2018 IEEE Conference on Decision and Control (CDC)
Miami Beach, FL, USA, Dec. 17-19, 2018

A logic-based speed limit control algorithm for Variable Speed Limits to reduce traffic congestion at bottlenecks*

José Ramón D. Frejo[1] and Bart De Schutter[1]

Abstract— **This paper proposes a logic-based control algorithm for Variable Speed Limits (VSLs) in order to reduce or avoid traffic jams created at bottlenecks. The proposed controller estimates, for each controller time step, the number of vehicles that have to be held back or released by the VSLs in order to maximize the outflow of the bottleneck (avoiding the capacity drop). Afterward, based on the estimated number of vehicles, the VSLs are increased or decreased sequentially. The proposed controller uses a feed-forward structure that allows to anticipate the future evolution of the bottleneck density in order to avoid or reduce traffic breakdowns. As a result, although the implementation of the controller is quite easy with an almost instantaneous computation time, the performance of the controller is effective in reducing Total Time Spent (TTS).**

The proposed controller is tested, using the macroscopic traffic flow model METANET, for 10 scenarios and the results are compared with the ones obtained with the Mainstream Traffic Flow Control (MTFC) algorithm, and with the optimal solution. The simulations show that the proposed controller is able to approach the optimal behavior and that its behavior is quite robust (especially comparing with MTFC) in cases where different demands are considered.

I. INTRODUCTION

Nowadays, Variable Speed Limits (VSLs) are one of the most promising dynamic control signals for reducing traffic jams on freeway (see [1] for a review about the use of VSLs for freeway traffic control). In fact, during the past years and in contrast to safety-oriented applications such as [2], VSLs have emerged as a potential traffic management measure for increasing freeway efficiency [3], [4], [5], [6]. However, previously proposed optimal algorithm for VSLs [3], [6], [7], [8] are too complex to be implemented in real time, mainly because their computation time quickly increases with the size of the network and because, moreover, they are not robust in case of communication or measurement errors.

In order to overcome this practical problem, a few easy-to-implement VSL controllers for increasing freeway efficiency have been proposed previously [4], [5], [9], [10]. The most well known are SPECIALIST [4], a control algorithm based on shock wave theory, and MTFC [5], a local VSL controller that uses a cascade control structure with feedback of the density at the bottleneck area and the flow downstream of the VSL application area. SPECIALIST is able to solve/reduce isolated shock waves that do not necessarily always happen at the same time or that do not have the same magnitude. However, SPECIALIST is not designed to deal with congestion at bottlenecks. On the other hand, MTFC can successfully avoid the capacity drop and the onset of congestion at bottlenecks [11]. However, its behavior highly depends on the scenario for which the control parameters have been calibrated. Therefore, although MTFC is usually able to perform close to the optimal solution for the scenario used for calibration, the performance is substantially decreased if these control parameters are applied for substantially different traffic conditions.

This paper proposes an easy-to-implement logic-based VSL control algorithm for bottlenecks that allows to robustly approach the performance of an optimal controller for different demand profiles while the computation of the control inputs is almost instantaneous and the tuning of the control parameters is very simple and intuitive (compared with previously proposed control algorithms for VSL). This performance is achieved through the use of a feed-forward structure that anticipates the activation of the VSLs before the critical density is reached. This control structure allows to naturally activate the VSLs at the right time without relying on parameters that multiply integral terms that depend on the scenario (and on how fast the flow is decreasing or increasing). Therefore, since the performance of a VSL controller depends crucially on proper activation, the proposed controller is able to locally approach the Total Time Spent (TTS) reduction of an optimal controller.

This paper is organized as follows: Section II describes and justifies the proposed control algorithm. Subsequently, the simulation results are shown in Section III and, finally, the main conclusions are drawn in Section VI.

II. A LOGIC-BASED VSL CONTROL ALGORITHM

A. Control structure

The implementation of the proposed Logic-Based control algorithm for Variable Speed Limits (LB-VSL), for each controller time step k, is based on the structure shown in Fig. 1. Firstly, the number of vehicles that have to be held back/released in order to keep the bottleneck flow around the capacity is computed based on the density of the bottleneck and on the mean flow and speed upstream of the bottleneck. Subsequently, the VSLs are increased or decreased sequentially (starting with the most upstream segment) until enough vehicles have been held back/released by the VSLs or until the entire set of VSLs have been decreased/increased.

*This research was supported by the European Union's Horizon 2020 research and innovation programee under the Marie Skłodowska-Curie grant agreement No 702579.

[1]José Ramón D. Frejo and Bart De Schutter are with the Delft Center for Systems and Control, Delft University of Technology, The Netherlands. J.R.DominguezFrejo@tudelft.nl, B.DeSchutter@tudelft.nl

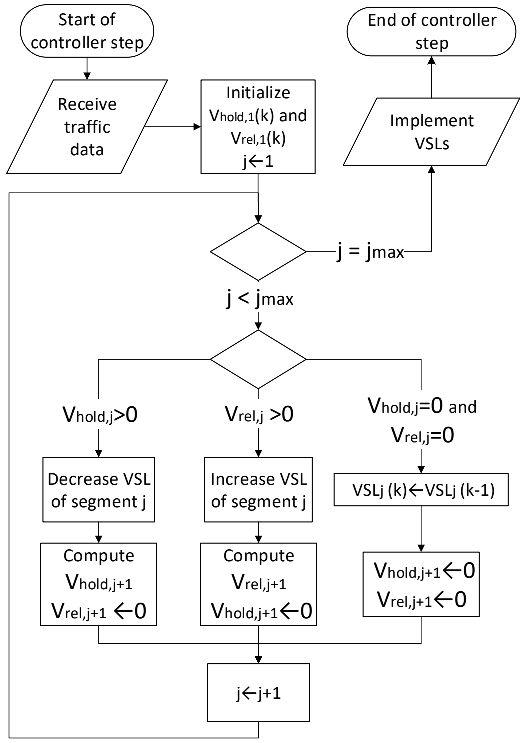

Fig. 1. Control structure for LB-VSL, where j is an index running over all the segments (upstream of the bottleneck) equipped with a VSL.

B. LB-VSL equations

This section shows the equations that are needed for the implementation of the proposed logic-based controller based on the structure shown in Fig. 1. The motivation and justification for using these equations is discussed in subsection II-D.

1) Total number of vehicles to hold back: Firstly, the total number of vehicles that have to be held back at time step k is estimated using the following equation:

$$V_{\text{hold},1}(k) = \max\left(0, T_{\text{ff}}(k)(Q_{\text{iB}}(k) - \overline{C_{\text{B}}}) - \lambda_{\text{B}}\text{L}_{\text{B}}(\rho_{\text{c,B}} - \rho_{\text{B}}(k))\right) \quad (1)$$

where $T_{\text{ff}}(k)$ is a certain period of time chosen such that all the vehicles located at time step k within a certain distance L_{A} upstream of the bottleneck, and not more, are able to reach the bottleneck, $Q_{\text{iB}}(k)$ is the average flow over all lanes (veh/h) entering the bottleneck during the considered period, λ_{B} is the number of lanes of the bottleneck, L_{B} is the length of the bottleneck, $\rho_{\text{c,B}}$ is the critical density of the bottleneck, and $\overline{C_{\text{B}}}$ is a tuning parameter that has to be set and that typically is around the capacity of the bottleneck. The period of time $T_{\text{ff}}(k)$ can be computed using the mean speed $\hat{v}_{\text{A}}(k)$ within the distance L_{A} upstream of the bottleneck: $T_{\text{ff}}(k) = L_{\text{A}}/\hat{v}_{\text{A}}(k)$. The selection of L_{A} and the estimation method used for $Q_{\text{iB}}(k)$ and $\hat{v}_{\text{A}}(k)$ are explained in section II-C.

2) Total number of vehicles to release: Equivalently, the total number of vehicles that have to be released at time step k is estimated using the following equation:

$$V_{\text{rel},1}(k) = \max\left(0, -T_{\text{ff}}(k)(Q_{\text{iB}}(k) - \underline{C_{\text{B}}}) + \lambda_{\text{B}}L_{\text{B}}(\rho_{\text{c,B}} - \rho_{\text{B}}(k))\right) \quad (2)$$

where $\underline{C_{\text{B}}}$ is a tuning parameter that has to be set between the the capacity of the bottleneck and the congested outflow (capacity minus capacity drop) of the bottleneck. It has to be taken into account that $V_{\text{hold},1}(k)V_{\text{rel},1}(k) = 0$ so, for any time step, either $V_{\text{hold},1}(k)$, $V_{\text{rel},1}(k)$, or both are equal to 0.

3) Decreasing a VSL: As shown in Fig. 1, if $V_{\text{hold,j}}(k))$ is higher than zero, the corresponding speed limit is decreased. The value of the VSL after the decrease is computed by using the equation below:

$$\text{VSL}_j(k) = \max\left(\frac{L_j\lambda_j v_j(k)\rho_j(k)}{(1+\alpha_j)(L_j\lambda_j\rho_j(k) + V_{\text{hold},j}(k))}, \underline{\text{VSL}}_j\right) \quad (3)$$

where $\text{VSL}_j(k)$ and $v_j(k)$ are the speed limit and the speed at time step k of the freeway segment j where the variable speed limit sign is located, λ_j, L_j, and α_j are the number of lanes, the length, and the compliance parameter of segment j, and $\underline{\text{VSL}}_j$ is the minimum speed limit that can be applied. If other implementation constraints, apart from the maximum and minimum value of the VSLs, are considered (maximum increase/decrease in one time step, spatial constraints...), they have to be included in (4).

4) Increasing a VSL: Equivalently, if $V_{\text{rel,j}}(k))$ is higher than zero, the corresponding VSL is increased:

$$\text{VSL}_j(k) = \begin{cases} \overline{\text{VSL}}_j & \text{if} \rho_j(k) \leq \frac{V_{\text{rel}}(k)}{L_j\lambda_j} \\ \min\left(\frac{L_j\lambda_j v_j(k)\rho_j(k)/(1+\alpha_j)}{(L_j\lambda_j\rho_j(k) - V_{\text{rel,j}}(k))}, \overline{\text{VSL}}_j\right) & \text{otherwise} \end{cases} \quad (4)$$

where $\overline{\text{VSL}}_j$ is the maximum speed limit that can be applied.

5) Updating V_{hold}: Finally, the remaining number of vehicles that have to be still held back (and thus considered for segment $j+1$) after decreasing the VSL of segment j is computed by:

$$V_{\text{hold},j+1}(k) = \max\left(0, (V_{\text{hold},j}(k) - V^{\text{hold}}_{\text{VSL},j}(k)\right) \quad (5)$$

where $V^{\text{hold}}_{\text{VSL},j}(k)$ is the number of vehicles that are going to be held back by the VSL of segment j which can be computed as:

$$V^{\text{hold}}_{\text{VSL},j}(k) = \lambda_j L_j \max\left(0, \frac{v_j(k)\rho_j(k)}{(1+\alpha_j)\text{VSL}_j(k)} - \rho_j(k)\right) \quad (6)$$

6) Updating V_{rel}: Equivalently, the remaining number of vehicles that have to be still released is computed by:

$$V_{\text{rel},j+1}(k) = \max\left(0, (V_{\text{rel},j}(k) + V^{\text{rel}}_{\text{VSL},j}(k)\right) \quad (7)$$

where $V^{\text{rel}}_{\text{VSL},j}(k)$ is the number of vehicles that are going to be released, which can be computed as:

$$V^{\text{rel}}_{\text{VSL},j}(k) = \lambda_j L_j \min\left(0, \frac{v_j(k)\rho_j(k)}{(1+\alpha_j)\text{VSL}_j(k)} - \rho_j(k)\right) \quad (8)$$

C. Entering flow and mean speed estimation

The implementation of (1) and (2) requires the selection of a distance L_{A} and the online estimation of $Q_{\mathrm{iB}}(k)$ and $\hat{v}_{\mathrm{A}}(k)$. The method used for this estimation can be adapted according to the number of detectors available and the topology of the network.

In this paper, we assume that there are measurements available in the entire set of segments upstream the bottleneck and that L_{A} is the distance between the segment where the first (upstream) considered VSL is installed and the bottleneck. Therefore, an easy and accurate way to estimate $Q_{\mathrm{iB}}(k)$ and $\hat{v}_{\mathrm{A}}(k)$ is to use the weighted summation of all the flows and speeds:

$$Q_{\mathrm{iB}}(k) = \frac{\sum_{i\in D_A} q_i(k)\hat{L}_i}{L_{\mathrm{A}}} \quad \hat{v}_{\mathrm{A}}(k) = \frac{\sum_{i\in D_A} v_i(k)\hat{L}_i}{L_{\mathrm{A}}} \quad (9)$$

where $q_i(k)$ and $v_i(k)$ are the flow and speed measured at detector on segment i, $\hat{L}_i$ is the distance between detector i and detector $i+1$, A is the stretch of the freeway between the first detector and the bottleneck, and D_A is the set of detectors in freeway stretch A.

In other cases (for instance, if only one detector upstream the bottleneck is available), other estimation methods can be used (as explained in [12]) without a substantial reduction in the obtained performance.

D. Justification

1) Number of vehicles to hold back/release: This paper proposes a control algorithm for VSLs that is able to anticipate the future evolution of the bottleneck density in order to avoid or reduce traffic breakdowns. This can be achieved by the use of a feed-forward control structure that makes use of the available measurements of speeds and flows upstream the bottleneck ($q_i(k)$, $v_i(k)$) in order to compute, for each control time step, the number of vehicles that have to be held back (or can be released) in order to avoid capacity drop after a certain period of time $T_{\mathrm{ff}}(k)$.

In order to define a convenient equation for these numbers of vehicles, a simplified model to predict the evolution of bottleneck density is used (because the goal is to propose an easy-to-implement controller). Firstly, the conservation of vehicles is used to predict the value of the density of the bottleneck after a given period of time $T_{\mathrm{ff}}(k)$:

$$\rho_{\mathrm{B}}(k+\frac{T_{\mathrm{ff}}(k)}{T}) = \rho_{\mathrm{B}}(k) + \frac{T_{\mathrm{ff}}(k)}{\lambda_{\mathrm{B}}L_{\mathrm{B}}}(Q_{\mathrm{iB}}(k) - Q_{\mathrm{ob}}(k)) \quad (10)$$

where $Q_{\mathrm{oB}}(k)$ is the outflow of the bottleneck during $T_{\mathrm{ff}}(k)$.

Subsequently, the previous equation is simplified using the following assumptions:

- The period of time $T_{\mathrm{ff}}(k)$ will be chosen such that all the vehicles located at time step k within the distance L_{A} upstream of the bottleneck, and not more, are able to reach the bottleneck.
- It is assumed that the vehicles entering the bottleneck between time step k and time step $k + T_{\mathrm{ff}}(k)/T$ have been traveling during this period at a mean speed $\hat{v}_{\mathrm{A}}(k)$ before they reach the bottleneck.
- The flow leaving the bottleneck is constant and equal to $\overline{C_{\mathrm{B}}}$ (for the computation of the number of vehicles to hold back) or $\underline{C_{\mathrm{B}}}$ (for the computation of the number of vehicles to release). This assumption is only true for $\overline{C_{\mathrm{B}}}$ if the bottleneck is being controlled around the critical density. However, this assumption is useful for control purposes as will be seen in the simulation results.

With these assumptions, and considering the we are computing the number of vehicles to hold back, (10) can be rewritten as:

$$\rho_{\mathrm{B}}(k+\frac{T_{\mathrm{kk}}(k)}{T}) = \rho_{\mathrm{B}}(k) + \frac{L_{\mathrm{A}}(Q_{\mathrm{iB}}(k) - \overline{C_{\mathrm{B}}})}{\lambda_{\mathrm{B}}L_{\mathrm{B}}\hat{v}_{\mathrm{A}}(k)} \quad (11)$$

Therefore, in order to keep $\rho_{\mathrm{B}}(k + T_{\mathrm{ff}}(k)/T)$ around the critical density it is necessary to hold back the following number of vehicles:

$$\begin{aligned} V_{\mathrm{hold,1}}(k) &= \lambda_{\mathrm{B}}L_{\mathrm{B}}(\rho_{\mathrm{B}}(k+\frac{T_{\mathrm{ff}}(k)}{T}) - \rho_{\mathrm{c,B}}) = \quad (12)\\ & T_{\mathrm{ff}}(k)(Q_{\mathrm{iB}}(k) - \overline{C_{\mathrm{B}}}) - \lambda_{\mathrm{B}}\mathrm{L}_{\mathrm{B}}(\rho_{\mathrm{c,B}} - \rho_{\mathrm{B}}(k)) \end{aligned}$$

Finally, taking account that $V_{\mathrm{hold,1}}(k) \geq 0$, (1) can be obtained from (12).

An equivalent reasoning can be undertaken for the number of vehicles to release ($V_{\mathrm{rel,1}}(k)$).

2) Decreasing/Increasing a VSL: In order to compute the number of vehicles that can be held back or released ($V_{\mathrm{VSL},j}(k)$) because a change in the value of the speed limit, the following assumptions are considered:

- The segment equipped with the VSL is uncongested.
- The flow of this segment after a certain period of time (relatively short) will be equal to the flow before the change of the speed limit value.
- The speed after the change of the speed limit value is the posted speed limit multiplied by a compliance factor $(1+\alpha_j)$.

Taking account of these assumptions, it can be stated that:

$$\begin{aligned} v_{\mathrm{bf},j}(k)\rho_{\mathrm{bf},j}(k) &= v_{\mathrm{af},j}(k)\rho_{\mathrm{af},j}(k) \quad (13)\\ v_{\mathrm{af},j}(k) &= (1+\alpha_j)\mathrm{VSL}_j(k) \end{aligned}$$

where $\rho_{\mathrm{bf},j}(k)$, $v_{\mathrm{bf},j}(k)$, $\rho_{\mathrm{af},j}(k)$, and $v_{\mathrm{af},j}(k)$ are, respectively, the densities and speeds of segment j before and after the change in the value of the speed limit.

In addition, the number of vehicles that are going to be held back/released by the VSL after a certain (relatively short) period of time is equal to:

$$V_{\mathrm{VSL},j}(k) = \lambda_j L_j(\rho_{\mathrm{af},j}(k) - \rho_{\mathrm{bf},j}(k)) \quad (14)$$

Therefore, using (13) and (14) and taking into account that $\rho_{\mathrm{bf},j}(k) = \rho_j(k)$ and $v_{\mathrm{bf},j}(k) = v_j(k)$, the value of the speed limit that has to be implemented in order to store or release $V_{\mathrm{VSL},j}(k)$ vehicles is obtained:

$$\mathrm{VSL}_j(k) = \frac{L_j\lambda_j v_j(k)\rho_j(k)}{(1+\alpha_j)(L_j\lambda_j\rho_j(k) + V_{\mathrm{VSL},j}(k))} \quad (15)$$

Finally, considering that $V_{\mathrm{VSL},j}(k) = V_{\mathrm{hold,j}}(k)$ if the vehicles have to be held back, that $V_{\mathrm{VSL},j}(k) = -V_{\mathrm{rel,j}}(k)$

if the vehicles have to be released, and that $V_{\mathrm{VSL},j}(k) \geq 0$, (3) can be obtained.

For (4), it has to be taken into account that, if $V_{\mathrm{rel,j}}(k) \geq L_j \lambda_j \rho_j(k)$, the corresponding VSL should be increased as much as possible but the computed variable speed limit (using (15)) would be negative. In order to avoid this undesirable effect, (4) is defined as a piece-wise function.

3) Updating V_{hold} and V_{hold}: Equations (5) and (7) update the value of $V_{\mathrm{hold}}(k)$ and $V_{\mathrm{rel}}(k)$ in order to compute the remaining number of vehicles that have to be held back or released by the VSLs located in downstream segments. This update is done by subtracting the number vehicles that have been already held back/released by the previous VSLs and taking into account that the remaining number of vehicles has to be higher than 0.

Finally, (6) and (8) compute, using (15) and taking account that $V^{\mathrm{hold}}_{\mathrm{VSL},j}(k)$ and $V^{\mathrm{rel}}_{\mathrm{VSL},j}(k)$ are higher than 0, the number of vehicles that have been already held back/released by the previous VSL.

E. Tuning of the controller parameters

The behavior of the proposed controller can be adapted by tuning the parameters $\overline{C_{\mathrm{B}}}$ and $\underline{C_{\mathrm{B}}}$.

The value of $\overline{C_{\mathrm{B}}}$ has to be initially set around the capacity of the bottleneck. However, $\overline{C_{\mathrm{B}}}$ can be reduced in order to advance in time the activation of the VSLs (decreasing the probability of reaching the capacity drop but increasing the probability of an unnecessary activation of the VSLs) or increased in order to delay the activation of the VSLs.

Equivalently, $\underline{C_{\mathrm{B}}}$ can be reduced or increased in order to advance or delay the deactivation of the VSLs. The value of $\underline{C_{\mathrm{B}}}$ has to be set between the congested congested outflow of the bottleneck (capacity minus capacity drop) and the capacity of the bottleneck. It has to be taken into that using a value for $\underline{C_{\mathrm{B}}}$ that is too close to the value of $\overline{C_{\mathrm{B}}}$ may create oscillations in the behavior of the VSLs.

III. Case Study

A. Considered scenarios

An hypothetical 12 km long freeway stretch (already used for the simulation of the FF-ALINEA algorithm in [12]), shown in Fig. 2, is used in order to simulate the proposed controller and to compare its performance with MTFC and the optimal solution:

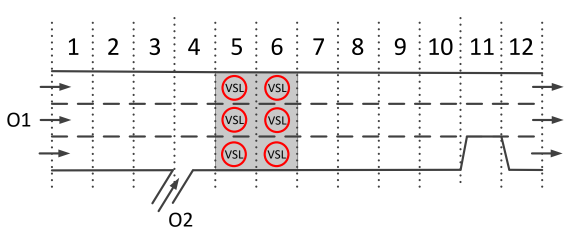

Fig. 2. Freeway stretch considered.

The freeway has $N = 12$ segments with $\lambda_i = 3$ lanes and with a length of $L_i = 1$ km for each segment, one controlled on-ramp at the beginning of segment 4 and one lane drop in segment 11 (i.e. segment 11 has only 2 lanes). Because of the lane drop, segment 11 is a bottleneck that will create congestion if the demands are high enough. There are two segments equipped with VSLs (segment 5 and 6). The VSLs are only allowed to take a limited number of discrete values ($\{40, 50, 60, 70, 80, 90, 100\}$) and they can only be increased or decreased with 10 km/h for each controller time step. These implementation constraints have been included, by rounding, in (3) and (4).

In order to simulate the considered freeway stretch, the METANET model [13] has been used. The effects of the VSLs have been included using the VSL model proposed in [3]. All the METANET parameters are considered to be the same for all the segments. The simulation time chosen is three hours, corresponding to 180 controller sample steps (because of the lenght of the controller time step is $T_{\mathrm{c}} = 60$ s) and 1080 simulation steps (because of the lenght of the simulation time step is $T = 10$ s). The on-ramp has a capacity of $C_{\mathrm{rel},4} = 2000$ veh/h, the free-flow speed v_{f} is 110 km/h, the critical density ρ_{c} is 32 veh/(km·lane), the maximum density ρ_{m} is 180 veh/(km·lane), and the time constant τ is 18 s. The rest of the model parameters can be seen in Table I. As proposed in [3], the model takes different values for μ (μ_{H} and μ_{L}) depending on the downstream density.

TABLE I
METANET PARAMETERS

a	μ_{H}	μ_{L}	ϕ	K	δ	α
2	40 km^2/h	80 km^2/h	0.1	40	0.01	0.1

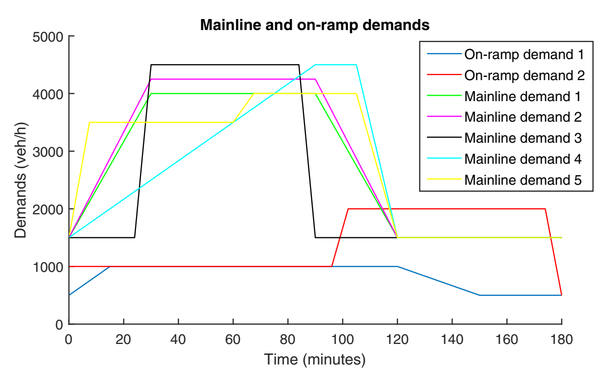

Fig. 3. Mainline and on-ramp demands

Five different mainline demands and two on-ramp demands (shown in Fig. 3) are considered. Combining the demands, ten scenarios, with different levels of congestion, are considered (the odd scenarios use the first on-ramp demand while the even scenarios use the second one). For the implementation of the proposed controller, it has been considered that there are flow and speed detectors available for all the segments between the on-ramp and the bottleneck (segments 4 to 11).

The critical density of the bottleneck (used by MTFC and the proposed controller) has been estimated using the flows and densities obtained by simulating METANET for Scenario 1 without any activation of the speed limits. As in previous references [12], [14], the obtained critical density ($\rho_{\mathrm{c,b}} = 36,78$ veh/(km·lane) is larger than the one given by the METANET fundamental diagram (32 veh/(km·lane).

The optimal solution for each scenario (an optimal controller optimized over the entire simulation horizon like in [12]) is also computed in order to obtain an estimation of the highest TTS reduction that can be achieved for each scenario. This computation of the optimal speed limits solution has been analyzed in many previous references [3], [6], [7].

B. Numerical results

In this section, the proposed controller is tested for the 10 scenarios considered and the results are compared with the ones obtained with MTFC and with the optimal solution for each scenario. The values of the controller parameters, for both the proposed algorithm and MTFC, have been found by minimizing the Total Time Spent (TTS) for the corresponding scenario. The obtained numerical results are shown in Table II:

TABLE II

TTS (VEH·H) FOR DIFFERENT CONTROLLERS AND SCENARIOS

Scen.	Uncon.	Optimal	MTFC	LB-VSL
1	2861	1904 (33.4%)	1966 (31.3%)	1942 (32.1%)
2	3957	2775 (29.9%)	2901 (26.7%)	2887 (27.0%)
3	3820	3205 (16.1%)	3209 (16.0%)	3213 (15.9%)
4	4909	4301 (12.4%)	4339 (11.6%)	4302 (12.4%)
5	3007	2609 (13.2%)	2725 (9.4%)	2645 (12.0%)
6	4082	3602 (11.8%)	3776 (7.5%)	3670 (10.1%)
7	2465	2104 (14.6%)	2153 (12.7%)	2126 (13.8%)
8	2896	2559 (11.7%)	2578 (11.0%)	2564 (11.5%)
9	2490	2025 (18.7%)	2065 (17.1%)	2049 (17.7%)
10	2782	2249 (19.2%)	2272 (18.3%)	2261 (18.7%)
Mean		-18.1%	-16.1%	-17.1%

Analyzing the results, it can be seen that the proposed controller is able to approach the optimal performance for the entire set of considered scenarios. In fact, the highest difference between the TTS reduction of the proposed controller and the optimal TTS reduction occurs for Scenario 2 (with a TTS reduction of 29.9% for the optimal case and 27.0% for the proposed controller). As a result, the mean TTS reduction for the 10 scenarios is also quite close the optimal one (17.1% versus 18.1%).

MTFC is also able to substantially improve the behavior of the traffic system with a 16.1% mean TTS reduction for the 10 scenarios. However, the performance is slightly worse than using the proposed controller for most of the considered scenarios.

C. Cross-validation

In real applications, the values of the controller parameters will be usually computed for one case (generally, the typical demand) and applied for different scenarios. Therefore, it is important that the controllers tuned for one scenario also perform properly in other circumstances. In other words, it is necessary to have a robust controller, especially against different demand profiles.

This section analyzes the robustness, against different mainline demands and ramp queue constraints, of MTFC and the proposed controller (by analyzing the TTS reduction for scenario j when using the parameters computed for scenario i). Moreover, the optimal value of the controller parameters (K_{I}^{*}, $K_{\mathrm{I}}^{'}{}^{*}$, $K_{\mathrm{P}}^{'}{}^{*}$, $\overline{C_{\mathrm{B}}}^{*}$ and $\underline{C_{\mathrm{B}}}^{*}$) that minimize the summation of the cost functions for the 10 scenarios have been also computed and included in the comparison (in the second-to-last column of the tables).

The results shown in Table III show that the proposed logic-based controller is quite robust for different traffic conditions. In fact, for the simulated scenarios, the performance obtained with different values of $\overline{C_{\mathrm{B}}}^{*}$ and $\underline{C_{\mathrm{B}}}^{*}$ is always close the one obtained with the optimal value of the controller parameters.

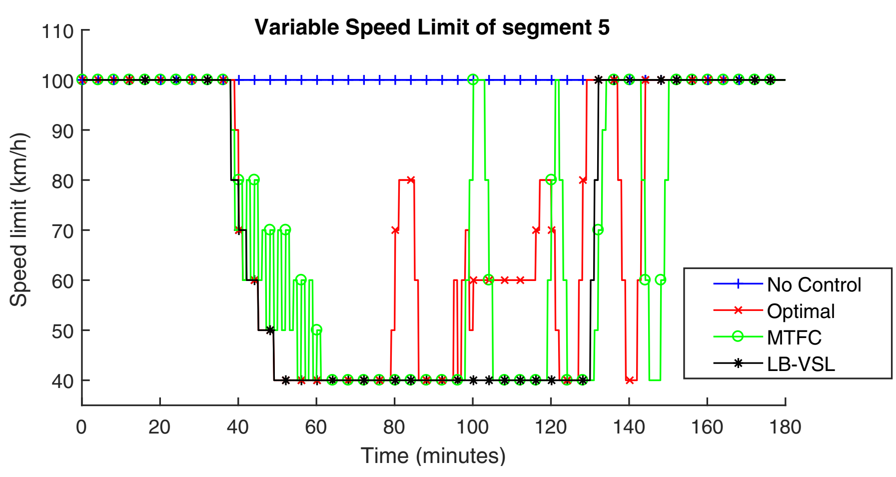

Fig. 4. Variable Speed Limit of segment 5 for Scenario 1

On the other hand, it can be seen in Table IV that the behavior of MTFC is much more dependent on the value of the parameters than the one obtained with the proposed controller. In fact, the best mean TTS reduction for the 10 scenarios that can be obtained using MTFC is 13.9% while using the proposed controller the best mean TTS reduction is 17.0%, slightly lower than the optimal one (18.1%). Moreover, for scenario 9, MTFC increases the TTS if incorrect values of the parameters are used.

D. Graphical results

Finally, Fig. 4, 5, and 6 show the graphical results for a representative scenario: Scenario 1, which uses the mainline demand 1 and on-ramp demand 1 (shown in Fig. 3).

Firstly, the VSLs obtained using each controller are shown in Fig. 4 and 5. It can be seen that the proposed controller accurately approaches the activation of the optimal controller for both segments. On the other hand, MTFC shows oscillations during this period (this could be avoided by changing the parameters of the controller entailing a reduction in performance in terms of TTS).

When a congestion queue is created (around minute 80), the behavior shown by the proposed controller and MTFC starts to differ from the optimal one. However, the impacts of VSLs (in terms of TTS and bottleneck outflow) are minimal if the congestion queue is already created (as can be seen in the numerical results of this paper and it has been observed by the authors in previous research). In fact, once the congestion tail has been created, one can argue that a safety-oriented VSL control algorithm may have more

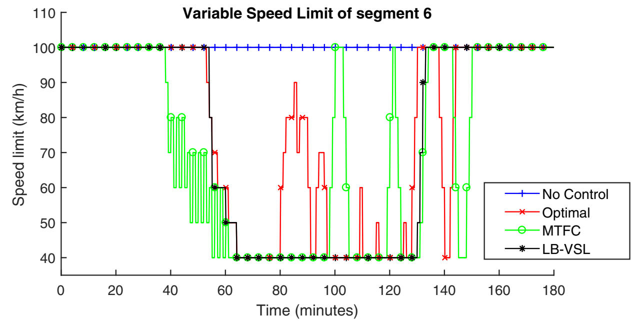

Fig. 5. Variable Speed Limit of segment 6 for Scenario 1

TABLE III

CROSS-VALIDATION: TTS REDUCTION WITH RESPECT TO THE NO-CONTROL CASE FOR LB-VSL WITH $\overline{C_{\mathrm{B}}}^i$ AND $\underline{C_{\mathrm{B}}}^i$ OPTIMIZED FOR SCENARIO i AND WITH THE OPTIMAL VALUE OF THE PARAMETERS $\overline{C_{\mathrm{B}}}^*$ AND $\underline{C_{\mathrm{B}}}^*$

	$\overline{C_{\mathrm{B}}}^1 =$ 4815.1 $\underline{C_{\mathrm{B}}}^1 =$ 3383.2	$\overline{C_{\mathrm{B}}}^2 =$ 4824.0 $\underline{C_{\mathrm{B}}}^2 =$ 3260.0	$\overline{C_{\mathrm{B}}}^3 =$ 4808.1 $\underline{C_{\mathrm{B}}}^3 =$ 3694.1	$\overline{C_{\mathrm{B}}}^4 =$ 4821.0 $\underline{C_{\mathrm{B}}}^4 =$ 3699.8	$\overline{C_{\mathrm{B}}}^5 =$ 4801.6 $\underline{C_{\mathrm{B}}}^5 =$ 3680.0	$\overline{C_{\mathrm{B}}}^6 =$ 4791.0 $\underline{C_{\mathrm{B}}}^6 =$ 3362.1	$\overline{C_{\mathrm{B}}}^7 =$ 4798.8 $\underline{C_{\mathrm{B}}}^7 =$ 3387.5	$\overline{C_{\mathrm{B}}}^8 =$ 4801.0 $\underline{C_{\mathrm{B}}}^8 =$ 3360.4	$\overline{C_{\mathrm{B}}}^9 =$ 4800.2 $\underline{C_{\mathrm{B}}}^9 =$ 2975.1	$\overline{C_{\mathrm{B}}}^{10} =$ 4824.0 $\underline{C_{\mathrm{B}}}^{10} =$ 3200.0	$\overline{C_{\mathrm{B}}}^* =$ 4824.0 $\underline{C_{\mathrm{B}}}^* =$ 3380.0	Mean
Scenario 1	**32.1**%	32.0%	31.4%	31.5%	31.9%	31.9%	31.9%	31.9%	31.7%	32.0%	32.1%	31.8%
Scenario 2	27.0%	**27.0**%	26.5%	26.6%	26.9%	26.7%	26.9%	26.8%	26.8%	26.4%	27.0%	26.8%
Scenario 3	15.7%	15.8%	**15.9**%	15.9%	15.6%	15.6%	15.6%	15.6%	15.6%	15.8%	15.7%	15.3%
Scenario 4	12.2%	12.2%	12.4%	**12.4**%	12.0%	12.0%	12.0%	12.0%	9.6%	12.2%	12.2%	11.8%
Scenario 5	11.9%	11.9%	12.1%	12.1%	**12.0**%	12.0%	11.9%	12.0%	9.5%	11.9%	11.9%	11.6%
Scenario 6	10.1%	9.9%	10.0%	10.0%	10.1%	**10.1**%	10.1%	10.1%	7.2%	9.8%	10.1%	9.7%
Scenario 7	13.7%	13.7%	13.2%	13.2%	13.7%	13.7%	**13.8**%	13.7%	12.8%	13.6%	13.7%	13.4%
Scenario 8	11.4%	11.4%	11.2%	11.2%	11.4%	11.5%	11.4%	**11.5**%	10.2%	11.3%	11.4%	11.2%
Scenario 9	17.5%	17.4%	16.9%	16.9%	17.5%	17.5%	17.4%	17.5%	**17.7**%	16.8%	17.5%	17.4%
Scenario 10	18.6%	18.6%	18.7%	18.7%	18.5%	18.5%	18.5%	18.5%	18.3%	**18.7**%	18.6%	18.6%
Mean	17.0%	17.0%	16.8%	16.8%	17.0%	16.9%	17.0%	17.0%	15.6%	16.9%	**17.0**%	16.8%

TABLE IV

CROSS-VALIDATION: TTS REDUCTION WITH RESPECT TO THE NO-CONTROL CASE FOR MTFC WITH K_{I}^i, $K_{\mathrm{I}}'^{\,i}$, AND $K_{\mathrm{P}}'^{\,i}$ OPTIMIZED FOR SCENARIO i AND WITH THE OPTIMAL VALUE OF THE PARAMETERS K_{I}^*, $K_{\mathrm{I}}'^{\,*}$, AND $K_{\mathrm{P}}'^{\,*}$

	$K_{\mathrm{I}}^1 =$ 0.0707 $K_I'^{\,1} =$ 2.6502 $K_{\mathrm{P}}'^{\,1} =$ 109.49	$K_{\mathrm{I}}^2 =$ 0.0695 $K_{\mathrm{I}}'^{\,2} =$ 3.2405 $K_{\mathrm{P}}'^{\,2} =$ 116.88	$K_{\mathrm{I}}^3 =$ 0.1013 $K_{\mathrm{I}}'^{\,3} =$ 3.2070 $K_{\mathrm{P}}'^{\,3} =$ 119.76	$K_{\mathrm{I}}^4 =$ 0.0632 $K_{\mathrm{I}}'^{\,4} =$ 3.1310 $K_{\mathrm{P}}'^{\,4} =$ 116.13	$K_{\mathrm{I}}^5 =$ 0.0531 $K_{\mathrm{I}}'^{\,5} =$ 4.4644 $K_{\mathrm{P}}'^{\,5} =$ 118.79	$K_{\mathrm{I}}^6 =$ 0.0619 $K_{\mathrm{I}}'^{\,6} =$ 2.4170 $K_{\mathrm{P}}'^{\,6} =$ 118.45	$K_{\mathrm{I}}^7 =$ 0.0598 $K_{\mathrm{I}}'^{\,7} =$ 2.4170 $K_{\mathrm{P}}'^{\,7} =$ 118.05	$K_{\mathrm{I}}^8 =$ 0.0661 $K_{\mathrm{I}}'^{\,8} =$ 1.7524 $K_{\mathrm{P}}'^{\,8} =$ 117.25	$K_{\mathrm{I}}^9 =$ 0.0675 $K_{\mathrm{I}}'^{\,9} =$ 1.5820 $K_{\mathrm{P}}'^{\,9} =$ 114.41	$K_{\mathrm{I}}^{10} =$ 0.0426 $K_{\mathrm{I}}'^{\,10} =$ 4.4031 $K_{\mathrm{P}}'^{\,10} =$ 116.73	$K_{\mathrm{I}}^* =$ 0.0879 $K_{\mathrm{I}}'^{\,*} =$ 3.37 $K_{\mathrm{P}}'^{\,*} =$ 117.77	Mean
Scen. 1	**31.3**%	27.0%	29.5%	29.9%	28.6%	25.7%	16.7%	15.6%	28.8%	26.6%	30.5%	26.4%
Scen. 2	24.9%	**26.7**%	25.6%	25.7%	24.7%	21.6%	12.0%	13.2%	25.3%	24.3%	25.5%	22.7%
Scen. 3	14.1%	14.4%	**16.0**%	14.4%	13.2%	12.9%	10.3%	9.2%	10.8%	13.7%	13.9%	13.0%
Scen. 4	9.6%	10.4%	9.9%	**11.6**%	10.2%	9.9%	7.7%	6.7%	8.2%	9.3%	11.4%	9.5%
Scen. 5	7.7%	6.2%	8.3%	6.2%	**9.4**%	8.7%	7.2%	7.9%	7.5%	6.7%	7.9%	7.6%
Scen. 6	5.3%	3.9%	6.7%	3.9%	7.5%	**7.5**%	5.9%	6.9%	5.1%	5.9%	6.5%	5.9%
Scen. 7	11.1%	11.1%	8.2%	11.1%	11.2%	9.2%	**12.7**%	11.6%	11.0%	11.0%	11.1%	10.8%
Scen. 8	6.2%	6.2%	6.4%	6.2%	6.7%	9.2%	8.7%	**11.0**%	6.8%	8.7%	7.0%	7.6%
Scen. 9	4.5%	4.1%	6.3%	5.8%	12.4%	−5.2%	−10.9%	−10.9%	**17.1**%	−1.3%	10.1%	2.9%
Scen. 10	17.5%	16.9%	17.0%	16.7%	14.6%	17.4%	9.7%	6.8%	13.2%	**18.3**%	15.0%	14.8%
Mean	13.2%	12.7%	13.4%	13.1%	13.9%	11.7%	8.0%	7.8%	13.4%	12.3%	**13.9**%	12.1%

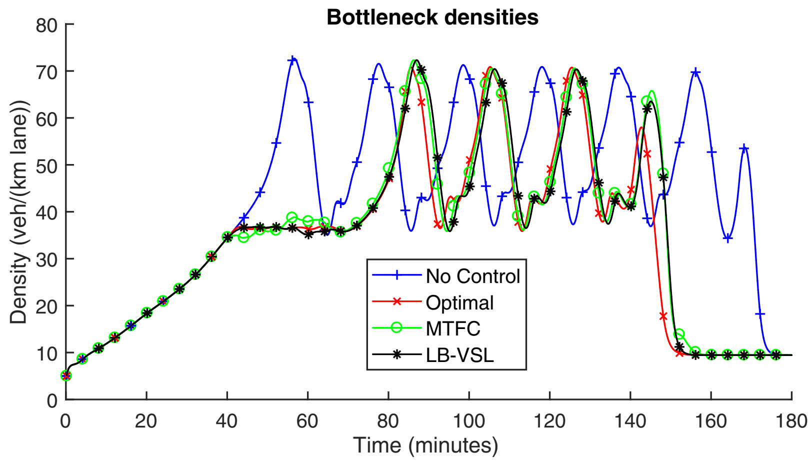

Fig. 6. Density of segment 11 for Scenario 1

benefits than a traffic-efficiency one.

Finally, in Fig. 6, the bottleneck densities obtained using each controller are shown. Again, it can be seen that MTFC and the proposed controller show similar behavior as the optimal controller (especially before the congestion tail is created), whereby the proposed controller approximates the optimal control more closely.

IV. CONCLUSIONS

This paper has proposed a new logic-based control algorithm for variable speed limits to reduce traffic congestion at bottlenecks. The proposed controller is based on the estimation of the number of vehicles that have be held back/released in order to keep the bottleneck outflow around the capacity. Based on this number, the variable speed limits are increased or decreased sequentially.

For the 10 scenarios considered, the simulation results have shown that the proposed controller is able to approach the optimal behavior while being quite robust for different demand profiles. Moreover, the results have also shown a performance increase, in terms of TTS reduction, with respect to MTFC.

The main advantage of the proposed controller, compared with previously proposed controllers for variable speed limits, is that it is quite easy-to-implement, since the computation of the control inputs is almost instantaneous and the tuning of the control parameters is very simple and intuitive, while its performance is effective and robust.

In future works, the proposed controller will be tested for real networks and integrated in the framework of a joint controller for ramp metering and variable speed limits (LB-TFC).

REFERENCES

[1] X. Lu and S. Shladover, *Review of variable speed limits and advisories: theory, algorithms, and practice*, Transportation Research Record: Journal of the Transportation Research Board, vol. 2423, pp. 15-23, 2014.

[2] Z. Li, P. Liu, W. Wang, and C. Xu, *Development of a control strategy of variable speed limits to reduce rear-end collision risks near freeway recurrent bottlenecks*, IEEE Transactions on Intelligent Transportation Systems, vol. 15, no. 2, pp. 866-877, 2014.

[3] A. Hegyi, B. De Schutter, and H. Hellendoorn, *Model predictive control for optimal coordination of ramp metering and variable speed limits*, Transportation Research Part C: Emerging Technologies, vol 13, no. 3, pp. 185-209, 2005.

[4] A. Hegyi, S.P. Hoogendoorn, M. Schreuder, H. Stoelhorst, and F. Viti, *SPECIALIST: A dynamic speed limit control algorithm based on shock wave theory*, 11th International IEEE Conference on Intelligent Transportation Systems, pp. 827-832, 2008.

[5] R.C. Carlson, I. Papamichail, and M. Papageorgiou, *Local feedback-based mainstream traffic flow control on motorways using variable speed limits*, IEEE Transactions on Intelligent Transportation Systems, vol. 12, no. 4, pp. 1261-1276, 2011.

[6] J.R.D. Frejo, A. Núñez, B. De Schutter, and E.F. Camacho, *Hybrid model predictive control for freeway traffic using discrete speed limit signals*, Transportation Research Part C: Emerging Technologies, vol. 46, pp. 309-325, 2014.

[7] R.C. Carlson, I. Papamichail, M. Papageorgiou, and A. Messmer, *Optimal motorway traffic flow control involving variable speed limits and ramp metering*, Transportation Science, vol. 44, no. 2, pp. 238-253, 2010.

[8] J.R.D. Frejo and E. F. Camacho, *Global versus local MPC algorithms in freeway traffic control with ramp metering and variable speed limits*, IEEE Transactions on Intelligent Transportation Systems, vol. 13, no. 4, pp. 1556-1565, 2012.

[9] H. Jin, and W. Jin, *Control of a lane-drop bottleneck through variable speed limits*, Transportation Research Part C: Emerging Technologies, vol. 58, pp. 568-584, 2015.

[10] M. Kontorinaki, I. Karafyllis, and M. Papageorgiou, *Global exponential stabilisation of acyclic traffic networks.* International Journal of Control, pp. 1-21, 2017.

[11] E.R. Muller, R.C. Carlson, W. Kraus, and M. Papageorgiou, *Microsimulation analysis of practical aspects of traffic control with variable speed limits*, IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 1, pp. 512-523, 2015.

[12] J.R.D. Frejo and B. De Schutter, *Feed-Forward ALINEA: A ramp metering control algorithm for nearby and distant bottlenecks*, IEEE Transactions on Intelligent Transportation Systems, accepted for publication.

[13] M. Papageorgiou, I. Papamichail, A. Messmer and Y. Wang, *Traffic simulation with METANET*, Fundamentals of Traffic Simulation, Springer, 2010.

[14] Y. Wang, E.B. Kosmatopoulos, M. Papageorgiou and I. Papamichail, *Local ramp metering in the presence of a distant downstream bottleneck: Theoretical analysis and simulation study*, IEEE Transactions on Intelligent Transportation Systems, vol. 15 (5), pp. 2024 - 2039, 2014.