Delft University of Technology

Fac. of Information Technology and Systems **Control Systems Engineering**

Technical report bds:99-03a

Optimal control of a class of linear hybrid systems with saturation: Addendum*

B. De Schutter

October 1999

Control Systems Engineering
Faculty of Information Technology and Systems
Delft University of Technology
Delft, The Netherlands
Current URL: https://www.dcsc.tudelft.nl

* This report can also be downloaded via https://pub.bartdeschutter.org/abs/99_03a.html

Optimal control of a class of linear hybrid systems with saturation: Addendum

Bart De Schutter

Abstract

In this addendum we give some extra propositions, proofs and results of numerical experiments in connection with the design of (sub)optimal switching time sequences for a class of first order linear hybrid systems with saturation that we have introduced in the paper "Optimal control of a class of linear hybrid systems with saturation" [A-1] (by B. De Schutter, *SIAM Journal on Optimization and Control*, vol. 38, no. 3, pp. 835–851, 2000).

All references in this addendum that are not preceded by a capital letter A, B or C refer to sections, equations, etc. of the paper [A-1].

A Additional remarks for [A-1]

A.1 Model

The model derived in [A-1] can accommodate varying amber durations. However, in many countries the amber time is fixed by regulation (e.g. to 3 s in France). If we assume that the duration of the amber phase is fixed, then we can adapt our model and reduce the number of variables (see also [8]).

A.2 Systems with varying rate functions

In [A-1] we have already explained that the MPC approach also allows us to deal with slowly-varying rate functions by updating the model after each switching and re-computing the optimal switching sequence.

Even if the rates are assumed to be non-constant within one MPC step, we can still use our approach if we approximate the time-varying rate functions by piecewise constant functions. Although in general we do not know the exact behavior of these functions in advance the behavior can often be predicted on the basis of historical data and measurements. Also note that we do not know the lengths of the phases in advance. In order to determine the average rates for each phase, we could therefore first assume that all phases have equal length. Then we compute an optimal or suboptimal switching time sequence and use the result to get better estimates of the lengths of the phases and thus also of the average queue length growth rates in each phase, which can then be used as the input for another optimization run. If necessary we could repeat this process in an iterative way.

Also note that in practice there is always some uncertainty and variation in time of the queue length growth rates, which makes that in general computing the exact optimal switching time sequence is utopian. Moreover, in practice we are more interested in quickly obtaining a good approximation of the optimal switching time sequence than in spending a large amount of time to obtain the exact optimal switching time sequence.

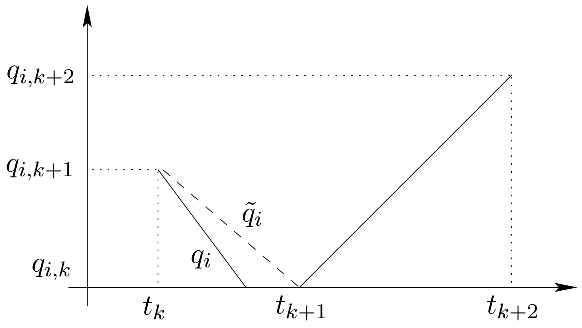

Figure A.1: The functions q_i (full line) and $\tilde{q}_i$ (dashed line) for a queue with a decrease phase and a subsequent positive growth phase and without saturation at an upper level. Note that in the growth phase the functions q_i and $\tilde{q}_i$ coincide.

A.3 Approximations of the objective functions

Let $l \in \{1, 4\}$. The value of the objective functions J_l and $\tilde{J}_l$ introduced in Sections 3.2 and 4.2 depends on the surface under the functions q_i and $\tilde{q}_i$ respectively. For a traffic signal controlled intersection where the traffic signals alternate between green and red (with a short amber phase in between) we will usually have a queue length evolution that is similar to the one represented in Figure A.1. An optimal traffic signal switching scheme implies the absence of long periods in which no cars wait in one lane (i.e. $q_i(t) = 0$) while in the other lanes the queue lengths increase. So in that case the surface under the function $\tilde{q}_i$ will be a reasonable approximation of the surface under the function q_i and then the optimal value of $\tilde{J}_l$ will be a reasonably good approximation of the optimal value of J_l.

Since $\tilde{q}_i$ is a piecewise-affine with breakpoints $(t_k, q_{i,k})$ for $k = 0, 1, \ldots, N_\mathrm{p}$, we have

$$\int_{t_k}^{t_{k+1}} \tilde{q}_i(t, x_q, x_\delta) dt \;=\; \frac{\delta_k}{2}(q_{i,k} + q_{i,k+1})$$

and thus

$$\tilde{J}_1(x_q, x_\delta) \;=\; \sum_{i=1}^{M} \left(\frac{w_i}{2\,(\delta_0 + \delta_1 + \ldots + \delta_{N_\mathrm{p}-1})} \sum_{k=0}^{N_\mathrm{p}-1} \delta_k (q_{i,k} + q_{i,k+1}) \right) \;.$$

B Additional numerical experiments and simulations for the example of Section 5.2

The switching interval vectors of the example of Section 5.2 are given by

$$\begin{aligned}
x^*_{\delta,\mathrm{elcp}} &= [\,10.04 \;\; 3.00 \;\; 38.75 \;\; 3.00 \;\; 39.88 \;\; 3.00 \;\; 70.00 \;\; 3.00\,]^T \\
x^*_{\delta,\mathrm{nlcon}} &= [\,10.04 \;\; 3.00 \;\; 30.75 \;\; 3.00 \;\; 39.88 \;\; 3.00 \;\; 70.94 \;\; 3.00\,]^T \\
x^*_{\delta,\mathrm{penalty}} &= [\,10.04 \;\; 3.00 \;\; 30.75 \;\; 3.00 \;\; 39.88 \;\; 3.00 \;\; 70.94 \;\; 3.00\,]^T \\
x^*_{\delta,\mathrm{relaxed}} &= [\,10.04 \;\; 3.00 \;\; 38.75 \;\; 3.00 \;\; 39.88 \;\; 3.00 \;\; 70.94 \;\; 3.00\,]^T \\
x^*_{\delta,\mathrm{approx}} &= [\,10.04 \;\; 3.00 \;\; 38.75 \;\; 3.00 \;\; 39.88 \;\; 3.00 \;\; 70.94 \;\; 3.00\,]^T
\end{aligned}$$

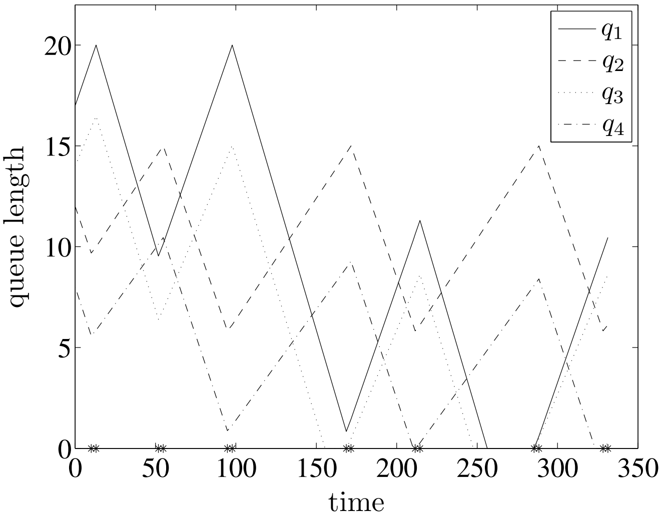

Figure A.2: The queue lengths in the various lanes as a function of time for the traffic signal switching sequence that corresponds to the switching interval vector $x^*_{\delta,\mathrm{elcp}}$ of the example of Section 5.2. The * signs on the time axis correspond to the switching time instants.

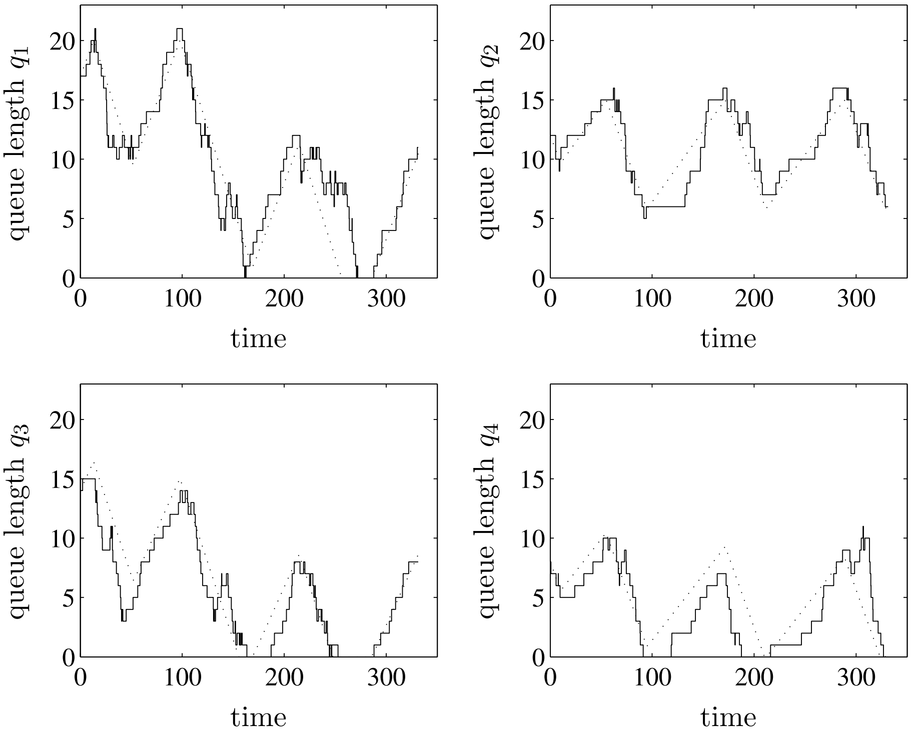

Figure A.3: The queue lengths in the various lanes as a function of time for an integer queue length simulation for the traffic signal switching sequence that corresponds to the switching interval vector $x^*_{\delta,\mathrm{elcp}}$. The integer queue length functions are plotted in full lines and their continuous approximations in dotted lines.

	$J_1(x^*_\delta)$			CPU time		
x^*_δ	$N_c = 4$	$N_c = 6$	$N_c = 10$	$N_c = 4$	$N_c = 6$	$N_c = 10$
$x^*_{\delta,\mathrm{elcp}}$	70.69	54.14	–	4.10	485.04	–
$x^*_{\delta,\mathrm{nlcon}}$	70.69	54.14	46.41	115.31	169.59	282.82
$x^*_{\delta,\mathrm{penalty}}$	70.69	54.14	46.41	9.59	20.92	39.28
$x^*_{\delta,\mathrm{relaxed}}$	79.69	54.14	46.41	0.16	0.26	0.50
$x^*_{\delta,\mathrm{approx}}$	70.81	56.24	53.66	2.73	2.96	3.73
$x^*_{\delta,\mathrm{lp}}$	70.69	54.14	47.54	0.14	0.15	0.19

Table A.1: The values of the objective functions J_1 and the CPU time needed to compute the (sub)optimal switching interval vectors of the example of Section 5.2 for $N_c = 4$, 6 and 10.

$$x^*_{\delta,\mathrm{lp}} = [\, 10.04 \;\; 3.00 \;\; 38.75 \;\; 3.00 \;\; 38.81 \;\; 3.00 \;\; 68.89 \;\; 3.00 \,]^T \ .$$

The evolution of the queue lengths for the optimal switching interval vector $x^*_{\delta,\mathrm{elcp}}$ is represented in Figure A.2. In Figure A.3 we have plotted the results of an integer queue length simulation for the traffic signal switching strategy that corresponds to the optimal switching interval vector $x^*_{\delta,\mathrm{elcp}}$. The effective average queue length over all lanes for this simulation is 45.17.

In order to show how the control horizon N_c influences the performance of the methods presented in [A-1] we have computed optimal and suboptimal switching time intervals for three different values of N_c: 4, 6 and 10. The other data and parameters have the same values as in Section 5.2. In Table A.1 we have given the optimal value of the objective function J_1 and the CPU time needed to compute the solution. We have not computed the ELCP solution for $N_c = 10$, since this would require too much CPU time.

We see again that the $x^*_{\delta,\mathrm{approx}}$ solution offers the best trade-off between efficiency and optimality.

While performing numerical experiments for the example of Section 5.2 and for similar examples, we noticed the following:

- The determination of the minimum value of J_1 over the solution set of the ELCP is a well-behaved problem in the sense that using a local minimization routine starting from different initial points almost always yields the same numerical result (within a certain tolerance). In a typical experiment in which for each face we computed the minimal value of the objective function for 20 random starting points, for almost every face 10 or more decimal places of the final objective function were the same. Therefore, we have only considered one run with an arbitrary random initial point for each face for the ELCP solution in Table 2.

- In order to obtain a good approximation to the optimal switching time vector using nonlinear constrained optimization or a penalty function approach, it is necessary to run the local minimization algorithm several times each time with a different initial starting point. In general a local minimization run for the approach that uses nonlinear constraints requires less time than a run for the penalty function approach. However, the nonlinear constraints approach requires more different starting points to obtain a good approximation of the global optimum than the penalty function approach.

- Apart from the quadratic penalty function defined by

$$F_{\text{penalty}} = 10\,000 \sum_{k=1}^{N_{\text{p}}} \sum_{i=1}^{M} \Big(\max\big(0,\, (q_k)_i - (q_{\max,\text{k}})_i\big)\Big)^2 \;, \tag{A.1}$$

  we have also used linear, exponential and mixed penalty functions in the penalty function approach. However, for the applications considered here the penalty function defined by (A.1) leads to the best performance.

- The relaxed problem (which has a convex feasible set) is much easier to solve using multistart local optimization than the original problem (which has a non-convex feasible set). In a typical experiment for $x^*_{\delta,\text{relaxed}}$ with 20 random starting points the first 11 decimal places of the final objective function J_1 always had the same value. For $x^*_{\delta,\text{approx}}$ the first 9 decimal places always had the same value. This implies that in practice performing one run with an random starting point is sufficient to obtain the globally optimal solution for $x^*_{\delta,\text{relaxed}}$ and $x^*_{\delta,\text{approx}}$.

C A generalization

C.1 A more general class of systems

In this section we show that the results of Sections 3 and 4 can be extended to a more general class of systems that can be described by equations of the form

$$q_{k+1} = \max\big(\min\big(A_k q_k + B_k u_k,\, b_k^{\text{us}}\big),\, b_k^{\text{ls}}\big) \tag{A.2}$$

for $k = 0, 1, 2, \ldots$ where $A_k \in \mathbb{R}^{M\times M}$, $B_k \in \mathbb{R}^{M\times L}$, and $u_k \in \mathbb{R}^L$ for some integer L. Note that (3) is a special case of (A.2) with $A_k = I$, $B_k = \alpha_k$, $u_k = \delta_k$ and $L = 1$.

Note that the description (A.2) does *not* correspond to a switched continuous time system the behavior of which is described by

$$\frac{dq_i(t)}{dt} = \begin{cases} (A_k q_i(t) + B_k u_k)_i & \text{if } b_{i,k}^{\text{ls}} < q_i(t) < b_{i,k}^{\text{us}} \\ 0 & \text{otherwise,} \end{cases} \tag{A.3}$$

for $t \in (t_k, t_{k+1})$ and for $i = 1, 2, \ldots, M$. Indeed, for $M = 1$ (A.3) results in

$$q_{k+1} = \begin{cases} \max\Big(\min\big((q_k + A_k^{-1}B_k u_k)e^{A_k(t_{k+1}-t_k)} - A_k^{-1}B_k u_k,\, b_k^{\text{us}}\big),\, b_k^{\text{ls}}\Big) & \text{if } A_k \neq 0 \\ \max\Big(\min\big(q_k + B_k u_k(t_{k+1}-t_k),\, b_k^{\text{us}}\big),\, b_k^{\text{ls}}\Big) & \text{if } A_k = 0\,. \end{cases}$$

For $M > 1$ the relation between q_{k+1}, q_k and $t_{k+1} - t_k$ is even more complex.

For the class of systems described by (A.2) the optimization problem that has to be solved in each major MPC step is given by:

$$\underset{u_0,u_1,\ldots,u_{N_{\text{c}}-1}}{\text{minimize}} \;\; J \tag{A.4}$$

subject to

$$
\begin{aligned}
u_k &= u_{k-K_c} && \text{for } k = N_c, N_c+1, \ldots, N_p-1, && \text{(A.5)}\\
u_{\min,k} \leqslant u_k &\leqslant u_{\max,k} && \text{for } k = 0, 1, \ldots, N_p-1, && \text{(A.6)}\\
q_{\min,k} \leqslant q_{k+1} &\leqslant q_{\max,k} && \text{for } k = 0, 1, \ldots, N_p-1 && \text{(A.7)}\\
z_{k+1} &= \min(A_k q_k + B_k u_k,\, b_k^{\text{us}}) && \text{for } k = 0, 1, \ldots, N_p-1, && \text{(A.8)}\\
q_{k+1} &= \max(z_{k+1},\, b_k^{\text{ls}}) && \text{for } k = 0, 1, \ldots, N_p-1. && \text{(A.9)}
\end{aligned}
$$

where $u_{\min,k}$ and $u_{\max,k}$ are respectively the minimum and the maximum values of u_k.

Using the same reasoning as in Section 3.3 we can show that the system (A.5) – (A.9) can be reformulated as an ELCP of the form

$$Ax_q + Bx_z + Cx_u + d \geqslant 0 \tag{A.10}$$

$$Ex_q + Fx_z + g \geqslant 0 \tag{A.11}$$

$$Hx_q + Kx_u + l \geqslant 0 \tag{A.12}$$

$$(Ax_q + Bx_z + Cx_u + d)^T (Ex_q + Fx_z + g) = 0 \ , \tag{A.13}$$

for appropriately defined matrices A, B, C, E, F, H, K and vectors d, g, l and with

$$x_u = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} .$$

If we introduce additional linear equality or inequality constraints on the components of x_u, we still obtain an ELCP. The additional linear inequality constraints lead to extra inequalities in (A.12), and the additional linear equality constraints lead to an extra equation of the form $Px_u + q = 0$, which also fits in the ELCP framework.

Now we can determine optimal input sequences using the ELCP approach or using multi-start local optimization.

C.2 Optimal and suboptimal input sequences for systems with saturation at a lower level only

In this section we consider systems with saturation at the lower level only. So $b_{i,k}^{\text{us}}$ is equal to ∞ for all i, k, or equivalently $(q_{\max,k})_i \leqslant b_{i,k}^{\text{us}}$ for all i, k. We also assume that $q_{\min,k} \leqslant b_k^{\text{ls}}$ for all k, i.e. we do not impose extra lower bound conditions on the queue lengths. Furthermore, we assume that $(A_k)_{ij} \geqslant 0$ for all i, j, k. Note that the latter assumption always holds for the class of first order linear hybrid systems that has been introduced in Section 3.1 since for this class we have $A_k = I$ for all k. The problem (A.4) – (A.9) then reduces to

$$\underset{x_u}{\text{minimize}} \ J$$

subject to

$$u_k = u_{k-K_c} \quad \text{for } k = N_c, N_c+1, \ldots, N_p-1,$$

$$
\begin{aligned}
u_{\min,k} \leqslant u_k \leqslant u_{\max,k} &\quad \text{for } k = 0,1,\ldots,N_\mathrm{p}-1,\\
q_{k+1} \leqslant q_{\max,k} &\quad \text{for } k = 0,1,\ldots,N_\mathrm{p}-1\\
q_{k+1} = \max(A_k q_k + B_k u_k,\, b_k^\mathrm{ls}) &\quad \text{for } k = 0,1,\ldots,N_\mathrm{p}-1.
\end{aligned}
$$

We call this problem $\mathcal{P}_\mathrm{g}$. We define the "relaxed" problem $\tilde{\mathcal{P}}_\mathrm{g}$ corresponding to the problem $\mathcal{P}_\mathrm{g}$ as:

$$\underset{x_q,x_u}{\text{minimize}}\ J$$

subject to

$$
\begin{aligned}
u_k = u_{k-K_\mathrm{c}} &\quad \text{for } k = N_\mathrm{c}, N_\mathrm{c}+1,\ldots,N_\mathrm{p}-1,\\
u_{\min,k} \leqslant u_k \leqslant u_{\max,k} &\quad \text{for } k = 0,1,\ldots,N_\mathrm{p}-1,\\
q_{k+1} \leqslant q_{\max,k} &\quad \text{for } k = 0,1,\ldots,N_\mathrm{p}-1\\
q_{k+1} \geqslant A_k q_k + B_k u_k &\quad \text{for } k = 0,1,\ldots,N_\mathrm{p}-1,\\
q_{k+1} \geqslant b_k^\mathrm{ls} &\quad \text{for } k = 0,1,\ldots,N_\mathrm{p}-1.
\end{aligned}
$$

Note that x_q and x_u are not directly coupled any more. The set of feasible solutions of $\tilde{\mathcal{P}}_\mathrm{g}$ is a convex set, whereas the set of feasible solutions of $\mathcal{P}_\mathrm{g}$ is in general not convex. Therefore, the relaxed problem $\tilde{\mathcal{P}}_\mathrm{g}$ will in general be easier to solve than the problem $\mathcal{P}_\mathrm{g}$.

The following proposition shows that for monotonically nondecreasing objective functions any optimal solution of the relaxed problem $\tilde{\mathcal{P}}_\mathrm{g}$ can be transformed into an optimal solution of the problem $\mathcal{P}_\mathrm{g}$.

Proposition C.1 *Let the objective function J be a monotonically nondecreasing function of x_q and let (x_q^*, x_u^*) be an optimal solution of $\tilde{\mathcal{P}}_\mathrm{g}$. If we define $x_q^\sharp$ such that*

$$
\begin{aligned}
q_1^\sharp &= \max\left(A_0 q_0 + B_0 u_0^*, b_0^\mathrm{ls}\right)\\
q_{k+1}^\sharp &= \max\left(A_k q_k^\sharp + B_k u_k^*, b_k^\mathrm{ls}\right) \qquad \textit{for } k = 1,2,\ldots,N-1.
\end{aligned}
$$

then $(x_q^\sharp, x_u^*)$ is an optimal solution of the problem $\mathcal{P}_\mathrm{g}$.

Proof: This proof is analogous to the proof of Proposition 4.1. The only difference is that now we have to include the fact that $(A_k)_{ij} \geqslant 0$ for all i,j,k in order to prove by induction that $q_k^\sharp \leqslant q_k^*$ for $k = 1,2,\ldots,N_\mathrm{p}$. □

Since the objective functions J_1, J_2, J_3, J_4 and J_5 do not explicitly depend on x_q, we have $J_l(\tilde{x}_q, x_u) = J_l(\hat{x}_q, x_u)$ for any $\tilde{x}_q, \hat{x}_q$ and for $l \in \{1,2,3,4,5\}$. This implies that J_1, J_2, J_3, J_4 and J_5 are monotonically nondecreasing functions of x_q. So we can use Proposition C.1 to transform the optimal control problem for the objective functions J_1 up to J_5 into an optimization problem with a convex feasible set.

The optimal solution of problem $\tilde{\mathcal{P}}_\mathrm{g}$ will in general not be a feasible solution of $\mathcal{P}_\mathrm{g}$, unless J is a monotonically increasing function of x_q:

Proposition C.2 *If J is a monotonically increasing function of x_q then any optimal solution of the relaxed problem $\tilde{\mathcal{P}}_\mathrm{g}$ is also an optimal solution of the problem $\mathcal{P}_\mathrm{g}$.*

Proof: This proof is similar to the proof of Proposition 4.2. □

Additional references

[A-1] B. De Schutter, "Optimal control of a class of linear hybrid systems with saturation," *SIAM Journal on Control and Optimization*, vol. 39, no. 3, pp. 835–851, 2000.