Ad Hoc Networks 7 (2009) 725–741

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

Bio-inspired multi-agent data harvesting in a proactive urban monitoring environment ☆

Uichin Lee [a], Eugenio Magistretti [b,1], Mario Gerla [a], Paolo Bellavista [b], Pietro Lió [c], Kang-Won Lee [d,*]

[a] Department of Computer Science, UCLA, Los Angeles, CA 90095
[b] Dipartimento di Elettronica, Informatica e Sistemistica, University of Bologna, Bologna, Italy 40136
[c] Computer Laboratory, University of Cambridge, England
[d] IBM T. J. Watson Research Center, Wireless Networking Group, 19 Skyline Drive, Hawthorne, NY 10532, United States

A R T I C L E I N F O

Article history:
Received 11 June 2007
Received in revised form 24 January 2008
Accepted 28 March 2008
Available online 16 May 2008

A B S T R A C T

Vehicular sensor networks (VSNs) enable brand new and promising sensing applications, such as traffic reporting, relief to environmental monitoring, and distributed surveillance. In our past work, we have designed and implemented *MobEyes*, a middleware solution to support VSN-based urban monitoring, where agent vehicles (e.g., police cars) move around and harvest meta-data about sensed information from regular VSN-enabled vehicles. In urban sensing operations, multiple agents typically collaborate in harvesting and searching for key meta-data in parallel. Thus, it is critical to effectively coordinate the harvesting operations of the agents in a decentralized and lightweight way. The paper presents a bio-inspired meta-data harvesting algorithm, called *datataxis*, whose primary goal is to effectively cover large urban areas *datataxis* alternate foraging behaviors inspired by *Escherichia coli* chemotaxis and by Lévy flights to favor agent movements towards "information patches" where the concentration of meta-data is high. The proposed scheme avoids harvesting duplication by preventing superfluous concentration of agents in the same region at the same time using stigmergy. We have validated datataxis via extensive simulations that demonstrate how the proposed bio-inspired behavior of harvesting agents effectively coordinates their movements, thus outperforming other decentralized strategies. Our solution was shown to be robust and to work well under a wide range of operation parameters, thus making it easily and rapidly deployable for different urban sensing operations.

1. Introduction

Vehicular sensor networks (VSNs) are becoming increasingly popular and relevant to the industry due to recent advances in inter-vehicular communication technologies and decreasing cost of communication devices. Differently from traditional wireless sensor nodes, vehicles are not typically affected by energy constraints and can easily be equipped with powerful processing units, wireless communication devices, GPS, and sensing devices such as chemical detectors, still/video cameras, vibration and acoustic sensors. Thus, they enable brand new and promising sensing applications, such as traffic reporting, relief to environmental monitoring, and distributed surveillance.

☆ Research was sponsored in part by the US Army Research Laboratory and the UK Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the US Army Research Laboratory, the US Government, the UK Ministry of Defence or the UK Government. The US and UK Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

* Corresponding author. Tel.: +1 914 784 7228; fax: +1 914 784 6205.
E-mail addresses: uclee@cs.ucla.edu (U. Lee), gerla@cs.ucla.edu (M. Gerla), paolo.bellavista@unibo.it (P. Bellavista), kangwon@us.ibm.com (K.-W. Lee).
[1] Electrical and Computer Engineering Department, Rice University, 6100 Main St. Houston, Texas 77005-1827.

1570-8705/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.adhoc.2008.03.009

In particular, there is an increasing interest in proactive urban sensing applications where vehicles continuously monitor events from urban streets, maintain sensed data in their local storage, process them (e.g., extracting license plate numbers from still images), and route messages to vehicles in their vicinity to achieve a common goal (e.g., to allow police agents to pursue the movements of runaway cars). In general, VSNs could be an excellent complement to the fixed deployment of static cameras/sensors. The distributed and opportunistic cooperation among sensor-equipped vehicles has the additional benefit of making it harder for potential attackers to thwart the surveillance. Of course, this added level of surveillance capability may raise privacy concerns, but in general people are willing to accept it when only recognized authorities can collect and process these data for forensic purposes, for counteracting terrorism, or simply for the common good (e.g., traffic jam reporting).

The above examples show that VSN-based urban monitoring often requires the collection, storage, and retrieval of massive amounts of sensed data. This is a major departure from conventional sensor networks, where data is dispatched to "sinks" for further processing. Obviously, it is impossible to deliver all the streaming data collected by video sensors on regular cars to a police authority sink because of sheer volume; also a priori input filtering is usually impossible because nobody can anticipate which data will be of use for future investigations. Thus, there is the hard technical challenge to find a completely decentralized solution for VSN applications, with low overhead, good scalability, and tolerance to disruption caused by mobility/attacks. To that purpose, we have designed and implemented the MobEyes middleware[1] MobEyes agents (e.g., police cars) move around and harvest meta-data (with features about sensed data and context information such as timestamp and location) when they are in direct communication range with regular vehicles; regular cars collect meta-data from other opportunistically encountered vehicles. In typical urban sensing operations, given the large scale of the targeted regions, multiple agents should collaborate in harvesting distributed meta-data and searching for key information in parallel. Thus, it is critical to design an effective mechanism to coordinate the movement of agents so that they can efficiently collect all meta-data of interest in a totally decentralized and lightweight way. Multi-agent harvesting is a challenging problem due to the very dynamic nature of the target environment (e.g., continuous meta-data creation/movement) and the scale of operation, without any a priori knowledge of the needed meta-data location.

Based on the observation that social animals (ranging from bacteria to vertebrae) efficiently solve a partially similar challenge – namely the *foraging* problem to find good food sources, this paper presents a novel multi-agent coordination algorithm for MobEyes harvesting agents that takes inspiration from biological systems. In particular, we have considered multiple biological phenomena: (a) Foraging behavior of *Escherichia coli* bacteria that operate in distinct modes of locomotion based on the level of nutrient concentration; (b) Lévy walk behavior of many biological organisms and groups, e.g., albatrosses and fishing boats, to improve food search over large-scale regions; and (c) Stigmergy found in ants and other social insects that use various types of pheromones to signal nest mates with potential conflicts, e.g., a sort of "no entry" sign. By delving into finer details, we propose a novel harvesting strategy, called *datataxis* (á la chemotaxis of *E. coli* bacteria), that guides the agents to stay and acquire meta-data on "information patches," the regions where newly created and not-harvested meta-data are concentrated (based on a simple metric for meta-data density estimation per road segment). Secondly, MobEyes harvesting agents adapt their behavior by following a 3-state transition diagram that sometimes forces them to change their area of exploration by exploiting Lévy walk-inspired movement patterns, which are considered suitable for the large scale of the typically targeted regions. Third, to avoid harvesting work duplication, MobEyes agents exploit stigmergy-inspired techniques for conflict resolution to prevent useless concentration of agents in the same region at the same time.

We have thoroughly validated the datataxis performance via an extensive simulation study taking into consideration different scenarios with different search areas, number of agents, and algorithm parameter settings. Using a Manhattan mobility model, we compare the harvesting efficiency of our datataxis foraging (DTF) with other decentralized harvesting strategies, such as random walk foraging (RWF), biased random walk foraging (BRWF), and an idealized preset pattern foraging (PPF) where agents have static knowledge on meta-data density distribution. From this study, we show that DTF effectively balances the movement of multiple agents and well distributes them notwithstanding the minimum agent coordination needed. The paper also shows that DTF always performs better than RWF and BRWF, and very close to PPF in all cases. Finally, we analyze the sensitivity of DTF performance with respect to two key parameters (namely, constrained walk duration and constrained walk radius), thus demonstrating the DTF robustness for a wide range of the operation parameter space.

The remainder of the paper is organized as follows: Section 2 presents a brief overview of MobEyes to allow the full understanding of the following agent coordination proposal. Section 3 defines the addressed coordination problem, while Section 4 describes the foraging behaviors in the nature that inspired our work. Then, we present our original datataxis solution for multi-agent coordination in Section 5, which is extensively evaluated in Section 6. Related work and conclusive remarks end the paper.

2. MobEyes overview

We first describe two application scenarios that are representative of the MobEyes applications. Based on these scenarios, we examine the key components of MobEyes, namely meta-data dissemination, meta-data harvesting, and data access.

[1] We provide a brief overview of MobEyes in Section 2; interested readers can refer to [39] for the detailed description of MobEyes architecture and protocols.

2.1. Urban sensing scenarios

In the first scenario, police agents use the background information collected by MobEyes-enabled vehicles to investigate a tip that criminals might spread poisonous chemicals in a particular section of the city (say, major city intersection). The MobEyes vehicles are equipped with cameras and chemical detection sensors. In this scenario, a search of the MobEyes distributed database will help locate and verify the attack and track the criminal vehicles leading to their capture. In the second scenario, we assume there has been a suicide bombing at the Embassy. The police agents investigate the MobEyes database to reconstruct the itinerary of the attack vehicle and detect possible colluding vehicles.

These scenarios are both representative MobEyes surveillance applications; yet, they are very different in terms of search requirements. In the first case, there is a great urgency to detect the location of the attack so that the authority can possibly prevent it. However, the cars will move in a regular traffic pattern. In the second case, the investigation is more of forensic nature, but the traffic pattern is very different because the cars near the explosion site may try to escape the scene as quickly as possible. However, in this case also the overall mobility will be constrained by the prevailing traffic conditions of the urban grid (traffic lights, congestion at major intersections, queues at freeway on ramps, etc). Experimental evidence from realistic urban simulations has shown that, as a result of the above traffic constraints, the vehicles carrying the meta-data related to a particular event (e.g., gas leak, explosion, attack trajectory, kidnapping, etc) tend to form clusters that are concentrated over a confined area. This observation is used in designing an effective search strategy.

2.2. MobEyes protocol

In MobEyes, vehicles continuously generate a huge amount of sensed data, store it locally, and periodically produce small size *meta-data chunks* obtained by processing the sensed data, e.g., license plate numbers or aggregated chemical readings. They include meta-data associated with the sensed data (e.g., vehicle position, timestamp, vehicle ID number and possible additional context such as simultaneous sensor alerts) and the features of interest extracted by local filters. A set of chunks are packed into a single packet for efficiency. This section briefly describes our diffusion protocol that is used by non-agent vehicles (i.e., regular nodes) to opportunistically spread meta-data by exploiting mobility. The diffused meta-data represent the first level distributed index that can be used to support targeted searches. It then describes the meta-data harvesting protocol used by agent vehicles which are police cars in our example. The police agents perform proactive harvesting in the background to build a second level distributed index of sensed data that resides on agent vehicles. The index is locally processed by police vehicles to detect possible anomalies and trigger alarms, and may also be interrogated electronically from the Central Office in response to time critical queries. Finally, we describe how the actual data can be retrieved. Note that additional details about the design and implementation of the MobEyes prototype are out of the scope of this paper. Interested readers are referred to [5].

2.2.1. Meta-data diffusion

A regular node periodically advertises newly generated meta-data to its neighbors in order to increase the opportunities for agents to harvest the meta-data. Clearly, excessive advertising will introduce too much overhead, whereas too few advertising will result in delayed data harvesting, as agents will need to contact more vehicles to complete the harvesting process. MobEyes tries to balance the trade-off between harvesting latency and advertisement overhead. In the meta-data advertisement, the packet header includes a packet type, generator ID, locally unique sequence number, packet generation timestamp, and generator's current position. Each packet is uniquely identified by the generator ID and its sequence number pair, and contains a set of meta-data locally generated during a fixed time interval.

Neighbor nodes receiving a packet store it in their local meta-data databases. Therefore, depending on the mobility and the encounters of regular nodes, packets are opportunistically diffused into the network of vehicles. We call this a *passive* diffusion. MobEyes can be configured to perform either single-hop passive diffusion (only the source advertises its packet to current single-hop neighbors) or k-hop passive diffusion (the packet travels up to k-hop as it is forwarded by j-hop neighbors with $j < k$). As detailed in [39], we find it is sufficient for MobEyes to use a lightweight k-hop passive diffusion strategy with very small k values to achieve a desired level of diffusion.

Fig. 1 depicts the operation of MobEyes with the case of two sensor nodes, $C1$ and $C2$, whose radio ranges are represented as dotted circles. In the figure, a black triangle with timestamp represents an encounter. According to the MobEyes meta-data diffusion protocol, $C1$ and $C2$ periodically advertise a new meta-data packet $S_{C1,1}$ and $S_{C2,1}$ respectively where the subscript denotes $\langle ID, Seq.\# \rangle$. At time $T - t_4$, $C2$ encounters $C1$, and thus they exchange

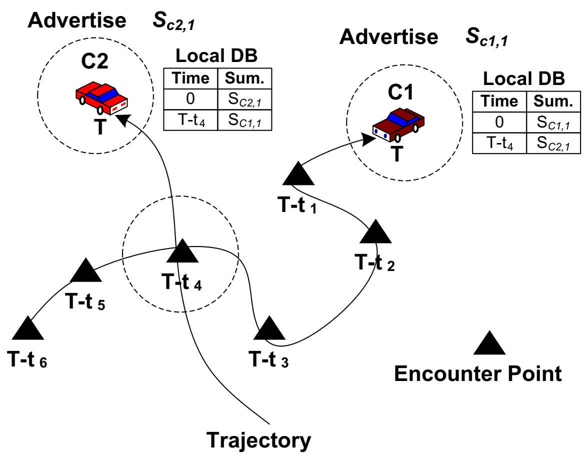


Fig. 1. MobEyes single-hop passive diffusion.

these advertisements. As a result, $C1$ carries $S_{C2,1}$ and $C2$ carries $S_{C1,1}$. We note that meta-data diffusion is time and location sensitive; i.e., it is spatial-temporal information diffusion. In fact, regular nodes keep track of the freshness of meta-data packets by using a sliding window with the maximum window size of a fixed expiration time. In addition, since a single advertisement packet may contain multiple meta-data records, it is possible to define packet sensing location as the average position of all meta-data in the packet. When a packet expires or the packet originator moves away more than a threshold distance from the packet sensing location, the packet is automatically disposed. The expiration time and the maximum distance are system parameters that should be configured depending on the requirements of an urban monitoring application.

2.2.2. Meta-data harvesting

In parallel with the meta-data diffusion, the MobEyes agents may collect meta-data from regular nodes by periodically querying the nearby nodes. The goal of meta-data harvesting is to collect all the meta-data generated in a specified area. For efficiency, a police agent should harvest only those meta-data packets that have not been collected already. In order to identify missing packets, a MobEyes agent node compares its list of meta-data packets with that of its neighbors, by exploiting a space-efficient data structure for membership checking. In our design, a MobEyes agent uses a Bloom filter to advertise its set of recently harvested meta-data packets. Since each meta-data has a unique node ID and sequence number pair, we use this as input for the hash functions of the Bloom filter. When an agent broadcasts a harvest request message it includes a Bloom filter for the meta-data it has already harvested. Upon receiving this, a regular node prepares a list of packets that are not already in the agent's packet list, and sends them to the agent using a random back-off to minimize collisions. In response, the agent sends back an acknowledgment with a piggybacked list of successfully received packets. Upon overhearing this, each neighbor updates their list of packets to send.

In the basic design of MobEyes, we have only considered a single agent scenario and a multi-agent scenario where the agents harvested data without coordination using random movement [39]. In general, however, we envision that there will be multiple collaborating agents that try to harvest important information concurrently. Thus MobEyes should support the operation of multiple agents. In effect, multiple agents create a distributed and partially replicated index of sensed data. The main goal of this paper is to strategically control the trajectory of agent nodes to efficiently harvest meta-data so that they can be collectively used for later data access. We present a detailed problem description for this paper in the next section.

2.2.3. Data access

In vehicular area networks (VANETs), geographic routing (or geo-routing) has been investigated extensively for its scalability. To handle intermittent connectivity due to heterogeneous vehicle distribution and time-of-day effects (e.g., during off-rush hours and in peripheral areas), mobility has been exploited to "assist" geo-routing; i.e., a vehicle carries packets and forwards them to a newly found vehicle that is moving towards the destination [38,69,73]. A prerequisite for geo-routing is a location service that tells where the destination is. However, it is challenging to devise an efficient, scalable, and robust location service protocol due to the dynamic nature of vehicular networks (mobility, channel errors) [72]. An elegant way of reducing this cost is by exploiting spatial-temporal correlation that exists in most realistic mobility patterns; i.e., the distance between two nodes is more or less correlated with the time elapsed since they last encountered each other. This observation brought forth Last Encounter Routing (LER) [31]. In MobEyes, since each vehicle can piggyback the current position into its meta-data advertisement, LER can be supported at no extra cost. LER, however, does not address the intermittent connectivity problem. In MobEyes, LER is enhanced with the carry-and-forward functionality. The enhanced LER plays a key role when an agent tries to retrieve the actual data, or to send a dump request to the target vehicle.

3. Problem definition: multi-agent coordination

Our primary objective of this paper is to *design an algorithm to control the movement of multiple agents to harvest dynamically generated meta-data in an unspecified region as efficiently as possible.*

The design goals for our multi-agent harvesting algorithm can be summarized as follows:

- *Low communication overhead*: The protocol should not involve a tight, close range control of agents' movement based on continuous streaming of agents coordinates since such a scheme would incur heavy communication overhead, and would not be robust in the face of intermittent connectivity.
- *Data harvesting efficiency*: The protocol must be efficient. Ideally, we want our algorithm to perform similarly to a centralized coordination algorithm, in terms of data harvesting efficiency (i.e., how fast we can collect all the data that we are interested in) and the control efficiency of agents movement (i.e., how much redundant data collection could be avoided).
- *Self-organization*: The protocol should be self-organizing and adaptive to the dynamics of the environment, such as the changes in the movement patterns and densities of the vehicles in VSN and the data carried by those vehicles, and also to the dynamic events and creation of interesting data.
- *Delay tolerance*: Some or most part of the network formed by vehicles may exhibit intermittent connectivity. Hence, we require our algorithm to work well in a delay tolerant scenario.

For all the requirements listed above, we notice that the nature has already solved a similar problem. In particular, social insects and animals coordinate their efforts to effectively collect food without prior knowledge of the food sources; yet they are known to solve this problem quite effectively, if not optimally [61]. According to the foraging theory, animals are presumed to search for nutrients and obtain them in a way to maximize the ratio of energy intake over the time spent for foraging. In effect, they are solving the multi-agent harvesting problem.

We also notice that our problem is related to a problem found in robotics, in which multiple cooperative robots try to complete a task quickly and reliably. In this case, three generic multi-agent tasks have been considered, namely forage, consume, and gaze. The forage task is to wander around the environment looking for the items of interests (attractors) [51]. After picking up these items, the robot agent takes them to a specified home base. Similarly to forage, the consume task involves wandering about the environment to find attractor, but upon finding an attractor, it performs a task on the object (e.g., toxic waste cleanup, assembly) [67]. In the gaze task, a robot searches for an area that has not been gazed, moves towards it, and then gazes over it until the entire environment has been covered (e.g., vacuuming house, lawn mowing) [43]. Our meta-data harvesting scenario is similar to the consume task. However, there is a major difference. In our case, the data sources are mobile and the meta-data are epidemically disseminated in the network. Since vehicles move in an urban grid, it is possible that after a while the same area may become "productive" again, as more data pours in from alternate paths or (in the case of trajectory tracking) from different view points. Thus, harvested areas are replenishable after a time-out and for a limited period. To the best of our knowledge, the literature in cooperative robots did not deal with this scenario.

4. Foraging behaviors in the nature

We review key foraging behaviors observed in the nature that can be applied to tackle our problem. We first review the social foraging behavior of insects such as ants and honey bees. These insects use stigmergy or dances to communicate information about the food sources and recruit other members. We then review the chemotaxes of *E. coli* bacteria, which operates in various modes, i.e., for searching nutrients and moving towards an area whose density in nutrients is high. Finally, we review the model for Lévy walk, which can be used to take agents to a relatively long distance so that when multiple agents *collide* in the same area, they can exhibit an efficient search behavior.

4.1. Stigmergy

Information harvesting, which is our main theme of this paper, is directly related to the food foraging problem solved by stigmergy. Ants need to find routes in an effective manner to possibly ephemeral food sources. Since it is not immediately obvious how long the current site will remain as a valid foraging site, they have to solve a dynamic problem of remembering a rewarding source while exploiting newly discovered food sites. It has been established that ants can optimize their foraging by selecting the most rewarding source. Several works have confirmed that the selection is the result of a trail modulation according to food quality, and have shown the existence of an optimal quantity of laid pheromone for which the selection of a source is at the maximum. Moreover, the selection of a rich source is more efficient if many individuals lay small quantities of pheromone, instead of a small group of individuals laying a higher trail amount. In many cases, the nutrients are distributed in *patches*, and the main issue of foraging is finding such patches, and deciding how long it will take before depleting and leaving the current food sources.

Physical contacts and other forms of direct communication, e.g., via sound or vibrations, are limited both spatially and temporally; only neighbors in the vicinity can receive the signal. On the contrary, pheromone trails are long lasting and can be considered a wide broadcast that slowly dissipates in time. Researchers have found that there are different types of pheromones used by ants. First, there are long-lasting pheromones, which are used to maintain the spatial organization of ant networks, and volatile pheromones, which are used to quickly mark routes leading to current food sources. For instance, the pygidial gland L. distinguenda produces a long-lasting trail pheromone (that lasts about 25 min), which guides the ants back to the trail or the colony when they are detached from the trail network [34]. Second, there is a short-live repellent pheromone, which effectively serves as a no-entry signal. This is quite different from attractant pheromones used to mark the routes to food sites.

Although, insects generally use pheromone to mark a trail leading to food sources, their usage in the biological world is not only to generate attraction but also as a flag of dominance, or "antagonism" as found in fish and mammals [28,19]. For instance, the urine signals social rank in tilapia [3], which has the same meaning as the fecal glucocorticoid laid down (and not accumulated) in some mammals [26,30]. In addition, there are several examples of non-accumulability and diffusive (or mobile) behavior. There is recent work on human [42,70] showing that diffusive, sub-threshold quantities (i.e., low quantity and not accumulable) of certain pheromones produce effects in humans.

In the literature, an algorithm following an ant-like behavior has been proposed for network routing (see Section 7 for more discussion); however, the concept of pheromones has not been explored in the context of mobile sensor networks. In this paper, we borrow the idea of stigmergy for indirect communication between agents to mark the areas that they have swept for information. In this way, the agents can move out from the area that has already been searched for by other agents.

4.2. Chemotaxis of E. coli

Another biological foraging behavior that we study in the context of information harvesting is the chemotactic (or foraging) behavior of E. coli bacteria [6]. *E. coli* is one of the main species of common bacteria that live in the lower intestines of mammals including humans. *E. coli* gets its locomotion from a set of rigid flagella that enables the bacteria to swim. When the flagella turn clockwise, the bacteria tumble and do not move to any particular direction. On the other hand, when flagella turn counter-clockwise, the bacteria will swim in a directional movement. The sensors of an *E. coli* are the receptor proteins that are stimulated by external substances. Based on the level of nutrients (or attractants) a bacterium will move in different modes. More specifically, when an *E. coli* is in some substance without food or noxious substances, its flagella will alternate between moving clockwise and counter-clockwise so that the bacterium will alternate between tumbling and swimming. This alternation will move the bacterium in random directions. We can consider this movement mode a *search* for food. If the bacteria are placed in a substance with a nutrient with homogenous concentration, they will exhibit a search behavior but with increased run length of swimming and decreased tumble time. In effect, they will search for nutrients more aggressively when they are in a nutrient environment. Finally, when the bacteria detect a change in the concentration level of nutrition, they will swim along the gradient of concentration toward the most nutrition rich area, and spend less time tumbling. If somehow an *E. coli* encounters a region where nutrient gradient does not increase after the swim, it will return to the baseline search mode to search for even higher concentrations.

The chemotactic behavior of *E. coli* is very simple. From an algorithmic perspective, it combines a greedy heuristic with random search – when the nutrient gradient is detected a bacterium moves along the highest gradient and when it hits a hill it starts a random search. A similar strategy can be employed by data harvesting agents – the agents can try to move towards an area where the density of new information is high, and before they detect enough concentration of new information, they can roam around to detect an area with promising outcomes. This is one of the main ideas that we will evaluate quantitatively using simulations in Section 6. Of course, the functional capability of E. coli is very simplistic and primitive, and we should not restrict our algorithms to exactly follow its behavior. Thus we leverage extra information and communication means, e.g., from pheromone trails left by other agents and the history of information density in the region, to coordinate the movements and the queries of the agents more effectively.

4.3. Lévy walk

The intense programs of observations and data collection of animal foraging strategies have attracted the interest of mathematicians and computer scientists. A freely diffusive particle is characterized by a mean square displacement which increases linearly in time, $\langle X^2(t)\rangle \sim t$. However, a variety of interesting physical and biological systems violate this temporal behavior. There is nowadays a growing agreement that foraging and movement patterns of some biological organisms may have so-called "Lévy-flight" characteristics. Lévy flights or random walks, named after the French mathematician Paul Pierre Lévy [41], are known to outperform Brownian random walks for searching when the precise location of the targets is not known a priori but their spatial distribution is uniform. A Lévy walk comprises random sequences of movement-segments, with lengths l, drawn from a probability distribution function having a power-law tail, $p(l) \sim \ell^{-a}$ where $1 < a < 3$. In other words, a Lévy walk has no intrinsic step length scale and thus steps of seemingly very long length may be observed. Such a distribution is said to have a "heavy" tail because large-length values are more prevalent than would be present within other random distributions, such as Poisson or Gaussian. Viswanathan et al., [66,4] demonstrated that $a = 2$ constitutes an optimal Lévy-flight search strategy for locating targets that are distributed randomly and sparsely. Under such conditions, the Lévy search strategy with $a = 2$ minimizes the mean distance traveled and presumably the mean energy expended before encountering a target.

It has been also reported that the Lévy strategy is optimal, which results in space filling paths, if the searcher has no prior knowledge of target locations, and if the mean spacing between successive targets greatly exceeds the searcher's perceptual range. A growing literature is showing that Lévy-flights with $a = 2$ characterize the movement patterns of a diverse range of animals [58,55,54,68,59,65] including, albatrosses, deer, bumblebees, ants, beetles, grasshoppers, spider mites, jackals, microzooplankton, bacterial motion, fruit fly flight patterns, spider monkeys and even human hunter-gathers, and human eye movements.

Although the validity of power law tails in the search pattern of some animal species such as albatrosses and deer is still under debate, there are growing evidences of Lévy search in human physiology and behaviors, which are crucial to our model. For instance, Segev et al. [56] recently reported that Lévy-stable distributions with an inverse-square law tail characterize the electrical activity of some neuronal networks. Odorant receptors whose sequence and structure are strongly conserved from low vertebrates to human, determine scale-free pheromone tracking [54,55]. More importantly, human scan paths and travels have Lévy characteristics [9,10]. One of the main conclusions from the past research on Lévy walks is that occasional long jumps combined with short range search makes an effective search pattern when the target locations are unknown, and randomly dispersed in a large area. We use this insight to design the wide area search behavior of the agents, especially when they discover multiple agents operating in the same region.

5. Bio-inspired agent coordination for meta-data harvesting

Inspired by the foraging behaviors and search patterns found in the nature that we studied above, we now present an algorithm for multi-agent meta-data harvesting in urban environments.

5.1. Information density and datataxis

One of the key insights learned from the common foraging behavior in the nature is that, to maximize energy intake efficiency over time and labor, each agent should move towards geographical areas where nutrition is currently richer. In MobEyes, vehicle mobility is exploited for effective and inexpensive meta-data dissemination, i.e., regular cars carry-and-forward meta-data to harvesting agents. Therefore, meta-data are likely located where the number of vehicles is greater. As an indicator to the concentration of information, we define the *information density* as the number of meta-data carriers, i.e., regular cars actually transporting meta-data, in a road segment. In our scenario, it is obvious that vehicle density is proportional to information density. Like *E. coli* bacteria, our goal is to find a *patch* that contains a large number of "useful" meta-data carriers with information not yet harvested by either the same or another cooperating harvesting agent. For instance, Fig. 2 shows an example of Manhattan grid style road layout, and the number of vehicles in S # 1 is greater than that in S # 2. As a first level approximation, a promising solution for agents is to mimic the foraging behavior of *E. coli* by estimating the gradient of *information density* and moving to a direction where this gradient increases (similarly to the swim of *E. coli* in a solution with a nutrient gradient), while performing a random search when there is no specific gradient (similarly to the tumble of *E. coli* in a homogeneous environment). We name this bio-inspired behavior of harvesting agents as *datataxis* (inspired by the chemotaxis of *E. coli*).

The key for effective datataxis is to estimate the vehicle density of a region in a decentralized way with minimum overhead. To achieve this goal, we use the concept of a virtual urban grid to divide an urban area into a set of segments of equal size. With GPS and similar location services available to MobEyes agents the concept of virtual grid can be easily implemented. In this paper, we use a special type of urban grid based on road segments that are valid in Manhattan-style urban area for illustration and evaluation. For the sake of simplicity, we define a road segment as segment of a road in between two nearby junction points, or intersections. For instance, the two junction points, S # 2 and A # 1, and S # 2 and A # 2 in Fig. 2, form a road segment.

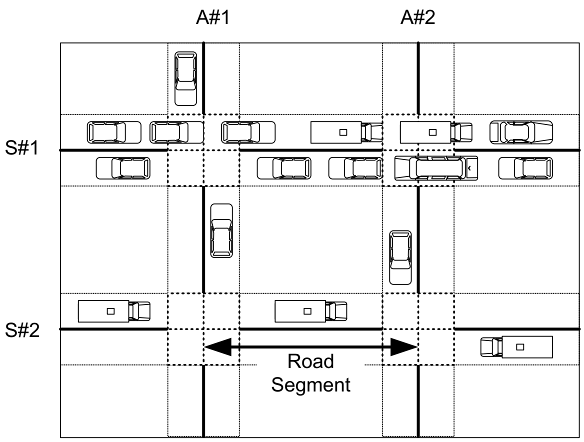


Fig. 2. Urban road layout (Manhattan grid) example.

While MobEyes regular nodes are in a specific road segment, they periodically estimate density of that segment by simply counting the number of their neighbors. The per-segment density is the average of the estimated density values of a specific road segment, and agents can collect such per-road segment density samples via the regular MobEyes protocol for meta-data harvesting, with no additional communication overhead. To obtain more accurate and statistically appropriate density estimations, agents locally compute average densities over time in the following way. For each road segment an agent maintains the density estimation over a fixed window of size T(seconds). Time is discretized into bins whose size is Δ(seconds) each, and each epoch starts every Δ(seconds). There are $K = T/\Delta$ bins for each road segment. Let D_i denote the density estimate during $[T_c - (i+1)\Delta, T_c - i\Delta]$ (i.e., the average density for a given epoch) where T_c is the latest epoch time. For each incoming density sample, an agent first checks whether the sample is within the range T; if so, the corresponding bin is updated. The average density is then calculated as the average of all the bins, i.e., $\sum_{i=0}^{K-1} D_i/K$. Note that this average density no longer reflects "absolute" density, but "relative" density over a period of time. In Section 6, we show that this can effectively estimate the real density in urban environments.

5.2. Differentiating the foraging-based behavior of MobEyes agents

Simply implementing the model of a simple *E. coli* behavior in all cooperating agents is insufficient to realize effective harvesting of meta-data in urban environments. As corroborated by the wide set of simulation results reported in the following section, a significant performance improvement can be obtained by differentiating the behavior of harvesting agents depending on their operation state. We have extensively explored bio-inspired coordination behaviors to identify, evaluate, and adopt the most suitable differentiated working modes to obtain the largest harvesting coverage with minimum overhead. In our design, MobEyes agents operate in one of the following three modes: (a) the Lévy Jump (LJ) mode, (b) the Biased Jump (BJ) mode, and (c) the Constrained Walk (CW) mode. The LJ/BJ modes are considered as the exploration stage to find the best possible location to start a more focused search, whereas the CW mode can be considered as the exploitation stage where agents try to harvest as much as possible by carefully and finely controlling their movements. Fig. 3 presents a transition diagram consisting of

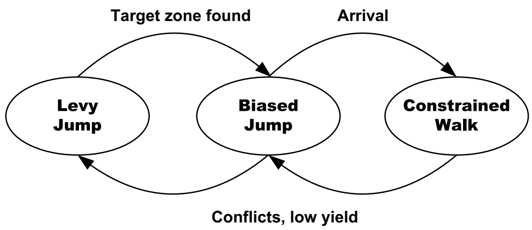


Fig. 3. Agent state diagram.

the three possible states of operation of MobEyes harvesting agents.

First of all, a MobEyes agent starts with the LJ mode and searches for dense areas with vehicles. In the Lévy jump literature, the jump distance following a power law distribution with the exponent of 2 is known to be optimal for non-destructive foraging, i.e., a foraging scheme where the agent can ‘‘productively” visit the same place many times [66].[2] The key idea of the LJ mode is that agents can choose a long distance with some probability, due to the heavy tail of the power law distribution. Thanks to the long jumps, the area covered by the agents will be much larger than the area that would have been covered by only random walk movement patterns [66]. In our model, the network size is finite, and thus, we use a truncated Lévy jump distribution: $f(d) = \frac{d_{\max} d_{\min}}{d_{\max} - d_{\min}} \frac{1}{d^2}$ where we set the $d_{\max}$ as the network diameter and $d_{\min}$ as the communication range. The angle of a jump from the current location is selected randomly. For each jump, the agent steers its movement towards the road segment that minimizes the distance to the new jump location. However, for a given location, it may not be feasible to jump toward a certain direction. For instance, if an agent is located at the bottom left corner of the network, a jump is feasible toward the first quadrant. The key idea of a Lévy jump is to have a long jump with some probability for efficient exploration. Thus, we modify the angle selection such that we only consider the region that can span a chosen distance. In the previous example, the jump direction is chosen from the first quadrant.

Once the agent finds a dense area above a certain threshold d_0, the agent changes its operation state to the BJ mode so that it can move toward that location. The target location is the mid-point of the densest road segment, which is also set as the reference point of the CW mode that will be used by the agent as described below. The agent steers its movement towards the road segment that minimizes the distance to the determined reference point (i.e., a simple greedy movement).

When entering the CW region (the circular area with center the reference point and radius R), the agent switches its mode to the CW mode and starts harvesting meta-data within that region. The default choice in MobEyes is to automatically set the distance parameter R as a function of the number of agents and the size of the overall search area. MobEyes supports two operating sub-modes for an agent in the CW state. First, the agent follows the road segment that maximizes the positive per-segment density change. In this case, since we exclude the current road segment from the candidate road segment for the next movement, it is possible that the rate change may be negative. If this occurs, the harvesting agent chooses the road segment that minimizes the change. Second, the agent can follow a biased random walk along a set of road segments in the vicinity; the set consists of the segments with density greater than a configurable threshold. If the explored urban area has the shape of a long strip, staying within a CW region could be inefficient. For this reason, the MobEyes agent periodically performs short range jumps to explore the nearby area after CW duration T_{cw}, thus changing its reference point. To avoid the worst case of continuous jumping around a region where there is not much gain, after a configurable time interval, the agent performs a long jump to a random direction, and switches its mode to the LJ mode to collect the density information again as in the initial phase. This behavior is repeated until the harvesting procedure has ended.

5.3. Conflict resolution

One crucial issue in multi-agent harvesting is to coordinate the search operations of multiple cooperating agents. Ideally, we want the agents to direct themselves in the richest information areas while not stepping other agents' toes. In other words, each agent coverage area should be non-overlapping with the others and, when agents encounter each other, one of them should be able to quickly move to a different non-overlapping region. This agent conflict problem can be generally handled in the following two ways. The first approach is based on an implicit detection mechanism. When multiple agents are present in the same region, the estimated information density will be lower than normal because much of the meta-data would have already been harvested by other agents. Thus, the agents can infer that there may be other agents if the information density is lower than usual or significantly drops suddenly. We call this implicit since it does not require any extra inter-agent communication.

The second conflict detection mechanism is explicit and is inspired by the stigmergy. Similar to the pheromone trail left by ants, a harvesting agent leaves a trail on the regular vehicles when it collects meta-data. The trail information will contain the ID of a collecting agent and the timestamp of data collection. The trail expires after a fixed period of time $T_{\exp}$. We set the time threshold based on the agent's average speed, and the size of the network. Each regular vehicle records this trail information which is returned to a newly encountered agent. If there are more than two agents harvesting in a same region, a conflict is detected. In this case, an agent with lower ID will perform a long jump to a random location that is outside the CW region of the conflicting agents. As a result, the LJ mode will be initiated and thus, the overall process starts over. Note that although we assume that the agent does not use the previously learned statistics, we can better optimize the knowledge of the history data such that the jump direction can be configured toward the direction that yields a better harvesting rate.

In addition to these two approaches, other more complex solutions are possible. For instance, a harvesting agent could proactively broadcast its presence to the other agents (say up to k-hop neighbors). Moreover, agents could steer their mobility patterns to minimize conflict possibilities, e.g., by forming a Voronoi constellation, at the cost of high communication overhead. However, we find that our simple pheromone trail is sufficiently efficient as conflict detection method as shown in the experimental result section, while imposing only a small amount of overhead.

[2] Recall that since vehicles move in the urban grid, it may be possible that after a while the same area may become ‘‘productive” again.

In this paper, we estimate the information density as proportional to vehicle density. We neglected the fact that in mobility-assisted data dissemination, how frequently a neighbor set changes has a great impact on the efficiency of data dissemination. For instance, if a group of nodes travel together, there will be no information dissemination within themselves. We consider the usefulness of information carriers using Neighborhood Changing Rate (NCR) [32]. Given a sample interval, NCR is defined as a fraction of leaving neighbors over the total number of neighbors (i.e., the number of nodes at the beginning and the number of incoming nodes). This statistics can be used along with density estimates to more accurately estimate the information density in MobEyes.

6. Evaluation

We evaluate the proposed meta-data harvesting algorithm via extensive ns2 [50] simulation experiments run in different urban deployment environments. After describing the simulation setup, we report the accuracy of density distribution estimates perceived by agents and compare various foraging schemes. Then, we analyze the performance of our proposal and its robustness (insensitivity) to configuration parameters by simulating its operation under various parameter ranges.

6.1. Simulation setup

In all the reported experiments mobile nodes communicate by using IEEE 802.11 with fixed bandwidth of 11Mbps and nominal radio range of 250 m. Two-ray ground propagation is the model used for physical layer characteristics [53]. We have chosen the model parameter values according to state-of-the-art work in the vehicular field [73,20].

Vehicle movements comply with the Manhattan mobility model (MT) by Bai et al. [25]. In MT, mobile nodes move on the streets defined by a map that is composed of a number of horizontal and vertical streets. Each street has two lanes for each direction, as shown in Fig. 4. Nodes are allowed to move along the grid of horizontal and vertical streets; at each intersection, vehicles make independent decisions about the next direction; the choice of direction (straight, left, right) is equally probable. To evaluate the independence on specific deployment environment characteristics, we have run simulations in two different sets of scenarios: 7×7 and 14×14 grids (the length of each grid segment is set to 300 m to avoid interference between nearby streets). As shown in Fig. 4, some horizontal streets are initially populated by vehicles: Streets 2 and 6 for 7×7 grids and Streets 3, 8, and 13 for 14×14 grids.

To model the above streets as dense areas, we set transition probabilities such that vehicles on these streets make left or right turns with probability 0.1, and go straight with probability 0.8. When nodes reach the boundary of the simulated region, they bounce back by inverting their direction (modeled by forcing U-turn with probability 1). If this happens, we reset the node and treat it as a new incoming node that carries no meta-data. In this way, we can keep constant the node density in the simulation area. Note that this is a worst case scenario for data harvesting since in reality new cars entering the targeted region may have had opportunities to encounter other cars outside the region and as a result may already carry some useful meta-data.

In addition, MT mimics the mobility of real vehicles by using acceleration and deceleration. Moreover, nodes driving on the same lane maintain a predefined safety distance and the speeds of two consecutive nodes are dependent, i.e., overtaking is not allowed. The speed is randomly drawn from $[v_{\min}, v_{\max}]$. We consider different numbers of nodes $N = 100, 200, 300$. The minimum speed is set to $v_{min} = 1$ m/s and the maximum speed is set to $v_{\max} = 10, 20, 30$ m/s. We fix the speed of harvesting agents to a constant, 10 m/s.

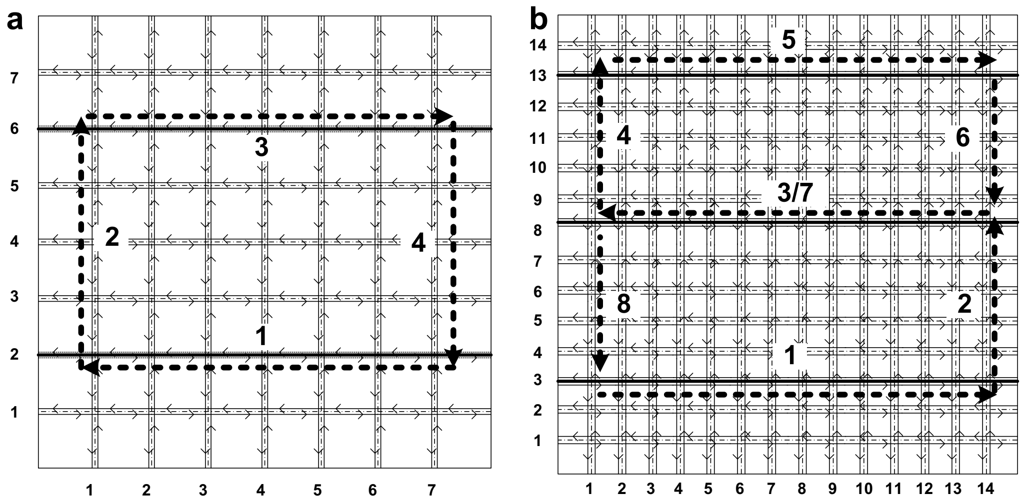


Fig. 4. Street maps used for the Manhattan mobility model: (a) 2400 m $\times$ 2400 m network area. Horizontal streets 2 and 6 (marked with thick solid lines) are initially populated. (b) 4800 m $\times$ 4800 m network area. Horizontal streets 3, 8, and 13 (marked with thick solid lines) are initially populated. In both cases, the regular mobility pattern (marked with thick dotted lines with sequence numbers) is traveled by agents in the Preset Pattern foraging strategy.

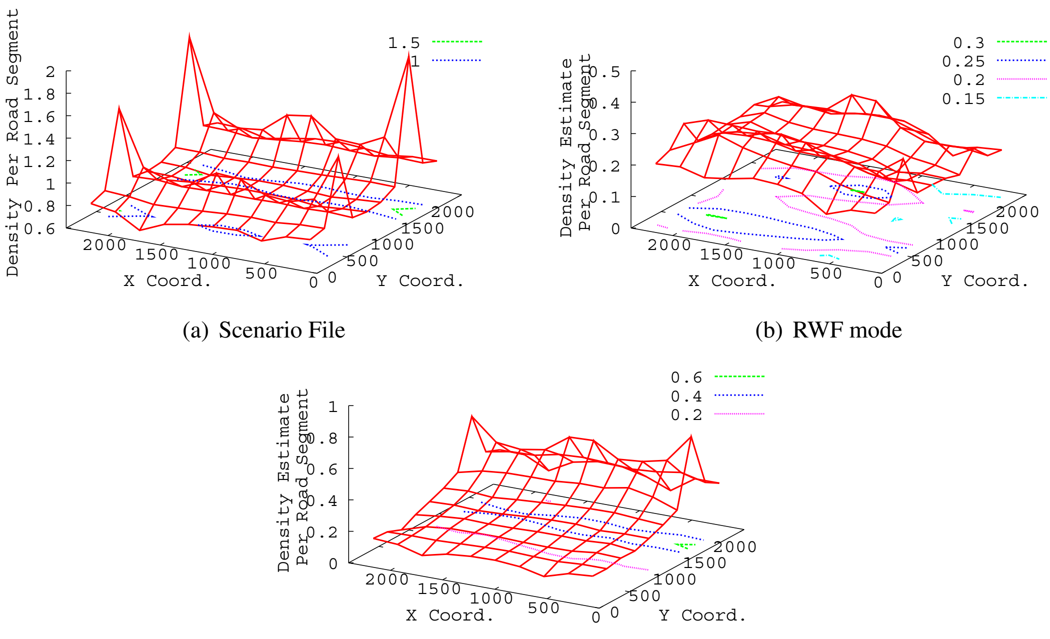
We consider and evaluate the following foraging schemes for meta-data harvesting agents: (a) Random Walk Foraging (RWF), (b) Biased Random Walk Foraging (BRWF), (c) Preset Pattern Foraging (PPF), and (d) Data-Taxis Foraging (DTF). Agents in RWF randomly choose road segments and harvest meta-data from encountered vehicles. Agents in BRWF operate similarly to RWF ones except that they choose road segments based on a defined transition probability that is biased by knowledge about "food sources" and thus, in our simulation setting, they are more likely to stay in initially populated streets (i.e., street 2 and 6 for 7×7 grids, and street 3, 8, and 13 for 14×14 grids). In PPF, we define a preset mobility pattern for the agents representing that the agents are fully aware of the movement patterns of others. To simulate this behavior, we configured the agents to move along the predetermined path that includes the populated streets (as shown in Fig. 4). When the static awareness of other agents' movements is feasible, PPF represents a good agent movement strategy, which ensures the coverage of most popular streets in our scenario. In this case, the agents are equally spaced on the route so that there will be no conflicts among them (recall that agent speed is fixed to 10 m/s).

Finally, DTF implements our proposed scheme for meta-data harvesting coordination. Unless otherwise mentioned, we use the following settings for DTF. An agent explores a region while in the Lévy Jump mode. To estimate meta-data density per road segment, the agent uses the window size of $T = 500$ s. An agent switches its mode to the CW mode if it finds an area where the aggregated density is above $d_0 = 2$. After moving to an information patch, an agent stays there for 300s (in CW mode). An agent performs short jumps within a CW region, where the radius of the CW region is set to 600 m. When a conflict is detected, the agent performs a long jump, where the maximum jump range is set to 900 m. The number of agents used in the simulation varies from 1 to 4 nodes for 7×7 grids and 1 to 6 nodes for 14×14 grids.

The starting positions of the agents are the same for all foraging schemes. The meta-data advertisement period of regular nodes and the harvesting request period are kept constant and equal to 3 s through all the simulations. A new meta-data is generated with the Poisson rate $\lambda = 1/10$, i.e., every 10 s on average. We measure the performance in terms of the total number of harvested meta-data packets. If multiple agents are used, we calculate both per-agent and aggregate harvesting rates. When calculating the harvesting rate, we only count the number of distinct meta-data that have been harvested. Unless otherwise specified, we report the average values out of 30 runs, each of which is 1500 s (25 min).

6.2. Simulation results

We first show the density distribution perceived by agents for different foraging schemes. As described in Section 5, agents collect the density sample points from harvested meta-data. Fig. 5 shows the actual density distribution (from offline computation) and the inferred one (from agent estimation) for the 7×7 grids. More specifically, Fig. 5a shows the distribution computed from the mobility scenario file. Fig. 5b and 5c report the density distribution estimated by a single agent in the RWF and DTF modes, respectively. In both cases, there are two agents in the area, and there are 100 regular nodes with the maximum speed of 10 m/s. In general, popular streets show higher density than the other streets in Fig. 5a. We observe that the density at the four corners is particularly high; this



(a) Scenario File

(b) RWF mode

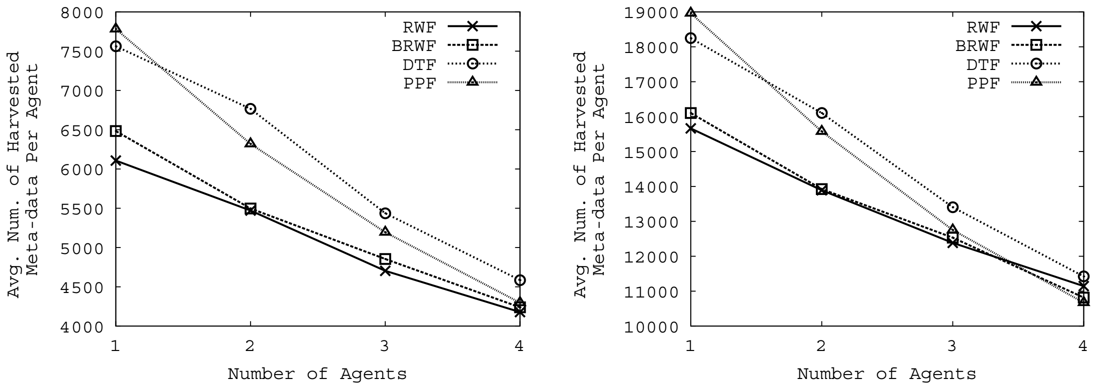
(c) DTF mode

Fig. 5. Density Distribution (7 × 7 girds).

is due to the artifact that when nodes reach the boundary they bounce back to the simulated region. With RWF foraging, agents collect the density samples quite uniformly over the simulated region and, thus, the density distribution estimate is quite uniform as well. However, in DTF, two agents effectively divide their regions of interest, for instance one around Street 2 and the other around Street 6. Therefore, from a single agent perspective, the agent sees much more opportunity along one of the two popular regions. Fig. 5c corroborates the above observation by reporting the estimate for one of the agents, which stayed mostly on Street 6 because of high perceived information density along that line. In general, this estimate provides more balanced and stable data harvesting behavior among multi-agents in DTF.

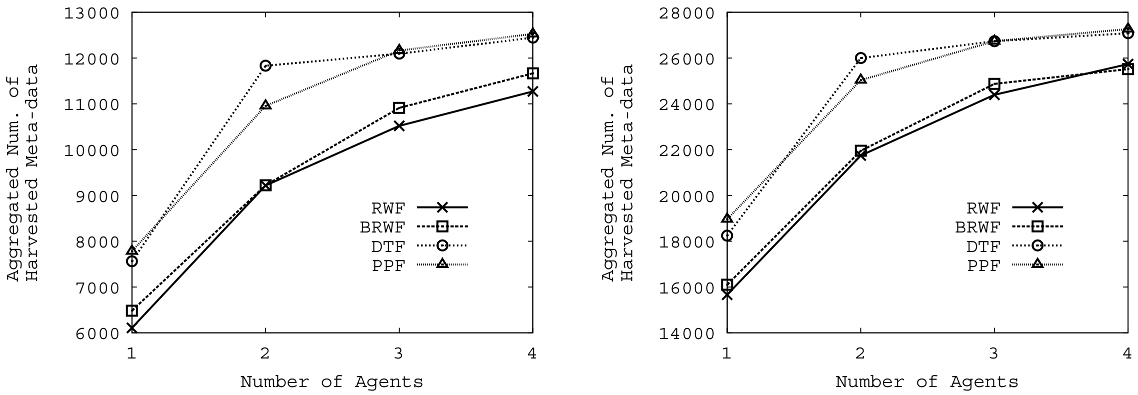
Here, we report the DTF efficiency compared with all the other introduced foraging schemes (RWF, BRWF, and PPF). We first show the impact of the number of foraging agents on harvesting performance. We simulate 100 and 200 nodes with the maximum speed of 20 m/s. We vary the number of harvesting agents in the region from one to four. Fig. 6 shows the number of harvested meta-data per agent. In general, the graph shows that the value decreases as the number of agents increases because we only count unique (non duplicated) collected meta-data. We note that BRWF shows only a slight improvement over RWF. This results from the fact that once the agents in BRWF deviate from the popular streets, it takes a long time for them to return to productive areas. The performance of DTF is consistently better than RWF and BRWF, and is quite close to PPF. Recall that PPF represents a very good foraging strategy in our simulation given the deployment environments and the meta-data density distribution. Thus, we find that our DTF algorithm is efficient, without requiring any static knowledge about the movements of other collaborating agents. Fig. 7 shows the total number of distinct meta-data harvested by all the agents. Also in this plot, we find that the aggregate harvesting ratio of DTF is much better than both RWF and BRWF, and very close to PPF.

Fig. 8 shows the impact of speed on harvesting rate. For the sake of simplicity, we restrict the analysis to DTF and RWF (from our experience we can note that the behavior of PPF is quite similar to DTF and likewise RWF has a similar trend to BRWF). The reported graph shows that the



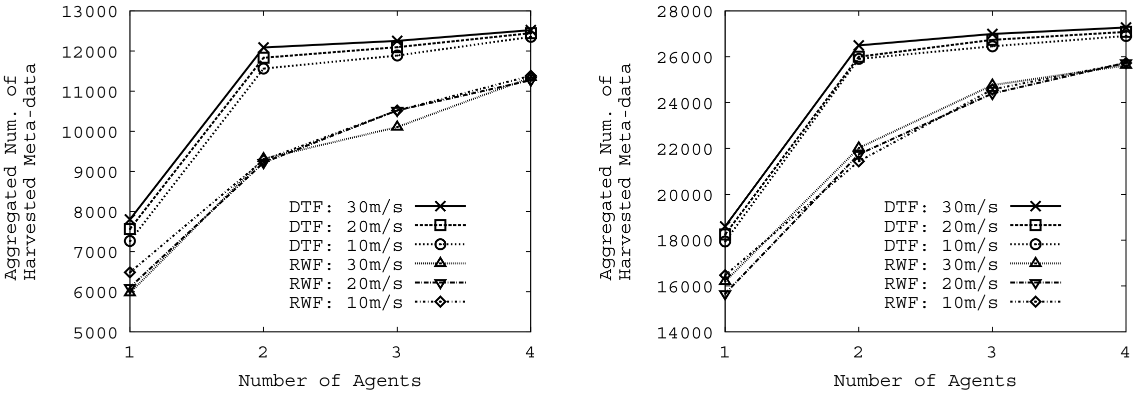
(a) $N = 100$ (b) $N = 200$

Fig. 6. Number of harvested meta-data per agent (7 × 7 grids).



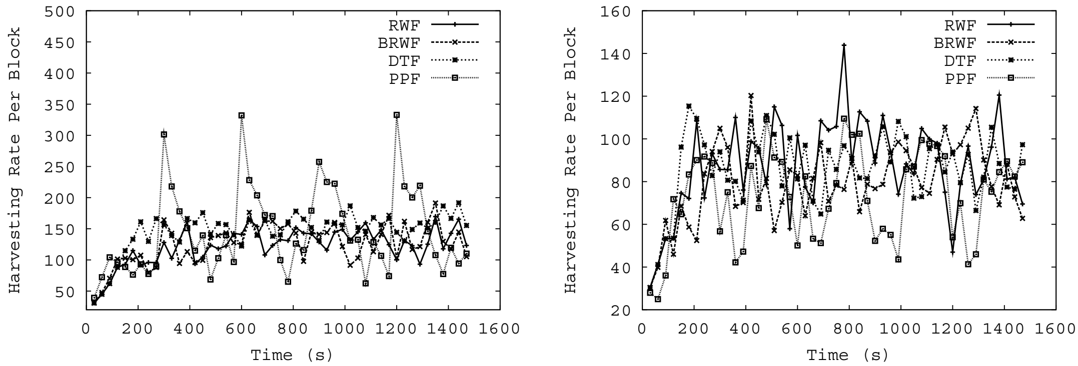
(a) $N = 100$ (b) $N = 200$

Fig. 7. Aggregated number of harvested meta-data (7 × 7 grids).



(a) $N = 100$ (b) $N = 200$

Fig. 8. Impact of speed (7 × 7 grids).



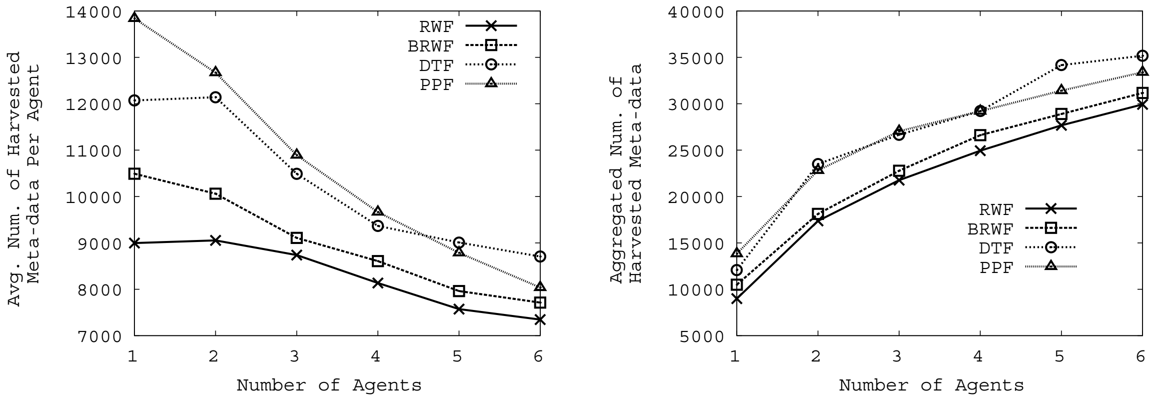
(a) 1 agent scenario (b) 4 agent scenario

Fig. 9. Harvesting rate per road segment over time (7 × 7 grids).

performance is only minimally affected by the typical range of vehicle speeds. In general, the higher the mobility, the faster the data dissemination. At the same time, this shortens the lifetime of a node because we treat a node as departed when it reaches the boundary. The tradeoff between speed and harvesting efficiency is different for each foraging scheme. Yet the performance difference is small. For instance, DTF shows a slightly better performance as speed decreases when the number of nodes is 100. In general, the overall harvesting efficiency is scarcely sensitive to the speed of vehicles for both the cases.

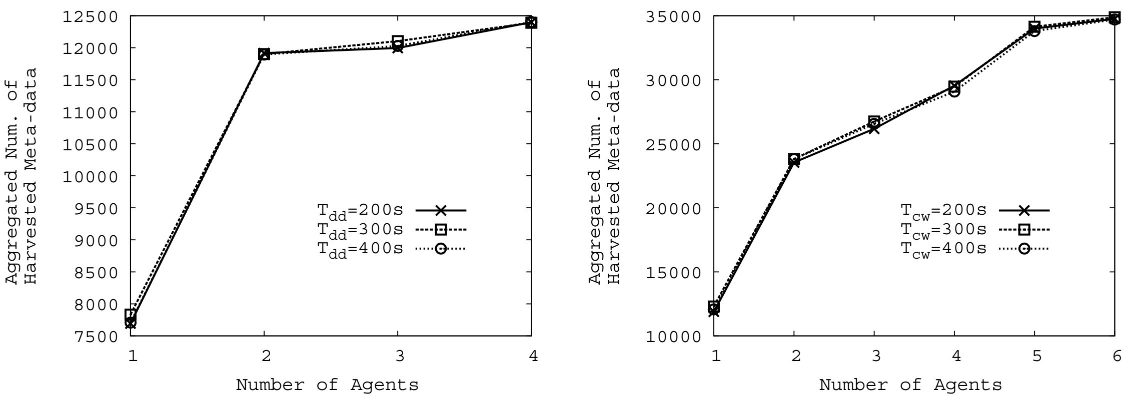
To further analyze the behavior of foraging schemes, we examine the per-agent harvesting rate over time. Each agent is programmed to print out the number of harvested meta-data when leaving its current road segment, and we collect the number over the period of 1500 s. Agent speed is 10 m/s and there are 100 regular nodes with maximum speed equal to 20 m/s. Fig. 9 shows the results for a single agent case (Fig. 9a) and a four agent case (Fig. 9b). From the graphs, we note that the harvesting rates of DTF fluctuate less than that of the other meta-data harvesting strategies. This result stems from the fact that DTF agents tend to remain in high density areas once they find them. On the contrary, we notice that PPF shows drastic changes since PPF agents periodically travel from a highly dense area to a less dense area on the rectangular path.

To evaluate the scalability of the proposed algorithm, we ran simulations with a larger scale network: 300 regular nodes move in 14 × 14 grids at the maximum speed of 20 m/s, with up to 6 meta-data harvesting agents. In Fig. 10, we report the average per-agent number and the aggregated number of harvested meta-data. The figure shows that the overall performance trend with respect to the number of agents does not change compared with the 7 × 7 grid scenario. DTF persistently outperforms RWF and BRWF, and DTF is comparable to PPF. The number of harvested meta-data rather gradually decreases compared to the 7 × 7 grid scenario. Since the area size foraging is quadrupled, the inter-conflict time among agents is longer enough for regular nodes to fill their local meta-data storage. In general, the impact of conflicts for a given foraging pattern is mainly determined by its conflict frequency. For instance, BRWF experiences more conflicts than RWF since agents tend to stay in populated streets; and each



(a) Average number of harvested meta-data per agent (b) Aggregate number of harvested meta-data

Fig. 10. Meta-data harvesting with larger area size (14 × 14 grids, N = 300).



(a) 7x7 grids (N=100) (b) 14x14 grids (N=300)

Fig. 11. Sensitivity of DTF: CW duration (T_{cw}).

agent in DTF is likely to stay in one of the populated streets. Conflicts are also influenced by meta-data generation rate and speed of meta-data diffusion, but these factors are invariant with respect to the choice of the foraging strategy.

Finally, we investigate the sensitivity of DTF performance to two parameters: CW duration (T_{cw}) and CW region radius (R). We use $T_{cw} = 200, 300, 400$ s and $R = 600, 900, 1200$ m. We note that when T_{cw} is too small, the agents tend to hop around the region frequently and, on the opposite, when T_{cw} is too long, it may hinder the agents from exploring other potentially fruitful areas. Fig. 11 shows that DTF well behaves with little dependency on the choice of T_{cw}. Thus, it shows that our algorithm is relatively robust to the T_{cw} parameter selection. The CW radius (R) determines the range of exploitation of agents when they found an information patch. In particular, we consider a range of R from $R = 600$ m, which corresponds to 4×4 grids in our simulation, to $R = 1200$ m, which corresponds to 8×8 grids. Fig. 12 shows the performance of DTF with respect to different R values with $T_{cw} = 300$ s. From the figure, we find that it is relatively insensitive to the choice of R.[3] The key is that an agent can effectively identify dense road segments, as shown in Fig. 5c. For large R, an agent will be able to effectively explore the populated streets; for small R, periodic short jumps after T_{cw} seconds help to explore other parts of the populated streets. In any case, DTF effectively enables agents to follow the gradient of information density. From these results, we conclude that the proposed meta-data harvesting algorithm is robust and scarcely sensitive to a particular choice of parameter values, thus fitting well different deployment environments and application requirements, without the need of a careful, manual, and fine tuning of its configuration settings.

7. Related work

In this section, we review VANETs, vehicular sensor networks, and bio-inspired networking systems. We then

[3] We note that we can draw the same conclusion with $T_{cw} = 200, 400s$.

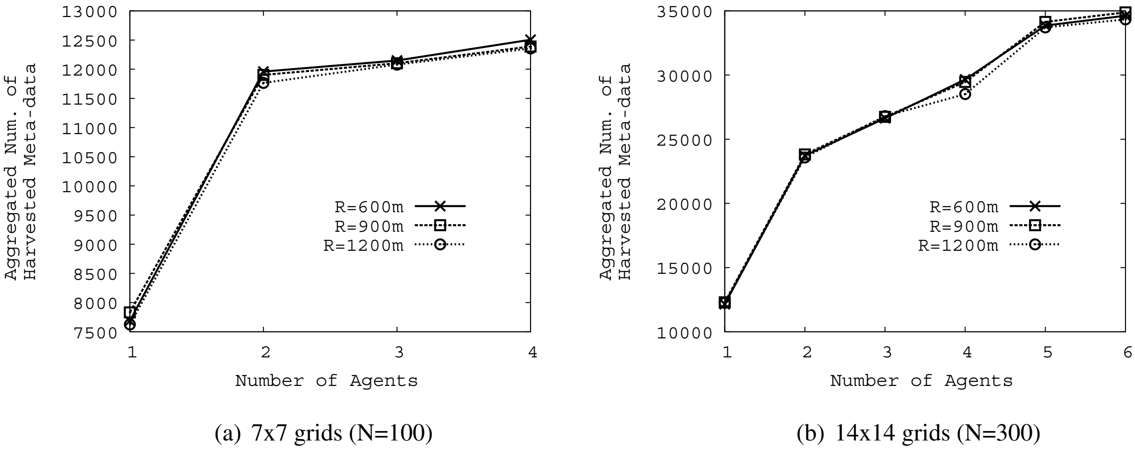


Fig. 12. Sensitivity of DTF: CW region radius (R).

present how our multi-agent collaboration is related to the field of multi-agent robotics.

VANET: VANETs have recently stimulated promising research by envisioning a large number of applications ranging from safe cooperative driving [71] to entertainment support [48,49,40] and distributed data collection [14,60]. So far, however, most VANET research has focused on the crucial primary issue of routing. On one hand, broadcast delivery to all nodes located within a certain area has proved to be crucial for safe driving applications [63,71,37]. On the other hand, packet delivery issues in areas with sparse vehicles have encouraged the investigation of carry-and-forward strategies [73,12].

Urban sensing: A few relevant research activities have addressed the provisioning of wide-scale applications on VANETs and, in particular, urban monitoring services, exploiting the concept of embedding sensors in vehicles. In MIT's CarTel [18,33] vehicles receive queries about sensed data and return replies, by locally running an intermittently connected database that exploits the opportunistic connectivity provided by open access points in their vicinity. Besides VANETs, Dartmouth's MetroSense [47,24] proposed an architecture including both stationary Sensor Access Points (SAP) and Mobile Sensors (MS) carried by users; MS opportunistically delegate tasks to each other, and "mule" [57] data to SAP. CENS recently started the Urban Sensing project [64,13], a multi-disciplinary project addressing "participatory" sensing, where urban monitoring applications receive data from mobile sensors operated by people. BIONETS project [7,17] considered two-tier network architecture for pervasive environment sensing where low-end sensor nodes (T nodes) monitor environments and high-end mobile nodes (U nodes) access information from the sensor nodes. Mobile nodes can access information via a service-oriented communication system; i.e., U nodes use epidemic information dissemination and for scalability, packets are filtered based on the age. To the best of our knowledge, MobEyes is the only system exploiting bio-inspired "behaviors" such as foraging and stigmergy for meta-data harvesting.

Bio-inspired networking systems: Understanding key ideas of how living organisms efficiently organize unreliable and dynamically changing resources and applying these ideas to distribute computing has been an active area of research for the past decade. Babaoglu et al. [1] summarized this by proposing a conceptual framework that captures a few basic biological processes such as diffusion, chemotaxis, and stigmergy. Readers can find the principles of collective animal behavior in [62]. Benefits of bio-inspired technologies for network embedded systems are well documented in [22]. In the following, we do not attempt to provide a complete overview of the huge amount of relevant work accomplished in this area; rather we simply try to sketch primary guidelines on how these bio-inspired ideas can be actually utilized in practice.

Several research activities, e.g., AntNet [15], have proposed Ant Colony Optimization (ACO) for routing in packet-switched networks. For ad hoc routing, a few proposals have already emerged, such as ARA [46], PERA [35], and AntHocNet [16]. ARA and PERA are quite similar to a reactive ad hoc routing protocol, e.g., AODV. On the contrary, AntHocNet is a hybrid (both proactive and reactive) multi-path ad hoc routing protocol and consists of two main processes: stigmergic learning and diffusion. During stigmergic learning, nodes send out ant-like agents (similar to RREQ control packets in AODV) which sample and reinforce good paths to the destination. Routing information is kept in an array of stigmergic variables, called pheromone tables. ARA and PERA share the same concept, but in AntHocNet, this mechanism is further supported by a diffusion process that spreads this information to other agents. Packets are routed under the probabilistic guidance of the learned pheromone tables. Note that bio-inspired technologies are also used for network security such as virus propagation/immunization [29], intrusion detection [21], fault-tolerance [8], attack modeling [36], etc.

Fiore et al., proposed Eureka, which identifies the regions of a network where the required information is more likely to be stored and steers the queries to those regions [27]. The concept of "information density" is proposed to estimate the amount of information cached by nodes in a specific area. Given this metric, queries are forwarded along density gradients, i.e., to nodes with higher density than the forwarder. Each node maintains local density information by monitoring the multi-hop information pulling. The measured local density is shared with the neighbors to make a better estimate. The Datataxis scheme shares the same density gradient idea; however, there are key differences between them: (1) we deal with massive sensor data (specifically, spatio-temporal data streams) and only a mere fraction of data is accessed; (2) we approximate the information density with the physical node density, because MobEyes uses controlled epidemic dissemination of spatio-temporal data, which is driven by physical density; and (3) we use a mechanical search method in which the trajectory of agents is controlled not only by the Datataxis density gradient direction, but also by the bio-inspired Lévy walk concept.

Bio-inspired multi-agent robotic systems: Cooperative multi-agent mobile robot design is based on a behavior-based paradigm where a mobile robot control system is decomposed based on task-achieving behaviors [11,2]. A finite state machine is typically used to describe these behaviors. Each active behavior processor computes its reaction to its perceptual stimuli from sensors, and perceptual triggers cause transitions between states. This paradigm is useful for controlling mobile robots in dynamic environments. In our system, the agent state transition diagram is composed of three states: Lévy Jump, Biased Jump, and Constrained Walk, and the state transition is triggered when certain events happen such as conflicts. Interested readers are referred to [23] for a taxonomy of cooperative robotics and [52] for a brief introduction to the current research topics in multi-robot systems.

The behavior-based paradigm is related to biological systems and thus, many researchers have examined the social characteristics of insects and animals when designing multi-robot systems. The simple local control rules of various biological societies such as ants, bees, and birds have been used to demonstrate the ability of multi-robot teams to flock, forage, etc [44,67]. Moreover, behaviors in higher animals such as cooperation, competition, and selfishness has been also studied in multi-robot control [45]. The effectiveness of communications on the performance of a multi-robot system has been also studied. Balch et al. found that communication provides benefits for particular types of tasks, and in many cases, minimal communication of small amount of information can lead to great performance improvement [2]. Our work is similar in concept to the foraging and cooperation among robots. However, the main difference is in that MobEyes agents are not tightly-controlled as in the robotics case, and we incorporated novel ideas such as Lévy jumps to improve the search efficiency and repulsive pheromone trails to coordinate multiple agents' movement.

8. Conclusion

In this paper, we presented a novel data harvesting algorithm in an urban sensing environment. The proposed algorithm has been designed based on biological inspirations such as (a) foraging behavior of *E. coli* bacteria, (b) stigmergy found in ants and other social insects, and (c) Lévy flight found in foraging and social movement patterns. The proposed algorithm called *datataxis* enables the MobEyes agents to move to "information patches" where new information concentration is high. This algorithm is guided by a practical metric for information density estimate per road segment. In our data foraging strategy, an agent performs a Lévy walk until it finds an information patch; then it performs a constrained walk to move toward a higher density region. When an agent encounters some other agents in the same region it moves to another regions using a conflict resolution algorithm that has been inspired by Lévy jump, so that their work is not duplicated.

We validated the performance of our proposed data harvesting scheme via an extensive simulation study. Using the Manhattan mobility model with linear information patches, we compared the harvesting efficiency of our datataxis foraging (DTF) with random walk foraging (RWF), biased random walk foraging (BRWF), and an idealized preset pattern foraging (PPF). From this study, we showed that DTF effectively balances the movement of multiple agents and distributes them. Also we showed that DTF always performs better than RWF and BRF, and close to PPF, which is optimal. This trend was consistent regardless of the number of agents, the speed of the regular vehicles, and the number of regular vehicles. Finally, we analyzed the sensitivity of DTF's performance with respect to two key parameters of DTF algorithm (namely, the constrained walk duration and the constrained walk radius), and reported that it is quite robust in a wide range of parameter space.

References

[1] O. Babaoglu, G. Canright, A. Deutsch, G.A.D. Caro, F. Ducatelle, L.M. Gambardella, N. Ganguly, M. Jelasity, R. Montemanni, A. Montresor, T. Urnes, Design patterns from biology for distributed computing, ACM Transactions on Autonomous Adaptations System 1 (1) (2006) 26–66.

[2] T. Balch, R.C. Arkin, Communication in reactive multiagent robotic systems, Autonomous Robots 1 (1) (1994) 27–52.

[3] E.N. Barata, P.C. Hubbard, O.G. Almeida, A. Miranda, A.V.M. Canario, Male urine signals social rank in the *Mozambique tilapia* (*Oreochromis mossambicus*), BMC Biology 5 (1) (2007).

[4] F. Bartumeus, M.G.E. da Luz, G.M. Viswanathan, J. Catalan, Animal search strategies: a quantitative random-walk analysis, Ecology 86 (2005) 3078–3087.

[5] P. Bellavista, E. Magistretti, U. Lee, M. Gerla, Standard integration of sensing and opportunistic diffusion for urban monitoring in vehicular sensor networks: the MobEyes architecture, in: IEEE ISIE, Vigo, Spain, June 2007.

[6] H. Berg, D. Brown, Chemotaxis in *Escherichia coli* analysed by three-dimensional tracking, Nature 239 (1972) 500–504.

[7] BIONETS Consortium. <http://www.bionets.org/>.

[8] D. Bradley, C. Ortega-Sanchez, A. Tyrrell, Embryonics + immunotronics: a bio-inspired approach to faulttolerance, in: The Second NASA/DoD Workshop on Evolvable Hardware, Palo Alto, CA, October 2000.

[9] D. Brockmann, T. Geisel, Are Human Scanpaths Lévy Flights? In Artificial Neural Networks (ICANN'99), Edinburgh, UK, 1999.

[10] D. Brockmann, L. Hufnagel, T. Geisel, The scaling laws of human travel, Nature 439 (2006) 462–465.

[11] R.A. Brooks, A robust layered control syste for a mobile robot, IEEE Journal of Robotics and Automation RA-2 (1) (1986) 14–23.

[12] J. Burgess, B. Gallagher, D. Jensen, B.N. Levine, MaxProp: routing for vehicle-based disruption-tolerant networks, In IEEE INFOCOM, Barcelona, Spain, 2006.

[13] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, M.B. Srivastava, Participatory sensing, In ACM WSW, Boulder, CO,USA, 2006.

[14] M. Caliskan, D. Graupner, M. Mauve. Decentralized discovery of free parking places, in: ACM VANET, Los Angeles, CA, USA, 2006.
[15] G. Di Caro, M. Dorigo, AntNet: distributed stigmergetic control for communications networks, Artificial Life 5 (2) (1999) 137–172.
[16] G. Di Caro, F. Ducatelle, L.M. Gambardella, AntHocNet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks telecommunications (ETT), Special Issue on Self Organization in Mobile Networking 16 (2) (2005) 137–172.
[17] I. Carreras, I. Chlamtacand, F.D. Pellegrini, D. Miorandi, BIONETS: bio-inspired networking for pervasive communication environments, IEEE Transactions on Vehicular Technology 56 (1) (2007) 218–229.
[18] MIT's CarTel Central. <http://cartel.csail.mit.edu/>.
[19] P. Chamero, T. Marton, D. Logan, K. Flanagan, J. Cruz, A. Saghatelian, B. Cravatt, L. Stowers, Identification of protein pheromones that promote aggressive behaviour, Nature 450 (2007) 899–902.
[20] Z.D. Chen, H. Kung, D. Vlah, Ad hoc relay wireless networks over moving vehicles on highways, In ACM MOBIHOC, Long Beach, CA, USA, 2001.
[21] D. Dasgupta, F.F. Gonzalez, An immunity-based technique to characterize intrusions in computernetworks, IEEE Transactions on Evolutionary Computation 6 (3) (2002) 281–291.
[22] F. Dressler, Benefits of bio-inspired technologies for networked embedded systems: an overview, In Dagstuhl Seminar 06031 on Organic Computing – Controlled Emergence, Dagstuhl, Germany, 2006.
[23] G. Dudek, M. Jenkin, E. Milios, D. Wilkes, A Taxonomy for swarm robots, In Intelligent Robots and Systems'93, Yokohama, Japan, 1993.
[24] S.B. Eisenman, G.-S. Ahn, N.D. Lane, E. Miluzzo, R.A. Peterson, A.T. Campbell, MetroSense project: people-centric sensing at scale, In ACM WSW, Boulder, CO, USA, 2006.
[25] N.S.F. Bai, A. Helmy, The IMPORTANT framework for analyzing the impact of mobility on performance of routing for ad hoc networks, AdHoc Networks Journal 1 (2005) 383–403.
[26] C. Fichtel, C. Kraus, A. Ganswindt, M. Heistermann, Influence of reproductive sea-son and rank on fecal glucocorticoid levels in free-ranging male Verreaux's si-fakas (*Propithecus verreauxi*), Hormone Behaviour 51 (5) (2007) 640–648. May.
[27] M. Fiore, C. Casetti, C.-F. Chiasserini, Efficient retrieval of user contents in MANETs, in: IEEE INFOCOM, Anchorage, AK, USA, May 2007.
[28] C. Gemeno, A. Sans, C. Lopez, R. Albajes, M. Eizaguirre, Pheromone antagonism in the European corn borer moth *Ostrinia Nubilalis*, Journal of Chemical Ecology 32 (5) (2006) 1071–1084.
[29] S. Goel, S. Bush, Biological models of security for virus propagation in computer networks, Login 29 (6) (2004) 49–56.
[30] L. Gould, T. Ziegler, D. Wittwer, Effects of reproductive and social variables on fecal glucocorticoid levels in a sample of adult male ring-tailed lemurs (*Lemur catta*) at the Beza Mahafaly Reserve Madagascar, American Journal of Primatology 67 (1) (2005) 5–23.
[31] M. Grossglauser, M. Vetterli, Locating nodes with ease: mobility diffusion of last encounters in ad hoc networks, In IEEE INFOCOM, San Francisco, CA, USA, 2003.
[32] J. Harri, B. Zhou, M. Gerla, F. Filali, C. Bonnet. Neighborhood Changing Rate: A Unifying Parameter to Characterize and Evaluate Data Dissemination Scenarios. In WONS'07, Innsbruck, Austria, January 2000.
[33] B. Hull, V. Bychkovsky, K. Chen, M. Goraczko, A. Miu, E. Shih, Y. Zhang, H. Balakrishnan, S. Madden, CarTel: A Distributed Mobile Sensor Computing System In ACM SenSys, Boulder, CO, USA, 2006.
[34] D.E. Jackson, F.L. Ratnieks, Primer: communications in ants, Current Biology 16 (15) (2006) 570–574.
[35] J.S. Baras, H. Mehta, A Probabilistic Emergent Routing Algorithm for Mobile Ad Hoc Networks In WiOpt'03, INRIA Sophia-Antipolis, France, 2003.
[36] J. Kong, X. Hong, D. Wu, M. Gerla, Complexity-theoretic modeling of biological cyanide poisoning as security attack in self-organizing networks, In IEEE 7th International Symposium on BioInformatics and BioEngineering (BIBE), Boston, MA, 2007.
[37] G. Korkmaz, E. Ekici, F. Ozguner, and U. Ozguner, Urban Multi-Hop Broadcast Protocols for Inter-Vehicle Communication Systems, In ACM VANET, Philadelphia, PA, USA, Oct. 2004.
[38] J. LeBrun, C.-N. Chuah, D. Ghosal, M. Zhang, Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks, In IEEE VTC'05, Dollas, TX, USA, 2005.
[39] U. Lee, E. Magistretti, B. Zhou, M. Gerla, P. Bellavista, A. Corradi, MobEyes: smart mobs for urban monitoring with a vehicular sensor network, IEEE Wireless Communications 13 (5) (2006).
[40] U. Lee, J.-S. Park, E. Amir, M. Gerla, FleaNet: a virtual market place on vehicular networks, In IEEE V2VCOM, San Francisco, CA, USA, 2006.
[41] P. Levy, Theorie de l'addition des variables aleatoires, Gauthier-Villars, Paris, 1954.
[42] J. Lundstrom, M. Goncalves, F. Esteves, M. Olsson, Psychological effects of sub-threshold exposure to the putative human pheromone 4,16-androstadien-3-one, Hormone Behaviour 44 (5) (2003) 395–401.
[43] D. MacKenzie, R.C. Arkin, Emergent bucket brigading: a simple mechanisms for improving performance in multi-robot constrained-space foraging tasks, In SPIE Conference on Mobile Robots VII, Boston, MA, 1993.
[44] M.J. Mataric, Designing Emergent behaviours: From Local Interactions to Collective Intelligenc, In International Conference on Simulation of Adaptive Behavior: From Animals to Animats, 1992.
[45] D. McFarland, Towards robot cooperation, From Animals to Animates (1995) 440–444.
[46] M. Gunes, U. Sorges, I. Bouazizi, ARA – The Ant-Colony Based Routing Algorithm for MANETs. In ICPPW'02, 2002.
[47] Dartmouth College MetroSense. http://metrosense.cs.dartmouth.edu/.
[48] A. Nandan, S. Das, G. Pau, M. Gerla, and M.Y. Sanadidi, Co-operative Downloading in Vehicular Ad-Hoc Wireless Networks. In IEEE WONS, St. Moritz, Swiss, Jan. 2005.
[49] A. Nandan, S. Tewari, S. Das, G. Pau, M. Gerla, and L. Kleinrock. AdTorrent: Delivering Location Cognizant Advertisements to Car Networks. In IFIP WONS, Les Menuires, France, Jan. 2006.
[50] ns-2 (The Network Simulator). http://www.isi.edu/nsnam/ns/.
[51] E.H. Ostergaard, G.S. Sukhatme, and M.J. Matari. Emergent bucket brigading: a simple mechanisms for improving performance in multi-robot constrained-space foraging tasks, In AGENTS'01, Montreal, Quebec, Canada, 2001.
[52] L.E. Parker, Current research in multirobot system, Artificial Life Robotics 7 (1/2) (2003) 1–5.
[53] T. Rappaport, Wireless Communications: Principles and Practice, IEEE Press, Piscataway, NJ, USA, 1996.
[54] A.M. Reynolds, Scale-free movement patterns arising from olfactory-driven foraging, Physical Review E 72 (4) (2005).
[55] A.M. Reynolds, M.A. Frye, Free-flight odor tracking in drosophila is consistent with an optimal intermittent scale-free search, PLoS ONE 2 (4) (2007).
[56] R. Segev, M. Benveniste, E. Hulata, N. Cohen, A. Palevski, E. Kapon, Y. Shapira, E. Ben-Jacob, Long term behavior of lithographically prepared in vitro neuronal networks, Physical Review Letters 88 (11) (2002).
[57] R.C. Shah, S. Roy, S. Jain, W. Brunette, Data MULEs: modeling a three-tier architecture for sparse sensor networks, Elsevier Ad Hoc Networks Journal 1 (2-3) (2003) 215–233.
[58] M. Shlesinger, J. Klafter, Lévy Walks Versus lévy Flights, in: H.E. Stanley, N. Ostrowsky (Eds.), The Netherlands: Martinus Nijhoff Publishers, Dordrecht, 1986, pp. 279–283.
[59] D. Sims, D. RIGHTON, J.W. PITCHFORD, Minimizing errors in identifying Le'vy flight behaviour of organisms, Journal of Animal Ecology 76 (2) (2007) 222–229.
[60] D. Sormani, G. Turconi, P. Costa, D. Frey, M. Migliavacca, L. Mottola, Towards lightweight information dissemination in inter-vehicular networks. In ACM VANET, Los Angeles, CA, USA, Sept. 2006.
[61] D. Stephens, K. Krebs, Foraging Theory, Princeton University Press, NJ, USA, 1986.
[62] D.J.T. Sumpter, The Principles of collective animal behaviour, Philosophical Transactions of the Royal Society of London: Series B 361 (2005) 5–22.
[63] M. Torrent-Moreno, D. Jiang, and H. Hartenstein. Broadcast reception rates and effects of priority access in 802.11-based vehicular ad-hoc networks. In ACM VANET, Philadelphia, PA, USA, Oct. 2004.
[64] CENS' Urban Sensing. http://research.cens.ucla.edu/projects/2006/Systems/Urban_Sensing/.
[65] G. Viswanathan, V. Afanasyev, S. Buldyrev, Lévy flight search patterns of wandering albatrosses, Nature 381 (1996) 413–415.
[66] G. Viswanathan, S.V. Buldyrev, S. Havlin, M.G.E. da Luz, E.P. Raposo, H.E. Stanley, Optimizing the success of random searches, Nature 401 (1999) 911–914.
[67] I.A. Wagner, Y. Altshuler, V. Yanovski, A.M. Bruckstein, Cooperative cleaners: a study in ant robotics, The International Journal of Robotics Research 27 (1) (2008) 127–151.
[68] R. Wehner, M.V. Srinivasan, Searching behaviour of desert ants genusCataglyphis, Journal of Comparative Physiology A: Neuroethology Sensory Neural and Behavioral Physiology 142 (3) (1981) 315–338.

[69] H. Wu, R. Fujimoto, R. Guensler, M. Hunter, MDDV: a mobility-entric data dissemination algorithm for vehicular networks, in: ACM VANET, Philadelphia, PA, USA, Oct. 2004.
[70] C. Wyart, W. Webster, J. Chen, S. Wilson, A. McClary, R. Khan, N. Sobel, Smelling a single component of male sweat alters levels of cortisol in women, Journal of Neuroscience 27 (6) (2007) 1261–1265.
[71] Q. Xu, T. Mak, J. Ko, R. Sengupta. Vehicle-to-vehicle safety messaging in DSRC. In ACM VANET, Philadelphia, PA, USA, Oct. 2004.
[72] Y. Yu, G.-H. Lu, Z.-L. Zhang, Enhancing location service scalability with HIGH-GRADE, in: IEEE MASS'04, Fort Lauderdale, Florida, USA, October 2004.
[73] J. Zhao, G. Cao, VADD: vehicle-assisted data delivery in vehicular ad hoc networks, In IEEE INFOCOM, Barcelona, Spain, 2006.

UICHIN LEE is a Ph.D. student in the Department of Computer Science at UCLA. His research interests include mobile wireless sensor networks (e.g., vehicular/underwater sensors), delay tolerant networks (DTNs), and wireless vehicular applications.

Eugenio Magistretti is a Ph.D student in the Department of Electrical and Computer Engineering at Rice University. He received a doctorate degree from the University of Bologna in 2007, and a graduate degree from the same university in 2003, both in computer engineering. His research interests include protocols and algorithms for wireless networks, mobile ad hoc networks, and sensor networks.

MARIO GERLA received a graduate degree in engineering from the Politecnico di Milano in 1966, and the M.S. and Ph.D. degrees in engineering from UCLA in 1970 and 1973. He became IEEE Fellow in 2002. After working for Network Analysis Corporation, New York, from 1973 to 1976, he joined the Faculty of the Computer Science Department at UCLA where he is now Professor. His research interests cover distributed computer communication systems and wireless networks. He has designed and implemented various network protocols (channel access, clustering, routing and transport) under DARPA and NSF grants. Currently he is leading the ONR MINUTEMAN project at UCLA, with focus on robust, scalable network architectures for unmanned intelligent agents in defense and homeland security scenarios. He is also conducting research on scalable TCP transport for the Next Generation Internet (see www.cs.ucla.edu/NRL for recent publications).

Paolo Bellavista graduated from University of Bologna, Italy, where he received the Ph.D. degree in computer science engineering in 2001; he is now an associate professor of computer engineering at the same university. His research activities span from middleware for mobile computing in general to location/context-aware services, from adaptive multimedia and vehicula sensor networks to mobile agent technologies. He has co-authored more than 25 journal/magazine articles and about 70 conference papers. He is senior member of IEEE, ACM, and ICST; he serves in the Editorial Board of the IEEE Communications Magazine, of the IEEE Transactions on Services Computing, and of the Springer Journal of Network and Systems Management. Additional details are available at http://lia.deis.unibo.it/Staff/PaoloBellavista/.

Pietro Lió is Senior Lecturer at the University of Cambridge, Computer Laboratory in England. He leads a group active in the general area of System Biology (stem cell behavior), combining Bayesian statistics and statistical mechanics, Bio-inspired computer science and social networks. He has studied Engineering and Biology at the University of Firenze and Pavia. Previous positions with Nick Goldman (European Bioinformatics Institute) and Newton Morton (University of Southampton).

Kang-Won Lee is a research staff member and the manager of the Wireless Networking Group at IBM Watson Research Center. He is also one of the 8 technical leaders of program, a long term fundamental research program funded by US ARL and UK MOD. Dr. Lee received a Ph.D. degree in Computer Science from University of Illinois at Urbana-Champaign in 2000, and received a B.S. and a M.S. from Seoul National University in 1992 and 1994, respectively. Dr. Lee has been a recipient for the C.W. Gear Award (1999), IBM Research Division Award (2003), and IBM Outstanding Technical Achievement Award (2007). In 2004, he was the president of Korean Computer Scientists and Engineers Association in America (KOCSEA), where he is now on the advisory board. Dr. Lee is a senior member of ACM.