UNIVERSITÀ
DEGLI STUDI
FIRENZE

PHD PROGRAM IN SMART COMPUTING
DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Using computational stigmergy for swarm-based data sensing and analysis: from samples to intelligent agents

Antonio Luca Alfeo

Dissertation presented in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Smart Computing

PhD Program in Smart Computing
University of Florence, University of Pisa, University of Siena

Using computational stigmergy for swarm-based data sensing and analysis: from samples to intelligent agents

Antonio Luca Alfeo

Advisor:

Prof. Gigliola Vaglini (University of Pisa)
Prof. Mario G. C. A. Cimino (University of Pisa)

Head of the PhD Program:

Prof. Paolo Frasconi (University of Florence)

Evaluation Committee:
Prof. Xiaowen Dong, *(University of Oxford)*
Prof. Antonella Santone, *(University of Molise)*

XXXI ciclo — October 2019

a Gian

Acknowledgments

First and foremost, I thank my tutors, Prof. Gigliola Vaglini, and Prof. Mario G. C. A. Cimino, for being so supportive and patient during these three intense years of personal and professional growth.

My deep gratitude goes to Prof. Sandy Pentland and Bruno Lepri, for their contribution within my research and for the fantastic experience of my research period at the M.I.T.

I would like to express my appreciation to my thesis reviewers, Xiaowen Dong, and Prof. Antonella Santone, for helping me improving this work with their useful and constructive comments.

I would like to acknowledge the fundamental contribution of the Pegaso grant from Regione Toscana that allowed me to attend this Ph.D. program.

I thank my families, the one in Apulia and the one in Tuscany, always there for me even when time and space would not allow it.

Finally, yet importantly, I would like to thank all my colleagues, the guys from VISTA, and every researcher, professor or student that has spent just a few minutes supporting, suggesting, or simply listening to my Ph.D. journey.

Abstract

Thanks to nowdays technology advancement, a large amount of data are generated from many different fields ranging from economy, health monitoring, communications, trasportation management, or robotics. While analyzing this kind of data, one of the main problem is to identify relevant or recurrent temporal patterns and recognize anomalous one. For instance: a relevant change in the trend of social and economic indicators is fundamental for policy makers investment decisions; the reduction of the speed of multiple vehicles can identify a traffic congestion; the changing of the movement patterns of a person can identify the behavioral shift in her/his daily activities; when applied to a sleep monitoring it can help to understand the sleep quality and its progressive degeneration due to subject's disease. While facing the analysis of such data, there are numerous levels of complexity i.e. the high dimensionality, the hard-to-represent dynamics underlying an observed phenomenon, and the definition of a similarity measure aimed at work with different dynamics without explicitly modelinging them, since such model may work only under the assumption formulated by the designer. In order to tackle those issues, we exploit a biologically-inspired computational approach based on stigmergy. In biology, stigmergy is a form of indirect communication and coordination used by social insects. Specifically, each individual releases a pheromone mark in a shared environment while performing a specific action (e.g. carrying a piece of food). At the same time, its behavior is affected by the pheromones perceived in the environment (e.g. following the pheromones trail towards the source of food). In that way, subsequent actions tend to reinforce and build on each other, leading to the spontaneous emergence of coherent, apparently systematic activities. Finally, this indirectly coordinated activity has a defined temporal extension since the pheromones, given their volatility, evaporate over time. This effect is counteracted only if many pheromones are subsequently deposited in proximity with each others (thus they aggregate), resulting in the appearance of a stable pheromone trail in correspondence of this regular depositing activity.

In computer science, stigmergy can be employed as a dynamic, agglomerative, computing paradigm able to embody both spatial and temporal domain. Computational stigmergy focuses on the low level processing, where individual samples are augmented with dynamic micro-structure to enable their spatio-temporal aggregation. Such aggregation summarizes micro and macro-dynamics in data, allowing the computation of a degree of similarity between different dynamics. Finally, this approach is specialized for each case study, by employing an adaptation mechanism based on a evolutionary algorithm. Here, different applications of computational stigmergy are studied, showing the feasibility and the capability of such approach to be adopted in heterogeneous fields. Moreover, at the final stage of the architecture development, we

compare the proposed approach with state-of-art techniques on classification task.

Stigmergy is also used as a self-organization mechanism that can be fruitifully exploited in the context of swarm robotics. Swarm robotics systems have the potential to shape the future of many applications, e.g. targeted material delivery, precision farming, and distributed target search. In this contexts, a virtual representation of the pheromone is used to steer the swarm toward the most convenient part of the scenario, e.g. the area with the higher probability to have the presence of target or material to carry. Here, different applications of stigmergy-based swarm coordination are presented, showing the convenience of such approach both with distributed target search via UAVs and distributed material collection via robot.

Contents

Chapter 1

Introduction and Motivation

We are all familiar with natural swarms. The word *swarm* brings to mind images of large groups of social animals (e.g. ants, bees, flocks, or wolves) in which each member performs a simple role, but their interaction results in a collectively intelligent behavior as a whole.

The principle on which these behaviors are based is known as Swarm Intelligence (SI). According to the SI approach, each individual of the swarm should: (i) act with a certain level of autonomy (ii) perform only local sensing and communication; (iii) operate without centralized control or global knowledge, and (iv) cooperate to achieve a global goal. When applied, SI approaches provide many benefits, such as: (i) robustness, for the ability to cope the loss of individuals or noisy inputs; (ii) scalability, since the swarm does not degrade its performance by increasing its size; (iii) flexibility, thanks to the ability to cope with a broad spectrum of different environments and tasks. Given these benefits, there is an increasing interest in exploiting the SI approaches to handle the numerosity and complexity of data and agents (i.e. sensors, interfaces, and actuators), due to the adoption of the pervasive computing paradigm. As an example, the application of swarm principles to multiple robots results in the swarm robotics paradigm, whereas in computer science 'swarm intelligence' refers to a set of well known biologically inspired algorithms.

Among the many SI metaheuristics, in this work we will focus on the collective behaviors of social insects which (i) correspond to the largest research corpus both theoretically and experimentally; (ii) are based on principles which are widely common in many other animal species (Garnier *et al.* , 2007).

The emergence of such complex behavior provides the swarm with a certain degree of self-organization, which is employed to achieve goals that are far beyond the aggregation of the capabilities of each individual. As an example, a colony of African termites consists of millions of tiny (1–2 mm long) and completely blind individuals, that are able to build complex mounds up to 30 m of diameter and a height of 6 meters (Grassé, 1984). Similarly, the ant colony is able to find the shortest

path to a food source despite having no global knowledge of the search area nor a leader or a coordinator. The overall mechanism behind this capabilities is known as *Stigmergy* (Theraulaz & Bonabeau, 1999).

The word stigmergy comes from two Greek words: *stigma* which means "mark", and *ergon* which means "action". Specifically, the action is the fundamental process that produces a change in the state of the environment (e.g. by leaving a mark) (Heylighen, 2015). Whereas, the agent is the performer of an action, whose performance depends on the state of the surrounding environment. This process can be deterministic or stochastic, i.e., it can happen with a certain probability. Finally, in a multi-agent system, multiple actions coexist, and can inhibit or balance with each other resulting in a pseudo-stable configuration of the system. The environment has the fundamental role of being a shared media in charge of the communication among the agents (Parunak, 2005). Moreover, the environment can maintain a sort of short-term memory of the action performed by agents by aggregating subsequent marks in a structure named trail.

Parunak in (Parunak, 2005) distinguishes four categories of stigmergy, according to two binary features: (i) the mechanism enabling the exploitation of the environment to communicate allows to distinguish between *marker-based stigmergy*, which is based on the releasing of a mark in the environment, and the *sematectonic stigmergy*, which uses the behavior exhibited by close agents as a stimuli; (ii) if the dimension of the signal is a scalar, we talk about *quantitative stigmergy*, *qualitative stigmergy* otherwise.

To facilitate the understanding of the biological metaphor taken as a reference in this work, it follows few biological examples of markers-based stigmergy.

While performing a collective foraging task, ants exploit the presence of pheromones to mark the path between a source of food and the nest. Moreover, different pheromones are used to foster or inhibit the activities of the colony. This is referred as *qualitative marker-based stigmergy*.

The nest building process performed by the termites is an example of *quantitative marker-based stigmergy* since the termites pile the mud-ball together by following the gradient (thus, a continuous quantity) of the pheromone trail.

Finally, another example of *quantitative marker-based stigmergy* occurs with humans, i.e. the behavior of the trade market. Here, the amount of transactions aimed at buying and selling a product (the actions) affects the value of the product itself (i.e. leave a trace in a shared market). Moreover, the value act as a stimulus for following transactions (Heylighen, 2015).

With marker-based stigmergy (stigmergy, for brevity), each individual releases a pheromone mark while performing a specific action (e.g. carrying food to the nest). On the other hand, the individuals that perceive the pheromones adjust their behavior accordingly (e.g. follow the pheromone trail toward the source of food). Since

pheromones evaporate over time, only subsequent actions build and reinforce on each other, resulting in a collective self-organized behavior. This mechanism can be reproduced, extended and employed in applications such as swarm robotics (Brambilla *et al.* , 2013a) and data analysis (Ramos *et al.* , 2002). Indeed, robots, UAVs and even data samples can be provided with simple interactions aimed at self-organizing a collective task (Cimino *et al.* , 2015c) or unfold structure (such as pattern) in data (Cimino *et al.* , 2015b).

1.1 SI for computation: Computational Stigmergy

The recent adoption of pervasive technologies in fields such as the economy, health monitoring, communications, and transportation management, has generated a large amount of data with the promise of providing better and strategic insights about each one of these application contexts. The analysis of the dynamics folded in this kind of data usually employs two types of models: (i) the knowledge-based model, explicitly designed as logical or mathematical rules, and determined by a domain expert; (ii) the data-driven model, in which the model is derived from prototypical data via machine learning. Specifically, knowledge-based models belong to the cognitivist paradigm (Vernon *et al.* , 2007). In this paradigm, the system aimed at analyzing the data has to be designed according to the model provided by an expert in the field, who translates his/her knowledge about that specific domain into a set of rules that can be exploited by an information processing system. However, a knowledge-based approach is highly context-dependent and hardly scalable. On the other hand, data-driven models are easier to adapt to different context or case study and more robust with respect to noisy and unexpected inputs. The data-driven approach discussed in this work takes inspiration from the emergent behavior of the ant colony, i.e. employ the principle of stigmergy to offer a model-free computational approach, characterized by adaptation and self-organization of data (Cimino *et al.* , 2015b).

One of the first applications of stigmergy in computer science were the approaches known as ant algorithms (Dorigo *et al.* , 2000), which have been used in solving several computational problems, such as the traveling salesman problem, structural engineering, digital image processing, scheduling problem, and routing (Mohan & Baskaran, 2012). Among the many application, only few research works address the exploitation of the paradigm of stigmergy to pattern analysis (Ramos & Almeida, 2004). As an example, in (Brueckner & Parunak, 2002) authors propose a stigmergy-based approach aimed at detecting patterns and applies it to the classification of synthetic images. However, the authors focused only on spatial patterns. On the other hand, our approach exploits both spatial and temporal dynamics, because it intrinsically embodies the time domain. Finally, our approach overcomes the ex-

plicit top-down domain-dependent modeling of the dynamics folded in the data under analysis. Indeed by using computational stigmergy, these dynamics spontaneously arises from the interactions between data sample and can be described with a domain-independent spatiotemporal logic (details in Chapter 3).

1.2 Environment Exploration with SI

Nowadays, unmanned vehicles (e.g. cars, robots, drones) are more and more exploited as a novel infrastructure together with new technologies (e.g. IoT, 5G) to build a cybernetic senseable, programmable, and actuable ecosystem.

In this vision, an increasing number of services will be provided in an autonomous and distributed manner. However, given the scale of the problem both structure and control logic may result in a high cost in terms of design, construction, and maintenance. Especially if the solution is modeled as a centralized system, since a single hardware or software fault may affect the whole service. Hence, to solve problems cooperatively while maintaining reliability and scalability, application designers are investigating the behavior-based solution belonging to swarm intelligent paradigm (Steels, 1990), with a specific focus on swarm robotics applications.

The main inspiration for a swarm of robots/UAVs comes from the observation of social animals, such as insects, birds, and fish, that exhibit a collective intelligence, i.e. achieve complex goals through simple rules and local interactions and no centralized control (Brambilla *et al.* , 2013a). By providing a swarm of robot/unmanned vehicles with these capabilities we are able to create an autonomous detection/disposal/delivery service (D'Andrea, 2012). As an example, UAV swarm can offer great potential in search and rescue operation since they can perform tasks in highly inhospitable environments, providing a quick "survey" of the area, and better investigate only key locations that provide some circumstantial evidence (Cimino *et al.* , 2016). As another example, swarms of robots are becoming more and more important as a part of industrial activities such as warehouse logistics (D'Andrea, 2012; Liu *et al.* , 2017).

In this context, many works in the field employ stigmergy as a coordination mechanisms of the swarm. As already specified, stigmergy is a fundamental swarm coordination mechanism, based on the release of information in the environment in the form of pheromone marks (Sauter *et al.* , 2005a). The pheromone is a volatile substance that diffuses locally, stays temporarily, and affects other individuals behavior if perceived (Parunak *et al.* , 2002). However, it may be very hard and costly to equip the swarms of robots/UAVs with the capability of producing a physical mark in the environment (i.e. with a physical substance) in which the task takes place (Kuyucu *et al.* , 2015). For this reason, most of the works in the field use simulated (i.e., virtual) pheromones. In a distributed environment, a pheromone map

of the search space can be maintained and made available for drones as a "remote brain" capability (Ermacora *et al.* , 2013) or stored locally via a properly "writable" environment (Johansson & Saffiotti, 2009).

For example, clustering tasks have been modeled with ant behavior like strategies (Holland & Melhuish, 1999). The authors in (Valckenaers *et al.* , 2007) applied stigmergy to model the problem of control and coordination in a manufacturing environment. Stigmergy has also been applied for the control of coordination of aerial drones (Sauter *et al.* , 2005a).

In this context, many SI approaches have been used or combined with stigmergy to provide the swarm with an autonomous coordination mechanism aimed at the exploration of an area while detecting and/or collecting objects.

As an example, in (Wagner *et al.* , 2008) authors provide a mechanism similar to cellular automata, in which robots coordinate their exploration according to the environment local status, seen as a grid. In (Dasgupta *et al.* , 2009) the proposed strategy employs the Reynold's flocking model so as to divide the robots into teams and explore the area. This solution allows the swarm to adjust its configuration (i.e., change dynamically the team membership of an individual) according to the constraints of the environment (e.g. obstacle presence, area overcrowding). In (Wang *et al.* , 2011) Particle Swarm Optimization is used to balance the exploration of nearby subareas and to disperse the robots in order to avoid redundancy coverage. However, among the many metaheuristics for environment exploration (Brambilla *et al.* , 2013a; Parker, 2003; Tan & Zheng, 2013) flocking (Cimino *et al.* , 2015b) and foraging (Zedadra *et al.* , 2015) seem to be the most widely exploited.

To briefly introduce the fundamental concepts about flocking and foraging strategy (more details and comparison with other works in Chapter 2), we present their ontologies. An ontology is a structured definition of basic concepts and relations among them (Siegemund *et al.* , 2011). In Fig. 1.1(a) we shows the ontology of foraging task (Steels, 1990), in which each *individual* starts its exploration of the environment from the *nest*, by randomly moving in the *environment*. The *environment* contains *food* which is collected by the individuals in the *nest* once it is found (Fig. 1.1(b)).

In Fig. 1.2(a) we show the ontology of a flocking agents, here called *boids* (as in the literature), which is based on three features: *separation* with respect to the closest flockmates to avoid collision and overcrowding, *alignment* with the flock, and *cohere* to avoid dispersion of the flock during the collective flight(Reynolds, 1987a). Flockmates are determined according to different radiuses for each rule, as represented in Fig. 1.2(b).

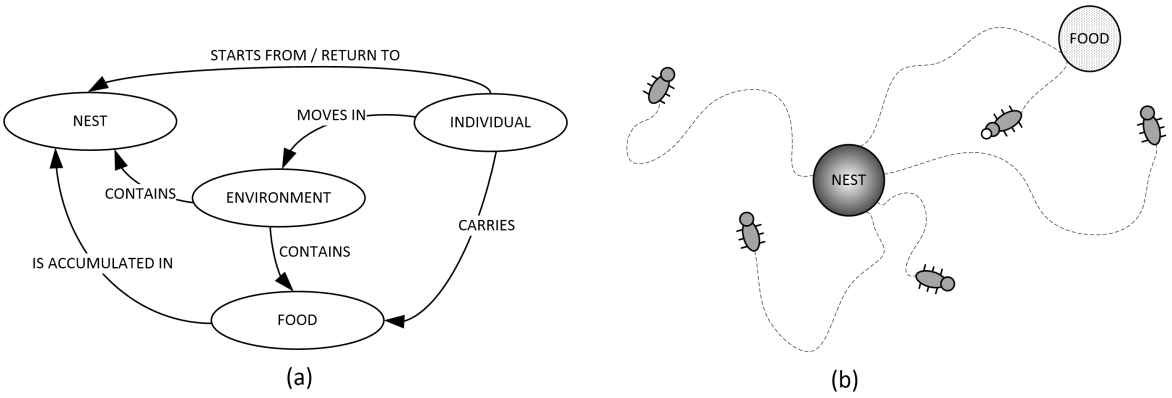


Figure 1.1: (a) Foraging ontology. (b) Illustration of the foraging task.

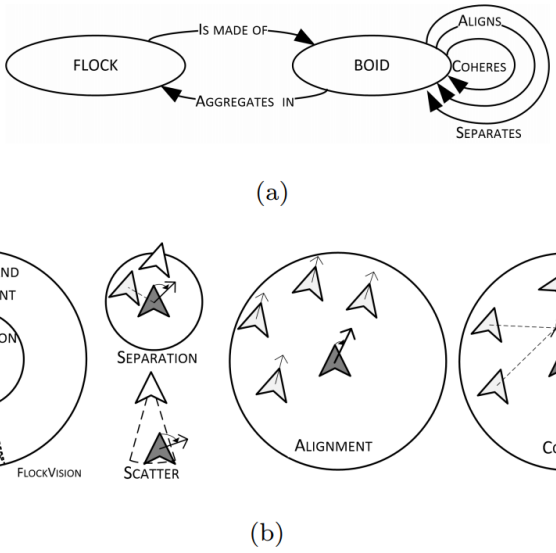


Figure 1.2: (a) Flocking agents ontology. (b) Illustration of separation, alignment, and cohere rules.

1.3 Adaptation with SI

Optimization problems can be solved by means of search methods, i.e. by trying out multiple attempts until a satisfactory result is obtained. In this context Swarm Intelligence (SI) methodologies have been applied to implement search mechanisms to address problems that cannot be modeled by means of exact and analytical techniques (Kachitvichyanukul, 2012). Indeed, each case employs different quality metrics. Then, an optimization method using a "black box" approach, which is not based on formal properties of the quality function, may be effective. SI approaches can find near-optimal solutions by iteratively trying to improve a population of candidate solutions with regard to a given measure of quality, or fitness. Solutions

are improved by means of stochastic transformation mechanisms inspired by biology, such as reproduction, mutation, recombination, selection, survival, swarm, and movement, in an environment whose dynamics are represented by the quality measure.

SI optimization methodology is based on three fundamental processes: the generation of the initial population, the evaluation of the fitness value and the generation of a new population. At first, the members of the population are initialized randomly or injecting some initial solutions. Each member represents a solution to the problem. The second step, the members are evaluated by means of a fitness function. The fitness is a measure of goodness of the solution and it is aimed at comparing two members of the population and selecting which one is better. The third key process is the generation of a new population via the perturbation of the current population and the selection of the best candidates. The second and the third step iterate until a stop condition is met. There are two main categories of stop condition: (i) static, e.g., a fitness value to reach, a number of generations to compute, etc.; (ii) dynamic, e.g., a percentage of improvement of the fitness between generations, convergence of the population members, and so on. In general, all SI approaches have (at least) two parameters to be set which are the number of members of the population and the stop condition, which usually is the number of generation to compute. Since the mid-sixties, many SI approaches have been proposed, and many efforts have also been devoted to comparing them. In the last decade, three SI approaches attracted the most of the attention: Genetic Algorithm (GA) (Holland, 1992), Particle Swarm Optimization (PSO)(Poli *et al.* , 2007), and Differential Evolution (DE) (Storn & Price, 1997).

GA (Holland, 1992), takes inspiration from the natural selection of genetic features during the reproduction of a living organism. In GA a member of the population is represented as a chromosome. The fitness value is used to rank the individuals of a given generation, whereas the members of the next generation are computed by means of three steps: selection, crossover, and mutation. Firstly, the chromosomes are selected to become the parents of the new generation according to their rank; secondly, the chromosomes of the selected parents are combined by means of the crossover process to generate a new chromosome. Finally, some modification is injected in the new chromosome by means of the mutation process to increase diversity and avoid stagnation.

In PSO (Poli *et al.* , 2007), a member is a particle and the population is a swarm of particles. The particle has a position, a velocity, and the historical best position. The position represents a solution in the search space of the problem, the velocity describes the particle movement, and the best position represents the best solution visited by the particle. The quality of the solution (thus, the quality of the position) is computed with the fitness function. The best position among all the individuals'

best position is referred as the best position of the swarm. Initially, the position and the velocity of each particle in the swarm is chosen randomly. The next position depends on its current position and velocity. According to the fitness of the new positions, the individual best and/or the swarm best position may be updated. Finally, depending on these best values, the velocity for each particle is updated.

DE (Storn & Price, 1997) considers a member of the population as a D-dimensional vector (also known as genotype), in which D is the dimensions of the search space. During each generation, the creation of new vectors follows three steps: mutation, crossover, and selection. In the mutation, three random members of the population are combined to obtain a mutant vector. Then, the crossover operator is applied between the candidate member and the mutant vector, generating the trial vector. The crossover is a random selection in which each element of the trial vector is taken from the candidate or the mutant. Finally, by means of the selection, a comparison between the candidate and the trial vector is computed, i.e. their fitness values are compared. The best one passes to the next generation, while the other is discarded.

Table 1.1 (Kachitvichyanukul, 2012) summarizes the qualitative properties of the three algorithms. The key difference is in the mechanism used to produce the new generation of individuals. In GA, the members of the population need to be ranked according to the fitness value and the best members are more likely chosen as parents of the new generation. For this reason, GA tends to produce solutions in clusters around some "good" solutions. Moreover, because of the ranking, the time to compute all the generation scales nonlinearly with the number of members of the population. In PSO, the new position of the particle is generated with the old position and the velocity. Thus, the particle can end in every possible location of the search space. In addition, the best particle found by the swarm influences the update step of the velocity that can lead to a premature convergence of the algorithm. DE has a similar exploration capability of PSO because the generation of the new solution is computed from (at least) three random members of the population. Moreover, the best member of the population does not take part in the generation, thus the algorithm does not suffer from premature convergence.

Table 1.1: An excerpt of the properties of the algorithms GA, PSO, and DE.

Property	GA	PSO	DE
Require ranking of solution	Yes	No	No
Influence of population size on solution time	Exponential	Linear	Linear
Influence of best solution on population	Medium	Most	Less
Average fitness cannot get worse	False	False	True
Tendency for premature convergence	Medium	High	Low
Density of search space	Less	More	More
Ability to reach good solution without local search	Less	More	More

Authors in (Vesterstrom & Thomsen, 2004) compared DE, PSO and a Simple Evolutionary Algorithm (SEA) on a set of 34 benchmark problems, including unimodal

and multimodal functions with correlated and uncorrelated variables, a problem with plateaus, and two noisy problems. Problem dimensionality varied from 2 and 30, and an extension to 100-dimensionality was provided to assess the performance on more difficult problems. Authors reported DE as the best performing algorithm, even considering that DE has slower convergence in the two noisy problems. Moreover, DE stagnates in a sub-optimal solution (the same problem occurred for PSO) in one of the benchmark function. As pointed out by the authors, such problems can be solved modifying the DE strategy or adjusting its parameters. In general, DE is robust and able to reproduce the same results consistently over many trials, in particular for 100-dimensionality problems. It is apparent from the previous discussion that DE is a simple, robust, fast and efficient adaptive scheme for global optimization. In addition, it has few parameters to set, and the same settings can be used in different problems. For this reason, it was selected to design the adaptation subsystem.

Fig. 1.3 shows the DE ontology. Briefly, candidate solutions to the parameterization problem play the role of *individuals* in a *population*. Each individual mutates and mates with its peer generating a new one, characterized by its own *genotype*. Each *genotype* generates a specific *behavior* of the solution, which corresponds to a *phenotype* that *fits* differently according to the *scenario*. Such *fitness* is the basis for the objective function.

Many variants of the DE algorithm have been designed, by combining different structure and parameterization of mutation and crossover operators (Mezura-Montes *et al.* , 2006a),(Zaharie, 2007). We introduce them according to a classic DE taxonomy, i.e. as $DE/x/y/z$, where:

- x defines the base choice (v_1) of the mutant vector (v_{mutant}) between (i) $rand$, random vector, which explores more, but requires more generations to converge; (ii) $best$, the best population individual, which converges faster, but risks to be trapped in local minima; and (iii) $rand-to-best$, a combination of the above strategies (i.e. the weighted sum of F).

- y is the number of differences in the mutation carriers, i.e. 1 means $v_{mutant} = v_1 + F \cdot (v_2 - v_3)$, whereas 2 means $v_{mutant} = v_1 + F \cdot (v_2 + v_3 - v_4 - v_5)$ where $v_2, ..., v_5$ are always random.

- z is the type of crossover, i.e. (i) bin (binomial), in which CR is the probability that an element of the vector is taken from the target or from the mutant vector; or (ii) 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.

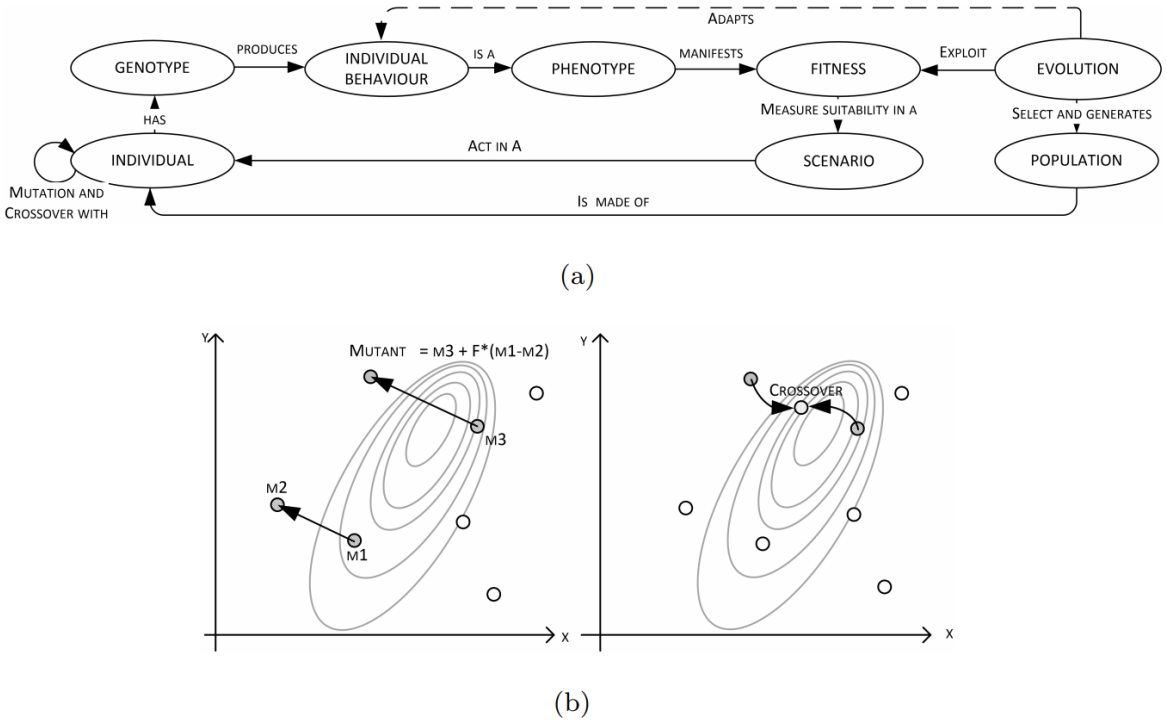


Figure 1.3: (a) Differential Evolution Ontology. (b) Illustration of mutation and crossover.

Many works in the field tried to compare these different versions. As an example, in (Mezura-Montes *et al.* , 2006b) authors concluded that the binomial crossover outperforms the exponential in almost every experiment. Moreover, DE/best/1/bin is a competitive DE variant for a wide set of benchmark problems. In (Zaharie, 2007) the authors explored the performance of DE with binomial and exponential crossover by varying the crossover rate CR. Specifically, the Crossover Rate CR $\in$ [0, 1] represents the probability to pick an element from the target or the mutant vector during the crossover operation; the other parameter exploited by DE is the differential weight F $\in$ (0, 2], i.e. the scaling factor between the mutant vectors generation (Fig. 1.3b). In any case, there does not seem to be any agreement in the literature on the versions and optimal parameterization of DE since each case study can be characterized by specific peculiarities (Das & Suganthan, 2011), thus further investigation in this direction should be done.

Finally, the DE approach is further detailed by means of pseudocode. In DE a solution is represented by a real K-dimensional vector. DE starts with a randomly generated population of N candidate solutions, i.e. $P^{(0)}$. At each iteration t and for each genotype $\boldsymbol{p}_i^{(t)}$ of the current population $P^{(t)}$, a mutant vector $\boldsymbol{m}$ is created by applying the mutation of randomly selected members. Then, a trial vector $\boldsymbol{p}_i^*$ is created by crossover of $\boldsymbol{m}$ and $\boldsymbol{p}_i^{(t)}$. Subsequently, the population is modified according

Algorithm 1: Differential Evolution Algorithm

```
function DifferentialEvolution(SearchSpace)
t = 0;
P^(0) = initializePopulation();
for each genotype p_i^(0) in P^(0) do
    M_i^(0) = genotypeToIndividual(p_i^(0));
    f_i^(0) = computeFitness(M_i^(0), SearchSpace);
do
    for each genotype p_i^(t) in P^(t) do
        m = generateMutant(P^(t), p_i^(t));
        p_i^* = binomialCrossover(p_i^(t), m);
        M_i^* = genotypeToIndividual(p_i^*);
        f_i^* = computeFitness(M_i^*, SearchSpace);
    for each genotype p_i^(t) in P^(t) do
        if (f_i^* < f_i^(t)) then
            p_i^(t+1) = p_i^*; f_i^(t+1) = f_i^*;
        else
            p_i^(t+1) = p_i^(t); f_i^(t+1) = f_i^(t);
        f_min^(t+1) = min{f_1^(t+1), ..., f_N^(t+1)};
    t = t + 1;
while (terminationCriterion(f_min^(t), t) = false);
return genotypeToIndividual(p_min^(t));
```

to the best fitting vector between the fitness of the trial vector (f_i^*) and the fitness of the initial genotype ($f_i^{(t)}$). When the termination criterion is true, the vector characterizing the solution with the best fitness (i.e. the shortest search time) in the current population is considered as the optimal problem solution.

As already specified, DE itself has at least two hyperparameters: the scaling factor $F\epsilon[0,2]$ from which results the mutant vector, and the crossover probability CR. The smaller CR, the higher probability of producing a vector that is more similar to the target vector rather than to the mutant vector. More formally, in the following we define the mutation and the crossover operators, respectively.

1.4 Thesis Outline

This dissertation has five Chapters: In Chapter 1 the motivations and background of the study are introduced. In Chapter 2 a literature review is presented. Specifically, the topics addressed are (i) an overview of the similarity measures for temporal pat-

Algorithm 2: Mutant vector generation

function generateMutant($P^{(t)}$, $\boldsymbol{p}_i^{(t)}$)
$\boldsymbol{p}'$ = randomExtraction($P^{(t)} \setminus \{\boldsymbol{p}_i^{(t)}\}$);
$\boldsymbol{p}''$ = randomExtraction($P^{(t)} \setminus \{\boldsymbol{p}_i^{(t)}, \boldsymbol{p}'\}$);
$\boldsymbol{p}'''$ = randomExtraction($P^{(t)} \setminus \{\boldsymbol{p}_i^{(t)}, \boldsymbol{p}', \boldsymbol{p}''\}$);
return $\boldsymbol{p}' + F \cdot (\boldsymbol{p}'' - \boldsymbol{p}''')$;

Algorithm 3: Binomial crossover

function binomialCrossover($\boldsymbol{p}_i^{(t)}$, $\boldsymbol{m}$)
k = randomInteger(1, K);
for each j-th gene $p_{j,i}^{(t)}$ **in** $\boldsymbol{p}_i^{(t)}$ **do**
  if (randomReal(0,1) $< CR$) or $(j = k)$) **then**
    $w_j = m_j$;
  else
    $w_j = p_{j,i}^{(t)}$;
return w;

terns; (ii) urban traffic congestions estimation; (iii) pattern analysis in urban mobility (iv) behavioral analysis via physiological signals; and (v) Swarm Intelligence approaches for distributed target detection, and distributed material disposal. Chapter 3 presents the Stigmergy-Based Architecture for temporal pattern analysis; moreover, its possible configuration according to the current application are discussed. Chapter 4 presents the provided applications in subsections, each one of them for a specific topic and application. Each subsection present the problem statement of the current application(s), the design of the proposed solution, and the obtained results. In Chapter 5 the conclusions of the study are drawn.

Chapter 2

Related work

In this section, a review of the research addressing the characterization and the enhancement of temporal dynamics is presented. Specifically, the review is presented by starting from a general view of the characterization of the temporal dynamics in data and then specializing the review of the literature on the applications covered during the Ph.D.

2.1 Similarity Measures for Temporal Pattern

A time series is a collection of chronologically ordered observation, regarding the same phenomenon, thus considered as a whole set rather than an aggregation of samples (Fu, 2011). The analysis of the time series has the purpose of identifying trends, cycles, and seasonal variances to aid in the forecasting of a future event. A fundamental mechanism to recognize trends and their variances is the ability to properly compare different time series. One of the main problems in this kind of analysis is due to the fact that time series are subjected to distortions such as noise, offset translation, scaling (amplitude or longitudinal), linear drift and discontinuities (Keogh & Kasetty, 2003). For these reasons, usual techniques relying on exact match making can't be adopted, and approximate techniques have to be considered. However, the similarity measure should be consistent with the intuitive notion of shape and provide the some fundamental properties (Esling & Agón, 2012), i.e. it should (i) provide recognition of perceptually similar objects, even though they are not mathematically identical; (ii) be consistent with human intuition; (iii) emphasize the most salient features on both local and global scales; (iv) allows to identify or distinguish arbitrary objects, that is, no restrictions on time series are assumed, and (v) robust with respect to distortions (Keogh & Kasetty, 2003).

In general, it is possible to distinguish between 4 categories of similarity measures for time series (Esling & Agón, 2012).

- The *edit-based* distances compare two time series according to the minimum number of modifications (insertion, deletion, and substitution) needed to transform one series into another one. As an example, the Longest Common Subsequence (Das *et al.* , 1997) allows having unmatched elements while measuring the cost to transform a time series to another without changing the sequence of the elements. Thus, it can handle the presence of the outliers (discontinuities) since the editing cost will not be incremented due to them. The Edit Distance on Real Sequence (EDR) (Chen *et al.* , 2005) assigns a certain edit cost if the difference between two elements is greater than a given threshold. Then, according to the gap between the two time series, an alignment cost is added. Finally, the Edit Distance with Real Penalty (ERP) (Chen & Ng, 2004) is a version of the EDR in which there is no threshold, and the alignment cost is measured by using the Euclidean distance.

- The *feature-based* distances extracts a set of features (descriptor) from a couple of time series and compute a similarity/distance measure between them. A common descriptor is the coefficients from the Discrete Fourier Transform (DFT) (Shatkay & Zdonik, 1996) or the Discrete Wavelet Transformation (DWT) (Chan & Fu, 1999). Indeed, Vlachos et al. (Vlachos *et al.* , 2005) present a combination of period-gram and autocorrelation functions which permits the selection of the most important periods of a series. Papadimitriou et al. (Papadimitriou *et al.* , 2006) propose a tracking of the local correlation extending Vlachos (Vlachos *et al.* , 2005). Janacek et al. (Janacek *et al.* , 2005) show that a likelihood ratio for DFT coefficients outperforms the Euclidean distance. Concerning symbolic representations, Mannila and Seppnen (Mannila & Seppänen, 2001) use random vector to represent symbols in a time series. Thus, the sum of the vectors weighted by the temporal distance is the representation of a sequence of symbols. Instead, Flanagan (Flanagan, 2003) uses weighted histograms of consecutive symbols as features.

- The *structure-based* similarity aim at finding a higher-level structure in the time series and then to compare them on a more global dimension. Popular approaches use Hidden Markov Model (HMM) with continuous output values or ARMA models (Xiong & Yeung, 2004). Ge and Smyth (Ge & Smyth, 2000) combine HMMs and piece-wise linear representation. Bicego et al. (Bicego *et al.* , 2003) use the similarity-based paradigm where HMM is used to determine the similarity between each object and a predetermined set of other objects. Keogh et al. (Keogh *et al.* , 2004) define a distance measure based on the Kolmogorov complexity called Compression-Based Dissimilarity Measure (CDM). The underlying idea is that concatenating and compressing similar series should produce higher compression ratios than when doing so with very

different data. This approach appears particularly efficient for clustering; it has been applied to fetal heart rate tracings (Santos *et al.* , 2006). Similarly, Degli Esposti et al. (Degli Esposti *et al.* , 2009) propose a parsing-based similarity distance in order to distinguish healthy patients from hospitalized ones on the basis of various symbolic codings of ECG signals. By comparing the performances of several data classification methods, this distance is shown to be a good compromise between accuracy and computational efforts.

- The *shape-based* distances compare the overall shape of the time series. These have been the most widely used measures for time series analysis. Among them, the Euclidean distance results to be easy to implement, fast to compute (linear) , and parameters-free. On the other hand, the Euclidean distance offer a fixed one-to-one mapping between points of two time series. Thus, it is sensitive to distortions such as offset translation and temporal drift. The same issue occurs with the Minkoski distance, i.e. the generalization of the Euclidean distance. Another measures employed for the similarity among time series is the Pearson's correlation coefficient (the closer to 1, the higher the similarity). The Short Time Series distance, STS, is the sum of the squared difference of the slopes of the two time series. The main advantage of this measure in comparison with the previous is that it embodies the temporal information of the time series. The more popular shape-based distance measure is the Dynamic Time Warping (DTW) (Berndt & Clifford, 1994). DTW measure the distance by considering the best alignment between two time series, by stretching them along the temporal dimension up to of a given amount (the time window). Thus, its robust to distortions such as the temporal shift. The Frechet distance (Driemel *et al.* , 2016) is a shape-based distance typically explained as the relationship between a person and a dog connected by a leash walking along the two curves and trying to keep the leash as short as possible. The maximum length the leash reaches is the value of the Frechet distance.

In the end of the day, there is no unique choice when it comes to compare different time series. In general, the most adequate similarity measure highly depends on the nature of the data to analyze as well as the required application-specific properties. According to them some guidelines can be drawn, as suggested by Esling et al. (Esling & Agón, 2012): (i) If the time series are relatively short and visual perception is a meaningful description, shape-based methods seem to be the appropriate choice; (ii) Feature-based methods seem more appropriate when periodicities are the central subject of interest and causality in the time series is not relevant; and (iii) if the time series are long and little knowledge about the structure is available, structure-based approaches have the advantage of being a more generic and parameter-free solution for the evaluation of similarity; Even with these general rec-

ommendations, the accuracy of the measure chosen has to be evaluated case by case. This evaluation is commonly performed within a 1-NN classifier framework. It has been shown by Ding et al. (Ding *et al.* , 2008) that, despite all proposals regarding different kinds of robustness, the forty year old DTW usually performs better. Thus, in the studies presented in this thesis DTW distance is often employed as a competitor of the presented stigmergy based similarity measure.

2.2 Behavioral Analysis via Physiological Signals

The human behavior can be considered as the collection of every physical action and observable condition associated with an individual. In the literature, many studies have recently proved that is possible to distinguish among different human activities, as well as evaluate user's physical condition, through wearable device and data-driven classification techniques (Abbate *et al.* , 2012). As an example, in (Bonomi *et al.* , 2010) 30 healthy subjects have been monitored for 14 days, using: (i) a triaxial accelerometer for movement registration to calculate the activity counts per day; (ii) a laboratory equipment (indirect calorimetry) to calculate the total energy expenditure in free living conditions; (iii) a respiration chamber to measure during an overnight stay the sleeping metabolic rate. The activity energy expenditure and the physical activity level are determined from total energy expenditure and sleeping metabolic rate. A direct linear association was observed between the activity counts per day and the physical activity level. A multiple-linear regression model predicted 76% of the variance in total energy expenditure, which is a very high accuracy for predicting free-living energy expenditure. (Guiry *et al.* , 2014) gathered samples from 10 subjects, each equipped with a smartphone and a smartwatch, exploiting all available sensors (tri-axial accelerometer, tri-axial magnetometer, tri-axial gyroscope, GPS, light and pressure sensors). Subjects were asked to perform specific physical activities during three different gathering phases. In the proposed approach, data samples are first preprocessed via Principal Component Analysis. Subsequently, the data set is used to classify the physical activities, by using five well-known learning algorithms: C4.5, CART, Naïve Bayes, Multi-Layer Perceptrons and Support Vector Machines. Results show that the system correctly classifies the activities with a percentage of 95% when using a smartphone and 89% when using a smartwatch. (Parkka *et al.* , 2007) estimate the intensity of physical activity attaching accelerometers and gyroscopes to ankle, wrist and hip. The results are compared to metabolic equivalent measures obtained by means of a portable system used for testing cardiopulmonary exercise. Experiments are made with 11 subjects carrying out everyday tasks, including ironing, vacuuming, walking, running, and cycling on exercise bicycle (ergometer). The authors have calculated a linear correlation between accelerometers signals and metabolic equivalent up to

0.86. (Zhu *et al.* , 2015) estimate physical activities energy expenditure using wearable devices in different activities: walking, standing, climbing upstairs or downstairs. More specifically, a Convolution Neural Networks is used to automatically detect important features from data collected from triaxial accelerometer and heart rate sensors. The results are compared with the state-of-the-art of linear regression and artificial neural networks applied to specific activities, obtaining a mean square error of 1.12 which is about 35% lower than existing models.

However, when the study involves the sleep quality assessment some of the above proposed techniques are no longer valid since they are too intrusive and may alter the sleep behavior of the subject. For this reason many of the studies refer actigraphy, and wearable devices in general, as an alternative to well known polysomnography, since they can provide similar insights if applied with the due restraint. Specifically, to evaluate the sleep quality independently of the used technology, we found in literature two main approaches. The first approach is based on measure metrics such as the time taken to initially fall asleep (sleep onset latency, SOL), time awake overnight after sleep onset (wake after sleep onset, WASO), and total sleep time (TST). When all these metrics are evaluated, an accurate assessment of the person's sleep efficiency (SE) can be made. SE is an overall measurement of a person's sleep quality and it simply is a ratio of the time spent asleep (TST) to the amount of time spent in bed (SOL + WASO + TST). This method is based on metrics used by sleep clinicians to infer if the patients do not suffer sleep problems (Frankel *et al.* , 1976). The other approach is based on the classification of the subjective quality ratings. The quality ratings are usually captured with self-reports via paper-based surveys and diaries. Examples include the Sleep Timing Questionnaire (Monk *et al.* , 2003) and the Epworth Sleepiness Scale (Johns, 1991). A complementary approach involves keeping a sleep diary. While tedious to collect, a diary-based approach has proven to be reliable (Bootzin & Engle-Friedman, 1981). Indeed, in (Espie *et al.* , 1989), the authors demonstrate that daily self-report is a valid index of sleep disturbance. Sleep diaries have also been found to be reliable for bedtime and wake-time estimates via actigraphy (Wilson *et al.* , 1998) and ambulatory electroencephalographic monitoring (Maquet, 2001). Nevertheless, data collected via diaries usually present a high level of uncertainty, thus requiring adaptation. A novel trend of work investigates explicit integration of machine learning algorithms into the data collection process to accomplish adaptation. For example, machine learning methods are deployed in (Huang & Oviatt, 2005) to achieve on-line adaptation to users' multimodal temporal thresholds within a human computer interaction application framework. Some other work studies application of reinforcement learning to adaptive fusion systems to perform dynamic data reliability estimation (Ansari *et al.* , 1996), (Hossain *et al.* , 2009). A recent work also proposed using kernel-based learning methods to achieve adaptive decision fusion rules (Fabeck & Mathar, 2008).

Machine learning techniques can improve the objectivity and reliability of the observations. However, since signal morphology vary widely between people, the use of machine learning algorithms frequently implies a careful tuning of their structural parameters. Unfortunately, this tuning is often a "black art" requiring expert experience, rules of thumb, or sometimes brute-force search (Snoek *et al.* , 2012). There is therefore great appeal for approaches aimed to save the effort required in model parameterization and management.

2.3 Traffic Congestion Estimation

Taking into account the technology involved in traffic state estimation, a number of methods have been developed. In (Tabibiazar & Basir, 2011) 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 (Kong *et al.* , 2009) 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 (Kong *et al.* , 2013) 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 (Chen *et al.* , 2007) 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 (Zhao *et al.* , 2011) 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 (Yoon *et al.* , 2007). 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 us-

ing 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 subsystems, 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 (Dorigo *et al.* , 2014). 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 (Kurihara *et al.* , 2009) 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 (Bedi *et al.* , 2007) 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 (Caselli *et al.* , 2015) 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 (Ito *et al.* , 2012) 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 (Ito *et al.* , 2012) the road is considered as a monolithic structure, in (Narzt *et al.* , 2010) roads are divided into segments.

2.4 Pattern Analysis in Urban Mobility

The increasing volume of urban human mobility data arises unprecedented opportunities to monitor and understand crowd dynamics. Identifying events which do not conform to the expected patterns can enhance the awareness of decision makers for a variety of purposes, such as the management of social events or extreme weather situations (Sagl *et al.* , 2012). For this purpose GPS-equipped vehicles provide a huge amount of reliable data about urban human mobility, exhibiting correlation with people daily life, events, and city structure (Veloso *et al.* , 2011). The majority of the methods approaching the analysis of vehicle traces can be grouped into three categories: *cluster-based*, *classification-based*, and *pattern mining-based*; whereas the main application problems include the hotspot discovery, the extraction of mobility profiles, and the detection and monitoring of big events and crowd behavior (Mazimpaka & Timpf, 2016). In this context, the wide availability of taxi trip data has produced a significant number of works aimed to mine urban dynamics by exploiting this kind of data. For example, in (Kuang *et al.* , 2015) the authors use a Multiscale Principal Component Analysis to analyze taxi GPS data in order to detect traffic congestion. In (Peng *et al.* , 2012) the authors use non-negative matrix factorization (NMF) algorithm to decompose taxi activity levels and extract three basic patterns. Those patterns represent respectively: (i) commuting between home and workplace, (ii) business traveling between two workplaces, and (iii) leisure trips from or to other places. Furthermore, authors model the relative daily deviation of the traffic flow in each category. In (Zhang *et al.* , 2015) authors analyze taxi traces in order to model the typical pattern of passenger flow in an urban area; by applying this model authors were able to compute the probability that an event happened, and measure the impact of the event by analyzing anomalous patterns in passenger flow via Discrete Fourier Transform. An Interactive Voting-based Map Matching Algorithm is used in (Pan *et al.* , 2013) to map GPS trajectories. This mapping is aimed to characterize typical drivers' behaviors and discover abnormal ones. Finally, the authors mine the cause of the anomaly by checking data gathered by social networks.

One of the main issues concerning the analysis of this kind of data is their dimensionality. Many approaches handle it by focusing on specific areas (*hotspots*) whose high concentration of events or samples can summarize the most relevant dynamics occurring in data (Hu *et al.* , 2014). In the literature, urban hotspots are typically divided in two categories: (i) regular and (ii) occasional. Areas comprising many points of interest such as movie theaters, commercial buildings, hospi-

tals, schools, colleges, etc. are prime examples of regular hotspots. Occasional hotspots are those areas where any incident has taken place. An incident is defined as an unexpected event that temporarily disrupts the mobility flow, e.g., car crash, marathon, VIP passing area, etc. However, the most of the studies firstly consider regular hotspots. Li. et al (Li *et al.* , 2012) proposed and developed an improved auto-regressive integrated moving average (ARIMA) for detecting urban mobility hotspots using taxi GPS traces; moreover, the patterns of pick-ups occurring in these city locations are extracted and analyzed. Other works, such as the one from Makrai (Makrai, 2016), provide a statistical approach for the detection of hotspots in New York City by means of a distributed environment. Authors in (Keler & Krisp, 2016) use OPTICS in order to exploit taxi drop-off positions, extracting hotspots from density-connected point clusters. Cluster results are then assigned as daily taxi drop-off hotspots. Recently, Lu et al. (Lu *et al.* , 2016c) developed a monitoring system performing spatiotemporal analysis on taxi trip data to find seasonal hotspots. This result is achieved by using DBSCAN clustering algorithm with pick up and drop-off locations every fixed amount of time. However, due to the complexity of mobility data, the modeling and comparison of their dynamics over timeremain hard to manage and parametrize (Castro *et al.* , 2013).

2.5 Distributed Target Detection with Swarm of UAVs

According to (Senanayake *et al.* , 2016), target search is the discovery of targets located in an unstructured environment, with no prior knowledge about their location and about the obstacles layout. The quality of the process can be measured by minimizing the overall time needed for completing the mission. A target search mission with swarms of robots (agents) can be defined according to (i) the number of targets and agents; (ii) the mobility of targets; (iii) the complexity of the environment; (iv) the prior knowledge about the target; (v) the type of swarm coordination.

Several research has been developed in the field, considering various problem setups when addressing the problem of target search. Such works either describe working systems, or focus on a single specific challenge to be studied. Thus, some systems have been deployed and tested in real-life scenarios, while others remain theoretical approaches. Among the proposed systems some approaches are tailored to suit requirements of a specific kind of event and are therefore domain specific. A recent survey (Zhang & Mehrjerdi, 2013) attempted to discuss the qualitative differences among the approaches using a taxonomy. For the sake of brevity, in this subsection some relevant works in the field are briefly summarized, discussing differences and similarities with our approach.

Table 2.1 shows a structured view of the techniques for coordinating swarms of UAVs according to (Gazi & Passino, 2011). Specifically, the *formation control* fo-

cuses on the spatial arrangement of the UAVs, and it can be distinguished as *Virtual Structure*, *Leader Following* or *Flocking*. *Virtual Structure* represents the formation by means of the reference points of a basic geometric structure (Krick *et al.* , 2008). It is useful when constraints on the mutual distance between UAVs should be managed (Zhang & Mehrjerdi, 2013). However, when obstacles are present, complex and multiple maneuvers are needed to recover the predefined arrangement. With *Leader Following* the drones keep a relative distance and angle from a leader, which has a higher awareness of the search field or a better equipment (Qiu & Duan, 2017). However, the leader can easily become a point of failure of the entire swarm due to its control (Dang & Horn, 2015). A more flexible schema is *Flocking*, which is still based on mutual distances among drones, but with a dynamic rearrangement capability based on: (i) maintaining the heading of flockmates, (ii) avoiding collisions and (iii) attempting to remain in range between them (Reynolds, 1987a), (Hauert *et al.* , 2011), (Cimino *et al.* , 2015c), and (De Benedetti *et al.* , 2017).

A *Trajectory Following* strategy aims at determining the path toward the targets by defining the swarm collective motion explicitly (Viguria *et al.* , 2010), or by using macroscopic rules such as the *artificial potential* (Qu *et al.* , 2015),(Luo & Duan, 2013). However, it requires the knowledge of the scenario layout in order to steer the swarm. Thus, it is not suitable for our research purpose. Similarly, with approaches based on *Swarm Aggregation* the ultimate goal of aggregation is to form a single aggregate that contains all the agents in a specific known location (Wilson *et al.* , 2012), (Soysal & Şahin, 2006).

Stigmergy is a mechanism aimed at achieving swarm coordination in collective tasks, such as *Social Foraging*. Specifically, each agent of the swarm releases a pheromone mark in a shared medium while performing a given action. The pheromone acts as a stimulus, i.e. other agents that perceive it will follow the pheromone trail toward the area where the target is located (Chi *et al.* , 2014), (Sauter *et al.* , 2005b), (Cimino *et al.* , 2016).

In the *Distributed Agreement or Consensus*, the agreement in the swarm is achieved if the variables of interest of all drones converge to the same value. These variables can represent the state of the agent, e.g. its heading or its current behavior (Kurdi *et al.* , 2016), or the state of the overall scenario, e.g. the nature of the targets (Han & Chen, 2014). A consensus-based approach for collaborative UAVs search provides an evident advantage when dealing with uncertainty about the target detection process (Sharma *et al.* , 2010). However, this situation is out of the scope of this paper.

The *Swarm Tracking* problem is based on the generalized pursuit-evasion problem for multi-agent systems. It assumes the existence of a moving target which is trying to evade the capture by the agents (Lee *et al.* , 2010), (Cui *et al.* , 2016), (Minaeian *et al.* , 2016). This research does not fit our requirements, since we assume that targets are static.

Table 2.1 summarizes the above presented taxonomy, which is provided according to (Yan *et al.* , 2013), (Senanayake *et al.* , 2016), and (Gazi & Passino, 2011).

Table 2.1: Characterization and qualitative comparison of the approaches aimed at coordinating a swarm of UAVs.

APPROACH	MAIN ADVANTAGES	MAIN DISADVANTAGES
Formation Control: Virtual Structure (Krick *et al.* , 2008) (Zhang & Mehrjerdi, 2013)	Consistent swarm performance in clear search space	Complex management of obstacles. Limited flexibility and scalability
Formation Control: Leader Following (Qiu & Duan, 2017) (Dang & Horn, 2015)	Flexible formations. No apriori knowledge of the search field layout	Single point of failure (the leader). No feedback from the followers to the leader to adjust the formation
Formation Control: Flocking (Hauert *et al.* , 2011) (De Benedetti *et al.* , 2017) (Cimino *et al.* , 2015c)	Dynamic formations rearrangement. No apriori knowledge of the search field layout	Poor predictability of the overall swarm behavior
Trajectory Following and Artificial Potential (Qu *et al.* , 2015) (Luo & Duan, 2013)	Efficient building of the search path	Easy to fall into local minima. Path or potential fields are explicitly defined according to search field layout
Swarm Aggregation (Wilson *et al.* , 2012) (Soysal & Şahin, 2006)	Predictable swarm performance	Requires the apriori knowledge of the targets positions. May require the synchronization among agents
Stigmergy (Sauter *et al.* , 2005b) (Chi *et al.* , 2014) (Cimino *et al.* , 2016)	No apriori knowledge of the search field layout	Poor predictability of the overall swarm behavior
Distributed Consensus (Kurdi *et al.* , 2016) (Sharma *et al.* , 2010)	Handle uncertainty in target detection	High communication rate
Swarm Tracking (Minaeian *et al.* , 2016) (Lee *et al.* , 2010)	Handle mobile target	Complex scalability, especially with multiple targets

According to the characterization provided in this literature review, flocking and stigmergy should properly fit an autonomous targets search in unknown environment, thus further investigation in this direction is addressed.

2.6 Swarm of Robot Coordination

The potential of robot swarms has been acknowledged as one of the ten robotics grand challenges for the next 5-10 years that will have notable socioeconomic impact (Yang *et al.* , 2018). Indeed, nowadays swarms of robots are starting to become an important part of industrial activities such as warehouse logistics (D'Andrea, 2012; Liu *et al.* , 2017). However, many important aspects of robot swarms which would need to be considered in realistic deployments are either underexplored or neglected (Castello *et al.* , 2016). The are two main reasons (Yang *et al.* , 2018): The first one is due to scalability issues. As the size of the swarm increases, the number of interactions grows exponentially, and tools to analyze them more precisely are lacking. Second, and foremost, because there is no consensus on how to design systems that include perception, action, and communication among large groups of robots. Currently, one of the main study areas of swarm robotics systems is on foraging behaviors. Foraging is the set of actions to explore and collect objects or information scattered in an environment. Foraging tasks can be projected to more complicated problems (e.g., exploration vs exploitation trade-offs, consumer and producer models, etc.), and currently they are one of the main benchmarks to evaluate swarm robotics systems (Lu *et al.* , 2018). Applications of swarm robotics foraging are wide-ranging from carrying objects and tokens to specific target locations (Dorigo *et al.* , 2005; Castello *et al.* , 2016) to rescuing natural disaster victims (Payton *et al.* , 2005). The similarity among these examples is that robots always leave from and return to a common central location (e.g., nest, headquarters, etc.). Central Place Foraging (CPF), as it is called, is in fact the most studied foraging approach in the field (Winfield, 2009; Brambilla *et al.* , 2013b; Castello *et al.* , 2016). Although CPF provides good results in simple missions and indoor scenarios, the overall performance (e.g., tokens collected, packages delivered, etc.) of the swarm decreases when the size of the scenario or the number of robots grow (Lu *et al.* , 2016b; Zia *et al.* , 2017). Due to this phenomenon, CPF-based systems might be inadequate for deployment in larger, more dynamic areas such as big cities or vast urban environments (Salvini, 2018). However, one possible solution to this issue could be Multiple Place Foraging (MPF). MPF is a bio-inspired problem (Schmolke, 2009) that relies on multiple nests rather than one central depot. Nests are scattered across the area and each robot inside the swarm can change its correspondent nest depending on its location and energy status (Lu *et al.* , 2016a, 2018). One of the main coordination mechanisms to steer the swarm is stigmergy (Zedadra *et al.* , 2015). With stigmergy, pheromones are released in a shared environment and are used as a type of indirect communication. Theoretically, CPF and MPF have a very similar set of parameters (Lu *et al.* , 2016a,b). However, MPF-based research has not been conducted outside simplistic scenarios (Halasz *et al.* , 2007; Berman *et al.* , 2008) and therefore further

studies are required to test its feasibility.

Chapter 3

Design

The computational approach aimed at matching time series presented in this study is called *Stigmergic Receptive Field* (SRF). This is designed to provide a general purpose measure of similarity for spatio-temporal dynamics occurring in the analyzed data. Figure 3.1 illustrates the architecture of an SRF, which is made of six main subsystems, i.e., clumping, marking, trailing, similarity, activation, and adaptation.

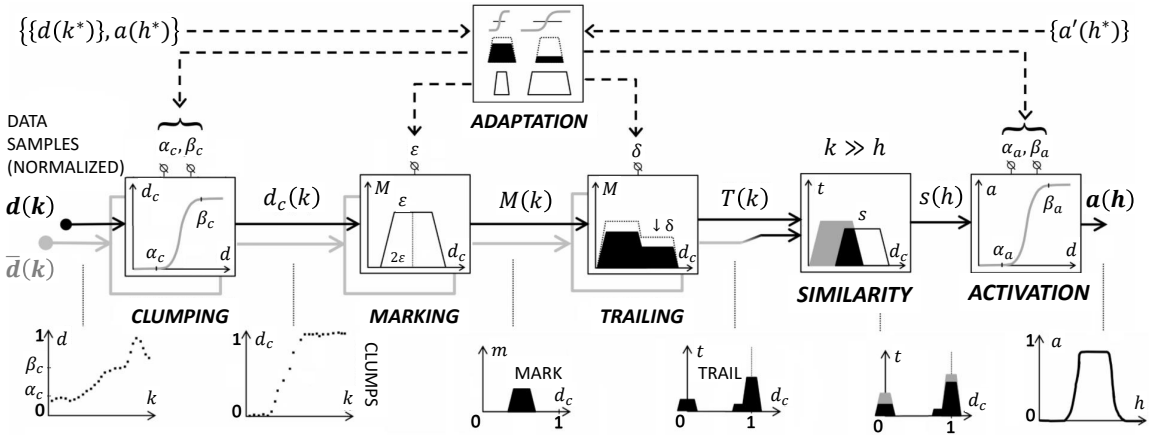


Figure 3.1: Architecture of a Stigmergic Receptive Field.

Input samples are firstly treated by the Clumping process. It is aimed to reduce microfluctuation in data while highlighting the dynamics occurring in correspondence of relevant information levels. In correspondence of each clumped sample, a mark (i.e. a trapezoid) is released in a bidimensional virtual environment by the Marking process. The Trailing process aggregates the marks in proximity with each other in the trail, whose intensity decreases with time. Thus, only mark subsequently deposited in proximity with each other's results in a consistent trail, which can be considered as a short-term memory summarization of the spatiotemporal dynamism occurring in data. Both time series $(d(k), \bar{d}(k))$ provided to the SRF undergo these processing stages independently until the Similarity process compares

their trails. Finally, the Activation process enhances only relevant similarity values whereas decrease the irrelevant ones.

The proposed mechanism works if structural parameters are correctly adapted for the given application context. Determining such correct parameters is not a simple task since different indicators may have different dynamics. For this purpose, we adopt a tuning mechanism based on the DE. In the next subsections, each module and subsystem is precisely described, by using a pilot data sample. In the following paragraph each of these modules is detailed.

3.1 Clumping, Unbiasing, Activation

At the input/output interfaces of a module, the input/output data may be treated with an S-shaped function. This is used for a better efficiency and alignment of the processing layers. Moreover, it let an input signal to reach a certain level before a processing layer passes it to the next layer, and allows a better distinction of the critical phenomena during unfolding events, with a better detection of the progressing levels. In Fig. 3.2 the effect (bold line) of the S-shape function on a real value time series (dotted line) is shown; here, every sample greater (lower) than a given threshold (i.e. 0.5) is enhanced (decreased).

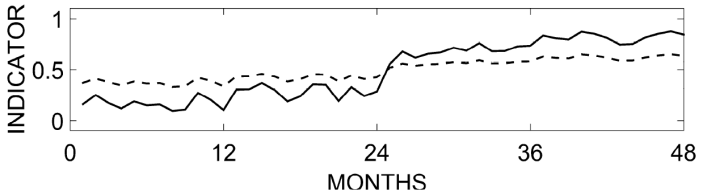


Figure 3.2: Example of application (bold line) of an S-shape function to an time series (dotted line).

The S-shaped function can be implemented by means of 2 functions:

- An S-shaped function (Eq. 3.1), parametrized with the parameters α and β by means of which the input values smaller (larger) than $(\beta - \alpha)/2$ are lowered (raised); values smaller (larger) than α (β) assume the minimum (maximum)

value, i.e., 0 (1).

$$\begin{cases} 0, & x \leq \alpha \quad (3.1) \\ 2 \cdot \dfrac{(x-\alpha)^2}{(\beta-\alpha)^2}, & \alpha \leq x \leq \dfrac{(\alpha+\beta)}{2} \quad (3.2) \\ 1 - 2 \cdot \dfrac{(x-\alpha)^2}{(\beta-\alpha)^2}, & \dfrac{(\alpha+\beta)}{2} \leq x \leq \beta \quad (3.3) \\ 1, & \geq \beta \quad (3.4) \\ & (3.5) \end{cases}$$

- A Sigmoidal function (Eq. 3.6) in which α β correspond to the sigmoid inflection points (α_a,β_a).

$$f(x, \alpha_a, \beta_a) = \frac{1}{(1 + e^{-\alpha_a(x-\beta_a)})} \quad (3.6)$$

According to the phase of the data processing in which this module is used, its aim, and so its name, will differ:

- *Clumping*: it is used at the initial phase of the data processing in order to undergo the input data to a soft discretization with respect of a number of levels; the levels correspond to the relevant ranges of values for the analysis (e.g. noise/relevant/outlier or low/medium/high). The number of S-shaped functions here corresponds to the number of levels.
- *Unbiasing*: it can be used in between of the data processing phases, in order to treat the semi-processed data and enhance (neglect) the relevant (insignificant) values, i.e. higher (lower) than a given threshold.
- *Activation*: it is used at the end of the data processing phases, in order to enhance (neglect) the relevant (insignificant) output values, i.e. higher (lower) than a given threshold.

3.2 Marking and Trailing

The *Marking* takes a clumped sample $d_c(k)$ of a normalized input time series $d(k)$, and releases a mark in a marking space whose codomain is called intensity (Cimino *et al.* , 2015b). The mark has five structural attributes: the center position p_c, the intensity I, the mark top, and bottom width, and the mark evaporation δ. The mark shape is a trapezoid, whose parameters allows to generalize both a rectangle (i.e. a mark with a uniform surface) and a triangle (i.e. a mark with the maximum intensity only in the position of the deposits). In Fig. 3.3 (a) shows, in thick line, we

depict the mark released in correspondence of a sample, with top width equal to zero (i.e. a triangle) and a bottom width equal to ϵ. The *Trailing* process handles the aggregation of multiple marks (i.e. the summation of their intensities) in the trail while the evaporation keeps decreasing the contribution (i.e. the intensity) of each subsequent deposit (Fig. 3.3 (b)). The parametrization of the marks enables their aggregation in the trail, given their spatial proximity (i.e. the mark width) and deposit frequency (i.e. the mark evaporation), providing the analysis with an information spatiotemporal granulation.

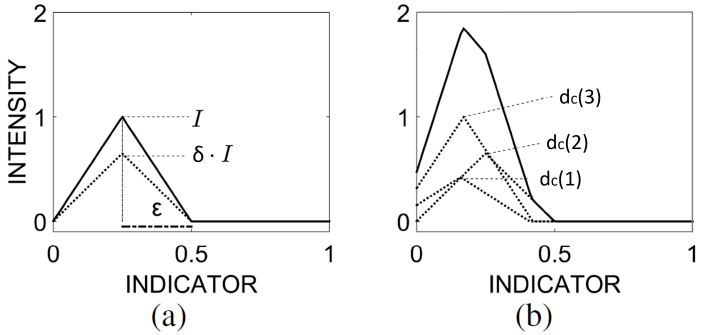


Figure 3.3: Example of a mark before and after the evaporation (a), and a trail obtained by depositing 3 marks (b).

3.3 Similarity

The *Similarity* process compares two trails (T_1, T_2) using the Jaccard coefficient (Peng *et al.* , 2016) as defined in Eq. 3.7. The resulting value ranges from 1 (identical trails) to 0 (completely different trails).

$$S = \frac{|T_1 \cap T_2|}{|T_1 \cup T_2|} \tag{3.7}$$

We adopt the notions of intersection and union from the fuzzy logic, thus the intersection of the two trails is their element-wise minimum, whereas their union is represented by the array of their element-wise maximum (Fig. 3.4).

3.4 Adaptation

In order to have an effective similarity computation, i.e. computing it in accordance with the features we are looking for in the time series, the SRF should be properly parameterized. For example, low trail evaporation causes early activation, whereas

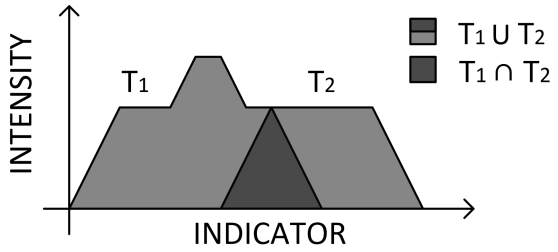


Figure 3.4: The comparison of 2 trails by means of the Jaccard similarity.

high trail evaporation generates a trail consisting of the latest marks only, and preventing trail reinforcement. Specifically, the SRF parameters are: (i) the clumping inflection points $\alpha_{c1}, \beta_{c1}, ..., \alpha_{cn}, \beta_{cn}$; (ii) the mark top and bottom widths ϵ_t, ϵ_b; (iii) the evaporation δ; and (iv) the activation inflection points α_a, β_a. The SRF parameters are adjusted by the *Adaptation* process. It uses the Differential Evolution (DE) algorithm (Cimino *et al.* , 2015b), in order to minimize Mean Square Error (MSE, Eq. 3.8), which is computed as the difference between desired ($a'(h^*)$) and actual ($a(h^*)$) output value on a training set of N labeled couples of time series ($\{a(k^*), a'(h^*)\}$).

$$Fitness = \frac{\sum_{i=1}^{N}(|a(i^*) - a'(i^*)|^2)}{N} \tag{3.8}$$

3.5 From the SRF to the Stigmergic Perceptron

Let us suppose to have a pure form time series which embodies a behavioral class (i.e., an archetype). The SRF can detect this specific behavior in the actual time series, by processing it together with the archetype, once the SRF has been specialized to unfold the features of the archetype itself (via adaptation). A set of SRFs aimed to recognize these archetypes can be arranged into a connectionist topology, obtaining a *Stigmergic Perceptron* (SP) (Alfeo *et al.* , 2018). By forming a linear combination of the SRFs outcomes, the Stigmergic Perceptron provides an assessment of the current behavior of the input time series among all the classes provided.

Specifically, the output of the SP is calculated as the average of the SRFs enumerations (represented as 1-to-N where N is the number of archetypes in our application case) weighted by their output similarities (Eq. 3.9). The SP output is called activity level and is defined between 0 and N (i.e., the number of archetypes).

$$ActivityLevel = \frac{\sum_{i=1}^{N}(a(i) * i)}{\sum_{i=1}^{N}(a(i))} \tag{3.9}$$

In order to prevent multiple activations of SRFs in the same SP, in this configuration their Adaptation process is a two phases procedure: (i) the Global Training

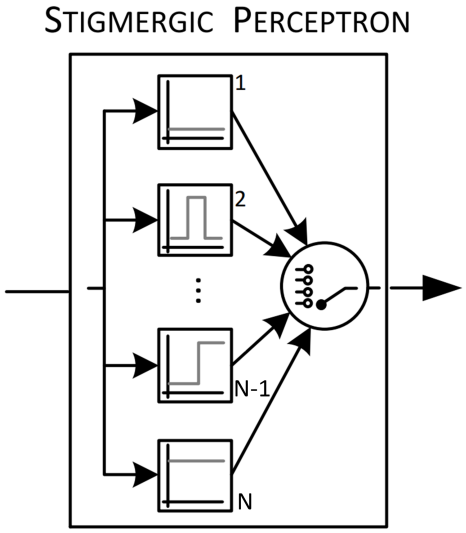


Figure 3.5: The architecture of a Stigmergic Perceptron.

phase is aimed to provide a suitable interval for each SRFs parameters according to their sensitivity; specifically, the interval for the evaporation rate, that is the most sensitive SRF parameter, is determined considering the narrowest interval including the fitness values above the 90th percentile, while the intervals for the other parameters can be provided according to the application domain constraints; (ii) the Local Training phase is aimed to find the optimal values for every parameter and each SRF by exploiting their Adaptation process and the interval determined in the Global phase; the training set for each SRF is made by half signals belonging to its behavioral class, and half belonging to the behavioral classes of adjacent SRFs.

Assuming to split a long time series according to a time window and pass them to a properly trained SP, this will transform the sequence of shorter time series in a new time series of archetypal similarity, that can be considered as a higher level characterization of the dynamics occurring in the time series.

Being a time series, even the SP output can be treated with the SRF providing a higher level similarity that can be profitably exploited by clustering techniques (see Chapter 4).

3.6 Bidimensional Stigmergy

What has been presented so far allows computational stigmergy to work with one-dimensional dynamics (real values time series). However, the same mechanism of aggregation and evaporation can be proposed in the multidimensional (for example in the two-dimensional) space in order to detect or enhance the dynamics of space-time on areas or planes.

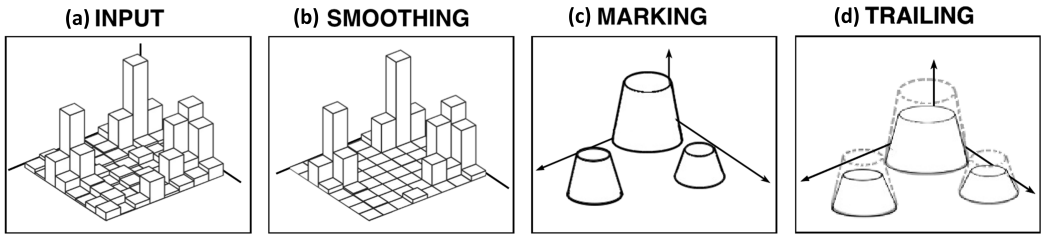


Figure 3.6: The stigmergic aggregation in a bidimensional space.

During each step of the analysis, samples corresponding to a given time slot are provided to the system and processed by a four-stages procedure. At the beginning, the Smoothing process removes irrelevant samples' values and highlights significant ones (Fig.3.6b)), by treating them with a sigmoidal function (Eq. 3.6). The marking process releases a mark in a tridimensional spatial environment in correspondence of each smoothed sample position (Fig.3.6(c)). Each mark is defined by a truncated cone with a given width and an intensity (height) equal to the sample value. The marks aggregate form the trail, whose intensity is subject to evaporation, i.e., the trail intensity is decreased by a constant value δ at each step of the analysis (Fig.3.6(d)). Eq. 4.12 describes the trail T at time instant i.

$$T_i = (T_{i-1} - \delta) + Mark_i \tag{3.10}$$

This approach produces a two-dimensional stigmergic trail which, just as in the one-dimensional case, unfold and summarize the spatiotemporal dense pattern in data. Indeed, it has been used to unfold hotspots in taxi activity and characterize the mobility of group of users (see Chapter 4).

A completely different application based on two-dimensional stigmergy is the exploration of unknown environments via swarms of drones. Here, the stigmergic trail acts as a short-term memory of the recently found targets in the search area. By following the trail other drones are pushed toward the part of the scenario with the higher probability of target presence.

During the Ph.D. many arrangements the above presented architectural modules has been tested, with different aim according to the application considered.

In order to highlight the benefits provided by our approach, the following chapter summarizes those applications and the obtained results.

Chapter 4

Applications

In this Chapter each application of the methodology presented in the former chapter is detailed. The arrangement of the applications is chosen according to their field. Specifically, we start with the problem of assessment of trend in technological indicators, it follows the applications concerning the behavioral analysis via physiological signals, then the applications targeting the event detection in urban mobility, and finally the application on the coordination of swarm of robots and UAVs.

4.1 Assessing the Trend of Innovation Indicators

Problem Statement

After years of economic crisis and the resulting reduction of resources available for research and development investments, Smart Specialization has immediately become a very relevant concept to get these two questions answered (Bellini, 2016). It represents an important chance for a progressive economical restart. In order to develop a policy prioritization logic to foster regional growth is important to have a deep knowledge of the potential evolutionary pathways related with the existing dynamics and the structures at regional level (McCann & Ortega-Argilés, 2015).

With this in mind, each region should initiate this process using knowledge-based sectors as points of reference where it already has a coherent "critical mass" or, at least, capabilities that refer to potential exploitable through right and targeted investment. On this line, the European Commission has established a program labelled 'Smart Specialization', consisting in a set of policies and guidelines aimed to promote the efficient and effective use of public investment in research and development (R & D). Smart Specialization is defined as "an industrial and innovation framework for regional economies that aims to illustrate how public policies, framework conditions, but especially R & D and innovation investment policies can influence economic, scientific and technological specialization of a region and con-

sequently its productivity, competitiveness and economic growth path. It is a logical continuation in the process of deepening, diversifying and specializing of more general innovation strategies, taking into account regional specificities and inter-regional aspects, and thus a possible way to help advanced economies (as well as emerging economies) to restart economic growth by leveraging innovation led / knowledge-based investments in regions" (Perianez-Forte & Cervantes, 2013). The long term aim of this work is exploring whether - and to what extent - different policies of 'technological specialization' and 'technological diversification' pays off in term of wealth creation at regional level. Then, we want to provide policy makers with computerized support in the analysis of innovation-relevant trends (Jin *et al.* , 2014).

Proposed Approach

It is known that diversification and specialization of patents applied in a region measure the Innovation of the region itself. Thus, it is important for a policy maker to analyse trends of innovation, to properly address the investments. Such trends cannot be directly sensed nor associated to the innovation. For this purpose, there are three important indicators which quantify innovation: specialization (S), related variety (R), and unrelated variety (U). The study of such Trends by the Policy Maker is fundamental to recognize scenarios of interest, i.e., the ways in which special situations may develop. Example of scenarios of interest are: (i) R or U decreases, while S increases; (ii) R or U decreases, while S is stable; (iii) R or U increases, while S is stable;(iv) R or U increases, while S increases. The problem is to detect variations of an indicator in terms of increase, decrease or stability.

To properly move into that direction, we start looking at this problem by analysing the trends of the aforementioned indicators for 268 EU-27 regions over 35 technological domains in the period 1990-2012, in order to obtain a model that can efficiently recognize significant event. Indeed, according to the region history and characterization the same indicator behavior may be significant in one region and not in another. For this purpose, we have designed an approach based on SRF.

In Fig. 4.1 we show the architecture of the approach used to analyze those trend. Specifically, at the input/output interfaces of each subsystem, an unbiasing module is used for a better efficiency and alignment of the processing layers. This lets an input signal to reach a certain level before a processing layer passes it to the next layer, and allows a better distinction of the critical phenomena during unfolding events, with a better detection of the progressing levels. The marking subsystem transforms input data into marks, whereas the trailing subsystem aggregates and evaporates marks as a track in the stigmergic space. The prototyping subsystem provides a simplified version of the track. It is a vehicle of abstraction, leading to the emergence of high-level information. The dissimilarity subsystem evaluates the difference be-

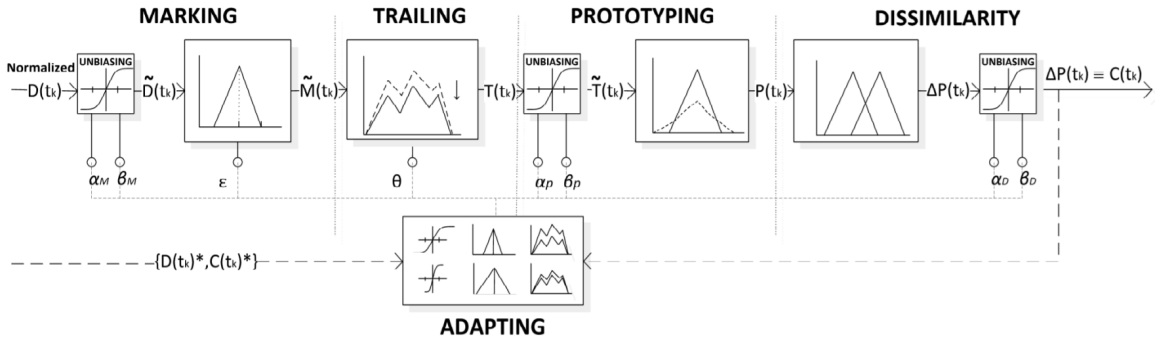


Figure 4.1: Architectural overview of our data analysis system based on Stigmergy.

tween consecutive prototypes in order to extract trend information of the indicators. The proposed mechanism works if structural parameters are correctly adapted for the given application context. Determining such correct parameters is not a simple task since different indicators may have different dynamics. For this purpose, we adopt a tuning mechanism based on the DE (version DE/1/rand-to-best/bin).

Obtained Results

The case study is based on a data set that contains the three annual indicators S, U, R, monitored for 15 years for 200 European Regions. The dataset contains 9000 samples. For each year we grouped regions via k-nearest neighbour algorithm. For each group we computed the annual mean μ and the standard deviation σ. We also determined that the resulting indicator samples are well-modelled by a normal distribution, using a graphical normality test. Finally, monthly samples have been derived considering normal distribution with mean and variance $\mu/12$ and $\sigma^2/12$, respectively. To choose the best value of CR and F, we first performed trials with CR in [0.3, 0.6, 0.9] and F in [0.4, 0.6, 0.8]. For each experiment, 5 trials have been carried out, by using the 20% of the dataset as a training set, and the remaining 80% as a testing set. We also determined that the resulting MSE samples are well-modelled by a normal distribution, using a graphical normality test. Hence, we calculated the 95% confidence intervals. Table 4.1 shows the results in the form "mean $\pm$ confidence interval". The best performance has been with CR=0.6 and F=0.6.

In order to assess the effectiveness of the approach, we adopted a 5-fold cross-validation. Indeed, each evaluation is also dependent on the data points, which end up in the training and test sets. For each trial, the training and test sets consist, respectively, of randomly extracted 20% and 80% of the original data. We carried out each trial 5 times. Table 4.2 summarizes, for indicator U, the results in terms of mean and standard deviation of the MSE for each trial. The low values of the MSE, for all

Table 4.1: 95% confidence interval of the MSE for the best setting of differential weight (F) and crossover rate (CR).

		F		
		0.4	0.6	0.8
CR	0.3	0.022 ± 0.0002	0.012 ± 0.00002	0.018 ± 0.0002
	0.6	0.019 ± 0.0002	**0.009± 0.0001**	0.011 ± 0.00001
	0.9	0.014 ± 0.00006	0.013 ± 0.00007	0.013 ± 0.00007

trials and for both training and testing sets, highlight the effectiveness of the system in terms of both performance and generalization properties. We replicated the same experiments and achieved similar performances for the other indicators. Finally, to highlight the great benefits of the adaptation subsystem, we also computed the MSE for the worst case of Table 4.1 (i.e., Trial 5), by using manual adaptation: this results in an MSE of 0.106, which is very higher than 0.022.

Table 4.2: MSE (mean $\pm$ confidence interval) for each trial extracted via 5-fold cross-validation, averaged over 5 repetitions.

Trial	**Training Set**	**Testing Set**
1	0.011 ± 0.010	0.018 ± 0.004
2	0.010 ± 0.010	0.020 ± 0.003
3	0.009 ± 0.006	0.020 ± 0.008
4	0.008 ± 0.008	0.020 ± 0.005
5	0.010 ± 0.007	0.022 ± 0.008

The experimental results show that using 20% of the data set as training set to recognize trends ranging from -1 to 1, the system achieved an MSE of 0.02. Thus, the proposed system appears to be profitability used to detect the trends of three different patent-based indicators.

4.2 Assessment of Sleep Quality

Problem Statement

One of the most important markers of a healthy lifestyle is represented by the quality and quantity of sleep. These factors directly affect the waking life, including productivity, emotional balance, creativity, physical vitality, and the general personal health. Indeed, poor long-term sleep patterns can lead to a wide range of health related problems, such as, high-blood pressure, high stress, anxiety, diabetes and depression (Chen *et al.* , 2013). In this context, the monitoring of sleep patterns becomes of major importance for various reasons, such as the detection and treatment of sleep disorders, the assessment of the effect of different medical conditions or medications on the sleep quality, and the assessment of mortality risks associated with sleeping patterns in adults and children (Metsis *et al.* , 2014). Traditionally, the Polysomnographic (PSG) recordings have been widely used in order to infer the sleep quality (Åkerstedt *et al.* , 1994). In this regard, the quality measure is usually captured with self-reports via paper-based surveys and diaries that, although being difficult and tedious to be collected, represent a reliable source of information (Bootzin & Engle-Friedman, 1981). Nevertheless, sleep diaries can be affected by cognitive bias related to the subject's sleep perception.

In recent years, because of the development of ubiquitous technology in health care, the research effort involving non-invasive sensors to assess and report sleep patterns is actively progressing. A relevant source of information on sleeping is represented by motion data coming from worn inertial sensors (i.e., accelerometers) embedded in smartphones or wristbands (Ong & Gillespie, 2016).

While commercially available activity trackers based on wearable devices can be considered valid for measuring sleep phases and heart rate (HR) during sleep (de Zambotti *et al.* , 2016), there exist many sleep analysis algorithms that, exploiting smartphone sensors only, have not been validated by scientific literature or studies (Ong & Gillespie, 2016). This is even more evident when considering long-term analysis. Many people track their sleep through mobile and wearable technology, together with contextual information that may influence sleep quality, like exercise, diet, and stress. However, there is limited support to help people make sense of this wealth of data, i.e., to explore the relationship between sleep data and contextual data. In (Liang *et al.* , 2016), authors try to bridge the gap between sleep-tracking and sense-making through the design of a web-based tool that helps individuals understand sleep quality. However, an automatic tool able to monitor the sleep over the long period and give a user-tailored quality measure is still missing.

Indeed, most sleep scoring algorithms provide a threshold based analysis of subject's activeness during the whole night. Unfortunately, due to peculiarities of each

subject's sleep, same thresholds cannot be effective for any user nor exhaustive for a sleep behavior analysis. As an example, same rem sleep ratio values can be obtained with nights characterized by different number and duration of Rem-NonRem cycles, which is an important behavioral difference. In contrast with more traditional scoring algorithms, novel machine learning approaches can provide greater accuracy due to their ability to generate nonlinear classification borders (Sadeh, 2011); moreover, they can improve the objectivity and reliability of the observations (Khaleghi *et al.* , 2013). On the other hand, the use of machine learning techniques often requires a careful tuning of their structural parameters, which can be provided by employing an expert in the field or even via brute-force search (Snoek *et al.* , 2012).

Proposed Approach

In this work, we present an automatic tool for monitoring sleep behavior that uses a commercially-available smartwatch, in order to sample heart rate and wrist inertial data, and a novel detection technique based on stigmergic receptive fields (SRFs). The SRF transforms an input time series into a stigmergic trail and provides a (dis-) similarity measure against another signal trail. The dissimilarity measure is parametrically adapted according to sleep quality annotations on a set of reference nights. In order to prove the adaptation capability of our approach we test it on 7 different subjects (from now on referred as subject A, B, ..., G).

The proposed solution employs a smartwatch to gather subjects' physiological and inertial data, i.e. heartbeat rate and wrist acceleration. The smartwatch embeds a heart rate monitor based on an optical sensor to detect peaks in blood flow. Thereafter it computes the heart rate over an interval of time established by the constructor, i.e. 1 second. The accelerometer embedded in the smartwatch measures both static (due to gravity) and dynamic acceleration on the three axes, sampling them at 10 Hz. These signals are splitted in time windows (1 minute each, partially overlapping) and normalized between 0 and 1. Moreover, the acceleration signals are summarized by the standard deviation of the acceleration magnitude, by now on referred only as acceleration. Both acceleration and heart rate signals are processed in order to derive an assessment of subjects sleep quality, which to be compared with respect to a sleep quality ground truth.

The ground truth is obtained from early sleep quality evaluation, provided by the subject as a Perceived Sleep Quality (PSQ) score annotated on a sleep diary. PSQ is scored as Normal (N) or Abnormal (A). Due to possible human misperception of sleep quality, we support the ground truth extraction from the sleep diary using a Sleep Stage Estimator (SSE): a software aimed to analyze user sleeping behavior based on physiological parameters (Bernardeschi *et al.* , 2016). The SSE provides a Computed Sleep Quality (CSQ) score based on an estimation of wake, REM, and NREM stages and their occurrence during the night.

Finally, the resulting CSQ and PSQ scores are compared: any night log whose computed score matches the perceived sleep quality becomes an entry of the ground truth nights set. System assessment performance will be evaluated using signals collected during these nights, in order to assess sleep quality. More in details, the assessment error will be computed as the mean square distance between sleep quality assessment and PSQ for each night.

Let us consider a single time series. What is actually interesting is not the continuous variation of the raw samples over time, but the transition from one type of behavior to another. For example, from bradycardia to tachycardia or from still wrist to moving wrist. Each type of behavior should be general and reusable for a broad class of subjects. More formally, each type is called archetype and is a pure form time series fragment representing a behavioral type of activity potentially occurring in a given time window. Thus, in this work we exploit the Stigmergic Perceptron to assess the current signal behavior (among the one provided as archetype) and a futher SRF to distinguish the regular from the anomalous collection of behaviors.

More specifically, Figures 4.2 and 4.3 show ten and six archetypes describing heart rate and arm motion. Here, ordered by increasing intensity, we present each archetype together with the corresponding stigmergic trail. For the heart rate signal we have: (Figure 4.2a) Bradycardia, that represents the lower heart rate activity level, which is exhibited during early deeper NonRem sleep phase; (Figure 4.2b) Episodic Burst, which depicts a single spike of the heart rate activity, a brief and sudden increase of the heart rate; (Figure 4.2c) Lowering, which usually characterize the deepen of the sleep in which the progressive drop of heart rate activity occurs; (Figure 4.2d) Irregular Bradycardia, instead, represents mostly high but irregular heart rate activity characterizing full REM sleep phase; (Figure 4.2e) False Lowering and (Figure 4.2f) False Rising depict long-lasting irregularities in heart rate behaviour; (Figure 4.2g)Irregular Tachycardia, that represents mostly low but irregular heart rate activity characterizing early REM sleep phase; (Figure 4.2h) Rising, which represents the progressive increase of heart rate activity characterizing the transition to REM sleep phase; (Figure 4.2i) Episodic Drop, that represents a single brief drop of the heart rate activity; (Figure 4.2l) Tachycardia, it represent the higher heart rate activity.

On the other hand for wrist motion we have: (Figure 4.3a) Still, which represents the complete paralysis characterizing REM sleep phase; (Figure 4.3b) Short Move, that depicts a brief and isolated movement; (Figure 4.3c) Intermittent Motion, which represents the occurrence of some movements characterizing early sleep phase; (Figure 4.3d) Long Pause that depicts long-lasting stop of wrist motion; (Figure 4.3e) Awakening, representing a sudden and steady increase of wrist motion, usually occurring during transitions from sleep to wake phase; (Figure 4.3f) Awake, in which the high occurrence of wrist motion probes the awakens of the subject.

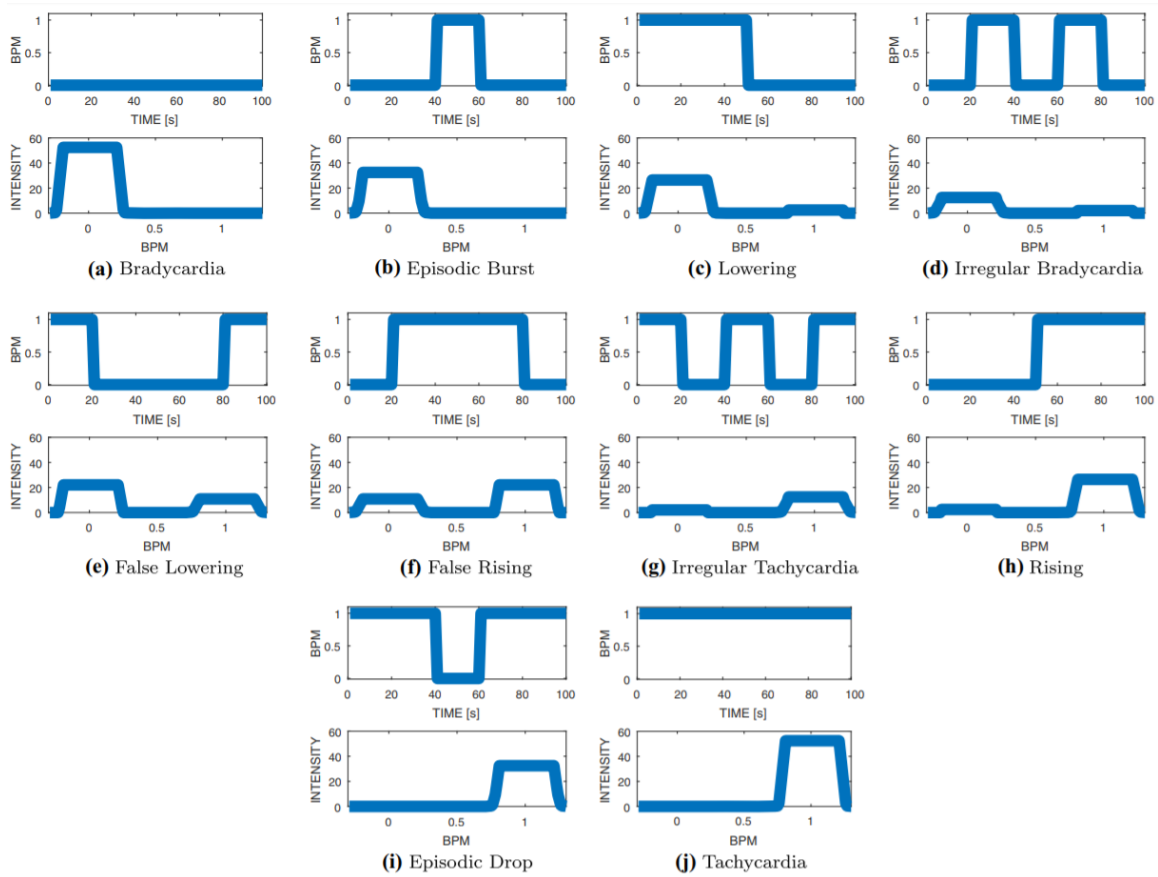


Figure 4.2: Heart rate archetypes.

As an example, in Figure 4.4 we show the outcome of the SP obtained by processing the heart rate time series in the time window corresponding to minute 318 of subject B. The SP processes this signal segment producing a value of 0.7778 as activity level. Indeed, the highest similarity detected by the SP is produced by the eighth SRF (i.e., the “Rising") while a minor activation comes from the seventh one. Thus, obtained activity level measure (i.e. 0.7778) describes the assessment of signal behavior as mostly similar to the one expressed by Rising archetype and secondly by the “Irregular Tachycardia” archetype. Figure 4.4, we show the actual Heart Rate Signals (bold line) together with the signal characterizing the “Rising” Archetype (gray dotted line).

In order to supply a clustering process, the second-level SRF is trained to distinguish similar and dissimilar signals, belonging to the cluster of normal sleep and abnormal sleep. Thus, a training set is provided selecting M (e.g. 5) nights of known sleep quality from the ground truth. The target similarity for a couple of similar or dissimilar nights is assumed to be 1 or 0, respectively.

The similarity between pairs of signal generates a similarity matrix, which is then

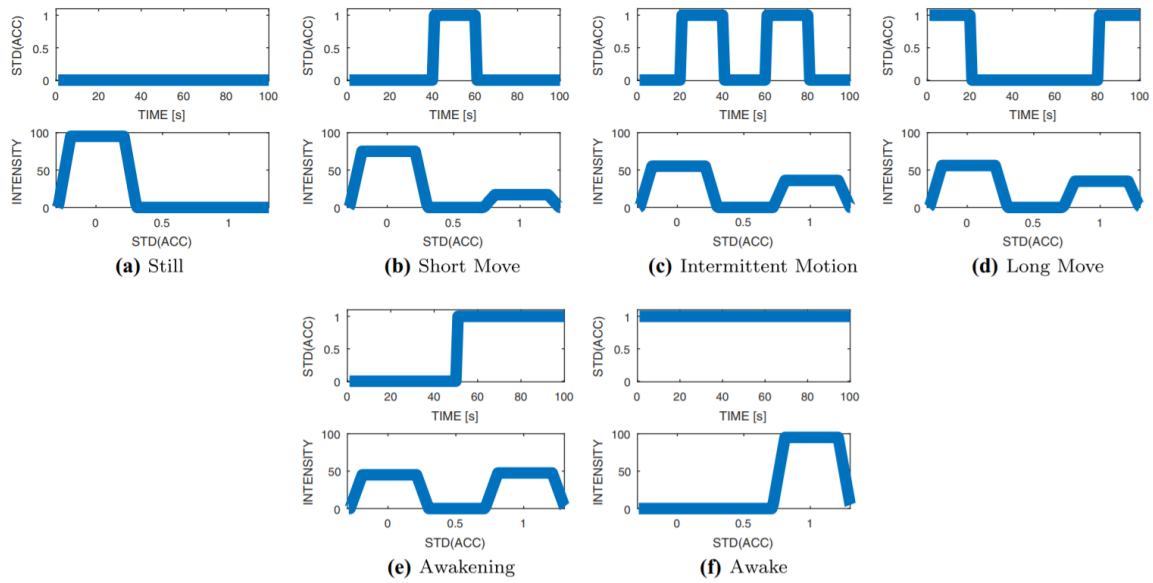


Figure 4.3: Arm motion archetypes.

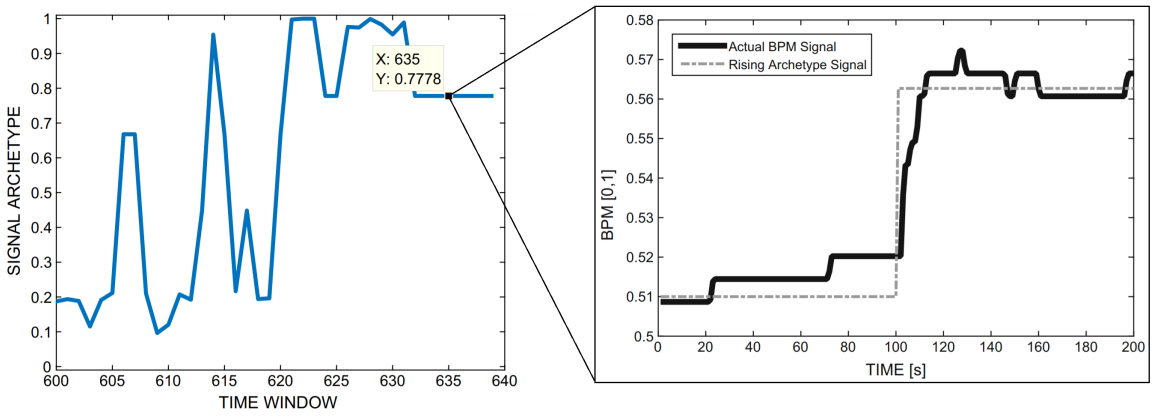


Figure 4.4: The processing of the heart rate time series in the time window corresponding to minute 318 of subject B.

processed by a fuzzy relational clustering technique (Cimino *et al.* , 2006). The fuzzy clustering generates, for each night, a membership degree to each sleep quality. In particular, let us consider the membership to the Normal sleep cluster to generate a normality index for each sleep night.

Finally, the Normality index obtained by Heart Rate and arm motion are combined via a weighted sum, obtaining an overall Sleep Quality Assessment. Sleep Quality Assessment is defined as a real value between 0 (Abnormal) and 1 (Normal). The weights are generated by minimizing the assessment error via Least Square Method (Hager, 2012). An interesting property of the proposed approach is that the provided mapping is not explicitly modeled at design-time but achieved by the

system in order to meet subject behavioral peculiarities.

The overall system architecture is shown in Fig. 4.5.

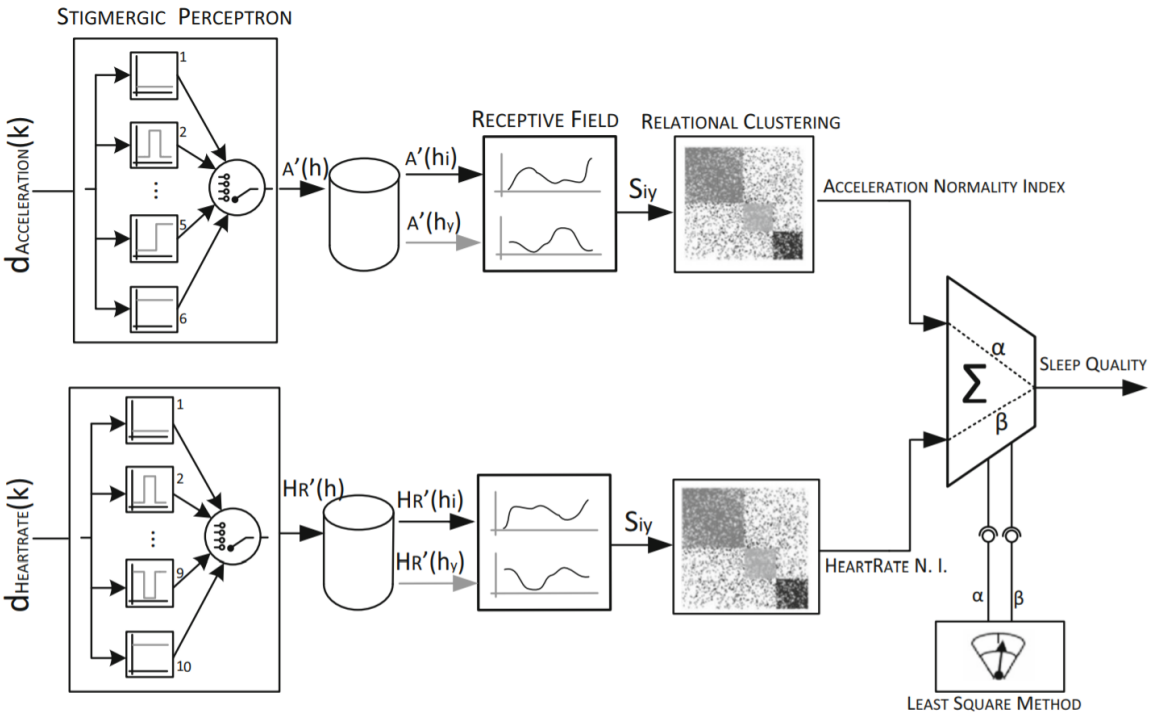


Figure 4.5: The overall signal processing architecture.

Obtained Results

The system analyzed data collected in 20 nights by seven subjects: a man aged 72 (subject A), a 22 year old student (subject B), a woman aged 88 affected by arterial hypertension (subject C), a 36 years old man (subject D), two 21 years old students (subject E and G), and a man aged 30 (subject F). Their perceived sleep quality has been used together with the sleep stage estimator, to extract the sleep quality ground truth (i.e. the nights characterized by a reliably known sleep behavior).

To show the effectiveness of our approach, we present the difference between the target and the calculated normality index for both the DTW distance and the SRF stigmergic similarity, for each subject and for each sleep night. We measure the performance obtained by using the SRF based approach in terms of assessment, by employing the Mean Square Error computed on the Normality Index Error values. Table 4.3 shows the Mean Square Error for both approaches, highlighting how the SRF based approach outperforms the DTW distance.

In order to test our system in terms of percentage of correctly classified nights (i.e. accuracy), we approximate the assessment of each night to its nearest integer

Table 4.3: Mean Square Error in sleep quality assessment via DTW and SRF-based approach.

Approach	Subject							Mean	Std
	A	B	C	D	E	F	G		
DTW	0.247	0.059	0.176	0.138	0.243	0.251	0.196	0.187	0.070
SRF	**0.195**	**0.052**	**0.120**	**0.112**	**0.212**	**0.238**	**0.185**	**0.159**	**0.066**

and compare it with target night classification. To this end, we compare our approach with state of the art classifiers, besides the DTW approach. In particular, we exploited the WEKA data mining software (Hall *et al.* , 2009) to select the best performing classifiers among the four most used families of learning schemes used in literature for similar scenarios: Bayesian frameworks, function-based (e.g., logistic regression, multilayer perceptron, SGD, SMO, etc.), rule-based, and tree-based. We trained the systems using as features the percentage of wake time, the percentage of rem time and the time to sleep, obtained with the previously introduced Sleep Stage Estimator (Bernardeschi *et al.* , 2016). The comparing systems have been selected as the best performing of each classification family using a 10 fold cross-validation over the entire ground truth dataset. With this method, the original sample is randomly partitioned into 10 equal sized subsamples. Of the 10 subsamples, a single subsample is retained as the validation data for testing the model, and the remaining 9 subsamples are used as training data. The cross-validation process is then repeated 10 times (the folds), with each of the 10 subsamples used exactly once as the validation data. The resulting systems chosen for the comparison are:

- Bayesian: *BayesNet* - A Bayes Network learning using various search algorithms and quality measures. This algorithm considers two assumptions: nominal values and no missing values. For estimating the conditional probability tables of network, simple estimator and K2 search algorithm are used to run the BayesNet (Cooper & Herskovits, 1992). This Bayes Network learning algorithm uses a hill climbing algorithm restricted by the order of the variables.

- Functions: *SGD* - A Stochastic Gradient Descent for learning an SVM-based linear model (Hinge loss) (Bottou, 2010). It globally replaces all missing values and transforms nominal attributes into binary ones. It also normalizes all attributes, so the coefficients in the output are based on the normalized data.

- Rules: *DecisionStump* - It builds simple binary decision “stumps” (1-level decision trees) for both numeric and nominal classification problems. It copes with missing values by extending a third branch from the stump (i.e. treating “missing” as a separate attribute value). It is highly capable of predicting the decision with single input (Quinlan, 1986).

Table 4.4: Accuracy obtained by each classifier for each user and their averages.

Subject	**SRF**	**DTW**	**BayesNet**	**SGD**	**Dec. Stump**	**Dec. Table**
10 Fold Cross-validation	-	-	75.73	74.76	77.67	79.61
A	66.66	80	60	53.33	60	60
B	93.75	93.75	100	100	100	100
C	87.5	75	25	25	12.5	25
D	83.33	83.33	43.75	43.75	43.75	43.75
E	68.42	57.89	52.94	70.59	64.71	64.71
F	64.71	58.82	83.33	91.67	83.33	91.67
G	81.82	72.73	78.95	73.68	78.95	84.21
Average	**78.03**	**74.50**	**63.42**	**65.43**	**63.32**	**67.05**

- Trees: *DecisionTable* - Decision tables are one of the simplest machine learning techniques (Kohavi, 1995). Basically, it consists of a hierarchical table in which each entry in the higher level table gets broken down by the values of a pair of additional features to form another table. The DecisionTable approach uses the simplest method of attribute selection: Best First. It searches the space of attributes by greedy hill climbing, augmented with a backtracking facility.

The first row of Table 4.4 shows the obtained performance for each selected classifier with the 10 fold cross-validation method. This validation method considers as overall dataset (i.e. training, test and validation) the nights of all users together. Our aim is, instead, to assess the sleep quality of each particular user independently. To this end, we validated the comparing classifiers using as test set the night of each user and as training set the night of the remaining users. The choice of a 10 cross-fold validation (with respect to a 5 cross-fold) is further motivated by the fact that the number of normal and abnormal nights is unbalanced. Therefore, to include the expected variability in the examples of the "normal2 nights and at least examples of "abnormal" nights, a larger training set is needed. Table 4.4 shows the accuracies obtained by each classifier for each user and their averages (last row in the table). We can see that our system, except for particular cases, performs better than supervised classifiers even if using a smaller dataset. Furthermore, the proposed solution overcomes the specificity of the selected classifiers (i.e. different classifiers performing better on different users) offering a general framework for the sleep quality assessment. Indeed, on average, it outperforms all the comparing classifiers: $\sim 4\%$ more than DTW and $\sim$11% more than the best performing classifier (DecisionTable).

4.3 Measuring user's Physical Activity Level

Problem Statement

Resistance and physiological reserves decrease in older people, resulting in a risk of adverse health effects. This state of vulnerability is called frailty (Fontecha *et al.* , 2011) and is assessed taking into account the physical activity level (PAL), among other factors (Fontecha *et al.* , 2012). Nowadays, physicians detect frailty by means of specialized questionnaires and physical tests performed in dedicated facilities. However, the number of pre-frail elder people, which identifies a high risk of progressing to frailty, is increasing beyond the facilities potential. On the other hand, human-driven test scores may be insufficient and inaccurate for detecting physical habits (Boletsis *et al.* , 2015), and can be affected by certain degree of subjectivity (Jansen *et al.* , 2015). Today the great availability of general purpose wearable devices offers a new opportunity for noninvasive healthcare monitoring. Some watch-like systems have been already developed to monitor specific user's physical activities, exploiting heart rate and motion signals. Actually, much work has to be done before such systems can be regularly managed: the detection of a specific physical activity usually implies complex techniques, including machine learning and probabilistic modelling. For a widespread adoption the system should be highly flexible, handle uncertainty, and allow a personalization of what to monitor and how to notice it. In this paper we propose to use a smartwatch to detect the physical activity level rather than a specific physical activity. This approach can provide enough benefits to warrant widespread adoption. For this purpose, we studied a suitable computational architecture with adaptive setting and configuration.

Proposed Approach

In the proposed architecture (Fig. 4.6), the time series generated by each smartwatch sensor (i.e. heart rate, wrist motion and pedometer) are normalized between 0 and 1 and splitted in time series (6 minute each, partially overlapped). Then, each time series is processed by a Stigmergic Perceptron to unfold its behavior with respect to a set of 5 Archeypes, representing the most frequently appearing time series behaviors. Subsequently, the outputs of the three Stigmergic Perceptron are fused via a weighted sum, in order to obtain a combined classification of the effort of each activity segment. Weight are set up via Linear Least Square Method, (Hager, 2012) using a training set made by the multi-sensory input and the expected effort for each type of physical activity, i.e. relaxing, walking, driving, stairs, biking, football, tennis, and excursion (as reported on the subject's diary). Finally, the real value representing the current activity segment PAL, it is passed to another SP aimed to analyze physical activities as a macropattern, i.e., the daily PAL. Again, this SP computes a

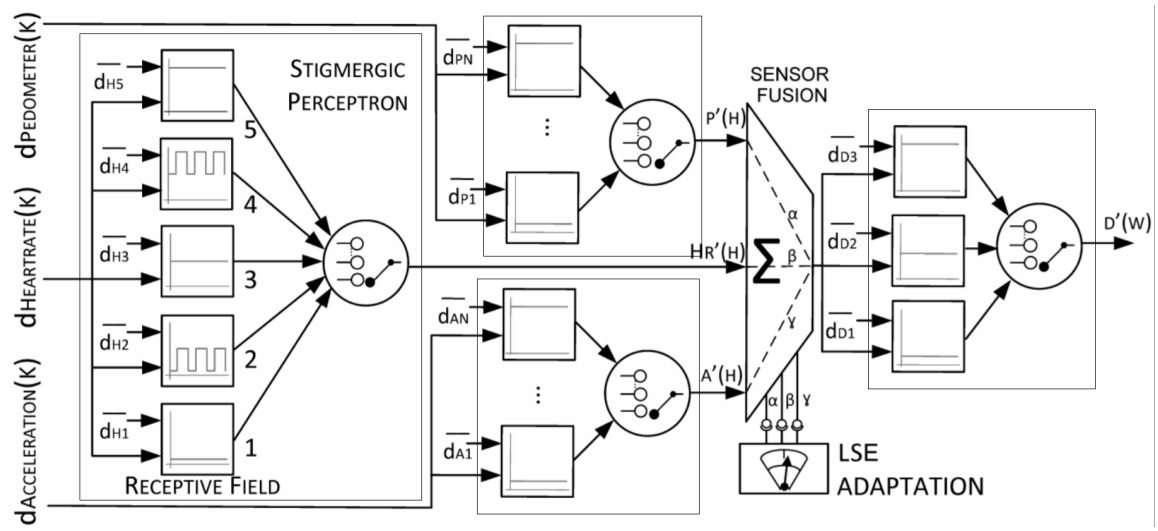


Figure 4.6: The architecture of our approach aimed at processing smartwatch sensors signals.

macro-level similarity between two daily time series. An example of class is a Low PAL Day, in which user does not perform any intense physical activity. Similarly, a linear combination of similarities among each archetype (Low, Medium and High) represents the daily PAL assessment.

Obtained Results

In order to prove the adaptation capability of our approach we test it with 3 different subjects (from now on referred as subject A, B, and C). Specifically, Subject A is a healthy and active 60 years man. He works and practises several sports. He does not present any frailty symptom, and is not under drug therapy. His activities data were been collected through smartwatch for a time period of 4 weeks of summer 2016. The activities performed and annotated on the diary spread from walking to excursion. The output provided by the system as a PAL is a real number in the interval [1,3], to represent any combination of the classes Low, Medium, High. In Fig. 4.8, each row comprises the samples related to a specific activity; each column represents a different PAL. On the top of each column, the activities with the expected PAL are also included. In practice, any activity involves a different life cycle with more or less different PALs (e.g. a recover process). In each row, the left and right side of the box represent the first and third quartiles of the distribution, the band inside the box is the second quartile (the median), while the ends of the whiskers represent the maximum and minimum of the distribution.

Overall, the fitting between the expected and the calculated PALs for subject A is good: the Mean Square Deviation over 165 time windows is 0.326. Indeed, we

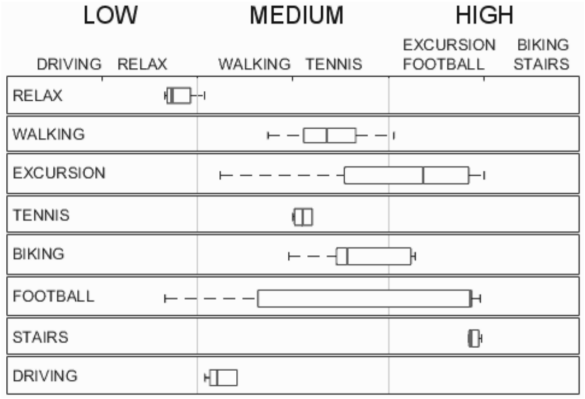


Figure 4.7: Physical Activity Levels of the Subject A over four weeks.

remark that relax, walking, virtual tennis, and stairs activities are mostly included in the expected class. Not surprisingly, excursion and five-a-side football are partially spread on the adjacent class, since the development of this kind of activities involves recovery processes with a lower physical. activity level. Similarly, the biking activity is expected to range from medium to high PAL, depending on the speed and the road slope. In contrast driving, which is an activity with constant PAL, is entirely included in the Medium PAL and not, as expected, in the Low. A deep investigation into the levels of processing shows that the most error for driving is located in the sensor fusion. In general, depending on the traffic and anxiety levels, driving may be an activity with high cognitive load, leading to a high heartbeat rate. In addition wrist acceleration is constantly high.

Since the purpose of the system is to assess physical activity on a daily basis, Fig. 5 shows the daily PAL computed by the system (white circles) along with the expected PAL (black circles). It is computed as the average PAL of the time windows of the day. Here we remark that, in 21 days, there is only one misclassification, on day 17. A deep investigation has shown that the error is derived by the driving activity, which is relevant for day 17. We remark that other 3 days in which driving was not the main activity are not affected by misclassification. Overall, the Mean Square Deviation with respect to the expected daily PAL is 0.158. The system was trained using 9 days (43%) of this data set.

In order to investigate the system behavior on older subjects, we have involved other two subjects into the experimentation. A problem is that older subjects are usually less active and less prone to manage a detailed diary. For this reason, we used the training carried out with the subject A for the initial roll-out of the system on the two subjects. The experimentation was made on three types of activity: relaxing, walking, and stairs climbing, and the diary entries were collected by the observer during direct observation. Although the number of activities and the gathering time are not relevant, results are very promising. More specifically, sub-

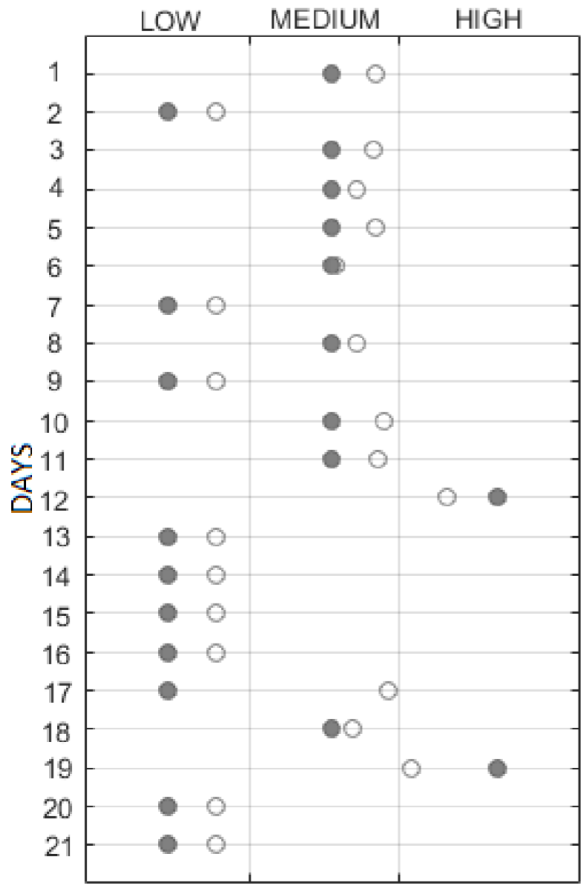


Figure 4.8: Daily PAL assessment on subject A.

ject B is a 74 years old man. He is retired, and is not physically active. He does at most 30 minutes of walking per day, for 5 days per week. He practises gardening, and does not present any frailty symptom. Occasionally he had some fall (recently, when taking the bus) without injuries. He is not under drug therapy. The data, on 14 time windows, were gathered on spring 2016. Each activity effort was classified by subject B as Low for Relax, High for Walking and Stairs climbing. The system performance is measured by a Mean Square Deviation of 0.0533.

Subject C is a sedentary 79 years old man. He is retired. He is not a very active individual: he walks for less than 15 minutes per day, for 5 days per week. He periodically does medical examinations, and is under drug therapy for blood pressure and cholesterol lowering therapy. The data, on 12 time windows, were gathered on summer 2016. The subject C classified his activity effort as Low for Relax and High for Walking and Stairs Climbing. The system performance is measured by a Mean Square Deviation of 0.0996. We remark that although both subjects have classified the walking activity effort as high, which is different than subject A, the system has correctly measured the walking. Actually, the direct observation of subjects B and C

has clearly shown that walking requires some degree of physical effort for them. The early results show that our system assesses PAL on how the activity is performed, despite of activity type.

In this work an innovative computational architecture for broad-spectrum assessment of the physical activity level of older adults is presented. The detection strategy is founded on computational stigmergy, a bio-inspired mechanism of emergent systems, which requires a continuous data gathering through general-purpose and non-intrusive devices, such as smartwatch. The architectural design is first presented. Then, the system experimentation is discussed on three subjects, making possible the initial roll-out of the approach in real environments. Experimental studies show promising results. A clinical trial could be interesting to validate the approach.

4.4 Traffic Congestion Estimation

Problem Statement

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 (Pellicer *et al.* , 2013). 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. By analyzing this data by means of our adaptive stigmergy based approach we aim at identifying the traffic congestion and exploring the different configurations of DE to find the best for this application.

Proposed Approach

We model a given urban street network as a directed graph. Fig. 4.9 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 geoposition $g_{v,t}$ of each vehicle 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 $t \in [\underline{t_k}, \bar{t_k}]$ the on-road positions of the queue head and tail can be denoted as $\underline{s_k^t}$ and $\bar{s_k^t}$.

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

$$E_k^{DETECTED} \equiv \{[\underline{t_k}, \bar{t_k}], [\underline{s_k^t}, \bar{s_k^t}], ..., [\underline{s_k^{\bar{t}}}, \bar{s_k^t}]\} \tag{4.1}$$

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:

$$E_k^{ACTUAL} \equiv \{[\underline{\tau_k}, \bar{\tau_k}], [\underline{\sigma_k^\tau}, \bar{\sigma_k^\tau}], ..., [\underline{\sigma_k^{\bar{\tau}}}, \bar{\sigma_k^\tau}]\} \tag{4.2}$$

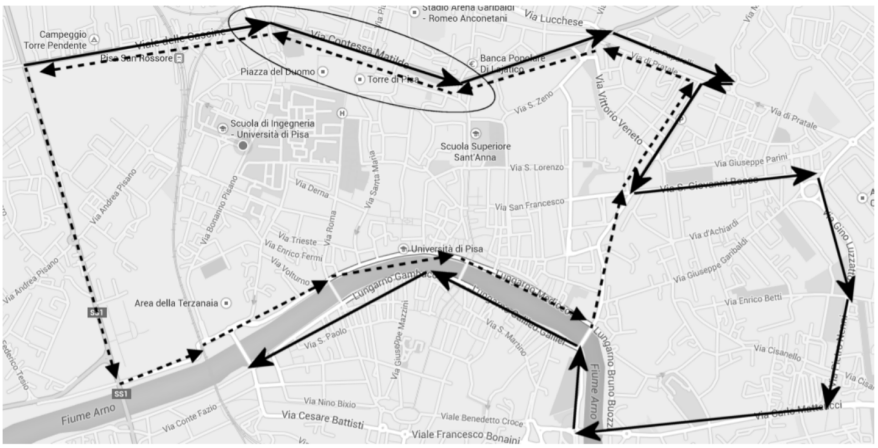


Figure 4.9: The Pisa urban area with two sample paths.

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

$$f_k = \frac{|\underline{\tau}_k - \underline{t}_k| + |\bar{\tau}_k - \bar{t}_k|}{|\bar{t}_k - \underline{\tau}_k|} + \sum_{\underline{i} \equiv min(\underline{\tau},\underline{t})}^{\bar{i} \equiv max(\bar{\tau},\bar{t})} \frac{|\underline{\sigma}_k^i - \underline{s}_k^i| + |\bar{\sigma}_k^i - \bar{s}_k^i|}{|\underline{i},\bar{i}||\bar{\sigma}_k^i - \underline{\sigma}_k^i|} \tag{4.3}$$

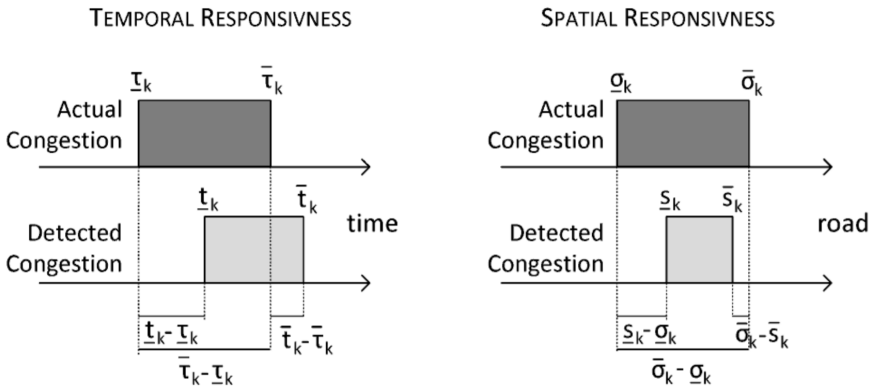


Figure 4.10: Representation of the main elements of the fitness in Formula 4.3.

More precisely, Fig. 4.10 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. 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:

$$Fit = \frac{1}{K} \cdot \sum_{k} f_k \tag{4.4}$$

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.

The proposed solution exploits the modules described in Chapter 3, exception made for the Input Activation module, specifically designed to process the vehicles localization information and produce a representation of the velocity in our system. 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 triangular mark is released in the virtual environment, and its intensity is proportional to the overlaps itself.

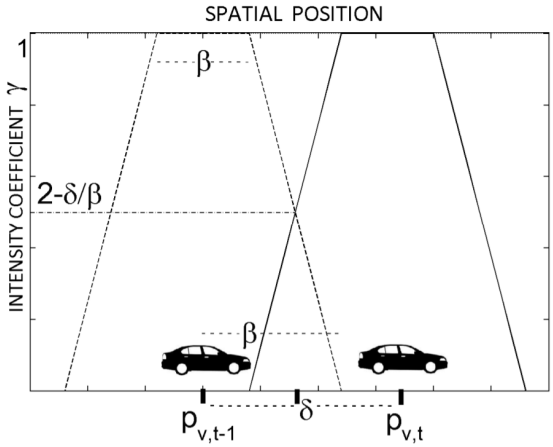


Figure 4.11: A scenario of input activation interface.

In Fig. 4.11 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, γ 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 λ, is:

$$\gamma_{v,t} = min\{1, 2 - \delta/\beta\} \in [0, 1] \tag{4.5}$$

The input activation is then processed by the following architecture, composed by the modules presented in Chapter 3. Specifically, we present together the Marking and Trailing modules, grouped as the "Stigmergic Layer". As known each module is parametrized (see Fig. 4.13) in order to be properly adjusted to the features of the current scenario. 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 timeconsuming, human-intensive and error-prone. Therefore, the adaptation (i.e. the Differential Evolution approach) is based on the evaluation of the fitness over a training set. In Fig. 4.13, 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.

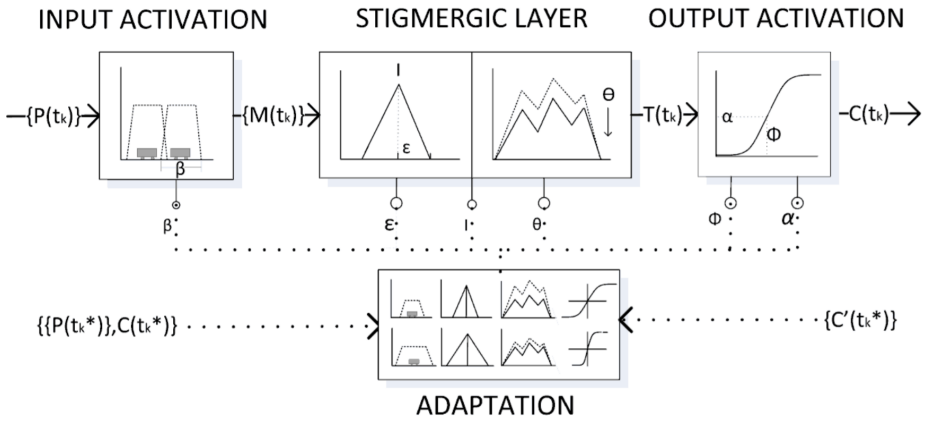


Figure 4.12: Overall system architecture.

However, even DE has few metaparameters to be tuned: the population size N, the crossover probability CR and the scaing factor F. Different population sizes are suggested in the literature (Wu *et al.* , 2016). 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 [2n, 40n]. Based on studies in (Cimino *et al.* , 2015a) 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] (Mezura-Montes *et al.* , 2006a). 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 (Wu *et al.* , 2016). To choose the best value, we performed trials with $CR \in \{ 0.2, 0.4, 0.6, 0.8 \}$ to compare the results. Finally, our aim is to identify which variant of DE is more suitable to identify traffic congestions, among DE/1/best/bin, DE/1/rand/bin, and DE/1/rand-to-best/bin.

Obtained Results

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 and the Matlab frameworks, respectively. Since we do not need to have a number of different traffic patterns to analyze, we prefer to use the knowledge we have of the scenario of our interest in an ad hoc simulator rather than using commercial solutions. Thus, 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. 4.9. In two hours of simulation, 116 congestion events occurred.

We ran DE for 30 generations, and for each setting of *CR-F-x* and 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 4.5 shows the fitness, in the form "mean $\pm$ confidence interval", for each strategy, together with the considered values of the parameters CR and F.

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. 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 4.5: 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**
F	**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**
	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**
F	**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
	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**
F	**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
	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 order to provide further details to the analysis, in Fig. 10 we 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.

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-

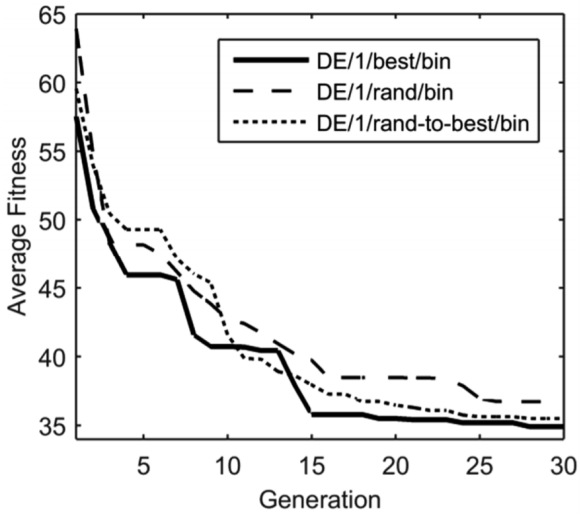


Figure 4.13: Fitness function versus generation, for *DE/1/best/bin*, *DE/1/rand/bin* and *DE/1/rand-to-best/bin* strategies.

best. However, the best strategy performs better and produces solutions with lower fitness than rand and rand-to-best strategies.

4.5 Anomaly Detection in Urban Mobility

Problem Statement

Smarness and sustainability are two key aspects of the forthcoming transportation systems. Smartness provides transportation monitoring and control with qualities like real-time sensing and fast decision making. Sustainability aims to manage travel demand efficiently by means of environmentally friendly strategies, providing transportation systems with policies for long-term economic suitability (Haque *et al.* , 2013).

Specifically, a sustainable urban development demands adequate policy instruments aimed to handle and mitigate the increasing volume of traffic congestion, carbon emission, and air pollution. One of the most frequently used policy tools for the measurement and evaluation of transportation sustainability performance are indicators. Indicators can be defined as quantitative measures aimed to explain and communicate complex phenomena simply, including trends and progress over time (EEA, 2005). In order to provide effective measures of sustainability of transportation activities, it is essential to define indicators' purpose and scope. During the last two decades, a number of international initiatives addressed the development of indicators aimed to achieve a more sustainable transportation on the local, regional, and global levels, by involving both scientific community and policy-makers (Litman, 2011). However, as of today, there is no standard or common agreement about the set of indicators to be used to assess transportation sustainability. Many works in the field perform an impact-based classification by employing a three-dimensional framework based on economic, environmental, and social impacts (Dobranskyte-Niskota *et al.* , 2007). The proposed indicators are in general calculated by exploiting commonly available data sources (Litman, 2011). Thanks to the pervasive technology supporting the smart city strategy, some of these indicators may be calculated via big data fed by on-board or fixed sensors. As an example, the GPS-enabled vehicles can provide a more comprehensive view of the factors shaping transportation emissions and efficiency, by analyzing passenger occupancy and trip density by location and time (An *et al.* , 2011). Finally, air quality monitoring systems can be used to monitor local pollution emission((Matte *et al.* , 2013),(Zheng *et al.* , 2013)) and noise emission (Zheng *et al.* , 2014a). These sources allow enhancing the precision of the investigation, providing insights about sustainability issues on specific urban locations and the moment in time. Among all the available sources of transportation data for smart city application, location aware vehicles provide us the opportunity to help urban planners and policy makers in mitigating traffic, planning for public services and resources, and properly manage infrequent events (Liu *et al.* , 2012). However, both public transportation and private vehicles provide quite predictable

GPS traces, because they are due to predetermined routes or personal routines (i.e. to and from work). On the other hand, GPS-enabled taxis, represent both a transit-complementary door-to-door transportation mode and a source of real-time human mobility information (Veloso *et al.* , 2011). Indeed, taxicabs play a prominent role as a transportation mode in metropolitan areas, e.g., in New York City, over 100 companies operate more than 13,000 taxicabs with a daily demand of 660,000 passengers (Zheng *et al.* , 2014a). Moreover, by continuously serving a wide diversity of passengers in the city, taxis GPS traces can provide a detailed glimpse into motivation and characterization of population's urban mobility. However, regular taxicab services becomes inefficient during urban-peak conditions, e.g., extreme weather or special events ((Zhan & Ukkusuri, 2014), (Neuwirth, 2016)), producing unnecessary traffic, pollution, energy consumption, and causing the increase of passenger's waiting time (Zheng *et al.* , 2014b). Thus, further investigation aimed at analyzing taxi-based transportation system is needed.

With this aim we exploit taxis' trip data provided by Taxi and Limousine Commission (TLC) of New York City. All taxis of NYC are equipped with FCD (floating car data) devices, which manage localization and card payments data, and enable taxicab drivers and passengers to receive information from the Taxi and Limousine Commission. FCD records include pick-up and drop-off positions, timestamp, and number of passengers, which feed the Taxi Trip Origin-Destination (OD) dataset. We analyze it in order to unfold urban hotspots, characterize human mobility patterns, and detect anomalous occurrences.

Proposed Approach

With the aim of discovering locations characterized by high density of taxi passengers activity (activity, for short), we build a bi-dimensional stigmergic trail by exploiting (as sample) the number of people being picked up or dropped off in a given time slot and in a given location. The hotspots are determined as the overlap of the city areas corresponding to the most relevant trails, obtained by analyzing data corresponding to early morning (i.e., 3a.m.-8a.m.), morning (i.e., 9am-2pm), afternoon/evening (i.e., 3pm-8pm), and night (i.e., 9pm-2am) time slots. Each hotspot is represented by a set of coordinates that bound a city area inside a polygon. The hotspots identified in Manhattan (New York City) are shown in Fig. 4.14. Their locations correspond to East Harlem - Upper East Side (A), Midtown East (B), Broadway (C), East Village - Gramercy - MurrayHill (D), Soho - Tribeca (E), Chelsea (F) and Time Square - Midtown West - Garment (G).

The result of the hotspot identification is a set of urban areas in which the most relevant activity dynamics occur. For each of them, we extract the activity time series by gathering the amount of activity occurred in the hotspots during the day. The hotspot activity time series are analyzed by means of Stigmergic Receptive Fields

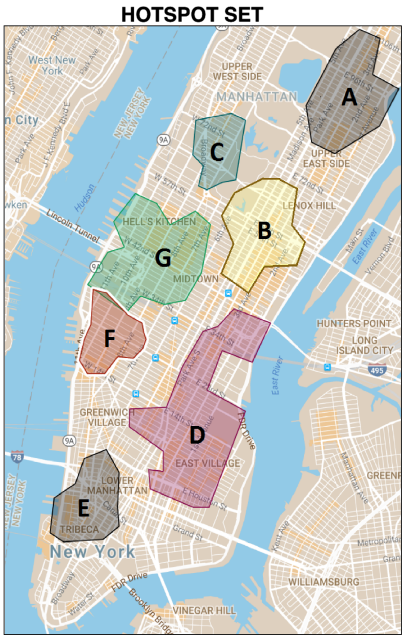


Figure 4.14: The hotspots found in Manhattan.

(SRF) arranged as a Stigmergic Perceptron. The SRF can detect a specific behavior in the actual time series, by processing it together with an archetype corresponding to that behavior. An example of behavioral class in our domain is “Rush-Hour” (Fig. 4.15a), which correspond to the behavior of the activity occurring in the hotspot when people movement is at its highest rate. Other classes provided are Asleep (Fig. 4.15g), i.e., the hotspot at its lowest activity level; Falling (Fig. 4.15f), i.e., the transition between regular activity and its calm down; Awakening (Fig. 4.15e), i.e., the waking up of urban activity following a calm phase; Flow (Fig. 4.15d), i.e., the hotspot at its operating capacity; Chill (Fig. 4.15c), i.e., the calm down of the hotspot activity after a rush hour; Rise (Fig. 4.15b), i.e., the transition to the most intense activity level.

In order to detect anomalous activity level patterns, we employ a further SRF aimed to measure of the similarity between two activity levels time series gathered in different days (Fig. 4.15h). The training set for this SRF is composed by different (i.e., 900) couples of activity level time series whose similarity is supposed to be 1, if they belong to the same activity level behavioral class, 0 otherwise. As an example, an activity level behavioral class can be “Working-Day" which is a day whose activity is mainly affected by working routines. Other activity level behavioral classes provided in our analysis are “Entertainment-Days" (usually occurring in Fridays and Saturdays) and “Leisure-Days" (usually occurring in Sundays).

By exploiting the SRF similarity measure we can match all the activity level time series of the SRF’s training set and store their similarity values into a similarity ma-

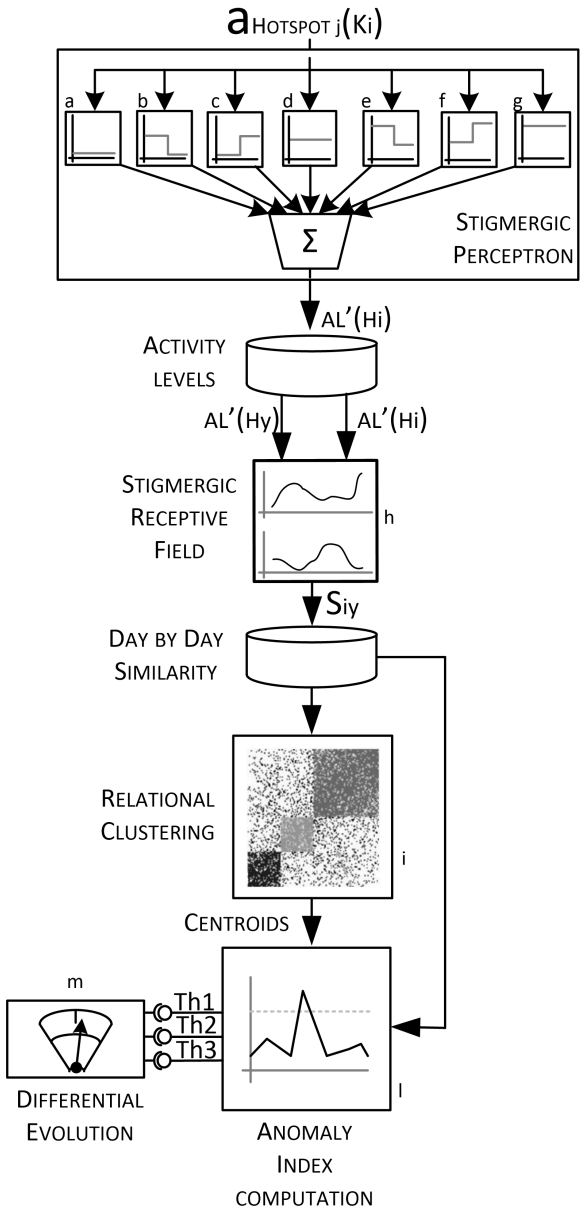


Figure 4.15: The overall processing of the activity samples.

trix. As an example, in Fig. 4.16, we show the similarity matrix obtained by analyzing patterns gathered during the year 2015. Patterns are arranged by behavioral class, i.e., Working-Day (days 1-10), Entertainment-Day (days 11-20), and Leisure-Day (days 21-30). Here, the similarity value obtained by matching two patterns gathered in different days is represented by the color of the corresponding box. The whitest the box, the higher the similarity. As expected, the similarity values appear to be higher only with couples of days belonging to the same activity level behavioral class.

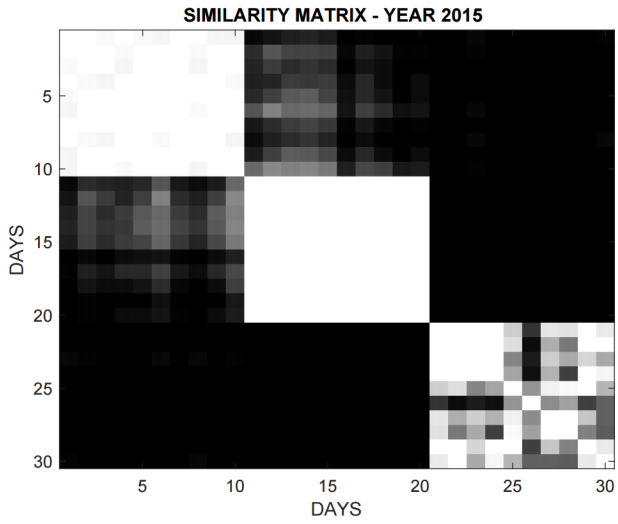


Figure 4.16: Similarity matrix obtained by analyzing patterns gathered during year 2015.

This similarity matrix is processed by a relational clustering technique (Fig. 4.15i) in order to group similar daily activity levels. Specifically, we employ Fuzzy C-Means using as number of clusters the number of daily activity behaviors taken into account in the analysis (i.e. 3). The fuzzy clustering generates, for each activity level time series, a membership degree for each behavioral class. Based on those, we can define a couple of measures aimed at finding the days characterized by anomalous activity:

- By treating as a distance the membership degrees of the daily activity level time series to each cluster, which are between 0 (not belonging to the cluster) and 1 (completely belonging to the cluster). Specifically, the membership degrees u_n, we measure the extraneousness of current activity level with respect to its expected cluster. The Extraneousness Index (EI) is defined as the Manhattan Distance between current daily activity level series d and the centroid of the cluster in which current day is assumed to belong. In Eq. 4.6, the computation case with 3 clusters is shown.

$$EI(d) = (|u_1(d) - u_1(C_2)| + |u_2(d) - u_2(C_2)| + |u_3(d) - u_3(C_3)|)/2 \quad (4.6)$$

- We can determine which are the most representative days for each behavioral class by selecting the activity level time series which are closest to the corresponding cluster centroid in terms of euclidean distance. An activity level time series obtained by a typical day is expected to exhibit high similarity with respect to the activity level time series obtained by the most representative days of its behavioral class. Thus, we compute the Anomaly Index of current day d

by exploiting the average of its similarity $S(d,i)$ with respect to its most representative N (i.e., 5) days, as detailed in Eq. 4.7. The Anomaly Index is defined between 0 (typical daily behavior) and 1 (very anomalous daily behavior).

$$AnomalyIndex(d) = |\frac{\sum_{i=1}^{N}(S(d,i))}{N} - 1| \tag{4.7}$$

In order to discern typical days from anomalies, an Anomaly Index threshold for each activity level behavioral class must be defined. These thresholds have been determined by using DE (Fig. 4.15m) in order to minimize the classification error (i.e., the percentage of correctly classified days) over all the days of the year under analysis, given a set of known anomalies.

Obtained Results

We analyze data provided by the Taxi and Limousine Commission of New York City, containing details about all taxi trips occurred during 2013, 2014 and 2015 in Manhattan. Each trip is reported with its taxi ID, number of passengers, together with latitude, longitude, and time-stamp of pick-up and drop-off. Data have been pre-processed in order to (i) remove missing values and (ii) discretize data in spatiotemporal buckets characterized by length and width of 10 foot, and duration of 5 minutes.

For the hotspots investigation, the period under analysis comprised both February and June. This period has been chosen since it can capture different seasonal behavior without being influenced by the presence of many holidays.

The activity time series extracted for each observed day has been normalized by using the min-max procedure.

Both global and local training phases of the SP are provided with a training set generated by applying random spatial noise and temporal shift to the pure archetype time series. The SP training set is composed of 70 time series (10 for each SRF), and the expected similarity is 1 if the current time series has been generated by the archetype on which current SRF must be specialized, 0 otherwise.

The SP outcome is processed by a further SRF, which is trained to measure activity level time series similarity, according to the daily behavioral classes provided, namely: (i) Working days which are expected to fall between Monday and Tuesday, when commuters and working routines deeply affects the crowd movements; (ii) Entertainment days, which are expected to fall on Friday and Saturday, and are characterized by high nocturnal activity due to the nightlife; (iii) Leisure days, which are expected to fall on Sunday, and are characterized by minor transportation usage. The training set is composed of 30 activity level time series, i.e., 10 time series representing the typical patterns of each behavioral class. The target similarity of each possible match is 1 if time series falls in the same behavioral class, 0 otherwise.

Depending on the land usage of the city area underlying each hotspot, some daily activity behaviors may not emerge. As an example, the Entertainment-Day behavior is mainly caused by the presence of clubs or other entertainment-oriented business that may attract the nightlife. Thus, the hotspots underlying a mixed usage zones are the most promising ones for the analysis, since our aim is to characterize all the aspects of the city life. According to official land use (publicy available at (zol, n.d.)), each city block can be classified into the following categories: commercial, residential, industrial, transportation space, institutional, open/recreational space, parking or vacant. By considering the distribution of these categories in each hotspot it can be evaluated how diversified the usage of that area is, and therefore the related amount of the mobility dynamics. Specifically (i) Hotspot A is primarily residential and secondly institutional; (ii) Hotspot B is mainly residential and commercial; (iii) Hotspot C is principally open space and residential; (iv) Hotspot D and E are characterized by an equal distribution of almost all usage classes; (v) Hotspot F is mainly commercial and residential, with some institutional blocks; (vi) finally, Hotspot G presents all usage categories, with a prevalence of the commercial category. The higher the variety of the usage of a hotspot, the better a candidate this hotspot is for our analysis: then hotspots D and E are chosen.

For hotspot D in 2015 we firstly employed the approach based on EI. In Fig. 4.17 the EI is computed with the data regarding September and October 2015, than this is compared with the maximum EI in the training set (red line in Fig. 4.17). Days with an EI greater than the maximum EI in the training set are recognized as an unexpected pattern (red spot in Fig. 4.17).

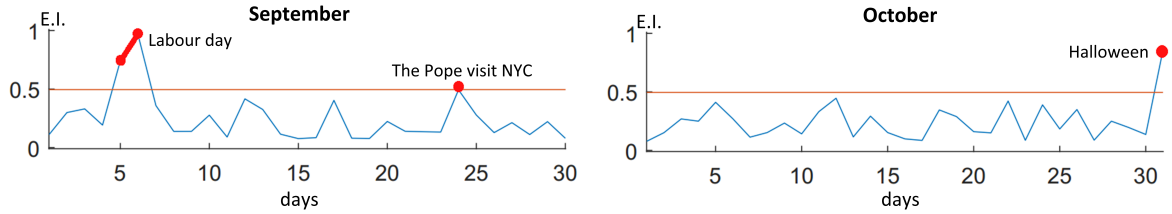


Figure 4.17: Extraneousness Index computed over days in September and October.

By repeating this procedure over the whole year 2015 we collect a number of relevant unexpected patterns, reported in Table 4.6. Each unexpected pattern date is shown together with their most probable cause, such as an occurred social event.

Than we experimented the approach based on the Anomaly Index. For the year 2015, Table 4.7 shows the percentage of correctly classified patterns (among the normal and anomalous classes) obtained with 5 different trials in the form mean $\pm$ 95% confidence. The classification performance is also calculated by using two well-known time series distance measures: the Dynamic Time Warping (Taylor *et al.* ,

Table 4.6: Most relevant unexpected patterns detected all over 2015.

EI	Date and occurred city event	
0.96	06-Sep,	Labour Day celebration
0.94	24-May,	Memorial Day
0.86	31-Oct,	Halloween
0.83	26-Nov,	Thanksgiving
0.83	28-Jun,	Gay Pride
0.82	25-Dec,	Christmas
0.81	01-Jan,	New Year's Eve
0.80	04-Apr,	Easter (holy Saturday)
0.79	27-Jan,	Winter Storm Juno (jun, 2015)
0.74	05-Sep,	Labour Day celebrations
0.63	03-Jul,	Independence Day
0.63	31-Dec,	New Year's Eve
0.61	15-Mar,	NYC Half Marathon
0.49	24-Sep,	Pope Francis in NYC (pop, 2015)

Table 4.7: Percentage of Correct Classification achieved by analyzing of hotspot D and E during 2015, and using 3 similarity measures.

Similarity Measure	Hotspot D	Hotspot E
SRF	95.61 ± 0.003	94.24 ± 0.24
DTW	90.57 ± 0.134	91.80 ± 1.387
FRECHET	90.14 ± 0.537	90.52 ± 2.34

2015), and the Fréchet distance (Driemel *et al.* , 2016). Clearly, the SRF measure outperforms both the DTW and the Fréchet distances.

In order to further test the assessment of anomalous patterns, the activity time series annotated as anomaly have been annotated by a triple according to their affinity with the typical pattern of each behavioral class. As an example, the triple W|E|L means that current time series is mostly similar to Working-Day typical pattern and secondly to Entertainment-Days one, whereas it shows only minor similarity with respect to Leisure-Days. With the aim for measuring the capability of our similarity measure to generate a corresponding affinity assessment, we compute the average of the similarity of each time series annotated as an anomaly with respect to the most representative days of each behavioral class. Sorting them by similarity, we obtain the triple. The Mean Assessment Error is computed as the number of non-matching sort constraints for each pair of triples, averaged over all the set of the anomalies. As an example, the triples W|L|E and W|E|L have just one non-matching sort con-

straint, which is L<E, whereas both triples state that W<E and W<L.

A comparison is provided by repeating this procedure using the Dynamic Time Warping (Taylor *et al.* , 2015) distance. Resulting Mean Assessment Error are equal to 1.135 (SRF-based similarity measure) and 1.115 (DTW distance). According to these results both methods are suitable for pattern analysis, thus we provide the comparison of their performance in anomalous pattern detection.

Specifically, in order to compare the classification performances of our approach with respect to DTW, we collect the percentage of correctly classified days among 5 trials. During each trial, the DE generates a new set of Anomaly Index thresholds. If the Anomaly Index of an activity level time series exceeds the threshold, the corresponding day is considered anomalous. Obtained results are presented in the form "mean $\pm$ 95% confidence interval" in Table 4.8.

Table 4.8: Percentage of Correct Classification achieved by analyzing data gathered during 2013, 2014, and 2015.

Year	**SRF**	**DTW**
2013	92.71 ± 0.321	90.57 ± 0.134
2014	96.65 ± 0.109	92.27 ± 0.106
2015	95.61 ± 0.003	91.28 ± 0.106

Based on obtained results, our approach provides an effective detection of major anomalies. But, handling minor or potential anomalies could be more difficult. In order to evaluate the effectiveness of our measure while handling this kind of anomalies, we select a set of events including official holidays and days affected by special events with documented effect on the road in (or in close proximity of the) hotspot D. Such events could be days characterized by adverse weather condition (e.g., Juno storm), street closure (e.g., due to the Gay Pride parade) and so on. This set is provided for each year under analysis.

A set of ordinary days is also included. An effective anomaly measure is supposed to exhibit high correlation between its value and the set (events or ordinary days) which current day belongs to. In Fig. 4.18 the correlation obtained by using our SRF-based measure is shown.

In order to compare obtained results in terms of correlation between events and computed Anomaly Index, we provide it by using SRF-based approach and DTW. In Table 4.9 we present obtained correlation coefficient for each year under analysis.

According to the provided results, our approach was able to identify city hotspots, characterize the daily patterns of their activity over time and detect days characterized by an anomaly. Our approach has been tested on real world dataset containing all taxi trips occurred in Manhattan during 2013, 2014 and 2015, for a total amount of 74GB of data. The performances of our approach is measured in terms of percentage

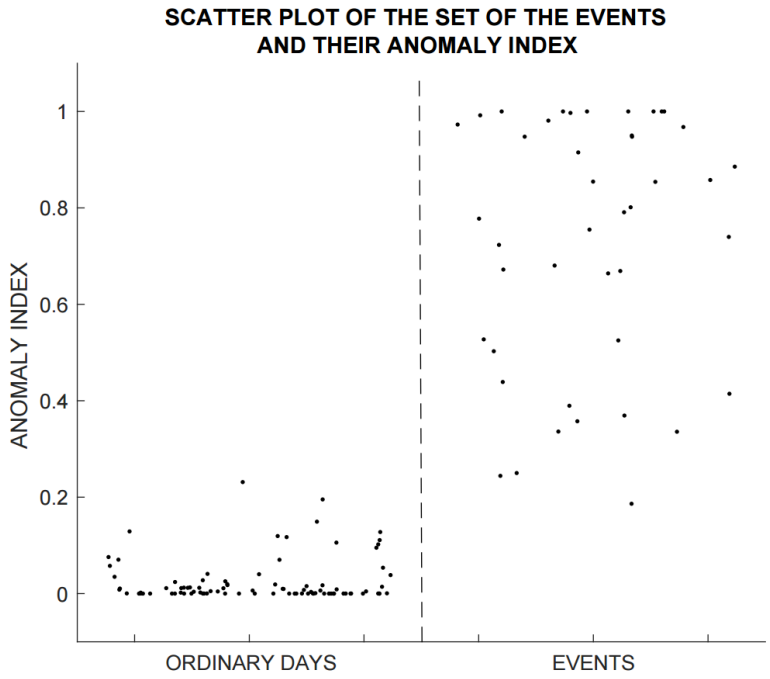


Figure 4.18: Scatter-plot generated by considering Events, Typical Days and their corresponding Anomaly Index.

Table 4.9: Correlation coefficient between Anomaly Index and day characterized by events occurring in the hotspot. Comparison between SRF-based approach and DTW.

Year	**SRF**	**DTW**
2013	0.8963	0.7177
2014	0.9289	0.7236
2015	0.9210	0.6828

of correctly classified daily patterns among typical and anomalous ones. Moreover, the effectiveness in handling minor anomalies is measured by computing the correlation between the Anomaly Index and the occurrence of urban events (e.g., local parade or official holiday).

Obtained results have been compared with respect to the one achieved by analyzing the activity time series with DTW. In both cases, and in every year under analysis, our approach outperforms DTW, achieving up to 96.65 percentage of correctly classified days and 0.9289 correlation coefficient.

4.6 Assessment of Refugees' Integration

Problem Statement

In the context of Syrian refugee crisis, Turkey is both an *effective* and *affected* country (Keyman, 2016). Indeed, it provides protection and facilities to more than three million refugees; but, on the other hand, an increasing hostility is emerging in the local Turkish communities, due to the magnitude and the duration of the humanitarian crisis. In order to prevent the growing of societal tensions over Syrian refugees, there is the need to formulate effective long-term integration policies (Carpi & Pınar Şenoğuz, 2018). However, the formulation of an effective policy demands tools aimed at evaluating and understanding the integration of refugees despite the complexity and the width of this phenomenon. In this context, great benefits can be provided by complementing the paper-and-pencil surveys, the interviews, and the focus groups with big data-driven indicators (Hardy & Maurushat, 2017).

One source of data that offers great potential for this kind of analysis are information captured from mobile phones (Gundogdu *et al.* , 2016), which have been used to analyze many effects of the migratory phenomena, i.e., the ones on political elections (Altindag & Kaushal, 2017), job markets (Silm & Ahas, 2014) or on the spread of epidemics (Tompkins & McCreesh, 2016).

In this work we analyze the Call Detail Records (CDR) datasets provided within the D4R data challenge (Salah *et al.* , 2018) with the aim of unfolding which conditions can contribute to the integration of refugees. Moreover, we aim at providing some data-driven indicators of the integration of Syrian refugees in Turkey, in order to allow policy makers at evaluating the effectiveness of the strategies aimed at fostering the integration of refugees.

Proposed Approach

In order to assess the integration of refugees, it is essential to establish metrics able to capture this phenomenon. These metrics should consider both on short (daily) and long (bi-weekly or monthly) term mobility and calling behavior of refugees and locals. Indeed, many works in the literature (Singh *et al.* , 2015) highlight the improvement obtained by including individual's mobility and behavior in the model, with respect to pure statistical one. It follows the list of the metrics we propose for our analysis:

- *Residential Inclusion by District (RI)*: we can assume that most of the calls during the night and early morning hours come from people's homes. Indeed, based on this assumption many works in the field of the CDR analysis infer the location of an individual's home as the place from which he/she mostly

call between 8 pm and 8 am (Alexander *et al.* , 2015). Thus, by observing the percentage of calls made by refugees between 8 pm and 8 am per antenna $a \in d$ is possible to assess the coexistence of resident locals and refugees in a given the districts d and a given month m. This metric is defined between 0 (no resident refugees' in the district) and 1 (only resident refugees' in the district).

$$RI_{d,m} = \frac{|calls_{a,m}(R)|_{a \in d}}{|calls_{BS,m}(R) + calls_{a,m}(L)|_{a \in d}} \tag{4.8}$$

- *District Attractiveness (DA)*: A district is considered attractive if the flow of people who move to it is on average higher than the flow of people who move from there in a given month (i.e. the people netflux). As for the assumptions used in the *RI* metrics, a person resides in a given district and month if that district is the most recurrent location from which he/she makes calls between 8 pm and 8 am. Specifically, given $residentRefugee_{d,m} = \{R | r : home_r(m) = d\}$ i.e. the set of the refugees who live in the district d during the month m, the District Attractiveness is defined as:

$$DA_{d,m} = |residentRefugee_{d,m+1}| - |residentRefugee_{d,m}| \tag{4.9}$$

- *Refugee's Interaction Level (IL)*: it is defined as the percentage of phone calls toward locals made by a given refugee in a given period. It represents how much the refugee is socially connected to the locals (Blumenstock & Fratamico, 2013), i.e. 0 means no calls toward locals and 1 means only calls toward locals. Each level is defined as a range of 20% within this scale.

$$IL_r = \frac{|calls_{r \to L}|}{|calls_{r \to L}| + |calls_{r \to R}|} \tag{4.10}$$

- *Refugee's Calling Regularity (CR)*: let us consider the time series of the call frequency (i.e. the calling pattern) made by each individual. Specifically, we build the calls pattern as the number of phone calls made by a person in a given hour of the day during a period of time. We normalize this amount with the average number of calls per hour in order to be comparable despite the different amount of calls made by each person. The period of time taken into account can be monthly or bi-weekly. In general the calling pattern may be due to several factors, e.g. daily routines, habits, or working schedule.

  Even if it is not possible to determine which component has a predominant role in generating a specific calling pattern, we can assume that similar routines will most likely generate similar calling patterns. Moreover, routines similarity is often linked to integration (Jansen *et al.* , 2006). Thus, the similarity

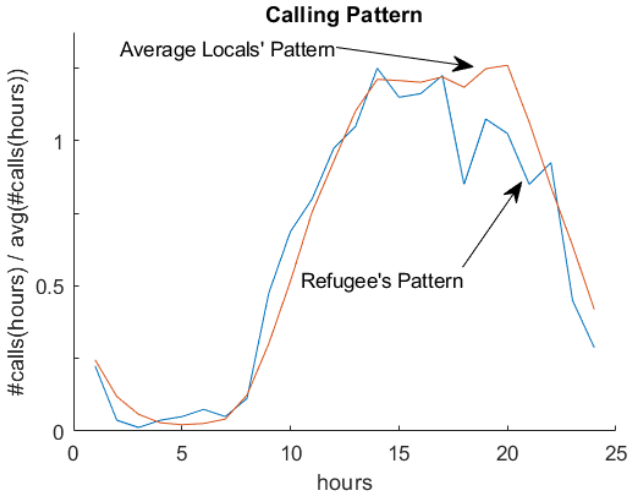


Figure 4.19: Calling patterns representation.

between the calling patterns of locals and refugees, may be a proxy of integration. As an example, each refugee that is employed is supposed to have a calling pattern (thus, a daily routine) similar to the average calling pattern of the locals, since they are mostly employed (Almaatouq *et al.* , 2016). In this context, the more a refugee's calling pattern CP_r is similar to the average local's calling pattern LCP the more it is considered regular. The similarity between two calling patterns is computed (Eq. 4.11) as their cosine similarity (Dong *et al.* , 2015). This metric is defined between 0 (completely different calling pattern w.r.t. locals) and 1 (identical calling pattern w.r.t. locals).

$$CR_r = \frac{CP_r \cdot LCP}{||CP_r|| \cdot ||LCP||} \tag{4.11}$$

- *Refugee's Mobility Similarity (MS)*: by collecting the locations of each call occurred during the day we can build the daily trajectories of an users' mobility. The similarity of the trajectories of refugees T_r and locals T_l implies the sharing of some urban space at the same time and may affect (or be affected by) the integration of the refugees (Hebbani *et al.* , 2017). The computation of this similarity is based on the principle of *stigmergy*. Stigmergy is a self-organization mechanism used in social insect colonies (Marsh & Onof, 2008). Basically, individuals in the colony affect each other behavior by marking a shared environment with pheromones when a specific condition occurs (e.g. the presence of food). The pheromone marks aggregate with each other in the trail if they are subsequently deposited in proximity to each other, otherwise they evaporate and eventually disappear. Thus, the resulting pheromone trail

steers the whole colony toward the region in which the condition above (e.g. the discovery of food) occurs consistently.

This pheromone-like aggregation mechanism can be employed in the context of data processing, providing self-organization of data (Vernon *et al.* , 2007) while unfolding their consistent spatio-temporal dynamics. By exploiting *computational stigmergy*, each sample of the trajectory is transformed in a digital pheromone deposits (i.e. mark) and released in a three-dimensional virtual environment in correspondence of each sample coordinate and time of appearance. Marks are defined by a truncated cone with a given width. Marks aggregate in the *stigmergic trail*, which is characterized by evaporation (i.e. temporal decay δ). The evaporation may be counteracted if marks are frequently released in proximity to each other, due to their aggregation, whereas isolated mark progressively evaporates and disappear. Eq. 4.12 describes the trail at time instant i.

$$T_i = (T_{i-1} - \delta) + Mark_i \tag{4.12}$$

Since only consistent spatio-temporal dynamics in data generate a stable pheromone trail, the trail itself can be considered as a summarization of these dynamics. By matching trails, we provide a general similarity measure for spatiotemporal trajectories. The similarity between trails is obtained by using the Jaccard similarity (Niwattanakul *et al.* , 2013), i.e. the ratio between the volume of the intersection and the union of the stigmergic trails (Fig.4.20).

The similarity of the spatiotemporal trajectories of refugees T_R and locals T_L (Eq. 4.13) is defined between 0 (completely different trajectories) and 1 (identical trajectories).

$$MS_{R,L} = \frac{|T_R \cap T_L|}{|T_R \cup T_L|} \tag{4.13}$$

Since our investigation includes an analysis of mobility, call behavior, and district characterization it is necessary to focus our research in areas that ensure (i) an high calling activity made by refugees. Indeed, in order to have representative behavioral models we have to avoid areas characterized by sparse data; and (ii) a good spatial resolution, which means an high density of antennas, since the granularity of the trajectories will be determined by this; in fact, with few antennas in the area under investigation, all trajectories will be roughly similar; and (iii) high number and diversification of districts per area; indeed, the district-based metrics can explain the settlement choice of each refugee. This effect is especially noticeable in the presence of many different districts close to each other since this allows refugees to move

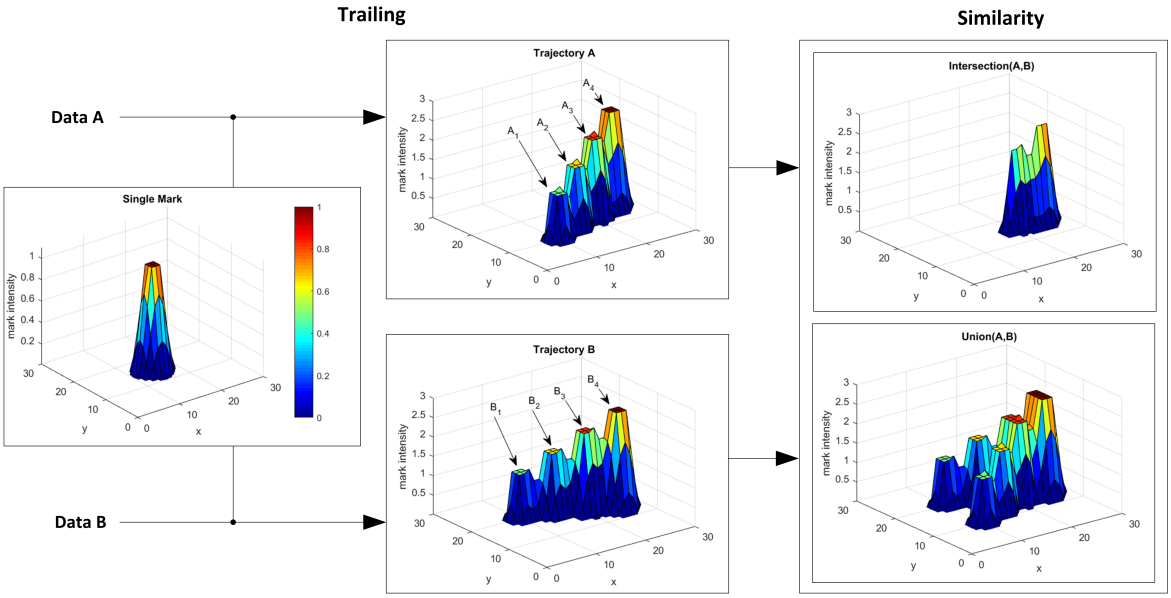


Figure 4.20: Phases of the mobility similarity computation. We represent the trails obtained from the deposit of 4 consecutive samples (A_1, A_2, A_3, A_4 and B_1, B_2, B_3, B_4) of the trajectories (A and B), their intersection and their union, which are used to compute their similarity

from one district to another according to their socio-economic integration level and its change in time. Therefore, our analysis focus on the cities of Istanbul, Ankara and Izmir, since they have the larger density of antennas and the larger calling activity made by refugees. In addition, Istanbul's metropolitan area alone consists of 69 districts with a variety of different characteristics (e.g., different housing costs or job opportunities). For this reason, the district-wise analyses focus on Istanbul.

Obtained Results

In order to verify if the Calling Regularity can be actually used as an integration proxy, we analyze the relationship between the Interaction Level and the calling regularity of each refugee in Istanbul. In order to have a reliable model of the calling pattern, we select the refugees with an average amount of calls per day equal or greater to 2. We compute the Pearson correlation coefficient between the average of the Calling Regularity of the refugees and their Interaction Level. In Fig. 4.21 the distribution of the resulting correlation coefficients is shown.

It is evident that the Interaction Level and the Calling Regularity are strongly and positively correlated, providing us with the insight that refugees that exhibit greater interaction with locals have also daily routines which are similar to them. This result comforts the findings of other studies in the field (Jansen *et al.* , 2006)

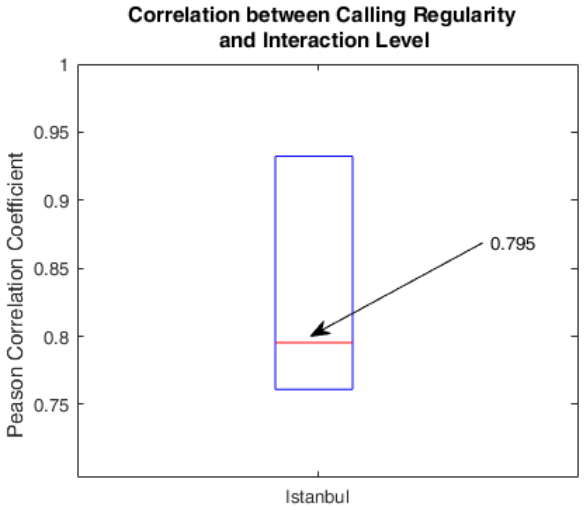


Figure 4.21: Boxplot of the correlation coefficient between Calling Regularity and the Interaction Level for each period in Istanbul.

and allows us to promote this measure as a metric for integration that should be taken into consideration by policy makers, at least in the case of Syrian refugees in Turkey.

Given the possibility of using Calling Regularity as an integration metric, we try to use it to obtain more insights at the district level. Specifically, we analyze the relationship between the District Attractiveness, the Residential Inclusion and the Calling Regularity of the refugees in each district of Istanbul according to the cost of living in the district itself. As an indicator of the cost of living per district, we consider the average rent cost per square meter in each district during 2016 (the data owner is an online housing website that would like to stay anonymous).

Firstly, we assess the impact of the presence of refugees on the attractiveness of a district. In order to do so, we compute the correlation between each district's yearly (i.e. averaged over 2017) Residential Inclusion (RI) and the District Attractiveness (DA). Figure 4.22 shows the correlation matrix obtained with the yearly RI and DA per district.

With a correlation coefficient equal to 0.494 and a P-value of 0.0016 we can consider RI and DA significantly and positively correlated. This means that refugees are more likely to move and stay in districts with a greater the number of refugees.

Pushing the investigation on a more fine level (i.e. Monthly-wise) we focus on the relation between the RI of a given district and month and the average CR of the refugees living in that district during that month. In order to work with representative calling patterns the CR is computed with refugees having at least 100 calls and settled in Istanbul for at least half of the whole year. Figure 4.23 shows the boxplot obtained with the correlation coefficients computed between the RI of each district

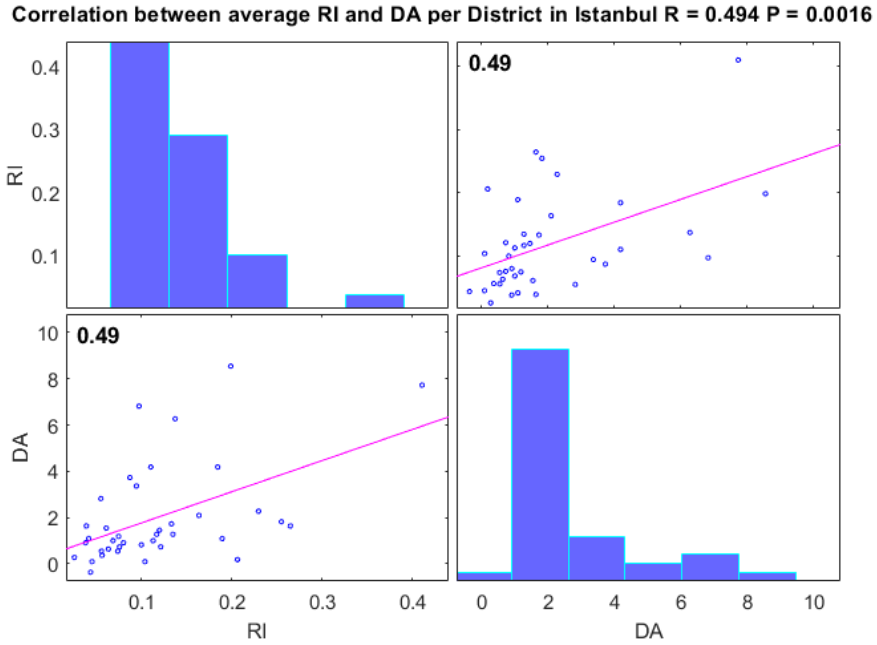


Figure 4.22: Correlation matrix obtained with the yearly RI and DA per district. On the diagonal the distribution of the average RI and DA respectively, whereas the others are the bivariate scatter plots with a fitted line.

and month and the average CR of the refugees living in that district during that month.

According to the results in Figure 4.23, even though its strength may vary from case to case, the RI in a given month and district appear to be positively correlated with the CR of the refugees living in that district during that month. In other words, the districts with the highest Residential Inclusion of refugees are also the districts where the routine of the refugees is more similar to the locals. This suggests that a minimum number of refugees per area is required for the dynamics of integration to be triggered, as suggested in (McIsaac, 2003). To understand the order of magnitude of the amounts we are talking about we analyze the distribution of RI by month and district. It is evident that many districts have a low RI, thus depicting a scenario of minor coexistence of refugees and locals in most of the districts. Moreover, in the few districts (and months) with higher RI, the RI value never exceeds 50%. Thus, the more evenly distributed the residents (locals and refugees in an area) are, the greater the similarity between the routines of locals and refugees.

Finally, we include the cost of living in a certain district in the analysis. The last results show that the Calling Regularity can be considered a proxy for social integration. However, the Calling Regularity may be even linked to the employment of a refugee. Unfortunately, it is not possible to verify directly this implication due to the lack of details about refugee employment, since they are often employed in

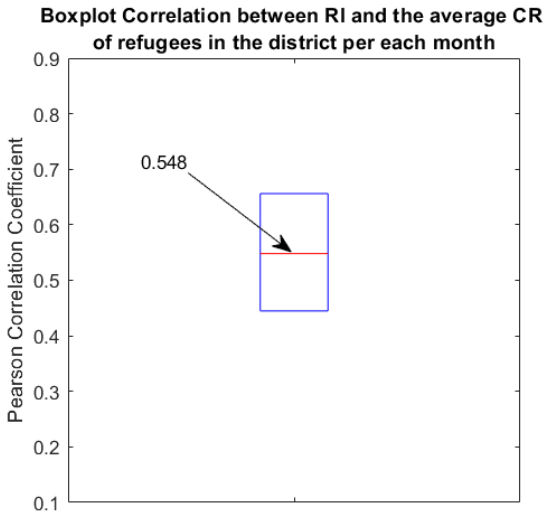


Figure 4.23: Boxplot of the correlation between RI of a given month and district and the average CR of the refugees living in that districts during that month.

the informal sector (Del Carpio & Wagner, 2015). Yet, it is possible to study this implication according to the social and economic characteristics of the location where refugees live. Indeed, depending on their economic well-being and the level of integration, migrants and refugees may choose different settlement solutions (Fawaz, 2017). For example, only an individual who has enough economical resources (e.g. who has some kind of job) can afford to live in an area that offers better opportunities. On the other hand, those who are not integrated and/or not working often find themselves socially isolated from the locals and relegated to poor neighborhoods.

In order to provide additional insights into this, we exploit the average rent cost per square meter in a given district in the year 2016 as an indicator of the cost of living for that specific district. Specifically, we compute the correlation between the cost of living in a given district and the average CR of refugees living in that district (Fig. 4.24).

With a correlation coefficient equal to 0.5 and a P-value equal to 0.003, the average CR of refugees living in the district exhibits a significant and positive correlation with the cost of living in that district. This means that, although it may be influenced by some factors not detectable by the data under analysis, the CR is a proxy for the daily routine similarity and for the economic capacity of refugees (i.e. the ability to meet a certain cost of living), thus it is a tool able to capture both necessary conditions occurring with the employment of refugees. For this reason, this metric should be taken into account when dealing with the problem of integration of refugees because having a job is one of the first promoter of refugees' integration (Bakker *et al.* , 2016), but is also hard to analyze it since the refugees' employment often happen

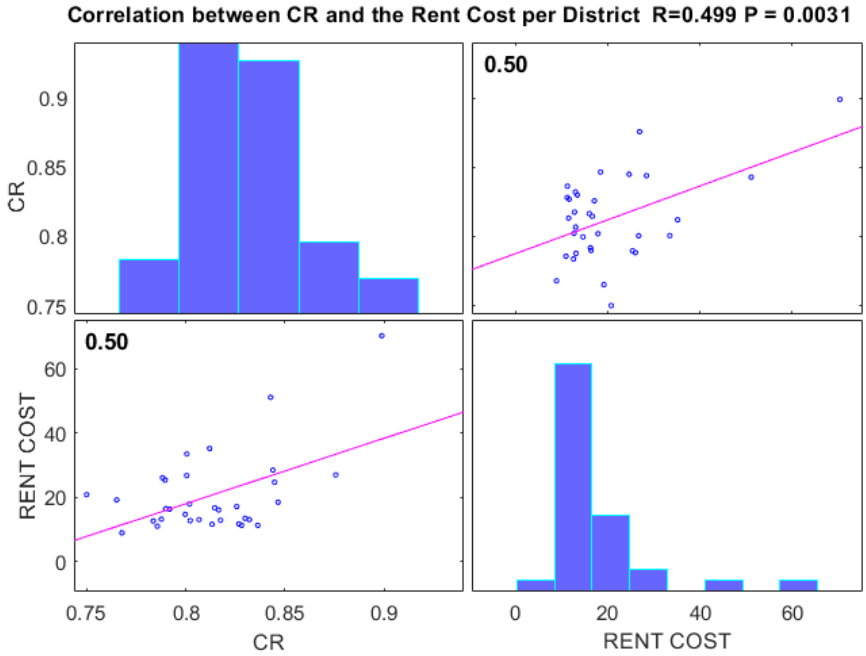


Figure 4.24: Correlation matrix obtained with the average CR in a district and the cost of living per district. On the diagonal the distribution of the average CR and cost of living respectively, whereas the others are the bivariate scatter plots obtained with these variables together with the fitted line.

in the informal sector (Balkan & Tumen, 2016).

Another fundamental driver of integration can be the sharing of urban spaces with the locals (Madanipour, 2015). However, its positive contribution in the integration dynamics it is not obvious. Indeed, it can allow the progressive integration in the social structure of the hosting city. However, on the other hand the shared urban areas may not be easily defined and perceived as a safe space (Lyytinen, 2015) thus leading to the occurrence of social friction in those areas.

In order to understand the contribution of sharing the same urban space with the locals, we analyze the relationship between the Mobility Similarity and the Interaction Level on a daily bases. Specifically, we create the cumulative trajectories of the group of refugees with a given Interaction Level, i.e. the stigmergic trails obtained with all the samples of the people in that group. Then, we compute the Mobility Similarity with the cumulative trajectories obtained with an equally sized group of locals.

We collect the Pearson correlation coefficients between the Interaction Level of each group and the resulting Mobility Similarity. We repeat this procedure multiple times by randomly subsampling the people for each group larger than the smallest one. In Fig. 4.25 we present the distribution of the obtained correlation coefficients by means of boxplots. It is evident that in the 3 cities analyzed the Mobility Simi-

larity is strongly correlated with the interaction level. Indeed, the 95% confidence interval of the correlation coefficients results as 0.91 ± 0.01 in Istambul, 0.83 ± 0.06 in Ankara, and 0.92 ± 0.04 in Izmir.

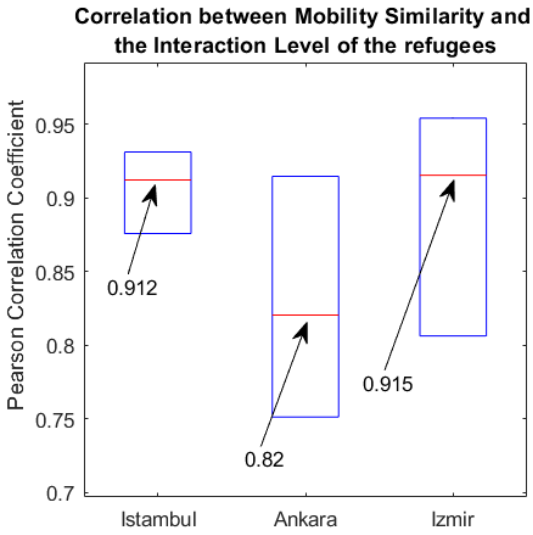


Figure 4.25: Boxplot of the correlation coefficients between the Mobility Similarity and the Interaction Level, over multiple trials with subsampling. The cases of Istanbul, Ankara and Izmir.

On the basis of the obtained results, it is possible to claim that the more the refugees have interactions with locals, the more they share urban spaces with the locals. This allows us to say that sharing of urban spaces is a positive factor in the dynamics of integration of refugees. Thus, the policies designed to improve refugees' integration should take into account Mobility Similarity to assess their impact.

Since we have seen that Mobility Similarity and Interaction Level are able to capture the integration of refugees, we now attempt to use them to study the effects of the events that are certainly caused or can cause the disruption of refugees' integration: the occurrence of social frictions. In order to look for the features that characterize a social friction, it is necessary to start with few examples of publicly known social frictions. Specifically, we collect a set of such events and we compare the Mobility Similarity and Interaction Level in 2 weeks before and after each event. We have found a number of occurrence of such events by searching for them over the internet. These are summarized in Table 4.10.

Once these events have been identified, we study the impact of these social friction by calculating the Mobility Similarity (with repeated trials according to the methodology described in the last section) and the percentage of calls made toward the locals, according to the Interaction Level of the refugees. Finally, we present the ratio between MS and the percentage of calls in the two weeks before and after each

Date	Location	Source
March 6	Izmir	(fri, 2018a)
April 12	Istanbul	(fri, 2018c)
May 15	Istanbul	(fri, 2018d)
May 16	Istanbul	(fri, 2018b)

Table 4.10: Dates and locations of the social friction events taken into account.

event. If this ratio is greater than 1, it indicates that after the event, the integration measure taken into consideration has decreased.

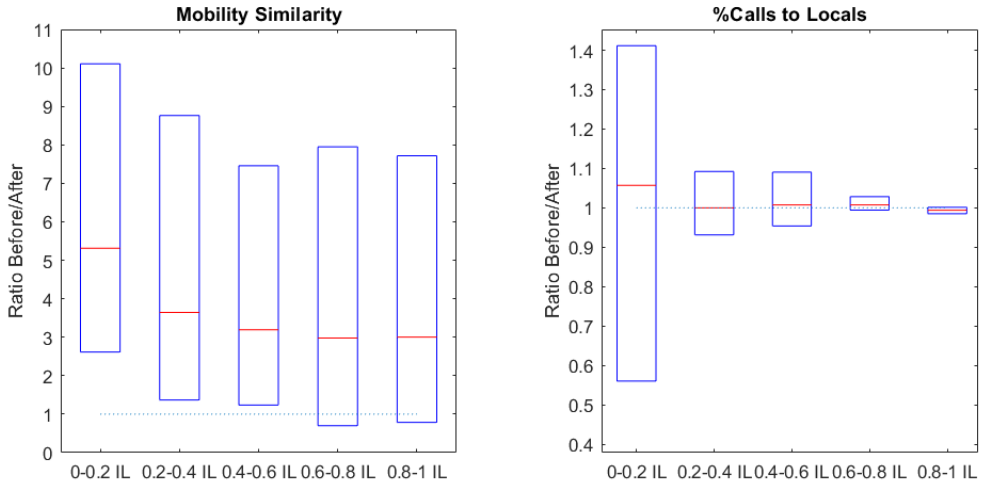


Figure 4.26: Ratio between the Mobility Similarity and the percentage of calls made toward refugees, 2 weeks before and after each social friction. The ratio is presented according to the IL of the group of refugees.

It is apparent that the social friction affects the behavior of the refugees by reducing the amount of shared urban space with the locals (i.e. lowering the Mobility Similarity after the event). Moreover, in terms of calls made toward locals, the social friction event affects the group of refugees with lower level of interaction with locals way more than the more integrated groups. Indeed, on average, they exhibit a lower percentage of calls made toward locals and a greater variability. Indeed, the quartiles of the percentage of calls made toward locals are arranged as [0.55, 1.05, 1.41] with the refugees with the lower Interaction Level, whereas are [0.98, 0.99, 1] with the refugees with the greater Interaction Level. Here, even the MS results more affected in the group of refugees with lower Interaction Level, who tend to be more segregated after the social friction event. Indeed, the median of the distribution of the ratios obtained with the Mobility Similarity with the lower and greater Interaction Level are respectively 5.31 and 3, which means that the Mobility Similarity of the refugees with lower Interaction Level decreased 77% more with respect to the

refugees with greater Interaction Level. Based on the obtained results, the proposed metrics are able to capture the effect of a social friction and should be taken into account when addressing application such as attempting to identify or measure the impact of social friction events.

In conclusion, the metrics proposed in the study should be considered when assessing the integration of refugees due to the results obtained, i.e. (i) Mobility Similarity and Calling Regularity are positively and significantly correlated with the level of interaction between refugees and locals, and have proved to offer great potential as measures of the integration related phenomenon with different applications; (ii) the integration is fostered by the simultaneous presence of refugees and locals who reside in the same area in a fair amounts; (iii) the Calling Regularity is also a proxy for refugee's economic capacity, which can imply refugee's employment, and (iv) both Mobility Similarity and the amount of calls made toward the locals are affected by events such as social friction involving refugees; however, the behavior of less integrated refugees appears to be significantly more affected by this kind of events.

4.7 Distributed target detection with swarm of UAVs

In the context of the coordination of UAVs' swarm we consider the problem of discovering static targets in an unstructured environment and minimizing the total time spent to discover targets. The main motivation for this study comes from the request to deal with circumstances where the target and the space of exploration are poorly specified, and the coordination strategy is autonomous, robust, resilient, and adaptive. Indeed, the current UAVs hardware and the available flight control software can offer good solutions to problems in many fields. However, the software available for coordinating the exploration of UAVs swarms is not sufficiently mature: limited flexibility, complex management and application-dependent design are the main issues to solve. We address these issues proposing a swarm coordination algorithm that is adaptive to different circumstances, combining three biologically-inspired processes: Stigmergy, Flocking and Evolution. Specifically, during our study we were able to simulate the proposed solution and test:

- The robustness with respect to error in the target sensing procedure
- The behavioral adaptation of the swarm to the search scenario
- The improvement provided by considering the technological characteristics of the drone in its behavioral model

Each one of the above reported characteristics has been the object of a research work, summarized in the following paragraphs. The swarm exploration algorithm has been implemented using NetLogo, a leading simulation platform for swarm intelligence (Net, 2017) and Matlab, an algorithmic development framework, for the evolutionary algorithm(Mat, 2017), respectively. Fig. 4.27 shows the six search scenarios that have been considered. Each of them is included in a squared area of 200 meters side length, the targets are represented by "x". More specifically:

(a) The Illegal Dump scenario is based on the Abusive Trash Map in Paterno (Sicily), and is composed by 140 trees, 19 differently-sized buildings, 11 groups of targets with an average of 4 targets per group (Tra, 2017).

(b) The Field scenario is a synthetic scenario made by 5 clusters of targets scattered over the area, with about 10 targets per group. There are no obstacles, 80 total drones arranged into 4 swarms, represented by triangular forms, and are placed at the antipodes of the area. Cluster of dots represent targets. There are neither obstacles nor buildings.

(c) The Forest scenario represents a synthetic reconstruction of spread targets in a stand of timber. Here, 20 targets, 400 trees (single obstacles), 80 total drones, arranged into 4 swarms, have been initially placed at the antipodes of the area.

(d) Urban Mine scenarios is derived from real-world examples of areas near Sarajevo, in Bosnia-Herzegovina, with landmine objects, selected from publicly available data (See, 2017). It is made by 40 targets, 59 trees and 28 buildings.

(e) The Urban scenario is characterized by two clusters of 110 total targets placed on two sides of 7 total buildings (represented by an area filled with single obstacles side by side). 40 drones, arranged into 4 swarms, are placed at the antipodes of the area, with no trees at all.

(f) The Rural Mine is derived from real-world examples of areas near Sarajevo, as well as Urban Mine (See, 2017). It is composed by 28 target, 281 trees and 3 buildings.

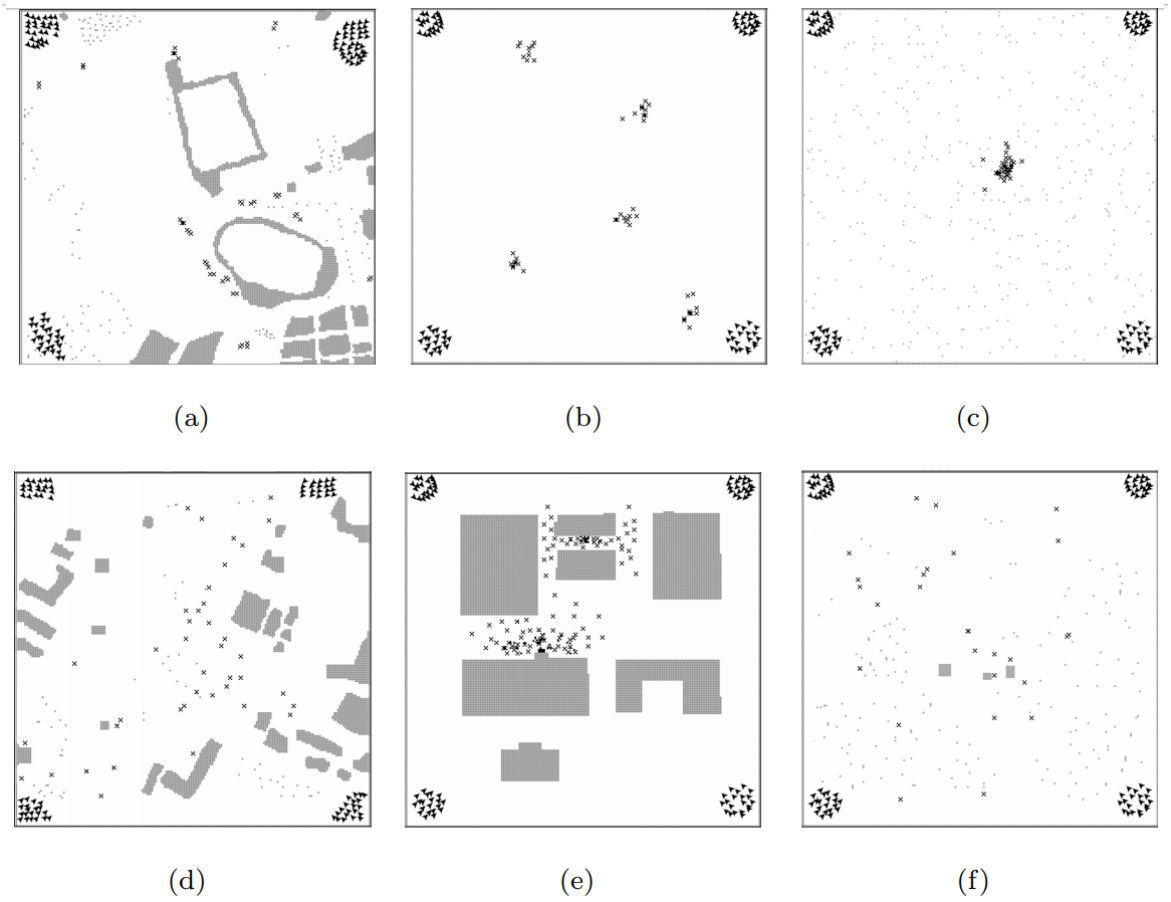


Figure 4.27: Search task scenarios.

Robustness

In recent years, several research groups are working on new procedures and technologies to operate and monitor complex scenarios. Two specific areas include search

and rescue and environmental monitoring. The remote/proximal sensing data obtained using mini-UAVs were validated in several environmental monitoring missions with complex scenarios as reported in previous research; these include: use of thermal imagery to monitor landfills (Lega & Napoli, 2008), surface waters contamination (Lega & Napoli, 2010) and to detect illegal dumping (Lega *et al.* , 2012). The detection, identification and localisation of a target are key elements in all the above operations. Groups of mini-UAVs equipped with self-localisation and sensing capabilities offer new opportunities; indeed, groups of mini-UAVs can explore cluttered outdoor environments, where access to conventional platforms is inefficient, limited, impossible, or dangerous. In this context, the swarming drones could be also considered as single array of sensors configured to the measure of a host of environmental parameters. In search and rescue tasks, for example, a more effective approach is to achieve a quick "survey" of the area to identify key locations as quick as possible. This exclusion process enables organisers to rescan the key locations that provided some circumstantial evidence. In this context, the quality of the sensing has also a direct impact on the overall mission performance (Bertuccelli & How, 2005). Therefore, an important aspect of the swarm coordination is the possibility to require a sufficient number of redundant samples of the target to reliably classify it as "detected" or "undetected". A cooperative approach that exploits drones sensing, minimizes the error in target recognition, moreover swarm intelligence methodologies can be investigated to solve problems cooperatively while maintaining scalability (Bethke *et al.* , 2007). Indeed, we propose a solution based on two swarm intelligence methodologies, Flocking and Stigmergy. Stigmergy, is an indirect coordination mechanism based on the release/perception of pheromones in a shared environment. specifically, a drone releases a particular amount of virtual pheromone in correspondence of the location of the sensed possible target, whose diffusion acts as an attractive potential on neighboring drones. To be attracted by pheromone trails, the available drones should be spatially organized into flocks. Flocking behavior allows the collective exploration of an environment by arranging the UAVs into flexible groups; it is an emergent effect of individual rules based on alignment, separation and cohesion (Reynolds, 1987b). As an effect of pheromone attraction, other drones can confirm the possible target through repeated sensing, and can surround the detected location in order to map the whole distribution. Thus a considerable amount of pheromone is aggregated for each possible target. Once a predefined number of drones confirmed the sensing of the possible target, it is definitively considered to be a true target. To model the sensing error we employ the notion of degradation of the sensing quality as a function of the proximity to the target: as the proximity increases, the sensing may generate an altered measure resulting in a wrong detection.

For the sake of simplicity, the environment is considered as a two dimensional

space. Indeed, it could be inefficient and unreliable to have more than one drone on the same location. In the worst case, drones could interfere with each other degrading the performance even more. From a model design perspective the environment is considered subdivided in cells (corresponding to 1 square meter) and the imperfect sensing probability is uniformly distributed in the cells surrounding the target (Fig. 4.28).

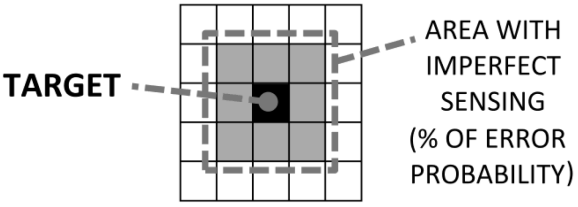


Figure 4.28: Imperfect sensing model.

The environment is mapped in the virtual environment in which the virtual pheromones are released/perceived by the drones. Specifically, the digital environment manages the aggregation, diffusion, and evaporation of the pheromones map. It can be seen as lattice of cells. When a single pheromone (intensity I) is released: (i) it is added to the underlying map; (ii) it is progressively diffused to the nearby cells, with a constant diffusion rate $\delta \in [0,1]$; (iii) it is subsequently evaporated, decreasing its intensity over time by the constant rate $\varepsilon \in [0,1]$. More formally, the pheromone intensity is the output of a scalar function $\mathrm{p}_{(x,y)}(t)$, released at the instant t on the cell (x,y), and characterized by the dynamics:

$$\mathrm{p}_{(x,y)}(t) = \varepsilon \cdot \Big[(1-\delta)\cdot \mathrm{p}_{(x,y)}(t-1) + \partial \mathrm{p}_{(x,y)}(t-1,t) + \partial \mathrm{d}_{(x,y)}(t-1,t)\Big] \quad (4.14)$$

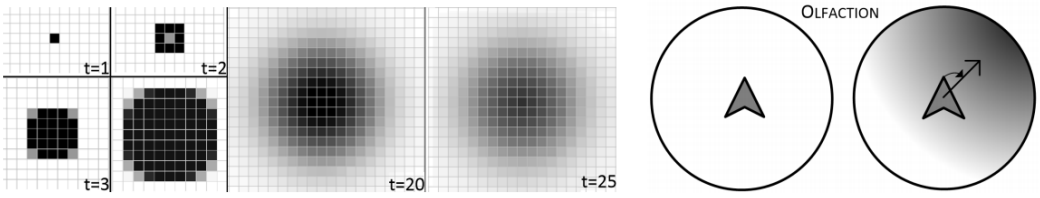
where $(1-\delta)\cdot \mathrm{p}_{(x,y)}(t-1)$ is the pheromone amount remaining on the cell (x,y) after diffusion to nearby cells, whereas $\partial \mathrm{p}_{(x,y)}(t_1,t_2)$ is the additional pheromone released on the cell (x,y) in the interval (t_1,t_2), and $\partial \mathrm{d}_{(x,y)}(t_1,t_2)$ is the additional pheromone diffused from all the nearby cells to the cell (x,y) in the interval (t_1,t_2). The total amount is also multiplied by ϵ to take into account evaporation. The diffused pheromone can be formally calculated in the interval $(t-1,t)$ as:

$$\partial \mathrm{d}_{(x,y)}(t-1,t) = \frac{\delta}{8} \cdot \sum_{i=-1}^{1} \sum_{j=-1}^{1} \mathrm{p}_{(x+i,y+j)}(t-1) \quad (4.15)$$

since each of the 8 neighbor cells propagates the portion δ of its pheromone to the cell (x,y) at each update cycle.

Fig. 4.29(b) illustrates the dynamics of pheromone release: after initial release (t=1) it is first diffused (t=2, 3, etc.) and then evaporated (t=20 and 25). Fig. 4.29(c)

shows the pheromone sensing and acting dynamics: here, the *social insect* called *agent* for concept abstraction follows the direction of major intensity.



(a) (b)

Figure 4.29: (a) Ontology of the attractive pheromone exhibited by ants in foraging process. (b) Illustration of the pheromone release dynamics. (c) Illustration of the pheromone sensing and acting dynamics.

On the other hand, the drone behavior is structured into a prioritized logic, where each priority level implements one basic behavior, or role. At each update cycle, or tick, the role assumed by the drone is a consequence of the environmental sensing. In descending order or priority, the roles are: *obstacle avoider*, *pheromone follower*, *flockmate* and *explorer*. Specifically, every tick period, the drone performs: (i) the target detection, in which case it releases pheromone controlled by *StigDiffusion* and *StigEvaporationRate* parameters; (ii) the *obstacle avoider*, i.e. if a close object is detected, within the *ObstacleVision* radius, the drone points toward a free direction, when available, and moves forward; (iii) if there are no close objects detected, the drones play the **tracker** role: it tries to sense pheromone within the *Olfaction* radius and, if detected, points toward the pheromone peak. (iv) if pheromone is not detected, the drone plays the *flockmate* role: it tries to detect surrounding drones within the *FlockVision* radius, in order to stay in the flock. Finally, (v) if there are no surrounding drones, as an *explorer* it performs a random turn within the *WiggleVar* angle, and then moves forward. The resulting behavior is depicted in the snapshot in Fig. 4.30.

Each one of this behavior is modeled and parametrized. The parametrization is performed in three phases: early analysis, under the assumption of reliable sensing (that is, sensing error probability and sensing redundancy set to 0.1 and 1, respectively); parameter sensitivity analysis on representative scenarios, by evaluating the uncertainty in the output for each parameter; finally, accurate setting on each of the most sensitive parameters, via a bisection method to find the value minimizing the time needed to find 95% of the targets (the search time). For the reader's convenience, Table 4.11 summarizes the main structural and behavioral parameters of the model, with their range and choosen values.

The proposed approach is experimented in the Field, Dumps, Urban and Urban Mine scenarios. To carry out the experiments under the requirement of imperfect

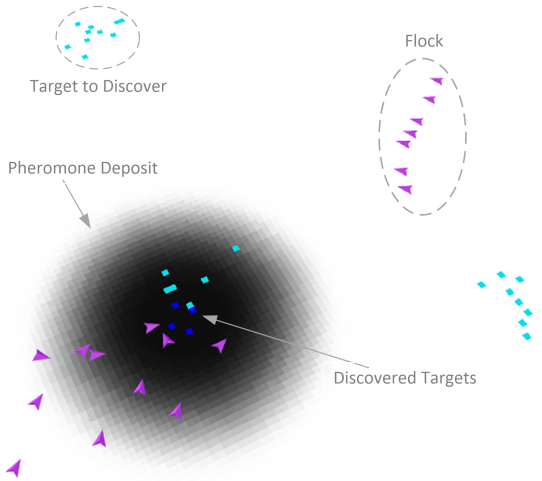


Figure 4.30: A simulation snapshot with flocks, targets and pheromone deposit.

Name	Description	Range	Set value
DroneVel	Drone horizontal speed	(0, 15)	1
WiggleVar	Drone max rand-fly turn angle	(0, 18)	150
ObstacleVision	Drone object sensing distance	(0, 5)	2
FlockVision	Flock visibility radius	(0, 50)	7
MinimumSeparation	Flock mobility distance	(0, 5)	3
MaxSeparateTurn	Flock separation angle	(0, 18)	30
MaxAlignTurn	Flock alignment angle	(0, 18)	20
MaxCohereTurn	Flock cohesion angle	(0, 18)	5
Olfaction	Pheromone sensing distance	(0, ∞)	1
StigIntensity	Pheromone release intensity	(0, ∞)	40K
StigDiffusion	Pheromone diffusion rate	(0, 1)	0.85
StigEvaporation	Pheromone evaporation rate	(0, 1)	0.05
SensingError	Sensing error probability	(0, 100)	{0.1,1}
Redundancy	Sensing Redundancy	(0, ∞)	{1,3,5}

Table 4.11: Behavioral parameters of the presented swarm coordination solution.

sensor model, a sensing error probability in the interval [0.1, 1] percent with uniform distribution has been added. Then, the search time has been evaluated by requiring a prefixed number of repeated measures of the targets in the termination criterion, that is, sensing redundancy values 3 and 5.

To assess the effectiveness of the proposed approach, the performance of the model has been evaluated on three approaches: Random Fly (“R”), Stigmergic approach (“S”), Stigmergic and Flocking aproach (“S+F”) and 3 redundancy values (in parenthesis). For each experiment, 10 trials have been carried out. It has been determined that the resulting performance indicator (i.e. the search time) samples are

well-modeled by a normal distribution, using a graphical normality test. Hence, the 95% confidence intervals have been calculated. Table 4.12 summarizes, for each scenario, the characteristics and the results in the form "mean $\pm$ confidence interval". The results confirm that the use of stigmergy speeds up the target search process in any scenario. Moreover, results become even better in combination with flocking.

Table 4.12: Characterization and performances obtained for each scenario.

	Field	**Dumps**	**Urban**	**Urban Mines**
# targets	50	30	110	40
# clusters	5	3	2	40
# trees	0	100	0	54
# buildings	0	0	7	28
# drones	80	80	40	25
R (1)	2,604 $\pm$ 248	2,252 $\pm$ 212	2,340 $\pm$ 229	651 $\pm$ 55
S (1)	1,383 $\pm$ 126	1,297 $\pm$ 102	1,748 $\pm$ 188	560 $\pm$ 49
S+F (1)	1,078 $\pm$ 106	1,009 $\pm$ 141	1,259 $\pm$ 102	487$\pm$ 29
R (3)	4,161 $\pm$ 269	3,993 $\pm$ 266	3,688 $\pm$ 286	944 $\pm$ 55
S (3)	1,758 $\pm$ 151	1,513 $\pm$ 116	2,089 $\pm$ 197	707 $\pm$ 84
S+F (3)	1,484 $\pm$ 147	1,289 $\pm$ 135	1,861 $\pm$ 166	594 $\pm$ 34
R (5)	6,173 $\pm$ 361	6,163 $\pm$ 399	4,647 $\pm$ 271	1,167 $\pm$ 51
S (5)	2,109 $\pm$ 246	2,208 $\pm$ 208	2,488 $\pm$ 280	770 $\pm$ 93
S+F (5)	1,591 $\pm$ 136	1,823 $\pm$ 233	2,102 $\pm$ 151	726 $\pm$ 32

To better highlight the scalability of our approach against redundancy, Fig. 4.31 shows the completion time for redundancy 1, 3 and 5, for each scenario. Here, it is apparent that Stigmergy introduces a significant improvement of trend over Random Fly, both alone and combined with flocking behavior.

Provided results prove the benefits of both stigmergy and flocking, in terms of tolerance to errors and scalability for increasing redundancy requirements. However, the overall mechanism can be better enabled if structural parameters are correctly tuned for the given scenario. Determining such correct parameters is not a simple task since different areas have different features. Thus, an appropriate tuning to adapt parameters to the specific search area is desirable to make the search more effective.

Adaptation

In this work we addressed the issues of providing an effective parametrization by proposing a swarm coordination algorithm that is adaptive to different circumstances. This is obtained by combining three swarm intelligence processes: Flocking, Stigmergy, and Evolution (FSE). *Stigmergy* is a reliable mechanism for generating autonomous swarm coordination. Basically, a mark is realized in the form of virtu-

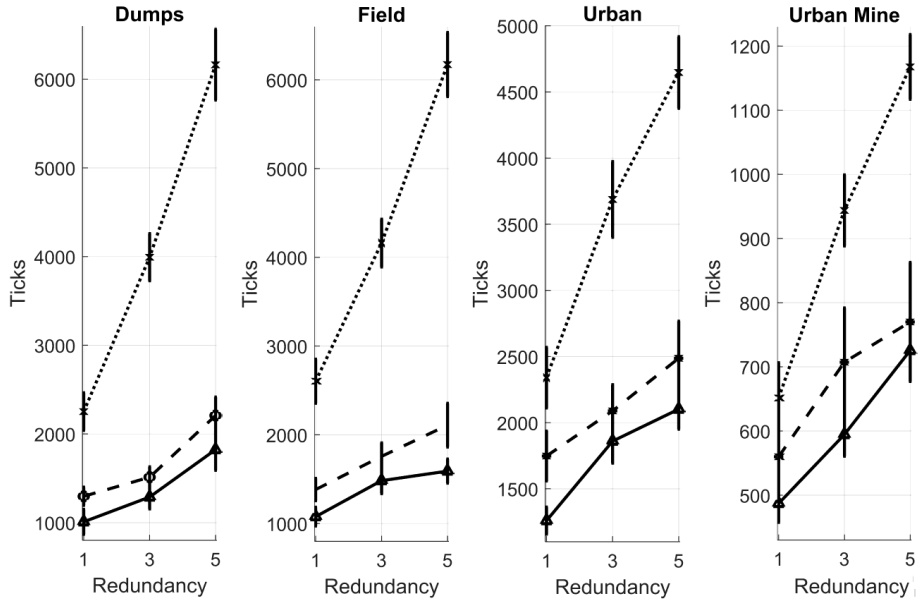


Figure 4.31: Completion time against redundancy, for each scenario and with different approaches: Random Fly (dotted line), Stigmergy (dashed), and Stigmergy + Flocking (solid).

alized pheromone, released in the corresponding position of the target just found. Such pheromones diffuses to be sensed by other drones and acts as an attractive potential on neighboring drones (Ermacora *et al.* , 2013). While unknown targets are discovered, additional pheromone is released by flock members, thus enabling an incremental positive feedback up to completion of all targets in the proximity of the initial target. After a certain time the pheromone intensity cannot be reinforced, and in practice disappears. The implementation of the virtual pheromone dynamics and of the virtual environment is based on the same model presented in the last paragraph. Furthermore, to reduce multiple explorations of the same zone, two additional coordination mechanisms are used: (i) olfactory habituation (Glanzman, 2011) and (ii) repulsive pheromone (Amrein, 2004). In essence, (i) a drone releases a repulsive pheromone where it does not sense a target, and (ii) a drone becomes unable to sense pheromone while moving in locations saturated by pheromones.

The available drones are spatially organized into a number of flocks. Flocking behavior is an effect of local rules based on alignment, separation and cohesion (Reynolds, 1987b). With respect to the original flock model presented in Chapter 1, we have verified that: (a) the cohere is not suitable for drones moving between many obstacles; (b) the separation can be better exploited with an area different than circular, as represented in Fig. 4.32.

Results in (Alfeo *et al.* , n.d.) prove the benefits of the combined strategy of stigmergy and flocking and raise the need of an appropriate adaptation of the stigmergy

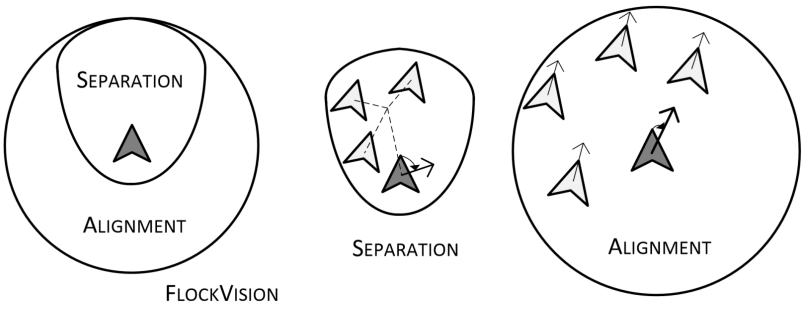


Figure 4.32: Flocking agent procedures for drones: search area, separate and alignment.

and flock parameters to the specific search area. For example, an area with a high density of obstacles can be efficiently explored via small flocks. In general great benefits may be provided by adjusting each swarm behavioral parameter to a given search scenario. Table 4.13 summarizes the parameters to adapt for each search scenario.

Table 4.13: Description of the parameters of the FS algorithm

Parameter	**Description (unit measure)**
droneVel	Drone horizontal speed (m/s)
wiggleVar	Drone max rand-fly turn angle (°)
obstacleVision	Drone object sensing distance (m)
flockVision	Flock visibility cone radius (m)
minimumSeparation	Flock separation distance (m)
maxSeparateTurn	Flock separation turn limit (°)
minFlockAngle	Flock separation front cone (°)
maxAlignTurn	Flock alignment turn limit (°)
stigIntensity	Pheromone release intensity
stigDiffusion	Pheromone diffusion rate on surrounding cells (%)
stigEvaporation	Pheromone evaporation rate (%)
olfaction	Pheromone sensing distance (m)

For the parameters adaptation we adopt an evolutionary computation technique, which improves the parameters with regard to a given measure of quality. Evolutionary computation has the advantage of making no assumptions about the problem being optimized, thus avoiding to bias the underlying mechanisms. It is based on a population of candidate solutions, iteratively improved via mechanisms inspired by natural evolution, such as mutation, selection, and crossover. Specifically, we adopted the Differential Evolution (DE) algorithm, which is one of the most powerful stochastic real-parameter optimization algorithms (Das & Suganthan, 2011). In

essence, DE operates through conventional operators of evolutionary computation, except that it perturbs the current population via scaled differences of randomly selected members.

As specified, to exploit DE we need to define a metric of quality of the swarm task, i.e. the fitness of the solution. The fitness can be defined as following. Given a simulated scenario Σ, made of: (i) simulation instants of time $t \epsilon \mathbb{N}^+$; (ii) a set of drones $\{D_i\}$, each drone having a dynamic position $(x_t, y_t)_{Di}$; (iii) a set of targets $\tau \epsilon \mathrm{T}$, each target having a fixed position $(x, y)_\tau$. The set of targets already found $\mathrm{T}_F(t) \subseteq \mathrm{T}$, at a given instant of time t, is the set of targets for which it exists a time $t' \leq t$ and a drone d such that the drone's position corresponds to the target's position:

$$\mathrm{T}_F(t) = \{\tau | \exists D_i, \exists t' \leq t : (x_{t'}, y_{t'})_{Di} = (x, y)_\tau\} \tag{4.16}$$

The fitness of the simulated scenario Σ is the minimum instant of time for which $\mathrm{T}_F(t)$ has cardinality greater than or equal to 0.95 of the cardinality of T:

$$\text{fitness}(\Sigma) = min_{t \epsilon \mathbb{N}^+} \{t : |\mathrm{T}_F(t)| \geq 0.95 \cdot |\mathrm{T}|\} \tag{4.17}$$

From an implementation point of view, the fitness is the results of the execution of a simulated search task, done in accordance to the drone behavior, executed on each drone until the search is over (i.e. 95% of the targets are found) as detailed in the following.

In the FS algorithm, each drone periodically carries out a target check on its location. If there is an unknown target, it releases an amount *stigIntensity* of attractive pheromone on its location. If no target is found, the drone releases repulsive pheromones. Subsequently, if there are drones or objects within *obstacleVision* radius, then the drone turns away. If there are attractive pheromones within *olfaction* radius, the drone turns toward the maximum amount of it. If drone is located at the maximum, it is subject to olfactory habituation. When there are no pheromones, if the drone detects flockmates in a separation area, it slows down and turns away from them (at most by *maxSeparateTurn*).

The separation area is created by union of a circular area of radius *minimumSeparation* around the drone, with a conical area in front of the drone, sized by *minFlockAngle* and *flockVision*. If there are no flockmates found in this area, the drone steers towards the drones in a larger circular area (with radius *flockVision*) by a quantity within *maxAlignTurn*. Subsequently, if neither drones, nor pheromones, nor obstacles are found, the drone steers towards the minimum of repulsive pheromone, if available, otherwise randomly. Finally, it moves forward. More formally, the part of

the FS algorithm can be summarized by the pseudocode presented in the following algorithm.

Algorithm 4: Coordination algorithm based on Flocking and Stigmergy

```
function executeFS(Swarm, Obstacles,
  Targets,AttractivePheromones,RepulsivePheromones)
tick = 0; cardT_F = 0; cardT = count(Targets) ;
do
    for each drone D_i in Swarm do
        if (D_i.position in Targets) then
            markTargetAsFound(Targets, D_i.position); cardT_F = cardT_F + 1;
            releasePheromone(AttractivePheromones, D_i.position);
        else
            releasePheromone(RepulsivePheromones, D_i.position);
        if (D_i.obstacleVision() intersects Obstacles) then
            turnDrone(D_i.heading);
            else if (D_i.olfaction() intersects AttractivePheromones) then
                turnDroneTowardsMax(D_i.heading,AttractivePheromones);
                else if (D_i.flockmates in D_i.separateArea()) then
                    turnDroneWithin(D_i.heading, maxSeparateTurn);
                    else if (D_i.heading != D_i.flockmates().meanHeading) then
                        turnDroneWithin(D_i.heading,maxAlignTurn);
                        else if (D_i.olfaction() intersects RepulsivePheromones) then
                            turnDroneTowardsMin(D_i.heading,RepulsivePheromones);
                        else turnDroneWithin(D_i.heading, wiggleWar);
        moveForward(D_i.pos,D_i.heading,D_i.vel);
    evaporatePheromone(AttractivePheromones,RepulsivePheromones);
    tick = tick + 1;
while cardT_F < 0.95 · cardT;
return tick;
```

To make realistic assumptions, some requirements and constraints of sensors and drones have been studied, such as the time limit and drone velocity. In (Neumann *et al.* , 2013) gas detection is provided via drones using a 1Hz rate sensor. In (Rodriguez *et al.* , 2014) drones to detect mines using image recognition through camera allow to have 80% recognition precision flying at 2.2 m/s. To fit among different needs of each scenario and to raise the precision of our recognition, the drone velocity is set to 1 m/s. Assuming a drone battery life of 25 minutes, the fly time is actually lower, due to the time needed to return to the place where drones are gathered (Ermacora *et al.* , 2013). Regarding the evolutionary approach, in this work, we refer to (Das & Suganthan, 2011) and use the most common DE variant, called "DE/rand/1/bin" according to a naming convention known as "DE/x/y/z". In this convention, "DE" stands for differential evolution, "x" the base vector to be

perturbed, "y" is the number of difference vectors considered for perturbation of "x", and "z" the type of crossover being used. Thus, the used variant is characterized by a perturbation with randomly ("rand") selected members, only one ("1") weighted difference vector, and a binomial ("bin") crossover used in conjunction. Apart from the DE variant, some meta-parameters must also be chosen. Specifically, in the optimization process, the differential weight F belongs to $[0, 2]$, and mediates the generation of the mutant vector. The component of a child vector is taken with probability CR from the mutant vector and with probability 1-CR from the target vector. In (Cimino *et al.* , 2016) an early application of DE is studied and proposed to parameterize the coordination of a group of drones whose schema is obtained by employing flocking and digital pheromones. The provided results reveal as good value for CR and F are respectively 0.5 and 0.7.

To obtain the best performance algorithm, we have experimented a number of variants of the proposed approach. More specifically: (i) in the $F_{ASC}S_AE$ algorithm, Flocking includes align, separation and cohere, Stigmergy includes attractive pheromone and finally Evolution is also available; (ii) in the $F_{AS}S_AE$ algorithm the cohere (Reynolds, 1987a) has been removed from Flocking; (iii) In the $F_{ASS}S_AE$ the scatter procedure (Fig.3b) has been incorporated into the separate flocking procedure (Cimino *et al.* , 2016); (iv) in the $F_{ASS}S_{AR}E$ the repulsive pheromone has been also added (Cimino *et al.* , 2015c); (v) in the $F_{ASS}S_{ARO}E$, to further speed the search, the olfactory habituation procedure has been added. Table 4.14 shows the performance of the different variants, in terms of 95% confidence interval over 10 trials of completion time (ticks) on each scenario.

Table 4.14: Performance of different variants of the FSE algorithm.

Version	Dump	Field	Forest	Rural Mine	Urban	Urban Mine
$F_{ASC}S_AE$	1043 ± 133	791 ± 109	569 ± 157	1658 ± 247	933 ± 218	1595 ± 296
$F_{AS}S_AE$	895 ± 179	507 ± 58	425 ± 36	1693 ± 276	818 ± 176	1356 ± 191
$F_{ASS}S_AE$	866 ± 205	546 ± 101	416 ± 76	1030 ± 123	932 ± 122	880 ± 59
$F_{ASS}S_{AR}E$	743 ± 45	511 ± 37	400 ± 28	981 ± 94	720 ± 56	878 ± 53
$F_{ASS}S_{ARO}E$	638 ± 33	356 ± 47	331 ± 25	783 ± 84	562 ± 33	846 ± 74

Results in Table 4.14 clearly shows that the different variants give an incremental improvement of performance. Considering the average performance over all scenarios, the improvement from the first to the last variant is 47% in all scenarios. Table 4.15 shows the adaptive capability of the $F_{ASS}S_{ARO}E$ algorithm.

As a final outcome, a performance comparison with previous approaches based on flocking and stigmergy is presented in Table 4.16, namely *Basic S+F* (Cimino *et al.* , 2015c) and *Adaptive S+F** (Cimino *et al.* , 2016).

Table 4.15: Performance before and after the Adaptation

Scenario	**Before Adaptation**	**After Adaptation**
Dumps	1132 ± 122	927 ± 71
Field	1149 ± 162	500 ± 55
Forest	1189 ± 128	515 ± 65
Rural Mine	1029 ± 59	947 ± 57
Urban	1224 ± 198	644 ± 51
Urban Mine	1228 ± 99	1109 ± 67

Table 4.16: Performance comparison with other approaches in the literature.

Scenario	*Basic S+F* (Cimino *et al.* , 2015c)	*Adaptive S+F** (Cimino *et al.* , 2016)	FSE approach
Dump	934 ± 216	757 ± 112	638 ± 33
Field	589 ± 86	582 ± 121	356 ± 47
Forest	602 ± 124	593 ± 146	331 ± 25
Rural Mine	1530 ± 225	1123 ± 116	783 ± 84
Urban	890 ± 93	666 ± 100	562 ± 33
Urban Mine	1704 ± 225	1025 ± 76	846 ± 74

Enhancing the behavioral model by including the UAVs technological capability

In this work the swarm behavior is modeled considering and comparing two different paradigms, namely *biological behavior*, which mimics social animal metaheuristics (exploited in last paragraphs), and *computational behavior*, which considers the enhancements obtained by exploiting UAVs configuration and information technology. Our aim is to verify whether some enhancements allow a reduction of complexity and a more effective optimization of the structural parameters. As an example, the digital environments can allow a specialization of biological models able to simplify both mechanisms and dynamics and then the research space. Nevertheless, this computational behavior should keep the essential benefits of the original biological behavior.

Specifically, the *computational behavior* expands the *biological behavior* with the following possible improvements:

(i) *computational stigmergy*: since the digital pheromone is maintained in a virtual space, called pheromone map, it can have an instant diffusion, to immediately propagate the environmental information to nearby UAVs; furthermore, a *linear evaporation* and a *streamlined shape* (Fig. 4.33(d))allow a better control of the aggregated pheromone potential;

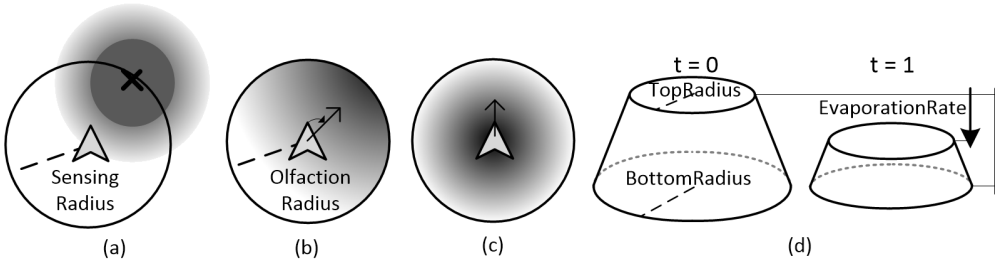


Figure 4.33: Computational Stigmergy. Pheromone Release (a), Pheromone sensing/attraction (b), Olfactory habituation (c), Pheromone Mark Implementation (d)

(ii) *computational flocking*: the simultaneous localization and mapping technology allows different-scaled flocking, ranging from visually-based flocks to large-scale localization-based formations, across physical barriers.

(iii) *computational sensing/actuation*: long-range sensing allows detection of remote targets and obstacles; as a consequence, computational pheromone can be released on a target remote with respect to the UAV position (Fig. 4.33(a)); moreover, a UAV can accelerate, decelerate and keep cruise speed while obstacles are not detected. Finally, olfactory receptors can decrease in sensibility over time to prevent overstimulation (olfactory habituation).

Given the reported characteristics, the behavioral parameters of the computational model are summarized in Table 4.17.

Table 4.17: Stigmergy and Flocking parameters.

Procedure	**Parameter Name**	**Measure**
Stigmergy	MarkTopRadius	*cells*
Stigmergy	MarkBottomRadius	*cells*
Stigmergy	MarkIntensity	*unit*
Stigmergy	EvaporationRate	*ratio*
Stigmergy	OlfactoryHabituationTime	*ticks*
Stigmergy	OlfactionRadius	*cells*
Flocking	FlockAngle	*degree*
Flocking	SeparationRadius	*cells*
Flocking	MaxSeparationTurn	*degree*
Flocking	AlignRadius	*cells*
Flocking	MaxAlignTurn	*degree*
Flocking	CohereRadius	*cells*
Flocking	MaxCohereTurn	*degree*
Flocking	MaxRandomTurn	*degree*

To have a more realistic simulation we take into account UAVs cruise speed, acceleration, angular velocity, battery duration, UAV size, and sensing. These charac-

teristics have been set considering the technical specifications of the UAV possibly used in the provided scenarios. Specifically, for the scenarios Dump, Field and Forest, the UAV used is DJI Inspire 2 with Zenmuse X5S video camera (Cruz *et al.* , 2016), *DJI2+X5S vc* for short. For the scenarios Rural Mine, Urban and Urban Mine, the UAV is equipped to detect gas and landmines (Gade & Moeslund, 2014) such as the DJI Inspire 1 with Zenmuse XT thermal camera (Chiaraviglio *et al.* , 2016), *DJI1+XT tc* for short. Table 4.18 summarizes all structural parameters.

Table 4.18: Structural parameters of the UAV models.

Parameter Name	**Measure**	***DJI2+X5S vc***	***DJI1+XT tc***
Max Speed	m/s	26	22
Cruising speed	m/s	3	3
Max Acceleration	m/s^2	4	4
Max Angular velocity	$degree/s$	150	150
Endurance time	s	1620	1080
UAV size	m	0.6	0.6
Sensing radius	m	2	4
ObstacleVision	m	3	3
ObstacleVisionAngle	$degree$	15	15

By considering the UAV camera pointing downwards, it is possible to calculate the radius of a circular sensing area (*Sensing Radius*). Moreover, we consider a UAV flight altitude of 3 meters, to navigate under the canopy or in a cluttered environment (Israel, 2011). A UAV cruising speed no higher than 3 m/s is assumed to ensure a good target recognition performance (Rodriguez *et al.* , 2014).

Finally, in order to investigate the performance and the properties of the proposed approach, we propose a set of quality measures :

(i) *Effectiveness*: since battery duration is a critical feature of current UAVs, a measure of the mission effectiveness is the average completion time, i.e. the average time to find the 95% of the targets.

(ii) *Efficiency*: the efficient swarm carries out a quick survey of the scenario (exploration) and an in-depth search of areas characterized by targeted cells (exploitation). The trade-off between exploration and exploitation is the target search efficiency. Formally, let $v_{(x,y)}$ be the number of visits that the cell (x, y) of the environment has received during the mission, and $d_{t(x,y)}$ the distance of (x, y) from the closest target. Thus, we expect that for an efficient search $v_{(x,y)}$ is large/small for cells that are close to/far from a target. In other terms, the

average ratio between $v_{(x,y)}$ and $d_{t(x,y)}$ is high:

$$SearchEfficiency = \frac{1}{n} \cdot \sum_{x=1}^{\sqrt{n}} \sum_{y=1}^{\sqrt{n}} \frac{v_{x,y}}{d_{t(x,y)}} \tag{4.18}$$

(iii) *Scalability*: By increasing the number of UAVs the search time is expected to be lower, due to the additional resources. However, the search time can be negatively affected by the number of path deviations caused by collision avoidance between many UAVs. The average search time over repeated trials, calculated for increasing numbers of UAVs, provides a scalability measure.

(iv) *Adaptive cooperation*: on average it is expected that: (a) stigmergy attracts mainly flock mates, i.e., the average flock size and the pheromone size should be similar; (b) a scenario with isolated/aggregated targeted cells determines small/large flocks; (c) obstacles determine flock fragmentation. To measure such effects, the following measures should be compared: (i) the (average) width of flocks, i.e. the (average) maximum distance between two flock mates; (ii) the average number of isolated UAVs, i.e. UAVs not belonging to any flock; (iii) the average number of non-flock mates attracted by a pheromone deposit (*flock dynamism*). Such measures are based on a procedure able to dynamically identify the flock composition. For this purpose, we adopt a well-known density-based clustering algorithm, DBSCAN. specifically, an agent p is a *core agent* if at least $minPts$ agents are within distance ϵ of it. Such agents are said to be directly reachable from p. An agent q is reachable from p if there is a path $p_1, ..., p_n$ with $p_1 = p$ and $p_n = q$, where each p_{i+1} is directly reachable from p_i and all the agents on the path are core agents. If p is a core agent, then it forms a flock together with all agents that are reachable from it.

By exploiting the maximum distance of flockmate's interaction (i.e. the CohereRadius) as ϵ we are able to generate clusters that correspond to the arrangement of the UAVs in the flocks. Specifically, a group of directly reachable UAVs are considered a flock, while the ones classified as noise are considered single UAVs (i.e. not belonging to any flock).

For each measure, we present the 95% Confidence Interval (CI) over 5 repeated trials. Table 4.19 shows the mission duration for different computational *sensing radiuses* (abbreviated as *Sens.Radius* in the tables), assuming the computational actuation, i.e., remote sensing and remote pheromone release. Here, the enhancement provided by computational sensing and actuation is apparent. Table 4.20 clearly shows the same experiment assuming the biological actuation, i.e., remote sensing and local pheromone release. Comparing Table 4.19 and Table 4.20 it is apparent

that the complete computational sensing and actuation sensibly improves performance.

Table 4.19: Mission duration with computational sensing and actuation.

Scenario	**Sens.Radius 2**	**Sens.Radius 6**	**Sens.Radius 15**
Dump	363.2 ± 102.6	238.4 ± 115.9	108.6 ± 42.8
Field	115.4 ± 45.7	52.8 ± 7.5	35.4 ± 1.41
Forest	334.8 ± 73.3	181.2 ± 88.0	123.2 ± 25.04
Rural Mine	195.8 ± 49.6	111.8 ± 68.5	38.6 ± 2.42
Urban	801.6 ± 310.7	727.8 ± 511.8	286.2 ± 179.1
Urban Mine	303 ± 85.7	160 ± 43.89	100.2 ± 23.46

Table 4.20: Mission duration with computational sensing.

Scenario	**Sens.Radius 2**	**Sens.Radius 6**	**Sens.Radius 15**
Dump	440.6 ± 192.58	234.2 ± 129.8	245.2 ± 91.4
Field	139.8 ± 66.4	61.2 ± 15.9	49.2 ± 11.59
Forest	372.4 ± 66.2	246.4 ± 73.2	186.6 ± 65.2
Rural Mine	193 ± 33.8	94.8 ± 23.28	40.6 ± 1.11
Urban	1070.8 ± 393.2	722.6 ± 360.0	942 ± 695.8
Urban Mine	363.8 ± 134.5	195.4 ± 35.2	129.4 ± 37.31

Based on these results, the computational behavior is compared with the biological behavior based only on adaptive Stigmergy and Flocking, called "*S+F**" presented in (Cimino *et al.* , 2016). It is worth noting that the *S+F** algorithm supports a basic obstacle avoidance that does not detect drones as obstacles. Thus, UAVs overlapping are possible. To have comparable results, the same UAV parameters reported in (Cimino *et al.* , 2016) have been used. The results in Tab. 4.21, clearly shows that the computational approach outperforms the "*S+F**" approach, although the latter is not constrained by UAV avoidance. Indeed, in the Urban scenario, which is characterized by highly dense targeted cells, the "*S+F**" approach outperforms the computational approach. However, it is worth noting that the number of UAV overlapping during the execution of the "*S+F**" approach on this scenario is very high. Thus, the related result is not realistic since it does not consider the real-world constraints by providing an excess of freedom to UAV flight.

Table 4.22 and Table 4.23 show the mission duration and the search efficiency when considering the adaptation of the evolution metaheuristics. Here, the human adaptation is provided by means of heuristics based on simple statistics: (i) *MaxRandomTurn* and *FlockAngle* are set to 120 and 60 degrees, respectively; (ii) to avoid overlapping the UAVs fields of view, *SeparationRadius* is set to 2 cells, whereas *AlignRadius* and *CohereRadius* are set to 4 and 8 cells, respectively; (iii) the flocking

Table 4.21: Mission duration with computational approach and *"S+F*"* approach.

Scenario	***"S+F*"***	**Computational Approach**
Dump	927 ± 71	919.9 ± 180.9
Field	500 ± 55	292 ± 41.9
Forest	515 ± 65	511.3 ± 38.9
Rural Mine	947 ± 57	501 ± 95.3
Urban	644 ± 51	930.4 ± 114.1
Urban Mine	1109 ± 67	902.6 ± 133.2

parameters, *SeparationRadius*, *AlignRadius* and *CohereRadius*, have been set accordingly to 10, 15 and 20 cells; (iv) *MarkTopRadius* has been set to 6 cells, since it must be greater than *ObstacleVision* to avoid UAV overcrowding; (v) *MarkBottomRadius* has been set to 8 cells, since this is the average distance between targeted cells in the scenarios; (vi) *OlfactoryHabituationTime* is set to 3 ticks to allow the UAV to cover a distance of *markBottomRadius* which allows to go far from its pheromone.

Table 4.22: Mission duration considering the adaptation process performed by DE.

Scenario	**Random Walk**	**Human Adaptation**	**DE Adaptation**
Dump	701.8 ± 87.8	624.4 ± 382.7	419.4 ± 157.0
Field	236.6 ± 14.2	198.6 ± 43.3	119.2 ± 16.8
Forest	532.2 ± 130.1	602.2 ± 171.1	318.2 ± 26.4
Rural Mine	206.0 ± 25.4	200 ± 31.4	110.4 ± 58.9
Urban	1760.2 ± 398.6	3056.2 ± 1919.8	801.6 ± 310.7
Urban Mine	217.2 ± 56.7	200.6 ± 17.7	198.2 ± 35.2

Table 4.23: Search Efficiency considering the adaptation process performed by DE.

Scenario	**Random Walk**	**Human Adaptation**	**DE Adaptation**
Dump	0.0838 ± 0.004	0.0737 ± 0.004	0.0652 ± 0.002
Field	0.0680 ± 0.057	0.0194 ± 0.001	0.0156 ± 0.001
Forest	0.0229 ± 0.002	0.0247 ± 0.002	0.0138 ± 0.001
Rural Mine	0.0274 ± 0.001	0.0269 ± 0.002	0.0264 ± 0.001
Urban	0.5294 ± 0.175	0.6630 ± 0.169	0.1937 ± 0.051
Urban Mine	0.0270 ± 0.002	0.0262 ± 0.002	0.0259 ± 0.001

The results obtained with the DE adaptation shows a good improvement of performances in all scenarios, with respect to the other approaches. Is it worth noting that in many scenarios the Human Adaptation does not outperforms the Random Walk. This result shows the structural importance of the DE optimization.

The better search efficiency of the DE adaptation is also confirmed by Fig. 4.34, where the distribution of number of cells with a given number of visits is shown. Here, an efficient search produces narrower and more-to-the-left distributions.

To measure the internal UAV organization caused by the DE adaptation, Table 4.24 and Table 4.25 shows the adaptive cooperation measures with Human and DE adaptation, respectively. As a first result, in the DE adaptation, each scenario is characterized by a lower number of isolated UAV with respect to the Human adaptation. This corresponds to a better swarm formation.

Table 4.24: Adaptive cooperation measures with human adaptation.

Scenario	**Flock Size**	**Isolated UAVs**	**Flock Dynamism**
Dump	28.463 ± 0.215	47.539 ± 0.347	0.970 ± 0.182
Field	24.473 ± 0.293	47.084 ± 1.119	0.983 ± 0.193
Forest	25.788 ± 0.198	51.196 ± 0.395	1.396 ± 0.280
Rural Mine	25.155 ± 0.343	47.830 ± 0.806	1.437 ± 0.231
Urban	25.899 ± 0.109	50.593 ± 0.129	1.333 ± 0.178
Urban Mine	29.326 ± 0.451	42.098 ± 0.757	1.000 ± 0.179

Table 4.25: Adaptive cooperation measures with DE adaptation.

Scenario	**Flock Size**	**Isolated UAVs**	**Flock Dynamism**
Dump	34.334 ± 0.350	5.104 ± 0.119	0.190 ± 0.070
Field	9.213 ± 0.365	35.721 ± 1.236	3.033 ± 0.367
Forest	24.901 ± 0.261	11.114 ± 0.214	0.842 ± 0.247
Rural Mine	16.816 ± 0.361	17.922 ± 0.498	0.148 ± 0.067
Urban	9.875 ± 0.076	29.726 ± 0.221	2.041 ± 0.234
Urban Mine	29.964 ± 0.488	7.910 ± 0.229	0.858 ± 0.174

On the other hand, the DE adaptation determines flock configurations that are peculiar to the scenario structure. For example, Dump, Forest and Urban Mine scenarios are characterized by numerous and small obstacles, which tend to crumble dense and rigid flocks. For this purpose, a good strategy is to create large and sparse flocks, which are more resilient to small obstacles getting across. This also results in a lower number of isolated UAVs shown in Table 4.25.

Figure 4.35 shows the scalability of the proposed approach, considering the confidence intervals of mission duration against the number of drones. Overall it shows that for each scenario, to increment the number of drones produces a different degree of improvements. It is worth noting that for some scenario characterized by dense targeted cells or dense obstacles such as Urban, Field, and Rural Mine, some nonlinear phenomenon can occur due to the complex obstacle avoidance situations.

In this work, we have enhanced basic biologically inspired metaheuristics for modeling and optimizing UAVs coordination in target search, taking into account technological and computational advances. A novel coordination logic is proposed, by designing stigmergy, flocking, and sensing/actuation metaheuristics. The combined metaheuristics are logically integrated and adapted to specific missions via a

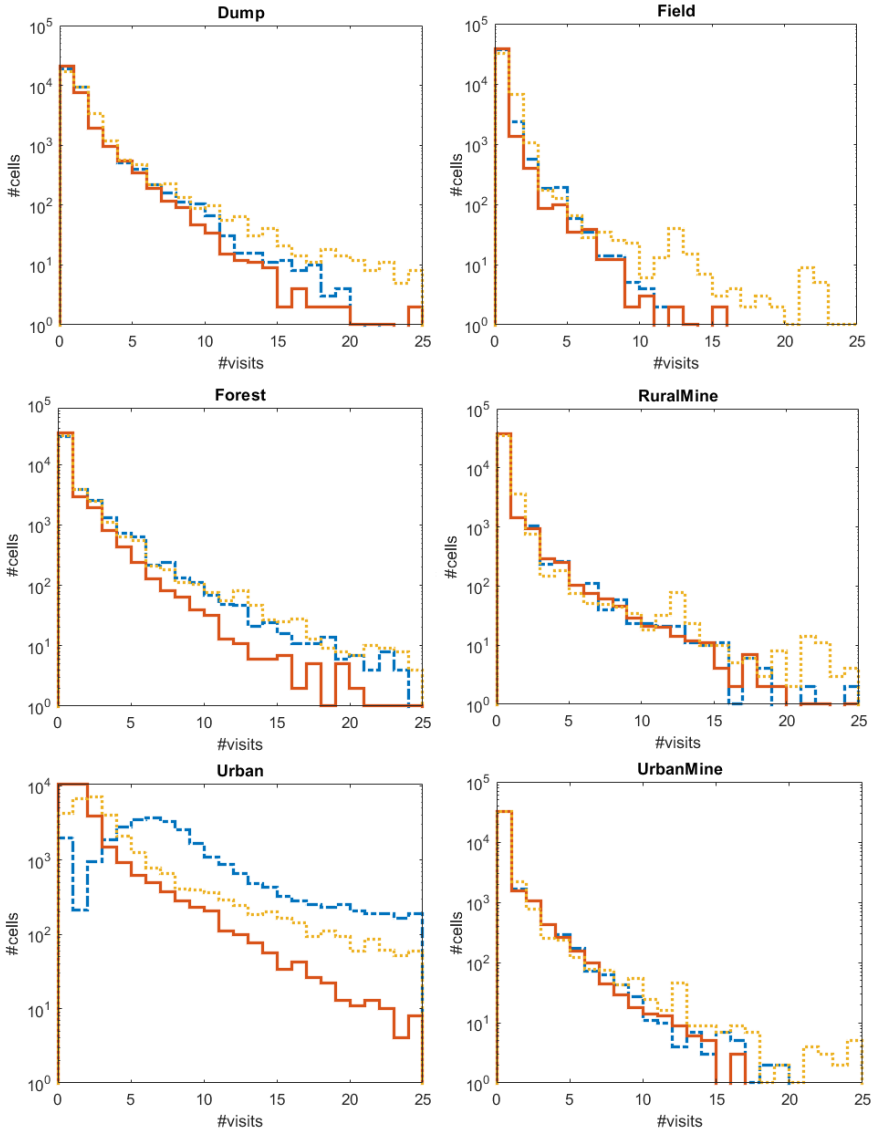


Figure 4.34: Search efficiency: number of cells against number of visits, with Random Walk (dotted line), Human Adaptation (dashed line), and DE adaptation (bold line).

differential evolution optimization. The system has been simulated on synthetic and real-world scenarios, by considering UAV commercial models. Experimental results show that (i) the computational advances sensibly improve the performance and the realism of the biologically metaheuristics; (ii) the differential evolution optimization provides significant and structural improvements to the coordination logic; (iii) the quality of the resulting cooperation is better in terms of swarm formation, search efficiency, strategy and scalability.

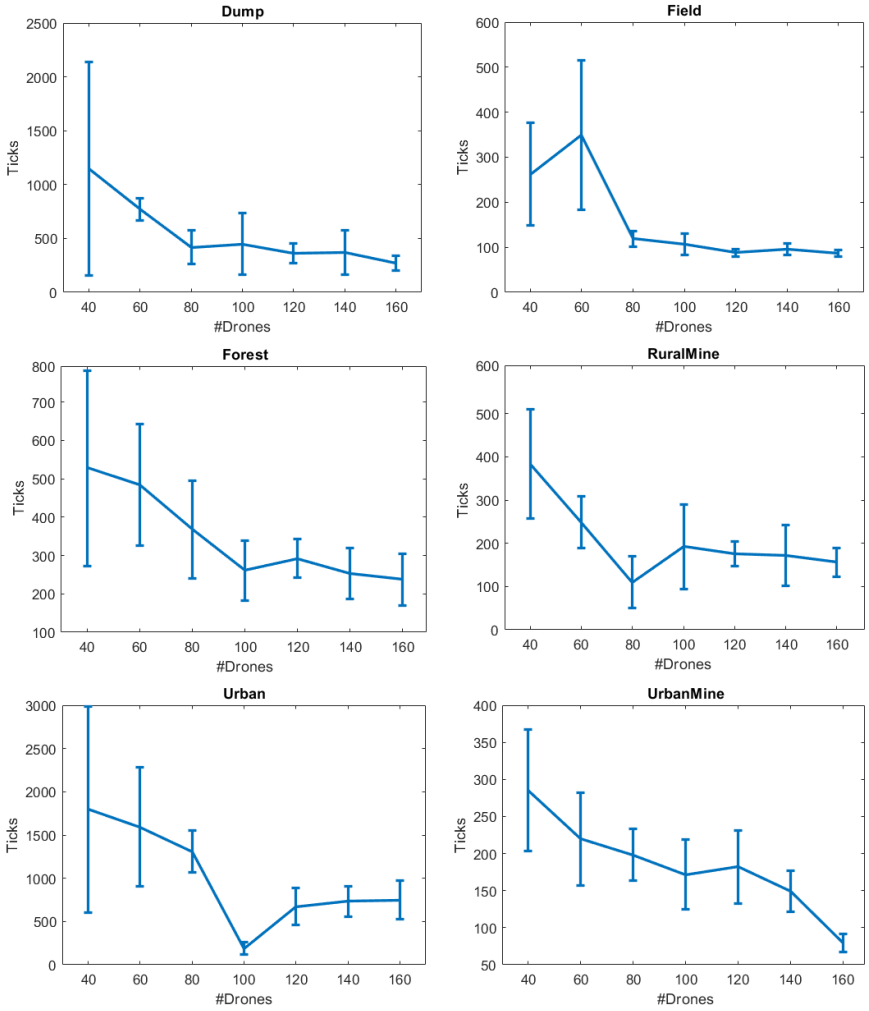


Figure 4.35: Mission duration with the computational approach: confidence intervals against UAVs numbers.

4.8 Urban Trash Disposal Service via Swarm of Robots

Problem Statement

The world is urbanizing at an unprecedented rate of which humanity has never witnessed before (Shahrokni *et al.* , 2014). UN-Habitat estimates that by 2050, 3.5 out of the 9.1-billion global residents will be living in informal urban communities (Habitat, 2016). On the one hand, novel urban infrastructures together with new technologies such as IoT, 5G, LiDAR, etc. allow to understand the city as a senseable, programmable, and actuable ecosystem (Shahrokni *et al.* , 2014). On the other hand, this urbanization implies important social and environmental challenges such as

fuel production, air pollution, etc. (Kumar *et al.* , 2015). Experts estimate that it will require $57 trillion US dollars to adapt traditional heavy infrastructures to the informal urban needs (Jahan, 2017) and that today's solutions will not be able to scale at the pace urbanization is taking place. One of the main urban services that could drastically benefit from the inclusion of novel technology is waste management due to its economical and environmental impact (Zanella *et al.* , n.d.). For instance, in areas that are experiencing fast growth, waste management has become a challenge since the basic resources are not adapted to such changes (Off, 2012).

The aim of this work is to explore the synergy of swarm robotics systems and urban environments by using MPF and stigmergy to improve the efficiency and autonomy of the urban waste management system. Specifically, we employ the concept of smart city. Indeed, in order to respond to the growing demands of more efficient, sustainable, and increased quality of life, cities are becoming more and more "smart". In this context, "to be smart" can be defined as the capability to gain insights about the current urban conditions, and to react dynamically to manage them properly. According to this view, smart cities can be seen as cybernetic urban environments where different agents (e.g., citizens) and actuators (e.g., swarm of robots) exploit the city wide infrastructure as a medium to operate synergistically.

Proposed Approach

The approach proposed in this work is presented as a multilayer simulation model where each layer represents one of these components: (a) The urban environment, (b) the waste management infrastructure, and (c) the actuation layer (see Fig. 4.36).

The urban environment (Fig. 4.36 (a)) contains (i) *buildings*, where agents stay in specific hours of the day, (ii) *roads*, used by agents to move between buildings, and *citizens* (iii). Citizens are special agents that move between *buildings* (e.g., home, workplace, amenities, etc.) at certain hours during the day using *roads*. In our approach, citizens recreate the daily activity of the urban area; their simulated behavior and mobility patterns were described in recent literature (Grignard *et al.* , 2018; Alonso *et al.* , 2018).

Due to the citizens' activity (e.g., shopping, eating out, etc.), waste is generated and deposited in urban Trash Bins (TB). Waste generation is a multi-step process (Fig. 4.37). Firstly, citizens are positioned on an initial location (1). When it is time to travel (2) (e.g., go to work, return home, etc.), the citizen chooses a destination and starts the trip. While travelling, if the citizen is bringing waste, is within a distance φ from a TB, and the TB is not full, the citizen drops λ liters of waste in the TB (3). After depositing waste in the TB, the citizen continues traveling (4). In case citizens find a TB that is already full, they do not drop any waste and continue traveling.

On top of the urban environment layer we have the waste management infrastructure (Fig. 4.36 (b)), which employs (i) Trash Bins (TBs). TBs are geolocated and

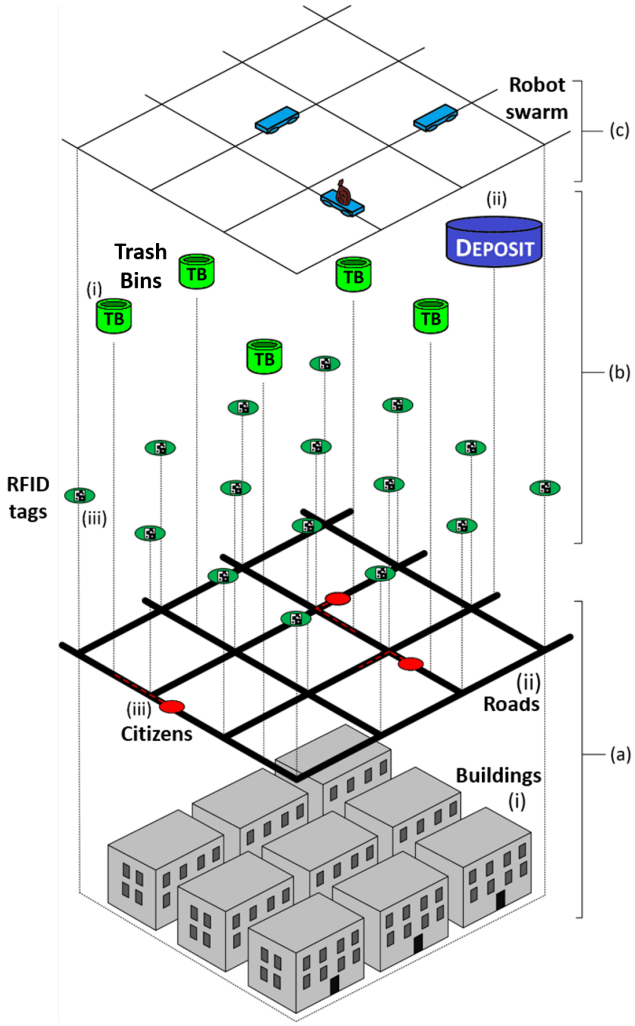


Figure 4.36: Multilayer model of the proposed scenario. In the image a "smart" city is divided into different "layers". (a) represents the urban environment layer, where buildings, roads, and citizens coexist. (b) represents the waste management infrastructure layer, where road RFID tags coexist with Trash Bins (TB) and Deposits (Ds). Finally, (c) represents the actuation layer, where swarm robots operate by exploiting the previous layers to carry out the waste management in the urban area.

arranged beside the roads. Each TB has an RFID tag containing a unique ID and the current amount of waste inside it (Ghadage & Doshi, 2017). The TB detects its amount of waste and updates the RFID tag accordingly. Once a minimum threshold is exceeded, the TB automatically packs the waste into a transportable unit (Chomik *et al.* , 2017). The number of packed waste units that can be kept in each TB is limited. Once this limit is exceeded, the TB is no longer usable. (ii) Deposits (Ds) are facilities that provide final trash disposal services (e.g., waste compactors, recycle processes, etc.) as well as robot battery refills. (iii) RFID tags at every crossroad store the in-

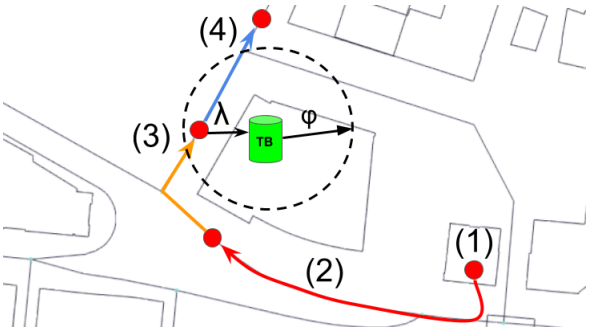


Figure 4.37: Representation of a citizen's travel path and waste depositing process. Citizens start their activity in an initial location (1). By travelling around (2) citizens generate waste. If citizens find a Trash Bin (TB) within distance φ (3), they drop λ amount of waste inside. Once this process is completed, or if the TB is full, they continue their journey (4).

formation needed to steer the swarm of robots. Specifically, each RFID tag contains the time-stamp of the last RFID operation, the amount of pheromone characterizing each road on that crossroad, and the distance and the direction toward the closest D. The path and distances between each road crossing and the closest D are fixed and known. This information allows robots to compute the shortest path between each crossing and the closest D and to store this information in its correspondent RFID tag.

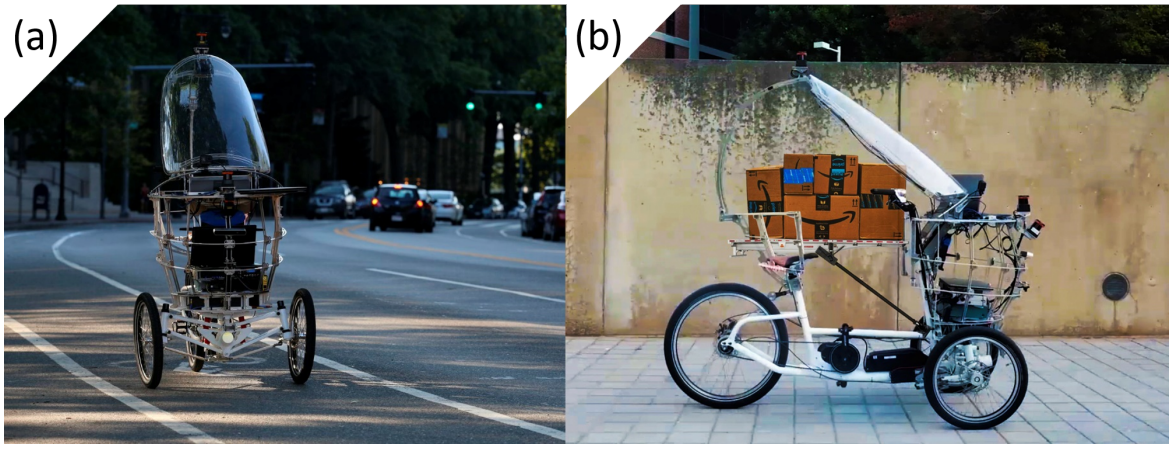


Figure 4.38: The Persuasive Electrical Vehicle (PEV). (a) PEV on the streets of the Kendall area (Cambridge, Massachusetts). (b) PEV carrying several packages as a payload.

The actuation layer (Fig. 4.36 (c)) is composed of a swarm of robots in charge of carrying the waste from each TB to the closest D. In order to increase the feasibility of our approach, we decided to model our robots using a real-world platform with

specifications suited for the task. The Persuasive Electric Vehicle (PEV), depicted in Fig. 4.38, is an autonomous tricycle developed at the MIT Media Lab aimed to be a hybrid between autonomous cars and bike sharing systems. The core idea behind the PEV is to provide an affordable, highly-customizable, self-driving solution to urban mobility. One of the main advantages of this platform over more traditional approaches is that it can operate on bike lanes; therefore, it would not stress the already saturated road infrastructure of a populated urban area. The main specifications for the PEV are: a maximum payload of 120.0 kg, maximum speed of 40.0 km/h, and 2 hours battery autonomy. In addition, the PEV is equipped with a wide variety of sensors such as R/W RFIDs, stereo cameras, LIDARs, IMUs, etc.

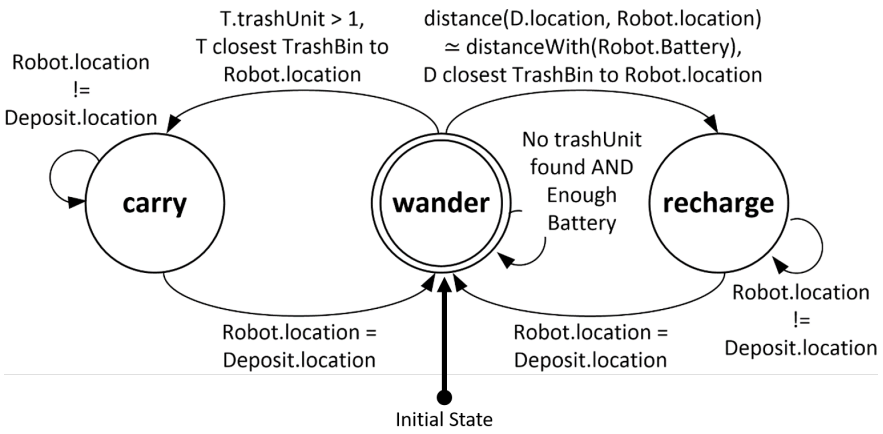


Figure 4.39: Robot behavior. Finite state machine diagram with 3 main actions: **Wander** allows robots to explore the environment. Once they run out of battery, robots move to the closest D to get a battery **Recharge**. Finally, if robots find waste in a TB along their path, they **Carry** it to the closest D.

Fig. 4.39 depicts the robot behavior via a finite state machine diagram. The robot behavior is designed by means of a set of 3 states and transition rules aimed at navigating the urban environment:

- The initial state of every robot is **Wander**. While in the wandering state, the robot travels from one road crossing to another by choosing the road with the strongest pheromone level. If this is not possible (i.e., if there is no pheromone on any road or all the roads have the same amount of pheromone), the next edge is chosen randomly. In order to avoid getting stuck on the same area until all the pheromones evaporate, robots can choose a random road with a probability of (1-X_r), where X_r represents the exploitation rate of the robots.

- At each road crossing, the robot estimates the distance that can be traveled with the remaining battery. If this distance is approximately the same to the

closest D, the robot state switches to **Recharge**. In this state, the robot moves towards the closest D by following the directions on the RFID tags on the way. The robot returns to the wander state when it reaches the closest D, since we assume that the deposit automatically swaps the battery of the robot.

- When the robot is close to a TB, it reads the RFID tag on the TB. If the TB contains at least one transportable waste unit, the robot changes its state to **Carry**. Then, the robot withdraws a waste unit from the TB and moves to the closest D by following the directions available on the RFID tags on the way. While the robot is in the carry state it ignores the TBs in its way. The robot returns to the wander state after reaching the D.

The fundamental mechanism on which the self-organization of the swarm of robots is based is the perception and distribution of pheromone amounts. In this work, an RFID tag is placed at each road crossing. In addition to the direction to the closest D and the time-stamp of the last operation, the RFID tag contains the pheromone amounts corresponding to each direction (thus, each road) that can be taken from the road crossing. The pheromones amounts are maintained in a consistent state by the robots, which manipulate them according to a precise set of rules that echo biological models of stigmergy-based foraging (**?**). In particular, the pheromone amount is subject to three processes:

- **Marking**, i.e., the addition and aggregation of pheromone to the already existing pheromone trail due to the performance of a given action (e.g., when an ant is carrying food). In our model, this is achieved by robots in the carry state by increasing the amount of pheromone in the crossroads from which the robot is coming, thus marking the path towards where the waste is being generated. This amount is proportional to the amount of waste found in the TB.

- **Evaporation**, i.e., the decay of the pheromone trail over time. In our model, this is achieved by each robot by decreasing the amount of pheromone corresponding to the current crossroad visited. The amount of pheromone decreasing through the evaporation mechanism is proportional to the difference between the current time instant (t) and the time-stamp (ts) of the last RFID operation. If the final amount of pheromone is less than zero, it is set to zero.

- **Diffusion**. To increase the probability that robots are attracted to a location with uncollected waste, we implemented the diffusion mechanism: the capability of marking a road with a small portion of the pheromone perceived on the last RFID tag, so as to make the pheromone perceptible even from roads immediately close to the marked path and steer the robots toward it. In our

model, this is achieved by each robot by increasing the amount of pheromone corresponding to the road from which the robot is coming.

In brief, when the robot interacts with an RFID tag, it decreases the amounts of pheromones on it depending on the time elapsed since the last RFID operation and the evaporation rate. Moreover, the amounts of pheromones regarding the direction (i.e., the road) from which the robot comes from is increased due to the diffusion and marking processes (if the robot is carrying waste). Specifically, the following formula describes the updating procedure of the amount of pheromone corresponding to each edge in the RFID tag:

$$P_t = P_{ts} - [E_r \cdot P_a \cdot (t - ts)] + (P_a \cdot T_a) + (D_r \cdot P_{max}) \tag{4.19}$$

In Eq. 4.19, P_t represents the amount of pheromone corresponding to the current edge at the current time instant. P_{ts} is the amount of pheromone corresponding to the current edge at the time-stamp (i.e., the last operation on the RFID tag). E_r is the evaporation rate ($0 \leq E_r \leq 1$), i.e., the amount of the pheromones disappearing per unit of time. P_a is the amount of pheromone to be added to the RFID tag for each unit of waste found in the TB from which the waste has been picked up (only if the robot is performing the carry action). T_a is the amount of waste found in the TB (only if the robot is performing the carry action and it comes from the current edge). D_r is the diffusion rate ($0 \leq D_r \leq 1$), in other words, the portion of the pheromone to diffuse. Finally, P_{max} is the maximum pheromone amount on the last RFID tag.

The presented system[1] was developed in GAMA (Grignard *et al.* , 2018); a realistic agent-based simulation tool applied in fields such as urban planning, disaster mitigation, etc. The urban environment layer was built using real-world GIS data by integrating the map of the Kendall (Cambridge, MA) urban area. The number of citizens was initialized to 10,000 following previous research works about the area of study (Alonso *et al.* , 2018). Regarding the waste generation process, according to the EPA (Environmental Protection Agency) Americans produce 2 kg of waste per day (Agency, 2014). However, not all of that waste goes into public TBs; a large portion of it is dropped in residential bins as well. According to (Patrick *et al.* , 2013), Cambridge public works collect an average of 1.18 kg of waste per citizen per day from Cambridge households. Thus, we estimated that 0.82 kg of waste per citizen was deposited in public TBs everyday. By using conversion data about the weight of different types of waste (Agency, 2016), we transformed the amount of kg of waste generated per citizen into liters. The result of these conversions was 8.42 liters/citizen. We initialized λ to this value throughout our simulations. Finally, φ was initialized to 50 meters.

[1] A copy of code repository can be found here: https://goo.gl/tqRvS4

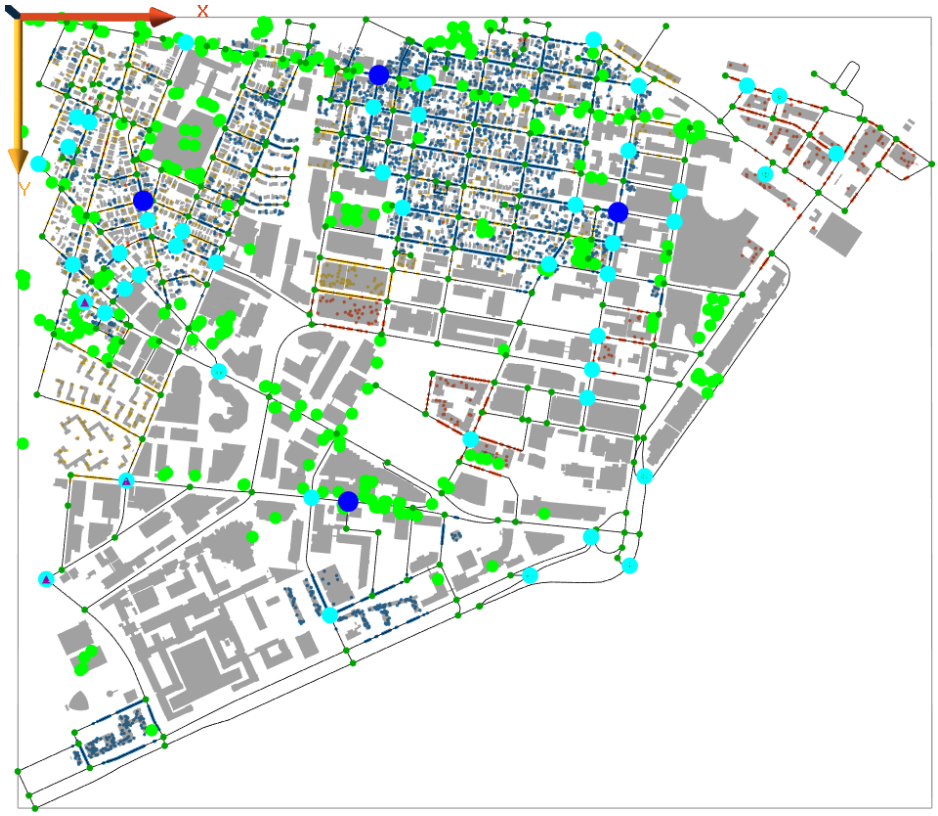


Figure 4.40: Simulation of the urban swarm in the Kendall area (Cambridge, MA). Citizens (small coloured dots) can move between buildings by using roads and during each movement they can drop waste in TBs (green dots) close by. The robots (cyan dots) can move over the roads to carry the waste from the TBs into the Ds (blue dots) using the information written on the RFID tags at each crossroad.

Fig. 4.40 shows a screenshot of the simulation tool where each building (gray), TB (green dots), road (black lines), and citizen (small dots over buildings and roads) is depicted. Moreover, we obtained the number and location of TBs from the Cambridge Geographic Information Service[2]. Ds (dark blue dots) were placed by using the k-means algorithm to minimize the distance between them and the TBs.

Finally, each robot is represented as a cyan dot. The effectiveness of the proposed approach was tested with different configurations (i.e., by changing the behavioral and scenario parameters). The ranges of the parameters were chosen to allow the implementation of significantly different behaviors on the swarm of robots. Specifically:

- **Number of Robots** (R_n): affects the effectiveness and the size of the overall system; the tested values for this parameter were 20, 35, and 50 robots.

[2]More information can be found here: https://goo.gl/os3nxN

- **Evaporation Rate** (E_r): affects the amount of time the system retains the information about the waste disposal demand; the tested values were 0.05%, 0.15%, and 0.3%.

- **Exploitation Rate** (X_r): affects the probability that the robots follow the path with the strongest pheromone rather than a random one. A higher value results in a higher exploitation of the information about the waste disposal process, whereas lower values increase the exploration of the overall scenario. A more exploratory swarm easily reaches isolated TBs, while a swarm more prone to the exploitation of the waste disposal information exhibits a more aggressive waste collection behavior toward the previously-discovered non-empty TBs. The exploitation rate affects also the diffusion rate, which is 1-X_r, since in a non-exploratory swarm the diffusion will just reinforce the already marked path. The tested values were 0.6, 0.75, and 0.9.

- **Carriable Waste** (C_w): affects the amount of carriable waste per robot. Lower values of this parameter result in a more responsive but slow reaction of the system since the waste can be picked only if the C_w is already present in the bin. Indeed, we assume that the C_w corresponds to the amount of waste that a TB can pack to be transported. The tested values were 6, 12, and 18 liters of waste. This was designed taking into account the PEV capabilities.

- **Number of Deposit** (D_n): affects the responsiveness of the overall system; the tested values were 2, 3, and 5 Ds.

In order to provide a better insight about the implications of the proposed approach, we compare it with the waste management model that is currently operating in the urban area of study (i.e., truck-based). We built this model based on the information provided by the Cambridge Department of Public Works (CDPW)[3]. According to CDPW, a single truck system in 5 working days (Monday-Friday) in 5 hours a day (7-12PM) is able to empty approximately 6000 TBs. This results in a capability of emptying about 240 trash bins in an hour. In our scenario, the number of TBs is fixed to 274. Therefore, the truck should be able to empty all TBs in about an hour and 10 minutes and should pass once a day.

Obtained Results

We conducted ten simulations for each possible parameter combination previously introduced. In order to analyze the effectiveness of both approaches, we introduced two performance metrics. First, the **Amount of Uncollected Trash** (AUT, measured

[3]More information about the specific route and timetables can be found here: https://goo.gl/cHXDYS

in liters) represents the amount of waste left unattended in the environment. Higher AUT levels correlate to the appearance of urban issues such as pests, air pollution, and public health problems. Second, the average number of **Full Trash Bins** (FTB, measured in units) in the scenario during a day. FTB shows the average number of unusable TBs that the system leaves in the urban environment during the day. Higher FTB values typically correlate to higher citizens' dissatisfaction rates since they might have to travel longer distances to dispose their waste. For the sake of interpretability, each of the measures is presented as a percentage (the lower, the better) considering that in our scenario there are 274 TBs with a capacity of 125 liters each[4].

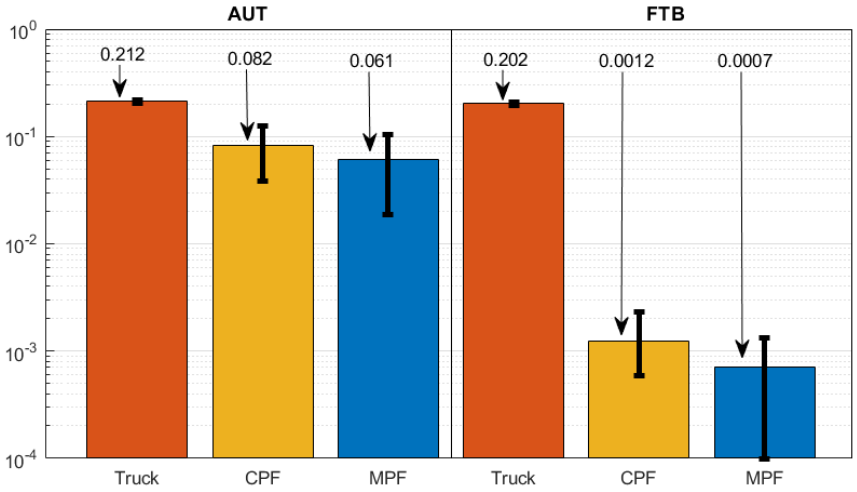


Figure 4.41: Percentage of AUT (Amount of Uncollected Trash) and FTB (Full Trash Bins) measures for the truck and the swarm (best parameterizations) with the CPF and the MPF approach. The results are presented in log scale to improve readability.

We compare the performance of our proposed approach against (i) the current trash disposal model i.e., truck-based model; and (ii) a CPF solution i.e., by using the stigmergy-based foraging with a single deposit. Thus, we simulated each model and computed the corresponding performance metrics. The results obtained with ten simulations are summarized in Fig.4.41. Results show that the current waste management system offers lower performance than the swarm-based solution proposed in this paper. In fact, in terms of the percentage of AUT and FTB, the MPF approach offers a decrease of 71% (0.061) and 99% (0.0007) compared to the results obtained with the truck-based model (0.212 and 0.202 respectively). Moreover, the MPF approach results to be more effective than the CPF approach for both AUT and FTB.

[4]This capacity correlates to the TB model (Big Belly BB5) currently operating in the study area. The BB5 is equipped with a solar-powered waste compactor and a wireless data link. More information can be found here: http://bigbelly.com/

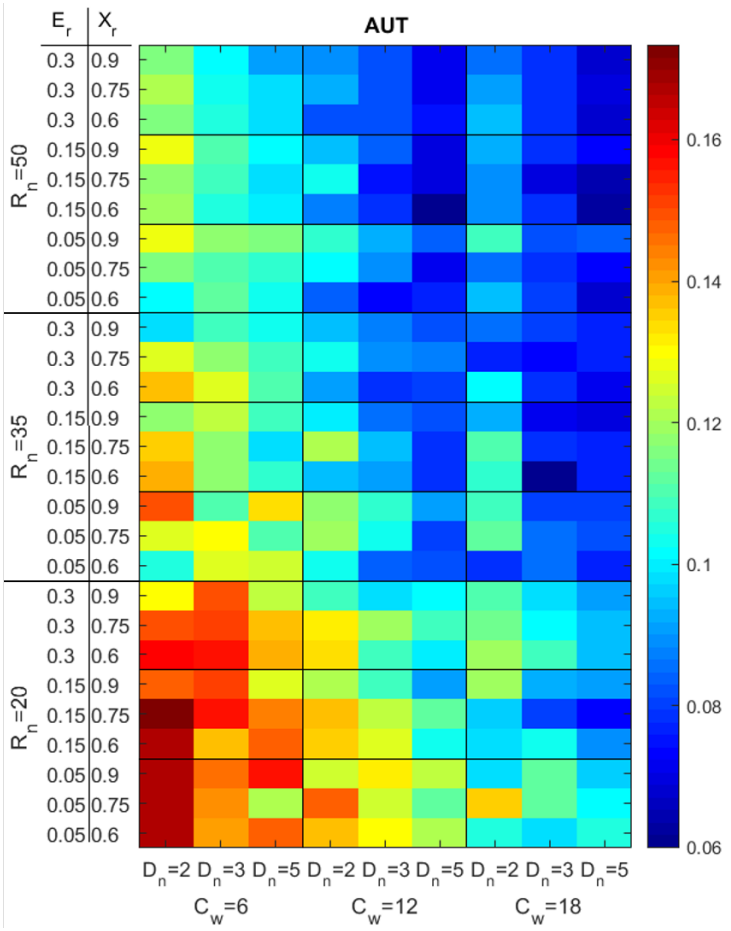


Figure 4.42: The average percentage of Amount of Uncollected Trash (AUT) in the scenario during a day by changing the parameterization.

It can be observed from the results in Fig. 4.42 that our system is able to collect most of the disposable trash on average during the day, leaving a AUT of 17% (worst case) and 6% (best case). Moreover, in general, the increase of D_n, C_w, and R_n results in a lower AUT. A greater number of deposits results in a shorter path to reach them, while more robots and a greater carriage capacity result in a system that collects and disposes waste more quickly.

The results in Fig. 4.43 prove that our system is able to empty the TBs fast enough to have less than 1 (0.0007%, best case) or 2 (0.0066%, worst case) FTB in the scenario on average during the day. Moreover, it can be noticed that by increasing E_r and R_n we obtain a lower percentage of FTB since it increases the responsiveness of the system.

Two parameter configurations provided the best performances: (1) R_n=35, E_r=0.15, X_r=0.6, C_w=18, D_n=3 which produces a 0.7% FTB and 6.1% AUT and (2) R_n=50, E_r=0.15, X_r=0.6, C_w=12, D_n=5 which obtains a 0.9% FTB and 5.9% AUT. The first so-

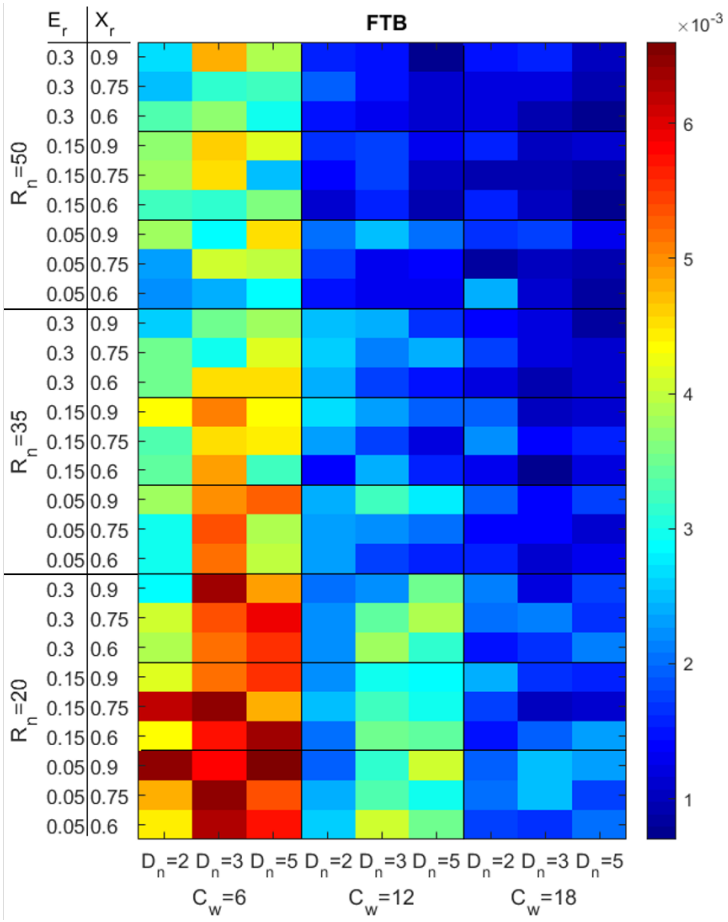


Figure 4.43: The average percentage of Full Trash Bins (FTB) in the scenario during a day by changing the parameterization.

	β_{Rn}	β_{Cw}	β_{Dn}
AUT	-0.6615	-0.7580	-0.3868
FTB	-0.4647	-0.9584	0.0407

Table 4.26: β coefficients, multiple standardized regression

lution is characterized by a medium number of deposits and robots, but can assure a good responsiveness of the system thanks to the relatively high C_w. On the other hand, the second solution, is characterized by a large D_n and R_n, and a medium C_w. These configurations suggest that there is a balance between the size of the system (D_n and R_n) and the amount of carriable waste (C_w). At the same time, both solutions are characterized by an E_r of 0.15 and a relatively high X_r with a value of 0.6. To assess the relationship between $R_n/D_n/C_w$ and the proposed performance measures, we computed a multiple standardized regression, i.e., the regression in which both dependent and independent variable are substituted by their Z-score. By considering the size of each β coefficient, we are able to compare the impact of

each variable on the corresponding performance measure despite the differences in their scale. Larger coefficients correspond to higher contributions, whereas the sign describes the direction (positive or negative) of the contribution. It is worth recalling that all performance measures proposed must be minimized to achieve a better performance. According to Table 4.26, we can see that by increasing R_n and C_w, the performance is considerably improved. Surprisingly, increasing D_n increases the AUT performance but decreases FTB's; indeed, a greater number of Ds means shorter travel distances for robots, therefore a more responsive system and a better AUT performance. However, since Ds are the destination of all robots, the paths around the D are more likely to be marked by digital pheromones which are aimed at steering the swarm. This means that around each D the robots' exploratory capability is reduced due to the overwhelming amount of deployed pheromones, therefore the probability of reaching TBs in areas that require to pass by a near D may be lower. This explains the decrease in the performance at the bottom-left corner of Fig. 4.43 when D_n is increased.

We showed that a swarm is able to handle the waste management in an effective and self-organized manner, without any external information source or prior knowledge about the trash disposal demand. Moreover, the proposed approach is not specific to waste management and can be used in a number of different applications such as package delivery, autonomous vehicle rides, etc.

chapterConclusion This section draws the conclusion of this Ph.D. thesis. First, we discuss the proposed approach and the obtained results. Then, a final section is dedicated to future directions of the research.

4.9 Discussion

In this thesis, we presented eight applications of computational stigmergy with very promising results. This confirms computational stigmergy as a versatile and transversal computation approach, useful for unfolding spatiotemporal patterns from the density of the analyzed data.

In contrast with other machine learning techniques (e.g., neural network), the SRF approach (i) exhibits robustness with respect to minor fluctuations occurring in data by adopting stigmergic information granulation, (ii) does not require time dynamics modeling, because it intrinsically embodies the time domain, and (iii) does not require the in-depth modeling of the dynamics under investigation since it relies on DE and requires a small training set to adapts the similarity computation to a specific problem.

Specifically, in terms of classification accuracy, our approach performs better than supervised classifiers even if using a smaller dataset and with different cases study (Chapters 4.1 and 4.2). Furthermore, the proposed solution overcomes the

specificity of a well-known set of classifiers (i.e., different classifiers performing better on different users) offering a general framework for the sleep quality assessment (Chapter 4.2). Indeed, on average, it outperforms all the comparing classifiers: 4% more than DTW and 11% more than the best performing classifier (DecisionTable).

The goodness of these results is confirmed also in the application presented in Chapter 4.3, where the daily physical activities of 3 subjects are assessed according to how the activity is performed, and despite activity type.

In the application detailed in Chapter 4.4. and 4.5 by using stigmergy, the spatiotemporal density in data has been exploited to identify city hotspots and characterize their dynamics, allowing to generate data-driven prototypes of typical daily activity. Moreover, by treating them via a clustering technique, we were able to discern expected patterns from unexpected ones, which were found to be usually related to various events (Chapters 4.5).

In Chapter 4.6 bidimensional stigmergy has been used to analyze the mobility of a group of people (i.e. refugees and locals) with the aim to understand the relationship between social integration and the mobility similarity between refugees and locals and assess the effect of social friction.

In Chapter 4.7 stigmergy has been used together with flocking strategy and DE to coordinate a swarm of UAVs employed in a distributed target search. Experimental results show that (i) the quality of the swarm cooperation is better in terms of swarm formation, search efficiency, robustness to fault, strategy and scalability; (ii) the differential evolution optimization provides significant and structural improvements to the coordination logic; (iii) taking into account the technological characteristics of the drone in its behavioral model sensibly improve the performance and the realism of the biologically metaheuristics.

In Chapter 4.8 we showed that thanks to an approach based on stigmergy and foraging heuristics a swarm of robots is able to handle the waste management in an effective and self-organized manner, without any external information source or prior knowledge about the trash disposal demand. Indeed, with the proposed approach, both the average amount of trash and the average number of full trash bins during the day are considerably reduced compared to the current solution.

Finally, it is important to note that our implementation also has reasonable execution times. Indeed, for data analysis applications that use the stigmergic perceptron, the training times have always been between 3 and 5 hours, with a variability due to the complexity (e.g. length, number of archetypes) of the time series analyzed. On the other hand, the algorithm of coordination of the swarms of drones has a variable training time depending on the scenario on which it is executed (for example, the "field" scenario has always had the shortest times as the simplest) but still between 4 and 7 hours. The times shown refer to the execution on a machine with the following characteristics: CPU Intel Core i7-6700HQ at 2.60-3.50 GHz, 6M Cache, 16 GB

DDR3L 1600MHz RAM, Windows 10 OS.

4.10 Future works

A direction for possible future developments is represented by the possibility to scale the already presented architecture in order to analyze multidimensional data. An initial effort in this direction has already been made but the part of the aggregation of trails and inference requires further research. As an example, the already presented two-dimensional computational stigmergy together with mobility data with an individual ID observed over a long period (which we have not had so far) to recognize individuals (e.g. taxi drivers) or groups of similar individuals.

Moreover, the aggregation of the SP presented in this work exploits the concepts taken from the connectionist paradigm. However, the aggregation could be performed with a logic ruled based approach in order to exploit the spatiotemporal aggregation provided by our approach in an inference system, i.e. implementing a hybrid obtained including a fuzzy logic approach.

Finally, the great properties provided by stigmergy should be systematically investigated to study: (I) the robustness against noise in data observations, (II) the computational complexity of the approach (III) the difference in terms of properties with other swarm approaches or reinforcement learning..

Appendix A

Publications

International Journals Papers

1. A. L. Alfeo, M.G.C.A. Cimino, A. Lazzeri and G. Vaglini, "Detecting urban road congestion via parametric adaptation of position-based stigmergy", published on Intelligent Decision Technologies Journal (2017). **Candidate's contributions**: theoretical analyses and manuscript production.

2. A. L. Alfeo, M.G.C.A. Cimino, A. Lazzeri, M. Lega and G. Vaglini, "Swarm coordination of small UAVs for target search with imperfect sensors", published on Intelligent Decision Technologies Journal (2017). **Candidate's contributions**: model design and testing, experimental analyses and manuscript production.

3. A. L. Alfeo, P. Barsocchi, M.G.C.A. Cimino, D.La Rosa, F. Palumbo, and G. Vaglini, "Sleep behavior assessment via smartwatch and stigmergic receptive fields", published on Personal and Ubiquitous Computing, (2017). **Candidate's contributions**: stigmergy-based approach testing and experimental analyses, manuscript production.

4. A. L. Alfeo, M. G. C. A. Cimino, S. Egidi, B. Lepri, and G. Vaglini, "A stigmergy-based analysis of city hotspots to discover trends and anomalies in urban transportation usage", published on IEEE Transaction on Intelligent Transportation Systems (2018). **Candidate's contributions**: theoretical analysis, approach modeling, testing, experimental analyses and manuscript production.

5. A. L. Alfeo, M.G.C.A. Cimino, M. Lega and G. Vaglini, "Design and control of the emergent behavior of small drones swarming for distributed target localization", accepted by Journal of Computational Science (2018). **Candidate's contributions**: theoretical analysis, approach modeling, testing, experimental analyses and manuscript production.

6. A. L. Alfeo, M. G. C. A. Cimino, and G. Vaglini, “Enhancing biologically inspired swarm behavior: metaheuristics to foster the optimization of UAVs coordination in target search”, **submitted to** international journal. **Candidate’s contributions**: approach modeling, testing, experimental analyses and manuscript production.

Peer Reviewed International Conferences Papers

1. A. L. Alfeo, F. Appio, M.G.C.A. Cimino, A. Lazzeri, A. Martini, and G. Vaglini. “An adaptive stigmergy-based system for evaluating technological indicator dynamics in the context of smart specialization”, in proceedings of INSTICC The 5th International Conference on Pattern Recognition Applications and Methods”, (ICPRAM 2016), pp. 497-502, Rome, Italy, Springer. **Candidate’s contributions**: theoretical analysis, approach modeling, testing, experimental analyses and manuscript production.

2. A. L. Alfeo, M.G.C.A. Cimino, and G. Vaglini, "Measuring Physical Activity of Older Adults via Smartwatch and Stigmergic Receptive Fields", in proceedings of INSTICC The 6th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2017), pp. 724-730, Porto, Portugal, 24-26 February 2017. **Candidate’s contributions**: experimental analyses and manuscript production.

3. A. L. Alfeo, M. G. C. A. Cimino, S. Egidi, B. Lepri, and G. Vaglini, “An emergent strategy for characterizing urban hotspot dynamics via GPS data”, in proceedings of “The 5th Conference on scientific analysis of mobile phone datasets”, organizzato da F. Calabrese, E. Moro, V. Blondlel e A. Pentland. (NetMob 2017) 5-7 April 2017, Milan, Italy. **Candidate’s contributions**: approach modeling, testing, experimental analyses and manuscript production.

4. A. L. Alfeo, M. G. C. A. Cimino, S. Egidi, B. Lepri, and G. Vaglini, “Stigmergy-based modeling to discover urban activity patterns from positioning data”, in proceedings of “Social, Cultural, and Behavioral Modeling: 10th International Conference”, (SBP-BRiMS 2017), vol. 10354, p.p. 292-302. Springer. Washington, DC, USA, July 5-8, 2017. **Candidate’s contributions**: approach modeling, testing, experimental analyses and manuscript production.

5. Antonio Luca Alfeo, Eduardo Castello Ferrer, Yago Lizarribar Carrillo, Arnaud Grignard, Luis Alonso Pastor, Mario G. C. A. Cimino, Bruno Lepri, Gigliola Vaglini, Kent Larson, Marco Dorigo, Alex 'Sandy' Pentland. "Urban Swarms: A new approach for autonomous waste management", **submitted to** international conference. **Candidate's contributions**: approach modeling, testing, experimental analyses and manuscript production.

International Workshops Paper

1. A. L. Alfeo, M. G. C. A. Cimino, B. Lepri, G. Vaglini. "Mobility and Calling Behavior to Unfold the Integration of Syrian Refugees in Turkey", **submitted to** international workshop.

Bibliography

Zola, interactive map for zone and land use of NYC. http://maps.nyc.gov/doitt/nycitymap/template?applicationName=ZOLA. [Online; accessed 18-April-2017].

2012. Putting the informal on the map: tools for participatory waste management. *Proceedings of the 12th Participatory Design ...*, 13–16.

2015. *Papal visit to bring blizzard of traffic to D.C., New York and Philadelphia. September 22, 2015.* UsaToday. [Online; accessed 14-March-2017].

2015. *Weather NYC: Thousands of transatlantic travellers face serious disruption caused by New York winter storm 'Juno'. January 26, 2015*. The Independent. [Online; accessed 13-March-2017].

2017. *Matlab*. www.mathworks.com/products/matlab/. [Online; accessed April 23, 2017].

2017. *Netlogo*. ccl.northwestern.edu/netlogo/. [Online; accessed September 18, 2017].

2017. *See-demining*. http://www.seedemining.org/main.htm. [Online; accessed May 19, 2017].

2017. *TrashOut*. https://www.trashout.ngo/trashmap/?ref=trashout.me. [Online; accessed May 27, 2017].

2018a. *130 migrants intercepted in Aegean heading for Chios*. http://www.ansamed.info/ansamed/en/news/nations/greece/2017/03/06/130-migrants-intercepted-in-aegean-heading-for-chios_b60eec09-29a6-42a6-b846-3e747f71f000.html. [Online; accessed August 13, 2018].

2018b. *Istanbul police evacuate refugees after clashes with locals*. https://www.trtworld.com/turkey/istanbul-police-evacuate-300-refugees-after-clashes-with-locals-358027. [Online; accessed August 13, 2018].

2018c. *Kurdish militants claim responsibility for Turkey tunnel attack*. https://www.reuters.com/article/us-turkey-blast/kurdish-militants-claim-

responsibility-for-turkey-tunnel-attack-idUSKBN17E0MW. [Online; accessed August 13, 2018].

2018d. *One killed in brawl between locals and Afghan, Syrian migrants in Istanbul*. http://www.hurriyetdailynews.com/one-killed-in-brawl-between-locals-and-afghan-syrian-migrants-in-istanbul-113111. [Online; accessed August 13, 2018].

Abbate, Stefano, Avvenuti, Marco, & Light, Janet. 2012. MIMS: A minimally invasive monitoring sensor platform. *IEEE Sensors Journal*, 12(3), 677–684.

Agency, United States Environmental Protection. 2014 (02). *Municipal Solid Waste Generation, Recycling, and Disposal in the United States Facts and Figures for 2012*. Tech. rept. US EPA.

Agency, United States Environmental Protection. 2016 (04). *Volume-to-Weight Conversion Factors*. Tech. rept. US EPA.

Åkerstedt, Torbjörn, Hume, Ken, Minors, David, & Waterhouse, Jim. 1994. The meaning of good sleep: a longitudinal study of polysomnography and subjective sleep quality. *Journal of Sleep Research*, 3(3), 152–158.

Alexander, Lauren, Jiang, Shan, Murga, Mikel, & González, Marta C. 2015. Origin–destination trips by purpose and time of day inferred from mobile phone data. *Transportation research part c: emerging technologies*, 58, 240–250.

Alfeo, Antonio L, Cimino, Mario GCA, De Francesco, Nicoletta, Lazzeri, Alessandro, Lega, Massimiliano, & Vaglini, Gigliola. Swarm coordination of mini-UAVs for target search using imperfect sensors. *Intelligent Decision Technologies*, 1–14.

Alfeo, Antonio Luca, Cimino, Mario Giovanni C. A., Egidi, Sara, Lepri, Bruno, & Vaglini, Gigliola. 2018. A Stigmergy-Based Analysis of City Hotspots to Discover Trends and Anomalies in Urban Transportation Usage. *IEEE Transactions on Intelligent Transportation Systems*, 19, 2258–2267.

Almaatouq, Abdullah, Prieto-Castrillo, Francisco, & Pentland, Alex. 2016. Mobile communication signatures of unemployment. *Pages 407–418 of: International conference on social informatics*. Springer.

Alonso, Luis, Zhang, Yan Ryan, Grignard, Arnaud, Noyman, Ariel, Sakai, Yasushi, ElKatsha, Markus, Doorley, Ronan, & Larson, Kent. 2018. CityScope: A Data-Driven Interactive Simulation Tool for Urban Design. Use Case Volpe. *Pages 253–261 of:* Morales, Alfredo J., Gershenson, Carlos, Braha, Dan, Minai, Ali A., & Bar-Yam, Yaneer (eds), *Unifying Themes in Complex Systems IX*. Cham: Springer International Publishing.

Altindag, Onur, & Kaushal, Neeraj. 2017. Do refugees impact voting behavior in the host country? Evidence from Syrian Refugee inflows in Turkey.

Amrein, Hubert. 2004. Pheromone perception and behavior in Drosophila. *Current opinion in neurobiology*, 14(4), 435–442.

An, S., Hu, X., & Wang, J. 2011. Urban taxis and air pollution: a case study in Harbin, China. *Journal of Transport Geography*, 19(4).

Ansari, Nirwan, Hou, Edwin SH, Zhu, Bin-Ou, & Chen, Jiang-Guo. 1996. Adaptive fusion by reinforcement learning for distributed detection systems. *IEEE Transactions on Aerospace and Electronic Systems*, 32(2), 524–531.

Bakker, Linda, Cheung, Sin Yi, & Phillimore, Jenny. 2016. The asylum-integration paradox: Comparing asylum support systems and refugee integration in the Netherlands and the UK. *International Migration*, 54(4), 118–132.

Balkan, Binnur, & Tumen, Semih. 2016. Immigration and prices: quasi-experimental evidence from Syrian refugees in Turkey. *Journal of Population Economics*, 29(3), 657–686.

Bedi, Punam, Mediratta, Neha, Dhand, Silky, Sharma, Ravish, & Singhal, Archana. 2007. Avoiding traffic jam using ant colony optimization-a novel approach. *Pages 61–67 of: Conference on Computational Intelligence and Multimedia Applications, 2007. International Conference on*, vol. 1. IEEE.

Bellini, Nicola. 2016. Smart Specialisation in Europe : Looking Beyond Regional Borders.

Berman, Spring, Halasz, Adam, Hsieh, M Ani, & Kumar, Vijay. 2008. Navigation-Based Optimization of Stochastic Deployment Strategies for a Robot Swarm to Multiple Sites. *In: 2008 Conference on Decision and Control (CDC'08*.

Bernardeschi, Cinzia, Cimino, Mario GCA, Domenici, Andrea, & Vaglini, Gigliola. 2016. Using smartwatch sensors to support the acquisition of sleep quality data for supervised machine learning. *Pages 1–8 of: The 6th EAI International Conference on Wireless Mobile Communication and Healthcare (MOBIHEALTH 2016)*. EAI.

Berndt, Donald J, & Clifford, James. 1994. Using dynamic time warping to find patterns in time series. *Pages 359–370 of: KDD workshop*, vol. 10. Seattle, WA.

Bertuccelli, Luca F, & How, JP. 2005. Robust UAV search for environments with imprecise probability maps. *Pages 5680–5685 of: Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*. IEEE.

Bethke, Brett, Valenti, Mario, & How, Jonathan. 2007. Cooperative vision based estimation and tracking using multiple UAVs. *Pages 179–189 of: Advances in cooperative control and optimization*. Springer.

Bicego, Manuele, Murino, Vittorio, & Figueiredo, Mário AT. 2003. Similarity-based clustering of sequences using hidden Markov models. *Pages 86–95 of: International Workshop on Machine Learning and Data Mining in Pattern Recognition*. Springer.

Blumenstock, Joshua, & Fratamico, Lauren. 2013. Social and spatial ethnic segregation: a framework for analyzing segregation with large-scale spatial network data. *Page 11 of: Proceedings of the 4th Annual Symposium on Computing for Development*. ACM.

Boletsis, Costas, McCallum, Simon, & Landmark, Brynjar Fowels. 2015. The Use of Smartwatches for Health Monitoring in Home-Based Dementia Care. *In: HCI*.

Bonomi, Alberto G, Plasqui, Guy, Goris, Annelies HC, & Westerterp, Klass R. 2010. Estimation of free-living energy expenditure using a novel activity monitor designed to minimize obtrusiveness. *Obesity*, 18(9), 1845–1851.

Bootzin, Richard R, & Engle-Friedman, Mindy. 1981. The assessment of insomnia. *Behavioral Assessment*, 3(2), 107–126.

Bottou, Léon. 2010. Large-scale machine learning with stochastic gradient descent. *Pages 177–186 of: Proceedings of COMPSTAT'2010*. Springer.

Brambilla, Manuele, Ferrante, Eliseo, Birattari, Mauro, & Dorigo, Marco. 2013a. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7(1), 1–41.

Brambilla, Manuele, Ferrante, Eliseo, Birattari, Mauro, & Dorigo, Marco. 2013b. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7(1), 1–41.

Brueckner, Sven A, & Parunak, H Van Dyke. 2002. Swarming agents for distributed pattern detection and classification. *Ann Arbor*, 1001, 48113–4001.

Carpi, Estella, & Pınar Şenoğuz, H. 2018. Refugee Hospitality in Lebanon and Turkey. On Making 'The Other'. *International Migration*.

Caselli, Federico, Bonfietti, Alessio, & Milano, Michela. 2015. Swarm-based controller for traffic lights management. *Pages 17–30 of: Congress of the Italian Association for Artificial Intelligence*. Springer.

Castello, Eduardo, Yamamoto, Tomoyuki, Libera, Fabio Dalla, Liu, Wenguo, Winfield, Alan F T, Nakamura, Yutaka, & Ishiguro, Hiroshi. 2016. Adaptive foraging for simulated and real robotic swarms: the dynamical response threshold approach. *Swarm Intelligence*, 10(1), 1–31.

Castro, Pablo Samuel, Zhang, Daqing, Chen, Chao, Li, Shijian, & Pan, Gang. 2013. From taxi GPS traces to social and community dynamics: A survey. *ACM Computing Surveys (CSUR)*, 46(2), 17.

Chan, Kin-Pong, & Fu, Wai-Chee. 1999. Efficient time series matching by wavelets. *Page 126 of: icde*. IEEE.

Chen, Lei, & Ng, Raymond. 2004. On the marriage of lp-norms and edit distance. *Pages 792–803 of: Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*. VLDB Endowment.

Chen, Lei, Özsu, M Tamer, & Oria, Vincent. 2005. Robust and fast similarity search for moving object trajectories. *Pages 491–502 of: Proceedings of the 2005 ACM SIGMOD international conference on Management of data*. ACM.

Chen, Yikai, Gao, Lingling, Li, Zhi-peng, & Liu, Yun-cai. 2007. A new method for urban traffic state estimation based on vehicle tracking algorithm. *Pages 1097–1101 of: Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*. IEEE.

Chen, Zhenyu, Lin, Mu, Chen, Fanglin, Lane, Nicholas D, Cardone, Giuseppe, Wang, Rui, Li, Tianxing, Chen, Yiqiang, Choudhury, Tanzeem, & Campbell, Andrew T. 2013. Unobtrusive sleep monitoring using smartphones. *Pages 145–152 of: 2013 7th International Conference on Pervasive Computing Technologies for Healthcare and Workshops*. IEEE.

Chi, TZ, Cheng, Hayong, Page, JR, & Ahmed, NA. 2014. Evolving swarm of UAVs. *Advances in aircraft and spacecraft science*, 1(2), 219–232.

Chiaraviglio, N, Artés, Tomàs, Bocca, R, López, J, Gentile, A, Ayanz, J San Miguel, Cortés, Ana, & Margalef, Tomàs. 2016. Automatic fire perimeter determination using MODIS hotspots information. *Pages 414–423 of: e-Science (e-Science), 2016 IEEE 12th International Conference on*. IEEE.

Chomik, Richard S, Yoho, Mark, Leslie, Stuart, Kim, Taek, Petrie, Aidan, & Sejnowski, Joe. 2017 (Aug. 1). *Waste storage device*. US Patent 9,718,614.

Cimino, Mario G. C. A., Lazzeri, Alessandro, & Vaglini, Gigliola. 2015a. Enabling swarm aggregation of position data via adaptive stigmergy: A case study in urban traffic flows. *2015 6th International Conference on Information, Intelligence, Systems and Applications (IISA)*, 1–6.

Cimino, Mario G. C. A., Lazzeri, Alessandro, & Vaglini, Gigliola. 2015b. Improving the Analysis of Context-Aware Information via Marker-Based Stigmergy and Differential Evolution. *In: ICAISC*.

Cimino, Mario GCA, Lazzerini, Beatrice, & Marcelloni, Francesco. 2006. A novel approach to fuzzy clustering based on a dissimilarity relation extracted from data using a TS system. *Pattern recognition*, 39(11), 2077–2091.

Cimino, Mario GCA, Lazzeri, Alessandro, & Vaglini, Gigliola. 2015c. Combining stigmergic and flocking behaviors to coordinate swarms of drones performing target search. *Pages 1–6 of: Information, Intelligence, Systems and Applications (IISA), 2015 6th International Conference on*. IEEE.

Cimino, Mario GCA, Lazzeri, Alessandro, & Vaglini, Gigliola. 2016. Using Differential Evolution to Improve Pheromone-based Coordination of Swarms of Drones for Collaborative Target Detection. *Pages 605–610 of: ICPRAM*.

Cooper, G., & Herskovits, E. 1992. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9(4), 309–347.

Cruz, Henry, Eckert, Martina, Meneses, Juan, & Martínez, José-Fernán. 2016. Efficient forest fire detection index for application in unmanned aerial systems (UASs). *Sensors*, 16(6), 893.

Cui, Yani, Ren, Jia, Du, Wencai, & Dai, Jingguo. 2016. UAV target tracking algorithm based on task allocation consensus. *Journal of Systems Engineering and Electronics*, 27(6), 1207–1218.

D'Andrea, Raffaello. 2012. Guest editorial: A revolution in the warehouse: A retrospective on Kiva Systems and the grand challenges ahead. *IEEE Transactions on Automation Science and Engineering*, 9(4), 638–639.

Dang, AnhDuc, & Horn, Joachim. 2015. Formation control of leader-following uavs to track a moving target in a dynamic environment. *Journal of Automation and Control Engineering Vol*, 3(1), 1–8.

Das, Gautam, Gunopulos, Dimitrios, & Mannila, Heikki. 1997. Finding similar time series. *Pages 88–100 of: European Symposium on Principles of Data Mining and Knowledge Discovery*. Springer.

Das, Swagatam, & Suganthan, Ponnuthurai Nagaratnam. 2011. Differential evolution: A survey of the state-of-the-art. *IEEE transactions on evolutionary computation*, 15(1), 4–31.

Dasgupta, Prithviraj, Cheng, Ke, & Fan, Li. 2009. Flocking-based distributed terrain coverage with dynamically-formed teams of mobile mini-robots. *Pages 96–103 of: Swarm Intelligence Symposium, 2009. SIS'09. IEEE*. IEEE.

De Benedetti, M, D'Urso, F, Fortino, G, Messina, F, Pappalardo, G, & Santoro, C. 2017. A fault-tolerant self-organizing flocking approach for UAV aerial survey. *Journal of Network and Computer Applications*, 96, 14–30.

de Zambotti, Massimiliano, Baker, Fiona C, Willoughby, Adrian R, Godino, Job G, Wing, David, Patrick, Kevin, & Colrain, Ian M. 2016. Measures of sleep and cardiac functioning during sleep using a multi-sensory commercially-available wristband in adolescents. *Physiology & behavior*, 158, 143–149.

Degli Esposti, MFCAMG, Farinelli, C, & Menconi, G. 2009. Sequence distance via parsing complexity: Heartbeat signals. *Chaos, Solitons & Fractals*, 39(3), 991–999.

Del Carpio, Ximena V, & Wagner, Mathis. 2015. *The impact of Syrians refugees on the Turkish labor market*. The World Bank.

Ding, Hui, Trajcevski, Goce, Scheuermann, Peter, Wang, Xiaoyue, & Keogh, Eamonn. 2008. Querying and mining of time series data: experimental comparison of representations and distance measures. *Proceedings of the VLDB Endowment*, 1(2), 1542–1552.

Dobranskyte-Niskota, A., Perujo, A., & Pregl, M. 2007. *Indicators to Assess Sustainability of Transport Activities—Part 1: Review of the Existing Transport Sustainability Indicator Initiatives and Development of an Indicator Set to Assess Transport Sustainability Performance.* `http://publications.jrc.ec.europa.eu/repository/handle/111111111/10416`. [Online; accessed 20-April-2017].

Dong, Yuxiao, Pinelli, Fabio, Gkoufas, Yiannis, Nabi, Zubair, Calabrese, Francesco, & Chawla, Nitesh V. 2015. Inferring unusual crowd events from mobile phone call detail records. *Pages 474–492 of: Joint European conference on machine learning and knowledge discovery in databases*. Springer.

Dorigo, Marco, Bonabeau, Eric, & Theraulaz, Guy. 2000. Ant algorithms and stigmergy. *Future Generation Computer Systems*, 16(8), 851–871.

Dorigo, Marco, Tuci, Elio, Groß, Roderich, Trianni, Vito, Labella, Thomas Halva, Nouyan, Shervin, Ampatzis, Christos, Deneubourg, Jean-louis, Baldassarre, Gianluca, Nolfi, Stefano, Mondada, Francesco, Floreano, Dario, & Gambardella, Luca Maria. 2005. The SWARM-BOTS Project. *Springer, Berlin, Heidelberg*, 31–44.

Dorigo, Marco, Birattari, Mauro, Garnier, Simon, Hamann, Heiko, de Oca, Marco Montes, Solnon, Christine, & Stutzle, Thomas. 2014. *Swarm Intelligence:*

9th International Conference, ANTS 2014, Brussels, Belgium, September 10-12, 2014. Proceedings. Vol. 8667. Springer.

Driemel, Anne, Krivosija, Amer, & Sohler, Christian. 2016. Clustering time series under the Fréchet distance. *In: SODA*.

EEA. 2005. *EEA core set of indicators: Guide. EEA Technical Report, Copenhagen*. `https://www.eea.europa.eu/publications/technical_report_2005_1`. [Online; accessed 19-April-2017].

Ermacora, Gabriele, Toma, Antonio, Bona, Basilio, Chiaberge, Marcello, Silvagni, Mario, Gaspardone, Marco, & Antonini, Roberto. 2013. A cloud robotics architecture for an emergency management and monitoring service in a smart city environment. *Polytech. Univ. Turin, Turin, Italy, Tech. Rep*.

Esling, Philippe, & Agón, Carlos. 2012. Time-series data mining. *ACM Comput. Surv.*, 45, 12:1–12:34.

Espie, Colin A, Lindsay, William R, & Espie, Lynne C. 1989. Use of the Sleep Assessment Device (Kelley and Lichstein, 1980) to validate insomniacs' self-report of sleep pattern. *Journal of Psychopathology and Behavioral Assessment*, 11(1), 71–79.

Fabeck, Gernot, & Mathar, Rudolf. 2008. Kernel-based learning of decision fusion in wireless sensor networks. *Pages 1–7 of: Information Fusion, 2008 11th International Conference on*. IEEE.

Fawaz, Mona. 2017. Planning and the refugee crisis: Informality as a framework of analysis and reflection. *Planning Theory*, 16(1), 99–115.

Flanagan, John A. 2003. A non-parametric approach to unsupervised learning and clustering of symbol strings and sequences. *Pages 128–133 of: Proceedings of the 4th Workshop on Self-Organizing Maps (WSOM03)*.

Fontecha, Jesús, Hervás, Ramón, & Bravo, José. 2011. A proposal for elderly frailty detection by using accelerometer-enabled smartphones.

Fontecha, Jesús, Navarro, Fco. Javier, Hervás, Ramón, & Bravo, José. 2012. Elderly frailty detection by using accelerometer-enabled smartphones and clinical information records. *Personal and Ubiquitous Computing*, 17, 1073–1083.

Frankel, Bernard L, Coursey, Robert D, Buchbinder, Rona, & Snyder, Frederick. 1976. Recorded and reported sleep in chronic primary insomnia. *Archives of General Psychiatry*, 33(5), 615–623.

Fu, Tak-chung. 2011. A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1), 164–181.

Gade, Rikke, & Moeslund, Thomas B. 2014. Thermal cameras and applications: a survey. *Machine vision and applications*, 25(1), 245–262.

Garnier, Simon, Gautrais, Jacques, & Theraulaz, Guy. 2007. The biological principles of swarm intelligence. *Swarm Intelligence*, 1(1), 3–31.

Gazi, Veysel, & Passino, Kevin M. 2011. Swarm Coordination and Control Problems. *Pages 15–25 of: Swarm Stability and Optimization*. Springer.

Ge, Xianping, & Smyth, Padhraic. 2000. Deformable Markov model templates for time-series pattern matching. *Pages 81–90 of: Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM.

Ghadage, Sudharani Ashok, & Doshi, Mrs Neeta Anilkumar. 2017. IoT Based Garbage Management (Monitor and Acknowledgement) System: A Review.

Glanzman, David L. 2011. Olfactory habituation: Fresh insights from flies. *Proceedings of the National Academy of Sciences*, 108(36), 14711–14712.

Grassé, PP. 1984. *Termitologia, Fondation des Sociétés. Construction. Tome II*.

Grignard, Arnaud, Alonso, Luis, Taillandier, Patrick, Gaudou, Benoit, Nguyen-Huu, Tri, Gruel, Wolfgang, & Larson, Kent. 2018. The Impact of New Mobility Modes on a City: A Generic Approach Using ABM. *Pages 272–280 of: International Conference on Complex Systems*. Springer.

Guiry, John J, Van de Ven, Pepijn, & Nelson, John. 2014. Multi-sensor fusion for enhanced contextual awareness of everyday activities with ubiquitous devices. *Sensors*, 14(3), 5687–5701.

Gundogdu, Didem, Incel, Ozlem D, Salah, Albert A, & Lepri, Bruno. 2016. Country-wide arrhythmia: emergency event detection using mobile phone data. *EPJ Data Science*, 5(1), 25.

Habitat, UN. 2016. Urbanization and development emerging futures. *World cities report*.

Hager, Gregory D. 2012. *Task-directed sensor fusion and planning: a computational approach*. Vol. 99. Springer Science & Business Media.

Halasz, a, Halasz, A, Hsieh, M Ani, Hsieh, M Ani, Berman, Spring, Berman, Spring, Kumar, Vijay, & Kumar, Vijay. 2007. Dynamic redistribution of a swarm of robots among multiple sites. *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, 2320–2325.

Hall, Mark, Frank, Eibe, Holmes, Geoffrey, Pfahringer, Bernhard, Reutemann, Peter, & Witten, Ian H. 2009. The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1), 10–18.

Han, Jinlu, & Chen, YangQuan. 2014. Multiple UAV formations for cooperative source seeking and contour mapping of a radiative signal field. *Journal of Intelligent & Robotic Systems*, 74(1-2), 323–332.

Haque, M.M., Chin, H.C., & Debnath, A.K. 2013. Sustainable, safe, smart—three key elements of Singapore's evolving transport policies. *Transport Policy*, 27.

Hardy, Keiran, & Maurushat, Alana. 2017. Opening up government data for Big Data analysis and public benefit. *Computer Law & Security Review*, 33(1), 30–37.

Hauert, Sabine, Leven, Severin, Varga, Maja, Ruini, Fabio, Cangelosi, Angelo, Zufferey, Jean-Christophe, & Floreano, Dario. 2011. Reynolds flocking in reality with fixed-wing robots: communication range vs. maximum turning rate. *Pages 5015–5020 of: Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*. IEEE.

Hebbani, Aparna, Colic-Peisker, Val, & Mackinnon, Mairead. 2017. Know thy neighbour: Residential integration and social bridging among refugee settlers in Greater Brisbane. *Journal of Refugee Studies*, 31(1), 82–103.

Heylighen, Francis. 2015. Stigmergy as a Universal Coordination Mechanism: components, varieties and applications. *Human Stigmergy: Theoretical Developments and New Applications; Springer: New York, NY, USA*.

Holland, John Henry. 1992. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press.

Holland, Owen, & Melhuish, Chris. 1999. Stigmergy, self-organization, and sorting in collective robotics. *Artificial life*, 5(2), 173–202.

Hossain, M Anwar, Atrey, Pradeep K, & El Saddik, Abdulmotaleb. 2009. Learning multisensor confidence using a reward-and-punishment mechanism. *IEEE Transactions on Instrumentation and Measurement*, 58(5), 1525–1534.

Hu, Yujie, Miller, Harvey J, & Li, Xiang. 2014. Detecting and analyzing mobility hotspots using surface networks. *Transactions in GIS*, 18(6), 911–935.

Huang, Xiao, & Oviatt, Sharon. 2005. Toward adaptive information fusion in multimodal systems. *Pages 15–27 of: International Workshop on Machine Learning for Multimodal Interaction*. Springer.

Israel, Martin. 2011. A UAV-based roe deer fawn detection system. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 38(1/C22), 51–55.

Ito, Takayuki, Kanamori, Ryo, Takahashi, Jun, Maestre, Iván Marsa, & De La Hoz, Enrique. 2012. The comparison of stigmergy strategies for decentralized traffic congestion control: Preliminary results. *Pages 146–156 of: Pacific Rim International Conference on Artificial Intelligence*. Springer.

Jahan, S. 2017. *Human Development Report, United Nations Development Programme*.

Janacek, Gareth J, Bagnall, Anthony J, & Powell, Michael. 2005. A likelihood ratio distance measure for the similarity between the fourier transform of time series. *Pages 737–743 of: Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer.

Jansen, F. Marijke, Prins, Rick G., Etman, A. A. M., van der Ploeg, Hidde P, de Vries, Sanne I., van Lenthe, Frank J, & Pierik, Frank H. 2015. Physical Activity in Non-Frail and Frail Older Adults. *In: PloS one*.

Jansen, Th, Chioncel, Nicoleta, & Dekkers, Hetty. 2006. Social cohesion and integration: Learning active citizenship. *British Journal of Sociology of Education*, 27(02), 189–205.

Jin, Bo, Ge, Yong, Zhu, Hengshu, Guo, Li, Xiong, Hui, & Zhang, Chao. 2014. Technology prospecting for high tech companies through patent mining. *Pages 220–229 of: Data Mining (ICDM), 2014 IEEE International Conference on*. IEEE.

Johansson, Robert, & Saffiotti, Alessandro. 2009. Navigating by stigmergy: A realization on an RFID floor for minimalistic robots. *Pages 245–252 of: Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE.

Johns, Murray W. 1991. A new method for measuring daytime sleepiness: the Epworth sleepiness scale. *sleep*, 14(6), 540–545.

Kachitvichyanukul, Voratas. 2012. Comparison of three evolutionary algorithms: GA, PSO, and DE. *Industrial Engineering and Management Systems*, 11(3), 215–223.

Keler, Andreas, & Krisp, Jukka M. 2016. Is there a relationship between complicated crossings and frequently visited locations? A case study with boro taxis and OSM in NYC. *Pages 111–121 of: 13th International Conference on Location-Based Services*.

Keogh, Eamonn, & Kasetty, Shruti. 2003. On the need for time series data mining benchmarks: a survey and empirical demonstration. *Data Mining and knowledge discovery*, 7(4), 349–371.

Keogh, Eamonn, Lonardi, Stefano, & Ratanamahatana, Chotirat Ann. 2004. Towards parameter-free data mining. *Pages 206–215 of: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM.

Keyman, E Fuat. 2016. Turkey at the heart of the refugee and ISIL crises: Can the buffer state be a solution. *Rev. Int'l L. & Pol.*, 12, 5.

Khaleghi, Bahador, Khamis, Alaa, Karray, Fakhreddine O, & Razavi, Saiedeh N. 2013. Multisensor data fusion: A review of the state-of-the-art. *Information Fusion*, 14(1), 28–44.

Kohavi, Ron. 1995. The power of decision tables. *Pages 174–189 of: European Conference on Machine Learning*. Springer.

Kong, Q-J, Chen, Yikai, & Liu, Yuncai. 2009. A fusion-based system for road-network traffic state surveillance: a case study of Shanghai. *IEEE Intelligent Transportation Systems Magazine*, 1(1), 37–42.

Kong, Qing-Jie, Zhao, Qiankun, Wei, Chao, & Liu, Yuncai. 2013. Efficient traffic state estimation for large-scale urban road networks. *IEEE Transactions on Intelligent Transportation Systems*, 14(1), 398–407.

Krick, Laura, Broucke, Mireille, & Francis, Bruce. 2008. Getting Mobile Autonomous Robots to Form a Prescribed Geometric Arrangement. *Pages 149–159 of: Recent Advances in Learning and Control*. Springer.

Kuang, Weiming, An, Shi, & Jiang, Huifu. 2015. Detecting traffic anomalies in urban areas using taxi GPS data. *Mathematical Problems in Engineering*, 2015.

Kumar, Prashant, Morawska, Lidia, Martani, Claudio, Biskos, George, Neophytou, Marina, Di Sabatino, Silvana, Bell, Margaret, Norford, Leslie, & Britter, Rex. 2015. The rise of low-cost sensing for managing air pollution in cities. *Environment International*, 75, 199–205.

Kurdi, Heba, How, Jonathon, & Bautista, Guillermo. 2016. Bio-inspired algorithm for task allocation in Multi-UAV search and rescue missions. *Page 1377 of: AIAA Guidance, Navigation, and Control Conference*.

Kurihara, Satoshi, Tamaki, Hiroshi, Numao, Masayuki, Yano, Junji, Kagawa, Kouji, & Morita, Tetsuo. 2009. Traffic congestion forecasting based on pheromone communication model for intelligent transport systems. *Pages 2879–2884 of: Evolutionary Computation, 2009. CEC'09. IEEE Congress on*. IEEE.

Kuyucu, Tüze, Tanev, Ivan, & Shimohara, Katsunori. 2015. Superadditive effect of multi-robot coordination in the exploration of unknown environments via stigmergy. *Neurocomputing*, 148, 83–90.

Lee, Geunho, Chong, Nak Young, & Christensen, Henrik. 2010. Tracking multiple moving targets with swarms of mobile robots. *Intelligent Service Robotics*, 3(2), 61–72.

Lega, M, & Napoli, RMA. 2008. A new approach to solid waste landfills aerial monitoring. *WIT Transactions on Ecology and the Environment*, 109, 193–199.

Lega, M, Ceglie, D, Persechino, G, Ferrara, C, & Napoli, RMA. 2012. Illegal dumping investigation: a new challenge for forensic environmental engineering. *WIT Transactions on Ecology and the Environment*, 163, 3–11.

Lega, Massimiliano, & Napoli, Rodolfo MA. 2010. Aerial infrared thermography in the surface waters contamination monitoring. *Desalination and water treatment*, 23(1-3), 141–151.

Li, Xiaolong, Pan, Gang, Wu, Zhaohui, Qi, Guande, Li, Shijian, Zhang, Daqing, Zhang, Wangsheng, & Wang, Zonghui. 2012. Prediction of urban human mobility using large-scale taxi traces and its applications. *Frontiers of Computer Science*, 6(1), 111–121.

Liang, Zilu, Ploderer, Bernd, Liu, Wanyu, Nagata, Yukiko, Bailey, James, Kulik, Lars, & Li, Yuxuan. 2016. SleepExplorer: a visualization tool to make sense of correlations between personal sleep data and contextual factors. *Personal and Ubiquitous Computing*, 20(6), 985–1000.

Litman, T. 2011. *Well measured: Developing indicators for sustainable and livable transport planning. Victoria Transport Policy Institute.* `http://www.vtpi.org/wellmeas.pdf`. [Online; accessed 20-April-2017].

Liu, Y., Wang, F., Xiao, Y., & Gao, S. 2012. Urban land uses and traffic 'source-sink areas': Evidence from GPS-enabled taxi data in Shanghai. *Landscape and Urban Planning*, 106(1), 73–87.

Liu, Yandong, Wang, Lujia, Huang, Huaiyang, Liu, Ming, & Xu, Cheng-zhong. 2017. A Novel Swarm Robot Simulation Platform for Warehousing Logistics. 2669–2674.

Lu, Qi, Moses, Melanie. E., & Hecker, Joshua P. 2016a. A Scalable and Adaptable Multiple-Place Foraging Algorithm for Ant-Inspired Robot Swarms.

Lu, Qi, Hecker, Joshua P., & Moses, Melanie. E. 2016b. The MPFA: A Multiple-Place Foraging Algorithm for Biologically-Inspired Robot Swarms. *Pages 3815–3821 of: IEEE/RSJ International Conference on Intelligent Robots and Systems*.

Lu, Qi, Hecker, Joshua P., & Moses, Melanie E. 2018. Multiple-place swarm foraging with dynamic depots. *Autonomous Robots*.

Lu, Yu, Chua, Gim Guan, Wu, Huayu, & Ong, Clement Shi Qi. 2016c. An Intelligent System for Taxi Service Monitoring, Analytics and Visualization. *Pages 4256–4257 of: IJCAI*.

Luo, QiNan, & Duan, HaiBin. 2013. An improved artificial physics approach to multiple UAVs/UGVs heterogeneous coordination. *Science China Technological Sciences*, 56(10), 2473–2479.

Lyytinen, Eveliina. 2015. Refugees' Conceptualizations of "Protection Space": Geographical Scales of Urban Protection and Host–Refugee Relations. *Refugee Survey Quarterly*, 34(2), 45–77.

Madanipour, Ali. 2015. Social exclusion and space. *Pages 237–245 of: The city reader*. Routledge.

Makrai, Gabor. 2016. Efficient method for large-scale spatio-temporal hotspot analysis (GIS Cup). *Pages 1–18 of: Proc. 24th ACM SIGSPATIAL Int. Conf. Adv. Geograph. Inf. Syst*.

Mannila, H, & Seppänen, J. 2001. Recognizing similar situations from event sequences. *Pages 1–16 of: First SIAM Conference on Data Mining*.

Maquet, Pierre. 2001. The role of sleep in learning and memory. *science*, 294(5544), 1048–1052.

Marsh, Leslie, & Onof, Christian. 2008. Stigmergic epistemology, stigmergic cognition. *Cognitive Systems Research*, 9(1-2), 136–149.

Matte, Thomas D., Ross, Zev, Kheirbek, Iyad, Eisl, Holger M, Johnson, Sarah, Gorczynski, John E, Kass, Daniel E., Markowitz, Steven, Pezeshki, Grant, & Clougherty, Jane Ellen. 2013. Monitoring intraurban spatial patterns of multiple combustion air pollutants in New York City: Design and implementation. *Journal of Exposure Science and Environmental Epidemiology*, 23, 223–231.

Mazimpaka, Jean Damascène, & Timpf, Sabine. 2016. Trajectory data mining: A review of methods and applications. *Journal of Spatial Information Science*, 2016(13), 61–99.

McCann, Philip, & Ortega-Argilés, Raquel. 2015. Smart specialization, regional growth and applications to European Union cohesion policy. *Regional Studies*, 49(8), 1291–1302.

McIsaac, Elizabeth. 2003. *Nation building through cities: A new deal for immigrant settlement in Canada*. Caledon Institute of Social Policy.

Metsis, Vangelis, Kosmopoulos, Dimitrios, Athitsos, Vassilis, & Makedon, Fillia. 2014. Non-invasive analysis of sleep patterns via multimodal sensor input. *Personal and ubiquitous computing*, 18(1), 19–26.

Mezura-Montes, Efrén, Velázquez-Reyes, Jesús, & Coello, Carlos A. Coello. 2006a. A comparative study of differential evolution variants for global optimization. *In: GECCO*.

Mezura-Montes, Efrñn, Velázquez-Reyes, Jesús, & Coello Coello, Carlos A. 2006b. A comparative study of differential evolution variants for global optimization. *Pages 485–492 of: Proceedings of the 8th annual conference on Genetic and evolutionary computation*. ACM.

Minaeian, Sara, Liu, Jian, & Son, Young-Jun. 2016. Vision-based target detection and localization via a team of cooperative UAV and UGVs. *IEEE Transactions on systems, man, and cybernetics: systems*, 46(7), 1005–1016.

Mohan, B Chandra, & Baskaran, R. 2012. A survey: Ant Colony Optimization based recent research and implementation on several engineering domain. *Expert Systems with Applications*, 39(4), 4618–4627.

Monk, Timothy H, Buysse, Daniel J, Kennedy, Kathy S, Potts, Jaime M, DeGrazia, Jean M, & Miewald, Jean M. 2003. Measuring sleep habits without using a diary: the sleep timing questionnaire. *Sleep*, 26(2), 208–212.

Narzt, W, Wilflingseder, U, Pomberger, G, Kolb, D, & Hörtner, H. 2010. Self-organising congestion evasion strategies using ant-based pheromones. *IET Intelligent Transport Systems*, 4(1), 93–102.

Neumann, Patrick P, Hernandez Bennetts, Victor, Lilienthal, Achim J, Bartholmai, Matthias, & Schiller, Jochen H. 2013. Gas source localization with a micro-drone using bio-inspired and particle filter-based algorithms. *Advanced Robotics*, 27(9), 725–738.

Neuwirth, Abraham. 2016. Fare Share : Flow and Efficiency in NYC ' s Taxi System.

Niwattanakul, Suphakit, Singthongchai, Jatsada, Naenudorn, Ekkachai, & Wanapu, Supachanun. 2013. Using of Jaccard coefficient for keywords similarity. *In: Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 1.

Ong, Adrian A, & Gillespie, M Boyd. 2016. Overview of smartphone applications for sleep analysis. *World Journal of Otorhinolaryngology-Head and Neck Surgery*, 2(1), 45–49.

Pan, Bei, Zheng, Yu, Wilkie, David, & Shahabi, Cyrus. 2013. Crowd sensing of traffic anomalies based on human mobility and social media. *In: SIGSPATIAL/GIS*.

Papadimitriou, Spiros, Sun, Jimeng, & Philip, S Yu. 2006. Local correlation tracking in time series. *Pages 456–465 of: Data Mining, 2006. ICDM'06. Sixth International Conference on*. IEEE.

Parker, Lynne E. 2003. Current research in multirobot systems. *Artificial Life and Robotics*, 7(1-2), 1–5.

Parkka, Juha, Ermes, Miikka, Antila, Kari, Van Gils, Mark, Manttari, Ari, & Nieminen, Heikki. 2007. Estimating intensity of physical activity: a comparison of wearable accelerometer and gyro sensors and 3 sensor locations. *Pages 1511–1514 of: Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*. IEEE.

Parunak, H Van, Purcell, Michael, & O'Connell, Robert. 2002. Digital pheromones for autonomous coordination of swarming UAV's. *Page 3446 of: 1st UAV Conference*.

Parunak, H Van Dyke. 2005. A survey of environments and mechanisms for human-human stigmergy. *Pages 163–186 of: International workshop on environments for multi-agent systems*. Springer.

Patrick, Deval L., Murray, Timothy P., & Sullivan, Richard K., Kimmel Kenneth L. 2013 (04). *Massachusetts 2010-2020 Solid Waste Master Plan*. Tech. rept. Commonwealth of Massachusetts, Executive office of Energy & Environmental Affairs.

Payton, David, Estkowski, Regina, & Howard, Mike. 2005. Pheromone Robotics and the Logic of Virtual Pheromones. 45–57.

Pellicer, Soledad, Santa, Guadalupe, Bleda, Andrés L., Maestre, Rafael, Jara, Antonio J., & Gómez-Skarmeta, Antonio F. 2013. A Global Perspective of Smart Cities: A Survey. *2013 Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, 439–444.

Peng, Chengbin, Jin, Xiaogang, Wong, Ka-Chun, Shi, Meixia, & Lio, Pietro. 2012. Collective Human Mobility Pattern from Taxi Trips in Urban Area. *PLoS ONE*, 7(8).

Peng, Jinglin, Wang, Hongzhi, Li, Jianzhong, & Gao, Hong. 2016. Set-based Similarity Search for Time Series. *In: SIGMOD Conference*.

Perianez-Forte, I, & Cervantes, M. 2013. *Innovation-driven Growth in Regions: The Role of Smart Specialisation (PRELIMINARY VERSION)*.

Poli, Riccardo, Kennedy, James, & Blackwell, Tim. 2007. Particle swarm optimization. *Swarm intelligence*, 1(1), 33–57.

Qiu, Huaxin, & Duan, Haibin. 2017. Pigeon interaction mode switch-based UAV distributed flocking control under obstacle environments. *ISA transactions*, 71, 93–102.

Qu, Yaohong, Zhang, Yintao, & Zhang, Youmin. 2015. A UAV solution of regional surveillance based on pheromones and artificial potential field theory. *Pages 380–385 of: Unmanned Aircraft Systems (ICUAS), 2015 International Conference on*. IEEE.

Quinlan, J. R. 1986. Induction of Decision Trees. *Machine Learning*, 1, 81–106.

Ramos, Vitorino, & Almeida, Filipe. 2004. Artificial ant colonies in digital image habitats-a mass behaviour effect study on pattern recognition. *arXiv preprint cs/0412086*.

Ramos, Vitorino, Muge, Fernando, & Pina, Pedro. 2002. Self-Organized Data and Image Retrieval as a Consequence of Inter-Dynamic Synergistic Relationships in Artificial Ant Colonies. *HIS*, 87, 500–512.

Reynolds, Craig W. 1987a. Flocks, herds and schools: A distributed behavioral model. *Pages 25–34 of: ACM SIGGRAPH computer graphics*, vol. 21. ACM.

Reynolds, Craig W. 1987b. Flocks, herds and schools: A distributed behavioral model. *Pages 25–34 of: ACM SIGGRAPH computer graphics*, vol. 21. ACM.

Rodriguez, J, Castiblanco, C, Mondragon, I, & Colorado, J. 2014. Low-cost quadrotor applied for visual detection of landmine-like objects. *Pages 83–88 of: Unmanned Aircraft Systems (ICUAS), 2014 International Conference on*. IEEE.

Sadeh, Avi. 2011. The role and validity of actigraphy in sleep medicine: an update. *Sleep medicine reviews*, 15(4), 259–267.

Sagl, Günther, Loidl, Martin, & Beinat, Euro. 2012. A visual analytics approach for extracting spatio-temporal urban mobility information from mobile network traffic. *ISPRS International Journal of Geo-Information*, 1(3), 256–271.

Salah, Albert Ali, Pentland, Alex, Lepri, Bruno, Letouzé, Emmanuel, Vinck, Patrick, de Montjoye, Yves-Alexandre, Dong, Xiaowen, & Dağdelen, Özge. 2018. Data for Refugees: The D4R Challenge on Mobility of Syrian Refugees in Turkey. *arXiv preprint arXiv:1807.00523*.

Salvini, Pericle. 2018. Urban robotics: Towards responsible innovations for our cities. *Robotics and Autonomous Systems*, 100, 278–286.

Santos, C Costa, Bernardes, João, Vitányi, Paul MB, & Antunes, Luis. 2006. Clustering fetal heart rate tracings by compression. *Pages 685–690 of: Computer-Based Medical Systems, 2006. CBMS 2006. 19th IEEE International Symposium on*. IEEE.

Sauter, John A, Matthews, Robert, Van Dyke Parunak, H, & Brueckner, Sven A. 2005a. Performance of digital pheromones for swarming vehicle control. *Pages 903–910 of: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*. ACM.

Sauter, John A, Matthews, Robert, Van Dyke Parunak, H, & Brueckner, Sven A. 2005b. Performance of digital pheromones for swarming vehicle control. *Pages 903–910 of: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*. ACM.

Schmolke, Amelie. 2009. Benefits of Dispersed Central-Place Foraging: An Individual-Based Model of a Polydomous Ant Colony. *The American Naturalist*, 173(6), 772–778.

Senanayake, Madhubhashi, Senthooran, Ilankaikone, Barca, Jan Carlo, Chung, Hoam, Kamruzzaman, Joarder, & Murshed, Manzur. 2016. Search and tracking algorithms for swarms of robots: A survey. *Robotics and Autonomous Systems*, 75, 422–434.

Shahrokni, Hossein, van der Heijde, Bram, Lazarevic, David, & Brandt, Nils. 2014. Big Data GIS Analytics Towards Efficient Waste Management in Stockholm. *Proceedings of the 2014 conference ICT for Sustainability*, 140–147.

Sharma, Rajnikant, Kothari, Mangal, Taylor, Clark N, & Postlethwaite, Ian. 2010. Cooperative target-capturing with inaccurate target information. *Pages 5520–5525 of: American Control Conference (ACC), 2010*. IEEE.

Shatkay, Hagit, & Zdonik, Stanley B. 1996. Approximate queries and representations for large data sequences. *Pages 536–545 of: Data Engineering, 1996. Proceedings of the Twelfth International Conference on*. IEEE.

Siegemund, Katja, Thomas, Edward J, Zhao, Yuting, Pan, Jeff, & Assmann, Uwe. 2011. Towards ontology-driven requirements engineering. *In: Workshop semantic web enabled software engineering at 10th international semantic web conference (ISWC), Bonn*.

Silm, Siiri, & Ahas, Rein. 2014. Ethnic differences in activity spaces: A study of out-of-home nonemployment activities with mobile phone data. *Annals of the Association of American Geographers*, 104(3), 542–559.

Singh, Vivek Kumar, Bozkaya, Burcin, & Pentland, Alex. 2015. Money walks: implicit mobility behavior and financial well-being. *PloS one*, 10(8), e0136628.

Snoek, Jasper, Larochelle, Hugo, & Adams, Ryan P. 2012. Practical bayesian optimization of machine learning algorithms. *Pages 2951–2959 of: Advances in neural information processing systems*.

Soysal, Onur, & Şahin, Erol. 2006. A macroscopic model for self-organized aggregation in swarm robotic systems. *Pages 27–42 of: International Workshop on Swarm Robotics*. Springer.

Steels, Luc. 1990. Cooperation between distributed agents through self-organisation. *Pages 8–14 of: Intelligent Robots and Systems' 90.'Towards a New Frontier of Applications', Proceedings. IROS'90. IEEE International Workshop on*. IEEE.

Storn, Rainer, & Price, Kenneth V. 1997. Differential Evolution - A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *J. Global Optimization*, 11, 341–359.

Tabibiazar, Arash, & Basir, Otman. 2011. Kernel-based modeling and optimization for density estimation in transportation systems using floating car data. *Pages 576–581 of: Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*. IEEE.

Tan, Ying, & Zheng, Zhong-yang. 2013. Research advance in swarm robotics. *Defence Technology*, 9(1), 18–39.

Taylor, Jeffrey, Zhou, Xuesong, Rouphail, Nagui M, & Porter, Richard J. 2015. Method for investigating intradriver heterogeneity using vehicle trajectory data: A Dynamic Time Warping approach. *Transportation Research Part B: Methodological*, 73, 59–80.

Theraulaz, Guy, & Bonabeau, Eric. 1999. A brief history of stigmergy. *Artificial life*, 5(2), 97–116.

Tompkins, Adrian M, & McCreesh, Nicky. 2016. Migration statistics relevant for malaria transmission in Senegal derived from mobile phone data and used in an agent-based migration model. *Geospatial health*, 11(1 Suppl), 408.

Valckenaers, Paul, Saint Germain, Bart, Verstraete, Paul, Van Brussel, Hendrik, *et al.* . 2007. MAS coordination and control based on stigmergy. *Computers in Industry*, 58(7), 621–629.

Veloso, Marco, Phithakkitnukoon, Santi, & Bento, Carlos. 2011. Urban mobility study using taxi traces. *Pages 23–30 of: Proceedings of the 2011 international workshop on Trajectory data mining and analysis*. ACM.

Vernon, David, Metta, Giorgio, & Sandini, Giulio. 2007. A survey of artificial cognitive systems: Implications for the autonomous development of mental capabilities in computational agents. *IEEE transactions on evolutionary computation*, 11(2), 151–180.

Vesterstrom, Jakob, & Thomsen, René. 2004. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No.04TH8753)*, 2, 1980–1987 Vol.2.

Viguria, Antidio, Maza, Ivan, & Ollero, Anibal. 2010. Distributed service-based cooperation in aerial/ground robot teams applied to fire detection and extinguishing missions. *Advanced Robotics*, 24(1-2), 1–23.

Vlachos, Michail, Yu, Philip, & Castelli, Vittorio. 2005. On periodicity detection and structural periodic similarity. *Pages 449–460 of: Proceedings of the 2005 SIAM international conference on data mining*. SIAM.

Wagner, Israel A, Altshuler, Yaniv, Yanovski, Vladimir, & Bruckstein, Alfred M. 2008. Cooperative cleaners: A study in ant robotics. *The International Journal of Robotics Research*, 27(1), 127–151.

Wang, Yiheng, Liang, Alei, & Guan, Haibing. 2011. Frontier-based multi-robot map exploration using particle swarm optimization. *Pages 1–6 of: Swarm Intelligence (SIS), 2011 IEEE Symposium on*. IEEE.

Wilson, Daniel B, Goktogan, AH, & Sukkarieh, Salah. 2012. UAV rendezvous: From concept to flight test. *Pages 1–10 of: Australasian Conference on Robotics and Automation (ACRA)*, vol. 25.

Wilson, Keith G, Watson, Shannon T, & Currie, Shawn R. 1998. Daily diary and ambulatory activity monitoring of sleep in patients with insomnia associated with chronic musculoskeletal pain. *Pain*, 75(1), 75–84.

Winfield, Alan F. T. 2009. *Towards an Engineering Science of Robot Foraging*. Berlin, Heidelberg: Springer Berlin Heidelberg. Pages 185–192.

Wu, Guohua, Mallipeddi, Rammohan, Suganthan, Ponnuthurai Nagaratnam, Wang, Rui, & Chen, Huangke. 2016. Differential evolution with multi-population based ensemble of mutation strategies. *Inf. Sci.*, 329, 329–345.

Xiong, Yimin, & Yeung, Dit-Yan. 2004. Time series clustering with ARMA mixtures. *Pattern Recognition*, 37(8), 1675–1689.

Yan, Zhi, Jouandeau, Nicolas, & Cherif, Arab Ali. 2013. A survey and analysis of multi-robot coordination. *International Journal of Advanced Robotic Systems*, 10(12), 399.

Yang, Guang-Zhong, Bellingham, Jim, Dupont, Pierre E., Fischer, Peer, Floridi, Luciano, Full, Robert, Jacobstein, Neil, Kumar, Vijay, McNutt, Marcia, Merrifield, Robert, Nelson, Bradley J., Scassellati, Brian, Taddeo, Mariarosaria, Taylor, Russell, Veloso, Manuela, Wang, Zhong Lin, & Wood, Robert. 2018. The grand challenges of Science Robotics. *Science Robotics*, 3(14), eaar7650.

Yoon, Jungkeun, Noble, Brian, & Liu, Mingyan. 2007. Surface street traffic estimation. *Pages 220–232 of: Proceedings of the 5th international conference on Mobile systems, applications and services*. ACM.

Zaharie, Daniela. 2007. A Comparative Analysis of Crossover Variants in Differential Evolution.

Zanella, Andrea, Bui, Nicola, Castellani, Angelo, Vangelista, Lorenzo, & Zorzi, Michele. Internet of Things for Smart Cities.

Zedadra, Ouarda, Seridi, Hamid, Jouandeau, Nicolas, & Fortino, Giancarlo. 2015. Design and analysis of cooperative and non cooperative stigmergy-based models for foraging. *Pages 85–90 of: Computer Supported Cooperative Work in Design (CSCWD), 2015 IEEE 19th International Conference on*. IEEE.

Zhan, Xianyuan, & Ukkusuri, Satish V. 2014. Measuring the Efficiency of Urban Taxi Service System.

Zhang, Wangsheng, Qi, Guande, Pan, Gang, Lu, Hua, Li, Shijian, & Wu, Zhaohui. 2015. City-scale social event detection and evaluation with taxi traces. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 6(3), 40.

Zhang, Youmin, & Mehrjerdi, Hasan. 2013. A survey on multiple unmanned vehicles formation control and coordination: Normal and fault situations. *Pages 1087–1096 of: Unmanned Aircraft Systems (ICUAS), 2013 International Conference on*. IEEE.

Zhao, Qiankun, Kong, Qing-Jie, Xia, Yingjie, & Liu, Yuncai. 2011. Sample size analysis of GPS probe vehicles for urban traffic state estimation. *Pages 272–276 of: Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*. IEEE.

Zheng, Yu, Liu, Furui, & Hsieh, Hsun-Ping. 2013. U-Air: when urban air quality inference meets big data. *In: KDD*.

Zheng, Yu, Liu, Tong, Wang, Yilun, Zhu, Yanmin, Liu, Yanchi, & Chang, Eric. 2014a. Diagnosing New York city's noises with ubiquitous data. *In: UbiComp*.

Zheng, Yu, Capra, Licia, Wolfson, Ouri, & Yang, Hai. 2014b. Urban Computing: Concepts, Methodologies, and Applications. *ACM TIST*, 5, 38:1–38:55.

Zhu, Jindan, Pande, Amit, Mohapatra, Prasant, & Han, Jay J. 2015. Using deep learning for energy expenditure estimation with wearable sensors. *Pages 501–506 of: E-health Networking, Application & Services (HealthCom), 2015 17th International Conference on*. IEEE.

Zia, Kashif, Din, Ahmad, Shahzad, Khurram, & Ferscha, Alois. 2017. A Cognitive Agent-based Model for Multi-Robot Coverage at a City Scale. *Complex Adaptive Systems Modeling*, 5(1), 1.