Green Swarm: Greener Routes with Bio-inspired Techniques

Daniel H. Stolfi[a,*], Enrique Alba[a]

[a]*Departamento de Lenguajes y Ciencias de la Computación, University of Malaga, Spain*

Abstract

This article proposes a mobility architecture, called Green Swarm, to reduce greenhouse gas emissions from road traffic in smart cities. The traffic flow optimization of four European cities: Malaga, Stockholm, Berlin, and Paris, is addressed with new case studies importing each city's actual roads and traffic lights from OpenStreetMap into the SUMO traffic simulator, so as to find the best ways to redirect the traffic flow, and advise drivers. Additionally, the proposal is compared with three other strategies, which are also combined with Green Swarm in order to improve metrics such as travel times, gas emissions, and fuel consumption. This results in reductions in gas emissions as well as in travel times and fuel consumption in more than 500 city scenarios. The proposal has also been tested in scenarios where not all drivers are using it, to observe the change in traffic conditions when it is only in partial use, successfully paving the way for future sustainable cities.

Keywords: Evolutionary algorithm, road traffic, smart city, smart mobility, gas emissions, Wi-Fi connections

1. Introduction

Currently, the number of inhabitants in urban areas is increasing and is expected to rise to 75% by 2050 [1]. As a result, a number of new services are required to help solve the new types of problems related to the huge number of people living in reduced geographical areas. According to [2], 50% of Europeans use a car every day, while 38% of them encounter problems as they travel around cities. An important number of Europeans believe that the truly serious problems within cities are caused by air pollution (81%), road congestion (76%), traveling costs (74%), accidents (73%), and noise pollution (72%).

Human health, economic development, traffic jams, environmental pollution, and waste management are some of the problems that strongly affect different aspects of our society. These problems represent a challenge for city governments if they wish to manage these growing issues in smarter ways. Research on smart cities and Intelligent Transportation Systems (ITS) is therefore a must and so is supported by major agencies worldwide.

Road traffic is a well-known source of air pollution in urban areas [3, 4]. Air quality is an important issue for the economy, the environment, and of course, human health. Poor air quality contributes to respiratory and cardiovascular diseases as well as to lung cancer [5]. It also has an

*Corresponding author

Email addresses: dhstolfi@lcc.uma.es (Daniel H. Stolfi), eat@lcc.uma.es (Enrique Alba)

economic impact, shortening lives, increasing medical costs, and reducing productivity through loss of working days. Additionally, air pollution can damage buildings, ecosystems, crops, and has an impact on the climate, since some air pollutants act as greenhouse gases [6].

In this article, the Green Swarm architecture is proposed. It is aimed at sustainability with an explicit, special focus on mobility in cities from a different point of view: the reduction of greenhouse gas emissions, specifically: i) Carbon Monoxide (CO), ii) Carbon Dioxide (CO_2), iii) Hydrocarbons (HC), iv) Particulate Matter (PM), and v) Nitrogen Oxides (NO).

Green Swarm is shown to be capable of reducing not only travel times, but also gas emissions from vehicles, as well as fuel consumption, by suggesting alternative routes and preventing traffic jams. Both actions contribute to a more environmentally friendly way of driving. This is a new approach consisting of rerouting vehicles through personalized, precalculated route segments at several points throughout the city which differs from previous related work not only in the *offline-online* strategy used, but also in the intelligent Wi-Fi nodes.

The rest of this paper is organized as follows: Section 2 reviews several publications related to our proposal. Section 3 describes the Green Swarm architecture and our proposed intelligent algorithms. In Section 4 three competitor strategies are presented. Section 5 presents the case study analysis. In Section 6 the experiments conducted and the results are discussed. Finally, in Section 7, conclusions and future work are given.

2. Related Work

In this section several articles which are directly related to our proposal are reviewed. They are considered to be similar because they consider the whole city and have similar motivations and approaches, at least in part.

A green Vehicle Traffic Routing System (VTRS) that reduces fuel consumption and consequently CO_2 emissions via a bio-inspired algorithm, combined with a fuel consumption model, is introduced in [7]. It consists of an Ant-based Vehicle Congestion Avoidance System (AVCAS) that uses the Signalized Intersection Design and Research Aid (SIDRA) fuel consumption and emission model in its vehicle routing procedure. This system is able to reduce fuel consumption by finding the least congested shortest paths in a simulation of Kuala Lumpur imported from OpenStreetMap [8] into the SUMO [9] traffic simulator. In contrast, the approach presented in this paper centers on reducing gases, namely CO_2, as a way of improving the rest of the metrics by preventing jams, in five case studies.

An approach for dynamic calculation of optimal traffic routes is presented in [10]. They use a new multi-objective algorithm called ISATOPSIS to avoid congestion by using the average travel speed of traffic (obtained from sensors deployed around smart cities) and the length of roads to find the optimal paths. The results of the proposed algorithm have been compared to the shortest path Dijkstra algorithm and other strategies in two real cities imported into the SUMO traffic simulator from OpenStreetMap. In our article an evolutionary algorithm is used to optimize the case studies instead of SA, hopefully avoiding local optima more frequently.

In [11] the authors propose an architecture to control and manage the utilization of road transport networks to prevent traffic congestion. Their architecture divides an urban area into smaller regions while the capacity of each road segment within these regions is reserved by users on demand. Additionally, a real-time scheduling algorithm to solve the route reservation problem is analyzed using a realistic road transport scenario in a large area in Nicosia, Cyprus, extracted from OpenStreetMap and imported into SUMO. Their results indicate that congestion can be avoided and travel times improved after the application of a route reservation algorithm over a specific region. In our article Green Swarm has been better tested on data taken from many cities with the intention of observing several metrics in detail, using our simulation approach.

A bi-level optimization framework to settle the optimal traffic signal setting problem is presented in [12]. By using a Hybrid Genetic Algorithm, the authors decouple the original bi-level problem into two single-level problems employing SUMO and then solve them sequentially. The upper-level problem sets the traffic signal to minimize the drivers average travel time, and the lower-level problem achieves network equilibrium using the settings calculated in the upper level. The experiments were conducted in an urban area of Chicago obtained from OpenStreetMap. In our proposal, the focus is on the rerouting of vehicles to prevent traffic jams, using traffic lights as rerouting nodes, without changing traffic light cycles.

In [13] the authors address the optimization of vehicular traffic flows by using road-side units (V2I) to gather information with which to redirect vehicles to less congested roads and reduce CO_2 emissions. The proposed algorithm, CAVE, implements a rerouting strategy for vehicles using the OMNet++ simulator with the Veins framework connected to SUMO to reduce several metrics by showing less congested routes to the drivers. Although our proposal also reduces travel times and gas emissions, our aim is to implement a lightweight infrastructure and a high reutilization of urban devices such as traffic lights and existing networks to reduce costs.

In summary, our contribution is an affordable system for reducing travel times, fuel consumption, and gas emissions of vehicles by suggesting alternative routes which are tailored to every single driver in the city, avoiding sending them through the same streets and preventing traffic jams. An explicit use of sustainability indicators and actions is implemented in order to achieve a system with the expected balance between the driver's convenience and the city's health.

3. The Green Swarm Architecture

The Green Swarm (GS) architecture, is an evolution of our preliminary work [14] redesigned and adapted in this case to reduce not only travel times, but also greenhouse gas emissions, and fuel consumption. This is a significantly different line of research where a large number of weak points have been addressed, to finally design and implement a new system (Green Swarm) based on the following, new contributions:

1. GS uses a new mathematical function to measure the quality of the solutions. This generates a new search landscape where new algorithms and unseen performances are analyzed.
2. The algorithms (now EfRA and GrA) have been revised and their performance improved to get better results in shorter times.
3. There is a study of the relationships between metrics (travel time, CO_2, fuel, etc.) which is unique to the present paper.
4. Four different cities (Malaga, Stockholm, Berlin and Paris) have been optimized, plus one extra scenario consisting of real traffic flows (Alameda) which amounts to more than 500 scenarios in five case studies. In previous work just one city or parts of it has been tested. The conclusions drawn from working with four cities give this study a robust endorsement as a comparison analysis for future work in this area.
5. We have pushed the boundaries of existing algorithms by significantly increasing the number of vehicles in each scenario (up to 5800 vs. 1200). This means considerably larger computational times, and improved realism.

6. There were no competitors whatsoever in past articles of the literature. Thus three competitors have been introduced, not only to test our strategy, but also to complement it. In this sense, our new proposal has been strongly tested compared to past, related systems.
7. As not everyone is keen on using new technologies until they are firmly established, in this article a user acceptance study has been conducted so as to address not only the scientific aspect of the proposal but also the social one.

GS can be installed in modern cities with a minimum investment as it is able to use already existing infrastructure such as traffic lights controlled by a computer, Wi-Fi connectivity, mobile phones, and tablets. GS comprises the following components: i) Nodes installed at traffic lights which communicate with vehicles to know their destination and send them a new route around the city; ii) The Eco-friendly Route Algorithm (EfRA) which calculates the configuration of the system; iii) The Green Algorithm (GrA) which is executed in the nodes to suggest eco-friendly routes to vehicles; and iv) Mobile devices such as smartphones and tablets for the user terminals, or even On Board Units (OBU) installed in vehicles.

In Figure 1 the schema of the GS architecture is depicted. It is divided into two stages: an offline stage called *Setup Stage* and an online stage called *Green Stage*. In the *Setup Stage*, the configuration of the nodes is calculated by EfRA so that each node will be able to suggest an eco-friendly route to a vehicle depending on its final destination, based on a probability value. These probabilities are calculated before deploying the system (training phase) by optimizing four different traffic distributions of the city, as this improves the robustness of GS [14].

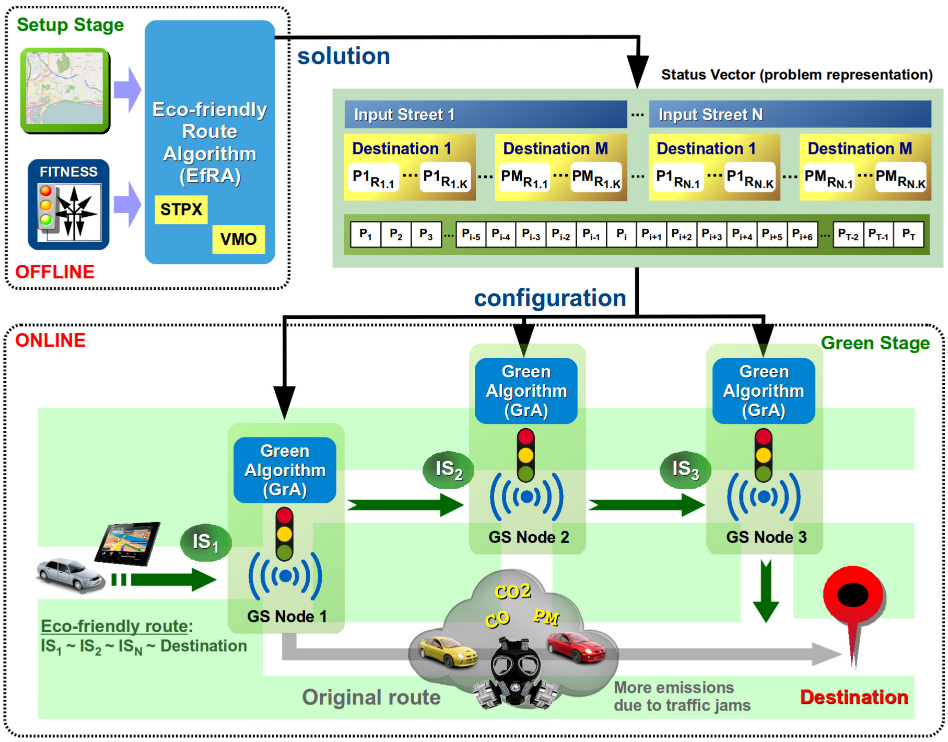


Figure 1: Green Swarm Architecture. In the *Setup Stage* the configuration of the GS nodes is calculated using the EfRA and then, in the *Green Stage*, vehicles are rerouted by the GrA to prevent traffic jams.

In the *Green Stage*, vehicles connect to the GS nodes as they pass by, which triggers the execution of the GrA. Then, GrA suggests a new route for the vehicles according to the configuration calculated by EfRA in the previous stage. These new routes are customized as they are determined by the final destination of each vehicle.

Each node is implemented using a Wi-Fi spot connected to a processing unit capable of running the GrA. Additionally, they can be remotely updated (via the mobile network or the already existing connectivity found in traffic lights) to change the GS configuration in the case of possibly closed streets, events, etc. The software running in the mobile devices consists of a navigator-like screen with a graphic user interface for entering the driver's destination. Finally, the communication between a device and a node implies the former sending the desired destination and the latter answering with the route to the next GS node or to the driver's destination. According to [15] we estimate an operational radius for each node of, on average, 77 meters.

The placement of the nodes has been manually set for this study as it represents a challenge in itself which needs and justifies a future, separate, scientific article. The main goal is: given a set of the more congested junctions controlled by a traffic light, identify those which better improve the rerouting of vehicles, preventing traffic jams, and use them as GS nodes.

An example of the rerouting performed by GS during the *Green Stage* is shown in Figure 1. When the vehicle connects to *Node 1* via a Wi-Fi link, the GrA suggests a new route toward *Node 2*, potentially different from the original one. It is assumed here that the driver accepts the new route, so that when he/she approaches *Node 2* by Input Street IS_2, the vehicle will be routed to *Node 3*. Finally, in *Node 3*, which is near the vehicle's destination, the driver will be sent directly (no intermediate node) to the end of his/her journey.

By using GS the vehicle has probably traveled a longer distance than when following the shortest path (which is usually the default choice made by drivers), but it has avoided possible traffic jams while driving in an eco-friendly way. As a result, the amount of gas emitted into the atmosphere and travel times have both been reduced. Even if this seems not to be an intuitive result, it is demonstrated that taking into consideration the global flow and driving events, leads to a greener trip in the end. In order to evaluate each case study, the traffic simulator SUMO [9] has been used. SUMO implements realistic car following models and it can be externally controlled by TraCI [16] to perform the reroutings suggested by the GrA.

3.1. Eco-friendly Route Algorithm (EfRA)

What is being attempted in this article is finding a solution to a very difficult real problem requiring high evaluation times and managing large vectors of numbers which encompass a huge search space, very hard to explore by exhaustive methods. Furthermore, there is no analytic equation, so traditional methods are not viable. In addition, low complexity operations as used in metaheuristics are needed. All these reasons make this problem suitable for solving with a bio-inspired algorithm [17]. Concretely, we have designed a new evolutionary algorithm, based on a (10+2)-EA [18] and called Eco-friendly Route Algorithm (EfRA).

As stated in [19] "Evolutionary algorithms mimic the process of natural evolution, the driving process for the emergence of complex and well-adapted organic structures". EfRA demonstrates these characteristics, working with a population of individuals which evolve through natural selection for reproduction, being crossed with each other to produce offspring which suffer from mutation. Finally, the next generation is formed by the fittest individuals which have survived. The EfRA was used to obtain a configuration for the GS nodes during the *Setup Stage*, which minimizes the gas emissions of vehicles in each case study.

Algorithm 1 Eco-friendly Route Algorithm.

procedure EfRA ▷ $O_{EfRA} = O(n^2)$
    $t \leftarrow 0$
    $P(0) \leftarrow createPopulation()$ ▷ P = population
    while not $terminationCondition()$ **do**
        $Q(0) \leftarrow \emptyset$ ▷ Q = auxiliary population
        $parents \leftarrow selection(P(t))$
        $offspring \leftarrow STPX(Pc, parents)$
        $offspring \leftarrow VMO(\pi_1, \pi_2, \theta, offspring)$
        $evaluateFitness(offspring)$
        $insert(offspring, Q(t))$
        $P(t+1) \leftarrow replace(Q(t), P(t))$
        $t \leftarrow t + 1$
    end while
end procedure

Moreover, an elitist steady state EA has been chosen, with a population of ten individuals, generating two new individuals at each step, mainly because the evaluation of each individual requires a simulation which takes more than 30 seconds to complete. EfRA is a light-weight algorithm (compared to other metaheuristics like common EAs, PSOs, etc.), it performs well without the need for an analytic equation which is impossible in this domain.

First, in EfRA (Algorithm 1), the number of steps t is set to zero and the population $P(0)$ (10 individuals) is initialized with random values. Then, while the termination condition is not fulfilled the main loop is executed. In our experiments EfRA ends when the maximum number of steps (5000) or the convergence criterion (500 generations without improvements) are reached.

Inside the main loop, after initializing the auxiliary population $Q(0)$, two parents are selected from the population by using binary tournament [20]. Next, the offspring (two individuals) are obtained after applying the recombination operator (STPX) [14] and after that, the offspring are mutated by applying our Variable Mutation Operator (VMO) [14], both described later. Then, the new individuals are evaluated and inserted in the auxiliary population $Q(t)$.

Finally, the new population $P(t + 1)$ is generated by replacing the current one ($P(t)$) with the individuals of the auxiliary one ($Q(t)$) in an elitist way, that is, the worst individuals in $P(t)$ (highest fitness values) are replaced by the individuals in $Q(t)$ if and only if the new ones have better (lower) fitness values and they are not yet in the population (no duplicates allowed). Note that the complexity of EfRA is $O(n^2)$ as there are two nested loops in its body.

3.1.1. Representation

The goal is to suggest routes to vehicles as they are approaching a junction controlled by a GS node, so the different probabilities for each route need to be stored in a configuration vector. These probabilities are computed by an intelligent automatic technique according to the layout and dynamic features of the traffic in the city: our EfRA. Additionally, the suggested routes have to be personalized for each driver depending on his/her destination. For this reason, the route probabilities have to be separated into groups assigned to each destination.

The problem representation chosen is shown in the middle of Figure 1 where it can be seen blocks of routes starting in the same street (Input Street 1, Input Street 2, etc.) which are inputs to a junction controlled by a GS node. These input streets are the points where the rerouting takes place (providing that the driver takes into account the suggestion given). Then, the available routes are replicated in several destination chunks in the same street block so as to personalize the trip based on drivers' destinations.

Finally, each route has a probability value associated with it, to define how likely it is to be assigned. Note that the summation of probability values in the same chunk must be equal to 1.

3.1.2. Evaluation Function

From our experimentation, explained in Section 6.1, several relationships have been observed between the metrics, which has led to only CO_2 being included in the evaluation function to calculate the fitness value of the individuals.

The fitness function for EfRA is presented in Equation 1 where two terms can be seen. The first is meant to penalize the individuals representing configurations for the GS nodes which are unable to route all vehicles to their destination within the analysis time. Therefore, N is the total number of vehicles and n is the number of vehicles which have completed their itineraries, so we penalize the resulting fitness value with the number of vehicles which are inside the area under analysis when the analysis time ends.

The second term of Equation 1 represents the average CO_2 emissions from vehicles. It is normalized by the α coefficient calculated as shown in Equation 2. There, λ represents the number of training scenarios (four in this article), and n_i is the number of vehicles in the training scenario i. By using α in Equation 1 the fitness function is normalized, so that the experts' solution we are hoping to improve has a fitness value equal to 1. As the idea is to minimize, values below 1 represent an improvement over the experts' solution, i.e. the lower, the better.

$$F = (N - n) + \alpha^{-1}\frac{1}{n}\sum_{i=1}^{n} CO_{2_i} \qquad (1)$$

$$\alpha = \frac{1}{\lambda}\sum_{i=1}^{\lambda}\frac{1}{n_i}\sum_{j=1}^{n_i} CO_{2_{ij}} \qquad (2)$$

3.1.3. Evolutionary Operators

Some of the operators tested in our previous experimentation [14] have again been used in this new problem. Binary Tournament is used as the selection operator; Street Two Point Crossover (STPX) as the recombination operator, where the cross points are blocks of input streets' configurations; Variable Mutation Operator (VMO) where two different mutation probability values are combined using a threshold value to switch between them (the first value is meant to explore the search space whilst the second is to refine the solution by exploitation); and Elitism in the replacement operator.

The STPX selects two input street blocks as crossover points and swaps their probability values. Figure 2 shows an example of STPX where the probability values in the blocks i to j are swapped. The VMO randomly selects one Input Street and then changes the probabilities in its destination chunks according to the mutation probabilities π_1 and π_2. Initially, π_1 is used until the fitness of the best individual is under the threshold θ. If this happens, then π_2 is used instead. Figure 3 presents an example of VMO where the Input Street i is selected and then the destination chunks 1, 4, and 5 have their probabilities changed.

The recombination probability value used is 0.6, the threshold $\theta = 1.0$, and the mutation probabilities are $\pi_1 = 0.04$ and $\pi_2 = 0.01$. Table 1 shows a summary of the parameters of EfRA.

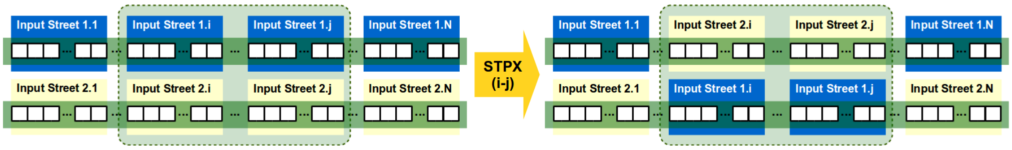


Figure 2: Street Two Point Crossover (STPX).

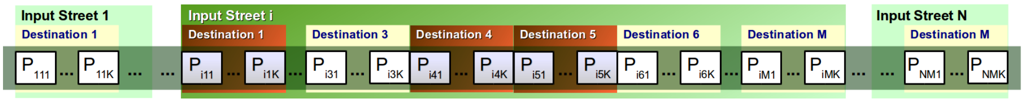


Figure 3: Variable Mutation Operator (VMO).

Table 1: Parameters of the EfRA.

Parameter	Value
Maximum iterations	5000
Crossover probability (P_C)	0.6
Mutation probabilities (π_1, π_2)	0.04, 0.01
Threshold (θ)	1.0

3.2. *Green Algorithm (GrA)*

Our GrA runs in each GS node. When a vehicle connects with a node via Wi-Fi, GrA reads the configuration previously calculated by EfRA for this node and suggests an alternative route based on the probability values and the vehicle's destination (*Green Stage*). Even though the GrA cannot guarantee that each vehicle will reach its destination (as each spot is only responsible for a section of the whole route), the evolution of the configurations in EfRA toward an optimum makes it highly likely that each vehicle will reach its final destination.

The pseudocode of GrA is presented in Algorithm 2. First, the current street and the vehicle's

Algorithm 2 Green Algorithm.

procedure GrA(vehicle) ▷ $O_{GrA} = O(n)$
  $current\_street \leftarrow getStreet(vehicle)$
  $destination \leftarrow getDestinationZone(vehicle)$
  if $current\_street \in destination$ **then**
    $route \leftarrow getCurrentRoute(vehicle)$
  else
    $route \leftarrow getRouteToDestination(current\_street, destination)$
    if $route = \emptyset$ **then**
      $nextInputStreet \leftarrow getStreetByProbability(current\_street, destination)$
      $route \leftarrow getRouteToInputStreet(nextInputStreet)$
    end if
  end if
  $suggestNewRoute(route, vehicle)$
end procedure

destination zone are obtained from the approaching vehicle itself. Second, the destination zone is checked to avoid rerouting a vehicle already in it. If the vehicle has not yet reached its destination zone, all the routes from the current street to the vehicle's destination zone are considered in the GS configuration. If the destination zone is not directly reachable from the current street ($route = \emptyset$), the algorithm obtains the next Input Street (belonging to another node) which is directly reachable from the vehicle's current street so that it is rerouted to another GS node. This Input Street is selected based on the probabilities stored in the GS configuration. Finally, the route from the current to the next Input Street is suggested to the vehicle in the last step.

Note that the complexity of GrA is $O(n)$ as it just retrieves the routes previously calculated by EfRA in the *Setup Stage*.

4. Competitor Techniques

Although comparing a contribution to existing competitors is a must in science, research papers in this area frequently do not consider competitor systems. The reason is not only the difficulty of finding closely similar work, but also that it is very difficult to find and manage studies reporting so many technological tools, open data and algorithms. Notwithstanding, an effort has been made on the part of the authors to include several competitors in this article to compare this proposal with others.

Consequently, three different strategies presented in [3] in order to reduce local traffic emissions have been chosen: i) reducing traffic demand by 20% (-20%), ii) introducing a speed limit of 30 km/h (30km/h), and iii) replacing heavy duty vehicles with 1.5 light duty vehicles (HDV-LDV). These strategies may seem at first glance to be trivial as they are not based on optimization. However, they are widely applied by local councils, particularly when the pollution levels are so high that people's health is put at risk.

The authors of the aforementioned article tested these strategies in a single intersection located in Bentinckplein in the city of Rotterdam, the Netherlands. Although they achieved reductions in emissions of between 13% and 30% depending on the strategy and metrics used, they analyzed only one intersection instead of large districts. This encouraged us to test those strategies in our case studies as an additional contribution. The modifications applied to the traffic demand implemented by each strategy are described as follows:

- Minus 20% (-20%): A reduction in the number of vehicles of 20% is implemented, while keeping the original proportion of vehicle types (Table 2). The result was 3282 vehicles in ALA, 3760 in MGA, 3680 in STO, 4640 in BER, and 4560 in PAR.

- Maximum 30km/h (30km/h): In this strategy the number and types of vehicles are the same as in the expert's solution. However, the maximum speed has been restricted to 30km/h for all of them.

- HDV-LDV: This strategy consists in replacing trucks, which have the worst emission class of all the vehicle types, with 1.5 light duty vehicles (sedan, van, and wagon). This represents an increase in demand of approximately 5% so that the number of vehicles in the case studies is 4308 in ALA, 4930 in MGA, 4830 in STO, 6086 in BER, and 5985 in PAR.

5. Case Studies

For this approach four large, important European cities: Malaga (Spain), Stockholm (Sweden), Berlin (Germany), and Paris (France) were chosen. This enabled the study of specific zones which are prone to traffic jams, with the aim of improving traffic flow and reducing gas emissions. Furthermore, a reduced area of Malaga (Alameda Principal) was also studied, where real traffic conditions could be faithfully recreated and the accuracy of the study, improved.

First, the GS system was applied to a small case study ($0.4Km^2$) comprising the *Alameda Principal* area of Malaga (Spain). In this case study data published by the local council for peak time traffic at 2 p.m. on working days [21] was used and its real traffic flows were generated by using the method described in [22]. Second, four new and larger geographical areas were used, representing zones not only in the city of Malaga, but also in three major European cities: Stockholm, Berlin, and Paris which are all depicted in Figure 4.

In spite of the fact that the real number of vehicles in the larger areas was unavailable, they were included to test our proposal against different cities, urban maps, and traffic distributions. By doing so, a real case study to validate GS was addressed, and then a variety of new case studies were analyzed using a distinct number of vehicles and flows (generalization and robustness).

To build each new case, the geographical areas in OpenStreetMap [8] were first selected and then exported to individual map files (.osm files). The maps were then modified by using the application JOSM (Java Open Street Map) thereby removing unhelpful, irrelevant data such as parks, housing blocks, and pedestrian walkways. Based on these, the working maps for SUMO were generated by NETCONVERT.

Finally, the traffic flows between the streets were defined using the DUAROUTE utility and used as the inputs to the areas being analyzed (source streets) and the streets which are destinations. Each flow contained several routes representing different, alternative paths between the same source and destination pairs. These were obtained by using the different weight metrics available in DUAROUTE such as travel times, emissions, and fuel consumption. By using these flows the difficulty of the problem being addressed increased, as vehicles do not always take the same routes toward their destination.

Other different case studies where vehicles are actually taking the fastest routes (calculated by minimizing travel times) in the cities under consideration have also been included. This was done so as to also address a more realistic problem (people usually drive along avenues).

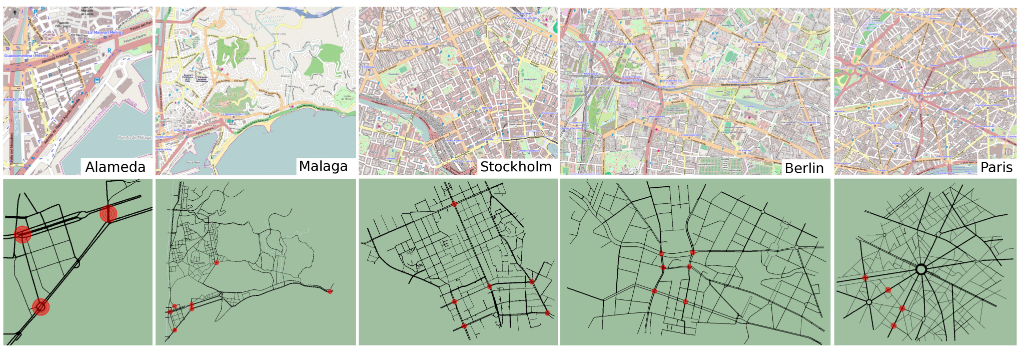


Figure 4: Case studies: *Alameda* (ALA), *Malaga* (MGA), *Stockholm* (STO), *Berlin* (BER), and *Paris* (PAR), imported from OpenStreetMap (upper row) into the SUMO traffic simulator (bottom row).

The traffic light cycles were assigned by NETCONVERT while generating the map, using the algorithms included in SUMO. However, some corrections, especially in the lights' synchronization, were made to avoid problems of misconfigured cycles. Note that both NETCONVERT and DUAROUTE are tools included in the SUMO software package.

Wishing to provide a more realistic study, four different vehicle types were used (Table 2) having different emission classes from the HBEFA [23] model, as it would not make sense to have sedans and trucks emitting the same amount of gas nor consuming the same liters per kilometer.

Table 2: Characteristics of the four types of vehicles.

Type	Arrival probability	MaxSpd. (Km/h)	Accel. (m/s^2)	Decel. (m/s^2)	Length (m)	Emission class
sedan	0.50	160	0.9	5.0	3.8	P_7_7
van	0.25	100	0.8	4.5	4.2	P_7_5
wagon	0.15	50	0.7	4.0	4.3	P_7_6
truck	0.10	40	0.6	3.5	4.5	HDV_3_1

In each working scenario, vehicles arrive at different times, through different streets and taking different routes, which generates a variety of situations to train and test the proposal. Since the assigned vehicles' type and route depend on the random number generator included in SUMO, by changing the simulation seed the different scenarios were defined for each case study. These mobility solutions based on traffic distributions are called the experts' solution as they were generated by the SUMO tools. The characteristics of the case studies are presented in Table 3. All of them were analyzed for one hour, while the rest of the characteristics were dependent on the road distribution obtained from OpenStreetMap as well as the size of the geographical area.

Table 3: Characteristics of the case studies: *Alameda* (ALA), *Malaga* (MGA), *Stockholm* (STO), *Berlin* (BER), and *Paris* (PAR). Note that the number of probability values in the solution vector denotes the complexity of each case study.

Case study	*ALA*	*MGA*	*STO*	*BER*	*PAR*
# Vehicles	4104	4700	4600	5800	5700
# Traffic lights	28	89	75	76	58
# GS nodes	3	7	6	6	4
# Vehicle flows	4	25	14	16	15
# Vehicle routes	15	430	196	229	210
Studied area (Km^2)	0.4	10.0	2.9	7.0	5.6
# Probability values	168	840	1314	450	732

6. Experimentation

First, several experiments were conducted to determine which metrics were best for inclusion in the evaluation function (Section 6.1). Second, the optimization of one case study (ALA) was addressed, consisting of real traffic flows. Then, four other case studies were optimized, where vehicles used various, different routes between their origin and destination (Section 6.2).

At this point, our proposal was compared with different, state of the art strategies where the behavior of GS when it is used after applying the other strategies was evaluated. This allowed us to know if they were compatible and if the metrics could be reduced even more (Section 6.3).

The best configuration obtained for GS in the previous experiments was tested in 500 unseen scenarios where vehicles either followed a number of different routes (more difficult to optimize) or just the fastest ones (a situation closer to reality). The other strategies were also included at this point and a combination of them were tested with GS in 1500 scenarios (Section 6.4).

Finally, a study was done to analyze how GS behaves when only a certain percentage of people are using it (Section 6.5), followed by a discussion on the GrA performance (Section 6.6).

6.1. Metric Study

For the present article, an initial series of tests were conducted to evaluate which emission metrics were more suitable for optimization. We exploited the case study called *Alameda* (ALA) because its size makes the analysis more affordable (in time) than the rest of the bigger scenarios.

Four hundred and twenty runs were carried out, which lasted 31.2 hours on average. From the results it could be observed that EfRA was able to reach the same optimal configuration in each optimization process when using different metrics in the evaluation function (CO_2, $Fuel$, $CO_2 + Fuel$, $CO + CO_2$, $CO + HC$, $PM + HC$, and $CO + CO_2 + NO$). Based on these results, CO_2 was chosen as the metric to be optimized, because not only is it a well-known gas causing global warming, but also because it keeps the evaluation function simple.

Figure 5 presents the graphs of the different metrics vs. CO_2 from 16416 vehicles (4 scenarios of *Alameda*) in order to visualize and confirm the similarities between them. The majority of the graphs show different slopes which correspond to the different emission classes of vehicles. Some of them are mostly coincident, especially in the case of the $Fuel$ consumption, where its linear relation with CO_2 is evident. This fact supports even further the decision made in respect to the variable (CO_2) evaluated to calculate the fitness value of our scenarios, as the rest of the metrics are reduced when reducing the CO_2 emissions.

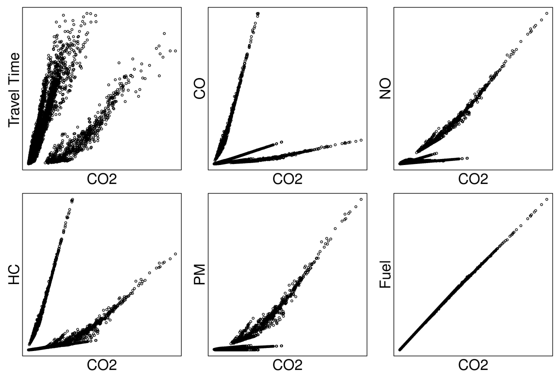


Figure 5: Similarities between CO_2 and the rest of the metrics. Different slopes correspond to the different emission classes of vehicles. Note that $Fuel$ consumption presents a linear relation with CO_2.

6.2. Optimization

In this section four training scenarios are optimized for each one of the five case studies. Each scenario presents different traffic distributions to EfRA so that the optimization processes can produce robuster solutions [14]. Thirty independent runs of EfRA were carried out to optimize each case study (150 runs) and the results are presented in Table 4 (GS strategy).

GS achieves improvements in all the metrics and in all the cities. The results are especially interesting in ALA, where there exists a real traffic challenge with a large number of vehicles in a reduced area. There, GS has shortened travel times by 70%, reduced CO emissions by 57%, CO_2 by 37%, and fuel consumption by 36% on average. In MAL, GS achieves 19% shorter travel times and a reduction in CO of 11% . Moreover, it can be seen that in STO there are important reductions in travel times (42%) and emissions (32% in CO and 29% in HC) when using GS. Vehicles driving through BER benefit from 19% shorter travel times when using GS and emit 13% less CO and 11% less HC in the atmosphere, on average. Finally, the best, improved metrics in PAR are travel times (10%), CO (8%), and HC (7%).

Table 4: Improvements in the experts' solution achieved by the strategies used to optimize our five case studies. Note that these results correspond to the scenarios used during the optimization. The best performing strategies are in bold.

Metric	Strategy	*ALA*	*MGA*	*STO*	*BER*	*PAR*
T.Time	GS	**69.7%**	18.7%	**41.7%**	19.0%	10.2%
	-20%	15.8%	**25.0%**	33.0%	**33.0%**	**37.8%**
	30Km/h	-5.0%	-12.5%	-10.3%	-12.8%	-22.7%
	HDV-LDV	-4.8%	0.7%	-4.1%	-2.8%	-5.1%
CO	GS	**56.7%**	10.6%	**31.8%**	12.8%	7.9%
	-20%	11.3%	15.3%	25.0%	**23.9%**	**23.6%**
	30Km/h	4.6%	**17.6%**	5.9%	10.2%	8.0%
	HDV-LDV	-14.3%	-6.2%	-15.8%	-7.9%	-13.7%
CO_2	GS	**36.6%**	5.3%	15.1%	5.2%	3.6%
	-20%	7.8%	7.8%	13.4%	12.4%	12.5%
	30Km/h	6.0%	10.2%	6.4%	7.2%	9.3%
	HDV-LDV	25.8%	**35.7%**	**30.6%**	**31.7%**	**31.8%**
HC	GS	**54.3%**	9.4%	**29.3%**	10.8%	7.3%
	-20%	10.3%	**13.5%**	23.2%	**21.2%**	**22.0%**
	30Km/h	0.5%	4.2%	0.1%	0.3%	-3.6%
	HDV-LDV	1.9%	1.1%	-1.5%	2.6%	-2.4%
PM	GS	47.6%	8.0%	24.6%	8.7%	5.7%
	-20%	8.6%	10.5%	20.6%	18.0%	18.2%
	30Km/h	2.1%	8.1%	4.2%	4.4%	3.6%
	HDV-LDV	**75.9%**	**68.2%**	**74.2%**	**70.0%**	**69.4%**
NO	GS	35.0%	5.4%	15.4%	4.8%	3.8%
	-20%	6.3%	7.3%	13.4%	11.8%	12.1%
	30Km/h	5.4%	10.5%	7.1%	7.6%	9.1%
	HDV-LDV	**66.7%**	**63.5%**	**65.6%**	**63.2%**	**63.2%**
Fuel	GS	**36.3%**	5.2%	14.8%	5.1%	3.6%
	-20%	7.8%	7.7%	13.2%	12.2%	12.4%
	30Km/h	6.1%	10.2%	6.5%	7.3%	9.5%
	HDV-LDV	25.3%	**35.6%**	**30.3%**	**31.5%**	**31.7%**

As a consequence of the rerouting strategy, some drivers have individually experienced longer travel times. Concretely, 25% of drivers have longer travel times in ALA, 38% in MAL, 39% in STO, and 47% in BER and PAR. This is a low price to pay for achieving global reductions of travel times and gas emissions in the city, especially if we take into account that it is not likely that the same drivers are penalized every day.

Next, the three competitor strategies were implemented as described in Section 4. Then, they were applied to our case studies (again the same four scenarios of each) to obtain improvements in each metric, also presented in Table 4. As can be seen, the improvements vary notably among the metrics and scenarios, which makes it difficult to conclude which strategy is the best one. Nevertheless, looking at the different strategies it can be appreciated that a reduction in the number of vehicles (-20%) has a positive impact on travel times as there are fewer vehicles on the streets of our case studies. Reducing the number of vehicles has worked well, especially in the reduction of CO, HC, and PM emissions. This strategy seems to achieve similar results to GS: the former reduces the number of vehicles directly while the latter reroutes them via alternative streets without restricting the drivers.

Fixing the maximum speed at 30Km/h has turned out to be the least effective measure to reduce emissions, demonstrating the worst travel times as well. All in all, the reduction of emissions is quite low in most of the cases, except for the case study MAL. Paradoxically, this is the method applied by the majority of city authorities when the pollution levels are high. Our conclusions in this matter are in keeping with those discussed in [24] where the authors illustrate the scientific uncertainties inherent in implementing speed management policies [25].

Replacing trucks with sedans, vans, and wagons (HDV-LDV) enables a huge reduction of CO_2, PM, and NO emissions, as they are the main gases emitted by trucks according to the HBEFA class selected for this type of vehicle (HDV_3_1). Furthermore, this strategy also reduces fuel consumption which is directly related to CO_2 emissions as we have stated in Section 6.1. The HDV-LDV strategy is a serious competitor to our system (emissions), but it definitely has a negative impact on the economy of the city, as it is difficult to implement, and will definitely incur protests. Our system is smoother and simultaneously more efficient with shorter travel times whereas the other strategies show longer ones (negative improvements).

Our conclusion after this study is that despite the fact that some competitor strategies perform better in some case studies, GS has competitive results. We must keep in mind that we do not restrict the number, type, or speed of vehicles which would not be desirable or even viable in many cities. Therefore, the next step taken was to optimize the same four training scenarios after applying the competitor strategies to know how GS behaves under these new conditions.

6.3. Green Swarm Combined with Other Strategies

In this section the combination of GS with other strategies is studied to discover not only if they are viable but also if the strategies achieve better results when are applied together.

We took the traffic distributions obtained when the -20%, 30Km/h, and HDV-LDV strategies were applied in our training scenarios and applied GS as the optimization algorithm in order to analyze how they combine with each other and see if some metrics could be improved even further. After performing a further 30 independent runs of the EfRA in four scenarios of our five case studies (150 runs per strategy) GS demonstrated the relative improvements over the other strategies shown in Table 5. At first glance, the best improvements are made when applying GS after limiting the vehicles' maximum speed (30Km/h+GS). However, the most important conclusion here is that all the metrics have been improved by complementing the competitors with GS which, in our opinion, validates our proposal as a promising solution for improving the city's streets reducing travel times, greenhouse gas emissions and fuel consumption. Focusing on the numbers, the maximum improvements are nearly 50% in travel times, 45% in CO, 30% in CO_2, 41% in HC, 38% in PM, 30% in NO, and 30% in fuel consumption.

The total number of runs performed in the optimization processes was 600 and the time spent on each of them was, on average, between 19 and 92 hours. The diversity of values depends on the case study, the number of vehicles and the heterogeneity of the hardware used.

Table 5: Relative improvements achieved by using GS after the other competitor strategies. These results correspond to the training scenarios used in the initial optimization. The best improvement in each metric and case study is in bold.

Metric	Strategy	*ALA*	*MGA*	*STO*	*BER*	*PAR*
$T.Time$	-20%+GS	43.1%	6.4%	31.9%	12.8%	7.4%
	30Km/h+GS	49.1%	15.1%	37.7%	14.1%	7.0%
	HDV-LDV+GS	**49.8%**	**18.8%**	**42.3%**	**16.8%**	**10.3%**
CO	-20%+GS	33.4%	2.0%	20.6%	6.0%	4.7%
	30Km/h+GS	**45.4%**	**11.9%**	**33.9%**	**11.5%**	6.9%
	HDV-LDV+GS	40.8%	10.6%	33.7%	9.6%	**8.9%**
CO_2	-20%+GS	20.0%	1.9%	8.2%	2.1%	1.8%
	30Km/h+GS	**30.4%**	**5.9%**	14.6%	3.5%	3.3%
	HDV-LDV+GS	**30.4%**	**5.9%**	**19.3%**	**4.1%**	**4.6%**
HC	-20%+GS	32.4%	1.8%	18.2%	4.4%	4.3%
	30Km/h+GS	**40.9%**	**8.4%**	**27.7%**	**8.6%**	5.9%
	HDV-LDV+GS	36.6%	8.2%	27.3%	5.6%	**7.2%**
PM	-20%+GS	28.0%	2.9%	13.5%	4.8%	3.3%
	30Km/h+GS	**38.1%**	**9.0%**	**23.8%**	**7.3%**	**5.9%**
	HDV-LDV+GS	20.1%	3.3%	10.0%	1.1%	2.4%
NO	-20%+GS	19.5%	2.1%	7.8%	2.5%	2.0%
	30Km/h+GS	**29.6%**	**6.3%**	**14.2%**	**3.3%**	**3.8%**
	HDV-LDV+GS	18.3%	2.9%	9.8%	0.6%	2.5%
$Fuel$	-20%+GS	19.8%	1.9%	8.0%	2.1%	1.8%
	30Km/h+GS	30.2%	5.8%	14.4%	3.4%	3.2%
	HDV-LDV+GS	**30.4%**	**5.9%**	**19.3%**	**4.1%**	**4.6%**

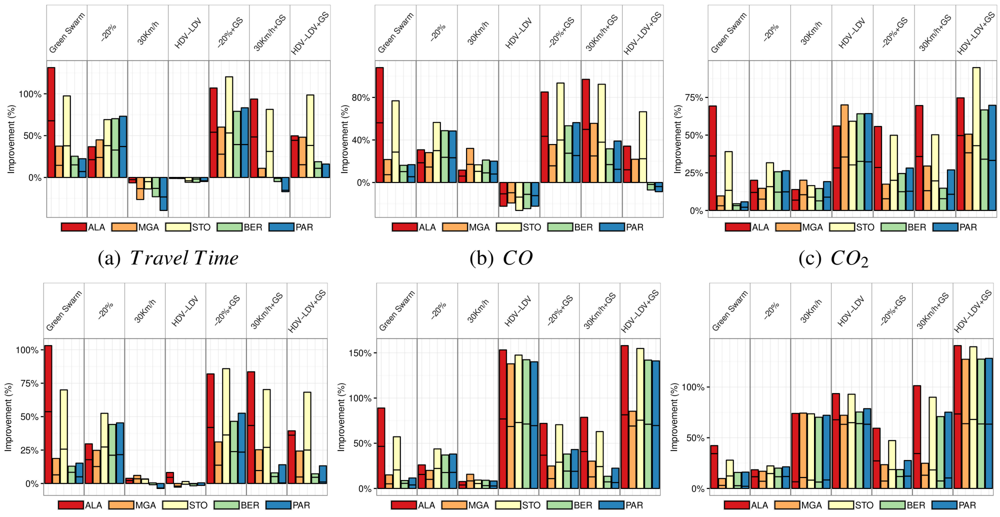
6.4. Study on Unseen Scenarios

After the aforementioned optimization processes we wanted to test GS in unseen scenarios. With this in mind, 50 new unseen scenarios were generated for each city, where the vehicles followed a variety of routes to their destination and another 50 in which they just flowed via the fastest routes (subscript $_{TT}$ which stands for travel time). Then, the seven optimization strategies on these scenarios (700 in total) were tested. The results are shown in Table 6 where the average improvements achieved by each strategy in each case study and metric are displayed. The GS configurations previously obtained were used here, so no extra optimization process was needed.

It can be seen in Table 6 that GS has improved the other strategies in this study as well, even turning some of their results that were worse than the experts' solution into actual improvements.

Table 6: Average improvement achieved by applying the seven strategies analyzed to 50 unseen scenarios of each case study (500 scenarios in total) during the *Green Stage*. The best performing strategies in each case study are in bold.

Metric	Strategy	ALA	MGA	STO	BER	PAR	ALA_{TT}	MGA_{TT}	STO_{TT}	BER_{TT}	PAR_{TT}	Average
$T.Time$	GS	**67.8%**	14.5%	37.8%	15.0%	7.1%	**63.5%**	23.0%	59.6%	10.3%	15.3%	31.4%
	-20%	21.3%	23.9%	38.1%	32.8%	37.0%	15.5%	21.0%	31.1%	37.4%	36.1%	29.4%
	30Km/h	-2.6%	-13.3%	-5.0%	-13.1%	-23.3%	-3.8%	-13.0%	-8.7%	-9.9%	-16.1%	-10.9%
	HDV-LDV	0.0%	-1.0%	-3.9%	-1.9%	-4.0%	-1.2%	-0.2%	-1.8%	-4.0%	-0.8%	-1.9%
	-20%+GS	54.1%	**27.8%**	**53.2%**	**39.4%**	**39.4%**	52.8%	32.4%	**67.0%**	**39.7%**	**43.9%**	**45.0%**
	30Km/h+GS	48.4%	0.5%	31.2%	-1.0%	-16.9%	45.3%	10.4%	50.2%	-3.9%	1.6%	16.6%
	HDV-LDV+GS	49.8%	15.3%	38.2%	11.0%	0.0%	-5.4%	**32.8%**	60.3%	7.8%	16.0%	22.6%
CO	GS	**56.1%**	7.4%	28.6%	10.3%	5.5%	**51.9%**	14.2%	48.2%	5.9%	11.3%	23.9%
	-20%	18.6%	14.5%	30.0%	23.7%	23.3%	12.2%	13.6%	26.5%	25.0%	25.2%	21.3%
	30Km/h	6.0%	17.1%	10.5%	9.1%	8.0%	5.6%	15.0%	6.2%	12.0%	12.1%	10.2%
	HDV-LDV	-10.7%	-9.6%	-13.8%	-11.2%	-12.4%	-11.7%	-9.6%	-12.5%	-13.4%	-9.9%	-11.5%
	-20%+GS	43.4%	15.7%	**40.0%**	**27.5%**	**25.3%**	41.7%	20.0%	53.5%	**25.8%**	**31.0%**	**32.4%**
	30Km/h+GS	49.9%	**25.0%**	37.9%	16.9%	12.4%	47.1%	**30.6%**	**54.5%**	14.8%	26.6%	31.6%
	HDV-LDV+GS	34.1%	0.1%	22.4%	-1.8%	-8.6%	-22.2%	21.8%	44.1%	-5.2%	4.8%	8.9%
CO_2	GS	36.2%	3.1%	13.3%	3.2%	2.1%	33.1%	6.5%	25.7%	1.2%	3.6%	12.8%
	-20%	12.0%	7.5%	15.8%	12.2%	12.4%	8.1%	7.2%	15.8%	13.5%	14.0%	11.8%
	30Km/h	6.9%	10.5%	8.8%	6.4%	8.9%	7.0%	9.6%	7.7%	8.2%	10.3%	8.4%
	HDV-LDV	28.2%	35.4%	30.2%	32.6%	32.3%	27.9%	**34.6%**	28.9%	31.6%	32.0%	31.4%
	-20%+GS	28.5%	7.8%	20.0%	12.5%	12.7%	27.2%	9.7%	29.9%	12.0%	15.4%	17.6%
	30Km/h+GS	35.8%	13.1%	19.6%	7.8%	10.7%	**33.8%**	16.4%	30.6%	6.9%	16.2%	19.1%
	HDV-LDV+GS	**49.7%**	**38.3%**	**43.0%**	**34.0%**	**33.2%**	25.0%	12.4%	**51.8%**	**32.7%**	**36.6%**	**35.7%**
HC	GS	**53.7%**	6.5%	25.7%	8.4%	5.0%	**49.4%**	12.3%	44.3%	4.6%	10.2%	22.0%
	-20%	17.8%	12.7%	27.3%	21.2%	21.6%	11.9%	12.1%	25.1%	**23.0%**	23.8%	19.6%
	30Km/h	2.2%	3.5%	3.4%	-0.9%	-3.7%	1.7%	2.5%	-0.2%	1.3%	0.3%	1.0%
	HDV-LDV	4.6%	-2.0%	-0.5%	-0.2%	-1.4%	3.7%	-0.8%	2.0%	-1.5%	1.9%	0.6%
	-20%+GS	41.9%	**13.7%**	**36.3%**	**23.8%**	**23.5%**	40.0%	17.4%	**49.5%**	22.6%	**29.1%**	**29.8%**
	30Km/h+GS	43.4%	9.7%	27.0%	5.2%	0.5%	40.2%	15.5%	43.2%	2.8%	13.6%	20.1%
	HDV-LDV+GS	39.4%	4.9%	25.0%	4.7%	1.1%	-3.3%	**19.4%**	43.2%	2.6%	12.1%	14.9%
PM	GS	46.6%	5.2%	20.5%	5.7%	3.9%	42.4%	9.9%	36.7%	3.0%	7.7%	18.2%
	-20%	15.5%	10.1%	22.1%	17.6%	17.8%	10.6%	9.9%	21.6%	19.4%	20.3%	16.5%
	30Km/h	3.8%	8.4%	5.5%	3.1%	2.8%	3.8%	7.3%	3.8%	5.9%	5.5%	5.0%
	HDV-LDV	76.9%	68.5%	73.0%	**71.3%**	69.5%	76.4%	**69.4%**	74.5%	**71.1%**	70.6%	**72.1%**
	-20%+GS	36.8%	10.9%	29.1%	19.3%	19.0%	35.1%	13.9%	41.5%	18.8%	24.1%	24.9%
	30Km/h+GS	40.9%	12.9%	24.3%	7.3%	6.5%	37.8%	17.4%	38.6%	6.1%	16.0%	20.8%
	HDV-LDV+GS	**81.4%**	**69.1%**	**75.5%**	71.1%	**69.6%**	**76.5%**	16.3%	**79.3%**	70.9%	**71.4%**	68.1%
NO	GS	34.3%	3.0%	12.4%	2.7%	2.2%	31.1%	6.0%	24.0%	0.8%	3.5%	12.0%
	-20%	11.6%	7.1%	14.8%	11.6%	11.8%	7.9%	6.8%	15.5%	13.1%	14.0%	11.4%
	30Km/h	6.6%	10.8%	8.2%	6.4%	8.4%	6.9%	9.9%	7.5%	8.5%	9.6%	8.3%
	HDV-LDV	67.7%	63.2%	64.9%	**64.0%**	63.3%	**67.4%**	**63.5%**	65.4%	**63.9%**	63.8%	**64.7%**
	-20%+GS	27.2%	7.3%	18.6%	11.8%	12.1%	25.8%	9.0%	28.0%	11.4%	15.3%	16.7%
	30Km/h+GS	34.4%	13.1%	18.3%	7.4%	10.4%	32.2%	16.1%	28.6%	6.8%	15.4%	18.3%
	HDV-LDV+GS	**73.5%**	**63.9%**	**68.0%**	63.6%	**63.4%**	66.9%	11.8%	**71.6%**	63.4%	**64.8%**	61.1%
$Fuel$	GS	35.9%	3.1%	13.1%	3.2%	2.1%	32.8%	6.5%	25.3%	1.2%	3.5%	12.7%
	-20%	11.9%	7.5%	15.6%	12.0%	12.2%	8.0%	7.1%	15.7%	13.4%	13.8%	11.7%
	30Km/h	7.0%	10.5%	8.9%	6.4%	9.0%	7.1%	9.7%	7.8%	8.2%	10.4%	8.5%
	HDV-LDV	27.7%	35.4%	29.9%	32.4%	32.1%	27.4%	**34.5%**	28.5%	31.4%	31.8%	31.1%
	-20%+GS	28.3%	7.7%	19.8%	12.3%	12.6%	27.0%	9.6%	29.5%	11.9%	15.2%	17.4%
	30Km/h+GS	35.6%	13.1%	19.5%	7.8%	10.8%	**33.6%**	16.3%	30.3%	6.9%	16.1%	19.0%
	HDV-LDV+GS	**49.4%**	**38.2%**	**42.7%**	**33.8%**	**33.0%**	24.5%	12.3%	**51.5%**	**32.5%**	**36.3%**	**35.4%**



(a) *Travel Time* (b) CO (c) CO_2

(d) HC (e) PM (f) NO

Figure 6: Average improvement of the strategies applied to 500 unseen scenarios (50 per case study). Note that the case studies of the same city are stacked in the same bar, e.g. ALA and ALA_{TT}, MGA and MGA_{TT}, etc.

Most of the best performing strategies in each metric involve GS, either alone or applied after another strategy. The HDV-LDV strategy shows the best reductions of PM and NO on average, -20%+GS reduces the most CO, HC, and travel times on average, and HDV-LDV+GS achieves the biggest reductions in CO_2 and fuel consumption on average.

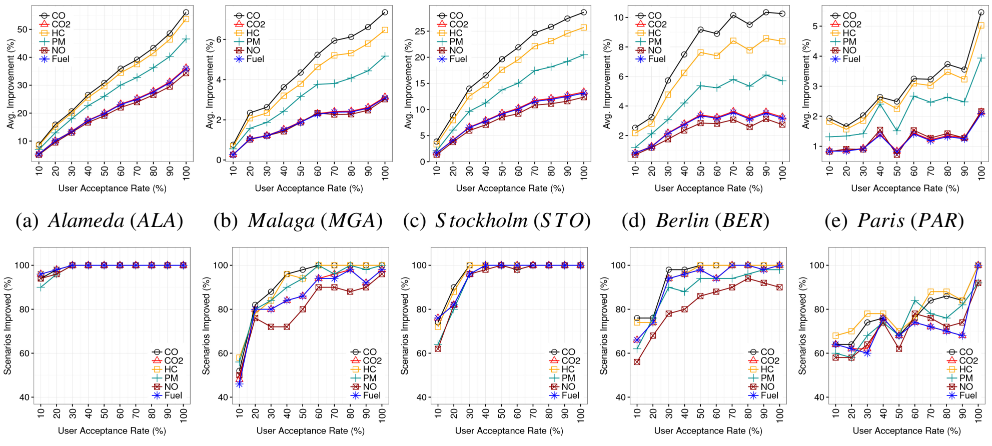
In Figure 6 a graphical comparison is given between strategies in each case study over six graphs for each metric. There, GS clearly performs especially well in our realistic congested case study (ALA) and it always presents a consistent improvement in all metrics. However, HDV-LDV and 30km/h encounter problems when improving travel times and reducing HC. HDV-LDV alone or combined with GS demonstrates the biggest reductions of CO_2, NO, and PM in most of the case studies according to our experiments. Finally, we have calculated the Wilcoxon p-value to be sure that the improvements reported on each metric are statistically significant. In all cases the p-value obtained was less than 0.01, that is, a confidence level greater than 99%.

6.5. Study of User Acceptance Rates

Since GS could be delivered as an app for smartphones, it is quite realistic to think that initially only a small number of drivers will have access to the system. Therefore, we have analyzed how the traffic behaves when just a subset of the vehicles use GS in our case studies.

In Figure 7 the graphs for the five case studies analyzed when the rates go from 10% to 100% in the best performing scenario are displayed. In the upper row, where the average improvement with respect to the experts' solution is plotted, it is clear that GS always reduces the average levels of gas emitted in each case study, even at low acceptance rates.

In the bottom row of Figure 7, the percentage of scenarios improved vs. GS acceptance rate is shown. The number of scenarios which are more eco-friendly when not all vehicles are using GS decreases, so that less use equals lower improvement, as one would expect. It is however noticeable that there is an average reduction in emissions in at least 48% of scenarios (the worst case: CO_2, *Malaga*), even when only 10% of drivers are using GS.



(a) *Alameda* (*ALA*) (b) *Malaga* (*MGA*) (c) *Stockholm* (*STO*) (d) *Berlin* (*BER*) (e) *Paris* (*PAR*)

(f) *Alameda* (*ALA*) (g) *Malaga* (*MGA*) (h) *Stockholm* (*STO*) (i) *Berlin* (*BER*) (j) *Paris* (*PAR*)

Figure 7: Graphs showing the average improvement achieved by GS for different user acceptance rates (upper row) and the percentage of the 50 scenarios improved (lower row) for the five case studies analyzed.

In addition, we observe that the behavior of GS in *Paris* has turned out to be a little different from the rest of the case studies. In figures 7(e) and 7(j) we can see that the metrics' variation for different usages is not as neat as in the rest of the case studies. This shows the different characteristics of *Paris*, especially its wide avenues and large roundabouts which leave little room for improvement. That being said, it is clear that there is also an improvement in each metric when GS is used, even for acceptance rates as small as 10% of all drivers in the city.

6.6. A Better Context for Understanding the Contributions of Green Swarm

In this section a study of the internal performance of Green Swarm is addressed. Concretely, a comparison of the EfRA with a state of the art Genetic Algorithm (GA) [26, 27] and Simulated Annealing (SA) [28, 29] is presented, followed by a convergence analysis.

The competitor GA implemented is a steady state ($\mu = 10, \lambda = 2$), using Binary Tournament as selection operator, Uniform Crossover as recombination operator ($P_C = 0.6$ as in EfRA), VMO with probability $1/L$ as the mutation operator, and an elitist replacement. The SA selected is a well-known metaheuristic applicable to a wide range of problems. In this comparison we have used $\alpha = 0.9$ and generated 50 random neighbors before each temperature decrement. Thirty independent runs of each algorithm were made, stopping after 2000 evaluations to make a fair comparison, which amounts to 284 equivalent days.

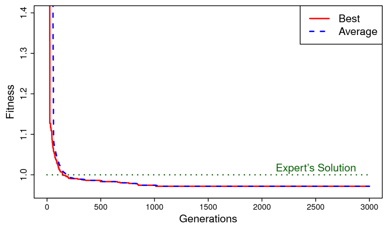
The objective of this study is to know how EfRA performs against its competitors and provide and internal statistical study [30] so as not to focus solely on the best fitness value. After testing the normality of the distributions using the Kolmogorov-Smirnoff test, we obtained p-values of 0.832 for the 30 runs of EfRA, 0.990 for GA, and 0.996 for SA. Consequently, non-parametric statistics (Friedman Rank and Wilcoxon) were used in the analysis. Table 7 shows the results of the comparison. EfRA achieved the best median value and was the best ranked algorithm. Additionally, the Wilcoxon test indicates that the differences between the results of the algorithms are statistically significant. We can therefore claim that our proposal overcomes existing results of the state of the art in the literature.

Moreover, a study on the EfRA fitness convergence over five independent runs (3000 generations, about 14 equivalent days) evaluating one instance of Malaga (MAL) was done. Figure 8(a)

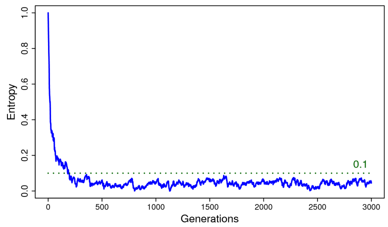
Table 7: EfRA compared with GA and SA.

Algorithm	Fitness		Friedman	Wilcoxon
	Median	Best	Rank	p-value
EfRA	**0.9625**	**0.9367**	**1.40**	—
GA	1.0145	0.9783	2.93	0.000
SA	0.9779	0.9441	1.67	0.032

shows that after the 180th generation, the experts' solution has been improved by our proposal. After that point the entropy, which had been falling until this moment, begins to fluctuate below 0.1 (meaning a very welcome exploration management of our algorithm) when the VMO changes the mutation probability from π_1 to π_2 to better exploit the solutions found (Figure 8(b)).



(a) Phenotype convergence of EfRA



(b) Genotype convergence of EfRA

Figure 8: Convergence of the EfRA over 3000 generations.

7. Conclusions and Future Work

In this article we have presented a system to reduce greenhouse gas emissions and used it to optimize road traffic in five cities in terms of not only emissions but also fuel consumption and travel times. Our conclusions can be summarized point-by-point as follows:

- Our proposal for a smarter mobility has performed notably well in almost all of the experiments conducted.
- GS has reduced average travel times (31% on average, 68% maximum), CO emissions (24% on average, 56% maximum), CO_2 (13% on average, 36% maximum), HC (22% on average, 54% maximum), PM (18% on average, 47% maximum), NO (12% on average, 34% maximum), and fuel consumption (13% on average, 36% maximum).
- There is a negligible increase in route lengths (2% on average) which is a consequence of the eco-friendly rerouting of vehicles via alternative streets which are not included in the shortest path (the needed trade-off between the individual and the community).
- GS has not only achieved competitive results when compared to the popular -20%, 30Km/h and HDV-LDV strategies, but has also worked perfectly as a complement to all of them, improving the metrics even further.
- In spite of the variations observed in the results, which was to be expected as we were considering different cities (cultures, locations, habits), we have consistently improved all the metrics, even when just 10% of vehicles were using GS.

As a matter for future work, we are working on different strategies to implement the rerouting of vehicles by using city districts as well as address the optimization of harder scenarios (computation time and hardware requirements) involving hundreds of thousands of vehicles. We are currently working on different strategies to address unforeseen events such as accidents, fires, public demonstrations, which could suddenly close streets, turning open routes into invalid ones.

8. Acknowledgments

This research is partially funded by the Spanish MINECO and FEDER projects TIN2014-57341-R, TIN2016-81766-REDT, and TIN2017-88213-R. Daniel H. Stolfi is supported by a grant (FPU13/00954) from the Spanish Ministry of Education, Culture and Sports.

References

[1] T. Bakici, E. Almirall, J. Wareham, A Smart City Initiative: the Case of Barcelona, Journal of the Knowledge Economy 4 (2) (2013) 135–148.

[2] TNS Opinion & Social, Attitudes of Europeans towards urban mobility, Tech. Rep. June (2013).

[3] M. Mahmod, B. van Arem, R. Pueboobpaphan, R. de Lange, Reducing local traffic emissions at urban intersection using ITS countermeasures, Intelligent Transport Systems, IET 7 (1) (2013) 78–86.

[4] H. Wahid, Q. P. Ha, H. Duc, M. Azzi, Neural network-based meta-modelling approach for estimating spatial distribution of air pollutant levels, Applied Soft Computing 13 (10) (2013) 4087–4096.

[5] D. Loomis, Y. Grosse, B. Lauby-Secretan, F. El Ghissassi, V. Bouvard, L. Benbrahim-Tallaa, N. Guha, R. Baan, H. Mattock, K. Straif, The carcinogenicity of outdoor air pollution, Lancet Oncol. 14 (13) (2013) 1262–1263.

[6] C. Guerreiro, F. de Leeuw, V. Foltescu, Air quality in Europe, Tech. rep., European Environment Agency (2013).

[7] M. R. Jabbarpour, R. M. Noor, R. H. Khokhar, Green vehicle traffic routing system using ant-based algorithm, Journal of Network and Computer Applications 58 (2015) 294–308.

[8] Open-Street-Map-Fundation, OpenStreetMap (jan 2017). URL `http://www.openstreetmap.org`

[9] D. Krajzewicz, J. Erdmann, M. Behrisch, L. Bieker, Recent Development and Applications of SUMO - Simulation of Urban MObility, International Journal On Advances in Systems and Measurements 5 (3) (2012) 128–138.

[10] H. Amer, N. Salman, M. Hawes, M. Chaqfeh, L. Mihaylova, M. Mayfield, An Improved Simulated Annealing Technique for Enhanced Mobility in Smart Cities, Sensors 16 (7) (2016) 1013.

[11] C. Menelaou, P. Kolios, S. Timotheou, C. Panayiotou, A congestion-free vehicle route reservation architecture, in: 2016 18th Mediterranean Electrotechnical Conference (MELECON), no. April, IEEE, 2016, pp. 1–6.

[12] M. Shahidehpour, Z. Li, S. Bahramirad, A. Khodaei, Optimizing Traffic Signal Settings in Smart Cities, IEEE Transactions on Smart Grid 3053 (4) (2016) 1–1.

[13] M. Tropea, A. F. Santamaria, Vehicular traffic optimization in VANETs: A proposal for nodes re-routing and congestion reduction, Advances in Electrical and Electronic Engineering 13 (4) (2015) 376–385.

[14] D. H. Stolfi, E. Alba, Red Swarm: Reducing travel times in smart cities by using bio-inspired algorithms, Applied Soft Computing Journal 24 (0) (2014) 181–195.

[15] J. Toutouh, E. Alba, Performance analysis of optimized VANET protocols in real world tests, in: Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International, IEEE, 2011, pp. 1244–1249.

[16] A. Wegener, M. Piórkowski, TraCI: An Interface for Coupling Road Traffic and Network Simulators, in: Proc. of the 11th Communications and Networking Simulation Symp., CNS '08, ACM, NY, USA, 2008, pp. 155–163.

[17] I. Boussaïd, J. Lepagnot, P. Siarry, A survey on optimization metaheuristics, Inf. Sciences 237 (2013) 82–117.

[18] T. Bäck, Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms, Oxford University Press, Oxford, UK, 1996.

[19] T. Bäck, U. Hammel, H. P. Schwefel, Evolutionary computation: comments on the history and current state, IEEE Transactions on Evolutionary Computation 1 (1) (1997) 3–17.

[20] D. E. Goldberg, K. Deb, A comparative analysis of selection schemes used in genetic algorithms, in: Foundations of Genetic Algorithms, 1991.

[21] Ayuntamiento de Malaga, Área de Movilidad (jan 2017). URL `http://movilidad.malaga.eu`

[22] D. H. Stolfi, E. Alba, An Evolutionary Algorithm to Generate Real Urban Traffic Flows, in: Advances in Artificial Intelligence, Vol. 9422 of Lecture Notes in Computer Science, Springer Int. Publishing, 2015, pp. 332–343.

[23] S. Hausberger, M. Rexeis, M. Zallinger, R. Luz, Emission Factors from the Model PHEM for the HBEFA Version 3, Tech. Rep. I (2009).

[24] L. Int Panis, C. Beckx, S. Broekx, I. De Vlieger, L. Schrooten, B. Degraeuwe, L. Pelkmans, PM, NOx and CO2 emission reductions from speed management policies in Europe, Transport Policy 18 (1) (2011) 32–37.

[25] A. D. Kontogianni, E. I. Papageorgiou, C. Tourkolias, How do you perceive environmental change? Fuzzy Cognitive Mapping informing stakeholder analysis for environmental policy making and non-market valuation, Applied Soft Computing 12 (12) (2012) 3725–3735.

[26] D. E. Goldberg, E. Smith, R. E. Smith, Nonstationary Function Optimization Using Genetic Algorithms with Dominance and Diploidy., in: International Conference on Genetic Algorithms, 1987, pp. 59–68.

[27] J. H. Holland, Adaptation in natural and artificial systems, The University of Michigan Press, 1992.

[28] V. Černý, Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm, Journal of Optimization Theory and Applications 45 (1) (1985) 41–51.

[29] S. Kirkpatrick, C. D. Gelatt, M. Vecchi, Optimization by simulated annealing, Science 220 (4598) (1983) 671–680.

[30] D. J. Sheskin, Handbook of parametric and nonparametric statistical procedures, crc Press, 2003.