Max-pressure Controller for Stabilizing the Queues in Signalized Arterial Networks

Anastasios Kouvelas *
Partners for Advanced Transportation Technology (PATH)
University of California, Berkeley
2105 Bancroft Way Suite 300, Berkeley, CA, 94720-3830, USA
kouvelas@berkeley.edu

Jennie Lioris
Partners for Advanced Transportation Technology (PATH)
University of California, Berkeley
2105 Bancroft Way Suite 300, Berkeley, CA, 94720-3830, USA
jennie.lioris@berkeley.edu

S. Alireza Fayazi
Department of Mechanical Engineering
Clemson University
138 Fluor Daniel Building, Clemson, 29634, SC, USA
sfayazi@clemson.edu

Pravin Varaiya
Department of Electrical Engineering and Computer Science
University of California, Berkeley
271M Cory Hall, Berkeley, 94720-1720, CA, USA
varaiya@berkeley.edu

*** Corresponding Author**

Paper submitted to TRB Annual Meeting 2014
August 1, 2013

5714 words + 5 figures + 2 tables => 7464 'words'

ABSTRACT

The problem of arterial signal control is considered here. Urban intersections face serious congestion problems and at the same time the installation and maintenance of centralized systems is deemed cumbersome. A decentralized approach which is relatively simple to implement is studied here. The recently proposed max-pressure controller, which provably stabilizes the queues of arterial traffic systems, is tested in simulations. Different modifications of the controller are analyzed and compared under the same demand scenarios. The mesoscopic model used for the simulation experiments is an extended version of the store-and-forward model and emulates the arterial traffic network as a queuing system. The obtained results demonstrate the efficiency of max-pressure algorithm, which, under certain conditions, can stabilize all queues of the system.

***Keywords*:** Traffic signal control; store-and-forward modeling; max-pressure controller; queuing networks; bounded queues; system stability; real-time adaptive control.

INTRODUCTION

Recent advancements in computing and communications have set the ground for the development and implementation of the next generation of traffic control strategies. Especially in arterial networks, there is an emerging effort towards the design and deployment of efficient signal control systems, as urban congestion continues to grow in most cities around the world. Although additional measures, such as road pricing, improved public transport operations, access restrictions of various kinds, driver information and guidance, may also help to alleviate the congestion problem, improved signal control strategies remains a significant objective. Real-time signal control systems that respond automatically to the prevailing traffic conditions are deemed to be potentially more efficient than clock-based fixed-time control settings. A variety of such strategies have been developed during the past few decades, some of which have been actually implemented while others are still in a research stage (see, e.g., (*1*), (*2*) for a good review).

Historically, SCOOT (*3*) and SCATS (*4*) systems were among the earliest efforts to develop adaptive traffic control systems for arterial networks. These well-known and widely-used traffic-responsive control systems are based on heuristic optimization algorithms. Other optimization methods for arterial traffic control that follow a centralized design avenue are OPAC (*5*), PRODYN (*6*) and RHODES (*7*), which are all based on dynamic programming and the rolling-horizon optimization scheme. More recently, TUC system (*8*) was introduced, which applies a multivariable feedback regulator approach that derives from a Linear-Quadratic (LQ) optimal control problem. This system has been successfully implemented in several large networks in Europe and South America; see (*9*) for some recent field results.

All the aforementioned systems have a centralized nature, that is, the control inputs are a function of all the measurements of the network. Hence, in order to apply one of these systems, the information from all intersections must be collected and transmitted to a central location (i.e. Traffic Management Center). This communication and computing architecture imposes significant installation, operating and maintenance costs, which has inhibited widespread adoption of traffic-responsive and adaptive control. According to (*10*), traffic-responsive and adaptive control achieve large benefits but fewer than 10% of intersections in U.S.A. use adaptive signals, because of the deployment cost of detection and communication and uncertainty about the benefits. By contrast, local controllers are much easier to implement as they only use the measurements around a certain area of interest and are considered to be more cost effective.

This paper presents the local feedback controller max-pressure which is applied intersection-by-intersection and uses only the adjacent measurements of queue lengths. The methodology was originally proposed in (*11*), considering the problem of routing and scheduling packet transmissions in a wireless network. In packet networks, the term backpressure policy has been adopted. The name max-pressure may have been coined by (*12*), and it seems to be the preferred term in scheduling and routing in flexible manufacturing networks. There is a large literature on max-pressure or backpressure algorithms. Different variations of the methodology that can be potentially applied to arterial road networks in real-time (depending on the available infrastructure and communication capabilities) are presented here, and evaluated by the use of a mesoscopic simulation environment.

For the purposes of the simulation experiments an event-based queuing model has been developed, which has been validated to capture the dynamics of queues in arterial streets. For the validation procedure, real data that have been collected from an arterial in Los Angeles area under the NGSIM initiative (*13*) have been used. The results of the simulation investigations demonstrate the ability of max-pressure to stabilize the queues of the studied system, in contrast to other local controllers, including priority service and fully actuated control, which are proven to not stabilize the queues of vehicles in arterial intersections (see (*14*) for details).

DECENTRALIZED FEEDBACK CONTROL

In this section the max-pressure control for arterial networks is introduced. This decentralized controller does not require any knowledge of the mean current or future demands of the network (in contrast to other model predictive control frameworks). Max-pressure stabilizes the network if the demand is within certain limits, thus it maximizes network throughput. However, it does require knowledge of mean turn ratios and saturation rates, albeit an adaptive version of max-pressure will have the same performance, if turn movements and saturation rates can be measured. It only requires local information at each intersection and provably maximizes throughput (*14*). Several variations of the basic method that can be applied in real-time (depending on the available infrastructure) are presented.

Notations

The arterial network is represented as a directed graph with links $z \in Z$ and nodes $n \in N$. For each signalized intersection n, we define the sets of incoming I_n and outgoing O_n links. It is assumed that the offsets and the cycle time C_n of node n are fixed or calculated in real-time by another algorithm. In addition, to enable network offset coordination, it is quite usual to assume that $C_n = C$ for all intersections $n \in N$ but this is not the case here as the coordination problem is not considered. The signal control plan of node n (including the fixed lost time L_n) is based on a fixed number of stages that belong to the set F_n, wherein v_j denotes the set of links that receive right of way at stage $j \in F_n$. Finally, the saturation flow S_z of link $z \in Z$ and the turning movement rates $\beta_{i,w}$, where $i \in I_n$ and $w \in O_n$ are assumed to be known and can be constant or time varying.

By definition, the constraint

$$\sum_{j \in F_n} g_{n,j}(k_n) + L_n = (or \ \leq) \ C_n$$

holds for every node n, where $k_n = 0, 1, 2, \dots$ is the control discrete-time index and $g_{n,j}$ is the green time of stage j. Inequality in the equation above may be useful in cases of strong network congestion to allow for all-red stages (e.g. for strong gating). In addition, the constraint

$$g_{n,j}(k_n) \geq g_{n,j,\min}, \ \ j \in F_n$$

where $g_{n,j,\min}$ is the minimum permissible green time for stage j in node n and is introduced in order to guarantee allocation of sufficient green time to pedestrian phases. The control variables of the problem are $g_{n,j}(k_n)$ which depict the effective green time of every stage $j \in F_n$ of every intersection $n \in N$.

Max-pressure

The state of each link $x_z(k_n)$ is defined by the number of vehicles waiting in the queue to be served for each control index k_n (i.e. at the beginning of time period $[k_n C_n, (k_n+1)C_n]$. Given that we are provided with real-time measurements or estimates of all the states we can compute the pressure $p_z(k_n)$ that each link exerts on the corresponding stage of node n at the beginning of cycle k_n as follows

$$p_z(k_n) = \left[\frac{x_z(k_n)}{x_{z,\max}} - \sum_{w \in O_n} \frac{\beta_{i,w} x_w(k_n)}{x_{w,\max}} g_{n,j}(k_n)\right] S_z, \ \ z \in I_n$$

where $x_{z,\max}$ is the storage capacity of link z (in vehicles). Storage capacity is used in the denominator in order to take into account the length of the links, so that the pressure of a short link with a number of vehicles waiting to be served is higher than the pressure of a longer link with the same number of vehicles. The measurements (or estimates) $x_z(k_n)$, $\forall \ z \in Z$ represent a feedback from the network under control, based on which the new pressures are calculated via the equation above in real-time

The pressure of link z during the control cycle k_n is the queue length of the link (first term within the brackets) minus the average queue length of all the output links (second term within the brackets). Regarding the second term as the (average) downstream queue length and the first as the upstream queue length, the definition of the pressure is simply the difference between the upstream and downstream queue lengths. It should be noted, that in the case where all output links are exiting the network (we assume that exit links have infinite capacity, i.e., they do not experience any downstream blockage), the second term in the brackets becomes zero. Hence, the pressure of the link is simply the queue length multiplied by its corresponding saturation rate.

If the equation above is applied $\forall\, z \in I_n$ the pressures of all incoming links of node n are calculated. The pressure of each stage j of the intersection can then be computed as follows

$$P_{n,j}(k_n) = \max\left\{ 0\,, \sum_{z\in v_j} p_z(k_n) \right\}\,,\ \ j \in F_n$$

and this metric can be used to calculate the splits for the different conflicting stages of the intersection.

Variations on max-pressure

This paper investigates different modifications of max-pressure control and their ability to stabilize the system queues via simulation experiments. A demand (i.e. time-series of incoming flow in the network origins) is said to be stabilizable if there exists a control plan that can accommodate it (i.e. the time-average of every mean queue length is bounded). The set of feasible (stabilizable) demands D is a convex set and can be easily defined for an intersection by solving a collection of linear inequalities involving only the mean values of the demands, turn ratios and saturation rates. If a demand D^0 is in the interior of the convex set D then there exists a fixed-time control that stabilizes the queues. Under this control the intersections may experience cycle failures but no queue is going to grow continuously.

Given that the pressure of each stage has been computed by the last equation of the previous section, the total effective green time G_n that is available to be distributed in node n

$$G_n = C_n - L_n - \sum_{j\in F_n} g_{n,j,\min}\,,\ \ n \in N$$

can be split to all stages in many different ways. One approach, is to select the stage with the maximum pressure and activate it for the next control cycle C_n. This implies that all the available effective green time G_n will be given to this stage. In the next cycle the queues of the system are updated, the new pressures are calculated and the stage with the maximum pressure is selected to be activated and so forth. This approach may not be the optimal one, as the control cycle may be large and queues can grow unexpectedly at the links that are not activated. Alternatively, max-pressure can be called several times within a cycle C_n. Every time the stage with the maximum pressure is activated, however, the frequency of the measurements/control is now higher. The frequency of max-pressure application to an intersection depends on two main factors: (a) the available infrastructure and communications (i.e. the appropriate measurements or estimates of queue lengths should be provided in real-time), and (b) an optimal frequency of max-pressure application which needs to be investigated and defined (and could be dependent on the special characteristics of each site).

Another approach, is to call max-pressure at the end of each cycle and split the green time G_n proportionally to the computed pressure of each stage. That is, for each decision variable $\tilde{g}_{n,j}(k_n)$ (where $\tilde{g}_{n,j}$ depicts the green time of stage j on the top of $g_{n,j,\min}$) the following update rule is applied

$$\tilde{g}_{n,j}(k_n) = \frac{P_{n,j}(k_n)}{\sum_{i\in F_n} P_{n,i}(k_n)} G_n\,,\ \ j \in F_n.$$

Thus, the total amount of green time allocated for each control variable $g_{n,j}(k_n)$ for cycle k_n is given by

$$g_{n,j}(k_n) = \tilde{g}_{n,j}(k_n) + g_{n,j,\min}\,,\ \ j \in F_n.$$

This procedure is repeated periodically (for every cycle) and requires minimum communication specifications, as the local controller is called once per cycle.

THE SIMULATION MODEL

A modeling avenue for network-wide simulation of the queue dynamics in arterial networks, especially for oversaturated traffic conditions, is based on the store-and-forward modeling paradigm, first proposed in (*15*). According to this model, the equation that describes the evolution of the queue for an arterial link z is as follows (see (*16*) for details)

$$x_z(t+1) = x_z(t) + T_{t\to(t+1)}[q_z(t) - s_z(t) + d_z(t) - u_z(t)]$$

where $x_z(t)$ is the number of vehicles within link z at the end of the discrete time period t (for the sake of brevity also called queue in this document); $q_z(t)$ and $u_z(t)$ are the inflow and outflow, respectively, in the sample period $T_{t\to(t+1)}$ (i.e. for the time period between the discrete time index t and $(t+1)$); $d_z(t)$ and $s_z(t)$ are the demand and the flow exiting the network within this link, respectively. As mentioned earlier, this is an event-based simulation model, hence the sample periods are not constant since they are triggered by different kind of events (e.g. vehicle arrival, signal change, etc.). The update equation above establishes the conservation of vehicles in link z. Consider that link z connects two intersections M_u and M_d such that $z \in O_{M_u}$ and $z \in I_{M_d}$. The exit flow $s_z(t)$ is given by

$$s_z(t) = \beta_{z,0} q_z(t)$$

where the exit rates $\beta_{z,0}$ are assumed to be known. The inflow to the link z is given by

$$q_z(t) = \sum_{i \in I_{M_u}} \beta_{i,z}\, q_i(t)$$

where $\beta_{i,z}$ with $i \in I_{M_u}$ are the turning rates towards link z from the links that enter junction M_u. Queues are subject to the constraints

$$0 \le x_z(t) \le x_{z,\max}\,, \quad z \in Z$$

where $x_{z,\max}$ is the maximum admissible queue length (in vehicles). Although the right part of the inequality is modeled in the simulator, it is not used for the investigations undertaken here as we want to demonstrate the ability of max-pressure to bound the queue length. Thus, no constraint on the maximum length of the queues is applied to the experiments presented here. For modeling the outflow $u_z(t)$, an approach that characterizes the utilized modeling approach and can be found in many places (see e.g. (*17*)) is introduced. According to this, the outflow $u_z(t)$ of link z is equal to the saturation flow S_z if the link has right of way, and equal to zero otherwise. Consequently,

$$u_z(t) = \begin{cases} \min\{x_z(t),\, S_z T_{t\to(t+1)}\} & \text{if sample period } T_{t\to(t+1)} \in \text{green phase} \\ 0 & \text{if sample period } T_{t\to(t+1)} \in \text{red phase.} \end{cases}$$

Another challenging part in the procedure of designing the simulation model has been the modeling of the turning movements, i.e., when there are more than one incoming links flowing to the same outgoing link simultaneously. Based on the signal plans that are used in the United States (permitted right and left turns need to consider gaps between opposite traffic), in an arterial network we can have multiple input links merging or diverging into output links simultaneously for every discretized time period t. This has been modeled with different queues for each movement and gap acceptance criteria but the details are beyond the scope of this document (the reader is referred to (*14*) where the simulation model is discussed in detail).

Model validation

The mesoscopic simulation model was validated with arterial data obtained by (*13*). A network with 4 signalized intersections in Los Angeles was studied (FIGURE 1). The inputs to our model are the demand profiles in all network origins (time-series of number of vehicles entering the network), the time varying profiles of split ratios in all nodes, and all the information about the signals (duration of phases, cycles, offsets). In reality, actuated control is applied to all the studied intersections, thus, the signal plans can be varying over time depending on the prevailing traffic conditions. It should be noted that there are two parking lots in the area, the inflows and outflows of which interact with the traffic of the main arterial.

Detailed information about the turn ratios towards the parking lots, as well as for the outflows from the lots to the main arterial were also provided.

The dataset that was simulated has a duration of 30 minutes (8:30am–9:00am) and the evaluation metric is the comparison between the total simulated outflow (in vehicles) for all the signalized links (link IDs: 1, 2, 3, 4, 6, 7, 8, 9 in FIGURE 1(b)). The location that the outflow is measured (both in the simulation and in reality) is the stop-line of each link. TABLE 1 demonstrates the obtained results from the validation procedure. More precisely, TABLE 1(a) presents the real measurements after the raw NGSIM data was processed, TABLE 1 (b) presents the outflows obtained by the simulation model, and TABLE 1(c) displays the comparison of the two aforementioned outflows (the (%) difference of the simulated over the real values). It should be noted, that the comparison is done for time intervals of 5 minutes (first column of TABLE 1) and the sampled number of vehicles is accumulated as time progresses. The simulated outflows of the signals are close to the real ones, illustrating that the store-and-forward model accurately simulates the real traffic conditions. Note, that for the corresponding dataset the network does not experience any severe congestion. Finally, the only tuning parameters for the simulation are the saturation flows S_z for each signal approach, which are assumed to be constant over the whole simulation horizon.

Closed-loop control and evaluation metrics

In order to investigate the efficiency and stabilizing property of the max-pressure controller, extensive simulation experiments have been carried out. The developed simulation model that was described in the previous section also tests the applicability of the methodology in real life conditions, as it comprises a replica of the real traffic conditions (as validated with real data). The arterial network is modeled as a queuing network with queues of vehicles waiting to be served at conflicting approaches of signalized intersections.

In our experiments, the overall closed-loop scheme uses a properly structured API module in order to control the traffic lights of the network in real-time. More specifically at each cycle k_n, the simulation engine delivers the queue lengths for all the incoming and outgoing links of node n, which are then used as inputs to max-pressure controller. The pressures of each stage of the intersection are calculated and according to the version of the controller used, the signal settings are computed (splits for every stage $g_{n,j}(k_n) \in F_n$). FIGURE 2 illustrates the way max-pressure controller is used in closed-loop with the simulation engine. The blue box indicates the computations that are done on the max-pressure side. The algorithm gets the simulated queue lengths $x_z(t)$, $\forall z \in Z$ and outputs the duration of all stages in real-time.

Another output of the simulation engine that is used in order to evaluate the applied control policies is the mean travel time (MTT). The model keeps track of every individual vehicle, the time-stamp that it entered the network and the time-stamps that it enters or exits each of the consecutive traveled links on its path. The time that each individual vehicle joins a queue is also stamped, thus, at any given time the user has access to the complete state of any vehicle in the network (i.e. its location and whether it is moving or not). Therefore, at the end of the simulation, statistics about the time that each vehicle has spent on each link of the network are available (time traveling on free flow and stopping time), and all travel times can be averaged (by approach, link, path, etc.) providing a performance metric for the enforced signal plans.

SELECTED SIMULATION RESULTS

Some selected illustrative results are presented in this section. Due to the stochastic nature of the simulator (i.e. all vehicles arrivals follow a Poison process, turn ratios in intersections follow stochastic distributions with predefined mean values) many replications of the same experiments have been run and the results shown here stem from scenarios that are close to the respective case's average. FIGURE 3 presents the fictitious arterial network that was used in the simulations. It consists of 4 signalized intersections and 12 links. All links are one-way and each intersection has 2 green stages (i.e. no left turns are allowed for simplicity) the duration of which can be controlled in real-time (cycle times and offsets are kept constant for all simulation scenarios). For all experiments we use $C_n = 62, \forall n$ and $g_{n,j,\min} = 5$, $\forall n, j$ and there is an intergreen of 5 seconds between successive stages. The simulation horizon is 1 hour.

Two demand scenarios have been generated D^1 and D^2 (i.e. vehicle arrivals in all origins of the network over time) and two fixed-time controls have been calculated (by solution of the system of linear inequalities that derive for this network and each demand scenario), L^1 and L^2 respectively, that can stabilize each demand. It should be noted, that the demand scenarios are created in such a way that no single fixed-time control can stabilize both demands. Each of the simulations is characterized by three attributes, i.e., the demand, the applied control (two variations of max-pressure are presented here) and the capacity of the links. The third attribute is based on the queue bound and whether or not the maximum queue inequality is applied. In the case that the inequality holds we say that we have a finite storage capacity for the link, whereas when the inequality is not applied links are assumed to have infinite storage capacity. The simulations have been run for both cases in order to investigate if this constraint is important for the model or not.

FIGURE 4(a) presents the evolution of the total number of vehicles in all the queues of the network ($\sum_{z \in Z} x_z(t)$) when the capacity of the queues if infinite, while FIGURE 4(g) shows the same run when the capacities are finite. FIGURE 4(c) presents the trajectory of the number of queued vehicles when MP1 control is applied. In MP1, max-pressure controller is called twice per cycle and the stage with the maximum current pressure is activated. FIGURE 4(e) displays the same scenario with MP2 control, where the total green duration is split proportionally to the pressures, however this is done once every C_n (i.e. at the beginning of each cycle, according to the current measurements). FIGURE 4(b) depicts what happens if after the end of the simulation presented in FIGURE 4(a) we continue with demand D^2 and with the same fixed-time control L^1. Obviously, this demand cannot be stabilized by this control and the sum of the queues is continuously growing until the end of the simulation. In contrast, if we apply either MP1 (FIGURE 4(d)) or MP2 (FIGURE 4(f)) the system is stabilized, as the sum of the queues is clearly bounded. FIGURE 4(h) is the same run as FIGURE 4(d) whereas the capacities of the queues are considered finite. In the experiment of FIGURE 4(f) the queues are assumed finite as well.

Similar runs have been conducted for demand D^2 and selective results are presented in FIGURE 5. FIGURE 5(a) shows the evolution of the queues for the fixed-time control L^2, FIGURE 5(c) for controller MP1 and FIGURE 5(e) for controller MP2. After the end of demand D^2 (3600 sec) the simulation is continued for another hour using demand D^1 and the same controllers are applied; the results are displayed in FIGURE 5(b), (d) and (f) respectively. All the experiments presented in FIGURE 5 have been run with infinite link capacities, except the simulation of FIGURE 5(e) where finite capacities have been used for all network links. Both in FIGURE 4 and FIGURE 5 the controller MP2 produces higher oscillations than MP1 for all experiments. This is an indication that calling max-pressure twice per cycle is better than once per cycle, as the recently updated measurements help the controller to track the changes in the queue lengths and adjust its inputs.

Evaluation of simulation results

The simulation results presented here are some representative runs and comprise a subset of all the conducted experiments. The main findings of the simulations in terms of control are twofold, (a) max-pressure will stabilize the queues of arterial networks if the demand is within the feasible area, and (b) the frequency that the controller is applied affects its performance. On the latter, one could say (by studying the Figures presented here) that the more frequently max-pressure is applied the narrower the bounds of the system can be. However, this statement is not obvious for any arbitrary network topology. For the simulation model, the capacity of the links (finite versus infinite) does not seem to play an important role at least for this kind of experiments, where the objective is to validate the stability of the system. The one approach is helpful to investigate if some of the queues are growing continuously with the corresponding control, while the other is important as it allows for consideration of spillback (or even gridlock) phenomena.

TABLE 2 displays the Total Travel Time (TTT) in vehicles∗hours for all the different major routes of the arterial network in FIGURE 3 and for all the reported simulated scenarios. As expected, the total number of vehicles in the queues explodes if the fixed-time control is not appropriate (something that can even happen with other local controllers), albeit max-pressure manages to stabilize the state of the system, given that a feasible fixed-time controller that can stabilize the system exists. The metric TTT is slightly higher for MP2 compared to MP1 and this is because of the different frequency (once per cycle versus twice per cycle respectively).

CONCLUSIONS

Two different versions of max-pressure controller are presented here and tested through simulation experiments. The first activates the stage with the maximum pressure every time the controller is called, while the latter distributes the green time proportionally to the respective pressures. The macroscopic simulation model that was used for the investigations has been developed in Python and is based on the store-and-forward model. It is an event-based simulator (i.e. the simulation step is not constant but rather defined by events that are triggered by an event planner module) and represents the arterial traffic as a queuing system. The results demonstrate the efficiency of max-pressure controller and validate the already published theoretical argument that it can stabilize networks of arbitrary topology. As shown, the frequency that the controller is applied plays a key role on the bounds of the summation of the network queues. Further experiments are needed in order to clarify the optimal frequency for different networks and demand scenarios.

ACKNOWLEDGMENTS

This research is funded primarily by the California Department of Transportation under the Connected Corridors program. S. Alireza Fayazi was sponsored in part by the National Science Foundation grant number CMMI–0928533 during his visit to University of California, Berkeley. The authors would like to thank Gabriel Gomes and Alex Kurzhanskiy for the constructive discussions and useful comments. Also, we are grateful to Dongyan Su for providing the processed NGSIM data.

REFERENCES

[1] Mirchandani, P., and Head, L. A Real-Time Traffic Signal Control System: Architecture, Algorithms, and Analysis. *Transportation Research Part C*, Vol. 9, No. 6, 2001, pp. 415–432.

[2] Papageorgiou, M., Diakaki, C., Dinopoulou, V., Kotsialos, A., and Wang, Y. Review of Road Traffic Control Strategies. *Proceedings of the IEEE*, Vol. 91, No. 12, 2003, pp. 2043–2067.

[3] Hunt, P., Robertson, D., Bretherton, R., and Royle, M. The SCOOT On-line Traffic Signal Optimization Technique. *Traffic Engineering and Control*, Vol. 23, 1982, pp. 190–192.

[4] Lowrie, P. SCATS: The Sydney Co-ordinated Adaptive Traffic System – Principles, methodology, algorithms. In *Proceedings IEE International Conference on Road Traffic Signalling*, London, England, 1982, pp. 67–70.

[5] Gartner, N., OPAC: A Demand-Responsive Strategy for Traffic Signal Control. *Transportation Research Record: Journal of the Transportation Research Board*, No. 906, 1983, pp. 75–84.

[6] Farges, J., Henry, J., and Tufal, J. The PRODYN Real-Time Traffic Algorithm. In *Proceedings 4th IFAC Symposium on Transportation Systems*, Baden, Germany, 1983, pp. 307–312.

[7] Mirchandani, P., and Head, L. RHODES – A Real-Time Traffic Signal Control System: Architecture, Algorithms, and Analysis. In *TRISTAN III*, San Juan, Puerto Rico, 1998, Vol. 2.

[8] Diakaki, C., Papageorgiou, M., and Aboudolas, K. A Multivariable Regulator Approach to Traffic-Responsive Network-Wide Signal Control. *Control Engineering Practice*, Vol. 10, 2002, pp. 183–195.

[9] Kosmatopoulos, E., Papageorgiou, M., Bielefeldt, C., Dinopoulou, V., Morris, R., Mueck, J., Richards, A., and Weichenmeier, F. International Comparative Field Evaluation of a Traffic-Responsive Signal Control Strategy in Three Cities. *Transportation Research Part A*, Vol. 40, No. 5, 2006, pp. 399–413.

[10] Lindley, J. Applying Systems Engineering to Implementation of Adaptive Signal Control Technology. In *19th ITS World Congress*, Vienna, Austria, 2012.

[11] Tassiulas, L., and Ephremides, A. Stability Properties of Constrained Queuing Systems and Scheduling Policies for Maximum Throughput in Multihop Radio Networks. *IEEE Transactions on Automatic Control*, Vol. 37, No. 12, 1992, pp. 1936–1948.

[12] Dai, J., and Lin, W. Maximum Pressure Policies in Stochastic Processing Networks. *Operations Research*, Vol. 53, No. 2, 2005, pp. 197–218.

[13] NGSIM: The Next Generation SIMulation Community. FHWA, U.S. Department of Transportation, www.ngsim.fhwa.dot.gov, 2006.

[14] Varaiya, P. The Max-Pressure Controller for Arbitrary Networks of Signalized Intersections. In *Advances in Dynamic Network Modeling in Complex Transportation Systems*. Springer, 2013, pp. 27–66.

[15] Gazis, D.C., and Potts, R.B. The Oversaturated Intersection. In *Proceedings 2nd International Symposium on Traffic Theory*, London, UK, 1963, pp. 221–237.

[16] Aboudolas, K., Papageorgiou, M., and Kosmatopoulos, E. Store-and-Forward Based Methods for the Signal Control Problem in Large-Scale Congested Urban Road Networks. *Transportation Research Part C*, Vol. 17, No. 2, 2009, pp. 163–174.

[17] Aboudolas, K., Papageorgiou, M., Kouvelas, A., and Kosmatopoulos, E. A Rolling-Horizon Quadratic-Programming Approach to the Signal Control Problem in Large-Scale Congested Urban Road Networks. *Transportation Research Part C*, Vol. 18, No. 5, 2010, pp. 680–694.

LIST OF TABLES

LIST OF FIGURES

TABLE 1 Total outflows comparison for the validation network (a) vehicles measured in real dataset (NGSIM); (b) simulated number of vehicles; (c) comparison of assessment metric (% difference of number of vehicles).

Link ID	1	2	3	4	6	7	8	9
Time	N-S				S-N			
8:30–8:35	180	170	158	110	43	99	90	90
8:35–8:40	355	340	320	221	99	215	192	186
8:40–8:45	539	513	484	337	139	337	301	291
8:45–8:50	755	714	667	457	195	454	406	395
8:50–8:55	973	924	875	601	247	601	541	525
8:55–9:00	1156	1093	1059	716	302	733	661	642

(a)

Link ID	1	2	3	4	6	7	8	9
Time	N-S				S-N			
8:30–8:35	178	177	166	104	43	98	100	84
8:35–8:40	355	352	319	213	100	220	201	179
8:40–8:45	541	524	495	344	138	338	316	287
8:45–8:50	757	738	697	470	194	483	423	390
8:50–8:55	977	951	918	626	246	630	534	501
8:55–9:00	1160	1135	1099	751	301	771	655	615

(b)

Link ID	1	2	3	4	6	7	8	9
Time	N-S				S-N			
8:30–8:35	-1.11	4.12	5.06	-5.45	0.00	-1.01	11.11	-6.67
8:35–8:40	0.00	3.53	-0.31	-3.62	1.01	2.33	4.69	-3.76
8:40–8:45	0.37	2.14	2.27	2.08	-0.72	0.30	4.98	-1.37
8:45–8:50	0.26	3.36	4.50	2.84	-0.51	6.39	4.19	-1.27
8:50–8:55	0.41	2.92	4.91	4.16	-0.40	4.83	-1.29	-4.57
8:55–9:00	0.35	3.84	3.78	4.89	-0.33	5.18	-0.91	-4.21

(c)

TABLE 2 Total Travel Time (TTT) criterion as computed by the simulation engine (a) scenarios presented in FIGURE 4; (b) scenarios presented in FIGURE 5.

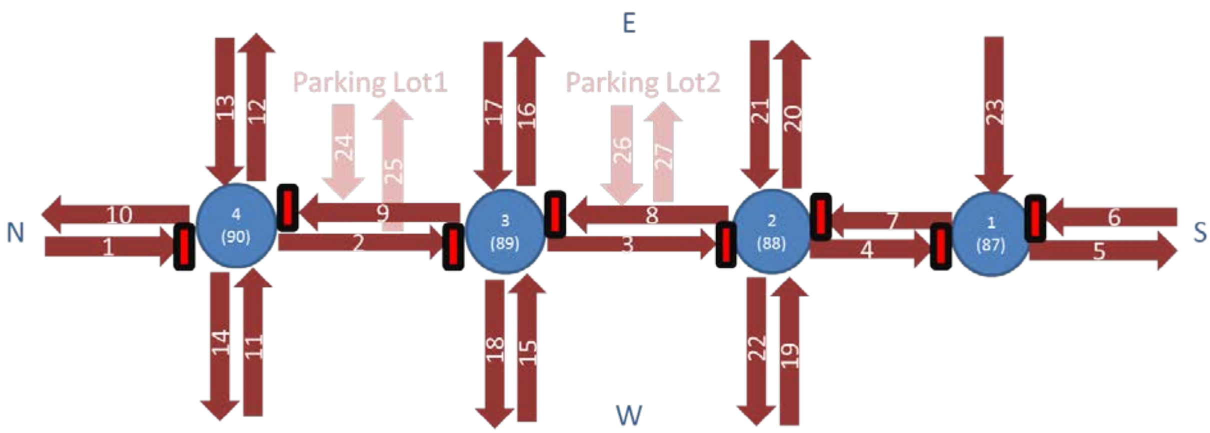
Route (Entry Link)–(Exit Link)	TTT – Total Travel Time (vehicles∗hours) Simulation scenario (FIGURE 4)							
	1(a)	1(b)	1(c)	1(d)	1(e)	1(f)	1(g)	1(h)
(1)–(3)	4.63	2.25	5.02	2.02	7.64	3.58	4.64	2.20
(4)–(6)	4.70	51.91	2.48	5.78	3.77	9.52	4.39	5.41
(7)–(9)	2.97	37.24	2.45	4.91	2.83	5.67	3.08	4.58
(10)–(12)	5.88	6.43	4.74	4.29	5.51	4.24	6.26	4.35
SUM	18.18	97.83	14.69	17.00	19.75	23.01	18.37	16.54

(a)

Route (Entry Link)–(Exit Link)	TTT – Total Travel Time (vehicles∗hours) Simulation scenario (FIGURE 5)					
	2(a)	2(b)	2(c)	2(d)	2(e)	2(f)
(1)–(3)	3.86	36.91	1.87	5.12	2.54	9.02
(4)–(6)	9.83	2.97	5.87	2.42	11.12	4.08
(7)–(9)	4.05	2.21	5.01	2.70	5.56	2.60
(10)–(12)	4.13	10.81	3.43	5.41	4.66	6.74
SUM	21.87	52.90	16.18	15.65	23.88	22.44

(b)

(a)



(b)

FIGURE 1 The NGSIM arterial network in Los Angeles: (a) satellite view of the network, (b) schematic representation of the simulation model (links, nodes and signals).

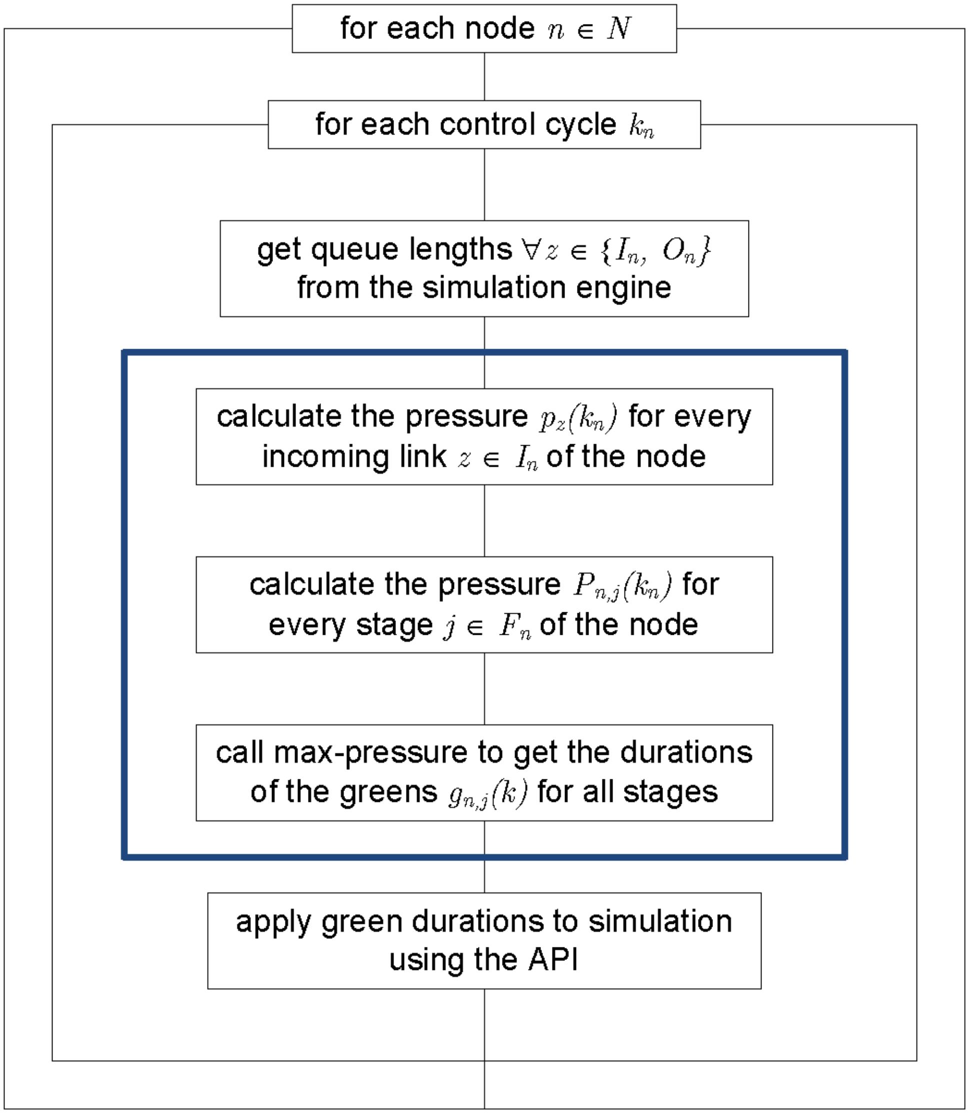


FIGURE 2 The closed-loop testing scheme of max-pressure controller using the macroscopic simulation engine.

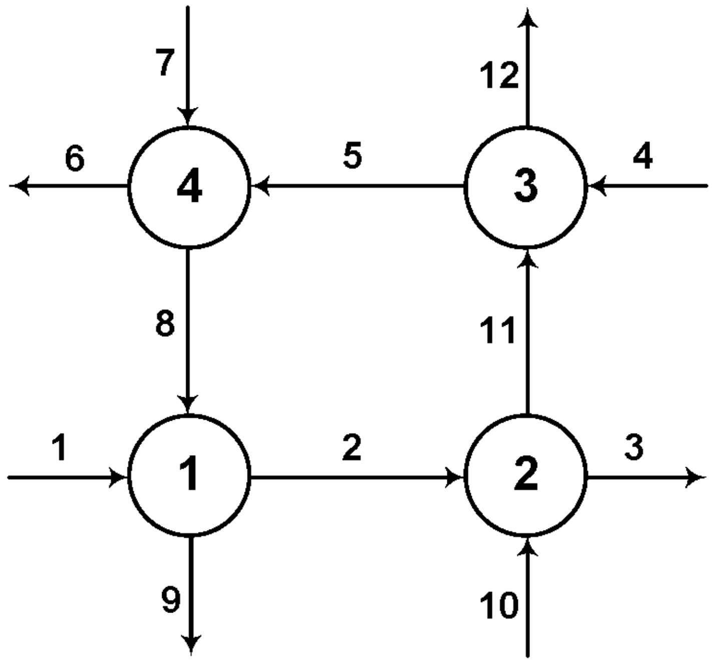


FIGURE 3 Directed graph of the fictitious arterial network used in the simulations with $\boldsymbol{N = \{1, 2, 3, 4\}}$ nodes and $\boldsymbol{Z = \{1, 2, \ldots, 12\}}$ links. We assume that links exiting the network do not experience any downstream blockage.

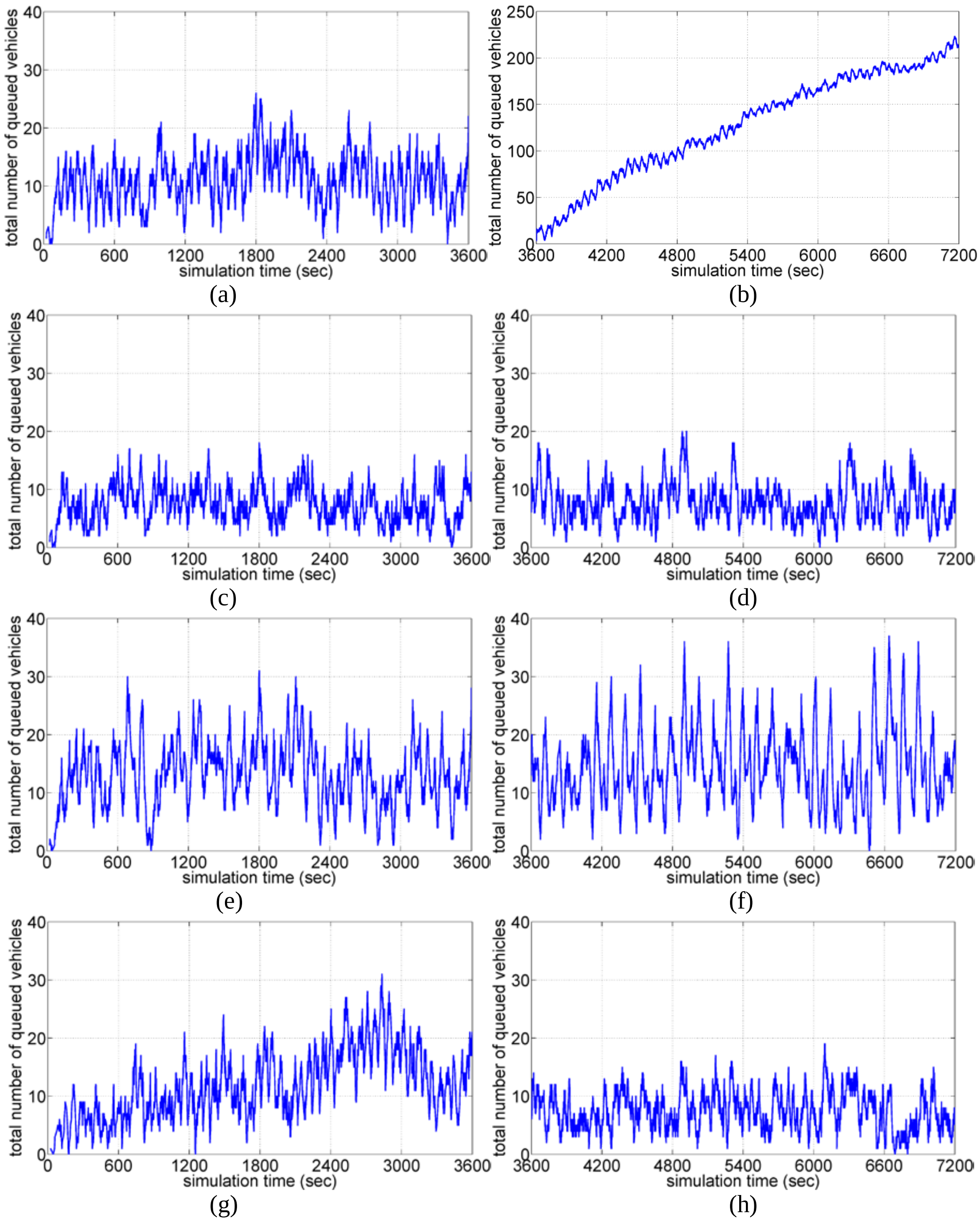


FIGURE 4 Evolution of total number of vehicles in all queues of the network for simulations that start with demand D^1 and different scenarios: (a) D^1 and L^1 (infinite capacities); (b) D^2 after D^1 and L^1 (finite capacities); (c) D^1 and MP1 (infinite capacities); (d) D^2 after D^1 and MP1 (infinite capacities); (e) D^1 and MP2 (finite capacities); (f) D^2 after D^1 and MP2 (finite capacities); (g) D^1 and L^1 (finite capacities); (h) D^2 after D^1 and MP1 (finite capacities).

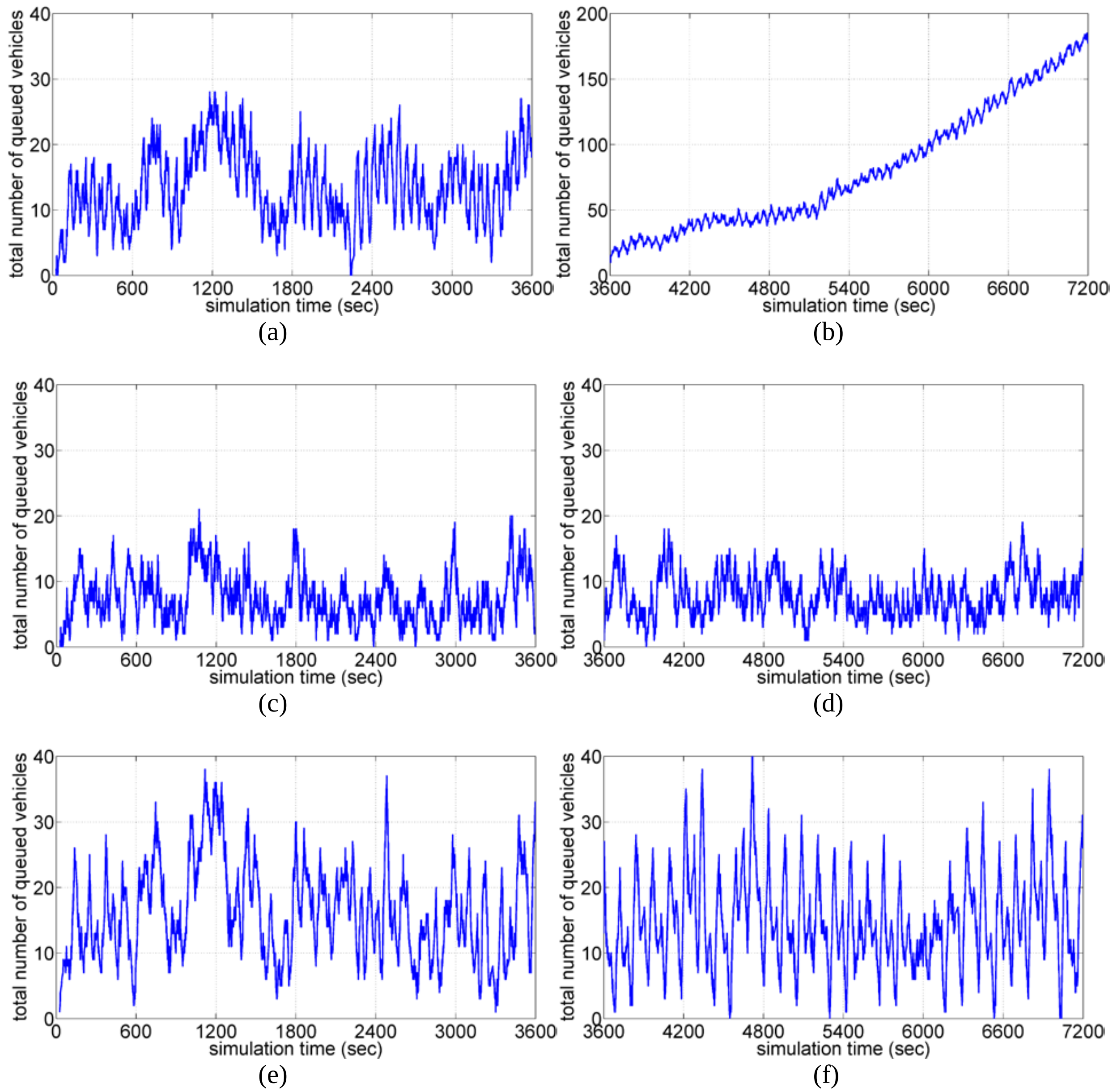


FIGURE 5 Evolution of total number of vehicles in all queues of the network for simulations that start with demand D^2 and different scenarios: (a) D^2 and L^2 (infinite capacities); (b) D^1 after D^2 and L^2 (finite capacities); (c) D^2 and MP1 (infinite capacities); (d) D^1 after D^2 and MP1 (infinite capacities); (e) D^2 and MP2 (infinite capacities); (f) D^1 after D^2 and MP2 (infinite capacities).