Delft University of Technology

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

Technical report CSE02-011

Model predictive control approach for recovery from delays in railway systems*

B. De Schutter, T. van den Boom, and A. Hegyi

If you want to cite this report, please use the following reference instead:
B. De Schutter, T. van den Boom, and A. Hegyi, "Model predictive control approach for recovery from delays in railway systems," *Transportation Research Record*, no. 1793, pp. 15–20, 2002. doi:10.3141/1793-03

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

Model Predictive Control Approach for Recovery from Delays in Railway Systems

B. De Schutter*, T. van den Boom, A. Hegyi

Control Systems Engineering
Fac. ITS, Delft University of Technology
P.O. Box 5031, 2600 GA Delft, The Netherlands
phone: +31-15-278.51.13, fax: +31-15-278.66.79
{b.deschutter,t.j.j.vandenboom,a.hegyi}@its.tudelft.nl

Abstract

We extend the model predictive control (MPC) framework, which is a very popular controller design method in the process industry, to transfer coordination in railway systems. In fact, the proposed approach can also be used for other systems with both hard and soft synchronization constraints, such as logistic operations. The main aim of the control is to recover from delays in an optimal way by breaking connections (at a cost). In general, the MPC control design problem for railway systems leads to a nonlinear non-convex optimization problem. We show that the optimal MPC strategy can also be computed using an extended linear complementarity problem. Furthermore, we present an extension with an extra degree of freedom to recover from delays by letting some trains run faster than usual (again at a cost). The resulting extended MPC railway problem can also be solved using an extended linear complementarity problem.

1 Introduction

We present a model predictive control (MPC) framework for a class of systems with a main emphasis on railway networks, although the approach can also be used for, e.g., logistic systems. MPC [1, 4, 8]. is a very popular on-line, adaptive control design technique in the process industry. The major advantages of MPC are that it allows the inclusion of constraints on the inputs and outputs, and that it can handle changes in the system parameters by using a moving horizon approach, in which the model and the control strategy are continuously updated. Conventional MPC uses discrete-time models (i.e., models consisting of a system of difference equations and that are sampled at regularly spaced instants of time). In [6] we have extended MPC to a class of systems with "hard" synchronization constraints, where "hard" means that the constraints should always be met. In this paper we further extend the MPC framework to a class of systems with both *hard* and *soft* synchronization constraints, i.e., in some cases we allow an activity to start although not all pre-scheduled predecessor activities have been completed, but at a cost. This could occur in a railway operations context, where a train should give pre-defined connections to other trains. However, if some of these trains have a too large delay, then it is sometimes better — from a global performance

*Corresponding author

viewpoint — to let the train depart anyway in order to prevent an accumulation of delays in the network. Of course, missed connections lead to a penalty due to dissatisfied passengers or due to compensations that have to be paid (In the Netherlands, NS, the main Dutch railway company has to pay back 50 % of the fare to passengers if they have an accumulated delay of more than 30 minutes, and 100 % of the fare if the delay is more than 1 hour). The main aim of the control approach presented here is to recover from past delays or known or expected future delays in an optimal way by breaking some connections if possible, or by letting the trains drive faster, where both types of control actions have a cost associated with them. Note that we do not consider re-routing or adapting the time schedule. Work in connection with the modeling and control of railway networks in a systems and control context can be found in [2, 3, 7, 12]. We also refer the interested reader to [9, 10, 11, 13] and the references therein.

This paper is organized as follows. After given a brief introduction to MPC, we derive a model for a railway system with both hard and soft synchronization constraints. Next, we define a control design problem for such a system where we can break a connection if delays occur and if this leads to a better global performance. We use an MPC approach (which has the following ingredients: a prediction horizon, a receding horizon procedure, and a regular update of the model and re-computation of the optimal control). In general, this leads to a hard non-convex nonlinear optimization problem. However, we show that the trajectories of the system can be described by an extended linear complementarity problem (ELCP) [5], for which we can compute a parameterized solution. Afterwards, we can then compute the optimal control over this solution set. The advantage is that we now have to solve a sequence of optimization problems with a convex feasible set (although the objective function is still nonlinear and non-convex). Computational experiments show that (for small sized problems or for a small control horizon) the ELCP approach is much faster and yields a better minimum than the straightforward nonlinear optimization approach. We also present an extension in which we allow the trains to drive faster if necessary. We conclude with a worked example.

2 Model predictive control

In this section we give a short introduction to conventional MPC [1, 4, 8] for discrete-time systems. In MPC we compute at each sample step k an optimal control input that minimizes a given performance measure over a given prediction horizon. Typically this performance measure represents a trade-off between reference signal tracking and minimizing the control effort. The optimization uses a prediction model that predicts the expected future behavior of the system for the current state of the system and for the expected or predicted external input signals. In order to limit the number of variables in the optimization procedure (which in general is equal to the number of control inputs multiplied by the prediction horizon) and to improve the stability of the system, a control horizon N_{c} ($\leqslant N_{\mathrm{p}}$) is introduced in MPC: after the control horizon has been passed the control signal is taken to be constant. In order to take changes in the system parameters into account, MPC uses an adaptive receding horizon approach in which at each sample step k only the first step of the optimal control signal is applied to the system, afterwards the model of the system or the predictions of the external input signals are updated as new measurements from the sensors or new information become available, next the time axis is shifted one sample, a new optimal control input is computed, and the whole procedure is repeated again.

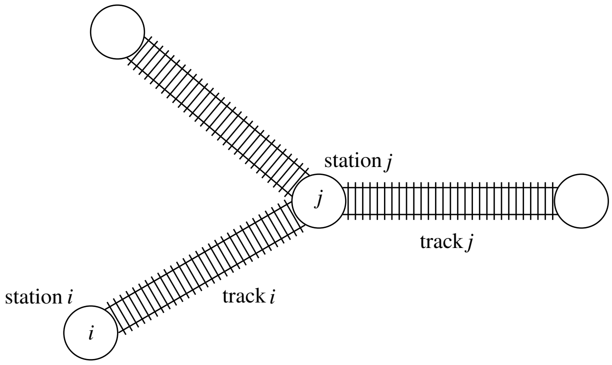

Figure 1: A part of a railway network.

Important parameters in this scheme are the prediction horizon N_p and the control horizon N_c. Too long control and prediction horizons can result in intractable optimization problems (recall that the optimization has to be performed on-line, so the size of the problem, i.e., the number of variables and the length of the prediction horizon, should not be too large). On the other hand, the prediction horizon N_p has to be long enough to represent the important process dynamics, and the control horizon N_c has to be long enough to be able to achieve a reasonable performance. In conventional model-based predictive control heuristic tuning rules have been developed to select appropriate values for N_p and N_c. When discussing the railway MPC problem we will also propose some rules of thumb to select values for N_p and N_c that are specifically oriented towards railway systems.

3 A model for railway systems

Consider a railway operations system. The nominal operation of the system follows a time schedule with a period T. We assume that all the trains follow a pre-scheduled route. Let n be the number of tracks in the network. Each track of the railway network has a number and a virtual train allocated to it. For the sake of simplicity we will say "(virtual) train j" to denote the (physical) train on track j, and "station j" to denote the station at the beginning of track j (cf. Figure 1). Let $x_j(k)$ be the time instant at which train j departs from station j for the kth time, and let $d_j(k)$ be the departure time for this train according to the time schedule.

The set of trains to which the kth train on track j gives a connection is denoted by $\mathcal{C}_j(k)$. This set can be divided in a set of hard connections $\mathcal{C}_{\mathrm{hard},j}(k)$ (e.g., if the train on track i and the train on track j are physically the same train, or if it is a very important connection that should be guaranteed at all cost), and a set of soft connections $\mathcal{C}_{\mathrm{soft},j}(k)$ (e.g., local trains to which the train j should give connection, but if the local train $i \in \mathcal{C}_{\mathrm{soft},j}(k)$ has a too large delay, then the connection may be broken; however, in that case a (maximal) cost $c_{\mathrm{broken},i,j}(k)$ is associated with the broken connection (see also (8)). Let $a_{i,j}(k)$ be the traveling time from station i to station j for each train $i \in \mathcal{C}_j(k)$. We also define a minimum connection time $t_{\mathrm{min},i,j}(k)$ for passengers to get from train i to train j for each train $i \in \mathcal{C}_j(k)$ (if virtual trains i and j are physically the same train, then this time corresponds to the minimum stopping time of train j at station j to allow passenger to get off or on the train).

We have the following constraints for the kth actual departure time $x_j(k)$ of train j:

- time schedule constraint
  Train j should not depart before the departure time according to the time schedule has passed:

$$x_j(k) \geqslant d_j(k) \ . \tag{1}$$

- hard synchronization constraints
  If there is another train i to which train j should give a hard connection, then train j may only depart if train i has arrived and the passengers have gotten enough time to get out of the train or to change trains. So for each $i \in \mathcal{C}_{\text{hard},j}(k)$ we have

$$x_j(k) \geqslant x_i(k - \delta_{i,j}(k)) + a_{i,j}(k) + t_{\text{min},i,j}(k) \ , \tag{2}$$

  where $\delta_{i,j}(k)$ denotes the cycle delay between train i and train j in the kth cycle: the kth train j gives connection to the $(k - \delta_{i,j}(k))$th train i (see also the worked example below).

- soft synchronization constraints
  If the connection takes place, then we have a constraint that is similar to (2). If the connection is broken, the train departs before the other train arrives and the passengers have gotten the time to change over. So for each $i \in \mathcal{C}_{\text{soft},j}(k)$ we have

$$x_j(k) \geqslant x_i(k - \delta_{i,j}(k)) + a_{i,j}(k) + t_{\text{min},i,j}(k) \tag{3}$$

  if the connection takes place, and

$$x_j(k) < x_i(k - \delta_{i,j}(k)) + a_{i,j}(k) + t_{\text{min},i,j}(k) \tag{4}$$

  if the connection is broken. If we introduce a control variable $u_{i,j}(k) \geqslant 0$, then we can combine these two equations into one equation of the following form:

$$x_j(k) \geqslant x_i(k - \delta_{i,j}(k)) + a_{i,j}(k) + t_{\text{min},i,j}(k) - u_{i,j}(k) \ , \tag{5}$$

  where $u_{i,j}(k)$ can be used to guarantee or to break a connection.

Since we let a train depart as soon as all connection conditions are satisfied, we have

$$\begin{aligned} x_j(k) = \max \Big(& d_j(k), \\ & \max_{i \in \mathcal{C}_{\text{hard},j}(k)} \big(x_i(k - \delta_{i,j}(k)) + a_{i,j}(k) + t_{\text{min},i,j}(k) \big), \\ & \max_{i \in \mathcal{C}_{\text{soft},j}(k)} \big(x_i(k - \delta_{i,j}(k)) + a_{i,j}(k) + t_{\text{min},i,j}(k) - u_{i,j}(k) \big) \Big) \ . \end{aligned} \tag{6}$$

Note that in a nominal, well-defined time schedule the term $d_j(k)$ in (6) will be the largest. However, if due to unscheduled circumstances (an incident, a late departure, works, etc.) train i has a delay, then the term corresponding to train i may become larger than the other terms. We define $t_{\text{slack},i,j}(k)$ as the slack time of the arrival of train $i \in \mathcal{C}_{\text{soft},j}(k)$ at station j (the transit time $t_{\text{min},i,j}(k)$ included) with respect to the actual kth departure time of train j:

$$t_{\text{slack},i,j}(k) = x_i(k - \delta_{i,j}(k)) + a_{i,j}(k) + t_{\text{min},i,j}(k) - x_j(k) \ . \tag{7}$$

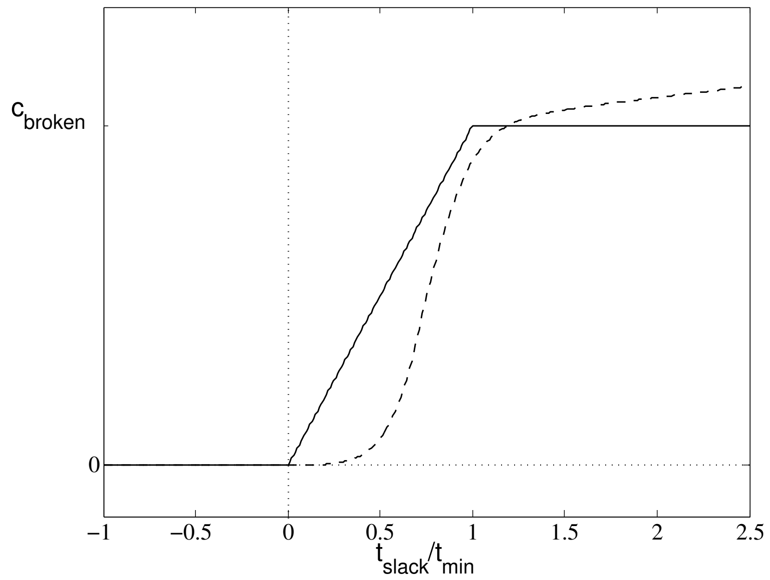

Figure 2: The piecewise-linear cost function J_{broken} (full line) and a smoother approximation (dashed line).

Note that this slack time is a function of the control variable $u_{i,j}(k)$ via $x_j(k)$. If $t_{\text{slack},i,j}(k) \leqslant 0$ then the connection is completely guaranteed (with enough time for the passengers to change trains). If $t_{\text{slack},i,j}(k) > t_{\text{min},i,j}(k)$, then train j leaves the station before the arrival of train i. If $0 < t_{\text{slack},i,j}(k) \leqslant t_{\text{min},i,j}(k)$ then the connection is guaranteed partly (i.e., fast-running passengers can get the connection, but slower ones may lose it). Therefore, we use the following piecewise-linear function to define the cost of a broken connection (see also Figure 2):

$$J_{\text{broken}}(t_{\text{slack}}, t_{\text{min}}, c_{\text{broken}}) = \begin{cases} 0 & \text{if } t_{\text{slack}} \leqslant 0, \\ \dfrac{c_{\text{broken}}}{t_{\text{min}}} \cdot t_{\text{slack}} & \text{if } 0 < t_{\text{slack}} \leqslant t_{\text{min}}, \\ c_{\text{broken}} & \text{if } t_{\text{slack}} > t_{\text{min}}\,. \end{cases} \tag{8}$$

4 The railway MPC problem

4.1 Problem definition

We define the following cost function over a given prediction horizon N_{p} for the kth operation cycle of the railway system:

$$\begin{aligned} J_{\text{cost}}(k) = & \sum_{l=0}^{N_{\text{p}}-1} \sum_{j=1}^{n} |x_j(k+l) - d_j(k+l)| + \\ & \lambda \sum_{l=0}^{N_{\text{p}}-1} \sum_{j=1}^{n} \sum_{i \in \mathcal{C}_{\text{soft},j}(k+l)} J_{\text{broken}}(t_{\text{slack},i,j}(k+l), t_{\text{min},i,j}(k+l), c_{\text{broken},i,j}(k+l)) \end{aligned} \tag{9}$$

where λ ¿ 0 is a weighting factor. This cost function has two components: the first tries to keep the trains running on schedule, whereas the second penalizes broken connections. The factor λ determines the trade-off or relative weight of the two components of the MPC cost function.

Now we consider the following controller design problem — which will be called the *railway MPC problem* at cycle k:

$$\min_{u_{i,j}(k),\ldots,u_{i,j}(k+N_\mathrm{p}-1)} \quad J_\mathrm{cost}(k)$$

$$\text{subject to (6) and to } u_{i,j}(k+l) \geqslant 0 \text{ for all } i,j \text{ and } l = 0,\ldots,N_\mathrm{p}-1\,. \tag{10}$$

In addition, to reduce the number of control variables we can — just as in conventional MPC — introduce a control horizon N_c ($\leqslant N_\mathrm{p}$) and set

$$u_{i,j}(k+l) = u_{i,j}(k+N_\mathrm{c}-1) \qquad \text{for } l = N_\mathrm{c},\ldots,N_\mathrm{p}-1\,. \tag{11}$$

This condition can be interpreted as follows: if after N_c cycles the delays have died out (i.e., it is not necessary to break connections any longer or equivalently, $u_{i,j}(k+N_\mathrm{c}) = 0$ for all i,j), then we do not break any connections in the subsequent cycles either. On the other hand, if the delays are still such that a connection should be broken in cycle $k+N_\mathrm{c}$, then we will also break these connections in the subsequent cycles. Alternatively, we can take the decrease or growth of the delays into account by using a constant growth/decrease rate condition of the form $\Delta u_{i,j}(k+l) = \Delta u_{i,j}(k+N_\mathrm{c}-1)$ for $l = N_\mathrm{c},\ldots,N_\mathrm{p}-1$ where we set negative values of $u_{i,j}(k+l)$ to 0 and where $\Delta s(k) = s(k) - s(k-1)$. Just like in conventional MPC we use a moving horizon approach, i.e., the railway MPC problem is solved for each cycle, then the computed controls are applied for the current cycle only, and meanwhile the model is updated, and the computation is performed again for the next cycle. This implies that we can also include predictable future delays (due to incidents, broken power lines, works, …) into our prediction model.

4.2 Tuning

The parameters N_p and N_c determine the size of the problem. If they are chosen too large, then the railway MPC problem is not tractable any more (recall that we have to solve the problem during each operations cycle, so the available time is limited). On the other hand, if we select N_p and N_c too small, then this will have a negative impact on the performance of the overall system. We propose the following rule of thumb for the selection of N_p: N_p should be chosen such that it covers the (expected) period over which the delays will die out. The choice of N_c mainly depends on the computational complexity of the problem. For small-sized networks we can take N_c rather large, whereas for large networks a small N_c will be necessary to be able to compute the MPC solution sufficiently fast (i.e., before the start of the next cycle of the railway network).

4.3 Algorithms for the railway MPC problem

In general, each step of the railway MPC problem leads to a non-convex nonlinear optimization problem. This problem can be solved using, e.g., a multi-start local optimization method such as multi-start sequential quadratic programming. Also note that the feasible set of the railway MPC problem is non-convex since (6) is non-convex. Now we will present an alternative approach to compute the optimal MPC control input that is based on a mathematical programming problem called the extended linear complementarity problem.

The Extended Linear Complementarity Problem (ELCP) is defined as follows [5]:

$$
\begin{aligned}
&\prod_{i \in \varphi_j} (Az - c)_i = 0 \qquad \text{for } j = 1, \ldots, m \\
&\text{subject to } Az \geqslant c \text{ and } Bz = d.
\end{aligned}
\tag{12}
$$

This problem can be interpreted as follows: Find solutions of a system of linear equations and inequalities ($Az \geqslant c$, $Bz = d$) where there are several groups of inequalities (one for each index set φ_j) such that in each group at least one inequality should hold with equality, i.e., for each j there should exist an index $i \in \varphi_j$ such that $(Az - c)_i = 0$. The formulation of the ELCP arose from our research on nonlinear resistive networks, discrete-event systems, hybrid systems, and traffic signal control.

Let us now show that the evolution equations and the constraints of the railway MPC problem can be recast as an ELCP. Clearly, the non-negativity constraint on $u_{i,j}(k)$ and the control horizon constraint (11) fit the ELCP framework. Now we show that (6) can also be written as an ELCP. This will be done by showing that an expression of the form $x_j(k) = \max(d_j(k), z_{1,j}(k), z_{2,j}(k))$ where $z_{1,j}(k)$ and $z_{2,j}(k)$ are auxiliary variables, is an ELCP. If we then add the conditions that $z_{1,j}(k)$ and $z_{2,j}(k)$ should be equal to the second term and the third term of the right-hand side of (6) and if we take into account that the merge of two ELCPs is also an ELCP, we have recursively shown that (6) can be written as an ELCP. The condition $x_j(k) = \max(d_j(k), z_{1,j}(k), z_{2,j}(k))$ can be rewritten as $x_j(k) - d_j(k) \geqslant 0$, $x_j(k) - z_{1,j}(k) \geqslant 0$, $x_j(k) - z_{2,j}(k) \geqslant 0$, with $x_j(k) = d_j(k)$ or $x_j(k) = z_{1,j}(k)$ or $x_j(k) = z_{2,j}(k)$. The latter condition can be rewritten as $(x_j(k) - d_j(k)) \cdot (x_j(k) - z_{1,j}(k)) \cdot (x_j(k) - z_{2,j}(k)) = 0$. Hence, we have obtained an ELCP. As a consequence, the trajectories of the railway system can be described by an ELCP.

In [5] we have developed an algorithm that yields a parametric description of the solution set of an ELCP. More specifically, the solution set of the ELCP defined by (12) is characterized by a set of vectors $V = \{z^i \,|\, i = 1, \ldots, r\}$ and a set of index sets $\Lambda = \{\psi_j \,|\, j = 1, \ldots, s\}$ such that for any j any combination of the form

$$
\sum_{i \in \psi_j} \rho_i z^i \qquad \text{with } \rho_i \geqslant 0 \text{ for all } i \text{ and } \sum_{i \in \varphi_j} \rho_i = 1 \tag{13}
$$

is a solution of the ELCP. The optimal MPC strategy can now be obtained by determining for each index set ψ_j the combination of the ρ_i's for which the objective function $J_{\text{cost}}(k)$ reaches a global minimum (note that for each index set ψ_j this amounts to an optimization over a convex set since (13) describes a convex combination of the vectors z^i) and afterwards selecting the overall minimum.

The advantage of this ELCP approach compared to straightforward nonlinear constrained optimization is that in the ELCP approach we have to solve a sequence of optimization problems with a convex feasible set instead of one big problem with a non-convex feasible set. Optimization problems with a convex feasible set (albeit with a non-convex objective function) are easier to solve numerically than problems with a non-convex feasible set. Note however that the algorithm of [5] to compute the solution set of a general ELCP requires exponential execution times, which means that the ELCP approach is not feasible if N_c is large. Our computational experiments have shown that in most cases the determination of the minimum value of the objective functions given above is a well-behaved problem in the sense that using a local minimization routine (that uses, e.g., sequential quadratic programming) starting from different initial points almost always yields the same numerical result (within a certain tolerance). So (for small sized problems or for a small control horizon) the ELCP approach is much faster and yields a better minimum than the straightforward nonlinear optimization approach.

To get a smoother optimization problem we can introduce another, smoother cost function for broken connections such as, e.g., the one represented by the dashed line in Figure 2. Such a cost function might also better correspond to what we could expect in reality than the piecewise-linear cost function defined by (8) and represented by the full line in Figure 2.

5 Extension

Up to now we have assumed that — in absence of any information about future delays — the traveling time from station i to station j in the kth cycle is given by the nominal traveling time $a_{i,j}(k)$. However, in practice we can also use this time to recover from delays by letting train i run faster if necessary. Of course, this will lead to extra costs (due to increased energy consumption or faster wear of the material). Let $a_{\min,i,j}(k)$ be the minimal time needed to get from station i to station j at full speed in cycle k, and let $a_{\text{nom},i,j}(k)$ be the nominal traveling time. We introduce an extra control variable $v_{i,j}(k)$ to modify the traveling time from station i to station j in the kth cycle. We get a model for this extended system by replacing all occurrences of $a_{i,j}(k)$ in (6) by $a_{\min,i,j}(k) + v_{i,j}(k)$ and by adding the extra condition

$$0 \leqslant v_{i,j}(k) \leqslant a_{\text{nom},i,j}(k) - a_{\min,i,j}(k) \ . \tag{14}$$

Note that if $v_{i,p}(k)$ and $v_{i,q}(k)$ correspond to the same traveling time for some indices p and q (i.e., stations p and q are in fact the same (physical) station), then we have to add the constraint $v_{i,p}(k) = v_{i,q}(k)$. To express the extra costs related to increasing the speeds of the trains, we add a term of the form

$$\mu \sum_{l=0}^{N_\text{p}-1} \sum_{j=1}^{n} \sum_{i \in \mathcal{C}_j(k+l)} c_{\text{speed},i,j} \cdot \big(a_{\text{nom},i,j}(k+l) - (a_{\min,i,j}(k+l) + v_{i,j}(k+l)) \big)^2 \tag{15}$$

to the cost function $J_{\text{cost}}(k)$, where μ is a nonnegative weight parameter that represents the relative importance of the speed term with respect to the other terms of the (extended) objective function, and where $c_{\text{speed},i,j}$ characterizes the extra cost per squared unit traveling time decrease for trains on the track going from station i to station j (this factor may also depend on k). This results in an extended railway MPC problem. It is easy to verify that this problem can also be solved using the ELCP.

6 Worked example

Consider the railroad network of Figure 3. There are 4 stations in this railroad network (S_1, S_2, S_3 and S_4) that are connected by 6 single tracks (T_1 up to T_6). There are two trains

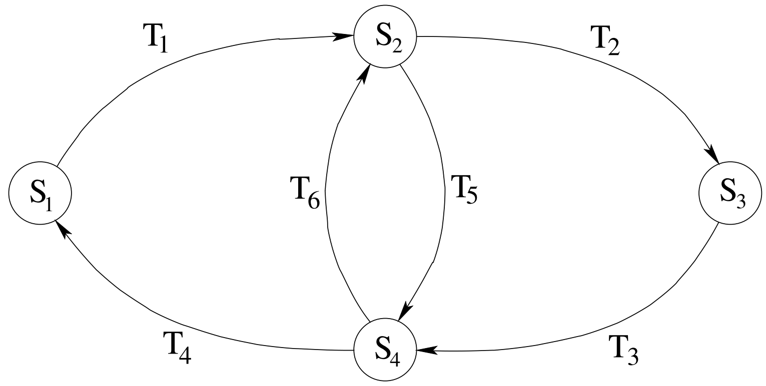

Figure 3: The railroad network considered in the worked example.

Track	From station	To station	Nominal traveling times (min)	Scheduled departure time modulo 60 (min)
T_1	S_1	S_2	16	00
T_2	S_2	S_3	8	19
T_3	S_3	S_4	10	31
T_4	S_4	S_1	12	45
T_5	S_2	S_4	20	22
T_6	S_4	S_2	25	50

Table 1: The nominal traveling times and the departure times for the railroad network considered in the worked example.

available. The first train follows the route S_1—S_2—S_3—S_4—S_1 and the second train follows the route S_2—S_4—S_2. We assume that there exists a periodic timetable that schedules the earliest departure times of the trains. The period of the timetable is $T = 60$ minutes. So if a departure of a train from station S_2 is scheduled at 5.13 a.m., then there is also scheduled a departure of a train from station S_2 at 6.13 a.m., 7.13 a.m., and so on. Table 1 summarizes the information in connection with the nominal traveling times and the departure times. All the times are measured in minutes. The indicated departure times are the earliest departure times in the initial station of the track expressed in minutes after the hour. The first period starts at time $t = 0$. At the beginning of the first period the first train is in station $S_1$and the second train is in station S_2. We will only consider the basic model here (so speed control will not be included).

Suppose that we have the following *soft* synchronization constraints in the network:

- the train on track T_2 has to wait for the train on track T_6,
- the train on track T_4 has to wait for the train on track T_5,
- the train on track T_5 has to wait for the train on track T_1,
- the train on track T_6 has to wait for the train on track T_3.

The hard connection constraints are that the (virtual) trains on tracks T_1, T_2, T_3 and T_4 are physically the same train, and the same holds for the (virtual) trains on tracks T_5 and

T_6. The passengers get 2 minutes to change trains (for soft connections) and 1 minute to get out of the train (for hard connections). Each train departs as soon as all the connections are guaranteed (except for a soft connection when it is broken), the passengers have gotten the opportunity to change over, and the earliest departure time indicated in the timetable has passed. We assume that in the first period all the trains depart according to schedule.

Now we write down the equations that describe the evolution of the $x_j(k)$'s. First we consider the train on track T_1 and we determine $x_1(k)$, the time instant at which this train departs from station $S_1$for the kth time. At the beginning of the first period the train is in station S_1. So if k is equal to 1, the train departs from station S_1 at time $t = 0$. If k is greater than 1, the train departs from station S_1 for the kth time as soon as it has arrived in station S_1 for the $(k-1)$th time, the passengers have gotten the time to get out of the train and the earliest departure time indicated in the timetable has passed. Note that under nominal operations the kth train on track T_1 (e.g., the one that departs from station S_1 at, e.g., 10.00 a.m.) gives connection to the $(k-1)$th train on track T_4 (which has departed from station S_4 at 9.45 a.m.) and *not* to the kth train on track T_4 (which will depart from station S_4 at 10.45 a.m.). This implies that the train arrives in station S_1 for the $(k-1)$th time at time instant $x_4(k-1) + a_{4,1}(k)$. Afterwards, the passengers have $t_{\min,4,1}(k) = 1$ minute to get out of the train. Since the system operates under a periodic timetable with period T, the kth departure time of the train on track T_1 according to the timetable is $0 + (k-1)T$. Hence, we have

$$x_1(k) = \max(x_4(k-1) + a_{4,1}(k) + 1,\ 0 + (k-1)T) \tag{16}$$

for $k = 1, 2, \ldots$, where we set $x_4(0) = -\infty$ to make the equation hold for $k = 1$ (note that setting $x_4(0) = -\infty$ makes that for $k = 1$ the first term of the max expression in (16) is always smaller than the second term), since in the first cycle of the day (i.e., for $k = 1$) the train is already present in station S_1 so that it departs according to the time schedule.

The train on track T_1 will arrive for the kth time in station S_2 at time instant $x_1(k) + a_{1,2}(k)$, after which the passengers have $t_{\min,1,2}(k) = 1$ minute to get out of the train. If k is greater than 1, then the train has to wait for the passengers of the train on track T_6, which arrives in station S_2 at time instant $x_6(k-1) + a_{6,2}(k)$. The passengers have $t_{\min,6,2}(k) = 2$ minutes to change trains. According to the timetable the train on track T_2 can only depart after time instant $19 + (k-1)T$. Furthermore, since the connection constraint is soft, we introduce a control variable $u_{6,2}(k)$ to break the connection if necessary. Hence, we have

$$x_2(k) = \max(x_1(k) + a_{1,2}(k) + 1,\ x_6(k-1) + a_{6,2}(k) + 2 - u_{6,2}(k),\ 19 + (k-1)T) \tag{17}$$

for $k = 1, 2, \ldots$ with $x_6(0) = -\infty$. Note that — referring to (6) — we have $\delta_{1,2}(k) = 0$ since the kth train on track T_2 is the same train as the kth train on track T_1, and $\delta_{6,2}(k) = 1$ since the kth train on track T_2 gives connection to the $(k-1)$th train on track T_6.

Using a similar reasoning as the one above, we find that the other departure times are given by

$$\begin{aligned}
x_3(k) &= \max(x_2(k) + a_{2,3}(k) + 1,\ 31 + (k-1)T) \\
x_4(k) &= \max(x_3(k) + a_{3,4}(k) + 1,\ x_5(k) + a_{5,4}(k) + 2 - u_{5,4}(k),\ 45 + (k-1)T) \\
x_5(k) &= \max(x_1(k) + a_{1,5}(k) + 2 - u_{1,5}(k),\ x_6(k-1) + a_{6,5}(k) + 1,\ 22 + (k-1)T) \\
x_6(k) &= \max(x_3(k) + a_{3,6}(k) + 2 - u_{3,6}(k),\ x_5(k) + a_{5,6}(k) + 1,\ 50 + (k-1)T)
\end{aligned} \tag{18}$$

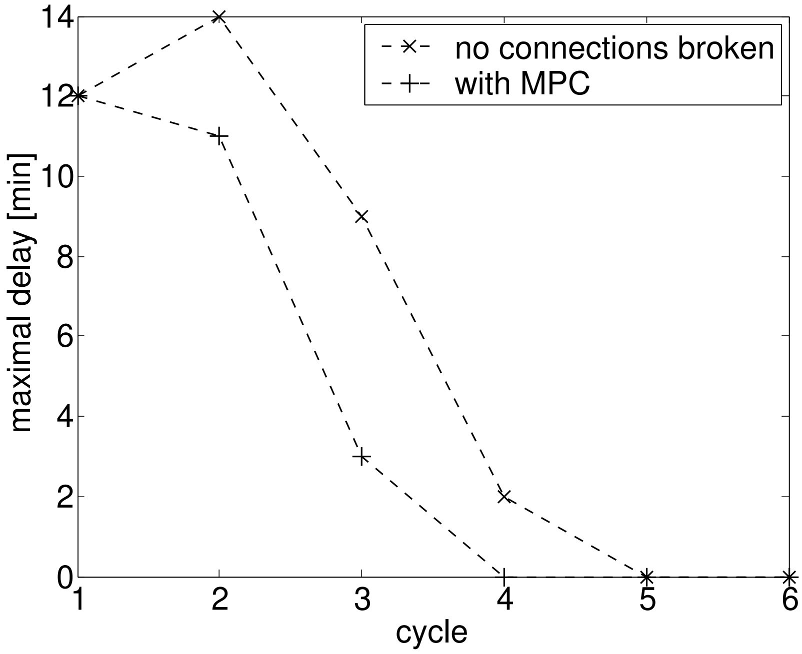

Figure 4: The delays with and without MPC control for the railroad network considered in the worked example.

for $k = 1, 2, \ldots$ with $x_j(0) = -\infty$ for $j = 1, 2, \ldots, 6$.

Let us now assume that all traveling times are nominal (cf. Table 1) except for $a_{1,2}(1) = a_{1,5}(1) = 30$ and $a_{1,2}(2) = a_{1,5}(2) = 25$. Let $N_\mathrm{c} = 4$, $N_\mathrm{p} = 6$, $\lambda = 0.75$, $c_{\mathrm{broken},6,2}(k) = c_{\mathrm{broken},5,4}(k) = 10$, and $c_{\mathrm{broken},1,5}(k) = c_{\mathrm{broken},3,6}(k) = 5$.

If we do not break any connections then we find a maximal delay w.r.t. the departure time schedule of 12 minutes in the first cycle (for the train on track T_2), 14 minutes in the second cycle (for the train on track T_2), 9 minutes in the third cycle (for the train on track T_1), and 2 minutes in the fourth cycle (for the train on track T_1); from the fifth cycle on, the trains will again ride on schedule (see Figure 4). If we do not break any connections, then the value of the MPC cost function is 134. If we consider the (basic) MPC railway problem for this network and if we compute the optimal MPC control input for $k = 1$, we find with both the nonlinear optimization approach and the ELCP approach the following solution: completely break the connection $\mathrm{T}_1 \to \mathrm{T}_5$ in the first and the second cycle. If we apply this control strategy, then we find a maximal delay w.r.t. the departure time schedule of 12 minutes in the first cycle (for the train on track T_2), 11 minutes in the second cycle (for the train on track T_2), and 3 in the third cycle (for the train on track T_1); in the fourth cycle all the trains again ride on schedule (see Figure 4). The corresponding value of the MPC cost function is 69.5.

7 Conclusions

We have presented an MPC-like control design method for a class of systems with both soft and hard synchronization constraints. A typical example of this class are railway systems.

The control action consists in breaking certain soft connections to prevent delays from accumulating, but this can only be done at a certain cost. We have also considered an extended problem in which we also allow trains to drive faster if necessary (again at a certain cost). We have shown that the resulting optimization problem for both the basic railway MPC problem and the extended railway MPC problem can be solved using ELCPs. Furthermore, due to the use of a moving horizon strategy and a control horizon this method can be used in on-line applications and it can deal with (predicted) changes in the system parameters. So if we can predict the delays that will occur due to an incident or to works, then we can include this information when determining the optimal control input for the next cycles of the operation of the network.

An important topic for future research is the development of efficient algorithms to solve the basic and the extended railway MPC problem. One option could be to develop a branch-and-bound algorithm to solve optimization problems defined over the solution set of an ELCP. So instead of first determining the solution set of the ELCP (which is a computationally intensive operation) and afterwards optimizing the objective function over the parameterized solution set, we could then perform the optimization and the (implicit) solution of the ELCP in one step, which should lead to a much more efficient approach. We will also compare the performance of this branch-and-bound algorithm with the straightforward nonlinear nonconvex optimization approach.

Acknowledgments

Research partially sponsored by the European Community TMR project ALAPEDES (Algebraic Approach to Performance Evaluation of Discrete Event Systems), and by the FWO Research Community ICCoS (Identification and Control of Complex Systems).

References

[1] F. Allgöwer, T.A. Badgwell, J.S. Qin, J.B. Rawlings, and S.J. Wright, "Nonlinear predictive control and moving horizon estimation – An introductory overview," in *Advances in Control: Highlights of ECC '99* (P.M. Frank, ed.), pp. 391–449, London, UK: Springer, 1999.

[2] J.G. Braker, "Max-algebra modelling and analysis of time-table dependent transportation networks," in *Proceedings of the 1st European Control Conference*, Grenoble, France, pp. 1831–1836, July 1991.

[3] J.G. Braker, "An extended algorithm for performance evaluation of timed event graphs," in *Proceedings of the 2nd European Control Conference*, Groningen, The Netherlands, pp. 524–529, June 1993.

[4] E.F. Camacho and C. Bordons, *Model Predictive Control in the Process Industry.* Berlin, Germany: Springer-Verlag, 1995.

[5] B. De Schutter and B. De Moor, "The extended linear complementarity problem," *Mathematical Programming*, vol. 71, no. 3, pp. 289–325, Dec. 1995.

[6] B. De Schutter and T. van den Boom, "Model predictive control for max-plus-linear discrete event systems," *Automatica*, vol. 37, no. 7, pp. 1049–1056, July 2001.

[7] P.R. de Waal, A. Overkamp, and J.H. van Schuppen, “Control of railway traffic on a single line,” in *Proceedings of the European Control Conference (ECC’97)*, Brussels, Belgium, paper 230, July 1997.

[8] C.E. García, D.M. Prett, and M. Morari, “Model predictive control: Theory and practice – A survey,” *Automatica*, vol. 25, no. 3, pp. 335–348, May 1989.

[9] T. Huisman and R.J. Boucherie, “Running times on railway sections with heterogeneous train traffic,” *Transportation Research Part B*, vol. 35, no. 3, pp. 271–292, Mar. 2001.

[10] O.K. Kwon, C.D. Martland, and J.M. Sussman, “Routing and scheduling temporal and heterogeneous freight car traffic on rail networks,” *Transportation Research Part E: Logistics and Transportation Review*, vol. 34, no. 2, pp. 101–115, 1998.

[11] D.R. Martinelli and H. Teng, “Optimization of railway operations using neural networks,” *Transportation Research Part C*, vol. 4, no. 1, pp. 33–49, Feb. 1996.

[12] R. Minciardi, M. Paolucci, and R. Pesenti, “Generating optimal schedules for an underground railway line,” in *Proceedings of the 34th IEEE Conference on Decision and Control*, New Orleans, Louisiana, pp. 4082–4085, Dec. 1995.

[13] İ. Şahin, “Railway traffic control and train scheduling based on inter-train conflict management,” *Transportation Research Part B*, vol. 33, no. 7, pp. 511–534, Sept. 1999.