UNIVERSITÀ DI PISA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

This is a preprint. Please cite this work as:

A.L. Alfeo, M.G.C.A. Cimino, A. Lazzeri, G. Vaglini, "Detecting urban road congestion via parametric adaptation of position-based stigmergy", Intelligent Decision Technologies, IOS Press, Vol. 11, Issue 4, Pages 465-475, 2017, (ISSN 1872-4981)

Machine Learning and Process Intelligence

Research Group

Detecting urban road congestion via parametric adaptation of position-based stigmergy

Antonio L. Alfeo, Mario G. C. A. Cimino*, Alessandro Lazzeri, Gigliola Vaglini

Department of Information Engineering, University of Pisa, Largo L. Lazzarino 1, 56127 Pisa, Italy

luca.alfeo@ing.unipi.it, alessandro.lazzeri@for.unipi.it, {mario.cimino, gigliola.vaglini}@unipi.it

* Corresponding author:

Mario G.C.A. Cimino, mario.cimino@unipi.it

Tel: +39 050 2217 455; Fax: +39 050 2217 600.

Abstract Urban traffic management requires congestion detection. Traffic shape changes over time and location in which it is observed. Moreover it depends on roads, lines and crossroads arrangement. In addition, each congestion event has its own peculiarities (e.g. duration, extension, flow). Therefore, to give correct responses any detection model needs some kind of parametric adjustment. In this paper, we present an adaptive biologically-inspired technique for swarm aggregation of on-vehicle GPS devices positions, able to detect traffic congestion. The aggregation principle of the position samples is based on a digital mark, released at each sample in a digital space mapping the physical one, and evaporated over time. Consequently, marks aggregation occurs and stays spontaneously while many stationary vehicles are crowded into a road. In order to identify actually relevant traffic events, marks aggregation has to be correctly configured. This is achieved by tuning the mark's structural parameters. Considering that each urban area has a specific traffic flow and density, determining a proper set of parameters is not trivial. Here, we approach the issue using different differential evolution variants, showing their impact on performance.

Keywords Urban traffic estimation, Swarm intelligence, Stigmergy, Parametric adaptation,

1 Introduction and motivation

Urban life issues are gaining more and more attention thanks to the rise of the Smart City paradigm. One of the main topics in this field is the traffic congestion management [1]. The main technologies employed in this field can be grouped in two categories: roadside infrastructure and on-vehicle devices. The former examines the traffic state via specific equipment (e.g. camera, loop detectors) installed on the roadside, while the latter refers to on board Global Position System (GPS) to portray traffic condition using vehicle distribution. Moreover, on board GPS offers widespread traffic observation in urban scenario with respect to roadside infrastructure, since the latter is mostly applied to highways and primary arteries. For this reason, we consider on-vehicle GPS a requirement in our approach as it will be used as data sources.

In this paper we rely on stigmergy to aggregate and analyze vehicles' positional data [2],[3]. More in depth, to exploit both spatial and temporal dynamics that characterize urban traffic, we use marker-based stigmergy as a computing paradigm. In the virtual space, digital marks are periodically released in correspondence of each monitored vehicle's position. Marks aggregate and strengthen when superimpose, otherwise they evaporate losing intensity. Marks agglomeration, referred to as track, pops up and stays spontaneously while high density roads and stationary vehicles occur. In addition, input/output activation interfaces are used before/after the stigmergic processing layer, in order to enhance scalability and distinction of congestion events. In this context "activation" (term that comes from neural network domain) refers to a function which triggers an output signal for the next layer only if the input signal reaches a certain level.

The processing layers are parameterized. For example, the mark extension is a parameter of the stigmergic layer. Since each application context has specific traffic flow and density,

determining suitable parameters is not trivial [4]. For this reason we adopt Differential Evolution (DE) as tuning mechanism.

In this paper, the problem statement, its formal characterization, as well as the proposed solving approach and experimental settings are covered. More specifically, Section 2 focuses on related work. Section 3 provides the analysis and the design of the processing layers. Section 4 is devoted to the analysis and the design of the parameters adaptation. Experimental studies are detailed in Section 5. Section 6 covers conclusions and future work.

2 Related work

Taking into account the technology involved in traffic state estimation, a number of methods have been developed. In [5] probe-vehicle data is used to determine kernel-based traffic density estimation. The method first models the traffic data with Gaussian density (centered in the sample position with predefined mean and variance) to extract the kernel parameters. Then, distance between their localized cumulative distributions is measured and optimized, in order to extract the weights of Gaussian kernels in the estimated distribution function. The approximation density function by optimized kernels' weights is finally used to estimate the mobile vehicles density in a specific time and space. In [6] the traffic flow is analyzed by means of GPS and GIS integrated system. In this approach roads are split up into segments, and mean car speed in it is estimated using loop detectors and taxi as probe vehicles, therefrom an approach based on Federated Kalman Filter and D-S Evidence Theory is used, to join such data. Finally, authors propose a curve-fitting method aimed to estimate mean speed in a urban road. It uses least-square method in order to fit data coming from GPS. In [7] the authors pursue a road-segment average traffic velocity estimation, achieved through two different approaches: vehicle tracking and curve-fitting. Experiments

show how a tracking-based method usually bears higher estimate accuracy but slower operational speed with respect to a model-fitting method. In [8] two subsequent GPS samples are used to define a vehicle track by means of the A* algorithm. The combination of tracks velocities passing through the road segment determines the average velocity of the current segment. In [9] an algorithm is proposed to estimate the traffic flow state by using the minimum GPS samples via a curve fitting method. The algorithm takes into account sample frequency, the road type, and the road section length. A spatial and temporal classification of road traffic state based on GPS data is proposed in [10]. Spatial classification aims to represent steady traffic, while temporal classification reflects traffic speed. Authors use GPS samples to calculate vehicles delay distribution over a road segment in order to classify the traffic. Time-location data is converted to spatiotemporal data and then classified using threshold-based quadrant clustering. Authors compare quadrant classifier with maximum likelihood and maximum a priori classifiers.

Traffic management systems are characterized by huge volumes of data that need to be timely analyzed (Big data) for detecting unfolding congestion. Multi-Agents-Systems (MASs) are a promising architecture that decompose the computation among several sub-systems, each operating with partial autonomy and local awareness in decentralized manner. More specifically, Swarm Intelligence is a biologically-inspired paradigm according to which self-organization and complex behavior can be realized by MASs composed by agents characterized by simple behavior [11]. In MASs, coordination between agents can occur in direct or indirect manner. The former is less scalable due to the overload of communication, while the latter works better with massive amount of agents. In the literature, *stigmergy* is a biologically-inspired pattern of indirect coordination. With stigmergy, each agent leaves a sign in a shared environment and stimulates the performance of a subsequent agent's action. In [12] traffic congestion forecast is realized via stigmergy.

Here vehicle flow is measured via fixed on-road sensors and traffic-density is processed via digital pheromone.

Another type of service is the recommendation of a path to avoid congestion. In [13] the authors proposed the DSATJ system, which computes alternative optimum path to avoid traffic jam. Here, digital pheromone evaporation and deposit on a virtual space mapping the roads is managed. The traffic jam is detected via upper bound on the pheromone value. Moreover, diversion of traffic on the roads which had been jammed was represented by normalization of pheromone. While this approach takes advantage from distributed computation that characterizes MAS, it requires that every vehicle involved in the analysis declares its destination and starting point.

In [14] a traffic lights control system based on swarm intelligence is presented. Here, control methods are divided into macroscopic and microscopic levels, and are based on stigmergic evaluation of traffic flow, by using pheromones deposits characterized by evaporation/diffusion dynamics.

In [15] the authors assume the following types of stigmergy: long term, short term, and anticipatory. The main differences lie in how and when the vehicles' position information is stored. Long term stigmergy is archived in a central storage, and provides stochastic traffic congestion information to vehicles. Short term stigmergy occurs while vehicles are sharing current data, and drivers can choose their routes more dynamically, on the basis of such real time information. Anticipatory stigmergy implies that vehicles can declare their destination, in order to distribute pheromones in advance and use them during routing task. Here, aprioristic knowledge on the phenomena is then required. The authors conducted several simulations on traffic analysis to compare the effectiveness of the different kinds of stigmergy. The results demonstrate that only if the traffic network is static, the combination of long term and short term stigmergy overcome the other kinds of stigmergy.

While in [15] the road is considered as a monolithic structure, in [16] roads are divided into segments. Here, congestion evasion strategies based on digital pheromone are investigated. More specifically, every vehicle deposits digital pheromone in the virtual environments, and takes into account the pheromone state to decide upon its subsequent route. The authors propose an algorithm for decentralized, self-organizing, traffic flow improvements adapting the mechanisms of the ant pheromone.

3 Analysis and design of the processing layers

The problem statement and the specification of the proposed system are detailed in this section. As a first insight, the ontology diagram of Fig. 1 shows an overview of the main concepts of our approach. Here, concepts are enclosed in gray ovals and connected by properties, represented by black directed edges. The core concepts are summarized by the following sentences. *Vehicle creates a Road Congestion Event*, which is *detected by a Monitoring System*. The *Monitoring System controls a Stigmergic Environment*, *containing marks* and its *aggregate*, i.e., *tracks,* as well as an *output activation* of the stigmergy, to signal the congestion event. A *Mark is enabled by a hypothetical track*, i.e., a preprocessing mechanism of *space-time position samples released by vehicles*. The performance of the *Monitoring System is evaluated by a Differential Evolution*, which *accordingly adapts the Stigmergic Environment*. More specifically, the adaptation is carried out on four fundamental parameters: *mark extension*, *mark intensity*, *track evaporation*, and *activation inflection*.

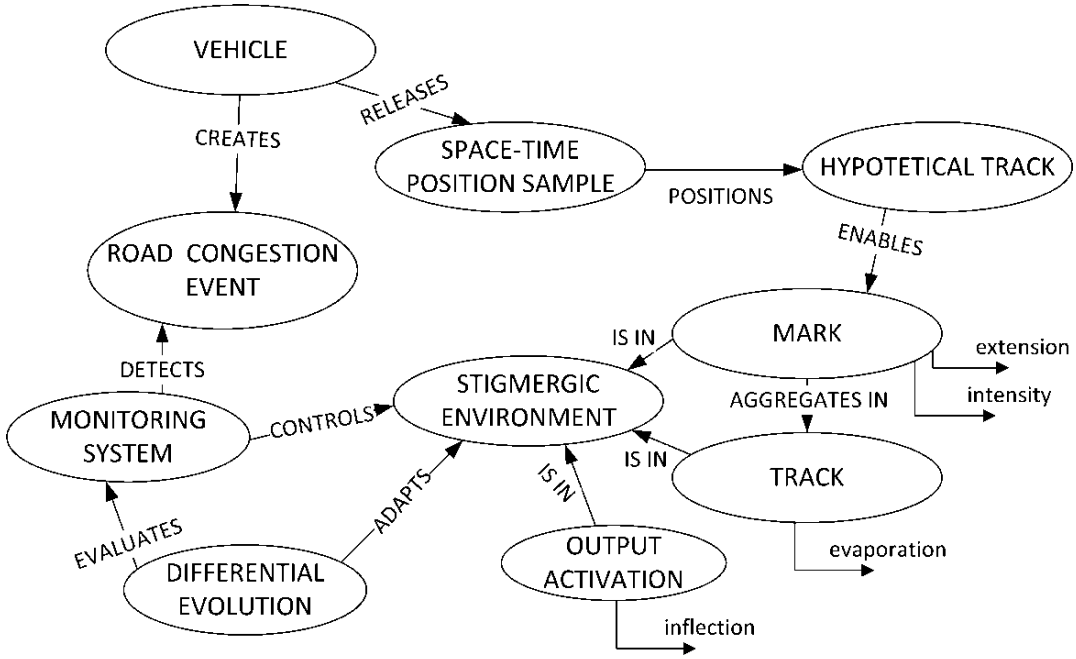

Fig. 1. Domain ontology of the proposed approach.

3.1 The problem statement

Let us model a given urban street network as a directed graph. Fig. 2 shows an example of the Pisa center urban street network (Italy). Here, two paths of the network are also shown. In the dynamic view of the system, each path can be modeled as a linear segment, because the position of each vehicle in the path can be measured by the on-road position from the initial point of the directed path.

The input of the monitoring system is made by periodical samples of the geo-position $g_{v,t}$, of each vehicle v at the time t in the given urban area. An occurred traffic congestion event E_k, is characterized by spatial and temporal coordinates, which correspond to congestion begin and end. Let us denote them as begin instant $\underline{t}_k$ and end instant $\overline{t}_k$. In each sampling instant $t \in [\underline{t}_k, \overline{t}_k]$, the on-road positions of the queue head and tail can be denoted as $\overline{s}_k^t$ and $\underline{s}_k^t$.

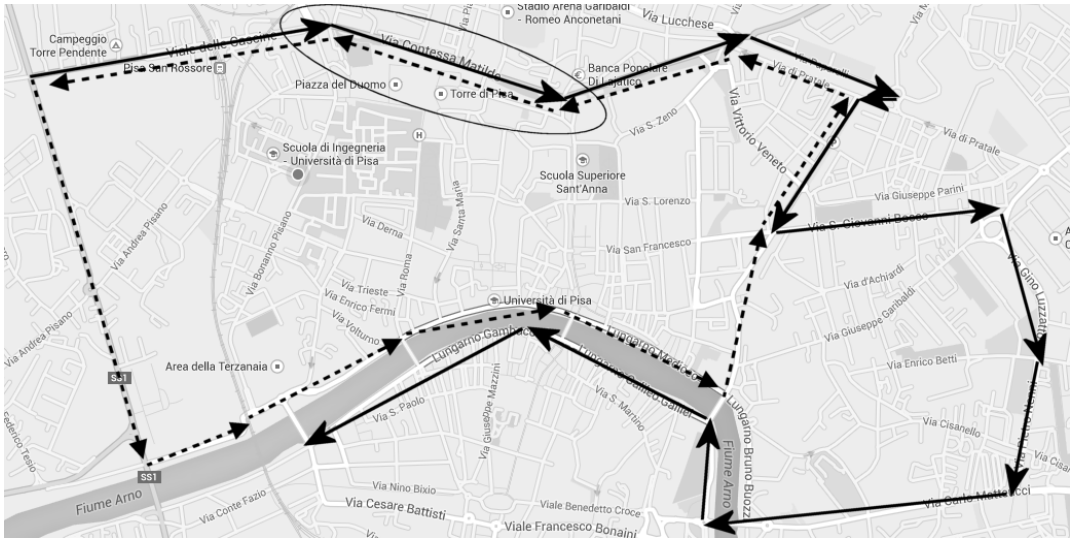

Fig. 2. The Pisa center urban area with two sample paths

With this characterization, the system output is made by a series of traffic congestion occurring *events*:

$$E_k^{DETECTED} \equiv \{[\underline{t}_k, \overline{t}_k], [\underline{s}_k^{\underline{t}}, \overline{s}_k^{\underline{t}}], ..., [\underline{s}_k^{\overline{t}}, \overline{s}_k^{\overline{t}}]\} \tag{1}$$

We measure the similarity between actual and detected events in order to design a fitness function which evaluate the system output quality. Real and detected event share the same representation format, but their values could be different because of detection error:

$$E_K^{ACTUAL} \equiv \{[\underline{\tau}_k, \overline{\tau}_k], [\underline{\sigma}_k^{\underline{\tau}}, \overline{\sigma}_k^{\underline{\tau}}], ..., [\underline{\sigma}_k^{\overline{\tau}}, \overline{\sigma}_k^{\overline{\tau}}]\} \tag{2}$$

Given the above definitions, a fitness function of the monitoring system is determined:

$$f_k = \frac{\left|\underline{\tau}_k - \underline{t}_k\right| + \left|\overline{\tau}_k - \overline{t}_k\right|}{\left|\overline{\tau}_k - \underline{\tau}_k\right|} + \sum_{\underline{i} \equiv \min(\underline{\tau}, \underline{t})}^{\overline{i} \equiv \max(\overline{\tau}, \overline{t})} \frac{\left|\underline{\sigma}_k^i - \underline{s}_k^i\right| + \left|\overline{\sigma}_k^i - \overline{s}_k^i\right|}{\left|[\underline{i}, \overline{i}]\right| \left|\overline{\sigma}_k^i - \underline{\sigma}_k^i\right|} \tag{3}$$

More precisely, Fig. 3 represents the main elements of the fitness function. The absolute differences between start and end times, normalized with respect to the time interval, is represented in the left addend, while the average absolute differences on the head and tail of

the queues, normalized with respect to the queues length and the number of samples is represented in the right addend.

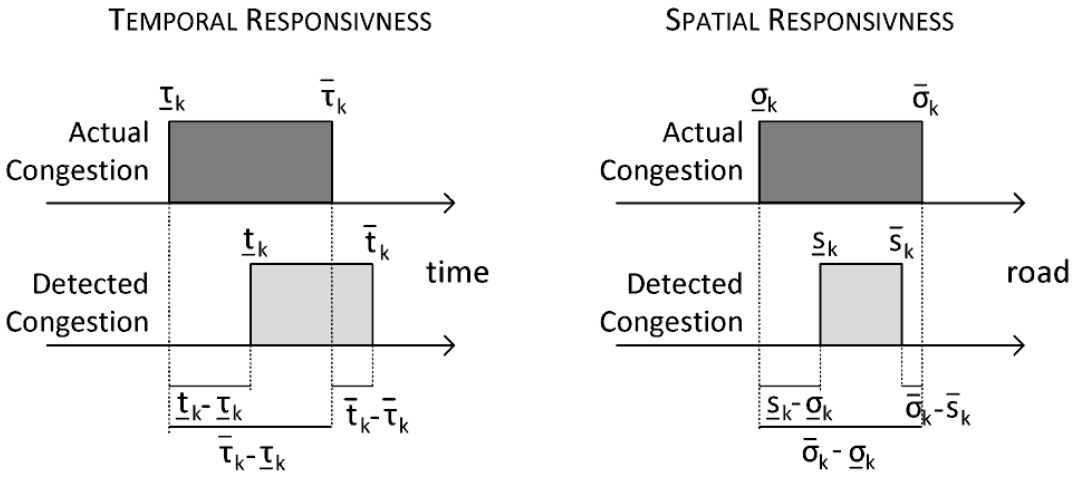

Fig. 3. Representation of the main elements of the fitness in Formula (3).

It is worth noting that f_k=0 for a perfectly detected event and that in general f_k is a positive real number. With this definition, the overall quality of the model is defined as the averaged fitness of all events:

$$Fit = \frac{1}{K} \cdot \sum_k f_k \tag{4}$$

Indeed the system may: (i) detect an event although a real counterpart does not occur (false positive); (ii) do not detect an actually occurred event (false negative). It follows that to find good match between actual and detected events corresponds to minimize *Fit*. The contributions of unmatched events are also entirely considered.

The overall problem is to detect all the traffic congestion events with the lowest fitness.

3.2 The input activation interface

The input activation interface aims to take vehicle positions $p_{v,t}$ (generated from $g_{v,t}$ via

projection on a linear path) and to establish whether they should be processed by the stigmergic layer or not. For this reason, we introduce the concept of *hypothetical track*, which is represented by an isosceles trapezoid placed on current vehicle position. If two hypothetical tracks generated by the same vehicle on two consecutive position samples overlap, then a mark is released in the stigmergic layer, and its intensity is proportional to the overlaps itself (5). In Fig. 4 a scenario of two overlapping hypothetical tracks centered on the vehicle positions is depicted. Here 1, β and 2β, are respectively height, upper and lower bases of the hypothetical track. Moreover, δ is the distance covered by the vehicle between $p_{v,t-1}$ and $p_{v,t}$. It can be demonstrated that, when the two hypothetical tracks overlap, the ordinate of the cross point of their diagonal edges, called intensity coefficient γ, is:

$$\gamma_{v,t} = \min\{1,\ 2-\delta/\beta\} \in [0,1] \tag{5}$$

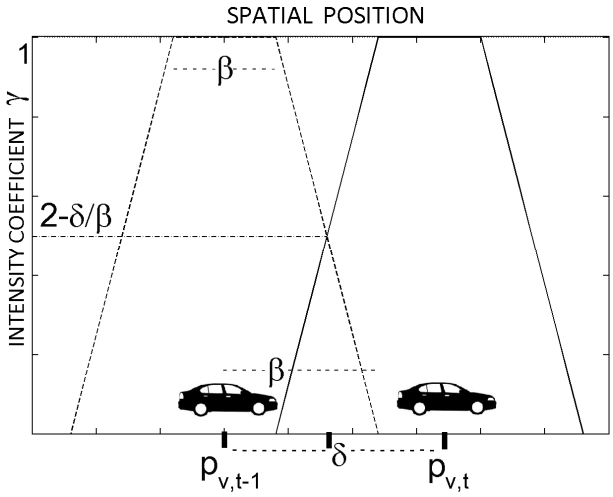

Fig. 4. A scenario of input activation interface with two hypothetical tracks.

The input activation interface activates the stigmergic layer and provides the pair ($p_{v,t}$, $\gamma_{v,t}$) when two consecutive hypothetical tracks overlap.

3.3 The stigmergic layer

Fig. 5 represents a mark (the overlying triangular shape), and the same mark after one step of evaporation (the underlying triangular shape). The mark is released by the vehicle v_1 in the path P_k, at the position $p_{v1,k}$, which is characterized by a central (maximum) *intensity* $\gamma_{v,t} \cdot I$, an *extension* $\varepsilon > 0$, and an *evaporation* $\theta \in [0,1]$. More precisely, θ corresponds to a percentage of the mark intensity that is kept step by step, until in practice the mark disappears. The decay time is longer than a marking step. Thus, if the vehicle is still, the new mark and the old one superimpose with each other, creating a track, and so on, up to reach a stationary level of the intensity. In contrast, if the vehicle speed is sufficiently high, the mark will not be reinforced and its intensity will decrease in time.

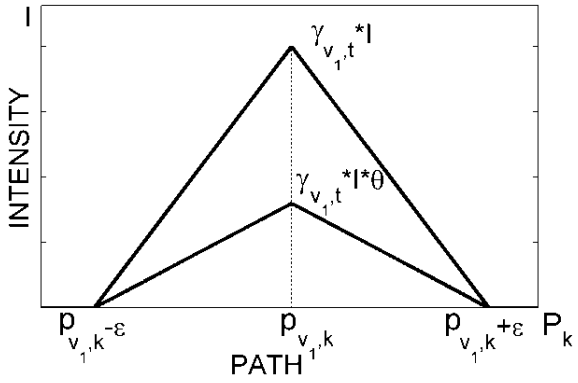

Fig. 5. A single mark released in the marking space (the overlying triangular shape) together with the same mark after a step of decay (the underlying triangular shape).

Similarly, marks generated by two vehicles sufficiently close will superimpose with each other. An example of track is shown in Fig. 6, by the overlying solid-line shape. It is generated by two vehicles releasing two marks, represented by the two underlying triangular shapes, at different instants of time. More precisely: vehicle v_1 released, at the previous time, t-1, a mark which accordingly evaporated by a *factor* θ, whereas vehicle v_2

released a mark at the current time, *t*, close to the mark of the vehicle v_1.

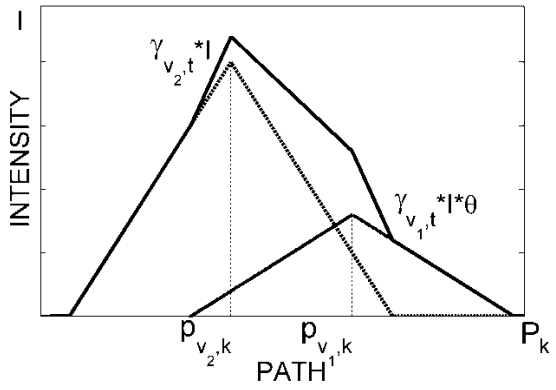

Fig. 6. Two marks released by two close vehicles (triangular shapes), with the corresponding track (overlying non-triangular shape).

It is apparent from Fig. 5 and Fig. 6 that mark extension and evaporation can serve as a means to detect mobility and proximity patterns of vehicles. In the context of road congestion detection, the overall purpose of the monitoring system is to control the mark aggregation process so as to maximize the intensity of marks when it is produced by a queue of vehicles, while minimizing such intensity when caused by other situations (e.g., a single and stationary parking vehicle, a short queue of vehicles at a traffic light). To this aim, the system does not represent explicitly the behavior of queues of vehicles. Its design is not a top-down process whereby a human observer abstracts and represents in a symbolic or statistic manner the situations of interest. Rather, it is tuned by an algorithm to achieve a certain level of adaptation to the local situation via examples of road congestion. Such adaptation is a bottom-up process: it consists in finding the right parameters at the micro-level (marks of a single vehicle) to produce a coherent emergent behavior at the macro-level (swarm of marks of multiple vehicles). To this aim, we will adopt an evolutionary computation technique, which improves the parameters with regard to a given measure of quality over the tuning examples. Evolutionary computation has the

advantage of making no assumptions about the problem being optimized, thus avoiding to bias the underlying emergent mechanism of aggregation. It is based on a population of candidate solutions, called agents, iteratively improved via operators inspired by natural evolution, such as inheritance, mutation, selection, and crossover. The adaptation process will be detailed in Section 4.

The next subsection is devoted to the problem of transferring the stigmergic potential achieved by the track intensity to a performance indicator with a clear interpretation, i.e., *stable*, meaning that it has to retain its identity in spite of some small fluctuations occurring in real-world data, and *distinguishable*, meaning that its identity should be distinct enough from each other.

3.4 The output activation interface

The information aggregation process based on stigmergy handles micro-fluctuations and leads to abstraction and emergence of high-level concepts. Significant phenomena of traffic congestion are better distinguished thanks to the output activation interface, which enhances the estimation of the congestion progressing levels. To this aim, a sigmoidal activation function is applied to the track intensity:

$$\Sigma(I_k) = 1/(1+e^{-\alpha(I_k-\phi)}) \tag{6}$$

In Fig. 7 an example of activation function with inflection point $\phi = 120$ and different values of α is shown. In essence, the activation function amplifies values of the intensity higher than ϕ, while decreases values lower than ϕ. As a consequence, major congestions are highlighted, while minor queues are hidden, and the micro fluctuations are smoothed. The sigmoidal inflection slope is determined by the value of the parameter α, which controls the width of the "gray zone", useful to deal with uncertainty in data: a high value makes the activation Boolean (suitable for stable events) whereas a low value enhances the

multi-class or “fuzzy” character of the output, which is useful to reduce information hiding when upper processing layer are available.

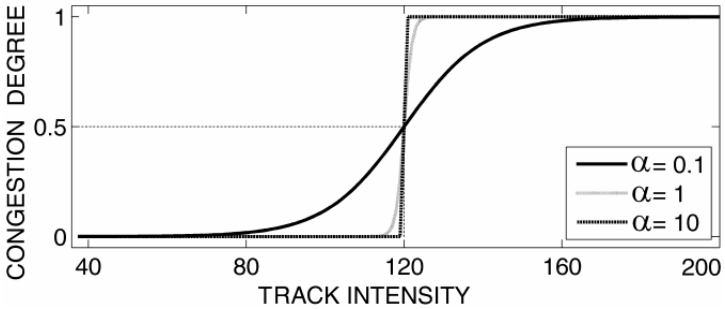

Fig. 7. Sigmoidal activation function with $\phi = 120$ and different values of α.

As an example, Fig. 8 shows vehicle positions, track intensity and congestion degree for the road highlighted with an oval in Fig. 2.

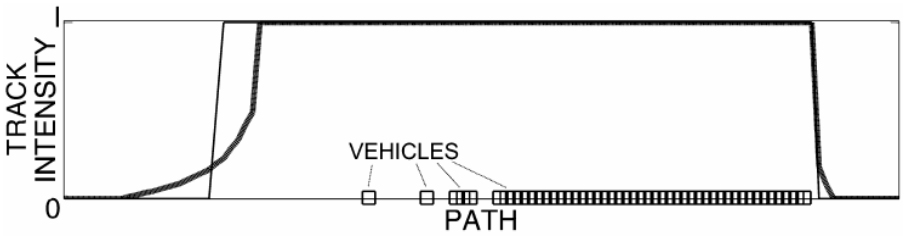

Fig. 8. Vehicles positions, track intensity (thick line), congestion degree (thin line), for the road highlighted with an oval in Fig. 1.

Overall, several structural parameters are involved in the swarm aggregation so far designed. Finding correct setting for such parameters is not trivial, since traffic flow and density vary with respect to the observed urban areas. Manual tuning is very time-consuming, human-intensive and error-prone. Furthermore, it depends on the intuition and experience, which are typically undocumented and therefore non-reproducible. Therefore, an automated parameter adaptation mechanism is needed [4]. Fig. 9 shows the overall

system architecture, inclusive of the adaptation module. In essence, at every instant considered the input activation interface takes as an input a set of vehicles' positions $\{P(t_k)\}$, and provides to the stigmergic layer a set of positions and intensities for related marks $\{M(t_k)\}$. The stigmergic layer provides a track, $T(t_k)$, to the output activation interface, which in turn provides a congestion degree, $C(t_k)$. This input-output flow is represented as a solid arrow in figure. In contrast the input-output of the adaptation subsystem is represented by a dotted arrow. More precisely, the adaptation is based on the evaluation of the fitness over a *tuning set*. In figure, the tuning set is denoted by asterisks: it is a sequence of (*input*, *desired output*) pairs, on the left side, together with a corresponding sequence of *actual output* values, on the right side. In a fitting solution, the desired and the actual output values corresponding to the same input are very close to each other. The next section is devoted to the adaptation subsystem.

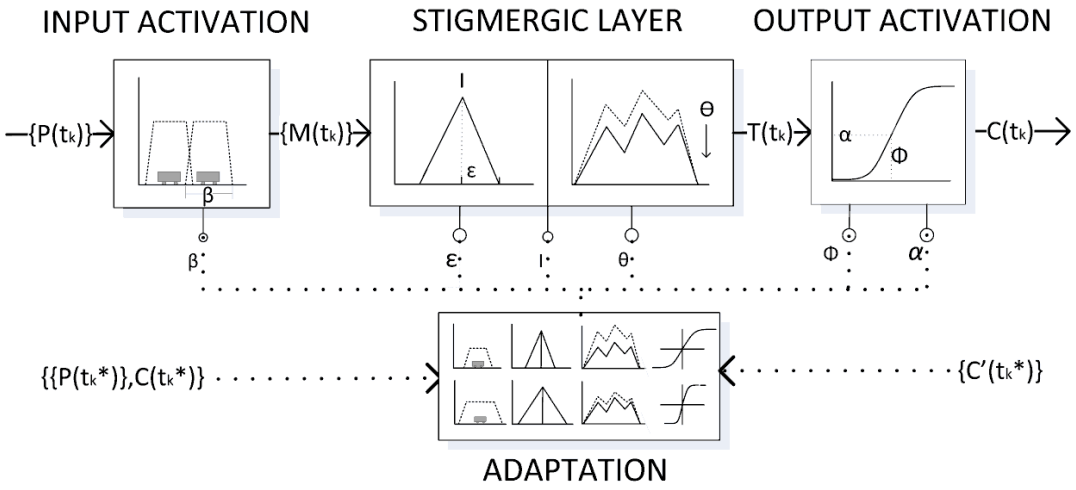

Fig. 9. Overall system architecture.

4 Analysis and design of the parameters adaptation

In this section, we first report on the role of each parameter in biasing the processing, and then we adopt a supervised data-driven parametric optimization based on DE.

In the literature, the approaches aimed to set up a group of parameters can be distinguished into *model-free* and *model-based* procedures [17]. Model-based procedures build a model upon the relation between the algorithm and the values of its parameters provided by human experience. On the other side, model-free procedures are faster in execution, but have no extrapolation potential (black-box approach). In general, the no-free-lunch theorem of optimization states that a general-purpose universal optimization strategy is impossible, and the only way one strategy can outperform another is by specializing it to the structure of the specific problem under consideration [18]. Specialization can be achieved applying constraints to the search space. A common solution to cope with a complex search space consists in applying a population-based method, such as Evolutionary Algorithms (EA) [17],[19]. We adopt a specific subclass of EA, namely DE. Since EAs are meta-heuristics, they have parameters to be tuned. However their effectiveness is already provided with default values.

4.1 Model Based Analysis

In Table 1, the structural parameters of the system are summarized. The *hypothetical track extension* (β) depends on statistics about specific road traffic. For instance, let us consider a road with a speed limit of 80 km/h and a sustainable average speed of 50 km/h = 833.2 meters/min. Two consecutive hypothetical tracks will not overlap if β = 416.6 meters while car speed is higher than 50 km/h. The *mark intensity* (I) represents the maximum intensity of the released mark. The intensity of the track generated from marks aggregation is directly influenced by value of I as well as the mark lifetime and the triggering of the output activation interface. For example, with an evaporation θ =0.4 and I=8, after 5 steps the mark intensity falls under 1, and then in practice disappears.

Table 1 Structural parameters of the system.

Parameter	Description	Section
$\beta > 0$	hypothetical track	3.2
$I > 0$	mark intensity	3.3
$\varepsilon > 0$	mark extension	3.3
$0 < \theta < 1$	mark evaporation	3.3
$\phi > 0$	inflection point	3.4
$\alpha > 0$	inflection slope	3.4

The *mark extension* (ε) is measured in unit and implies the distance within which marks interact with each other. The mark of a vehicle in a unit should interact with both the next and the previous occupied units. Thus, 1 unit = 10 meters can be used as lower bound. In addition, considering a 100 meters congestion, vehicle marks in the head and in the tail have to interact at most with the mark produced by the vehicle in the middle of the queue. A range of 5 units = 50 meters allows this interaction. Hence, taking ε between 1 and 5 will allow marks aggregation between close vehicles, while preventing interaction between vehicles too far from each other in any urban context, thus increasing the system error on the start and the end positions of a queue. The mark lifetime is strictly related to the *mark evaporation* (θ). Short-life marks prevent marks aggregation, whereas long-life marks lead to track saturation. Both behaviors increase system error: the former affects detection of the temporal start of the event, while the latter affects detection of the temporal end of the event. Therefore, considering a sampling period of 1 minute, mark lifetime should be higher than 2 minute ($\theta = 0.5$) and lower than 5 minutes ($\theta = 0.75$). The *inflection point* (ϕ) is in the domain of the mark intensity, and the inflection slope (α) is a multiplicative factor of the transient dynamics, and can be set according to the structural mark parameters. As shown in Fig. 7, the maximum value for α is 10. Indeed it provides an almost-Boolean transition.

4.2 The model-free parametric tuning

The parametric adaptation subsystem is based on the DE algorithm, which optimize the system parameters with respect to the fitness defined in (4). In our context, an *n*-dimensional vector represents a solution of the optimization problem delegated to DE, where *n* is the number of parameters to tune. At the begin of its execution, if solutions to inject are unavailable, DE generate randomly a population of *N* candidate solutions. For each step and member (target), some population members are selected for mutation, obtaining a mutant vector, and a trial vector is generated applying the crossover to the mutant vector and the target. Finally, the target is replaced by the trial if the latter provides a lower fitness value.

Different population sizes are suggested in the literature [21]. Generally, a larger population size corresponds to a higher probability to find a global optimum. On the contrary, a smaller population size increases the convergence rate, and reduces the number of needed function evaluations. Smaller populations are suitable to separable and unimodal fitness functions, while larger populations are appropriate to multi-modal function in order to avoid premature convergence. Population size can vary in a range of [2*n,* 40*n*]. Based on studies in [22] we set the population size to 20 members. The scaling factor $F \in [0,2]$ mediates the generation of the mutant vector. *F* is usually set in [0.4-1) with an initial value in [0.5-0.9] [23]. To choose the best value for our application, we performed trials with $F \in \{0.4, 0.8, 1.2, 1.6, 2.0\}$. About the crossover probability, larger values generate a vector which is more similar to the mutant vector, while the opposite favors the target vector. In general, large *CR* speeds up the convergence. A good value for *CR* is between 0.2 and 0.9 [21]. To choose the best value, we performed trials with $CR \in \{0.2, 0.4, 0.6, 0.8\}$ to compare the results.

Many variants of the DE algorithm have been designed, by combining different structure

and parameterization of mutation and crossover operators [23],[24]. Any specific DE strategy is formally described as *DE / x / y / z*, where:

- *x* defines the base choice (v_1) of the mutant vector (v_{mutant}) between:
  - *rand*, random vector, which explores more, but requires more generations to converge;
  - *best*, the best population individual, which converges faster, but risks to be trapped in local minima;
  - *rand-to-best*, a combination of the above strategies (weighted sum of *F*).
- *y* is the number of differences in mutation carriers:
  - *1*: $v_{mutant} = v_1 + F \cdot (v_2 - v_3)$
  - *2*: $v_{mutant} = v_1 + F \cdot (v_2 + v_3 - v_4 - v_5)$, where v_2 , ..., v_5 are always random
- *z* is the type of crossover:
  - *bin* (binomial), in which *CR* is the probability that an element of the vector is taken from the target or from the mutant vector;
  - *exp* (exponential), in which, starting from a random element of the vector, the mutation proceeds sequentially in a circular manner. It stops with probability *CR* after each item, or if you changed all the elements;

Clearly DE variants vary in the terms of vector choice, recombination operator used and also in the way in which the mutation is computed. Our aim is to identify which variant of them is more suitable with respect to our application. The next Section presents the experimental studies.

5 Experimental studies

To prove the effectiveness of the proposed approach we developed a Java-based system architecture. More specifically, the stigmergic environment and the adaptation subsystem have been developed under the Repast[1] and the Matlab[2] frameworks, respectively. A traffic simulator based on Java and the Google Maps API has been developed to feed the system. To generate traffic data, as a pilot urban area we considered about 8 km of the network of Fig. 2. In two hours of simulation, 116 congestion events occurred.

For the setting of *CR* and *F*, and the comparison between strategies, we took into account the model-based analysis of Section 4.2 [22], thus using human experience. We refer to this approach as the "HU+DE". Differently from "HU+DE", we relaxed the constraints earlier identified in order to better investigate the behavior of DE. In fact, relaxing the constraints increase the search space, i.e., the difficulty of the task. This is important also to distinguish between the performances of each setting. More exactly, increasing the maximum mark extension allows the marks more spatial interaction; enlarging the evaporation constraints permits temporal flexibility; we consequently adjusted the inflection point according to Section 4.2; finally we relaxed the slope to permit the system resiliency to previous changes. To sum up, we constrained the parameters to the following parameters bounds: $\varepsilon \in$ [1, 10], $\theta \in$ [0.35, 0.9], $\phi \in$ [1, 500], $\alpha \in$ (0, 20]. We refer to this approach as "DE". We ran DE for 30 generations, and for each setting we repeated the experiment for 5 times. We also determined that the resulting fitness values are well-modeled by a normal distribution, using a graphical normality test. Hence, we calculated the 95% confidence intervals.

Table 2 shows the fitness, in the form "mean ± confidence interval", for each strategy, together with the considered values of the parameters *CR* and *F*. We remark that the confidence interval is slightly wider than in [22], because we considered a larger search

[1] http://repast.sourceforge.net/
[2] http://www.mathworks.com

space for DE. DE achieves the best performance in term of fitness with a greater population (N=20) independently of the strategy, CR and F. However, the time to compute a solution increases almost linearly with the population: respectively to set N=10, N=15 needs 50% more time, and to set N=20 100% more.

Table 2 Settings of the optimization parameters: (a) DE/1/best/bin, (b) DE/1/rand/bin, and (c) DE/1/rand-to-best/bin

N=20		CR			
		0.2	**0.4**	**0.6**	**0.8**
	0.4	37.11±1.66	36.24±1.32	35.26±1.13	34.98±1.24
	0.8	40.71±3.35	36.78±0.79	37.17±1.54	**34.89±1.04**
F	**1.2**	46.27±2.09	44.36±2.65	38.75±2.37	37.86±1.27
	1.6	43.66±7.05	43.36±4.07	42.09±2.10	39.97±2.99
	2.0	43.35±6.31	43.83±3.16	46.61±6.27	43.38±2.50

(a)

N=20		CR			
		0.2	**0.4**	**0.6**	**0.8**
	0.4	40.81±1.92	38.43±1.61	37.22±1.18	**36.73±1.00**
	0.8	44.02±5.14	40.76±2.50	38.29±1.95	37.05±2.31
F	**1.2**	45.25±4.40	41.38±1.68	40.84±3.97	40.22±3.68
	1.6	43.72±4.75	42.94±4.13	44.69±5.31	42.56±2.12
	2.0	45.39±6.29	45.86±6.12	46.86±2.58	42.81±3.67

(b)

N=20		CR			
		0.2	**0.4**	**0.6**	**0.8**
	0.4	40.20±4.12	35.94±0.90	37.01±1.68	**35.49±1.03**
	0.8	38.03±1.09	38.20±1.04	35.98±0.89	35.55±1.23
F	**1.2**	41.30±2.94	39.69±1.60	37.70±2.03	39.46±0.95
	1.6	49.41±6.99	41.77±5.60	40.75±2.49	41.35±4.61
	2.0	53.58±4.71	46.22±3.29	48.48±9.30	47.73±4.91

(c)

In Table 2, we present the results with the used combinations of CR and F, and the three evaluated strategies. In all strategies, DE performance improves for higher CR and lower F. When CR is low (0.2 and 0.4), very few elements of the mutant vector enter the trial vector. This implies the trial vector to be very similar to the target vector (which is already a member of the population). Therefore the crossover is pretty inefficient. The scaling factor

F seems to affect negatively the performance of DE when higher than 1 (1.2, 1.6, and 2.0). The mutation process with lower values of *F* performs small modifications of the mutant vector, especially with the DE/1/best/bin, and this positively affects the performance of DE. To sum up, DE operates very well with high *CR* (0.6 and 0.8) and low *F* (0.4 and 0.8). With this setting, there is a small mutation of the mutant vector, but it is more likely that during the crossover an element of the trial is picked from the mutant than the target vector. In general the DE/1/best/bin strategy performs better than both the DE/1/rand/bin and DE/1/rand-to-best/bin. For values of $F = 1.6$ and 2.0, the strategy DE/1/rand-to-best/bin has the lowest performance, while for lower values of *F* (0.4 and 0.6) is better than DE/1/rand/bin and almost as good as DE/1/best/bin. Finally, we repeated the experiment for all the strategies with the promising combination of $CR=0.9$ and $F=0.2$. However, no improvement of the performance has been detected.

Table 3 presents the optimal parameters setting and the related best fitness of each experiment. To provide an absolute quality measure, the table also shows separately the average absolute errors on the start/end times, and the average absolute errors on the head/tail of the queues. It can be observed that the model-free approach ("DE") significantly improved the quality of the detection with respect to the model-based ("HU"), from 65.5 to 33.8, and that the hybrid approach ("HU+DE") provided further improvements. The experiments confirm the solution found by [22] with the model-based approach to be the one with the best fitness value. In fact, most of the final solutions identified by "DE" converge close to that same value, but without any improvement. The best solution has been identified by DE/1/best/bin with $CR = 0.8$ and $F = 0.8$. The fitness is 33.8, which is very close to the fitness value of 33.6 provided in [22]. It is worth noting that the two solutions differ from the structural parameters of the mark. The "DE" solution has wider mark and lower evaporation. This implies that marks interact at greater distance but for shorter time.

Due to that, the system is able to better identify the spatial extension of the traffic congestion (average error of 17.3 meters) at the expense of the temporal duration (average error of 20.8 minutes). Table 3 also shows that the major impact of the parametric adaptation by [22] is on the temporal error (from 28.4 to 9.5 minutes), whereas the impact on the spatial error is not significant (about one meter).

Table 3 Human-driven vs. DE-driven parameterization

Approach	Parameterization				Performance		
	ε	θ	α	ϕ	*Fit*	*Avg Time Err (min.)*	*Avg Position Err (mt.)*
HU+DE	4	0.579	8.5	25	33.6	9.5	36.4
DE	10	0.469	13.8	82	33.8	20.8	17.3
HU	3	0.675	1	23	65.5	28.4	35.7

Fig. 10 shows the fitness versus the number of generations for the three strategies: DE/1/best/bin with CR = 0.8 and F = 0.8; DE/1/rand/bin with CR = 0.8 and F = 0.4; DE/1/rand-to-best/bin with CR = 0.8 and F = 0.4. We observe that for all the strategies the fitness function gets stable under a value of 40 after a small number of generations (about 15). It is worth noting that DE/1/best/bin improves the solution with subsequent drops and plateaus (generation 7 and 13) of the fitness, and finally small adjustments are made to the best member; differently, the DE/1/rand/bin, has a softer decrease of the fitness with small improvements over all the generations; the hybrid strategy DE/1/rand-to-best/bin shows both patterns: drops and plateaus occurred in the first generations (6 and 9), and then the fitness slowly decreases for the remaining generations.

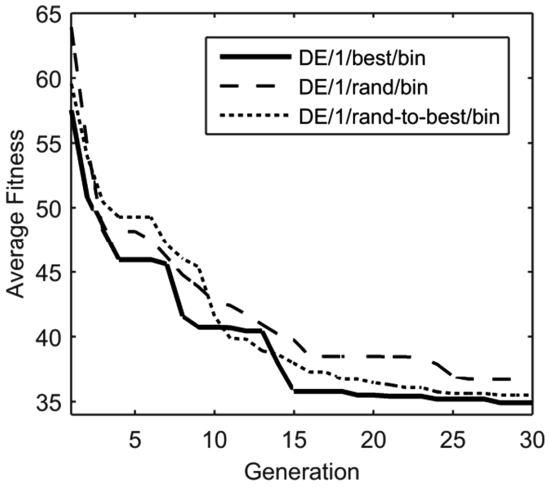

Fig. 10. Fitness function versus generation, for DE/1/best/bin, DE/1/rand/bin and DE/1/rand-to-best/bin strategies.

6 Conclusions

This paper proposes a novel approach for traffic congestion estimation, based on swarm aggregation of vehicle positions. At the core of the system is the marker-based stigmergy, properly interfaced with two input and output activation mechanisms, and adapted to different application contexts. For this purpose, we designed a fitness function and adopted the differential evolution as an optimization strategy. We explored DE with three strategies and different combination of both the crossover rate and the differential weight. The experiments shows that lower values of the differential weight ($F \leq 0.8$) and higher values of the crossover rate ($CR \geq 0.6$) produce better solutions. This parameterization has been effective with all the three strategies: *best*, *rand* and *rand-to-best*. However, the *best* strategy performs better and produces solutions with lower fitness than *rand* and *rand-to-best* strategies. In comparison with [22], the best solution has better accuracy in the spatial detection of the traffic congestion, at the expense of the temporal accuracy. This result indicates that the parameterization of the system can be oriented for different purposes,

according to the user needs. As a future work, we plan to reduce the human knowledge in the parametric adaptation, i.e., to relax the constraints of the search space.

Acknowledgements

This research has been partially supported in the research project entitled “Smarty” (SMARt Transport for sustainable city), which has been co-financed by the Tuscany Region (Italy).

References

[1] S. Pellicer, G. Santa, A.L. Bleda, R. Maestre, A.J. Jara, A. Gomez Skarmeta, "A Global Perspective of Smart Cities: A Survey," Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), Seventh International Conference on. IEEE pp.439-444, 3-5 July, 2013.

[2] V. H. Parunak, "A survey of environments and mechanisms for human-human stigmergy," in Environments for Multi-Agent Systems II, Springer Berlin Heidelberg, pp. 163-186, 2006.

[3] M. G. C. A. Cimino, A. Lazzeri, G. Vaglini, “Improving the Analysis of Context-Aware Information via Marker-Based Stigmergy and Differential Evolution”, 14th International Conference on Artificial Intelligence and Soft Computing (ICAISC), pp.341-352, 14-18 June, 2015.

[4] A. Ciaramella, M.G.C.A. Cimino, B. Lazzerini, F. Marcelloni, "Using Context History to Personalize a Resource Recommender via a Genetic Algorithm", in Proceeding of the International Conference on Intelligent Systems Design and Applications, ISDA’10, IEEE, pp. 965-970, 2010.

[5] A. Tabibiazar, B. Otman, "Kernel-based modeling and optimization for density estimation in transportation systems using floating car data," Intelligent Transportation Systems (ITSC), 14th International IEEE Conference, pp. 576-581, 2011.

[6] Q.J. Kong, Y. Chen, Y. Liu, "A fusion-based system for road-network traffic state surveillance: a case study of Shanghai," Intelligent Transportation Systems Magazine, vol. 1(1), pp. 37-42, 2009.

[7] Q.J. Kong, Q. Zhao, C. Wei, Y. Liu, "Efficient traffic state estimation for large-scale urban road networks," Intelligent Transportation Systems, IEEE Transactions, vol. 14(1), pp. 398-407, 2013.

[8] Y. Chen, L. Gao, Z. Li, Y. Liu, "A new method for urban traffic state estimation based on vehicle tracking algorithm," Intelligent Transportation Systems Conference (ITSC) IEEE, pp. 1097-1101, 2007.

[9] Q. Zhao, Q.J. Kong, Y. Liu, "Sample size analysis of GPS probe vehicles for urban traffic state estimation," Intelligent Transportation Systems (ITSC), 14th International IEEE Conference, pp. 272-276, 2011.

[10] J. Yoon, B. Noble, M. Liu, "Surface street traffic estimation," Proceedings of the 5th international conference on Mobile systems, applications and services, ACM, p. 220-232, 2007.

[11] M., Dorigo, M. Birattari, S. Garnier, H.Hamann, M.M. de Oca, C. Solnon, T. Stützle, Swarm Intelligence, Springer International Publishing, Proceedings of the 9th International Conference, ANTS 2014, Brussels, Belgium, September 10-12, 2014.

[12] S. Kurihara, H. Tamak, M. Numao, J. Yano, K. Kagawa, T. Morita, "Traffic congestion forecasting based on pheromone communication model for intelligent transport systems," in Proceedings of the 11th Congress on Evolutionary Computation, IEEE, pp. 2879-2884, 2009.

[13] P. Bedi, N. Mediratta, S. Dhand, R. Sharma, A. Singhal, "Avoiding traffic jam using ant colony optimization-a novel approach." Conference on Computational Intelligence and Multimedia Applications, 2007. International Conference on. Vol. 1. IEEE, 2007.

[14] F. Caselli, A. Bonfietti, M. Milano. "Swarm-Based Controller for Traffic Lights Management." Congress of the Italian Association for Artificial Intelligence. Springer International Publishing, 2015.

[15] I. Takayuki, R. Kanamori, J. Takahashi, I. M. Maestre, E. de la Hoz. "The comparison of stigmergy strategies for decentralized traffic congestion control: Preliminary results." Pacific Rim International Conference on Artificial Intelligence. Springer Berlin Heidelberg, 2012.

[16] W. Narzt, U. Wilflingseder, G. Pomberger, D. Kolb, H. Hörtner. "Self-organising congestion evasion strategies using ant-based pheromones." IET Intelligent Transport Systems 4.1 (2010): 93.

[17] F. Dobslaw, "Recent Development in Automatic Parameter Tuning for Metaheuristics," in Proceedings of the 19th Annual Conference of Doctoral Students - WDS 2010, pp.54-63, 2010.

[18] Y.-C. Ho, D. L. Pepyne. "Simple explanation of the no free lunch theorem of optimization," Cybernetics and Systems Analysis, vol. 38(2), pp. 292-298, 2002.

[19] P. Pellegrini, T. Stützle, M. Birattari, "A critical analysis of parameter adaptation in ant colony optimization", Swarm Intelligence, vol. 6(1), pp. 23-48, 2012.

[20] M. Avvenuti, D. Cesarini, M.G.C.A. Cimino, "MARS, a Multi-Agent System for Assessing Rowers' Coordination via Motion-Based Stigmergy," Sensors, vol. 13, pp. 12218-12243, 2013.

[21] R. Mallipeddi, P.N. Suganthan, Q.K. Pan, M.F.Tasgetiren, "Differential evolution algorithm with ensemble of parameters and mutation strategies," Applied Soft Computing, vol. 11(2), pp.1679-1696, 2011.

[22] M.G.C.A. Cimino, A. Lazzeri, G. Vaglini. "Enabling swarm aggregation of position data via adaptive stigmergy: a case study in urban traffic flows." Information, Intelligence, Systems and Applications (IISA), 2015 6th International Conference on. IEEE, 2015.

[23] E. Mezura-Montes, J. Velázquez-Reyes, C.A. Coello, "A comparative study of differential evolution variants for global optimization," Proceedings of the 8th annual conference on Genetic and evolutionary computation, ACM, pp.485-482, 2006.

[24] D. Zaharie, "A comparative analysis of crossover variants in differential evolution," Proceedings of IMCSIT 2007, 2nd International Symposium Advances in Artificial Intelligence and Applications, pp. 171-181, 2007.