Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 8873962, 19 pages
https://doi.org/10.1155/2020/8873962

Research Article

Performance Analysis of Decentralized VS Centralized Control for the Traffic Signal Synchronization Problem

Ludovica Adacher and Marco Tiriolo

Roma Tre University, Via Della Vasca Navale 79, Rome 00146, Italy

Correspondence should be addressed to Ludovica Adacher; adacher@dia.uniroma3.it

Received 6 July 2020; Revised 22 September 2020; Accepted 19 October 2020; Published 9 November 2020

Academic Editor: Zhixiang Fang

This paper suggests the adoption of a spatial decomposition method to solve the signal synchronization problem. A good signal setting maximizes the number of vehicles passing through intersections, while minimizing gas emissions and possible delays experienced by drivers. The signals synchronization issue can be defined as the problem of finding the offsets, the green timings, and the cycle length for a series of controlled intersections, minimizing the total delay of the network subject to admissibility constraints. In this paper, the authors optimized the signal setting through a new Surrogate Method calculating the objective function via the $\mathrm{CTM}_{\mathrm{UT}}$ model while performing a simulation. A spatial decomposition approach is here suggested with a simultaneous analysis of different levels of cooperation among subnetworks. This study tries to identify a subnetwork that might be representative of the entire network while taking into consideration two factors: efficiency and efficacy. A comparison between centralized and decentralized control is performed.

1. Introduction

Traffic congestion can be reduced through a traffic signal control. A traffic signal control gives an improvement both for the drivers (minimizing travel time and delay) and for the environment (reducing both the energy consumption and gas emission). Nevertheless, controlling the traffic signals of a transportation network is a significant challenge due to its large-scale and complexity. Traffic congestion on roads is a serious problem, especially for big cities in the world.

The traffic signal synchronization is a nonconvex problem and sometimes finding a quick and optimal solution even for small networks can be difficult [1, 2]. It has been showed that the improvement in the traffic flow can reduce fuel consumption, gas emissions and accidents [3–6].

The signal synchronization problem consists in the simultaneous optimization: the offsets, the green timings and the cycle length at each junction (computed by delay minimization) for a series of junctions see [7].

According to the classification presented in [8], there are three different approaches to solve the problem:

(i) Centralized approach: The majority of signal timing optimization algorithms use a centralized formulation and architecture. At the same time, for all intersections, they optimize various signal timing parameters (i.e., cycle length, green times, and offsets). However, network signal timing optimization is an NP-hard problem and a central optimization technique will not be scalable and applicable to large transportation networks [1, 9–12].

(ii) Hierarchical or distributed approaches decompose the network optimization problem into a multilevel control problem with distinct objectives at each level. The underlying concept of most hierarchical approaches is to make network level decisions at the upper (or central) level and the real-time, small-area computations in the lower (or intersection) level. The exchange of information is a crucial aspect [13–17].

(iii) Decentralized approaches decompose the network into regions with varying number of intersections.

> As the result of the decentralization, these approaches are scalable and can be real-time; however, rather than global optimization, they mostly locally control signals and may find a suboptimal solution [8, 18–23].

It is evident that a centralized system will theoretically be able to find optimal solutions, though with a higher amount of information on the system than its distributed counterpart. However, the computation time of a centralized control increases exponentially together with the size of the urban network thus preventing its processing. It is also expected that a centralized system may theoretically provide a more effective control policy than its decentralized counterparts with a better coordination among network components. As a result of the decentralization, these approaches are scalable and can be on real-time basis; however, they control local signals and often find suboptimal solutions. The first developed commercial software is based on a centralized control system, where one computing unit decides for all intersections (e.g., SCOOT and SCATES) or hierarchical architectures where one part of the decision is centralized whereas the other is local (e.g., MOTION and RHODES).

Often the literature suggests a decentralized approach which optimizes each intersection separately and the information used is not sufficient to optimize the offset among intersections, finding suboptimal solutions. Several authors tend to suggest a decomposition based on a single intersection either without or a limited exchange of information. The adaptive traffic signal control systems are used to accommodate real-time traffic conditions. In fact recent development of artificial intelligence, especially the success of deep learning, gives the possibility to use information of individual vehicles to control traffic signals. However, those studies are limited to isolated intersections and their effectiveness was only evaluated in ideal simulated traffic conditions by hypothetical benchmarks (see [24–27]). On the other hand, in [28] the authors evaluating an algorithm through real-world coordinated actuated signals, in a simulated suburban traffic corridor, emulate the real-field traffic condition. However, a corridor (a main street) with eight signals is considered under control. In [29], the author divides the network into individual intersections and proposes a dynamic programming algorithm to minimize total queue length at each intersection. The authors of [30] propose a rolling horizon based on predictive microscopic simulation algorithms based on cumulative vehicle delay. The max-pressure control is of great theoretical significance in terms of the decentralized control, which is able to guarantee the global stability by implementing a localized control policy. The back-pressure controller is a distributed feedback system which does not require knowledge of the global network inflow. Also the back-pressure controller adjusts just local green splits based on both upstream and downstream local queue length measured at each intersection. In [31], the author proposes an a-cyclic distributed max-pressure signal controller based on measurements of queue lengths at adjacent intersections. In [21], the author extends Varaiya's work to cycle-based control with the exertion of logic function for the network-wide coordination. Others authors propose a decomposition approach but their test cases are not detailed enough (often they are corridors instead networks) or introduce more simplification for the optimization process. In [32], the authors propose for the freeway ramp metering a distributed optimization algorithm based on the multi-agent A-ADMM formulation, applied to both shared control and shared state systems. In [22], the authors introduce a decomposition mechanism for the global anticipatory network traffic control problems, based on dynamic clustering of traffic controllers. Their technique gives the possibility to recognize when and which controllers should be grouped in clusters, and when they can be optimized separately.

Despite the decentralized approaches presented in literature, there are only a few studies comparing the performance of the distributed system to its centralized version. The authors of [20] propose a decentralized approach based on max-pressure controller, to evaluate their performances; the control systems are applied to a two-dimensional three-by-three grid network. The decentralized approach reaches an optimal level of about 22%. In [19], they present an alternative decentralized solution approach based on the neighbourhood concept, analyzing a real urban network of 58 signal-controlled. The discrepancies between the centralized and decentralized controllers are about 20%–30% with a reduction to 8%–15% once the network traffic router is introduced. In [23], the authors propose a distributed traffic signal control based on Cell Transmission Model. For lager topology up to 72 intersections, the decentralized approach, which requires no communication, reaches an optimal level of about 30%, with two rounds of communication about 8%. In all these studies that examine the decentralized approaches they present different objectives functions (delay, total travel time, and queue) and they solve the signal setting problem fixing only the green time of the controlled intersections whilst the cycle and the offset are not taken into consideration, thus reducing both the complexity of the problem and computational time. The main contributions of this study are the following:

(i) The objective function is the delay minimization. The delay is considered as the number of vehicles obstructed in the road section. Consequentially, the time saving due to signals synchronization reduces pollution produced by traffic and fuel loss due to low running speed.

(ii) The decision variables for the delay minimization are: (i) the green time, (ii) the offset, and (iii) the cycle for each controlled intersection (rather than only the green time).

(iii) The network is decomposed in sub-networks (rather than in individual intersections); network, rather than the neighborhood concept.

(iv) The proposed clustering is based on the physical topology and on the features of the urban road network (rather than the neighborhood concept).

(v) The analysis is performed on a real-world network with 56 signalized intersections and 39 intersections under control (rather than on small networks or corridors).

In our previous studies, we have demonstrated the effectiveness of the Surrogate Method to solve the traffic signal synchronization problem proposing a centralized approach. However, the computing time grows with the size of decision vector (i.e., the urban network that needs to be optimized). The Surrogate Method is applied to single sub networks, minimizing the total delay and optimizing the cycle, the offset, and the green time ratio. Different levels of exchange of information between subnetworks are considered. In this paper, a decentralized approach is presented, and a comparison with a centralized approach is reported. Generally, a decentralization approach is created as a way to improve efficiency and take advantage of potential economies of scale. In fact, decentralization looks to improve the speed and flexibility. Our aim is to find a decentralized approach with a good trade off between optimality and computational time.

1.1. Contribution of the Paper. As shown previously, signal timing optimization in an urban network is an NP problem and a central approach is not able to find the optimal solution in a reasonable amount of time. The hierarchical approaches can find solutions faster but require significant investment in infrastructure to provide communications between a central unit and each local optimizer. The existing decentralized approaches find suboptimal signal timing parameters. Given the interactions among travellers and between travellers and the network in the transportation system, it is difficult to formulate pure mathematical models to evaluate performances. Simulation has been more widely used as the tool to evaluate transportation system performance under different policies. When the system operates in a stochastic environment and no closed form expression for objective function is available, the problem is further complicated by the need to estimate the function. In this case, traditional optimization methods based on derivatives cannot be applied. Most known approaches are based on some form of random search, or ordinal optimization approach. In addition, also being the simulation computationally expensive, the extensive exploration of the entire solution domain would imply unacceptable calculation time [33]. To avoid these problems, the Surrogate Method is introduced, first for manufacturing problem and then also for transportation problem [1, 9, 34]. This method combines the advantages of stochastic approximation type of algorithm with the ability to obtain sensitivity estimates. The gradient information necessary to drive the stochastic approximation part of the Surrogate Method is simplified considering the estimation of the objective function for a selection set (it will be describe in Section 4). In previous studies, it is demonstrated the capacity of the Surrogate Method (SM) to find central optimal solution to problems concerning signal setting and combined signal setting, due to its ability to jump out of local minimum. Hence, it is showed the efficacy and the efficiency of the SM with respect to the Projected Gradient Algorithm (PGA), the Particle Swarm Optimization (PSO), and the Genetic Algorithm (GA) (see [9, 35]). Given the complexity of this problem, a spatial decomposition method is introduced. The methodology here suggested can efficiently find optimal solutions without the need of a central unit. Specific contributions of this paper are as follows:

(i) Different decomposition of the network is proposed

(ii) Different decentralized approaches based on different levels of cooperation are suggested

(iii) Some simplifications to reduce the calculation time of SM are introduced

(iv) A comparison between centralized and decentralized approach is presented

Given the characteristics of the signal setting problem, in this paper, some simplifications for the SM have been introduced, while suggesting to reduce the complexity of the problem thought the network decomposition. Given the subnetworks and considering different levels of cooperation, the new SM is applied. The decomposition approach wants to reduce the computational time in order to find a solution that might be compatible with the central optimal solution. It is important to notice that the SM complexity is strictly related to the dimension of the network. Given a reduction in the network dimension, it is possible to obtain a significant reduction of the computational time.

The signal setting improves driver safety, but it also provokes delays. For this reason, many researches try to minimize total delay, being the sum of all vehicles delays. This objective function is often calculated through simulation approaches. The $\mathrm{CTM}_{\mathrm{UT}}$ is assumed in order to calculate the total delay caused by the signalized intersections. A centralized solution will provide a better performance, but it is often unrealistic. Unfortunately, the traffic signal synchronization is a complex (NP-hard) problem, and it is usually hard to be managed, hence it is unfit for online decision making, especially when the problem involves several intersections and a large time horizon. The centralized solution also presents disadvantages because it requires global information concerning the status of the network, and it is not robust, if a failure occurs in the network the system must recalculate a new solution. On the other hand, distributed strategies can be more robust to failures, and distributed approaches can find good application in different fields [8, 18, 23].

The approach is described in Figure 1. In Urban Traffic Simulation, the value of the objective function is calculated via simulation utilizing the $\mathrm{CTM}_{\mathrm{UT}}$; in Traffic Control, the values of signal settings are defined by the Surrogate Method and the Network Clustering Method identifies different subnetworks that are considered for the signal setting optimization. In other words, thanks to the Simulator the flow in the nodes of the network, and the value of the objective function is determined. Selecting the nodes Priority through the clustering method, the Surrogate Method finds the best signal settings for a sub network, optimizing the entire network.

The remainder of the paper is organized as follows: The problem description is given in Section 2. In Section 3, the

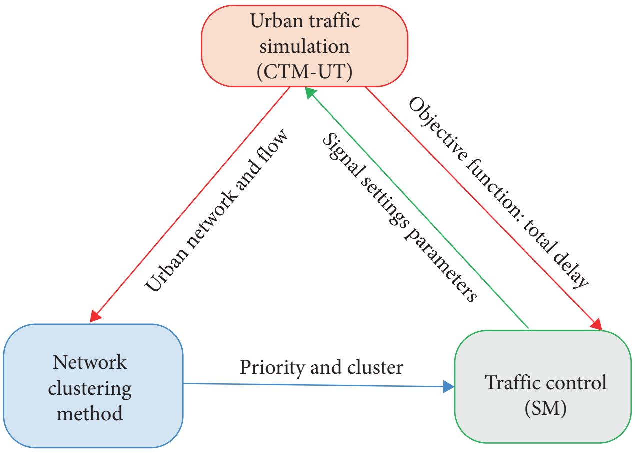

Urban traffic simulation (CTM-UT)
Urban network and flow
Signal settings parameters
Objective function: total delay
Network clustering method
Priority and cluster
Traffic control (SM)

Figure 1: Scheme of our approach.

suggested method for the spatial decomposition and the levels of cooperation are described. Surrogate Method and its improvements are reported in Section 4. Section 5 provides some simulation results, and Section 6 draws the conclusions. In APPENDIX, the CTM$_{\mathrm{UT}}$ and the clustering methods are reported. These concepts are known in the literature and are reported to facilitate understanding of the proposed approach.

2. Problem Description

Over the past years, the well-known Cell Transmission Model (CTM by Daganzo [36]) has been extended, modelling practical applications. Considering the queue formation, queue dissipation, and kinematic waves, the CTM can be used to represent the real traffic dynamic. Recently, many extensions of CTM have been used for signal setting problem [37–41]. The Cell Transmission Model for Urban Traffic (CTM$_{\mathrm{UT}}$) is a good compromise between accuracy and computational complexity.

The CTM$_{\mathrm{UT}}$ represents for each node the connection between the demand upstream intersections and the supplies downstream intersection, and it is also takes into consideration the percentages of the demand of turns for every single lane.

The CTM$_{\mathrm{UT}}$ considers also behaviors of urban drivers (i.e., the movements that model drivers that by mistake take the channelized lane for right-turn).

Given the urban network $U_N(N, E)$, where N is the set of controlled intersection i (nodes), and E is the set of links $e_{i,j}$ (edges), for a fixed value of the cycle C, the green split ratio is represented by $(g_1, \ldots, g_M)$ and the offsets by $(\theta_1, \ldots, \theta_M)$ and M are the controlled intersection links (i.e., the traffic lights). The CTM$_{\mathrm{UT}}$ is used to evaluate the objective function: the global delay of the network $J_D(C, g_1, \ldots, g_M, \theta_1, \ldots, \theta_M)$.

The delay is considered as the number of vehicles obstructed in the road section (as in the paper of Lo see [42]). For each cell, the delay is defined as the difference between the number of vehicles that could travel in the downstream cell less the vehicles travelling in the downstream cell. The total delay includes both the upstream and the downstream node delay (for more details see APPENDIX).

The delay of cell i at time k is equal to the number of its vehicles in it minus the number of vehicles flowing into the next cell $i+1$. The delay of the whole network is obtained by aggregating all cells during the time horizon (T).

(i) $T =$ the time horizon

(ii) $M =$ are controlled intersection links

(iii) $k =$ period time

(iv) $N^a =$ the total number of cells of the link a

(v) $n_i^a(k) =$ the nominal flow in the cell i of the link a for the period k

(vi) $y_i^a(k) =$ flow into cell i of the link a for the period k

(vii) $y_{N^a+1}^a(k) =$ outflow from the next cell N^a+1 of the link a for the period k

The problem is formulated as follows:

$$\min_{(C,g,\theta)\in A_d} J_D(C, g, \theta, T) = \sum_{k=1}^{T} \sum_{a=1}^{M} \sum_{i=1}^{N^a} \left(n_i^a(k) - y_{i+1}^a(k)\right), \tag{1}$$

with the capacity constraints

$$A_d = \left\{ g := [g_1, \ldots, g_M]', g_{\min} \le g_i \le g_{\max}, \sum_{\forall j:\, e_{j,i}\in E} g_i = C, g_i \in \mathbb{Z}^+; i = 1, \ldots, M \theta := [\theta_1, \ldots, \theta_M]', 0 \le \theta_i \le 1; \theta_i \in \mathbb{Z}^+, i = 1, \ldots, M \right\}, \tag{2}$$

where

(i) C is the cycle, which is the same for all intersection links i

(ii) g is an M-dimensional decision vector with $g_i \in \mathbb{Z}^+$ denoting the green time ratio for intersection link i

(iii) where θ is an M-dimensional decision vector with $\theta_i \in \mathbb{Z}^+$ denoting the offset for intersection link i

(iv) $g_{\min} =$ minimum green time ratio for signalized link a fulfilling capacity constraint

(v) $g_{\max} =$ maximum green time ratio for signalized link a fulfilling capacity constraint

$J_D(C, g, \theta, T)$ is the total delay on the network when the decision variables (green split vector, offsets, and cycle) are fixed.

The $g_{\min}$ and $g_{\max}$ guarantee a minimum of green and red for a traffic light. The decision variables concern only the links connected to the signalized intersection (i.e., intersection with the traffic light).

The analysis of the objective function shows that this function is convex with respect to the cycle; instead it presents many local minimum with regard to the green split vector and offsets. Since the Surrogate Method calculation time is strictly dependent on the size of the problem, the cycle is not optimized by Surrogate Method. The Cycle is the same for all junctions and fixed by the binary search, and only after it has been found its optimum value the Surrogate Method is applied. The shape of the objective function looks convex with respect to the cycle variations and it is quasiconvexity respect when the green ratio changes and non convex with many peaks, with respect to the offset variations. From this analysis, it seems that the hardest task for signal synchronization problem is the offset setting. Figure 2 shows an example of the shape related both to one component of the green split vector variation and one offset variation. To visualize the characteristics of the solution space, it is not necessary to perform the average system time for all possible combinations of cycle/green time ratio/offset. In order to show the characteristics of the solution space, the cycle, the offset, and all green time ratio are kept constant, except for one component at time for each decision vector (or cycle or offset or green time). It is done for different values of green/offset/cycle that are kept constants.

3. Decentralized Approach

Spatial problem decomposition is the process that divides in small areas the space used for the optimization. The optimization of the decomposition is to identify some subsets of traffic lights that can be representative for the entire network [18, 21, 23].

Based on the physical topology model of the urban road traffic network, urban network $U_N(N, E)$ is defined considering the functional properties of urban road network such as the length and traffic capacity of the road sections, and the following definitions can be taken into consideration:

(i) $N = \{n_1, n_2, \ldots, n_n\}$ is the finite set of the nodes which means the controlled intersections, n is the number of controlled intersections.

(ii) $E = \{e_{ij}\colon\ (i, j) \in N\}$ is the finite set of the edges that means the sections of the road through which the two intersections can be connected directly. $|E|$ is the number of elements in set E and it represents the number of the controlled links ($e_{ij} \neq e_{ji}$).

(iii) EL: $E \longrightarrow \mathbb{R}$ is the mapping function from an edge to a positive real number, $\mathrm{EL}(e_{ij}) = l_{ij}$ is the length of the road section e_{ij}.

(iv) $P^*_{i,j}\colon N \longrightarrow \mathbb{R}$ is the mapping function from a couple of nodes $(i, j) \in N$ to a positive real number, $P^*_{i,j}$ is the length of shortest path from i to j.

(v) ETC: $E \longrightarrow \mathbb{Z}$ is the mapping function from an edge to a positive integer, $\mathrm{ETC}(e_{ij}) = tc_{ij}$ is the traffic capacity of the road section e_{ij}.

3.1. Hybrid Clustering. A classification for the nodes of the urban network is introduced to define a new clustering method. A Node Priority considering three different parameters is here introduced:

Degree $D(j)$ is the simplest measure of the node. The degree of the node is the number of incident edges. Given

$$X_{i,j} = \begin{cases} 1, & e_{ij} \in E, \\ 0, & \text{Otherwise.} \end{cases} \tag{3}$$

The degree of the node j is calculated as follows:

$$D(j) = \sum_{i \in N} \left(X_{i,j} + X_{j,i} \right). \tag{4}$$

Betweenness $B(j)$: Betweenness is a measure of the importance of the node with respect to the network. It is based on the idea that a node is central if it lies between many other nodes, in the sense that it is traversed by many of the shortest paths connecting couples of nodes.

Given $d_{i,k}$ is the number of shortest paths between i and k and $d_{i,k}(j)$ is the number of shortest paths between i and k that contain node j. The betweenness of node j $B(j)$ is

$$B(j) = \frac{1}{(n-1)(n-2)} \sum_{(i \neq j \neq k) \in N} \frac{d_{(i,k)}(j)}{d_{(i,k)}}. \tag{5}$$

Flow $F(j)$ is the amount of the flow directed into node. Obviously the flow incoming into the node must be equal to the one coming out. Given the $Y^{e_{i,j}}$ the flow in to link $e_{i,j} F(j) = \sum_{\forall i \in N\colon\, e_{ij} \in E} Y^{e_{i,j}}$.

Definition 1 (Node Priority $P(j)$). The Node Priority $P(j)$ is a function assigning a weight to each node j that takes into consideration the importance of the node with respect to the network, based on its degree, betweenness, and flow. The values normalized are

$$P(j) = \frac{D(j)}{D} + B(j) + \frac{F(j)}{F}. \tag{6}$$

$D = \sum_{i=1}^{n} D(i)$
$F = \sum_{i=1}^{n} F(i)$

Definition 2 (Network Priority $P(U_N)$). Given the network $U_N(N, E)$, the Network Priority is a function that assigns a weight to the network.

$$P(U_N) = \sum_{\forall j \in N} P(j). \tag{7}$$

The node priority is calculated by the sum of three normalized elements, this implies that each element varies

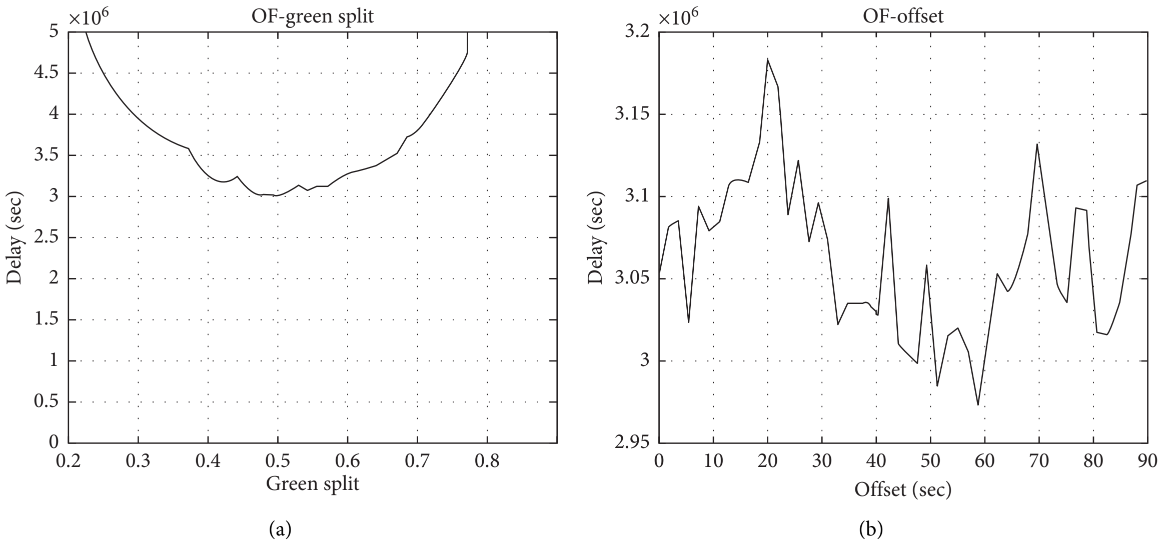


Figure 2: Shape of the objective function.

from 0 to 1. For a node, the normalized degree increases with the number of the incident edges on it, and the betweenness increases with the number of the shortest paths traversed it and the normalized flow increases with the flow directed into the node. It is evident that the first two terms can change only if the urban network changes. While considering these priorities it is possible to understand the importance of each node with respect to the network. The Node Priority is also used to determine the Priority of each subnetwork. The Priority of the sub-network is given by the sum of the Priority of each node in the subnetwork. Given the partition, for each node and for each subnetwork, the Priority is calculated.

Given the procedure to calculate the Priority, the proposed algorithm to clustering the network is introduced.

The Hybrid algorithm suggests a clustering of the network considering the Priority method. The Hybrid algorithm adds to the clustering one subnetwork S_{K+1} composed by the nodes with highest priority on the network. Probably, these nodes are the nodes with more traffic; in fact, highest priority means this node has big probability to be chosen by a driver, because it is traversed by many shortest paths and has many incident edges. The aim of the Hybrid algorithm is to compose a subnetwork representing the whole urban network. It is evident that these most important nodes for the network must be optimize and synchronize, minimizing the total delay of the network. When the green time and the offset of these nodes are optimized, it is possible to optimize the others nodes considering different levels of cooperation. Given a clustering S from well known cluster algorithms and K is the number of clusters, the Hybrid algorithm adds a cluster composed by a set of nodes (not more then $(K - 1 + (N/K)))$, while selecting the nodes with highest Priority, and at least one node for each cluster in S, deleting these selected nodes from their clusters. The dimension of this new cluster S_{K+1} is defined through a preprocessing analysis. If in S the cluster with the highest Priority contains few nodes, the Hybrid clustering will consist in K sub networks, otherwise in $K + 1$. The new cluster is composed by the nodes with the highest Priority from each sub network in S. Algorithm 1 reports the steps of the Hybrid algorithm. In Step 1, the main sub network S_{k+1} is formed by the sub network S^* with the highest Network Priority; Step 2 adds to the main sub network S_{k+1} the node with the highest Node Priority from the other subnetworks in S. Step 3 guarantees that the dimension of S_{k+1} does not exceed $(K - 1 + (N/K))$.

Example 1. In Figure 3, it is reported on the left side the clustering given by Newman algorithm $S = \{(1, 2, 3, 5, 6), (4, 7), (8, 9)\}$ (in bold the nodes with the highest Priority) and on the right the Hybrid clustering $S^H = \{(1, 2, 4, 5, 8), (3, 6), (7), (9)\}$. The red nodes are the nodes with the the highest Priority. Given the small dimension of the network, $|S_{K+1}|$ is set to be equal to 5. The Hybrid cluster gives 4 subnetworks and the new cluster $S_{K+1} = (1, 2, 4, 5, 8)$ is composed by 5 nodes with the highest Priority almost one for each of the three subnetowrks of Newman: {1,2,5} from the sub network with higher Priority, {4} from the second, and {8} from the third.

Given a clustering of the entire urban network, we proposed different decentralized approaches based on different levels of cooperation between the individuated subnetworks.

3.2. The Different Levels of Cooperation.

Before introducing the different approaches, the concept of Traffic Signal Control is introduced.

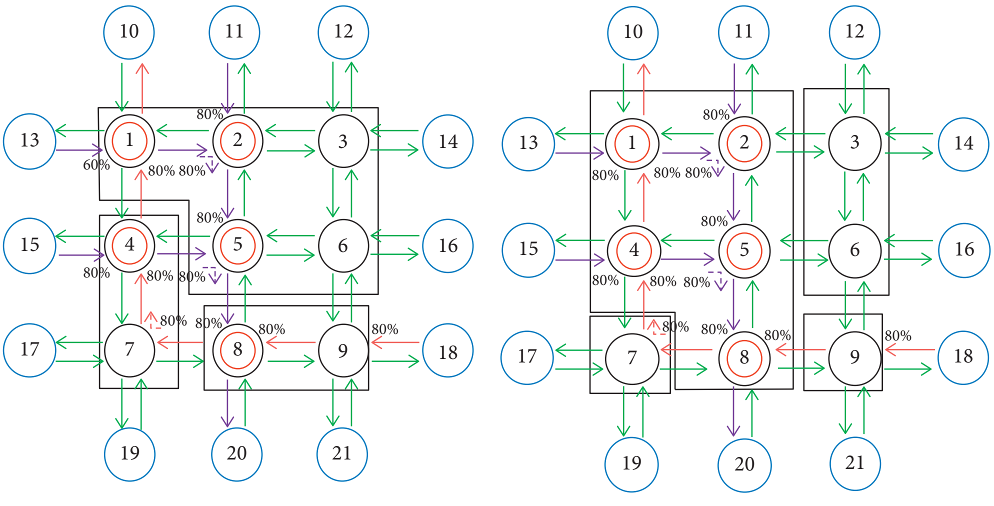
Definition 3. Traffic Signal Control SM: $U_N \longrightarrow \mathbb{Z}$.

Given the network $U_N(N, E)$, the Traffic Signal Control applies the SM to U_N fixing all the variables of the signals control in the network minimizing the global delay of the network $U_N(N, E)$.

(i) C_j is the cycle for the controlled intersection link j

```
Step 0 Given:
    the network $U_N(N,E)$ and
    $S=\{S_1,S_2,\dots,S_K\}$ S is a cluster of $K$ sub-networks given by a clustering method
    $S_{k+1}=\{\}$ is the new sub-network
    $S^H$ is the Hybrid clustering set;
    $i,m,j,a$ int;
    $P(i),\forall i\in N$
    $P(S_j),\forall S_j\in S$
Step 1 Calculate $S^*=\arg\max\{P(S_j)\}, \forall S_j\in S$
    $S=S-S^*$
    $S_{k+1}=S^*$
Step 2 WHILE $(S\neq\{\})$ DO
    $S_{K+1}=S_{K+1}+\{i=\arg\max\{P(i)\}, i\in S$
    FOR $m=1,\leq k,m++$
    IF $(i\in S_m)$
    $\{S_m=S_m-\{i\}$
    $S^H=S^H\cup S_m$
    $S=S-S_m\}$
Step 3 WHILE $(|S_{K+1}|\geq K-1+(N/K))$ DO
    $i=\arg\min\{P(i)\}, \forall i\in S_{K+1}$
    $S^*=S^*+\{i\}$
    $S_{k+1}=S_{k+1}-\{i\}$
Step 4 RETURN $S^H=S^H\cup S^*\cup S_{k+1}$
```

ALGORITHM 1: Principal steps of Hybrid algorithm.



(a)

(b)

FIGURE 3: Example of Hybrid clustering based on Newman partition. (a) Newman clustering. (b) Hybrid clustering.

(ii) g_j is the green time ratio for controlled intersection link j

(iii) θ_j is the offset for controlled intersection linkj

(iv) $J_D^{U_N}(C_1,\dots,C_{|E|},g_1,\dots,g_{|E|},\theta_1,\dots,\theta_{|E|})$. is the global delay of the network when the parameters are fixed

The C_j is the same for all intersection links and it is fixed by binary search (C).

$$\mathrm{SM}(U_N)=\left\{\left(C,g_j^*,\theta_j^*\right)\forall j\in E\colon\ \min J_D\right\}. \tag{8}$$

In practice, the Traffic Signal Control, based on the SM, fixes the control variables (C,g,θ) to minimize the total delay in a given network U_N. How the SM optimizes the traffic signal setting is shown in the next section. Given the network U_N and the clustering S, different levels of cooperation are considered and reported. In a decentralized system, there is a lack of cooperation every time the subsystems do not exchange information. In this case, one cluster gives information to other clusters about one or more of its decision variables.

Definition 4. Coopnet CP: $(U_N, S) \longrightarrow S_{\text{CP}}$.

The Coopnet is a function that, given the network $U_N(N, E)$ and the clustering $S = \{S_1, \dots, S_K\}$, returns a subnetwork S_{CP} composed by the subnetwork with the highest Priority in S and the nodes with highest Priority in U_N, at least one for each subnetworks in S.

(i) $S^* = (S_i\colon\ \max(P(Si), \forall S_i \in S))$

(ii) $n_i^* = (j\colon\ \max(P(j), \forall j \in S)_i)$

$$S_{\text{CP}} = S^* \bigcup_{j=1}^{\|S\|} n_j^*. \quad (9)$$

This process utilizes the same criteria of Hybrid algorithm without limiting the dimension of the subnetwork.

Definition 5. COOP means that SM optimized only S_{CP}.

The level of cooperation is minimal, and the Surrogate Method is applied only on S_{CP}. The optimization is applied only on the nodes of the subnetwork, minimizing $(J_D(S_{\text{CP}}))$, the total delay calculated on S_{CP}. The clusters give information only related to the nodes with the highest Priority.

(i) Apply $\text{SM}(S_{\text{CP}}, J_D^{S_{\text{CP}}})$

(ii) SM Optimizes $(C^*, g_i^*, \theta_i^*)\forall i \in S_{\text{CP}}$

(iii) Fix randomly $(C, g_i, \theta_b i)\forall i \in (N - S_{\text{CP}})$

COOP is the method that optimizes only the subnetwork S_{CP}, taking into account the subnetwork S_{K+1} introduced by the Hybrid method and the nodes with highest Priority, one for each subnetwork of the clustering. The choice to take the most important node for each subnetwork guarantees the synchronization between subnetworks. Obviously, when the cooperation increases, the results improve.

Definition 6. COOP1 means that SM at first optimized S_{CP}, and then in a parallel way on the others subnetworks, considering fixed all the traffic parameters of the nodes in S_{CP}. The clusters provide information related to the nodes with the highest Priority, and the SM fixes the control variables of these nodes (all nodes in S_{CP}) and then SM optimizes all the others clusters but the control variables in S_{CP} are kept constant.

(i) Apply $\text{SM}(S_{\text{CP}}, J_D^{S_{\text{CP}}})$

(ii) SM Optimizes $(C^*, g_i^*, \theta_i^*)\forall i \in S_{\text{CP}}$

(iii) Update $S_i = S_i - \{n_i^*\}\forall S_i \in S$

(iv) Apply $\text{SM}(S_1, J_D^{S_1}, \dots, S_{|S|}, J_D^{S_S})$SM Optimizes $(C^*, g_i^*, \theta_i^*)\forall i \in (S_1, \dots, S_{|S|})$

Definition 7. COOP2 means that SM is at first applied on S_{SC} and then in series on the other subnetworks. The SM is applied to other subnetworks in a descending order of Priority. The clusters provide information related to the nodes and those clusters with the highest Priority, and the SM fixes the control variables of nodes in S_{CP}, minimizing $J(S_{\text{CP}})$, and then the SM optimizes the second cluster with the highest Priority (i.e., S^*) but the control variables in S_{CP} are kept constant minimizing $J(S^*)$ and so on.

(i) Apply $\text{SM}(S_{\text{CP}}, J_D^{S_{\text{SC}}})$

(ii) SM Optimizes $(C^*, g_i^*, \theta_i^*)\forall i \in S_{\text{SC}}$

(iii) Update $S_i = S_i - \{n_i^*\}\forall S_i \in S$

(iv) UNTIL $S \neq \{\}$ DO

(v) $\{\ S^* = (S_i\colon\ \max(P(S_i), \forall S_i \in S))$

(vi) $\text{SM}(S^*, J_D^{S^*})$

(vii) $S = S - S^*\}$

(viii) SM Optimizes $(C^*, g_i^*, \theta_i^*)\forall i \in S^*$

Definition 8. COOP3 means that the SM is applied as in COOP2 but the objective function examined in the SM is calculated on the subnetwork considered together with all those nodes already optimized. The clusters give information related to the nodes and to the clusters with highest Priority, and the SM first fixes the control variables of nodes in S_{CP}, minimizing $J(S_{\text{CP}})$; then the SM optimizes the second cluster with higher Priority (i.e. S^*) but the control variables in S_{CP} are keep constant minimizing $J(S_{\text{CP}} \cup S^*)$ and so on.

(i) Apply $\text{SM}(S_{\text{CP}}, J_D^{S_{\text{SC}}})$

(ii) SM Optimize $(C^*, g_i^*, \theta_i^*)\forall i \in S_{\text{SC}}$

(iii) Update $S_i = S_i - \{n_i^*\}\forall S_i \in S$

(iv) UNTIL $S \neq \{\}$ DO

(v) $\{\ S^* = (S_i\colon\ \max(P(S_i), \forall S_i \in S))$

(vi) $\text{SM}(S^*, J_D^{S_{\text{CP}} \cup S^*})$

(vii) $S = S - S^*\}$

(viii) SM Optimizes $(C^*, g_i^*, \theta_i^*)\forall i \in S^*$

A scheme of the Network Clustering method and the cooperation methods is reported in Figure 4. The white boxes are the input, the grey boxes are the methods, and the green boxes are the outputs. Given the network $U_N(N, E)$ and the flow F, the Network clustering method calculates the clustering set S, the Priority Node for each node of the network, and the Priority Network for each subnetworks.

Considering the different levels of cooperation, the signal setting parameters for a subset nodes (just the node in S_{CP}) is optimized, if the cooperation is limited and for all the nodes in $U_N(N, E)$ in the other cases. The different level of cooperation is based on different levels of knowledge of urban network. COOP optimizes the Priority subnetwork only. The other methods optimize the whole network applying the SM on clustering in series or parallel way. When the cooperation is limited, the optimization is obtained considering the delay separately for each subnetwork; instead through a ful cooperation, the delay is calculated considering the union of the subnetworks. Section 6 reports a numerical example of the different levels of cooperation.

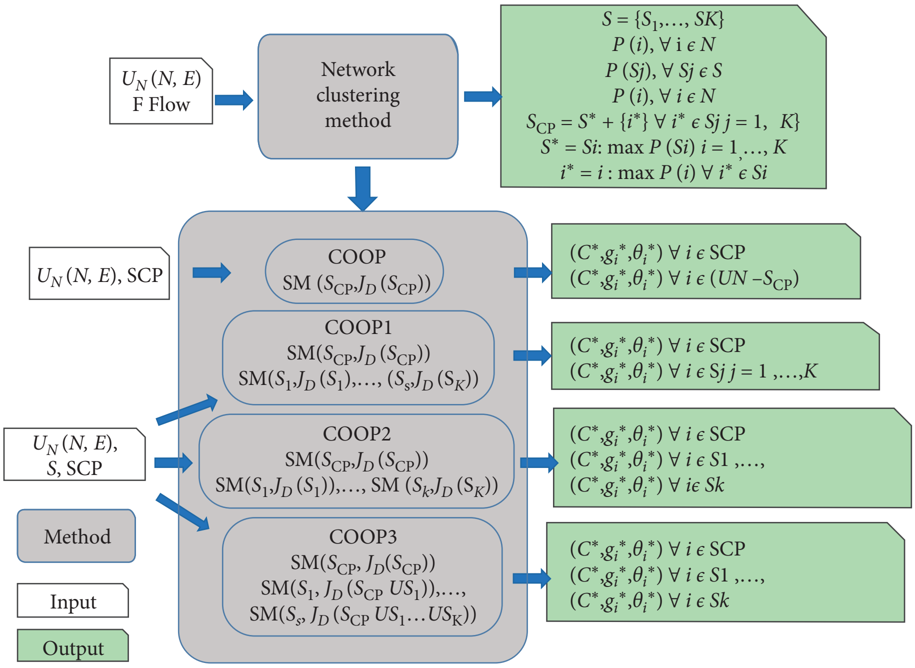

$U_N(N, E)$
F Flow
Network clustering method
$S = \{S_1,\ldots, SK\}$
$P(i), \forall \text{ i} \in N$
$P(Sj), \forall Sj \in S$
$P(i), \forall i \in N$
$S_{\text{CP}} = S^* + \{i^*\} \forall i^* \in Sj\, j = 1, \ K\}$
$S^* = Si$: max $P(Si)\ i = 1_, \ldots, K$
$i^* = i$: max $P(i) \forall i^* \in Si$
$U_N(N, E)$, SCP
COOP
SM $(S_{\text{CP}}, J_D(S_{\text{CP}}))$
$(C^*, g_i^*, \theta_i^*) \forall i \in$ SCP
$(C^*, g_i^*, \theta_i^*) \forall i \in (UN - S_{\text{CP}})$
COOP1
SM$(S_{\text{CP}}, J_D(S_{\text{CP}}))$
SM$(S_1, J_D(S_1), \ldots, (S_s, J_D(S_K))$
$(C^*, g_i^*, \theta_i^*) \forall i \in$ SCP
$(C^*, g_i^*, \theta_i^*) \forall i \in Sj\, j = 1, \ldots, K$
$U_N(N, E)$,
S, SCP
COOP2
SM$(S_{\text{CP}}, J_D(S_{\text{CP}}))$
SM$(S_1, J_D(S_1)), \ldots$, SM $(S_k, J_D(S_K))$
$(C^*, g_i^*, \theta_i^*) \forall i \in$ S1 ,…,
$(C^*, g_i^*, \theta_i^*) \forall i \in Sk$
Method
COOP3
SM$(S_{\text{CP}}, J_D(S_{\text{CP}}))$
SM$(S_1, J_D(S_{\text{CP}}\ US_1)), \ldots$,
SM$(S_s, J_D(S_{\text{CP}}\ US_1 \ldots US_K))$
Input
Output

Figure 4: Procedure providing the clustering and the signal setting parameters of the urban network.

4. Traffic Control: The Surrogate Method (SM)

The Surrogate Method (SM) solves the optimization problem by using the gradient method. This procedure presents an iterative structure that, in each cycle, transforms the original problem with discrete decision variables, into an optimization problem with continuous decision variables. The latter problem is denoted as Surrogate. Subsequently, the gradient estimate, which allows to update the solution, is realized in the discrete field.

The steps sequence of the algorithm is reported in Algorithm 2.

Vector Z is an $2M$-dimensional decision vector, subject to the capacity constraints A_d and $J_D(Z)$ is the delay incurred when the state is Z. The cycle is fixed a priori by the binary search, whereas the green time ratio and the offset are optimized for each controlled link. The integer capacity constraint is relaxed and a resulting surrogate problem is obtained.

The basic idea of this method is to solve a continuous optimization problem by stochastic approximation methods and establish the fact that when (and if) a solution of the relaxed problem ρ^* is obtained it can be mapped into a discrete point $z = f(\rho^*) \in A_d$ which is in fact the solution to the problem.

Note, however, that the sequence $\{\rho_k\}$, $k = 1, 2, \ldots$ generated by an iterative scheme to solve the relaxed problem consists of real-valued solutions which are unfeasible, since the actual system involves only discrete resources. Thus, a key feature of the Surrogate algorithm is that at every step k of the iteration scheme, the discrete state is updated through $z_k = f_k(\rho_k)$ as ρ_k is updated.

This has two advantages:

(i) The cost of the original system is continuously adjusted (in contrast to an adjustment that would only be possible at the end of the Surrogate optimization process)

(ii) It allows us to make use of information typically employed to obtain cost sensitivities from the *actual* operating system at every step of the process

Note that there is an additional operation: the $\{z_k\}$ corresponds to feasible states based on which one can evaluate estimates $\nabla J_c(\rho_k)$, calculated on actual system z_k (not the surrogate state ρ_k, see Step 3). We can therefore see that this scheme is intended to combine the advantages of stochastic approximation type of algorithm with the ability to obtain sensitivity estimates with respect to discrete decision variables.

For each updating step of the Surrogate state (Step 4), it is computed $(2M + 1)$ times the objective function value by the simulation module. Hence, it becomes crucial to develop an efficient simulation module which performs an acceptable computation time. Some simplifications to reduce the calculation time of SM are introduced: the dynamic gradient step, the database of solution and green splits constraints.

4.1. Estimation of the Dynamic Gradient Step. A different green time ratio vector is able to provide the same value of J_D; hence, the gradient ∇J_D will be equal to 0 and there is no update of the state (see Step 3 of Algorithm 2), forcing the convergence of the algorithm in a nonoptimal solution. For this reason, a perturbation of the gradient is introduced. The next step is to modify the component of the gradient equal to 0 through the counter z as follows (Algorithm 3):

```
0 Initialize $\rho_0 = z_0$ and perturb $\rho_0$ to have all components non-integer.
For any iteration $k = 0, 1, \ldots$ repeat the following steps
1 Determine the selection set $S(\rho_k)$ using these steps:
  Initialize
  $I = \{1, \ldots, 2M\}$
  $v = \rho - \lfloor \rho \rfloor$
  Repeat the following steps Until $I \neq \varnothing$
    $i = \mathrm{argmin}_{j \in I}(v_j)$
    $y_i = v_i$
    *
    $Wi = \sum_{j \in I} e_j$
    *
    $v = v - y_i Wi$
    *
    $I = I / \{i\}$
    *
  $S(\rho_k) = \{Wi - \lfloor \rho \rfloor,\ i = 0, \ldots, 2M\}$
2 Select a transformation function $f_k$ such that
(i) $z_k = f_k(\rho_k) = \mathrm{argmin}_{z \in S(\rho_k)} \|z - \rho_k\|$.
3 Evaluate the gradient estimation
(i) $\nabla J_D(\rho_k) = [\nabla_1 J_D(\rho_k), \ldots, \nabla_N J_D(\rho_k)]^T$,
(ii) using the following relationship
(iii) $\nabla_j J_D(\rho_k) = J_D(z^j) - J_D(z^k)$, where $k$ satisfies
(iv) $z^j - z^k = e_j$ (versor with $j$-th component equal to 1).
4 Update state: $\rho_{k+1} = \pi_{k+1}[\rho_k - \eta_k \nabla J_D(\rho_k)]$.
5 If some stopping condition is not satisfied, repeat steps for $k + 1$. Else set $\rho^*$.
```

Algorithm 2: Steps of the surrogate method SM.

4.2. Database of the Solutions. For every gradient estimation, the SM requires $2M + 1$ values (see Step 3 of Algorithm 2). To improve the efficiency of the algorithm, each solution with relative delay function is memorized. If the value of green ratio is already calculated, the SM algorithm saves the traffic simulation.

4.3. Green Splits Constraints. The green splits constraints imply a simplification of the decision vector.

$$\sum_{\forall j:\, e_{j,i} \in E} g_i = C. \tag{10}$$

For every intersection, there is one independent variable only; as a matter of fact, the sum of green for every links of an intersection is equal to the cycle time. One decision variable for each intersection is taken into consideration (i.e., if the cycle is 100, and the intersection is composed by three intersection links, given the green times vector $[30, 35, 35]$ only the first component will be the decision variable). It is evident that if the decision green time variable changes (i.e., $30 + 1 = 31$), also the neighbouring feasible states will change based on the time cycle (i.e., $[31, 34, 35]$, $[31, 34, 35]$). The decision vector is chosen among the neighbourhood feasible states considering the green times vector that minimizes the objective function J_D. This simplification reduces the dimension of the decision green time variables, and there is a green time variables for each intersection rather than each intersection link.

Section 5.2 reports an analysis of these simplifications SM.

5. Numerical Results

The results of the optimization approach applied on the entire network (centralized control) are compared to the results obtained while considering different network partitions and different levels of cooperation. The goal is to analyze whether the optimization must be applied on the entire network or obtain satisfying results just applying the Surrogate Method on subnetworks.

The aim of this paper is to evaluate if the decentralized control can be used to solve the synchronization signal problem. For this reason the evaluation of the trade-off between efficiency (goodness of the solution) and efficacy (computational time) is fundamental. Each method is compared in terms of efficacy (Δ_{J_D} means that the variation on the total delay with respect to the centralized solution) and efficiency (Δ_{CT} implies a variation in the calculation time with respect to the centralized solution), a positive percentage determines a worsening while a negative one an improvement if compared to the centralized solution.

(i) $\Delta_{J_D} = (J_D(\text{Decentralized}) - J_D(\text{Centralized})) / (J_D(\text{Centralized}))^* 100$

(ii) $\Delta_{\mathrm{CT}} = (\text{Calculation Time}(\text{Decentralized} - \text{Calculation Time}(\text{Centralized})))/(\text{Calculation Time}(\text{Centralized}))^* 100$

The Hybrid Algorithm is applied to Newman and K-means clustering methods. The Newman clustering is based on betweenness, considering in part our Priority Method. K-means takes in consideration the neighborhood concept, often used in the literature when a set of nodes are detected to be optimize. The results put in evidence that the best results are given by the Hybrid Method considering the Newman clustering, which provides the same solution of the centralized method. This highlights the efficiency and efficacy of our Priority Method that finds a subset of nodes that represent the whole network.

5.1. Small Case Study. In this section, to better explain the Network Clustering Method, it is carried out a small case study, and some results are presented. The small network is the same presented in [35] and for simplicity the considered parameters are here reported. The network is composed by 9 four way signal-controlled intersections and 12 centroids for the input/output flow on the network. The links (300 meters in length) have a capacity of 900 (veic/hour) and two lanes. The capacity and demand of the links are split over the two lanes fifty-fifty, except for the principal paths where the demands are represented in Figure 5.

The $\mathrm{CTM_{UT}}$ models the behavior of urban drivers. It is considered an additional movement belonging to the right-turn so that the demand lane is split in right-turn ($100\% - \omega$ turn demand) and through (ω turn demand) movements. This through demand is applied to all lanes for the right-turns of network and it is part of phase 1 for east-west/west-east directions and phase 2 for south-north/north-south directions. These percentages represent drivers that must move forward but, due to a mistake, take the channelized lane for right-turn. It is examined the ω varying from 2% to 6%. It is necessary to model flows crossing the channelized lanes, flow upstream intersection, and the complex flow intersection with detail level quite close to the microscopic models. The inflow demands is 500 (veic/h) from the nodes 11, 13, 15, 18 during a simulation time of 400 sec.

An inflow demand of 800 (veic/h) to exam different levels of congestion is also performed. The delay is calculated in seconds. All experiments are performed on a desktop computer with an Intel i5-3470 processor (3.2 GHz) with 8 GB of DDR3 RAM running 64-bit Windows 7. Even if each iteration takes 100 msec to complete, plus the communication overhead to calculate the objective function by $\mathrm{CTM_{UT}}$ requiring approximately 10 sec, the SM (centralized version) would require about 35 minutes to provide the result.

5.2. Surrogate Model with Simplifications. It is here first reported a comparison for the extensions on the SM. What follows is the nomenclature:

(i) SM is the traditional version

(ii) $\mathrm{DS_{SM}}$ considered the Database of solutions

(iii) $\mathrm{DG_{SM}}$ used the Dynamic Gradient step estimation

(iv) $\mathrm{DSDG_{SM}}$ used both the extensions

The extensions of Surrogate Method are compared in terms of efficiency (number of times that the J_D must be calculated NJ_D) Table 1. Δ_{NJ_D} represents the savings percentage on the calculation time.

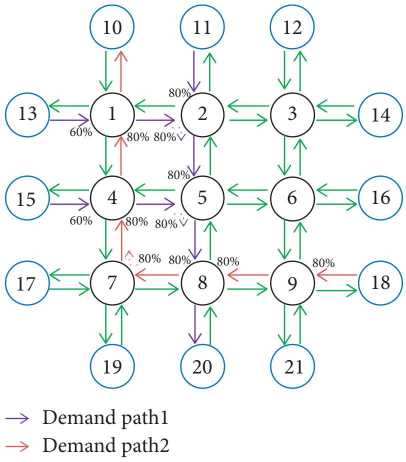


FIGURE 5: Small case study. The centroids are red, signalized intersections are red, and links are green.

```
if $J(g^k) \leq \min J(g^{k-1})$ then
    $p = 1$
else
    $p = p + 1$
end if
for $i = 1$: $n$ do
    if $\nabla_i J < -\varepsilon$ and $\nabla_i J > \varepsilon$ then
        $z = z + 1$
        $\nabla_i J = \text{rand} \in [-z, z]$
        $\eta_i = 0.5$
    else
        $\eta_i = (0.5/p)$
    end if
end for
```

ALGORITHM 3: Dynamic gradient step estimation.

TABLE 1: Efficiency of extensions.

	J_D	NJ_D	Δ_{NJ_D}
SM	7906	291	
$\mathrm{DS_{SM}}$	7906	106	−64%
$\mathrm{DG_{SM}}$	7743	271	−7%
$\mathrm{DSDG_{SM}}$	7743	208	−29%

Despite the simplifications on the MS able to decrease the calculation time, the new dynamic gradient step estimation determines an improvement in the solution of 2% with respect to the traditional SM. The results obtained with the suggested SM extensions showed that when the DS is applied, a strong reduction in the efficiency occurs (64%).

5.3. Clustering Algorithms. An application of the clustering method is given in Figure 6 on the left the clustering obtained by K-means partition ($K = 4$) and on the right the main subnetwork (i.e., the subnetwork with the highest Priority) and the most important nodes for each subnetwork are highlighted in red. The clustering given by the 4-means

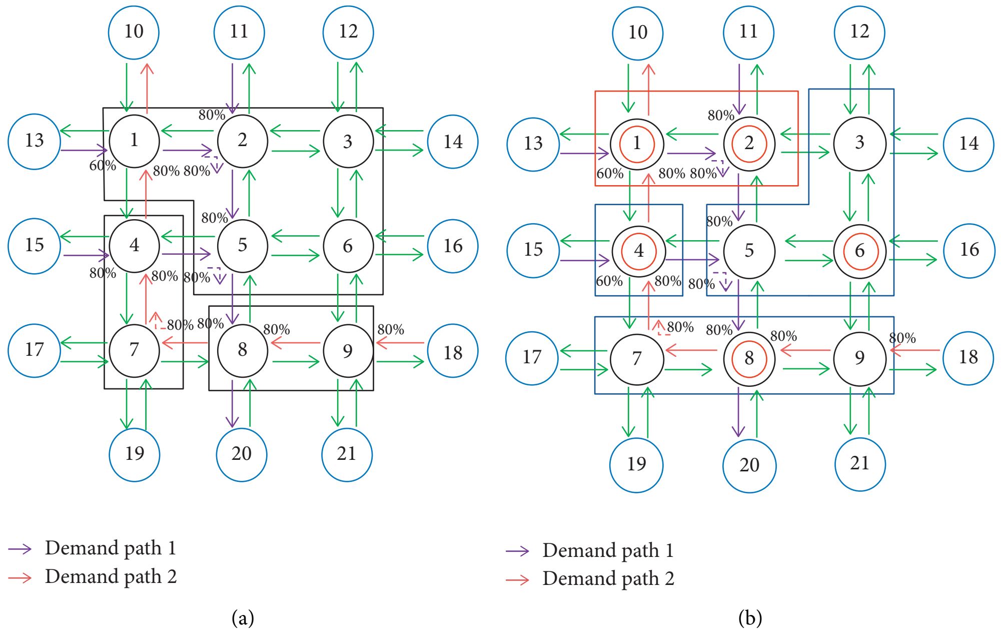


Figure 6: Clustering individuated by 4-means partition. (a) 4-means clustering. (b) Priority process.

algorithm is { (1,2),(7,8,9),(4),(3,5,6) }, in bold the main subnetwork and the nodes with the highest Priority in each subnetwork).

Example 2 (Hybrid-ALGORITHM). Given the small dimension of the test network the dimension of the sub network introduced by Hybrid method is fixed to 5. The Hybrid clustering adds a cluster formed by the cluster with high Priority (1,2) together with the node with high Priority. One is for the others clusters, obtaining the clustering: $\{(1,2,4,6,8),(7,9),(3,5)\}$.

Example 3 (COOPERATION). In COOP, the SM is applied on the main subnetwork together with the nodes with higher Priority one for each subnetwork of the clustering (i.e., $S_{\text{CP}} = (1,2,4,6,8)$). In COOP1, the SM is applied at first on S_{CP} optimizing all the control variables of the nodes (1,2,4,6,8) that are kept constant for the other subnetworks. Then the SM is applied in a parallel way on (7,8,9) considering the value of signal setting parameters for intersection 8, on (4) considering set 4, and on (3,5,6) considering set 6. In practice, the signal setting for the nodes with the highest Priority for each subnetworks are fixed by the main subnetwork. In COOP2, the approach is the same but the SM is applied in series on the subnetworks in a decreasing Priority order. In COOP3 we need more information. The SM is applied on S_{CP} (1,2,4,6,8), then on (7,8,9) considering the value of signal setting parameters for intersection 8 set, though the total delay is calculated on the subnetwork (1,2,4,6,8,7,9), and so on.

Table 2 takes into consideration the Newman partition and in Tables 3, and 4 the K-means partition for $K = 4$ and $K = 3$ are reported. The results of Hybrid approach based on 4-means clustering are also reported. In the tables, CT is the computation time and it is given in minutes. The results of

Table 2: Decentralized vs centralized with Newman decomposition.

	Subnetworks	J_D	Δ_{J_D}	Δ_{CT}	CT
Newman	[1,2,3,5,6],[4,7],[8,9]				
Tot net	[1,2,3,4,5,6,7,8,9]	5997			35
Main net	[1,2,3,5,6]	6330	5.6%	−84%	5.6
COOP	[1,2,3,4,5,6,8]	6318	5.3%	−34%	23
COOP1	[1,2,3,4,5,6,8],[4,7],[8,9]	6214	3.6%	−84%	5,6
COOP2	[1,2,3,4,5,6,8],[4,7],[8,9]	6195	3.3%	−82%	6
COOP3	[1,2,3,4,5,6,8],[4,7],[8,9]	5999	0.03%	−68%	11

Table 3: Decentralized vs centralized with 4-means inflow decomposition.

	Sub-networks	J_D	Δ_{J_D}	Δ_{CT}	CT
4-means	[1,2],[3,5,6],[7,8,9],[4]				
Tot net	[1,2,3,4,5,6,7,8,9]	5997			35
Main net	[1,2]	6489	7.6%	−98%	0.7
COOP	[1,2,4,6,8]	6411	6.4%	−82%	6
COOP1	[1,2,4,6,8],[5,6,9],[7,8,9],[4]	6411	6.4%	−82%	6
COOP2	[1,2,4,6,8],[5,6,9],[7,8,9],[4]	6382	6.0%	−79%	7
COOP3	[1,2,4,6,8],[5,6,9],[7,8,9],[4]	6017	0.3%	−70%	10.5

Table 4: Decentralized vs centralized with 3-means adjacent decomposition.

	Subnetworks	J_D	Δ_{J_D}	Δ_{CT}	CT
3-means	[5,7,8,9],[2,3,6],[4,1]				
Tot net	[1,2,3,4,5,6,7,8,9]	5997			35
Main net	[5,7,8,9]	6411	6, 9%	−82%	6
COOP	[2,4,5,7,8,9]	6175	2.9%	−52%	17
COOP1	[[2,4,5,7,8,9],[2,3,6],[4,1]	6173	2.9%	−52%	17
COOP2	[[2,4,5,7,8,9],[2,3,6],[4,1]	6102	1.6%	−47%	19
COOP3	[[2,4,5,7,8,9],[2,3,6],[4,1]	6093	1.7%	−48%	18

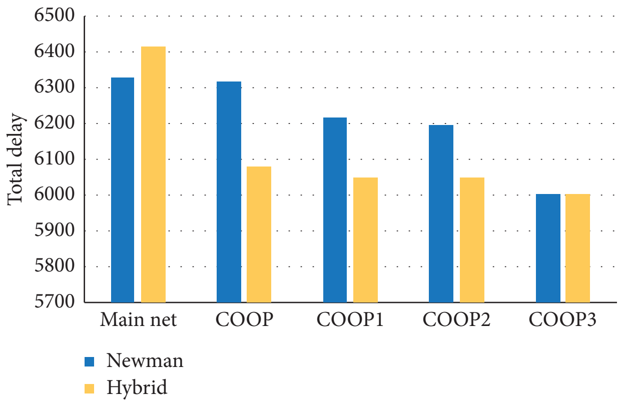
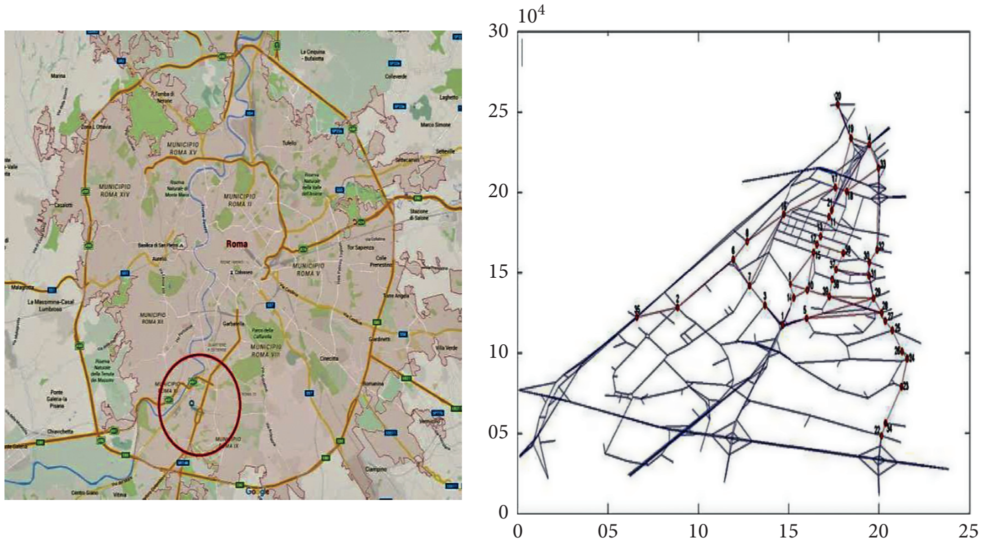


Figure 7: Comparison between Newman clustering and Hybrid clustering (the delay is given in seconds).



(a)

(b)

Figure 8: Real case study.

Hybrid approach based on Newman clustering are optimal but not significant since the clustering is composed by four subnetworks, two of them composed by only one node and the main network is representative of the whole network (see Figure 3). It can be noticed that if the cooperation increases the solution improves, though the control requires a higher time interval.

These preliminary results stress that it is possible to find a good compromise between subnetwork and efficiency. The comparison is performed between the centralized approach (TotNet) and decentralized approaches with different levels of cooperation (main net without cooperation and COOP/1/2/3 with increasing levels of cooperation). The objective function gets worse from 0.03% to 8.2%, against an improved calculation from 34% to 98%. It is evident that the cooperation provides optimal solutions which approximate the centralized control, though with a significant reduction in the calculation time. In Figure 7, a comparison between Newman clustering and the Hybrid clustering based on 4-means is reported. It highlights the fast convergence of Hybrid method that provides solutions approaching the optimal level also with limited cooperation (COOP). It is possible to understand that a clustering based on Priority is able to best represent the whole network.

5.4. Big Networks. The real case study has been conducted on a large-size network located in Rome, the area of Eur. Figure 8 reports the network; it presents 56 signalized intersections and 39 intersections under control, 194 links and 26 centroids. The large network simulates the supply and demand of the real network, the in flow demand is about 1000 [veic/h] from the input centroids of principal paths during a simulation time of 600 sec. All experiments are performed on a desktop computer with an Intel i5-3470 processor(3.2 GHz) with 8 GB of DDR3 RAM running 64-bit Windows 7. Even if each iteration takes 100 msec to complete, plus the communication overhead to calculate the objective unction by CTM$_U T$ requiring approximately 80sec, the SM (centralized version) would require 3,7 hours (i.e., 222 minutes) approximately to provide the result.

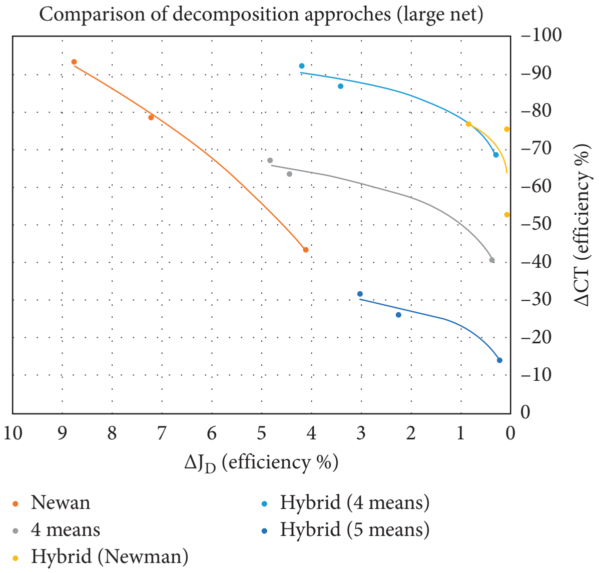


Figure 9: Trade-off efficiency-efficacy.

Table 5: Decentralized vs centralized with Hybrid decomposition based on Newman clustering.

	Sub-networks	J_D	Δ_{J_D}	Δ_{CT}	CT
Hybrid	[4 7 9 14 16 18 35 41 45 48], [1 2 3 5 6 8], [31 32 33 38 39 40 46 47], [20 21 36 37 42 43 44],[22 23 24 25 26 27 28 29 30]				
Tot net		15146			222
COOP	[4 6 7 9 14 16 18 19 20 28 31 35 41 45 48]	15177	0.21%	−85%	33
COOP1		15275	0.9%	−77%	51
COOP2		15159	0.09%	−75%	55
COOP3		15159	0.09%	−62%	106

Table 6: Decentralized vs centralized with Hybrid decomposition based on 4-means.

	Sub-networks	J_D	Δ_{J_D}	Δ_{CT}	CT
Hybrid	[1,2,4,5,8],[6,3],[7,9]				
Tot net	[1,2,3,4,5,6,7,8,9]	5997			35
Main net	[1,2,4,5,8]	6415	6.5%	−92%	3
COOP	[1,2,4,5,6,7,8]	6081	1.3%	−49%	17
COOP1	[1,2,4,5,6,7,8],[6,3],[7,9]	6051	0.8%	−48%	18
COOP2	[1,2,4,5,6,7,8],[6,3],[7,9]	6051	0.8%	−48%	18
COOP3	[1,2,4,5,6,7,8],[6,3],[7,9]	5999	0.03%	−45%	19

The Newman algorithm gives 4 clusters, and this is why K is set to 4 and 5. In Figure 9 reports the relation of the most promising algorithms. The Hybrid (Newman) is the Hybrid Method applied on the Newman Clustering, instead Hybrid (K-means) is the Hybrid Method applied on the K-means clustering. The Hybrid Method improves the solution both in efficiency and efficacy. The best performance is provided by the curve on top of the right side of the figure. As a matter of fact the solution with small Δ_{J_D} introduces the higher efficacy and with big Δ_{CT} higher efficiency.

It is important to notice that the decomposition approaches provide optimal results, and the worsening of the objective function can reach a maximum of 10% against a reduction of the computation time of 93%; see Figure 9.

The best results are given by the Hybrid Method considering the Newman clustering, which provides the same solution of the centralized method. The results of Hybrid decomposition based on Newman clustering are explicitly reported in Table 5. CT means the computation time and it is given in minutes. As a matter of fact, it presents discrepancies between the centralized and decentralized controllers of 0.09% and an improvement of 75% in the calculation time. The results of the Hybrid Method considering the Newman clustering are explicitly reported in Table 6. These results confirm that the clustering based on the topology of the network and flow information provides better performances. It is evident that for the signal setting problem, the decomposition method gives optimal results. Bring the

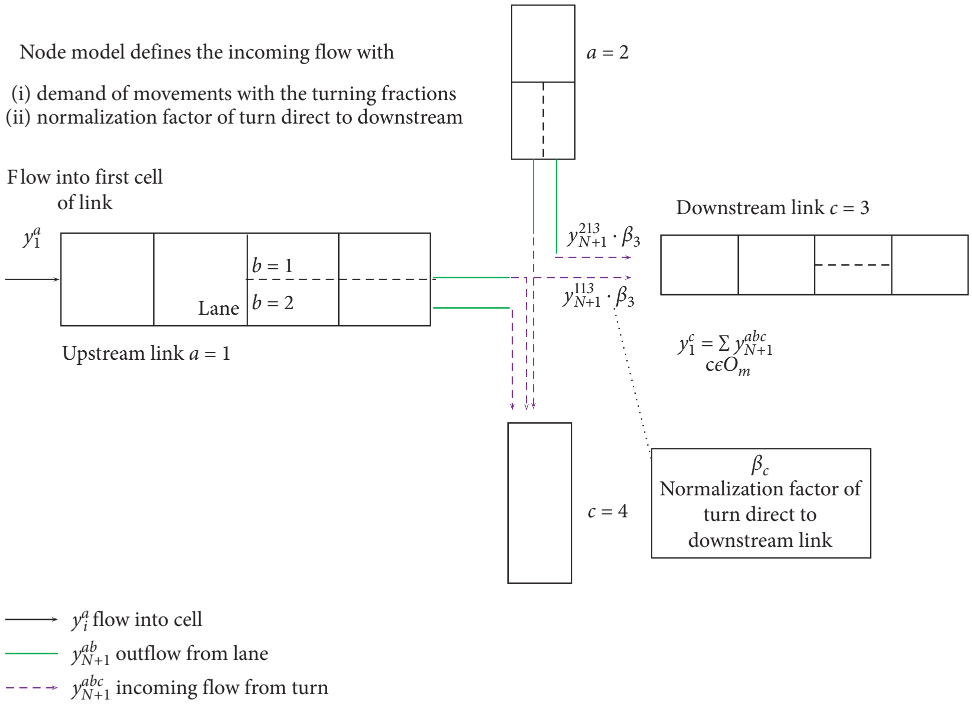

Node model defines the incoming flow with
(i) demand of movements with the turning fractions
(ii) normalization factor of turn direct to downstream
Flow into first cell of link
y_1^a
$a = 2$
$b = 1$
Lane
$b = 2$
Upstream link $a = 1$
$y_{N+1}^{213} \cdot \beta_3$
$y_{N+1}^{113} \cdot \beta_3$
Downstream link $c = 3$
$y_1^c = \sum_{c \epsilon O_m} y_{N+1}^{abc}$
$c = 4$
β_c
Normalization factor of turn direct to downstream link
y_i^a flow into cell
y_{N+1}^{ab} outflow from lane
y_{N+1}^{abc} incoming flow from turn

Figure 10: Representation of the node.

Principal steps of the K-means algorithm

(0) Given K the number of clusters;
(1) Set the centers, one for each cluster
(2) Assign each data element to the closest cluster
(3) Set the centers of each cluster on the mean of all elements in the cluster
(4) Repeat steps 2 and 3 until the centres will no longer move

Figure 11: Steps of K-means algorithm.

Surrogate Method to be an excellent tool for the optimization of signal setting problem for urban networks bypassing the critical aspect of the calculation time.

6. Conclusions

The signal setting problem is a nonconvex problem; usually to find an optimal solution for simple networks may take long time, when it is possible. It is here suggested a decentralized control, considering different clustering approaches for the network, together with a procedure to classify the nodes. The Surrogate Method is applied to solve the Traffic Signal Synchronization problem, for each identified subnetwork. This decomposition provides comforting results for other indexes concerning the composition is currently under study. The reduced computation time is not sufficient to run online; however, it is possible to run the decentralized approach more times in a day during principal time slices, while obtaining optimal results. An improvement of calculation time can give the possibility to apply the SM real time, although it is necessary to reduce the communication time for evaluating the total delay on the network. We are studying the possibility to dynamically change the network clustering based on traffic conditions and/or to fix a desired number of traffic lights for each subnetwork.

Principal steps of Newman algorithm

(1) Calculate the betweenness of each arc in the network
(2) Remove the edge with the highest betweenness
(3) Recalculate the betweenness of all edges that are connected with the removed edge
(4) Steps 2 and 3 are repeated until no edges remain.

Figure 12: Steps of Newman algorithm.

Appendix

A. CTM$_{\text{UT}}$ (Cell Transmission Model Urban Traffic)

In this section, the principal characteristics of CTM$_{\text{UT}}$ are summarized. Considering the urban channelized zone, CTM$_{\text{UT}}$ gives a correct representation of the urban dynamics with a loss of efficiency of about 4% respect to the traditional CTM. It is due to the representation of the microscopic aspects introduced by CTM$_{\text{UT}}$. The CTM$_{\text{UT}}$ models the congestion well and in fact predicts the mean speed and density well, introducing relative errors of about 4% – 10% with respect to microscopic models (i.e., SUMO and VISSIM). But when the calibration method is introduced, the error becomes 0%. The CTM$_{\text{UT}}$ divides the arterial into two zones: a zone in which the vehicles are split into specific lanes representing different turning movements (the downstream queue storage area) and a zone in which the turning movements are mixed (the upstream merging zone). Given N, the total number of cells of the link, and I, the number of cells belong to merging zone, the arterial is divided in two zones: an upstream merging zone ($1 \leq i \leq N - I + 1$) where the turning movements are mixed and a downstream queue storage zone ($N - I + 1 < i \leq N$) where vehicles are split into specific dedicated lanes, one for each different turning movements. The CTM_{UT} proposes a realistic model of turning movements considering the vehicular conflicts in channelized zone. The CTM_{UT} represents for each node the connection between the demand upstream intersections and the supplies downstream intersection, and it also considers the percentages of the demand of turns for every single lane. An example of the $\text{repCTM}_{\text{UT}}$ is reported in Figure 10.

A.1. Flow Conservation. The flow conservation equation used for CTM_{UT} is expressed as the difference between the inflows and the outflows of the earlier time interval. The following formulation allows to update the number of vehicles contained in each lane:

$$n_i^a(k+1) = n_i^a(k) + y_i^a(k) - y_{i+1}^a(k), \quad 1 \leq i \leq N. \tag{A.1}$$

The number of vehicles present in each cell i in period $k+1$ ($n_i^a(k+1)$) is equal to the sum of the number of vehicles present, in period k, in the cell i and vehicles moving from upstream cell $(i-1)$ to cell i and less than the number of vehicles moving from the cell i to the downstream cell $(i+1)$.

$$n_i^{ab}(k+1) = n_i^{ab}(k) + y_i^{ab}(k) - y_{i+1}^{ab}(k), \quad 1 \leq i \leq N. \tag{A.2}$$

A.2. Propagation into Link. Inflow of the cells belongs to the *merging zone* of link a:

$$y_i^a(k) = \min\begin{cases} n_{i-1}^a(k), Y_i^a(k) \\ \dfrac{w_i^a\left(F_i^a(k) - n_i^a(k)\right)}{v_i^a} \end{cases}, \quad 1 < i \leq N - I, \tag{A.3}$$

and for estimate flow into cell i of link a and direct to lane b, we have

$$y_i^{ab}(k) = \Phi_{ab} y_i^a(k), \quad 1 < i \leq N - I. \tag{A.4}$$

When $i = N - I + 1$, max *flow of downstream channelized zone* can be calculated by

$$\tilde{y}^{ab}(k) = \min\begin{cases} \Phi^{ab} n_{N-I}^a(k), \alpha^{ab} Y_{N-I+1}^a(k), \\ \dfrac{w_{N-I+1}^a\left(\alpha^{ab} F_{N-I+1}^a(k) - n_{N-I+1}^{ab}(k)\right)}{v_{N-I+1}^a}. \end{cases} \tag{A.5}$$

It permits to maximize the demand of upstream lane n_{N-I}^{ab} considering the maximum capacity of lane ($\alpha^{ab} Y_{N-I+1}^a$) and the necessary restriction to ensure that the inflow y^{ab} does not exceed the available capacity. ($w_{N-I+1}^a(\alpha^{ab} F_{N-I+1}^a(k) - n_{N-I+1}^{ab}(k))/v_{N-I+1}^a$) represents the total space available in the downstream cell i.

Because of conflict between turning vehicles and ahead vehicles, the total inflow of channelized zone can be formulated as follows:

$$y_{N-I+1}^a(k) = \min_{b \in B_m}\left\{\frac{\tilde{y}^{ab}(k)}{\alpha^{ab}}\right\}. \tag{A.6}$$

Inflow of each direction can be calculated as

$$y_{N-I+1}^{ab}(k) = \Phi_{ab} y_{N-I+1}^a(k). \tag{A.7}$$

To access the channelized zone, the vehicles directed to different turns may obstruct each other. For this reason, in oversaturated conditions, their behavior could block different movements. The following simple case considers only the interactions between left-turn (L) and through (T) movements incoming in cell 3. In order to give a realistic representation of the vehicular conflict occurring between neighboring turning movements when entering the channelized zone, the CTM_{UT} has a formulation based on the inflow of the blocking movement: specifically, this conflict is assumed to be proportional to the difference of the values of blocking inflow when passing from the merging zone to the channelized one.

$$y_3^{aL}(k) == \min\begin{cases} \tilde{y}_3^{aL}(k) \cdot \left[1 - \left(\dfrac{y_2^{aT}(k) - y_3^{aT}(k)}{y_2^{aT}(k)}\right) \cdot \left(1 - \Phi^{aL}\right)\left(1 - \alpha^{aL}\right)\right], \\ \Phi^{aL} n_2^a(k), \alpha^{aL} Y_3^a(k). \end{cases} \tag{A.8}$$

It considers the maximum flow given by (A.5) and substitutes (A.6) and (A.7) in model formulation. Moreover, capacity and supply constraints must be respected.

Channelized zone, for $N - I + 1 < i \leq N$ the inflow of each cell, can be represented as follows:

$$y_i^{ab}(k) = \min \begin{cases} n_{i-1}^{ab}(k), \alpha^{ab} Y_i^a(k) \\ \\ \dfrac{w_i^a\left(\alpha^{ab} F_i^a(k) - n_i^{ab}(k)\right)}{v_i^a}, \end{cases} \quad N - I + 1 < i \leq N. \tag{A.9}$$

For each $b \in B_a$ we can calculate the total inflow of cell i as

$$y_i^a(k) = \sum_{b \in B_a} y_{N+1}^{ab}(k), \quad N - I + 1 < i \leq N. \tag{A.10}$$

A.3.Demand Constraint for Conflicting Flow Interactions. The CTM_{UT} represents for each node the connection between the demand upstream intersections and the supplies downstream intersections, and it also considers the percentages of the demand of turns for every single lane.

This model is based on the same method used by Flotterod to define the demand constraint function (gap acceptance method). The CTM_{UT} avoids the problems and the complexities caused by the calibration the parameters (critical gap and follow-up times) for the capacity determination giving the possibility to apply the model for dynamic traffic. In this node, the model present different turning movements belonging to intersection as y_{N+1}^{abc}, where a is the arterial link upstream intersection, b is the lane belonging to the link, c is the link downstream intersection, and $N + 1$ represents the outgoing flow from the last cell of link a (uspstream intersection) direct to link c (downstream intersection). y_{N+1}^{a} represents the total outflow by link and y_{N+1}^{abc} is the flow divided for turning movements at intersection.

$$\widehat{\Delta}_{abc}(f_h, \ldots, f_{\text{cf}}) == \frac{\Delta_{abc}}{2} \left[\left(\frac{\left(\sum_{h=1}^{\text{cf}} Y_h - \sum_{h=1}^{\text{cf}} f_h \right)}{\sum_{h=1}^{\text{cf}} Y_h} \right) \exp \cdot \left(- \sum_{h=1}^{\text{cf}} f_h \left(t f^{\text{cf}} 1, 5^{\text{cf}} \right) \right) \right]. \tag{A.11}$$

In order to have a realistic capture of the potential capacity of minor flow $\widehat{\Delta}_{abc}$ when obstructed, we have proposed a formulation based on the demand of the minor flow Δ_{abc} considering the flow rate of traffic (the conflicting flow rate) that conflicts with a specific minor flow. Respect to the classical capacity determination method, the new formulation considers that the half of minor flow that wants to cross the intersection is inversely proportional to the crossing major flows f_h with respect to their capacities Y_h. The exponential elements depend by the flows fh, number conflict flows cf, and number of total flows tf for the movement direct to c (including the minor flow). Higher are the values of principal flows, conflict flows, and total flows, higher the minor flow is obstructed when try to cross the intersection. The following example shows the application of the previous equation considering one principal flow ($y^{abT} N + 1$ through flow T), one minor flow (y_{N+1}^{abL} left-turn L), one conflict point of crossing flows cf and total flow conflicts $tf = 2$ (i.e.., the minor flow plus all principal flows in conflict with it).

$$y_{N+1}^{abL}\left(y_{N+1}^{abT}\right) = \frac{y_{N+1}^{abL}}{2} \left[\left(\frac{\left(Y_{N+1}^{T} - y_{N+1}^{abT}\right)}{Y_{N+1}^{T}} \right) \exp\left(-y_{N+1}^{abT}\left(2^1 1, 5^1\right)\right) \right]. \tag{A.12}$$

This equation can be applied on the node model to evaluate the max capacity of minor flow. The capacity determination of minor streams of the CTM_{UT} produces a smoother representation of the flow when a conflict occurs at the intersection. Moreover, it gives a good accuracy respect to the other macroscopic and microscopic models. The CTM_{UT} can represent the behaviors of urban drivers, and the movements represent drivers that by mistake take the channelized lane for right turn as well.

B. Clustering Algorithms

B.1. K-means Clustering Algorithm. K-means is one of the simplest unsupervised learning algorithms able to solve the well known clustering problem; for more details, see [43]. Given the number of clusters K, the procedure follows a simple and easy way to cluster a given data set. For each cluster, a center is defined, so K centers are fixed. It is important to notice that choosing the centers is fundamental, if the centers change the solution changes. These centers should be placed in a cunning way. It would be better if the centers are set far from each other. Once all the centers are fixed, the other elements of data set will be associated with the closest center. If no element is pending, the first step can be considered completed and the early cluster is done. K new centers are calculated as barycenter of the clusters resulting from the previous step and the elements of data set are reassociated with these closest new centers. These two steps are performed until the centers will not move any more. The clustering procedure is summarized in Figure 11.

Obviously, the algorithm is also highly sensitive to the choice of initial centers. The K-means algorithm can be run multiple times to reduce this effect.

B.2. Newman Clustering Algorithm. In Girvan-Newman algorithm the number of clusters is not fixed a priori, and it detects the clusters by removing edges from the original network; for more details, see [44]. The cluster is given by the connected subnetwork obtained. The Girvan Newman algorithm is based on the betweenness concept. The betweennees of an edge represents the number of times that this edge is part of a shortest path in the network. The edge with the highest betweenness is removed. The betweenness of remaining edges is recalculated every time that an edge is removed. While removing these edges, the groups will be separated from the rest. The steps of algorithm are summarized in Figure 12.

Data Availability

The data used in the study are available on request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] L. Adacher, E. Cipriani, and A. Gemma, "The global optimization of signal settings and traffic assignment combined problem: a comparison between algorithms," *Advances in Transportation Studies*, vol. 36, pp. 35–48, 2015.

[2] H. S. Goliya and K. J. Nitin, "Synchronization of traffic signals: A case study-eastern ring road, indore," *International Journal of Advanced Technology in Civil Engineering*, vol. 1, no. 2, 2012.

[3] A. M. Khaki and P. J. Haghighat, "The impacts of traffic signal timings optimization on reducing vehicle emissions and fuel consumption by Aimsun and Synchro software's (case study: Tehran intersections)," *International Journal of Civil and Structural Engineering*, vol. 5, no. 2, 2014.

[4] A. Stevanovic, J. Stevanovic, k. Zhang, and s. Batterman, "Optimizing traffic control to reduce fuel consumption and vehicular emissions," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2128, no. 1, pp. 105–113, 2009.

[5] M. Xiaoliang, J. Junchen, and L. Wei, "Multi-criteria analysis of optimal signal plans using microscopic traffic models," *Transportation Research Part D: Transport and Environment*, vol. 32, pp. 1–14, 2014.

[6] X. Hu, J. Lu, W. Wang, and Z. Ye, "Traffic signal synchronization in the saturated high-density grid road," *Network Computational Intelligence and Neuroscience*, vol. 2015, Article ID 532960, 11 pages, 2015.

[7] G. E. Cantarella, S. De Luca, R. Di Pace, and S. Memoli, "Network signal setting design: meta-heuristic optimisation methods," *Transportation Research Part C: Emerging Technologies*, vol. 55, pp. 24–45, 2015.

[8] S. M. A. B. A. Islam and A. Hajbabaie, "Distributed coordinated signal timing optimization in connected transportation networks," *Transportation Research Part C: Emerging Technologies*, vol. 80, pp. 272–285, 2017.

[9] L. Adacher and A. Gemma, "A robust algorithm to solve the signal setting problem considering different traffic assignment approaches," *International Journal of Applied Mathematics and Computer Science*, vol. 27, no. 4, pp. 815–826, 2017.

[10] M. Di Gangi, G. E. Cantarella, R. Di Pace, and S. Memoli, "Network traffic control based on a mesoscopic dynamic flow model," *Transportation Research Part C: Emerging Technologies*, vol. 66, pp. 3–26, 2016.

[11] B. Park, C. J. Messer, and T. Urbanik, "Traffic signal optimization program for oversaturated conditions: genetic algorithm approach," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1683, no. 1, pp. 133–142, 1999.

[12] L. Tsai-Yun, "A fuel-based signal optimization model," *Transportation Research Part D: Transport and Environment*, vol. 23, pp. 1–8, 2013.

[13] S. Dazhi, R. F. Benekohal, and S. T. Waller, "Bilevel programming formulation and heuristic solution approach for dynamic traffic signal optimization," *Computer-Aided Civil and Infrastructure Engineering*, vol. 21, pp. 321–333, 2006.

[14] D. Manolis, T. Pappa, C. Diakaki, I. Papamichail, and M. Papageorgiou, "Centralised vs decentralised signal control of large-scale urban road networks in real time: a simulation study," *IET Intelligent Transport Systems*, vol. 12, no. 8, 2018.

[15] A. Petrillo, A. Salvi, S. Santini, and A. S. Valente, "Adaptive multi-agents synchronization for collaborative driving of autonomous vehicles with multiple communication delays," *Transportation Research Part C: Emerging Technologies*, vol. 86, pp. 372–392, 2018.

[16] M. Tajalli and A. Hajbabaie, "Distributed optimization and coordination algorithms for dynamic speed optimization of connected and autonomous vehicles in urban street networks," *Transportation Research Part C: Emerging Technologies*, vol. 95, pp. 497–515, 2018.

[17] M. K. Tan, H. S. E. Chuo, R. K. Y. Chin, K. Beng Yeo, and T. K. TeoK, "Optimization of traffic network signal timing using decentralized genetic algorithm," in *Proceedings of the 2017 IEEE 2nd International Conference on Automatic Control and Intelligent Systems (I2CACIS)*, pp. 62–67, Kota Kinabalu, Malaysia, 2017.

[18] L. Adacher, A. Gemma, and G. Oliva, "Decentralized spatial decomposition for traffic signal synchronization," *Transportation Research Procedia*, vol. 3, pp. 992–1001, 2014.

[19] A. Chow, R. Sha, and S. Li, "Adaptive control strategies for urban network traffic via a decentralized approach with user-optimal routing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 4, 2019.

[20] A. H. F. Chow and R. Sha, "Performance analysis of centralized and distributed systems for urban traffic control," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2557, no. 1, pp. 66–76, 2016.

[21] T. Le, P. Kovács, N. Walton, H. L. Vu, L. L. H. Andrew, and S. S. P. Hoogendoorn, "Decentralized signal control for urban road networks," *Transportation Research Part C: Emerging Technologies*, vol. 58, pp. 431–450, 2015.

[22] M. Rinaldi, W. Himpe, and C. M. J. Tampère, "A sensitivity-based approach for adaptive decomposition of anticipatory network traffic control," *Transportation Research Part C: Emerging Technologies*, vol. 66, pp. 150–175, 2016.

[23] S. Timotheou, C. G. Panayiotou, and M. M. Polycarpou, "Distributed traffic signal control using the cell transmission model via the alternating direction method of multipliers," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 919–933, 2015.

[24] J. GaoGao, Y. Shen, J. Liu, M. Ito, and N. Shiratori, "Adaptive traffic signal control: deep re-inforcement learning algorithm with experience replay and target network," 2017, https://arxiv.org/abs/1705.02755.

[25] L. Hoogendoorn, Y. Lv, and F.-Y. Wang, "Traffic signal timing via deep reinforcement learning," *IEEE/CAA Journal of Automatica Sinica*, vol. 3, no. 3, pp. pp247–254, 2016.

[26] S. S. Mousavi, M. Schukat, and E. Howley, "Traffic light control using deep policy-gradient and value-function-based reinforcement learning," *IET Intelligent Transport Systems*, vol. 11, no. 7, pp. 417–423, 2017.

[27] M. Muresan, L. Fu, and G. Pan, "Adaptive traffic signal control with deep reinforcement learning-an exploratory investigation," in *Proceedings of the 97th Annual Meeting of the Transportation Research Board*, Washington, DC, USA, 2018.

[28] Y. Gong, M. Abdel-Aty, Q. Cai, and M. Rahman, "Decentralized network level adaptive signal control by multi-agent deep reinforcement learning," *Transportation Research Interdisciplinary Perspectives*, vol. 1, 2019.

[29] C. Priemer, “A decentralized adaptive traffic signal control using V2i communication data,” *Intelligent Transportation Systems*, vol. 33, pp. 1–6, 2009.

[30] N. J. Goodall, B. L. Smith, and B. Park, “Traffic signal control with connected vehicles,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2381, no. 1, pp. 65–72, 2013.

[31] P. Varaiya, “Max pressure control of a network of signalized intersections,” *Transportation Research Part C: Emerging Technologies*, vol. 36, pp. 177–195, 2013.

[32] J. Reilly and A. M. Bayen, “Distributed optimization for shared state systems: applications to decentralized freeway control via subnetwork splitting,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3465–3472, 2015.

[33] X. He, X. Chen, C. Xiong, Z. Zhu, and L. Zhang, “Optimal time-varying pricing for toll roads under multiple objectives: a simulation-based optimization approach,” *Transportation Science*, vol. 51, no. 2, pp. 412–426, 2016.

[34] K. Gokbayrak and C. G. Cassandras, “Generalized surrogate problem methodology for online stochastic discrete optimization,” *Journal of Optimization Theory and Applications*, vol. 114, no. 1, pp. 97–132, 2002.

[35] L. Adacher and M. Tiriolo, “A distributed approach for traffic signal synchronization problem,” in *Proceedings of the Third International Conference on Mathematics and Computers in Sciences and in Industry (MCSI)*, pp. 191–196, Chania, Greece, 2016.

[36] C. F. Daganzo, “The cell transmission model: a dynamic representation of highway traffic consistent with the hydrodynamic theory,” *Transportation Research Part B: Methodological*, vol. 28, no. 4, pp. 269–287, 1994.

[37] L. Adacher and M. Tiriolo, “A macroscopic model with the advantages of microscopic model: a review of cell transmission model’s extensions for urban traffic networks,” *Simulation Modelling Practice and Theory*, vol. 86, pp. 102–119, 2018.

[38] X. Ai, J. Yu, and S. Yang, “Optimization of signal timing schedule of traffic network based on CTM,” in *Proceedings of the 33rd Chinese Control Conference*, pp. 3367–3372, Nanjing, China, 2014.

[39] H. S. E. Chuo, M. K. Tan, B. L. Chua, R. K. Y. Chin, and K. T. K. Teo, “Computation of cell transmission model for congestion and recovery traffic flow,” in *Proceedings of the 2016 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia)*, pp. 1–4, Seoul, South Korea, 2016.

[40] L. Zhang, Y. Yin, and Y. Lou, “Robust signal timing for arterials under day-to-day demand variations,” *Transportation Research Record*, vol. 2192, pp. 156–166, 2010.

[41] S. Zhao, S. Liang, H. Liu, and M. Ma, “CTM based real-time queue length estimation at signalized intersection,” *Mathematical Problems in Engineering*, vol. 2015, Article ID 328712, 13 pages, 2015.

[42] H. K. Lo and A. H. F. Chow, “Control strategies for oversaturated traffic,” *Journal of Transportation Engineering*, vol. 130, no. 4, pp. 466–478, 2004.

[43] M. P. Malakapalli and C. J. Messer, “Enhancements to the PASSER II-90 delay estimation procedures,” *Transportation Research Record*, vol. 1421, pp. 94–103, 1983.

[44] M. Xu, K. An, Z. Ye, Y. Wang, J. Feng, and J. Zhao, “A Bi-level model to resolve conflicting transit priority requests at urban arterials,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 4, pp. 1353–1364, 2019.