22nd National Conference of the Australian Society for Operations Research, Adelaide, Australia, 1–6 December 2013
www.asor.org.au/conferences/asor2013

Pickup and delivery with a solar-recharged vehicle

A. Albrecht[a], **P. Pudney** [b]

[a] *Centre for Industrial and Applied Mathematics, University of South Australia, Mawson Lakes, South Australia, 5095*
[b] *Barbara Hardy Institute, University of South Australia, Mawson Lakes, South Australia, 5095*
Email: Amie.Albrecht@unisa.edu.au

Abstract: Zimbabwe has one of the highest maternal death rates in the world. A key contributing factor is the difficulty that expectant mothers have in getting from rural villages to health care facilities. Specifically-designed electric vehicles, called African Solar Taxis, are currently being developed for deployment. The vehicles will be charged at solar charging stations located at health care facilities. The limited speed and range means that these vehicles must be efficiently scheduled so that expectant mothers are transported to health facilities in a timely manner.

We describe two methods for determining good single-vehicle schedules to transport expectant mothers to health facilities. We consider two objectives: maximising the total requested trip distance completed during the day, and minimising the schedule span. One of these methods is both simple and effective—it requires the scheduler to select from a list of precomputed trip patterns each day.

Keywords: *Pickup and delivery, electric vehicles.*

1 INTRODUCTION

Rural Zimbabwe has one of the highest maternal death rates in the world. A key contributing factor is the lack of transport; women are much more likely to die giving birth at home than if they can get to a hospital.

We have been approached by a non-government organisation in Zimbabwe to design a taxi service for transporting women to hospital to give birth. However, the taxis must be powered from solar energy, since conventional vehicle fuels are not available. The taxis will be low-mass electric vehicles designed to minimise energy use. They will carry a driver and two passengers—the pregnant woman and her companion. The taxis will be recharged from solar-powered recharging stations based at key locations. The range of the taxis will be limited, since increased range requires a greater battery capacity, which increases the mass and reduces the efficiency of the taxi. We will therefore need to plan schedules that include recharging stops.

The initial stage of the Solar Taxi project will be based at the St Albert's Mission Hospital, near the northern border of Zimbabwe. During the first stage of the project, taxis will collect women from four clinics to the south of the hospital (see Figure 1). The distances between the locations are:

Always – David Nelson	19 km
Chidkamwedzi – David Nelson	32 km
David Nelson – Chiyani	15 km
Chiyani – St Albert's Hospital	27 km

Figure 1. Map of the region to be serviced by the taxis. The hospital is at the northern edge of the region.

There will be a charging station at St Albert's Hospital. A second charging station will be located at David Nelson, to reduce the maximum distance between charges from 148 km (St Albert's – Chidkamwedzi – St Albert's) to 64 km (David Nelson – Chidkamwedzi – David Nelson).

Eventually we will have several taxis operating in the area, with some based at the hospital and some at David Nelson. For now, however, we will consider the problem of scheduling a single taxi.

2 Single taxi problem formulation

Suppose we have a single taxi, based at the hospital. At the beginning of each day we will have a list of pending trip requests. Trip requests could include trips from clinics to the hospital and trips from the hospital to clinics.

Possible objectives are:

- maximise the total requested trip distance completed during the day
- minimise the time required to complete all requested trips.

At the end of each day we will transfer any requested trips not completed to the list of pending trips for the next day, along with any new trip requests.

The first objective—completing as much trip distance as possible during the day—will leave the system in a good state for the next day, but will not necessarily be as effective as the second objective over many days. However, if we generate a solution that minimises the time required to complete all journeys, the schedules we use on the first day may not be optimal for updated problems that include new trip requests.

Another possible objective is to maximise the minimum energy content of the battery, which will extend the life of the battery. If we were to include this objective then we would favour solutions that had similar distances travelled each day rather than travelling as far as possible on the first day. But transporting passengers is more important, so we will ignore the battery objective; if we have slack time in our solution for a day, we can adjust the charging durations to maximise the minimum energy content for the given daily schedule.

The vehicle will not have enough battery capacity to be able to complete a round trip between St Albert's Hospital and Always or Chidkamwedzi; the vehicle will have to stop at David Nelson to take on enough energy to ensure that it can reach at least the next charging station. The time required to charge will be proportional to the recharge energy.

How much energy the vehicle takes at each charging station will affect the minimum battery level, but also affect how much energy is taken from each charging station. Ultimately we will have constraints on how much energy each charging station can deliver in a day. We can decide the capacities of the charging stations depending on how the vehicles recharge, or design the schedules to fit the available charging capacity. For now, we will assume that the daily charging capacity of the charging stations is not a constraint.

The time required to travel between any pair of locations is given, and is independent of whether the taxi has passengers. The energy required to travel between any pair of locations is also given, but will be greater for a trip with passengers than for travel without passengers.

The taxi must start and finish each day at the hospital. The span of a daily schedule must not exceed, say, 10 hours. The taxi will start each day with a full battery; the final level of the battery is not important, since the taxi can completely recharge overnight.

Let the set of locations be $\mathcal{L} = \{1, \ldots, L\}$. The problem parameters are:

E_{ij}	the energy required to travel from location $i \in \mathcal{L}$ to location $j \in \mathcal{L}$, with passengers
E'_{ij}	the energy required to travel from location $i \in \mathcal{L}$ to location $j \in \mathcal{L}$, without passengers
t_{ij}	the time required to travel from location $i \in \mathcal{L}$ to location $j \in \mathcal{L}$
D	the maximum daily duration.

The set of requested trips is $\mathcal{R} = \{r_k\}, k \in \{1, \ldots, R\}$, where $r_k = (o_k, d_k)$ and where $o_k \in \mathcal{L}$ and $d_k \in \mathcal{L}$ are the origin and destination locations of trip k.

Vehicle routing problems have been well-studied (see, for example, Laporte (2009); Parragh et al. (2008a,b)). The paired pickup and delivery problem is a vehicle routing problem in which each transportation request specifies a pair of pickup and delivery locations. A fleet of one or more vehicles is available to satisfy the requests.

It is common to formulate a vehicle routing problem as a network flow model in which the vertices of the graph represent customers and other locations to be visited, such as a depot. Arcs between vertices have an associated distance and travel time. A corresponding mathematical linear integer programming model can be formulated in which the decision variables identify whether a particular transportation request is assigned to a particular vehicle, the order in which each vehicle visits locations to which it is assigned, and the arrival and departure times of each vehicle at each location it visits. Pickup and delivery problems are NP-hard as they generalise the well-known Travelling Salesman Problem.

The solar taxi scheduling problem is a paired pickup and delivery model, but with additional constraints on the range of the vehicle. Therefore, we must keep track of the battery level of the vehicle and schedule visits to charging stations as required. For obvious reasons, researchers have only recently begun to incorporate energy considerations into vehicle routing problems. We have found four related studies in which the possibility of vehicle recharging has been considered: Conrad and Figliozzi (2011); Erdogan and Miller-Hooks (2012); Schneider et al. (2012); Wang and Cheu (2013). Earlier studies (eg. Laporte et al. (1985)) constrained the distance travelled by vehicles, but did not allow the opportunity for this to be extended en route.

In our problem the order of the trips within each day is not important—we can tell each passenger what time they will be collected. This allows us to consider simpler formulations of the problem.

3 Simulating trips

For any given sequence of trip requests, we can use discrete event simulation to calculate a sequence of (time, location, battery energy) trip points to form a schedule. Each trip request generates an unloaded trip to the pickup location, if the vehicle is not already there, followed by a loaded trip to the delivery location. We calculate the time and battery energy at each node of the network. If the vehicle arrives at a charging location and the battery energy $E < 0$, we look back through the partial schedule to the previous charging location and add a charging session with energy E and duration E/p, where p is the charging power, at that location. We then adjust the times and energy values for the remainder of the path before continuing with the remaining trip requests.

Once we know which trips will be done on a day, we can increase charging times during the day to avoid taking the battery energy lower than necessary.

4 Daily trip patterns

Suppose we consider only requests for trips from clinics to the hospital. Figure 2 shows the eighteen possible trip patterns that can be completed within a 10-hour day. The horizontal axis is time, in hours. The vertical axis represents location (red curve) with the hospital at the bottom, or battery energy (blue curve). Battery energy decreases when the taxi is moving, and increases during stationary charging sessions at the St Albert's Hospital and at David Nelson.

Because the order of the trips within each day is not important, we can represent a pattern by a vector indicating the number of trips from each clinic. For example, the first pattern in Figure 2 (top left) is represented by the vector $[3, 1, 0, 0]$, indicating that the pattern does three trips from Chiyani, one trip from David Nelson, no trips from Always and no trips from Chidkamwedzi.

The first eight of these patterns are maximal, in that it is not possible to fit another trip into the day. The remaining ten patterns are subsets of the first eight.

5 Maximising the daily trip distance

If our objective is to maximise the useful distance covered during the current day, we can simply look down the list of daily patterns, in decreasing order of distance covered, and pick the first pattern where the pattern does not do any unnecessary trips.

This method is simple enough that it could be done manually. For example:

- if we have two women waiting at each of the clinics then we would skip the first pattern, which does not cover the demand and has a superfluous trip to Chiyani, and choose the second pattern, which collects two women from Always

- if we have one woman waiting at each clinic then we would choose the third pattern, which collects one woman from David Nelson and one woman from Chidkamwedzi.

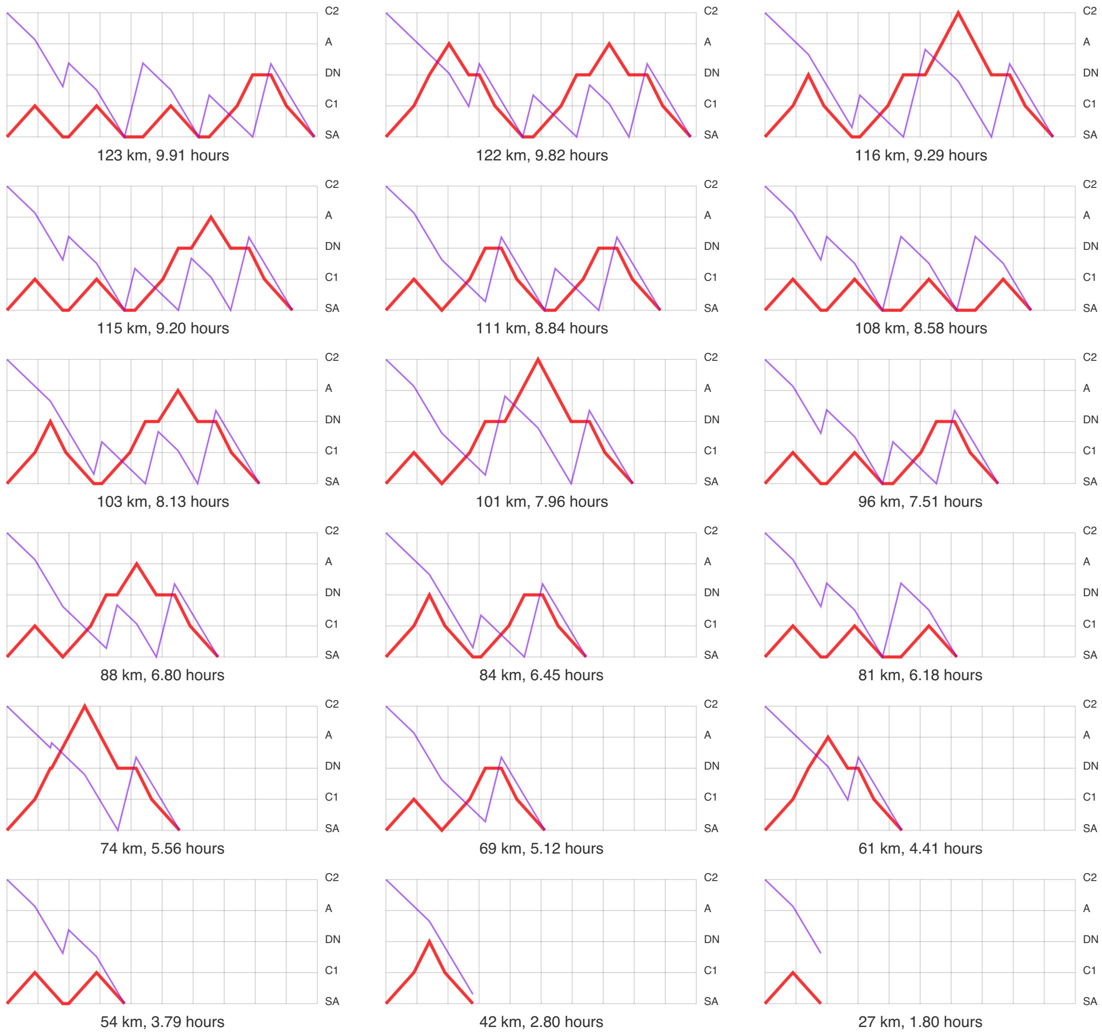


Figure 2. The 18 daily patterns for trips from clinics to the hospital. In each pattern, the horizontal axis is time, in hours. The vertical axis represents battery energy (blue curve), or location (red curve). The locations are SA (St Albert's Hospital), C1 (Chiyani), DN (David Nelson), A (Always), C2 (Chidkamwedzi).

6 Minimising schedule span

Our second possible objective is to find the schedule that completes all current trip requests with the least span. We can formulate this as a set covering problem with the objective function

$$\text{minimise } z = \sum_{i \in \mathcal{P}} x_i \left(m_i D + (1 - m_i) T_i \right)$$

where x_i is an integer decision variable that indicates the number of times pattern i is used, m_i is a binary parameter that indicates whether pattern i is maximal, D is the maximum allowable daily duration, and T_i is the duration of pattern i. We constrain the solution to have at most one non-maximal pattern, which we use on the last day. We may have unused trips on non-last days, so we solve the optimisation as a set covering problem. The constraints are:

$$\begin{aligned} \sum_{i \in \mathcal{P}} x_i p_{ij} &\geq d_j \qquad \forall j \in \mathcal{C} \\ \sum_{i \in \mathcal{P}} (1 - m_i) x_i &\leq 1 \end{aligned}$$

where p_{ij} is the number of trips from clinic j in pattern i, d_j is the demand for trips from clinic j, $\mathcal{C} = \{1, \ldots C\}$ is the set of clinics and $\mathcal{P} = \{1, \ldots P\}$ is the set of patterns.

The table below shows solutions to some example problems:

demand	solution	z
$[2, 0, 0, 0]$	$1 \times [2, 0, 0, 0]$	3.79
$[2, 0, 0, 1]$	$1 \times [1, 0, 0, 1] + 1 \times [1, 0, 0, 0]$	11.80
$[0, 0, 0, 3]$	$2 \times [0, 1, 0, 1] + 1 \times [0, 0, 0, 1]$	25.56

The last example uses maximal patterns with unused trips on the first two days.

In situations where the demand will change before the schedule is complete, the first day should use the trip pattern with the greatest distance and without any unused trips.

The schedule that minimises span does not necessarily use the trip pattern that maximises the daily trip distance. For example, consider the demand $[2, 4, 1, 0]$. The fully-utilised trip pattern that maximises the daily trip distance is $[2, 0, 1, 0]$, whereas the minimum span schedule uses

$$2 \times [1, 2, 0, 0] + 1 \times [0, 0, 1, 0].$$

Figure 3 shows the three-day schedule that results from iteratively selecting the fully-utilised trip pattern that maximises daily trip distance (top), and the three-day schedule that minimises span (bottom).

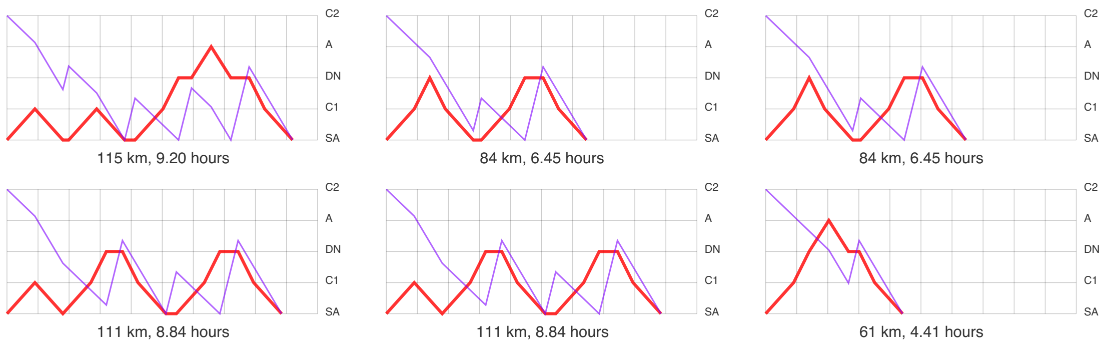


Figure 3. Schedules obtained by iteratively selecting the fully-utilised trip pattern that maximises daily trip distance (top), and by minimising span (bottom).

7 Iterated examples

We can analyse the performance of each of the two scheduling methods—longest distance today, and minimum span—by applying them iteratively over many days with random arrivals at each clinic. Arrivals at clinics are modelled by a Poisson process where the mean time between arrivals, λ, is the same for each clinic and is given. Given the demand sequence, we use each of the two scheduling methods to determine which trip pattern should be used on each day. Any trips not completed are added to the demand for the next day.

Figure 4 shows time-series plots of the number of pending trip requests at each clinic for each of the two scheduling methods, for $\lambda \in \{1.5, 1.2, 1.1, 1.0\}$ days. With $\lambda > 1.1$, the taxi can keep up with demand. With $\lambda = 1.0$, requests arrive slightly faster than can be handled by the single taxi, and so the number of pending trips increases over time. In this example, after 100 days:

- the 'longest distance today' method has remaining demand $[2, 2, 2, 15]$, which could be cleared in 16 days if there were no more arrivals
- the 'minimum span' method has remaining demand $[20, 3, 0, 9]$, which could be cleared in 13 days if there were no more arrivals.

Either method would be adequate in practice provided that the arrival rate is not too large. The 'minimum span' method appears to be slightly better.

8 Conclusion and future work

The 'longest distance today' method is easy to implement—it requires the scheduler to select the first trip pattern that does not do any unnecessary trips from a list of precomputed trip patterns. It is also effective and achieves similar results to the method which finds the minimum span schedule for the currently requested trips.

Further work is required to consider problems where trips are requested from the hospital to the clinics, and where there are multiple taxis operating. We may need to consider limiting the amount of energy taken from each charging station in a day. When we have multiple vehicles, we may also have constraints on the number of vehicles that can charge at any time. We can rearrange the sequence of trips within each vehicle's pattern to minimise overlaps at charging stations.

Finally, with multiple vehicles, we may need to decide where each vehicles should be based.

References

Conrad, R. and M. Figliozzi (2011). The recharging vehicle routing problem. In T. Doolen and E. V. Aken (Eds.), *Proceedings of the 2011 Industrial Engineering Research Conference*.

Erdogan, S. and E. Miller-Hooks (2012). A green vehicle routing problem. *Transportation Research Part E 48*, 100–114.

Laporte, G. (2009). Fifty years of vehicle routing. *Transportation Science 43*(4), 408–416.

Laporte, G., Y. Nobert, and M. Desrochers (1985). Optimal routing under capacity and distance restrictions. *Operations Research 33*(5), 1050–1073.

Parragh, S., K. Doerner, and R. Hartl (2008a). A survey on pickup and delivery problems. part I: Transportation between customers and depot. *Journal für Betriebswirtschaft 58*(1), 21–51.

Parragh, S., K. Doerner, and R. Hartl (2008b, April). A survey on pickup and delivery problems. part II: Transportation between pickup and delivery locations. *Journal für Betriebswirtschaft 58*(1), 81–117.

Schneider, M., A. Stenger, and D. Goeke (2012). The electric vehicle routing problem with time windows and recharging stations. Technical Report 02/2012, University of Kaiserslautern.

Wang, H. and R. Cheu (2013). Operations of a taxi fleet for advance reservations using electric vehicles and charging stations. In *TRB 2013 Annual Meeting*.

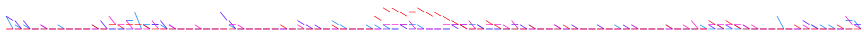
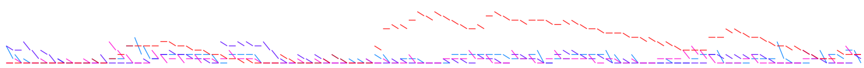
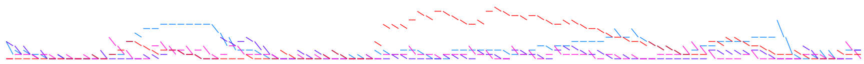
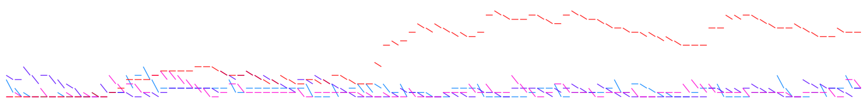
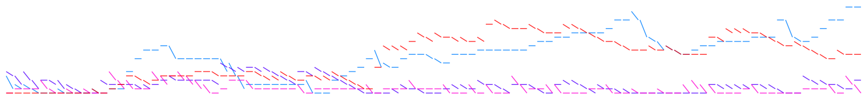
λ	pending graph	remaining
1.5		$[0, 1, 0, 0]$
		$[0, 1, 0, 0]$
1.2		$[2, 0, 1, 0]$
		$[2, 0, 1, 0]$
1.1		$[0, 1, 2, 2]$
		$[2, 2, 0, 1]$
1.0		$[2, 2, 2, 15]$
		$[20, 3, 0, 9]$

Figure 4. Time series graphs of the number of pending trip requests from each clinic for each to the two scheduling methods: longest distance today (top graph for each value of λ) and minimum span (bottom graph for each value of λ). The horizontal axes represent days, from 1 to 100. The vertical axes represent the number of trips requests. Clinics are indicated by colour: Chiyani is light blue, David Nelson is dark blue, Always is purple, Chidkamwedzi is red. The last column shows the number of pending trip requests from each clinic after 100 days.