Consensus-Based Cooperative Control Based on Pollution Sensing and Traffic Information for Urban Traffic Networks

Antonio Artuñedo *, Raúl M. del Toro and Rodolfo E. Haber

Centre for Automation and Robotics (CSIC—UPM), Ctra. Campo Real Km. 0.2, Arganda del Rey, 28500 Madrid, Spain; raul.deltoro@car.upm-csic.es (R.M.d.T.); rodolfo.haber@car.upm-csic.es (R.E.H.)
* Correspondence: antonio.artunedo@car.upm-csic.es; Tel.: +34-918-711-900

Academic Editors: Stefano Mariani, Francesco Ciucci, Dirk Lehmhus, Thomas Messervey, Alberto Vallen and Stefan Bosse
Received: 6 February 2017; Accepted: 21 April 2017; Published: 26 April 2017

Abstract: Nowadays many studies are being conducted to develop solutions for improving the performance of urban traffic networks. One of the main challenges is the necessary cooperation among different entities such as vehicles or infrastructure systems and how to exploit the information available through networks of sensors deployed as infrastructures for smart cities. In this work an algorithm for cooperative control of urban subsystems is proposed to provide a solution for mobility problems in cities. The interconnected traffic lights controller (TLC) network adapts traffic lights cycles, based on traffic and air pollution sensory information, in order to improve the performance of urban traffic networks. The presence of air pollution in cities is not only caused by road traffic but there are other pollution sources that contribute to increase or decrease the pollution level. Due to the distributed and heterogeneous nature of the different components involved, a system of systems engineering approach is applied to design a consensus-based control algorithm. The designed control strategy contains a consensus-based component that uses the information shared in the network for reaching a consensus in the state of TLC network components. Discrete event systems specification is applied for modelling and simulation. The proposed solution is assessed by simulation studies with very promising results to deal with simultaneous responses to both pollution levels and traffic flows in urban traffic networks.

Keywords: air pollution monitoring; cooperative systems; discrete event systems; traffic sensing; urban traffic network

1. Introduction

Nowadays, the challenge of making sustainable and environmentally friendly cities is driving the activities of many stakeholders such as local and national authorities, the private sector, environmental pressure groups, university research institutes, and neighborhood associations. They are jointly promoting system-of-systems-based solutions for smart cities [1–3]. From a physical and system engineering standpoint, a city or an urban region can be considered as a physical system composed of several coupled physical subsystems with thousands of networked sensors and actuators (e.g., transportation, energy distribution systems, water supply systems, etc.). Thereby, the overall dynamic performance of the city can be assessed on the basis of the emerging behavior of these coupled physical subsystems having three important features: complexity, uncertainty and heterogeneity. Strict requirements for sustainability, resource availability, and optimality are then imposed. The ability to anticipate and control daily situations (e.g., traffic congestion and high levels of pollution) and unexpected events (e.g., power failures, traffic accidents and exceptional climate conditions) is a key
issue to deal with complex and potentially unstable conditions. In addition, many sensors, actuators, communication systems, and computing platforms are available in each subsystem. Therefore, the development of integrated methods based on these sensory data, signal processing strategies and decision-making procedures is a challenge with no straightforward mathematical tractability, and difficult representation and evaluation by traditional modeling and simulation tools. New computing platforms are able to measure representative variables, perceive physical events that may occur in the environment, preprocess the events with embedded computing capabilities and send information to networked and distributed applications that can perform more complex tasks for managing complete urban systems.

Current research on smart cities aims to integrate urban subsystems [3]. These subsystems are usually from different domains or with different timing and bandwidths. Two examples are the pollution measurement systems and the traffic control and monitoring systems. The heterogeneity of the systems imposes new challenges in model-based design approaches. On the other hand, the increasing number of sensors, actuators, communication systems and computing devices already deployed in cities, enable new applications to go beyond particular solutions and cover wider urban systems and scenarios. These challenges have been recently addressed in other application fields by using machine learning techniques, data mining or artificial cognition-based strategies [4,5].

The System-of-Systems (SoS) paradigm emerges from the need to address the complexity of interconnected systems. In this context, systems are independently designed, evolve independently and can operate autonomously [6,7]. Nevertheless, the constituent systems of a SoS have their own individual goals while the whole SoS has global goals. The expected goal of operating sub-systems that work together is a better overall yield of the sum of the systems in comparison with each individual system. Their joint operation provides new functionalities; subsystems are also operationally independent and autonomous, so interactions between them are generally asynchronous and can be represented as discrete-event models [8]. This heterogeneity of the systems brings new challenges to the model-based design approach.

How can cooperative control and sensing systems be designed on the basis of SoS paradigm to provide new sensor-based solutions for smart urban systems?

The design of cooperative control strategies for urban subsystems can be supported on centralized and distributed coordination schemes [9]. In a centralized coordination scheme, it is assumed that each system has the ability to communicate with a central unit or share information via a fully connected network. On the contrary, distributed coordination schemes reduce communication requirements using local neighbor-to-neighbor interactions, thus improving scalability, flexibility, reliability and robustness.

Consensus is a fundamental issue to be considered for designing cooperative control for distributed multi-agents coordination. Some reports on cooperative control of sets of multiple subsystems such as vehicles, robots and rovers are inspired in consensus-based approaches [10]. Nanayakkara et al. [8], suggested a consensus-based control using the cooperative-control paradigm, to increase the expected benefits of the resulting system. These authors proposed to create a set of systems each pursuing its own objectives as well as their common goals, using communications between subsystems.

Consensus algorithms have been widely applied to distributed cooperative systems such as vehicle groups [9,11,12], swarm robots [10], traffic control systems [13–15], traffic speed density prediction [16] and wireless-sensor networks [17], among others. In these cases, consensus control enables each individual system or agent to ascertain the global goal and then, based on the state of the other system, to make decision on the best control signal to achieve the corresponding goal.

The problem of traffic optimization in urban environments based on city-pollution information can be addressed by a consensus-based approach between different control agents [18]. Focused on a similar problem (urban-traffic management and control), Castro et al. proposed a multilayer distributed model for predictive control of urban traffic. The control system was composed of local and global layers of controls where each control agent is responsible for traffic lights at one intersection.
The system uses the principles of consensus between control agents and behavioral coordination, in order to conciliate local and global control objectives.

In another work reported in [19], the distributed control principle was applied to a network of intersections, with the aim of controlling the traffic at a macroscopic level. Wuthishuwong et al. implemented a discrete time-consensus algorithm, to coordinate the gross traffic density of an intersection and its neighborhoods in the network. The traffic density at each intersection was collected by using Vehicle-to-Infrastructure (V2I) communication, and the information was distributed to the neighborhood via Infrastructure-to-Infrastructure (I2I) communication link. Each intersection performed as a centralized controller that computed the control signal using both, own and neighborhood information on traffic density. Intersections were considered individual nodes, in order to apply the consensus strategy, and each node shared information on its traffic density and coordination variables.

In this work, the System-of-Systems (SoS) engineering paradigm sets the foundation for developing a cooperative control strategy applied to an urban scenario. For this purpose, the air-pollution information service and the traffic control subsystem are put together, and the control system makes use of available information to modify or adjust traffic-light cycles at intersections. Due to the distributed nature of the main subsystems, a distributed consensus-based control method is developed. Pollution levels and traffic densities are considered inputs of the proposed control strategy based on subsystems cooperation. It should be noted that sources of air pollution in cities are not only coming from vehicle emissions but also from industrial activity, domestic fuel burning, boilers for heating, etc.; which increases the complexity for considering the whole problem. Modeling and simulation are not straightforward according to the complexity of main subsystems, instead both can be powerful tools to address the problem and generate the basic knowledge on this field. The discrete event systems specification (DEVS) is applied for modeling and simulation in this work. The main rationale for using DEVS relies on the suitability of this framework for modeling complex dynamical systems and their interactions [20].

From the best of authors’ knowledge the main contribution of this work is the design and evaluation of a strategy for improving the performance of urban traffic networks in specific regions of a city taking into account traffic and pollution sensory information. A consensus-based cooperative control method is proposed by taking advantage of new developments on technologies and concepts for smart cities. This paper is organized as follows: in Section 2, a brief review of pollution and traffic sensing systems is presented. Section 3 introduces the problem and the proposed solution. Section 3.1 deals with the modeling technique for representing the behavior of the main subsystems components. Section 3.2 addresses the consensus-based cooperative control and presents the proposed cooperative control solution. Sections 3.3 and 3.4 show open-loop and closed-loop simulation results. Discussions on the main results are given in Section 4. Finally, the conclusions are presented in Section 5.

2. Review of Pollution and Traffic Sensing Systems

In the last years, the interest in collecting data in cities has increased substantially. The current trend is to make a better use of public resources while increasing the quality of the services that citizens can access [21]. In particular, urban air quality sensing and traffic information are attracting a lot of research because of growing population in cities. In this section, a brief review about pollution and traffic sensing systems in cities is presented.

2.1. Pollution Sensing Systems

The basis for the consensus-based cooperative control is the pollution sensing system. Current sensors for pollution concentration measurement are based on different chemical or physical principles such as chemoresistance, solid electrolyte, absorption, etc. These sensors can measure different pollutants such as O2, O3, CO, carbon dioxide (CO2), mono-nitrogen oxides (NOx), PM, and volatile organic compounds (VOCs), in different range of sensitivity, selectivity and response time. Depending
on the application, the sensor can be chosen with the required specific parameters e.g., in [22] an O\textsubscript{2} with a sampling time of 1.6 s and NO\textsubscript{2} with a sampling time of 41 s are selected.

Traditional approaches for pollution monitoring in urban or rural areas are based on networks of fixed stations for measuring air quality. Typically, these networks can provide detailed data but limited to the location of the stations. This makes that modelling approaches are usually used for obtaining representative pollution information in the area of interest [23].

The current challenges in the assessment of human exposure to pollution are how to make available reliable pollution data in cities, and how to handle large amounts of data from high resolution sensors and process it. Most recent methods propose the deployment of low-cost sensors that can provide emission information that allows the development of mitigation strategies [23]. In contrast with traditional approaches, the advances in sensor compactness, robustness, and wireless communications let sensor networks to be remotely managed, to easily transmit collected data and to report high spatial-resolution information in near-real time [24]. Data from these new pollution information systems enable a better pollution assessment and the development of new control strategies for pollution reduction [23].

2.2. Traffic Sensing Systems

The performance and safety of traffic control and management systems depend directly on traffic sensing systems. As in the case of pollution sensing systems, in the last decade there has been significant progress in the application of computer, sensing and communication technologies to traffic management and sensing systems [25]. Nowadays, there exist different technologies for traffic sensing that are suitable for several scenarios depending on the installation of the sensors, maintenance, performance, atmospheric conditions, etc. Some of them are: inductive loops, radio-frequency identification (RFID), microwave radar, acoustic, magnetic, ultrasonic and video image processor (VIP) [25]. Inductive loop detectors are commonly used for traffic sensing in cities and highways due to their characteristics: they are suitable for different weather conditions and different traffic volumes [26]. They also provide consistent and accurate measurements. Induction loops are also capable of measuring vehicle speed and vehicle classification, what make them appropriate and reliable for provide data to traffic control systems [27].

3. General Scenario and the Proposed Solution

We have defined a general scenario based on a smart mobility application with the aim of improving the performance of urban traffic networks in urban regions of a city. The scenario is based on an emission control scheme proposed by Andó et al. [28]. These authors suggested the idea of a vehicles emission control scheme for traffic flow control as a possible solution to address the problem of urban air quality (see Figure 1). However, they do not provide specific solution for designing the emission controller. Authors proposed the number of allowed vehicles as the main control signal whereas the difference between the innocuous allowed CO level and the CO measured level is the input signal.

![Figure 1. The schematic concept of a vehicle emission control system.](image)

A new control scheme and a procedure that can be integrated into the Adaptive Traffic Control System (ATCS) of an urban area are proposed in this work [29]. The ATCS adjusts, in real time, signal
timing plans based on the current traffic conditions, demand, system capacity, among other conditions. In the control scheme suggested in this work, we focused on designing the adaptive component $\Delta u$ for traffic lights cycles $U$. The traffic lights cycles can vary on the basis of the following relationship $U_0 = U_0 + U_0 \Delta u$, where $U_0$ is the value set by other components of the ATCS or by traffic engineers.

Figure 2 depicts a general diagram of the scenario and the strategy proposed in this work. Interconnected traffic lights control (TLC) network adapts the traffic-light cycles based on air pollution data provided by the city information service and local traffic measurements. The air pollution information service provides data ($\xi$) from the city pollution sensing system. Moreover, traffic information ($x$) is given by the vehicle detection sensor placed at each intersection of the urban traffic network. The traffic lights cycles ($U$) are updated by TLC units which make decisions based on a consensus variable ($\varepsilon$) derived from the consensus-based control method.

The following subsections address the modeling, control design, and simulation of the whole system.

3.1. Modeling

In order to study the dynamic behaviour and the interaction between the different components of the scenario, Discrete Event Systems Specification (DEVS) formalism is selected as a framework for SoS modeling and simulation. This formalism can be used for modeling and simulating complex dynamical systems and their interactions [20]. The DEVS modeling-based approach enables specification of basic models and how they are connected together. These basic models, known as atomic models, are modular systems with inputs (through input ports), changing states, and outputs (through output ports) running in a time frame. The couplings between atomic models generate coupled models.
In this work we used Parallel DEVS formalism (PDEVS) which solves collisions between internal and external transitions allowing all components to be activated and to send their output to other components. A basic PDEVS model can be represented as:

\[ M = \langle X_M, Y_M, S, \delta_{\text{ext}}, \delta_{\text{int}}, \delta_{\text{con}}, \lambda, t_a \rangle \]  

where,

- \( X_M = \{ (p, v) \mid p \in \text{Inputs}, \ v \in X_p \} \) is the set of input ports (p) and values (v)
- \( Y_M = \{ (p, v) \mid p \in \text{Outputs}, \ v \in Y_p \} \) is the set of output ports (p) and values (v)
- S is the set of states
- \( \delta_{\text{int}} : S \rightarrow S \) is the internal transition function
- \( \delta_{\text{ext}} : Q \times X_M^b \rightarrow S \) is the external transition function
- \( \delta_{\text{con}} : Q \times X_M^b \rightarrow S \) is the confluent transition function
- \( Q = \{ (s, e) \mid s \in S, \ e \in [0, t_a] \} \) is the total state set and e is the time since the previous transition
- \( \lambda : S \rightarrow Y \) is the output function
- \( t_a : S \rightarrow R^+ \cup \infty \) is the time advance function

Basic PDEVS models have a bag of inputs. A bag is a set with possible multiple occurrences of its elements e.g., \{a,b,a,c\} [30]. PDEVS also introduces the confluent transition function (\( \delta_{\text{con}} \)), which decides the next state in case of collision between external and internal events (\( e = t_a(s) \)) without the need for a priority scheme. This option provides a complete control over the collision behavior. Instead of serializing model behavior through the select function at the coupled-model level, PDEVS leaves this decision to the individual component. After modeling, it is necessary to conduct simulation studies. DEVS simulation methods generate the corresponding behaviors of each model, which are trajectories that can even reach illegal states by following execution protocols. This modeling paradigm serves to represent each subsystem as an atomic model that can be coupled with other model, forming coupled models.

This section addresses the representation of the system based on a conceptual model and the data flows between subsystems. Every system is represented as an atomic discrete system. The coupling of atomic models forms a coupled model. Figure 3 depicts the structure of a DEVS model of the general scenario and how the subsystems are connected.

![Figure 3. Structure of DEVS models.](image)

In the above model, the following systems are defined as atomic models:

- Traffic-light control unit (TLC)
- Pollution-monitoring service
• Traffic system (i.e., road network, vehicles, traffic lights, etc.)
• Other pollution sources

The network of interconnected TLCs shares information (consensus variable, \( \varepsilon_i \)) to achieve a consensus between each other. This network is modeled as a coupled model. Each TLC is connected with a Traffic system to adapt the cycle length (\( u_i \)) of the corresponding traffic lights. The output of the Traffic subsystem is the traffic information (i.e., traffic flow at every intersection, \( x_i \)) and the pollution data from urban road traffic (\( E_v \)). The system called “Other pollution sources” emulated the pollution contribution of other urban subsystems (e.g., building heating systems, other transportation systems, etc., \( E_o \)). The Pollution monitoring system is represented by the urban pollution-measurement system that brings information on air quality to the TLC network (\( \xi \)). Data flows illustrated in Figures 2 and 3 are summarized in Table 1.

Table 1. Description of data flows.

| Variable | Description |
|----------|-------------|
| \( E_v \) | Vehicle emissions monitoring |
| \( E_o \) | Other emissions |
| \( \xi \) | Area-wide air-quality information. Includes current pollution-status details for a given geographic area. |
| \( \varepsilon \) | Consensus variable that represents the TLC dynamics (see Section 3.2.1) |
| \( x \) | Processed traffic-detector data which allows derivation of traffic-flow variables (density, occupancy, flow measures, etc.). It can be represented as a vector that refers to the signal of each sensor. |
| \( u \) | Data flow contains the system configuration data for a traffic signal controller. It includes the parameters required to reconfigure its operations. |

3.2. Consensus-Based Cooperative Control Design

Consensus algorithms are thought to be distributed algorithms that assume only neighbor-to-neighbor interactions between subsystems. Subsystems update their information states based on their neighbors. The aim is to design a control strategy for updating the information states of all of the subsystems in the network, so that they converge to a common value.

The main rationale of a consensus-based procedure is to impose similar dynamics on the information states of each subsystem of the network [9]. Depending on the typology of the network for connecting subsystems, two different approaches can be applied. Firstly, if the network allows continuous communication, then the information state is modeled using differential equations. Secondly, if the data are communicated through discrete packets, the information state is represented by difference equations.

The topology for interactions in a network of agents is represented using a graph \( G = (V, E) \), where \( V = \{1, 2, \ldots, n\} \) is a finite none-empty node set and \( E \subseteq V \times V \) is an edge set of ordered nodal pairs, called edge. The edge \((i, j)\) in the edge set of a graph denotes that node \( j \) can obtain information from node \( i \). The neighbors of a node \( i \) are denoted by \( N_i = \{ j \in V : (i, j) \in E \} \). An iterative form of the consensus algorithm to reach an agreement in relation to the state of \( n \) integrator nodes with dynamics \( x_i = u_i \) can be represented in discrete time as follows:

\[
    x_i(k + 1) = x_i(k) + \lambda \sum_{j \in N_i} a_{ij}(x_j(k) - x_i(k)), \quad k \in \mathbb{N}
\]

where, \( a_{ij} \) is the entry in row \( i \) and column \( j \) of the adjacency matrix \( A_n \in \mathbb{R}^{n \times n} \) associated with \( G \), and \( x_i \) is the information state of the \( i \)-th system. A consequence of this equation is that the information...
state \( x_i(k) \) of the system \( i \) is driven toward the information states of its neighbors. The consensus is achieved if, for all \( x_i(0) \) and for all \( i, j = 1, \ldots, n \), \( |x_i(k) - x_j(k)| \rightarrow 0 \), as \( k \rightarrow \infty \).

The procedure to apply the consensus-based control and the results are then presented. The goal of the control system is to achieve a consensus between the intersections related to an estimation of local emissions (consensus state variable) by adapting local traffic-light cycles. The consensus-based decision-making takes into consideration the air-pollution state, provided by the air pollution information service, and the local traffic state. Specifically, a discrete consensus-based control algorithm is applied to control the coordination of the TLC network. The main steps for designing the control system are described in the following subsections.

3.2.1. System Dynamics

A consensus variable that represents the system dynamics should be defined. Each TLC uses local traffic and pollution sensing information from a pollution information service. Therefore, an estimation of the pollutant concentration at each intersection as the consensus state variable is taken. The dynamics of each intersection is the same and it is assumed to behave as a local linear system. The discrete dynamic model of the node, that includes traffic information and pollution sensing data, is defined by the following recursive equation:

\[
\varepsilon_i(k+1) = \varepsilon_i(k) + a_i \xi(k-n) + \beta x_i(k-m) + \gamma \Delta u_i
\]

where,

- \( \varepsilon_i(k) [\text{gNOx/m}^3] \) is the state of the system \( i \) at instant \( k \). The state of the system is related to the pollution levels and the traffic state at each intersection. The pollutant considered in this study is NOx (mononitrogen oxides).
- \( \xi(k-n) [\text{gNOx/m}^3] \) is a system input. It contains the pollution information provided to the TLCs at instant \( k-n \) where \( n \) is the delay between pollution production and pollution information reception by the TLCs.
- \( a_i \) is a dimensionless parameter that represents the contribution of intersection \( i \) to overall city pollution. It is computed on the basis of the maximum occupancy of intersection \( i \) over the total maximum occupancy.
- \( x_i(k-m) [\text{veh}] \) is a system input. It contains the summation of vehicle queues at every approach of the intersection controlled by TLC\(_i\). \( m \) represents the delay between traffic queue measurement and traffic information reception at TLC\(_i\).
- \( \beta [\text{gNOx/veh/m}^2] \) is the pollutant emissions of a given traffic queue at an intersection. \( \beta = \frac{qF}{100\Delta t} \), where \( q \) is the emission factor of the pollutant (gNOx/veh/km), \( F \) is the dispersion factor of the pollutants (s/m\(^2\)) and \( \Delta t \) is the simulation step [31].
- \( \Delta u_i [% \text{T.L. cycle}] \) is the consensus-based cooperative control output of TLC\(_i\). The control signal is a percentage change of the cycle length of the traffic lights with respect to the initial cycle length in a limited range. This component is defined later (control law design).
- \( \gamma [\text{gNOx/m}^3/\% \text{T.L. cycle}] \) represents the influence of the pollutant emissions on traffic-light timing. \( \gamma = \beta \gamma' \), where \( \gamma' [\text{veh}/\% \text{T.L. cycle}] \) is a parameter that represents the queue length with respect to changes in traffic-light cycle lengths. Therefore there is a proportional relationship between \( \gamma \) and \( \gamma' \).
3.2.2. Design of a Consensus-Based Control Strategy

The control strategy adopted for each TLC (nodes of the network) makes use of pollution and traffic information as well as information from neighbors. Taking into account the dynamic model considered in (2) and applying the consensus control described in (1), the control strategy is represented by:

\[ \Delta u_i(k) = -\frac{1}{\gamma} \left( a_i \xi_i(k-n) + \beta x_i(k-m) + \lambda \sum_{j \in N_i} a_{ij} (\varepsilon_i - \varepsilon_j) \right) \quad (4) \]

where,

- \( \lambda \) is a parameter that refers to system stability. If \( \lambda \in (0, \theta^{-1}) \), where \( \theta \) is the maximum degree of the graph, consensus convergence is guaranteed in a connected graph [32].
- \( a_{ij} \) is the corresponding value of the adjacency matrix.

This control law contains three components, shown in Table 2.

| Component | Expression | Description |
|-----------|------------|-------------|
| 1         | \( a_i \xi_i(k-n) \) | Feed-forward action related to local pollution data. |
| 2         | \( \beta x_i(k-m) \) | Feed-forward action related to local traffic data. |
| 3         | \( \lambda \sum_{j \in N_i} a_{ij} (\varepsilon_i - \varepsilon_j) \) | Consensus-based control signal that makes use of information from the neighbor of each network node. |

In order to avoid large dissimilarities with the original traffic-light cycle lengths, the values of the control input \( \Delta u_i \) are constrained to a variation of \( \pm 50\% \) over the initial value.

3.3. Open-Loop Simulation for Test Scenario: Pollution and Traffic-Based Control Are Switched Off

The scenario is based on an urban-like road network depicted in Figure 4. In order to simplify the description of the proposed solution, we considered a test scenario composed of four signalized traffic intersections (junctions J1, J2, J3 and J4) and interconnected TLCs. Vehicles are moving on the road network, following predefined routes randomly generated for imitating urban traffic. The vehicles included in the scenario are of the same type and use the same car-following model which is a version of the model defined by Krauß et al. [33]. In this version, different deceleration capabilities of the vehicles are handled without violating safety (the original model allowed for collisions in this case), and the formula for safe velocity is adapted to maintain safety when using the current Euler-position update rule. The parameters defined for vehicles and the car-following model are shown in Table 3.
The number of trips is defined by the repetition rate (number of trips per second). The repetition rate is calculated randomly by using a normal distribution with a mean of 5 and standard deviation of 0.25. Routes were generated, ensuring a minimum straight-line distance between the start and the end edges of a 170-m trip.

In this scenario, TLCs work with a fixed timing \( u_i^0 \) (\( \Delta u_i = 0 \)), because there is no control system running. This initial timing \( (u_i^0) \) is shown in Figure 5. The simulation ran 2 h (7200 s) with a simulation step size of 1 s.

Table 3. Vehicle and car-following model parameters.

| Variable                  | Value  |
|---------------------------|--------|
| Length (m)                | 5.00   |
| Minimum gap (m)           | 2.50   |
| Maximum speed (m/s)       | 55.56  |
| Maximum acceleration (m/s²) | 2.60  |
| Maximum deceleration (m/s²) | 4.50  |
| Imperfection              | 0.50   |
| Reaction time (s)         | 1.00   |
| Person capacity           | 4      |

We used the SUMO simulator [34] for emulating the urban traffic network. This is a microscopic traffic simulator which enables interaction with an external application through an interface (TraCI: Traffic Control Interface). This simulator feeds traffic information from the traffic detectors, taking raw-pollution data of vehicles and traffic-light actions. TraCI4Matlab [35] is a library of functions for interaction with SUMO from the Matlab command line. It was used for linking SUMO simulator to Matlab models.

Figure 6a shows the traffic queues obtained at all junctions. The plotted data corresponds to the average vehicle queues at every approach to each junction in a time window of 20 s. Figure 6b shows...