

A Crowd-Cooperative Approach for Intelligent Transportation Systems

Davide Cerotti, Salvatore Distefano, Giovanni Merlino, and Antonio Puliafito

Abstract—As embedded and mobile systems grow pervasive in people lives and expand their reach, paradigms related to Mobile Crowdsensing (MCS) are going to play an ever more prominent role. Innovative methodologies and applications able to unlock such a huge potential are required. A domain where MCS really fits is mobility and transportation. However, for a full exploitation of this paradigm in Intelligent Transportation Systems (ITS), distributed and self management capabilities for the involved nodes-vehicles have to be provided.

This paper is a first step in this direction, laying out an optimization system by exploiting feedback-driven patterns in a distributed-opportunistic way. In this sense, collective intelligence and stigmergic, swarm-based paradigms are adapted to an innovative decentralized MCS pattern towards new approaches in ITS. Their effectiveness is demonstrated through a traffic engineering case study, where route planning services are designed according to the proposed approach and then modeled and evaluated by Markovian Agents.

Keywords—Mobile Crowdsensing; Stigmergy; Traffic Engineering; Traffic Monitoring; Route Planning; Markovian Agents.

I. INTRODUCTION AND MOTIVATIONS

Among the strategic services addressing societal challenges, many governments give priority to mobility and transportation, pushing for Intelligent Transportation Systems (ITS). Core issues in this area are those addressed by traffic engineering, due to wide-scale urbanization processes. In particular the route planning problem, whilst addressed thoroughly for a single traveler in terms of shortest path computation, becomes quickly unwieldy when dealing with a set of travelers. In presence of contrasting optimization objectives, additional data, e.g., traffic information, may help in exploring some domain-specific problem space. Interesting solutions in this direction may come from Information and Communication Technologies (ICT) through, e.g., crowdsourcing.

A promising way to exploit this potential is Mobile Crowdsensing (MCS) [1]. It aims at gathering and harnessing the power and wisdom of the crowds to deal mainly with human-related problems, typically in social, urban and citizen science applications. MCS comprises by definition applications where individuals carrying sensor-hosting embedded systems such as smartphones get collectively engaged in information gathering and sharing efforts to monitor and georeference events which may be of interest for individuals and communities alike. It gets applied successfully in transportation systems endowed

with varying degrees of intelligence, e.g., monitoring of traffic and road conditions, mapping of road network features or other elements of interest.

Most MCS applications typically feature a common, simplified, two-component architecture, one running on the embedded device to collect and disseminate measurements, and a second one as backend. The main drawback of such siloed pattern lies in missing exploitation of proximity or density in topologies. In particular this last point is crucial, since in any kind of high-density scenario, especially in presence of real-time constraints as in certain route planning systems, there needs to be a smart approach to proactively take advantage of proximal nodes and crowded areas. Given the MCS paradigm and forthcoming use cases with specific regard to ITS, where the scope for mobility is really going to match crowds at scale, an opportunistic design pattern may be conceived. In particular, the MCS approach perfectly fits with traffic application requirements, asking for frequent, nearly real-time, updates in both monitoring and route planning decision phases. More in general, a distributed, cooperative, crowd-based solution may be especially useful in case of partial, incomplete or even missing information on the area of interest, e.g., from sudden jam-inducing events to unavailable maps. An opportunistic MCS pattern would also allow the generation of a (live) map of the area by adopting a totally decentralized approach.

Due to the limitation of current (mainly participatory) patterns, MCS has been effectively adopted just in traffic monitoring applications [2], [3]. In this paper we want to fill this gap, demonstrating that the MCS paradigm, through novel opportunistic patterns, can also support the implementation of self-organizing systems. These should be able to autonomously decide the best route for a vehicle based on the traffic information gathered from neighbors, overcoming the drawbacks of existing solutions, mainly on real-time constraints, through a cooperative approach. We propose to use this new MCS development in route planning, broadly defined as the regulation of transport networks in terms of the adaptation process of the theoretical schedules to the real traffic conditions. Indeed, we aim at merging traffic monitoring and route planning tasks according to a decentralized pattern in nodes, resulting in a single combined time-effective iterative strategy.

To this purpose, the main contribution of this paper is threefold: i) an innovative, distributed, opportunistic MCS paradigm; ii) a stigmergic algorithm for implementing MCS-enabled route planning applications, on top of this distributed pattern; iii) an analytical modeling technique, based on Markovian Agents, to evaluate the effectiveness of this approach through relevant traffic stream metrics. Our framework, and in particular the stigmergic algorithm for probabilistic route

D. Cerotti is with Dip. di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Italy. Email: davide.cerotti@polimi.it

S. Distefano is with Dip. di Matematica ed Informatica (MIPT), Università di Messina, Italy. Email: salvatore.distefano@unime.it and with

Kazan Federal University, Russia. Email: s_distefano@it.kfu.ru

G. Merlino and A. Puliafito are with Dip. di Ingegneria, Università di Messina, Italy. Email: {gmerlino, apuliafito}@unime.it

planning, including the distributed MCS paradigm, has been named MoCSACO, from MCS and ACO (ant colony optimization). Indeed, the algorithm adapts and extends the well-known ant colony optimization [4] metaheuristic to the distributed, autonomous MCS environment we propose, adopting a probabilistic routing strategy rather than a deterministic one (e.g., shortest path). This way, our goal is to investigate if a probabilistic approach, which gives more room than a deterministic one for the traffic-sensitive system of vehicles to escape local minima in the search for an optimal solution, performs better than existing solutions.

In the following, we are going to lay out a background about ITS and MCS in Section II, highlighting pros and cons of the MCS paradigm in relation to ITS and proposing our ideas to address route planning related issues. Thus, focusing on the opportunistic approach, cooperative stigmergic strategies for route planning are developed in Section III. A Markovian agent modeling technique, described in Section IV, has been then adopted in the case study of Section V to demonstrate the effectiveness of the MoCSACO cooperative route planning approach against the traffic-unaware one. Finally, in Section VI a discussion on MoCSACO exploitation, some remarks and hints for future work close the paper.

II. BACKGROUND AND RELATED WORK

A. An overview of ITS

ITS naturally spans a wide range of technologies, in particular ICT ones, starting from basic management systems such as navigation ones, possibly to be augmented in the future by systems where artificial “co-drivers” may assist humans during their duties [5]. Yet, there are many other examples of instances of subsystems prone to be enhanced through ICT, e.g., traffic signal control systems, which may leverage some kind of system-optimal routing algorithm for road networks as well, such as game-theory based ones [6]. Moreover, from a technological viewpoint, any delay in information dissemination for vehicle-to-vehicle communication networks [7], so called VANETs [8], considering a traffic-dense configuration as the relevant scenario, can be identified as one of the main challenges to be overcome for any coordination system to really work as expected. Some authors [9] have leveraged Deep Learning to predict traffic flows by dealing with Big Data sources. Such problems were also analyzed by model-based solutions: for instance in [10] a stochastic (hazard-based) model to evaluate the impact of a reliability-safety tradeoff on the travel-time is proposed.

In terms of route planning, apart from shortest path algorithms [11] and system-optimal strategies [12], the problem has also been explored in multiple source-target pairs scenarios [13], for instance dealt with by modifying relevant heuristics with time-dependent variables, e.g., arc weights. In particular these solutions tend to optimize against slightly different metrics, e.g., *earliest arrival times* [14], also considering user preferences [15], [16] and multimodal transport [17]. A good overview on the topic is given in [18], and an up-to-date survey in [19]. Real-time [20] algorithms have been also proposed, in some cases adapting to actual traffic conditions

by exploiting a centralized information infrastructure [21]. Anyway, due to the tight real-time requirements this is still an open issue, and a solution better integrating traffic monitoring and route planning on a local basis, could be promising in reducing the delays due to communication with servers.

B. MCS for ITS

Several works presented and characterized the crowdsensing approach in terms of opportunistic [22], participatory [23], social, and crowdsourcing [24] interaction paradigms, then grouped into the MCS framework [1]. One of the main categorizations of MCS thus spans the participatory-opportunistic spectrum. On the one hand, participatory sensing may be defined as any crowd-sourced sensing activity where each member of the crowd is actively involved, giving feedback when asked or otherwise tagging measurements on a voluntary basis. Conversely, sensing under an opportunistic perspective is essentially *unmanned*: MCS would tap into mobile devices just because people carry those around in their pockets all day long anyway. Thus also the device owners may be included in the data feeding process. Their mobility and situation awareness may be leveraged, in an opportunistic and participatory fashion, to support the collection of fine grained information and semantically tagged data.

Several success stories demonstrate the capability of the MCS paradigm to support the development of effective ITS applications. Among them, mapping activities such as OpenStreetMap [25] and BikeNet [26], aimed at creating an open geographic map of the road networks or bike routes worldwide, respectively, or more specific pothole detection and mapping application [2], gained consensus and large scale participation. Anyway, due to the MCS potential for providing near real-time information, the ITS-related killer application for this paradigm is traffic monitoring. CarTel [27] and NeriCell [2] pioneered tackling this problem by a sensing infrastructure deployed on-purpose, or mobiles, respectively. Then, more advanced solutions have combined both static and mobile sensors, involving street cameras as well as smartphones or vehicle sensors. For instance in [28] a traffic monitoring system for a public transportation service is implemented, and similarly in [3] for vehicle traffic in general. Also large-scale, commercial solutions have developed traffic monitoring services (partially) based on mobile contributions, such as those included in Google Maps (Google traffic [29]), MapQuest and Baidu services. Another interesting application of crowdsourcing in ITS contexts is proposed in [30] where a situation-aware music recommendation system leveraging on crowdsourcing for improving the mood-fatigue status of drivers through intelligent selection of music and songs is developed. A comprehensive overview of crowdsourcing exploitation in ITS is surveyed in [31].

These works demonstrate that, on the one hand the crowdsourcing paradigm perfectly fits with traffic monitoring goals but, on the other hand, it is not adopted in applications requiring active involvement of end-users/contributors through a control loop or a feedback, such as, in the ITS context, real-time global route planning [20]. This is likely due to some

limitations of the current usage of the MCS paradigm itself, mainly focused on participatory patterns instead of opportunistic ones. Indeed, in the development of ITS for traffic engineering and optimization of flows and transportation resources, two main solutions exist: the managed or the unsupervised (cooperative) approaches. The former resorts to centralized systems, such as municipality-controlled street lights, timetables and related information systems. The latter requires leveraging cooperation among traffic vectors, such as private vehicles or buses. Furthermore, a traffic engineering system may be translated into a purely distributed, (network) mesh-dependent subsystem. Identifying *Internet* as the connection facility and *things* as vehicles we have a straightforward mapping onto the Internet of Things (IoT) paradigm, sometimes also declined in literature [32] as *Internet of Vehicles*. Under such premise, this ITS characterization of MCS may just become a pattern under the IoT umbrella term, i.e. a specialization of the platform that an IoT would represent for sensing-related, mobility-enabled, crowd-sourced use cases.

III. A NOVEL COOPERATIVE STRATEGY

In this section we are going to propose an innovative distributed MCS scenario, and therefore a stigmergic strategy for achieving global optimization goals, e.g., aggregate travel time with respect to the whole population affected by route planning services, through distributed cooperation (information exchange, coordinated traveling) of autonomous MCS nodes, e.g., vehicles.

A. A distributed MCS pattern

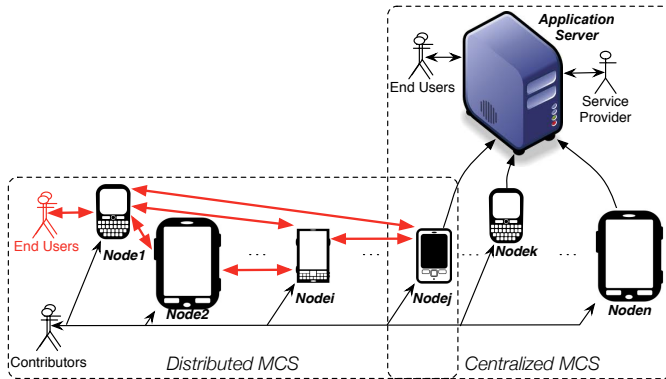


Fig. 1: Centralized and distributed MCS patterns.

Framing the discussion in the aforementioned characterization, we briefly describe a scenario, where cooperation among nodes is important, even essential, especially in the context of mobility and transportation applications. This way, we focus on MCS opportunistic patterns, mainly characterized by push fruition modality and distributed interaction model. As shown on the right part of Fig. 1, typical (*centralized*) MCS applications mainly implement a client-server interaction pattern where a *service provider* offers MCS-based services to *end users*, leveraging *contributors* willingness to provide their physical (sensing) resources. Data are therefore collected and processed by (backend and frontend) *application servers* to carry out analytics and feed back relevant results to end users.

As discussed above this approach does not allow to properly exploit the power of the underlying resources at the edge of the IoT, restricting the applicability of the MCS paradigm to just client/server applications, thus requiring a centralized coordination.

A way to fully exploit this unexpressed potential is by adopting a *distributed* paradigm. This approach is depicted on the left of Fig. 1, where differences between the decentralized and the traditional centralized MCS patterns are highlighted in red. The main differentiation lies in the opportunistic cooperation, a collaborative approach by which nodes may interact one another to aid local computations and perform distributed optimization on a small/medium scale. This way, end users may leverage an MCS application by just exploiting cooperation among nodes. In this scenario, contributors usually also act as end users and viceversa, directly interacting through their nodes/devices, even without any server provider. Furthermore, since a contributor could be involved in different MCS applications, the same node may be simultaneously exploited in both centralized and distributed contribution patterns.

To the best of our knowledge, this is the first attempt at exploiting opportunistic, cooperative approaches in MCS contexts. Some work on opportunistic IoT and sensing environment is available in literature. For example, in [33] an IoT framework is proposed, mainly extending opportunistic networking towards participatory sensing, enabling information sharing among things to also support mobile social networking. Similarly, opportunistic mobile networking is the topic of [34], mainly focusing on low level data forwarding issues through a framework able to support and optimize opportunistic sensing. Differently from these approaches, here we are working at a high level, proposing an alternative or, more specifically, a complementary approach to the centralized MCS paradigm. An alternative, that is, where backend-less operation is possible, as cooperation would work unimpaired anyway, at least in steady-state stages of execution. Existing solutions, as the aforementioned ones, mainly operate at network level and are surely of interest for the implementation of our ideas when dealing with such concerns, which anyway are out of the scope of the present work, focused on introducing and motivating an opportunistic-cooperative approach specifically geared for MCS in ITS.

B. Stigmergic approach

To demonstrate the suitability of the distributed MCS pattern, effective methods and tools enabling opportunistic features are required, in particular with a focus on an ITS scenario, e.g., dynamic route planning for vehicles in a urban setting. Routing protocols for MANETs may be viewed as an enablement step, dealing with both path discovery and node addressing. A promising implementation belongs to the ServalMesh project [35], available for Android mobiles. Over this set of (dynamic) meshes, such as a vehicular crowd, we propose a stigmergic approach for cooperation and optimization, tackling the problem from a probabilistic perspective.

1) *Ant colony optimization*: Ant colony optimization (ACO) [4] is a relatively recent metaheuristic based on the

behavior of ants seeking a path between their colony and a source of food. In nature wandering ants have exhibited in this sense a provable capability to discover nearly optimal paths. The collective intelligence of the swarm derives from the indirect exchange of information among ants via the environment (the so-called *stigmergy*). While traveling to search for food, ants lay down pheromones on their way back to the nest (i.e., home colony) only when sources of food are found. As other colony members step into pheromone trails, they tend to stick to the beaten path accordingly. Moreover, the trace gets reinforced as more individuals follow the same trail, leaving pheromone of their own, in turn resulting increasingly attractive for other ants. For any complex problem which can be reduced to a search for optimal paths, ACO may work as a probabilistic solver, by emulating such naturally occurring behavior. Indeed, stigmergic approaches have been applied to a wide range of problems, including adaptive traffic routing [36] and route planning [37] among others. Nor are choices in this space limited to ACO as, e.g., gossip protocols provide similar [38] distinctive features, i.e., relying on local information, being round-based and relatively simple, and having a bounded information transmission and processing complexity at each round. Yet ACO is more geared toward time-insensitive computation of optima, whereas gossip is mostly meant for quick dissemination of recently acquired knowledge, and thus tends to be more biased towards transients.

In details, the probability p_{ij}^k for an artificial ant k , placed in vertex i , to move toward node j is defined as follows:

$$p_{ij}^k = \frac{\tau_{ij}^\alpha \cdot \eta_{ij}^\beta}{\sum_{l \in \mathbf{N}_i^k} (\tau_{il}^\alpha \cdot \eta_{il}^\beta)} \quad (1)$$

where τ_{ij} corresponds to the *quantity of pheromones* laid over arc a_{ij} , η_{ij} to *a-priori attractiveness* or *gain* of the move, computed by some heuristic embedding the cost of choosing arc a_{ij} along the path that leads to the destination (see eq. (6)), and \mathbf{N}_i^k is the *set of neighbors* in node i for ant k , i.e., the nodes directly reachable by the ant. Coefficients α and β are global parameters of the algorithm, and are typically both set equal to 1, so that pheromones and a-priori information have the same importance in the choice of the arc. In typical ACO variants, ants bring food back home after being done with their movement. Denoting T^k as the *tour* of ant k and n as the number of elapsed rounds, C^k is defined as the *length* of T^k , and specifies the amount of pheromones to be placed by ant k on each arc on the trail leading to the food source:

$$\Delta\tau_{ij}^k = \begin{cases} \frac{1}{C^k} & \text{if arc } (i,j) \text{ belongs to } T^k \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

$$\tau_{ij}(n+1) = \tau_{ij}(n) + \sum_{k=1}^M \Delta\tau_{ij}^k \quad (3)$$

where M is the total number of ants in the colony.

At the end of a round, after each ant has completed a move, the extent of pheromones laid over each arc gets reduced (e.g. evaporates), according to:

$$\tau_{ij}(n+1) = (1 - \rho)\tau_{ij}(n) \quad (4)$$

where ρ is a global *evaporation parameter* as well, ranging around 0.5, i.e., halving the pheromone value at each iteration. ACO algorithms achieve their best performance when some form of local search algorithm is employed in combination with the ACO.

2) *ACO-based MCS*: In order to adapt ACOs to the envisioned cooperative MCS-based route planning application, we propose *MoCSACO*, where an ant corresponds to a (physical) mobile device, a vehicle. The intuition of a vehicle behaving like a stigmergic agent is enough to assume active, implicit cooperation as a byproduct. The application of the ACO paradigm to investigate crowd behavior is not new: for instance, in [39] it was used to analyze the pedestrian crowd dynamics during emergency conditions. In other ACO approaches to route planning [20], the focus is on public transportation and the heuristic was modified to deal with reconfiguration of the route plans stop by stop, whilst taboo lists were employed to let the algorithm also look for alternative routes. By comparison here we are redefining the general objective of finding the shortest path on a (weighted) graph in terms of leveraging common, state-of-the-art and readily available heuristics for path discovery. The A^* [11] search algorithm is such a solution, allowing us to apply the stigmergic approach to arc choice and weighting only, i.e., the *admissible heuristic* function in case of A^* , where each arc has a cost defined by a certain metric.

In order to make the a-priori cost (i.e., of choosing an arc along the path towards destinations) explicit, we define:

$$c_{i \rightarrow d} = c_{ij} + \min_{k \in \mathbf{N}_j} c_{j \rightarrow d} \quad (5)$$

the *cost* $c_{i \rightarrow d}$ from node i towards destination d along a neighboring node j as the sum of the cost associated with traversing the arc between i and j , c_{ij} , and that from j to destination along the choice of node k , belonging to the neighborhood of j \mathbf{N}_j^k , which minimizes this distance. The cost of choosing the a_{ij} arc, c_{ij} , has to be defined taking into account the metrics of interest, e.g., the distance, the amount of pheromones (for example see eq. (8)). Given the aforementioned definition, the value of the a-priori gain, $\eta_{i \rightarrow d}$, for a certain choice leading to destination d is computed according to the following formula:

$$\eta_{i \rightarrow d} = \delta / c_{i \rightarrow d} \quad (6)$$

where the relationship is inversely proportional to the (weighted) distance, i.e., a cost, and δ is just a constant, which may be set to 1 according to literature, when referring to such kind of formula, i.e., tying a-priori information to costs.

A further fix, also applicable to the standard ACO variant, would consist in relaxing the requirement that agents, i.e., vehicles, travel back home after finding food, in its stead leveraging the opportunistic inter-node communication for near-instant swarm-wide dissemination of pheromone trails. This way, probability p_{ij}^k of eq. (1) has to be adapted to any MoCSACO artificial ant, placed in vertex i , to move toward

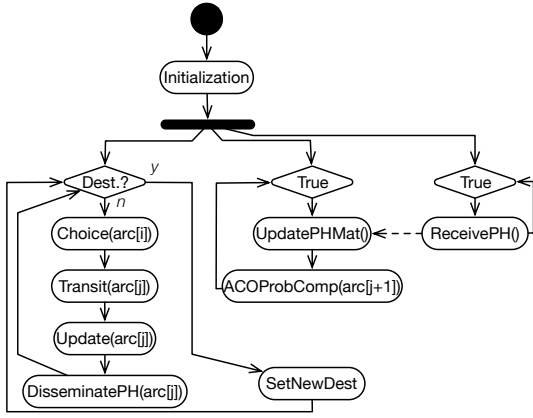


Fig. 2: MoCSACO activity diagram.

node j , along the path to destination d , as follows:

$$p_{i \rightarrow d}^k = \frac{\tau_{ij}^\alpha \cdot \eta_{i \rightarrow d}^\beta}{\sum_{l \in \mathbf{N}_i^k} (\tau_{il}^\alpha \cdot \eta_{i \rightarrow d}^\beta)} \quad (7)$$

where τ_{ij} corresponds to the quantity of pheromones laid over arc a_{ij} , $\eta_{i \rightarrow d}$ to a-priori attractiveness of the choice, computed by some heuristic embedding the cost of choosing arc a_{ij} along the path that leads to the destination d , and \mathbf{N}_i^k is the set of neighbors in node i for ant k , i.e., the admissible transitions for the ant. The main difference between eq. (1) and eq. (7) lies in accounting for the destination d selected by the ant (driver) k in eq. (7). Even in this case, pheromone gets updated as stated by eq. (3).

In MoCSACO, there are as many objectives as destinations, choices are unpredictable (in the sense that an autonomous agent, e.g., a driver - or even a self-driving vehicle if overriding policies are enforced - may choose to disregard indications, or even just drop off the cooperative efforts by stopping its own instance of MoCSACO), and there cannot be a notion of rounds for such kind of agents. It follows that there is not an applicable notion of convergence, and pheromone laid over each arc evaporates, still according to eq. (4), but on a time basis, by setting per-arc countdown timers (possibly preset to a default value), to be reset at each pheromone update. Moreover, A^* may then be considered a degenerate version of the MoCSACO algorithm, in the sense that a sparse density of agents, or just slow transitions, for whatsoever reason, may induce depletion of pheromones, which may be counter-acted upon by choosing a very low value for the evaporation parameter and/or the timer frequency. In turn depleted pheromones would lead to perturbations in the computation of probabilities, translating into unreliable estimates ultimately. This is to be accounted for by reverting arc choice to (deterministic) A^* computations each time $\sum_{l \in \mathbf{N}_i^k} (\tau_{il})$ falls below a predefined threshold.

More in detail, in Fig. 2 are depicted the activities pertaining to a single ant joining the distributed system, i.e., starting up and being connected. An *Initialization* phase corresponds to downloading initial pheromone matrix P , possibly from a centralized MCS backend, with constantly available information about traffic, if global Internet connectivity is available. Otherwise, the matrix gets reset to some predefined

defaults with respect to the graph topology, such as an inverse proportionality with respect to the length of road segments. Afterwards we have three concurrent (infinite) loops. In the leftmost (nested) loop, unless the destination has been reached, the system waits for the next destination to be set to re-enter the inner loop. In the latter, the *Choice* of the arc to be traversed is made by randomly extracting one of the neighboring arcs according to the aforementioned probability, followed by an agent-performed (arc) *Transit* phase, where the arc may correspond exactly to the previously chosen one (autonomous vehicles) or be just a suggestion for the driver to follow (possibly leading to $j <> i$), triggering a new pheromone value (*Update*) for the arc actually just traversed, before the dissemination (*DisseminatePH*) of such update. Otherwise updates may only happen (and thus be promptly disseminated) as soon as timers (one for each arc) expire, producing evaporation. The loop in the center of the diagram, periodically updating the pheromone matrix (*UpdatePHMat*) before every computation of the probability for the neighboring arc(s) to be traversed (*ACOProbComp*), a function which gets triggered by edge visits. The rightmost constantly listening for, and collecting, updates (*ReceivePH*) from the mesh.

C. MoCSACO route planning

Here we contextualize the general MoCSACO approach described above in the ITS domain. In particular, we analyze an instance of a route planning problem implemented through a partially decentralized navigation system for automotive usage, where both the single user and the (global transportation) system must satisfy a given set of criteria. Such high-level requirement would translate into the unambiguous, lower-level goals of minimizing possibly diverging attributes, thus leading to a tradeoff between the number of vehicles per road segment and the duration of the path of each user.

From this (ITS) perspective, MCS nodes represent embedded automotive devices, e.g., GPS-based navigation systems, installed on each vehicle, or available as detachable devices (e.g., mobiles). The urban-level penetration rate of the GPS signal may impact coordinate sampling accuracy. Yet, as typical for map-based navigation assistance software, it is expected that useful metrics for the algorithm, i.e., travel times, can nevertheless be extracted with arbitrary precision. The accuracy may indeed be tuned by interpolating geolocation timestamping with other clues about road segment boundaries, e.g., measuring jerks with onboard accelerometers.

Through the mesh MCS network the pheromone-encoded (implicit) information about the traffic in the region of interest (RoI) is disseminated to all vehicles. In such context, due to the opportunistic nature of the communications [8], vehicles may have an approximated and partially incomplete view of the whole traffic situation in the area. An aid could however be provided by planning a number of nodes, available as fixed infrastructure (e.g., totems) at predefined sites, helping at least with dissemination [40] duties by coping with uneven sparsity of agents. In this sense, apart from the aforementioned application of plain A^* in each vehicle as degenerate function

for path building, costs may always be adjusted according to a global view of the traffic provided by a centralized server, as shown by the black arrows in Fig. 1 connecting a subset of Internet-connected nodes to the Application Server.

Each vehicle executes the algorithm of Fig. 2 acting as an ant, where the matrix updates originate from the continuous exchange of information with other vehicles. The results of such process are the quantities of pheromones laid over the roads (τ_{ij}), which are proportional to traffic flow intensities, and the probabilities to choose such roads given a specific destination ($p_{i \rightarrow d}$), in turn inversely proportional to traffic flow intensities. These values are used to define the costs associated with the arcs that in turn define the heuristic function used by the A^* algorithm executed by each embedded device. In order to implement the proposed routing strategy eq.s (5)-(6) are affected. In particular, we define the cost for the route planning problem:

$$c_{ij} = l_{ij} \cdot w_{ij} \quad (8)$$

with l_{ij} as the (physical) *length* of the (road) segment, and the *weight* of the arc w_{ij} defined as:

$$w_{ij} = \gamma \cdot \tau_{ij} \quad (9)$$

where γ is a constant of proportionality (may be typically set to 1) and τ_{ij} represents the amount of pheromones placed on arc a_{ij} . Being T^k the tour or *traveled path* of ant (i.e., vehicle) k , C^k of eq. (2) is redefined as:

$$C^k = \frac{t_e}{T_l^k} \quad (10)$$

where t_e is the time spent since the trip beginning, assuming vehicles continuously traveling, and T_l^k is the length of T^k . This way, the amount of pheromones to be placed by ant k on each arc is still specified by eq. (2). Referring to eq. (7), to establish an inverse proportionality between traffic flow intensities and probability implies that $\alpha \leq \beta$, assuming that an amount of pheromone lower than 1 identifies relatively uncongested segments, thus modifying eq. (4) as follows:

$$\tau_{ij}(n+1) = \max\{1, (1-\rho)\tau_{ij}(n)\}. \quad (11)$$

IV. MODELING MOCSACO

To evaluate the MoCSACO approach a specific technique able to stochastically represent the ant colony interactions is required. In particular we are mainly interested in evaluating the effectiveness of MoCSACO in terms of traffic distribution. The usual way to study ACO problems is through simulation, since classical analytical techniques, such as state space-based ones, are affected by the well-known state space explosion problem, due to the large number of involved elements (i.e., ants and roads). However, new stochastic entities, called Markovian Agents (MAs) [41], have been introduced to provide a flexible, powerful, and scalable technique for modeling complex systems of distributed interacting objects thus evaluated through feasible analytical and numerical solution algorithms. Moreover, MAs are suitable to represent systems able to self-organize their topology adapting to environmental changes [42] such as ant colonies.

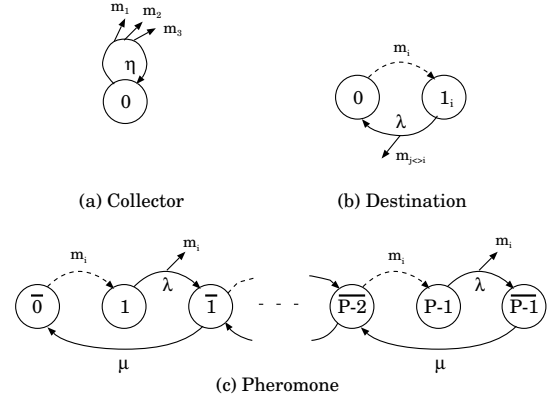


Fig. 3: MA agents representing: (a) collector nodes MA^h , (b) destination nodes MA^d , (c) pheromone amount MA^p .

An MA is an entity that can evolve autonomously according to its local behavior, but interacts with the environment and with the other agents. In particular, an MA is a finite-state continuous-time homogeneous Markov chain (CTMC) that evolves according to a given transition rate matrix and is located in a specific geographical position. The interaction among MAs is represented by the exchange of relational entities, called messages, which are emitted by an MA and perceived by its neighbors influencing their dynamics. The specific interactions among agents are formalized through a *perception function* that rules the aptitude of receiving messages according to agent positions. To model heterogeneous systems, different classes of agents and types of messages are allowed: agents belonging to the same class behave in the same way (i.e., same CMTC structure but different rates), and when receiving a message they may react in different ways according to the type of the message received.

In this paper a preliminary MA model of MoCSACO is proposed to evaluate the traffic distribution on the road network, focusing on traffic flows and disregarding low level details, such as network communication delays. We assume to always work in highly populated conditions, thus ignoring degenerate situations when the application of the A^* algorithm is required (see Section III-B). In such model MAs of different classes are associated with the vertices of a graph \mathcal{G} according to the type of node. A class h agent corresponds to *collector* nodes as entrance points in the RoI for vehicles, a class d is associated with *destination* nodes and a class p agent is used for the other ones. A vehicle, or ant, moving from vertex i to j is represented by a message emitted by an MA located in vertex i , and received by an MA in vertex j . As many types of messages as the number of destinations have to be specified in the MA model.

The model of a collector agent of class h located in vertex v (hereinafter $MA^h(v)$) is shown in Fig. 3(a). It is characterized by a single state with a self loop where the rate of incoming vehicles is η . During its transitions it can emit messages of type m_i with equal probability (in Fig. 3 they are shown as little labeled arrows starting from the self loop) representing incoming vehicles with a probabilistic destination. The MA^d agent is depicted in Fig. 3(b): in state 0 it waits for the arrival of messages; when a message m_i arrives, the agent moves to

state 1_i (depicted as a dashed arrow in the figure) and then comes back to 0 retransmitting a new message $m_j \neq m_i$. This represents a vehicle that, once reached its final destination, decides to change its destination moving towards other nodes. Assuming that the mean time to traverse a node is equal to Tr , we can set $\lambda = 1/Tr$. Finally, the MA^p agent encodes in its state-space the amount of pheromone in the node. In our model, such value is discretized in P levels ranging from 0 to $P - 1$. Thus, state 0 of the MA^p agent represents a node without pheromone, whereas states \bar{p} or p mean the presence of p units of pheromone. At the arrival of a vehicle with destination i (dashed arrow labeled m_i), the amount of pheromone gets increased by one unit, when the vehicle leaves the node moving to a neighbor node (continuous arrow with generation of message m_i) the amount of pheromone is preserved. Pheromone evaporation is represented by a local transition from a state \bar{p} to a state $\bar{p} - 1$, thus pheromone decrements by one unit at time with rate μ .

Let us denote the total density of agents of class c in position \mathbf{v} with $\xi^c(\mathbf{v})$ and $\rho_i^c(t, \mathbf{v})$ the density of agents in state i and position \mathbf{v} at time t . We collect the state densities into a vector $\boldsymbol{\rho}^c(t, \mathbf{v}) = [\rho_i^c(t, \mathbf{v})]$. The routing of messages exchanged by MA^p agents is ruled by the perception function $u_m(\cdot)$ defined similarly to eq. (7). For each destination m_i we have:

$$u_m(\mathbf{v}, \mathbf{v}', t) = \frac{(E[\boldsymbol{\rho}^p(t, \mathbf{v})])^\alpha \cdot \eta_{\mathbf{v}' \rightarrow \mathbf{v}}^\beta}{\sum_{\mathbf{v}'' \in \text{Next}(\mathbf{v}')} (E[\boldsymbol{\rho}^p(t, \mathbf{v}'')])^\alpha \eta_{\mathbf{v}'' \rightarrow \mathbf{v}}^\beta} \quad (12)$$

where the average amount of pheromone $E[\boldsymbol{\rho}^p(t, \mathbf{v})]$ is obtained by the pheromone level corresponding to the states of the agents MA^p . The routing probabilities among other agent classes (i.e. MA^h, MA^d) are obtained in a similar way.

The evolution of the whole model can be studied by solving $\forall \mathbf{v}, c$ the following differential equations:

$$\dot{\boldsymbol{\rho}}^c(0, \mathbf{v}) = \xi^c(\mathbf{v}) \boldsymbol{\pi}_0^c \quad (13)$$

$$\frac{d\boldsymbol{\rho}^c(t, \mathbf{v})}{dt} = \boldsymbol{\rho}^c(t, \mathbf{v}) \mathbf{K}^c(t, \mathbf{v}). \quad (14)$$

where $\boldsymbol{\pi}_0^c$ is the initial probability distribution vector of a class c agent and $\mathbf{K}^c(t, \mathbf{v})$ are the time-dependent infinitesimal generator matrices ruling the whole behavior of agent of class c in position \mathbf{v} . Eq.s (13) and (14) are discretized in time and solved by resorting to standard numerical techniques for differential equations. Details on both the $\mathbf{K}^c(t, \mathbf{v})$ matrices' computation and the solution technique can be found in [41].

V. CASE STUDY

To demonstrate the effectiveness of the MoCSACO approach the traffic of an urban area shown in Fig. 4, close to downtown Messina, has been investigated. A graph $\mathcal{G}(V, E)$ has been derived from this map, where V and E are the sets of nodes and arcs, respectively. Hexagonal vertices represent higher-order nodes, e.g., collector roads or parking lots, while cross-shaped vertices are destination nodes, for instance railway stations or schools in the morning. Both higher-order and destination nodes are randomly placed in the RoI. Finally, the circle vertices are just plain transit nodes. Arcs of the graph represent road segments weighted by the

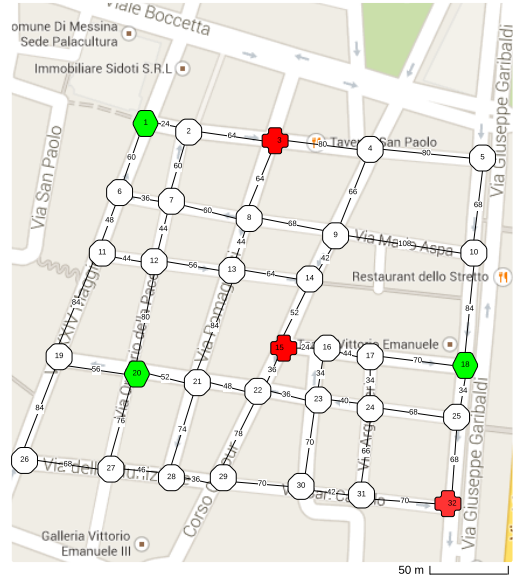


Fig. 4: A distance graph extracted from a road map of the example RoI in Messina.

corresponding lengths l_{ij} . Given such scenario, the objective of the MoCSACO decentralized navigation system is a system-optimal [12] behavior, reducing the overall traffic flows and densities, while limiting straying from user equilibrium, albeit stochastic in the choice of next arc to traverse at each road intersection.

A. Metrics

To evaluate the MoCSACO performance through the case study above described, a specific MA model has been developed. Three different approaches have been therefore compared: i) the well-known Shortest Path algorithm (SP), ii) a degenerate version of an ACO, where the probabilistic arc choice does not depend on the pheromone values (setting $\alpha = 0$ in eq. (12)) (PR), and iii) the full version of MoCSACO (M). In this case study we are interested in evaluating some specific transportation domain metrics able to provide useful insights on the proposed approach. This way, considering a generic arc in the graph from node i to j , we specifically focus on: the *traffic flow* (number of vehicles traversing the road in a time unit) $q_{i,j}$, the *traffic density* (number of vehicles per unit length of the road) $k_{i,j}$ ($q_{i,j} = k_{i,j} \cdot v_{i,j}$ where $v_{i,j}$ is the speed), the *travel time* $t_{i,j}$ and the *travel distance* $d_{i,j}$, the latter two evaluated on the traveled path. Assuming that all the source-destination tours have the same request or arrival rate, in the SP case the number of times an arc belongs to any of the resulting shortest paths for these tours is proportional to the traffic density and flow. In the other cases (PR and M), these parameters are obtained by the model.

To analyze the complex and dynamic behavior of the traffic flow in the MoCSACO model, we first need to identify an instant in time that well represents the average (i.e., steady-state) condition of the system. We say that the $MA^p(\mathbf{v})$ is in a stable state when its average pheromone intensity does not vary anymore and that the whole system reaches the stability when all its nodes are stable (details in [41]). We denote with \bar{t} the first time instant where stability is achieved and, to evaluate

the steady-state behavior of MoCSACO, all the results will be computed in such state.

Since in the MA model the message exchanges represent vehicle movements, the total rate of messages traversing an arc is a proper metric for evaluating the traffic flow in a road. The rate $\gamma(\tilde{t}, \mathbf{v}', \mathbf{v})$ of the whole traffic over the arc $(\mathbf{v}, \mathbf{v}')$ of the graph can be obtained as the sum of the rates for all messages emitted by any agent classes from \mathbf{v} to \mathbf{v}' :

$$\gamma(\tilde{t}, \mathbf{v}', \mathbf{v}) = \sum_m \sum_{c=1}^C u_m(\mathbf{v}', \mathbf{v}, \tilde{t}) \phi^c(m) \rho^c(\tilde{t}, \mathbf{v}). \quad (15)$$

where, for a given message m and a class- c agent, the rate can be computed as the product of the agent density of class- c agents ρ^c that generate messages of type m and the corresponding generation rate $\phi^c(m)$ modulated by the perception function. By the aforementioned indices we can easily derive both the traffic flow ($q_{i,j} \propto \gamma(\tilde{t}, \mathbf{v}_i, \mathbf{v}_j)$) and the traffic density ($k_{i,j} = q_{i,j}/v_{i,j}$) of each road segment as well as the total travel distance and travel time required for the source-destination tours above specified.

B. Evaluation

We analyze the model shown in Fig. 4 with the destination vertices (MA^d) in positions $\{3, 15, 32\}$ and the collector ones (MA^h) in positions $\{1, 18, 20\}$. In the evaluation we set the following parameters for the MA model as reported in Fig. 3: $P = 16$, $\lambda = 10$, $\mu = 2$, $\eta = 5$. In the M case the choice of a high-pheromone node is discouraged by setting $\alpha = -1$ (as can be inferred by eq. (12)), while in the PR case the pheromone is not taken into account in routing ($\alpha = 0$). In both scenarios $\beta = 0.5$, so that the values of $E[\rho^p(t, \mathbf{v})]$ and $\eta_{\mathbf{v}', \mathbf{v}, m}^\beta$ weigh in similarly, thus setting a balanced trade-off between choosing the shortest path and avoiding congested roads. At first, we investigated about the pheromone distribution over the nodes, evaluating the mean value μ of pheromone intensity over the set of vertices of graph \mathcal{G} and its coefficient of variation c_v . In the PR case we obtained: $\mu^{PR} = 2.463$ and $c_v^{PR} = 1.195$; while for M : $\mu^M = 2.734$ and $c_v^M = 0.837$. The higher coefficient of variation c_v^{PR} indicates the presence of congested nodes in the PR case, whereas the higher mean value μ^M shows that the traffic is more evenly distributed in the M case.

Such preliminary considerations have been further investigated by analyzing the traffic flow. The traffic within the urban area in a given road is evaluated by summing up the flow rate $\gamma(\tilde{t}, \mathbf{v}', \mathbf{v})$ between node \mathbf{v} and \mathbf{v}' in both directions. Based on the map of Fig. 4, the three graphs of Fig. 5 show, from left to right, the traffic flow of the SP , PR and M algorithms, respectively. The arcs are colored according to their flow rate: high traffic segments are darker, whereas less congested ones feature lighter hues. The SP solution (Fig. 5(a)) causes the highest level of congestion especially in the segments located in the shortest source-destination paths. The probabilistic algorithms are instead able to redistribute the traffic towards the less congested segments, across the whole RoI. However, also in the PR case there are some very

congested roads, in particular at branches 17-18-10 and 12-20-19. MoCSACO allows to further decrease the traffic of such roads, as shown in Fig. 5(c), at the cost of slightly increasing the traffic at 27-26-19. As above, we can also analyze the mean value and the coefficient of variation of the traffic flow over the set of arcs of graph \mathcal{G} obtaining for PR $\mu^{PR} = 0.045$ and $c_v^{PR} = 4$, while for M $\mu^M = 0.055$ and $c_v^M = 3.636$. Similarly to the pheromone level, also the traffic flow is more evenly distributed in the M case than in the PR one. The analysis of traffic density distribution provides similar results, as shown in Fig. 6. From these results we can argue that MoCSACO allows for the overall traffic to be more evenly spread out over the urban area. Thus the results of the probabilistic (PR or M) models can be used as input for the initialization step of the algorithm described in Fig. 2 to improve the overall accuracy.

Finally, we compare the three algorithms with respect to the travel time and travel distance along a chosen set of tours. The computation of the travel distance is performed in two steps. First, the MA model is analyzed and the probabilities of traversing arcs (given by eq. (12)) in steady state condition are collected for each node of the network. Then, the distance accumulated during a random walk towards a specific destination is computed. Each arc of the walk is chosen according to the previous probabilities and the walk ends the first time the destination is reached. Such cumulated distance can be exactly computed through a probabilistic model enriched with rewards and solved by a probabilistic model checker such as Prism [43]. The travel time is computed in a similar way, but adding a penalty on the time spent in the traversed arcs according to their respective traffic flows.

In Fig. 7 each tour starting from a node s to a destination d is identified by a pair (s, d) . The travel times for the three algorithms is shown on the left. In 7 out of 9 tours there is a significant reduction on the travel time achieved by PR and M algorithms with respect to SP , till about 70%, 30 – 40% on average. The improvement of M on PR is about 15% on average. However, as expected, the saving in time causes an increment of the travel distance, as shown in Fig. 7(b). Indeed, to avoid a congested segment, a vehicle could do a large deviation from the shortest path. In average, in the SP case the traveled path length is 210 m in 5.5 min (with average speed $\bar{v}_{SP} = 2.3$ km/h), in the PR case the traveled path is 2500 m in 4 min ($\bar{v}_{PR} = 37.5$ km/h) and in the M one the traveled length is 2200 m in 3.5 min ($\bar{v}_M = 37.7$ km/h).

The choice to revert to deterministic routing, which would be preferable in certain cases like, e.g., in (18,3) and (18,15), pays off even in terms of limited impact on the detour of vehicles. This condition is met when under a predefined traffic threshold, imposed by aforementioned constraints, namely the fact that insufficient levels of pheromones would lead to an unreliable estimate of the routing probabilities.

VI. DISCUSSION AND CONCLUSIONS

This paper defines a scenario, focusing on opportunistic contribution patterns and self-organizing, distributed approaches, to unlock the MCS potential for the design of innovative ITS applications. The algorithmic solution proposed to exploit

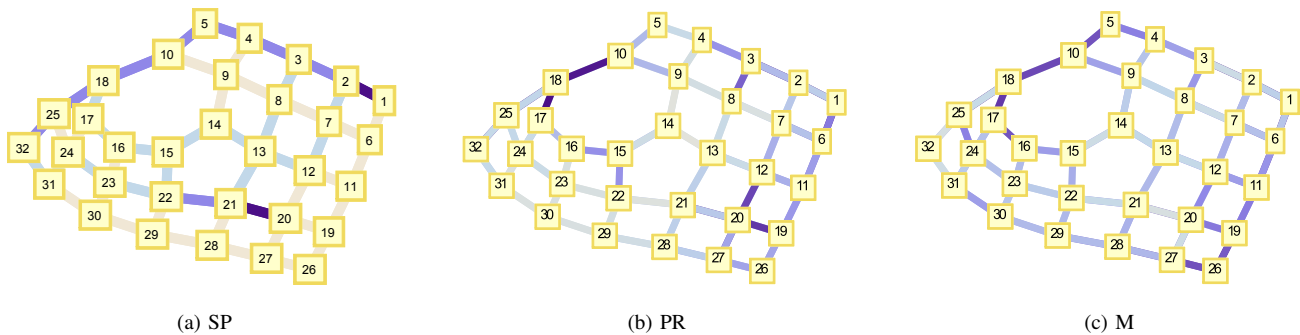


Fig. 5: Traffic flow obtained with (a) shortest path, (b) probabilistic and (c) MoCSACO approaches.

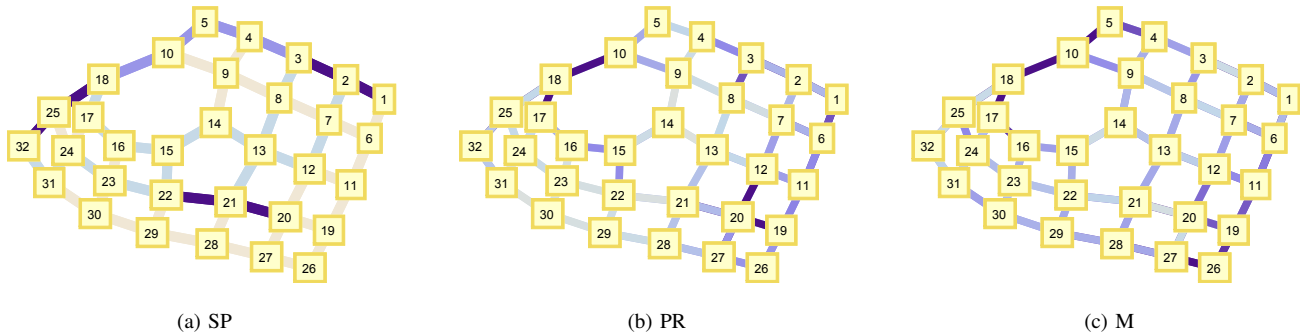


Fig. 6: Traffic density obtained with (a) shortest path, (b) probabilistic and (c) MoCSACO approaches.

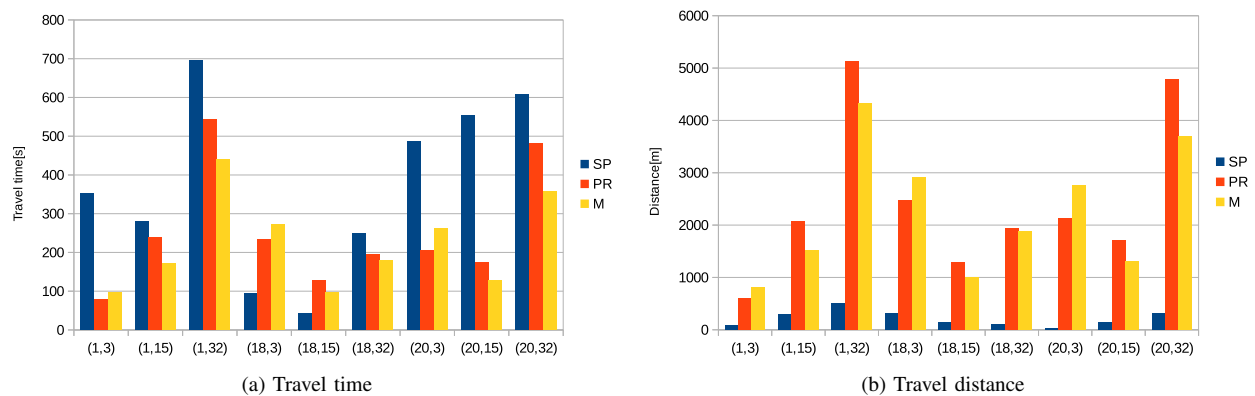


Fig. 7: Comparison of the three algorithms in terms of (a) travel time and (b) travel distance.

this distributed MCS pattern, MoCSACO, adapts and extends an ant colony optimization metaheuristic to a problem of pathfinding and graph traversal according to a given distance metric. The proposed solution has been then characterized into the ITS application domain, by dealing specifically with a traffic engineering problem, exploiting the opportunistic pattern for route planning. To evaluate its effectiveness we used a model based on Markovian Agents, able to represent the interactions and the dynamics of a complex MoCSACO-based system. The model has been analyzed to evaluate the impact on traffic over a certain area due to the MoCSACO approach. Results thus obtained confirmed the effectiveness of MoCSACO in global route planning, indeed spreading the traffic over a given area.

In terms of future work, a challenge may lie in evaluating the approach and algorithm in a real-world scenario, even though the availability of our main target demographics, i.e., autonomous fleets of vehicles, is quickly approaching viability. Still, as long as volunteering owners with their mobiles, or

in-dash car systems, are equipped with up-to-date maps and navigation software libraries, as well as WiFi (for mesh-like communication or even point-to-point upload to base stations along routes), GPS and accelerometers, the most pressing requirement consists in engaging a high enough number of volunteers to set up a suitable experimentation activity. In that sense, the #SmartME [44] project, an experimental testbed for a Smart City in Messina, which is going to include buses and taxis as participating fleets, is poised to provide valuable resources for any crowdsensing effort of this scope.

A development of the algorithm for hierarchical meshes and wider-scope optimization interplay is ongoing. Moreover, hybrid (mesh plus centralized) mechanisms could possibly be part of forthcoming refinements. At last, a custom simulator will be implemented to further validate these approaches.

REFERENCES

- [1] R. Ganti, F. Ye, and H. Lei, "Mobile crowdsensing: current state and future challenges," *Communications Magazine, IEEE*, vol. 49, no. 11, pp. 32–39, November 2011.

- [2] P. Mohan, V. N. Padmanabhan, and R. Ramjee, "Nericell: rich monitoring of road and traffic conditions using mobile smartphones," in *Proc. of the 6th ACM conference on Embedded network sensor systems*, ser. SenSys08. New York, NY, USA: ACM, 2008, pp. 323–336.
- [3] S. Hu, L. Su, H. Liu, H. Wang, and T. Abdelzaker, "Smartroad: A crowd-sourced traffic regulator detection and identification sys." in *Proc. of the 12th Intern. Conf. on Information Processing in Sensor Networks*, ser. IPSN '13. New York, NY, USA: ACM, 2013, pp. 331–332.
- [4] M. Dorigo and T. Stützle, *Ant Colony Optimization*. Scituate, MA, USA: Bradford Company, 2004.
- [5] M. Da Lio, F. Biral, E. Bertolazzi, M. Galvani, P. Bosetti, D. Windridge, A. Saroldi, and F. Tango, "Artificial Co-Drivers as a Universal Enabling Technology for Future Intelligent Vehicles and Transportation Systems," *Int. Transp. Sys., IEEE Trans. on*, vol. 16, no. 1, pp. 244–263, Feb 2015.
- [6] N. Groot, B. De Schutter, and H. Hellendoorn, "Toward sys-optimal routing in traffic networks: A reverse stackelberg game approach," *Int. Transp. Sys., IEEE Trans. on*, vol. 16, no. 1, pp. 29–40, Feb 2015.
- [7] L. Du and H. Dao, "Information dissemination delay in vehicle-to-vehicle communication networks in a traffic stream," *Int. Transp. Sys., IEEE Trans. on*, vol. 16, no. 1, pp. 66–80, Feb 2015.
- [8] E. Lee, E.-K. Lee, M. Gerla, and S. Oh, "Vehicular cloud networking: architecture and design principles," *Communications Magazine, IEEE*, vol. 52, no. 2, pp. 148–155, February 2010.
- [9] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *Int. Transp. Sys., IEEE Trans. on*, vol. PP, no. 99, pp. 1–9, 2014.
- [10] S. Hamdar, A. Talebpoor, and J. Dong, "Travel time reliability versus safety: A stochastic hazard-based modeling approach," *Int. Transp. Sys., IEEE Trans. on*, vol. 16, no. 1, pp. 264–273, Feb 2015.
- [11] P. Hart, N. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *Systems Science and Cybernetics, IEEE Transactions on*, vol. 4, no. 2, pp. 100–107, July 1968.
- [12] A. Chow, "User equilibrium, system optimum, and externalities in time-dependent road networks," in *86th Annual meeting of the Transportation Research Board*, January 2007.
- [13] S. Knopp, P. Sanders, D. Schultes, F. Schulz, and D. Wagner, "Computing many-to-many shortest paths using highway hierarchies," in *2007 Proceedings of the Ninth Workshop on Algorithm Engineering and Experiments (ALENEX)*, pp. 36–45.
- [14] K. L. Cooke and E. Halsey, "The shortest route through a network with time-dependent intermodal transit times," *Journal of Mathematical Analysis and Applications*, vol. 14, no. 3, pp. 493 – 498, 1966.
- [15] M. Baum, J. Dibbelt, T. Pajor, and D. Wagner, *Dynamic Time-Dependent Route Planning in Road Networks with User Preferences*. Cham: Springer International Publishing, 2016, pp. 33–49.
- [16] D. Dellling, A. V. Goldberg, T. Pajor, and R. F. Werneck, "Customizable route planning in road networks," *Transportation Science*, 2015.
- [17] J. Dibbelt, T. Pajor, and D. Wagner, "User-constrained multimodal route planning," *J. Exp. Algorithmics*, vol. 19, pp. 3.2:1–3.2:19, Apr. 2015.
- [18] D. Dellling, P. Sanders, D. Schultes, and D. Wagner, "Algorithmics of large and complex networks," J. Lerner, D. Wagner, and K. A. Zweig, Eds. Berlin, Heidelberg: Springer-Verlag, 2009, ch. Engineering Route Planning Algorithms, pp. 117–139.
- [19] H. Bast, D. Dellling, A. Goldberg, M. Müller-Hannemann, T. Pajor, P. Sanders, D. Wagner, and R. F. Werneck, "Route planning in transportation networks," *arXiv preprint arXiv:1504.05140*, 2015.
- [20] S. Zidi, S. Maouche, and S. Hammadi, "Real-time route planning of the public transportation system," in *Intelligent Transportation Systems Conference, 2006. ITSC '06. IEEE*, Sept 2006, pp. 55–60.
- [21] J. Lübmann, P. Mathias, M. Mann, and M. Fullerton, "Route distribution control in road traffic networks to make on-board navigation systems become part of the traffic management," in *mobiTUM 2013*, 2013.
- [22] N. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. Campbell, "A survey of mobile phone sensing," *Communications Magazine, IEEE*, vol. 48, no. 9, pp. 140–150, 2010.
- [23] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava, "Participatory sensing," in *In: Workshop on World-Sensor-Web (WSW'06): Mobile Device Centric Sensor Networks and Applications*, 2006, pp. 117–134.
- [24] M. Demirbas, M. Bayir, C. Akcora, Y. Yilmaz, and H. Ferhatosmanoglu, "Crowd-sourced sensing and collaboration using twitter," in *World of Wireless Mobile and Multimedia Networks (WoWMoM), 2010 IEEE International Symposium on a*, 2010, pp. 1–9.
- [25] M. Haklay and P. Weber, "Openstreetmap: User-generated street maps," *Pervasive Computing, IEEE*, vol. 7, no. 4, pp. 12–18, Oct 2008.
- [26] S. B. Eisenman, E. Miluzzo, N. D. Lane, R. A. Peterson, G.-S. Ahn, and A. T. Campbell, "Bikenet: A mobile sensing system for cyclist experience mapping," *ACM Trans. Sen. Netw.*, vol. 6, no. 1, pp. 1–39, Jan. 2010.
- [27] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden, "CarTel: A Distributed Mobile Sensor Computing Systems," in *Proc. of the 4th Intern. Conf. on Embedded Networked Sensor Sys.*, ser. SenSys '06. New York, NY, USA: ACM, 2006, pp. 125–138.
- [28] A. Petkovics and K. Farkas, "Efficient event detection in public transport tracking," in *Telecommunications and Multimedia (TEMU), 2014 Intern. Conf. on*, July 2014, pp. 74–79.
- [29] D. Barth, "The bright side of sitting in traffic: Crowdsourcing road congestion data," 2009, <http://googleblog.blogspot.ca/2009/08/bright-side-of-sitting-in-traffic.html>.
- [30] X. Hu, J. Deng, J. Zhao, W. Hu, E. C.-H. Ngai, R. Wang, J. Shen, M. Liang, X. Li, V. C. M. Leung, and Y.-K. Kwok, "Safedj: A crowd-cloud codesign approach to situation-aware music delivery for drivers," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 12, no. 1s, pp. 21:1–21:24, Oct. 2015.
- [31] X. Wang, X. Zheng, Q. Zhang, T. Wang, and D. Shen, "Crowdsourcing in its: The state of the work and the networking," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 6, pp. 1596–1605, June 2016.
- [32] M. Gerla, E.-K. Lee, G. Pau, and U. Lee, "Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds," in *Internet of Things (WF-IoT), 2014 IEEE World Forum on*, March 2014, pp. 241–246.
- [33] B. Guo, D. Zhang, Z. Wang, Z. Yu, and X. Zhou, "Opportunistic iot: Exploring the harmonious interaction between human and the internet of things," *Journal of Network and Computer Applications*, vol. 36, no. 6, pp. 1531 – 1539, 2013.
- [34] D. Zhao, H. Ma, S. Tang, and X.-Y. Li, "COUPON: A Cooperative Framework for Building Sensing Maps in Mobile Opportunistic Networks," *Parallel and Distributed Sys., IEEE Trans. on*, vol. 26, no. 2, pp. 392–402, Feb 2015.
- [35] "Serval project," <http://www.servalproject.org/>.
- [36] P. Kromer, J. Martinovic, M. Radecky, R. Tomis, and V. Snasel, "Ant colony inspired algorithm for adaptive traffic routing," in *Nature and Biologically Inspired Computing (NaBIC), 2011 Third World Congress on*, Oct 2011, pp. 329–334.
- [37] L. Feng, Z. Lv, G. Guo, and H. Song, "Pheromone based alternative route planning," *Digital Communications and Networks*, pp. –, 2016.
- [38] P. Costa, V. Gramoli, M. Jelasity, G. P. Jesi, E. Le Merrer, A. Montessor, and L. Querzoni, "Exploring the interdisciplinary connections of gossip-based systems," *SIGOPS Oper. Syst. Rev.*, vol. 41, no. 5, pp. 51–60, Oct. 2007. [Online]. Available: <http://doi.acm.org/10.1145/1317379.1317388>
- [39] N. Shiwakoti, M. Sarvi, G. Rose, and M. Burd, "Animal dynamics based approach for modeling pedestrian crowd egress under panic conditions," *Transportation Research Part B: Methodological*, vol. 45, no. 9, pp. 1433 – 1449, 2011.
- [40] Y.-T. Yu, X. Li, M. Gerla, and M. Sanadidi, "Scalable vanet content routing using hierarchical bloom filters," in *Wireless Communications and Mobile Computing Conf. (IWCMC), 2013 9th Intern.*, July 2013, pp. 1629–1634.
- [41] D. Bruneo, M. Scarpa, A. Bobbio, D. Cerotti, and M. Gribaudo, "Markovian agent modeling swarm intelligence algorithms in wireless sensor networks," *Performance Evaluation*, vol. 69, pp. 135–149, 2012.
- [42] —, "An intelligent swarm of markovian agents," in *Springer Handbook of Computational Intelligence*, J. Kacprzyk and W. Pedrycz, Eds., 2015, pp. 1345–1359.
- [43] F. Dannenberg, E. M. Hahn, and M. Kwiatkowska, *Computing Cumulative Rewards Using Fast Adaptive Uniformisation*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 33–49.
- [44] "#SmartME [URL]," <http://smartme.unime.it/>.
- Davide Cerotti** is a research associate at Politecnico di Milano. His research interests include performance evaluation and stochastic models.
- Salvatore Distefano** is an associate professor at University of Messina and professorial fellow at Kazan Federal University. His research interests include performance evaluation, parallel and distributed computing, software engineering, and dependability.
- Giovanni Merlino** is a postdoc research fellow at University of Messina. His research interests include mobile and distributed systems with particular emphasis on Cloud, Internet of Things, Smart sensors and Crowdsensing.
- Antonio Puliafito** is a full professor of computer engineering at University of Messina. His interests include parallel and distributed systems, networking, IoT, mobile and Cloud computing.