OPTIMAL TRAFFIC CONTROL IN FREEWAY NETWORKS WITH BOTTLENECKS

A. Hegyi [*] B. De Schutter [*] J. Hellendoorn [*]

[*] *Delft Center for Systems and Control, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands, {a.hegyi,b.deschutter,j.hellendoorn}@dcsc.tudelft.nl*

Abstract:
We consider optimally coordinated freeway traffic control for networks containing bottlenecks with capacity drop. Due to the multitude of traffic jams, and the spatial and temporal relationships between control actions and traffic behavior, this coordinated control problem is not as straightforward as for local control. The order in which the measures are applied may be relevant, or it may be possible that not all jams can be resolved. In that case the best possible locations of jams should be determined. We develop an approach that addresses these problems, where we use a generalized representation of flow-limiting control measures and bottlenecks. The approach supplies the necessary sequence of control actions and the necessary relocation of traffic jams to minimize the total time that vehicles spend in the network. *Copyright $^{\copyright}$ 2005 IFAC*

Keywords: freeway traffic control, traffic networks, capacity drop, hybrid systems

1. INTRODUCTION

1.1 Traffic control in networks

In the past dynamic traffic control measures (such as ramp metering, route guidance, and dynamic speed limits) have been applied mainly locally to resolve traffic jams. However, due to the high traffic demands the spatial and temporal relationships in traffic networks have become stronger: a control measure applied at a certain place and time may also influence (positively or negatively) the traffic later and/or at more distant parts of the network. Therefore, it is necessary to take the network-wide effects of the control measures into account, which implies coordination between the measures.

Even though a network-wide coordinated approach may result in general in more effective traffic control, in many cases it may not be possible to solve all traffic jams in the network or it may be necessary to create temporarily a traffic jam somewhere else in the network in order to solve a given jam (similarly to ramp metering, which locally creates a queue to solve a freeway jam). With the approach presented in this paper an answer is given to which jams need to be solved (or created temporarily) in order to achieve the best performance of the network.

Furthermore, there may be limitations on the traffic control signals, such as minimum and maximum metering rates, bounds on dynamic speed limits, or the limited rerouting effects of route guidance, so that these measures may not limit the flow sufficiently to solve a jam individually. The combination with other measures may than result in a more effective control that can solve the jam. However, using these measures may also reduce flow in other parts of the network. The selection of the appropriate measures in such a case is a trade-off, and may depend on the traffic demands, the drivers' route choice, and other factors.

In this paper these issues will be addressed, and an approach will be developed to find the control mea-

sures that result in the optimal network behavior over a given time horizon. Optimality will be defined as the control strategy that minimizes the total time spent (TTS) by the vehicles in the network (cf. (Kotsialos *et al.*, 2002; Alessandri *et al.*, 1998)).

1.2 Capacity drop

There are several possible causes why freeway networks do not always perform optimally. One of them is the so-called capacity drop (or two-capacity phenomenon), which is the phenomenon that the outflow of a jam at a bottleneck (the so-called queue discharge rate) is lower than the capacity of the bottleneck in free-flow (the so-called pre-queue capacity). As long as the jam remains existent, the performance of a jammed freeway link will be sub-optimal.

Another reason for sub-optimally performing freeway networks is blocking, which occurs when the tail of a traffic jam propagates back to a bifurcation where it also blocks the traffic that has a route that does not go via the bottleneck location that caused the jam.

In this paper we focus on the capacity drop and develop an approach to determine whether the network performance can by improved by dynamic traffic control measures. For the sake of simplicity we will assume that the queues occurring at bottlenecks and flow-limiting control measures do not become so long that they reach bifurcations or other bottlenecks. Although this is a restricting assumption, extension of the model to include blocking and bottleneck interaction is possible. The capacity drop may occur at several types of bottlenecks, such as on-ramps, upstream propagating jams (shock waves), off-ramps, curves, grades, tunnels and bridges. The value of the capacity drop has been estimated for some of these bottlenecks: for on-ramps (Kerner, 2002; Zhang and Levinson, 2004; Hall and Agyemang-Duah, 1991; Cassidy and Bertini, 1999) it was found to be in the range of 0–15%, and for upstream propagating jams around 30% (Kerner, 2002; Hegyi, 2004).

2. PROBLEM DESCRIPTION

We represent the evolution of the traffic network over time in *stages* where the index k refers to the k-th stage. The time duration of stage k is denoted by $t(k)$, with $t(k) \geq 0$.

The benefit of having $t(k)$ as an independent variable is that – as we will see later – for a given k the system behaves linearly as a function of the control inputs $u(k)$ and $t(k)$. This results in a simpler description of the dynamic behavior of the system.

However, in practice most traffic control devices, such as ramp metering or route guidance, accept inputs at discrete time steps, e.g., every 30 or 60 seconds.
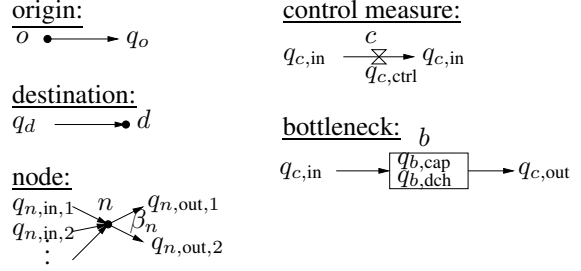

Fig. 1. The network elements.

The expectation is that the optimal switching sequence (explained in Section 3.1) will result in time durations $t(k)$ that are significantly larger than the sampling time of the control measures. Therefore the rounding of the time durations $t(k)$ will result in a small performance loss only [1].

2.1 Network elements

We model a traffic network by a directed graph that contains problem-specific elements. Each network consists of the following elements (see also Figure 1 for the symbols of the elements and the related variables):

Origins. We assume that the origins are the sources of traffic. Origin o is an element of the set of all origins $\{O_1, O_2, \dots\}$, and provides a constant inflow to the network of q_o (veh/h).

Destinations. Destinations are the sinks of traffic. The average flow at destination $d \in \{D_1, D_2 \dots\}$ in stage k, is denoted by $q_d(k)$ (veh/h).

Nodes. At nodes traffic from several incoming links may be joined and redistributed over one or two [2] outgoing links. The flows of the incoming links of node $n \in \{N_1, N_2, \dots\}$ are denoted by $q_{n,\text{in},i}(k)$ (with $i \in \mathcal{I}_n$, where $\mathcal{I}_n$ denotes the set of indexes of the incoming links of node n), and the outgoing links are denoted by $q_{n,\text{out},j}(k)$ (with $j \in \{1, 2\}$). The inflows and outflows are related by $q_{n,\text{out},j}(k) = \beta_{n,j}(k) \, q_n(k)$, where $q_n(k) = \sum_{i \in \mathcal{I}_n} q_{n,\text{in},i}(k)$, and $\beta_{n,j}(k)$ is the fraction of traffic that leaves node n through link j. Of course, $\beta_n(k) \geq 0$ and $\sum_{j \in \mathcal{O}_n} \beta_{n,j}(k) = 1$, where $\mathcal{O}_n$ is the set of indices of leaving links from node n.

If there is no route guidance, a constant turning rate $\beta_{n,j}(k) = \beta_{n,j}$ is assumed. If there is route guidance at the node, we will consider $q_{n,\text{ctrl}}(k) = q_{n,\text{out},1}(k) \, (= \beta_{n,1}(k) \, q_n(k))$ as the control variable.

There may be bounds on the route guidance signal $q_{n,\text{ctrl}}(k)$, which are expressed by $\beta_{n,1,\text{min}} \, q_n$ and $\beta_{n,1,\text{max}} \, q_n$, with $0 \leq \beta_{n,1,\text{min}} \, q_n \leq \beta_{n,1,\text{max}} \, q_n \leq 1$. This leads to the following relations:

---

[1] The rounding has to be chosen such that the resulting control signals satisfy the constraints developed later in this section.
[2] The extension to more outgoing links is straightforward. We allow here one or two outgoing links for the sake of simplicity.
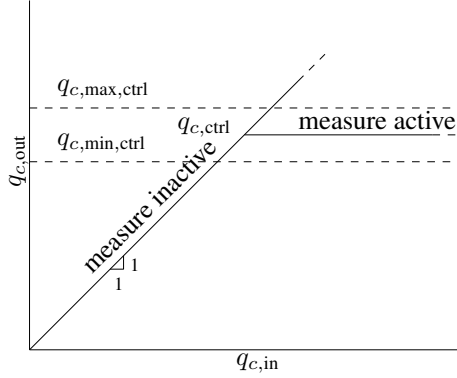
(a) The relations between inflow and outflow of the flow-limiting control measure. The relations depend on the activity status of the measure.
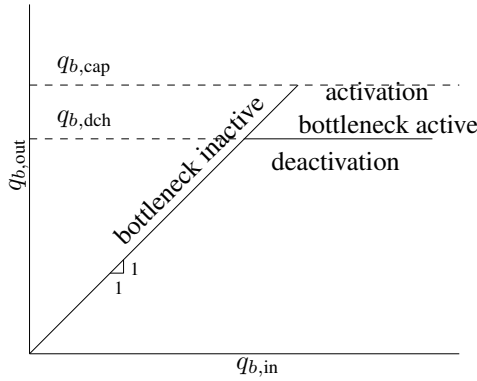
(b) The relations between inflow and outflow of a bottleneck. The relations depend on the activity status of the bottleneck.

Fig. 2. The relations between inflow and outflow for control measures and bottlenecks.

$$q_{n,\text{out},1}(k) = q_{n,\text{ctrl}}(k)\ ,$$
$$q_{n,\text{out},2}(k) = q_n(k) - q_{n,\text{ctrl}}(k)\ ,$$
$$\beta_{n,1,\text{min}}\, q_n \leq q_{n,\text{ctrl}}(k) \leq \beta_{n,1,\text{max}}\, q_n\ .$$

Flow-limiting control measures. Traffic control measures, such as ramp metering, main-stream metering, and dynamic speed limits can be represented by a generalized control measure $c \in \{C_1, C_2, \dots\}$ that describes the corresponding flow limitation.

A flow limitation can be active or inactive. The flow limitation is called active when the outflow is limited by the controller (see also Figure 2(a)). So, when it is active there will be a queue present of length $w_c(k)$. The following relations hold:

$$u_c(k) = q_{c,\text{out}}(k)\, t(k)$$
$$u_c(k) \leq q_{c,\text{in}}(k)\, t(k) + w_c(k)$$
$$u_c(k) \geq 0$$
$$w_c(k+1) = w_c(k) - u_c(k)$$

where $u_c(k)$ is the control input and represents the number of vehicles that leave the control measure in stage k, $q_{c,\text{in}}(k)$ is the inflow and $q_{c,\text{out}}(k)$ the

outflow at measure c. In addition, the bounds on the control input are defined as: $q_{c,\text{min}}\, t(k) \leq u_c(k) \leq q_{c,\text{max}}\, t(k)$, with $0 \leq q_{c,\text{min}} \leq q_{c,\text{max}}$.

If the flow limitation is not active, then there is no queue and the outflow is not limited:

$$u_c(k) = q_{c,\text{in}}\, t(k)$$
$$q_{c,\text{out}}(k) = q_{c,\text{in}}(k)$$
$$w_c(k) = 0$$
$$w_c(k+1) = w_c(k)$$

The activity status of c is denoted by χ_c, which has a value 1 if the control measure is active and 0 if it is inactive.

Bottlenecks. A generalized bottleneck $b \in \{B_1, B_2, \dots\}$ may represent several kinds of bottlenecks, such as on-ramps, bridges, tunnels, curves, grades, shock waves [3], merges, and bifurcations. The common factor in these bottlenecks is that they have a limited capacity $q_{b,\text{cap}}$, and that there may be a capacity drop if the bottleneck is jammed. The queue discharge rate is denoted by $q_{b,\text{dch}}(\leq q_{b,\text{cap}})$, where equality holds if there is no capacity drop, but only a limited capacity.

Similarly to flow-limiting control measures a bottleneck can also be active or inactive, and the relation between the inflow and outflow depends on the activity status. The basic idea for the bottleneck modeling is that if the inflow exceeds the capacity then the bottleneck will become active (congested) and the outflow will drop to the queue discharge rate (see also Figure 2(b)). In order to resolve the jam at the bottleneck the inflow must be limited to a value lower than the outflow (the queue discharge rate) and the queue length must be zero. When the jam is resolved, the bottleneck becomes inactive and the outflow may increase up to the capacity again.

So, if the bottleneck is active, then

$$q_{b,\text{out}}(k) = q_{b,\text{dch}}$$
$$w_b(k) - (q_{b,\text{dch}} - q_{b,\text{in}}(k))\, t(k) \geq 0$$
$$w_b(k+1) = w_b(k) - (q_{b,\text{dch}} - q_{b,\text{in}}(k))\, t(k)\ ,$$

and if the bottleneck is inactive, then

$$q_{b,\text{out}}(k) = q_{b,\text{in}}(k)$$
$$q_{b,\text{in}}(k) \leq q_{b,\text{cap}}$$
$$w_b(k) = 0$$
$$w_b(k+1) = w_b(k)$$

The activity status of b is denoted by χ_b, which has a value 1 if the bottleneck is active and 0 if it is inactive.

Links. Links provide the connection between any two other elements. A link connects the outflow of the upstream element with the inflow of the downstream

---

[3] The representation of moving shock waves is valid in our framework as long as the shock wave does not propagate upstream to other network elements.

element. The capacity[4] of a link is assumed to be unlimited. If a freeway link with limited capacity is modeled, a bottleneck element should be included.

Now we can build networks with the elements from Section 2.1 (see Figure 3 for an example).

2.2 Network properties

Network mode. We define the network mode $\mathcal{M}$ as the vector of the activity states of all bottlenecks and control measures $\mathcal{M} = [\chi_{B_1}, \chi_{B_2}, \ldots, \chi_{C_1}, \chi_{C_2}, \ldots]^T$. The current network mode can be acquired from speed, flow and density measurements at the bottlenecks and control measures. The bottleneck modes cannot be controlled directly, only through the available control measures, which may change the inflow of a bottleneck such that a mode change is triggered.

Mode changes. If the network is in a given mode, then other modes may be reached autonomously or by varying the control inputs. A queue of a control measure or bottleneck may become zero, which may cause a state change from active to inactive, or if the control inputs are changed, then the activity state of a control measure or a bottleneck may change. An activity state change is triggered if the inflow of a measure or bottleneck violates the condition that would guarantee the current state (e.g., the condition that for an active bottleneck the inflow must be higher than the queue discharge rate) and if the corresponding queue is zero.

So, in addition to the relations of Section 2.1 for the (current and next) modes at stages k and $k + 1$, the following relations must hold for the mode change. If a bottleneck changes from active to inactive then

$$w_b(k) - (q_{b,\text{dch}} - q_{b,\text{in}}(k)) \, t(k) = 0$$
$$q_{b,\text{in}}(k + 1) < q_{b,\text{dch}},$$

and if the activity state changes from inactive to active then

$$q_{b,\text{in}}(k + 1) > q_{b,\text{cap}} \,.$$

If a control measure changes from active to inactive then

$$w_c(k) - u_c(k) + q_{c,\text{in}}(k) \, t(k) = 0 \,,$$

and there are no additional constraints for control measures if the activity state changes from inactive to active.

Note that it is necessary to require strict inequalities for the bottleneck activity changes since the transition will be triggered only in that case. Since in practice the

state transition of bottlenecks (when congestion is created or resolved) may be influenced by disturbances and stochastic effects, we include an extra margin ϵ in the inequalities related to the mode changes. E.g., the inequality of the change from inactive to active becomes

$$q_{b,\text{in}}(k + 1) \geq q_{b,\text{cap}} + \epsilon \,.$$

For the same reason we also include a constraint to require that each mode has a minimum dwell time (Morse, 1995)

$$t(k) \geq t_{\min} \,.$$

2.3 Problem statement

According to the formulation in Section 2.1 and Section 2.2 the time duration $t(k)$ of stage k is a free variable.

The outflow of a network is uniquely defined for any control input sequence $u_c(k)$, $q_{n,\text{ctrl}}(k)$, $t(k), k = 0, 1, \ldots, K$, where K is the index of the last stage. The purpose here is to find the control input sequence and the corresponding mode sequence that minimize the total time that vehicles spend in the network (TTS).

In order to compare the performance of different control signals we define a time horizon t_{hor} over which the control strategies are compared.

Now we are ready to formulate the control problem:

Given a traffic network as defined in Section 2.1 and Section 2.2, an initial network mode $\mathcal{M}(0)$ and initial queue lengths $[w_c(0)^T w_b(0)^T]^T$ at time t_0 find the control inputs $u_c^*(k)$, $q_{n,\text{ctrl}}^*(k)$, $t^*(k)$, $k = 0, \ldots, K$, and K, that minimize the total time that vehicles spend in the network over $[t_0, t_0 + t_{\text{hor}}]$.

3. APPROACH

In this section we present an approach to solve the stated problem. First, we derive the expression for the TTS, and show that for a given mode sequence minimizing the TTS is a quadratic programming problem. Next, we discuss the procedure for finding the mode sequence that minimizes the TTS.

3.1 Optimizing for a given mode sequence

In this section we consider a given mode sequence $\mathcal{S} = \{\mathcal{M}(k)\}_{k=0}^M$. From the relations between flows, queues, and control inputs as given in Section 2.1 a discrete-time mode-dependent state-space model can be derived for the evolution of the queues:

$$w(k + 1) = w(k) + B(k) \, u(k) + e(k) \, t(k), \quad (1)$$
$$k = 0, \ldots, K,$$

where $w(k) = [w_c(k)^T, w_b(k)^T]^T$ is the vector of all queues at control measures and bottlenecks, $u(k) =$

---

[4] Here we mean flow-capacity. Storage capacity is not relevant due to the assumption of no blocking and no bottleneck interaction.

$[u_c(k)^\mathrm{T}, u_n(k)^\mathrm{T}]^\mathrm{T}$ is the vector of the control inputs at flow-limiting control measures and at nodes, and the matrix $B(k)$ and the vector $e(k)$ express the mode-dependent relationship between the inflow to a queue and the control measures, and between the inflow to a queue and the constants in the network (such as queue discharge rates and origin flows). The matrix $B(k)$ and the vector $e(k)$ can be determined by tracking back (in the upstream direction) from the network element (flow-limiting control measure or bottleneck) to all origins, active bottlenecks, and flow-limiting control measures that supply flow to the considered network element. Matrix $B(k)$ and the vector $e(k)$ are mode-dependent, since the relations in the network are mode-dependent.

The total time that vehicles spend in the network (TTS) is most easily expressed in terms of the number of vehicles $N(k)$ in the network at stage k (we assume a linear growth of queues between two mode transitions):

$$J_{\mathrm{TTS}}(\mathcal{S}, K) = \sum_{k=0}^{K} \left\{ \frac{N(k) + N(k+1)}{2}\, t(k) \right\} \quad (2)$$

and the number of vehicles in the network is given by $N(k) = \sum_{i=1}^{n_w} w_i(k)$, where n_w is the length of the vector $w(k)$. Combining this with (1) and (2) results in

$$J_{\mathrm{TTS}}(\mathcal{S}, K) = \frac{1}{2} \left\{ t(0) \sum_{i=1}^{n_w} w_i(0) + \right. \quad (3)$$

$$t(K) \sum_{i=1}^{n_w} \left(w_i(0) + \sum_{j=0}^{K} B_i(j)\, u(j) + e_i(j)\, t(j) \right) +$$

$$\left. \sum_{k=0}^{K} t(k) \sum_{i=1}^{n_w} \left(w_i(0) + \sum_{j=0}^{k-1} B_i(j)\, u(j) + e_i(j)\, t(j) \right) \right\}$$

where $B_i(k)$ is the i-th row of $B(k)$ and $e_i(k)$ the i-th element of $e(k)$. Note that $J_{\mathrm{TTS}}(\mathcal{S}, K)$ is quadratic in the control inputs $u(k)$ and $t(k)$.

The constraints for a given mode $\mathcal{M}(k)$ can be found in a similar way as $B(k)$ and $e(k)$, i.e., by tracking from the network elements (bottlenecks and flow-limiting control measures) to all upstream origins, active bottlenecks and active control measures that contribute to the flow of the considered network element. After adding the constraints resulting from the mode transitions the resulting set of constraints can be written as

$$F\, x \le g, \quad (4)$$
$$Q\, x = r, \quad (5)$$

where x is the vector of all control inputs $x = [u(0)^\mathrm{T}, t(0), \ldots, u(K)^\mathrm{T}, t(K)]^\mathrm{T}$, and F, g, Q, and r are matrices and vectors of appropriate sizes.

The minimization of the objective function (3) subject to the constraints (4) and (5) form a quadratic programming problem (QP) that can be solved by standard techniques, such as the modified simplex algorithm and the interior point methods (Pardalos and Resende, 2002; Nesterov and Nemirovskii, 1994).

3.2 Finding the optimal mode sequence

The procedure for finding the optimal mode sequence will be based on the A* algorithm (Russel and Norvig, 1995). This algorithm is based on a cost function that has the property that for a sequence of K decisions the cost is higher than or equal to the cost for any subsequence of the first $L, (L < K)$ decisions (which is called monotonicity). Consequently any subsequence leading to the optimal solution sequence will have a cost lower than or equal to the optimal cost. So, any subsequence that has a higher cost than a solution sequence (a candidate for the *optimal* solution sequence) can be discarded, since all solution sequences that contain this sequence will have an even higher cost. The algorithm always expands the subsequence with the lowest cost with all possible decisions. The algorithm stops if all subsequences and solution sequences have higher cost than a given solution sequence. This sequence is guaranteed to be the optimal one.

Since we compare mode sequences for $t \in [t_0, t_0 + t_{\mathrm{hor}}]$ we are eventually interested in minimizing $J_{\mathrm{TTS,hor}}(\mathcal{S}, K)$ (including finding the optimal mode sequence $\mathcal{S}$ and the corresponding K) for which the additional constraint $t_{\mathrm{hor}} = \sum_{k=0}^{K} t(k)$ holds. However, to guarantee monotonicity in the cost we will need to evaluate $J_{\mathrm{TTS}}(\mathcal{S}, K)$ for subsequences without this constraint. To distinguish between the two objective functions, we will denote $J_{\mathrm{TTS}}(\mathcal{S}, K)$ when this additional constraint is not imposed, and $J_{\mathrm{TTS,hor}}(\mathcal{S}, K)$ when it is imposed.

For solution sequences $J_{\mathrm{TTS,hor}}(\mathcal{S}, K)$ needs to be minimized, since this gives the TTS for $t \in [t_0, t_0 + t_{\mathrm{hor}}]$. For subsequences $J_{\mathrm{TTS}}(\mathcal{S}, K)$ is minimized, which may result in a duration that is more or less than t_{hor}. However, the optimized $J_{\mathrm{TTS}}(\mathcal{S}, K)$ provides a lower bound for the cost $J_{\mathrm{TTS,hor}}(\tilde{\mathcal{S}}, \tilde{K})$ of any solution sequence $\tilde{\mathcal{S}}$ that starts with $\mathcal{S}$ ($K \le \tilde{K}$). This follows from the fact that $J_{\mathrm{TTS,hor}}(\tilde{\mathcal{S}}, K) \le J_{\mathrm{TTS,hor}}(\tilde{\mathcal{S}}, \tilde{K})$, when $J_{\mathrm{TTS,hor}}(\tilde{\mathcal{S}}, \tilde{K})$ is optimized and $J_{\mathrm{TTS,hor}}(\tilde{\mathcal{S}}, K)$ is computed using the same control inputs, since $K < \tilde{K}$, and from the fact that $J_{\mathrm{TTS}}(\mathcal{S}, K) \le J_{\mathrm{TTS,hor}}(\tilde{\mathcal{S}}, K)$ since $J_{\mathrm{TTS}}(\mathcal{S}, K)$ is optimized.

We now describe the algorithm that finds the optimal control inputs. For brevity we use $J_{\mathrm{TTS}}(\mathcal{S})$, and $J_{\mathrm{TTS,hor}}(\mathcal{S})$ to denote $J_{\mathrm{TTS}}(\mathcal{S}, K)$, and $J_{\mathrm{TTS,hor}}(\mathcal{S}, K)$ respectively, with K being the counter corresponding to the last mode in $\mathcal{S}$. Let us denote the initial mode by $\mathcal{M}_0$. We will also make use of the set $\mathcal{X}$ in which we store mode sequences, and of the set $\mathcal{Y}$ in which we will store tuples $(S, u(k), t(k), J_{\mathrm{TTS}}(\mathcal{S}))$ consisting of a state sequence and the corresponding control signals, the duration of each mode, the resulting performance.

begin

   1. Initialize $\mathcal{S} = \{\mathcal{M}_0\}$, and $\mathcal{Y} = \emptyset$.

   2. Minimize $J_{\text{TTS}}(\mathcal{S})$, i.e., find $u^*(k)$, and $t^*(k)$ for $k = 0, \dots, K$ that minimize $J_{\text{TTS}}(\mathcal{S})$.

   3. Mark $\mathcal{S}$ as a subsequence and add the tuple $(\mathcal{S}, u(k), t(k), J_{\text{TTS}})$ to $\mathcal{Y}$.

   4. Minimize $J_{\text{TTS,hor}}(\mathcal{S})$.

   5. Mark $\mathcal{S}$ as a solution sequence and add the tuple $(\mathcal{S}, u(k), t(k), J_{\text{TTS,hor}})$ to $\mathcal{Y}$.

   6. Select from $\mathcal{Y}$ a sequence $\hat{\mathcal{S}}$ with the lowest performance index.

   7. (a) If $\hat{\mathcal{S}}$ is marked as a subsequence
      (b) then expand $\hat{\mathcal{S}}$ with all possible modes; denote the set of these new sequences by $\mathcal{X}$.
      7.1 For all sequences $\tilde{\mathcal{S}}$ in $\mathcal{X}$ **do**
         7.1.1 Minimize $J_{\text{TTS}}(\tilde{\mathcal{S}})$.
         7.1.2 If the problem is feasible
            then mark $\tilde{\mathcal{S}}$ as a subsequence and add $(\tilde{\mathcal{S}}, \tilde{u}^*(k), \tilde{t}^*(k), J_{\text{TTS}})$ to $\mathcal{Y}$.
      (c) else $\hat{\mathcal{S}}$ and the corresponding $\hat{u}(k)$, and $\hat{t}(k)$, $k = 1, \dots, K$ are the optimal mode sequence and optimal control inputs respectively. **Stop.**

   8. Minimize $J_{\text{TTS,hor}}(\tilde{\mathcal{S}})$. If the problem is feasible then add $(\tilde{\mathcal{S}}, \tilde{u}^*(k), \tilde{t}^*(k), J_{\text{TTS,hor}})$ to $\mathcal{Y}$, and mark it as a solution sequence.

   9. Go to step 6.

end

By step 6 all other solution sequences in $\mathcal{Y}$ will have higher or equal performance index than the one found in 7.(c). By monotonicity all solution sequences resulting from the subsequences in $\mathcal{Y}$ will result in a higher cost than the one found in step 7.(c). So, the one found in 7.(c) is optimal, and the problem formulated in Section 2.3 is solved.

4. CONCLUSIONS AND FUTURE RESEARCH

We have developed an approach that finds the optimal control inputs (such as ramp metering, dynamic speed limits, and route guidance signals) that minimizes total time that vehicles spend on a freeway network with bottlenecks and capacity drop behavior. The physical interpretation of this result is that this method finds the sequence of necessary control measures that solve all traffic jams – or if that is not possible – to reach the best possible relocation of traffic jams.

The topics for future research are the inclusion of blocking effects, the extension to dynamic demands, and the investigation of the computational complexity of the algorithm.
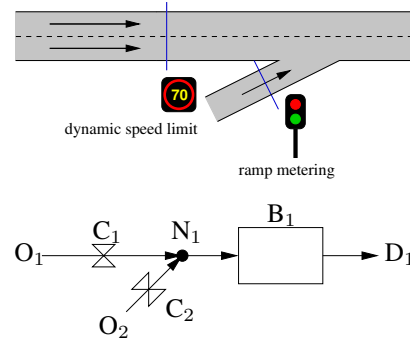
ACKNOWLEDGMENTS

Fig. 3. Example network with ramp metering and dynamic speed limits. Top: The physical lay-out of the network. Bottom: The network in terms of the defined network elements.

REFERENCES

Alessandri, A., A. Di Febbraro, A. Ferrara and E. Punta (1998). Optimal control of freeways via speed signalling and ramp metering. *Control Engineering Practice* **6**, 771–780.

Cassidy, M. J. and R. L. Bertini (1999). Some traffic features at freeway bottlenecks. *Transportation Research Part B* **B33**, 25 – 42.

Hall, F.L. and K. Agyemang-Duah (1991). Freeway capacity drop and the definition of capacity. *Transportation Research Record* (1320), 91–98.

Hegyi, A. (2004). Model Predictive Control for Integrating Traffic Control Measures. Ph.D. thesis, TRAIL thesis series T2004/2. Delft University of Technology. Delft, The Netherlands. ISBN 90-5584-053-X.

Kerner, B. S. (2002). Empirical features of congested patterns at highway bottlenecks. In: *Proceedings of the 81st Annual Meeting of the Transportation Research Board*. Washington, D.C.

Kotsialos, A., M. Papageorgiou, M. Mangeas and H. Haj-Salem (2002). Coordinated and integrated control of motorway networks via non-linear optimal control. *Transportation Research Part C* **10**, 65–84.

Morse, A.S. (1995). *Trends in Control*. Chap. Control using logic-based switching, pp. 69–114. Springer. London.

Nesterov, Y. and A. Nemirovskii (1994). *Interior-Point Polynomial Algorithms in Convex Programming*. SIAM. Philadelphia, Pennsylvania.

Pardalos, P.M. and Resende, M.G.C., Eds.) (2002). *Handbook of Applied Optimization*. Oxford University Press. Oxford, UK.

Russel, S. J. and P. Norvig (1995). *Artificial Intelligence – A modern approach*. Prentice Hall International.

Zhang, L. and D. Levinson (2004). Ramp metering and the capacity of active freeway bottlenecks. *Proceedings of the 83nd Annual Meeting of the Transportation Research Board*.