The cooperative orienteering problem with time windows

Martijn van der Merwe[a,*], James P. Minas[a], Melih Ozlen[a], John W. Hearne[a]

[a]*School of Mathematical and Geospatial Sciences, RMIT University, GPO Box 2476, Melbourne, Victoria, 3001, Australia*

Abstract

In this we paper we define a new class of the team orienteering problem; the cooperative orienteering problem with time windows (COPTW). The COPTW is a generalisation of the TOPTW, which requires multiple vehicles to cooperatively collect the reward from a location. The COPTW is demonstrated with the aid of a wildfire scenario in South Hobart, Tasmania Australia. Computational testing suggests that it is feasible to apply the COPTW to realistic problems.

Keywords: Team orienteering problem; Time windows; Integer Programming; Wildfire

1. Introduction

In the *team orienteering problem with time windows* (TOPTW) a number of locations are given. Each location has an associated time window, a service duration and a reward. A fleet of vehicles are available to collect these rewards. To collect the reward from a location a vehicle has to visit the location, staying for the duration of the service time. The service time may only commence within the location's associated time window and it is not required that all locations are serviced. The aim is to maximise the total reward collected. Often an overall time limit is associated with each vehicle route.

The TOPTW may be used to model a variety of decision problems including tourist trip planning and sport talent scouting [1]. Another application where the TOPTW presents itself is in managing the allocation of resources during wildfires. The response efforts to wildfires are coordinated by *incident management teams* (IMT). Often during large wildfires these teams have to make optimal use of limited resources. Large wildfires may result in the loss of or damage to houses, schools,

*Corresponding author. Phone number: +61 3 9925 3767

Email addresses: martijn.vandermerwe@rmit.edu.au (Martijn van der Merwe), james.minas@rmit.edu.au (James P. Minas), melih.ozlen@rmit.edu.au (Melih Ozlen), john.hearne@rmit.edu.au (John W. Hearne)

bridges, factories and hospitals, often referred to as community assets. In some cases the loss of assets may be prevented if sufficient resources are dispatched to the assets in a timely manner. The protection requirement of an asset is the amount of resources required to provide an adequate level of protection to defend an asset. The typical resource units being assigned are fire trucks, commonly referred to in wildland fire fighting parlance as tankers.

The advancing fire front imposes a time window during which any task to protect an asset must be carried out. The time windows are determined by the anticipated *time to impact*, this is the time remaining before the asset is impacted by the fire. The time to impact may be determined using fire spread modelling. Extensive research has been carried out in the modelling and prediction of fire spread, which is summarised in a series of reviews undertaken by Sullivan [2, 3, 4]. In the case where all the assets have identical protection requirements, the problem described is the TOPTW. However, in the wildfire asset protection problem, multiple vehicles may be required to cooperatively protect a single asset. This differs from the TOPTW where only a single team member is needed to claim a reward from a location. The problem of protecting assets during wildfires thus inspires the formulation of a new model, the *cooperative orienteering problem with time windows* (COPTW) to take the varying protection requirements of locations into account.

Optimisation models are often employed in emergency management and wildfire response. The development of decision support tools to assist with the management of wildfires is an active field [5, 6, 7].However, to our knowledge the problem of protecting assets during a wildfire has not previously been modelled.

Kantor and Rosenwein [8] were the first to solve the (single vehicle) orienteering problem with time windows, while the TOPTW was first studied by Vansteenwegen et al. [9] and Montemanni and Gambardella [10]. The TOPTW is in the class of vehicle routing problems with profits and is closely related to the selective vehicle routing problem with time windows (SVRPTW) Gueguen and Dejax [11]. The SVRPTW generalises the TOPTW by adding a capacity constraint to each vehicle [12]. Vehicle routing problems with profits have been reviewed by Archetti et al. [13] and orienteering problems by Vansteenwegen et al. [1].

The split delivery capacitated team orienteering problem allows multiple vehicles to service a single location [14], but differs from the COPTW in the sense that it does not require the service times at a single location to coincide.

Recently there has been a marked interest in heuristics methods for the TOPTW, see for example the recent paper by Hu and Lim [15]. An exact algorithm for team orienteering problems was proposed by Boussier et al. [16], demonstrating its implementation for the team orienteering problem and the SVPRTW. Gueguen and Dejax [11] and Butt and Ryan [17] proposed exacts algorithms for the SVRPTW, both these approaches are neatly summarised in [12]. The column generation step in all of these exact methods require the calculation of the reward contributed to the objective function value by each vehicle route. This is straightforward in the TOPTW. In the COPTW, however, vehicles must collect rewards cooperatively, it is therefore not possible to calculate the contribution of a single vehicle to the total objective value independently. These branch and price methods cannot be applied directly to solving the COPTW.

In this paper we present the COPTW which generalises the TOPTW by including a resource requirement for each location. The resource requirement of a location specifies the number of vehicles required to provide service to that location. All the required vehicles have to arrive at the location before service may commence. We demonstrate the COPTW by applying it to a wildfire asset protection scenario in South Hobart, Tasmania Australia.

The remainder of the paper is structured as follows. A mixed integer programming formulation for the COPTW is presented in §2. Computational testing of the COPTW is carried out in §3. The COPTW's functionality is demonstrated in §4. The paper concludes in §5 with a discussion of the results, application of the COPTW and possible extension and variations on the COPTW.

2. Mathematical programming formulation

Next we introduce a two-index vehicle flow formulation for the COPTW.

Consider the connected weighted graph G of order N with vertex set V and arc set E. Each arc $e_{ij} \in E$ has an associated weight $w(e_{ij}) = t_{ij}$. The arc weight is analogous to the travelling time (or cost) between vertices. Each vertex has an associated reward S_i and resource requirement r_i. Each vehicle starts at a depot located at vertex v_1 and ends at a depot located at vertex v_N. There is no value associated with the initial and final vertices, therefore $S_1 = 0$ and $S_N = 0$. Let a_i be the service duration associated with vertex v_i. Each vertex v_i has an associated time window, the time window's opening time is O_i and its closing time is C_i. The vehicle fleet consists of P identical vehicles.

The following decision variables are defined. The decision variable $y_i = 1$ if vertex v_i is serviced, otherwise $y_i = 0$. The non-negative integer decision variable x_{ij} is the number of vehicles travelling from vertex v_i to vertex v_j. The binary variable $z_{ij} = 1$ if $x_{ij} > 0$, otherwise $z_{ij} = 0$.

A vertex is considered serviced if r_i vehicles visit the vertex, all arriving at or before the start of service, s_i. The vehicles then cooperatively provide the service for a duration of a_i.

The next step is to eliminate those arcs that are infeasible due to the time window constraints. Consider two vertices v_i and v_j, chosen such that the earliest possible departure from vertex v_i results in an arrival at vertex v_j which is later than the closing time of vertex v_j. Since no feasible solution will contain the arc e_{ij} it is possible to ignore this arc. Let $\mathcal{E}$ be the index set excluding the infeasible arcs, that is $(i, j) \in \mathcal{E}$ if and only if $e_{ij} \in E$ and $O_i + a_i + t_{ij} \leq C_j$.

Two sets $\Delta^-(i)$ and $\Delta^+(i)$ are defined to simplify the model notation: $\Delta^-(i)$ is the index set of vertices adjacent *to* vertex v_i, that is $j \in \Delta^-(i)$ if $(j, i) \in \mathcal{E}$, and $\Delta^+(i)$ is the index set of vertices adjacent *from* vertex v_i, that is $j \in \Delta^+(i)$ if $(i, j) \in \mathcal{E}$.

Based on the notation introduced above, the COPTW may be formulated as a mixed integer program:

$$\text{Maximise} \sum_{i=2}^{N-1} S_i y_i \tag{1}$$

subject to

$$\sum_{j \in \Delta^+(1)} x_{1j} = \sum_{i \in \Delta^-(N)} x_{iN} = P; \tag{2}$$

$$\sum_{i \in \Delta^-(k)} x_{ik} = \sum_{j \in \Delta^+(k)} x_{kj} \quad \forall\ k = 2, \dots, N-1; \tag{3}$$

$$r_k y_k = \sum_{j \in \Delta^+(k)} x_{kj} \quad \forall\ k = 2, \dots, N-1; \tag{4}$$

$$x_{ij} \leq P z_{ij} \quad \forall\ (i, j) \in \mathcal{E}; \tag{5}$$

$$s_i + t_{ij} + a_i - s_j \leq M(1 - z_{ij}) \quad \forall\ (i, j) \in \mathcal{E}; \tag{6}$$

$$O_i \leq s_i \quad \forall\ i = 1, \dots, N; \tag{7}$$

$$s_i \leq C_i \quad \forall\ i = 1, \dots, N; \tag{8}$$

$$x_{ij} \in \{0, 1, 2, \dots, P\} \quad \forall\ (i, j) \in \mathcal{E}; \tag{9}$$

$$y_i, z_{ij} \in \{0, 1\} \quad \forall\ (i, j) \in \mathcal{E}. \tag{10}$$

The objective function (1) is to maximise the total collected reward. Constraint (2) ensures that each vehicle starts at vertex v_1 and ends at the vertex v_N. The vehicle flow to and from each vertex is balanced by constraints (3). Constraints (4) enforce the condition that assets located at a vertex are protected if and only if a sufficient number of vehicles visit the vertex during the time window. Constraints (5) and (6) ensure that service at a vertex may only start after service at a previously visited vertex has been completed and sufficient time for travel has been allowed, with M representing a large constant. The assignment $M = \max(C_i) + \max(t_{ij}) + \max(a_i) - \min(O_i)$ is a sufficiently large value for M. The start of service times at vertices are limited to their respective time windows by constraints (7) and (8). The integer and binary conditions are enforced on the decision variables by constraints (9) and (10) respectively.

3. Computational study

In this section the solution time of the COPTW is demonstrated. The aim of the testing was to obtain an indication of the size of problems that can be solved within a reasonable time. Computational testing was done on randomly generated problem instances. These instances were generated using the existing TOPTW benchmarks by adding a column for the resource requirements. The resource requirement values were generated by drawing randomly from the set {1,2,3}. The original TOPTW benchmark instances are described by Vansteenwegen et al. [1].

Computational testing was carried out on a single node of a computer cluster. The node had two Intel Xeon E5-2670 processors and 64GB of RAM. CPLEX 12.6 was used to solve the problem instances and performance was measured in CPU time. The solver's parallel optimisation mode was set to deterministic while all the remaining CPLEX solver parameters were left at their default values.

The results of the computational testing are summarised in Table 1. Each instance is truncated to the size of inidicated in the table.

The results indicate that an optimal solution may be found for the majority of problems with 20 locations or fewer. In wildfire asset protection the time windows are determined by the anticipated time to impact, therefore the time windows are correlated to their spatial positions. New problem instances were generated to investigate the effect of this correlation on computational performance. These new problem instances were generated with uniform spatially distributed vertices. Ten problem instances with a 100 vertices each were generated. The vertices of each instance are uniformly distributed in a 80km by 80km square region. The travel time is directly proportional to the distance between vertices. The opening time of each time window is correlated to the x-

Set	Size		Solution time $P = 3$		$P = 4$		$P = 5$
c100	15		1		0		0
	20	**(3)**	3 615		0		1
	25	**(5)**	6 006	**(5)**	6 005	**(1)**	1 223
c200	20		1		0		1
	25		10		2		2
	30		48		14		3
pr1-10	15		5		6		4
	10		119		6		116
	20	**(7)**	8 269	**(8)**	9 882	**(4)**	7 119
pr11-20	10		2		0		0
	15	**(7)**	7 849	**(4)**	4402		5
	20	**(10)**	10 803	**(4)**	4 401	**(1)**	1 390
r100	10		2		0		0
	15	**(2)**	3 292	**(2)**	3 088	**(3)**	2 850
	20	**(7)**	7 131	**(2)**	3 089	**(7)**	6 657
r200	20		2		1		2
	25		10		2		2
	30		3 231		38		12
rc100	10		0		0		0
	15	**(2)**	3 027	**(2)**	3 427	**(1)**	2 608
	20	**(4)**	7 977	**(2)**	3 432	**(5)**	6 967
rc200	20		9		1		1
	25		206		25		4
	30	**(3)**	8 151		1 804		49

Table 1: The average solution time of each benchmark set in seconds. The number of problems which could not be solved within the 10 800s time limit is shown in boldface in brackets. The problems were truncated to the indicated sizes.

coordinate of it's vertex. This was done to capture the spatially correlated property of time windows in wildfire scenarios. A parameter w is used to determine the length of the time window of each vertex ($C_i = O_i + w$). The smaller problem instances (25, 50 and 75 vertices) are subsets of the ten 100-vertex instances that were generated.

The solution times of these problems are summarised in Table 2. The solution times were limited to three hours (10 800 seconds). Small problems, those with 25 vertices, are solved within a couple of seconds, while the solution times of the larger problems are highly dependent on the problems' parameters. In the instances considered in Table 2, increasing either the length of the time windows, or the number of vehicles, resulted in increased solution times. In some cases it may be possible to easily solve problems with a 100 locations.

	Solution time			
	$P = 3$	$P = 6$	$P = 3$	$P = 6$
Size	$w = 20$	$w = 20$	$w = 40$	$w = 40$
25	1	1	2	2
50	5	24	92	1 828
75	29	122	**5 382**	-
100	58	**2 216**	-	-

Table 2: The solution times in seconds for a number of test instances with each N uniformly distributed vertices. Here P is the number of vehicles and w is the length of the time windows. A dash indicates that none of the problems could be solved within the three hour time limit and bold face entries are sets in which solutions where found for all problems except one.

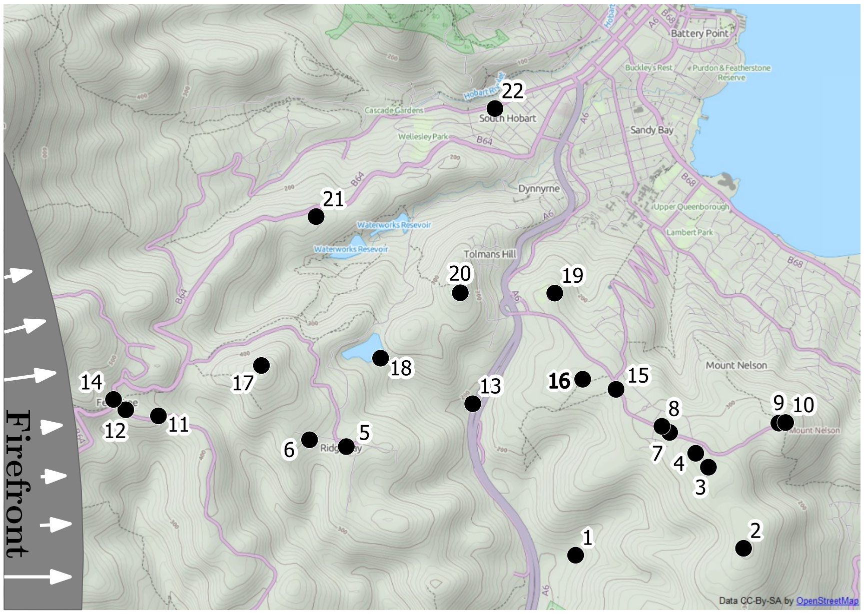
4. Case study - Wildfire asset protection in Tasmania

The working of the COPTW is demonstrated in this section by aid of a hypothetical wildfire scenario. The Tasmanian Fire Service's community protection plans identify various community assets such as communication towers, hotels, and historical significant buildings. A number of these community assets located in South Hobart are shown in Figure 1. For the scenarios presented here, a simple fire spread radiating outwards at rate of 3km/h in a circular fashion from a single point of origin is considered. The fire front is indicated as the dark grey shaded area on the map in Figure 1. Further, it is assumed that each asset requires 30 minutes of protection at the time of impact. Travel times between assets were calculated using Google Maps' Distance Matrix service. Random values were generated for the protection requirement and value of each asset.

An optimal solution utilising five vehicles is shown in Figure 2(a). Note the convergence of paths at vertices requiring multiple vehicles. The locations are visited from left to right as the fire spreads across the landscape. In this scenario not all assets can be saved with only five vehicles. Twenty vehicles would be required to provide adequate protection to all of the assets. The roads used by the vehicles are highlighted in Figure 2(b).

5. Conclusion

The computational results demonstrate that it would be possible to apply the model to realistic sized problems in wildfire asset protection. Problems may arise, however, which are difficult to solve, due to either the underlying properties of the problem, or simply due to a large number of locations. In these cases the development of fast heuristics approaches or efficient exact methods may be very useful.



i	S_i	r_i	$O_i = C_i$
1	10	2	91
2	10	3	126
3	30	1	119
4	30	2	117
5	20	3	45
6	10	3	38
7	30	2	112
8	10	3	110
9	20	2	134
10	10	1	136
11	30	1	8
12	30	1	2
13	10	3	72
14	10	2	0
15	10	2	102
16	10	1	95
17	20	1	31
18	20	2	55
19	10	2	93
20	20	1	75
21	20	3	51
22	30	2	94

Figure 1: Community assets located in South Hobart. The table contains the parameters associated with each location.

Futurework may consider the case where vehicles have different capabilities and each location requires a mix of vehicle types to provide cooperative service. Taking different vehicle types and their respective capabilities into account would require significant modifications to the COPTW.

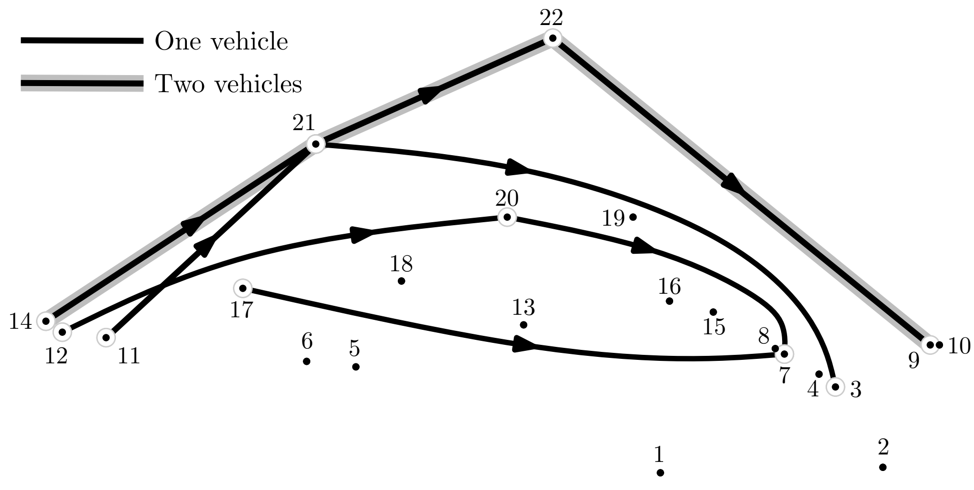
test

Acknowledgements

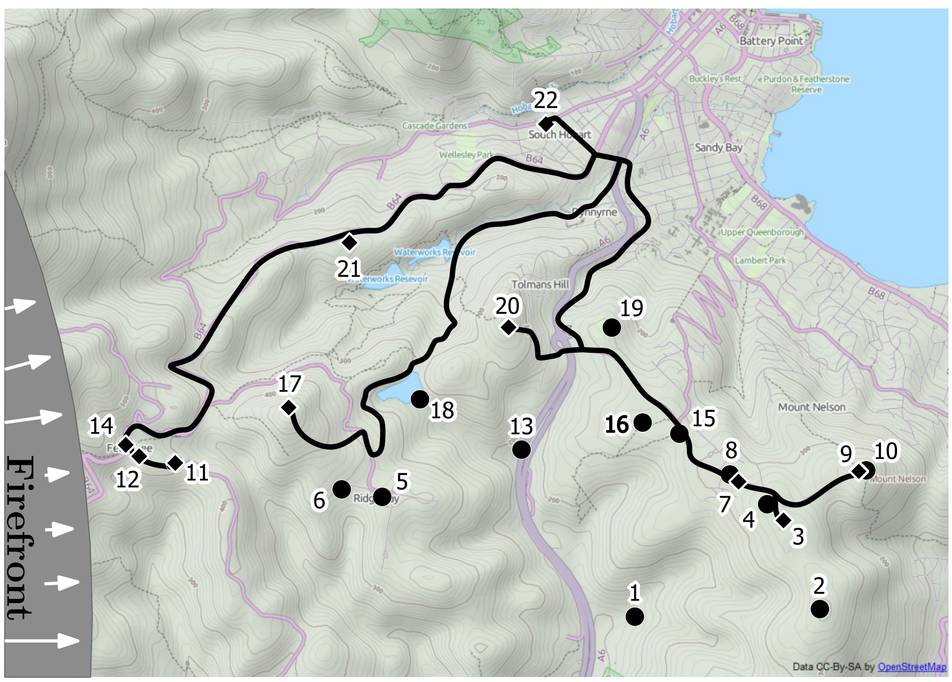
We would like to thank Damien Killalea from the Tasmania Fire Service for providing valuable information on wildfire incident management. This work was supported by the Bushfire CRC in the form of scholarship funding to Martijn van der Merwe. Computational work was done on Trifid, a VPAC high performance computer cluster. The third author is supported by the Australian Research Council under the Discovery Projects funding scheme (project DP140104246).

[1] P. Vansteenwegen, W. Souffriau, D. Van Oudheusden, The orienteering problem: A survey, European Journal of Operational Research 209 (1) (2011) 1–10.

[2] A. L. Sullivan, Wildland surface fire spread modelling, 1990–2007. 3: Simulation and mathematical analogue models, International Journal of Wildland Fire 18 (4) (2009) 387–403.



(a)



(b)

Figure 2: Applying the COPTW to a wildfire asset protection scenario. The problem parameters are contained in the table in Figure 1. An optimal solution is shown for five vehicles, a total reward of 240 is collected by servicing locations 3,7,9,11, 12, 14,17,20, 21 and 22. (a) The solution in graph representation. (b) The roads utilised by vehicles in this solution.

[3] A. L. Sullivan, Wildland surface fire spread modelling, 19902007. 1: Physical and quasi-physical models, International Journal of Wildland Fire 18 (4) (2009) 349–368.

[4] A. L. Sullivan, Wildland surface fire spread modelling, 19902007. 2: Empirical and quasi-empirical models, International Journal of Wildland Fire 18 (4) (2009) 369–386.

[5] D. L. Martell, A review of operational research studies in forest fire management, Canadian Journal of Forest Research 12 (2) (1982) 119–140.

[6] D. L. Martell, E. A. Gunn, A. Weintraub, Forest management challenges for operational researchers, European Journal of Operational Research 104 (1) (1998) 1–17.

[7] J. P. Minas, J. W. Hearne, J. W. Handmer, A review of operations research methods applicable to wildfire management, International Journal of Wildland Fire 21 (3) (2012) 189–196.

[8] M. G. Kantor, M. B. Rosenwein, The Orienteering Problem with Time Windows, The Journal of the Operational Research Society 43 (6) (1992) 629–635.

[9] P. Vansteenwegen, W. Souffriau, G. Vanden Berghe, D. Van Oudheusden, Iterated local search for the team orienteering problem with time windows, Computers & Operations Research 36 (12) (2009) 3281–3290.

[10] R. Montemanni, L. M. Gambardella, An ant colony system for team orienteering problems with time windows, Foundations of Computing and Decision Sciences 34 (4) (2009) 287–306.

[11] C. Gueguen, P. Dejax, Méthodes de résolution exacte pour les problèmes de tournées de véhicules (1999).
URL `http://cat.inist.fr/?aModele=afficheN&cpsidt=203211`

[12] D. Feillet, P. Dejax, M. Gendreau, Traveling Salesman Problems with Profits, Transportation Science 39 (2) (2005) 188–205.

[13] C. Archetti, M. Speranza, D. Vigo, Vehicle Routing Problems with Profits, *Working Papers WPDEM 2013/3*, Department of Economics and Management, University of Brescia (2013).

[14] C. Archetti, N. Bianchessi, M. Speranza, A. Hertz, The split delivery capacitated team orienteering problem, Networks 63 (1) (2014) 16–33.
URL `http://doi.wiley.com/10.1002/net.21519`

[15] Q. Hu, A. Lim, An iterative three-component heuristic for the team orienteering problem with time windows, European Journal of Operational Research 232 (2) (2014) 276–286.

[16] S. Boussier, D. Feillet, M. Gendreau, An exact algorithm for team orienteering problems, 4OR 5 (3) (2006) 211–230.

[17] S. E. Butt, D. M. Ryan, An optimal solution procedure for the multiple tour maximum collection problem using column generation, Computers & Operations Research 26 (4) (1999) 427–441.