Delft University of Technology

Delft Center for Systems and Control

Technical report 13-011

An MPC scheme for routing problem in baggage handling systems: A linear programming approach*

Y. Zeinaly, B. De Schutter, and H. Hellendoorn

If you want to cite this report, please use the following reference instead:
Y. Zeinaly, B. De Schutter, and H. Hellendoorn, "An MPC scheme for routing problem in baggage handling systems: A linear programming approach," *Proceedings of the 10th IEEE International Conference on Networking, Sensing and Control*, Paris, France, 6 pp., Apr. 2013. Paper ThB01.01.

Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
phone: +31-15-278.24.73 (secretary)
URL: https://www.dcsc.tudelft.nl

An MPC Scheme for Routing Problem in Baggage Handling Systems: A Linear Programming Approach

Yashar Zeinaly
Delft Center for Systems and Control
Delft University of Technology
Delft, The Netherlands

Bart De Schutter
Delft Center for Systems and Control
Delft University of Technology
Delft, The Netherlands

Hans Hellendoorn
Delft Center for Systems and Control
Delft University of Technology
Delft, The Netherlands

***Abstract*—This paper considers the routing problem of DCVs in state-of-the-art baggage handling systems. In previous work, this problem was considered within the framework of model predictive control, where the optimization problem was recast as a mixed integer linear programming problem. We make two significant improvements with respect to this previous work. We show that by choosing an alternative model, the optimization problem can be recast as a linear programming problem, which has significant computational advantages. Furthermore, we use a new cost function that reflects the control objectives in a better way. This leads to a computationally efficient routing controller.**

I. Introduction

Many big airports nowadays implement modern baggage handling systems. Such systems help to automate the baggage handling process in an efficient way achieving high baggage throughput otherwise unreachable by traditional baggage handling systems. This enables the airport to accommodate higher baggage demands with less delays, hence reducing the baggage handling costs. State-of-the-art baggage handling systems are composed of the following elements: loading stations, where the luggage originating from a check-in desk or a transit flight enters the system; unloading stations, which are the final destinations of the luggage from where the luggage are transported to the aircraft; a network of tracks that connects the loading stations to the unloading stations; and destination coded vehicles (DCVs), which are high-speed vehicles moving on the network of tracks transporting pieces of luggage from their origins (loading stations) to their destinations (unloading stations). The tracks of the network are composed of links connected via junctions. At each junction a switch will determine the next link to travel along. Given the origin and destination for each DCV, the switch controller determines a route for each DCV by manipulating the switches.

From a control perspective there are usually two sets of control challenges associated with modern baggage handling systems; low-level control issues and high-level control issues. Low-level control problems are mostly related to the safety and operational requirements of the baggage handling system such as controlling the speeds of the DCVs, keeping a minimum safe distance between two consecutive DCVs traveling on the same link, collision avoidance, synchronization of loading each piece of baggage into a DCV, etc. High-level control problems relate to the performance requirements of the baggage handling system such as on-time delivery of the baggage at the unloading stations and minimizing the energy consumption and wear and tear due to switching and dispatching of DCVs.

The high-level control issues can be further divided into three categories: i) routing of DCVs, ii) line balancing, and iii) empty-cart management. The routing problem is the problem of assigning a route for each DCV from its origin to its destination such that a certain performance criterion is satisfied [1], [2]. Once the DCVs deliver the baggage at the end point, they have be re-assigned to the loading stations. Line balancing is the problem of dynamically assigning empty DCVs to the loading stations [3]. Very closely related to this problem, is empty-cart management, which is the problem of route assignment for empty DCVs from their end points to the loading stations such that they cause as little interruption as possible with the flow of loaded DCVs.

In [2], optimal control and model predictive control (MPC) based schemes are developed for dynamic route assignment of DCVs by controlling the switches at the junctions in an optimal way. Of particular interest are the MPC approaches due to their capability to directly include constraints. One particular approach in [2] is MPC-based routing based on flow models, where it is shown the associated optimization problem can be recast as a mixed integer linear program (MILP). However, for a large-scale baggage handling system, MPC based on MILP (MILP-MPC) with a large prediction horizon becomes computationally prohibitive [1], [4]. To avoid this issue, in [2] a short prediction horizon was used, which comes at the cost of a loss in performance.

In this paper, the dynamic route assignment problem for DCVs considered in [2] is revisited. A slightly different flow-based model of the baggage handling system is presented, which will be used within the framework of MPC to compute optimal

Research supported by the European Union Seventh Framework Programme [FP7/2007-2013] under grant agreement no. 257462 HYCON2 Network of Excellence, and the BSIK project “Next Generation Infrastructures (NGI)”.

routes for DCVs. In particular, we will show that using this model, the associated optimization problem can be recast as a *linear* program. This allows us to achieve a better closed-loop performance, in a feasible amount of time, by choosing longer horizons. As a second improvement with respect to the work of [2], the cost function will be modified in such a way to better reflect the control requirements, hence improving the overall performance of the closed-loop system.

II. Common Notation and Assumptions

First, we will present some notation and assumptions that are common for the model of [2] and the alternative model, which will be presented in Section V. We make the following assumptions:

A0 Each link in the network of tracks belongs to at least one directed path from an origin node (i.e., a loading station) to a destination node (i.e., an unloading station).

A1 A DCV is present at the loading station whenever a piece of baggage arrives.

A2 There is a unique outgoing link for each loading station and a unique incoming link for each unloading station.

A3 The movement of the DCVs is approximated by continuous flow variables.

A4 The DCV queue lengths are very short compared to the length of the link and the DCV travel time on each link is an integer multiple of the controller sampling time T_{s}.

By Assumption A0, there are no redundant links in the network. By Assumption A1, the pieces of baggage are immediately dispatched from the loading stations as they arrive. This also implies that we do not need to distinguish between baggage flows and DCV flows within the system. Otherwise, the movement of empty DCVs from the unloading stations to the loading stations would have to be modeled as well. One then needs to distinguish between baggage and DCV flows. Assumption A2 is necessary to properly define the inflow of the link connected to a loading station and the outflow of the link connected to an unloading station. Assumption A3 is necessary for tractability of the control problem. Even though the number of DCVs is an integer in reality, for a fairly large number of DCVs, the movement of DCVs can be approximated by continuous flows without introducing a large error. The actual time required to travel from the beginning of the link to the end of the link depends on the length of the DCV queue, which is formed at the end of the link. However,by assumption A4, when the queue lengths are small compared to the length of the links, the variation in the travel time is negligible.

The baggage handling system can be represented by a directed graph $G = (V, L)$ with $V = V_{\mathrm{org}} \cup V_{\mathrm{inter}} \cup V_{\mathrm{des}}$, where the V_{org} is the set of origin nodes, V_{inter} is the set of intermediate nodes, and V_{des} is the set of destination nodes. The set of links of the network is $L = L_{\mathrm{org}} \cup L_{\mathrm{inter}} \cup L_{\mathrm{des}}$, where L_{org} is set of origin links (i.e., the links that are directly connected to the loading stations), L_{inter} is the set of intermediate links, and V_{des} is the set of destination links (i.e., the links that are directly connected to the unloading stations).

III. Original Model

In this section we briefly present the flow-based model as proposed in [2]. The model uses the nodes of the graph to describe the system. First we consider the following notation:

- L_d denotes the set of links that are on some directed path to destination node d, $d \in V_{\mathrm{des}}$.
- L_v^{in} is the set of incoming links of node v, $v \in V$.
- L_v^{out} is the set of outgoing links of node v, $v \in V$.
- For each $d \in V_{\mathrm{des}}$ and for each $o \in V_{\mathrm{org}}$, $Q_{o,d}(k)$ is the baggage inflow (demand) at origin node o with destination d during the time interval $[kT_{\mathrm{s}}, (k+1)T_{\mathrm{s}})$, with T_{s} being the sampling time.

For each destination node $d \in V_{\mathrm{des}}$ and for each link $l \in L_d$, $q_{l,d}(k)$ is defined as the flow of DCVs moving towards destination node d that enter link l during the time interval $[kT_{\mathrm{s}}, (k+1)T_{\mathrm{s}})$. This model assumes that at each node v, with exception of destination nodes, the DCVs stack up in vertical queues according to their destination. The queue lengths at destination nodes are considered to be zero. This is because we assume either destination nodes have unlimited capacity or there is no restriction on the outflow of destination nodes so the baggage are immediately taken to the planes upon arrival. Let $x_{v,d}(k)$ denote the length of the vertical queue at node v for DCVs going to destination node $d \in V_{\mathrm{des}}$ during the time interval $[kT_{\mathrm{s}}, (k+1)T_{\mathrm{s}})$. Then, the evolution of queue lengths in discrete time is given by

$$x_{v,d}(k+1) = \max\big(0, x_{v,d}(k) + T_{\mathrm{s}}(F_{v,d}^{\mathrm{in}}(k) - F_{v,d}^{\mathrm{out}}(k))\big) \quad (1)$$

where $F_{v,d}^{\mathrm{in}}(k)$ is the total inflow of node v to destination d being defined as

$$F_{v,d}^{\mathrm{in}}(k) = \begin{cases} Q_{v,d}(k) & \text{if } v \in V_{\mathrm{org}} \\ \sum\limits_{l \in L_v^{\mathrm{in}} \cap L_d} q_{l,d}(k - k_l) & \text{if } v \notin V_{\mathrm{org}} \end{cases} \quad (2a)$$

with $k_l T_{\mathrm{s}}$ being the travel time on link l[1]. $F_{v,d}^{\mathrm{out}}(k)$ is the total outflow of node v to destination d being defined as

$$F_{v,d}^{\mathrm{out}}(k) = \sum_{l \in L_v^{\mathrm{out}} \cap L_d} q_{l,d}(k) \quad (2b)$$

In [2], it is assumed the total outflow from node v satisfies the following condition:

$$F_{v,d}^{\mathrm{out}}(k) \leq F_{v,d}^{\mathrm{in}}(k) + \frac{x_{v,d}(k)}{T_{\mathrm{s}}} \quad (3)$$

Additionally, the following constraints on the DCV queue lengths and on the total DCV flows of the links are imposed:

$$\sum_{d \in V_{\mathrm{des}}} x_{v,d}(k) \leq x_z^{\max} \quad (4a)$$

$$\sum_{d \in V_{\mathrm{des}}} q_{l,d}(k) \leq q_l^{\max} \quad (4b)$$

where $x_v^{\max}$ is the maximum queue length at node v and $q_l^{\max}$ is the maximum flow on link l.

[1] Assuming a constant speed for DCVs v_{DCV}, k_l is given by $k_l = \frac{s_l}{T_{\mathrm{s}} v_{\mathrm{DCV}}}$, where s_l is the length of link l.

The evolution of the queue length as given by (1) is a piecewise affine function in the control variable $q_{l,d}(k)$. By introducing some binary variables as described in [5], the evolution of the queue lengths as well as the above mentioned constraints is recast as system of equalities and inequalities of the following form [2]:

$$\boldsymbol{x}_{k+1} = \mu^{\text{eq}}(\boldsymbol{x}_k, \boldsymbol{q}_k)$$
$$\mu^{\text{ineq}}(\boldsymbol{x}_k, \boldsymbol{q}_k) \leq 0$$

where $\boldsymbol{x}_k$ is the state vector that includes all queue lengths $x_{v,d}(k)$ for $(v,d) \in V \times V_{\text{des}}$ and delayed samples of $q_{l,d}(k)$ and $Q_{o,d}(k)$, and $\boldsymbol{q}_k$ is the input vector that includes all flows $q_{l,d}(k)$ for each $l \in L_d$ and each $d \in V_{\text{des}}$.

IV. Real and Virtual Flows

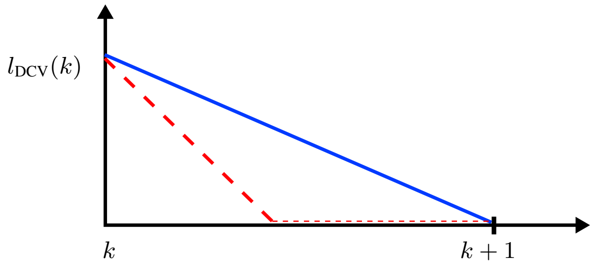


Fig. 1. Evolution of a queue length associated with real flow (solid line) and virtual flow (dashed-line) models between time-steps k and $k+1$. The queue length computed using the virtual flow model is only valid at time instants kT_s and $(k+1)T_s$ whereas the one of real flow is valid for the whole interval.

The model presented in [2] is based on "real flows", which, for a sufficiently small sampling time, can model the evolution of the queue lengths during the time interval $[kT_s, (k+1)T_s)$. However, if we are only interested in the queue lengths at time instants kT_s, "virtual flows" can be used, where the flows are artificially restricted such that the queue lengths are always non-negative during the time interval $[kT_s, (k+1)T_s)$. Note that the two models yield the same queue lengths at time instants $t = kT_s$ for integer k. The difference between these two models is highlighted in Fig. 1

Even though [2] uses (1) to describe the evolution of queue lengths using the "real flows", it is easily observed that constraint (3) restricts the flows such that the queue lengths will never be zero during the time interval $[kT_s, (k+1)T_s)$. This is essentially equivalent to modeling the system with virtual flows, which makes the max operator in (1) redundant. However, this fact is not noticed in [2] as the model is expressed by mixed linear equalities and inequalities as mentioned in Section III. Since [2] uses a linear objective function, the optimal control problem in [2] is an MILP problem.

V. New Alternative Model

This model is based on the links of the graph. The queue lengths are associated with the links and the control variables are defined as the flow of DCVs from a link to its neighbor links. In a similar manner to the model in [2], the flows are indexed by their destination, enabling us to distinguish between baggage with different destinations. This is important as the baggage must end up in the right destination. Accordingly, at the end of each link $l \in L$ there is a partial queue of DCVs associated with destination $d \in L_{\text{des}}$. Consider the following notation:

- L_d denotes the set of links that are on some directed path to destination link d, $d \in L_{\text{des}}$.
- L_l^{in} is the set of incoming links of link l, $l \in L$.
- L_l^{out} is the set of outgoing links of link l, $l \in L$.
- For each $d \in L_{\text{des}}$ and for each $o \in L_{\text{org}}$, $Q_{o,d}(k)$ is the baggage inflow (demand) at origin link o with destination d during the time interval $[kT_s, (k+1)T_s)$.

For each link $l \in L_d$ and each link $p \in L_l^{\text{out}}$ and each $d \in L_{\text{des}}$, we define the control variable $q_{l,p}^d(k)$ that is the partial flow of DCVs with destination link d from link l to link p during the time interval $[kT_s, (k+1)T_s)$. Accordingly, $x_l^d(k)$ denotes the partial queue length at the end of link l associated with destination link d. The total inflow of DCVs, associated with destination link d, to link z is given by

$$F_{l,d}^{\text{in}}(k) = \begin{cases} \sum\limits_{p \in L_l^{\text{in}}} q_{p,l}^d(k) & \text{if } l \in L_d \cap (L_{\text{inter}} \cup L_{\text{des}}) \\ Q_{l,d}(k) & \text{if } l \in L_d \cap L_{\text{org}} \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Equation (5) states that the total inflow to link $l \in L_d$ with destination link $d \in L_{\text{des}}$ is the sum of the flows with destination link d from the incoming links of l. Obviously, if l is not on a path to destination link d, its inflow with destination d is zero. If l is an origin link (i.e., $l \in L_{\text{org}}$), then its inflow with destination d is equal to the demand.

The total outflow of DCVs, associated with destination link d, from link l is given by

$$F_{l,d}^{\text{out}}(k) = \begin{cases} \sum\limits_{p \in L_l^{\text{out}} \cap L_d} q_{l,p}^d(k) & \text{if } l \in L_d \cap (L_{\text{org}} \cup L_{\text{inter}}) \\ F_{l,d}^{\text{in}}(k - k_l) & \text{if } l \in L_d \cap L_{\text{des}} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

Equation (6) states that the total outflow of link $l \in L$ with destination link $d \in L_{\text{des}}$ is the sum of flows with destination link d from link l to each outgoing link that is on a path to the destination link d. Clearly, if l is not on a path to destination d, its outflow to destination link d is zero. If l is a link that is directly connected to an unloading station (i.e., $l \in L_{\text{des}}$) then its outflow to destination d is considered to be equal to the delayed version of its inflow. This implies that there will be no queue shaping up at the end of such link.

Finally, the evolution of the partial queue length of DCVs with destination link d at the end of link l, x_l^d in discrete time is given by

$$x_l^d(k+1) = x_l^d(k) + T_s(F_{l,d}^{\text{in}}(k - k_l) - F_{l,d}^{\text{out}}(k)) \tag{7a}$$
$$x_l^d(k) \geq 0 \tag{7b}$$
$$q_{l,p}^d(k) \geq 0 \tag{7c}$$

where T_s is the sampling time of the system and k_l is the number of time steps that the DCVs need to travel the length of link l. The control variables are the partial outflows $q_{l,p}^{d}(k)$. Equation (7a) describes the evolution of the queue lengths. Constraint (7b) is equivalent to (3). Constraint (7c) guarantees non-negativity of the control variables. Let $\boldsymbol{x}(k)$ be the state vector that includes all queue lengths $x_{l,d}(k)$ and delayed samples of $q_{l,p}^{d}(k)$ and $Q_{o,d}(k)$ and let $\boldsymbol{q}(k)$ be the input vector that includes all flows $q_{l,d}(k)$, we obtain

$$\begin{aligned}
&\boldsymbol{x}(k+1) = A\boldsymbol{x}(k) + B_1\boldsymbol{q}_(k) + B_2\boldsymbol{d}(k)\\
&\boldsymbol{x}(k) \geq 0\\
&\boldsymbol{q}(k) \geq 0
\end{aligned}$$

where $\boldsymbol{d}(k)$ is the demand vector composed of all individual demand $Q_{o,d}(k)$ and the matrices A, B_1, and B_2 are defined properly.

VI. Optimal Control Problem

In this section, we define the MPC problem for routing the DCVs through the network. At each controller sampling time, given the current state of the system as the initial state and an estimate of future demand, the model presented in Section V is used to predict the trajectories of the system. Based on this prediction, a constrained optimal control problem is solved over a horizon yielding an optimal sequence of control actions. According to the receding horizon policy, only the first step of such sequence will be applied to the system. This process is then repeated at the next controller sampling time [6].

The objective function must reflect the following performance criteria: i) the pieces of baggage assigned to a certain destination (unloading station) must be reach the destination within a given time window, ii) the energy consumption of the system should be minimized. The time window represents the time duration in which the end point is ready to receive the luggage. It is undesirable to have the luggage arrive at the destination out of this time window. If the pieces of luggage arrive too late, they will miss the flight. Too early arrival of the luggage to the destination point also might inflict a high cost on the operator for storing them until they can be loaded on the plain. The energy consumption is associated with manipulating the actuators in the system and wear and tear inflicted on the actuators. There are two contributors to the energy consumption in the system: i) movements of DCVs in the system, which is related to the magnitude of DCV flows, and ii) variation in the DCV flows. This is particularly important when the DCV flows obtained here will be realized using switch controllers at each junction of the network. The variation in the flow then translates to the frequency of switching.

The constrained linear model proposed in Section V cannot determine the time instant at which a certain flow of baggage reaches to its destination explicitly. However, we can consider a cost function to indirectly penalize baggage arrival time deviation from a given time window. The cost function is composed of three penalty terms. The first penalty term penalizes the queue lengths being defined as

$$J_d^{\mathrm{tw}}(k) = C_d^{\mathrm{tw}}(k)\sum_{l\in L_d} x_l^d(k) \tag{8}$$

where $C_d^{\mathrm{tw}}(k)$ as illustrated in Fig. 2 is given as

$$C_d^{\mathrm{tw}}(k) = \begin{cases} 0 & \text{if } k \leq k_d^{\mathrm{open}} \\ c^{\mathrm{tw}}(k - k_d^{\mathrm{open}}) & \text{if } k_d^{\mathrm{open}} < k \leq k_d^{\mathrm{close}} \\ c^{\mathrm{tw}}(k_d^{\mathrm{open}} - k_d^{\mathrm{close}}) & \text{if } k > k_d^{\mathrm{close}} \end{cases} \tag{9}$$

where k_d^{close} and k_d^{close} are respectively the opening time instant and closing time instant of destination d. Since the $C_d^{\mathrm{tw}}(k)$ is zero before the destination is open, the queue lengths associated with destination d are not penalized before the destination is open. Therefore, the controller will not try to minimize the cost by increasing the DCV flows to destination d to reduce the queue length of DCVs associated with destination d. During the time window the penalty of DCV queues to destination d increases linearly with time. This will make the controller send more DCVs to destination d as the end of time window is approaching. The penalty function saturates to its maximum value at the end of time window otherwise late arrival of DCVs would be preferred.

In line with this objective, the second penalty term penalizes the flows given by

$$J_d^{\mathrm{flow}}(k) = C_d^{\mathrm{flow}}(k)\sum_{l\in L_d}\sum_{p\in L_l^{\mathrm{out}}\cap L_d} q_{l,p}^d(k) \tag{10}$$

with C_d^{flow} as depicted in Fig. 3 being

$$C_d^{\mathrm{flow}}(k) = \begin{cases} -c_1^{\mathrm{flow}}(k - k_d^{\mathrm{open}}) & \text{if } k \leq k_d^{\mathrm{open}} \\ 0 & \text{if } k_d^{\mathrm{open}} < k \leq k_d^{\mathrm{close}} \\ c_2^{\mathrm{flow}}(k - k_d^{\mathrm{close}}) & \text{if } k > k_d^{\mathrm{close}} \end{cases} \tag{11}$$

Before the opening time of the destination d, all flows to destination d are assigned a large penalty that is reducing linearly as the opening time is approaching. During the time window the penalty term is zero, which allows sending more DCVs to destination d. After the time window, the penalty term increases with time to penalize late arrival of DCVs. However, the slope is smaller to allow late luggage to reach destination.

The last two terms of the cost function penalize the energy consumption. We consider a the following penalty terms:

$$J^{\mathrm{e}}(k) = \sum_{d\in D}\sum_{l\in L_d}\sum_{p\in L_l^{\mathrm{out}}\cap L_d} q_{l,p}^d(k) \tag{12}$$

$$J^{\mathrm{sw}}(k) = \sum_{d\in D}\sum_{l\in L_d}\sum_{p\in L_l^{\mathrm{out}}\cap L_d} |q_{l,p}^d(k) - q_{l,p}^d(k-1)| \tag{13}$$

The total cost at time step k is therefore given as

$$J(k) = \sum_{d\in D} J_d^{\mathrm{tw}}(k) + \alpha_1 \sum_{d\in D} J_d^{\mathrm{flow}}(k) + \alpha_2 J^{\mathrm{e}}(k) + \alpha_3 J^{\mathrm{sw}}(k) \tag{14}$$

where $0 < \alpha_i$ is the weight factor showing the relative importance of the associated term in the objective function.

The MPC performance index over the prediction horizon of N_p is thus given as

$$J(k, N_\mathrm{p}) = \sum_{i=k}^{k+N_\mathrm{p}-1} J(i) \tag{15}$$

Now we would like to highlight the following remarks:

R1 The plots of Fig. 2 and Fig. 3 show respectively coefficients of the penalty terms (8) and (10), not the penalty terms themselves. In fact, at the given time step k and for a prediction horizon N_p the value of this coefficients is known for $k, \dots, k+N_\mathrm{p}-1$. Therefore, these coefficients have fixed values and hence the associated penalty terms (8) and (10) are linear in the control variable.

R2 By introducing some dummy variables according to standard techniques in optimization [7], terms of the form (13) can be recast as a linear programming problem with linear constraints.

Therefore, at every time step k we solve the following optimization problem:

$$\min_{\boldsymbol{y}(k)} F^\mathrm{T} \boldsymbol{y}(k) \tag{16}$$

subject to

$$C\boldsymbol{y}(k) \leq \boldsymbol{b} \tag{17}$$

where the vector F is defined based on the MPC objective function (14) and the vector $\boldsymbol{y}(k)$ includes the control inputs $\boldsymbol{q}(k), \dots, \boldsymbol{q}(k+N_\mathrm{p}-1)$ and the dummy variables mentioned in Remark R2. Moreover, $\boldsymbol{b}$ is a constant vector that depends on the current state $x(k)$ and the demand values $\boldsymbol{d}(k), \dots, \boldsymbol{d}(k+N_\mathrm{p}-1)$.

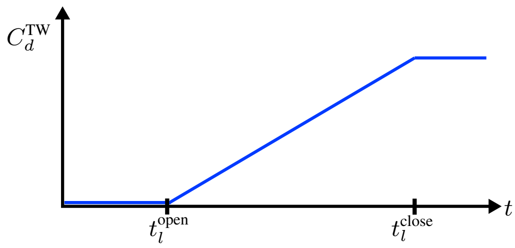


Fig. 2. The coefficient for the queue length penalty term.

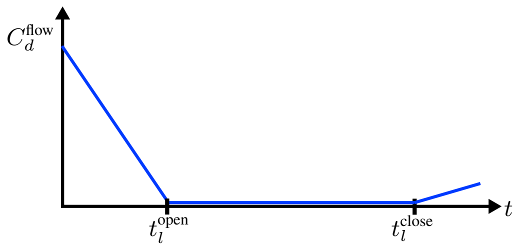


Fig. 3. The coefficient for the flow penalty term.

VII. Case Study

In this section we will illustrate the closed-loop performance of our LP-based control scheme under a given scenario for the simulation parameters listed in Table. I, where the numerical values for the average speed of DCVs v_DCV, the sampling time

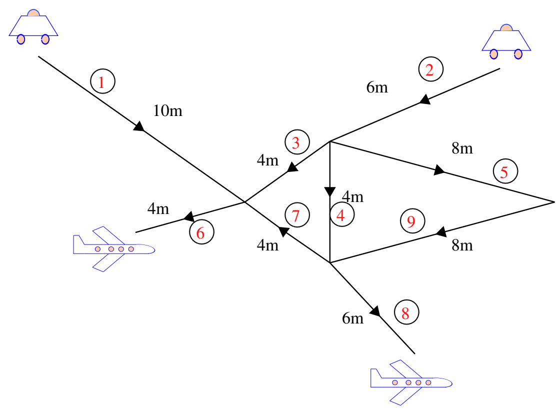


Fig. 4. The network layout. The link label (in circle) and the length of each link in meter is shown next to each link.

TABLE I
SIMULATION PARAMETERS

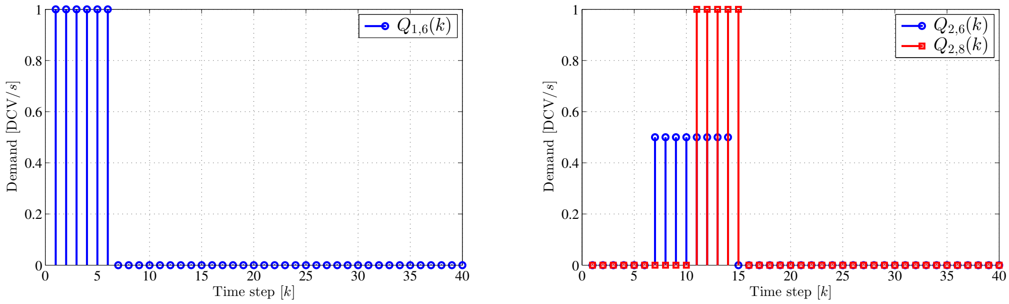
v_DCV [m/s]	T_s [s]	N_p	N_sim	α_1	α_2	α_2
2	1	15	40	1	1	1

T_s, the prediction horizon N_p, the total number of simulation steps N_sim and the weights used in (15) are listed. Then we also compare our method with the MILP-based controller based on the work in [2] in the sense of computational efficiency.

We consider the network layout depicted in Fig. 4, in which the length of each link in meter is indicated. The loading stations are connected to links 1 and 2. The destination links are links 6 and 8. A demand profile as shown in Fig. 5 arrives at the loading stations. We have assigned the time window $[10T_\mathrm{s}, 17T_\mathrm{s}]$ for destination link 6 and the time window $[20T_\mathrm{s}, 25T_\mathrm{s}]$ for destination link 8. The demands with destination link 6 originate from origin links 1 and 2 and arrive at destination link 6 via links 1, 3 or 6. The demands with destination 8 originate from origin link 2 and arrive at destination link 8 via links 4 or 9. Note that there is no path from the origin link 1 to destination link 8. Hence, the demand to destination 8 can only originate from origin link 2.

To analyze the optimal performance of the system, we consider the flow of DCVs arriving at the destination links. See Fig. 5. It it observed from Fig. 6(a) that the flows arrive within the specified time window for destination 6. The DCV flow from link 3 to link 6 falls slightly outside the specified time window as the demand $Q_{2,6}(k)$ of Fig. 5(b) is generated too late. The flow to destination link 8 arrives within the specified time window as depicted in Fig. 6(b). The MILP-based approach with objective function (15) yields the same results closed-loop results under this scenario and for the simulation parameters of Table I.

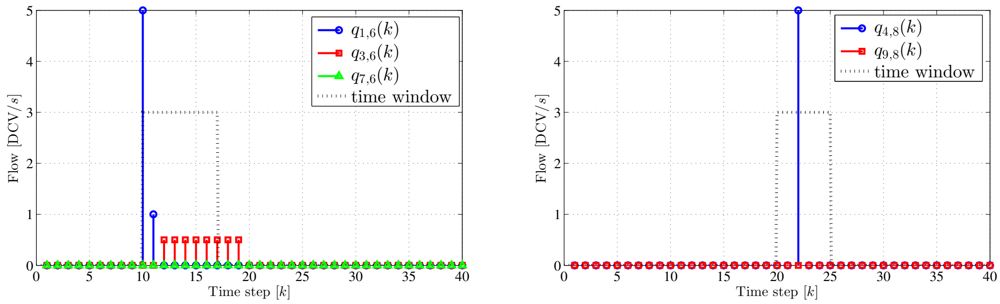
Table II lists the average CPU times for the closed-loop simulation based on the LP approach and on the MILP approach for $N_\mathrm{p} = 4, 8, 12$, where the LP programming problem is solved using SIMPLEX solver of MATLAB Optimization Toolbox and the MILP programming problem is solved using the



(a) Demand profile on link 1.

(b) Demand profile on link 2.

Fig. 5. Demand profile on the loading stations.



(a) DCV flows in link 6.

(b) DCV flows in link 8.

Fig. 6. DCV flows at the unloading stations.

TABLE II
THE AVERAGE, MINIMUM AND MAXIMUM COMPUTATION TIMES FOR LP AND MILP APPROACHES.

N_p	method cpu time[s]	avg. cpu time[s]	min. cpu time[s]	max. cpu time[s]
4	LP	0.0146	0.0049	0.0342
	MILP	0.0692	0.0200	0.2316
8	LP	0.0150	0.0053	0.0351
	MILP	0.1330	0.0033	0.6590
12	LP	0.0165	0.0055	0.3595
	MILP	1.4200	0.0055	31.644

CPLEX solver. Clearly, the LP-based approach is much faster than the MILP-based approach and the difference increases sharply as the prediction horizon N_p increases.

VIII. CONCLUSIONS AND FUTURE WORK

The routing problem in the baggage handling system has been considered. In particular, we addressed an MPC formulation for optimal routing of DCVs within the network of tracks. Previous work in this context had arrived at an MILP formulation. Since MILP is NP hard, a short prediction horizon has to be used then, which comes at a loss of performance. In this paper we have proposed an alternative model for the baggage handling system. Using this model, we showed that due to some assumptions made in the original model, the MPC optimization problem can be recast as an LP problem, which has obvious computational advantages over MILP. Moreover, we have modified the cost function used in the original work to better reflect the objectives of the control system. Closed-loop simulation shows that the LP based approach achieves the same performance as the MILP based approach with far less computational effort. Closed-loop simulation for a simple case study show that the LP approach can be up to 1-2 orders of magnitude faster than the MILP approach in particular if large prediction horizons are considered.

For future work, the optimality of the LP approach will be compared to the optimal solution of a nonlinear optimization approach based on a detailed simulation model of the system.

REFERENCES

[1] A. Tarău, "Model-based control for postal automation and baggage handling," Ph.D. dissertation, Delft University of Technology, 2010.

[2] A. Tarău, B. De Schutter, and H. Hellendoorn, "Hierachical route choice control for baggage handling systems," in *Proceedings of the 12th International IEEE Conference on Intelligent Transportation Systems*, St. Louis, Missouri, USA, Oct. 2009, pp. 679–684.

[3] Y. Zeinaly, B. De Schutter, and J. Hellendoorn, "A model predictive control approach for the line balancing in baggage handling systems," in *Proceedings of the 13th IFAC Symposium on Transportation Systems*, Sofia, Bulgaria, Sep. 2012.

[4] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.

[5] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, pp. 407–427, 1999.

[6] C. Garcia, D. Prett, and M. Morari, "Model predictive control: Theory and practice – A survey," *Automatica*, vol. 25, no. 3, pp. 335 – 348, 1989.

[7] A. Antoniou and W.-S. Lu, *Practical Optimization: Algorithms and Engineering Applications*. Springer, 2007.