

Enabling swarm aggregation of position data via adaptive stigmergy: a case study in urban traffic flows

Mario G. C. A. Cimino, Alessandro Lazzeri, Gigliola Vaglini

Department of Information Engineering

University of Pisa

Pisa, Italy

mario.cimino@unipi.it, alessandro.lazzeri@for.unipi.it, gigliola.vaglini@unipi.it

Abstract—Urban road congestion estimation is a challenge in traffic management. City traffic state can vary temporally and spatially between road links, depending on crossroads and lanes. In addition, congestion estimation requires some sort of tuning to “what is around” to trigger appropriate reactions. An adaptive aggregation mechanism of position data is therefore crucial for traffic control. We present a biologically-inspired technique to aggregate position samples coming from on-vehicle devices. In essence, each vehicle position sample is spatially and temporally augmented with digital mark, locally deposited and evaporated. As a consequence, a marks concentration appears and stays spontaneously while many stationary vehicles and high density roads occur. Marks concentration is then sharpened to achieve a better distinction of critical phenomena to be triggered as detected traffic events. The overall mechanism can be actually enabled if structural parameters are correctly tuned for the given application context. Determining such correct parameters is not a simple task since different urban areas have different traffic flux and density. Thus, an appropriate tuning to adapt parameters to the specific urban area is desirable to make the estimation effective. In this paper, we show how this objective can be achieved by using differential evolution.

Keywords—urban traffic estimation; swarm intelligence; stigmergy; parametric adaptation; differential evolution

I. INTRODUCTION AND MOTIVATION

To reduce traffic congestion is one of the main issues in the Smart City strategy [1]. Recently more and more advanced sensor techniques for traffic measurement have attracted a number of researchers. The available technology can be basically grouped into two categories: on-vehicle devices and roadside infrastructures. The former mainly employs the Global Positioning System (GPS) and its variants to report real-time information on a vehicle position, whereas the latter typically involves dedicated equipment, such as loop detectors and cameras. Actually, any use of roadside infrastructures is essentially constrained to highways, freeways and primary arteries, and then cannot be used for urban traffic estimation. For this reason, the on-vehicle positioning device is considered as the reference data source in this paper. Hence, the availability of vehicle position data is supposed to be a requirement in our approach.

In this paper we present a new design of swarm aggregation

of vehicle positions based on marker-based stigmergy [2]. We use marker-based stigmergy as a computing paradigm for exploiting both spatial and temporal dynamics that characterize urban traffic. Basically, in our approach each vehicle of a monitored urban network releases periodical *marks* in a computational environment, according to its position. Marks may aggregate in the environment reinforcing in strength, whereas lose intensity evaporating over time. As a consequence a marks concentration, called *track* hereafter, appears and stays spontaneously while many stationary vehicles and high density roads occur. For a better scalability and a better distinction of unfolding congestion events, we adopt activation interfaces at the input-output of the stigmergic layer. Here, the term “activation” is taken from neural sciences and it is related to the requirement that a signal must reach a certain level before a processing layer fires to the next layer.

The proposed mechanism works if structural parameters, as mark extension, for example, are correctly tuned for the given application context [3]. Determining such correct parameters is not a simple task since different urban areas have different traffic flux and density. For this purpose, we adopt a tuning mechanism based on differential evolution for adapting parameters to the specific urban area.

The paper includes the problem statement and its formal characterization, as well as the proposed solving approach and experimental settings. More specifically, Section II focuses on related work. Section III provides the problem statement and discusses the design of the aggregation subsystem. Section IV is devoted to the design of the parametric adaptation. Experimental studies are detailed in Section V. Section VI covers conclusion and future works.

II. RELATED WORK

A number of methods to estimate traffic state on the basis of both categories have been developed for urban traffic. In [4] a kernel-based density estimation method based on probe-vehicle data has been proposed. 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 to extract the weights of Gaussian kernels in the estimated distribution function. The approximation density function by optimized

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).

kernels' weights can be used to estimate the mobile vehicles density in a specific time and space. In [5] a GPS and GIS integrated system for urban traffic flow analysis is presented. The authors use loop detectors and taxi as probe vehicles to estimate mean speed in road segment, then they integrate the two kinds of data using Federated Kalman Filter and D-S Evidence Theory. Moreover, they proposed a curve-fitting method to analyze GPS data for the mean-speed estimation in the urban road network. The least-square method is used to fit the data. In [6] a curve-fitting method and a vehicle tracking method have been compared for estimating spatiotemporal average velocity, i.e., the mean speed of vehicles on a road segment during a period of time. Result showed that the tracking-based method usually bears higher estimate accuracy but slower operation speed compared with the model-fitting method. In [7] a vehicle tracking method is proposed. Authors use the A* algorithm to determine the track of a vehicle between two following GPS samples. Then, the average velocity of each road segment is obtained by combining the velocities of the tracks that pass through the road segment. In [8] authors proposed an algorithm to calculate the minimum sample size for curve fitting methods. In the algorithm, the road type, the length of road section, and sample frequency are taken into account. The result showed that the errors of the estimated speeds become smaller with increasing number of the GPS samples. In [9] GPS data has been used to classify road conditions as "good" or "bad," both spatially and temporally. The first reflects the steadiness, the latter the speed of traffic. They take GPS samples and calculate the car delay distribution over the road segment. This "cumulative time-location" data is converted to spatiotemporal data and then classified using threshold-based quadrant clustering. Authors confront quadrant classifier with maximum likelihood and maximum a priori classifiers.

From the architectural standpoint, the deployment of traffic management system is characterized by huge volumes of data ("Big data") that need to be timely analyzed for detecting unfolding congestion. A promising architecture is the so-called multi-agents-system (MAS), a decentralized environment made by sub-systems each operating with partial autonomy and local awareness. According to the biologically-inspired principles of Swarm Intelligence, MASs should be designed to manifest self-organization and complex behaviors, although the individual behavior of each agent is simple [10]. The coordination mechanisms of MAS can generally be divided into two types: direct and indirect. The former is less scalable due to the overload of coordination. In the literature, the coordination for the latter is usually called *stigmergy*. With stigmergy, the influence in the environment left by the behavior of one agent stimulates the performance of a subsequent action of this agent or a different agent. The stigmergic mechanism can work with massive numbers of agents. For instance, in [11] stigmergy is used for traffic congestion forecast, where fixed on-road sensors measure vehicle flow and use traffic-density pheromone to predict congestions.

III. THE DESIGN OF THE STIGMERGIC AGGREGATION

The architecture of the proposed aggregation technique consists in three sub-systems. This section is devoted to the problem statement and to the specification of each sub-system.

A. Problem statement

Given an urban street network, which can be modeled as a *directed graph*, let us consider the least number of paths such that every link belongs to at least one path. As an example, Fig.1 shows a static view with two paths of the Pisa center urban street network (Italy). 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.

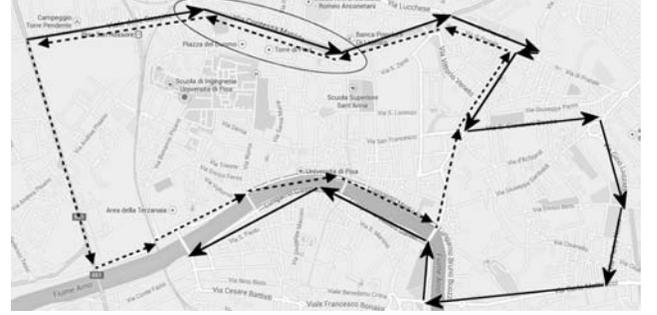


Fig. 1. The Pisa center urban area considered for pilot experiments, with two sample paths.

Let us assume that the geo-position $g_{v,t}$ of each vehicle v at the time t in a given urban area is periodically sampled and provided as input to the system. When a traffic congestion event E_k occurs, this is characterized by temporal begin and end, at the instant \underline{t}_k and \bar{t}_k , respectively. In addition, at each sampling instant $t \in [\underline{t}_k, \bar{t}_k]$, the on-road positions of the queue head and tail \bar{s}_k^t and \underline{s}_k^t are periodically detected. In conclusion, the output of the system is characterized by a series of traffic congestion occurring *events*:

$$E_k^{DETECTED} \equiv \{[\underline{t}_k, \bar{t}_k], [\underline{s}_k^{\underline{t}_k}, \bar{s}_k^{\underline{t}_k}], \dots, [\underline{s}_k^{\bar{t}_k}, \bar{s}_k^{\bar{t}_k}]\} \quad (1)$$

To model the quality of the output, let us distinguish between the actual and the detected (estimated) congestion event. The real event is characterized by the same format but slightly different values, since the detection process is never perfect:

$$E_k^{ACTUAL} \equiv \{[\underline{\tau}_k, \bar{\tau}_k], [\underline{\sigma}_k^{\underline{\tau}_k}, \bar{\sigma}_k^{\underline{\tau}_k}], \dots, [\underline{\sigma}_k^{\bar{\tau}_k}, \bar{\sigma}_k^{\bar{\tau}_k}]\} \quad (2)$$

Given the above definitions, a fitness function of the system can be also determined, to evaluate how the system approximates the k -th event:

$$f_k = \frac{|\underline{\tau}_k - \underline{t}_k| + |\bar{\tau}_k - \bar{t}_k|}{|\bar{\tau}_k - \underline{\tau}_k|} + \sum_{i=\min(\underline{\tau}, \underline{t})}^{\bar{i}=\max(\bar{\tau}, \bar{t})} \frac{|\sigma_k^i - s_k^i| + |\bar{\sigma}_k^i - \bar{s}_k^i|}{|[\underline{i}, \bar{i}]| |\bar{\sigma}_k^i - \underline{\sigma}_k^i|} \quad (3)$$

where the left addend represents the absolute differences on the start and end times, normalized with respect to the time interval, whereas the right addend represents the average absolute differences on the head and tail of the queues, normalized with respect to the queues length and the number of samples. It is worth noting that $f_k=0$ for a perfectly detected event and that in general f_k is a positive real number. The overall quality of the model is then measured by the average fitness on all events:

$$Fit = 1 / K \cdot \sum_k f_k \quad (4)$$

Actually the system may generate: (i) detected events which do not correspond to actual events (false positive); (ii) undetected actual events (false negative). To find the match between actual and detected events we choose at each step the pair that minimizes Fit . Unmatched events are also entirely considered as contribution.

Finally, given the above definitions, the problem is to detect all the traffic congestion events with the lowest fitness.

B. The input activation interface

The role of the input activation interface is to take vehicle positions $p_{v,t}$ (generated from $g_{v,t}$ via projection on a linear path) to establish whether they should be processed by the stigmergic layer or not and the marks intensity. For this purpose, the interface adopts the concept of *hypothetical track*, placed on the vehicle position to assess whether two hypothetical tracks generated on two consecutive positions of the vehicle overlap, i.e., whether a mark can be released. Fig.2 shows a scenario of two overlapping hypothetical tracks centered on the vehicle positions, with the form of isosceles trapezoid, whose height, upper and lower bases are 1, β and 2β , respectively. Here, the vehicle covered the distance δ 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:

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

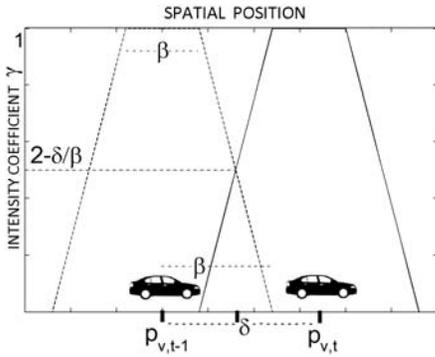


Fig. 2. A scenario of the input activation interface with hypothetical tracks.

The input activation interface works according to the following rule: when two consecutive hypothetical tracks overlap, then the stigmergic layer is activated and supplied with the pair $(p_{v,t}, \gamma_{v,t})$.

C. The stigmergic layer

Fig. 3 shows a mark (with solid line) released by the vehicle v_1 in the path P_k , at the position $p_{v_1,k}$, which is characterized by a central (maximum) intensity $\gamma_{v_1,t}$, an extension $\varepsilon > 0$, and an evaporation $\theta \in [0,1]$. Fig. 3 shows, with a gray line, the same mark after a step of marking. More precisely, θ corresponds to a proportion of the intensity of the previous step. Hence, after a certain decay time, the single mark in practice disappears. The decay time is longer than the step of marking. Thus if the vehicle is still, a new mark superimpose on the old marks, creating a track, whose intensity will reach a stationary level. In contrast, if the vehicle speed is sufficiently high, the mark intensities will decrease with time without being reinforced.

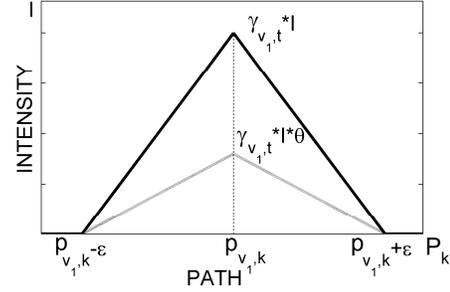


Fig. 3. A single mark released in the marking space (solid line), together with the same mark after a step of decay (gray line).

Similarly, two vehicles sufficiently close and still superimpose their marks. Fig. 4 shows an example of track (the overlying non-triangular shape) generated by two vehicles releasing marks (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 .

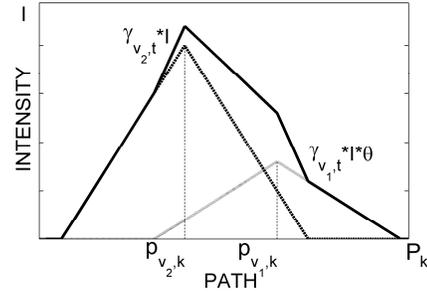


Fig. 4. Two marks released by two close vehicles (triangular shapes) with the corresponding track (overlying non-triangular shape).

It can be realized from Fig. 3 and Fig. 4 that mark extension and evaporation take into account the mobility and the proximity of vehicles, and that the track is a kind of short term memory of the activities on a path. It is worth nothing that a single and stationary vehicle (e.g. parking) can produce a stationary mark with poor intensity with respect to a queue of vehicles, which in contrast can lead to a high aggregation of marks.

D. The output activation interface

The process of information aggregation leads to abstraction and emergence of high-level concepts beyond occurring micro-fluctuations. The output activation interface allows achieving a better distinction of the critical phenomena during unfolding traffic congestion, with a better detection of the progressing levels. For this purpose we apply a sigmoidal activation function to the track intensity:

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

Fig. 5 shows the activation function with inflection point $\phi = 120$ and different values of α . As an effect of the activation, values of the intensity higher than ϕ are further amplified to evidence major congestion effects, whereas values lower than ϕ are further decreased to hide minor queues. In both cases, micro fluctuations are smoothed. The parameter α controls the inflection slope, and then the width of the “gray zone”, 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. As an example, Fig. 6 shows vehicle positions, track intensity and congestion degree for the road highlighted with an oval in Fig. 1.

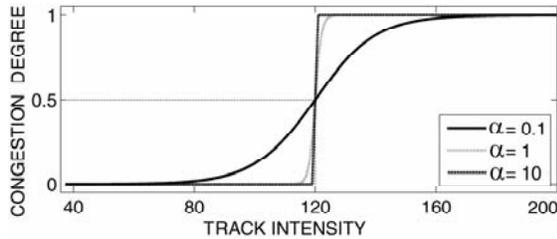


Fig. 5. Sigmoidal activation function with $\phi = 120$ and different values of α .

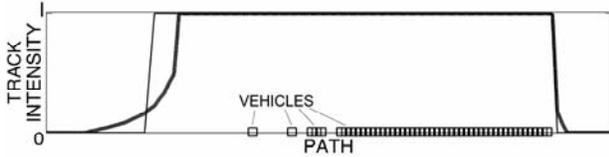


Fig. 6. Vehicles positions, track intensity (thick line), congestion degree (thin line), for the road highlighted with an oval in Fig.1.

IV. THE DESIGN OF THE PARAMETRIC ADAPTATION

The swarm aggregation designed in Section II involves a number of structural parameters to be set appropriately for each given application context. Determining such correct parameters is not a simple task since different urban areas have different traffic flux and density. Manual tuning is very time-consuming, human-intensive and error-prone. Moreover, it depends on the intuition and experience, which are typically undocumented and therefore non-reproducible. Hence, means for automated parameter tuning are required [3]. 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 differential evolution.

In the literature, the approaches for setting a set of parameters can roughly be divided into *model-free* versus *model-based* procedures [12]. The main difference is that model-based procedures build a model interpreting the relation between the algorithm and its parameter values derived from the human experience. Model-free procedures are more lightweight and 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 [13]. A well-known method of specialization is to apply constraints to the search space. Given the complexity of the search space, a population-based method, called evolutionary algorithm (EA), is commonly applied [12][14]. More specifically, we adopt Differential Evolution (DE), a method based on vector differences and therefore primarily suited for numerical optimization problems. Since EAs are meta-heuristics, they have parameters to be tuned. However they show effectiveness with default values when sufficient domain constraints are applied.

A. Model-based analysis

Table I summarizes the main structural parameters. The *hypothetical track extension* (β) depends on the traffic statistics of the specific urban area. Let us consider, for instance, a speed limit of 50 km/h and a sustainable average speed of 25 km/h = 416.6 meters/min. To set $\beta = 208.3$ meters implies that two consecutive hypothetical tracks do not overlap when car speed is higher than 25 km/h. The *mark intensity* (I) is the maximum intensity of a mark. The value of this parameter influences directly the intensity level of the track and then the triggering at the output activation interface, as well as the lifetime of a mark. For example, with an evaporation of 0.5 and $I=5$, after 3 steps the mark intensity falls under 1, and then in practice disappears.

TABLE I. STRUCTURAL PARAMETERS

Parameter	Description	Section
$\beta > 0$	hypothetical track extension	II.B
$I > 0$	mark intensity	II.C
$\varepsilon > 0$	mark extension	II.C
$0 < \theta < 1$	mark evaporation	II.C
$\phi > 0$	inflection point	II.D
$\alpha > 0$	inflection slope	II.D

The *mark extension* (ε) implies the distance of interaction between marks, and it is measured in units. The mark of a vehicle in a unit should interact with both the next and the previous occupied units. Thus, a lower bound can be 1 unit = 10 meters. In addition, considering a 100 meters congestion, vehicle marks in the middle should interact at most with the queue head and tail. This is allowed with an extension of 5 units = 50 meters. Hence, values lower than 1 may prevent mark aggregation, whereas values higher than 5 allow 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 evaporation* (θ) affects the lifetime of a mark. Short-life marks cannot aggregate, increasing the system error on the temporal start of the event, long-life marks cause saturation, increasing the error on the temporal end of the event. Hence, considering a sampling period of 1 minute, mark lifetime should be higher than 2 minutes ($\theta = 0.5$) 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. It is apparent from Fig. 5 that the maximum value for α is 10, which causes an almost-Boolean transition. Considering ϕ , a single vehicle still in a long-duration congestion produces a stationary mark intensity whose maximum value is $I / (1-\theta) = 5/(1-0.5)=10$ [15]. Thus, the minimum value of ϕ is 10 because the single vehicle does not represent a congestion event. Let us consider $\theta = 0.75$ and $\beta = 5$ to compute the track intensity with 5 overlapping triangular marks at the head or tail of the queue. With $\delta = 0$ (vehicles in queue) the intensity coefficient (Formula 4) is 2. Thus, the track intensity [15] of the 5 marks is $2 \cdot (1 + 4/5 + 3/5 + 2/5 + 1/5) \cdot 5 / (1 - 0.75) = 120$. In practice, 120 is the maximum value of ϕ because it is the track intensity that should be detected at the queue extremity. In conclusion, $\phi \in [10, 120]$.

B. The model-free parametric tuning

The parametric tuning uses DE algorithm to optimize the parameters of the system with respect to the fitness defined in (4). In DE algorithm, a solution is represented by a real n -dimensional vector, where n is the number of parameters to tune. DE starts with a population of N candidate solutions, injected or randomly generated. At each iteration and for each member (target) of the population, a mutant vector is created by mutation of selected members and then a trial vector is created by crossover of mutant and target. Finally, the best fitting among trial and target replaces the target.

In the literature different ranges of population are suggested [16]. In general, the larger the population size, the larger is the probability to find a global optimum. However, large population size decreases the convergence rate and the algorithm needs more function evaluations. Separable and unimodal functions require smaller population, while multi-modal function requires larger population to avoid premature convergence. Population size spread can vary from a minimum of $2n$ to a maximum of $40n$. To balance speed and reliability we use $N=20$. Many variants of the DE algorithm have been designed, by combining different structure and parameterization of mutation and crossover operators [17][18]. We adopted the *DE/1/best/bin* version, which uses the best individual of the population to perform the mutation. It is fast but it can be trapped in a local minimum when insufficiently constrained. 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 $[0.5-0.9]$ [17]. To choose the best value, we performed trials with $F=0.7-0.8-0.9$ and compared the results. Section IV presents this experimental study. There are different crossover methods in DE. Results show that a competitive approach can be based on binomial crossover [18]. With binomial crossover, a component of the offspring is taken with

probability CR from the mutant vector and with probability $1-CR$ from the target vector. A small crossover probability leads to a vector that is more similar to the target vector while the opposite favors the mutant vector. A large CR speeds up convergence. A good value for CR is between 0.3 and 0.9 [16]. To choose the best value, we performed trials with $CR = 0.6-0.7-0.8$ and compared results. The next Section presents the experimental study.

V. EXPERIMENTAL STUDIES

A Java-based system architecture for the proposed approach has been developed and experimented, under a research program co-funded by a regional government. The stigmergic aggregation module and the tuning modules have been developed under the Matlab¹ and the Repast² frameworks. 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. 1. In two hours of simulation, 116 congestion events occurred.

For the setting of CR and F we took into account the model-based analysis of Section III.A, thus using human experience. We call the approach “HU+DE”. More exactly, we constrained the parameters to the following values or interval-values: $\beta = 208.3$; $I = 5$; $\varepsilon \in [1, 5]$, $\theta \in [0.5, 0.75]$, $\phi \in [10, 120]$, $\alpha \in (0, 10]$. For each experiment, 5 trials have been carried out, with 30 generations. 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 II shows the fitness, in the form “mean \pm confidence interval”, for the considered values of the parameters CR and F . Here it can be observed that the variation of both parameters does not significantly affect the performance of the algorithm, and that the best performance (represented with boldface style in Table II) is achieved with $CR=0.7$ and $F=0.8$.

TABLE II. OPTIMIZATION PARAMETERS SETTING WITH THE HU+DE APPROACH: F (WITH $CR=0.7$); CR (WITH $F=0.8$)

F	Fit	CR	Fit
0.7	35.16 \pm 0.47	0.6	35.44 \pm 0.80
0.8	33.72 \pm 0.72	0.7	33.72 \pm 0.72
0.9	35.91 \pm 0.89	0.8	34.81 \pm 0.31

(a)

(b)

We carried out three experiments with different approaches, to optimize the aggregation parameters: (a) “HU+DE”, i.e., a DE constrained with the same model-based analysis used for the setting of CR and F , and then using human experience; (b) “DE”, with the following model-free parameters bounds: $\varepsilon \in [1, 10]$, $\theta \in [0.35, 0.9]$, $\phi \in [1, 500]$, $\alpha \in (0, 20]$; (c) “HU”, i.e., a manually-parameterized experiment, using default values for each parameter: $\varepsilon = 3$; $\theta = 0.675$, $\phi = 23$; $\alpha = 1$. Table III shows 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

¹ <http://www.mathworks.com>

² <http://repast.sourceforge.net/>

errors on the start and end times, and the average absolute errors on the head and 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 36.0, and that the hybrid approach (“HU+DE”) provided further improvements. Table III also shows that the major impact of the parametric adaptation 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 III. HUMAN-DRIVEN VS./WITH 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	4	0.611	18.5	26	36.0	12.3	34.9
HU	3	0.675	1	23	65.5	28.4	35.7

Fig. 7 shows the fitness versus the number of generations for the HU+DE and DE approaches. We observe that the fitness function gets stable around a small number of generations (about 30). It is also apparent that DE optimization sensibly compensates the leverage of human knowledge.

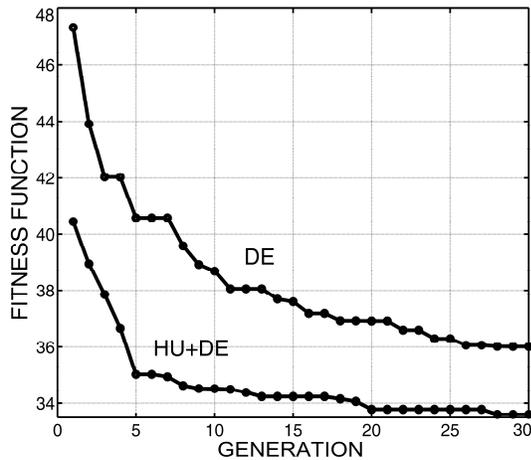


Fig. 7. Fitness function versus generation, for HU+DE and DE approaches.

VI. CONCLUSIONS AND FUTURE WORKS

In this paper, we have presented a new design of swarm aggregation of vehicle positions applied to traffic congestion estimation. The design is based on marker-based stigmergy, properly interfaced with input-output activation mechanisms for a better interoperability with the sensing and the application layers. Since the emergent character of stigmergy depends on biases and scale factors that can vary for different application contexts, an essential component of the design is the parametric adaptation. For this purpose, we designed a fitness function and adopted the differential evolution as an optimization strategy. Experimental results show the effectiveness of the approach and relevant improvements with respect to a human parameterization. Other evolutionary-based strategies may be effective in solving the type of optimization problem tackled in this work. For this purpose, to use more explorative version of

DE is considered a key investigation activity for future works, in the attempt to further reduce the need of human knowledge.

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), 2013 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] A. Ciaramella, M.G.C.A. Cimino, B. Lazzarini, 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.
- [4] 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.
- [5] 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.
- [6] 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.
- [7] 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.
- [8] 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.
- [9] 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.
- [10] 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.
- [11] 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.
- [12] F. Dobsław, "Recent Development in Automatic Parameter Tuning for Metaheuristics," in Proceedings of the 19th Annual Conference of Doctoral Students - WDS 2010, pp.54-63, 2010.
- [13] Y.C. Ho, D.L. Papyne, 2001, You-Chi, "Simple explanation of the no free lunch theorem of optimization," Cybernetics and Systems Analysis, vol. 38(2), pp. 292-298, 2002.
- [14] 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.
- [15] 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.
- [16] 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.
- [17] 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.
- [18] 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.