International Journal of Operations Research

International Journal of Operations Research Vol. 5, No. 4, 233−241 (2008)

Waiting Strategies for the Dynamic and Stochastic Traveling Salesman Problem

Gianpaolo Ghiani[1,*], Gilbert Laporte[2], Emanuele Manni[1], and Roberto Musmanno[3]

[1]Dipartimento di Ingegneria dell'Innovazione University of Salento via per Monteroni 73100, Lecce, Italy

[2]Canada Research Chair in Distribution Management HEC Montréal 3000, chemin de la Côte-Sainte-Catherine

Montréal, Canada H3T 2A7

[3]Dipartimento di Elettronica, Informatica e Sistemistica University of Calabria via P. Bucci 41C 87036 Rende (CS), Italy

Received March 2008; *Revised June 2008*; *Accepted October 2008*

Abstract—In the *Dynamic and Stochastic Traveling Salesman Problem* (DSTSP) a vehicle has to service a number of requests which are disclosed in a dynamic fashion over a planning horizon. When the vehicle is temporarily idle, one option is to reposition it in anticipation of future demand. The aim of this paper is to study waiting strategies for the DSTSP under a probabilistic characterization of customer requests. We determine an optimal policy through a Markov decision process and we develop both lower and upper bounds (analytically and heuristically, respectively) on the optimal policy cost. The behavior of these procedures is illustrated on a numerical example and tested on a set of random instances.
Keywords—Traveling Salesman, Real-time fleet management, Markov decision processes, Transportation.

---

1. INTRODUCTION

The purpose of this article is to introduce the *Dynamic and Stochastic Traveling Salesman Problem* (DSTSP), as well as to study exact and heuristic waiting policies for it. The DSTSP is defined on a graph $G = (V, A)$, where V is a vertex set and A is an arc set. A single vehicle based at a *depot* i_0 has to service a number of pick-up requests, or delivery requests, but not both. The choice to use an uncapacitated vehicle reflects, for instance, the situation in the courier and package-express industries, where the size of parcels is small enough, so that vehicle capacity is not a crucial aspect. Request $i_k \in V' \subseteq V$ ($k = 1, ..., n$) may arise at time instant T_k with probability p_k. A customer i_k may not require service if $p_k < 1$. Time instants T_k ($k = 0, ..., n$), with $T_0 = 0$, are assumed to be integer and the requests are supposed to be statistically independent. The vehicle may wait at any vertex (both a customer or a non-customer vertex) in order to anticipate future demand. It is worth noting that, unlike what happens in the classical (static) *Traveling Salesman Problem*, in which the vehicle follows a shortest path between two consecutive customers, in the DSTSP the vehicle may wait for some time at some vertices outside the current route. This may be useful, for example, to promptly collect enough demand in an area before moving in another part of the service territory. Let t_{ij} be the shortest travel time from vertex $i \in V$ to vertex $j \in V$. As is common in dynamic vehicle routing problems, the aim is to maximize overall customer service level rather than minimize the total travelled distance. Let τ_k be the service time of a customer i_k requiring service. To each customer is associated a non-decreasing and convex function $f_k(\tau_k)$ expressing the penalty associated with customer i_k. This definition includes the case where $f_k(\tau_k)$ represents customer *waiting time* (i.e., $f_k(\tau_k) = \tau_k - T_k$, $\tau_k \geq T_k$) or a more involved penalty function (e.g., $f_k(\tau_k) = 0$, $T_k \leq \tau_k \leq D_k$ and $f_k(\tau_k) = \tau_k - D_k$, $\tau_k \geq D_k$, where D_k is a soft deadline associated with customer i_k). Decision epochs occur at time instants T_k ($k = 1, ..., n$). The DSTSP consists of determining a policy such that at any epoch a decision is made: a) on the order in which pending customers have to be visited; b) on how to reposition the vehicle in anticipation of future demand. The latter issue includes deciding how long the vehicle has to wait at various locations along its route as well as whether to relocate the vehicle to some vertices outside the current route. The objective function to be minimized is the expected total penalty:

$$z = \sum_{k=1}^{n} E[f_k(\tau_k) \mid k] p_k \tag{1}$$

where $E[f_k(\tau_k) \mid k]$ is the expected penalty associated to customer i_k requiring service.

---

[*] Corresponding author's email: emanuele.manni@unile.it

Assume that the order in which customers are serviced is given ($i_1 \leq i_2 \leq ... \leq i_n$ without any loss of generality) and we develop exact and heuristic *waiting policies* for the DSTSP.
In Section 2 we discuss related literature, while in Section 3 we propose a lower bound on the expected penalty of an optimal policy, and in Section 4 we assess the expected penalty associated to two heuristic policies introduced by Mitrović-Minić and Laporte (2004) in a dynamic PDPTW setting. In Section 5 we determine the optimal waiting policy through a Markov decision process. Section 6 illustrates these procedures through a numerical example, while Section 7 presents some results on randomly generated instances. Finally, conclusions follow in Section 8.

2. LITERATURE REVIEW

The DSTSP falls under the broad category of real-time fleet management in which vehicle routes are built in an on-going fashion as customer requests, vehicle locations and travel times are revealed over the planning horizon. Early work on dynamic and stochastic routing problems is mainly focused on the Dynamic Routing and Dispatching Problem (DRDP). In DRDPs, the dynamic nature of the problem is not usually considered in the solution approach, because stochastic information about future requests is typically ignored. Rather, the research presented in these papers is characterized by algorithms reacting to new requests only once they have occurred, without any effort to anticipate future requests. Overviews of these problems can be found in Powell et al. (1995), Psaraftis (1988, 1995), Gendreau and Potvin (1998), and Ghiani et al. (2003).

Even the work that follows the DRDP work does not exploit stochastic information, although it implicitly accounts for future arrivals. By waiting at the last advanced request customer, Kilby et al. (1998) demonstrate the advantage achievable by implicitly anticipating future service requests. Larsen et al. (2002) consider a problem in which some service requests are known at the beginning of the planning horizon, but the locations of dynamic customers are unknown. Their objective is to minimize expected travel time.

Recent work aims at incorporating explicit waiting strategies. Particularly relevant to our paper is the article by Mitrović-Minić and Laporte (2004) in which four waiting strategies are examined for the dynamic *Pickup and Delivery Problem with Time Windows* (PDPTW). In the dynamic PDPTW, the presence of time windows allows the vehicles to wait at various locations along their routes. The authors show that an adequate distribution of this waiting time may affect the planner's ability to make good decisions at a later stage. Research in Mitrović-Minić et al. (2004) extends the earlier DPDPTW work by introducing a double- horizon heuristic, which has different objectives for the short-term and the long-term, minimizing route length and preserving flexibility so that future service requests can more easily be accommodated, respectively. Branke et al. (2005) considers waiting strategies in dynamic vehicle routing problems without time windows in contexts where the objective is to maximize the probability that an additional customer can be integrated into a fixed tour without violating time constraints. The authors propose several waiting strategies as well as an evolutionary algorithm to optimize the selected waiting strategy.

Another line of research examines dispatching and routing policies whose performance can be determined analytically if specific assumptions are satisfied. See, e.g., Bertsimas and van Ryzin (1991, 1993), where demands are distributed in a bounded area in the plane and arrival times are modeled as a Poisson process. The authors identify optimal policies both in light and heavy traffic cases. Papastavrou (1996) describes a routing policy that performs well both in light and heavy traffic, while Swihart and Papastavrou (1999) examine a dynamic pickup and delivery extension.

In a vehicle dispatching context, Powell et al. (1988) introduce a truckload dispatching problem, and Powell (1996) provides formulations, solution methods, and numerical results. In these papers, the authors use forecasts of future demand to determine which loads should be assigned to what vehicles in a truckload environment to account for forecasted capacity needs in the next period.

Ichoua et al. (2006) extend the work presented in Gendreau et al. (1999) to exploit probabilistic information about future arrivals. The heuristic allows a vehicle to wait in its current zone if the probability of a future request reaches a particular threshold. In Thomas and White (2004) a vehicle may serve several requests at a time and may wait for future demand both at a customer and non-customer locations. Not all requests have to be serviced and the objective function to be minimized is the expected value of a combination of travel costs, terminal costs, and revenue generated from a pickup. Bent and Van Hentenryck (2004) consider a vehicle routing problem where customer locations and service times are random variables which are realized dynamically during planned execution. They develop a multiple scenario approach which continuously generates plans consistent with past decisions and anticipating future requests. Similarly exploiting probabilistic information about future requests, van Hemert and La Poutré (2004) introduce the concept of fruitful regions. Fruitful regions represent clusters of known customer locations that are likely to require service in the near future. In the fashion of Bent and Van Hentenryck (2004), potential schedules are created by sampling fruitful regions. The authors then provide an evolutionary algorithm for determining when to move to one of the fruitful regions in anticipation of future service requests, whit the objective to maximize the number of customers served. The paper does not discuss about waiting. More recent is the work presented in Thomas (2007). Thomas extrapolates the structure for the optimal policy for one late-requesting customer to develop a real-time heuristic that performs well when the percentage of late-request customer is 25% or less. The author shows that a strategy that distributes waiting time across advance request customer locations works well as the percentage of late-request customers increases. The objective in this problem is to maximize the expected number of late-request customers served.

This paper makes the following contributions to the literature. First, we develop a lower bound on the expected penalty of an optimal policy for the DSTSP, as well as we assess the expected penalty associated to two heuristic policies. Then, we use a Markov decision process in order to determine the optimal waiting policy.

3. A LOWER BOUND ON THE OPTIMAL POLICY EXPECTED PENALTY

In this section a lower bound on the expected penalty of the optimal policy is computed under the hypothesis of perfect information, i.e., when all occurring requests are known at the beginning of the planning horizon.

Let $\sigma(r)$ be the r-th occurring request ($r = 1, ..., m \leq n$) and $\sigma(0) = 0$. Under perfect information an optimal policy can be devised straightforwardly. Indeed the vehicle should drive immediately to the first occurring customer, then if $t_{i_{\sigma(0)} i_{\sigma(1)}} < T_{\sigma(1)}$ wait until $T_{\sigma(1)}$, service $i_{\sigma(1)}$, then drive to $i_{\sigma(2)}$, etc. Let τ'_k be the service time of customer i_k under an optimal policy in case of perfect information. For every realization of the demand (such that request i_k occurs), the following inequalities are valid:

$$\tau_k \geq \tau'_k \geq \sum_{\substack{j=0,\ldots,m-1: \\ \sigma(j+1)\leq k}} t_{i_{\sigma(j)} i_{\sigma(j+1)}} \tag{2}$$

This relationship holds since τ'_k is the right-hand side of Eq. (1), plus the waiting times at the occurring customers $i_{\sigma(j)}$ ($\sigma(j) \leq k$). Consequently, assuming request i_k is issued, the expected value of the right-hand side of Eq. (2) is a lower bound on the expected value $E[\tau_k]$. The probability associated with $t_{\sigma(j)\sigma(j+1)}$ in this expected value computation is the probability that both $i_{\sigma(j)}$ and $i_{\sigma(j+1)}$ occur and no intermediate customer issues an order:

$p_{\sigma(j)} p_{\sigma(j+1)} (1 - p_{\sigma(j)+1}) (1 - p_{\sigma(j)+2}) \ldots (1 - p_{\sigma(j+1)-1})$, $\quad (\sigma(j + 1) < k)$

$p_{\sigma(j)} (1 - p_{\sigma(j)+1}) (1 - p_{\sigma(j)+2}) \ldots (1 - p_{\sigma(j+1)-1})$, $\quad (\sigma(j + 1) = k)$

where we have assumed $p_{\sigma(0)} = p_0 = 1$. Hence,

$$E[\tau_k] \geq \sum_{\substack{r=0,\ldots,k-2 \\ s=r+1,\ldots,k-1}} [t_{rs} p_r p_s \prod_{u=r+1}^{s-1} (1-p_u)] + \sum_{r=0,\ldots,k-1} [t_{rk} p_r \prod_{u=r+1}^{k-1} (1-p_u)] \tag{3}$$

Let L_k be the right-hand side of Eq. (3). Based on the Jensen inequalities (Birge and Louveaux, 1997) and the monotonicity of penalty functions $f_k()$, we can write:

$E[f_k(\tau_k)] \geq f_k(E[\tau_k]) \geq f_k(L_k)$

We then obtain the required lower bound:

$$LB = \sum_{k=1}^{n} f_k(L_k) p_k \tag{4}$$

It is worth noting that this lower bound requires $O(n^3)$ computations provided that functions $f_k()$ ($k = 1, ..., n$) can be evaluated in constant time.

4. HEURISTIC POLICIES

In this section we assess the expected penalty of two heuristic policies, called *Wait-First* (WF) and *Drive-First* (DF), introduced by Mitrović-Minić and Laporte (2004) in a purely dynamic setting, and we use such penalties as upper bounds for the optimal policy. The WF strategy requires an idle vehicle to wait at its current location until a new customer request arrives, while the DF strategy requires an idle vehicle to drive to its next *potential* customer. Under a WF policy, the time between the service of customers i_r and i_s ($s > r$) is equal to

$$A_{rs}(\tau_r) = \max\{T_s - \tau_r,\ 0\} + t_{i_r i_s}$$

provided that customer i_r is serviced at time τ_r and no intermediate request is issued. Indeed, $\max\{T_s - \tau_r, 0\}$ represents the waiting time at vertex i_r, whereas $t_{i_r i_s}$ is the travel time between the two vertices. The probability that the service time τ_k is equal to t under a WF policy can be computed through the iterative formula:

$$\Pr(\tau_k = t \mid k) = \sum_{l<k} \sum_{t'<t-t_{lk}} \left[\Pr\{\tau_l = t - A_{lk}(t') \mid l\} \prod_{r=l+1}^{k-1} (1 - p_r) \right] \tag{5}$$

with the initialization $Pr(\tau_0 = 0) = 1$. Once these probabilities have been computed we can calculate the expected value of the total penalty associated to the WF policy by applying the definition:

$$z_{WF} = \sum_{k=1}^{n} p_k \sum_t \Pr(\tau_k = t \mid k) \tag{6}$$

Under a DF policy, the previous procedure still applies, except that the computation of $A_{km}(\tau_k)$ is more elaborate. If the vehicle services customer i_k at time τ_k, then it moves along a shortest path from i_k to i_{k+1}. Let i'_r be the vertex where the vehicle is located at time instant T_r ($r = 1, ..., n$). If $\tau_k + t_{i_k i_{k+1}} \leq T_{k+1}$, then $i'_{k+1} = i_{k+1}$, where the vehicle waits for $\max\{T_{k+1} - \tau_k - t_{i_k i_{k+1}}, 0\}$ time instants. Otherwise, i'_{k+1} is a vertex along a shortest path from i_k to i_{k+1}, where the vehicle is diverted to i_{k+2}. In any case the vehicle then follows a shortest path from i'_{k+1} to i_{k+2}. By iteratively applying these procedures, vertices $i'_{k+1}, ..., i'_m$ are identified and $A_{rs}(\tau_r)$ is computed as

$$A_{rs}(\tau_r) = t_{i_r i'_{r+1}} + \max\{T_{r+1} - \tau_r - t_{i_r i_{r+1}},\ 0\} + \sum_{j=r+2}^{s} t_{i'_j i'_{j+1}} + \sum_{j=r+2}^{s} \max\{T_j - T_{j-1} - t_{i'_{j-1} i_j},\ 0\} \tag{7}$$

Formula (7) can be used to compute the expected penalty associated with the DF policy through relations Eq. (5) and Eq. (6).

5. A MARKOV DECISION PROCESS

We determine the optimal policy through a Markov decision process (MDP) which is a well-known approach for modeling and solving dynamic and stochastic decision problems. Much has been written about dynamic programming. Some recent books in this area are Puterman (1994), Bertsekas (1995), and Sennot (1999).

In our MDP, decisions are made at time instants T_k ($k = 1, ..., n$) (*decision epochs*) at which it becomes known whether or not customers need service. In particular, at time T_k we have to decide, in case the vehicle becomes idle before T_{k+1}, to which vertex the vehicle should be repositioned at time T_{k+1}.

A fundamental concept in MDPs is that of a *state*, denoted by s. The set S of all possible states is called the *state space*. The decision problem is often described as a controlled stochastic process that occupies a state at each point in time. The state should be a sufficient and efficient summary of the available information affecting the future of the stochastic process. In our problem, at every time instant T_k (*stage*) the state is represented by the triple (T_k, i_k^*, t_k^*), where i_k^* and t_k^* are, respectively, the vertex and time where the vehicle will become idle (i.e., with no pending requests) at the next epoch. Let S_k be the set of possible states at stage T_k. Obviously, $S = \cup_{k=0,\ldots,n} S_k$. The set S_0 contains a single state $s = (T_0, i_0, T_0)$ since the vehicle is idle at the depot at time $T_0 = 0$.

At any stage T_k, we first know whether customer i_k requires service, and we may then decide how to reposition the vehicle. Let $s = (T_k, i_k', t_k') \in S_k$ be the state before information about i_k becomes known (*chance state*). If this request occurs, i_k is appended to the route so that the state becomes (T_k, i_k'', t_k'') with $i_k'' = i_k$ and $t_k'' = t_k' + t_{i_k' i_k}$. Otherwise, the state remains unchanged ($i_k'' = i_k'$ and $t_k'' = t_k'$). These two states (T_k, i_k'', t_k'') are called the *decision states* associated with chance state s. Let $V^+(i,\Delta t)$ be the set of vertices that can be reached from $i \in V$ within no more than Δt (> 0) time units, and let $V^+(i,\Delta t) = \{i\}$ if $\Delta t \leq 0$. Hence, once it is known whether request i_k has occurred, we can reposition the vehicle to a vertex $i'_{k+1} \in V^+(i_k'', T_{k+1} - t_k'')$ where the vehicle will arrive at time $t'_{k+1} = t_k'' + t_{i''_k i'_{k+1}}$. Consequently, at stage T_{k+1}, the state may be chosen from the subset $\{(T_{k+1}, i'_{k+1}, t'_{k+1}) : i'_{k+1} \in V^+(i_k''\ T_{k+1} - t_k''),\ t'_{k+1} = t_k'' + t_{i''_k i'_{k+1}}\} \subseteq S_{k+1}$. It is worth noting that, if the k-th request i_k occurs, the service time τ_k of customer i_k is then equal to $t_k'' = t_k' + t_{i_k' i_k}$. Let z_s be the expected penalty $p_k f_k(\tau_k)$ associated with the service of customer i_k if the vehicle is in state $s \in S_k$ and let Z_s be the total expected penalty associated to an optimal policy servicing customers $\{i_k, i_{k+1}, ..., i_n\}$ starting from state $s \in S_k$. Moreover, let $\Sigma(s)$ be the set of successors of a state s, i.e., those states s reachable through a single transition from s. We can now outline our Markov decision process.

Step 0 (Initialization).

$S_0 = \{(T_0, i_0, T_0)\}$.

$z_s = 0$ *for* $s \in S_0$.

Step 1 (Forward Computation).

for $k = 1$ ***to*** n

Determine the set of feasible states S_k.

for *any state* $s = (T_k, i_k', t_k') \in S_k$ ***do begin***

Determine the state transition associated with the occurrence of customer i_k *and compute the associated* i_k'' *and* t_k''.

If the k*-th request* i_k *occurs, compute* τ_k.

Compute $z_s = p_k f_k(\tau_k)$.

end

Set $Z_s = z_s$ *for any state* $s \in S_n$.

Step 2 (Backward Computation).

for $k = n - 1$ ***to*** 0

for *any state* $s = (T_k, i_k'', t_k'') \in S_k$ ***do begin***

Determine the decision associated to state s as the transition from s to state:

$$s'^{(*)} = \arg\min_{s' \in \Sigma(s)} Z_{s'}.$$

Then, set $Z_s = Z_{s'^{(*)}}$.

end

for *any state* $s = (T_k, i_k', t_k') \in S_k$ ***do begin***

Determine $Z_s = z_s + p_k(Z_{s'}) + (1-p_k)Z_{s''}$, *where* s' *and* s'' *are the two decision states associated to the occurrence or non-occurrence of the* k*-th request provided the vehicle is in state s;*

end

Z_{T_0, i_0, T_0} *represents the expected cost of an optimal waiting policy.*

The number of states is bounded above by $O(n|V|\overline{T})$, where $\overline{T}$ is an upper bound on the service time of customer i_n in an optimal policy (e.g., $\overline{T} = \sum_{r=0}^{n-1} t_{i_r i_{r+1}}$). Hence, the above MDP requires $O(n|V|\overline{T}^2)$ time since $O(\overline{T})$ operations are required for every state.

6. A NUMERICAL EXAMPLE

We now illustrate the above procedures on a numerical example. Let $G(V,A)$ be the graph represented in Figure 1, where $V' = \{1,2,3\}$ and $V = \{0\} \cup V' \cup \{a,b,c,d,e\}$. With each vertex in V' are associated the corresponding arrival time and probability, and with each arc is associated its traversal time. Penalties $f_k(\tau_k)$ ($k = 1,2,3$) are constituted by customers' *waiting times*, i.e., $f_k(\tau_k) = \tau_k - T_k$, $\tau_k \geq T_k$.

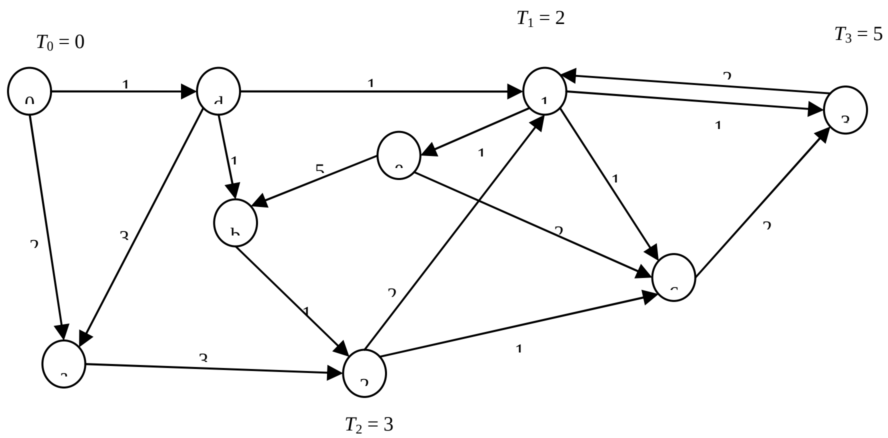

Figure 1. Sample network.

Firstly, we compute a lower bound on the expected penalty of an optimal policy. The right-hand side of inequalities (3) are:

$$L_1 = t_{01} p_0 = 2$$
$$L_2 = t_{02}(1-p_1) + (t_{01}+t_{12})p_1 = 1.8 + 3.6 = 5.4$$
$$L_3 = (t_{01}+t_{12}+t_{23})p_1 p_2 + (t_{01}+t_{13})p_1(1-p_2) + (t_{02}+t_{23})(1-p_1)p_2 + t_{03}(1-p_1)(1-p_2)$$
$$= 4.32 + 0.12 + 3.24 + 0.18 = 7.86\ .$$

Hence, formula (4) provides a lower bound equal to:

$$LB = 0 + f_1(2)\cdot 0.4 + f_2(5.4)\cdot 0.9 + f_3(7.86)\cdot 0.7$$
$$= 0 + 0\cdot 0.4 + 2.4\cdot 0.9 + 2.86\cdot 0.7 = 4.16\ .$$

Secondly, we determine an optimal policy through a Markov decision process. In Figure 2 are reported, for each stage, the associated chance and decision states (represented by circles and squares, respectively). Labels z_s and Z_s are shown in Figure 3 separated by a semi-colon. The expected cost of an optimal policy is equal to $Z_{T_0,\, i_0,\, T_0}$= 6.32. Figure 4 illustrates the optimal policy. Figures 5 and 6 depict the Wait-First and Drive-First policies which yield a total expected penalty equal to 9.23 and 9.95, respectively.

7. COMPUTATIONAL RESULTS

In addition, we have solved a number of randomly generated instances on a PC with a Pentium IV processor clocked at 2.8 GHz. Two sets of 50 instances were generated as follows. First, a graph $G(V,A)$ was generated by randomly choosing points in a 100×100 square. Then n (≤ $|V|$) customers were chosen at random and an order of visit was determined in a random fashion. In our experiments, we choose $|V|$=50 and n=20, 25, 30, 35, 40. Hence, request occurrence times were chosen as realizations of a Poisson process with $\lambda = 1$ in the first set and $\lambda = 2$ in the second set. Request probabilities were chosen as uniform random numbers in [0, 1]. Computational results reported in Table 1 indicate that the average lower bound gap is 12.33% for $\lambda = 1$ and 2.42% for $\lambda = 2$. The Markov Decision Process was able to determine the optimal policy always within 1500 seconds for the first set and within 3000 seconds for the second set, while the number of states was always less than 40000 and 60000 for $\lambda = 1$ and $\lambda = 2$, respectively.

Moreover, in order to compare the results of the heuristic policies with the optimal policy, we have evaluated the average performance ratio computed as the heuristic solution value divided by the optimal policy value. We have observed an average performance ratio of the WF heuristic equal to 1.22 for $\lambda = 1$ and to 1.13 for $\lambda = 2$. Similarly, the average performance ratio of the DF heuristic was 1.02 for $\lambda = 1$ and 1.01 for $\lambda = 2$. These values indicate that the DP policy performs better than the WF policy, thus empirically showing that moving towards the next customer in anticipation of its possible demand may give some benefit.

In addition, results reported in Table 1 show that the instances with $\lambda = 1$ were more difficult to solve, resulting in larger gaps for both the lower and upper bounding techniques and in larger computing times. This can be explained by the fact that larger arrival rates give rise to a busier vehicle. For every test set, the DF policy outperformed the WF policy both in terms of solution quality and computing time.

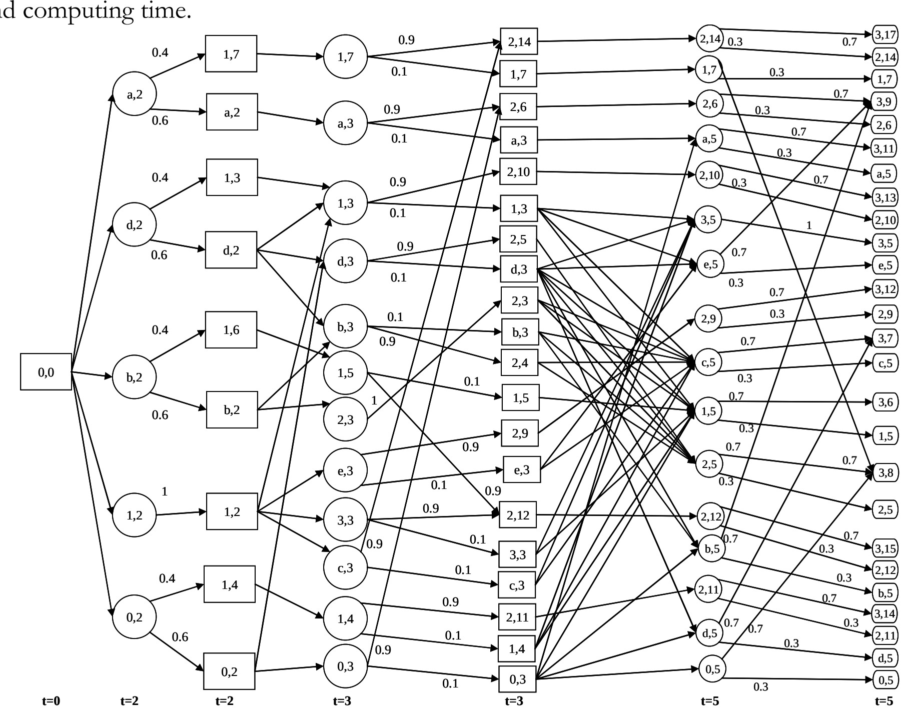

Figure 2. State space of the sample problem.

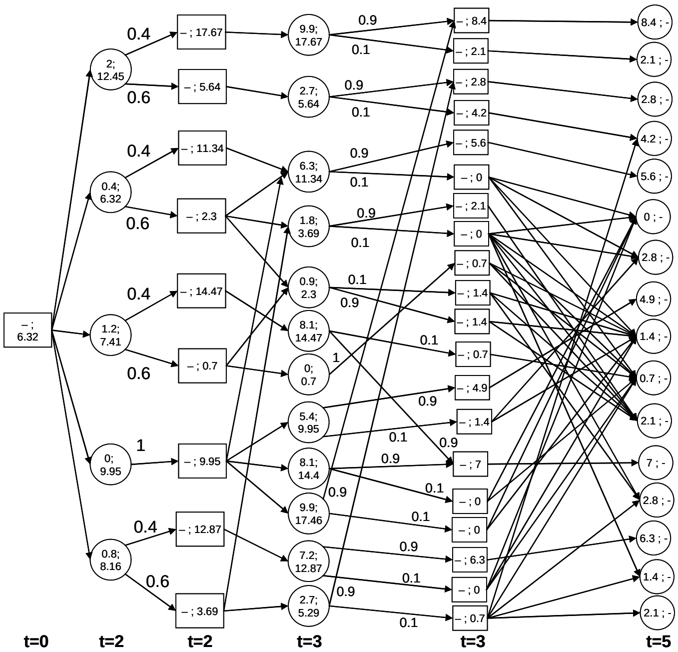

Figure 3. Expected penalties in the sample problem.

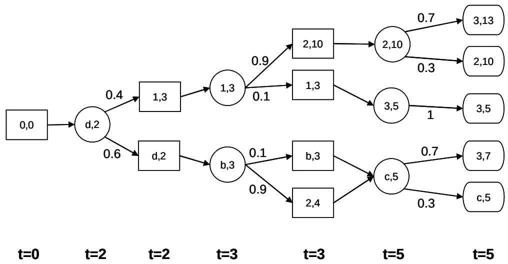

Figure 4. Optimal policy.

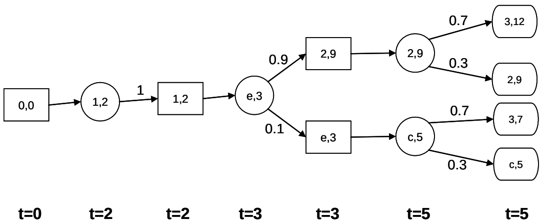

Figure 5. Drive-First policy.

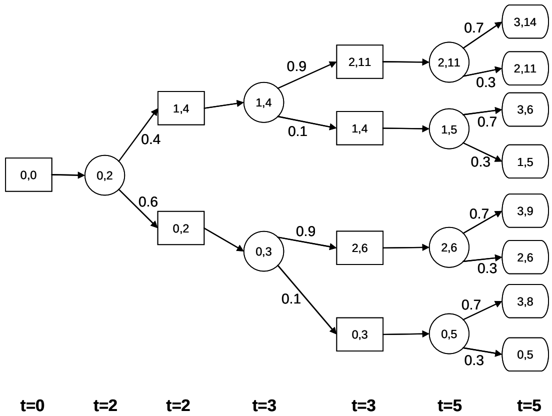

Figure 6. Wait-First policy.

Table 1. Computational results.

λ	Customers	LB gap (%)	MDP States	MDP time (sec)	WF gap (%)	WF time (sec)	DF gap (%)	DF time (sec)
1	20	13.12	8923	120.7	27.41	61.8	2.01	13.5
	25	19.80	12527	193.4	26.48	156.8	2.41	77
	30	7.62	28267	753	20.13	463.8	0.95	169
	35	2.88	33524	1075.5	15.04	807.8	1.19	457.6
	40	18.22	38901	1383	21.47	1135.6	2.07	858.9
2	20	1.67	9430	118.5	14.90	58.4	0.39	17.9
	25	1.17	14740	222.8	13.11	171.4	0.29	78.7
	30	3.96	22930	431	12.34	337.4	0.28	189.9
	35	3.83	30211	777.5	15.97	741.5	1.50	543.4
	40	1.47	58968	2361.1	10.18	2262.3	0.51	1910.1

8. CONCLUSIONS

In this article we have introduced the *Dynamic and Stochastic Traveling Salesman Problem* (DSTSP), and we have examined exact and heuristic waiting policies for it under the hypothesis that a probabilistic characterization of the customer requests is available. We have developed a Markov Decision Process (MDP) in order to determine the optimal policy, as well as a lower bound based on the availability of perfect information. We have assessed the value of two heuristic waiting strategies (previously introduced in a purely dynamic setting) against this lower bound. We have tested both the MDP and the heuristic policies on two different sets of randomly generated instances, and we have compared the results of the heuristic policies with the optimal policy, by means of the average performance ratio. The experiments have shown that a heuristic waiting strategy requiring an idle vehicle to drive to its next potential customer outperforms a waiting strategy in which an idle vehicle is requested to wait at its current location until a new customer request arrives.

Our results are based on a number of assumptions that should gradually be removed: a) the hypothesis that request occurrence times $T_1 \leq T_2 \leq \ldots \leq T_n$ are sorted in non-decreasing order; b) the assumption that the order of service is given; c) the hypothesis that a customer request may arise at a single time instant. These extensions are left as a future research. In addition, when removing the over mentioned hypothesis, the MDP will not be able to handle instances with many customers. Thus, a heuristic will be needed to account for this aspect.

ACKNOWLEDGMENTS

This research was partially supported by the Ministero dell'Istruzione, dell'Università e della Ricerca Scientifica (MIUR), and by Regione Puglia (Progetto Esplorativo e-FLEET). This support is gratefully acknowledged. Moreover, we thank three anonymous referees for their useful comments, which helped us to improve the paper.

REFERENCES

1. Bent, R. and Van Hentenryck, P. Scenario-based planning for partially dynamic vehicle routing with stochastic customers. *Operations Research*, 52:977-987, 2004.
2. Bertsekas, D.P. *Dynamic Programming and Optimal Control.* Athena Scientific, Belmont, MA, 1995.
3. Bertsimas, D.J. and Van Ryzin G. A stochastic and dynamic vehicle routing problem in the euclidean plane. *Operations Research*, 39:601-615, 1991.
4. Bertsimas, D.J. and Van Ryzin, G. Stochastic and dynamic vehicle routing problem in the euclidean plane with multiple capacitated vehicles. *Operations Research*, 41:60-76, 1993.
5. Birge, J.R. and Louveaux, F. *Introduction to Stochastic Programming.* Springer-Verlag, New York, 1997.
6. Branke, J., Middendorf, M., Noeth, G., and Dessouky, M. Waiting strategies for dynamic vehicle routing. *Transportation Science*, 39:298-312, 2005.
7. Gendreau, M. and Potvin, J.-Y. Dynamic vehicle routing and dispatching. In Crainic, T.G. and Laporte, G. editors, *Fleet Management and logistics*, pages 115-126. Kluwer, Boston, 1998.
8. Gendreau, M., Guerten, F., Potvin, J.-Y., and Taillard, É. Parallel tabu search for real-time vehicle routing and dispatching. *Transportation Science*, 33:381-390, 1999.
9. Ghiani, G., Guerriero, F., Laporte, G., and Musmanno, R. Real-time vehicle routing: Solution concepts, algorithms and parallel strategies. *European Journal of Operational Research*, 151:1-11, 2003.
10. Ichoua, S., Gendreau, M., and Potvin, J.-Y. Exploiting knowledge about future demands for real-time vehicle dispatching. *Transportation Science*, 40:211-225, 2006.
11. Kilby, P., Prosser, P., and Shaw, P. Dynamic VRPs: A study of scenarios. Technical Report APES-06-1998, Department of Computer Science, Strathclyde University, 1998.
12. Larsen, A., Madsen, O.B.G., and Solomon, M.M. Partially dynamic vehicle routing - models and algorithms. *Journal of the Operational Research Society*, 53:637-646, 2002.

13. Mitrović-Minić, S. and Laporte, G. Waiting strategies for the dynamic pickup and delivery problem with time windows. *Transportation Research Part B*, 38:635-655, 2004.
14. Mitrović-Minić, S., Krishnamurti, R., and Laporte, G. Double-horizon based heuristics for the dynamic pickup and delivery problem with time windows. *Transportation Research Part B*, 38:669-685, 2004.
15. Papastvrou, J.D. A stochastic and dynamic routing policy using branching processes with state dependent migration. *European Journal of Operational Research*, 95:167-177, 1996.
16. Powell, W.B. A stochastic formulation of the dynamic assignment problem, with an application to truckload motor carriers. *Transportation Science*, 30:195-219, 1996.
17. Powell, W.B., Sheffi, Y., Nickerson, K.S., Butterbaugh, K., and Atherton, S. Maximizing profits for North American Van Lines' truckload division: A new framework for pricing and operations. *Interfaces*, 18: 21-41, 1988.
18. Powell, W.B., Jaillet, P., and Odoni, A. Stochastic and dynamic networks and routing. In Ball, M.O., Magnanti, T.L., Monma, C.L., and Nemhauser, G.L. editors, *Network Routing*, volume 8 of *Handbooks in Operations Research and Management Science*, pages 141-295. Elsevier Science, Amsterdam, 1995.
19. Psaraftis, H.N. Dynamic vehicle routing problems. In Golden, B.L. and Assad, A.A. editors, *Vehicle Routing: Methods and Studies*, pages 223-248, Amsterdam, 1988. North-Holland.
20. Psaraftis, H.N. Dynamic vehicle routing: Status and prospects. *Annals of Operations Research*, 61:143-164, 1995.
21. Puterman, M.L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, New York, 1994.
22. Sennot, L.I. *Stochastic Dynamic Programming and the Control of Queueing Systems*. Wiley, New York, 1999.
23. Swihart, M.R. and Papastavrou, J.D. A stochastic and dynamic model for the single-vehicle pick-up and delivery problem. *European Journal of Operational Research*, 114:447-464, 1999.
24. Thomas, B.W. Waiting strategies for anticipating service requests from known customer locations. *Transportation Science*, 41:319-331, 2007.
25. Thomas, B.W. and White III, C.C. Anticipatory route selection. *Transportation Science*, 38:473-487, 2004.
26. Van Hemert, J.I. and La Poutré, J.A. Dynamic routing with fruitful regions: Models and evolutionary computation. In Yao, X., Burke, E., Lozano, J.A., Smith, J., Merelo-Guervós, J.J., Bullinaria, J.A., Rowe, J., Tino, P., Kabáan, A., and Schwefel, H.-P. editors, *Parallel Problem Solving from Nature VIII*, pages 690-699. Springer-Verlag, 2004.