



## **CENTRUM Católica's Working Paper Series**

**No. 2012-09-0011 / September 2012**

### **Coverage-Based Service Vehicle Routing when only some Tasks are Known in Advance**

José Antonio Larco  
Rommert Dekker  
Uzay Kaymak

**CENTRUM Católica – Pontificia Universidad Católica del Perú**

Working papers are in draft form. This working paper is distributed for purposes of comment and discussion only. It may not be reproduced without permission of the author(s).

# Coverage-based service vehicle routing when only some tasks are known in advance

Jose Antonio Larco<sup>1</sup>, Rommert Dekker<sup>2</sup>, Uzay Kaymak<sup>1</sup>

<sup>1</sup>CENTRUM Catlica, Pontificia Universidad Catlica del Per, Jr. Aloma Robles 125, Santiago de Surco, Peru, (corresponding author) jlarco@pucp.pe, Tel. +51 1 626 7100 Ext 7185

<sup>2</sup>Econometric Institute, Erasmus University Rotterdam P. O. Box 1738, 3000 DR, Rotterdam, The Netherlands, rdekker@eur.nl,

<sup>3</sup>School of Industrial Engineering, Eindhoven University of Technology, P.O. Box 513, 5600 MB, Eindhoven, the Netherlands, j.a.larco.martinelli@tue.nl (corresponding author), u.kaymak@ieee.org Tel. +31 10 40 247 4984 Fax +31 10 408 90 14

---

## Abstract

In this paper we deal with a common problem found in the operations of security and preventive/corrective maintenance services: that of routing a number of mobile resources (vehicles) to perform foreseen and unforeseen tasks during a shift. We define the (Stochastic Service Team Orienteering Problem) SSTOP as the problem of making a routing strategy to maximize the expected weighted number of tasks served within the specified time-windows. To obtain solutions to this problem, we propose to solve successively the Coverage Team Orienteering Problem with Time Windows (C-TOPTW). The C-TOPTW considers information related to known tasks and also information about the arrival process of new unforeseen tasks. We find that the information about the arrival process of new unforeseen tasks is of value in generating routes for vehicles to maximize the expected proportion of tasks accomplished within the specified time windows.

*Key words:* Routing, location, reliability, distributed services.

---

## 1 Introduction

The domain of geographically distributed security services and preventive/corrective maintenance services involves the task of routing mobile resources (e.g. field engineers, security guards) over a number of sites so that various

services can be performed. While the clients are mainly concerned with receiving timely and reliable services so that their security/maintenance problems are solved effectively, the operational managers of the service providers are mainly concerned with offering competitive and profitable services. In a competitive environment, where the firms choose to differentiate themselves from competitors by focusing on service quality, rather than on costs, responsiveness is one of the most important quality dimensions of service operations. Responsiveness in security/maintenance contexts means that the clients' security/maintenance concern is being dealt within specified time windows.

The time windows depend on the type of service/task to be performed. We distinguish between two kinds of tasks: foreseen tasks which are known in advance at the start of a shift and unforeseen tasks which are only known at the time an alarm is triggered. While foreseen tasks involve routine activities such as verification of alarm systems or preventive maintenance, unforeseen tasks are emergency related activities addressing critical situations such as a robbery or a failure of an important piece of equipment. As a result, in the case of foreseen tasks, the latest allowable starting times are specified in advance upon request of the client, whereas in the case of unforeseen tasks, these are defined by a given standard response-time which is a service-guarantee that can be easily understood (for a discussion on service guarantees see Hart, 1988).

Operational managers and/or dispatchers have the task of ensuring that the routing decisions, which are taken before the start of a shift, and the re-routing decisions are such that they allow servicing as many foreseen and unforeseen tasks as possible within the specified time-windows and within the shift's length. Although we acknowledge that there are several sources of uncertainty in the distributed services domain, such as non-deterministic travelling times or task service durations, we restrict our study to the main source of uncertainty: that of the release of unforeseen tasks. Hence, in this paper we study the challenge of (re-)routing when dealing with (un)foreseen tasks so as to meet specifications of service contracts and offer reliable and timely services. Un-foreseen tasks generally involve emergencies where fast reactions are needed. These are considered of higher priority than foreseen tasks.

To deal with incomplete information in terms of the tasks known in advance we introduce a new problem that we name the Stochastic Service Team Orienteering Problem (SSTOP). In contrast to the multiple traveling salesman problem with time windows (m-TSPTW) (Mitrovic-Minic, and Krishnamurti, 2006) the SSTOP deals with two types of tasks and recognizes that it is not always possible to guarantee that all tasks will be served within the time windows when unforeseen tasks are released simultaneously. Hence, the objective of the SSTOP is to maximize the expected number of tasks that are serviced

by mobile resources within the specified contractual deadlines. Consistent with the vehicle routing literature, we use the general term “vehicles” to refer to the mobile resources within vehicle.

The SSTOP problem may be stated as follows: a planner must decide on the sequence of tasks a number of vehicles must be serviced within a given shift. The tasks are located in sites on a plane and each task has time-windows indicating the time at which a task is allowed to start to be serviced. In addition, emergency events may occur within the shift in any of a set of candidate set of sites according to a Poisson process with sampling. Each emergency event generates an unforeseen task and such task can only start to be serviced within a certain amount of time counting from the time the emergency event occurs. Each task has an associated weight indicating its importance, where unforeseen tasks have a higher weight than foreseen tasks.

To solve the SSTOP, we propose to solve successively a static bi-objective problem. The first criterion uses complete information about known tasks. It aims to serve as many known weighted tasks as possible within the given time-windows and a given shift by using a variant of the Team Orienteering Problem (Golden et al. 1996). For a recent review on the Orienteering Problem and its variants see Vansteenwegen et al. (2011). The second criterion uses the incomplete information about unforeseen tasks. It is assumed that the stochastic process that generates these tasks is known. It recognizes that although the arrival time of unforeseen tasks is not known a-priori, it is important to know the risk that a certain set of vehicle routes implies. To evaluate such a risk, we extend Church and Reville’s (1974) Maximum Coverage Location Problem (MCLP) where it is possible to identify which sites are covered so as to serve potential requests of unforeseen tasks on time.

In the original version of the MCLP, the problem is applied to static coverage situations where emergency vehicles (e.g. ambulances) return to their bases immediately after serving *every* unforeseen task. Hence, only the fixed location of the base is relevant to identify which sites are covered. In contrast, the extended MCLP version we propose is applied to dynamic coverage situations where vehicles have to modify their routes to accommodate new unforeseen tasks without having to return to the base, implying that the location of each vehicle at each point in time is relevant for identifying the sites being covered and therefore must be updated. The location of vehicles can now be identified using the tracking capabilities of GPS that is already being used in the geographical distributed services domain. By incorporating the proposed extended MCLP model as the second criterion, we investigate the added value of the information that the arrival process can bring for the design of more reliable routes. To our knowledge, there is no available literature that deals with the combination of routing and location models where the location refers not to static bases or sites but to the vehicles themselves.

A number of related problems exist in the literature. First, we differ from Weintraub (1999) that also deals with unforeseen tasks (but without foreseen tasks) in that we focus on the fulfillment of contracts that define time windows and maximum response times rather than on simply minimizing response times for all tasks. By considering also unforeseen tasks of a generally higher priority than foreseen tasks, the routes will be modified accordingly such that the risk of not being able to serve an unforeseen task is minimized. Second, our situation is different from the Orienteering Problem with stochastic profits (Ilhan et al., 2008) in which a continuous distribution of profits is associated to each site and that it is independent of the time at which it is served. Another related problem is that of the Probabilistic Traveling Salesman Problem (Jaillet, 1985) where a route is designed a-priori and only a subset of the sites will eventually be needed to be served by a probabilistic process. It differs from our problem in that we require a service time guarantee in servicing the unforeseen tasks and that a number of foreseen tasks are also known in advance. Other variants of stochastic Orienteering problems exist where the length of the routes are stochastic instead of the profits and thus a balance must be struck between the probability that a tour remains feasible and the objective value of such a feasible tour (Evers et al., 2011). In our case, to service an unforeseen task is only profitable for a given time window (between the release time of the unforeseen task and the maximum allowable time to be able to serve it).

This paper is organized as follows. In Section 2, we describe the SSTOP and its assumptions. In Section 3, we outline the multi-criteria problem; while in Section 4 we provide a heuristic to solve it. In Section 5, we present the set-up of a series of simulation experiments used to evaluate the effectiveness in using coverage considerations in solving the SSTOP. The results of these experiments are reported and discussed in Section 6. Conclusions of the paper are given in Section 7.

## 2 The Stochastic Service Team Orienteering Problem (SSTOP)

The Stochastic Service Team Orienteering Problem consists of a number of sites where tasks need to be served. These tasks may be of foreseen type (i.e. known in advance at the start of a shift) or of unforeseen tasks (i.e. tasks may be originated at sites during the shift where the scheduler has no knowledge about them before they are originated). Thus, in the SSTOP, we start the shift with a schedule for foreseen tasks and then this schedule can be modified at any moment as soon as new (unforeseen) tasks become available. In the following, we describe the dynamic characteristics of the SSTOP as a problem with incomplete information and then provide a precise definition of the SSTOP. To conclude, we illustrate the dynamics of the SSTOP with an

example.

## 2.1 General assumptions

We assume that the traveling times between sites and two points in the plane are deterministic and that the durations of both foreseen and unforeseen tasks are known beforehand. Furthermore, we assume that all sites may be reached from any other location.

During a shift of length  $TT$ , all the vehicles start at a base and have to return to the base at the end of the shift. Moreover, we assume that a single dispatcher schedules each vehicle and that vehicle is able to serve any task. In this problem setting, the fixed information of the SSTOP (i.e. known at any point in time) is given by the following:

- $I$  : the set of sites,
- $K$  : the set of vehicles,
- $\delta$  : the system-wide standard response time indicating the time available to start serving an unforeseen task since its release,
- $P_i$  : is the probability that if an unforeseen task is generated, such unforeseen task will be located in site  $i_1 \in I$ ,
- $\mu$  : the fixed duration of any unforeseen tasks generated,
- $\Delta_n$  : the fixed duration of task  $n$ , when the task is of unforeseen type  $\Delta_n = \mu$ , otherwise  $\Delta_n$  is a fixed value known at the start of the shift.

## 2.2 Tasks

The services delivered by vehicles are a series of tasks. Every task  $n \in N$  has the following tuple associated to it:  $\wp_n = (L(n), w_n, r_n, \Delta_n, e_n, l_n)$  where  $L(n) = i : i \in I$  is the site at which the task is to be performed,  $w_n$  is the importance of servicing the task,  $r_n$  is the release time at which the task is known to the dispatcher and  $e_n$  and  $l_n$  are the earliest and latest allowable starting times respectively. Note that the release time allows to distinguish between a foreseen and an unforeseen task. If  $r_n = 0$  the task  $n \in F$  is of foreseen type and if  $r_n > 0$  the task  $n \in U$  is of unforeseen type (note that  $F \cap U = \emptyset$ ). The location of a site is mapped to an Euclidean space with a function  $LS$  such that  $LS : I \mapsto \mathbb{R}^2$ .

Without loss of generality, we assume that unforeseen tasks are generated with a Poisson process with intensity  $\lambda$ , that the duration of these are known and constant  $\mu$  (i.e  $\Delta_n = \mu, n \in U$ ) and that the importance of unforeseen tasks is assumed fixed in advance with a constant value of  $w_E \in \mathbb{R}^+$ , thus,  $w_n = w_E, n \in U$ . Note that given the importance of servicing unforeseen tasks generated by emergency events, in general  $w_E > w_n$  if  $n \in F$ .

Using a poisson sampling process, we may introduce heterogeneity in the relative probability that an unforeseen task will be generated in a given site. For such purpose, we introduce  $v_i$  to denote the probability that an unforeseen task will be originated at site  $i \in I$  if an unforeseen task is generated. The starting time windows are also derived differently for each type of task. While for foreseen tasks,  $e_n$  and  $l_n$  are determined a-priori, for unforeseen tasks these are determined with the following relations:  $e_n = r_n$  and  $l_n = r_n + \delta$ .

### 2.3 Valid schedules

In the SSTOP, the main task of a central dispatcher is that of defining valid schedules for each of the vehicles available. However, as new information is available each time a new unforeseen task is released, new schedules need to be devised each time the set of unforeseen tasks is updated with additional unforeseen tasks that are originated and become known to the scheduler. We use the index  $j \in J$  to denote the dates at which the list of already released unforeseen tasks is updated. To reflect the incomplete information nature of the SSTOP problem, we define  $TK_j$  as the set of tasks for which service has not started at or before the  $j$ -th unforeseen task has been released. Note we use  $j = 0$  to denote the start of a shift before any unforeseen task has been released (i.e. there has been no emergency event yet). A valid schedule at instant  $j$  is then defined as follows.

**Definition 1:** *A valid schedule for a vehicle  $k \in K$  at a given evaluation instant  $j \in J$ , is defined as an ordered set of known pending tasks denoted by  $\Pi_j^{(k)} \subseteq TK_j$  to be accomplished by the vehicle that fulfills the following dynamic conditions.*

- (1) *The vehicles start in the home base (common to all vehicles, i.e.  $i=b$ ) at the start of their shift and otherwise a known location evaluated at any other time before the end of the shift according to previously scheduled routes. The vehicles must end their shift at the base (i.e.  $i=b$ ); and can only service tasks that allow them to return to the base before the termination of their shift at  $t=TT$ .*
- (2) *Each task included in the schedule must start within the service starting time windows:  $[e_n, l_n]$ .*
- (3) *Once a task is executed, the vehicle re-allocates immediately to service another task or returns to the base. If the vehicle arrives at a site before the corresponding earliest allowable starting time, it is assumed that it waits just outside the premises of the site until the earliest allowable starting time,  $e_n$  time at which it starts servicing the task.*
- (4) *If a vehicle is en-route to service an unforeseen task, it can not be re-allocated to another task until it finishes servicing the unforeseen task.*

- (5) *If a vehicle is servicing a foreseen or an unforeseen task, such a vehicle can not be re-allocated to serve another task until the vehicle finishes servicing its current task.*

The first two conditions enforce that vehicles start servicing tasks within the specified time windows so that the vehicles are able to return to the base before the end of the shift. The third condition is used for simplification purposes where a sequence of tasks fully determines the actual position of vehicles at given time instants. The last two conditions assure that immediate re-allocation is allowed only in the case a vehicle is traveling to serve a foreseen task so that the clients remain unaware of service interrupts.

Since we assume that the time to transfer between sites is pre-determined, it is possible to know in advance if a released task can be started within the given time window. Note that the SSTOP's objective only counts the tasks served within the specified time-windows. Tasks that are known to be impossible to be serviced within time windows are never scheduled and/or serviced.

#### 2.4 Information set and strategy

At the start of the shift or when an unforeseen task is released, updated information is available to the dispatcher to generate new schedules. The information set,  $\mathcal{I}_j$ , at evaluation instant  $j \in J$ , includes the updated information of the tasks available for servicing as well as information about the current situation of every vehicle  $k \in K$ .

$$\mathcal{I}_j = (TK_j, \bigcup_k VL_j^{(k)}, \bigcup_k VS_j^{(k)}, \bigcup_k LT_j^{(k)} \bigcup_k LL_j^{(k)}) \quad (1)$$

We use the following notation.

- $VL_j^{(k)}$  : the actual location of vehicle  $k \in K$  in the Euclidian plane at evaluation instant  $j \in J$ ,
- $VS_j^{(k)}$  : identifies the state at which a vehicle  $k$  is at evaluation instant  $j \in J$  such that:
  - $VS_k^{(k)} = 1$  if vehicle  $k$  is traveling to serve an unforeseen task,
  - 2 if vehicle  $k$  is currently servicing a (un)foreseen task,
  - 3 if vehicle  $k$  is traveling to serve a foreseen task, traveling back to the base, or waiting at a site to serve a foreseen task,
- $LT_j^{(k)}$  : the first time at which a dispatcher can reschedule a vehicle  $k$  given the current time at evaluation instant  $j \in J$  in accordance with the conditions defined in Definition 1 ,



$LL_j^{(k)}$  : the location of the vehicle  $k \in K$  at time  $t = LT_k^{(j)}$  in the Euclidian plane. If the vehicle is traveling to serve a unforeseen task, i.e.  $VS_j^{(k)} = 1$  then, the location will be that of the unforeseen task the vehicle is traveling to, i.e.  $LL_j^{(k)} = LS(L(n))$ . Otherwise, the location of the vehicle would be the same as the location at instant  $j \in J$ , i.e.  $LL_j^{(k)} = VL_j^{(k)}$ , the site in the Euclidian space at which the vehicle  $k \in K$  is traveling to at evaluation instant  $j \in J$ . Alternatively, if the vehicle is not traveling, it identifies the location of the site at which the vehicle is at evaluation  $j \in J$ .

We define a strategy as a mechanism that generates valid schedules for each vehicle upon the occurrence of each event (i.e. a request to serve a new unforeseen task). More formally, by defining a system schedule  $\Gamma_j$  as the collection of all valid vehicle schedules  $\Pi_j^{(k)}$  at a given evaluation instant, i.e.  $\Gamma_j = \bigcup_{k \in K} \Pi_j^{(k)}$ , we define a strategy as follows:

**Definition 2:** A strategy denoted by  $s$  is a function that maps the information set at a given evaluation instant  $j \in J$  to a system schedule:  $s : \mathcal{I}_j \rightarrow \Gamma_j$ .

## 2.5 Performance evaluation

To evaluate the quality of the schedules generated by a given strategy during the realization of a certain shift, we introduce the concept of weighted fulfilment yield which is in-line with the service operational managers' objective of servicing as many tasks as possible while recognizing that not all the tasks are of equal importance.

**Definition 3:** The weighted fulfilment yield,  $\Psi$ , is a random variable defined as the weighted proportion of tasks  $n \in N$  (weighted by  $w_n$ ) that are completely serviced within their time-windows during a shift  $[0, TT]$ .

As a problem with incomplete information, the objective of the SSTOP is to maximize the *expected* weighted fulfilment yield  $\gamma$  over all possible realizations of unforeseen tasks in a shift. Therefore, we define the SSTOP as follows:

**Definition 4:** The objective of the SSTOP is to devise a strategy  $s \in S$  for the allocation and re-allocation of vehicles to tasks so that the strategy maximizes the *expected* weighted fulfilment yield  $E(\Psi)$  during a shift.

Recall that the expected weighted fulfilment yield,  $E(\Psi)$ , considers that unforeseen tasks arise following a Poisson process with intensity  $\lambda$  and that the

Table 2

Initial information available of basic example.

Task index {Site Location}	1{a}	2{b}	3{c}	4{d}	5{e}	6{f}
X-coordinate (km)	-1.0	1.0	-1.0	1.0	-1.0	1.0
Y-coordinate (km)	1.0	1.0	0.0	0.0	-1.0	-1.0
Earliest allowable starting time (hr)	2.0	1.0	1.0	5.0	0.5	2.5
Latest allowable starting time (hr)	5.0	7.0	9.5	8.5	3.5	4.5
Duration shift (hr)	12.0					
Duration of foreseen $\Delta_n$ {unforeseen $\mu$ } tasks (hr)	1.0{1.2}					
Relative importance of foreseen {unforeseen} tasks	16.7%{100%}					
Mean rate of unforeseen tasks (# tasks in $[0, TT - \mu]$ )	1					
Speed of vehicles (km/hr)	1					
Response time standard $\delta$ (hr)	1.5					

probability that the unforeseen task is located at a given site is given by  $P_i$  for every  $i \in I$ . The constraints of this incomplete information optimization problem are stated implicitly by constraining possible solutions to strategies that generate *only* valid schedules.

## 2.6 Risks inherent to routes

To illustrate the SSTOP, consider the next example where the information initially available is given in Table ???. There are six sites and a base at coordinates (0,0) which are fully connected. Further, assume that two vehicles are available and that the number of unforeseen tasks that occur in a shift are Poisson distributed with an equal likelihood of being located at any site.

For illustration purposes, consider two alternative solution sets as an initial schedule for the SSTOP. In Solution Set A, vehicle 1 follows an anticlockwise pattern visiting the sites in the sequence b-a-c, while vehicle 2 also follows an anticlockwise pattern visiting the sites in the sequence e-f-d. Solution Set B differs from Solution Set A only in the route of vehicle 1 that visits the sites in a clockwise manner in the sequence c-a-b. Both sets of plans are feasible solutions, fulfilling the corresponding time windows and returning to the base before the end of the shift (see Table ???). If no alarms occur during the duration of the shift then it is clear that both solution sets serve all the foreseen tasks achieving a 100% weighted fulfilment yield. However, unforeseen tasks do occur and hence it is worth verifying if both sets of plans yield also the same risk.

To assess such a risk, it is useful to track the positioning and state of the vehicles at different instants. While in Solution Set A (see Figure 1(1A)) both vehicles tend to be positioned in opposite sides at the same instants, in Solu-

Table 3

Arrival times to sites in time units and base per vehicle and solution set.

Solution Set - Vehicle	1{a}	2{b}	3{c}	4{d}	5{e}	6{f}	Base
Solution Set A: Vehicle 1	4.41	1.41	6.41	-	-	-	8.41
Solution Set B: Vehicle 2	-	-	-	6.41	1.41	4.41	8.41
Solution Set A: Vehicle 1	3.00	6.00	1.00	-	-	-	8.41
Solution Set B: Vehicle 2	-	-	-	6.41	1.41	4.41	8.41

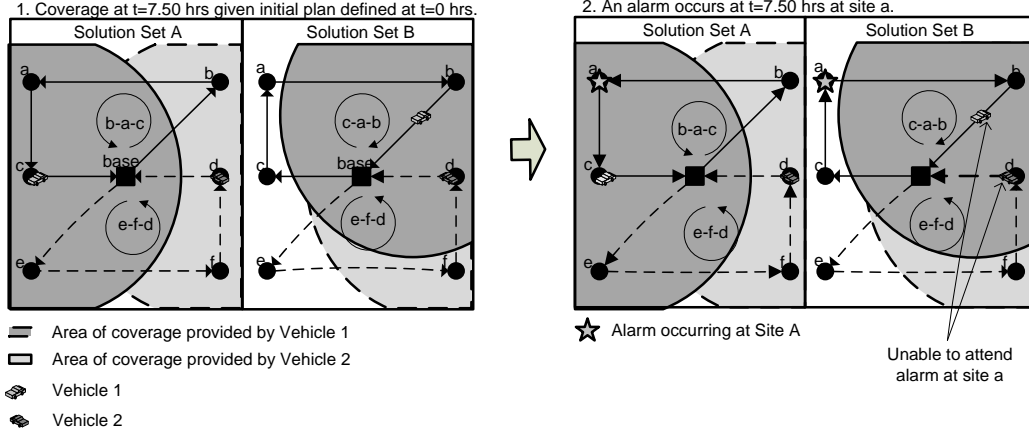


Fig. 1. Coverage of vehicles for unforeseen tasks.

tion Set B (see Figure 1(1B)), the vehicles are located in the same side (i.e. left or right) of the graph. If we then draw a circle with radius  $\delta = 1.5$  hrs (the response time standard) centered in the position of each vehicle, we can identify the sites that can be reached on time if an unforeseen task is released and thus gain insight about the inherent risks to routes.

For example, at time instant  $t = 7.50$  hrs Figure 1 shows that while Solution Set A covers all the sites (Figure 1(2A)), the Solution Set B (Figure 1(2B)) does not cover sites a, c and e. If, for example, an alarm occurs at site a at  $t = 7.50$  hrs, vehicle 1 in Solution Set A will be able to serve it returning to the base on time. However, none of the vehicles in Solution Set B will be able to serve the alarm because these are located too far from the site as shown in Figure 1(2B). In the case only one unforeseen task is released at site a at  $t = 7.50$  hrs during the shift, then Solution Set A provides a  $\Psi = 100\%$  of fulfilment yield. In contrast, Solution Set B, provides only  $\Psi = 50\%$  of fulfilment yield. Thus, given the higher number of sites covered in time of Solution Set A because of the increased separation of vehicles compared to Solution Set B, it is reasonable to choose Solution Set A over Solution Set B on the grounds of being a less risky choice.

The example illustrates that for obtaining higher fulfilment yields it is important to design routes that serve as many known tasks as possible, while maintaining at the same time the capacity of vehicles to handle unforeseen tasks on time by covering most sites throughout the duration of the shift. The

next section deals with how to address the complete and incomplete information aspects of the SSTOP.

### 3 A valid strategy for the SSTOP: the C-TOPTW

In this section we present a multi-objective optimization problem to be solved successively each time a new unforeseen task is generated at instants  $j \in J$ . We call such a multi-objective problem the Coverage Team Orienteering Problem with Time Windows (C-TOPTW). The problem is in itself composed of two sub-problems, each with its own objective function to be solved at instants  $j \in J$ : (1) the Team Orienteering Problem with Time Windows (TOPTW) with objective function  $Z_{S1}^{(j)}$ ; and (2) the Time-averaged Maximum Coverage Location Problem (TACMLP) with objective function  $Z_{S2}^{(j)}$ .

The TOPTW addresses the known tasks by maximizing the weighted number of tasks served on time while the TAMCLP, addresses the stochastic process of the arrival of new unforeseen tasks by maximizing the capacity of vehicle routes to attend new unforeseen tasks on time. Both sub-problems share the same solution structure: an ordered sequence of tasks to be served per vehicle. In the case of the TOPTW, the ordered sequence of tasks directly provides the weighted number of tasks accomplished on time. In the case of the TAMCLP, the sequence of tasks influences the objective function in a more indirect way. Namely, given that we assume that the sites are represented in a complete graph in the Euclidean plane, the sequence of tasks determines the locations of vehicles at certain points in time, defining which sites are covered and subsequently what is the inherent capacity of vehicles to serve unforeseen tasks on time.

To combine the individual objectives we propose two approaches. The first approach is by selecting the lexicographic solution for  $(Z_{S1}^{(j)}, Z_{S2}^{(j)})$  such that from a set of alternative optimal solutions of the TOPTW the solution with the highest value of  $Z_{S2}^{(j)}$  is chosen (Ehrgott, 2005). The second approach is by direct integration  $Z^{(j)} = Z_{S1}^{(j)} + Z_{S2}^{(j)}$ , calculating the weighted sum of actual foreseen tasks been served and the expected capacity to serve the next unforeseen task to be released.

The suitability of choosing a certain weighing value (i.e.  $\alpha$ ) will be studied later in the experimental section. Our overall approach is to solve the SSTOP by solving successively the C-TOPTW as follows.

We note that since any unforeseen task generated after  $TT - \mu$  can not be able to be served in a way that allows the base to be reached before the end of the shift we assume that the release time of the last unforeseen task to be generated

is before  $TT - \mu$ . In the next sub-sections we present the formulations and advantages of including each sub-problem in the C-TOPTW. The formulations are general and valid for the start of the shift and for any re-scheduling action needed thereafter.

### 3.1 Subproblem 1: The Team Orienteering Problem with Time Windows (TOPTW)

The objective of the TOPTW is similar to the objective of the Orienteering Problem (OP), which has been classified as part of the TSP with Profits set of problems (Feillet, 2005). The OP (Golden, 1987) is inspired by an outdoor sport where the participants are equipped with a compass and a map and have to visit a number of checkpoints displayed on the map. Each check point has an associated score which is collected when a participant visits the check point. The goal is to maximize the total score collected by visiting a number of check-points within a limited time-span. Hence, the OP and the SSTOP have certain similarities, where the relative importance of servicing a task is equivalent to a reward and the limited time-span is equivalent to a shift length.

The OP can be further extended to incorporate time windows for each task. We call this problem the Team Orienteering Problem with Time Windows (TOPTW). It is important to observe that if we set the mean number of unforeseen tasks per shift to zero (i.e.  $\lambda = 0$ ) then no unforeseen tasks can be generated in a shift. Then, solving the SSTOP is reduced to solving the TOPTW.

To extend the OP to the TOPTW we review the existing variants of the OP. The multi-vehicle version of the OP already exists in the literature and is known as the Team Orienteering Problem (TOP) (Chao, 1996). Kantor and Rosenwein (1992) developed a time-windows version (one time-window per site) for the OP but the time windows refer to the end of the service time rather than the start service time. Later, Nguyen and Gao (2003) extended the possibility for several time-windows per site without requiring to create artificial sites. However, both formulations of time windows consider deadlines instead of starting servicing times. Hence we have to adapt existing formulations and develop a new one. The formulation defined for every occurrence (date)  $j \in J$  of the TOPTW is as follows using the notation introduced in Section 2.

Known parameters:

- $LS(L(m))$  : the location in the Euclidian plane of the site at which task  $m \in TK_j$  is to be accomplished
- $D(a, b)$  : the transfer time between two locations in the Euclidian plane;  $a, b \in \mathbb{R}^2$ ,

- $\hat{D}(m, n)$  : the transfer time between the location of two tasks in the Euclidian plane;  $m, n \in TK_j$  such that  $\hat{D}(m, n) = D(LS(L(m)), LS(L(n)))$
- Decision variables:
- $x_{mn}^{(k)}$  : 1, if task  $n \in TK_j$  is scheduled at date  $j \in J$  immediately after task  $m \in TK_j$ , by vehicle  $k \in K$  such that  $n \neq m$ ; 0 otherwise,
- Derived values from  $x_{mn}^{(k)}$  :
- $z_n$  : 1, if task  $n \in TK_j$  is scheduled to be served; 0 otherwise,
- $y_n^{(k)}$  : 1, if task  $n \in TK_j$  is scheduled to be served by vehicle  $k \in K$ ; 0 otherwise,
- $a_n^{(k)}$  : the arrival time at site  $L(n)$  for  $n$  element of  $TK_j$  by vehicle  $k \in K$ .
- $s_n^{(k)}$  : the starting servicing time of task  $n \in TK_j$  by vehicle  $k \in K$ .
- $N \in TK_j$  : artificial task that acts as final depot, located at the base  $b \in I$  included in the set of sites  $I$ , so that  $L(N) = b$
- $o(j, k) \in TK_j$  : artificial initial tasks for every vehicle  $k \in K$  at evaluation instant  $j \in J$  such that  $\Delta_{o(j,k)} = 0$  and  $w_{o(j,k)} = 0$  (i.e. the duration and weight of the task is zero),  $e_{o(j,k)} = l_{o(j,k)} = LT_j^{(k)}$  (i.e. from Condition 4 of Definition 1 the earliest allowable time and the latest allowable time to start servicing must be equal to the time at which the vehicle is ready to be scheduled for instance  $j \in J$ ) and  $L(o(j, k)) = LL_j^{(k)}$  (i.e. the location of the provisional depot is the same as the location of the vehicle at  $t = LT_j^{(k)}$ ).  
If  $j = 0$  then, it means that the artificial task is located at the base such that it starts from the base  $L(o(0, k)) = b$ .
- $g_n^{(k)}$  : 1, if task  $n \in TK_j$  is served by vehicle  $k \in K$  and  $e_n^{(k)} \leq a_n^k$ .

The TOPTW problem with  $Z_{S1}^{(j)}$  objective to be solved at evaluation instant  $j \in J$  is then defined by the following mixed integer programming formulation, where “ $M$ ” represents a large number.

$$\max Z_{S1}^{(j)} = \sum_{n \in TK_j} \sum_{k \in K} w_n y_n^{(k)} \quad (6)$$

s.t.

$$\sum_{k \in K} y_n^{(k)} \leq 1, \quad \forall n \in TK_j \setminus \{N\}, \quad (7)$$

$$\sum_{\substack{l \in TK_j, \\ l \neq m}} x_{lm}^{(k)} = \sum_{n \in TK_j, n \neq m} x_{mn}^{(k)} = y_m^{(k)} \quad \forall m \in TK_j \setminus \{N\} \cup \{o(j, k), k \in K\}, \forall k \in K, \quad (8-9)$$

$$e_n \leq s_n^{(k)} \leq l_n \quad \forall n \in TK_j, \forall k \in K, \quad (10-11)$$

$$M(1 - x_{mn}^{(k)}) \geq s_m^{(k)} + \Delta_m + \hat{D}(m, n) - a_n^{(k)} \quad \forall m, n \in TK_j : n \neq m, \forall k \in K, \quad (12)$$

$$M(x_{mn}^{(k)} - 1) \leq s_m^{(k)} + \Delta_m + \hat{D}(m, n) - a_n^{(k)} \quad \forall m, n \in TK_j : n \neq m, \forall k \in K, \quad (13)$$

$$e_n^{(k)} - a_n^{(k)} \leq M(g_n^{(k)}) \quad \forall n \in TK_j, \forall k \in K, \quad (14)$$

$$a_n^{(k)} - e_n > M(g_n^{(k)}) \quad \forall n \in TK_j, \forall k \in K, \quad (15)$$

$$s_n^{(k)} \leq a_n^{(k)} + M(1 - g_n^{(k)}) \quad \forall n \in TK_j, \forall k \in K, \quad (16)$$

$$s_n^{(k)} \geq a_n^{(k)} - M(1 - g_n^{(k)}) \quad \forall n \in TK_j, \forall k \in K, \quad (17)$$

$$s_n^{(k)} \leq e_n + M(g_n^{(k)}) \quad \forall n \in TK_j, \forall k \in K, \quad (18)$$

$$s_n^{(k)} \geq e_n - M(g_n^{(k)}) \quad \forall n \in TK_j, \forall k \in K, \quad (19)$$

$$\sum_{k \in K} \sum_{\substack{n \in TK_e, \\ n \neq 0}} x_{0n}^{(k)} = \sum_{\substack{m \in TK_j, \\ m \neq N}} x_{mN}^{(k)} = |K| \quad (20)$$

$$s_0^{(k)} = LT_j^{(k)}, s_N^{(k)} \leq TT \quad \forall k \in K, \quad (21-22)$$

$$y_n^{(k)} \in \{0, 1\}, s_n^{(k)} \in [LT_j^{(k)}, TT] \quad \forall n \in T, \forall k \in K, \quad (23-24)$$

$$x_{mn}^{(k)} \in \{0, 1\} \quad \forall m, n \in TK_j : n \neq m. \quad (25)$$

The normalized objective (6) maximizes the weighted proportion of tasks accomplished on time. Constraints (7) assure that a task will be served by at most one vehicle. Constraints (8-9) guarantee the connectivity for each path a vehicle follows. Constraints (10) and (11) assure that any task  $n$  that is serviced by a vehicle starts to be executed within the respective time windows.

To ensure that the TOPTW generates valid schedules for vehicles in accordance with Condition 3 of Definition 1, constraints (12)-(19) have been included. The set of constraints (12) and (13) force that if task  $m$  immediately precedes task  $n$  (i.e.  $x_{mn}^{(k)}$ ), then the arrival time of a task  $n$  must be *exactly* equal to the starting time of the preceding task  $m$  plus the time taken to service task  $n$  and to transfer to task  $m$ . Hence, the relation  $a_n = s_m + \Delta_m + \hat{D}(m, n)$  is forced by making the upper and lower bounds equal. Constraints (12) and (13) allow  $s_m^{(k)}$  and  $a_n^{(k)}$  take any positive value if task  $n$  does not precede task  $m$  in vehicle  $k$  route using the “big  $M$ ” method, reducing the expressions to  $-M \leq s_m + \Delta_m + \hat{D}(m, n) - a_n^{(k)} \leq M$  when task  $n$  does not follow task  $m$ . The equality constraint is enforced when task  $n$  follows task  $m$ . Note also that these constraints also eliminate the possibility of sub-tours as in this case, as only one service time per task  $m$  is allowed to exist. The set of constraints (14) and (15), use the “big”  $M$  method to determine if a vehicle arrives to a site later than the earliest allowable time or not (i.e.  $a_n^{(k)} \leq s_n^{(k)}$ ). Similarly, con-

straints (14) to (19) force that the task is served as soon as possible when the vehicle arrives on site. Constraints (14) and (15) define the indicator variable  $g_n^{(k)}$  that determines whether  $a_n^{(k)} \geq e_n$  or otherwise. The set of constraints (16) and (17) assure that if a vehicle arrives later than the earliest allowable time i.e.  $a_n^{(k)} \geq e_n$ , then the vehicle is forced to start servicing the task as soon as the vehicle arrives, i.e.  $s_n^{(k)} = a_n^{(k)}$ . Otherwise, constraints (18) and (19) assure that if a vehicle arrives earlier than the earliest allowable time i.e.  $a_n^{(k)} < e_n$ , then the vehicle is forced to start servicing the task as early as the earliest allowable time, i.e.  $s_n^{(k)} = e_n$ .

Next, constraint (20) ensures that the vehicles start and finish at the base. Constraints (21) and (22) ensure that all vehicles start at their corresponding liberation time, i.e.,  $LT_j^{(k)}$  and finish within the termination time of the shift, i.e.  $TT$ . Finally, constraints (23) to (25) define the domains of the decision variables in the model.

### 3.2 Subproblem 2: The Time-averaged Maximum Coverage Location Problem (TAMCLP)

In Section 3.1 it was shown that it is possible to identify which sites are covered for servicing potential unforeseen tasks on time, suggesting that it is possible to obtain a measure of the capacity of vehicles to serve unforeseen tasks on time. A useful measure of the capacity for servicing potential unforeseen tasks is the weighted proportion of sites that can be served by a vehicle on time if an unforeseen task is released at a given point in time. In fact, for a given time instant, weighted demand covered by the vehicles is equivalent to the objective function of the Maximum Coverage Location Problem or MCLP (Church and Reville, 1974), which is used for locating emergency vehicles at certain bases to cover possible sites that request emergency services. The difference with the situation found in the SSTOP lies in the fact that emergency vehicles in the MCLP serve emergency requests and then return to the base while vehicles in the SSTOP follow a route and return to the base only when there are no more tasks to perform. This means that the location in the SSTOP situation needs to be constantly updated because a static situation as in the MCLP does not exist.

Hence, by taking a representative number of “snapshots” of the system at regular intervals of length  $\epsilon$  it is possible to measure the instantaneous capacity of vehicles to serve unforeseen tasks. Nonetheless, it is important to note that this is only an approximation of the expected demand covered at each snapshot as the vehicles’ position can only be tracked for known routes. As unforeseen tasks may be released, the vehicles’ routes may be changed as a result. As computing routes for all possible future unforeseen releases becomes



intractable, we limit ourselves to computing the expected demand covered assuming that the vehicles routes remain unchanged. To qualify this assumption we weight the coverage in a “snapshot” by the probability that the first unforeseen task may be released in the interval of such “snapshot” at a given site. If the frequency of unforeseen tasks is sufficiently small, this one-look ahead perspective of tracking current routes may cover most possibilities of the SSTOP problem.

If we apply the concept of sampling MCLP objective values over time to the example from Figure 1, we obtain a graph (see Figure ??) that shows the proportion of sites covered by both solution sets at different time shots. In this way, Figure ?? clearly shows that averaging over time Solution Set A is less risky than Solution Set B by covering more sites during the shift. The MCLP measured over time can be interpreted as a measurement of the quality of coordination between vehicles. When vehicles are located further apart and do not execute tasks simultaneously, there is less overlapping of coverage and thus the vehicles can more efficiently cover sites for unforeseen tasks.

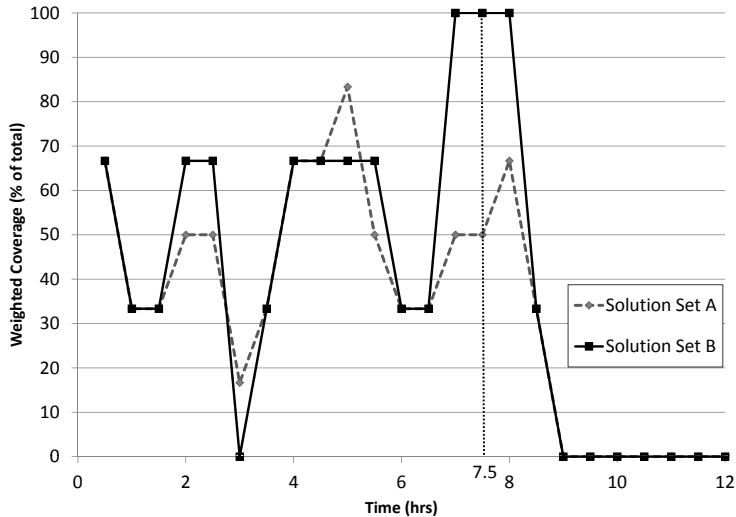


Fig. 2. Weighted coverage over time per solution.

To devise a strategy for the SSTOP, we propose to measure the MCLP over time and average it to construct a new time-dependent dynamic coverage problem: the Time-Averaged Maximum Expected Location Problem (TAMCLP). In the next lines we present the general structure for the formulation of the TAMCLP. It must be noted that due to the application of interpolation and Euclidian distances in the formulation it is not possible to model it as a mixed integer program. In addition we introduce an auxiliary variable  $ST_j$  to mark the start at which coverage will be calculated given that different vehicles may have different liberation times (i.e.  $\{LT_j^{(k)} : k \in K\}$ ).

The formulation uses the following variables.

Known parameters:

$ST_j = \min_{k \in K} \{LT_j^{(k)} : k \in K\}$	:	the starting time at which coverage is to be calculated at the $j$ th evaluation episode,
$\epsilon$	:	the time between discrete points in time,
$TS_j = \lceil \frac{TT - \mu - ST_j}{\epsilon} \rceil + 1$	:	number of time instants to evaluate in interval $[ST_j, TT - \mu]$ ,
$p$	:	index identifying discrete point in time; $t_p = \epsilon p + ST_j, t_p = t_1, \dots, t_{TS_j}$
$w_i$	:	the relative importance associated to each site $i$ , assuming that if an unforeseen task is generated at this site it will have such a weight,
Derived values from $x_{mn}^{(k)}$		
$q_{ip}$	:	1, if at least one vehicle is covering site $i$ at time $t_p$ ; 0, otherwise,
$g_{ip}^{(k)}$	:	1, if vehicle $k$ is covering site $i$ at time $t_p$ ; 0, otherwise,
$\rho_p^{(k)}$	:	radius covered (in time units) by vehicle $k$ at time $t_p$ ,
$LV(p, k) = (X_p^{(k)}, Y_p^{(k)})$	:	Location in the Euclidian plane of vehicle $k \in K$ at time $t = \epsilon p + ST_j$ . The coordinates are described by $X_p^{(k)} \in \mathbb{R}$ and $Y_p^{(k)} \in \mathbb{R}$ ,

We can then define the TAMCLP as follows, further detailed in Sections 3.2.1 to 3.2.4.

$$Z_{S2}^j = \sum_{p=1}^{TS_j} \sum_{i \in I} (e^{-\lambda p})(1 - e^{-\lambda \epsilon}) q_{ip} w_E v_i \quad (26)$$

s.t.

$$\text{Determination of } LK_p^{(k)} \text{ constraints} \quad \forall k \in K, \forall p = 1, \dots, TS_j \quad (27)$$

$$\text{Determination of } \rho_p^{(k)} \text{ constraints} \quad \forall k \in K, \forall p = 1, \dots, TS_j \quad (28)$$

$$\text{Determination of } g_{ip}^{(k)} \text{ constraints} \quad \forall i \in I, \forall k \in K, \forall p = 1, \dots, TS_j \quad (29)$$

$$\sum_{k \in K} g_{ip}^{(k)} \geq q_{ip} \quad \forall i \in I, \forall p = 1, \dots, TS_j \quad (30)$$

$$g_{ip}^{(k)} \in \{0, 1\} \quad \forall i \in I, \forall k \in K, \forall p = 0, 1, \dots, TS_j \quad (31)$$

$$q_{ip} \in \{0, 1\} \quad \forall i \in I, \forall p = 0, 1, \dots, TS_j \quad (32)$$

$$\rho_p^{(k)} \in \mathbb{R}, LC_p^{(k)} \in \mathbb{R}^2, q_p^{(k)} \in [0, 1] \quad \forall k \in K, \forall p = 0, 1, \dots, TS_j \quad (33-35)$$

All the variables of the TAMCLP formulation are derived from the sequence of tasks scheduled per vehicle constrained by the TOPTW sub-problem. Thus, the TAMCLP is structured in the logical sequence in which these variables should be derived. An overview of this sequence is given as follows.

For calculating the average expected demand covered across time, it is first necessary to track the guards' locations at discrete points in time given by constraints (27) given in Section 3.2.2. Next, given that the SSTOP conditions for a valid strategy defined in Definition 1 imply that the actual area covered per vehicle is time dependent, the coverage radius must be determined with

constraints (28) given in Section 3.2.3. Having established the coverage radius, it is then possible to identify whether a certain vehicle covers a certain site as defined in constraints (29) given in Section 3.2.4. Then, in constraints (30) if at least one vehicle covers a site, the site is considered to be covered and used as input for the objective function in (26). Finally, constraints (31)-(35) define the derived variables. Although all of these constraints can be formulated in a standard mathematical programming format using the “big- $M$ ” method, such formulations become highly complex given the number of conditions involved for the constraints to be active. Hence, for clarity, we chose to explicitly describe the conditions required.

### 3.2.1 TAMCLP Objective function

The normalized objective function of the TAMCLP (26) is based on that of the Maximum Coverage Location Problem MCLP (Church and Reville, 1974). In effect, the TAMCLP objective function samples repeatedly at each snapshot,  $p = 0, 1, \dots, TS_j$ , estimating the expected weighted number of unforeseen tasks covered for a  $[t_p, t_{p+1})$  interval. We assume that the positions and state of a vehicle remain unchanged in such an interval.

The expected weighted number of unforeseen tasks covered for a  $[t_p, t_{p+1})$  interval considers the probability that an unforeseen task will be generated in such interval at a given site  $i \in I$ . This probability is in fact the probability that an unforeseen task will be generated at any given interval i.e.  $(1 - e^{-\lambda\epsilon})v_i$  multiplied by the probability that no unforeseen task has been generated before, i.e.  $(e^{-\lambda p})$ . Although it may be the case that positions of vehicles remain unchanged even after the arrival of an unforeseen task, we ignore such a possibility due to its difficulty to estimate a-priori. In addition, the probability of releasing two or more unforeseen tasks for a sufficiently small  $\epsilon$  interval is considered to be negligible.

### 3.2.2 Determination of $LK(p, k)$

For tracking the vehicles positions over time given by  $LK(p, k)^{(k)} = (X_p^{(k)}, Y_p^{(k)})$  as specified in constraints (27), we distinguish between two situations in which a vehicle can be found (see Figure ??).

- (1) The vehicle is at a site. Hence the coordinates of the vehicle correspond to that of the site’s.
- (2) The vehicle is transferring to serve a task at a site. Hence the coordinates are linearly interpolated between the origin and the destination.

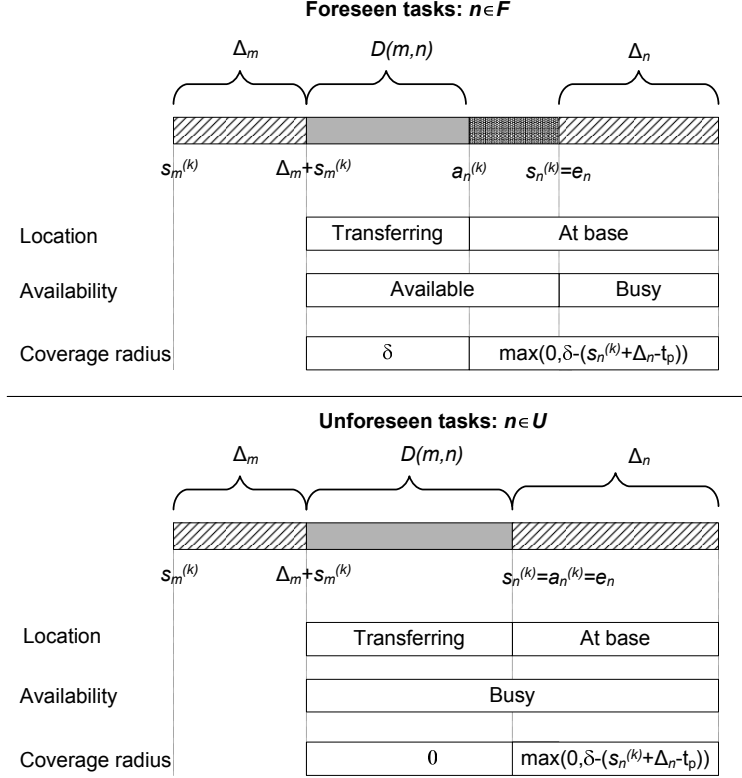


Fig. 3. The status and coverage radius of vehicles according to a time line.

Therefore, for calculating  $X_p^{(k)}$  we obtain the following relationship for  $\forall p = 0, 1, \dots, TS_j$  such that  $\epsilon p + ST_j \geq LT_k^{(j)}$ ,  $\forall k \in K$  (note that the calculation for the  $Y$ -coordinate  $Y_p^{(k)}$  is analogous):

$$X_p^{(k)} = \begin{cases} X_m & \text{if } \exists m, n \in TK_j: n \neq m, \\ & x_{mn}^{(k)} = 1, a_n^{(k)} \leq t_p \leq s_n^{(k)} + \Delta_n \\ X_m + \frac{(t_p - s_m^{(k)} - \Delta_m)(X_n - X_m)}{\hat{D}(m, n)} & \text{if } \exists m, n \in TK_j: n \neq m, \\ & x_{mn}^{(k)} = 1, s_m^{(k)} + \Delta_m < t_p \leq a_n. \end{cases} \quad (36)$$

where:

$$\begin{aligned} X_m, X_n &: X \text{ coordinates of locations } L(m) \text{ and } L(n), \text{ and} \\ a_n^{(k)} &: \text{the arrival time of vehicle } k \text{ for servicing task } n \text{ such that} \\ & a_n^{(k)} = s_m^{(k)} + \Delta_m + \hat{D}(m, n). \end{aligned}$$

Given Condition 4 of Definition 1 of what is a valid schedule, it may be the case that at the time an unforeseen task is released one or more vehicles are unable to be reassigned immediately to serve different tasks because at the time of the evaluation instant  $j \in J$ . The reason for is that the vehicles may be either travelling to serve an unforeseen task (i.e.  $VS_j^{(k)} = 1$ ) or serving a task on site (i.e.  $VS_j^{(k)} = 2$ ). In case a vehicle  $k \in K$  is in such situation, it

holds that  $\epsilon p + ST_j \leq LT_k^{(j)}$ . For this reason, to track vehicles before all of them can be reassigned immediately to serve different tasks it is necessary to take into account information from the information set  $\mathcal{I}_j$  that is derived from past schedules.

For calculating  $LK(p, k)$  in these cases, we first define the coordinates of vehicle  $k \in K$  at the time of the evaluation  $j \in J$  as:  $VL_j^{(k)} = X_{VL_j}^{(k)}, Y_{VL_j}^{(k)}$  consistent with the notation introduced in describing the information set  $\mathcal{I}_j$ . Similarly, we also define the coordinates of vehicle  $k \in K$  at the earliest allowable time that the vehicle can be rescheduled after evaluation instant  $j \in J$  as:  $LL_j^{(k)} = X_{LL_j}^{(k)}, Y_{LL_j}^{(k)}$ . Thirdly, we define the arrival time at a site where a previously scheduled (i.e. at instant  $j - 1 \in J$ ) unforeseen task has a vehicle either transferring to it or currently servicing without having reached completion yet introducing a new variable, namely,  $AU_j^{(k)}$  such that  $AU_j^{(k)} = D(LL_j^{(k)}, LV_j^{(k)}) + LL_j^{(k)}$ . We introduce this variable to distinguish whether a vehicle is servicing a task from a previous evaluation instant or not. Hence, to calculate  $X_p^{(k)}$ ,  $\forall p = 0, 1, \dots, TS_j$  such that  $\epsilon p + ST_j < LT_k^{(j)}$ ,  $\forall k \in K$  the following relationship is needed.

$$X_p^{(k)} = \begin{cases} X_{VL_j}^{(k)} + \frac{(\epsilon p + TS_j - LT_j^{(k)})(X_{LL_j}^{(k)} - X_{VL_j}^{(k)})}{D(LL_j^{(k)}, VL_j^{(k)})} & \text{if } \epsilon p + TS_j < AU_j^{(k)}, \\ X_{LL_j}^{(k)} & \text{if } \epsilon p + TS_j \geq AU_j^{(k)}. \end{cases} \quad (37)$$

### 3.2.3 Determination of $\rho_p^{(k)}$

To determine the coverage radius  $\rho_p^{(k)}$  as specified in constraints (28), we identify three possible situations that a vehicle  $k$  may encounter at  $t_p = \epsilon p$  (see Figure ??):

- (1) Transferring or waiting to serve a foreseen task  $n \in F$ . The vehicle can be immediately re-scheduled to serve another task and the coverage radius corresponds to that of the standard response time.
- (2) Servicing a foreseen or unforeseen task. The vehicle can only serve a new task after it finishes servicing the task it currently is servicing and hence has a reduced coverage area.
- (3) Transferring to serve an unforeseen task  $n \in U$ . If we assume for simplicity that  $\Delta_n \geq \delta$ , then the vehicle has an effective coverage of zero as it will never finish servicing the unforeseen task early enough to be able to serve the new unforeseen task on time.

Consolidating the three cases described above, we have that  $\rho_p^{(k)}$  is defined as follows  $\forall p = 0, 1, \dots, TS_j$  such that  $\epsilon p + ST_j \geq LT_k^{(j)}$ ,  $\forall k \in K$ :

$$\rho_p^{(k)} = \begin{cases} \delta & \text{if } \exists m, n \in TK_j : n \neq m, x_{mn}^{(k)} = 1, \\ & n \in F, s_m^{(k)} + \Delta_m \leq t_p \leq s_n^{(k)} \\ \max(0, \delta - (s_n^{(k)} + \Delta_n - t_p)) & \text{if } \exists n \in TK_j : y_n^{(k)} = 1, \\ & s_n^{(k)} \leq t_p \leq s_n^{(k)} + \Delta_n \\ 0 & \text{if } \exists m, n \in TK_j : n \neq m, x_{mn}^{(k)} = 1, \\ & n \in U, s_m^{(k)} + \Delta_m \leq t_p \leq s_n^{(k)} \end{cases} \quad (38)$$

As with tracking vehicles, the coverage radius should be adjusted for the case that  $\epsilon p + ST_j \geq LT_k^{(j)}$  so that the earliest time at which the vehicle  $k \in K$  after the  $j$ th evaluation instant is considered. Hence, it follows  $\forall p = 0, 1, \dots, TS_j$  such that  $\epsilon p + ST_j < LT_k^{(j)}$ ,  $\forall k \in K$  that:

$$\rho_p^{(k)} = \max\{0, \Delta - (LT_k^{(j)} - \epsilon p + ST_j)\} \quad (39)$$

### 3.2.4 Determination of $g_{ip}^{(k)}$

For determining if a vehicle  $k \in K$  is actually covering a site at a given time step (i.e.  $g_{ip}^{(k)} = 1$ ) as defined by constraints (29), two conditions are needed. The first is that the vehicle is effectively able to cover the site in question,  $i \in I$  given its coverage radius  $\rho_p^{(k)}$ . The second is that for a site to be considered to be covered by a vehicle, such vehicle must be able to have sufficient time to relocate to such a site to serve a hypothetical unforeseen task and return to the base before the end of the shift in accordance to Condition 4 of Definition 1. Recalling that  $LS(i) = (X_i, Y_i)$  defines the location of a site and  $LK(p, k)$  the location of a vehicle,  $g_{ip}^{(k)}$  is determined by the following constraints for  $\forall i \in I \setminus \{0\}, \forall p = 0, 1, \dots, TS_j, k \in K$ .

$$g_{ip}^{(k)} = \begin{cases} 1 & \text{if } \rho_p^{(k)} \geq D(LK(p, k), LS(i)) \\ 0 & \text{if } \rho_p^{(k)} < D(LK(p, k), LS(i)) \end{cases} \quad (40)$$

$$TS_j + \epsilon p + \rho_p^{(k)} - \delta + \mu + D(LS(i), LS(0)) \leq TT + M(1 - g_{ip}^{(k)}) \quad (41)$$

### 3.2.5 Calculation example

In Table ?? we illustrate for the example from Figure 1 the coverage calculation for Solution Set B at  $t_p = 7.50$  hrs. Note that “ $D(\cdot)$ ” is an abbreviation for  $D((LK(p, k), (LS(i)))$ . In addition, “ $\geq \rho_p^{(k)}$ ?” is an abbreviation for the condition of whether the site is at a shorter distance from the vehicle than the coverage radius as in constraints (40) indicating a 1, if the condition is met; 0 otherwise. Similarly, “ $\leq TT$ ?” is an abbreviation of the condition depicted in constraints (41), namely whether the vehicles can serve an unforeseen task in such a site and return before the end of the shift.

Table 5

Example for calculating coverage for Solution Set B at t=7.50 hrs.

$\lambda$	=	1			$\rho_p^{(1)}$	=	1.5			
$\Delta_E$	=	7.5			$\rho_p^{(2)}$	=	1.5		$\forall i$	
$TT$	=	12			$(e^{-\lambda p})$	=	0.4994		(a)	
$\delta$	=	1.5			$(1 - e^{-\lambda \epsilon})$	=	0.0452		(b)	
$w_i$	=	100	$\forall i$		(a)x(b)	=	0.0226			
$v_i$	=	0.167	$\forall i$							
<hr/>										
		$D(\cdot)$		$\geq \rho_p^{(k)}?$		$\leq TT?$		$g_{ip}^{(k)}$		
i	$k = 1$	$k = 2$	$k = 1$	$k = 2$	$k = 1$	$k = 2$	$k = 1$	$k = 2$	$q_{ip}$	$q_{ip}w_iv_i$
a	1.684	2.160	0	0	1	1	0	0	0	0.000
b	0.500	1.004	1	1	1	1	1	1	1	16.667
c	1.769	1.914	0	0	1	1	0	0	0	0.000
d	0.737	0.086	1	1	1	1	1	1	1	16.667
e	2.328	2.160	0	0	1	1	0	0	0	0.000
f	1.684	1.004	0	1	1	1	0	1	1	16.667
Coverage for time step: $\sum_{i \in I} (e^{-\lambda p})(1 - e^{-\lambda \epsilon})q_{ip}w_iv_i =$										1.130

#### 4 Solving the C-TOPTW

To solve moderately sized C-TOPTW problems (i.e. with two vehicles and less than 20 tasks) each time a new problem instance is generated we first generate a number of feasible solutions for the TOPTW (a sequence of tasks to be performed by each vehicle) calculating the associated TOPTW scores for each of them. Next, we track the vehicles' paths of each solution at regularly spaced discrete points in time to then calculate the TAMCLP associated objective function of each proposed solution using the formulation in (26). We consider two ways for integrating the TOPTW and the TAMCLP in the C-TOPTW. One way is by finding the lexicographic optimum with the objective of the TOPTW as the primary one. The second way is by adding directly the objectives of TOPTW and the TAMCLP, where the first objective is the actual weighted number of foreseen tasks and the TAMCLP is a measure of the expected weighted unforeseen task that can be served if a new unforeseen task is released.

The main underlying idea of the presented heuristic structure is to take advantage of the fact that the TOPTW is more constrained than the TAMCLP. For solving the TOPTW we first have to recognize that the computational complexity of the TOPTW is at least as difficult as the OP which is already NP-hard (Laporte, 1990). Hence, most of the approaches available in the literature to solve the OP and its derivatives are based on metaheuristics (see Chao et al., 1996; Tasgetiren, 2002; Liang and Smith, 2006). Furthermore, the relationship between the TOPTW and the TAMCLP is poorly understood. As the purpose of this paper is to show whether including coverage considerations (i.e. the TAMCLP) improves the quality of solutions for the SSTOP, we choose a heuristic that is fast and simple, that can provide "good" solu-

tions for the TOPTW and at the same time provides a diversity of alternative solutions with a variety of TAMCLP reliability measures to choose from.

In this context, we construct “good” feasible plans by extending a stochastic-based algorithm proposed by Tsiligirides for the Orienteering Problem (OP) (Tsiligirides, 1984). Tsiligirides’ S-algorithm is based on devising a reasonable measure of the desirability to append a task to an existing route by taking the ratio between the reward collected and the extra distance needed to serve such a task. The S-algorithm constructs a sequence of tasks by adding one task at a time and making the probability of adding a certain task in a route dependent on such desirability ratio. Therefore, the algorithm provides a certain directionality in the search for good “solutions” using a greedy approach, but also allows for exploring neighborhood areas in the solution space by introducing stochasticity. Tests by Tsiligirides’(1984) reveal a satisfactory performance in solving the OP.

In this paper, we adapt Tsiligirides’ S-algorithm which has proven to have a performance (Tsiligirides, 1984) for solving the TOPTW. Instead of the extra distance required to service a candidate task, we use the extra time needed to serve the candidate task in the denominator of the fitness function. In this way, the time required for servicing a task and also the time required to wait at a site for an earliest starting time  $e_n$  to occur are considered in the desirability measure. Therefore, the further in the future the earliest allowable starting time is, the lower the desirability is for appending such a task to the current schedule. The fitness function is then defined as follows, where  $m$  is the last executed task, which in a first step is equivalent to the artificial task (i.e.  $o(j, k) = m$ ) and  $n$  is the candidate task to be evaluated.

$$f(m, n) = \left[ \frac{w_n}{\max\{s_m^{(k)} + \Delta_m + D(m, n) + \Delta_n, e_n\} - s_m^{(k)} + \Delta_m + D(m, n)} \right]^3 \quad (42)$$

A power of 3 is used in the fitness function in order to amplify the differences between ratios, and bias further the probability of selecting tasks with higher desirability ratios.

Another extension made to Tsiligirides’ S-algorithm is that of dealing with more than one vehicle. In Chao et al. (1996), Tsiligirides’ S-algorithm is extended by scheduling the vehicles either sequentially or concurrently. The sequential construction of routes involves constructing first a route for the first vehicle, then for the second and so on. In the concurrent approach, each vehicle takes turns to add tasks to their routes. We propose to randomly select the turn of a vehicle instead so as to further increase solution variety and do not restrict the search solutions to concurrent type of routing sequences.

The logic of the extended algorithm is simple. First it is checked whether a task



$n$  can be feasibly added to the route of a vehicle  $k$ . Such feasibility depends on the possibility of meeting the task deadline on time (i.e. start servicing the task before the latest allowable starting time) and to have sufficient time to return back to the base before the shift ends. Next, the fitness functions  $f(n)$  are computed and then ranked. To limit the running time only four tasks are considered for inclusion in the vehicles' route (similar to Tsiligirides (1984)). These tasks are referred to as members of the elite set,  $ES$ . Finally, task  $n^*$  is selected based on the following probability.

$$P(n^*) = \frac{f(n^*)}{\sum_{n \in ES} f(n)} \quad (43)$$

## 5 Simulation set-up

To study whether incorporating coverage considerations (i.e. embedding the TOPTW in the TAMCLP) yield superior schedules for the SSTOP we designed a series of simulation experiments (with 200 simulated shifts per experiment) where we benchmark in each simulation run three suggested strategies. Each time an unforeseen task is released, the information set of known tasks and sites is updated and one of the selected strategies is used to reschedule the vehicles paths. The first strategy consists of (re)solving the TOPTW; as the TOPTW only considers known tasks, it is used as a base to compare it to other strategies. We refer to this strategy as simply the (TOPTW) base strategy. The other two strategies consider stochastic information of the generation of unforeseen tasks. The second strategy is to solve a multi-objective problem of the TOPTW and the TAMCLP by selecting a lexicographic optimum solution of both problems (i.e. selecting a solution that maximizes the TAMCLP among alternative optima of the TOPTW). We refer to this strategy as the (C-TOPTW) lexicographic strategy. The third strategy, integrates the TOPTW and TAMCLP objectives with an affine combination that assigns equal weights to both objectives, selecting the optimum of such affine combination. We refer to this strategy as the (C-TOPTW) integration strategy.

The performance of the three strategies are tested in four scenarios. These scenarios have certain parameters in common for ease of comparison (see Table ??). However, to obtain independent confirmation of simulation results, the four scenarios differ in four key variables (see Table ??): the distribution of sites in the Euclidian space (i.e. type A and B shown in Figure ??), the arrival rate of emergencies that generate unforeseen tasks (i.e. Scenario 4 has a higher arrival rate) and whether the probability of released unforeseen tasks is equal for every site or not (i.e. Scenario 3 has heterogenous arrival rates per site). The scenarios were checked to verify if a-priori alternative optimal solutions of the TOPTW may yield different values of the TAMCLP objective function as shown in Table ?. This means that different levels of risks exist

Table 6

Common parameters in simulation experiments

Number of vehicles, $ K $	2
Number of sites, $ I $	12
Number of foreseen tasks, $ F $	12
Maximum weight possible of foreseen tasks, $\sum_{n \in F}$	100
Weight of unforeseen tasks, $n \in U$	100
Shift length in time units, $TT$	100
Duration of foreseen tasks in time units, $\Delta_n, n \in F$	Max: 8 Median: 4.5 Min: 3.5
Duration of unforeseen tasks in time units, $\mu$	7.5
Response time in time units, $\delta$	7.5
Time windows: earliest allowable time, $e_n$	Generated by $U(0, 40)$
Time windows: latest allowable time, $e_n$	Generated by $U(70, 100)$
Number of shifts simulated per scenario	200
Time between “snapshots” in time units, $\epsilon$	1
Number of generated solutions per (re-)scheduling instance	4000

Table 7

Scenarios description

Scenario	Layout type	Alarm intensity	Site risk	$Z_{S1}^{(j)}$	Max	$Z_{S2}^{(j)}$ Average	Min
1	A	1	Homogeneous	100	47.63	40.58	30.27
2	A	1	Heterogeneous	100	49.51	39.78	27.88
3	B	1	Homogeneous	100	50.25	40.66	30.27
4	B	2	Homogeneous	100	70.82	57.29	46.07

among alternative optima and thus may yield different expected results for the SSTOP.

Note that because the purpose of the experiments is to explore the possibility of embedding coverage considerations in the strategies devised for the SSTOP, these are kept computationally simple. Only twelve sites (each with one foreseen task) and two vehicles are considered. The running time of a day shift simulation with on average 1 unforeseen task generated is 2.5 minutes running on a Intel i5 Core processor. In addition, emergency events are rare and important. Thus, a maximum mean number of unforeseen tasks per shift (i.e.  $\lambda$ ) of two (see scenario 4) with a weight equal to the sum of all the foreseen tasks available is set. A few initial runs were also made prior to this so as to assure that all the foreseen tasks may be served a-priori within the duration of a shift with a certain time to spare to still be able to service at least two unforeseen tasks per shift as would occur in desirable operating conditions.

## 6 Results

We ran 200 simulated shifts for each strategy and each scenario. Table ?? shows a summary of the results of these simulations. The results show that across four scenarios that the two strategies that incorporate coverage consid-

erations (i.e. lexicographic and integration strategies) have a significant better capability of being able to service unforeseen tasks on time.

Table 8

Summary of results of simulation experiments( $F$ :Foreseen tasks,  $U$ : Unforeseen tasks)

a. Mean fulfillment yield at the end of shift (i.e.  $\bar{\Psi}$ )

Scenario	Base			Lexicographic			Integrated		
	$F$	$U$	$F \cup U$	$F$	$U$	$F \cup U$	$F$	$U$	$F \cup U$
1	99.48	61.00	160.48	99.78	72.00	171.78	99.67	72.50	172.17
2	99.73	49.50	149.23	99.62	62.00	161.62	99.69	63.50	163.19
3	99.63	58.50	158.13	99.62	68.00	167.62	99.69	72.00	171.69
4	98.92	115.50	214.42	98.87	130.00	228.87	99.14	126.00	225.14

b. P-values for pairwise 1-tailed t-test comparisons of weighted score collected

Scenario	Lexicographic vs. Base			Integration vs. Base			Integration vs. Lexicographic		
	$F$	$U$	$F \cup U$	$F$	$U$	$F \cup U$	$F$	$U$	$F \cup U$
1	0.0489	0.0032	0.0023	0.1426	0.0030	0.0025	0.2091	0.4331	0.4471
2	0.1940	0.0005	0.0005	0.3940	0.0001	0.0001	0.2412	0.2825	0.2709
3	0.4610	0.0032	0.0030	0.3213	0.0001	0.0001	0.3052	0.0513	0.0462
4	0.4288	0.0017	0.0014	0.1940	0.0250	0.0212	0.1167	0.1630	0.1779

c. Counting comparisons of performance

Scenario	Lexicographic vs. Base			Integration vs. Base			Integration vs. Lexicographic		
	$F$	$U$	$F \cup U$	$F$	$U$	$F \cup U$	$F$	$U$	$F \cup U$
1									
Better	14	38	46	12	38	47	4	16	18
Same	181	143	135	182	143	133	191	168	163
Worse	5	19	19	5	19	19	5	16	19
2									
Better	5	38	41	6	34	39	5	15	19
Same	189	147	142	188	157	150	191	173	168
Worse	6	15	17	9	15	17	4	12	13
3									
Better	10	28	35	7	33	39	9	14	23
Same	182	160	150	188	158	150	184	179	166
Worse	8	12	15	10	12	15	7	7	11
4									
Better	19	50	59	20	50	59	19	25	34
Same	161	125	108	164	118	104	169	146	131
Worse	20	25	33	25	25	33	12	29	35

We used pairwise t-test comparisons as recommended by Law and Kelton (1998) to compare policies in a simulation environment. The results show that the C-TOPTW lexicographic strategy significantly served more unforeseen tasks than the TOPTW base strategy at significance levels of  $p < 0.005$  for all the scenarios tested. Similarly, the direct integration approach yielded similar significance levels of outperforming the base strategy in servicing unforeseen tasks except for Scenario 4 where it still outperforms the base case but at a significance level of  $p < 0.05$ . The reason for such drop in significance levels for

Scenario 4, may have to do with the lower statistical power of the experiment as the alarm rate is doubled and the variability of results increases. The t-test comparisons is illustrated with the count comparisons in part c) of Table ??.

For every scenario, both strategies considering coverage are able to service more unforeseen tasks in more than twice the number of shifts compared to the base strategy.

At the same time, the results show similar results in servicing unforeseen tasks for both coverage based strategies (i.e. C-TOPTW lexicographic and C-TOPTW direct integration). This result may stem for the fact that for the direct strategy to outperform the lexicographic strategy it requires that there exist cases in which not servicing a foreseen task can be traded-off with better TAMCLP values that yields a higher expected number of weighted unforeseen tasks covered. For this to be the case, either the importance of the unforeseen tasks should be higher and/or the rate of unforeseen tasks should be higher. In either way, the variance on the results of the experiments requiring longer simulation runs to be able to observe such effects. Moreover, from an operational perspective having to trade-off servicing foreseen task for a future unforeseen task is undesirable as it means that not enough slack is built on the service system to be able to service all foreseen tasks.

In the case of the capacity for servicing foreseen tasks, no significant difference was found for all the scenarios across the three strategies evaluated. The reason for this absence of significance differences is related to the same reason why the C-TOPTW lexicographic and C-TOPTW direct integration strategies do not differ significantly in handling unforeseen tasks: enough slack time was added to the shift duration such that servicing unforeseen tasks could be done without having to sacrifice servicing a foreseen task.

By observing specific solutions in the Euclidian plane as depicted in Figure ??, we can also derive an important qualitative observation about incorporating coverage considerations into routing problems. In Figure ?? we show solutions in the two layouts used with “snapshots” at the time each unforeseen task is released and at the end of the shift. Consider for example, the solution depicted for Layout B in Figure ?. At time  $t = 5.2$  an unforeseen task is released located where the triangle is. By observing the snapshot at time  $t = 46.7$  we see what decision was taken at  $t = 5.2$  to service such unforeseen task. Whereas Vehicle 1 travels backward to service the unforeseen task, Vehicle 2, reschedules its original foreseen task and to serve another task located at another position. This is an illustration of how routes consider coverage, as in effect, Vehicle 2, is re-allocating to a more central position (instead of going to the original extreme position), not because this is more efficient for a route, but because it can then cover a greater number of sites. In this way, including the coverage objective helps to coordinate the position of vehicles to minimize the risks of not servicing an unforeseen task released in the future. Later,

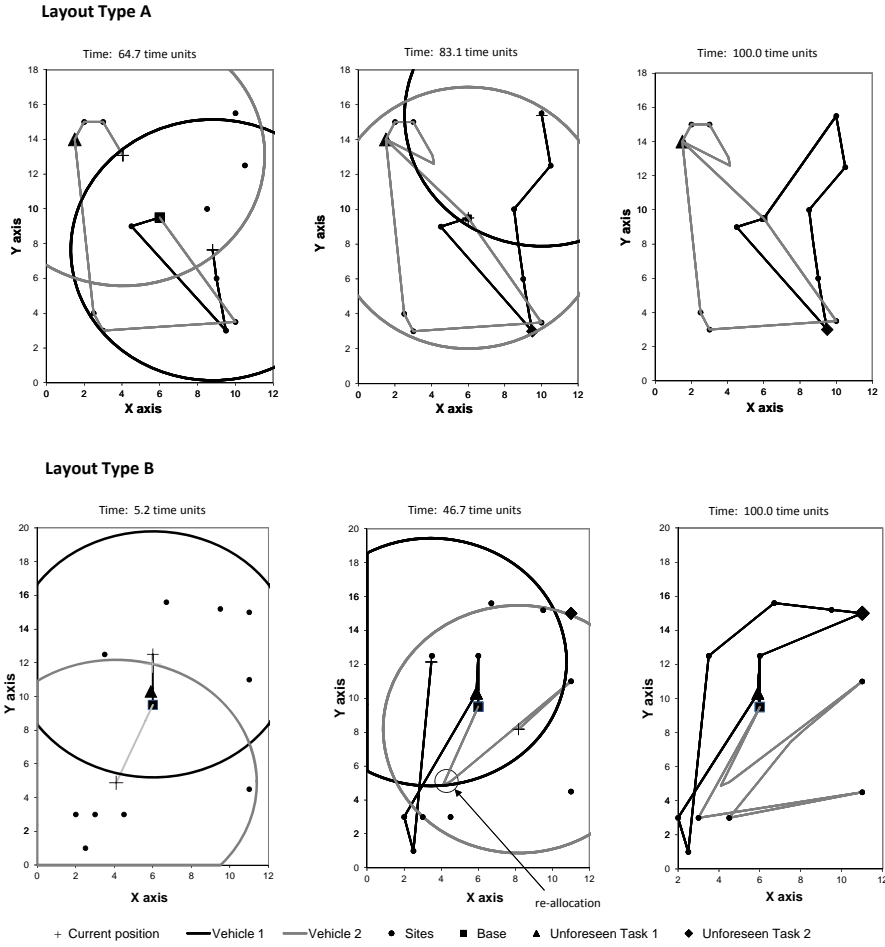


Fig. 4. Two simulation runs for different layouts

Vehicle 1 avoids servicing a near-by site and instead, travels further upwards towards the site located at (3.5, 12). In this way, the system avoids having both vehicles servicing at the same time, and having at least one vehicle partially covering covering sites for potential unforeseen tasks. This emphasizes that coverage is not only about coordinating vehicle positions but also the timing at which these are available for unforeseen tasks.

## 7 Conclusions

The SSTOP applied to service engineering is a new approach focusing on service quality rather than on costs through the consideration of specific service deadlines and response times required by clients. In addition to information about known tasks, we have also used in this paper incomplete information about possible unforeseen tasks. Moreover, it was shown that by tracking the position and state of each vehicle it is possible to devise a measure of reliability of constructed routes. We have observed that improvements in reliability

can be made by choosing among alternative equivalent optima of the TOPTW problem or at the expense of the performance of the known tasks. Through simulation experiments we observed that incorporating coverage considerations via the TAMCLP, improvements can be achieved making it possible to serve more unforeseen tasks on time.

Further investigation is required to distinguish the best way to best combine the TAMCLP and the TOPTW objectives in solving the SSTOP. The SSTOP can be further extended by adding the possibilities of delays at sites without immediate re-allocation to serve other tasks. Increased control on the vehicles' path may add new possibilities to obtain higher route reliabilities. Other possible extensions for the SSTOP include the use of individual response times, multiple traveling speeds and stochastic travelling times.

## References

Chao I., Golden B. and Wasil E. The team orienteering problem. *European Journal of Operational Research* 1996; 88 3; 464-474.

Daskin M., The maximal expected coverage location model: Formulation, properties and heuristic solution. *International Journal of Production Research* 1983; 17 1; 48-70.

Ehrgott M., *Multicriteria Optimization*, second ed. Springer Science & Media 2005.

Evers L, Dollevoet T.A.B., Barros A.I. and Monsuur H. *Robust UAV Mission Planning*. EI 2011-07, Econometric Institute, Erasmus University Rotterdam, 2011.

Feillet D., Dejax P. and Gendreau M. Traveling salesman problems with profits. *Transportation Science* 2005; 39 2; 188-205.

Golden B., Levy L. and Vohra R. The orienteering problem. *Naval Research Logistics* 1987; 34 3; 307-318.

Golden B. and Assad A. *Vehicle routing: methods and studies*. North-Holland: Amsterdam; 1988.

Hart C. The power of unconditional service guarantees. *Harvard Business Review* 1988; 88 4; 54-62.

Ilhan, T., Irvani, S. and Daskin, M., The orienteering problem with stochastic profits. *IIE Transactions* 2008; 40; 406421.

- Jaillet, P. Probabilistic traveling salesman problem (Ph.D. Thesis) Technical Report 1985; 185. Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA.
- Kantor M. and Rosenwein M. The orienteering problem with time windows. *Journal of the Operational Research Society* 1992; 43; 629–635.
- Laporte G. and Martello S. The selective traveling salesman problem. *European Journal of Operational Research* 1990; 177 1; 1-21.
- Mitrovic-Minic S. and Krishnamurti R. The multiple TSP with time windows: vehicle bounds based on precedence graphs. *Operations Research Letters* 2006; 34; 11-120.
- Nguyen M. and Gao M. Modeling and prototype development of an electronic tourist guide. Master thesis, Centre for Industrial Management, Katholieke Universiteit Leuven, Herverlee, Belgium; 2003.
- Liang Y. and Smith A. An ant colony approach to the orienteering problem. *Journal of the Chinese Institute of Industrial Engineers* 2006; 23 5; 403-414.
- Revelle C. Review, extension and prediction in emergency service siting models. *European Journal of Operational Research* 1989; 40 1; 58-69.
- Saydam C. and Atuğ C. Accurate estimation of expected coverage: revisited. *Socio-Economic Planning Sciences* 2003; 37 1; 69-80.
- Steuer R. Multiple criteria optimization theory, computation and application. 1986 John Willey and Sons; United States.
- Tasgetiren M. A genetic algorithm with an adaptive penalty function for the orienteering problem. *Journal of Economic and Social Research* 2002; 4 2; 1-27.
- Tsiligirides T. Heuristic methods applied to orienteering. *Journal of the Operations Research Society* 1984; 35 9; 797-909.
- Weintraub A., Aboud J., Fernandez C, Laporte G. and Ramirez E. An emergency vehicle dispatching system for an electric utility in Chile. *Journal of the Operational Research Society* 1999; 50 7; 690-696.
- Vansteenwegen P., Souffriau W. and Oudheusden D.V. The orienteering problem: a survey. *European Journal of Operational Research*. 2010; 209; 1-10.