

Section 1

Introduction

Traffic incidents not only greatly impact individuals, but also affect the general population. Depending on the incident severity, incidents are likely to cause both private and public property damage and possibly cause injury and fatalities. Due to both economic and humanitarian importance, maintaining roadway travel safety has aroused widespread interest in government officials, industry, and researchers. The National Highway Traffic Safety Administration estimates the cost of improving safety through various law enforcement activities to be as high as \$230.6 billion annually, nearly 2.3 percent of the nation's gross domestic product (Blincoe et al., 2002). Recently, a number of researchers (Steil and Parrish, 2009; Keskin et al., 2011; Lou et al., 2011; Willemse and Joubert, 2012) study the effectiveness of law enforcement plans, including how to improve patrolling plans. One way of improving patrolling efficiency is to focus on patrolling critical locations with high crash frequencies.

Given historical crash data, a “hot spot”(HS) is defined as a certain stretch of highway with high frequency of crashes of different severity over a given time period. In this problem, we are interested in finding the right start and stop locations (temporary stations) for state troopers at the beginning and end of their shift, respectively, as well as the patrol routes to visit time-critical HSs. Our overall goal is to maximize the visibility of state troopers during the *hot times* of the HSs while minimizing the costs associated with utilization of state troopers, traveling from one HS to another, and potential fees for temporary stations. Therefore, we tackle a bi-criteria (benefit maximization and cost minimization) optimization problem.

Specifically, we assume that at the beginning of a shift, state troopers start their patrol at temporary stations whose locations need to be determined from a list of potential stations. During their shift, starting at their selected temporary stations, the state troopers travel from one HS to another and stop at HSs during the effective coverage time, i.e. during the time interval that particular HS is critical. At the end of the shift, the state troopers go to other temporary stations so that the travel time from the last covered HS in the previous shift and the travel time to the next HS in the next shift is optimized. The locations of the ending temporary stations need to be determined as well. With these characteristics, this problem is similar to a *multi-depot (multiple temporary stations), dynamic location (changing locations), and routing problem (LRP)*. At the same time, since we are aiming to maximize the presence of the state troopers at the defined HSs and the service time at a HS can be viewed as the “variable profit,” the problem has resemblance to the *team orienteering problem with time windows (TOPTW)*.

We note that our paper is closely related to the work by Keskin et al. (2011), which focuses on patrol coverage of HSs. Considering a single station, Keskin et al. (2011) propose a mixed integer linear optimization model, called the maximum covering patrol routing problem (MCPRP), to maximize the presence of state troopers at the defined HSs for a given patrol shift. They show that the problem of interest is related to the TOPTW and prove that the MCPRP is *NP-hard*. They develop efficient local search- and tabu search-based heuristics to solve real life instances. In their results, they note that despite the effectiveness of the solutions, even with unlimited number of

state troopers, it is not possible to cover all of the time-sensitive HSs by just starting from a single station. HSs are geographically dispersed and time sensitive. By the time the state troopers reach a distant HS, the effective coverage would have already lapsed. Our work extends their paper in three directions:

- (i) We consider multiple temporary stations whose locations need to be determined as opposed to a single depot. This way, more HS are covered which are located out of the accessibility range with just one station.
- (ii) Our model spans multiple periods (shifts) as the locations of the HSs and temporary stations dynamically change and temporary station locations tie the multiple periods together.
- (iii) In addition to “coverage benefit” maximization, we also consider the minimization of total system costs (cost of utilization of troopers, travel costs, and temporary station location costs). With the addition of temporary stations, the coverage is expected to go up. But, it is also important to account for how much this coverage is going to cost. The costs included in the analysis create an immediate trade-off with respect to resource utilization and hot spot coverage. For instance, if fewer state troopers are dispatched or fewer temporary stations are available, the state troopers need to travel farther and spend more time on the road rather than covering HSs. On the other hand, if more state troopers are dispatched and more temporary stations are opened, there may not be enough monetary resources to pay for patrolling costs.

Since the MCPRP, which arises as a subproblem for our problem, and the dynamic location-routing problem are shown to be NP-hard, we resort to heuristic approaches. We first present a mixed integer programming formulation of the problem that can be solved via off-the-shelf software. Then, we develop efficient, tailored heuristics based on effective neighborhood searches embedded within a simulated annealing framework. When we compare the tailored heuristics with the off-the-shelf software, we see that our solutions provide good quality solutions in short periods of time. Additionally, we provide additional service measures including the percentage of number of HSs covered and percentage of coverage length based on the outcome of the heuristics. These service measures provide additional insights into the solutions.

The remainder of this paper is structured as follows: in Section 2, we present the literature review. In Section 3, the details of the general mathematical model are discussed, including necessary assumptions and notations. Next, in Section 4, we present the analysis of the problem and the solution approaches based on the characteristics of the problem. In Section 5, we discuss the computational results based on the heuristics and their implications. Finally, in Section 6, we summarize our results and offer recommendations for effective implementation.

Section 2

Literature Review

As our problem has similarities to TOPTW and LRP, we review both of these areas.

2.1 TOPTW

The OP is first introduced by Tsiligirides (1984) for the orienteering competition. The goal is to identify a circuit that maximizes collected profit such that travel costs do not exceed a preset value C . Some of its important variants include the team orienteering problem (TOP) where a fixed number of paths is considered, the orienteering problem with time windows (OPTW), and the team orienteering problem with time windows (TOPTW).

Boussier et al. (2007), Montemanni and Gambardella (2009), and Vansteenwegen et al. (2009) are the only people known to have solved the TOPTW. The exact branch-and-price algorithm proposed by Boussier et al. (2007) is generic enough to handle different kinds of OP, including the TOPTW. Montemanni and Gambardella (2009) develop local search and ant colony system algorithms based on the solution of a hierarchic generalization of TOPTW. Lastly, Vansteenwegen et al. (2009) present a straightforward and very fast iterated local search heuristic, which combines an insertion step and a shaking step, reverse insertion operation, to escape from local optima. Note that all of these papers only consider single period problems. To the best of our knowledge, only Tricoire et al. (2010) work on a multi-period OPTW problem. They design a variable neighborhood search based metaheuristic. However, they specify a fixed starting depot and a fixed stopping depot for each period for only one car, whereas in our case, the starting and ending locations are decision variables for multiple state troopers.

2.2 LRP

Since Salhi and Rand (1989) show that LRP consistently produces better solutions than solving sub-problems of facility location and vehicle routing sequentially, LRP has received increased attention from researchers. Laporte (1988) summarizes two-index or three-index vehicle flow formulations for static, deterministic LRP. For more information, please refer to the reviews by Balakrishnan et al. (1987), Min et al. (1998), and Nagy and Salhi (2007).

Both exact algorithms and heuristics are designed to solve LRP, but exact algorithms (see Labbé et al. (2004) and Laporte et al. (1986)) are still limited to small to medium size problems and heuristics are far more prevalent. Nagy and Salhi (2007) categorize heuristics into sequential, clustering, iterative (Hansen et al. (1994), Perl and Daskin (1985), and Wu et al. (2002)), and hierarchical heuristics (Albareda-Sambola et al. (2007), Melechovskỳ et al. (2005), and Nagy and Salhi (1996)). Among these four categories, the last two are preferred as sequential and clustering heuristics fail to utilize feedback between location and routing subproblems. Especially when

there is hierarchy involved, hierarchical heuristics are shown to be more effective. A hierarchical heuristic divides LRP into a master problem location and its subordinate routing problem. We follow this logic in the development of our heuristics.

We note two important differences between our work and the literature. First, our problem has time window limitation that has not been addressed in LRP literature before to the best of our knowledge. Even though Nagy and Salhi (2007) point out in their survey that work by Semet and Taillard (1993) belongs to this category, that paper should be viewed as VRPTW literature instead of LRP since there are no location decisions. Second, instead of locating long-term depots, we locate temporary stations while optimizing routing schedules. For our problem, both location and routing are short-term decisions, avoiding the common criticism that LRP has conflicting planning horizons in the short and long run.

Section 3

General Model

3.1 Problem Definition

As discussed earlier, it is assumed that at the beginning of each shift, state trooper cars are dispatched from temporary stations (TS), where the potential locations are given as $i \in I = \{1, 2, \dots, |I|\}$, that also include state trooper posts. $\mathcal{K} = \{1, 2, \dots, |\mathcal{K}|\}$ is the set of the available state troopers, and each trooper on duty incurs a cost of v (\$/shift/trooper). Let $\mathcal{P} = \{1, 2, 3\}$ be the set of shifts, representing morning, afternoon, and night shifts, and $\mathcal{D} = \{1, 2, \dots, |\mathcal{D}|\}$ be the set of days, where 1 represents the first day and so on. As a simplification, pairs of $p \in \mathcal{P}$ and $d \in \mathcal{D}$ can be represented by a single period index $t \in \mathcal{T} = \{1, \dots, |\mathcal{P}| \times |\mathcal{D}|\}$.

Within a subset of regions with given potential locations for TS $i \in I$ and during a particular period $t \in \mathcal{T}$, there are historically established HS, $j = 1, \dots, n$. In a period $t \in \mathcal{T}$, HS j has three attributes: (i) location on the mile-posted road network; (ii) the time window $[e_j^t, l_j^t]$ when HS j becomes “hot” where e_j^t and l_j^t are the start and end times of the “hotness” window; and (iii) weight w_j^t representing severity level. By definition, $e_j^t \leq l_j^t$. Furthermore, we assume that without loss of generality, \mathcal{N} is an ordered set according to e_j^t such that $e_1^t \leq e_2^t \leq \dots \leq e_n^t$. Note that a location can be listed as two different HSs i and j , where $e_i^t < e_j^t$ if it becomes “hot” twice within the same period.

Let $\mathcal{V} = \mathcal{N} \cup I$ denote the union of the sets of HS and locations of potential TS. Additionally, we let $\mathcal{E} = \{(i, j) : i, j \in \mathcal{N} \cup I, i \neq j\}$ define the set of edges. The connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ represents the underlying road network. $d_{ij}^t > 0$ denotes the shortest travel time from HS i to j , $\forall (i, j) \in \mathcal{E}$, and in period $t \in \mathcal{T}$. Meanwhile, we define:

- $\Delta^+(i) = \{j \in \mathcal{V}, t \in \mathcal{T} : (i, j) \in \mathcal{E}, e_i^t + d_{ij}^t \leq l_j^t\}$ as the set of vertices that are directly reachable from $i \in \mathcal{V}$ within the time window, and
- $\Delta^-(i) = \{j \in \mathcal{V}, t \in \mathcal{T} : (j, i) \in \mathcal{E}, e_j^t + d_{ji}^t \leq l_i^t\}$ as the set of vertices from which i is directly reachable.

The additional assumptions of the model include the following:

1. The fixed cost of TS is negligible.
2. There is no capacity limit at TS; i.e., multiple state trooper cars can start/stop at the same TS, if desired.
3. Visits of state troopers at HSs are only effective within the time windows of HSs.
4. At the beginning of a shift, a state trooper leaves a selected TS, and at the end of the shift, he may or may not come back to the same TS.
5. State trooper cars travel at a constant speed of 60 miles/hour, thus 1 minute corresponds to 1 mile. This way, distance and time can be easily translated to one other.

6. State troopers can choose whether to visit a HS or not, as well as time to begin and end the coverage. If a HS is chosen by a state trooper, it cannot be visited by others.

Our goal is to optimize the dynamic selection of TS utilized each period, allocate state troopers to TS, and route state troopers to HSs simultaneously. Figure 1 shows an example with 5 potential TS, 3 available state troopers, 2 periods, and 16 HSs per period. At the beginning, these 3 cars are parked at TSs 2, 3, and 5. The routes, represented by the directed arrows, form a feasible solution while meeting the time windows of the visited HSs. In the first period when $t = 1$, all 3 troopers are utilized; when $t = 2$, only 2 troopers are utilized due to budget limitations. That is, the cost minimization outplays the benefit maximization. When $t = 1$, $k = 1$ starts at TS= 2 but ends at TS= 1, $k = 3$ starts at TS= 5 but ends at TS= 3, and only $k = 2$ starts and ends at the same TS.

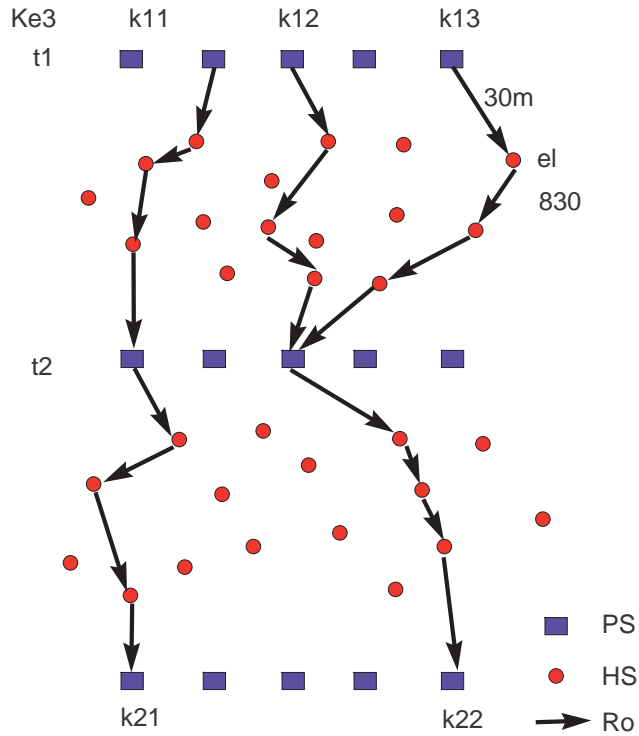


Figure 1: A representative example

3.1.1 Decision Variables

We define five sets of decision variables: (i) $x_{ijk}^t = 1$, if state trooper car $k \in \mathcal{K}$ travels from i to j , $(i, j) \in \mathcal{E}$ during $t \in \mathcal{T}$, and 0, otherwise. (ii) $s_{ik}^t \geq 0$, the starting time of service for state trooper car $k \in \mathcal{K}$ at HS $i \in \mathcal{V}$ during $t \in \mathcal{T}$. (iii) $f_{ik}^t \geq 0$, the time state trooper car $k \in \mathcal{K}$ leaves HS $i \in \mathcal{V}$ during $t \in \mathcal{T}$, i.e., the end of service. (iv) $y_{ik}^t = 1$ if state trooper k serves $i \in \mathcal{V}$ during $t \in \mathcal{T}$, 0, otherwise. (v) $R_{ijk}^t = 1$ if state trooper car $k \in \mathcal{K}$ is relocated from one TS i to another TS j at the end of $t \in \mathcal{T}$, $i, j \in I$.

3.1.2 Objective

We have a multi-objective optimization problem, including cost (trooper utilization cost, routing cost, and facility cost) minimization and benefit (coverage) maximization. All cost parameters are scaled down to the same time span, that is, one shift. We set $v = \$36.63/\text{shift}/\text{trooper}$ assuming an average wage of $\$40,000/\text{year}/\text{trooper}$. We set the trip cost as $c = \$0.14/\text{mile}$ or $\$0.14/\text{minute}$ assuming that fuel price is $\$3.5/\text{gallon}$ and fuel consumption is $0.04 \text{ gallon}/\text{mile}$. Then our objective is:

$$\min_{\mathbf{x}} \left(v \sum_{t \in \mathcal{T}} \sum_{i \in I} \sum_{j \in \mathcal{N}} \sum_{k \in \mathcal{K}} x_{ijk}^t + c \sum_{t \in \mathcal{T}} \sum_{(i,j) \in \mathcal{E}} \sum_{k \in \mathcal{K}} d_{ij}^t x_{ijk}^t \right) \quad (1)$$

$$\max_{\mathbf{f}, \mathbf{s}} \sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{N}} \sum_{k \in \mathcal{K}} (f_{jk}^t - s_{jk}^t) w_j^t \quad (2)$$

For multi-objective optimization problems, it is very common that objectives may not be commensurate with each other. Similarly, for our problem, the coverage benefit is measured in minutes whereas the total cost is measured in dollars. Facing this dilemma, the vast majority of researchers use either weighted sum of the objectives or ϵ -constraint approach. The first group of researchers (Alçada-Almeida et al., 2009; Alumur and Kara, 2007; Caballero et al., 2007) transformed conflicting objectives into a weighted sum by attaching each objective with a coefficient. However, due to the arbitrary choices of coefficients, we adopt the other commonly used method: ϵ -constraint approach (Bérubé et al., 2009; Chankong and Haimes, 1983; Laumanns et al., 2005, 2006; Mavrotas, 2009; Miettinen, 1999). This approach considers the single most important objective and puts all of the other objective(s) into the formulation as constraint(s). Thereafter, our problem is transformed into a benefit maximization problem by setting an upper limit on the budget, say B , $v \sum_{t \in \mathcal{T}} \sum_{i \in I} \sum_{j \in \mathcal{N}} \sum_{k \in \mathcal{K}} x_{ijk}^t + c \sum_{t \in \mathcal{T}} \sum_{(i,j) \in \mathcal{E}} \sum_{k \in \mathcal{K}} d_{ij}^t x_{ijk}^t \leq B$. In our computational experiments, we test different levels of B to demonstrate the effect of costs and available budgets on the patrol routes.

3.1.3 Constraints

We categorize our constraints under five groups: schedule feasibility constraints (1a)-(1d), route structuring constraints (2a)-(2e), TS updating constraints (3a)-(3d), car related constraints (4), and last but not the least, integrality and non-negativity constraints (5a)-(5b). In constraints (1a), $M_{ij}^t = \max\{l_i^t + d_{ij}^t - e_j^t, 0\} \geq 0$, and in constraints (3d), D_{limit} is a constant that is set to 20 minutes.

Schedule feasibility	$f_{ik}^t + d_{ij}^t - s_{jk}^t \leq (1 - x_{ijk}^t)M_{ij}^t,$	$\forall t \in \mathcal{T}, (i, j) \in \mathcal{E}, k \in \mathcal{K}.$	(1a)
	$e_i^t \sum_{j \in \Delta^+(i)} x_{ijk}^t \leq s_{ik}^t,$	$\forall t \in \mathcal{T}, i \in \mathcal{V}, k \in \mathcal{K}.$	(1b)
	$l_i^t \sum_{j \in \Delta^+(i)} x_{ijk}^t \geq f_{ik}^t,$	$\forall t \in \mathcal{T}, i \in \mathcal{V}, k \in \mathcal{K}.$	(1c)
	$s_{ik}^t \leq f_{ik}^t,$	$\forall t \in \mathcal{T}, i \in \mathcal{V}, k \in \mathcal{K}.$	(1d)
Route structuring	$\sum_{i \in \Delta^-(j)} x_{ijk}^t = \sum_{i \in \Delta^+(j)} x_{jik}^t,$	$\forall t \in \mathcal{T}, j \in \mathcal{N}, k \in \mathcal{K}.$	(2a)
	$\sum_{j \in \Delta^+(i)} x_{ijk}^t \leq y_{ik}^t,$	$\forall t \in \mathcal{T}, i \in I, k \in \mathcal{K}.$	(2b)
	$\sum_{j \in \Delta^-(i)} x_{jik}^t \leq y_{ik}^{t+1},$	$\forall t \in \mathcal{T}, i \in I, k \in \mathcal{K}.$	(2c)
	$\sum_{j \in \Delta^+(i)} x_{ijk}^t = y_{ik}^t,$	$\forall t \in \mathcal{T}, i \in \mathcal{N}, k \in \mathcal{K}.$	(2d)
	$\sum_{k \in \mathcal{K}} y_{jk}^t \leq 1,$	$\forall t \in \mathcal{T}, j \in \mathcal{N}.$	(2e)
TS updating	$R_{ijk}^t \leq y_{ik}^t,$	$\forall t \in \mathcal{T}, i, j \in I, j \neq i, k \in \mathcal{K}.$	(3a)
	$R_{ijk}^t \leq y_{jk}^{t+1},$	$\forall t \in \mathcal{T}, i, j \in I, j \neq i, k \in \mathcal{K}.$	(3b)
	$R_{ijk}^t \geq y_{ik}^t + y_{jk}^{t+1} - 1,$	$\forall t \in \mathcal{T}, i, j \in I, j \neq i, k \in \mathcal{K}.$	(3c)
	$d_{ij}^t R_{ijk}^t \leq D_{\text{limit}},$	$\forall t \in \mathcal{T}, i, j \in I, j \neq i, k \in \mathcal{K}.$	(3d)
Car related	$\sum_{i \in I} y_{ik}^t \leq 1,$	$\forall t \in \mathcal{T}, k \in \mathcal{K}.$	(4)
Integrity and non-negativity	$s_{ik}^t, f_{ik}^t \geq 0,$	$\forall t \in \mathcal{T}, i \in \mathcal{V}, k \in \mathcal{K}.$	(5a)
	$x_{ijk}^t, y_{ik}^t, R_{ijk}^t \in \{0, 1\},$	$\forall t \in \mathcal{T}, i, j \in \mathcal{V}, k, g \in \mathcal{K}, g > k.$	(5b)

This model generalizes the MCPRP by Keskin et al. (2011) from single-depot to multi-depot and from a static depot location to dynamic depot locations. The main modification to the formulation involves the newly added third and fourth sets of constraints. The third set of Constraints (3a)-(3c) are confining $R_{ijk}^t = 1$ if and only if $y_{ik}^t = y_{jk}^{t+1} = 1$, i.e., we relocate a state trooper from one TS to another. If relocation occurs, the distance between the starting and the stopping TS should not exceed D_{limit} , which is achieved by Constraints (3d). This is a practical constraint required by the state troopers. In the fourth set, Constraints (4) stipulate that one car can only be parked at one TS.

3.1.4 Overall Model

The overall model is subject to constraints (1a)–(5b). We call this model, the dynamic multi-depot MCPRP, in short, DMD-MCPRP.

Remark 1 *If a state trooper must go back to where he starts his shift, R_{ijk}^t and its related constraints are not needed any more. This is a special case of DMD-MCPRP, which can be solved by period and independently.*

3.2 Extension with Fixed Charge Considerations

We now consider the case when there is a fixed cost associated with each utilized TS. This case is more applicable if, for instance, each TS is charged with some parking fee, denoted as F_i .

To incorporate the fixed costs into the model, another set of decision variables is needed. We define $z_i^t = 1$ if TS $i \in I$ is open in $t \in \mathcal{T}$, 0, otherwise. With this new variable, the model requires the following two updates. First, the budget constraint has one additional term of the total fixed cost, that is,

$$v \sum_{t \in \mathcal{T}} \sum_{i \in I} \sum_{j \in \mathcal{J}} \sum_{k \in \mathcal{K}} x_{ijk}^t + c \sum_{t \in \mathcal{T}} \sum_{(i,j) \in \mathcal{E}} \sum_{k \in \mathcal{K}} d_{ij}^t x_{ijk}^t + \sum_{t \in \mathcal{T}} \sum_{i \in I} F_i z_i^t \leq B.$$

Second, the model is augmented with one additional set of constraints, guaranteeing that TS is marked as open if selected.

$$\text{TS selection} \quad z_i^t \geq y_{ik}^t, \quad \forall t \in \mathcal{T}, i \in I, k \in \mathcal{K}. \quad (7)$$

Section 4

Solution Approaches

We observe that both DMD-MCPRP and its extended model are mixed integer linear programs (MILP) and can be solved by CPLEX 12.1*. Unfortunately, even for very small instances such as the example in Figure 1, CPLEX runs out of memory.

Among different solution options, we choose a hierarchical heuristic as our problem has an obvious hierarchical structure. As our objective is transformed into a benefit maximization by stating the incurred costs under a budget limit, we first solve the multi-depot MCPRP (MD-MCPRP), and then the locations of the temporary stations are determined. This decomposition makes the multi-depot MCPRP problem solvable by period and by shift. Therefore, it provides an opportunity to utilize the solution of Keskin et al. (2011) with slight modifications due to multi-depot considerations. We solve the location problem via a greedy heuristic. We iterate among these two problems to search for better feasible solutions for the overall problem. We first discuss the details of the heuristic for the base model (without TS location costs) and then present a modified heuristic to handle the extended model (with TS fixed costs) next.

4.1 Heuristic for DMD-MCPRP

For the base model when the fixed costs of TS are negligible, the optimal solution has the following characteristic:

Observation 1 *If the optimal HS routes in MD-MCPRP are known, the nearest TSs to the first and last HSs in the routes are selected as start and stop locations.*

Based on this observation, we build our heuristic approach. First, note that the first problem is the multi-depot MCPRP that determines the multi-car routing among HSs to maximize the benefits of visiting HSs. This problem ignores the selection of locations for TSs (depots) and the budget limit temporarily. However, in order to initiate the building of the routes, we need initial starting locations for the routes. For this purpose, we use three initialization strategies: (i) **STR1**: start at the HS with the earliest time window; (ii) **STR2**: start at the HS with the highest weight; (iii) **STR3**: use a combination of STR1 and STR2: that is, out of the first 5 earliest HSs, pick the HS with the highest weight. The heuristic is run using one of these strategies, and we report the computational results with different strategies in Section .

The heuristic has five components that include **initialization**, **MCPRP algorithm** (Keskin et al., 2011), **add/drop**, **insert/erase**, and **simulated annealing**. The pseudo-code of the algorithm that explains how these components are utilized is given in Display 1. Next, we explain the details of each component.

*CPLEX is a trademark of IBM.

Display 1 *DMD – MCPRP heuristic*(Obj^* , Res^* , Rou^* , Car_t^*)

- 1: **Initialization:** $Obj^* = \infty$, $Res^* = 0$, $Rou^* = \emptyset$, $Car_t^* = 0$. Equally allocate $Car_t^* = \min\{(1 - p) \frac{\mathcal{B}}{|\mathcal{T}|v}, |\mathcal{K}|\}$ cars $\forall t \in \mathcal{T}$. Use a starting strategy to pick starting location for Car_t^* , $\forall t \in \mathcal{T}$.
 - 2: **MCPRP**[Obj^* , Res^* , Rou^*].
 - 3: **Add/Drop**[Obj^* , Res^* , Rou^* , Car_t^*].
 - 4: Simply pick the closest TS to the starting and stopping end points within D_{limit} .
 - 5: **if** Budget allows **then**
 - 6: **Insert**[Obj^* , Res^* , Rou^*] between the starting TS and starting end point;
 - 7: **else**
 - 8: **Erase**[Obj^* , Res^* , Rou^* , Car_t^*].
 - 9: **end if**
 - 10: **Simulated annealing**[Obj^* , Res^* , Rou^* , Car_t^*].
 - 11: Return Obj^* , Res^* , Rou^* , Car_t^* .
-

4.1.1 Initialization

In the **Initialization** step, we first initialize the objective coverage Obj^* , resource consumption level Res^* , and the set of route sequence information Rou^* . To determine the number of cars available in each period Car_t^* , we compare the available budget for employees with the total number of cars. To allocate the budget per car appropriately, we assume that employee salary portion of the budget is divided equally among each period and that the maximum number of cars in a given period cannot exceed $(1 - p) \frac{\mathcal{B}}{|\mathcal{T}|v}$, where $(1 - p)$ is the portion of the budget spent on employee salary. Then, the initial number of available cars in period t is calculated as $\min\{(1 - p) \frac{\mathcal{B}}{|\mathcal{T}|v}, |\mathcal{K}|\}$. Afterwards, using one of the aforementioned starting strategies, we initialize the starting locations of each car.

Given the number of cars and their starting locations, we utilize the MCPRP algorithm developed by Keskin et al. (2011). This algorithm builds Car_t^* routes in a greedy fashion, improved with exchange and relocate operators.

4.1.2 Add/Drop Component

After the MCPRP algorithm, the initial routes are built up for all periods. Using this information, we calculate the resource consumption Res^* by taking into account the travel costs incurred by the formed routes. If the consumed resource level Res^* is less than $\mathcal{B} - (1 + \frac{p}{1-p})v$ and there is an available (unused) state trooper car, we can add one more patrol route to the period with the largest number of uncovered HSs, i.e., $Car_t \leftarrow Car_t + 1$. Note that $(1 + \frac{p}{1-p})v$ is an approximate cost for utilizing one car (v) and traveling to hot spots ($\frac{pv}{1-p}$). Until all of the budget is used or all of the state trooper cars are utilized, we keep adding a new patrol route. Each new route is again built using the MCPRP algorithm.

On the other hand, if the total resource consumption Res^* after the initial route construction

exceeds the available budget by more than ν , i.e., $Res^* > \mathcal{B} + \nu$, we eliminate the route with the least coverage time until the total budget is controlled.

4.1.3 Selecting TS Locations

The next step in the overall algorithm involves selecting TS locations. By observation 1, for each state trooper, we simply pick the closest TSs to the starting and stopping HSs in his route to begin and end his shift as long as the start and end TSs are within distance D_{limit} . In other words, as opposed to considering all of the candidate TS locations, we only consider the ones within distance D_{limit} . We repeat this process for all $t \in \mathcal{T}$. In essence, this step achieves the goal of picking a common TS which has the shortest travel distance from the ending HS of one period to the starting HS of the next period in a myopic fashion. After all of the TS locations are selected, the routes are formed. After this component, the heuristic completes a location-routing cycle. However, the budget may still be violated. Therefore, the next two components (**Insert** and **Erase**) improve this location-routing solution by taking the budget limit into account.

4.1.4 Insert/Erase Component

As traveling from the selected TS locations to the HSs increases the resource consumption, the new resource consumption may exceed the budget limit. If the budget is exceeded, **Erase** keeps deleting the HS with the least coverage time until the resource consumption is within budget limits. One possible result of this operation is that all of the HSs of a route are removed. If this is the case, then that route does not cover any HS other than TS, and this route is closed. The state trooper car is, therefore, freed up. The number of cars used in that period decreases by one and the resource consumption is reduced by the utilization cost ν . Since now additional resources are made available, the **Insert** component is called to insert any uncovered HS while considering the travel costs as well as the coverage benefit obtained from the inclusion of this HS.

On the other hand, if the inclusion of travel costs from and to selected TS locations into the resource consumption does not exceed the budget limitation, we may re-call the **Insert** component to include uncovered HSs until all of the budget is utilized.

Display 2 Procedure $SA(Obj^0, Res^0, Rou^0, Car_t^0, T_0, \alpha, \beta, M, MaxTime)$

```
1:  $T = T_0$ .
2:  $Res^{current} = Res^0; Rou^{current} = Rou^0; Car_t^{current} = Car_t^0; Obj^{current} = Obj^0$ .
3:  $Time = 0$ .
4: while  $Time \leq MaxTime$  do
5:   Discard chosen TS, and update  $Obj^{current}$ ,  $Res^{current}$ , and  $Rou^{current}$ .
6:    $Res^* = Res^{current}; Rou^* = Rou^{current}; Car_t^* = Car_t^{current}; Obj^* = Obj^{current}$ .
7:   Add/Drop $[Obj^{new}, Res^{new}, Rou^{new}, Car_t^{new}]$ .
8:   Select TSs.
9:   if Budget allows then
10:    Insert $[Obj^{new}, Res^{new}, Rou^{new}]$ ;
11:   else
12:    Erase $[Obj^{new}, Res^{new}, Rou^{new}, Car_t^{new}]$ .
13:   end if
14:   Call Metropolis $(Res^{current}, Rou^{current}, Car_t^{current}, Obj^{current}, T, M)$ ;
15:    $Time = Time + M$ ;
16:    $T = \alpha \cdot T; M = \beta \cdot M$ .
17: end while
18: return  $Res^*, Rou^*, Car_t^*, Obj^*$ .
```

4.1.5 Simulated Annealing Component

To optimize the patrol routes and TS locations, we develop a simulated annealing algorithm. Simulated annealing (SA), first proposed by Kirkpatrick et al. (1983), is one of the most well-developed and widely used iterative techniques for solving optimization problems (Sait and Youssef, 1999). The basic requirements of the SA algorithm are a neighborhood structure on the set of feasible solutions and a number of parameters which govern the acceptance or rejection of new solutions generated during the search. In our SA implementation, we utilize relocate and exchange neighborhoods to improve the routes by considering different HS inclusions.

SA is a randomized search method that tries to improve a solution by a random walk in the solution space and gradually adjusting a parameter called *temperature*. The sequence of temperatures and the number of iterations, for which they are maintained, are called the *annealing schedule*. The quality of the solution is very sensitive to both of these factors. Therefore, the SA algorithm requires an initial temperature, T_0 ; a cooling rate, α ; a progressive factor, β ; the total allowed time for the annealing process, $MaxTime$; and, finally, the time until the next parameter update, M (Sait and Youssef, 1999, pages 53-55). In our implementation, we experimented extensively to find an effective combination of these parameters. We set $T_0 = 1000$, $\alpha = 0.9$, $\beta = 2$, $MaxTime = 8000$, and $M = 2$. The details of the simulated annealing metaheuristic are given in Display 2. The core of the SA algorithm is the *Metropolis* procedure. The *Metropolis* procedure, after receiving the current solution $Res^{current}, Rou^{current}, Car_t^{current}$, the temperature, T , and the number of metropolis loops, M , as inputs, simulates the annealing process at a given temperature T . In the *Metropolis* procedure, we utilize exchange and relocate neighborhoods, similar to Keskin et al. (2011), to define a new solution. We accept the “first-best-solution” in the neighborhoods. The *Metropolis* procedure is presented in Display 3.

Display 3 Procedure *Metropolis*($Res^{current}, Rou^{current}, Car_t^{current}, Obj^{current}, T, M$):

```
1: while  $M > 0$  do
2:   Relocate & exchange operator[ $Obj^{new}, Res^{new}, Rou^{new}$ ].
3:    $\Delta Obj = Obj^{current} - Obj^{new}$ .
4:   if  $\Delta Obj \leq 0$  then
5:      $Rou^{current} = Rou^{new}; Car_t^{current} = Car_t^{new}; Obj^{current} = Obj^{new}$ .
6:     if  $Obj^{new} \leq Obj^*$  then
7:        $Rou^* = Rou^{new}; Car_t^* = Car_t^{new}$ ; and  $Obj^* = Obj^{new}$ .
8:     end if
9:   else
10:    if  $Random < \exp(-\frac{\Delta Obj}{T})$  then
11:       $Rou^{current} = Rou^{new}; Car_t^{current} = Car_t^{new}$ ; and  $Obj^{current} = Obj^{new}$ .
12:    end if
13:  end if
14:   $M = (M - 1)$ .
15: end while
16: return  $Obj^*, Res^*, Rou^*, Car_t^*$ 
```

4.2 Modification of the Heuristic for the FC Model

For the extended model, we revise DMD-MCPRP heuristic to encompass the fixed cost of TS. Specifically, the inclusion of fixed costs changes two main components of the algorithm. First, instead of locating the TS locations based on proximity to the starting and ending HSs in the route, we utilize a cost-based approach. We select TS locations with the smallest $cd_{ij}^t; F_i$ among the potential TS locations that conform to D_{limit} . Secondly, since **Add/Drop** most aggressively adjusts the resource consumption by changing the number of routes Car_t , the algorithm moves onto the modification of TS locations after the resource consumption reaches the total available budget. To improve on the selection of TS locations, we include a **Decrease TS** component that adjusts the resource consumption less aggressively by dropping one open TS at one time until the resource consumption Res drops to $\mathcal{B} + pv$. The rest of the algorithm, including the SA component, stays intact.

Section 5

Computational Experiments

In order to test the proposed models and solution approaches, we design small- to medium-size instances from crash history data in the state of Alabama. All of the crash data in the state of Alabama since 2001 is collected by Critical Analysis Reporting Environment (CARE), a data analysis software package developed by researchers at the University of Alabama (Steil and Parrish, 2009). To determine the effects of various factors on the performance of the heuristics as well as the coverage benefits, we design a set of experiments by varying the number of periods $|\mathcal{T}|$, the number of HSs per period $|\mathcal{N}|$, the number of depots $|\mathcal{A}|$, the number of cars $|\mathcal{K}|$, and the number of TSs $|I|$. We assume $|\mathcal{K}|$ is correlated with $|\mathcal{A}|$ and $|I|$ is correlated with $|\mathcal{T}|$ and $|\mathcal{N}|$. That is, if there are more depots, there should also be proportionally more cars; likewise more $|I|$. Once the number of HSs are determined by the experimental design, we use CARE to extract the necessary HS information related to location, HS duration, and time window considerations. With this construction, our design has $2^5 = 32$ instances. The details are provided in Table 1.

Table 1: Design of experiment.

Item	Small	Medium
$ \mathcal{T} $	2	4
$ \mathcal{N} $	16	32
$ \mathcal{A} $	2	3
$ \mathcal{K} $	$2 \mathcal{A} $	$3 \mathcal{A} $
$ I $	$1/8 \mathcal{T} \times \mathcal{N} $	$1/4 \mathcal{T} \times \mathcal{N} $

Based on the aforementioned design, we test all instances for

- two weight schemes w^f : high variance (1, 1.5, 2), and low variance (1, 1.1, 1.2);
- three starting strategies: **STR1**, **STR2**, and **STR3**;
- three routing cost allocation percentage levels p : 0.25, 0.5, and 0.75; and
- five budget levels: 20%B, 40%B, 60%B, 80%B, and 100%B, where B is the total cost estimated when all $|\mathcal{K}|$ troopers are used for each period, and all HSs are covered on a straight-and-back basis.

In total, we run $32 \times 2 \times 3 \times 3 \times 5 = 2880$ instances. We conduct all of these experiments using C++ on an Intel Core 2 Duo E8400 with 2.94GB of memory. Our proposed metaheuristics return the coverage benefit under the given budget limit. Meanwhile, since our model is a MILP, CPLEX is able to generate lower bound (LB)- a feasible solution as a benchmark to heuristics. Note that in many instances, CPLEX could not solve the problem. Therefore, it is possible that the heuristics developed are better than the lower bound of the CPLEX. To avoid running out of memory, CPLEX is set to run up to 3600 seconds.

5.1 Experiment for DMD-MCPRP

After obtaining the coverage objective from the heuristic and the LB from CPLEX, we evaluate our solution approach by examining the gap: $(Objective - LB)/LB$. If the gap is positive, our heuristic finds better solution than the best feasible solution that CPLEX is able to find within the given runtime. However, it is also possible that the LB of CPLEX is better than our heuristic, i.e., the gap is negative. We report both the average and maximum gap, in short ‘‘Avg.’’ and ‘‘Max.’’ in Table 2.

Table 2: Performance gap between the metaheuristic and CPLEX.

	$w_i^d=(1, 1.1, 1.2)$								$w_i^d=(1, 1.5, 2)$							
	Avg. (%)				Max. (%)				Avg. (%)				Max. (%)			
	Str1	Str2	Str3	Best	Str1	Str2	Str3	Best	Str1	Str2	Str3	Best	Str1	Str2	Str3	Best
p=0.25																
100%B	2.3	0.8	1.4	2.6	22.2	20.9	18.9	22.2	0.6	-0.4	0.0	1.2	22.8	19.7	19.6	22.8
80%B	1.7	0.2	0.8	2.0	18.0	14.8	16.0	18.0	1.1	0.0	0.4	1.6	23.4	20.2	20.2	23.4
60%B	1.6	0.1	0.8	2.0	20.5	19.2	17.3	20.5	0.6	-0.4	0.0	1.2	13.6	11.1	10.4	13.6
40%B	0.4	-1.6	-0.7	0.8	21.5	19.9	18.1	21.5	0.9	-0.5	-0.2	1.4	21.3	20.0	19.3	21.3
20%B	-1.4	-5.0	-2.8	-0.6	55.9	53.3	55.9	55.9	-0.2	-3.7	-1.7	0.9	56.1	56.1	51.5	56.1
p=0.5																
100%B	2.3	0.8	1.4	2.6	22.2	20.9	18.9	22.2	0.6	-0.4	0.0	1.2	22.8	19.7	19.6	22.8
80%B	1.7	0.2	0.8	2.0	18.0	14.8	16.0	18.0	1.1	0.0	0.4	1.6	23.4	20.2	20.2	23.4
60%B	1.2	-0.5	0.5	1.7	20.5	19.2	17.3	20.5	0.6	-0.6	-0.3	1.0	13.6	11.1	10.4	13.6
40%B	-1.6	-2.2	-1.4	-0.2	21.5	19.9	18.1	21.5	-0.5	-1.0	-0.3	0.6	21.3	20.0	19.3	21.3
20%B	-3.7	-4.8	-4.6	-2.2	57.0	57.9	58.5	58.5	-3.1	-2.9	-2.5	-0.4	60.0	53.7	53.5	60.0
p=0.75																
100%B	1.6	-0.1	0.8	2.0	22.2	20.9	18.9	22.2	-0.1	-1.1	-0.8	0.5	22.8	19.7	19.6	22.8
80%B	-0.5	-1.2	-0.4	0.6	18.0	14.8	16.0	18.0	-0.6	-0.8	-0.4	0.6	23.4	20.2	20.2	23.4
60%B	-1.9	-1.5	-0.8	0.2	17.7	16.0	16.5	17.7	-2.3	-1.4	-1.7	-0.3	13.1	12.1	14.1	14.1
40%B	-4.6	-3.7	-3.6	-2.4	19.4	18.8	18.5	19.4	-4.5	-2.0	-2.5	-1.0	15.5	18.4	20.0	20.0
20%B	-6.7	-6.6	-5.5	-3.5	54.8	57.9	56.7	57.9	-3.8	-3.2	-3.7	-1.1	52.1	52.5	56.7	56.7

In Table 2, if we compare different budget levels, there is a general trend: as the budgets become tighter and tighter, the average gaps become slightly worse. At 100%B, 80%B, and 60%B, with the best starting strategy, our heuristic outperforms the LB returned by CPLEX. Therefore, if there is enough budget, our metaheuristic displays a dominating advantage over CPLEX. On the other hand, if the budget is tight, this dominance is only compromised slightly. The inclusion of the budget limit allows us to conduct the benefit-cost tradeoff analysis; its result shows how much a change in the budget will affect the patrol effectiveness.

Next, we compare different starting strategies. Both with lower and higher variance weights, **Str1** has the best gap for $p = 0.25$ with all budget levels and for $p = 0.50$ with most budget levels. However, for $p = 0.75$, there is no consistent result with respect to which one is the best. For instance, with lower variance weights, **Str3** has the best gap for most budget levels and with high variance weights, **Str2** has the best gap for budget levels 60%B, 40%B, and 20%B. Because of the lack in consistency, we recommend to try all of the starting strategies and proceed with the best one. Since the heuristic is running fast, this does not create additional problems.

Third, different route cost allocation factors p do not affect the coverage benefit. Especially, in the first two rows with $p = 0.25$ and those with $p = 0.5$, the results are exactly the same. Regardless of how much budget is allocated to gas consumption in the beginning of the algorithm, the inherent **Add/Drop** component adjusts the number of cars very effectively. Therefore, the

proposed metaheuristic is robust. This point is substantiated further, if we compare different w_i^t . Low-variance weights and high-variance weights have quite similar coverage benefits and similar heuristic performances. The robustness of our approach is critical when decision makers have different perceptions of different crash types and assign different weights to them.

Overall, the best performances by the heuristic outperform those by CPLEX. The largest improvement reaches up to 60.0%, that is, optimistically speaking, our method provides state troopers with 60.0% more coverage than the commercial software does. All five factors have positive impacts on the objective. Especially, the positive relation between the number of TSs and the coverage objective forms the root cause of the necessity to incorporate the choice of TSs in the patrol routes of state troopers.

As for runtime, our metaheuristic takes a couple of seconds, while CPLEX takes an hour to obtain a LB. The solution time is very critical, especially when state troopers need to respond to accidents in a timely manner. Therefore, our solution approach is more favorable.

5.2 Experiment for the Extended Model

Next, we investigate the performance of the revised metaheuristic to solve the extended model with F_i . Keeping all other parameters the same, we test our algorithm with identical $F_i = \{2, 8\}$ \$/TS/period, since each TS is charged the same. If F_i is TS dependent, our algorithm is generic enough to handle as well. Since weights (1, 1.1, 1.2) and (1, 1.5, 2) have very similar results, we only report the results of one weight scheme - (1, 1.1, 1.2) to avoid redundancy. The reported items are the gaps compared with CPLEX, shown in Table 3.

Table 3: Performance gap between the revised metaheuristic and CPLEX for $w_i^t = (1, 1.1, 1.2)$ with different fixed costs.

	$F_i = 2$								$F_i = 8$							
	Avg. (%)				Max. (%)				Avg. (%)				Max. (%)			
	Str1	Str2	Str3	Best	Str1	Str2	Str3	Best	Str1	Str2	Str3	Best	Str1	Str2	Str3	Best
p=0.25																
100%B	0.6	-0.8	-0.2	1.0	23.6	22.3	20.4	23.6	-0.6	-2.0	-1.4	-0.2	27.9	26.5	24.5	27.9
80%B	0.2	-1.2	-0.6	0.6	13.5	14.1	13.2	14.1	0.4	-1.0	-0.4	0.8	15.4	14.2	12.4	15.4
60%B	0.1	-1.3	-0.7	0.5	16.1	14.9	13.0	16.1	-0.2	-1.7	-1.1	0.1	28.1	26.8	24.7	28.1
40%B	0.7	-1.3	-0.2	1.2	21.6	20.3	18.4	21.6	-0.3	-1.9	-0.9	0.5	16.6	17.2	16.3	17.2
20%B	-2.5	-4.4	-3.7	-1.1	34.8	28.7	36.4	36.4	-4.2	-6.4	-4.6	-1.0	71.4	66.0	75.5	75.5
p=0.5																
100%B	0.6	-0.8	-0.2	1.0	23.6	22.3	20.4	23.6	-0.6	-2.0	-1.4	-0.2	27.9	26.5	24.5	27.9
80%B	0.2	-1.2	-0.6	0.6	13.5	14.1	13.2	14.1	0.4	-1.0	-0.4	0.8	15.4	14.2	12.4	15.4
60%B	-0.3	-1.9	-1.0	0.2	16.1	14.9	13.0	16.1	-0.7	-2.3	-1.4	-0.2	28.1	26.8	24.7	28.1
40%B	-1.3	-2.2	-1.3	0.1	21.6	20.3	18.4	21.6	-2.0	-2.6	-1.8	-0.6	16.6	17.2	16.3	17.2
20%B	-5.4	-4.4	-5.1	-2.6	30.6	37.0	34.6	37.0	-3.6	-3.2	-3.1	-0.1	80.1	89.9	76.9	89.9
p=0.75																
100%B	0.0	-1.7	-0.8	0.4	23.6	22.3	20.4	23.6	-1.2	-2.9	-2.0	-0.8	27.9	26.5	24.5	27.9
80%B	-2.0	-2.6	-1.8	-0.8	13.5	14.1	13.2	14.1	-1.8	-2.4	-1.7	-0.6	15.4	14.2	12.4	15.4
60%B	-3.3	-2.9	-2.2	-1.3	13.4	11.8	12.9	13.4	-3.7	-3.3	-2.7	-1.7	25.1	23.4	23.7	25.1
40%B	-4.5	-3.5	-3.7	-2.3	18.5	16.5	18.2	18.5	-4.9	-4.3	-4.4	-2.9	16.8	13.9	14.6	16.8
20%B	-6.0	-6.6	-5.8	-3.0	28.1	27.5	34.9	34.9	-5.1	-5.7	-3.4	-1.6	86.8	80.4	79.5	86.8

When we compare the results for different budget levels, starting strategies, and p values, we get similar results as in the previous subsection. However, if we compare the results with the fixed cost and those without the fixed cost, additional insights can be drawn. When $F_i = 0$, state

troopers are more spread out with respect to where they start and stop; when $F_i > 0$, state troopers tend to share the starting or stopping places in order to save money on paying for the fixed cost of TS. The tighter the budget and the higher the fixed cost of TS, the more obvious this phenomenon is. It can be projected that if the budget is really tight and the fixed cost of TS is high enough, all state troopers will share only one TS each period, which becomes a single depot problem.

5.3 Performance Measures

In addition to comparing with CPLEX, we also benchmark on work by Keskin et al. (2011), since DMD-MCPRP is an extension of MCPRP. Other than the objective, Keskin et al. (2011) also introduce two performance measures to evaluate the proposed coverage plan. They are “Percentage of Hot Spots Covered (HS%)” and “Percentage of Coverage Length (TW%)”. For the sake of completeness, we present the following definitions:

HS%: This performance measure calculates, among all of the hot spots, the percentage covered as a result: $HS\% = \frac{\sum_{t \in \mathcal{T}} \sum_{i \in \mathcal{N}} \sum_{k \in \mathcal{K}} y_{ik}^t}{|\mathcal{T}| \times |\mathcal{N}|}$, where the numerator represents the total number of visited HSs.

TW%: This performance measure calculates the percentage of total available time serviced: $TW\% = \frac{\sum_{t \in \mathcal{T}} \sum_{i \in \mathcal{N}} \sum_{k \in \mathcal{K}} (f_{ik}^t - s_{ik}^t)}{\sum_{t \in \mathcal{T}} \sum_{i \in \mathcal{N}} (l_i^t - e_i^t)}$. In this measure, the numerator is the service time returned, and the denominator is the total time window length.

We compare the objectives, HW%, and TW% of DMD-MCPRP with those of MCPRP. Since MCPRP does not have a budget limit, it is only compared with DMD-MCPRP without the fixed cost when the budget is 100% \mathcal{B} . The best objectives of all p and all starting strategies of DMD-MCPRP are compared with objectives of MCPRP returned by local search. Therefore, there are a total of 32 pairs of comparisons for each weight scheme. We report the results of weight (1, 1.1, 1.2) in Table 4.

To compare objectives, we report the improvements $\frac{\text{Obj of DMD-MCPRP} - \text{Obj of MCPRP}}{\text{Obj of MCPRP}}$, referred to as “Imp” in the last column of the table. The results confirm our intuition that DMD-MCPRP outperforms MCPRP. The worst performance of DMD-MCPRP is a tie with MCPRP in instance 7, in which both DMD-MCPRP and MCPRP cover all of the HSs. In contrast to the best performance, the biggest improvement is as high as 9.7%, found in instance 3. The improvement is attributed to the dynamic selection of a TS, with state troopers starting at a TS closer to a HS than the central depot and stopping at a TS closer to a HS in the next period. In the meantime, we report the average values of all reported items “Avg.” at the bottom of the table. DMD-MCPRP, on average, has 3.1% more time coverage benefits than MCPRP. Even though this percentage may seem low, in real terms, this translates to almost two extra hours of effective coverage. Improvement at such scale helps state troopers increase their patrol effectiveness.

TW% performances of both MCPRPR and DMD-MCPRP are consistent with objectives, as they have the same denominators. On average, DMD-MCPRP returns 88% and MCPRP returns 85%

TW% coverage, thus DMD-MCPRP manages to stay effectively at a HS longer. With respect to HS%, sometimes DMD-MCPRP is better than MCPRP; at other times MCPRP is better. On average, DMD-MCPRP returns 90% and MCPRP returns 92% HS% coverage, thus MCPRP is forced to switch more often from one HS to another due to reaching the latest time window. Interestingly, in some instances TW% is 100%, but HS% is less than 100%, e.g. instances 7 and 8. The reason lies in the fact that there are some HSs whose time window length is 0. On average, both performance measures are higher than 80%, which is quite satisfactory.

Table 4: Comparison of performance measures between MCPRP and DMD-MCPRP.

Inst	MCPRP			DMD-MCPRP			Imp
	Obj	TW%	HS%	Obj	TW%	HS%	
1	1654	86%	88%	1699	89%	91%	2.7%
2	3379	91%	98%	3436	93%	97%	1.7%
3	1959	91%	100%	2148	100%	100%	9.7%
4	3774	94%	95%	3952	98%	98%	4.7%
5	2178	94%	100%	2272	98%	100%	4.3%
6	3581	97%	98%	3606	98%	98%	0.7%
7	1947	100%	97%	1947	100%	97%	0.0%
8	4299	99%	98%	4322	100%	98%	0.5%
9	1708	89%	94%	1751	91%	91%	2.5%
10	3516	82%	88%	3633	84%	91%	3.3%
11	1978	95%	97%	2018	97%	97%	2.0%
12	3843	95%	98%	3934	97%	95%	2.4%
13	2044	96%	100%	2101	99%	100%	2.8%
14	3910	94%	95%	4030	97%	98%	3.1%
15	2301	98%	100%	2349	100%	100%	2.1%
16	3847	98%	100%	3901	99%	100%	1.4%
17	2556	63%	73%	2591	64%	70%	1.4%
18	4628	59%	70%	4783	61%	66%	3.4%
19	2903	71%	88%	2969	72%	81%	2.3%
20	6547	83%	90%	6758	85%	91%	3.2%
21	3278	76%	92%	3366	78%	84%	2.7%
22	7018	81%	92%	7130	82%	91%	1.6%
23	4022	92%	97%	4049	93%	98%	0.7%
24	8336	90%	98%	8498	99%	91%	1.9%
25	2589	57%	61%	2727	60%	56%	5.3%
26	4685	58%	66%	4799	60%	64%	2.4%
27	3469	76%	91%	3645	80%	94%	5.1%
28	6822	76%	92%	7032	78%	86%	3.1%
29	2854	79%	91%	3119	87%	91%	9.3%
30	6445	74%	88%	6654	88%	76%	3.2%
31	3530	86%	98%	3721	90%	100%	5.4%
32	7393	90%	98%	7608	92%	99%	2.9%
Avg	3843	85%	92%	3955	88%	90%	3.1%

Section 6

Conclusions

In conclusion, to improve the efficiency of state trooper patrols, we allow for dynamically changing patrol routes and starting and stopping locations. For this purpose, we develop a new dynamic, multi-depot, location-routing model, extending MCPRP in the literature. Gaining insights from solutions of LRP, we decompose this problem into multi-depot MCPRP and facility location, and then solve them in an iterative way with custom built heuristics. We test the model and solution approach for the situations without and with a fixed cost of TS, and compare with the LB of CPLEX. We also compare the time and HS coverage performances of this model with the single depot MCPRP, and significant improvements are found in the objectives.

There are several possible extensions for future research. One possible extension is to include dynamic travel times with real time traffic conditions. Another one, in addition to covering predetermined HSs, is to consider on-call responses of state troopers.

Section 7

References

- Albareda-Sambola, M., E. Fernandez, G. Laporte. 2007. Heuristic and lower bound for a stochastic location-routing problem. *European Journal of Operational Research* **179**(3) 940–955.
- Alçada-Almeida, L., L. Tralhão, L. Santos, J. Coutinho-Rodrigues. 2009. A multiobjective approach to locate emergency shelters and identify evacuation routes in urban areas. *Geographical Analysis* **41**(1) 9–29.
- Alumur, S., B. Y. Kara. 2007. A new model for the hazardous waste location-routing problem. *Computers and Operations Research* **34**(5) 1406–1423.
- Balakrishnan, A., J. E. Ward, R. T. Wong. 1987. Integrated facility location and vehicle routing models: Recent work and future prospects. *American Journal of Mathematical and Management Sciences* **7**(1) 35–61.
- Bérubé, J.F., M. Gendreau, J.Y. Potvin. 2009. An exact-constraint method for bi-objective combinatorial optimization problems: Application to the traveling salesman problem with profits. *European Journal of Operational Research* **194**(1) 39–50.
- Blincoe, L., A. Seay, E. Zaloshnja, T. Miller, E. Romano, S. Luchter, R. Spicer. 2002. The economic impact of motor vehicle crashes, 2000. Crash Statistics DOT HS 809 446, NHTSA.
- Boussier, S., D. Feillet, M. Gendreau. 2007. An exact algorithm for team orienteering problems. *4OR: A Quarterly Journal of Operations Research* **5**(3) 211–230.
- Caballero, R., M. González, F. M. Guerrero, J. Molina, C. Paralera. 2007. Solving a multiobjective location routing problem with a metaheuristic based on tabu search. Application to a real case in Andalusia. *European Journal of Operational Research* **177**(3) 1751–1763.
- Chankong, V., Y.Y. Haimes. 1983. *Multiobjective decision making: theory and methodology*, vol. 8. North-Holland.
- Hansen, P. H., B. Hegedahl, S. Hjortkjaer, B. Obel. 1994. A heuristic solution to the warehouse location-routing problem. *European Journal of Operational Research* **76**(1) 111–127.
- Keskin, B. B., S. R. Li, D. Steil, S. Spiller. 2011. Analysis of an integrated maximum covering and patrol routing problem. *Transportation Research Part E: Logistics and Transportation Review* **48**(1) 215–232.
- Kirkpatrick, S., C. D. Gelatt, Jr., M.P. Vecchi. 1983. Optimization by simulated annealing. *Science* **220** 671–680.
- Labbé, M., I. Rodríguez-Martin, J. J. Salazar-González. 2004. A branch-and-cut algorithm for the plant-cycle location problem. *Journal of the Operational Research Society* **55**(5) 513–520.

- Laporte, G. 1988. Location-routing problems. B. L. Golden, A. A. Assad, eds., *Vehicle Routing: Methods and Studies*. Noth-Holland, Amsterdam, 163–198.
- Laporte, G., Y. Nobert, D. Arpin. 1986. An exact algorithm for solving a capacitated location-routing problem. *Annals of Operations Research* **6**(9) 291–310.
- Laumanns, M., L. Thiele, E. Zitzler. 2005. An adaptive scheme to generate the pareto front based on the ϵ -constraint method. *Practical Approaches to Multi-Objective Optimization* .
- Laumanns, M., L. Thiele, E. Zitzler. 2006. An efficient, adaptive parameter variation scheme for metaheuristics based on the ϵ -constraint method. *European Journal of Operational Research* **169**(3) 932–942.
- Lou, Y., Y. Yin, S. Lawphongpanich. 2011. Freeway service patrol deployment planning for incident management and congestion mitigation. *Transportation Research Part C: Emerging Technologies* **19**(2) 283–295.
- Mavrotas, G. 2009. Effective implementation of the ϵ -constraint method in multi-objective mathematical programming problems. *Applied Mathematics and Computation* **213**(2) 455–465.
- Melechovský, J., C. Prins, R. W. Calvo. 2005. A metaheuristic to solve a location-routing problem with non-linear costs. *Journal of Heuristics* **11**(5) 375–391.
- Miettinen, K. 1999. *Nonlinear multiobjective optimization*, vol. 12. Springer.
- Min, H., V. Jayaraman, R. Srivastava. 1998. Combined location-routing problems: A synthesis and future research directions. *European Journal of Operational Research* **108**(1) 1–15.
- Montemanni, R., L. M. Gambardella. 2009. Ant colony system for team orienteering problems with time windows. *Foundations of Computing and Decision Sciences* **34**(4) 287–306.
- Nagy, G., S. Salhi. 1996. Nested heuristic methods for the location-routeing problem. *Journal of the Operational Research Society* **47**(9) 1166–1174.
- Nagy, G., S. Salhi. 2007. Location-routing: Issues, models and methods. *European Journal of Operational Research* **177**(2) 649–672.
- Perl, J., M. S. Daskin. 1985. A warehouse location-routing problem. *Transportation Research Part B: Methodological* **19**(5) 381–396.
- Sait, S. M., H. Youssef. 1999. *Iterative Computer Algorithms with Applications in Engineering: Solving Combinatorial Optimization Problems*. IEEE Computer Society, Los Alamitos, CA.
- Salhi, S., G. K. Rand. 1989. The effect of ignoring routes when locating depots. *European Journal of Operational Research* **39**(2) 150–156.
- Semet, F., E. Taillard. 1993. Solving real-life vehicle routing problems efficiently using tabu search. *Annals of Operations Research* **41**(4) 469–488.

- Steil, D., A. Parrish. 2009. Hit: A gis-based hotspot identification taxonomy. *International Journal of Computers and their Applications* **16**(2) 81–90.
- Tricoire, F., M. Romauch, K. F. Doerner, R. F. Hartl. 2010. Heuristics for the multi-period orienteering problem with multiple time windows. *Computers and Operations Research* **37**(2) 351–367.
- Tsiligirides, T. 1984. Heuristic methods applied to orienteering. *Journal of the Operational Research Society* **35**(9) 797–809.
- Vansteenwegen, P., W. Souffriau, G. Vanden Berghe, D. Van Oudheusden. 2009. Iterated local search for the team orienteering problem with time windows. *Computers and Operations Research* **36**(12) 3281–3290.
- Willemse, E. J., J. W. Joubert. 2012. Applying minmax k postmen problems to the routing of security guards. *Journal of the Operational Research Society* **63**(2) 245–260.
- Wu, T. H., C. Low, J. W. Bai. 2002. Heuristic solutions to multi-depot location-routing problems. *Computers and Operations Research* **29**(10) 1393–1415.