A Network Transformation Heuristic Approach for the Deviation Flow Refueling Location Model

Jong-Geun Kim\textsuperscript{a*}, Michael Kuby\textsuperscript{b}

\textsuperscript{a}Department of Geography Education, Seoul National University, Seoul, 151-742, South Korea
\textsuperscript{b}School of Geographical Sciences and Urban Planning, Arizona State University, Tempe, AZ, 85287-5302, USA

\* Corresponding author. Tel.: +82 2 880 7718; fax: +82 2 882 9873.
E-mail address: jgkim25@snu.ac.kr (J. Kim), mikekuby@asu.edu (M. Kuby).

ABSTRACT

In the early stages of development, alternative-fuel vehicles will tend to have shorter driving ranges than conventional vehicles, and the availability of stations will be limited. Given these conditions, it is important to consider the willingness of drivers to deviate to some extent from their shortest paths in order to refuel their vehicles and complete their trips. Previously, we proposed the Deviation-Flow Refueling Location Model (DFRLM) for locating a given number of refueling facilities to maximize the total alternative-fuel vehicle flows that can be refueled by drivers traveling on or deviating from their shortest paths. On a real-world problem, however, the large number of possible deviations from each path and of combinations of facilities that can cover each path would make it extremely difficult to generate and solve the mixed-integer formulation. This paper develops heuristic algorithms for the DFRLM that overcome this difficulty through network transformation. Specifically, a greedy heuristic constructs and edits an artificial feasible network in which each node represents a station, origin, or destination, and each arc represents a feasible path between two nodes given the assumed driving range of vehicles. At each step of the greedy and greedy-substitution algorithms, the feasible network is edited and a shortest path algorithm is run that determines whether each origin-destination round trip can be completed. This method allows any possible detour to be taken (up to some user-defined maximum) while also ensuring that drivers take the smallest possible detour. Computational experiments on a simple network and a real-world network for Florida show the heuristics to be efficient in solving the problems. Comparisons between the results of the DFRLM and the FRLM indicate that taking driver deviations into account in the model can have a significant effect on the locations chosen and demand covered.

Keywords: Path deviation, Detour, Heuristics, Location, Network design, Refueling station, Energy
1. Introduction

This paper provides a new network transformation method for solving the problem of locating refueling facilities for alternative fuel vehicles (AFVs) when the drivers are assumed to deviate from their shortest paths if required. Refueling stations are a type of “discretionary service facility” [1, 2], which customers visit as part of pre-planned trips rather than make a special trip to the facilities. Designing a network of refueling facilities for AFVs requires more considerations than for conventional gasoline stations. The short driving range of AFVs implies that AFV drivers may need to refuel multiple times in order to finish their trips [3] and the scarcity of stations implies that they will sometimes need to deviate from their usual routes to refuel their vehicles [4, 5].

The Deviation-Flow Refueling Location Model (DFRLM) developed in [5] locates \( p \) refueling facilities to maximize the total demand that can be refueled or recharged. Demand consists of flows—trips (vehicles) that are traveling between O-D pairs on paths through a network. Whether a flow can be refueled depends on whether the AFV is able to stop at stations that are located on or near the drivers’ pre-planned paths and whether those stations are adequately spaced in relation to the driving range of the AFVs so that the round trip can be completed. Kim and Kuby [5] formulated a mixed-integer linear programming (MILP) model for the DFRLM and optimally solved small problem instances. However, solving a realistically sized DFRLM problem using the MILP formulation is not advisable. The reason is that the proposed MILP formulation of the DFRLM in [5] requires pre-generation of 1) every possible path for each O-D pair and 2) every possible combination of facilities that enables the trip to be completed without the vehicle running out of fuel on each of those paths. The large number of possible deviations from each path and of combinations of facilities that can cover each path would make it extremely difficult to generate and solve the mixed-integer formulation. In this paper, we are primarily concerned with development of a heuristic solution approach for solving DFRLM problems.

Heuristic solution methods have been successfully applied to solve the related problems such as the Flow Intercepting Location Model with deviation cases (FILM-D) [4] and Flow Refueling Location Model (FRLM) [3]. We, however, are not aware of any heuristic approach for solving the DFRLM. One of the common features of the previous heuristic algorithms for the related models is that they eliminated the need of pre-generating a large amount of data: all
deviation paths and all feasible node combinations. For instance, the heuristic algorithms for the FRLM developed in Lim and Kuby [6] do not pre-generate every possible combination of facilities. Likewise, a greedy heuristic for the FILM-D suggested in [4] does not generate deviation paths until a set of candidate facilities is selected for evaluation, and in doing so it ensures that the minimum number of deviation paths is generated dynamically.

There is a difference, however, in the deviation paths considered by the DFRLM and by the FILM-D, which is essential to understand for successful development of a heuristic DFRLM solution algorithm. One of the ways to construct a deviation path for FILM-D [4] is to choose one node from the network that is at not on the shortest path from the origin to the destination and generate a shortest path from the origin to the chosen node and from there to the destination. We can think of these paths as “one-step” deviation paths; they are consistent with the assumption of flow capturing/intercepting models in that a single facility anywhere on a path is all that is needed to serve the flows on the path. When a flow-based model is used for refueling, however, and the vehicles have a limited driving range, a one-step deviation path may not enable drivers to complete their trips if more than one refueling is needed. Therefore, any algorithm for solving DFRLM problems needs to include “multi-step” deviation paths that are generated from the origin to one or more candidate stations on or off the original shortest path and then to the final destination.

Review of the heuristics in [4] and [6] suggest that a desirable approach for solving the DFRLM is to generate as few paths as possible while ensuring that travelling on each of the paths is feasible in terms of fuel availability given the spatial configuration of candidate facilities and the AFV’s driving range after each refueling. The method should also allow any possible detour to be taken (up to some user-defined maximum) while ensuring that drivers take the smallest possible detour. In order to achieve these goals efficiently, we developed a network transformation method that constructs an artificial feasible network depicting functional linkages among candidate facility locations and O-D nodes. This paper reviews the related literature (Section 2); explains the transformation method and heuristic algorithm (Section 3); examines the performance of the algorithm and illustrates how deviations in the model affect the locations selected for stations (Section 4); and provides conclusions and suggestions for future research (Section 5).
2. Literature Review

2.1 Flow refueling and deviation-flow refueling

Most classical location models involve providing service to point-based demands. Recently there has been increasing research interest in path-based demand that is expressed by flows travelling on paths between origin-destination (O–D) pairs in a traffic network. The flow-intercepting location model (FILM) that was introduced independently in Hodgson [1] and Berman et al. [2] sites facilities within a transportation network and explicitly considers the flow on shortest paths over the network arcs. Refueling stations [3], convenience stores, and automated teller machines, vehicle inspection stations [7, 8], and billboards [9] are examples of flow-dependent facilities. Many extensions of the basic FILM have been introduced [10-14] to incorporate classical p-median or set-covering types of constraints or to consider competition among facilities. Given that the deterministic FILM requires knowledge of the flows on all the paths in the network, Berman, Krass, and Xu [15] re-formulated the FILM to use more readily obtainable data on the proportion of flows originating from each node and the probability of turns estimated at each node.

Kuby and Lim [3] extended the FILM to locate a given number of refueling stations. The key new element in their Flow Refueling Location Model (FRLM) is the vehicle driving range, which implies that one facility anywhere on the path might not be sufficient for refueling a trip on a given shortest path—a combination of facilities may be needed. Whereas the FILM counts a flow as captured if a facility is located anywhere along the path of the flow because one stop will satisfy consumers’ need, the FRLM regards a flow as refueled only when a satisfactory number of facilities (stations) are spaced properly along the path because consumers on the path may need their fuel tank refilled or battery recharged or swapped out multiple times. To improve the FRLM’s solution quality for a given network, Kuby and Lim [16] proposed methods to add candidate locations along arcs. Upchurch, Kuby, and Lim [17] extended the FRLM to consider capacity of facilities. Given that solving a FRLM problem instance to optimality is computationally challenging, heuristic algorithms [6] and a new formulation of the model [18] were also proposed. The FRLM was used to provide strategic station locations for the state of Florida at two different scales of analysis: metropolitan Orlando and statewide [19].

While the FRLM takes an all-or-nothing approach to coverage of flows, the Deviation Flow Refueling Location Model (DFRLM) of Kim and Kuby [5] considers consumers’ necessary
deviations off the shortest paths, and incorporates partial coverage of flows. They assumed that: (i) drivers take the shortest or least-cost deviation path to their required refueling stations and then to their final destination; (ii) the fraction of customers deviating to a facility can be specified as a decreasing function of deviation distance; (iii) there is an upper limit of deviation distance that drivers can tolerate, even though a longer path might be refuelable; (iv) facility location is limited to network nodes; (v) fuel consumption is strictly a function of distance; and (vi) vehicles begin each trip with the tank filled halfway. The latter two assumptions ensure that the round trip will be feasible and repeatable, because if an AFV can begin with a half tank and make it to the first station along the way, then it can refuel at that same station on the way back and return to the origin with a half tank, and thus be able to make the same trip (or any other similarly feasible trip) from the same origin. They provided a framework to explore the effects of five different types of deviation penalty functions: linear, exponential, sigmoid, inverse, and a no-decay step function. A MILP formulation of the DFRLM was presented as follows:

Formulation of the DFRLM

Maximize \( \sum_q \sum_r f_q g_{qr} y_{qr} \)  

Subject to

\[ \sum_{r \in R_q} y_{qr} \leq 1 \quad \forall q \in Q \]  \hfill (2)

\[ \sum_{h \in H_q} v_h \geq y_{qr} \quad \forall r \in R_q, q \in Q \]  \hfill (3)

\[ x_k \geq v_h \quad \forall h \in H, k \in K_h \]  \hfill (4)

\[ \sum_{k \in K} x_k = p \]  \hfill (5)

\[ x_k, v_h, y_{qr} \in \{0,1\} \quad \forall k \in K, h \in H, q \in Q, r \in R_q \]  \hfill (6)

where:

Indices
- \( q \) = a particular O-D pair
- \( r \) = index of deviation paths
- \( k \) = a potential facility location
- \( h \) = index of combinations of facilities

Sets
- \( R \) = set of all deviations
- \( R_q \) = set of deviation paths \( r \) for O-D pair \( q \)
- \( Q \) = set of all O-D pairs
\( K \) = set of all potential facility locations  
\( K_h \) = set of facilities \( k \) that are in combination \( h \)  
\( H \) = set of all potential facility combinations  
\( H_{qr} \) = set of facility combinations \( h \) that can refuel deviation path \( r \) that is originated from O-D pair \( q \)

**Parameters**

\( p \) = the number of facilities to be located  
\( f_q \) = flow between O-D pair \( q \)  
\( g_{qr} \) = fraction of O-D pair \( q \) customers who would be willing to take deviation path \( r \) (that is, the penalty function value for deviation \( r \))

**Decision Variables**

\( x_k \) = 1 if there is a facility at location \( k \), 0 if not  
\( y_{qr} \) = 1 if path \( r \) is the least-deviating path for O-D pair \( q \) that can be refueled, 0 otherwise  
\( v_h \) = 1 if all facilities in combination \( h \) are open, 0 otherwise

The objective function (1) maximizes the total flow that can be refueled. The model provides a flexible framework for modeling the fraction of customers on O-D pair \( q \) who are willing to deviate from their shortest paths by the amount required on path \( r \), denoted as \( g_{qr} \). Let \( DD, d_q \), and \( g(DD) \) be the deviation distance, a reference distance, and the fraction of flows specified by function \( g \) and input \( DD \). Users can choose a distance decay function type from five available types: no decay, linear, exponential, inverse distance, and sigmoid. Each function was expressed in [5] as:

\[
g(DD)_{\text{No Decay}} = 1
\]

\[
g(DD)_{\text{Linear}} = 1 - \frac{DD}{\beta d_q}
\]

\[
g(DD)_{\text{Exponential}} = 1 - (ae^{\beta(DD-d_q)})
\]

\[
g(DD)_{\text{Inverse Distance}} = (ae^{-\beta \cdot DD})
\]

\[
g(DD)_{\text{Sigmoid}} = \frac{1}{1 + ae^{(\beta \cdot DD) - d_q}}
\]

By specifying parameters \((\alpha \text{ or } \beta)\), the shape of the function is determined. The reference distance, in combination with the other specified parameters, defines the bandwidth. A large reference distance implies that the spatial extent of distance decay is large, which will result in
smooth distance decay, whereas a small reference distance will result in a rapidly decreasing weighting. The reference distance can be specified as fixed, or it can be derived individually from each O-D path (i.e., shortest-path distance). When the shortest-path distance is specified, different impedance functions for each O-D path will be "adaptively" generated. If a fixed distance is used, on the other hand, the same distance decay model will be globally applied to all O-D pairs.

Constraints (2) limit the contribution to the objective function to at most one deviation path \( r \). There could be many deviation paths for an O-D pair but only the feasible path with the highest \( g_{qr} \) will be counted within the objective function. The set of paths \( r \) for O-D pair \( q \) includes the shortest path with deviation distance = 0 and \( g_{qr} = 1 \). Constraints (3) ensure that for any deviation path \( r \) to be refueled, at least one valid combination \( h \) has to be open. Determination of the eligible combinations is exogenous in that they are generated outside the model and depend on the network structure and vehicle range. An algorithm to generate the combination \( h \) for each path \( q \) and other considerations such as obtaining a tighter set \( H \) by removing superset sets are discussed in Kuby and Lim [3] and can also be applied to deviation paths \( r \). More specifically, the algorithm uses the driving range parameter to determine whether a particular combination \( h \) of stations is capable of refueling a vehicle on the round trip on path \( r \) without the vehicle running out of fuel. Constraints (4) ensure that all the facilities in combination \( h \) must be open before \( v_h \) is able to equal one. Constraints (5) specify the number of facilities to open, and (6) are the integrality constraints for the variables.

2.2 Solution approaches for solving FILM and FRLM problems

The FILM and extensions such as the FRLM are NP-hard [1, 2], and therefore heuristics have been proposed for solving large problem instances. A greedy heuristic for the basic FILM was originally proposed by Hodgson [1] and also studied in [2, 7, 8, 20]. Berman, Larson, and Fouska [2] proposed a heuristic and a branch-and-bound algorithm, while Gendreau, Laporte, and Parent [20] proposed ascent search and Tabu search to improve on the greedy solution by utilizing an iterative local search to transform the current solution to a set of neighboring solutions in the search space. Gzara and Erkut [21] applied Lagrangian relaxation for the FILM, while Selmic, Teodorovic, and Vukadinovic [22] proposed a bee colony optimization heuristic.
Lim and Kuby [6] developed three heuristic algorithms (greedy, greedy with substitution, and genetic) for the FRLM. Some heuristics focused on manipulation of flow data. A bi-level programming approach was proposed by Yang et al. [23] where the FILM was the upper-level problem and a stochastic traffic assignment problem was the lower level. Zeng, Castillo, and Hodgson [24] suggested a data manipulation framework to exploit the FILM’s property whereby nodes with heavy traffic flows are good candidates for the optimal solution. Their iterative procedure systematically removes O-D pairs with lower flow volumes; removes nodes with low passing flows; identifies nodes with higher probability of being included in the optimal solution; and then finally solves the problem to optimality when the size is small enough.

2.3 Needs and requirements of a new approach

The literature shows an absence of efficient heuristic solution algorithms for the DFRLM. The MILP formulation of the DFRLM in [5] requires pre-generation of (1) all possible deviation paths for each O-D pair using a $k$-shortest path algorithm, and (2) all possible combinations of facilities that can refuel each of those paths. These tasks, as well as solving the resultant MILP, require extensive computation and memory, and may not be applicable to real-world networks. On the other hand, a simple mixture of previous heuristic approaches suggested for FILM-D [4] and FRLM [6] would not work because the generation of every deviation path is impractical and the deviations that the DFRLM needs to consider are different from those needed for FILM-D. There are some important considerations to take into account before an efficient approach for the DFRLM is developed.

Firstly, one should decide, between path generation and flow estimation, which task should come first. The greedy heuristic for the FILM-D suggested in [4] is limited by the fact that it cannot handle all possible deviations but only special cases. Given the complexity of considering all possible paths among O-D pairs and the fact that DFRLM ultimately maximizes the coverage of flows, one approach in the literature with potential to avoid enumerating all possible deviation paths for the DFRLM is the use of turn probabilities, as suggested by Berman, Krass, and Xu [15] for a probabilistic FILM. In the case of refueling stations, however, the turn probabilities at nodes would have to be endogenous depending on the spatial configuration of the stations, as drivers turn as needed to navigate to the few available stations. Therefore, flows on
paths would need to be estimated after deviation paths are generated rather than from exogenous turn probabilities. Moreover the DFRLM requires knowledge of at least the shortest deviation path for each O-D pair that passes by the facilities, which cannot be obtained directly from turn probabilities.

A related question is then the number of paths to generate and when to generate them. Note that, as we discussed in the introduction, the deviation paths should be able to reach more than one facility off the shortest path. In addition to one-step deviation paths, the algorithm should also include multi-step deviation paths, the number of which increases exponentially not to mention the difficulty of generating them.

Another requisite for the deviation paths for the DFRLM is that drivers must be able to complete their round trips traveling on each of the paths given an AFV’s driving range per tank of fuel. This directs us to identify and focus on the nodes that are important in terms of refuelability: the origin, the destination, and the nodes with refueling stations. Based on the driving range, the driver can draw a service area that he or she can reach from the origin or after refueling the vehicle at a station. These service areas may form a service corridor if the stations are close enough to allow the driver to reach anywhere in the combined area by refueling at the stations. Eventually the driver would be interested in identifying a shortest path that falls in a service corridor that provides connection from his origin to destination.

Therefore, if O-D paths need to be explicitly generated prior to estimating flows, but it is too computationally burdensome to generate and solve for all possible deviation paths and all possible facility combinations, the most promising avenue for developing an efficient heuristic for solving the DFRLM would seem to include a procedure that dynamically generates shortest deviation paths given the origin, the destination, the locations of facilities, and vehicle’s driving range per refueling. That is the approach taken in this paper. In addition to solving the DFRLM, such a heuristic would also be able to solve the original FILM [1, 2], FILM-D [4], and the original FRLM [3] as simpler cases of the DFRLM.

3. Network Transformation Heuristic

We develop a network transformation approach to solve the DFRLM heuristically. As in previous greedy approaches [4, 6], additional facilities are sequentially located at nodes that add the greatest increment of previously unserved vehicle flows in the network, but the algorithms
have been modified to consider vehicle flows that deviate from the shortest paths subject to the
vehicle’s driving range. The major innovation of this approach is the introduction of a feasible
network, where feasible paths among the temporary locations and O-D nodes are represented as
edges. The creation and editing of the feasible network and solution of a shortest path algorithm
on the network is embedded within a classic greedy adding and substitution framework.

This modified procedure involves the following steps:

1. Initialize $p$, network, range, O-D matrix, and pre-existing facilities.
2. Add 1 to $p$ new facilities (outer loop)
   a. Test each possible candidate site and add the best node to the solution (inner loop).
      Four sub-steps build and test each temporary solution:
      i. Construct a temporary solution by adding a candidate node to the set of
         previously determined facility nodes.
      ii. Construct an artificial feasible network from the temporary solution.
      iii. Generate shortest deviation paths among all the O-D pairs on the feasible
           network and evaluate the objective function in terms of total vehicle flows
           that can be refueled.
      iv. Go to 2-b if substitution is desired. Select the best node and add it to the
           solution.
   b. Optional substitution procedure:
      i. Select a node in the solution (excluding pre-existing facilities) for possible
         substitution.
      ii. Replace the selected node with a node that is not selected:
         a) Repeat steps i-iii above.
         b) If this substitution improves the objective value, accept it.
         c) Continue substitutions until a user-specified number of
            substitutions has been reached or all possible one-for-one
            substitutions have been attempted.
3. Stop when the locations of $p$ new facilities have been finalized.

Next we present the key steps of the above procedure in greater detail.
3.1 Concept of a feasible network

This research introduces a conceptual network to represent the feasibility of refueling among O-D pairs. Assume an undirected weighted network $G = (N, A)$ where $N$ includes O-D nodes, candidate facility locations, and transfer nodes with cardinality $|N| = n$ and $A$ is the set of arcs. This is referred to as the physical network in that it depicts the physical properties of a road network of interest. Based on the physical network, the algorithm creates an undirected weighted network $G_f = (V, E)$ where $V$ is the union of O-D nodes in $G$ and the set of temporarily selected facilities, and $E$ is the set of shortest paths among the nodes in $V$ that are feasible given the assumed vehicle driving range. $G_f$ is referred to as the feasible network.

Adding or substituting a candidate node to the temporary solution is essentially an update of $V$, which in turn requires an update of $E$. Paths are added as single links to the set $E$ if they meet the following criteria, which consider node type, link length, and vehicle range:

- If either end of a path is an O-D node, and a facility exists at either end, and the length of the path is equal to or shorter than half the vehicle range, then this segment of the round trip is feasible and the entire path is added to $E$ as a single link.
- If facilities exist at both ends of a path and the length of the path is equal to or shorter than the vehicle range, this segment of a round trip is feasible and the entire path is added to $E$ as a single link.

After the feasible network $G_f$ is constructed for $V$ given a set of stations in a temporary solution $S$, shortest paths can be generated for all O-D pairs. The shortest path length on $G_f$ represents the shortest feasible route on the physical network via the network of stations.

To illustrate the logic of the feasible network, consider the 7-node network in Figure 1. In addition, let us distinguish between O-D nodes and transfer nodes, where transfer nodes are neither at the origin nor the destination. With this distinction, first consider a case where there is a facility at node 3 and the vehicle range is 6 (Figure 2). With a facility located at 3, links 1-3 and 2-3 become feasible, and therefore $G_f$ is a graph with $V = \{1, 2, 3, 4, 6, 7\}$, $E = \{1, 3\}$ and $\{2, 3\}$. Note that $G_f$ may include vertices that are not connected via links, some of which would be members of O-D pairs but traveling between them is not feasible. With the two links added into $G_f$, no deviation is required for O-D pair 1-3 (30 trips) and therefore the entire flow volume is covered. In addition, a deviation path 1-3-2 for O-D pair 1-2 is also feasible, and a reduced

11
fraction of its flow (< 20) will also be covered by the facility at 3. Note that path 1-3-2 is the shortest deviation path for the pair 1-2 that can be obtained from the current feasible network $G_f$. Flows between 1 and 2 will not be fully covered (that is, with no deviation via node 3 necessary and no penalty) until there are stations at both nodes 1 and 2, at which time the algorithm will add a direct link from 1 to 2 to the feasible network.

3.2 Generation of shortest feasible deviation paths

Once a feasible network is constructed from a temporary solution, we can generate shortest feasible deviation paths. Because each link in a feasible network not only represents the shortest path between nodes in $G$ by design and its refueling feasibility is already verified while constructing $G_f$, the shortest path between a pair of nodes in $G_f$ is in effect the shortest feasible path. Therefore, by running an all-pairs shortest path algorithm on $G_f$, we can compute shortest feasible path distances among all O-D pairs. For each OD pair and set of stations in a temporary solution, one of three results is possible:

- A shortest path exists in $G_f$ and its length is equal to the shortest path distance in $G$. In this case, no deviation was required because all stations that need to be visited are on the shortest path in $G$.
- A shortest path exists in $G_f$ and its length is longer than the shortest path distance in $G$. In this case, some deviation was required from the shortest path in $G$.
- A shortest path does not exist in $G_f$, which means there is no feasible path in $G$.

Some shortest deviation paths obtained from $G_f$ may contain multiple cycles at the origin or destination in the physical network $G$. Because it is not a probable refueling behavior, we include a procedure that makes sure at most one cycle exists at either end of the path. Refer to [25] for further discussion of this situation.

3.3 Vehicle flows on the deviation paths

The fraction of vehicle flows on each feasible path ($g_{qr}$) is calculated from the deviation distance and penalty function, both of which are easily obtained. The product of $g_{qr}$ and $f_q$ equals the captured vehicle flows on the feasible deviation path; summing them up them over all $q$ results in the objective function value of the temporary solution. Comparing the objective
function values of temporary solutions, we can decide which node to add to the solution at each iteration from 1 to $p$.

### 3.4 A technical note on the complexity

Given that generation of shortest paths is computationally complex, the implemented shortest path (SP) algorithm is the main component that drives the overall efficiency of the heuristics. This research implements Dijkstra’s SP algorithm [26] to compute all O-D paths from each node $v \in V$ and employs a Fibonacci heap to store candidate nodes, as in Zhan [27] and Zhan and Noon [28].

The GRD-ADD needs to run an SP algorithm as many as $\sum_{i=1}^{p} (|V| - i + 1)$ times to obtain $p$ facilities. In the GRD-ADD-SUB, the number can increase multiplicatively up to the number of substitution iterations. Therefore, the complexity of the procedure that generates shortest deviation paths depends on the density of $G_r^*(V, E)$, which is affected by the structure of $G$, the deviation penalty function, the vehicle range, and the current feasible solution.

Empirical results in a test setting show that the procedure took 10 milliseconds for a temporary feasible graph $G_r^*(V, E)$ when $|V| = 12$ and $|E| = 11$, and 44 milliseconds when $|V| = 59$ and $|E| = 188$.\(^1\)

### 4. Numerical Experiments

This section provides the experimental design and results to test the performance of the heuristics for the DFRLM using two test networks. The first test was performed on a 25-node network that was used in previous research [1, 3, 29]. The test network has 25 nodes, 43 edges, and 300 O-D pairs with flow volumes estimated using a gravity model. Each node is both an O-D node and a candidate site (Figure 3). The second test used an aggregated road network for the US state of Florida [19, 30] with 302 nodes, 495 edges, 74 O-D nodes, and 2,701 O-D pairs and flow volumes estimated using a gravity model (Figure 4). The 25-node network is small enough to obtain optimal solutions and compare them with heuristics, but for the Florida network, the

\(^1\) Average of 10 runs. Both have $|O-D|$ of 74
optimal solutions were not obtained; the GRD-ADD is therefore compared with GRD-ADD-SUB for Florida. The DFRLM solutions are also compared with FRLM solutions without deviations to illustrate the benefit of considering driver deviations, which is followed by a discussion on the cost incurred by driver deviations.

For both networks, solutions for \( p = 1 \) to 25 were obtained while measuring the computation time of the algorithms. Different deviation penalty functions and maximum deviation distances were used to estimate the penalty for deviation. For the 25-node network, two different vehicle ranges (8 and 12) were used, and for the Florida state network, 100 miles was specified for the vehicle range. For the smaller 25-node test network, optimal solutions were obtained using Xpress 7.0 software with the MILP formulation [5]. The heuristic algorithms were implemented using C# language, and ESRI ArcGIS 10.0 was used as a platform to process data, to visualize the results, and as the platform for the heuristics. All the problem instances were solved on a computer with four 2.4 GHz cores and 4 GB memory.

4.1 Test on the 25 node network

4.1.1 Solution time

The GRD-ADD is much faster than GRD-ADD-SUB in all instances (Tables 1-2). Note that GRD-ADD-SUB solves in a similar amount of time regardless of the number of substitution iterations allowed. This implies that, for most of the steps from 1 to \( p \), no substitutions are found, and therefore only one substitution iteration is performed regardless of how many are allowed. Exact solutions were found fairly quickly for smaller problems (13 seconds for \( DD_{max} = 10\% \) of SP with a range of 8). However, as the driving range and maximum allowed deviation distance increased, the exact solution time increased substantially (2,661 seconds for \( DD_{max} = 50\% \) of SP with a range of 12), and considering the small size of the test network, this result confirms the necessity of using the heuristics for bigger problems.

4.1.2 Optimality gap

The objective values of optimal solutions were compared with the results from heuristics and are summarized in Table 2. As the vehicle range becomes longer, the GRD-ADD performed well. Even though GRD-ADD-SUB finds better solutions with substitution iterations being increased, it does not always find the optimal solution because the substitution was carried out one facility at a time.
We can observe a special case with the range = 12, where the solutions of heuristics sometimes had higher objective values than the MILP solution. The reason for this abnormality lies in the difference of input data. The k-shortest path (KSP) algorithm that was implemented in generating the paths for the Xpress model can generate a large number of paths, but did not generate all possible deviation paths. In particular, some paths with a cycle were not generated by the KSP algorithm used in this research. As a result, the MILP solution includes a facility that resulted in an inferior solution.

4.2 Results for the Florida state network

4.2.1 Tradeoff between objective gain and time

Results for the Florida state network are summarized in Table 3. The GRD-ADD-SUB produces better solutions with more flow coverage than GRD-ADD in most cases, yet with substantial increase of computation time.

Specification of parameters affects the solution performance. Specifically, as the \( DD_{\text{max}} \) is increased from 10% to 50% of the SP, the solution time increases 1.2~1.6 times (Table 3). The objective function gain from the increase in \( DD_{\text{max}} \) ranges between 0.67% and 6.74% with an average of 3.6%. The maximum gain is observed when the algorithms find a small number of facilities (\( p = 3\text{-8} \)). This result confirms that drivers will need larger deviations when there are fewer stations.

4.2.2 Comparison of DFRLM with FRLM

The results from DFRLM and FRLM greedy heuristics are compared by setting a very small number (30 seconds) for \( DD_{\text{max}} \) so that any driver would regard the deviation as negligible. Figure 5 shows the solutions for both models for \( p = 2 \) obtained from the GRD-ADD-SUB 3. Both models select node 256 (Fort Lauderdale) that provides service to heavy vehicle flows with full coverage. The FRLM, however, chooses node 270 in the central Miami (I-95 & I-195) as the second facility whereas DFRLM selects node 271 in North Miami. In this example, by allowing 30 seconds of deviation, the DFRLM provides a better solution for \( p = 2 \) that can provide 0.6% more coverage than the FRLM.

4.2.3 Benefits and costs of allowing deviations

Three deviation scenarios are devised to compare the effects of allowing deviations to AFV stations and summarized in Table 4 and Figure 6. We also examined the “cost” of the
deviations to the drivers by calculating the percentage of trips requiring deviations and the average deviation distance (measured in minutes). Table 4 clearly shows that the amount of additional coverage that can be gained from deviations will increase if we assume the drivers are more tolerant of deviations. The cost associated with each scenario also increases with the increase of the coverage. Figure 6 illustrates that the specification of $DD_{max}$ and $g(DD)$ affects the spatial arrangement of selected facilities as well.

This result implies that a rollout plan for a refueling station network would benefit from a careful estimation of drivers’ sensitivity to the required deviation at each development phase. Such practice is important not only because drivers’ willingness to deviate may change as the refueling network and alternative-fuel vehicle market become mature but because the way in which deviation behavior is modeled affects the optimal facility locations.

5. Conclusions and Future Research

In this paper, greedy-adding and greedy-adding with substitution heuristic algorithms based on network transformation are developed for solving real-world DFRLM problems. The heuristics, which are the first in the literature for the DFRLM, are based on the concept of a feasible network, the arcs of which are feasible paths between pairs of facilities or between facilities and origins and destinations, given the vehicle driving range and the temporary solution being evaluated. The procedures in the heuristic efficiently generate all shortest feasible deviation paths among all O-D pairs given the vehicle range and probability of drivers deviating while removing unrealistic multiple cycles at origins or destinations. The optimality gap tends to decrease as the number of substitutions and vehicle range increase. Comparison of the greedy and greedy-substitution heuristics showed that substitutions enhance the objective with the cost of increased solution time, but multiple substitutions do not add much to the solution time because no substitutions are made during most iterations. Most of the solution time is used to generate the all-pairs shortest paths on the feasible network.

The choice of deviation decay function and maximum allowed deviation both have effects on solution quality and optimal facility locations. Therefore, careful modeling of deviation behavior in practice is suggested. For example, infrastructure developers and government agencies will need to determine how sensitive potential (and actual) AFV drivers are
to the required deviations. Such assessment may be required in every important phase of the infrastructure development.

Additional future research is necessary to extend the reliability and usability of the DFRLM heuristics. Evaluation of refueling feasibility needs to take into account more diverse refueling behavior such as refueling or recharging at home or work, or refueling within a time window. It is expected that if candidate sites are not restricted to nodes and are numerous enough relative to vehicle range, the optimality gap will decrease as in [16]. Future implementation of different algorithms for solving the most time-consuming sub-problem in the heuristics will reduce computational effort and may provide more competitive performance. Development of a column generation method [31] to solve the DFRLM is another interesting research topic. Empirical research on the willingness of AFV drivers to deviate from their shortest paths is needed [32]. Finally, given that the DFRLM assumes each facility site is uncapacitated, the development of a model that can simultaneously consider deviation and station capacity is a promising future direction for research.

ACKNOWLEDGEMENTS
We gratefully acknowledge FICO for use of the Xpress Optimization Suite under their Academic Partnership Program. We would like to thank to the reviewers whose thoughtful comments help improve the previous version of the paper. Thanks also to Dr. Ozgur Araz for sharing his insights on related alternative-fuel problems.

REFERENCES


Figure 1. A 7-Node Physical Network and its O-D Flow Volume.

(The OD flow volumes shown are the full volume assuming the trip can be made on the shortest path. If a deviation to refuel is necessary, these volumes are reduced.)
Figure 2. The Feasible Network with a Facility at Node 3 (Range of 6).
Figure 3. 25-Node Test Network.
Figure 4. State of Florida Network.
Figure 5. Different Solutions for $p = 2$ from FRLM (left) and DFRLM (right).
Figure 6. Two solutions for $p = 4$ using different deviation assumptions with no facilities in common. On the left: $g(DD)_{\text{linear}}$, on the right: $g(DD)_{\text{NoDecay}}$. The shortest path from Bradenton to Titusville, highlighted in yellow, can be served by different set of stations.
<table>
<thead>
<tr>
<th>Range</th>
<th>$DD_{max}$, Distance Decay Function</th>
<th>Computation Time (CPU seconds)</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th>Optimal</th>
<th>GRD</th>
<th>Greedy Sub 1</th>
<th>Greedy Sub 2</th>
<th>Greedy Sub 3</th>
<th>Greedy Sub 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>10% of SP, No Decay</td>
<td>13.20</td>
<td>&lt; 1</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>50% of SP, No Decay</td>
<td>1605.53</td>
<td>&lt; 1</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>10% of SP, No Decay</td>
<td>23.24</td>
<td>1</td>
<td>9</td>
<td>10</td>
<td>10</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>50% of SP, No Decay</td>
<td>2660.54</td>
<td>1</td>
<td>9</td>
<td>10</td>
<td>10</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p</td>
<td>DDmax: 10% of SP, g(DD)$_{NoDelay}$</td>
<td>DDmax: 50% of SP, g(DD)$_{NoDelay}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>-------------------------------------</td>
<td>-------------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>OOFV (%)</td>
<td>Optimality Gap (%)</td>
<td>OOFV (%)</td>
<td>Optimality Gap (%)</td>
<td>OOFV (%)</td>
<td>Optimality Gap (%)</td>
<td>OOFV (%)</td>
<td>Optimality Gap (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>17.13</td>
<td>18.23</td>
<td>17.13</td>
<td>18.23</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>32.58</td>
<td>34.34</td>
<td>32.58</td>
<td>34.34</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>44.41</td>
<td>47.90</td>
<td>44.41</td>
<td>49.04</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>55.97</td>
<td>58.14</td>
<td>56.08</td>
<td>62.64</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>63.52</td>
<td>67.70</td>
<td>64.06</td>
<td>72.46</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>68.08</td>
<td>-</td>
<td>75.00</td>
<td>0.36</td>
<td>71.61</td>
<td>-</td>
<td>81.80</td>
<td>-0.43</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>72.32</td>
<td>-</td>
<td>82.68</td>
<td>2.03</td>
<td>74.40</td>
<td>0.82</td>
<td>91.46</td>
<td>1.80</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>77.87</td>
<td>3.98</td>
<td>88.88</td>
<td>4.31</td>
<td>84.56</td>
<td>6.62</td>
<td>95.61</td>
<td>2.69</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>82.77</td>
<td>1.21</td>
<td>92.93</td>
<td>3.85</td>
<td>92.18</td>
<td>-</td>
<td>97.59</td>
<td>2.35</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>90.06</td>
<td>-</td>
<td>96.83</td>
<td>4.61</td>
<td>95.99</td>
<td>-</td>
<td>98.97</td>
<td>2.08</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>94.41</td>
<td>-</td>
<td>97.81</td>
<td>2.64</td>
<td>98.25</td>
<td>-</td>
<td>99.54</td>
<td>1.50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>96.80</td>
<td>-</td>
<td>98.66</td>
<td>1.85</td>
<td>98.76</td>
<td>-</td>
<td>99.80</td>
<td>0.99</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>97.78</td>
<td>-</td>
<td>99.30</td>
<td>1.50</td>
<td>99.03</td>
<td>-</td>
<td>99.85</td>
<td>0.31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>98.43</td>
<td>-</td>
<td>99.85</td>
<td>1.19</td>
<td>99.45</td>
<td>0.33</td>
<td>99.95</td>
<td>0.15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>98.74</td>
<td>-</td>
<td>99.85</td>
<td>0.55</td>
<td>99.72</td>
<td>0.51</td>
<td>100.00</td>
<td>0.11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>99.71</td>
<td>-</td>
<td>100.00</td>
<td>0.15</td>
<td>99.81</td>
<td>-</td>
<td>100.00</td>
<td>0.05</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>99.77</td>
<td>-</td>
<td>100.00</td>
<td>0.07</td>
<td>99.87</td>
<td>0.02</td>
<td>100.00</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>99.86</td>
<td>0.09</td>
<td>100.00</td>
<td>-</td>
<td>99.96</td>
<td>0.11</td>
<td>100.00</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>99.92</td>
<td>0.15</td>
<td>100.00</td>
<td>-</td>
<td>100.00</td>
<td>0.15</td>
<td>100.00</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>99.92</td>
<td>0.15</td>
<td>100.00</td>
<td>-</td>
<td>100.00</td>
<td>0.15</td>
<td>100.00</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>99.92</td>
<td>0.15</td>
<td>100.00</td>
<td>-</td>
<td>100.00</td>
<td>0.15</td>
<td>100.00</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>99.92</td>
<td>0.15</td>
<td>100.00</td>
<td>-</td>
<td>100.00</td>
<td>0.15</td>
<td>100.00</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>99.92</td>
<td>0.15</td>
<td>100.00</td>
<td>-</td>
<td>100.00</td>
<td>0.15</td>
<td>100.00</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>99.92</td>
<td>0.15</td>
<td>100.00</td>
<td>-</td>
<td>100.00</td>
<td>0.15</td>
<td>100.00</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>99.92</td>
<td>-</td>
<td>100.00</td>
<td>-</td>
<td>100.00</td>
<td>-</td>
<td>100.00</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: a Optimal Objective Function Value, b Greedy, c Greedy Sub 4 Iterations
<table>
<thead>
<tr>
<th>P</th>
<th>G r d y</th>
<th>T i m e</th>
<th>G r d y</th>
<th>T i m e</th>
<th>G r d y</th>
<th>T i m e</th>
<th>G r d y</th>
<th>T i m e</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(% )</td>
<td>( s )</td>
<td>(% )</td>
<td>( s )</td>
<td>(% )</td>
<td>( s )</td>
<td>(% )</td>
<td>( s )</td>
</tr>
<tr>
<td>1</td>
<td>30.75</td>
<td>&lt;1.00</td>
<td>30.75</td>
<td>&lt;1.00</td>
<td>31.12</td>
<td>&lt;1.00</td>
<td>31.12</td>
<td>&lt;1.00</td>
</tr>
<tr>
<td>2</td>
<td>43.05</td>
<td>&lt;1.00</td>
<td>43.05</td>
<td>&lt;1.00</td>
<td>45.22</td>
<td>&lt;1.00</td>
<td>45.22</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>52.67</td>
<td>1</td>
<td>55.20</td>
<td>4</td>
<td>54.85</td>
<td>1</td>
<td>57.00</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>61.99</td>
<td>1</td>
<td>61.29</td>
<td>7</td>
<td>63.79</td>
<td>1</td>
<td>64.36</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>68.08</td>
<td>2</td>
<td>66.71</td>
<td>11</td>
<td>71.15</td>
<td>2</td>
<td>69.78</td>
<td>11</td>
</tr>
<tr>
<td>6</td>
<td>72.05</td>
<td>3</td>
<td>70.67</td>
<td>16</td>
<td>75.14</td>
<td>3</td>
<td>75.41</td>
<td>24</td>
</tr>
<tr>
<td>7</td>
<td>75.95</td>
<td>3</td>
<td>76.31</td>
<td>32</td>
<td>78.75</td>
<td>3</td>
<td>79.02</td>
<td>30</td>
</tr>
<tr>
<td>8</td>
<td>79.09</td>
<td>4</td>
<td>79.45</td>
<td>40</td>
<td>81.52</td>
<td>4</td>
<td>81.79</td>
<td>38</td>
</tr>
<tr>
<td>9</td>
<td>81.16</td>
<td>5</td>
<td>81.74</td>
<td>49</td>
<td>83.59</td>
<td>6</td>
<td>84.17</td>
<td>47</td>
</tr>
<tr>
<td>10</td>
<td>82.77</td>
<td>7</td>
<td>83.39</td>
<td>71</td>
<td>85.05</td>
<td>7</td>
<td>85.63</td>
<td>60</td>
</tr>
<tr>
<td>11</td>
<td>84.31</td>
<td>8</td>
<td>84.94</td>
<td>84</td>
<td>86.47</td>
<td>8</td>
<td>87.05</td>
<td>76</td>
</tr>
<tr>
<td>12</td>
<td>85.68</td>
<td>9</td>
<td>86.33</td>
<td>101</td>
<td>87.88</td>
<td>9</td>
<td>88.46</td>
<td>97</td>
</tr>
<tr>
<td>13</td>
<td>86.97</td>
<td>10</td>
<td>87.70</td>
<td>121</td>
<td>89.15</td>
<td>11</td>
<td>89.93</td>
<td>129</td>
</tr>
<tr>
<td>14</td>
<td>88.30</td>
<td>12</td>
<td>88.94</td>
<td>142</td>
<td>90.45</td>
<td>14</td>
<td>91.07</td>
<td>203</td>
</tr>
<tr>
<td>15</td>
<td>89.51</td>
<td>13</td>
<td>90.15</td>
<td>209</td>
<td>91.55</td>
<td>17</td>
<td>92.38</td>
<td>247</td>
</tr>
<tr>
<td>16</td>
<td>90.96</td>
<td>15</td>
<td>91.29</td>
<td>254</td>
<td>92.88</td>
<td>20</td>
<td>93.37</td>
<td>297</td>
</tr>
<tr>
<td>17</td>
<td>92.10</td>
<td>18</td>
<td>92.27</td>
<td>303</td>
<td>93.70</td>
<td>24</td>
<td>94.14</td>
<td>357</td>
</tr>
<tr>
<td>18</td>
<td>93.02</td>
<td>20</td>
<td>93.19</td>
<td>358</td>
<td>94.47</td>
<td>27</td>
<td>94.91</td>
<td>420</td>
</tr>
<tr>
<td>19</td>
<td>93.68</td>
<td>24</td>
<td>93.83</td>
<td>417</td>
<td>95.25</td>
<td>31</td>
<td>95.59</td>
<td>489</td>
</tr>
<tr>
<td>20</td>
<td>94.32</td>
<td>28</td>
<td>94.45</td>
<td>488</td>
<td>95.93</td>
<td>35</td>
<td>96.20</td>
<td>632</td>
</tr>
<tr>
<td>21</td>
<td>94.92</td>
<td>31</td>
<td>95.07</td>
<td>566</td>
<td>96.54</td>
<td>39</td>
<td>96.65</td>
<td>711</td>
</tr>
<tr>
<td>22</td>
<td>95.44</td>
<td>36</td>
<td>95.63</td>
<td>650</td>
<td>97.00</td>
<td>43</td>
<td>97.01</td>
<td>794</td>
</tr>
<tr>
<td>23</td>
<td>95.87</td>
<td>40</td>
<td>96.14</td>
<td>741</td>
<td>97.36</td>
<td>47</td>
<td>97.36</td>
<td>885</td>
</tr>
<tr>
<td>24</td>
<td>96.27</td>
<td>44</td>
<td>96.51</td>
<td>837</td>
<td>97.71</td>
<td>51</td>
<td>97.70</td>
<td>981</td>
</tr>
<tr>
<td>25</td>
<td>96.64</td>
<td>48</td>
<td>96.83</td>
<td>938</td>
<td>97.98</td>
<td>55</td>
<td>97.98</td>
<td>1085</td>
</tr>
</tbody>
</table>
Table 4. Gains and Costs of Allowing Deviations ($p = 5$, GRD-SUBS-4)

<table>
<thead>
<tr>
<th>FRLM</th>
<th>(g(DD)_{\text{NoDecay}})</th>
<th>(g(DD)_{\text{linear}})</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solution</td>
<td>{108, 184, 242, 256, 263}</td>
<td>{107, 183, 248, 271, 273}</td>
</tr>
<tr>
<td>Vehicle flows that can be refueled (%)</td>
<td>64.89</td>
<td>72.33</td>
</tr>
<tr>
<td>Percentage of trips requiring deviations (%)</td>
<td>0</td>
<td>39.47</td>
</tr>
<tr>
<td>Average deviation distance (minutes)</td>
<td>0</td>
<td>4.6</td>
</tr>
</tbody>
</table>