle types and some other factors. Vehicles follow a kind of certain mobility pattern that can be a combination of the road map, the traffic light, the speed limit and traffic distribution. We try to learn from these traffic patterns and make it to assist data delivery in vehicular DTNs. We design a directional routing named DRSS that can help to relay, carry and forward the packets by selecting the optimal routes within the “preferred” areas that enable good performance toward the destination. An intelligent network control based on reinforcement learning is used to predict the knowledge of the network environment. The probabilistic routing strategies are updated according to reward/cost from the current network environment. We know that directional routing problem in vehicular DTNs complicates because of mobility. Reinforcement learning based network control framework can help to learn and predict the network traffic pattern according to history experiences. The probabilistic routing

strategies converge toward an optimal route from the reinforce learning process.

Figure 2 shows the network control framework by using reinforcement learning. To deal with the varied opportunistic connectivity in the network environment, the data communications should be based on the prediction of future contacts by taking advantages history of network topology and traffic distribution. During the intelligent learning procedure, the vehicle adapts the strategies according to its reward or cost with different actions without known modeling of network environment. With reinforcement learning, the vehicle can update its strategy by predict the network environment and converge to an optimal strategy.

According to the above framework, we adopt Q-learning approach to circumvent the routing and scheduling problem we defined in Sect. 3.5. Q-learning is used for learning and decision under dynamic and unknown environment model. The DTN network system has random connection events because of mobility. The Q-learning based routing scheme observe the transmission activity of its nearby nodes and gather the information of network traffic and topology as well as route resource utilization, which enables the node to build the knowledge of nearby routing environment. Then, the knowledge is used by the vehicle node to decide which relay direction and schedule to choose so that the desired performance can be achieved. Provided that the learned information is efficient, it guides routing and schedule strategies to make efficient use of network resources by routing packets toward the optimal direction.

The objective of DRSS is to learn from the environment states (dynamic connection events) to decide the actions so as to maximize the reward (i.e., minimize the cost function) [32]. The DTN network control system is formulated by a tuple $\langle S, A, \check{R}, \check{T} \rangle$, where S is the discrete state space. A is the discrete action space that is dependent on strategies taken. $\check{R}: S \times A \rightarrow \mathbf{R}$ is the cost function, which implies the quality of a state-action combination of the network system. $\check{T}: S \times A \rightarrow \Delta S$ is the state transition function, where ΔS is the probability distribution over state space S .

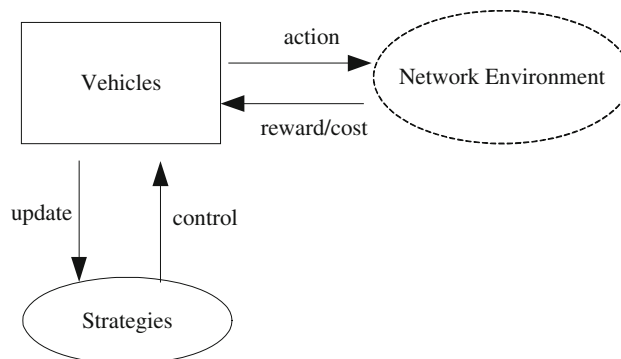


Fig. 2 The intelligent network control framework

In DRSS, the actions include sets of selected relay and rate strategies denoted as: $\pi(\text{relay}(u), \text{rate}(u))$. In which, $\text{relay}(u)$ is the optimal relay and $\text{rate}(u)$ is the assigned transmission slots for the outgoing link according to the chosen relay. Once an action (i.e., a strategy) is taken, the network system produces a new performance signal of cost according to it. With the update cost \check{R} , DRSS scheme evaluates the effectiveness of the action and then utilizes it for the further decisions. This is achieved by Q-learning procedure with update cost of actions.

The Q-learning procedure is achieved by updating the Q-value. The Q-learning approach converges to an optimal strategy as long as the state-action pairs are continually updated. When the transmission slots assigned for the outgoing link connected to the relay node start at time slot τ and ends at time slot $\tau + \lambda$. In which, λ is the number of time slots assigned for transmission schedule. The Q-value for state-strategy pair is updated as shown in Formula (11). Q' is the updated Q-value of taking action a in state s and then following the optimal policy thereafter. For current state s is with time τ , and the next transmission state is s' with time $\tau + \lambda$. For each step, the selected optimal strategy is to minimize the expected cost. This is achieved through an iterative search method, i.e., to find a minimal $Q(s', a)$ with a optimal strategy. In this formula, α is the learning rate among the range of (0,1). β is the discount factor among the range of (0,1) too. We use a constant learning factor, and the learning procedure can track the dynamic network situations.

$$Q'(s, a) = (1 - \alpha)Q(s, a) + \alpha(\check{R}_\tau + \beta \min_a Q(s', a)) \quad (11)$$

4.2 Cost function

The DRSS scheme aims to provide green energy-efficient routing and rate allocation. The cost function is achieved by the average amount of energy consumption per bit on the current route path from the 1st to h th hops, which embodies the tradeoff among energy efficiency, connection duration and communication efficiency. The Formula (12) shows the cost function. T_c is the slots of opportunistic contacting time in a frame, which includes transmission and receiving time slots for outgoing and incoming communications on current node u .

$$\begin{aligned} \check{R}(\text{path}) &= \sum_1^h \check{R}(u) = \sum_1^h \frac{E}{\text{rate}(u) * T_c} = \sum_1^h \frac{E_t + E_r + E_l}{\text{rate}(u) * T_c} \\ &= \sum_1^h \frac{(e_e + e_a R^n) \lambda_t + (e_e + e_p) \lambda_r + e_l (1 - \lambda_t - \lambda_r)}{(\lambda_t + \lambda_r) C * (\lambda_t + \lambda_r)} \end{aligned} \quad (12)$$

Considering the flow constraints and buffer constraints, we formally express the cost function as shown in Formula (13).

$$\check{R}(u) = \begin{cases} \frac{(e_e + e_a R^n) \lambda_t + (e_e + e_p) \lambda_r + e_l (1 - \lambda_t - \lambda_r)}{(\lambda_t + \lambda_r) C * (\lambda_t + \lambda_r)} & \text{if satisfies (4)–(8)} \\ \infty & \text{otherwise} \end{cases} \quad (13)$$

For one-commodity routing situation, if node $u \notin \{\text{src}, \text{dst}\}$ on the route path, then the traffic data bits sent out equal to the received data bits, i.e., $\lambda_t = \lambda_r$. The received traffic rate depends on the traffic rate sent out of the last hop on the route. So, the DRSS scheme is mainly responsible for selecting relay node and allocating transmission slots. For on-line routing, the available slots of current node u can be calculated as the total slots in a frame minus the slots used for communications and slots that may collide with it (among 2-hop communication range). This information will be known by periodic neighbor messages and roadside infrastructure broadcast.

4.3 Nash Q-learning approach

For on-line multi-commodity scenarios, DRSS utilizes extended Q-learning approach for multi-agent decision making. In this case, each agent is selfish for routing desire. The network routing and allocation problem is modeled as a stochastic non-cooperative game. Let $\langle N, \Pi, \check{R} \rangle$ denote the stochastic non-cooperative routing and scheduling game, where N is the number of routing commodities in the network. Π is the strategy set, and \check{R} is the cost set. The objective of multi-commodity routing is to minimize the overall cost. Different agents have their own objectives which may contend among one another, so the overall cost is dependent on the strategy selection of each agent. The strategy of each agent is to select relays and transmission rates (i.e., transmitting slots) on each route path hop. The overall cost achieves minimized cost during the learning process. According to this, the multi-agent Q-learning approach is utilized to find out the optimized DRSS strategy and achieve the minimized cost for all routes. Let $P(\pi_i)$ be the probability for i th routing commodity to select a specific strategy π at state s (i.e., the state with time τ), which achieves the game equilibrium. Π_i is the set of all possible strategies. The overall objective in the stochastic non-cooperative routing and scheduling game can be express as shown in Formula (14):

$$\min_{\pi_i \in \Pi_i} E[\check{R}_i(S)] \quad \text{for all } i \quad (14)$$

where

$$E[\check{R}_i(\pi)] = \sum_i P(\pi_i) \check{R}_i(\pi) \quad (15)$$

In the stochastic non-cooperative game based network system, each routing commodity finds a strategy with Nash equilibrium to achieve objective in Formula (14).

Let $NashQ_i$ be agent i 's cost for state s when all agents follow specified Nash equilibrium strategies. According to [31], the Nash equilibrium $NashQ_i$ is the agent i 's payoff in state s for the selected equilibrium, which is with equilibrium strategy $(\pi_1^* \cdots \pi_m^*)$ for the stage game $(Q_i(s) \cdots Q_m(s))$ computed from Formula (16). In the formula, m is the number for players in the game, i.e., the number of routing commodities in DRSS. In order to calculate the Nash equilibrium, each agent i needs to know the other agents' Q -values. Then, each agent observes the other agents' immediate cost and action. Then the agent i can update its Q -value according to other agents' Q -values as shown in Formula (17). In each time step at state s (with time τ), a player observes the cost \tilde{R}_i of its current state s and takes action a . An immediate cost for the next state s' (with time $\tau + \lambda$) are learned and predicted in this way.

$$NashQ_i(s, a) = \pi_1^*(s) \cdots \pi_m^*(s) \cdot Q_i(s, a) \tag{16}$$

$$Q'_i(s, a_1, \dots, a_m) = (1 - \alpha)Q_i(s, a_1, \dots, a_m) + \alpha(\overset{\vee}{R}_i + \beta NashQ_i(s', a_1, \dots, a_m)) \tag{17}$$

To minimize the network cost, the multi-agent Q-learning approach has to explore all possible strategies randomly and greedily, and then chooses the “good” strategy. The strategy exploration probability is updated as shown in Formula (18). The learning policy satisfies the GLIE (greedy in the limit with infinite exploration) property and also based on routing indicator priority. The relay is based on the priority of routing indicator I (defined in Sect. 3.3), and the bigger indicator value means higher priority. Therefore, for all the possible strategies, the probability of choosing strategy π for the i th commodity in state s' is decided by the probability of choosing π in state s , the Nash equilibrium strategy of state s , and routing indicator.

$$P(\pi_i, s') = \begin{cases} P(\pi_i, s) + I(1 - P(\pi_i, s)), & \text{if } \pi_i = \arg NashQ_i(s, a) \\ \max(I \cdot P(\pi_i, s), 0) & \text{if } \pi_i \neq \arg_{\pi} NashQ_i(s, a) \end{cases} \tag{18}$$

The Nash Q-learning based centralized algorithm is described in Algorithm 1. The algorithm is a half-distributed algorithm that needs global network information to select equilibrium point. The routing and scheduling process is executed distributed hop by hop, while the node decides the next hop and transmission slots according to the global game information. Line 1–6 is the algorithm initialization. In line 4, $|\Pi|$ is the number of possible

strategies, which is bounded by the multiplication of maximal degree of the network graph Δ and frame slot number L , i.e., $|\Pi| = \Delta * L$. Line 7–27 is the Nash Q-learning procedure. Line 13–19 is the routing mode selection, i.e., to use the “probabilistic forwarding” or “directional replication”. In line 20, $\tilde{R}_1 \cdots \tilde{R}_m$ denote the cost for all players, and $\pi_1 \cdots \pi_m$ denote the strategy taken by the other players except a_i . In line 21, slot time τ is updated as $(\tau + \lambda) \bmod L$ in the next state, where L is the frame length. Line 23 shows the Q -value update for each user in the next state according to Formula (17).

As explained in [29], the time complexity and space requirement of this learning algorithm is high when agent number is big. For 2-player Nash Q-learning, it has exponential worst-case time complexity. The space complexity is also exponential in the number of users. In the network, the game of routing resources occurs among the routing commodities within the interference range during the contacting time. The transmissions diverged from interference range will not affect one another. So, the routing game process of the network system can be achieved by the local game with local routing commodities, when there are joint routes within the interference range from one another during the contacting time.

An improved algorithm based on local games is illustrated in Algorithm 2. In Algorithm 2, we only consider the local game that happens among the interference range to reduce the complexity. So, each node makes distributed routing and scheduling decisions according to local game information. Of course, it still needs some global information for network initialization process.

In line 21, j is the routing commodity with routes that are within the interference range of i for current contacting state by satisfies: $\|route(i) - route(j)\| \leq D$. D is the interference range. The distance of the routes are defined as the minimal distances of the relay nodes from the two routes respectively. So, it only need to observe the cost and

action of contending commodity with current commodity i , but not all the other commodity in the network. For general DTN networks and their applications, the networks are usually sparsely or loosely connected. The on-line routing commodity number is often small within the interference range, so the performance of local game based DRSS is acceptable. From the above algorithms, DRSS selects the contact strategies toward the energy-efficient and

Algorithm 1 Basic algorithm with Nash Q-learning

```

1  for  $i=1 \dots m$  //  $m$  agents with  $src_i$  and  $dst_i$ 
2    Let  $\tau=0$ , get the initial state  $s=s(\tau)$ 
3    for all  $s \in S$  and  $a \in A$ 
4       $Q_i(s, a) = 0$ , for all  $\pi^k \in \Pi, P(\pi_i^k) = 1 / |\Pi|$ 
5    endfor
6  endfor
7  while (network execution condition is TRUE)
8     $u_i = src_i$ 
9    for current node  $u_i$  //  $u_i$  is the current hop for  $i^{\text{th}}$  agent
10   if ( $u_i == dst_i$ )    break;
11   endif
12   Choose action  $a_i$  according to (18)
13   if  $relay\_pattern == \text{"straightway"}$  then
14     forward the packet; //forward one copy
15   else if  $relay\_pattern == \text{"intersec-2-des"}$  then
16     replicate the packet; //replicate the packet
17   else if  $relay\_pattern == \text{"intersec-against-des"}$  then
18     break; //carry the packet until next contact
19   endif
20   Take  $\pi_i$  for  $s(\tau)$  and observe  $\check{R}_1 \dots \check{R}_m, a_1 \dots a_m$ 
21    $\tau = (\tau + \lambda) \bmod L$  and the next state  $s' = s(\tau)$  //change to the
     next state with time update
22   for all  $j \neq i, j = 1 \dots m$ 
23     Update  $Q_j$  according to (17)
24   endfor
25    $s = s', u_i = \pi(relay(u_i))$  //update state and the next hop
26   endfor
27 endwhile

```

interference-aware direction toward the destination within the timeliness by optimal relay and feasible schedule. According to Q-learning cost, DRSS converges to select the relay node that is moving with velocity equal to or higher than the expected average velocity towards the destination direction with good connectivity and traffic

situations. Otherwise, the node carries the packets until it finds a better node that can help to make data delivery.

5 Simulations

We make simulations by using ns2 simulator and evaluate the performance. We simulate a vehicular ad hoc delay-tolerant network with 50 vehicles travelling among the area of 2,500 m \times 2,500 m. The vehicles travel with the speed from 5 m/s to maximal 20 m/s around the test area. A fixed static node is placed at the center of the area, which is used to simulate a roadside access point in real vehicular networks. By default, all the vehicle nodes move in the network according to the Random Waypoint mobility model. In the simulations, the vehicles connect with each other by a half-duplex wireless network interface card. The communication range is 150 m, while the interference range is set to 300 m. At the beginning of the simulation, the routing request with CBR rate is periodically generated from a source node that is randomly selected. The load is 40Bps generated on the source node. By default, each node allocates a limited buffer with maximal 30 packets. One slot is enough time for transmit or receive a packet. The simulated roadside access point is chosen as the routing destination. We initialize the parameter values: $\alpha = 0.1$, $\beta = 0.9$ and $\gamma = 0.5$. The value of energy consumption related parameter e_e , e_a , e_p , and e_l is chosen based on [33].

In this part, we simulate and evaluate the convergence of Q-learning based DRSS, learning ability when traffic load changes and energy efficiency, and then the performance of data delivery delay. We make simulations when the routing commodity number is 1, 2, and 4 respectively. In the 4-commodity situations, we evaluate the performance of the local game based algorithm.

5.1 Convergence

Directional routing and scheduling scheme utilizes Q-learning approach, so convergence is one of the main concerns for performance evaluations. We study the convergence speed of the cost function as simulation time increases. The result is shown in Fig. 3. It shows the average path cost from 20 to 1,000 s during the simulations. We observe that the average path cost begins to converge around 800 s for all the situations. When comparing multi-agent results with 1-agent situation, results of multi-agent seem to incur a little more cost. This is because Nash equilibrium strategy is chosen to provide optimal routing cost for all the agents in the network, which may not be the optimal strategy for each single agent.

Algorithm 2 Improved algorithm with local game

```

1  for  $i=1\dots m$  //  $m$  agents with  $src_i$  and  $dst_i$ 
2    let  $\tau=0$ , get the initial state  $s=s(\tau)$ 
3    for all  $s\in S$  and  $a\in A$ 
4       $Q_i(s, a) = 0$ , for all  $\pi^k \in \Pi, P(\pi_i^k) = 1/|\Pi|$ 
5    endfor
6  endfor
7  while (network execution condition is TRUE)
8     $u_i = src_i$ 
9    for current node  $u_i$  //  $u_i$  is the current hop for  $i^{\text{th}}$  agent
10   if ( $u_i == dst_i$ ) break;
11   endif
12   choose action  $a_i$  according to (18)
13   if  $relay\_pattern == \text{"straightway"}$  then
14     forward the packet; //forward just one copy
15   else if  $relay\_pattern == \text{"intersec-2-des"}$  then
16     replicate the packet; //replicate the packet
17   else if  $relay\_pattern == \text{"intersec-against-des"}$  then
18     break; //carry the packet until next contact
19   endif
20   take  $\pi_i$  for  $s(\tau)$  and get  $\check{R}_i$ 
21   observe  $\check{R}_j, a_j (j \neq i, j=1\dots m)$ , if  $\|route(i)-route(j)\| \leq D$ 
22   update  $\tau = \tau + \lambda$ , the next state  $s'=s(\tau)$ 
23   for all  $j$  //  $j$  is the local contender
24     Update  $Q_j$  according to (17)
25   endfor
26    $s=s', u_i = \pi(relay(u_i))$  //update state and next hop
27   endfor
28 endwhile

```

5.2 Learning ability

The advantage of Q-learning approach based DRSS is the ability to predict the unknown network environment and then adapt the strategies toward it by learning from it. In

practical road scenarios for vehicular DTNs, the network topology and traffic varies with time. We then simulate the varied network topology and traffic by changing the number of vehicles in a specific area.

In the first scenario: we add 20 more default vehicles into the network area to increase the connections at 200 s after the simulation begins, and then we remove them at 800 s. The corresponding results are shown in Fig. 4. We observe that DRSS can track the change and adapt the strategies toward the network environment. The average path cost of routing situations with 20 more vehicles added into the network is lower than the original situations. It implies that the added vehicles help to increase the contacting opportunity in the network without extra congestions because of effective schedule control. At 800 s of the simulations, we remove those newly added vehicles. This incurs small fluctuations of the average path cost, because each agent learns from the change of network scenario and finds new strategy toward it.

In the second scenario: we double the source traffic rate from 200 s of the simulations, and then change back to the original traffic rate at 800 s. The corresponding results are shown in Fig. 5. We observe that the average path cost increases a little from 200 to 800 s, when compared with the previous results. This is because that the increased load incurs more possible congestions. According to DRSS, the routing and rate allocation strategies will be adapted to optimize the cost. More idle energy consumptions are incurred in a long schedule for the increased traffic. The results show there are many result fluctuations after 800 s, because DRSS keeps learning and selecting new strategies according to the varied network environment.

5.3 Delay-bounded delivery

Another concern of DRSS performance is the data delivery ratio within the given delay bound. Figure 6 shows the average real-time delivery ratio as delay bound increases from 200 to 4,000 ms, which is based on the results of 800 s' simulations. We observe that the data delivery ratio increases as we relax the delay bound. The 1-agent achieves better delivery when compared with 2-agent and 4-agent situations. More routing agents bring more possible congestions in the networks and the need more schedule time for transmission according to DRSS rate allocation.

5.4 Comparisons

We compare our DRSS with epidemic [11] and modified Dijkstra algorithm [18]. Figure 7 shows the average energy consumption per bit during the simulations. From the results, we can see that DRSS has much better energy efficiency when compared with the other two algorithms.

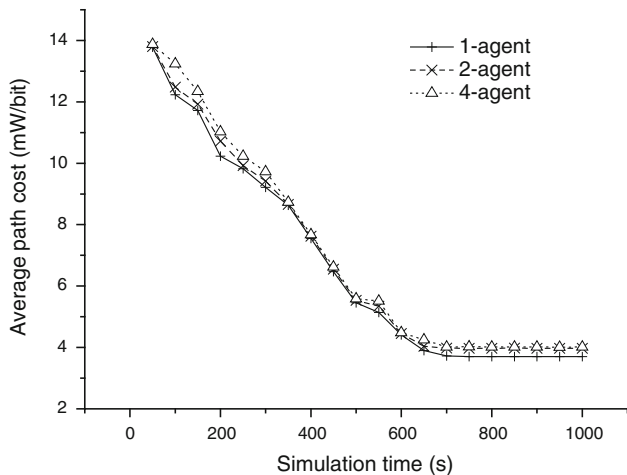


Fig. 3 Convergence of DRSS

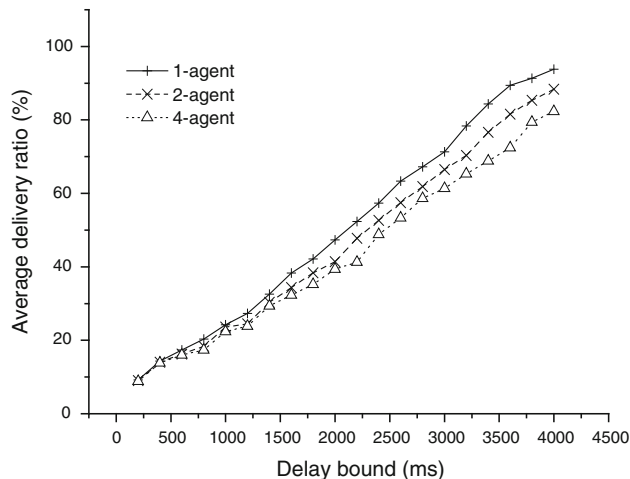


Fig. 6 Data delivery ratio as delay bound increases

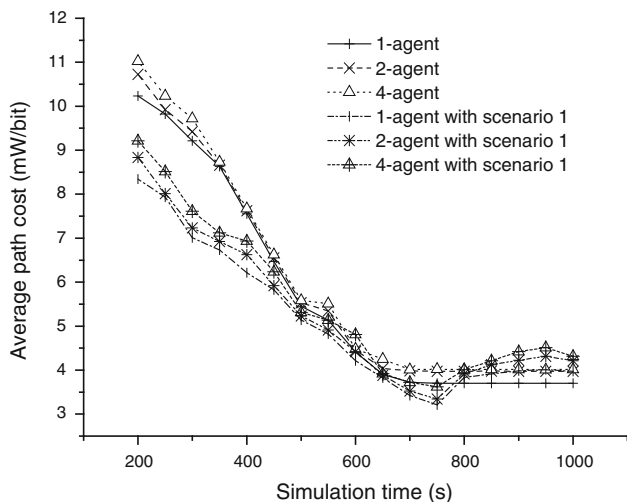


Fig. 4 Learning ability of DRSS with scenario 1

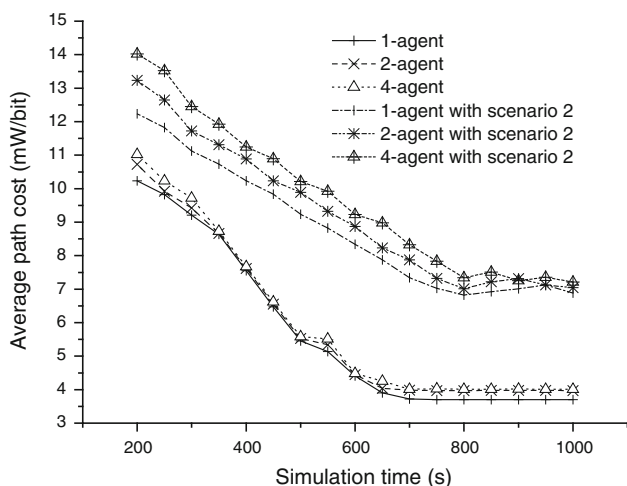


Fig. 5 Learning ability of DRSS with scenario 2

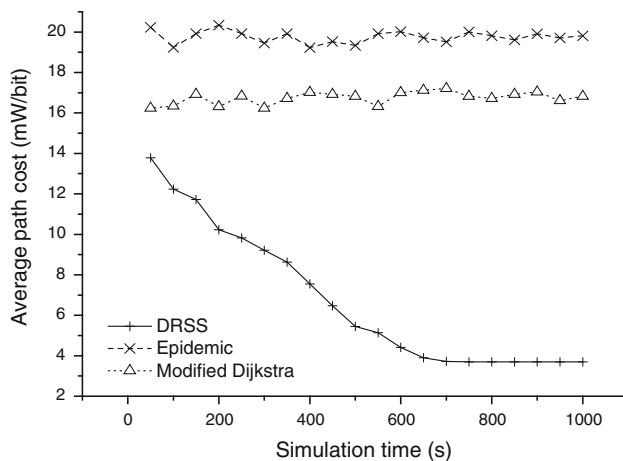


Fig. 7 Average energy consumption per bit

Epidemic algorithm uses the replication method to deliver data packets, so the more energy is consumed. And this is even worse when simulation goes on, because much replicated routing occurs during the packet delivery. The modified Dijkstra algorithm uses the forward method to minimize the average delivery delay, but it does not consider the energy consumption of packets and cannot adapt toward the varied network environment. Actually, sometimes it leads to the wrong direction that is close to the destination without connection and may consume more energy.

6 Conclusion

We make research on an energy-efficient routing and scheduling scheme in vehicle delay tolerant networks in this paper. To predict the unknown network environment such as traffic pattern in varied network situations, we

utilize Nash Q-learning approach to try to find the optimized routing strategies optimizing the overall network efficiency. We present a novel DRSS to provide energy efficient and delay-bounded data delivery by combining forwarding and replication method. We make simulations on vehicular DTN networks and make performance evaluations. The results show DRSS can effectively improve the energy efficiency and data delivery ratio by learning and predicting the network environment. Our further work includes simulations on road networks with different network deployment and more realistic mobility models.

Acknowledgments This work was supported by Chinese National Science Foundation Project (no. 61103218), Hubei Province National Science Foundation Project (no. 2011CDB446), Chinese National Program on Key Basic Research Project (no. 2011CB707106) and Wuhan University Automatic Research Project (no. 111093).

References

- Blum, J., Eskandarian, A., & Hoffman, L. (2004). Challenges of intervehicle ad hoc networks. *IEEE Transaction on Intelligent Transportation Systems*, 5(2004), 347–351.
- Grossglauser, M., & Tse, D. (2002). Mobility increases the capacity of ad hoc wireless networks. *IEEE/ACM Transaction on Networking*, 10(4), 477–486.
- Fall, K. (2003). A delay-tolerant network architecture for challenged internets. *ACM SIGCOMM* (pp. 27–34).
- Fletcher, R., & Hasson, A. (2004). Daknet: Rethinking connectivity in developing nations. *IEEE Computer*, 37(1), 78–83.
- Hull, B., Bychkovsky, V., Zhang, Y., Chen, K., Goraczko, M., Miu, A., et al. (2006). CarTel: A distributed mobile sensor computing platform. *ACM SenSys* (pp. 125–138).
- DTN Research Group (DTNRG). <http://www.dtnrg.org/>.
- Spyropoulos, T., Rais, R. N. B., Turetli, T., Obraczka, K., & Vasilakos, A. (2010). Routing for disruption tolerant networks: taxonomy and design. *Wireless Networks*, 16(8), 2349–2370.
- Hodges, R., & White, W. (2008). *Go green in ICT*. Technical report, GreenTech News.
- Lee, K. C., Lee, U., & Gerla, M. (2009). Survey of routing protocols in vehicular ad hoc networks. In *Advances in vehicular ad-hoc networks: Developments and challenges*, chapter 8, IGI Global (pp. 149–170). doi:10.4018/978-1-61520-913-2.ch008.
- Zhao, J., & Cao, G. (2006). Vehicle-assisted data delivery in vehicular ad hoc networks. In *25th IEEE InfoCom* (pp. 1–12).
- Mitchener, W., & Vahat, A. (2000). *Epidemic routing for partially connected ad hoc networks*. Technical report CS-2000-06, Duke University.
- Spyropoulos, T., Psounis, K., & Raghavendra, C. S. (2005). Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In *ACM WDTN* (pp. 252–259).
- Burns, B., Brock, O., & Levine, B. N. (2005). MV routing and capacity building in disruption tolerant networks. In *IEEE InfoCom* (pp. 398–408).
- Lindgren, A., Doria, A., & Schelen, O. (2004). Probabilistic routing in intermittently connected networks. *Lecture Notes in Computer Science*, 3126, 239–254.
- Small, T., & Haas, Z. (2005). Resource and performance trade-offs in delay-tolerant wireless networks. In *ACM SIGCOMM workshop on delay-tolerant networking (WDTN)* (pp. 260–267).
- Burgess, J., Gallagher, B., Jensen, D., & Levine, B. N. (2006). MaxProp: Routing for vehicle-based disruption-tolerant networks. In *IEEE InfoCom* (pp. 1–11).
- Balasubramani, B., Levine, N., & Venkataramani, A. (2007). DTN routing as a resource allocation problem. In *ACM SIGCOMM* (pp. 373–385).
- Jain, S., Fall, K., & Patra, R. (2004). Routing in a delay tolerant network. In *ACM SIGCOMM* (pp. 145–157).
- Hay, D., & Giaccone, P. (2009). Optimal routing and scheduling for deterministic delay tolerant networks. In *IEEE WONS* (pp. 27–34).
- Bulut, E., Geyik, S. C., & Szymanski, B. K. (2010). Conditional shortest path routing in delay tolerant networks. In *IEEE international symposium on a world of wireless mobile and multimedia networks (WoWMoM)* (pp. 1–6).
- Ahmed, S., & Kanhere, S. S. (2010). A Bayesian routing framework for delay tolerant networks. In *Wireless communications and networking conference (WCNC)* (pp. 1–6).
- Huang, C.-J., Shen, H.-Y., Liao, J.-J., Hu, K.-W., Yang, D.-X., Chen, C.-H., et al. (2009). A fuzzy logic-based routing for delay-tolerant heterogeneous networks. In *IEEE international conference on granular computing (GRC)* (pp. 254–259).
- Dvir, A., & Vasilakos, A. V. (2010). Backpressure-based routing protocol for DTNs. In *ACM SIGCOMM* (pp. 405–406).
- Pereira, P., Casaca, A., Rodrigues, J., Soares, V., Triay, J., & Cervelló-Pastor, C. (2011). From delay-tolerant networks to vehicular delay-tolerant networks. *IEEE Communications Surveys & Tutorials*, 99, 1–17.
- Leontiadis, I., & Mascolo, C. (2007). GeOpps: Geographical opportunistic routing for vehicular networks. In *IEEE international symposium on a world of wireless, mobile and multimedia networks (WoWMoM)* (pp. 1–6).
- Ros, F., Ruiz, P., & Stojmenovic, I. (2012). Acknowledgment-based broadcast protocol for reliable and efficient data dissemination in vehicular ad hoc networks. *IEEE Transactions on Mobile Computing*, 11(1), 33–46.
- Zhou, L., Wang, X., Tu, W., Muntean, G., & Geller, B. (2010). Distributed scheduling scheme for video streaming over multi-channel multi-radio multi-hop wireless networks. *IEEE Journal on Selected Areas in Communications*, 28(3), 409–419.
- Sanctis, M. D., Cianca, E., & Joshi, V. (2011). Energy efficient wireless networks towards green communications. *Wireless Personal Communications*, 59(3), 537–552.
- Wang, X., Vasilakos, A. V., Chen, M., Liu, Y., & Kwon, T. T. (2011). A survey of green mobile networks: Opportunities and challenges. *Mobile Networks and Applications*, 1, 1–17.
- Mangrulkar, R. S., & Atique, M. (2010). Routing protocol for delay tolerant networks: A survey and comparison. In *IEEE international conference on communication control and computing technologies (ICCCCT)* (pp. 210–215).
- Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press.
- Hu, J., & Wellman, M. P. (2003). Nash Q-learning for general-sum stochastic games. *Journal of Machine Learning Research*, 4, 1039–1069.
- Feng, W., & Elmihani, J. M. H. (2009). Green ICT: Energy efficiency in a motorway model. In *3rd international conference on next generation mobile applications, services, and technologies* (pp. 389–394).

Author Biographies



Yuanyuan Zeng received her B.Sc. and Ph.D. degree in Computer Science from Wuhan University, China in 2002 and 2007, respectively. She was a research student and then conducted as research assistant in Department of Computer Science, City University of Hong Kong from 2005 to 2007. She was a lecturer in Shanghai University of Finance and Economy in 2007. She then became a postdoctoral researcher in University College Cork, Cork, Ireland and Wuhan University,

Wuhan, China from 2008 till 2010, respectively. She is currently working in School of Electronic Information, Wuhan University. Her research interests involve in network embedded systems, wireless sensor networks, stochastic optimization, and wireless communication. She has been a program committee and reviewer of about 20 international conferences and journals.



Kai Xiang received his M.Sc. degree in Computer Science from Huazhong Normal University, China in 2006. He is currently a Ph.D. candidate in National Engineering Research Centre for Multimedia Software, Wuhan University, Hubei, China. His research interests involve in Wireless Sensor Networks, Multimedia Information Processing and Communication technologies, and Streaming Media technologies.



Deshi Li received his B.S. degree in Information Engineering from Chengdu Technology University, China in 1990. He then received the M.S. in Electronic Engineering from Wuhan University of Hydraulic and Electronic Engineering in 1993. He received the PhD degree in Computer Science from Wuhan University in 2001. From 2000 to 2005, He was an associate professor in Wuhan University. From December 2001 till August 2004, He

committed his post-doc research in Geophysics at Wuhan University.

From 2005 till 2007, He was a senior visiting scholar at University of California, Davis, USA, where he was involved in underwater wireless sensor networks. Dr. Li is currently a professor and associate dean in the School of Electronic Information, Wuhan University, China. His research is involved in wireless sensor networks, networked robots, wireless communication, etc. Dr. Li has been in charge of many Chinese NSF projects, national 973 and 863 projects and province-ranked projects. He received technology-developing award in Hubei Province, and achieved numerous patents in China. He is now a member of Chinese Association of Internet of Things.



Athanasios V. Vasilakos is currently Professor at the Department of Computer and Telecommunications Engineering, University of Western Macedonia, Greece, and Visiting Professor at the Graduate Programme of the Department of Electrical and Computer Engineering, National Technical University of Athens (NTUA). He is coauthor (with W. Pedrycz) of the books *Computational Intelligence in Telecommunications Networks* (CRC Press, USA, 2001), *Ambient*

Intelligence, Wireless Networking, Ubiquitous Computing (Artech House, USA, 2006), coauthor (with M. Parashar, S. Karnouskos, W. Pedrycz) *Autonomic Communications* (Springer), *Arts and Technologies* (MIT Press), coauthor (with M. Anastasopoulos) *Game Theory in Communication Systems* (IGI Inc., USA). He has published more than 150 articles in top international journals and conferences. He is the Editor-in-chief of the Inderscience Publishers journals: *International Journal of Adaptive and Autonomous Communications Systems* (IJAACS, <http://www.inderscience.com/ijaacs>), *International Journal of Arts and Technology* (IJART, <http://www.inderscience.com/ijart>). He was or he is at the editorial board of more than 20 international journals including: *IEEE Communications Magazine* (1999–2002 & 2008–), *IEEE Transactions on Systems, Man and Cybernetics* (SMC, Part B, 2007–), *IEEE Transactions on Wireless Communications* (invited), *ACM Transactions on Autonomous and Adaptive Systems* (invited). He is chairman of the Telecommunications Task Force of the Intelligent Systems Applications Technical Committee (ISATC) of the IEEE Computational Intelligence Society (CIS). Senior Deputy Secretary-General and fellow member of ISIBM (International Society of Intelligent Biological Medicine). He is member of the IEEE and ACM.