Delft University of Technology

Delft Center for Systems and Control

Technical report 08-012

Travel time control of destination coded vehicles in baggage handling systems*

A. Tarău, B. De Schutter, and H. Hellendoorn

If you want to cite this report, please use the following reference instead:
A. Tarău, B. De Schutter, and H. Hellendoorn, "Travel time control of destination coded vehicles in baggage handling systems," *Proceedings of the 17th IEEE International Conference on Control Applications*, San Antonio, Texas, pp. 293–298, Sept. 2008.

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

*This report can also be downloaded via https://pub.deschutter.info/abs/08_012.html

Travel time control of destination coded vehicles in baggage handling systems

Alina Tarău, Bart De Schutter, and Hans Hellendoorn

Abstract— **The process of handling baggage in an airport is time critical. Currently, the fastest way to transport the luggage is to use destination coded vehicles (DCVs). These vehicles transport the bags at high speed on a "mini" railway network. The typical issues of the baggage handling system are coordination and synchronization of the processing units, minimization of the baggage travel time, avoidance of collisions between vehicles, prevention of buffer overflows, and minimization of the transportation costs. In this paper we determine an event-based model of the baggage handling system using DCVs and we investigate the use of centralized control for this system. The proposed control methods are optimal control and model predictive control. This way the optimal route and the optimal speed profile of each vehicle are determined. The considered control methods are compared for several scenarios. Results indicate that optimal control becomes intractable when a large stream of bags has to be handled, while model predictive control can still be used to suboptimally solve the problem.**

I. Introduction

The baggage handling system of an airport plays a decisive role in the airport's efficiency and comfort, which are among the most important factors that determine the airport's ability to attract new airlines or to stay a major airline hub.

The baggage handling system is successful if all the bags are transported to the corresponding lateral (a lateral is the place where the bags are lined up, waiting to be loaded in containers) before the plane has to be loaded. Hence, the process is time critical. The faster the transportation is performed, the more efficient the baggage handling system is.

In order to transport the bags in an automated way, a baggage handling system could incorporate technology such as scanners that scan the labels on each piece of luggage, baggage screen equipment for security scanning, networks of conveyors equipped with junctions that route the bags through the system, and destination coded vehicles (DCVs). The DCVs are carts propelled by linear induction motors, and these carts are mounted on tracks. They transport the bags at high speed on a "mini" railway network.

Briefly, the main control problems of a baggage handling system are coordination and synchronization of the processing units, route assignment of each bag (and implicitly the switch control of each junction), velocity control of each DCV, line balancing, and prevention of buffer overflows.

All authors are with Faculty of Mechanical, Maritime and Materials Engineering, Delft Center for Systems and Control, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands a.n.tarau@tudelft.nl, b@deschutter.info, j.hellendoorn@tudelft.nl

Bart De Schutter is also with the Marine and Transport Technology department of Delft University of Technology.

Synchronization and coordination are required when loading the bags onto the system in order to avoid damaging the bags and blocking the system, or when unloading them to the corresponding lateral.

When using DCVs, the route and the velocity profile of each vehicle have to be determined in order to assure the system optimum. In the literature, the route assignment problem has been discussed in e.g. [1], [2], [3], [4].

The loading conveyor belt can only advance when there is an empty cart onto which the bag on the conveyor belt can be placed. But, since the number of DCVs is limited, empty carts will arrive only after they have delivered their previous loads. Hence, the availability of individual carts is critical for the performance of the entire system. The key in controlling the available capacity of the system consists in fact in assuring a balanced transport service. This problem has been named in [5] the "line balancing" problem.

The buffer overflow problem appears when the capacity of the conveyor belt or any other buffer is exceeded due to e.g. improper line balancing or congestion.

Our goal is to investigate the use of centralized control of the baggage handling system. In this paper we implement advanced control methods such as optimal control and model predictive control to determine the optimal route and velocity for each DCV in the network. We do not consider yet line balancing and prevention of the buffer overflows.

The paper is organized as follows. In Section II, the baggage handling process using DCVs is described, and afterwards, the continuous-time event-driven model of the system is presented. In Section III, several control approaches are proposed for computing the optimal route and optimal velocity of each DCV transporting a bag. The analysis of the simulation results and the comparison of the proposed control methods are elaborated in Section V. Finally, in Section VI, conclusions are drawn and the future directions are presented.

II. Event-driven model

A. Operation of the system

The baggage handling process begins after the bags have passed the check-in. Then they enter the conveyor network, being routed to loading conveyors towards loading stations. Depending on the availability of empty DCVs, at each loading station a queue of bags may be formed. In this paper we focus on the transporting-using-DCVs part of the process. Therefore, one may consider that each loading station has a buffer of bags waiting to be handled as sketched in Figure 1. The baggage handling system operates as follows: given a finite sequence of bags (identified by their unique code) and a buffer of empty DCVs for each loading station, together with the network of tracks, the optimal route and the optimal velocity profile of each DCV have to be computed subject to operational and safety constraints such that the system optimum is assured.

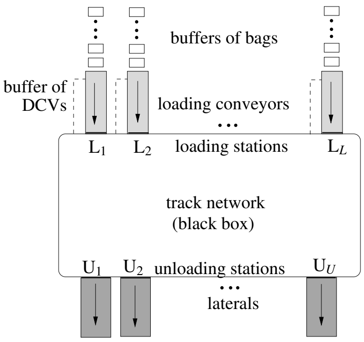


Fig. 1. Baggage handling system using DCVs.

We consider a baggage handling system with L loading stations and U unloading stations as depicted in Figure 1. Accordingly, we have L FIFO (First In First Out) buffers of bags waiting to enter the system.

B. Modeling assumptions

Later on we will use the model for on-line model-based control. So, in order to balance between a detailed model that requires large computation time and a fast simulation we make the following assumptions:

1) a sufficient number of DCVs are present in the system.
2) the capacity of the network is large enough so that no overflow will occur.
3) each loading station has a finite buffer of bags waiting to be handled.
4) all buffers have the same maximum capacity $b_{\max}$.
5) assume there are X bags with random destinations to be handled. They are numbered $1, 2, \cdots, X$. When using a baggage handling system with L loading stations, we split this stream $\mathbf{b} = [1\ 2\ \cdots\ X]^{\mathrm{T}}$ with $X \leq Lb_{\max}$, in L new streams $\mathbf{b_1} = [1\ 2\ \cdots\ l]^{\mathrm{T}}$, $\mathbf{b_2} = [l+1\ l+2\ \cdots\ 2l]^{\mathrm{T}}, \cdots, \mathbf{b_L} = [(L-1)l+1\ (L-1)l+2\ \cdots\ X]^{\mathrm{T}}$ with $l = \left\lfloor \frac{X}{L} \right\rfloor$, where $\lfloor x \rfloor$ denotes the largest integer less than or equal to x.
6) the "mini" railway network has single-direction tracks.
7) a route switch at a junction can be performed in a negligible time span.
8) the speed of a DCV is piecewise constant.
9) the laterals have infinite capacity.
10) the destinations to which the bags have to be transported are allocated to the laterals when the process starts.

Since we consider the line balancing problem solved, these assumptions are reasonable and give a good approximation of the real baggage handling system.

C. Model

There are four types of events that can occur:

- loading a new bag into the system.
- unloading a bag that meets the corresponding lateral.
- updating the route switches at the junction that the DCV has to pass.
- updating the speed of a DCV.

The model of the baggage handling system is an event-driven one consisting of a continuous part describing the movement of the individual vehicles transporting the bags through the network, and of the discrete events listed above.

Define N_i as the number of junctions that a DCV has to pass in order to reach its destination. A track segment is the portion of the track on which a DCV is running either between a loading station and a junction, or between two junctions, or between a junction and an unloading station. Let DCV_i be the DCV that transports the ith bag that entered the system. The following situation has been assumed: given the velocity sequence of the DCV_i say $\mathbf{v_i} = [v_i(0)\ v_i(1)\ \cdots\ v_i(N_i)]^{\mathrm{T}}$ and the sequence of segment lengths $\mathbf{l_i} = [l_i(0)\ l_i(1)\ \cdots\ l_i(N_i)]^{\mathrm{T}}$, on each segment j of length $l_i(j)$ the velocity of DCV_i equals $v_i(j)$ as illustrated in Figure 2. The velocity of the DCV_i that passed segment j, $j = 0, 1, \cdots, N_i - 1$ is updated at time instant $t_{j+1} = t_j + \frac{l_i(j)}{v_i(j)}$ with t_0 the initial time.

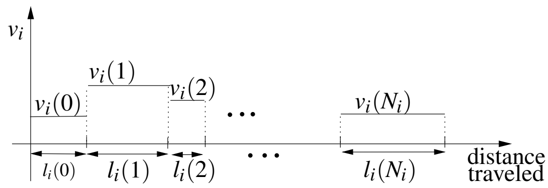


Fig. 2. Speed evolution of the DCV_i.

The model of the baggage handling system is given by the algorithm below, where the loading stations are denoted by $\mathrm{L}_1, \mathrm{L}_2, \cdots, \mathrm{L}_L$, and the unloading stations are denoted by $\mathrm{U}_1, \mathrm{U}_2, \cdots, \mathrm{U}_U$. We also define S as the number of junctions of the track network and $X_{\text{current}}(t)$ as the number of bags that entered the baggage handling system up to the current time instant t.

Algorithm 1. Baggage handling

```
t ← t_0
while there are bags to be handled do
    for ℓ = 1 to L do
        t_load(ℓ) ← time that will pass until the next
                    loading event from L_ℓ's point of view
    end for
    for ℓ = 1 to U do
        t_unload(ℓ) ← time that will pass until the next
                    unloading event from U_ℓ's point of view
    end for
    for s = 1 to S do
        t_switch(s) ← time that will pass until the next
                    route switch event from the junction s's
                    point of view
    end for
    for i = 1 to X_current(t) do
```

13: **if** bag i is not at a lateral **then**
14: $t_{\text{speed\_update}}(i) \leftarrow$ time that will pass until the next speed-update event from the point of view of the DCV$_i$
15: **end if**
16: **end for**
17: $t_{\min} \leftarrow \min(\min_{\ell=1,\cdots,L} t_{\text{load}}(\ell), \min_{\ell=1,\cdots,U} t_{\text{unload}}(\ell), \min_{s=1,\cdots,S} t_{\text{switch}}(s), \min_{i=1,\cdots,X_{\text{current}}(t)} t_{\text{speed\_update}}(i))$
18: $t \leftarrow t + t_{\min}$
19: update the state of the system
20: **if** $t_{\min} = \min_{i=1,\cdots,X_{\text{current}}(t)} t_{\text{speed\_update}}(i)$ **then**
21: update the speed of the DCV$_i$
22: **end if**
23: **end while**

If multiple events occur at the same time, then we take all these events into account when updating the state of the system at step 19.

D. Operational constraints

The operational constraints derived from the mechanical and design limitations of the system are the following:

- the velocity of each DCV is bounded between 0 and $v_{\max}$.
- a bag can be loaded onto a DCV only if there is an empty DCV under the loading station.
- a DCV can transport only one bag.
- collisions between DCVs have to be avoided on each track segment and at each intersection.

III. Control approaches

In this paper we consider several centralized control approaches that determine the route and the speed of each DCV such as finite-horizon optimal control and model predictive control.

A. Optimal control with variable speed profile

Several methods for solving dynamic optimization problems have been developed. The optimal control problem consists of finding the time-varying control law $u(\cdot)$ for a given system such that a performance index $J(u(\cdot))$ is optimized while satisfying the operational constraints imposed by the model, see e.g. [6].

In this paper the function J is being defined as follows:

$$J = \sum_{i=1}^{X} t_{\text{dwell}}(i) \tag{1}$$

where X is the number of bags and $t_{\text{dwell}}(i)$ is the time that bag i spends in the baggage handling system.

The performance index J is influenced by the route that each bag takes and by the velocity profile of each DCV.

Assuming that there are R possible routes named 1, 2, $\cdots$, R, the route of DCV$_i$ is $r(i)$, $i = 1, 2, \cdots, X$. Then the route sequence is defined as $\mathbf{r} = [r(1)\, r(2) \cdots r(X)]^{\mathrm{T}}$.

First, we consider the case where Assumption 8 does not hold. One could denote by $\mathbf{V}_{\mathrm{t}}(\cdot)$ the time-varying speed vector of the DCVs transporting bags, defined as $\mathbf{V}_{\mathrm{t}}(\cdot) = [v_{\mathrm{t},1}(\cdot)\, v_{\mathrm{t},2}(\cdot) \cdots v_{\mathrm{t},X}(\cdot)]^{\mathrm{T}}$ where the function $v_{\mathrm{t},i} : [t_0, t_{\text{end}}] \to [0, v_{\max}] : t \mapsto v_{\mathrm{t},i}(t)$ with $v_{\mathrm{t},i}(t)$ the speed of the DCV$_i$ at time t and t_{end} the time instant when all X bags have been sorted. Then the optimal control problem would be defined as follows:

P1: $\min_{\mathbf{r}, \mathbf{V}_t} J(\mathbf{r}, \mathbf{V}_t(\cdot))$
subject to
the system dynamics
operational constraints

But, continually adjusting the velocity of each DCV transporting bags through the system so as to minimize the performance index J requires extremely high computational effort. In practice, the problem P1 becomes intractable when the number of possible routes and the number of bags to be transported are large. The computational effort can be reduced when considering e.g. the piecewise constant speed profile of each DCV. Therefore, Assumption 8 is necessary.

B. Optimal control with piecewise constant speed profile

One way to simplify P1 is to consider that the speed of a DCV remains constant while it is running along one segment.

The piecewise constant speed profile of the DCV$_i$ is defined as $v_i : \{0, 1, \cdots, N_i\} \to (0, v_{\max}]$. Then we have to determine the route control sequence $\mathbf{r} = [r(1)\, r(2) \cdots r(X)]^{\mathrm{T}}$ and the tuple $\mathscr{V} = (\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_X)$ with $\mathbf{v}_i = [v_i(0)\, v_i(1) \cdots v_i(N_i)]^{\mathrm{T}}$ for $i = 1, 2, \cdots, X$ that minimize the performance index J defined in (1) subject to the dynamics of the system and the operational constraints:

P2: $\min_{\mathbf{r}, \mathscr{V}} J(\mathbf{r}, \mathscr{V})$
subject to
the system dynamics
operational constraints

The computational effort of solving the optimal control problem P2 increases with the number of junctions in the network and with the number of bags to be handled.

C. Optimal control with constant speed profile

The simplest optimal control problem to solve would be to determine the route and the constant velocity of each DCV transporting bags so as to minimize the performance index J previously defined:

P3: $\min_{\mathbf{r}, \mathbf{v}_{\text{ct}}} J(\mathbf{r}, \mathbf{v}_{\text{ct}})$
subject to
the system dynamics
operational constraints

where $\mathbf{r} = [r(1)\, r(2) \cdots r(X)]^{\mathrm{T}}$ is the route sequence, $\mathbf{v}_{\text{ct}} = [v_{\text{ct}}(1)\, v_{\text{ct}}(2) \cdots v_{\text{ct}}(X)]^{\mathrm{T}}$ is the velocity sequence and $v_{\text{ct}}(i) \in (0, v_{\max}]$ for $i = 1, 2, \cdots, X$ is the constant speed of DCV$_i$.

Solving P3 gives the smallest computational effort, but at the cost of suboptimal results.

D. Model predictive control

In order to make a trade-off between the optimality and the time required to compute the optimal velocity profile of each DCV transporting bags, model predictive control (MPC) is introduced.

Model predictive control is an on-line control design method that uses the receding-horizon principle, see e.g. [7].

In the basic MPC approach, given a prediction horizon N_p and a control horizon N_c with $N_\mathrm{c} \leq N_\mathrm{p}$, at time step k, the future control sequence $u(k|k), \cdots, u(k+N_\mathrm{c}-1|k)$ is computed by solving a discrete-time optimization problem over a given time step period $[k, k+N_\mathrm{p}]$ so that the cost criterion J is optimized subject to constraints on the inputs and outputs. The input signal is typically assumed to become constant beyond the control horizon i.e. $u(k+j|k) = u(k+N_\mathrm{c}-1|k)$ for $j \geq N_\mathrm{c}$. After computing the optimal control sequence, only the first control sample is implemented, and subsequently the horizon is shifted. Next, the new state of the system is measured or estimated, and a new optimization problem at time step $k+1$ is solved using this new information. In this way, a feedback mechanism is introduced.

In our case, k is not time index, but bag index. Also, computing the control $u(k|k)$ consists of determining the route and the piecewise constant speed profile of a DCV.

In this variant of MPC the prediction horizon corresponds to the number of bags that we let to enter the baggage handling system. The control horizon is equal to the prediction horizon ($N_\mathrm{p} = N_\mathrm{c} = N$) since, in this case, the control horizon constraint cannot be applied. This happens due to the fact that the DCVs transporting the bags do not have the same route and implicitly the number of segments they have to pass is different. At step k, where k is the number of bags in the system, the controls $u(j|k) = (r(k+j), \mathbf{v}_{k+j})$ for $j = 1, 2, \cdots, N$ are computed such that the total dwell time of the next N bags that enter the system is minimized. The MPC optimization problem at bag step k is defined as follows:

P4: $\min\limits_{\mathbf{r}(\mathbf{k}), \mathscr{V}(k)} J_N(\mathbf{r}(k), \mathscr{V}(k))$
subject to
the system dynamics
operational constraints

where $\mathbf{r}(k) = [r(k+1)\ r(k+2)\ \cdots\ r(k+N)]^\mathrm{T}$ is the future route sequence and $\mathscr{V}(k) = (\mathbf{v}_{k+1}, \mathbf{v}_{k+2}, \cdots, \mathbf{v}_{k+N})$ with $\mathbf{v}_{k+i} = [v_{k+i}(0)\ v_{k+i}(1)\ \cdots\ v_{k+i}(N_i)]^\mathrm{T}$ for $i = 1, 2, \cdots, N$ is the future velocity profile for the next N bags entering the baggage handling system.

Only the control $(r(k+1), \mathbf{v}_{k+1})$ will be applied. Given the state of the system after applying the MPC control, a new optimization will be solved over the prediction horizon.

The main advantage of MPC consists in a smaller computation time than the one needed when using optimal control with piecewise constant speed profile. Even more, the velocity of each DCV may be computed on-line. However, this happens at the cost of a suboptimal performance of the baggage handling system.

IV. Optimization methods

In order to solve the optimization problems presented in the previous section, the route for each DCV and its speed profile have to be determined. The route can be represented by an integer value, while the velocity of each DCV at any time instant is a real value. Therefore, to solve any of the optimization problems P2, P3, or P4, one might use *mixed-integer* algorithms such as branch and bound methods, *genetic* algorithms, or *simulated annealing* algorithms, see e.g. [8], [9], [10].

Another way to solve the optimization problems is to determine, in an outer loop, the route for each DCV using e.g. *genetic* algorithms and, in an inner loop, the optimal velocity profile using e.g. multi-start *pattern search* or a *sequential quadratic programming* algorithm, see [11], [12].

Finally, one could apply a greedy two step approach, split the mixed-integer problem and determine first the optimal route (considering e.g. that each DCV can run with the maximal speed, and neglecting the possible collisions); afterwards, only the problem of computing the velocity profile of each DCV remains to be solved.

V. Case study

In this section we compare the proposed control methods and the optimization techniques that could be used to solve the optimal and MPC control problems. The comparison will be made based on simulation examples.

A. Set-up

We consider the network of tracks depicted in Figure 3 with two loading stations L_1 and L_2, two unloading stations U_1 and U_2, and two junctions I_1 and I_2. The route of any bag entering the system can be either "direct" or "indirect", depending on whether, in its way to the corresponding lateral, the bag passes one or two junctions respectively. We have considered this network because on the one hand it is simple, allowing an intuitive understanding of and insight in the operation of the system and the results of the control approaches, and because on the other hand, it also contains all the relevant elements of a real set-up.

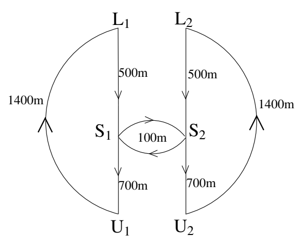


Fig. 3. Network of tracks as a directed graph.

We assume that the velocity of each DCV varies between 0 m/s and 20 m/s. The lengths of the track segments are indicated in Figure 3.

In order to faster assess the efficiency of our control method we assume that we do not start with an empty network but with a network populated by autonomous DCVs that transport bags in such a way that no collisions occur. Starting at the time instant t_0, the bags that enter the system will be transported by DCVs that have their velocities determined by one of the controllers developed in this paper.

B. Scenarios

We have defined twenty-five scenarios where the stream of bags that will enter the system after t_0 has the length $1, 2, \cdots, 25$, the destination of each bag being randomly assigned. All these scenarios start from the same initial state of the system.

C. Control problem

Note that since for the network of Figure 3 there is only one correct route between a loading station and a lateral, we only control the velocity of the DCVs.

First we compare several optimization methods that may solve the optimal control problem with constant speed profile. Afterwards, we select the optimization technique that gives a good trade-off between performance and computational effort, and we compare the proposed control approaches.

As optimization techniques we have considered the Matlab functions *pattern search*, *genetic algorithm*, and *simulated annealing* incorporated in the Genetic Algorithm and Direct Search Toolbox, and the *glcFast* solver of the *mixed-integer nonlinear problems* incorporated in the Matlab/Tomlab Optimization Environment.

The performance index J is the total dwell time of the bags that enter the system starting at the time instant t_0.

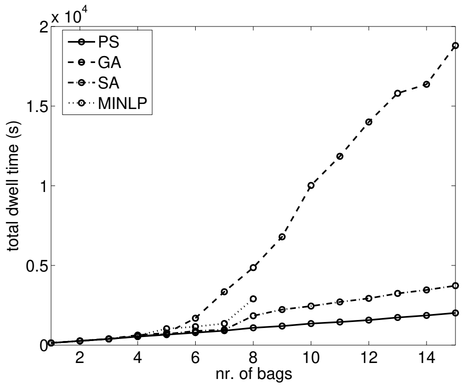
D. Results

Using the same scenarios for each of the optimization methods, we illustrate in Figure 4 the evolution of the total dwell time, of the computation time[1] required to determine the optimal velocities, and of the number of evaluations with collision result, with respect to the number of bags to be handled. For the *multi-start* functions *pattern search* and *simulated annealing* we have used both structured and random starting points. A structured starting point is a feasible initial guess chosen based on a priori knowledge. In our case this starting solution consists of all non-autonomous DCVs being assigned the same low velocity i.e. the smallest velocity of the autonomous DCVs. This choice has been considered because (due to Assumption 7) using the same very low velocity for all the non-autonomous DCVs usually yields a local solution.

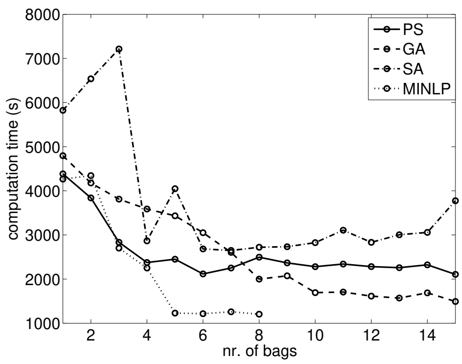
If there is a collision or if a bag does not reach its corresponding lateral before the plane has to be loaded, the total dwell time is set to a very high value, say 10^8 s. In order to make a fair comparison, we have imposed the same number of function evaluations (10^4) for each of the considered optimization methods. The evaluation of a possible sequence of speed profiles ends either when all the bags have been transported, or when a collision occurs. Therefore, the more the number of evaluations with collision result increases, the more the computation time decreases.

These results show that the *glcFast* solver of the *mixed-integer nonlinear problems* finds a local solution only for up to 8 bags to be handled. Also, the more the number of bags to be handled increases, the more the number of evaluations with collision result increases and implicitly the computation time decreases, but the obtained results are less optimal. The *pattern search* algorithm gives the best performance while the computational effort required to compute a local optimum starting from a feasible initial guess is the smallest. Therefore, it will be further used in comparing the proposed control methods.

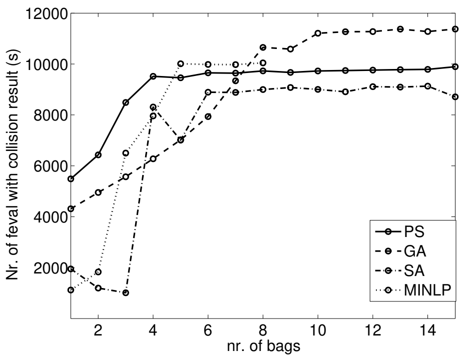
We first consider improving the performance and afterwards, the computational effort is taken into account. Usually Assumption 7 does not hold. Therefore, one may only consider random starting points and repeat the search for a local



(a) total dwell time versus the number of bags to be handled



(b) computation time versus the number of bags to be handled

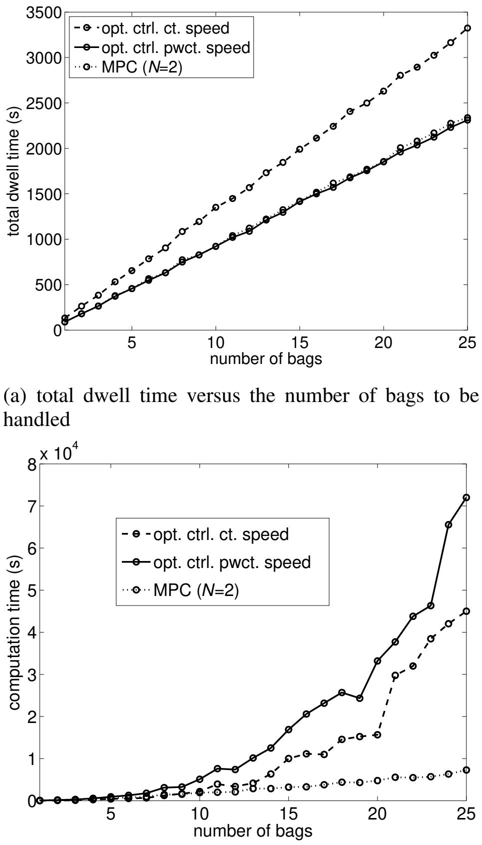


(c) number of function evaluations with collision result versus the number of bags to be handled

Fig. 4. Comparison of the proposed optimization methods when using optimal control with constant velocity profile.

[1]The simulations were performed on a 3.0 GHz P4 with 1 GB RAM.

optimum until a solution is found. Then the computation time required to find a solution using e.g. the optimal control with constant speed profile method when handling 12 bags reached $5 \cdot 10^5$ s. So, the optimal control method becomes intractable when the number of bags to be handled is large. Therefore, a trade-off has to be made between the performance and the computation time. In this context MPC provides a balanced trade-off due to the receding-horizon principle that the approach uses, each optimization being solved for a relative small stream of bags. As a consequence, MPC assures the speed control of each DCV even for a large number of bags.



(a) total dwell time versus the number of bags to be handled

(b) computation time versus the number of bags to be handled

Fig. 5. Comparison of the proposed control approaches.

For all of the proposed control approaches, we have illustrated in Figure 5 the performance index J i.e. the total dwell time and the computation time with respect to the number of bags to be handled. To faster solve the optimization problems, we have used the same structured starting point.

Simulations confirm that applying optimal control with piecewise constant speed profile yields better performance than applying optimal control with constant speed profile. Also, MPC gives results that are very close to the ones obtained when using optimal control with piecewise constant speed profile.

VI. Conclusions and Future Work

In this paper we have considered the baggage handling process in large airports using destination coded vehicles (DCVs) running at high speed on a "mini" railway network, together with the main control problems of the baggage handling systems. A fast event-driven model of the continuous-time baggage handling process has been determined. We have investigated the use of centralized control of the baggage handling systems using DCVs. The considered control methods are optimal control and model predictive control.

Results confirm that optimal control can be used for determining the optimal piecewise constant or constant velocity profile of each DCV. However, this is only possible for a limited number of bags to be handled. Therefore, receding-horizon approaches are needed so as to efficiently transport a large number of bags. Simulations show that MPC can be successfully used to suboptimally solve the baggage handling problem due to the receding-horizon principle that the approach uses.

In future work other (receding-horizon) control methods will be considered such as fast heuristic approaches, fuzzy control, case-based control, distributed control, etc. We will also include more complex dynamics of the system than the ones considered in this paper.

Acknowledgment

Research supported by the STW-VIDI project "Multi-Agent Control of Large-Scale Hybrid Systems", by the BSIK project "Next Generation Infrastructures" (NGI), by the Transport Research Centre Delft, and by the Delft Research Centre Next Generation Infrastructures.

References

[1] T. Friesz, J. Luque, R. Tobin, and B. Wie, "Dynamic network traffic assignment considered as a continuous time optimal control problem," *Operations Research*, vol. 37, no. 6, pp. 893–901, 1989.

[2] D. Kaufman, J. Nonis, and R. Smith, "A mixed integer linear programming model for dynamic route guidance," *Transportation Research Part B: Methodological*, vol. 32, no. 6, pp. 431–440, 1998.

[3] H. Gang, J. Shang, and L. Vargas, "A neural network model for the free-ranging AGV route-planning problem," *Journal of Intelligent Manufacturing*, vol. 7, no. 3, pp. 217–227, 1996.

[4] M. Duinkerken, J. Ottjes, and G. Lodewijks, "Comparison of routing strategies for AGV systems using simulation," in *Proceedings of the 2006 Winter Simulation Conference*, 2006, pp. 1523 – 1530.

[5] R. de Neufville, "The baggage system at Denver: Prospects and lessons," *Journal of Air Transport Management*, vol. 1, no. 4, pp. 229–236, 1994.

[6] F. Lewis, *Optimal Control*. New York, USA: John Wiley & Sons, Inc., 1986.

[7] J. Maciejowski, *Predictive Control with Constraints*. Harlow, UK: Prentice Hall, 2002.

[8] C. Floudas, *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications*. New York, USA: Oxford University Press, 1995.

[9] C. Reeves and J. Rowe, *Genetic Algorithms - Principles and Perspectives: A Guide to GA Theory*. Norwell, MA, USA: Kluwer Academic Publishers, 2002.

[10] K. Dowsland, "Simulated annealing," in *Modern heuristic techniques for combinatorial problems*, C. Reeves, Ed. New York, USA: John Wiley & Sons, Inc., 1993, pp. 20–69.

[11] C. Audet and J. Dennis, "Analysis of generalized pattern searches," *SIAM Journal on Optimization*, vol. 13, no. 3, pp. 889–903, 2003.

[12] S. Han, "A globally convergent method for nonlinear programming," *SIAM Journal on Optimization*, vol. 22, no. 3, pp. 297–309, 1977.