Transportation Research Part C 18 (2010) 757–769

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Hybrid predictive control for real-time optimization of public transport systems' operations based on evolutionary multi-objective optimization

Cristián E. Cortés [a,*], Doris Sáez [b], Freddy Milla [b], Alfredo Núñez [b], Marcela Riquelme [c]

[a] *Civil Engineering Department, Universidad de Chile, Santiago, Chile*
[b] *Electrical Engineering Department, Universidad de Chile, Santiago, Chile*
[c] *CONTAC S.A., Santiago, Chile*

ARTICLE INFO

Article history:
Received 2 January 2009
Received in revised form 11 May 2009
Accepted 16 May 2009

Keywords:
Public transport system
Hybrid predictive control
Multi-objective optimization
Genetic algorithms

ABSTRACT

A hybrid predictive control formulation based on evolutionary multi-objective optimization to optimize real-time operations of public transport systems is presented. The state space model includes bus position, expected load and arrival time at stops. The system is based on discrete events, and the possible operator control actions are: holding vehicles at stations and skipping some stations. The controller (operator) pursues the minimization of a dynamic objective function to generate better operational decisions under uncertain demand at bus stops. In this work, a multi-objective approach is conducted to include different goals in the optimization process that could be opposite. In this case, the optimization was defined in terms of two objectives: waiting time minimization on one side, and impact of the strategies on the other. A genetic algorithm method is proposed to solve the multi-objective dynamic problem. From the conducted experiments considering a single bus line corridor, we found that the two objectives are opposite but with a certain degree of overlapping, in the sense that in all cases both objectives significantly improve the level of service with respect to the open-loop scenario by regularizing the headways. On average, the observed trade-off validates the proposed multi-objective methodology for the studied system, allowing dynamically finding the pseudo-optimal Pareto front and making real-time decisions based on different optimization criteria reflected in the proposed objective function compounds.

1. Introduction

The design of a public transport system run by buses requires the optimization of the number of bus lines and their associated routes, the fleet composition of each line, and the optimal frequency, for the expected passenger demand during peak periods. Even though operational variables, such as the frequency, are optimized for different periods and lines, it is difficult to regularize the movement of buses as they are affected by different disruptions as the day progresses, such as traffic congestion, unexpected delays, randomness in passenger demand (both spatial and temporal), irregular vehicle dispatching times, incidents and so on. In the literature, as an attempt to reduce the negative effects of service disturbance, researchers have devoted significant effort to develop flexible control strategies, either in real-time or offline, depending on the specific features of the problem.

These strategies have been designed to allow the operator reacting dynamically to real-time system disturbances. The most studied strategy of this type in the last years is the holding strategy, in which vehicles are held at certain stations

* Corresponding author. Fax: +56 26894206.
E-mail addresses: ccortes@ing.uchile.cl (C.E. Cortés), dsaez@ing.uchile.cl (D. Sáez), fmilla@ing.uchile.cl (F. Milla), alfnunez@ing.uchile.cl (A. Núñez), mriquelme@contac.cl (M. Riquelme).

0968-090X/$ - see front matter
doi:10.1016/j.trc.2009.05.016

for a determined time, in most cases designed to keep the headway between successive buses deterministic as far as possible.

With regard to the most recent contributions in holding modeling, Hickman (2001) developed a stochastic holding model at a given control station, obtaining a convex quadratic program in a single variable corresponding to the time lapse during which buses are held. More recent research has explored holding models relying on online vehicle location. Among them, Eberlein (1995) and Eberlein et al. (1999, 2001) developed deterministic quadratic programs under a rolling horizon scheme. In their approaches, the holding decision for a specific vehicle affects the operation of a specific subset of the precedent vehicles. The authors concluded that having two or more holding stations over a corridor is not necessary. On the opposite, Sun and Hickman (2004) concluded that holding multiple vehicles in several control stations would be better than having a single station to hold buses.

Another interesting dynamic control strategy is the real-time expressing problem (station skipping). Their approach consists of speeding up buses by skipping stations (one or more) with the objective of minimizing the total waiting time, considering the extra waiting time of passenger whose station has been skipped. In the literature, expressing is formulated as a pre-planned strategy (Jordan and Turnquist, 1979; Furth, 1986) and, more frequently, as a real-time control strategy (Eberlein, 1995; Eberlein et al., 1999; Fu and Liu, 2003; Sun and Hickman, 2005). In most cases, a station skipping decision is made before the buses depart from the terminal, except by the work by Sun and Hickman (2005), who consider as decision variables the first and last stations of the skipped section of the bus route. They found that in some cases a strategy that allows buses to stop at a skipped station if there are passengers who need to get off there (allowing passengers to get on the bus there) outperforms the basic strategy, where passengers whose destination is inside the skipped section must get off before their destination station.

Eberlein (1995) developed an integrated model, considering holding and expressing. Additionally, Adamski and Turnau (1998) and Adamski (1996) developed a simulation decision-support tool for dynamic optimal dispatching control, including punctuality, regularity and synchronizing control based on a linear quadratic feedback control scheme, while considering system state constraints. They extended their work to a linear quadratic stochastic control with real-time estimation of model parameters.

Most of the revised models for real-time control strategies as found in the public transport literature are based on heuristics to solve the problems due to the mathematical complexity of the formulations, lacking of a dynamic control framework. Sáez et al. (2007, 2009) formalized the problem by integrating the two aforementioned strategies (holding and expressing) to solve a real-time public transport control problem with uncertain passenger demand, relying on online information of system behavior. The model is formulated as a hybrid predictive control (HPC) problem, in which state space equations and a proper predictive objective function are crucial steps in the model definition, as the proposed approach has the capability of optimizing system performance in real-time based on such a properly chosen objective function (see e.g. Hegyi, 2004). The model is able to estimate the effects of the control actions on the behavior of the bus system, and also allow the inclusion of complex system constraints.

Sáez et al. (2007, 2009) highlights the fact that the underlying theory behind HPC fits nicely into the dynamic conditions of typical public transport problems. Moreover, predictive approaches are suitable for dynamic environments with high uncertainty of future events, which can become relevant for the making decision process that has to be performed in real-time. One feature of this model is the multidimensionality of the proposed dynamic objective function (comprising a variety of different terms associated with different components of the system), which is solved as a mono-objective optimization problem.

In this paper, we extend the mono-objective hybrid predictive control strategy to a formulation based on evolutionary multi-objective optimization in order to optimize real-time control operations of a bus system considering the different aspects of the multidimensionality of the embedded problem. The hybrid model based on discrete events considers some relevant state space variables, such as bus position, expected load and arrival time at stops. The possible control actions to be applied by the operator in real-time are holding vehicles at specific stations and skipping some bus stops to coordinate the joint operation. The dynamic formulation of the system requires the demand forecast based on offline as well as online data.

The real-time passenger demand, which is unknown and uncertain, is modeled as a disturbance for the predictive scheme. The control strategies permit to incorporate in the model the future behavior of the system associated with the operation of the buses, by using an online prediction system for the disturbances (demand). Nevertheless, the formulation does not depend on the specific demand prediction methodology. In such a context, an estimate of the state space variables at future events can be obtained (Sáez et al., 2007, 2009).

The predictive controller (bus operator) uses such information to minimize a proper dynamic objective function, generating better current decisions under uncertain demand at bus stops. He (she) dynamically provides the control actions to the bus system in order to optimize the performance according to a two-dimensional objective function. The two dimensions correspond to the regularization of bus headways on the one hand, and the minimization of the impact on the system due to the application of the strategies on the other. The former term is related to the minimization of the waiting time of passengers at bus stops, while the latter penalizes the extra travel and waiting time of some passengers affected by the strategies (holding and station skipping). In this paper, we formalize these two apparent conflicting (opposite objectives) in a dynamic evolutionary multi-objective optimization (EMO) framework for the real-time control of a bus system based on hybrid predictive control as formulated by Sáez et al. (2007, 2009).

The structure of the formulation and solution of the EMO problem for a HPC model was recently developed for a dynamic pick-up and delivery problem by our research group (Núñez et al., 2007, 2008), where EMO is used to solve such a problem, considering the operator and user costs as the opposite dimensions in the objective function. The formulation is implemented for a test system and efficiently solved with genetic algorithms (GA), which turn out to be very effective to build dynamic pseudo-optimal Pareto fronts as discussed later in this paper.

In the evolutionary multi-objective optimization literature, most of the problems are static (Hajri-Gabouj, 2003). Literature on dynamic EMO problems is scarce and it lacks of clear evaluation methodologies (Farina et al., 2004). There is an interesting work by Tan et al. (2007), where a multi-objective stochastic vehicle routing problem is solved via EMO.

In the literature, predictive control based on multi-objective optimization was reported under different approaches. Alvarez and Cruz (1998) propose a multi-objective dynamic optimization method for discrete time systems. First, a multi-objective programming sub-problem is solved with general constraints at each time-step. Then, policies that satisfy the necessary optimality conditions for this problem are derived. The priorized policies are used as criteria for choosing the optimal control action. Kerrigan et al. (2000) present several methods for handling a large class of multi-objective formulations and priorizations for model predictive control of hybrid systems. The methods are based on mixed logical dynamical (MLD) models for prioritizing soft constraints of predictive control strategies, guaranteeing the satisfaction of a maximum number of hard constraints.

Next, Kerrigan and Maciejowski (2002) solve the multi-objective predictive control problem based on priorized constraints and objectives. In this case, the most important optimization problem is solved first and the solution to this problem is then used to impose additional constraints on the second optimization, etc. Also, the control action of predictive controller proposed is solved based on convex programming techniques by considering certain convexity assumptions. Thus, priorized multi-objective predictive controller can be solved online without redesigning the controller offline; however, this increase in flexibility also demands an increase in the amount of online computational power. Nunez-Reyes et al. (2002) present a comparison of three different predictive controllers applied to an olive oil mill, which are a typical model-based predictive control (MPC) approach based on a mono-objective function, a priorized multi-objective predictive controller and a structured MPC controller. The structured MPC uses a decision list to select the current objective function, which must be supplied to the control action. Based on simulation tests, the priorized multi-objective predictive controller gives the best results without the need of tuning weights as the mono-objective MPC, although a relevant computational cost is required. An intermediate solution is the structured MPC.

Zambrano and Camacho (2002) describe a multi-objective model predictive control algorithm based on a goal attainment method, which considers the different objective functions as constraints for the minimization of the relaxation variable. This multi-objective predictive controller allows the specification of different goals, like economic factor, at different operation points and was applied to a solar refrigeration plant and formulated for variable configuration systems. The results show benefits of including the multi-objective approach. On the other hand, Laabidi and Bouani (2004) present a multi-objective control strategy for non-linear uncertain dynamic systems by using a neural network for modeling. Non dominated sorting genetic algorithm is used for solving the multi-objective optimization problem. Each objective function corresponds to the conventional MPC objective function (minimizing tracking error and control effort) obtaining predictions with different neural networks models of the system. The criterion for choosing the optimal control action considers taking only the solution which gives the minimal sum of the objective functions.

Regarding recent applications, Subbu et al. (2006) present a multi-predictive multi-objective optimization approach for thermal power plants and Hu et al. (2007) discusses the development of a dynamic multi-objective predictive control system for generating cost-effective control strategies for a bioremediation site.

Thus, multi-objective predictive controllers reported are interesting developments; however, the systematic tuning methodology design is not complete. Then, in this work a general methodology for dynamic multi-objective hybrid predictive controller that provides generic solutions based on genetic algorithms is proposed.

In our proposed hybrid predictive control approach based on evolutionary multi-objective optimization (HPC-EMO), we include discrete (number of passengers on the buses) as well as continuous (bus position and speed) variables. For this reason, a hybrid predictive approach is utilized, in which control actions are optimized considering both kinds of variables.

The present paper is structured as follows. In the next section we describe the model formulation. Later, in Section 3 we present the HPC-EMO strategy and the genetic algorithm designed for solving the resulting problem under the dynamic EMO approach. The presentation proceeds with illustrative applications of the methodology in Section 4, to finish in Section 5 with a synthesis, conclusions and further research lines.

2. Dynamic modeling for the hybrid predictive control design

2.1. Formulation and state space model

The objective of this section is to summarize the HPC approach proposed in Sáez et al. (2007, 2009) for a real-time bus system optimization. To be efficient in terms of computation time, the HPC framework is written in discrete time. The problem is then formulated as a hybrid predictive system, where events are triggered by specific actions. Unlike traditional HPC formulations written for a fixed step-size, this scheme is based on the occurrence of relevant events (corresponding to the instants at which control actions have to be taken); in this case the formulation results in a variable step-size as a proxy for expected bus arrival times at bus stops.

For the sake of simplicity, in this work the HPC framework is constructed for a single loop bus system. The system is represented in Fig. 1 below. The network is a one-way loop route, with N equidistant stations and b buses running around the loop, under the control of the dispatcher. Station 1 is the terminal of the bus route. All passengers have to get off the bus there. Passengers arrive at each station at certain rate, with destination among the stations ahead of the station the passenger is getting on. From historical data, a representative stop-to-stop demand matrix can be estimated for each modeling period, which helps us add the predictive feature in the model. Complementarily, online demand data can be used to enrich predictions.

Under this scheme, the events are triggered when a bus arrives at a bus stop, which determines a variable time-step. Hereafter, we denote t as the continuous time, k as the event, and t_k as the continuous time at which event k occurs. Note that an event k is always associated with the arrival of a specific bus to a specific bus stop.

Sáez et al. (2007, 2009) define two state space variables in order to check the bus status and consequently trigger the events. These are the position of the bus at any continuous instant t, x_i, and the expected remaining time for the bus to reach the next stop, $\hat{T}_i$. Specifically, the manipulated variables are the holding $h_i(k)$ and the station skipping $Su_i(k)$ actions associated with bus i and event k. Thus, $h_i(k)$ is the lapse during which bus i is held at the stop associated with event k, while $Su_i(k)$ is a binary variable that is equal to one if passengers at allowed to board bus i at the stop associated with event k, zero otherwise.

The output variables correspond to the estimated passenger load $\hat{L}_i(k+1)$ and the estimated departure time $\hat{T}d_i(k+1)$ once the bus departs from its current stop, associated with the bus i that triggered event k.

The analytical expressions for such a dynamic model associated with bus i that triggered event k can be summarized as follows (Sáez et al., 2007, 2009):

$$\begin{aligned}
x_i(t) &= x_i(t_k) + \int_{t_k}^{t} v_i(\vartheta)d\vartheta \\
\hat{T}_i(t) &= t_k + h_i(k) + \hat{T}r_i(k) + Tv_i(k) - t \\
\hat{L}_i(k+1) &= L_i(k) + Su_i(k)(\hat{B}_i(k) - \hat{A}_i(k)) \\
\hat{T}d_i(k+1) &= t_k + h_i(k) + \hat{T}r_i(k)
\end{aligned} \tag{1}$$

where $\hat{B}_i$ corresponds to the expected number of passenger that will board bus i while it is at the stop, and $\hat{A}_i$ represents the estimated number of passenger alighting from bus i at event k.

In (1), v_i is the bus instantaneous speed as a function of the continuous time, and $\hat{T}r_i,\ Tv_i$ show the estimated passenger transference delay (maximum between the boarding and alighting times and corrected by the skipping action) and the travel time between two consecutive stations, respectively. Note that the equations in (1) are function of both the control actions $h_i(k)$ and $Su_i(k)$.

In this paper, we extend this HPC scheme to a dynamic evolutionary multi-objective optimization approach by considering two apparent conflicting objectives (passenger level of service versus penalty due to strategies) for the real-time control of a bus system. Fig. 2 summarizes the HPC-EMO strategy for this problem, leaving the details of the HPC-EMO strategy to Section 3.

The additional variable associated with the demand estimator in Fig. 2 is the number of passengers $\hat{\Gamma}(k+1)$ waiting for a bus i that triggered the event k (bus-stop load). The estimator of demand generates the prediction $\hat{\Gamma}(k+1)$, based on both off and online historical data. The two variables $\hat{B}_i(k)$ and $\hat{\Gamma}(k+1)$ are modeled as disturbances in the dynamic model scheme.

Thus, with the state space model, including state space variables and model outputs as well, the next step is to properly define a predictive objective function in order to make real-time decisions and optimize the dynamic system.

2.2. Objective function

In this case, we will pursue the minimization of expressions (2) and (3) next, which comprises four components oriented to the improvement of the passengers level of service by means of waiting time and penalty due to control actions, which are grouped in two expressions J_1 and J_2. Analytically,

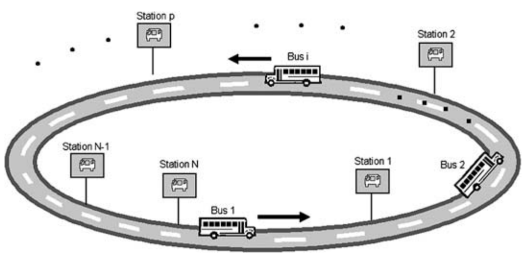


Fig. 1. Public transport system.

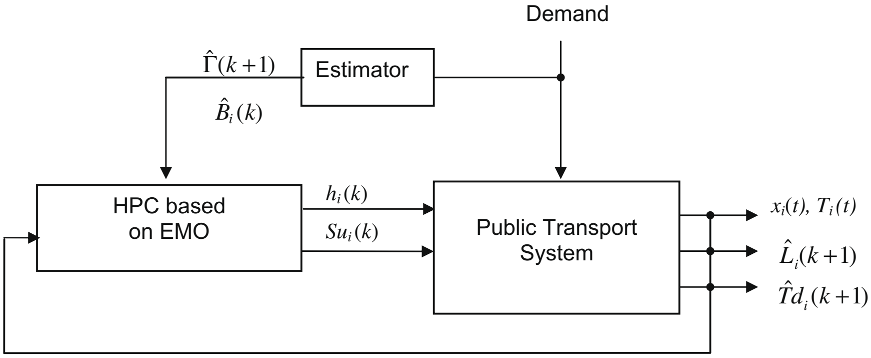


Fig. 2. HPC-EMO strategy for the public transport system.

$$J_1 = \sum_{\ell=1}^{N_p} \left[\theta_1 \cdot \hat{H}_i(k+\ell)\hat{\Gamma}(k+\ell) + \theta_2 \cdot (\hat{H}_i(k+\ell) - \bar{H})^2 \right]\Big|_{i=i(k+\ell-1)} \tag{2}$$

$$J_2 = \sum_{\ell=1}^{N_p} \left[\theta_3 \cdot \hat{L}_i(k+\ell)h_i(k+\ell-1) + \theta_4 \cdot \hat{\Gamma}(k+\ell)\hat{H}_{i+1}(k+\ell+z_{i+1})(1 - Su_i(k+\ell-1)) \right]\Big|_{i=i(k+\ell-1)} \tag{3}$$

with $u(k+\ell-1) = \begin{bmatrix} h_i(k+\ell-1) \\ Su_i(k+\ell-1) \end{bmatrix}$ where bus i triggers event $k+\ell-1$. N_p is the prediction horizon and b is the number of buses in the fleet.

Note that $i = i(k+\ell-1) \in \{1, \ldots, b\}$, if we consider that the future event $k+\ell-1$ is triggered by one bus $i(k+\ell-1)$ arriving to a specific station downstream. In expressions (2) and (3), θ_j are weighting parameters, and have to be tuned depending on the specific problem to be treated and on the physical interpretation of the different components as well. Moreover, these parameters allow the modeler to give different importance to the specific terms of J_1 and J_2.

Note that expressions (2) and (3) depend on the predicted headway between consecutive buses. By using the prediction of the departure time as detailed in (1), it is possible to predict the headway $\hat{H}_i(k+\ell)$ of bus i that triggers the event $k+\ell-1$, with respect to its precedent bus $i-1$ when it reaches the same stop, which corresponds to event $k+\ell-z_{i-1}$. Analytically:

$$\hat{H}_i(k+\ell) = \hat{T}d_i(k+\ell) - \hat{T}d_{i-1}(k+\ell-z_{i-1}) \tag{4}$$

where $\hat{T}d_i(k+\ell)$ is associated with the bus i that triggers the event $k+\ell-1$, and $\hat{T}d_{i-1}(k+\ell-z_{i-1})$ represents the predicted departure time of precedent bus $i-1$ that triggers the event $k+\ell-z_{i-1}-1$, at the same stop. The variable z_{i-1}represents the number of events between the arrival of the precedent bus $i-1$ and the bus i, both reaching the same stop.

The last term of expression (3) includes the headway $\hat{H}_{i+1}(k+\ell+z_{i+1})$ associated with the next bus $i+1$ that trigger the event $k+\ell+z_{i+1}-1$ and is calculated with respect to the bus i that trigger the event $k+\ell-1$, both buses reaching the same stop. Analogously, the variable z_{i+1} represents the number of events between the arrival of the bus i and the next bus $i+1$ at the same stop.

In expression (2), $\bar{H}$ corresponds to the desired headway (set-point) designed for servicing the system demand during a certain time period. Normally, the desired headway $\bar{H}$ is related to the design frequency that directly depends on the segment loads, and can be determined as the minimum required for moving the passengers on the most loaded segment along the bus route. The first term in (2) quantifies the total passenger waiting time at stops, and depends on the predicted headway along with the bus-stop load, which at the same time quantifies the level of service. The second term captures the regularization of bus headways, and that intends to maintain the headway as close as possible to the desired headway. The first term in (3) measures the delay associated with passengers on-board a vehicle when they are held at a control station due to the application of the holding strategy. Finally, the second component in (3) is the extra waiting time of passengers whose station is skipped by an expressed vehicle, associated with the station skipping strategy.

In the next section we describe the solution algorithm implemented in order to dynamically solve the hybrid predictive control problem under the evolutionary multi-objective optimization approach, considering the minimization of both (2) and (3), and complemented by the state space model and the demand prediction algorithm.

3. Hybrid predictive control based on evolutionary multi-objective optimization (hpc-emo) solution algorithms

3.1. Description of HPC-EMO strategy

The HPC-EMO strategy is a generalization of HPC, in which the control action is selected based on a criterion which takes solutions from the Pareto region considering the following multi-objective problem:

$$\underset{\{u(k),u(k+1),\ldots,u(k+N_p-1)\}}{Min}\{J_1,J_2\} \tag{5}$$

with J_1 and J_2 corresponding to the defined objective functions in (2) and (3). Note that this scheme does not need to define an arbitrary weighting factor (parameter λ) as the conventional hybrid predictive control does (say, to minimize $\lambda J_1 + (1-\lambda)J_2$). The solution of the HPC-EMO problem at every event corresponds to a set of control sequences which form the optimal Pareto set. Considering that $u^r = \{u^r(k),\ldots,u^r(k+N_p\text{-}1)\}$ is a feasible control action sequence, the following definitions associated with a multi-objective optimization problem can be stated:

A solution u^r is said to Pareto-dominate another solution u^m if and only if:

$$(J_1(u^r) \leqslant J_1(u^m) \quad \text{and} \quad J_2(u^r) < J_2(u^m)) \quad \text{or} \quad (J_2(u^r) \leqslant J_2(u^m) \quad \text{and} \quad J_1(u^r) < J_1(u^m))$$

A solution u^r is said to be Pareto optimal if and only if there is not another solution u^m which dominates it according to the previous criterion.

The optimal Pareto set P_s contains all the Pareto optimal solutions.

The set of all the values that the objective functions take for each solution P_s is known as the optimal Pareto front P_F and it is defined as:

$$P_F = \{(J_1(u^r), J_2(u^r)) : u^r \in P_S\}$$

In this case, as the control sequences are integer and also defined within a feasible finite set, the resulting optimal Pareto front corresponds to a set with a finite number of elements. From the Pareto front solutions for the dynamic HPC problem, it is necessary to select only one control sequence $u^r = \{u^r(k),\ldots,u^r(k+N_p-1)\}$ and from that, apply the control action $u^r(k)$ to the system according to the receding horizon concept. A relevant application of this approach in the controller's dispatching decision is the definition of criteria to select the best control action at each event under the HPC-EMO approach. In this application, a criterion related to the importance given to both components (J_1 and J_2) is used in the selection of the sequence and the immediate application of the control $u^r(k)$ on the dynamic system. The idea of this application is to utilize a criterion oriented to reduce the waiting time of passengers at bus stops (J_1) by varying the intensity of the control actions applied to the system, which normally annoy the users (J_2).

In order to efficiently find the dynamic set of solutions, we propose the use of genetic algorithms (GA) for each dynamic decision (Man et al., 1998). The better the performance of the algorithm, the better the quality of the resulting set of solutions, which theoretically are pseudo-optimal since GA is not an exact optimization method. However, we claim that our implementation of GA is quite simple, efficient and consequently, it constructs a pseudo-optimal Pareto front very close to the optimal one.

Next, the GA implementation for this particular problem is explained in detail.

3.2. Genetic algorithms for the HPC-EMO problem

Genetic algorithms (GA) are used to solve the stated HPC-EMO problem. In GA, a potential solution of the genetic algorithm is called individual. The individual can be represented by a set of parameters related to the genes of a chromosome and can be described in a binary or integer form. The individual represents a possible control-action sequence $\{u(k),\ldots,u(k+N_p-1)\}$, where each element is a gene, and the individual length corresponds to the control horizon N_p.

Using genetic evolution, the fittest chromosome is selected to assure the best offspring. The best parent genes are selected, mixed and recombined for the production of an offspring in the next generation. For the recombination of genetic population, two fundamental operators are used: crossover and mutation. For the crossover mechanism, the portions of two chromosomes are exchanged with a certain probability in order to produce the offspring. The mutation operator alters each portion randomly with a certain probability.

At each stage of the algorithm, to find the Pareto set the best individuals will be those who belong to the best Pareto set found until the current iteration. As mentioned earlier, the GA in HPC-EMO provides sub-optimal Pareto fronts, but very close to the global optimum. The most important computational effort of applying this algorithm is in computing the predictions, which are recursively, calculated using the model and the control action given by the individual. The tuning GA parameters are the number of individuals, number of generations, crossover probability, mutation probability and stop criterion (Man et al., 1998).

In our application, there are two manipulated variables: holding action and station skipping. Therefore, the following states of the manipulated variables are defined:

$$u(k) = \begin{bmatrix} h_i(k) \\ Su_i(k) \end{bmatrix} \in \{U^1, U^2, \ldots, U^i, \ldots, U^Q\}$$

where U^i corresponds to one of the Q specific control actions.

Moreover, the following constraints for the control actions should be satisfied:

- If the passenger needs to get off, the bus should stop, and therefore station skipping action cannot be applied.
- The holding action is defined for some specified bus stops.

The complete procedure for the GA applied to this EMO control problem is as follows:

1. Set the iteration counter to $i = 1$, and initialize a random population of n individuals, i.e., create n random integer feasible solutions of the manipulated variable sequence. As the control horizon is N_p, there are Q^{N_p} possible individuals. Not all individuals are feasible because of the constraints explained above. The size of the population is n individuals per generation.

$$\text{Population } i \iff \begin{pmatrix} \text{Individual 1} \\ \text{Individual 2} \\ \vdots \\ \text{Individual } n \end{pmatrix}$$

In general, individual j means that the vector of the future control action is:

$$\text{Individual } j = [u^j(k)u^j(k+1),\ldots,u^j(k+N_p-1)]^T_{Npx1}$$

2. For every individual, evaluate J_1 and J_2 corresponding to the defined objective functions in (2) and (3). Then, obtain the fitness function of all individual in the population. In fact, when considering individuals belonging to the best pseudo-optimal Pareto set, a fitness function equal to 0.9 will be set; otherwise 0.1 will be used, in order to maintain the solution diversity. If the individual is not feasible, penalize it (pro-life strategy).
3. Select random parents from the population i (different vectors of the future control actions).
4. Generate a random number between 0 and 1. If the number is less than the probability p_c, choose an integer $0 < c_p < N_p - 1$ (c_p denotes the crossover point) and apply the crossover to the selected individuals in order to generate an offspring. The next scheme describes the crossover operation for two individuals, U^j and U^l, resulting in U^j_{cross} and U^l_{cross}.

$$U^j = \left\{\boxed{u^j(k),u^j(k+1),\ldots,u^j(k+c_p-1)},\boxed{u^j(k+c_p),\ldots,u^j(k+N_p-1)}\right\}$$
$$U^l = \left\{\boxed{u^l(k),u^l(k+1),\ldots,u^l(k+c_p-1)},\boxed{u^l(k+c_p),\ldots,u^l(k+N_p-1)}\right\}$$
$$\Downarrow$$
$$U^j_{cross} = \left\{\boxed{u^l(k),u^l(k+1),\ldots,u^l(k+c_p-1)},\boxed{u^j(k+c_p),\ldots,u^j(k+N_p-1)}\right\}$$
$$U^l_{cross} = \left\{\boxed{u^j(k),u^j(k+1),\ldots,u^j(k+c_p-1)},\boxed{u^l(k+c_p),\ldots,u^l(k+N_p-1)}\right\}$$

5. Generate a random number between 0 and 1. If the number is less than the probability p_m, choose an integer $0 < c_m < N_p - 1$ (c_m denotes the mutation point) and apply the mutation to the selected parent in order to generate an offspring. Select a value $u^j_{mut} \in U$ and replace the value in the c_mth position in the chromosome. The next scheme describes the mutation operation for an individual U^i resulting in U^j_{mut}.

$$U^j = \left\{u^j(k),u^j(k+1),\ldots,u^j(k+c_m-1),\boxed{u^j(k+c_m)},u^j(k+c_m+1),\ldots,u^j(k+N_p-1)\right\}$$
$$\Downarrow$$
$$U^j_{mut} = \left\{u^j(k),u^j(k+1),\ldots,u^j(k+c_m-1),\boxed{u^j_{mut}},u^j(k+c_m+1),\ldots,u^j(k+N_p-1)\right\}$$

6. Evaluate the objective functions J_1 and J_2 of all the individuals of the offspring population. Then obtain the fitness of each individual by following the fitness definition described in step 2. If the individual is unfeasible, penalize its corresponding fitness.
7. Select the best individuals according to their fitness. Replace the weakest individuals from the previous generation with the strongest individuals of the new generation.
8. If the tolerance given by the maximum generation number is reached (stopping criteria, i equals the number of generation), then stop. Otherwise, go to step 3. Note that since we are proposing a real-time control strategy, the best stopping algorithm criterion corresponds to the number of generations.

The tuning parameters of the EMO method based on GA are the number of individuals, the number of generations, the crossover probability p_c, the mutation probability p_m and the stopping criteria.

At each stage of the algorithm, to find the pseudo-optimal Pareto set, the best individuals will be those who belong to the best Pareto set found until the current iteration.

From the pseudo-optimal Pareto front, it is necessary to select only one control sequence $u^* = \{u^*(k), \ldots, u^*(k + N_p - 1)\}$ and from that, apply the current control action $u^*(k)$ to the system according to the receding horizon concept. For the selection of this sequence, a criterion related to the importance given to both the user (J_1) and operator (J_2) costs in the final decision is needed, as we show in the experiments conducted and detailed in Section 4 next.

4. Simulation experiments

The proposed strategy is applied to a bus corridor of 4 (km) comprising 10 stations evenly distributed over the bus route with a fleet of six buses circulating. For operational reasons, we assume that holding can be applied only to a subset of stations, which must be not consecutive. In this experiment, the holding control action is applied at the bus stops 1, 5 and 10, while the skipping actions can be applied in all stations. The simulation assumes uncertain demand dynamically arriving at stations by following a Poisson process with different demand rates, differentiated by station and period. The total simulation period was 2 h, with a warm-up time (discarded for statistics) of 15 min at the beginning and at the end of the simulation.

As explained before, we utilize two manipulated variables: holding and station skipping. For simplicity, in this application the holding will take only four possible values: 0, 30, 60 and 90 (s) at the selected bus stops. The station skipping is defined with "0" value when the bus skips the stop and "1" otherwise. Both manipulated variables are exclusive of each bus-stop. Thus, when the station skipping is applied, the holding action cannot be applied at the same station. Thus, the following states of the manipulated variables are defined:

$$u(k) = \begin{bmatrix} h_i(k) \\ Su_i(k) \end{bmatrix} \in \left\{ \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 30 \\ 1 \end{bmatrix}, \begin{bmatrix} 60 \\ 1 \end{bmatrix}, \begin{bmatrix} 90 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\}$$

where the first row represents the holding action and the second one represents station skipping. In order to apply GA, the following coding is proposed:

$$U^1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad U^2 = \begin{bmatrix} 30 \\ 1 \end{bmatrix}, \quad U^3 = \begin{bmatrix} 60 \\ 1 \end{bmatrix}, \quad U^4 = \begin{bmatrix} 90 \\ 1 \end{bmatrix}, \quad U^5 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Additionally, in the experiments we considered two different prediction horizons N_p of 2 and 5 steps.

Tables 1 and 2 shows the average waiting time, travel time and total time per passenger over the simulation period, applying HPC-EMO based on GA, for N_p of 2 and 5, respectively. The averages are taken over 17 replications of the experiment, representing 17 different days of operation.

With regard to the different cases summarized in Tables 1 and 2, the open-loop (OL) response (system without control) is first reported. When a new event occurs (i.e. when a bus arrives at a station), the operator must decide what to do next, based on one solution chosen among those available from the dynamic pseudo-optimal Pareto front constructed by the GA. In these experiments, we consider five cases. Case 1 considers a 100% importance to J_1 for each dynamic decision; Case 2 considers an 80% importance to J_1 for each dynamic decision; Case 3 gives equal importance to J_1 and J_2; Case 4 is analogous to Case 1 but in which the 80% is now assigned to J_2. Finally Case 5 is analogous to Case 1 but in which the 100% is now assigned to J_2.

Depending on the case, the solution chosen by the operator to proceed with the operation at each decision instant will be the one that not only belongs to the pseudo-optimal Pareto Front but also is the closest – in terms of euclidean distance – with respect to a virtual point in the (J_1, J_2) space that represents the criterion that define each case. For case *i*, the virtual point has coordinates $(\theta_i \cdot M_1, (1 - \theta_i) \cdot M_2)$, with M_1 and M_2 representing the maximum J_1 and J_2 values obtained among the dynamic pseudo-optimal Pareto set solutions associated with each event. θ_i is the weight (importance) of J_1 in the final decision, normalized between 0 and 1. For example, in Case 3, $\theta_3 = 0.5$.

Table 1
Average and standard deviation of waiting time, travel time and total time per passenger using HPC-EMO for prediction horizon $N_p = 2$.

Cases	Waiting time (min)		Travel time (min)		Total time (min)	
	Mean	Std.	Mean	Std.	Mean	Std.
OL	9.54	0.90	6.57	0.30	16.11	0.94
1	4.60	0.86	6.54	0.29	11.14	0.91
2	4.67	0.80	6.51	0.27	11.18	0.85
3	4.68	0.80	6.56	0.30	11.25	0.86
4	4.78	0.69	6.54	0.29	11.33	0.75
5	4.94	0.82	6.51	0.33	11.45	0.88

Table 2
Average and standard deviation of waiting time, travel time and total time per passenger using HPC-EMO for prediction horizon $N_p = 5$.

Cases	Waiting time (min)		Travel time (min)		Total time (min)	
	Mean	Std.	Mean	Std.	Mean	Std.
OL	9.54	0.90	6.57	0.30	16.11	0.94
1	4.51	0.68	6.52	0.29	11.03	0.74
2	4.59	0.69	6.50	0.27	11.09	0.74
3	4.73	0.76	6.50	0.24	11.24	0.79
4	5.15	0.79	6.58	0.26	11.74	1.10
5	5.10	0.74	6.52	0.29	11.56	0.68

Cases 1 and 5 are the extreme situations, both mono-objective giving 100% importance to either J_1 or J_2. The objective of these two cases is to visualize the trade-off between the two apparent conflicting objectives.

From the reported results, we can realize, first that the HPC strategy outperforms the myopic OL strategy, and second, the HPC-EMO allows the operator to dynamically decide the importance of each term in the proposed objective function.

The first observation is that in all cases the predictive model considerably improves the quality of the solution compared with the OL system. In the best cases, a 20% savings of total time for users is observed when using this HPC strategy in comparison with the OL system. From the results we also observe that the predictive control scheme primordially improves the waiting time of passengers, with almost no benefit in terms of travel time, which means that the objective function in Section 2.2 was not written to really take into account a component summarizing the potential savings in travel time as a relevant issue. Savings in waiting time due to the HPC strategy are significant (around 50% in Case 1), which validates the HPC model proposed, always in the line of improving the regularity of the service (reflected in J_1).

We also appreciate from Tables 1 and 2 that, independent of the case, the reduction in waiting time is considerable with respect to the OL base, which means that (mainly looking at the results for the extreme cases) even though J_1 and J_2 seem to be opposite and adequate for the EMO formulation, in the experiments both are useful to improve the quality of the service in terms of waiting time (regularity of the service). However, the tendency from Cases 1 to 5 shows a slight deterioration of the level of service via waiting time, which should be compensated by an improvement of the level of service of users affected by the control actions, in order to validate the multi-objective framework proposed for this problem.

Standard deviations are all in the same range that seems to be reasonable. The only point that is not following the expected tendency is the average waiting time for Case 5 in Table 2. This small unexpected behavior is most likely due to the uncertainty added to the model when a longer prediction horizon is considered ($N_p = 5$).

In order to visualize the trade-off between the two objectives, we have to somehow measure the impact on the passengers affected by the strategies. Thus, in Table 3 we present two indicators *PTH* and *PTS*, associated with holding and station skipping respectively, defined as follows:

$$PTH = P_{H30} \cdot N_{H30} \cdot 30 + P_{H60} \cdot N_{H60} \cdot 60 + P_{H90} \cdot N_{H90} \cdot 90$$
$$PTS = P_S \cdot N_S$$

where P_{H30}, average number of passengers held during 30 (s) at any station; P_{H30}, average number of passengers held during 60 (s) at any station; P_{H30}, average number of passengers held during 90 (s) at any station; N_{H30}, number of holding actions of 30 (s); N_{H60}, number of holding actions of 60 (s); N_{H90}, number of holding actions of 90 (s); P_S, average number of passengers affected by a skipping action; N_S, number of skipping actions.

These indicators represent an estimator of the total passenger-time spent by those passengers affected by holding in the former case (*PTH*) and an estimator of the total number of passengers affected by a skipping in the latter (*PTS*), both computed considering the whole simulation period. They are obtained by counting holding and skipping actions during the valid simulation period. From the 17 days of observation, averages and standard deviations are obtained for all the statistics required to compute *PTH* and *PTS*. In Appendix A we detail the average and standard deviation of the aforementioned statistics, for each case and prediction horizon.

In Table 3, we report *PTH* and *PTS* for all studied cases, and for $N_p = 2$ and 5.

Table 3
PTH and *PTS* indicators.

Cases	$N_p = 2$		$N_p = 5$	
	PTH	*PTS*	*PTH*	*PTS*
1	4506.05	8743.24	5978.77	9700.93
2	1431.69	8764.92	3889.64	9599.74
3	2835.56	7272.10	2764.57	6390.44
4	1715.75	6245.87	2883.84	1061.24
5	1283.54	6386.14	1567.34	544.57

Results are quite reasonable. The impact of the different weight given to the two objectives is consistent with the definition of the different cases in almost all cases. First, we can note that the behavior of station skipping seems to follow better the tendency expected across the different cases (decreasing from cases 1 to 5), except *PTS* for $N_p = 2$, cases 4 and 5. The other indicator *PTH* also follows the expected tendency. The only point really strange is *PTH* for case 3, $N_p = 2$. These apparent not

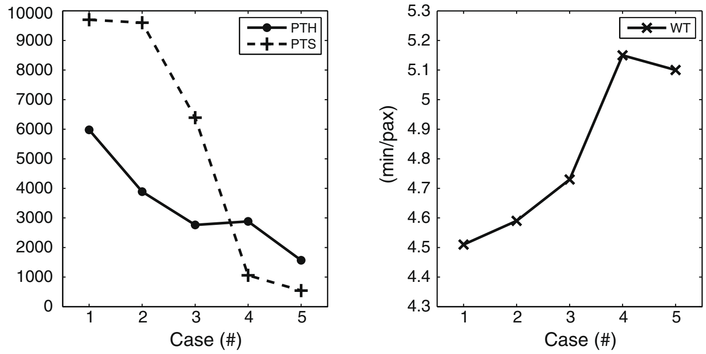
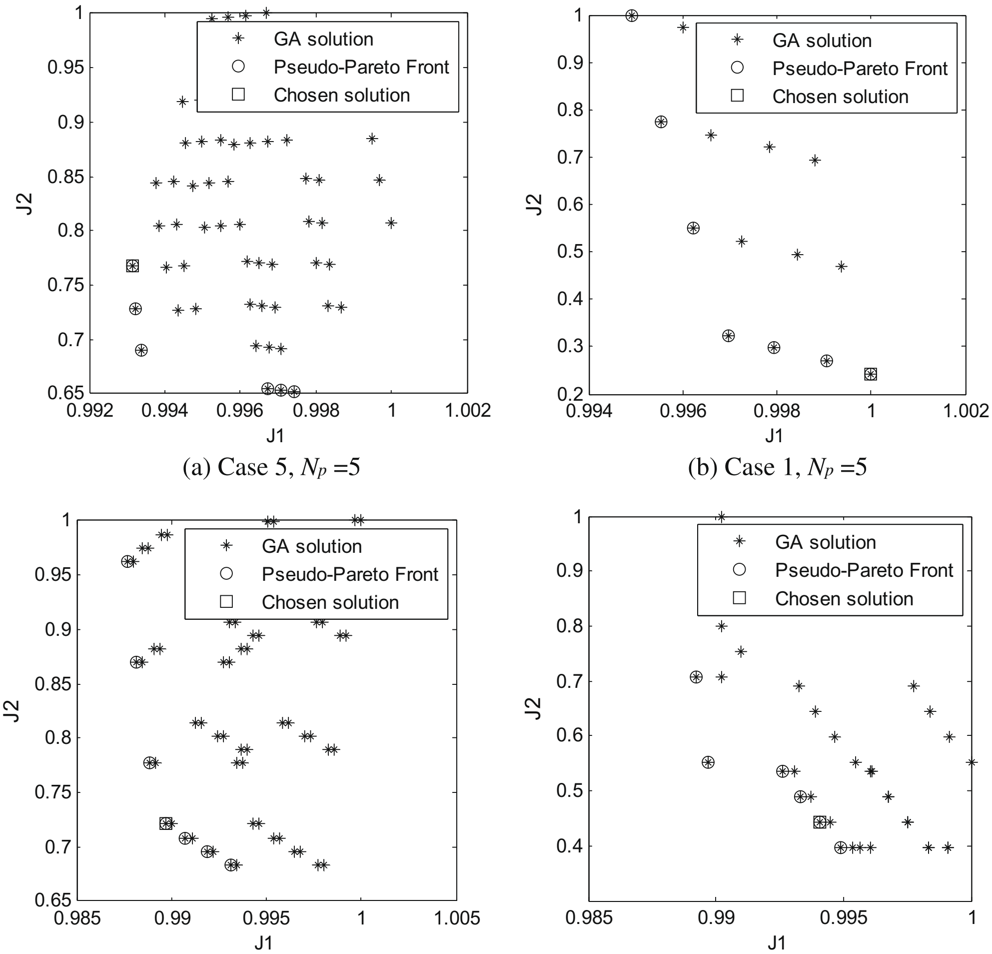


Fig. 3. Trade-off between the two objectives J_1 and J_2.



(a) Case 5, N_p =5

(b) Case 1, N_p =5

(c) Case 3, N_p=5

(d) Case 2, N_p =5

Fig. 4. Illustrative pseudo-optimal Pareto fronts generated with HPC-EMO.

reasonable points can be explained by the premise we pointed our before: even though J_1 and J_2 are opposite, they have certain degree of overlapping, since an objective function influenced only by the penalty of the strategies result in a big improvement in regularization and waiting time with respect to the OL scenario, almost comparable with that obtained by using an objective function oriented only to the minimization of waiting times and regularization of headway.

In Fig. 3 we depict the resulting trade-off between both objectives, through the average waiting time per passenger (in min/pax), *PTH* (in pax-sec) and *PTS* (in pax) across all cases, for $N_p = 5$.

The graphs in Fig. 3 nicely shows the relevance of considering the spectrum of solutions provided by the dynamic HPC-EMO scheme in this case, through the opposite tendency of the indicators reflecting the impact of each objective. Having the dynamic pseudo-optimal Pareto front available at each decision time can significantly affect the final action applied by the operator, which finally depends on the final objective behind the operation of a public transport system.

In Fig. 4 next, a set the explored solutions is depicted for the three cases (J_1 vs. J_2), at certain event k properly chosen for illustration purposes, with $N_p = 5$. The points belonging to the pseudo-optimal Pareto front are highlighted with circles, indicating with a square the solution finally chosen by the operator in the simulation. From the figures, curves look like reasonable Pareto sets in all cases.

During the simulation, at certain events, we obtained pseudo-Pareto fronts comprising just one point. In such situations, we removed such a point and considered the next pseudo-Pareto front in order to generate enough point to have a reasonable set for deciding the action to follow according to the virtual point method explained above.

5. Synthesis and conclusions

In this paper we have shown a Hybrid Predictive Control strategy based on Evolutionary Multi-objective Optimization to dynamically optimize the performance of a public transport system along a linear corridor with uncertain demand at bus stops (stations). The optimization is conducted by applying holding and station skipping. The proposed HPC-EMO strategy was formulated under a discrete event simulation environment, and is developed in order to optimize real-time control operations of the bus system considering the different aspects of the multidimensionality of the embedded problem. The dynamic formulation of the system requires the demand forecast based on offline as well as online data.

The multi-objective was defined in terms of two objectives: waiting time minimization on one side, and impact of the strategies on the other. This flexibility in the formulation allows the controller to accommodate his (her) actions to different service policies, depending on the case. In this formulation, the term J_2 controls the possible penalization of the impact on users of applying the different strategies, reflected by the extra travel and waiting time due to buses stopped at stops (holding) and passengers waiting two intervals when stations are skipped. J_1 on the other hand, helps the operator regularize headways around a predefined desired headway $\bar{H}$ that could eventually change if medium and long term demand modifications are observed. From the conducted experiments, we found that the two objectives are opposite (as finally summarized in Fig. 4) but with a certain degree of overlapping, in the sense that in all cases both objectives significantly improve the level of service with respect to the OL scenario by regularizing the headways. Therefore, even though the objectives have certain similarities, on average they show an observed trade-off anyway, which validates the HPC-EMO methodology for the studied system and based on the proposed objective function components.

In addition, one major contribution of the dynamic EMO approach together with the GA solution method is that they provide dynamic pseudo-optimal Pareto fronts allowing the operator (or the planner) to make online decisions based on a variety of options. Therefore, they are able to decide at each event-time what is better for the system depending on a specific policy or other factors, but with a long range of not dominated solutions to make a better choice and improve the operational scheme.

In further applications, other objective functions can be tested, for example by adding a component directly related to operational costs or additional fleet necessary to deal with some unexpected situations. Moreover, we recommend developing detailed sensitivity analyses with respect to the multi-objective criteria, prediction horizon, weight parameters, etc., in order to have better criteria to define operational policies. Other control actions can be also tested (injection of vehicles, signal priority for buses, etc.) under a HPC-EMO scheme, properly identifying the different dimensions that could result in opposite objectives.

Acknowledgments

This research was partially financed by the ACT-32 Project ‘‘Real Time Intelligent Control for Integrated Transit Systems”, Fondecyt Chile Grant 1061261, and the Millennium Institute ‘‘Complex Engineering Systems”.

Appendix A

In Tables A1 and A2 we report the number of passenger (Pax) affected by the holding strategy in the five different options defined in the formulation (30 s, 60 s, 90 s) and the number of times those passengers were affected by such a strategy, and that for the same cases reported in the text, and for the same two and five prediction horizons as well. Besides, the last two columns in Tables A1 and A2 show the same statistics, but associated with the station skipping strategy.

Table A1
Passengers affected by the holding and station skipping actions using HPC-EMO. Prediction horizon $N_p = 2$.

Cases		Holding 30 (s)		Holding 60 (s)		Holding 90 (s)		Station skipping	
		Pax	Number	Pax	Number	Pax	Number	Pax	Number
1	Mean	9.47	1.70	1.52	2	3.05	13.94	119.29	73.29
	Std.	8.19	1.57	2.42	1.41	3.79	2.10	92.26	12.71
2	Mean	8.00	2.52	0.12	1.70	0.89	10.05	111.53	78.58
	Std.	6.74	1.66	0.43	1.64	0.98	1.88	64.12	14.44
3	Mean	12.00	6.35	1.11	2.47	0.45	9.29	92.88	78.29
	Std.	9.65	2.17	2.78	1.46	0.34	2.33	56.68	9.36
4	Mean	6.64	2.41	2.70	6.47	0.23	8.70	93.88	66.52
	Std.	8.01	1.54	2.28	2.06	0.75	2.46	74.66	15.87
5	Mean	6.29	2.70	0.13	1.64	0.88	9.52	101.18	73.00
	Std.	5.15	1.44	0.13	1.36	0.85	1.97	57.98	15.56

Table A2
Passengers affected by the holding and station skipping actions using HPC-EMO. Prediction horizon $N_p = 5$.

Cases		Holding 30 (s)		Holding 60 (s)		Holding 90 (s)		Station skipping	
		Pax	Number	Pax	Number	Pax	Number	Pax	Number
1	Mean	10.76	3.82	0.941	3.11	4.47	11.35	97.35	99.64
	Std.	9.33	1.33	2.56	1.45	4.40	2.52	36.91	9.00
2	Mean	10.64	3.88	0.74	3.11	3.64	7.64	159.06	60.35
	Std.	8.63	2.11	0.86	1.53	0.74	2.28	67.83	9.80
3	Mean	9.05	7.52	0.93	3.41	0.96	6.05	110.18	58.00
	Std.	6.49	2.06	0.66	1.73	0.78	2.07	57.60	11.97
4	Mean	8.82	4.05	4.29	6.35	0.10	7.58	27.29	38.88
	Std.	6.66	1.56	7.50	1.69	0.12	1.73	22.47	8.12
5	Mean	10.11	2.88	0.12	2.82	0.87	8.52	50.58	10.76
	Std.	6.09	1.40	0.46	1.46	0.74	1.73	36.94	6.30

References

Adamski, A., 1996. Flexible dispatching control tools in public transport, In: Bianco, L., Toth, P. (Eds.), Advanced Methods in Transportation Analysis, pp. 481–506.

Adamski, A., Turnau, A., 1998. Simulation support tool for real-time dispatching control in public transport. Transportation Research – Part A 32 (2), 73–87.

Alvarez, J., Cruz, C., 1998. Multi-objective dynamic optimization of discrete-time systems. In: Proceedings of the 37th IEEE Conference on Decision & Control, Tampa, Florida USA, December 1998.

Eberlein, X.-J., Wilson, N., Bernstein, D., 2001. The holding problem with real-time information available. Transportation Science 35 (1), 1–18.

Eberlein, X.-J., Wilson, N., Bernstein, D., 1999. Modeling real-time control strategies in public transit operations. In: Wilson, N. (Ed.), Computer-aided Transit Scheduling, Lecture Notes in Economics and Mathematical Systems, vol. 471. Springer-Verlag, Heidelberg, pp. 325–346.

Eberlein, X.-J., 1995. Real-time Control Strategies in Transit Operations: Models and Analysis. PhD Thesis, Massachusetts Institute of Technology, Cambridge, MA.

Farina, N., Deb, K., Amato, P., 2004. Dynamic multi-objective optimization problems: test cases, approximations, and applications. IEEE Transactions on Evolutionary Computation 8 (5), 425.

Fu, L., Liu, Q., 2003. A real-time optimization model for dynamic scheduling of transit operations. In: 82nd Annual Meeting of the Transportation Research Board (CD-ROM), Washington, DC.

Furth, P., 1986. Zonal route design for transit corridors. Transportation Science 20 (1), 1–12.

Hajri-Gabouj, S., 2003. A fuzzy genetic multi-objective optimization algorithm for multilevel generalized assignment problem. IEEE Transactions on Systems, Man and Cybernetics 33 (2), 214–224.

Hegyi, A., 2004. Model Predictive Control for Integrating Traffic Control Measures. PhD Thesis, Delft University of Technology, Delft Center for Systems and Control.

Hickman, M., 2001. An analytic stochastic model for the transit vehicle holding problem. Transportation Science 35 (3), 215–237.

Hu, Z., Chan, C., Huang, G., 2007. Multi-objective optimization for process control of the in-situ bioremediation system under uncertainty. Engineering Applications of Artificial Intelligence 20, 225–237.

Jordan, W., Turnquist, M., 1979. Zone scheduling of bus routes to improve service reliability. Transportation Science 13 (3), 242–268.

Kerrigan, E., Maciejowski, J., 2002. Designing model predictive controllers with prioritised constraints and objectives. In: Proceedings of 2002 IEEE International Symposium on Computer Aided Control System Design, Glasgow, Scotland, September 18–20, pp. 33–38.

Kerrigan, E., Bemporad, A., Mignone, D., Morari, M., Maciejowski, J., 2000. Multi-objective priorisation and reconfiguration for the control of constrained hybrid systems. In: Proceedings of the American Control Conference, Chicago Illinois, June 2000, pp. 1694–1698.

Laabidi, K., Bouani, F., 2004. Genetic algorithms for multi-objective predictive control. In: Proceedings of the 2004 IEEE International Symposium on Control, Communications and Signal Processing, Hammamet, Tunisia, 21–24 March, 2004, pp. 149–152.

Man, K., Tang, K., Kwong, S., 1998. Genetic Algorithms, Concepts and Designs. Springer-Verlag.

Núñez, A., Riquelme, M., Sáez, D., Cortés, C.E., 2007. Control predictivo híbrido para el problema de ruteo dinámico de vehículos basado en optimización multiobjetivo evolucionaria (EMO) (Spanish). XIII Congreso Chileno de Ingeniería de Transporte, Santiago, Chile, October 22–26, 2007.

Núñez, A., Sáez, D., Cortés, C.E., 2008. Hybrid predictive control for the vehicle dynamic routing problem based on evolutionary multi-objective optimization (EMO). In: 17th IFAC World Congress. Seoul, Korea, July 6–11, 2008.
Nunez-Reyes, A., Scheffer-Dutra, C., Bordons, C., 2002. Comparison of different predictive controllers with multi-objective optimization. Application to an olive oil mill. In: Proceedings of the 2002 International Conference on Control Applications, Glasgow, Scotland, September 18–20, 2002, pp. 1242–1247.
Sáez, D., Cortés, C.E., Riquelme, M., Núñez, A., Tirachini, A., Sáez, E., 2007. Hybrid predictive control strategy for a public transport system with uncertain demand. TRISTAN IV Congress, Phuket Island, Thailand, June 10–15, 2007.
Sáez, D., Cortés, C.E., Milla, F., Núñez, A., Riquelme, M., Tirachini, A., 2009. Hybrid predictive control strategy for a public transport system with uncertain demand. Under review in *Transportmetrica*.
Subbu, R., Bonissone, P., Eklund, N., Weizhong, Y., Iyer, N., Feng, X., Rasik, S., 2006. Management of complex dynamic systems based on model-predictive multi-objective optimization. In: Proceedings of 2006 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, La Coruña, Spain, July 12–14, pp. 64–69.
Sun, A., Hickman, M., 2004. The holding problem at multiple holding stations. In: Proceedings of 9th International Conference on Computer-aided Scheduling of Public Transport (CASPT) San Diego, California, EE.UU, August 9–11, 2004.
Sun, A., Hickman, M., 2005. The real-time stop-skipping problem. Journal of Intelligent Transportation Systems 9 (2), 91–109.
Tan, K., Cheong, C., Goh, C., 2007. Solving multi-objective vehicle routing problem with demand via evolutionary computation. European Journal of Operational Research 177, 813–839.
Zambrano, D., Camacho, E.F., 2002. Application of MPC with multiple objective for a solar refrigeration plant. In: Proceedings of the 2002 International Conference on Control Applications, September 18–20, Glasgow, Scotland, 1230–1235.