Linear-parameter-varying approximation of nonlinear dynamics for model predictive flow control of urban multi-region systems

Conference Paper

Author(s): Kouvelas, Anastasios; Saeedmanesh, Mohammadreza; Geroliminis, Nikolas

Publication date: 2020

Permanent link: https://doi.org/10.3929/ethz-b-000367957

Rights / license: In Copyright - Non-Commercial Use Permitted

Originally published in: https://doi.org/10.1109/CDC40024.2019.9029502
Linear-parameter-varying approximation of nonlinear dynamics for model predictive flow control of urban multi-region systems

Anastasios Kouvelas, Mohammadreza Saeedmanesh, Nikolas Geroliminis

Abstract—An alternative approach for real-time network-wide traffic control in cities that has recently gained a lot of interest is perimeter flow control. The focus of the current work is to study two aspects that are not covered in the perimeter control literature, which are: (a) integration of appropriate external demand information that has been considered system disturbance in the derivation of feedback control laws in previous works, and (b) mathematical formulation of the original nonlinear problem in a linear-parameter-varying (LPV) form, so that optimal control can be applied in a (rolling horizon) model predictive concept. This work presents the mathematical analysis of the optimal control problem, as well as the approximations and simplifications that are assumed in order to derive the formulation of a linear optimization problem. The developed scheme is applied to microsimulation in order to better investigate its applicability to real-life conditions. The simulation experiments demonstrate the effectiveness of the scheme compared to fixed-time control as all the performance indicators are improved significantly.

I. INTRODUCTION

Many efforts have been carried out to optimize signal settings during the peak hours, where networks face serious congestion problems and the performance of the infrastructure degrades significantly. The state-of-practice strategies fail to deal efficiently with oversaturated conditions (i.e. queue spillbacks and partial gridlocks), as they are either designed by use of simplified models that do not accurately replicate some traffic flow phenomena (e.g. propagation of congestion), or based on application-specific heuristics. An alternative approach for real-time network-wide traffic control that has recently gained a lot of interest is perimeter flow control (or gating). The basic concept of such an approach is to partition heterogeneous cities into a small number of homogeneous regions (zones) and apply perimeter control to the inter-regional flows along the boundaries between regions. The inter-transferring flows are controlled at the traffic intersections located at the borders between regions, so as to distribute the congestion in an optimal way and minimize the total delay of the system (an alternative objective could maximize the total throughput).

Perimeter flow control can be viewed as a high-level regional control scheme and might be combined with other strategies (e.g. local, distributed [1] or coordinated controllers) in a hierarchical control framework (see e.g. [2]): this topic has gained a lot of attraction in the research community lately. For a recent review on this research direction the reader is referred to [3], [4]. The original model for the dynamics of the multi-region process (plant) is highly nonlinear and the modelling tool that is utilized is the Macroscopic Fundamental Diagram (MFD) (see e.g. [5], [6]). MFD provides a concave, low-scatter relationship between network vehicle accumulations [veh] or density [veh/km] and network circulating flow [veh/h] or production [veh·km] for every region of the system which is homogeneously congested [7]. Recently, there has been some effort to simplify this model; the authors in [8] proposed a hybrid model for formulating this problem and applied model predictive control (MPC) in conjunction with switching plans control. A pre-defined set of MFDs is introduced (which correspond to regional switching plans), and some affine approximations are made in order to reduce the complexity of the derived hybrid MPC approach.

One key aspect that is investigated in the perimeter control literature is the treatment of model parameters uncertainties. A recent work [9] utilizes the context of distributed adaptive control in order to improve the performance of feedback controllers under uncertainties. The derived controllers incorporate input delay and can deal with bounded external dependencies. These approaches can effectively deal with parameters uncertainties, but, on the other hand, the control actions may in some cases be quite conservative (if many stochastic scenarios are generated). Finally, the studies in [10], [11] reveal some fruitful insights about the use of MFD-based systems in order to control road congestion through route guidance approaches.

In the current work we derive a linear-parameter-varying (LPV) formulation to solve the finite-time optimal perimeter flow control problem. Moreover, we integrate realistic external demand information that has been considered system disturbance in the derivation of multivariable feedback control laws in previous works [12], [13], [14]. The LPV system is optimized in a rolling time horizon using a feedback MPC framework, and the control decisions are applied to a microsimulation model (plant) for evaluation. The efficiency of the control decisions is compared to the case where the real pre-timed signal plans are applied to the network. The effort of comparing a developed MFD control framework versus a microsimulation instead of a simplistic plant is significantly more tedious and this is among the first efforts to test an MPC MFD framework in such a large network. It
is also known that nonlinear optimization problems can be stuck in local optima and the derived solution might have some optimality gap compared to the best possible solution (which is actually unknown). Our developed framework guarantees optimality in each applied step of the MPC.

In the literature of perimeter control [5], [15] a complex model is utilized to solve the nonlinear MPC problem. In the current work we simplify this model and create a linear-parameter-varying (LPV) system that is used for control purposes. One limitation of the derived LPV system is that it cannot be used for long prediction horizons as the time varying parameters need to be updated frequently. Nevertheless, if the prediction horizon is appropriate (i.e. within certain limits) the LPV system provides a good approximation of the original nonlinear system as demonstrated later in this paper. Given that online estimates of the model parameters are provided in real-time by an estimation engine [16], the resulting LPV system is updated online and utilized for solving the rolling horizon MPC problem. In order to evaluate the applicability to real life conditions the developed control scheme is tested in microsimulation experiments. For this purpose we apply the commercial software Aimsun where we can apply our MPC controller through the embedded API. The estimation engine presented in [16] is utilized in real-time in order to estimate some states that cannot be measured easily. Then, these estimates are fed to the linear MPC component that calculates the optimal control actions to be applied to the traffic lights of the network.

II. AGGREGATED DYNAMICS FOR A PARTITIONED CITY

This section builds on [17], [18] model formulation and is used as a reference and for the development of notation. Consider an urban network partitioned in \( N \) homogeneous regions with well-defined MFDs. The index \( i \in \mathcal{N} = \{1, 2, \ldots, N\} \) denotes the region of the system, \( n_i(t) \) the total accumulation (number of vehicles) in region \( i \), and \( n_{ij}(t) \) the number of vehicles in region \( i \) with final destination region \( j \in \mathcal{N} \), at a given time \( t \). Let \( \mathcal{N}_i \) be the set of all regions that are directly reachable from the borders of region \( i \), i.e. adjacent regions to region \( i \). The discrete time MFD dynamics of the \( N \)-region system can be described by the following first order difference equations

\[
n_{ii}(k) = n_{ii}(k-1) + T \left( q_{ii}(k-1) - M_{ii}(k-1) - \sum_{h \in \mathcal{N}_i} M^h_{ii}(k-1) + \sum_{h \in \mathcal{N}_i} M^h_{ii}(k-1) \right) \]

\[
n_{ij}(k) = n_{ij}(k-1) + T \left( q_{ij}(k-1) - \sum_{h \in \mathcal{N}_i} M^h_{ij}(k-1) + \sum_{h \in \mathcal{N}_i} M^h_{ij}(k-1) \right) \]

where \( i, j \in \mathcal{N}, i \neq j \); \( k = 1, 2, \ldots, K \) is the model discrete time index, \( T \) [sec] the sample time period of the model (i.e. time \( t = kT \)). The variables \( q_{ii}(k) \) [veh/sec] denote the exogenous traffic flow demand that is generated in region \( i \), at time step \( k \), with final destination in region \( j \) (i.e. \( q_{ij}(k) \) is the demand generated in region \( i \) that has final destination in region \( i \)). The variables \( M^h_{ij}(k) \) [veh/sec] denote the transfer flows from region \( i \) to region \( h \), that have final destination region \( j \), while \( M_{ii}(k) \) [veh/sec] is the internal trip completion rate of region \( i \) (without going through another region). Consequently, the total accumulation \( n_i(k) \) for region \( i \) can be computed by \( n_i(k) = \sum_{j \in \mathcal{N}} n_{ij}(k) \).

We assume that for each region \( i \) there exists a production MFD between accumulation \( n_i(k) \) and total production, \( P_i(n_i(k)) \) [veh/m/sec], which describes the performance of the system in an aggregated way. To this end, the transfer flows \( M^h_{ij}(k) \) and internal trip completion rates \( M_{ii}(k) \) are calculated according to the corresponding production MFD of the region, and proportionally to the accumulations \( n_{ij}(k) \) as follows

\[
G_i(n_i(k)) = \frac{P_i(n_i(k))}{L_i}, \quad M_{ii}(k) = \theta_i(n_i(k)) \frac{n_{ii}(k)}{n_i(k)} G_i(n_i(k)) \tag{3}
\]

\[
M^h_{ij}(k) = \min \left\{ C_{ih}(n_i(k)), \frac{u_{ih}(k)}{\theta^h_{ij}(k)} \frac{n_{ij}(k)}{n_i(k)} G_i(n_i(k)) \right\} \tag{4}
\]

where \( i, j \in \mathcal{N}, h \in \mathcal{N}_i; \ k = [k/N_c], \) with \( N_c \) some positive integer, is the control discrete time index (i.e. the control cycle is always a multiple of the plant sample time \( T \), and \( L_i \) [m] is the average trip length for region \( i \), which is assumed to be independent of time and destination, internal (inside region \( i \)) or external (to some other region \( j \)). The parameters \( \theta_i(n_i(k)), \theta^h_{ij}(k) \) reflect the route choice of drivers and are assumed to be exogenous (i.e. can be constant or time varying and they are provided by another methodology, e.g. an estimation technique). The transfer flows \( M^h_{ij}(k) \) are the minimum between the sending flow from region \( i \) (which only depends on the accumulation of the region), and the receiving capacity \( C_{ih}(n_i(k)) \) [veh/sec] of region \( h \). Moreover, each outflow MFD \( G_i(n_i(k)) \) is approximated by a third degree polynomial function of \( n_i(k) \), including the Gaussian white noise term \( \xi_i(k) \) which is constructed to be proportional to \( n_i(k) \), i.e.

\[
G_i(n_i(k)) = c_{i,1} n_i^2(k) + c_{i,2} n_i^3(k) + c_{i,3} n_i(k) + n_i(k) \xi_i(k) \tag{5}
\]

where \( \xi_i(k), i \in \mathcal{N} \) denotes noise that acts on the outflow MFD, which is empirical and not exact, in order to reflect the modelling inaccuracies. By imposing the noise proportional to the regional accumulation \( n_i(k) \) we achieve to introduce noise on the outflow MFD which is lower for uncongested conditions and higher for congested conditions, something that is more realistic. Note that in (1)–(4) there is no modelling noise as they describe the conservation of vehicles which holds strictly in any case.

The nonlinear difference equations (1)–(5) represent a stochastic version of equations presented in [5] and represent the traffic dynamics (plant) of an \( N \)-region urban network integrated with an aggregated routing model. Note, that these equations allow the drivers to choose any arbitrary sequence of regions as their route and their path can cross region boundaries multiple times (because of variables \( \theta_i(n_i(k), \frac{n_{ii}(k)}{n_i(k)} G_i(n_i(k)) \)).
The complete process plant can be written in a compact vector form as
\[
\mathbf{DS}_1(n,u,d,\xi) : \quad \mathbf{n}(k) = g\left[n(k-1), \mathbf{u}(k-1), \mathbf{q}(k-1), \xi(k-1)\right]
\]
where \( g \) is a nonlinear vector function corresponding to the \( N^2 \) model equations of the multi-region system accumulations. Including all \( i,j \in \mathcal{N} \), the vector \( \mathbf{n}(k) \) represents the plant state; \( \mathbf{d}(k) \), comprises all exogenous demand variables; \( \mathbf{u}(k) \) is the vector with the control inputs, and \( \xi(k) \) represents the vector with all modelling errors. The utilization of (6) requires real-time availability of \( \mathbf{d}(k) \) and determination of \( \mathbf{u}(k) \) by a control law. However, some elements of \( \mathbf{d}(k) \) may not be measured directly or even not at all measurable, but can rather be estimated by appropriate methods [16].

Furthermore, each region \( i \) has a maximum accumulation \( n_{i,\text{max}} \) such that \( 0 \leq n_i(k) \leq n_{i,\text{max}}, \forall i \in \mathcal{N} \). If \( n_i(k) = n_{i,\text{max}} \), the region reaches gridlock and all the inflows along the periphery are restricted. Finally, the control variables \( u_{ih}(k) \), \( \forall i \in \mathcal{N}, h \in \mathcal{M} \) denote the fraction of the flow that is allowed to transfer from region \( i \) to region \( h \). Physical constraints are applied to the values of the control variables, i.e. \( 0 \leq u_{ih}(k) \leq 1, \forall i \in \mathcal{N}, h \in \mathcal{M} \), but also – depending on the implementation – operational constraints of the following form might apply, i.e., \( |u_{ih}(k) - u_{ih}(k-1)| \leq u_{ih}^R, \forall i \in \mathcal{N}, h \in \mathcal{M} \), where \( u_{ih}^R \) is a user-defined parameter that is used to prevent big fluctuations in the control variables.

### A. Nonlinear model predictive control (NMPC)

The MFD dynamics described in the previous section derive a nonlinear model that has been used in other works [15], [5] to apply nonlinear model predictive control (NMPC). In order to have a well defined problem, the following assumptions are considered:

- the quantities \( q_{ii}(k), q_{ij}(k) \) and \( \theta_{ij}(k), \theta_{ij}^h(k) \) are considered exogenous variables that can be measured or given by another algorithm beforehand,
- as in many similar works [15], the capacity constraint \( C_{ih}(n_{ih}(k)) \) in (4) is dropped, since from a control viewpoint it is not necessary.

Given these two reasonable assumptions the nonlinear optimal control problem for a horizon of \( N_p \) model steps is defined as follows

\[
\begin{align*}
\text{maximize} & \quad \sum_{k=k_0}^{k_0+N_p-1} \sum_{i \in \mathcal{N}} L_i \left[ M_{ii}(k) + u_{ih}(k)M_{ij}^h(k) \right] \\
\text{subject to} & \quad \text{equations (1), (2), (3), (5), and saturation constraints} \\
M_{ij}^h(k) &= u_{ih}(k)\theta_{ij}^h(k) \frac{n_{ij}(k)}{n_i(k)} G_i(n_i(k)) \\
n_i(k) &= \sum_{j \in \mathcal{N}} n_{ij}(k) \\
k &= k_0, k_0+1, \ldots, k_0+N_p-1 \\
\kappa &= k_c, k_c+1, \ldots, k_c + \left\lfloor (N_p - 1)/N_c \right\rfloor, k_p = k_c \times N_c \\
\forall i,j \in \mathcal{N}, h \in \mathcal{M}, & \quad \text{where } \kappa = \lfloor k/N_c \rfloor \text{ is the control discrete time index, with } N_c = T_c/T \in \mathbb{Z}^+ \text{ some positive integer (i.e. the control cycle } T_c \text{ is always a multiple of the plant sample time } T). \quad (6)
\end{align*}
\]

where the constraints are applied to the values of the control variables, \( u_{ih}(k), \forall i \in \mathcal{N}, h \in \mathcal{M} \) denote the fraction of the flow that is allowed to transfer from region \( i \) to region \( h \) for the time interval \( \left\lfloor (N_c \kappa)T, (N_c \kappa + N_c)T \right\rfloor \).

The objective function (7) tries to maximize the total production of the system for a horizon of \( N_p \) model steps (or \( N_p T \) seconds). This problem can be solved in reasonable time by use of advanced nonlinear optimization toolboxes (e.g. ipopt\(^1\), see also [17]). Note that in [18] we have utilized this approach as a benchmark and compared the results with the LPV approach in a macroscopic simulation environment; here we apply the LPV approach to microsimulation which constitutes a very different plant and requires a significant effort for an implementation of online estimation and control.

### B. Linearising the problem

In the current work we derive a linear approximation of the model described in the previous section, and we formulate a linear MPC problem that can be utilized for real-time control purposes. In order to linearise the dynamic equations we assume the following simplifications and approximations:

- We introduce the model parameters for the accumulation proportions, i.e. \( \beta_{ii}(k) = n_{ii}(k)/n_{i}(k) \) and \( \beta_{ij}(k) = n_{ij}(k)/n_{i}(k), \forall i \in \mathcal{N}, j \in \mathcal{N} \). One approach, that is implemented here, is to get feedback for the values of these parameters every time that we roll the horizon. They can be estimated in real-time from measurements (e.g. using extended Kalman filter [16] or maximum likelihood approximation) and then be kept constant for all the optimization horizon. Note that the parameters can be time varying but they need to be exogenous signals for the MPC framework.
- New “dummy” control variables \( u_{ih}(k) \) are introduced \( \forall k = k_0, k_0+1, \ldots, k_0+N_p-1 \), that restrict the trip completion rates at every region \( i \). Although these variables are not reasonable from a physical point of view, they are required in order for the problem to be linear. A conjecture is that the solution of MPC will always result in \( u_{ih}(k) = 1, \forall i \in \mathcal{N}, k = k_0, k_0+1, \ldots, k_0+N_p-1 \), but this needs to be validated through the numerical results. Note that in the linear formulation, a separate control variable \( u_{ih}(k) \) exists for each model step (i.e. \( k = k_0 \)) rather than one for each control step.
- Most importantly, we introduce new decision variables

\[
\begin{align*}
\psi_{ii}(k) &= \psi_{ii}(k)G_i(n_i(k)) \theta_{ii}^h(k) \\
\psi_{ih}(k) &= \psi_{ih}(k)G_i(n_i(k)) \sum_{j \in \mathcal{N}} \theta_{ij}^h(k) \beta_{ij}(k) \\
\forall i,j \in \mathcal{N}, h \in \mathcal{M}, & \quad \text{that help linearise the equations and derive an LPV system. In (12)–(13) the variables } \theta_{ii}^h(k), \theta_{ij}^h(k), \beta_{ii}(k), \beta_{ij}(k) \text{ are considered time varying exogenous signals}
\end{align*}
\]

\(^1\)http://www.i2c2.aut.ac.nz/Wiki/OPTI/index.php/Solvers/IPOPT
and as a result the nonlinearity of the problem comes from the product of the control inputs $u_{ih}(k)$ with the MFD functions. Figure 1 provides a graphical representation of the new decision variables $f_{ih}(k)$, $f_{ih}(k)$ that help linearise the problem.

- To overcome this, we approximate the MFDs of the regions with piecewise affine (PWA) functions that form a convex set. Each MFD can be approximated with $l = 1, 2, \ldots, N$ affine functions, and we denote as $G_l^i(n_l(k))$ each affine function $l$.

In conclusion, the control variables have the property of being bounded (i.e. $u_{ih}(k) \in [0, 1]$, $\forall \ i \in \mathcal{N}$, $h \in \mathcal{N}_i$, $k = k_p, k_p + 1, \ldots, k_p + N_p - 1$) and the MFDs that can be approximated by PWA functions. As a result, we are looking for an optimal solution within a convex set, and in this particular case the product can be linearised by introducing the new variables $f_{ih}(k)$ (see [19] for some theoretical analysis of a similar convexification in a ramp metering control problem). Finally, once the optimal solution is computed, there is a unique transformation between the new variables and the original control variables $u_{ih}(k)$, $u_{ih}(k)$. This is a modeling trick that allows us to simplify the problem without loosing any accuracy in the dynamics.

III. LINEAR MODEL PREDICTIVE CONTROL (LMPC)

The assumptions outlined above are reasonable approximations/simplifications of the nonlinear model in order to derive a linear formulation that can be used for online MPC. In this section we formulate a linear model predictive control (LMPC) problem, which does not keep track of the origin-destination information of vehicles. This dynamic model has less online data requirements (as it carries lower level of information, i.e. only state and demand trajectories $d_i$, instead of $d_{ij}$), but under certain optimization horizons can provide similar optimal solutions for the control variables (see Section with simulation results). Note that the aggregated demands $q_i$ and accumulations $n_i$ are easy to be measured, while detailed demands $q_{ij}$ and accumulations $n_{ij}$ would require more advanced sensors configurations.

A. LMPC without OD information

Moving one step forward with our approximation, the new model does not need to keep track of the OD information (aggregated information about each region can be sufficient for control purposes). Hence, by adding all the states $n_{ij}$ and $n_i$ for each region $i$, we get a linear model that does not consider OD data, but only aggregated demands in the region level. In that case, the derived LPV that approximates the original system and can be solved online is as follows

$$\begin{align*}
\text{maximize} & \quad \sum_{k=k_p}^{k_p+N_p-1} \sum_{i \in \mathcal{N}} L_i \left[ f_{ih}(k) + f_{ih}(k) \right] \\
\text{subject to} & \quad n_i(k) = n_i(k-1) + T_p \left( q_i(k-1) - f_{ih}(k-1) - \right. \\
& \quad \left. \sum_{h \in \mathcal{N}_i} f_{ih}(k-1) + \sum_{h \in \mathcal{N}_i} f_{ih}(k-1) \right) \\
& \quad 0 \leq f_{ih}(k) \leq \theta_{ih}(k) \beta_{ih}(k) G_l^i(n_l(k)) \\
& \quad 0 \leq f_{ih}(k) \leq G_l^i(n_l(k)) \sum_{j \in \mathcal{N}} \theta_{ij}(k) \beta_{ij}(k) \\
& \quad 0 \leq n_i(k) \leq n_{i,\text{max}} \\
& \quad k = k_p, k_p + 1, \ldots, k_p + N_p - 1 \\
& \quad \forall \ i, j \in \mathcal{N}, \ h \in \mathcal{N}_i, \ l = 1, 2, \ldots, N_i
\end{align*}$$

where the objective function (14) again represents the total production of the system. All the constraints of this problem are linear, and, as a consequence, the computational requirements are quite low, even for a network with many regions and large prediction horizons. Note that in this rolling horizon framework the model parameters $\theta_{ij}(k)$ as well as the exogenous demands $q_i(k)$ are estimated and updated online through the methodology presented in [16].

B. Closing the loop with online estimation

In a real life implementation it is not reasonable to assume that we have perfect information of the state feedback. However, the measurements required for the LMPC framework can be estimated online via appropriate algorithms. For this reason an estimation engine has been developed in [16] that can produce estimates of the state vector by utilizing loop detectors data. The complete closed loop system of dynamic estimation and control is described in Figure 2.

![Fig. 1. Introduction of new variables $f_{ih}(k)$: (a) original transferring flow variables (on the MFD curve); (b) new flow variables (below the convex set of the MFD curve (gray area)).](a)

![Fig. 2. Real-time control loop (including estimation) for the multi-region dynamical system.](b)
measurements of accumulation $\bar{n}_i(k)$ and transferring flows from region $i$ to region $j$ $\bar{M}_{ij}(k)$ are forwarded to the extended Kalman filter (EKF) methodology presented in [16]. All measurements include zero-mean Gaussian noises. The EKF outputs the state estimates $\hat{n}_i(k)$, $\hat{q}_i(k)$, $\hat{\beta}_i(k)$ that are fed to the LMPC framework as feedback measurements. The solution of the LPV problem for the steps of the prediction horizon $N_p$ provides the optimal control inputs that are forwarded to the plant for implementation. Figure 2 provides a graphical description of the real-time closed loop system of estimation and control ($DS_2(x,u,y)$ refers to the dynamical system that is used by EKF methodology in [16]).

IV. APPLICATION TO MICROSIMULATION

The efficiency of the linear MPC scheme is also tested in microsimulation experiments. The Aimsun microscopic environment (Version 8.0.8) is used and the real-time implementation of the control scheme is replicated through the simulator API. Only loop detector data is utilized to estimate the state of the system, highlighting the feasibility and applicability of the developed framework in real life conditions.

A. Microsimulation model description

The urban network of Barcelona, Spain is used as the test site (Figure 3). The network covers an area of 12 square kilometers with about 600 intersections and 1500 links of various lengths and is modeled and calibrated in Aimsun. For the simulation experiments, typical loop-detectors have been installed around the middle of each network link. The OD-based demand that is used for the simulations consists of 123 origin centroids and 132 destination centroids and provides a good replication of real life conditions as it generates realistic traffic congestion patterns in the network. The duration of the simulation is 6 hours including a 15 minutes warm-up period. In the no control case (where a set of the real fixed-time plans of the city are applied to the intersections) the network faces some serious congestion problems, with queues spilling back to upstream intersections. Traffic lights at signalized intersections are operating on multi-phase fixed-time plans with constant (but not equal) cycle lengths. Note, that drivers adapt to the traffic conditions through a C-Logit route choice model (standard module of the simulator) that is activated every 3 minutes, therefore the distribution of demand into the network is more realistic. Note that the developed controller does not utilize any information related to OD; only loop detector data is required.

Traffic congestion in the city of Barcelona is unevenly distributed, creating multiple pockets of congestion in different areas of the network. Partitioning is a possible solution to take advantage of well-defined MFDs. The algorithm used in this study is an optimization framework called “Snake” algorithm [20], which considers heterogeneity index as a main objective function and contiguity is forced explicitly by imposing constraints. By applying this algorithm, the network of Barcelona is partitioned into 4 homogeneous regions that are shown in Figure 3(b). This partitioning simplifies the network dynamics as there is no need for routing information/decisions (i.e. due to the configuration of the regions there is only one choice to move from one region to another). Note, however, that the methodological control approach proposed in the previous sections can be applied to any arbitrary partitioning outcome.

B. Simulation results

In this section we report the results of the experiments. Figure 4 compares the trajectories of accumulations for the 4 regions, for fixed-time control (FTC) and LMPC, when the detailed measured data of the simulator are used as input to the controller. The congestion is much less in the LMPC case, improving significantly the delays in the system (area between solid and dashed curves). In the LMPC case there are less links that are saturated and we have less spillbacks and partial gridlocks. Moreover, the network empties earlier as more vehicles can be served for the same amount.

Then, we run the same experiment by utilizing the EKF methodology for real-time state estimation. Instead of taking the complete set of real measurements from Aimsun, we measure the accumulations $n_i(k)$ and the transferring flows $M_{ij}(k)$, and EKF provides the complete vector for the state estimation to LMPC (this is happening for every time interval $k$). Figure 5 presents the accumulations of the regions for that case in comparison again to FTC. The online estimation and control framework can again improve traffic conditions significantly. The control inputs fluctuate smoothly according to the prevailing traffic conditions. Finally, Table I provides the statistics that we obtain from Aimsun software for the three aforementioned scenarios; FTC, LMPC with perfect measurements (LMPC-PM) and LMPC-EKF. We can see the improvement of the linear controller described in this work in the case that it is fed with perfect measurements or in the case that some measurements are not available (and others are noisy) and the real-time estimation scheme is used to provide the state of the system. Interestingly, even if the

---

2Intersections that are considered for perimeter flow control have different constant cycle durations; these durations are not affected here (i.e. only the durations of green phases are changed) in order to maintain the coordination.
same number of vehicles is served in all cases, in the fixed-time control case vehicles travel slightly longer distances due to the re-routing of vehicles.

V. CONCLUSIONS

An LPV formulation is derived for solving the perimeter control problem in multi-region cities. The originally nonlinear system is relaxed and approximated by a simplified linear model that under certain assumptions can track the behaviour of the original multi-region system. The new model requires less information in terms of real-time measurements (e.g., traffic states, OD demands), and, because it is formulated as an LPV, it guarantees optimality and fast convergence of the solver. The solution of the linear MPC approximation provides the control inputs to be applied to the network and the overall performance of the controller is deemed beneficial for the network congestion.

An interesting research direction for future work is to use perimeter control as a first-layer controller in cities (as it deals with zone interactions) and develop a second-layer of distributed control (e.g., [21]) for optimizing locally (see also [2]). The combination of the two provides a hierarchical control scheme that could potentially be more efficient in alleviating traffic congestion in cities.

REFERENCES


TABLE I

<table>
<thead>
<tr>
<th>Criteria</th>
<th>NC</th>
<th>LMPC-PM</th>
<th>LMPC-EKF</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delay</td>
<td>539.48</td>
<td>369.34</td>
<td>468.53</td>
<td>sec/km</td>
</tr>
<tr>
<td>Mean Speed</td>
<td>5.96</td>
<td>8.3</td>
<td>6.75</td>
<td>km/h</td>
</tr>
<tr>
<td>Stop Time</td>
<td>450.52</td>
<td>294.7</td>
<td>388.23</td>
<td>sec/km</td>
</tr>
<tr>
<td>Total Time</td>
<td>73923.62</td>
<td>47454.54</td>
<td>60695.78</td>
<td>h</td>
</tr>
<tr>
<td>Total Distance</td>
<td>425745.89</td>
<td>390783.2</td>
<td>396844.67</td>
<td>km</td>
</tr>
<tr>
<td>Vehicles Served</td>
<td>201706</td>
<td>201706</td>
<td>201706</td>
<td>veh</td>
</tr>
</tbody>
</table>

Fig. 4. Vehicle accumulations for the 4 regions, FTC and LMPC, when we have access to the error-free measurements from the simulator.

Fig. 5. Vehicle accumulations for the 4 regions, FTC and LMPC, when we include the real-time EKF state estimation algorithm in the control loop.