Technical report 13-021

Model predictive control for freeway traffic using discrete speed limit signals[*]

J.R.D. Frejo, A. Núñez, B. De Schutter, and E.F. Camacho

---

[*]This report can also be downloaded via https://pub.deschutter.info/abs/13_021.html

Model predictive control for freeway traffic using discrete speed limit signals

José Ramón D. Frejo[1], Alfredo Núñez,[2], Bart De Schutter,[3] and Eduardo F. Camacho[1]

Abstract— **A model predictive control (MPC) approach for freeway traffic control using discrete signals for the activation of variable speed limit panels is proposed. The discrete characteristics of the panels and some necessary constraints for their real operation are usually underestimated in the literature, so first we propose a way to include them using the macroscopic traffic model METANET within an MPC framework. For obtaining practical discrete signals, the MPC controller has to solve a highly non-linear optimization problem, including mixed-integer variables. Since solving such a problem is complex and difficult to execute in real-time, we propose some methods to obtain reasonable control actions in a limited computation time. The methods consist of first relaxing the discrete constraints for the inputs; and then, based on this continuous solution and together with different search methods, to find discrete speed limit signals that provide the best performance, keeping the number of simulations reduced. The proposed methods are tested by simulation, showing not only a good performance but also keeping the computational time reduced.**

I. INTRODUCTION

Traffic congestion on freeways is a critical problem due to its negative impact on the environment and many other important consequences (higher delays, waste of fuel, a higher accident risk probability, etc.) It has been reported in the literature under different conditions and particularities that dynamic traffic control is a good solution to decrease congestion [1] [2] [3]. In general, dynamic traffic control uses measurements of the traffic conditions over time and computes dynamic control signals to influence the behavior of the drivers and then to generate a response in such a way that the performance of the network is improved, by reducing for example the delays, emissions, etc.

Variable speed limits (VSL), ramp metering, and route guidance are most used examples of measures that can be used to dynamically control traffic. These measures have been already successfully implemented in USA, Germany, Spain, Netherlands and other countries [4]. When selecting the control signals, among the available options described in the literature, the methods based on the use of advanced control techniques like Model Predictive Control (MPC)

[5] have by simulation proved to substantially improve the performance of the controlled traffic system [6] [2]. In this paper, we focus on control using variable speed limits (VSL) panels. In most of the works about VSL computed with MPC, the VSL signals are assumed to have continuous values, meaning that the real VSL panels implemented in the network should display to the driver those values [2], [6], [1], [7], [8]. However, in real implementations of VSL panels, the displayed signals are just allowed to take a limited set of discrete values. Moreover, some extra constraints should be considered like a limited variation in time (for each panel) and space (consecutive panels) so to avoid drastic changes in speed for safety reasons. As a real implementation example, in [3] a method to reduce the effect of shock waves in the Dutch freeway A12 using VSL panels is reported, where the signals of the panels were just allowed to take values in the set $\{60, 80, 100\}$[km/h].

A few works have tried to deal with discrete VSL. In [9], a traffic model with variable length segments is used to compute a simple best-effort controller that reduces congestion considering VSL signals that can only be decreased or increased by steps of 10[km/h]. A discrete VSL controller based on shock wave theory is proposed and tested in [3]. Both controllers, [9] and [3], use simple control laws that are not explicitly designed to optimize a performance index of the network. In [10] and [11] the VSL are discretized (by rounding, ceiling or flooring) after computing them in a continuous way. These papers conclude that the performance of the discrete safety-constrained speed limits was comparable with the continuous case. However, those results are dependent of the network configuration and the demand conditions, as in our case study we found some important loss of performance due to the discretization.

In this paper, we consider explicitly the effects of using discrete signals for the VSL panels in an MPC framework. After the formulation of the control problem in Section II, III and IV, methods to obtain discrete signals for the VSL panels are presented in Section V. The first algorithm evaluates all the feasible discrete solutions near the continuous solution provided by the relaxed MPC controller. The second algorithm evaluates a subset of the feasible discrete solutions using a genetic algorithm. Finally, in Section IV, numerical results are presented and discussed.

II. MODEL PREDICTIVE CONTROL

The main concepts behind a MPC strategy are the use of a prediction model to obtain the trajectories of relevant variables of the system, the optimization of a dynamic

objective function to determine the best sequence of control actions for the system, and the application of the rolling horizon procedure, so from the best sequence of control actions only the first component is applied to the system and in the next instant the initial conditions are updated and the procedure is repeated again [5]. Since MPC is based in an optimization procedure, in the case of systems with discrete control actions we can include that characteristic as a constraint, and obtain suitable control laws by using a proper mixed-integer optimization algorithm.

To formalize all these concepts, consider the discrete-time non-linear system whose dynamic evolution is described by the following state space model:

$$x(k+1) = f(x(k), u(k), d(k)) \tag{1}$$

with $x(k)$ the state, $u(k)$ the discrete and continuous input vector, and $d(k)$ the non-controllable input vector, usually demands profiles and other exogenous variables.

In an MPC controller, the core is the optimization of a cost function $J(x_t(k), u_t(k), d_t(k))$, which is used to measure the performance of the system. The vectors $x_t(k)$ and $d_t(k)$ are the state and non-controllable input predictions along the prediction horizon N_p and u_t is the control sequence along the control horizon N_u. As the formulation is based on the solution of an optimization problem, it is possible to explicitly include constraints. Assume that the feasible values of the states and the inputs are given by the following generic constraints $x(k) \in \mathbb{X}$, $u^\mathrm{c}(k) \in \mathbb{U}$, and $u^\mathrm{d}(k) \in \mathbb{S}$ for all k, representing explicitly physical or operational constraints of the system. When considering the case of discrete inputs, the MPC problem can be formulated as the following mixed-integer non-linear optimization problem:

$$\min_{u_t(k)} J(u_t(k), x_t(k), d_t(k)) \tag{2}$$

subject to:
$$x(k+\ell+1) = f(x(k+\ell), u(k+\ell), d(k+\ell)),$$
$$x(k) = x_k \quad x(k+\ell+1) \in \mathbb{X},$$
$$u^\mathrm{c}(k+\ell) \in \mathbb{U}, \qquad u^\mathrm{d}(k+\ell) \in \mathbb{S},$$
$$\text{for } \ell = 0, 1, ..., N_\mathrm{p} - 1,$$

with $\mathbb{S}$ the set of possible control values $\mathbb{S} = \{u_1, u_2, ..., u_M\}$ for the corresponding feasible discrete input components (in our case, the VSL). Using the rolling horizon procedure, only the first control action $u(k)$ of the optimal sequence is applied to the system, and in the next time step the initial conditions are updated and the procedure is repeated.

For this formulation, the main disadvantage is the computation time needed to solve the optimization problem. In cases when the problem can be recast into a linear one, well-known optimization methods are efficient and many software/toolboxes are available to solve them. However, in the case of traffic systems, the problem is intrinsically non-linear and any simplification of the model may lead to non-acceptable predictions, which are incapable of incorporating the real behavior of traffic. As a way to deal with the complexity, and to include the discrete characteristic explicitly in

the solution, we propose first to reduce the search space of possible solutions by including some operational constraints that are actually very necessary for safety reasons, and then to use some fast heuristic methods to find a good solution within that limited space.

III. TRAFFIC MODEL METANET

In this paper, we have selected the traffic model METANET [12]. However, the methods we propose are independent of the traffic model used, so they can be equivalently applied using other macroscopic traffic models, if they are capable to include the effect of VSL in its formulation (like some versions of the CTM [13]).

The METANET model is a macroscopic second-order traffic model that provides a good trade-off between simulation speed and accuracy. The METANET model is deterministic and can be adapted to motorway networks of arbitrary topology and characteristics, including motorway stretches, bifurcations, on-ramps and off-ramps, and the effects of control actions such as ramp metering, route guidance, and VSL panels. This model discretizes the freeway in consecutive sections in segments of length L_i and uses density $\rho_i(k)$ and speed $v_i(k)$ as state variables. For simplicity, in this paper the authors do not differentiate between links and segments. The model is as follows:
- **Density equation**:

$$\rho_i(k+1) = \rho_i(k) + \tag{3}$$
$$+ \frac{T}{\lambda_i L_i}(q_{i-1}(k) - q_i(k) + q_{\mathrm{r},i}(k) - \beta_i(k)q_{i-1}(k))$$

where λ_i is the number of lanes, $\beta_i(k)$ is the split ratio for an off ramp on segment i $\beta_i(k) = 0$ for a segment without an off-ramp), T is the model sample time, $q_i(k)$ is the flow leaving segment i and $q_{\mathrm{r},i}(k)$ is flow entering by an on-ramp on segment i $q_{\mathrm{r},i}(k) = 0$ for a segment without an on-ramp).
- **Speed equation**:

$$v_i(k+1) = v_i(k) + \frac{T}{\tau_i}(V(\rho_i(k)) - v_i(k)) + \tag{4}$$
$$+ \frac{T}{L_i}v_i(k)(v_{i-1}(k) - v_i(k)) - \frac{\mu_i T}{\tau_i L_i}\frac{\rho_i(k+1) - \rho_i(k)}{\rho_i(k) + K}$$
$$- \frac{\delta_i T q_{r,i}(k) v_i(k)}{L_i \lambda_i (\rho_i(k) + K)}$$

where K_i, τ_i, δ_i and μ_i are model parameters and $V(\rho_i(k))$ is the speed desired for the drivers. As proposed in [1], the model can take different values for μ_i, depending on whether the downstream density is higher or lower than the density in the corresponding segment.
- **Desired speed equation**:

$$V(\rho_i(k)) = \tag{5}$$
$$\min\left(v_{\mathrm{f},i}\exp\left(-\frac{1}{a_i}\left(\frac{\rho_i(k)}{\rho_{\mathrm{c},i}}\right)^{a_i}\right), (1+\alpha_i)V_{\mathrm{c},i}(k)\right)$$

where a_i, α_i are model parameters, $\rho_{\mathrm{c},i}$ is the critical density, $v_{\mathrm{f},i}$ is the free flow speed and $V_{\mathrm{c},i}$ is the variable speed limit.

- **Ramp flow equation**:

$$q_{r,i}(k) = \qquad\qquad\qquad\qquad\qquad (6)$$
$$= \min\left(r_i(k)C_{\mathrm{r},i}, D_i(k) + \frac{w_i(k)}{T}, C_{\mathrm{r},i}\frac{\rho_{\mathrm{m},i} - \rho_i(k)}{\rho_{\mathrm{m},i} - \rho_{\mathrm{c},i}} \right)$$

where $C_{\mathrm{r},i}$ is the ramp capacity, $D_i(k)$ is the ramp demand, $w_i(k)$ is the ramp queue, $\rho_{\mathrm{m},i}$ is the maximum density, and $r_i(k)$ is the ramp metering rate.

- **Flow equation**:

$$q_i(k) = \lambda_i \rho_i(k) v_i(k) \qquad\qquad (7)$$

- **Queue length equation:**

$$w_i(k_{\mathrm{m}} + 1) = w_i(k_{\mathrm{m}}) + T_{\mathrm{m}} \cdot (D_i(k_{\mathrm{m}}) - q_{\mathrm{r},i}(k_{\mathrm{m}})) \quad (8)$$

For the sake of simplicity, merge and join nodes, and other extensions are not considered in this paper (See [12], [1], [14] for further details).

In [12], VSLs are included in the model rendering the three parameters of the fundamental diagram $\rho_{\mathrm{c},i}$, $v_{\mathrm{f},i}$ and a_i. However, in this paper, the VSL are included by a minimum term in the desired speed equation (5) as proposed in [1]. the effect of VSLs on aggregate traffic flow behavior is studied in [15] on the basis of traffic data comparing the equation used in this paper (5) with other options.

IV. VSL IMPLEMENTATION ISSUES

In the majority of the references about VSL computed by MPC, the variable speed limits are considered to be implemented in the network without any operational constraints. Hereafter, we will assume the MPC controllers will have to consider the following constraints:

A. Temporal constraint

In real implementations, the VSL cannot change abruptly due to driver safety and comfort. In this section, it is supposed that the VSLs are just allowed to change γ km/h as maximum in a controller sample. Therefore, it is necessary to recompute the control inputs in order to have a feasible solution including the following hard constraint in the MPC approach:

$$|V_{c,i}(k + l) - V_{c,i}(k + l - 1)| \leq \gamma \qquad (9)$$
$$for\ i \text{ with a VSL and}$$
$$for\ l = 0, 1, ... N_{\mathrm{u}} - 1$$

Hereafter, the MPC proposed with VSL temporally constrained will be denoted by T-MPC. In this paper, a value of $\gamma = 10$ will be used.

B. Spatial and temporal constrained case

Moreover, in real implementations it is necessary to limit the difference between the VSLs of two adjacent segments. This can be done by including the following constraint:

$$|V_{c,i}(k + l) - V_{c,i}(k + l - 1)| \leq \gamma \qquad (10)$$
$$|V_{c,i+1}(k + l) - V_{c,i}(k + l)| \leq \zeta$$
$$for\ [i + 1, i] \text{ with VSL and}$$
$$for\ l = 0, 1, ... N_{\mathrm{u}} - 1$$

Hereafter, the MPC proposed with VSL spatially and temporally constrained will be denoted by ST-MPC. In this paper, a value of $\zeta = 10$ will be used.

C. Discrete case

Even with MPC constrained spatially and temporally for the VSLs, the solution obtained cannot be implemented in practice since the freeway signs are only allowed to show a limited set of speed limits. However, the MPC controller is considering the VSL as continuous signal and, therefore, is providing real numbers for implementation. In practical case, it is necessary to approximate this decimal control inputs by an allowed VSL value. Some methods for this discretization will be proposed in the following section. In the following sections, it will be assumed that the set of allowed VSL is $\mathbb{S} = \{20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120\}$, $\zeta = 10$, and $\gamma = 10$.

V. VSL DISCRETIZATION

A. Rounding discretization

The easiest and most used way to do the discretization is by rounding to the closest available value for the VSLs:

$$V_{c,i}(k) = \mathrm{argmin}_{s \in \mathbb{S}}(|V_{\mathrm{cST},i}(k) - s|) \qquad (11)$$

where $V_{cST,i}(k)$ is the VSL of segment i computed by ST-MPC controller explained in the previous section IV.B. In [1] the discretization is also done by ceiling and flooring obtaining a slightly better performance for the ceiling case and a worse performance for the flooring case. In the rest of this section alternative methods for the discretization step will be proposed in order to decrease the loss of performance due to the quantization.

B. Search tree

During the discretization process, it is desirable to choose the discrete control input profile that minimizes the cost function. This can be done by solving the following optimization problem:

$$\min_{V_{\mathrm{c}}} \quad J(k) \qquad\qquad\qquad (12)$$
$$s.t: \quad |V_{\mathrm{c},i}(k + l) - V_{\mathrm{c},i}(k + l - 1)| \leq 10$$
$$|V_{\mathrm{c},i+1}(k + l) - V_{\mathrm{c},i}(k + l)| \leq 10$$
$$20 \leq V_{\mathrm{c},i+1}(k + l) \leq 120$$
$$for\ l = 0, 1, ... N_{\mathrm{u}} - 1$$
$$with\ V_{\mathrm{c}} \in \mathbb{S}$$

where V_{c} is the discrete control input profile: $[V_{\mathrm{c},1}(k), ..., V_{\mathrm{c},1}(k+N_{\mathrm{u}}-1), ..., V_{\mathrm{c},N_{\mathrm{VSL}}}(k), ..., V_{\mathrm{c},N_{\mathrm{VSL}}}(k + N_{\mathrm{u}} - 1)]^T$ and N_{VSL} is the number of VSL.

Most of the model predictive control approaches for hybrid large-scale systems do not have a standard strategy to relax the problem in order to obtain an acceptable good solution in a reasonable amount of computation time. Limiting the number of feasible nodes in the mixed integer optimization problem is a strategy that has been used before in the context of MPC for sewer networks [16], as a way to reduce

the combinatorial explosion related to the search tree of binary solutions at the expense of a suboptimal solution. In this paper, this limitation will be done by including a new constraint in the optimization problem making use of the ST-MPC solution $V_{\mathrm{cST},i}(k)$:

$$|V_{\mathrm{c},i}(k+l) - V_{\mathrm{cST},i}(k+l)| \leq \theta \qquad (13)$$

where θ is a tuning parameter. If the parameter θ is chosen to be 10, we are allowing that the VSLs take the closest upper or the closest lower value with respect to the continuous optimum $V_{\mathrm{cST},i}(k)$:

$$V_{\mathrm{c},i}(k) = \lfloor \frac{V_{\mathrm{cST},i}(k)}{10} \rfloor, \lceil \frac{V_{\mathrm{cST},i}(k)}{10} \rceil \qquad (14)$$

The feasible VSL set can be represented by a search tree. In Fig. 1 an example of a search tree with $N_u = 2$, $V_c(k) = [40, 50]^T$, $V_{\mathrm{cST}}(k) = [43, 53]^T$, $V_{\mathrm{cST}}(k) = [52, 61]^T$ and $\theta = 10$ is shown. It is supposed that there are only VSL on segments 3 and 4, as in the scenario in section IV. In this case, the constraints reduce the number possible combinations of discrete VSL from 81 to only 6.
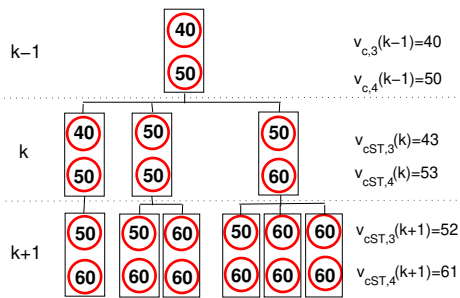

Fig. 1. Search tree example with $N_u = 2$

For higher horizons the number of feasible VSL profiles is increased exponentially. However, it is still possible to significantly reduce this number by eliminating the non-feasible solutions. For example, in one of the simulation done in this paper (with $N_u = 5$ and $\theta = 10$) the search tree is reduced from 59049 points to an average number of 700 points. If θ is chosen to be 14, the average number of points in the reduced tree is 5850.

C. Explicit enumeration

The easiest way to solve problem (12) with constraint (14) is to evaluate the cost function for all the feasible points in the reduced search tree. The main problem is the computation time needed for the evaluation of such a large possible combinations of discrete VSL. Therefore, this solution is just available for small networks with small horizons. In order to be able to solve the problem for higher networks and horizons with a limited computational time available, a genetic algorithm is proposed in the following section.

D. Genetic discretization

In order to determine a good discrete VSL within the limited time available, a genetic algorithm (GA) is proposed. It

has been reported that genetic algorithms can efficiently cope with the optimization of mixed integer nonlinear problems [17] in the context of model predictive control applications [18]. The main advantages of this method are that the gradient of the objective function does not need to be calculated and that by fixing few parameters it is possible to limit the number of function evaluations and iterations so as to obtain near optimal results within a fixed sampling time.

A candidate solution in the genetic algorithm is called an individual, and each individual has a fixed number of genes (emulating a chromosome). Each gene, in the context of the control of the VSL panels, will represent a possible input during the control horizon. The individual can be seen then as a candidate control action sequence:

$$\mathrm{Individual}_1 = \qquad (15)$$
$$= [u_1(k), ..., u_1(k+N_u-1), u_2(k), ..., u_m(k+N_u-1)]$$

where an element $u_i(k+j-1) \in \mathbb{S}$, $j \in \{1, N_u-1\}$ is a gene, i denotes the control actions related with the ith VSL, and the individual length corresponds to the control horizon N_u times the number of VSL m.

In genetic algorithm the idea is to find the fittest individuals (solutions with good objective function value) within a generation, to apply genetic operators for the recombination of those individuals, and to generate a good offspring [17]. For the selection, a roulette method is applied, having the best individuals more chances to be selected for recombination. For the recombination, two fundamental operators are used: crossover and mutation. For the crossover, portions of the chromosomes of two individuals are exchanged with a given probability p_c; and the mutation operator modify each gene randomly with a given probability p_m.

In the case a solution does not satisfy a constraint, we will penalize with a high objective function value. We have to point out that due to the limited time reaching the global optimum is not guaranteed. Since for traffic control the optimization is a complex mixed integer and nonlinear problem, using the GA optimization is justified. Many different approaches and adaptations of genetic algorithms have been proposed in the literature [19] in order to deal with many issues like constraint handling, diversity of the solutions, combination with classical optimization methods to assure a local convergence, etc. The algorithm we used for the traffic application is the simplest one [18].

VI. SIMULATION

A. Particularities of the MPC controllers used in this paper

This subsection explains the main particularities of the MPC controllers used in this paper, as in [20]:
–The MPC controller uses an objective function containing one term for the Total Time Spent (TTS) and two terms that penalize abrupt variations in the ramp metering and VSL.
–The controlled system is subject to constraints on the maximum and minimum values of densities, speeds, queues, ramp metering rates and VSL.
– The constraints on speed, density, and queue are made soft

by including them as penalization terms in the overall cost function.

–The optimization is computed by SQP optimization techniques by the Matlab function fmincon with a control and prediction horizon of $N_{\mathrm{u}} = 5$ and $N_{\mathrm{p}} = 10$, respectively.

—In order to try to avoid that the algorithm ends up in a local minimum, the algorithm runs an evaluation procedure before the optimization. During the procedure, the TTS is evaluated for a grid of control values. The best control values obtained are taken as initial values for the optimizations.
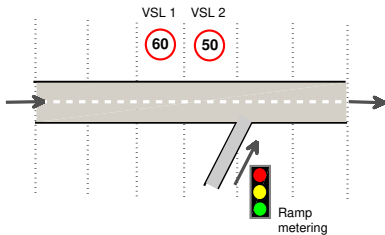
B. Set-up and scenario

Fig. 2.   Stretch used as example.

In order to simulate the analyzed controllers, the benchmark network in Fig. 2 has been used. The network has been taken from [1]. The freeway has $N = 6$ segments with a longitude of $L_i = 1000$ m and with $\lambda = 2$ lanes. There are 3 control signals: two VSL (for segments 3 and 4) and a ramp metering. Thirteen variables (density and speed of each segment and queue of the ramp) are supposed to be measured at each controller sample step and used for the computation of the control signals. The simulation time chosen is two and half hour corresponding to 75 controller sample steps ($T_{\mathrm{c}} = 120$s) and 900 simulation steps ($T = 10$s). See [1] [20] for further details about the chosen benchmark including the model parameters. The set of allowed VSL is supposed to be: $\mathbb{S} = \{20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120\}$.

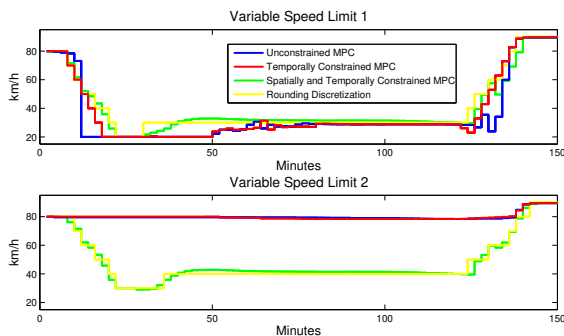### C. Continuous MPC controllers results

Fig. 3.   VSL of segment 3 and 4 for continuous MPC and rounding discretization

The results of the closed-loop simulations for the VSL computed by each continuous MPC controller and the rounding discretization of the ST-MPC can be seen on Fig. 3. It can

be seen that VSL1 of unconstrained MPC suddenly change from 73 km/h to 20 km/h at minute 12. However, this change is done at a rate of ten km/h for the T-MPC. Moreover, the difference between VSL1 and VSL2 is around 50 km/h for unconstrained MPC and T-MPC during a long period of time. When this difference is constrained in ST-MPC, the value of the VSL2 is strongly decreased in order to obtain a difference of 10 kilometers/hour with the VSL1 (which is also slightly increased). Finally, the discrete case is just a quantization of ST-MPC.

	TTS Reduction (%)
Uncontrolled System	0
Unconstrained MPC	12.66
Temporally constrained MPC	11.92
Spatially and temporally constrained MPC	8.10
Rounding discretization	4.99

The numerical results can be seen on Table I. It can be seen how the VSL that could be really implemented (the discrete ones) have a TTS reduction that is the 39.5% of the corresponding to the unconstrained MPC (12.66% versus 4.99%). This shows that the supposition done in many previous of paper about the continuous implementation of the VSL entails a large loss of performance for the controlled system.
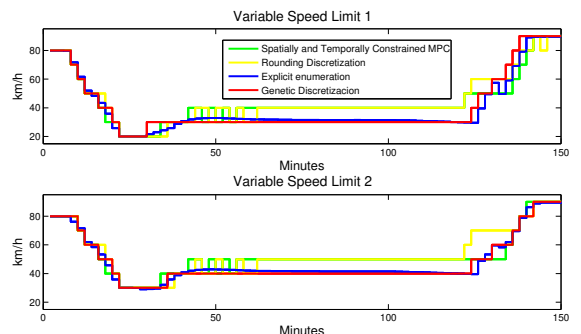
D. Discretization results

Fig. 4.   VSL discretization of segment 3 and 4.

The VSL discretization of ST-MPC and the continuous ST-MPC solution are shown on Fig. 4. The numerical results can be seen on Table II where it can be seen that θ is a key parameter in the performance of the discretization. The explicit enumeration with $\theta = 10$ is only a 6.5% better than the rounding discretization (5.28% versus 4.96% in the TTS reduction). However, choosing $\theta = 14$, the explicit enumeration is a 44% better than the rounding case (7.14% versus 4.96%). In this benchmark, values of θ higher than 14 do not bring much increase in the performance of the controlled system as can be seen in TTS reduction of the case with $\theta = 18$.

TABLE II

DISCRETIZATION PERFORMANCES

	TTS Reduction (%)
Rounding discretization	4.96
Explicit enumeration with $\theta = 10$	5.28
Explicit enumeration with $\theta = 14$	7.14
Explicit enumeration with $\theta = 18$	7.15
Genetic discretization with $\theta = 10$	5.20
Genetic discretization with $\theta = 14$	7.06

Finally, it can be seen than the genetic algorithm gives a solution that is very close to the explicit one but with a lower computational times (5.20% versus 5.28% for $\theta = 10$ and 7.06% versus 7.14% for $\theta = 14$).

In fact, the computation times show that the proposed genetic algorithm reduces substantially the computational load with respect to the explicit enumeration: For $\theta = 10$, the average computational time needed for the discretization is 25.69 s for the explicit case and 2.52 s for the genetic algorithm and, for $\theta = 14$, the computation time is 170.65 s and 9.38 s, respectively.

VII. CONCLUSIONS

Firstly, this paper has analyzed the effect of converting the continuous unconstrained Variable Speed Limits (VSL) signal to an implementable VSL signal. In the network simulated, the performance is reduced with respect to ideal case (i.e. the case with continuous VSL) in a 60.5% (12.66% versus 4.99%). Therefore, it can be concluded that the supposition about the continuous implementation of the VSL entails in some cases a large loss of performance for the controlled system. Since the majority of the literature makes this supposition, the authors propose two methods to reduce part of this loss of performance.

The first algorithm (explicit enumeration) evaluates all the feasible discrete solutions around the solution provided by the continuous MPC optimization problem using a search tree. This method is able to improve the solution in a 44% for the simulation analyzed. Unfortunately, this method usually entails unacceptable computation times.

The second algorithm (genetic discretization) relaxes the previous algorithm using a genetic algorithm. The algorithm is able to closely approximate the behavior of the explicit enumeration (A total time spent reduction of 7.06% versus 7.14% for explicit enumeration). However, with the genetic discretization, the computation time is acceptable (9.38 seconds of average computational time versus 170.65 seconds for the explicit case). Moreover, the trade-off between simulation speed and accuracy can be adapted online to the available time in each controller sample time making the algorithm very useful for the application in practical cases.

REFERENCES

[1] 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*, 13(3):185–209, June 2005.

[2] A. Muralidharan and R. Horowitz. Optimal control of freeway networks based on the link node cell transmission model. In *American Control Conference (ACC), 2012*, pages 5769 –5774, june 2012.

[3] A. Hegyi and S.P. Hoogendoorn. Dynamic speed limit control to resolve shock waves on freeways - field test results of the specialist algorithm. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 519 –524, sept. 2010.

[4] E. van den Hoogen and S. Smulders. Control by variable speed signs: results of the dutch experiment. In *Road Traffic Monitoring and Control, 1994., Seventh International Conference on*, pages 145 –149, apr 1994.

[5] E. F. Camacho and C. A. Bordons. *Model Predictive Control*. Springer-Verlag, London, 2004.

[6] J.R.D. Frejo and E.F. Camacho. Feasible cooperation based model predictive control for freeway traffic systems. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pages 5965 –5970, dec. 2011.

[7] S.K. Zegeye, B. De Schutter, J. Hellendoorn, E.A. Breunesse, and A. Hegyi. A predictive traffic controller for sustainable mobility using parameterized control policies. Technical Report 12-005, Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands, February 2012. Accepted for publication in *IEEE Transactions on Intelligent Transportation Systems*.

[8] G. Gomes and R. Horowitz. Optimal freeway ramp metering using the asymmetric cell transmission model. *Transportation Research Part C: Emerging Technologies*, 14(4):244 – 262, 2006.

[9] C.C. de Wit. Best-effort highway traffic congestion control via variable speed limits. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pages 5959 –5964, dec. 2011.

[10] A. Hegyi, B. De Schutter, and J. Hellendoorn. Optimal coordination of variable speed limits to suppress shock waves. *IEEE Transactions on Intelligent Transportation Systems*, 6(1):102–112, 2005.

[11] R. Castelan Carlson, I. Papamichail, and M. Papageorgiou. Local feedback-based mainstream traffic flow control on motorways using variable speed limits. *Trans. Intell. Transport. Sys.*, 12(4):1261–1276, December 2011.

[12] M. Papageorgiou, I. Papamichail, A. Messmer, and Y. Wang. Traffic simulation with metanet. In Jaume Barcel and Frederick S. Hillier, editors, *Fundamentals of Traffic Simulation*, volume 145 of *International Series in Operations Research and Management Science*, pages 399–430. Springer New York, 2010.

[13] C. F. Daganzo. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, 28(4):269 – 287, 1994.

[14] Jos Ramn Domnguez Frejo, Eduardo Fernndez Camacho, and Roberto Horowitz. A parameter identification algorithm for the metanet model with a limited number of loop detectors. In *CDC 2012*, pages 6983–6988. IEEE, 2012.

[15] M. Papageorgiou, E. Kosmatopoulos, and I. Papamichail. Effects of variable speed limits on motorway traffic flow. *Transportation Research Record*, 2047(-1):37–48, 2008.

[16] A. Ingimundarson, C. Ocampo-Martinez, and A. Bemporad. Model predictive control of hybrid systems based on mode-switching constraints. In *CDC'07*, New Orleans, LA, 2007.

[17] K. F. Man, K. S. Tang, and S. Kwong. *Genetic Algorithms: Concepts and Designs with Disk*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2nd edition, 1999.

[18] A. Núnez, D. Sáez, S. Oblak, and I. Akrjanc. Fuzzy-model-based hybrid predictive control. *ISA Transactions*, 48(1):24 – 31, 2009.

[19] H. Sarimveis and G. Bafas. Fuzzy model predictive control of non-linear processes using genetic algorithms. *Fuzzy Sets and Systems*, 139(1):59 – 80, 2003.

[20] J.R.D. Frejo and E.F. Camacho. Global versus local mpc algorithms in freeway traffic control with ramp metering and variable speed limits. *Intelligent Transportation Systems, IEEE Transactions on*, PP(99):1 –10, 2012.