INTERNATIONAL JOURNAL OF ADAPTIVE CONTROL AND SIGNAL PROCESSING
Int. J. Adapt. Control Signal Process. 2008; **22**:103–123
Published online 15 October 2007 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/acs.1005

Hybrid adaptive predictive control for a dynamic pickup and delivery problem including traffic congestion

Cristián E. Cortés[1,*,†], Alfredo Núñez[2] and Doris Sáez[2]

[1] *Civil Engineering Department, Universidad de Chile, Blanco Encalada 2002, Santiago, Chile*
[2] *Electrical Engineering Department, Universidad de Chile, Av. Tupper 2007, Santiago, Chile*

SUMMARY

This paper presents a hybrid adaptive predictive control approach to incorporate future information regarding unknown demand and expected traffic conditions, in the context of a dynamic pickup and delivery problem with fixed fleet size. As the routing problem is dynamic, several stochastic effects have to be considered within the analytical expression of the dispatcher assignment decision objective function. This paper is focused on two issues: one is the extra cost associated with potential rerouting arising from unknown requests in the future, and the other is the potential uncertainty in travel time coming from non-recurrent traffic congestion from unexpected incidents.

These effects are incorporated explicitly in the objective function of the hybrid predictive controller. In fact, the proposed predictive control strategy is based on a multivariable model that includes both discrete/integer and continuous variables. The vehicle load and the sequence of stops correspond to the discrete/integer variable, adding the vehicle position as an indicator of the traffic congestion conditions. The strategy is analyzed under two scenarios. The first one considers a predictable congestion obtained using historical data (off-line method) requiring a predictive model of velocities distributed over zones. The second scenario that accepts unpredictable congestion events generates a more complex problem that is managed by using both fault detection and isolation and fuzzy fault-tolerant control approaches. Results validating these approaches are presented through a simulated numerical example. Copyright © 2007 John Wiley & Sons, Ltd.

Received 21 November 2006; Revised 1 August 2007; Accepted 8 August 2007

KEY WORDS: dynamic pickup and delivery problem; traffic congestion; predictive control; fault detection and isolation

[*] Correspondence to: Cristián E. Cortés, Civil Engineering Department, Universidad de Chile, Blanco Encalada 2002, Santiago, Chile.
[†] E-mail: ccortes@ing.uchile.cl

Contract/grant sponsor: Fondecyt, Chile; contract/grant numbers: 1061156, 1061261
Contract/grant sponsor: Millennium Institute ‘Complex Engineering Systems’
Contract/grant sponsor: ACT-32

1. INTRODUCTION AND BACKGROUND

The dynamic pickup and delivery problem (DPDP) can be formulated as a set of transportation requests (characterized by pickup and delivery loads, time windows and spatial coordinates) served by a fleet of vehicles located initially at several depots [1, 2]. The dynamic dimension appears when a subset of the requests is not known in advance and most dispatch decisions have to be taken in real time. The DPDP is of great interest for practitioners, mainly due to the fast growth in communication and information technologies, as well as the current interest in real-time dispatching and routing.

In the literature, dynamic vehicle routing problems (dynamic VRPs) are formulated assuming that inputs may change or have to be updated during the execution of the solution algorithm. Within this family of problems, the DPDP has been designed to solve the dynamic dial-a-ride problem (DDRP), which has been intensely studied in the last 20 years [3–6]. The final output of such a problem is a set of routes for all vehicles, which dynamically change over time. With regard to real applications Madsen *et al.* [7] adapt the insertion heuristics by Jaw *et al.* [8] and solve a real-life problem for moving elderly and handicapped people in Copenhagen, while Dial [9] proposes a modern approach to many-to-few dial-a-ride transit operation autonomous dial-a-ride transit (ADART), currently implemented in Corpus Christi, TX, U.S.A.

With regard to other interesting dynamic VRPs, let us start mentioning dynamic travelling salesman problem (DTSP) introduced by Psaraftis [4]. This work motivates the Dynamic Traveling Repairman Problem (DTRP), defined by Bertsimas and Van Ryzin [10] and next extended in Bertsimas and Howell [11]. Lately Swihart and Papastavrou [12] and Thomas and White [13] formulate and solve two variants of the DTRP. Kleywegt and Papastavrou [6, 14] and Papastavrou *et al.* [15] study a problem called the Dynamic and Stochastic Knapsak Problem (DSKP), in which demands for a given resource occur according to some stochastic process. Larsen [16] develops a nice review of the different dynamic problems.

There are several key aspects for improving the efficiency of a real implementation behind a DPDP instance. Fundamentally, it is crucial to utilize a correct definition of a decision objective function for dispatching, including total travel and waiting times for users as well as a performance measure for vehicles (proxy of operational costs). When the problem is dynamic, a proper objective function must consider prediction of both future demand and expected waiting and travel times experienced by customers in the system due to potential re-routing decisions decided in the future. This last issue has been mostly underestimated in the dynamic vehicle routing literature, restricting the development of algorithms to myopic models (current decisions not affected by unknown future demand events).

Nevertheless, there exists some recent literature in the field of vehicle routing and dispatching (of both freight and passengers) trying to exploit information about future events to improve decision-making. Some examples are the works of Ichoua *et al.* [17] and Spivey and Powell [18]. For such schemes, the reported solution approaches are diverse, with formulations based on dynamic network models as in Powell [19], dynamic and stochastic programming schemes developed by Godfrey and Powell [20] and Topaloglu and Powell [21], respectively. Besides, Cortés [22] realizes that the problem could be modeled under a hybrid adaptive predictive control (HAPC) scheme, considering that potential re-routing of vehicles could affect the current dispatch decisions, through the extra cost of inserting real-time service requests into predefined vehicle routes while vehicles are moving. Cortés *et al.* [23] write a formal formulation of the DPDP as an HAPC, by stating the state-space variables and models. Based on such an approach, Sáez *et al.* [24]

DOI: 10.1002/acs

develop a family of solution algorithms based on artificial intelligence for solving real-size instances.

These approaches incorporate an important source of stochasticity in real-time routing decisions, which is the extra cost over the system caused by unexpected vehicle re-routing coming from serving future unknown requests arising in real time. However, there is another relevant source of stochasticity that could affect dynamic routing decisions, mainly in the context of urban transport systems. That is, the uncertainty behind the traffic network conditions, interfering the normal operation of the routed vehicles. This new source of uncertainty has not been treated extensively in the literature associated with dynamic routing problems, mainly because of the computational complexity arising from the resulting formulations. Nevertheless, lately we found some interesting research effort for adding traffic congestion into dynamic as well as probabilistic/stochastic VRP that are worth to mention.

Berman and Simchi-Levi [25] consider a variant of the probabilistic traveling salesman problem (PTSP), including a random subset of customers requiring service and random travel times as well. With regard to stochastic VRP, Kao [26], Sniedovich [27], and Carraway *et al.* [28] solve the stochastic TSP, considering arcs having independent and normally distributed travel times. Laporte *et al.* [29] study the stochastic VRP with stochastic travel as well as service times. They solve instances on networks with 10–20 nodes and 2–5 scenarios. Lambert *et al.* [30] solve an optimization of collection routes through bank branches in a network with stochastic travel times. Keyton and Morton [31] also solve stochastic VRPs on a network with random travel and service times, by using a branch-and-cut scheme within a Monte Carlo sampling-based procedure. Most of the work described above is based on static models that do not reoptimize routes after realizing the random parameters.

With regard to VRPs including traffic conditions, we can mention the work by Hill and Benton [32], defining the nodes of the road network with time-dependent piecewise constant speeds and compute the travel time on a link from the average speed of the incident nodes. Malandraki and Daskin [33] formulate a mixed integer program for the VRP with time windows (VRPTW) and piecewise constant travel times, which is solved via heuristic methods.

To our knowledge, there are just a couple of examples of dynamic VRPs, in which routes can be modified in real-time from updated information of travel time on links and some prediction of the system based on updated data [34, 35]. The former considers a dynamic routing system that dispatches a fleet of vehicles according to customer requests asking for service randomly over a planning period. The authors propose a solution to such a problem, relying on online travel time information from a traffic management center, formulating three routing procedures for event-based dispatching. On the other hand, the latter examines the value of real-time traffic information to optimal vehicle routing in a non-stationary stochastic network. The authors develop optimal routing policies under time-varying traffic flows based on a Markov decision process formulation.

In this paper, an HAPC formulation for a DPDP is proposed, which combines both sources of uncertainty when taking real-time vehicle routing decisions. On the one hand, the formulation considers uncertainty from possible future demand influencing routes of current customers, which follows the original scheme proposed by Sáez *et al.* [24]. Apart from that, the uncertainty regarding traffic congestion conditions is also added into the scheme, under the premise that this disturbance should also affect the preplanned vehicle routes dynamically based on traffic information around them. The proposed approach allows modeling not only predictable congestion conditions but also unpredictable situations, such as incidents occurring unexpectedly at any location on the traffic network.

DOI: 10.1002/acs

In the following section, we formally develop the HAPC formulation for the DPDP, including prediction not only on demand but also on traffic conditions. The state-space equations are written, and later a proper objective function for this problem is proposed. Then it is shown how the formulation and its solution work for several simulation scenarios, under several demand and speed conditions. The unpredictable events are detected with real-time data observed by the fleet, by using a fault detection and isolation (FDI) scheme. Then, the model is corrected in order to avoid the affected zones by the incident, through a fuzzy fault-tolerant control (FFTC) approach.

2. ANALYTICAL FORMULATION: HAPC APPROACH

In this section, the formulation of the DPDP under an HAPC scheme as proposed by Sáez *et al.* [24] is extended to capture the network traffic conditions and provide a more realistic representation of the transport system uncertainty. For doing that, it is necessary to define a set of state-space variables, which is used in order to characterize the key elements of the system at certain instant and are needed to provide a formal predictive control formulation to the DPDP problem.

In this case, three state-space variables are considered: departure time, vehicle load at stops and position of the vehicles. The last variable (position of vehicles) is added in order to incorporate the traffic conditions as a function of the network speed distribution. Regarding the objective function, it includes both user and operational costs. The operational cost is approximated by the total vehicle time traveled and the user cost considers both waiting and travel time. The fleet size is assumed known, and the cost function does not include time windows on either pickup or delivery points.

Next, in Section 2.1 the extended dynamic model for representing the DPDP is formulated. Then, in Section 2.2, the corresponding objective function formulation is established, completing the presentation with a description of the optimization method in Section 2.3.

2.1. Dynamic model formulation

Let us assume an influence urban area and a fleet of homogenous vehicles of size F. The fleet is currently in operation traveling within the area according to predefined routing rules. When a new call for service appears, a selected vehicle is then routed in order to insert the new request into its predefined route. The procedure to find the optimal vehicle-request assignment requires a proper objective function that depends on predictions of state-space variables as described hereafter.

As in Sáez *et al.* [24], the modeling approach is discrete in time and the time-steps are triggered whenever a new relevant event happens, such as the occurrence of a real-time request for service (namely η_k). The index k represents the kth instant in the discrete sequence of events. Note that η_k is unknown, comes up in real time and can be characterized by two positions, indicating the pickup and the delivery, the time of the call, a label for the request and by the number of passengers. In expression (2), the specification of η_k is exposed and later discussed.

At any instant k, each vehicle j has been assigned to follow a sequence of tasks that include pickups and deliveries. Such a sequence can be represented by a function $S_j(k)$, in which the ith row represents a specific ith stop along vehicle j's route, and $w_j(k)$ is the number of scheduled stops. The control or manipulated variable corresponds to the set of sequences $u(k)=S(k)=\{S_1(k),\ldots,S_j(k),\ldots,S_F(k)\}$ associated with all the vehicles in the fleet. The proposed HAPC dispatcher selects the optimal sequences based on the minimization of an *ad hoc* objective function (as shown in Section 2.2 next). Thus, a sequence of stops assigned to vehicle j at time k, $S_j(k)$ is given by

$$S_j(k)=\begin{bmatrix} z_j^0(k) & P_j^0(k) & r_j^0(k) & \Omega_j^0(k) \\ z_j^1(k) & P_j^1(k) & r_j^1(k) & \Omega_j^1(k) \\ z_j^2(k) & P_j^2(k) & r_j^2(k) & \Omega_j^2(k) \\ \vdots & \vdots & \vdots & \vdots \\ z_j^{w_j(k)}(k) & P_j^{w_j(k)}(k) & r_j^{w_j(k)}(k) & \Omega_j^{w_j(k)}(k) \end{bmatrix} \tag{1}$$

In expression (1), $z_j^i(k)$ is a binary variable defined at instant k, which is equal to 1 if the stop i is a pickup, 0 if the stop i is a delivery. $P_j^i(k)\in R^2$ is a two-dimensional vector that shows the geographical position of stop i assigned to vehicle j in terms of spatial coordinates x and y, $r_j^i(k)$ is a tag to identify the passenger who is calling and $\Omega_j^i(k)$ is the number of passengers to be transported between the origin and destination associated with request $r_j^i(k)$. The first row of the sequence of stops in (1) represents the initial conditions, which correspond to the last stop already visited by the corresponding vehicle j.

Figure 1 shows a sequence assigned to a vehicle j at time $k\,(S_j(k))$, which is a picture of the assigned vehicle tasks. $\hat{T}_j^i(k)$ represents the expected departure time of the vehicle j at stop i, $\hat{L}_j^i(k)$ is the expected vehicle load when vehicle j leaves stop i.

$X_j(k,\varphi(t_k))$ is the current position (coordinates) computed at instant time k that depends on the traffic conditions $\varphi(t)$ in the way we will explain later. t_k is a variable connecting the continuous time (clock time) with the discrete model in time (index k). Note that $X_j(k,\varphi(t_k))$ must be in between $P_j^0(k)$ and $P_j^1(k)$. To simplify the notation, hereafter we will simply denote $X_j(k)$ to represent $X_j(k,\varphi(t_k))$. Note that the traffic conditions (φ) affect the current position of each vehicle $X_j(k,\varphi(t_k))$, which is a measurable output of the system. The vehicle position is a random variable, and $X_j(k,\varphi(t_k))$ is a realization of such a variable.

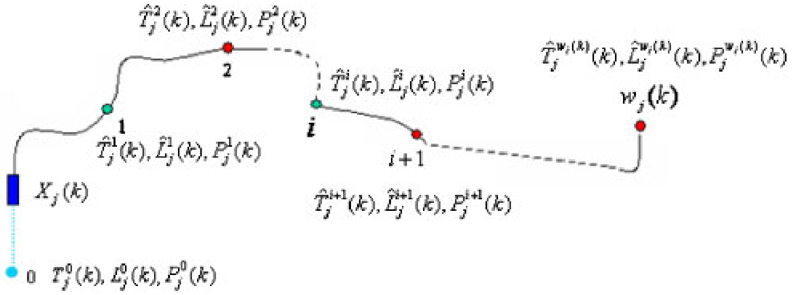

$\hat{T}_j^2(k), \hat{L}_j^2(k), P_j^2(k)$
2
$\hat{T}_j^{w_j(k)}(k), \hat{L}_j^{w_j(k)}(k), P_j^{w_j(k)}(k)$
$\hat{T}_j^i(k), \hat{L}_j(k), P_j^i(k)$
$w_j(k)$
1
i
$\hat{T}_j^1(k), \hat{L}_j^1(k), P_j^1(k)$
$i+1$
$X_j(k)$
$\hat{T}_j^{i+1}(k), \hat{L}_j^{i+1}(k), P_j^{i+1}(k)$
0 $T_j^0(k), L_j^0(k), P_j^0(k)$

Figure 1. Vehicle sequence representation.

These three types of variables $(\hat{T}_j^i(k), \hat{L}_j^i(k), X_j(k))$ conform the state-space vector as described next. Moreover, $L_j^0(k)$ and $T_j^0(k)$ are the vehicle conditions when the last call request was satisfied located at $P_j^0(k)$.

For simplicity, in this application a conceptual network with Euclidean norm as a distance estimator is considered. Although the distance is computed through a fixed measure depending on the coordinates of the initial and final conditions, the modeled travel times on segments experienced by vehicles are not fixed, since the speed is variable.

Analytically for any vehicle j, the state-space model is given by

$$\chi_j(k+1) = \begin{bmatrix} \hat{X}_j(k+1) \\ \hat{T}_j(k+1) \\ \hat{L}_j(k+1) \end{bmatrix} = \begin{bmatrix} f_X(X_j(k, \varphi(t_k)), S_j(k), \hat{v}(t,p), \eta_k) \\ f_T(X_j(k, \varphi(t_k)), \hat{T}_j(k), S_j(k), \hat{v}(t,p), \eta_k) \\ f_L(\hat{L}_j(k), S_j(k), \eta_k) \end{bmatrix} \tag{2}$$

where $\chi_j(k+1)$ is the vector of state-space variables defined for vehicle j at next instant $k+1$, as function of the control action $S_j(k)$, the disturbances $\varphi(t_k), \eta_k$, the speed model $\hat{v}(t,p)$and the state-space variables at instant k, $(\hat{T}_j^i(k), \hat{L}_j^i(k), X_j(k))$.

The estimated departure time vector $\hat{T}_j(k) = [T_j^0(k)\ \hat{T}_j^1(k)\ \cdots\ \hat{T}_j^{w_j(k)}(k)]^{\mathrm{T}}$ and the estimated load vector $\hat{L}_j(k) = [L_j^0(k)\ \hat{L}_j^1(k)\ \cdots\ \hat{L}_j^{w_j(k)}(k)]^{\mathrm{T}}$ are both vectors of dimension $w_j(k)+1$. Note that only the first component of both the expected departure time and expected load vectors at instant k is known, since the remaining components of both vectors are really expectations of what is supposed to happen at the scheduled stops of each vehicle defined in each sequence, which will depend on the expected disturbances along the vehicle routes. Thus, to compute the estimated departure time at stops the predictive model is utilized starting from the current vehicle position $X_j(k, \varphi(t_k))$ (continuously being affected by the disturbance φ). Besides, the expected load as well as the expected departure time at future stops will also depend on the demand over space and time, from where potential re-routings could affect the future load and departure times at stops.

In addition, the demand is characterized by four attributes, namely $\eta_k = (P_k, r_k, \Omega_k, \tau_k)$, which correspond to the last call and have all the information about the request (position, label, load and time).

In the proposed approach, traffic congestion is modeled through the distribution of commercial speed of the vehicles on both relevant dimensions: time and space, since traffic conditions of an urban area normally change along the day, and are different depending on where each vehicle is traveling. The real speed distribution is unknown $v(t, p, \varphi)$ and it depends on a stochastic source that comes from the network traffic conditions $\varphi(t)$ (if the specification is additive, then $\varphi(t)$ will be measured in speed units, as shown in the simulation test Section 4). Also a known velocity distribution of the urban area during a typical period of recurrent congestion is assumed available based on historical data, which is represented by a model of the speed $\hat{v}(t,p)$. All of them are specified in terms of the continuous time t and the spatial coordinate p. The functions f_X, f_T and f_L in Equations (2) define the state-space model and are specified in Equations (3)–(6) next.

DOI: 10.1002/acs

First, the dynamic model for the position associated with vehicle j is given by

$$\hat{X}_j(k+1) = X_j(k) + \int_{t_k}^{t_k+\tau} \hat{v}(t, p(t)) \frac{(P_j^0(k) - P_j^1(k))}{\| P_j^0(k) - P_j^1(k) \|_2} \, \mathrm{d}t \tag{3}$$

where $t_k \leqslant t \leqslant t_k + \tau$. So, the model requires a variable stepsize (τ), defined by the interval between the occurrence of a probable future call asking for service ($t_k + \tau$) and the occurrence of the previous call t_k. τ is calculated as a tuning parameter for the HAPC by using a sensitivity analysis described in [24]. Note that $P_j^0(k) - P_j^1(k)$ provides the information with regard to the direction of the vehicle j speed. If $P_j^1(k)$ is reached in $t < t_k + \tau$, an adaptive mechanism uploads $P_j^0(k)$ and $P_j^1(k)$ since these variables represent the last visited stop and the next scheduled stop respectively, at every instant t.

Besides, the departure time vector depends on the vehicle speed and can be computed as follows:

$$\hat{T}_j(k+1) = \begin{bmatrix} T_j^0(k) & t_k + \kappa_j^1(k) & t_k + \sum_{s=1}^{2} \kappa_j^s(k) & \cdots & t_k + \sum_{s=1}^{w_j(k)} \kappa_j^s(k) \end{bmatrix}^{\mathrm{T}} \tag{4}$$

where

$$\kappa_j^1(k) = \int_{X_j(k,\varphi(t))}^{P_j^1(k)} \frac{1}{\hat{v}(t_j(\omega), \omega)} \, \mathrm{d}\omega, \quad \kappa_j^i(k) = \int_{P_j^{i-1}(k)}^{P_j^i(k)} \frac{1}{\hat{v}(t_j(\omega), \omega)} \, \mathrm{d}\omega, \quad i = 2, \ldots, w_j(k) \tag{5}$$

$\kappa_j^i(k)$ is an estimate of the time interval between stop $i-1$ and stop i in the sequence of vehicle j, at time k. When $i=1$, the reference for computing the arrival time is the current position of the vehicle instead of the previous stop. $t_j(\omega)$ is the continuous time at which vehicle j reaches position ω. In (5), the integration is performed along the line between two consecutives stops.

Finally, the dynamics embedded in the vehicle load vector depends exclusively on the current sequence and the previous load variable at instant k. Analytically,

$$\hat{L}_j(k+1) = \begin{bmatrix} L_j^0(k) & L_j^0(k) + (2z_j^1(k) - 1)\Omega_j^1 & L_j^0(k) + \sum_{s=1}^{2} (2z_j^s(k) - 1)\Omega_j^s & \cdots \\ & L_j^0(k) \sum_{s=1}^{w_j(k)} (2z_j^s(k) - 1)\Omega_j^s & & \end{bmatrix}^{\mathrm{T}} \tag{6}$$

with z_j^s and Ω_j^s defined in expression (1).

Vehicle sequences as well as state-space variables have to satisfy a set of constraints that depend on the real conditions of the modeled DPDP. Specifically, precedence, capacity and consistency constraints are added into the dynamic model to generate only feasible sequences.

2.2. *Objective function*

The request-vehicle assignment is decided by the dispatcher (controller) based on a proper objective function that depends on predictions of the state-space variables and consequently, on the future control actions applied to the system. The objective function is specified in terms of both the total expected waiting and travel time for passengers. The idle travel time (vehicles moving around without passengers) is also included in the formulation in order to consider a *proxy* for the operational cost in the decision.

The major issue in the definition of the objective function is to define a reasonable prediction horizon N, which depends on the studied problem. A prediction at one-step ahead is equivalent to performing a myopic assignment, since only the new request (arising at k) is considered when taking the routing decision. When a predictive horizon greater than 1 is assumed, the decision maker (controller) adds the predictive feature into the formulation, since decisions taken at k will depend not only on the new request at k but also on possible events (new service requests unknown at the decision instant k) occurring at future instants $(k+1, k+2, \ldots)$. These new requests are estimated by using fuzzy clustering based on historical demand data. A set of consecutive expected calls $\{\eta_{k+1}^h, \eta_{k+2}^h, \ldots, \eta_{k+N-1}^h\}$ define a trip pattern h (note the superscript h in the call representation above to join a pattern with the calls associated with it). Thus, the central dispatcher (controller) computes the following set of sequences $S(k)\cup\bigcup_{h=1}^{H}\{S(k+1)|_{\eta_{k+1}^h}, \ldots, S(k+N-1)|_{\eta_{k+N-1}^h}\}$, which corresponds to the decisions for the entire control horizon N and for each pattern h. Then, the dispatcher applies just the next step sequence $S(k)$ based on receding horizon control. It is important to note that $S(k)$ includes the new request to be assigned (η_k), which is known (deterministic) at the decision time. The quality of the dispatcher routing decisions will depend on how well the system predicts the impact of re-routing passengers due to unknown insertions as well as traffic congestion. Note that deterministic decisions are continuously made by the dispatcher based on the information of each call that enters the system along with a forecast of a future decision corresponding to each possible pattern (scenario).

The objective function for a generic prediction horizon, N, can be written as follows:

$$\underset{S(k)\cup\bigcup_{h=1}^{H}\{S(k+1)|_{\eta_{k+1}^h},\ldots,S(k+N-1)|_{\eta_{k+N-1}^h}\}}{\text{Min}} \sum_{j=1}^{F}\sum_{h=1}^{H} p_h\cdot C_j(k+N)|_h \tag{7}$$

$$C_j(k+N)|_h = \sum_{i=1}^{w_j(k+N)} \left(\underbrace{(\hat{L}_j^{i-1}(k+N)+1)(\hat{T}_j^{i}(k+N)-\hat{T}_j^{i-1}(k+N))}_{J \text{ travel time}} + \underbrace{z_j^i(k+N-1)\alpha(\hat{T}_j^i(k+N)-T_j^0(k+N))}_{J \text{ waiting time}} \right)\Bigg|_h \tag{8}$$

where $C_j(k+N)|_h$ in (8) is the cost function of vehicle j at instant $k+N$, provided that the trip pattern h, characterized by $\{\eta_{k+1}^h, \eta_{k+2}^h, \ldots, \eta_{k+N-1}^h\}$, occurs. Such a cost also depends directly on the set of sequences to be applied, namely $\{S(k), S(k+1)|_{\eta_{k+1}^h}, \ldots, S(k+N-1)|_{\eta_{k+N-1}^h}\}$, which are the optimization variables. H is the number of trip patterns considered, p_h is the probability of occurrence of the hth trip pattern (future demand). $w_j(k+N)$ is the number of stops estimated for vehicle j at instant $k+N$.

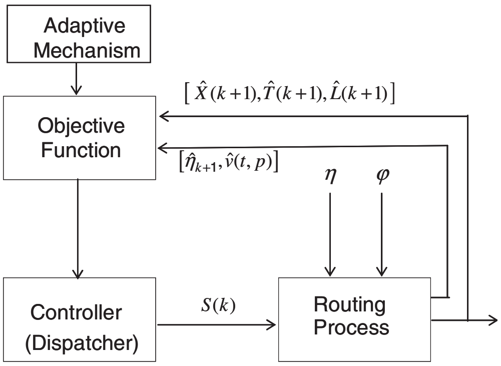


Figure 2. Closed-loop diagram of a hybrid adaptive predictive approach for DPDP.

The future instants $k+1$, $k+2, \ldots,$ are generated by using a variable time step. Then, the expected call associated with pattern h to happen n steps ahead is $\eta^h_{k+n} = (P^h_{k+n}, r^h_{k+n}, \Omega^h_{k+n}, \tau^h_{k+n})$, where τ^h_{k+n} is the expected occurrence time of such a call in the future. Owing to the large number of parameters, we simplify the computations by assuming $\tau^h_{k+n} = \tau_{k+n}$ $\forall h$. Besides, $\tau_{k+n} = \tau_{k+n-1} + \Delta\tau$ with $\Delta\tau$ tuned through sensitivity analysis. Finally, α is a weight for the waiting time to differentiate its contribution compared with that of travel time in the objective function. The number of future demand patterns H and their probabilities of occurrence p_h are parameters in the objective function, and they have to be computed based on either real-time data historical data or a combination of both. In this case, Fuzzy clustering is used to model the demand ($\hat{\eta}_{k+1}$) by considering only historical data (see [24] for details).

In the case of the one-step ahead strategy (myopic), the new requirement is one and known, and therefore its probability is equal to 1. In case of the two-step ahead prediction, the objective function requires the estimation of probabilities that the new call entering the system two steps ahead falls into each demand pattern. A distribution for the time interval between successive calls is also assumed in order to compute time interval probabilities.

In summary, the closed-loop routing system is shown in Figure 2. The HAPC represented by the dispatcher takes the routing decisions in real time based on the information it has from the routing system (process) and the values for the attributes of the vehicle fleet and the transport system (model). The demand (η) and the traffic conditions (φ) are disturbances (stochasticity). An adaptive mechanism for the proposed controller is added to the chart in Figure 2, representing the necessity of adapting the size of the model when either a new request arrives or a request has been satisfied. Moreover, the objective function is influenced by the prediction of the uncertain demand and traffic conditions ($\hat{\eta}_{k+1}$ and $\hat{v}(t,p)$, respectively).

2.3. Optimization method

In this application, the optimization is performed over a reduced space of solutions that satisfy the *no-swapping* constraint. For each vehicle, the '*no-swapping*' simplification searches the best insertion position, in which the candidate call (whether it is the current deterministic call or a

DOI: 10.1002/acs

future call representative of certain future pattern) is inserted into a predefined sequence without changing the previous order of the points defined in the vehicle route (sequence). The decision of adopting this simplification was motivated from the results by Cortés [22], who showed empirically through simulation that in most pickup and delivery problem configurations, the optimal solution of inserting a new request does not alter the order of the previous sequences. Besides, there are practical reasons for considering the *no-swapping* case taking into account that the optimization is dynamic, and the previous information (previous sequences) is crucial to take fast decisions constantly. Note that the '*no-swapping*' constraint is added to simplify the NP-hard problem embedded, and consequently to reduce the time spent by the algorithm when computing the cost of each possible insertion.

The HAPC approach for the DPDP problem generates a highly non-linear optimization problem, which is NP-Hard. Owing to this feature of the problem, it is not feasible to solve it by using traditional algorithms for mixed-integer problems. In Sáez *et al.* [24], genetic algorithms (GA) were proposed to find good-quality solutions for the DPDP problem.

In summary, GA is used as an efficient optimization solver for the DPDP problem, where the optimization variables identify the stops that must be satisfied by the vehicle fleet. The individuals are the feasible sequences, fulfilling the load, precedence and no swapping constraints defined in Section 2.1. The gene of an individual considers the following components: the vehicle j used for the new insertion and the sequence position of the new call (for both pickup and delivery) within the previous sequence, assuming the *no-swapping* policy. Owing to the precedence and *no-swapping* constraints, the previous sequence is held.

For more than one step ahead, GA is conducted for each scenario associated with a specific demand pattern. Previously, the demand patterns are categorized by a fuzzy clustering technique, as detailed in Sáez *et al.* [24]. As GA considers random generation of individuals, the genetic operators (mutation or crossover) could provide infeasible solutions that have to be removed or repaired (typically through the capacity constraint). The number of individuals for each population has to be smaller than the total number of feasible combinations in order to avoid solving the explicit enumeration method.

The complexity of the GA is proportional to the number of individuals for each iteration (generation) multiplied by the number of generations. Both, the number of individuals and the number of generations are parameters to be tuned by the GA designer. The individuals at each iteration are randomly chosen by using genetic operators (mutation and crossover) and the number of generations is stated as the GA stopping criterion. This procedure allows a considerable reduction in computation time providing near optimal solutions.

3. FAULT DETECTION AND ISOLATION, FUZZY FAULT-TOLERANT CONTROL (FDI-FFTC) SYSTEM FOR UNPREDICTABLE TRAFFIC CONGESTION EVENTS

The approach described so far seems useful when a speed distribution is available and calibrated in both relevant dimensions, time and space. For that, a statistical work has to be conducted from historical data of the studied area, which allows us to have a good prediction of recurrent (predictable) traffic conditions. However, in real transportation networks, the unpredictable congestion events can also affect the expected vehicle travel times, resulting in bad quality routing with the occurrence of a big incident close to the dispatch areas. In order to incorporate such an effect, we propose an FDI method for detecting the unpredictable traffic jam and an FFTC to force the vehicles avoiding the affected zones. Both systems will permit to reduce the effect of the incident over the users waiting and travel times. The unpredictable events will be detected and modeled by using real-time information from our vehicle fleet, noting that the method is easily extended to the use of any other sources of online speed data. In the literature, there are some preliminary results for fault detection problems and diagnosis in the transport infrastructure, like traffic monitoring sensors and vehicle mechanical systems [36]. Related with anomalies, Aronson *et al.* [37] consider the re-route problem as incident repair method for a multimodal transport system; the considered incidents are changes in freight orders, traffic jams and vehicles faults. Weinstein [38] presents a model oriented to objects to describe the planning of multiagent systems, which enables to diagnose the anomalies executions.

In this work, the measurements of $v(t, p, \varphi)$ are available for each position p at every instant time t. Besides, a recurrent model of the speed $\hat{v}(t, p)$ is assumed. The speed measurements are compared with the results of the speed distribution model and used for the FDI method. Analytically, the speed residual is given by $e(t) = \hat{v}(t, p) - v(t, p, \varphi)$. Thus, the residual $e(t)$ for a reasonable period of time TT is analyzed in order to activate the FDI system. If the system detects a fault during the entire period TT, the FDI system will be activated. During TT, the information of the real velocity is recorded to modify the recurrent model of velocity $\hat{v}(t, p)$ used by the HAPC control strategy in order to avoid the negative effects of the incident. This procedure corresponds to the FFTC method.

After the FDI system is activated, a set of rules have to be defined in order to model the incident impact. These rules generate the new recurrent model that includes the original recurrent model $\hat{v}(t, p)$ and the fuzzy rules for the incident representation. The fuzzy approach is used in order to capture the non-linear behavior of the incident impact. Moreover, these fuzzy rules permit to distinguish different magnitudes and features of the incident.

First of all, the definition of the fuzzy rules requires to establish the velocity associated with each type of incident, which is modeled by a Gaussian function (μ, σ, m). In the Gaussian model, μ is the location of the center of the incident, σ is the affected zone radio and m represents the minimum velocity located at the center where the incident is supposed to happen. These three parameters are adjusted based on the type of the incident. The duration of Gaussian model is assumed constant. The parameter σ is assumed to be inversely proportional to the Euclidean distance associated with the vehicle movement during TT, and μ is associated with the linear trajectory traveled by the vehicle. Analytically,

$$\sigma = \frac{1}{\|P_{\mathrm{D}} - P_{\mathrm{F}}\|}, \quad \mu = P_{\mathrm{D}} + \lambda \cdot (P_{\mathrm{F}} - P_{\mathrm{D}}), \quad 0 \leqslant \lambda \leqslant 1 \tag{9}$$

where P_{D} is the position of the vehicle where the fault is detected and P_{F} the position of the same vehicle after TT. Next, once the type of incident is established, the corresponding fuzzy rules are defined based on the expected behavior of the system under incident conditions. These rules are fed by two inputs: the speed residual $e(t)$ and the increment of the residual along the trajectory $\mathrm{d}e(t) = e(t) - e(t-1)$. The rule outputs are the movement size λ and the minimum velocity m for each type of incident, the latter proportional to $m^* = \max\{\mathrm{d}e(t), \mathrm{d}e(t-1)\}$. The fuzzy rules and their corresponding membership functions are defined in Figure 3.

R_1: If $e(t)$ is M and $de(t)$ is N then λ is 0.0, m is m^*

R_2: If $e(t)$ is M and $de(t)$ is Z then λ is 0.5, m is $1.1 \cdot m^*$

R_3: If $e(t)$ is M and $de(t)$ is P then λ is 1.0, m is $1.3 \cdot m^*$

R_4: If $e(t)$ is H and $de(t)$ is N then λ is 0.5, m is m^*

R_5: If $e(t)$ is H and $de(t)$ is Z then λ is 1.0, m is $1.2 \cdot m^*$

R_6: If $e(t)$ is H and $de(t)$ is P then λ is 1.0, m is $1.5 \cdot m^*$

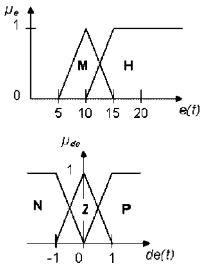


Figure 3. Fuzzy rules and membership functions for the incident velocity model.

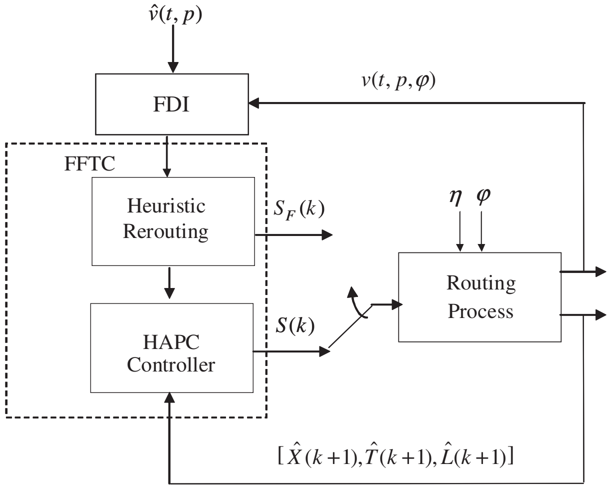


Figure 4. FDI-FFTC system for the DDPP.

The proposed procedure FDI-FFTC method (as shown in Figure 4) consists of the following steps:

1. When some vehicle detects the incident traffic jam for a certain period of time (FDI is activated):
   1.1. A new recurrent model is generated by considering both the $\hat{v}(t,p)$ and the proposed fuzzy rules. The incident model based on fuzzy rules intends to represent the effects of the unpredictable event.
   1.2. The requests located somewhere inside the affected zone are re-assigned as new calls for the dispatcher system based on HAPC, now considering the new recurrent model according to the new traffic conditions detected. As re-routing decisions of the re-assignment calls need to be fast, a one-step ahead HAPC is proposed ($S_F(k)$).

DOI: 10.1002/acs

1.3. After the re-routing, the new call requests are assigned by the HAPC strategy ($S(k)$) considering the same new recurrent model, and for the two-step ahead case.

2. Otherwise, when the FDI system does not detect the incident, the HAPC strategy described in Section 2 is used directly ($S(k)$) for the two-step ahead case as well.

4. SIMULATION TESTS

In this section, some simulation tests are carried out in order to quantify the potential benefits of such a methodology in the context of a DPDP. In the experiments a transportation fleet of nine vehicles, with capacity for four passengers each is used. The simulation tests are implemented in Matlab version 7.0.1 release 14 running on a Pentium® D CPU 3.20 GHz processor.

The future origin–destination trip patterns are unknown. However, historical demand obtained from the average demand measured over a week before or so is available. This scenario is not real. However, the demand patterns follow a heterogeneous distribution inspired on real data from the Origin–Destination Survey in Santiago, Chile, 2001. An urban service area of approximately 81 km^2 is considered and all simulations are performed over two representative hours (14:00–14:59,15:00–15:59) of a working day. We assume vehicles traveling straight between stops and the embedded network following the speed distribution stated in the following equation:

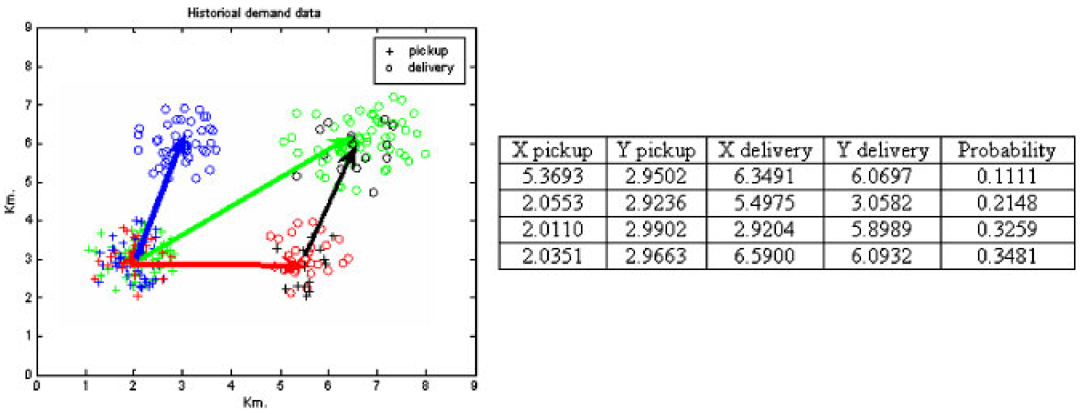
$$v(t,p,\varphi)=20+\left(5-\frac{t}{12}\right)\cdot \mathrm{e}^{-\frac{(p_x-4)^2+(p_y-4)^2}{2}}+\left(\frac{t}{12}-5\right)\cdot \mathrm{e}^{-\frac{(p_x-7)^2+(p_y-6)^2}{2}}+\varphi(t) \tag{10}$$

where t (min) is the clock time, $t=0$ min corresponds to 14:00, and $t=120$ min to 16:00. $p=(p_x,p_y)$ (km) denotes a position in terms of the plane coordinates inside the urban area. $\varphi(t)$ is the white noise that captures the stochasticity coming from traffic congestion.

The speed distribution shows how the congestion moves from one side of the urban area to the other along the 2-h simulation. The historical data generated via simulation follow the trips' patterns shown in Figure 5 with arrows. From historical data and a fuzzy zoning method proposed by Sáez *et al.* [24], Figure 5 also shows the pickup and delivery coordinates and the probabilities for most relevant trip patterns.

For the simulation test, 120 calls were generated following the same behavior as that of the historical data. Regarding the temporal dimension, we assume a negative exponential distribution for time intervals between calls with rate of 0.9 call/min. In terms of spatial distribution, pickup and delivery points were generated randomly within each corresponding zone. A reasonable warm up period was considered to avoid boundary distortions (10 calls at the beginning and 10 at the end). Fifty replications of each experiment were conducted to obtain global statistics. With regard to the objective function, a weight $\alpha=1$ was used, which means that travel time is as important as waiting time into the cost function expression. In order to analyze and evaluate the performance of HAPC strategies, simulation tests were conducted for one- and two-step ahead algorithms under the same conditions. Two-step ahead algorithm was performed considering the four trip patters shown in Figure 5. We present the results of 50 replications with GA solver by using 20 individuals and 20 generations.

Table I shows the effective waiting and travel times of passengers by using the HAPC based on GA for one- and two-step ahead prediction, and for the two velocity estimations. A constant estimation of velocity means that the expected departure time is computed based on the constant



X pickup	Y pickup	X delivery	Y delivery	Probability
5.3693	2.9502	6.3491	6.0697	0.1111
2.0553	2.9236	5.4975	3.0582	0.2148
2.0110	2.9902	2.9204	5.8989	0.3259
2.0351	2.9663	6.5900	6.0932	0.3481

Figure 5. Origin–destination trip patterns. Pickup and delivery coordinates and probabilities: fuzzy zoning.

Table I. Performance comparison for one- and two-step ahead algorithms. Constant velocity estimation.

	Variable velocity estimation				Constant velocity estimation			
	Waiting time (min)		Travel time (min)		Waiting time (min)		Travel time (min)	
Strategy	Mean	St. dev.	Mean	St. dev.	Mean	St. dev.	Mean	St. dev.
One step ahead	15.443	1.64	17.879	0.61	15.844	1.25	18.346	0.78
Two step ahead	13.618	1.90	16.939	0.65	14.077	1.78	17.002	0.74
Savings 2 step	1.82		0.94		1.76		1.34	
Improvement (%)	11.81		5.26		11.15		7.32	

speed. The second estimation (variable velocity) is more realistic since it is adapted to the network velocity conditions through the recurrent model $\hat{v}(t, p)$. We observe that waiting time is significantly reduced by using the two-step ahead method (12%) when compared against the myopic one-step ahead method. In addition, an improvement in travel time is also observed.

Table II describes the operational costs for the entire vehicle fleet. In addition, total effective costs are also reported in the table. We observe that vehicle operational costs and the total effective costs are still reduced by running both the constant velocity (8.81%) and the variable velocity (8.00%) methods.

From this example, we found an improvement of 3.26% in waiting time, and still one improvement of 1.68% in total time, only due to the fact of including a more sophisticated prediction of the velocity over the space and time, based on historical data (recurrent congestion).

From the results described above, we found that including a good estimation of the distribution of the speed into the prediction always improves the routing decisions, just from recognizing the variability of the speed (from historical data) as part of the prediction. Even though the improvement of this modeling scheme above the improvement resulting from the demand prediction seems not very impressive, we claim that the integrated approach should produce much better results as the variability of the speed (not only in time but also in space) became larger.

DOI: 10.1002/acs

Table II. Vehicle and passenger operational costs.

	Variable velocity estimation				Constant velocity estimation			
	Operational costs (min)		Effective total costs (min)		Operational costs (min)		Effective total costs (min)	
Strategy	Mean	St. dev.	Mean	St. dev.	Mean	St. dev.	Mean	St. dev.
One step ahead	143.68	7.3172	3809.1	183.23	145.13	7.84	3906.0	189.51
Two step ahead	142.95	8.7826	3504.3	256.51	143.21	7.83	3562.0	258.02
Savings 2 step	0.73		304.82		1.91		344.07	
Improvement (%)	0.51		8.00		1.32		8.81	

Apart from the basic formulation, we developed a methodology (Section 3) to deal with unpredictable congestion, under the same HAPC formulation developed for recurrent congestion in Section 2. By following the same line of reasoning as in the previous paragraph, in this case we will try to measure the impact of applying this approach to a scenario in which suddenly a big incident occurs, generating for a while a big congestion around the affected area. The system should react in real time to the occurrence of such an incident and take proper routing decisions taking into account such a change. We intuitively expect considerable cost savings in this case, as shown next.

4.1. *Fault detection simulation test*

In order to test the fault detection proposal a reduced fleet of four vehicles is used. For the simulation test, 75 calls were generated over the whole simulation period of 2 h. In Figure 6, the speed distribution defined in Equation (10) is shown for four instant times. Figure 7 shows the recurrent model $\hat{v}(t,p)$ considered for the HAPC before the incident. At 15:00, an incident happens (as shown in Figure 8) and thus, the fault detection module becomes active by checking the detection rules described in Section 3.

Table III reports the waiting time, travel time, total time, operational cost and effective total cost for two cases. The former (Case 1) considers the HAPC controller by using the speed distribution from the initial recurrent model, without incorporating the incident that start getting reflected in the real speed data taken online by the fleet of vehicles. The latter (Case 2) considers the HAPC scheme together with the proposed FDI detection system. Thus, the HAPC approach considers a more realistic recurrent model that provides the effect of the incident. In addition, we include a third case as a benchmark, in which the HAPC is applied by assuming completely known the distribution of the speed as a result of the incident occurrence (Case 3), and therefore, the routing decisions are preformed based on a velocity model including the fault effect (Figure 8).

The last row in Table III shows the additional improvement of Case 3 above Case 2 with respect to Case 1, to have an idea of how far the solution is from the ideal situation (Case 3) in which the incident (fault) is completely known at any time. As we can appreciate, the improvement in this particular case is of the order of 4% (effective total cost) above the improvement of Case 1 over the model without including speed distribution in the prediction. We appreciate a relevant improvement in terms of waiting time in case of using the FDI-FFTC method (16.45%), in this case even better than having the information of the fault beforehand. More tests have to be run in order to completely explain this last result. The intuition suggests that this apparent contradiction

DOI: 10.1002/acs

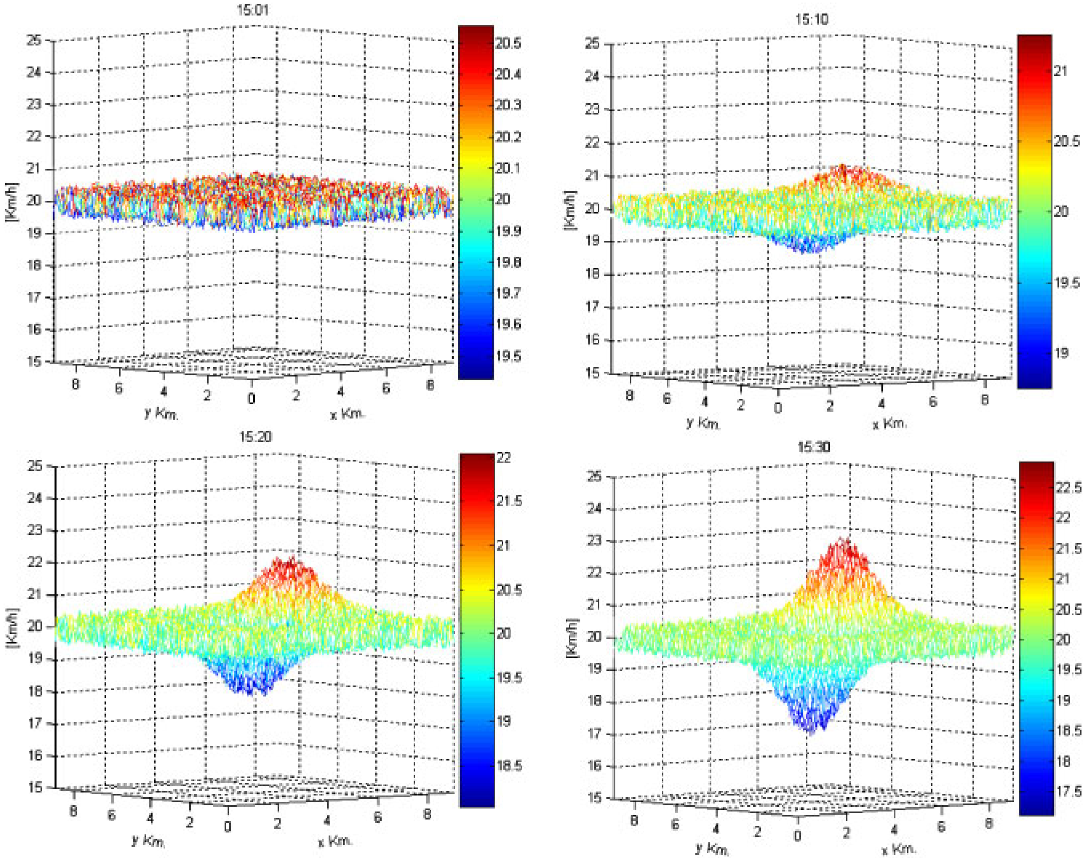


Figure 6. Real speed distribution without incident.

can be explained from a trade-off between travel and waiting time, favoring the former in Case 3 due to the extra available information with regard to the fault location and impact. Case 2 anyway performs quite well when compared against the benchmark (Case 3) in all cases, except in travel time, in which the fault detection does not help. Finally, in Figure 9 the real situation is compared with the new speed model, which adaptively updates the fault detector whenever the vehicles of the fleet enter the fault impact zone and report its experienced speed. Thus, Figure 9(a) has to be compared with Figure 9(b), while Figure 9(c) has to be compared with Figure 9(d), for the real and modeled speed, respectively, at two instants. Results could improve considerably if more speed measurement stations were added to the detection system (both fixed and mobile stations).

5. SYNTHESIS, CONCLUSION AND FURTHER RESEARCH

In this paper, we present an HAPC formulation for a DPDP that combines two sources of uncertainty when taking real-time vehicle routing decisions. On the one hand, the formulation considers

DOI: 10.1002/acs

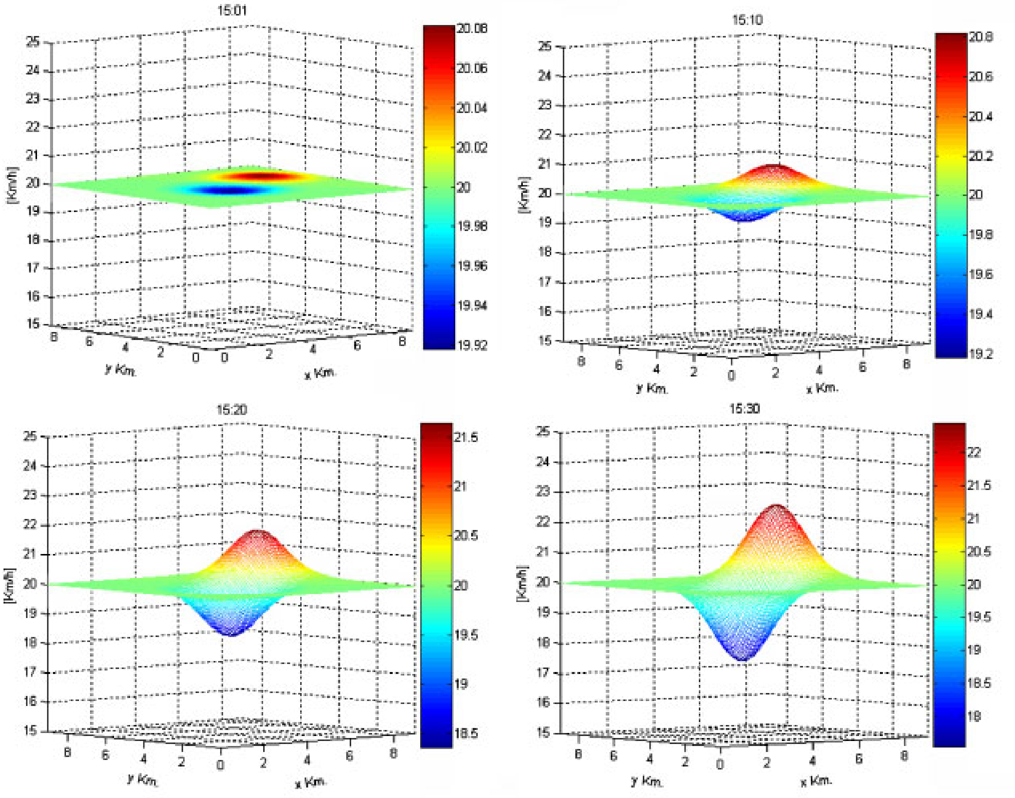


Figure 7. Speed distribution for the initial recurrent model.

uncertainty from possible future demand influencing routes of current customers, which follows the original scheme proposed by Sáez *et al.* [24], and on the other hand, the scheme also considers the uncertainty behind the traffic congestion conditions. The predictive model is proposed in order to modify the preplanned schedule of vehicle routes based on traffic information around their routes as well as future insertions coming from unknown real-time service requests. In our approach, traffic congestion is modeled through the distribution of commercial speed of the vehicles on both relevant dimensions: time and space. The approach allows modeling not only predictable congestion conditions but also unpredictable situations, such as incidents occurring unexpectedly at any location on the traffic network. In the second case, we also utilize online (real-time) data regarding speed conditions from the fleet of vehicles moving around serving the demand.

Results show the potential benefits of such an approach. We can immediately mention several important contributions of this paper. First, the integrated HAPC allows systematizing the formulation of the DPDP as a control problem, which open more possibilities for using sophisticated techniques not only to characterize the dynamic problem properly but also to solve complex DPDP configurations unable to be treated without such a framework. Second, as far as we know, in the specialized literature there is no other DPDP formulation allowing prediction of both, future demand as well as future traffic conditions. Third, we found very attractive (in terms of both

DOI: 10.1002/acs

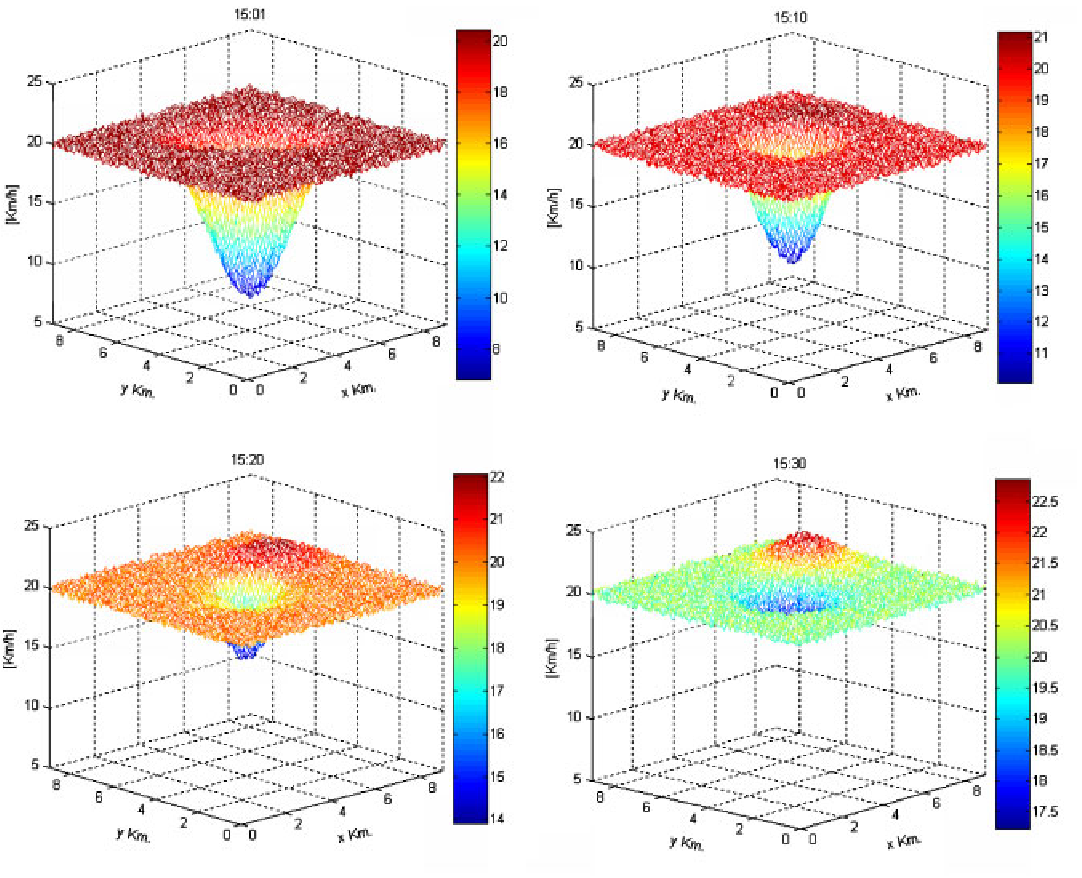


Figure 8. Real speed distribution with incident.

Table III. Performance comparison for fault detection method.

	Waiting time (min)	Travel time (min)	Total time (min)	Operational cost (min)	Effective total cost (min)
	Mean	Mean	Mean	Mean	Mean
Case 1	9.5110	12.6994	22.2104	132.3360	687.3965
Case 2	7.9461	12.9906	20.9367	132.0360	659.7205
Improvment (%)	16.45	−2.3	5.73	0.2	4.01
Case 3	8.1758	11.8525	20.0283	131.9050	632.6113
Δ Improvment (%)	−2.42	8.96	4.09	0.1	3.94

computation time and quality solutions) the use of solution methods coming from the artificial intelligence literature (such as GA, Fuzzy logic and others) in the context of this problem. Additional tests have to be conducted to adjust the embedded parameters and to sophisticate the methods

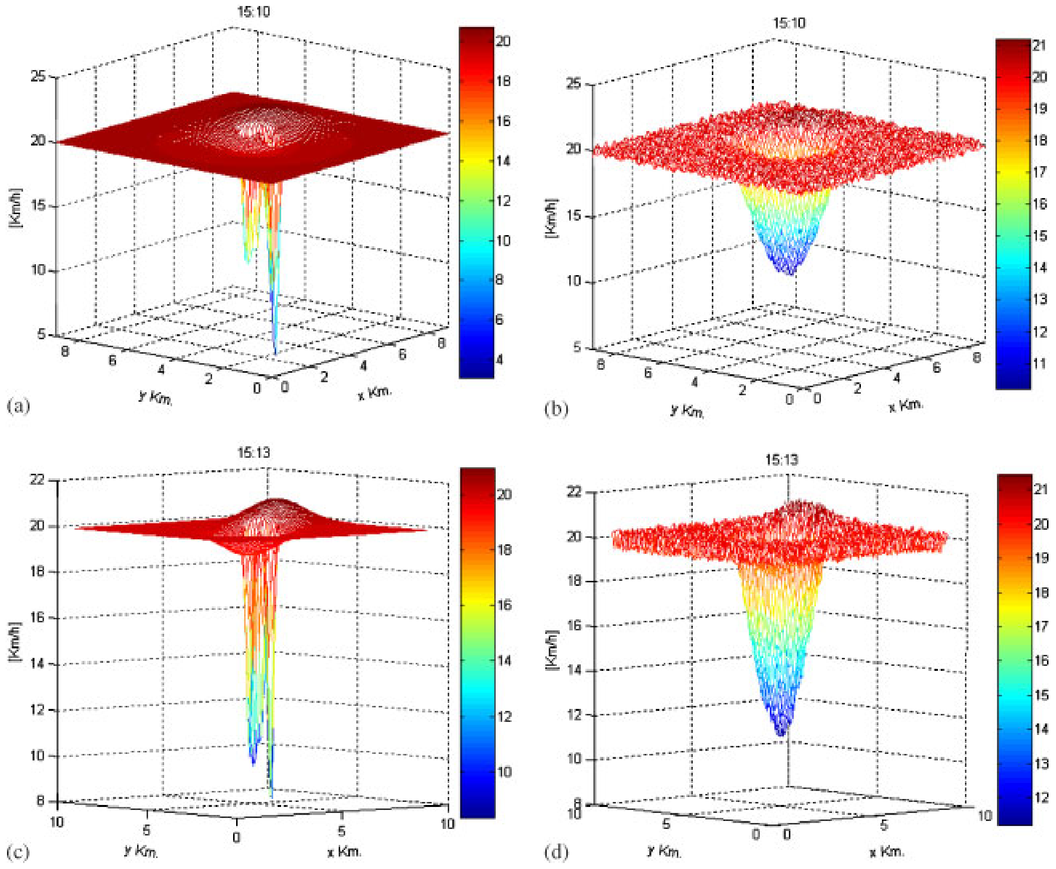


Figure 9. Comparison between model and real speed distribution with incident.

in order to get better solutions under realistic scenarios. Fourth, we realize that the occurrence of an incident can be treated under an FDI-FFTC scheme, allowing the reaction of the controller and the adjustment of the speed distribution parameters to significantly improve the dispatch rules under such a distorted scenario. The addition of the speed distribution into the model ensures a better estimation of both waiting and travel times not only due to demand prediction but also because of traffic congestion predictions, generating better real-time routing decisions, and consequently better performance of the dispatch service. The more information we have the system, the better performance can be obtained from the HAPC framework.

This paper represents a first step in the elaboration of a sophisticated HAPC approach to model DPDP and use prediction in the current decisions. The next step is to consider a real network configuration (with specific links and nodes) and replace the generic speed model in space by a velocity distribution model at a link level. This extension requires the coding of a time-dependent shortest path algorithm to compute optimal routes from point to point through the network, with link travel times depending on the time at which vehicles reach the upstream node of such a link. The coding can result harder; however, the general framework (Section 2) remains the same. We propose

DOI: 10.1002/acs

the use of traffic microsimulation in order to have a better quantification of the performance of the system in real time (simulation time). Better velocity models should result in better performance of the HAPC scheme. In the case of unexpected incidents, we propose an FDI-FFTC method. However, we recognize that the proposed rules can be further improved, sophisticating the way in which the system reacts to the occurrence of the detected fault. One straight extension is to somehow re-route those vehicles whose sequence path falls into the fault area, even though the associated stops are not inside the affected zone. Besides, the present formulation can be extended to the use of fixed stations monitoring traffic conditions at strategically chosen locations over the urban area, in order to have more data available to better trigger the FDI detection.

ACKNOWLEDGEMENTS

This research was partially financed by Fondecyt, Chile, grants 1061156 and 1061261, the Millennium Institute 'Complex Engineering Systems' and the ACT-32 Project 'Real Time Intelligent Control for Integrated Transit Systems'. Also, we thank Mr. Pablo Medina for his valuable comments and the anonymous referees.

REFERENCES

1. Desrosiers J, Soumis F, Dumas Y. A dynamic programming solution of a large-scale single-vehicle dial-a-ride with time windows. *American Journal of Mathematical and Management Sciences* 1986; **6**:301–325.
2. Savelsbergh M, Sol M. The general pickup and delivery problem. *Transportation Science* 1995; **29**(1):17–29.
3. Psaraftis H. A dynamic programming solution to the single many-to-many immediate request dial-a-ride problem. *Transportation Science* 1980; **14**(2):130–154.
4. Psaraftis H. Dynamic vehicle routing problems. In *Vehicle Routing Methods and Studies*, Golden BL, Assad AA (eds), 1988; 223–248.
5. Gendreau M, Guertin F, Potvin J, Taillard E. Parallel Tabu search for real-time vehicle routing and dispatching. *Transportation Science* 1999; **33**:381–390.
6. Kleywegt AJ, Papastavrou JD. The dynamic and stochastic Knapsack problem. *Operations Research* 1998; **46**:17–35.
7. Madsen O, Raven H, Rygaard J. A heuristics algorithm for a dial-a-ride problem with time windows, multiple capacities, and multiple objectives. *Annals of Operations Research* 1995; **60**:193–208.
8. Jaw J, Odoni A, Psaraftis H, Wilson N. A heuristic algorithm for the multivehicle many-to-many advance-request dial-a-ride problem. *Working Paper MITUMTA-82-3*, MIT, Cambridge, 1986.
9. Dial R. Autonomous dial a ride transit—introductory overview. *Transportation Research—Part C* 1995; **3**:261–275.
10. Bertsimas D, Van Ryzin G. A stochastic and dynamic vehicle routing problem in the Euclidean plane. *Operations Research* 1991; **39**:601–615.
11. Bertsimas D, Howell LH. Further results on the probabilistic traveling salesman problem. *European Journal of Operational Research* 1993; **65**:68–95.
12. Swihart MR, Papastavrou JD. A stochastic and dynamic model for the single-vehicle pick-up and delivery problem. *European Journal of Operational Research* 1999; **114**:447–464.
13. Thomas BW, White III CC. Anticipatory Route selection. *Transportation Science* 2004; **38**(4):473–487.
14. Kleywegt AJ, Papastavrou JD. The dynamic and stochastic Knapsack problem with random sized items. *Operations Research* 2001; **49**(1):26–41.
15. Papastavrou JD, Rajagopalan S, Kleywegt AJ. The dynamic and stochastic Knapsack problem with deadlines. *Management Science* 1996; **42**:1706–1718.
16. Larsen A. The dynamic vehicle routing problem. *Ph.D. Thesis*, Technical University of Denmark, 2000.
17. Ichoua S, Gendreau M, Potvin JY. Exploiting knowledge about future demands for real-time vehicle dispatching. *Transportation Science* 2006; **40**(2):211–225.
18. Spivey M, Powell WB. The dynamic assignment problem. *Transportation Science* 2004; **38**(4):399–419.
19. Powell WB. A comparative review of alternative algorithms for the dynamic vehicle allocation problem. In *Vehicle Routing: Methods and Studies*, Golden BL, Assad AA (eds). North-Holland: New York, 1988; 249–292.

DOI: 10.1002/acs

20. Godfrey G, Powell WB. An adaptive dynamic programming algorithm for stochastic resource allocation problems I: single period travel times. *Transportation Science* 2002; **36**:21–39.
21. Topaloglu H, Powell W. A distributed decision-making structure for dynamic resource allocation using non-linear functional approximations. *Operations Research* 2005; **53**(2):281–297.
22. Cortés CE. High-coverage point-to-point transit (*HCPPT*): a new design concept and simulation-evaluation of operational schemes for future technological deployment. *Ph.D. Dissertation*, University of California at Irvine, U.S.A., 2003.
23. Cortés CE, Sáez D, Murcia F, Núñez A, Control Predictivo Híbrido para un sistema personalizado de transporte público puerta a puerta programado en tiempo real (in Spanish). *Proceedings of the XII Chilean Conference in Transport Engineering*, Valparaíso, Chile, October 2005; 375–388.
24. Sáez D, Cortés CE, Núñez A. Hybrid adaptive predictive control for the multi-vehicle dynamic pickup and delivery problem based on genetic algorithms and fuzzy clustering. *Computers and Operations Research* 2007, DOI: 10.1016/j.cor.2007.01.025.
25. Berman O, Simchi-Levi D. The traveler salesman location problem on stochastic networks. *Transportation Science* 1989; **23**:54–57.
26. Kao E. A preference order dynamic problem for a stochastic traveling salesman problem. *Operations Research* 1978; **26**:1033–1045.
27. Sniedovich M. Analysis of the preference order traveling salesman problem. *Operations Research* 1981; **29**: 1234–1237.
28. Carraway RL, Morin TL, Moskowitz H. Generalized dynamic programming for stochastic combinatorial optimization. *Operations Research* 1989; **37**:819–829.
29. Laporte G, Louveaux FV, Mercure H. The vehicle routing problem with stochastic travel times. *Transportation Science* 1992; **26**:161–170.
30. Lambert V, Laporte G, Louveaux F. Designing collection routes through bank branches. *Computers and Operations Research* 1993; **20**:783–791.
31. Keyton A, Morton D. Stochastic vehicle routing with random travel times. *Transportation Science* 2003; **37**(1): 69–82.
32. Hill A, Benton W. Modelling intra-city time-dependent travel speeds for vehicle scheduling problems. *Journal of the Operation Research Society* 1992; **43**:343–351.
33. Malandraki C, Daskin MS. Time dependent vehicle routing problems: formulations, properties and heuristic algorithms. *Transportation Science* 1992; **26**:185–200.
34. Fleishmann B, Gietz M, Gnutzmann S. Time-varying travel times in vehicle routing. *Transportation Science* 2004; **38**(2):160–173.
35. Kim S, Lewis M, White C. Optimal vehicle routing with real time traffic information. *IEEE Transactions on Intelligent Transportation Systems* 2005; **6**:178–188.
36. Capriglione D, Liguori C, Pietrosanto A. Analytical redundancy for sensor fault isolation and accommodation in public transportation vehicles. *IEEE Transactions on Instrumentation and Measurement* 2004; **53**(4):993–999.
37. Aronson LD, van der Krogt RPJ, Zutt J. Automated transport planning using agents. *Proceedings of the International Congress on Freight Transport Automation and Multimodality*: *Organisational and Technological Innoventions* (*FTAM'02*), Delf, The Netherlands, May 2002. Available on http://pds.twi.tudelft.nl/~jonne/pubs.html.
38. Weinstein R. RFID: a technical overview and its application to the enterprise. *IT Professional* 2005; **7**(3):27–33.

DOI: 10.1002/acs