

Accepted Manuscript

An Efficient Heuristic Algorithm for the Alternative-Fuel Station Location Problem

Trung Hieu Tran, Gábor Nagy, Thu Ba T. Nguyen, Niaz A. Wassan

PII: S0377-2217(17)30906-2
DOI: [10.1016/j.ejor.2017.10.012](https://doi.org/10.1016/j.ejor.2017.10.012)
Reference: EOR 14736



To appear in: *European Journal of Operational Research*

Received date: 29 July 2016
Revised date: 1 July 2017
Accepted date: 7 October 2017

Please cite this article as: Trung Hieu Tran, Gábor Nagy, Thu Ba T. Nguyen, Niaz A. Wassan, An Efficient Heuristic Algorithm for the Alternative-Fuel Station Location Problem, *European Journal of Operational Research* (2017), doi: [10.1016/j.ejor.2017.10.012](https://doi.org/10.1016/j.ejor.2017.10.012)

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

Highlights

- We solve the alternative-fuel station location problem.
- Our heuristic algorithm is based on solving a sequence of sub-problems.
- A parallel computing strategy is used to reduce computation time of the algorithm.
- Our algorithm outperforms current best-known heuristic algorithms.
- Our algorithm can obtain optimal solutions for all tested benchmark instances.

ACCEPTED MANUSCRIPT

An Efficient Heuristic Algorithm for the Alternative-Fuel Station Location Problem

Trung Hieu Tran^{1*}, Gábor Nagy², Thu Ba T. Nguyen³, Niaz A. Wassan²

¹*Laboratory for Urban Complexity and Sustainability, University of Nottingham, Nottingham NG7 2RB, UK*

²*Kent Business School, University of Kent, Canterbury, Kent CT2 7PE, United Kingdom*

³*Southampton Business School, University of Southampton, Southampton SO17 1BJ, United Kingdom*

Abstract

We have developed an efficient heuristic algorithm for location of alternative-fuel stations. The algorithm is constructed based on solving the sequence of subproblems restricted on a set of promising station candidates, and fixing a number of the best promising station locations. The set of candidates is initially determined by solving a relaxation model, and then modified by exchanging some stations between the promising candidate set and the remaining station set. A number of the best station candidates in the promising candidate set can be fixed to improve computation time. In addition, a parallel computing strategy is integrated into solving simultaneously the set of subproblems to speed up computation time. Experimental results carried out on the benchmark instances show that our algorithm outperforms genetic algorithm and greedy algorithm. As compared with CPLEX solver, our algorithm can obtain all the optimal solutions on the tested instances with less computation time.

Keywords: location; alternative-fuel vehicle; heuristic algorithm; parallel computing.

1 Introduction

The problem of alternative-fuel station location is a recent, but very applicable research topic within location science. In essence, what make the problem of determining locations of alternative-fuel refuelling stations different from those of petrol stations is the scarcity of current infrastructure. In fact, alternative-fuel vehicles require a very dense refuelling infrastructure, as these vehicle typically have a short driving range. The alternative-fuel industry is suffering from a “vicious circle”: there is little appetite for infrastructure investment as there are not a sufficient number of alternative-fuel vehicles, the automotive industry can only produce these vehicles at high process as there are not sufficient economies of scales due to limited demand, and customers are discouraged from buying such vehicles due to both their price and the limited refuelling infrastructure. For more information on the current state of the alternative-fuel infrastructure, see European Commission (2013). This topic is especially timely in the light of the recent European directive requiring Member States to provide a minimum coverage of refuelling points for alternative fuels (European

*Correspondence author.

Email address: trunghieu.tran@nottingham.ac.uk (T.H. Tran).

Commission, 2014). The directive provides a regulatory framework for alternative fuels such as hydrogen, electricity, liquefied natural gas and compressed natural gas. The targets are very ambitious. Compressed natural gas stations and hydrogen stations are to be built along the European TEN-T core network at intervals of 400 and 300 kilometres, respectively. The electricity refuelling network is to be multiplied significantly, from about 12,000 to 800,000 charging stations. For further information of the European plans, see European Commission (2013). Thus, this is the right time for Operational Researchers to devote their energies to finding optimal or near-optimal locations for alternative-fuel refuelling facilities.

From an Operational Research point of view, the problem is clearly one of location. However, traditional covering models, such as maximum-cover, are not applicable, as these related to demand arising at nodes. In the context of refuelling, the demand arises from vehicle journeys, thus origin-destination flows rather than single points. Since the seminal paper of Hodgson (1981), a steady stream of research has been devoted to such “flow-capturing” problems of which our flow-refuelling problem is a subtle variation. The main aim of this paper is to take this research somewhat further by developing an efficient heuristic algorithm based on the concept of restricted subproblems. Moreover, we wish to present a comparison of models and formulations to enable the reader to get an insight into the state of the art.

We first review the literature review of this problem, then discuss the mathematical models proposed for it in detail. Sections 4 and 5 present our new heuristic algorithm and show our numerical experimentation with it, respectively. Finally, we draw some conclusions from our study and offer ideas for future research.

2 Literature review

We aim to present a relatively extensive although not comprehensive review of relevant literature. We believe this will help the reader understand the variety of approaches and model variants that exist for our problem. Moreover, the method put forward in this paper is potentially extensible to the variants discussed in this section.

2.1 The flow-capturing location model

The flow-refuelling location model (FRLM), introduced by Kuby and Lim (2005), has its origins in the flow-capturing location model (FCLM) of Hodgson (1990). Hodgson originally investigated the concept of “locating facilities on the home-to-work journey”, see Hodgson (1981). He observed that, unlike traditional location models, where facilities are to be sited near customer locations, in some cases it makes more sense to locate facilities near routes that customers already take. The example mentioned is that of locating childcare centres. Hodgson (1990) formalised this, creating the so-called flow-capturing location model. A main theme of this paper is that of cannibalisation, that is, the multiple (and thus unnecessary) capturing of the same flows by different facilities. A cannibalising and a non-cannibalising heuristic are compared, showing the superiority of the latter. The author showed that basing locational decisions simply on arc flow volumes is not sufficient as locating facilities on this basis may lead to cannibalisation, because origin-destination flows traverse multiple arcs on a single path. Instead, models should be based on detailed origin-destination flow data, which can be counted as captured only once. This seminal work spurred a number of flow-capturing papers, some of which we review here. It may be interesting to point out that models that combine both flow

and node capturing exist, see for example Hodgson and Rosing (1992) and Berman (1997), who combined flow-capturing and p-median/max-cover objectives.

An important aspect of the FCLM is that any flow (origin-destination pair) is captured by a single facility. This is sensible as one would not, for example, stop at every roadside supermarket on the way home, one stop is sufficient to satisfy one's shopping needs. However, some variations exist. For example, Hodgson and Berman (1997) considered facilities (roadside advertisement billboards) where one billboard can be considered to "capture" a passing motorist, but seeing the same advertisement again will "reinforce the message" and is thus beneficial. Another interesting example is given in Hodgson et al. (1996), where a facility cannot capture the entire flow (origin-destination trip). The problem at hand was locating inspection stations where drunk drivers or drivers of unlicensed hazardous waste material may be intercepted. The drivers are prevented from causing damage "downstream" from the inspection station but the station cannot prevent accidents or spillages on the previous part of the journey.

It is often assumed that in order to capture a flow, a facility must lie on the origin-destination path. However, it may also be reasonable to assume, especially if the network of facilities is very sparse, that drivers would make some reasonable detours to visit a facility. Berman et al. (1995) allowed a flow to be captured if it passes a facility within a specified distance. Variations of the model are also considered, most importantly an assumption whereby a portion of the flow is captured proportional to the distance between facility and path. Berman (1997) considered a median-objective problem with deviation distances.

2.2 The flow-refuelling location model

The main difference of the FRLM from the FCLM is that a single facility may be unable to capture an entire flow. This is due to the issue of "limited range", namely, that a vehicle may not be able to undertake a given origin-destination journey with a single refuelling stop. This model is most applicable to vehicles powered by alternative fuels, such as hydrogen or electricity. Such vehicles normally can cover a shorter distance on a full tank than traditional gas-guzzling vehicles. In addition, the availability of alternative fuel refuelling stations is very limited. However, as pointed out by Zockaie et al. (2016), the model is also applicable to the location of conventional refuelling stations in developing countries where infrastructure is as yet lacking. Likewise, it can be applicable to territories with sparse population (and hence sparse refuelling infrastructure).

Kuby and Lim (2005) introduced the FRLM, motivating the new model with the above concept of vehicle range. They observe that origin-destination data, rather than simple traffic count on edges, is required to model this problem properly. Multiple facilities may be required to serve individual journeys. Unlike in the FCLM, it can be shown that it is not sufficient to consider only node locations for facilities, thus making the problem harder to solve. An integer programming formulation is provided; this will be looked at in Section 3. The authors drew the following conclusions from their experimentation:

- The longer the vehicles' range the fewer facilities are needed to capture all the demand, but placing facilities only at nodes (junctions) may be unable to provide total coverage.
- There is a lack of convexity in the trade-off curve between the number of refuelling stations opened and the volume of flow they capture, unlike in the FCLM that exhibits convexity.
- Greedy solution approaches tend to give very poor results, much poorer for the same instances as they give for the FCLM.

- Unlike in maximum cover location problems, alternate optima do not often occur in the FRLM. (The authors hypothesise that in this respect the FCLM may fall between the FRLM and the max-cover problem.)

Upchurch et al. (2009) extended the above model to the case of capacitated facilities. An integer programming formulation is presented. (We do not consider the case of capacity limitations and hence this model will not be discussed in detail.) While in the previous papers (and indeed in a large part of the literature), the objective is to maximise the flow captured, the model of Wang and Lin (2009) aims to minimise costs such that all flows are served. The authors devised a "vehicle refueling logic" that is more involved but also more flexible than that of previous models. Another important difference is that this model requires only knowledge of origin-destination distances, but not of the origin-destination flow data. Lim and Kuby (2010) designed some heuristic algorithms for the FRLM. One of their motivations for doing so is the complexity of the Kuby and Lim (2005) mathematical formulation. There are three heuristics but with a common subroutine to evaluate the objective function value:

- The "greedy-adding" or "add" algorithm simply adds one more facility in each iteration so as to maximise the increase in flow capture.
- The "greedy-adding with substitution" or "add-swap" algorithm also attempts in each iteration to replace an existing facility with a potential facility. Thus, each iteration consists of an "add" and a "swap" move.
- The genetic algorithm is based on the chromosome representation of a list of open facilities. (As the number of facilities is fixed in advance, this is more reasonable than a 0-1 representation.)

Unlike, say, the maximum covering problem, the evaluation of a given solution is not a straightforward task. For a given solution, i.e. a set of facilities, the evaluation subroutine must evaluate every origin-destination path to see whether it is refuelable – if so, its flow is added to the objective function value. We note that all the algorithms are capable of handling pre-existing facilities. The authors found that the greedy algorithms perform quite well, nearly as well as the genetic algorithms (except for the case of a very short vehicle range), and are significantly faster.

Lately, research has focused on obtaining more efficient formulations to the FRLM. The motivation for this is that the original Kuby and Lim (2005) model requires a massive preprocessing effort. All facility combinations must be checked whether they can refuel each origin-destination journey and the resulting coefficients inserted as input into the integer programming model. This takes an immense amount of time, so much so, that the authors could not even generate the integer programming model for their benchmark instance, let alone solve it. Capar and Kuby (2012) put forward a more complex model, but without the above preprocessing requirement. This new formulation is in fact as fast as the greedy heuristics of Lim and Kuby (2010). Capar et al. (2013) offered a more efficient formulation than Capar and Kuby (2012). While the previous model used a "node-cover/path-cover" logic, the authors propose an "arc-cover/path-cover" model. In the model of MirHassani and Ebrazi (2013), the number of facilities is not fixed in advance, as it explicitly takes into account their establishment costs. The logic of their formulation is developed from a single-path to a multi-path formulation. The differences between these models will be analysed in more detail in Section 3.

Nie and Ghamami (2013) considered the particular case of locating refuelling stations along a travel corridor. Their objective was cost minimisation rather than flow maximisation, and this included battery production

cost. Apart from recharging electric vehicles (which takes considerable time) the authors have also considered the concept of "battery swapping" (replacing a depleted car battery with a fully charged one which takes just a few minutes). Wen et al. (2014) investigated both maximal flow capture and total flow capture (cost-minimising) models. Their formulations are based on set covering and do not require the evaluation of all feasible combinations of locations. Ghamami et al. (2016) extended the corridor model of Nie and Ghamami (2013), allowing for congestion and delay at charging stations. Their formulation is based on the assignment problem. The authors have also designed a simulated annealing metaheuristic.

2.3 FRLM with deviation

The FRLM assumes that refuelling stations are located on origin-destination paths. However, just like in the FCLM (see Berman et al., 1995, and Berman, 1997), one could consider the situation where driver make some detour from their shortest path in order to refuel. This is a very reasonable assumption given that the network of alternative-fuel stations is a very sparse one at the moment.

Kim and Kuby (2012) introduced the deviation-flow refuelling location model. It is assumed that drivers are willing to make some detour from their shortest paths to visit a refuelling facility, that they would visit the facility/facilities that represent the smallest possible detour, and that the volume of flow captured by a facility decreases with the deviation distance required to reach it. Similarly to the FRLM the preprocessing is not a straightforward task. Evaluating all possible detour journeys is even more cumbersome than evaluating simple origin-destination paths. (This is further complicated by the fact that shortest paths do not contain loops but shortest feasible deviation paths may contain loops.) Then, as facilities on detours do not capture 100% of the flow, another algorithm is required to calculate the volume of flow captured. Finally, as in the FRLM, these paths must be evaluated to see whether they can refuel an origin-destination journey. (This is further complicated by the possibility that a round-trip journey may be optimally refuelled by different facilities on the outbound and the inbound legs.) The authors presented an extension of the Kuby and Lim (2005) integer programming model to account for deviations. Their discussion focuses on the effects of vehicle range, lack of convexity in the trade-off curve, effects of the deviation distance, effects of the function relating the volume of flow captured to the deviation distance, and the effects of multiple shortest paths. Kim and Kuby (2013) applied a network transformation heuristic to mitigate the preprocessing issues encountered by Kuby and Lim (2005) and then solve the deviation-flow FRLM using the "add" and "add-swap" greedy heuristics of Lim and Kuby (2010). Kang and Recker (2015) considered drivers' routing decisions and thus explicitly allow detours in their location-routing type model. The multi-period models of Bhatti et al. (2015) and Miralinaghi et al. (2016) also allow drivers to make detours. Huang et al. (2015) considered multiple deviation paths; this model also requires finding the vehicle routes as here drivers do not necessarily take the shortest path. Yildiz et al. (2016) also integrated finding drivers' deviation routes into the refuelling location problem; they present improved models and solves these via branch-and-price. Zockaie et al. (2016) considered capacitated refuelling stations. Their objective was to minimise refuelling detours. A simulated annealing metaheuristic was designed.

We note that this problem is somewhat similar to the vehicle routing allocation problem, a subproblem of the family of location-routing problems. (The reader is referred to Nagy and Salhi, 2007, Prodhon and Prins, 2014, and Drexl and Schneider, 2015, for more information.) In the next subsection we look at location-routing type refuelling models.

2.4 Location-routing type refuelling models

Location-routing is a generic name for locational problems where routing aspects are taken into account. This approach is useful as ignoring these aspects can easily lead to suboptimal locational solutions. Such problems are usually solved by an iterative approach (one that iterates between locational and routing phases) or an integrated approach (where the routing algorithm is integrated into the location algorithm). For good overviews and a summary of recent work see the review papers of Nagy and Salhi (2007), Prodhon and Prins (2014) and Drexl and Schneider (2015).

Kang and Recker (2015) devised a location-routing type model for location alternative-fuel stations. The location part is based on set covering and the routing part is formed by the “household activity pattern problem.” This latter problem captures people’s day-to-day activities and is a more involved model than just considering home-to-work journeys as in the FRLM. As it is more suited to shorter journeys, the model does not need to consider the issue of multiple refuelling stops. However, time windows considerations are introduced. The authors present an integer programming formulation, based on set covering. The differences from standard location-routing problems are highlighted. Numerical experimentation shows that including tour-planning aspects in choosing locations gives a better model with more robust locational decisions; in particular, it shows that a sparser infrastructure network is sufficient than a nodal-demand based model would suggest. Jung et al. (2014) named this type of problem “itinerary interception” problem, and furthermore consider dynamic problems with stochastic service demand and queueing delay. Yang and Sun (2015) considered a location-routing model where the facilities are refuelling stations. An iterative heuristic is put forward.

2.5 Multi-objective FRLM

The FRLM, like its predecessor the FCLM, is based on the concept of demand arising out of flows (journeys), rather than nodes (customer locations), as in traditional location models. However, already the proponents of the FCLM noted that both type of demand may coexist, see Hodgson and Rosing (1992) and Berman (1997). There is a very limited literature on the FRLM that takes into account also the fact that customers may prefer to refuel close to home, possibly to enable them to undertake short frequent local trips not captured by FRLM models that focus on long-distance journeys. Evidence for the use of this model is limited, with a survey by Kelley and Kuby (2013) concluding that drivers, if faced with a choice, are ten times more likely to refuel at a facility with the shortest detour rather than at the one closest to their home.

Wang and Wang (2010) extended the model of Wang and Lin (2009) to take into account both “intercity and intra-city travel” by adding set covering constraints to account for node coverage. The authors consider the dual objectives of minimising facility cost and maximising population coverage. They discuss the trade-offs between objectives and the influence of the vehicle range and the coverage distance on the solution. Badri-Koohi and Tavakkoli-Moghaddam (2012) observed that the number of stations was fixed in most previous models and stated a preference for the model to determine this value. To this end, they considered the costs of establishing alternative-fuel stations. A long-term planning approach was adopted, taking into account depreciation and inflation. Flow demand in their model is broken up into segments of routes and then approximated by node demand. Moreover, the location model used is continuous, rather than the discrete location model used in all other FRLM papers.

2.6 Multi-period FRLM

Most refuelling models (as indeed most locational models) assume that the locational decision is based on static data and is a one-off decision. However, in reality demand and other input data may change over time and locations may either need to be adjusted periodically to account for this or planned at the outset in such a way that they serve well such stochastic demand. Often such models are called dynamic or stochastic, see Ballou (1968) and Rosenhead et al. (1972). They are also used in combined location-routing models, see Nambiar et al. (1989), Salhi and Nagy (1999) and Albareda-Sambola et al. (2012). These models are especially applicable to the FRLM, as demand for alternative fuels is likely to rise. Bhatti et al. (2015) and Hosseini and MirHassani (2015) both considered two-stage models where facilities are established in two tranches. The former propose a two-stage model with learning, whereby the decision-maker is able to reap information about demand once the first set of facilities is established. In the latter model, firstly permanent stations are established and these are supplemented later with mobile refuelling stations. This model uses the refuelling logic of MirHassani and Ebrazi (2013). Chung and Kwon (2015) and Miralinaghi et al. (2016) allowed for more than two time periods; the former allows locations to change in each time period while the latter requires the locations to be fixed for the entire planning horizon. Miralinaghi et al. (2016) observed that this difference makes the former model more applicable to electric and the latter to hydrogen vehicles. Chung and Kwon (2015) based their model on the expanded network of MirHassani and Ebrazi (2013). Miralinaghi et al. (2016) used branch-and-bound and Lagrangean relaxation.

3 Models for the alternative-fuel station location

3.1 Nomenclature and parameters

In the subsection, we present all notations of indices, sets, parameters and decision variables that are used in the formulations and the proposed algorithm for the alternative-fuel station location problem. The formulations of Kuby and Lim (2005), Capar and Kuby (2012), Capar et al. (2013) and MirHassani and Ebrazi (2013), denoted by [P1], [P2], [P3] and [P4] respectively, will be described in details in next subsection. Our heuristic algorithm is labeled as [HA]. Since some common notations are used in four of the formulations, we notice the referred formulations in the last column. In addition, to present the formulations in a uniform way we changed some notations from the original notations used in the literature. This should hopefully help the reader in appreciating the differences between the formulations.

Indices and sets:

i, j, k, n	Indices for nodes (i.e., station locations/sites)	[P1]-[P4]
q	Index for paths (i.e., origin-destination pairs)	[P1]-[P4]
h	Index for combinations of nodes	[P1]
m, r	Indices of the order of candidate nodes on a given path	[P2]
t	Index of the state of a candidate node ($t = 0$: station built, $t = 1$: station not built)	[P2]
k_m^q	Index i corresponding to the m^{th} candidate node in N_q	[P2]
N	Set of nodes, $N = \{1, 2, \dots, n\}$	[P1]-[P4]
Q	Set of paths	[P1]-[P4]
H	Set of all potential node combinations	[P1]
\bar{N}_h	Set of nodes in combination h : $\bar{N}_h = \{i \in N a_{hi} = 1\}$	[P1]

M_q	Number of candidate nodes within the distance interval $(0, D_q - R/2)$ on path q , if $(D_q - R/2 \leq 0)$, then $M_q = 0$	[P2]
N_q	Set of candidate nodes on path q sorted in sequential order from origin to destination $N_q = \{1 \text{ (origin)}, 2, 3, \dots, n_q \text{ (destination)}\}$	[P2]
A_q	Set of arcs on path q in the original network $G = (N_q, A_q)$	
N_{mt}^q	Set of candidate nodes accessible from the m^{th} candidate node on path q :	[P2]
	$N_{mt}^q = \begin{cases} \{N_q d_{mr}^q \leq R, r > m\}, & \forall q \in Q, \quad m = 1, 2, \dots, M_q, t = 1, \\ \{N_q d_{mr}^q < R, r > m\}, & \forall q \in Q, \quad m = 2, \dots, M_q, t = 0, \\ \{N_q d_{mr}^q \leq R/2, r > m\}, & \forall q \in Q, \quad m = 1, t = 0. \end{cases}$	
\vec{a}_{jk}	Directed arc starting from node j and ending at node k	[P3]
\vec{A}_q	Set of directed arcs on path q , sorted from origin to destination and back to origin	[P3]
K_{jk}^q	Set of candidate nodes that can refuel the directed arc $\vec{a}_{jk} \in \vec{A}_q$	[P3]
Q_i	Subset of Q which contains all the paths passing candidate node i	[P4]
\hat{N}_q	Extended set of candidate nodes on path q (including source and sink dummy nodes)	[P4]
\hat{A}_q	Extended set of arcs in the extended network, $G_E = (\hat{N}_q, \hat{A}_q)$	[P4]
N^*	Set of promising nodes for restricted subproblems	[HA]
N^0	Set of non-promising nodes, $N^0 \cup N^* = N$	[HA]
N^{*1}	Set of promising nodes in which x_i is fixed to be 1 when solving subproblems	[HA]
N^{*0}	Set of non-fixed promising nodes, $N^{*1} \cup N^{*0} = N^*$	[HA]
N_m^*	m^{th} neighboring set of promising nodes generated by k -exchange	[HA]
Q^*	Set of promising paths defined by N^*	[HA]

Parameters:

R	Range of vehicles	[P1]-[P4]
f_q	Volume of traffic flow on path q	[P1]-[P4]
p	Number of stations to be located	[P1]-[P4]
a_{hi}	Coefficient that equals to 1 if station i is in combination h and 0 otherwise	[P1]
b_{qh}	Coefficient that equals to 1 if combination h can refuel path q and 0 otherwise	[P1]
d_{mr}^q	Distance between the m^{th} and r^{th} candidate nodes in N_q	[P2]
D_q	Length of the shortest path for path q	[P2]
ϵ	A very small positive number	[HA]
UB	Upper bound	[HA]
LB	Lower bound	[HA]

Decision variables:

$$x_i = \begin{cases} 1 & \text{if station is located at node } i, \\ 0 & \text{otherwise.} \end{cases} \quad [P1]-[P4]$$

$$y_q = \begin{cases} 1 & \text{if the flow on path } q \text{ is refueled,} \\ 0 & \text{otherwise.} \end{cases} \quad [P1],[P2],[P3]$$

$$v_h = \begin{cases} 1 & \text{if all facilities in combination } h \text{ are opened,} \\ 0 & \text{otherwise.} \end{cases} \quad [\text{P1}]$$

$$c_{m0}^q = \begin{cases} 1 & \text{if } x_{k_m^q} = 0 \text{ and vehicles on path } q \text{ have enough fuel remaining} \\ & \text{at the } m^{\text{th}} \text{ candidate site to be able to reach the next} \\ & \text{open fuel station on path } q \text{ without running out of fuel,} \\ 0 & \text{if } x_{k_m^q} = 0 \text{ and it cannot reach to an open fuel station.} \end{cases} \quad [\text{P2}]$$

$$c_{m1}^q = \begin{cases} 1 & \text{if } x_{k_m^q} = 1 \text{ and vehicles on path } q \text{ have enough fuel after} \\ & \text{refueling at the } m^{\text{th}} \text{ candidate site to be able to reach the next} \\ & \text{open fuel station on path } q \text{ without running out of fuel,} \\ 0 & \text{if } x_{k_m^q} = 1 \text{ and it cannot reach to an open fuel station.} \end{cases} \quad [\text{P2}]$$

$$y_{ij}^q = \text{flow on an arc } (i, j) \in \hat{A}_q \text{ in the extended network of path } q. \quad [\text{P4}]$$

3.2 Formulations of the alternative-fuel station location

Kuby and Lim (2005) introduced the FRLM, an extension of the FCLM, which locates p refuelling facilities to maximise the number of trips refueled. In the first model, the authors proposed a mixed-integer programming formulation for the nodes-only version of the problem. The model is supported by an algorithm to determine all combinations of nodes that can refuel a path given. The combinations of nodes depend on the length of the path and the maximum vehicle range assumed. Based on the definitions of sets, parameters and decision variables described for [P1] in Section 3.1, the mathematical formulation can be presented as follows

[P1]: (Kuby and Lim, 2005)

$$\max \sum_{q \in Q} f_q y_q \quad (1)$$

$$\text{s.t. } \sum_{h \in H} b_{qh} v_h \geq y_q, \quad \forall q \in Q, \quad (2)$$

$$a_{hi} x_i \geq v_h, \quad \forall h \in H, i \in \bar{N}_h, \quad (3)$$

$$\sum_{i \in N} x_i = p, \quad (4)$$

$$x_i, y_q, v_h \in \{0, 1\}, \quad \forall i \in N, q \in Q, h \in H. \quad (5)$$

The objective function (1) aims to maximise the total traffic volume which can be refueled. Constraints (2) require at least one valid combination of nodes available for path q to be refueled. Constraints (3) keep v_h to be zero unless all the facilities i in combination h are built. Constraint (4) ensure exactly p facilities to be located. Finally, constraints (5) define binary variables.

Unfortunately, generating the valid combinations of nodes H can be computationally burdensome. Some efforts have been performed to reformulate the problem by eliminating the usage of pre-generation of valid combinations. Capar and Kuby (2012) presented such a mixed-binary-integer programming formulation for solving efficiently the problem. By introducing new decision variables c_{m0}^q and c_{m1}^q , along with removing variables v_h relevant to the combinations, and modifying some corresponding constraints, a new mathematical model is formulated by

[P2]: (Capar and Kuby, 2012)

$$\max \sum_{q \in Q} f_q y_q \quad (6)$$

$$\text{s.t. } c_{mt}^q + (-1)^t x_{k_m^q} \leq 1 - t, \quad \forall q \in Q, t \in \{0, 1\}, m = 1, 2, \dots, M_q, \text{ if } M_q \neq 0, \quad (7)$$

$$c_{mt}^q - \sum_{n \in N_{mt}^q} x_n \leq 0, \quad \forall q \in Q, t \in \{0, 1\}, m = 1, 2, \dots, M_q, \text{ if } M_q \neq 0, \quad (8)$$

$$c_{m1}^q - \sum_{n \in x_{k_m^q} \cup N_{m1}^q} x_n \leq 0, \quad \forall q \in Q, m = M_q + 1, \quad (9)$$

$$c_{m0}^q = 0, \quad \forall q \in Q, m = M_q + 1, \quad (10)$$

$$\sum_{m=1}^{M_q+1} \sum_{t \in \{0,1\}} c_{mt}^q = (M_q + 1) y_q, \quad \forall q \in Q, \quad (11)$$

$$\sum_{i \in N} x_i = p, \quad (12)$$

$$x_i, y_q, c_{m0}^q, c_{m1}^q \in \{0, 1\}, \quad \forall i \in N, q \in Q, m = 1, 2, \dots, M_q. \quad (13)$$

The objective function (6) is identical to that in [P1]. Constraints (7) governs the relationship between variables c_{mt}^q for the m^{th} node on path q and variables x at the same node to ensure that c_{mt}^q equals to one in the right scenario. Constraints (8) governs the relationship between variables c_{mt}^q for the m^{th} node on path

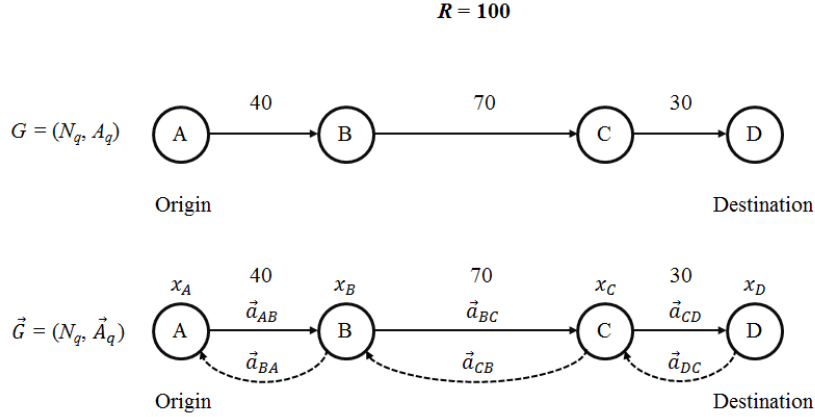


Figure 1: An example of fuel station location on a single path for formulation [P3].

q and variables x further along the path to ensure that c_{mt}^q equals to zero unless another refuelling station can be reached from the m^{th} node on path q . Constraints (9) and (10) handle the scenario of nodes within half the range of the destination of path q . Constraints (11) allow the traffic flow on path q to be completely refueled if the sum of the refueled stations on the path equals to the number of the stations considered (i.e., $M_q + 1$). Constraint (12) is the same with constraint (4) in [P1] that build exactly p facilities. Finally, constraints (13) define binary variables.

Although [P2] is more efficient than [P1] as it eliminates the usage of combination pre-generation, the number of new decision variables and constraints significantly increases the size of the model. Hence, [P2] may not be efficient for real-world applications with large size. Capar et al. (2013) developed an arc cover-path-cover formulation for the alternative-fuel station location problem as replacing the usage of combination pre-generation by the concept that a path can be refueled if all directed arcs on the round-trip path are served. For example, in Figure 1 path q (i.e., A-D pair) can be refueled if the traffic volumes in all directed arcs (A, B), (B, C), (C, D), (D, C), (C, B), and (B, A) are served. For each directed arc \vec{a}_{jk} on path q , a set of candidate sites K_{jk}^q which can refuel the arc has to be determined based on the vehicle range R . For example, in Figure 1 (with $R = 100$) the set of candidate sites for directed arc \vec{a}_{CD} is $\{B, C\}$, since only if refuelling at station B or C we can drive to D without running out of fuel. The arc cover-path-cover mathematical formulation can be described as follows:

[P3]: (Capar et al., 2013)

$$\max z_0 = \sum_{q \in Q} f_q y_q \quad (14)$$

$$\text{s.t.} \quad \sum_{i \in K_{jk}^q} x_i \geq y_q, \quad \forall q \in Q, \vec{a}_{jk} \in \vec{A}_q, \quad (15)$$

$$\sum_{i \in N} x_i = p, \quad (16)$$

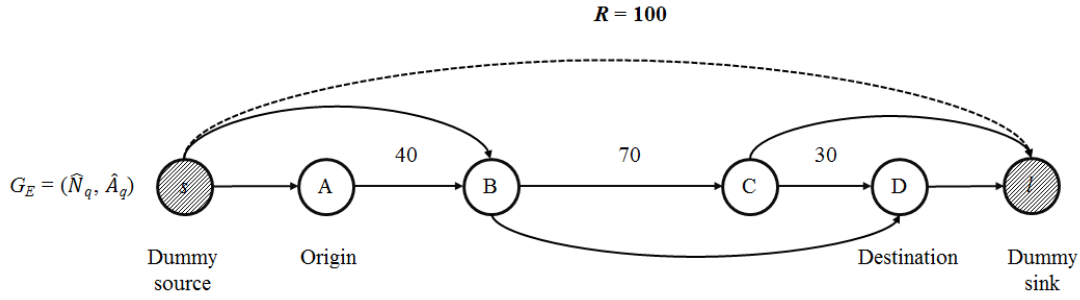


Figure 2: An example of original network vs. extended network for formulation [P4].

$$x_i, y_q \in \{0, 1\}, \quad \forall i \in N, q \in Q. \quad (17)$$

The objective function (14) is still to maximise the total traffic volume which can be refueled. Constraints (15) represent the innovative formulation in which path q is refuelable if and only if every directed arc along the path is travelable after refuelling at one of the facilities built. Constraint (16) makes sure that only p facilities are located, while constraints (17) are to define binary variables.

MirHassani and Ebrazi (2013) considered the case where the number of facilities is not fixed in advance but is determined from the input data of the facility establishment cost. The authors proposed a flexible reformulation for a general refuelling station location problem, in which some paths may not be covered by station sites just located at the nodes of the network. In particular, the formulation allows to consider the number of candidate sites modified on the arcs of the paths to refuel the traffic flows by the additional midarc sites. In their paper, the midarc sites were uniformly generated (i.e., identical length of midarcs) along the arc in order to prevent the network from existing uncovered paths. A variation of their formulation was also presented for the case of a set value of p . This includes a new variable y_{sl}^q given by

$$y_{sl}^q = \begin{cases} 1 & \text{if the flow on path } q \text{ cannot be refueled,} \\ 0 & \text{otherwise,} \end{cases}$$

where s and l denote the dummy source and sink nodes respectively, which are added to construct the extended network. In Figure 2, $G = (N_q, A_q)$ represents the original network with the set of candidate nodes and arcs on path q (known as N_q and A_q respectively), where q is the path from node A to D . The distance between nodes in the network is given, along with a range of vehicles $R = 100$. In the extended network $G_E = (\hat{N}_q, \hat{A}_q)$, the dummy source and sink nodes are included into the set of candidate nodes (referred to as \hat{N}_q). In addition, other additional arcs with respect to new decision variables y_{ij}^q are built by a four-step procedure in MirHassani and Ebrazi (2013). In particular, an arc is added if the distance between two nodes (defined by the arc) does not exceed the range of vehicle (for example, the additional arcs sB, BD, Cl). Finally, an arc between two dummy nodes is modified to construct the set of arcs \hat{A}_q in the extended network.

[P4]: (MirHassani and Ebrazi, 2013)

Table 1: A comparison of the sizes of the four formulations for the alternative-fuel station location problem.

Formulation	No. of binary variables	No. of continuous variables	No. of constraints
[P1]	$ N $	$ H + Q $	$\sum_{h=1}^H \bar{N}_h + Q + 1$
[P2]	$ N + Q $	$2 \sum_{q=1}^Q M_q $	$4 \sum_{q=1}^Q M_q + 3 Q + 1$
[P3]	$ N $	$ Q $	$\sum_{q=1}^Q \bar{A}_q + 1$
[P4]	$ N $	$\sum_{q=1}^Q \hat{A}_q $	$\sum_{q=1}^Q \hat{N}_q + \sum_{i=1}^N Q_i + 1$

$$\max \sum_{q \in Q} f_q (1 - y_{sl}^q) \quad (18)$$

$$\text{s.t.} \quad \sum_{\{j|(i,j) \in \hat{A}_q\}} y_{ij}^q - \sum_{\{j|(j,i) \in \hat{A}_q\}} y_{ji}^q = \begin{cases} 1, & i = s, \\ -1, & i = l, \\ 0, & i \neq s, l, \end{cases} \quad \forall q \in Q, i \in \hat{N}_q, \quad (19)$$

$$\sum_{\{j|(j,i) \in \hat{A}_q\}} y_{ji}^q \leq x_i, \quad \forall i \in N, q \in Q_i, \quad (20)$$

$$\sum_{i \in N} x_i = p, \quad (21)$$

$$x_i \in \{0, 1\}, \quad \forall i \in N, \quad (22)$$

$$y_{ij}^q \geq 0, \quad \forall q \in Q, (i, j) \in \hat{A}_q. \quad (23)$$

The objective function (18) is to maximise the total traffic volume which can be served. Constraints (19) are flow conservation equations for the extended network G_E . Constraints (20) make sure that an inflow passes through a site only if a refuelling station is located at that site. Constraint (21) ensures that only p facilities are located. Constraints (22) and (23) define location variables as binary variables and flow variables as nonnegative, respectively.

A comparison of the four formulations based on the number of variables (i.e., binary and continuous) and the number of constraints is shown in Table 1. In the table, $|\cdot|$ is the number of elements in a set. Since some binary variables in the formulations can be relaxed in a mixed-integer program (solved by CPLEX solver)

as continuous variables with an upper bound of 1 without impact to the optimal integer solution, to make a fair comparison we relaxed as many as possible binary variables of the formulations. Only the formulation introduced by Kuby and Lim (2005) needs a support procedure to generate valid combinations of stations H . It takes a lot of time to do and is thus known as the less efficient model.

Common assumptions used to formulate the problem consist of

- The traffic flow between an origin-destination pair is only through a single path (i.e., shortest path) and its volume is given.
- Drivers have full knowledge of refuelling station locations along their path and know when to refuel in order to complete their trip without running out of fuel.
- All vehicles have the same vehicle range R .
- Only nodes in the network are considered to be refuelling station locations.
- Refuelling stations are uncapacitated.
- Fuel consumption is proportional to traveling distance.

4 A heuristic algorithm for the FRLM

The idea of using the optimal solution obtained by relaxing the integrality constraints of the mixed-integer linear programming (MILP) problems to generate a set of initial solutions for well-known meta-heuristic algorithms appeared in some research works (for example, Singh and Sharma (2008) with simulated annealing for facility layout problem, Tran and Ng (2011) with water flow algorithm for vehicle routing problem, etc.). However, such metaheuristic algorithms only used the information of the optimal solution in the first iteration, but did not use to support search process in next iterations. Recently, the idea of using the information of the optimal solution to support search process further (i.e., to establish a set of promising candidate variables) has been developed by Angelelli et al. (2010), known as Kernel Search. This method identifies subsets of decision variables for the MILP problem by solving the relaxation problem and then solves the restricted problems to optimality by commercial MILP solvers. It has been successfully applied for several optimisation problems, such as the multi-dimensional knapsack problem (Angelelli et al., 2010), the portfolio selection problem (Angelelli et al., 2012), the index tracking problem (Guastaroba and Speranza, 2012a), the multi-source capacitated facility location problem (Guastaroba and Speranza, 2012b), and the single-source capacitated facility location problem (Guastaroba and Speranza, 2014). Following the successful applications, we develop an efficient heuristic algorithm, based on the concept of restricted subproblems, for solving the alternative-fuel station location problem. Compared with Kernel Search algorithm, our algorithm has some small differences as follows:

- Although there are two sets of binary variables in the problem, we only explicitly restricted on the location variables x_i . The number of path variables y_q is determined based on the restricted location variables. Thus, there is implicit restriction. Restricting on one set of binary variable may help the proposed algorithm obtain the good balance of solution quality and CPU time, instead of restricting on all the sets of binary variables as in Kernel Search algorithm.

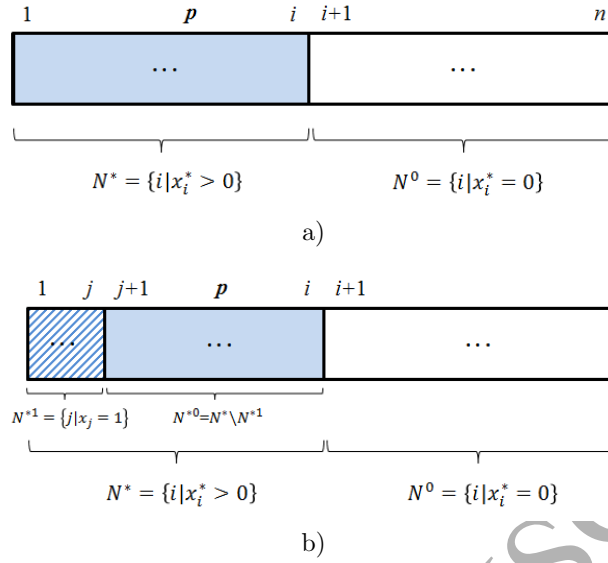


Figure 3: Illustration of a) constructing the set of restricted nodes and b) fixing a number of station locations in the set for the subproblems.

- The size of the restricted subproblems equals to the number of location variables with positive value (i.e., $x_i^* > 0$, where x^* denotes the optimal location as solving the relaxation model initially). In Kernel Search algorithm, size of the restricted subproblems is usually a given arbitrary parameter. As a result, initial promising variable set may include $x_i^* = 0$ or remove some promising variables $x_i^* > 0$, which may lead to spend additional CPU time to find the best solutions.
- k -exchange neighborhood is used to generate a pool of the restricted subproblems for parallel computing strategy in the proposed algorithm.

In the next subsection, we describe how to construct the restricted subproblems in the proposed algorithm. Then, we show how to integrate a parallel computing strategy with k -exchange neighborhood structure for improvement of the computation time of the algorithm. Finally, we present the framework of the proposed algorithm for solving the FRLM.

4.1 Restricted subproblems

The formulation of alternative-fuel station location problem [P3] in Capar et al. (2013) is referred as the original problem (labeled by ASLP) in the heuristic algorithm. As described in the section above, the objective function (14) is to maximise the round-trip traffic volume served. Constraints (15) ensure that a flow on path q is refueled if and only if each directed arc on the path is travelable after refueled by one of the opening stations. Constraint (16) makes sure that only p stations are located to serve vehicles in the transport network. Finally, constraints (17) define binary variables. The formulation [P3] is modified to generate restricted subproblems in the algorithm.

To construct the restricted subproblems, we need to determine the set of promising nodes N^* and the corresponding set of promising paths Q^* . In the algorithm, the set of promising nodes can be obtained by

solving the relaxation model of the original problem [ASLP] with the relaxed constraints of binary variables. Then, we define $N^* = \{i \in N | x_i^* > 0\}$, and $Q^* = \{q \in Q | N_q \cap N^* \neq \emptyset\}$. Figure 3a illustrates how to build N^* . To improve the computation time for solving the restricted subproblem, we fixed a number of the best promising station locations in N^* by adding constraints $x_j = 1, \forall j \in N^{*1}$ (Figure 3b). To determine set N^{*1} , we sorted node $i \in N^*$ by two following criteria (with descending order): solution value x_i^* obtained by solving the relaxation model, and the traffic volume through node i . Then, we choose $|N^{*1}|$ the best promising candidate nodes in N^* to be located.

The restricted subproblems are defined by

[R-ASLP]

$$\max z_1 = \sum_{q \in Q^*} f_q y_q \quad (24)$$

$$\text{s.t.} \quad \sum_{i \in K_{jk}^q \cap N^*} x_i \geq y_q, \quad \forall q \in Q^*, \vec{a}_{jk} \in \vec{A}_q, \quad (25)$$

$$\sum_{i \in N^*} x_i = p, \quad (26)$$

$$x_i = 1, \quad \forall i \in N^{*1}, \quad (27)$$

$$x_i, y_q \in \{0, 1\}, \quad \forall i \in N^*, q \in Q^*. \quad (28)$$

The size of set Q^* may be reduced by removing paths that cannot be refueled by a set of restricted nodes. Then, the size of the restricted subproblems is improved significantly.

4.2 Parallel computing strategy

To integrate a parallel computing strategy into the algorithm, we need to generate a pool of restricted subproblems, and divide them into L disjoint sets of subproblems corresponding to the number of CPU cores. By allocating a disjoint set of subproblems to each CPU core and then solving them simultaneously, we are able to save significantly computation time.

A pool of restricted subproblems is constructed on k -exchange neighborhood structure. In particular, we replace k less promising nodes in N^* by k other nodes in N^0 to generate a new subproblem. The neighborhood structure for the alternative-fuel station location problem can be illustrated in Figure 4. From the preliminary experiment, we see that 2-exchange neighborhood can produce efficiently the restricted problems, as considering balance between solution quality and computation time. In this work, we thus performed 2-exchange to generate a pool of the restricted subproblems for the parallel computing strategy of the proposed algorithm.

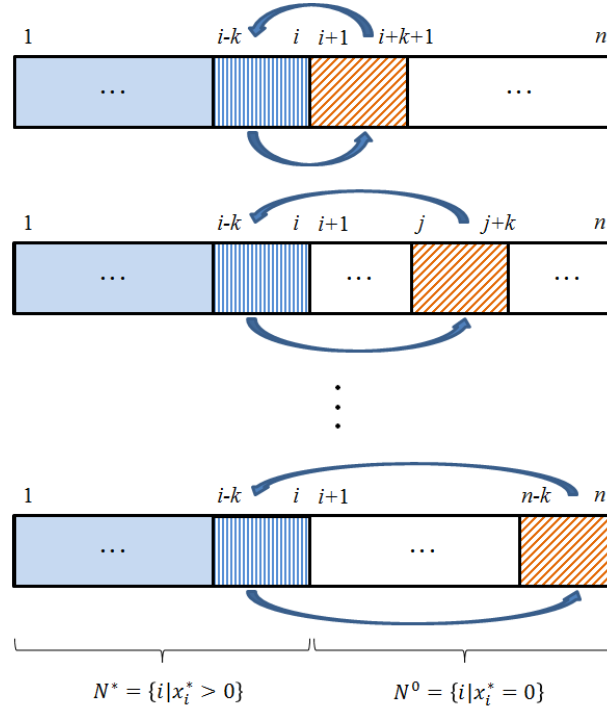


Figure 4: k -exchange neighborhood structure to construct the restricted subproblems.

4.3 Framework of the proposed algorithm

A framework of the proposed algorithm can be described as follows.

Algorithm

-
- Step 1:** Solve the original problem [ASLP] relaxed on the constraints of binary variables;
Denote (x^*, y^*) and z_0^* be the optimal solution and the objective value, respectively;
- Step 2:** **If** (x^*, y^*) are integer values, **then** stop and output the optimal solution (x^*, y^*) ;
Else set upper bound $UB = z_0^*$, and update the set of promising nodes $N^* = \{i \in N | x_i^* > 0\}$
and the set of remaining nodes $N^0 = \{i \in N | x_i^* = 0\}$;
- Step 3:** Determine the corresponding set of restricted paths Q^* , defined by N^* ;
Solve the restricted subproblem [R-ASLP], update the best solution and lower bound $LB = z_1^*$;
- Step 4:** **If** $(UB - LB < \epsilon)$, **then** stop and output the best solution found;
Else go to next step;
- Step 5:** Solve the restricted subproblems [R-ASLP] with a parallel computing strategy as follows
Do the following procedure until one of termination conditions is satisfied:
- (i) Construct sets of restricted nodes $N_1^*, N_2^*, \dots, N_L^*$ by exchanging k nodes between N^* and N^0
 - (ii) Assign each set of restricted nodes into each CPU core and solve [R-ASLP];
 - (iii) Update the best solution and LB , and check termination conditions:
 - + $UB - LB < \epsilon$
 - + All the restricted subproblems solved
 - + Reach the number of restricted subproblems given
-

In the algorithm, we used three termination conditions:

- The algorithm stops if we obtain $UB - LB < \epsilon$.
- The algorithm terminates if all the restricted subproblems have been considered (in which generating the restricted subproblems is based on k -exchange neighborhood structure).
- The algorithm stops if the number of restricted subproblems given is reached.

5 Experimental results

In this section, we investigate the computational efficacy of solving the FRLM with the heuristic algorithm proposed. We evaluate the performance of the heuristic algorithm on three well-known datasets from the literature and one randomly generated dataset, and then compare the obtained results with the optimal solutions from CPLEX solver as well as other heuristic algorithms.

The models and the proposed algorithm were implemented in Visual C++ and run on the same Microsoft Windows 7 Enterprise PC with an Intel Core i7-4810MQ processor (2.80 GHz per chip) and 16 GB of RAM. The models were built and solved using the IBM ILOG CPLEX version 12.4 callable library. Based on some preliminary experiments, parameter values for the proposed algorithm were chosen as follows: the number of the best promising station locations fixed $|N^{*1}| = \lceil p/5 \rceil$; 2-exchange neighborhood used; $\epsilon = 0.01$; the number of restricted subproblems given was set to 16 and 32 for $p \leq 15$ and $p > 15$, respectively; and the number of parallel computing threads was 8 (the number of cores on the Intel Core i7 processor). In addition, to reduce computation time for solving the restricted subproblems, we added the constraint of setting a cut-off value to the objective function. The cut-off value is retrieved from LB .

5.1 Datasets and parameters

The computational experiments were run on three well-known benchmark datasets and one randomly generated dataset of the FRLM, taken from the literature.

- Hodgson dataset (Hodgson, 1990): This is a 25-node alternative-fuel station location network. The flow volumes f_q in the 25×25 origin-destination matrix are estimated using a gravity model. The flows are then assigned to their shortest paths. The candidate sites are limited to the 25 nodes of the network. The network has 300 origin-destination pairs.
- Florida dataset (Kuby et al., 2009): This is a Florida state highway network consisting of 302 nodes (i.e., junctions) and 495 arcs. Each of nodes serves as a candidate site. Of the 302 candidate sites, there are 74 origin-destination nodes for trips. Since the return trip is assumed to be refuelable, the network of 74 origin-destination nodes only requires 2,701 unique origin-destination pairs. The road network includes all inter-state, toll, U.S. highways and other important state highways. Florida's inter-city volumes were based on a spatial interaction model, but short intra-zonal flows are excluded.
- California dataset (Arslan et al., 2014): This is a California state road network consisting of 339 nodes and 617 arcs. We consider all 1,167 possible origin-destination pairings of the urban population centers in California, in which their population is more than 50,000 and their distance is not closer than 30 kilometers. Volume of the flow on each pair is calculated using the gravity model of Hodgson (1990).

- Capar-Kuby dataset: A set of instances are randomly generated by Capar and Kuby (2012). In the experiment, only the instances with large network size (e.g., $n = 500, 750$ and $1,000$ nodes) are considered to test our algorithm. For each instance, m out of n are randomly chosen to be origin-destination nodes, and then $m(m-1)/2$ origin-destination pairs are determined to build the shortest paths. Here, we test our algorithm on $m = 40$ and 80 nodes (i.e., 780 and $3,160$ paths respectively). The traffic flow for each pair is generated using the gravity model of Fotheringham and O’Kelly (1989).

For the evaluation of computational experiments, a set of scenarios are generated by changing the range of vehicles R and the number of stations to be located p . Range of vehicles $R = 4, 8$, and 12 are used for Hodgson network; $60, 100$, and 200 miles are used for Florida network; $100, 150$ and 200 kilometers are used for California network; and $250, 500$ and $1,000$ are used for Capar-Kuby network. Hodgson, Florida and California networks are investigated with $p = \{5; 10; 15; 20; 25\}$, while $p = \{5; 10; 15; 20; 25; 30; 35; 40; 45; 50\}$ is for Capar-Kuby network.

5.2 Computational results

We compare the performance of the proposed heuristic algorithm (i.e., HA) with genetic algorithm (LKGenA) and greedy algorithm (LKGreA) developed by Lim and Kuby (2010). In addition, we evaluate our solutions over the optimal solutions obtained by CPLEX solver with 8 threads mode setting (for the formulations of Capar and Kuby (2012), Capar et al. (2013) and MirHassani and Ebrazi (2013), denoted by CK, CKLT and ME respectively) on four benchmark datasets mentioned. Tables 2, 3, 4 and 5 show the comparison results with the objective function measured by trips (%) on Hodgson, Florida, California and Capar-Kuby instances, respectively. In the tables, $Opt.$ represents the optimal objective value, Δ is the percentage gap between the objective value found ($Best$) and the optimal objective value (see Equation (29)), $\bar{\Delta}$ is the average percentage gap, and the computational time is measured by seconds.

$$\Delta = \frac{(Opt. - Best)}{Opt.} \times 100. \quad (29)$$

From Table 2, we can see that our algorithm obtained the optimal solutions for all Hodgson instances with less average CPU time than CPLEX solver on the formulations CK, CKLT and ME. Since the CPU time of the compared algorithms (i.e., LKGenA and LKGreA) was not reported in Lim and Kuby (2010), we do not present them in this paper. In terms of solution quality, our algorithm outperforms LKGreA in four cases, while there is no difference as compared with LKGenA.

Table 2: Comparison for the formulations and the algorithms on Hodgson instances.

R	p	$Opt.$	CK		CKLT		ME		LKGenA	LKGreA	HA	
			Δ	Time (s)	Δ	Time (s)	Δ	Time (s)	Δ	Δ	Δ	Time (s)
4	5	25.88	0	0.24	0	0.12	0	0.13	0	0	0	0.12
	10	50.72	0	0.20	0	0.14	0	0.12	0	10.75	0	0.13
	15	62.64	0	0.18	0	0.08	0	0.10	0	0	0	0.05
	20	68.58	0	0.18	0	0.08	0	0.10	0	0	0	0.05
	25	69.14	0	0.16	0	0.10	0	0.10	0	0	0	0.05
8	5	58.56	0	0.28	0	0.14	0	0.14	0	0.49	0	0.15
	10	82.81	0	0.32	0	0.18	0	0.18	0	0.93	0	0.26
	15	97.24	0	0.28	0	0.14	0	0.15	0	0	0	0.27
	20	98.33	0	0.20	0	0.12	0	0.11	0	0.15	0	0.05
	25	98.33	0	0.18	0	0.10	0	0.10	0	0	0	0.06
12	5	61.61	0	0.84	0	0.16	0	0.44	0	0	0	0.21
	10	93.99	0	0.30	0	0.16	0	0.16	0	0	0	0.07
	15	99.85	0	0.24	0	0.14	0	0.13	0	0	0	0.07
	20	100.00	0	0.28	0	0.14	0	0.14	0	0	0	0.07
	25	100.00	0	0.20	0	0.12	0	0.11	0	0	0	0.07
Average			0	0.27	0	0.13	0	0.15	0	0.82	0	0.11

Table 3 continues to show that our algorithm can find the optimal solutions for all Florida instances with less average CPU time than CK, CKLT and ME. Especially, the average CPU time of CK and ME is **about 8 and 4 times slower** than our CPU time, respectively. Also, our algorithm outperforms the three formulations in terms of CPU time on all Florida instances. As compared with LKGenA and LKGreA, our algorithm even outperforms significantly these algorithms in terms of solution quality and CPU time. For example, we only need 129.02 seconds to find the optimal solution for Florida instance ($R = 100, p = 25$), while LKGenA and LKGreA take 10,646 seconds (approximate 3 hours) and 1,800 seconds (approximate a half hour) respectively but do not find the optimal solution. In Table 3, we denoted “-” for the instances which were not solved and reported by LKGenA and LKGreA.

The efficiency of our algorithm, as compared with the formulations CK, CKLT and ME, are more clearly demonstrated in solving other large-sized instances. Tables 4 and 5 show that our algorithm can obtain the optimal solutions on all California and Capar-Kuby instances with less computation time. In particular, the proposed algorithm improved the average CPU time about 39-86 times than CK, twice than CKLT and 35-65 times than ME as solving the instances.

Table 3: Comparison of the formulations and the algorithms on Florida instances.

R	p	$Opt.$	CK		CKLT		ME		LKGenA		LKGreA		HA	
			Δ	Time (s)	Δ	Time (s)	Δ	Time (s)	Δ	Time (s)	Δ	Time (s)	Δ	Time (s)
60	5	56.12	0	56.74	0	14.70	0	24.31	-	-	-	-	0	11.04
	10	77.17	0	74.50	0	14.25	0	38.52	-	-	-	-	0	12.56
	15	83.55	0	132.00	0	21.75	0	64.10	-	-	-	-	0	20.15
	20	87.60	0	234.58	0	21.21	0	127.19	-	-	-	-	0	20.54
	25	91.39	0	197.18	0	31.80	0	95.69	-	-	-	-	0	30.55
100	5	64.89	0	148.08	0	34.02	0	76.34	0.71	182	0	33	0	27.76
	10	81.22	0	209.84	0	29.04	0	101.22	0.41	1,352	0.43	174	0	28.58
	15	87.84	0	340.82	0	32.13	0	165.31	0.16	2,412	0.50	456	0	30.25
	20	92.44	0	491.62	0	75.03	0	250.89	0.10	5,800	0.31	1,132	0	73.56
	25	95.37	0	999.12	0	138.42	0	512.36	0.10	10,646	0.38	1,800	0	129.02
200	5	69.19	0	240.36	0	41.85	0	123.58	-	-	-	-	0	34.44
	10	86.03	0	342.66	0	48.54	0	177.23	-	-	-	-	0	39.49
	15	92.92	0	533.72	0	57.33	0	260.76	-	-	-	-	0	52.71
	20	96.65	0	841.50	0	104.07	0	415.39	-	-	-	-	0	100.46
	25	98.17	0	1,172.02	0	174.72	0	592.12	-	-	-	-	0	150.16
Average			0	400.98	0	55.92	0	201.67					0	50.75

Table 4: Comparison of the formulations and our algorithm on California instances.

R	p	$Opt.$	CK		CKLT		ME		HA	
			Δ	Time (s)	Δ	Time (s)	Δ	Time (s)	Δ	Time (s)
100	5	67.08	0	25.15	0	2.46	0	23.23	0	1.83
	10	87.98	0	28.63	0	4.10	0	26.33	0	2.29
	15	95.01	0	26.55	0	3.89	0	23.68	0	2.03
	20	98.41	0	197.50	0	7.25	0	75.47	0	4.28
	25	99.79	0	442.92	0	17.99	0	443.26	0	9.80
150	5	79.94	0	30.56	0	6.45	0	27.39	0	3.12
	10	92.98	0	76.70	0	9.56	0	62.98	0	4.96
	15	98.35	0	205.46	0	12.05	0	108.19	0	5.69
	20	99.89	0	336.50	0	11.36	0	268.58	0	5.52
	25	100.00	0	1,979.90	0	5.97	0	1,225.62	0	3.87
200	5	85.18	0	31.18	0	11.23	0	32.33	0	4.21
	10	95.64	0	69.64	0	13.54	0	64.86	0	4.96
	15	99.21	0	195.32	0	18.51	0	116.57	0	5.74
	20	99.97	0	630.88	0	19.79	0	830.78	0	6.07
	25	100.00	0	1,638.47	0	15.20	0	1,187.77	0	4.65
Average			0	394.36	0	10.62	0	301.14	0	4.60

Table 5: Comparison of the formulations and our algorithm on Capar-Kuby instances.

n	m	R	CK		CKLT		ME		HA	
			$\bar{\Delta}$	Time (s)	$\bar{\Delta}$	Time (s)	$\bar{\Delta}$	Time (s)	$\bar{\Delta}$	Time (s)
500	40	250	0	400.51	0	7.07	0	367.13	0	3.69
		500	0	420.26	0	8.25	0	435.18	0	3.82
		1,000	0	859.58	0	15.89	0	829.65	0	8.60
	80	250	0	1,429.43	0	16.36	0	1,309.33	0	8.36
		500	0	2,298.23	0	27.60	0	2,033.15	0	13.80
		1,000	0	5,405.47	0	65.41	0	5,325.93	0	31.25
750	40	250	0	1,999.48	0	81.95	0	2,053.62	0	49.02
		500	0	2,579.40	0	118.20	0	2,374.37	0	62.04
		1,000	0	3,760.46	0	169.58	0	3,310.76	0	87.66
	80	250	0	2,015.79	0	104.13	0	2,084.84	0	58.32
		500	0	3,716.89	0	197.32	0	3,629.60	0	105.69
		1,000	0	8,540.25	0	385.59	0	8,350.28	0	244.17
1,000	40	250	0	1,661.72	0	201.24	0	1,356.83	0	83.72
		500	0	3,318.69	0	326.70	0	2,708.65	0	159.67
		1,000	0	5,524.13	0	447.29	0	3,967.22	0	227.97
	80	250	0	3,789.75	0	207.68	0	2,914.45	0	97.51
		500	0	6,183.12	0	315.33	0	5,829.33	0	156.54
		1,000	0	10,327.50	0	489.45	0	8,715.05	0	261.77
Average			0	3,568.37	0	176.95	0	3,199.74	0	92.42

As parallelisation is an integral part of our solution method we do not present here results without parallel computing. We note though that our heuristic is approximately twice as fast in parallel mode than using just a single thread.

6 Conclusions and future work

The design of a heuristic algorithm for the alternative-fuel station location problem is an important issue that has not received appropriate attention in the research. In the paper, we thus develop an efficient heuristic algorithm to locate optimal fueling stations for the maximisation of round-trip traffic volume. The algorithm is constructed on solving the sequence of restricted problems by a set of promising station candidates, and by a number of the best promising stations to be located. To determine the initial set of candidates we solve a relaxation model of the original problem with the constraints of integer variables relaxed, and then update the set in next iterations by performing 2-exchange between the set of promising candidates and the remaining station set. As solving the restricted problems, we locate the best stations in the set of promising candidates to improve the computation time of the algorithm. Besides that, we use a parallel computing strategy to simultaneously solve a number of restricted problems with less computation effort for large-sized instances. Experimental results show that the proposed algorithm can obtain the optimal solutions with less computation time (compared with CPLEX solver), and outperforms the other compared algorithms (i.e., genetic algorithm and greedy algorithm) with respect to solution quality as well as computation time.

From the successful results obtained, we can extend the heuristic algorithm to handle other interesting alternative-fuel station location problems, such as:

- The capacitated FRLM (Upchurch et al., 2009, Zockaie et al., 2016),
- The FRLM with deviation (Kim and Kuby, 2012, Kim and Kuby, 2013, Zockaie et al., 2016),
- The **budget-constrained** FRLM (Badri-Koochi and Tavakkoli-Moghaddam, 2012, Capar et al. (2013), MirHassani and Ebrazi, 2013),
- The multi-period FRLM (Chung and Kwon, 2015, Miralinaghi et al., 2016).

It would be very interesting to apply our algorithm to practical applications, which we believe may arise in the near future, especially in the light of the recent EU directive on the establishment of a Europe-wide alternative fuel infrastructure (European Commission, 2014). In this respect, the reader is referred to the recent application study by Kuby et al. (2017), focusing on natural gas refuelling stations in the E.U. Another possible application would be for the location of alternative fuel stations for the railways. While the algorithms presented in the literature could be just as applicable to rail transport as to automobiles, most papers tackle the FRLM in the context of automobile refuelling stations. Yet, as Kuby and Lim (2005) has already pointed out, there is much better origin-destination flow data available for railways, making this mode of transport an ideal field of applying FRLM models.

Acknowledgment

This work has been funded by a grant from the Spanish Ministry of Economy and Competitiveness (ECO2011-24927).

References

- M Albareda-Sambola, E Fernandez, and S Nickel. Multiperiod location-routing with decoupled time scales. *European Journal of Operational Research*, 217(2):248–258, 2012.
- E Angelelli, R Mansini, and MG Speranza. Kernel search: A general heuristic for the multi-dimensional knapsack problem. *Computers and Operations Research*, 37(11):2017–2026, 2010.
- E Angelelli, R Mansini, and MG Speranza. Kernel search: A new heuristic framework for portfolio selection. *Computational Optimization and Applications*, 51(1):345–361, 2012.
- O Arslan, B Yildiz, and OE Karasan. Impacts of battery characteristics, driver preferences and road network features on travel costs of a plug-in hybrid electric vehicle (phev) for long-distance trips. *Energy Policy*, 74(0):168–178, 2014.
- B Badri-Koochi and R Tavakkoli-Moghaddam. Determining optimal number and locations of alternative-fuel stations with a multi-criteria approach. *Proceedings of the 8th International Industrial Engineering Conference*, 2012.
- RH Ballou. Dynamic warehouse location analysis. *Journal of Marketing Research*, 5(3):271–276, 1968.
- O Berman. Deterministic flow-demand location problems. *Journal of the Operational Research Society*, 48(1):75–81, 1997.

- O Berman, MJ Hodgson, and D Krass. *Flow intercepting models*. In: Z. Drezner (Ed.), *Facility Location: A Survey of Applications and Methods*, pages 389-426. Springer, New York, 1995.
- SF Bhatti, MK Lim, and HY Mak. Alternative fuel station location model with demand learning. *Annals of Operations Research*, 230(1):105–127, 2015.
- I Capar and M Kuby. An efficient formulation of the flow refueling location model for alternative-fuel stations. *IIE Transactions*, 44(8):622–636, 2012.
- I Capar, M Kuby, VJ Leon, and YJ Tsai. An arc cover-path-cover formulation and strategic analysis of alternative-fuel station locations. *European Journal of Operational Research*, 227(1):142–151, 2013.
- SH Chung and C Kwon. Multi-period planning for electric car charging station locations: a case of Korean expressways. *European Journal of Operational Research*, 242(2):677–687, 2015.
- European Commission. http://europa.eu/rapid/press-release_MEMO-13-24_en.htm, 2013. Accessed 01/12/2015.
- European Commission. http://europa.eu/rapid/press-release_IP-14-1053_en.htm, 2014. Accessed 01/12/2015.
- M Drexler and M Schneider. A survey of variants and extensions of the location-routing problem. *European Journal of Operational Research*, 241(2):283–308, 2015.
- AS Fotheringham and ME O’Kelly. *Spatial Interaction Models: Formulations and Applications*. Kluwer Academic Publishers, 1989.
- M Ghamami, A Zockaie, and YM Nie. A general corridor model for designing plug-in electric vehicle charging infrastructure to support intercity travel. *Transportation Research Part C: Emerging Technologies*, 68:389–402, 2016.
- G Guastaroba and MG Speranza. Kernel search: An application to the index tracking problem. *European Journal of Operational Research*, 217(1):54–68, 2012a.
- G Guastaroba and MG Speranza. Kernel search for the capacitated facility location problem. *Journal of Heuristics*, 18(6):877–917, 2012b.
- G Guastaroba and MG Speranza. A heuristic for BILP problems: The single source capacitated facility location problem. *European Journal of Operational Research*, 238(2):438–450, 2014.
- MJ Hodgson. The location of public facilities intermediate to the journey to work. *European Journal of Operational Research*, 6(2):199–204, 1981.
- MJ Hodgson. A flow-capturing location-allocation model. *Geographical Analysis*, 22(3):270–279, 1990.
- MJ Hodgson and O Berman. A billboard location model. *Geographical and Environmental Modeling*, 1(1): 25–45, 1997.
- MJ Hodgson and KE Rosing. A network location-allocation model trading off flow capturing and p-median objectives. *Annals of Operations Research*, 40(1):247–260, 1992.

- MJ Hodgson, KE Rosing, and J Zhang. Locating vehicle inspection stations to protect a transportation network. *Geographical Analysis*, 28(4):299–314, 1996.
- M Hosseini and SA MirHassani. Refueling-station location problem under uncertainty. *Transportation Research Part E*, 84:101–116, 2015.
- Y Huang, S Li, and ZS Qian. Optimal deployment of alternative fueling stations on transportation networks considering deviation paths. *Networks and Spatial Economics*, 15(1):183–204, 2015.
- J Jung, JYJ Chow, R Jayakrishnan, and JY Park. Stochastic dynamic itinerary interception refueling location problem with queue delay for electric taxi charging stations. *Transportation Science Part C*, 40:123–142, 2014.
- JE Kang and W Recker. Strategic hydrogen refueling station locations with scheduling and routing considerations of individual vehicles. *Transportation Science*, 49(4):767–783, 2015.
- S Kelley and M Kuby. On the way or around the corner? Observed refueling choices of alternative-fuel drivers in Southern California. *Journal of Transport Geography*, 33:258–267, 2013.
- JG Kim and M Kuby. The deviation-flow refueling location model for optimizing a network of refueling stations. *International Journal of Hydrogen Energy*, 37(6):5406–5420, 2012.
- JG Kim and M Kuby. A network transformation heuristic approach for the deviation flow refueling location model. *Computers and Operations Research*, 40(4):1122–1131, 2013.
- M Kuby and S Lim. The flow-refueling location problem for alternative-fuel vehicles. *Socio-Economic Planning Sciences*, 39(2):125–145, 2005.
- M Kuby, L Lines, R Schultz, Z Xie, JG Kim, and S Lim. Optimization of hydrogen stations in Florida using the flow-refueling location model. *International Journal of Hydrogen Energy*, 34(15):6045–6064, 2009.
- M Kuby, I Capar, and JG Kim. Efficient and equitable transnational infrastructure planning for natural gas trucking in the european union. *European Journal of Operational Research*, 257(3):979–991, 2017.
- S Lim and M Kuby. Heuristic algorithms for siting alternative-fuel stations using the flow-refueling location model. *European Journal of Operational Research*, 204(1):51–61, 2010.
- M Miralinaghi, BB Keskin, Y Lou, and AM Roshandeh. Capacitated refueling station location problem with traffic deviations over multiple time periods. *Networks and Spatial Economics*, 2016. URL doi: 10.1007/s11067-016-9320-3.
- SA MirHassani and R Ebrazi. A flexible reformulation of the refueling station location problem. *Transportation Science*, 47(4):617–628, 2013.
- G Nagy and S Salhi. Location-routing: Issues, models and methods. *European Journal of Operational Research*, 177(2):649–672, 2007.
- JM Nambiar, LF Gelders, and LN Van Wassenhove. Plant location and vehicle routing in the Malaysian rubber smallholder sector: a case study. *European Journal of Operational Research*, 38(1):14–26, 1989.
- Y Nie and M Ghamami. A corridor-centric approach to planning electric vehicle charging infrastructure. *Transportation Research Part B*, 57:172–190, 2013.

- C Prodhon and C Prins. A survey of recent research on location-routing problems. *European Journal of Operational Research*, 238(1):1–17, 2014.
- J Rosenhead, M Elton, and SK Gupta. Robustness and optimality as criteria for strategic decisions. *Journal of the Operational Research Society*, 23(4):413–431, 1972.
- S Salhi and G Nagy. Consistency and robustness in location-routing. *Studies in Locational Analysis*, 13:3–19, 1999.
- SP Singh and RRRK Sharma. Two level modified simulated annealing based approach for solving facility layout problem. *International Journal of Production Research*, 46(13):3563–3582, 2008.
- TH Tran and KM Ng. Two-level water flow algorithm for vehicle routing problems. *Proceedings of the 5th Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA)*, Arizona, US, 2011.
- C Upchurch, M Kuby, and S Lim. A model for location of capacitated alternative-fuel stations. *Geographical Analysis*, 41(1):85–106, 2009.
- YW Wang and CC Lin. Locating road-vehicle refueling stations. *Transportation Research E*, 45(5):821–829, 2009.
- YW Wang and CR Wang. Locating passenger vehicle refueling stations. *Transportation Research E*, 46(5):791–801, 2010.
- M Wen, G Laporte, OBG Madsen, AV Norrelund, and A Olsen. Locating replenishment stations for electric vehicles: application to Danish traffic data. *Journal of the Operational Research Society*, 65(10):1555–1561, 2014.
- J Yang and H Sun. Battery swap station location-routing problem with capacitated electric vehicles. *Computers and Operations Research*, 55:217–232, 2015.
- B Yildiz, O Arslan, and OE Karasan. A branch and price approach for routing and refueling station location model. *European Journal of Operational Research*, 248(3):815–826, 2016.
- A Zockaie, HZ Aashtiani, M Ghamami, and YM Nie. Solving detour-based fuel stations location problems. *Computer-Aided Civil and Infrastructure Engineering*, 31(2):132–144, 2016.