Fleet Management for Vehicle Sharing Operations

The Pennsylvania State University ❖ University of Maryland
University of Virginia ❖ Virginia Polytechnic Institute and State University
West Virginia University

The Pennsylvania State University ❖ The Thomas D. Larson Pennsylvania Transportation Institute
Transportation Research Building ❖ University Park, Pennsylvania 16802-4710
Phone: 814-863-1909 ❖ Fax: 814-863-37073
www.pti.psu.edu/mautc

1. Report No. UMD-2008-02	2. Government Accession No.	3. Recipient's Catalog No.

4. Title and Subtitle	5. Report Date
Fleet Management for Vehicle Sharing Operations	May 2010
	6. Performing Organization Code
7. Author(s) Elise Miller-Hooks, Rahul Nair	8. Performing Organization Report No
9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
University of Maryland 1173 Glenn Martin Hall College Park, MD 20742	11. Contract or Grant No. DTRT07-G-0003
12. Sponsoring Agency Name and Address US Department of Transportation Research & Innovative Technology Admin UTC Program, RDT-30 1200 New Jersey Ave., SE Washington, DC 20590	13. Type of Report and Period Covered Final
	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract Transit, touted as a solution to urban mobility problems, cannot match the addictive flexibility of the automobile. 86.5% of all trips in the U.S. are in personal vehicles (USDOT 2001). A more recent approach to reduce automobile ownership is through the use of vehicle sharing programs (VSPs). A VSP involves a fleet of vehicles located strategically at stations across the transportation network. In its most flexible form, users are free to check out vehicles at any station and return them to stations close to their destinations. Vehicle fleets can be comprised of bicycles, low emission cars or electric vehicles. Such systems offer innovative, low-cost, and flexible solutions to the larger mobility problem and can have positive impacts on the transportation system as a whole by reducing urban congestion.

To match automobile flexibility, users are free to determine all trip characteristics (where to checkout and return vehicles, duration of travel and time of travel). This places exceptional logistical challenges on operators who must ensure demand in the near future is met. Since flow from one station to another is seldom equal to flow in the opposing direction, the VSP fleet can become spatially imbalanced. To meet near-future demand, operators must then redistribute vehicles to correct this asymmetry. The focus of this report is to provide efficient, cost-effective operational strategies for fleet management.

A stochastic, mixed-integer program (MIP) involving joint chance constraints is developed that generates least-cost vehicle redistribution plans for shared-vehicle systems such that a proportion of all near-term demand scenarios are met. The model aims to correct short term demand asymmetry in shared-vehicle systems, where flow from one station to another is seldom equal to the flow in the opposing direction. The model accounts for demand stochasticity and generates partial redistribution plans in circumstances when demand outstrips supply. This stochastic MIP has a non-convex feasible region that poses computational challenges. To solve the proposed program two solution procedures are developed. The first procedure is based on enumerating p-efficient points, used to transform the problem into a set of disjunctive, convex MIPs. A novel divide-and-conquer algorithm for generating p-efficient points that handles dual-bounded chance constraints is developed. Our technique has a smaller memory and computational footprint than previously proposed methods. Since this method can be computationally prohibitive for large shared-vehicle systems, we develop a faster cone-generation method that assumes that the random demand at each station is independent. Finally, using an equal-failure apportionment assumption we develop a bound on the problem that can also be used to generate redistribution strategies.

17. Key Words	18. Distribution Statement No restrictions. This document is available from the National Technical Information Service, Springfield, VA 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified		

FLEET MANAGEMENT FOR VEHICLE SHARING OPERATIONS

FINAL REPORT

UMD-2008-02

Prepared for

Mid-Atlantic University Transportation Center (MAUTC)

By

Dr. Elise Miller-Hooks, Principal Investigator
Rahul Nair, Graduate Research Assistant

Department of Civil & Environmental Engineering
University of Maryland
1173 Glenn Martin Hall,
College Park, MD 20742

May 2010

This work was sponsored by the U.S. Department of Transportation, Federal Highway Administration. The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the sponsor at the time of publication. This report does not constitute a standard, specification, or regulation.

Executive Summary

Transit, touted as a solution to urban mobility problems, cannot match the addictive flexibility of the automobile. 86.5% of all trips in the U.S. are in personal vehicles (USDOT 2001). A more recent approach to reduce automobile ownership is through the use of *vehicle sharing programs* (VSPs). A VSP involves a fleet of vehicles located strategically at stations across the transportation network. In its most flexible form, users are free to check out vehicles at any station and return them to stations close to their destinations. Vehicle fleets can be comprised of bicycles, low emission cars or electric vehicles. Such systems offer innovative, low-cost, and flexible solutions to the larger mobility problem and can have positive impacts on the transportation system as a whole by reducing urban congestion.

To match automobile flexibility, users are free to determine all trip characteristics (where to check-out and return vehicles, duration of travel and time of travel). This places exceptional logistical challenges on operators who must ensure demand in the near future is met. Since flow from one station to another is seldom equal to flow in the opposing direction, the VSP fleet can become spatially imbalanced. To meet near-future demand, operators must then redistribute vehicles to correct this asymmetry. The focus of this report is to provide efficient, cost-effective operational strategies for fleet management.

A stochastic, mixed-integer program (MIP) involving joint chance constraints is developed that generates least-cost vehicle redistribution plans for shared-vehicle systems such that a proportion of all near-term demand scenarios are met. The model aims to correct short term demand asymmetry in shared-vehicle systems, where flow from one station to another is seldom equal to the flow in the opposing direction. The model accounts for demand stochasticity and generates partial redistribution plans in circumstances when demand outstrips supply. This stochastic MIP has a non-convex feasible region that poses computational challenges. To solve the proposed program two solution procedures are developed. The first procedure is based on enumerating p-efficient points, used to transform the problem into a set of disjunctive, convex MIPs. A novel divide-and-conquer algorithm for generating p-efficient points that handles dual-bounded chance constraints is developed. Our technique has a smaller memory and computational footprint than previously proposed methods. Since this method can be computationally prohibitive for large shared-vehicle systems, we develop a faster cone-generation method that assumes that the random demand at each station is independent. Finally, using an equal-failure apportionment assumption we develop a bound on the problem that can also be used to generate redistribution strategies.

The developed model and associated solution methods are implemented for a car-sharing system in Singapore. Historical usage patterns for the system are derived using available data from 2003 till 2006. Five different strategies were implemented for various reliability levels. Several measures of effectiveness (system reliability, cost of redistribution plans, and dropped demand in simulation studies) are used for comparison of various strategies. These measures demonstrate that the methods that account for the inherent stochasticity of the demand process consistently outperform methods that assume known or static demand. Operators employing redistribution plans generated by considering uncertainty can provide better level-of-service to users. Additionally, the trade-offs between achieved reliability and redistribution costs are explored for various strategies.

Contents

List of Figures

List of Tables

1. Introduction

Transit, touted as a solution to urban mobility problems, cannot match the addictive flexibility of the automobile. 86.5% of all trips in the U.S. are in personal vehicles (USDOT 2001). A more recent approach to reduce automobile ownership is through the use of *vehicle sharing programs* (VSPs). A VSP involves a fleet of vehicles located strategically at stations across the transportation network. In its most flexible form, users are free to check out vehicles at any station and return them to stations close to their destinations. Vehicle fleets can be comprised of bicycles, low emission cars or electric vehicles. Such systems offer innovative, low-cost, and flexible solutions to the larger mobility problem and can have positive impacts on the transportation system as a whole by reducing urban congestion.

For a user, a shared vehicle reduces cost of ownership without sacrificing flexibility. Vehicles, viewed as a resource, spend most of their time idle and depreciating in value. More efficient use of this resource implies lower costs for users. Shared vehicle fleets offer a mechanism for exploiting this down-time. In addition to cost benefits to individual users, there are system-wide benefits from a decrease in motor vehicles in the system (a recent European study estimates that one shared vehicle lead to reduction of between four to 10 privately owned cars (Rydén and Morin 2005); estimates for North America range from six to 23 cars (Shaheen and Cohen 2007)). These programs typically have pricing structures that charge based on usage, which has been shown to reduce travel amongst participants (Shaheen et al. 2003). In essence, a smaller fleet of vehicles offers a comparable level-of-service to users, and provides system-wide benefits. This concept of pooling resources has parallels in other sectors, such as air transport (Kuby and Gray 1993, Barnhart and Schneur 1996, Kim et al. 1999) and consolidated freight systems (Taniguchi et al. 1999, Fusco et al. 2003), for the same reasons.

In addition to being environment-friendly, socially responsible and economical, operators and users around the world have found VSPs to be profitable. As of 2007, car sharing programs exist in 600 cities around the world (Shaheen and Cohen 2007) and bicycle sharing programs in 102 cities (Britton 2008). *Vélib'*, the bicycle sharing program in Paris, has 20,600 bicycles spread over 1,450 stations across the city with an average of 120,000 trips daily (Erlanger 2008).

To match automobile flexibility, VSPs transfer control of vehicles to users. This places exceptional logistical challenges on operators who must ensure demand in the near future is met. Since flow from one station to another is seldom equal to flow in the opposing direction, the VSP fleet can become spatially imbalanced. To meet near-future demand, operators must then redistribute vehicles to correct this asymmetry. The focus of this report is to provide efficient, cost-effective operational strategies for fleet management.

The management of VSP fleets differs from previously studied models in related areas. These differences preclude the direct application of prior work and motivate the development of problem specific tools. Firstly, since users determine the trip characteristics, critical system attributes (where vehicles are checked out and returned, and the duration of lease) are beyond the control of operators. Secondly, there is a 'duality' of demand between vehicles and slots. A successful trip needs a vehicle available at the origin station and an available docking slot at the destination. A vehicle checkout reduces vehicle inventory at a station, but increases the slot inventory. This duality has significant implications and reduces the range of acceptable inventory levels for vehicles. Thirdly, the inventory is never consumed, but is merely moved. The fleet management strategy involves correcting imbalances at the various stations.

Past research in vehicle sharing programs has originated from social perspectives of sharing resources (Matsuura 2003, DeMaio and Gifford 2004, Katzev 2003, Wray 2008), policy considerations (TRB 2005, Shaheen et al. 2004), integration of transit and slow modes (Rodier et al. 2004, Shaheen et al. 2005, Shaheen and Rodier 2006, Shaheen et al. 2006), travel behavioral implications (Cervero et al. 2002), challenges in increasing awareness (Dawn Haines 2005), and analyzing potential market characteristics in terms of demographics, etc. (summarized in Shaheen and Cohen (2007)). These studies take a demand-side approach.

Supply-side analyses of VSPs are predominantly qualitative and the literature dealing specifically with fleet management for VSP's is limited. Barth and Todd (1999) proposed three strategies to generate redistribution plans. These are based on immediate demand, expected demand and perfect demand information. The time period looks 20 minutes into the future and uses simulation to evaluate the redistribution strategies. No details on how redistribution plans are generated are presented. Barth et al. (2004) studied the redistribution problem, but attempt to shift the burden of redistribution on users through two mechanisms of ride splitting and joining. Kek et al. (2009) used an mixed-integer program (MIP) to generate redistribution plans and allocate operator staff to redistribution and maintenance activities. Their model uses a time-expanded network, with static, known demand. Unserviced demand is penalized by a penalty cost in the objective function. A simulation model is used to evaluate the redistribution strategy. In these works, the redistribution plans are based on static demand.

In this report, the problem of fleet management for shared-vehicle systems is formulated considering demand uncertainty (Section 2). The management strategy involves anticipative fleet redistribution that operators initiate to correct short-term demand asymmetry (since flow from one station to another is seldom equal to flow in the opposite direction). When operators have inadequate resources to meet demand, then the model generates partial redistribution plans. The model takes a form of a stochastic MIP with joint chance constraints. Stochastic programs of this class have non-convex feasible regions. Two solution methods are presented. One approach to deal with this non-convexity uses the concept of p-efficient points (PEPs) to transform the problem to a disjunctive MIP that is more readily solvable. A divide-and-conquer algorithm to generate PEPs is designed to reduce the computational burden of existing methods (Section 3). The algorithm can be applied to any problem with joint chance constraints given that the vector of random variables is discrete. This contribution transcends the current application. A second cone generation solution method (Section 3.2), akin to the column generation procedure, can be employed when the demand at each station is assumed to be independent. Under this limiting assumption, this method provides quick solutions even for shared systems with large number of stations. Additionally, we derive an equal-apportionment bound to the problem (Section 4). We compare various redistribution strategies in a real-world application to a car sharing system in Singapore (Section 5). Extensive computational experiments and simulation studies show that when the redistribution strategies developed herein are employed, the system operates at a reliability level that would otherwise be possible only with capital improvements to the system.

2. Problem Formulation

Given (a) the system configuration (stations, capacities, fleet size), (b) current inventory levels at each station, (c) relocation costs, and (d) a probabilistic characterization of demand at each station, we wish to find a least-cost fleet redistribution plan such that *most* near future demand scenarios are satisfied.

VSP operators have substantial ITS infrastructure for various functions, including tracking of vehicles for theft prevention, smart cards for member access, vehicle availability across the network, charging consoles for electric vehicles, payment systems, and traveler information services. This data-rich environment provides a real-time awareness of the system that can be leveraged for fleet management. Since individual users decide where and when trips are made, demand at each station is uncertain from a system perspective. The aim of operators is to serve all demand. However, it is typically cost-prohibitive to design the system to satisfy all possible demand realizations and operators can expect demand to outstrip supply in high-demand scenarios. By characterizing demand probabilistically, as can be done using historical information, operators can quantify the existing level-of-service. If the desired level-of-service at a station is not met, then a fleet redistribution action can be initiated to bring the system to an acceptable state.

The VSP system can be defined on a network of n stations. Each station i has capacity, C_i, the maximum number of vehicles it can accommodate. The capacity represents parking bays for car-sharing, docking slots in bike-sharing systems, or charging stations for electric vehicles. The number of vehicles at station i, termed the station inventory, is denoted by V_i. The cost of relocating vehicles from station i to station j, $i \neq j$, is denoted by a_{ij}. There is a penalty δ to move each vehicle between two stations. The system operator has perfect information on available inventories at each station. The system operator plans for a fixed short-term planning horizon for which demand is known probabilistically. Redistribution tasks are assumed to be completed before the planning period commences. At each station i, there are two types of demand processes, one to check out vehicles, ξ_i^1, and the other to return vehicles, ξ_i^2. Both ξ_j^1 and ξ_j^2 are random variables with known probability distributions. The operator seeks a least-cost redistribution plan that would make the system p-reliable during the planning period. That is, the system satisfies all demand at every station $(1, \ldots, n)$, for p proportion of all possible realizations. This can be described by the following joint-chance constraint.

$$\mathbb{P}\begin{pmatrix} \text{Available vehicles at stn } 1 \geq \xi_1^1, & \text{Available spaces at stn } 1 \geq \xi_1^2 \\ \text{Available vehicles at stn } 2 \geq \xi_2^1, & \text{Available spaces at stn } 2 \geq \xi_2^2 \\ \vdots & \vdots \\ \text{Available vehicles at stn } n \geq \xi_n^1, & \text{Available spaces at stn } n \geq \xi_n^2 \end{pmatrix} \geq p. \tag{1}$$

Equation (1) represents a level-of-service constraint for the operator who seeks a p-reliable system. To achieve this, the operator looks at available inventory at each station. If the available resources, both vehicles and free spaces, are adequate to satisfy p-proportion of all possible demand scenarios, then no further corrective actions are necessary. If the available vehicles are insufficient, then vehicles can be 'borrowed' from adjacent stations. If available spaces are inadequate, then vehicles can be 'lent out' to other stations to free up spaces. Since there are costs involved in these actions, the operator seeks an optimal method to perform this redistribution.

To derive the level-of-service constraint, we note that the available vehicle inventory at each station depends on the current inventory and the number of returns and checkouts during the time period. If the redistribution plan calls for vehicles to be relocated into (or out of) the station, then these vehicles are assumed to be available (or unavailable) at the start of the planning period. This assumption is not restrictive, since redistribution tasks can commence well before the planning period begins. Similarly, the available spaces inventory at each station depends on the current inventory, the number of returns, and the number of vehicles relocated in and out of the station during the planning period. Therefore,

$$\text{Available vehicles at Stn } i = \begin{cases} \text{Vehicle inventory at } i \qquad (V_i) \\ + \\ \text{Returns at } i \qquad (\xi_i^2) \\ + \\ \text{Vehicles relocated to } i \qquad (\sum_j y_{ji}) \\ - \\ \text{Vehicles relocated out of } i \qquad (\sum_j y_{ij}) \end{cases} \tag{2}$$

$$\text{Available spaces at Stn } i = \begin{cases} \text{Spaces inventory at } i \qquad (C_i - V_i) \\ + \\ \text{Checkouts at } i \qquad (\xi_i^1) \\ + \\ \text{Vehicles relocated out of } i \qquad (\sum_j y_{ij}) \\ - \\ \text{Vehicles relocated to } i \qquad (\sum_j y_{ji}). \end{cases} \tag{3}$$

It is assumed that redistribution is completed *before* the planning period. Therefore return (ξ_i^2) and checkout(ξ_i^1) random variables are for the future planning period, while the inventory (V_i) and redistribution variables (y_{ij}) denote operator actions before the planning period commences. The current vehicle inventory, V_i, is known as is the corresponding spaces inventory, $C_i - V_i$. Let x_{ij} denote a binary decision variable indicating if vehicles are moved from i to j in anticipation of future demand. Let y_{ij} denote an integer decision variable indicating the number of vehicles moved from i to j. In terms of decision variables y_{ij}, the level-of-service constraint (1), can be written as

$$\mathbb{P}\begin{pmatrix} V_i + \sum_{j=1}^{n}(y_{ji} - y_{ij}) + \xi_i^2 \geq \xi_i^1, \qquad i = 1, ..n \\ C_i - V_i + \sum_{j=1}^{n}(y_{ij} - y_{ji}) + \xi_i^1 \geq \xi_i^2, \; i = 1, ..n \end{pmatrix} \geq p. \tag{4}$$

Let ξ_i to be the net demand at a station i for the planning period. That is, $\xi_i = \xi_i^1 - \xi_i^2$. The two types of demand (vehicles and spaces) exhibit duality (reduction of one type implies an increase of the other), so the net demand ξ_i encodes both types of demand in one random variable and represents the imbalance between demand for vehicles and checkouts. For a particular time period, if the realization of ξ_i is positive, there are more checkouts than returns. Similarly, if ξ_i is negative, there are more returns than checkouts.

The level-of-service constraint (4) cannot be met for every demand realization. For example, in scenarios where the available resources outstrip demand, this constraint is infeasible. To recover *partial redistribution plans* that help operators make the best possible use of available resources (though still shy of the desired level-of-service), phantom vehicle and space variables are introduced. For each station, let w_i be the number of phantom vehicles and z_i be the number of phantom spaces. Additionally, let γ be a large penalty cost that forces the use of phantom resources only if necessary. The variables w_i and z_i relax the level-of-service constraint as shown in constraint (6). This relaxation allows the model to generate *partial* redistribution plans and maintains feasibility even if resources are inadequate. The optimal fleet redistribution plan can be formulated as a chance constrained model with desired reliability p (CCM-p) as follows.

$$(\mathrm{CCM}-p) \qquad \min \sum_{i=1}^{n}\sum_{j=1}^{n}(a_{ij}x_{ij}+\delta y_{ij})+\sum_{i=1}^{n}\gamma\left(w_i+z_i\right), \tag{5}$$

$$\text{s.t. } \mathbb{P}\begin{pmatrix} V_i+\sum_{j=1}^{n}(y_{ji}-y_{ij})+w_i\geq \xi_i, & i=1,\ldots,n \\ C_i-V_i+\sum_{j=1}^{n}(y_{ij}-y_{ji})+z_i\geq -\xi_i, & i=1,\ldots,n \end{pmatrix}\geq p, \tag{6}$$

$$\sum_{j=1}^{n} y_{ij}\leq V_i \qquad i=1,\ldots,n, \tag{7}$$

$$\sum_{j=1}^{n} y_{ji}\leq C_i-V_i \qquad i=1,\ldots,n, \tag{8}$$

$$y_{ij}\leq M\cdot x_{ij} \qquad i=1,\ldots,n, j=1,\ldots,n, \tag{9}$$

$$y_{ij}, w_i, z_j\in \mathbb{Z}_+ \qquad i=1,\ldots,n, j=1,\ldots,n, \tag{10}$$

$$x_{ij}\in [0,1] \qquad i=1,\ldots,n, j=1,\ldots,n. \tag{11}$$

The objective (5) represents the fixed cost for relocating vehicles, the cost of moving additional vehicles, and the penalty costs for utilizing phantom resources. Fixed cost between of redistribution two stations can be based on distance. The operator seeks to minimize the total cost of redistribution. The probabilistic level-of-service constraint (6) states that the redistribution plan must result in inventories that satisfy p proportion of all demand scenarios in the planning horizon. If available resources are insufficient, then this constraint is relaxed using phantom resources. There are capacity constraints (7) that limit the number of vehicles relocated out of a station to be no greater than the vehicles available at the start of the planning period. Similarly, there are capacity constraints (8) for slots at a station stating that the number of vehicles relocated to a station does not exceed the number of slots available. Constraints (9) relates the decision variables. All decision variables are non-negative integer valued (10), except x_{ij} which is binary (11).

Program CCM-p is a stochastic MIP for determining the optimal redistribution plan to satisfy p proportion of demand. In situations where available resources are inadequate to match anticipated demand, the program yields a partial redistribution plan. In this case, the phantom resources are utilized ($w_i > 0$ or $z_i > 0$) and the system is no longer p-reliable. The true reliability must be recomputed as shown in Section 3.1.2. If phantom resources are not utilized, the joint chance constraint (6) states that the probability that demand (in terms of vehicles and slots) exceeds supply is no greater than $1-p$.

Unless the joint distribution of ξ is log-concave, this stochastic MIP has a non-convex feasible region, making it computationally challenging to solve. The next section presents a specialized technique for a solution that deals with this non-convexity. This technique applies when the random vector is only on the right-hand side (RHS) and is discrete as is the case in this application.

3. Solving Program CCM-p

CCM-p is a stochastic MIP with joint chance constraints. In general these programs are difficult to solve, but since the random vectors are discrete and appear only in the RHS of the constraints, a specialized technique involving p-efficient points (PEPs) can be employed. Two solution procedures are presented. In the first method (Section 3.1), the main idea is to transform the non-convex feasible space to a disjunctive set of convex spaces. This transformation leads to a family of MIPs, one for each convex set. A single PEP characterizes one such convex set by substituting the chance constraint by linear constraints. A PEP is formally defined shortly, but generally speaking, a PEP is the smallest possible (non-dominated) vector for which the joint chance constraint is valid. For example, if v is a PEP, then the chance constraint $\mathbb{P}(Ax \geq \xi) \geq p$ can be substituted by a linear constraint $Ax \geq v$, since v represents a realization of ξ that ensures that the chance constraint is met (see Figure 1). Solving the family of MIPs yields a set of solutions, the best amongst which is optimal for the original non-convex program.

To generate the family of MIPs, the set of PEPs needs to be enumerated. When the dimension of the random vector is large, enumerating PEPs can be problematic, since the set of PEPs can be very large. Once the set of PEPs is enumerated, the family of MIPs is solved using existing MIP solvers. While this method is not new, our contribution is a PEP enumeration algorithm that aims to address the major bottleneck in the enumeration phase of the algorithm. The proposed divide-and-conquer procedure is more efficient that existing methods (Prékopa et al. 1998, Prékopa 2003, Beraldi and Ruszczyński 2002). Additionally, we extend the PEP concept to dual-bounded chance constraints.

The second solution method (Section 3.2) reduces the computational burden of PEP enumeration but makes a limiting assumption on the independence of the random vector. The main idea is similar to column generation where only necessary columns (or PEPs) that improve the objective are generated. The master problem is a convexified linear approximation of the CCM-p. The simplex multipliers from the master problem are used in the subproblem to direct the PEP enumeration phase of the algorithm.

The idea of PEPs was first proposed by Prékopa (1990) who also presented ways to deal with the chance constraint if the marginal distribution of the random vector is log-concave (Prékopa 1995). Beraldi et al. (2004) documents an application. Prékopa et al. (1998) presented a nested algorithm for generating PEPs. The main idea is to recursively explore the search space while keeping certain dimensions fixed. Beraldi and Ruszczyński (2002) proposed two enumeration schemes, backward and forward, along with hybrid schemes that attempt to avoid complete enumeration of PEPs under some restrictive conditions on the properties of the random vector. They also derived conditional bounds that aid in determining if a candidate vector is a PEP. Dentcheva et al. (2002) proposed hybrid methods, called convexification and cone generation methods, with the aim of avoiding explicit PEP enumeration. Saxena et al. (2008) combined the enumeration scheme with the solution phase for the same reasons. They introduced the concept of p-*in*efficiency to reduce constraints in the resulting program.

For a joint chance constraint of the form $\mathbb{P}(Ax \geq \xi) \geq p$, where ξ is discrete, the enumeration of PEPs is challenging, since the search space includes all possible realizations of the discrete random vector. The performance of any enumeration scheme depends on (a) the dimensionality of the random vector, (b) the support of the random vector, (c) complexity of evaluating the joint distribution function, and (d) the value of p. Increasing the dimensionality causes a combinatorial explosion in the search space. Increasing the support of the random vector also increases the search space, although not combinatorially. All enumeration schemes must evaluate the joint distribution function repeatedly. Thus, even moderate complexity in calculating the distribution function negatively impacts performance. Lastly, if the value of p is closer to either 0 or 1, the number of possible PEPs is considerably less than if it is close to 0.5, since there are fewer combinations along different dimensions of the random vector.

3.1. Solution based on PEP Enumeration

Since a MIP needs to be solved for each PEP, the solution procedure is designed to reduce the number of MIPs that need to be solved using some problem specific properties. First, since *complete* redistribution plans are preferable to *partial* ones, conditions for which a particular PEP will yield a guaranteed sub-optimal (partial) solution are derived in section 3.1.2. If a complete redistribution plan has been found, these *infeasibility* conditions help in screening PEPs that will yield partial solutions, thereby reducing the number of MIPs to be solved. Second, a zero-cost redistribution plan implies that no imbalance exists, so this forms the absolute lower bound on the problem. Third, since the set of PEPs S_p is large, it may preclude a complete enumeration and alternate termination criteria can be used to settle on an acceptable solution. A partial enumeration provides a solution that is not guaranteed to be optimal. The algorithm for solving CCM-p, given the inventories V_i, $i=1,\ldots,n$ at each station and the desired reliability level p, is as follows.

Algorithm ENUM-p:

Step 0. **Initalization.** Initialize the objective value of best solution $z^{opt}=\infty$. Set the PEP counter $k=1$. Let CR be a boolean flag that is true if a complete redistribution plan exists. Set $CR=$ false.

Step 1. **PEP Enumeration.** Generate the set of all PEPs S_p (see Section 3.1.1).

Step 2. For the k-th PEP $(u_k,v_k)\in S_p$ proceed to Step 3.

Step 3. **Feasibility Test.** If $CR=$ true, check all $n+2$ feasibility conditions (14), (15), and (16) (see section 3.1.2) for (u_k,v_k). If feasible or if $CR=$ false, proceed to Step 4; otherwise, set $k=k+1$ and return to Step 2.

Step 4. **Solve Deterministic Equivalent.** For a PEP (u_k,v_k), solve the program (5)–(11) by replacing the joint chance constraint (6) by the two linear constraints

$$V_i+\sum_{j=1}^{n}(y_{ji}-y_{ij})+w_i\geq v_k \qquad \text{and} \tag{12}$$

$$-\left(C_i-V_i+\sum_{j=1}^{n}(y_{ij}-y_{ji})+z_i\right)\leq u_k. \tag{13}$$

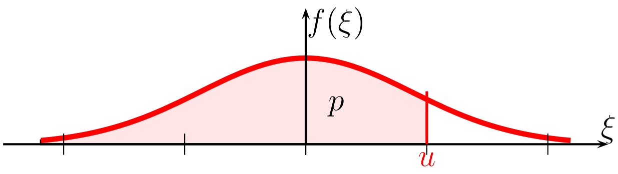
Step 5. **Solution Check.** If the objective value $z_k<z^{opt}$, then $z^{opt}=z_k$ and save the redistribution plan corresponding to z_k as optimal. If this redistribution plan does not use phantom resources, set $CR=$**true**. If $z_k=0$, absolute lower bound reached, terminate.

Step 5. If termination criteria are met, stop; else $k=k+1$ and return to Step 2.

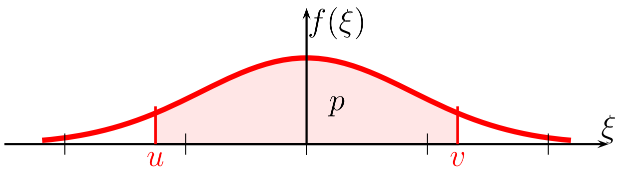
3.1.1. A Divide-and-Conquer PEP Enumeration Algorithm Since all past developments of PEPs have dealt with random vectors having an upper or lower bound but not both, the concept is extended to the case when the random vector is dual-bounded as is the case in constraint (6). Essentially, the procedure is developed for chance constraints of the form $\mathbb{P}(A'x\leq\xi\leq Ax)\geq p$, but also applies to the classical chance constraint $\mathbb{P}(Ax\geq\xi)\geq p$. The principle difference in the treatment of dual-bounded constraints is that PEPs are expressed as vector pairs and the cumulative distribution function is replaced by a function $g(u,v)$ that handles dual bounds. Barring these distinctions, the following development emulates concepts proposed by Prékopa (2003) and others (Beraldi and Ruszczyński 2002, Dentcheva et al. 2002).

Definition 1. For $p\in(0,1)$ a vector pair(u,v), $u\in\mathbb{Z}^n$ and $v\in\mathbb{Z}^n$, is said to be a p-efficient point (PEP) if $g(u,v)\geq p$, where $g(u,v)=\mathbb{P}(u_i\leq\xi_i\leq v_i,\ \forall i=1,..,n)$ and there exists no vectors y and z such that $g(y,z)\geq p$, $z\leq v$, and $y\geq u$.

For dual-bounded chance constraints, the definition employs a function $g(u,v)$ instead of the cumulative distribution function to ensure that the lower bound is also met as shown in Figure 1.



(a)Standard definition when $Ax \geq u \Rightarrow \mathbb{P}(Ax \geq \xi) \geq p$



(b)Modified dual-bounded definition when $A'x \leq u$ and $Ax \geq v \Rightarrow \mathbb{P}(A'x \leq \xi \leq Ax) \geq p$

Figure 1 Illustration of PEP definitions for a single dimension

For this modified dual-bounded case, we first derive a bound that will serve to test if a candidate vector is a PEP similar to the bounds based on the conditional marginal distribution derived by Beraldi and Ruszczyński (2002).

Proposition 1. *A vector pair (u,v) is PEP if and only if $l(u) = u$ and $h(v) = v$, where*

$$l_i(u) = \arg\max \{k \mid g(u,v) \geq p, u_i = k\},\ i = 1,..,n \text{ and}$$
$$h_i(v) = \arg\min \{k \mid g(u,v) \geq p, v_i = k\},\ i = 1,..,n$$

for $g(u,v) = \mathbb{P}(u_i \leq \xi_i \leq v_i, \forall i = 1,..,n)$.

Proof. **Bounds $\Rightarrow$ PEP**. The proof is by contradiction and is shown for the lower bound only. Similar arguments can be applied to the upper bound. Assume a pair of vectors (u,v), where $l(u) = u$ and $h(v) = v$. Take a vector pair (y,v) that is PEP, such that $y \leq u$ with $y_k < u_k$ for an arbitrary dimension k. Since $g(u,v)$ monotonically decreases in u, $g(y,v) \geq g(u,v)$ since $y \leq u$. The bounds $l(u) = u$ and $h(v) = v$ imply that $g(u,v) \geq p$. Therefore, (y,v) cannot be PEP, since there exists a larger vector u such that $g(u,v) \geq p$. Now assume another vector pair (y,z) that is a PEP, such that $y \geq u$. For an arbitrary dimension k, where $y_k > u_k$, construct a new vector w such that $w_i = u_i, \forall i, i \neq k$ and $w_i = y_i$, $i = k$. We know $g(y,z) \geq p$ as (y,z) is a PEP. Since $y \geq u$ and the function $g(w,z)$ monotonically decreases in w, $g(w,z) \geq p$. Therefore, $l_k(u) \leq y_i$, which implies (y,z) is not a PEP, contradicting our assumption.

PEP $\Rightarrow$ Bounds. Follows from the definition, since if (u,v) is PEP, then there exists no larger vector y such that (y,v) is PEP and no smaller vector z such that (u,z) is PEP. $\square$

This bound is used to check if a candidate vector is PEP. The concept of the proposed enumeration algorithm is that instead of a linear traversal suggested by previous methods, we exploit the monotonic property of the cumulative distribution function (or $g(u,v)$) to allow us to focus on areas of the search space that contain the p-frontier where $g(u,v)=p$. The function $g(u,v)$ monotonically increases with v and monotonically decreases with u. The search space can be construed as a lattice, since the random vector takes only discrete values. Any arbitrary hyper-rectangle within the search space can be defined by two 'corner' points. The 'low' corner point, consisting of the smallest possible v component and the largest possible u component within the hyper-rectangle and is denoted by (u^s,v^s). The 'high' corner point consists of the largest v and the smallest u components within the hyper-rectangle and is denoted by (u^e,v^e). All lattice points within the hyper-rectangle are candidate PEPs. Due to the monotonic nature of $g(u,v)$, the lowest possible value within the hyper-rectangle is $g(u^s,v^s)$ and the highest possible value it can take is $g(u^e,v^e)$. Based on the two corner points, three cases may occur as illustrated in Figure 2.

Case 1. If $g(u^e,v^e)<p$, then the entire hyper-rectangle can be ignored, since it is guaranteed not to contain a PEP.

Case 2. If $g(u^s,v^s)>p$ then the hyper-rectangle is 'above' the p-frontier. The only possible PEP is the corner point (u^s,v^s). If the corner point is PEP, then it is the sole PEP in the hyper-rectangle, since it would dominate all other candidate solutions. If it is not PEP, then no other PEPs exist within the hyper-rectangle, since they would be dominated by (u^s,v^s).

Case 3. If $g(u^s,v^s)\le p\le g(u^e,v^e)$, then the hyper-rectangle may contain one or more PEPs and is marked for further exploration.

Large swaths of the search space can be disregarded quickly using Cases 1 and 2. When the hyper-rectangle is marked for further exploration (Case 3), it can be partitioned arbitrarily with the same cases applied recursively. With each iteration, the partitions get smaller until they can no longer be divided. The terminal partition is a hyper-rectangle with at most two lattice points along any dimension. For an n-dimensional random vector, enumeration of PEPs in the terminal partition could, in the worst-case, require examination of 2^{2n} candidate vector pairs. The PEPs in this case can be completely enumerated using existing enumeration schemes (Prékopa et al. 1998, Prékopa 2003, Beraldi and Ruszczyński 2002).

This procedure obviates the need for complete enumeration by focusing on areas of the search space that contain the p-frontier where $g(u,v)=p$. Only two evaluations of $g(u,v)$ are needed to

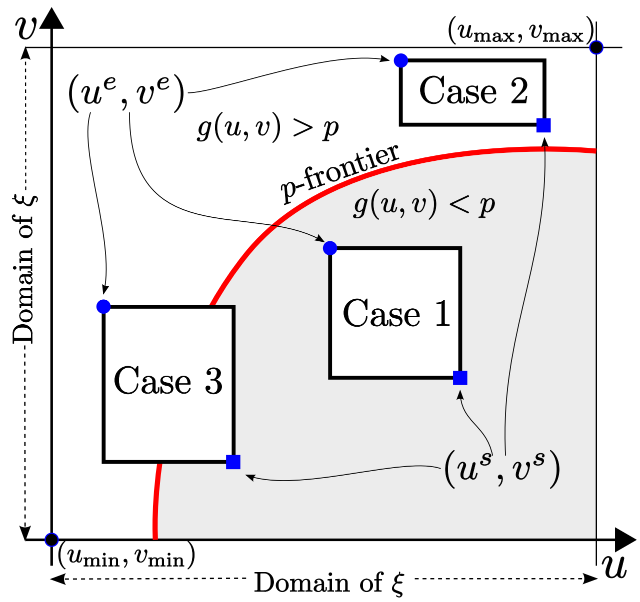


Figure 2 **Three cases for arbitrary hyper-rectangles in 2-dimensions**

determine if a candidate hyper-rectangle contains the p-frontier. Complete enumeration is saved for portions of the search space that are promising. The procedure has a small memory footprint, since the algorithm only keeps track of two vector pairs for each hyper-rectangle. While Prékopa's procedure (Prékopa et al. 1998) nests the search along the different dimensions of the random vector, here we nest in the domain of each component of the vector.

The recursive PEP enumeration algorithm is presented for a random vector $\xi = (\xi_1, \xi_2, \ldots, \xi_r)$. Each component of the random vector ξ_i can take values between l_i and u_i.

Step 0. **Initialization**. Define the starting corner vector pairs (u^s, v^s) and (u^e, v^e), where $u_i^s = v_i^e = u_i$ and $v_i^s = u_i^e = l_i$. Initialize the set of PEPs $S_p = \emptyset$ and p the desired probability level.

Step 1. **p-Frontier check**. For two vector pairs (u^s, v^s) (start) and (u^e, v^e) (end) if $g(u^s, v^s) \leq p \leq g(u^e, v^e)$ then proceed to Step 2, otherwise, terminate.

Step 2. **Partition**. Along an arbitrary dimension $k, k = 1, \ldots, 2r$, determine a scalar w_k that partitions the hyper-rectangle defined by (u^s, v^s) and (u^e, v^e) into two non-empty, non-overlapping hyper-rectangles. If no partition exists, then go to Step 4.

Step 3. **Recurse**. If $k \leq r$, then construct $s = (u_1^e, u_2^e, \ldots, w_k, \ldots, u_k^e)$. Go to Step 1 first with the vector pairs $(u^s, v^s), (s, v^e)$ and then again with the vector pairs $(s, v^s), (u^e, v^e)$. If $k > r$, then construct $s = (v_1^e, v_2^e, \ldots, w_k, \ldots, v_r^e)$ and go to Step 1 first with vector pairs $(u^s, v^s), (u^e, s)$ and then with $(u^s, s)(u^e, v^e)$.

Step 4. **Enumerate**. For each candidate vector pair (u, v) in the hyper-rectangle, compute the conditional bounds $l(u)$ and $h(v)$. If $l(u) = u$ and $l(v) = v$, add to set of PEPs $S_p = S_p \cup (u, v)$. Stop.

This procedure terminates with the set S_p required in the solution procedure of CCM-p (Section 3.1).

3.1.2. Reducing Computational Effort Conditions that provide a quick test on whether a PEP (u,v) will provide a sub-optimal solution (without solving the MIP) are derived. These conditions are based on the premise that the cost of a partial redistribution plan always exceeds that of a complete redistribution plan, since the phantom resources (the decision variables w_i and z_i) are utilized with a high penalty (γ). As the solution algorithm involves determining redistribution for a series of PEPs, if a complete plan has been found (that is not necessarily optimal), then all successive PEPs that lead to partial solutions can be safely ignored. A partial plan is used only when resources are outstripped by demand and operators cannot cover p-proportion of demand. These conditions utilize the physical signifance of a PEP pair (u,v). v_i represents the number of vehicles (if positive) needed at station i for the desired level-of-service and u_i, if negative, represents the required number of spaces.

At any station the total number of spaces and vehicles needed cannot exceed the capacity. This condition is termed the *capacity infeasibility* condition and can be expressed as

$$-\min(u_i,0)+\max(v_i,0)>c_i \qquad i=1,\ldots,n. \tag{14}$$

These capacity constraints are 'local', since they are applied to each station. There are 'global' *supply infeasibility* conditions when the available inventory in the system is insufficient for the operator to meet anticipated demand. These supply infeasibilities can be expressed as

$$\sum_{i=1}^{n} V_i < \sum_{i=1}^{n} \max(v_i,0) \qquad \text{and} \tag{15}$$

$$\sum_{i=1}^{n} (C_i - V_i) < -\sum_{i=1}^{n} \min(u_i,0). \tag{16}$$

Eq. (15) states that the total inventory of vehicles available across the network is exceeded by total anticipated demand across the entire network. Eq. (16) states the same principle for spaces.

When the model suggests partial redistribution, the system operates at reliability levels that are lower than the desired p. The true system reliability in this case can be computed as follows. Let (u^*,v^*) be the PEP for which the (partial) redistribution plan is optimal. Let w_i^* and z_i^* be the optimal values of the phantom resources. Then, the true system reliability $\hat{p}$ can be expressed as

$$\begin{aligned} \hat{p} &= \mathbb{P}\left(u_i^* + w_i^* \le \xi_i \le v_i^* - z_i^*,\, i=1,\ldots,n\right) \\ &= g(u^*+w^*, v^*-z^*) \end{aligned} \tag{17}$$

3.2. A Cone Generation Method

When demand is assumed to be independent, the cone generation method proposed by Dentcheva et al. (2000, 2002) can be employed. The method can generate redistribution plans quickly and is suitable for large systems where PEP enumeration is prohibitive. The master problem is an approximation of CCM-p and the subproblem generates p-efficient points as needed. The basic idea is similar to column generation where each PEP can be viewed as a column. The solution algorithm presented here mirrors the procedure of Dentcheva et al. (2000, 2002) but proposes a new subproblem to deal with a dual-bounded chance constraint.

Algorithm CGM-p:

Step 0. **Initialization**. Choose an arbitrary starting PEP pair (u^0, v^0) and set $J_0 = 0$, $k = 0$.

Step 1. **Master problem**. Solve the convexified linear program

$$\min \sum_{i=1}^{n}\sum_{j=1}^{n}(a_{ij}x_{ij} + \delta y_{ij}) + \sum_{i=1}^{n}\gamma\,(w_i + z_i)\,, \tag{18}$$

$$V_i + \sum_{j=1}^{n}(y_{ji} - y_{ij}) + w_i \geq \sum_{j\in J_k}\mu_j v_i^j \qquad i = 1,\dots,n, \tag{19}$$

$$C_i - V_i + \sum_{j=1}^{n}(y_{ij} - y_{ji}) + z_i \geq -\sum_{j\in J_k}\mu_j u_i^j \qquad i = 1,\dots,n, \tag{20}$$

$$\sum_{j\in J_k}\mu_j = 1, \tag{21}$$

$$\sum_{j=1}^{n} y_{ij} \leq V_i \qquad i = 1,\dots,n, \tag{22}$$

$$\sum_{j=1}^{n} y_{ji} \leq C_i - V_i \qquad i = 1,\dots,n, \tag{23}$$

$$y_{ij} \leq M \cdot x_{ij} \qquad i = 1,\dots,n, j = 1,\dots,n, \tag{24}$$

$$x_{ij}, y_{ij}, w_i, z_j \geq 0 \qquad i = 1,\dots,n, j = 1,\dots,n, \tag{25}$$

$$\mu_j \geq 0 \qquad j \in J_k \tag{26}$$

Let λ_v^k and λ_u^k be the simplex multipliers of constraints (19) and (20) respectively.

Step 2. **Upper bound**. Calculate the bound for the dual function over the set of generated PEPs.

$$\bar{d}(u^k, v^k) = \min_{j\in J_k}(\lambda_v^k)^T v^j - (\lambda_u^k)^T u^j \tag{27}$$

Step 3. **Solve subproblem**. Find the PEP pair (u^{k+1}, v^{k+1}) by solving

$$\min\left\{(\lambda_v^k)^T v^{k+1} - (\lambda_u^k)^T u^{k+1} \mid g(u^{k+1}, v^{k+1}) \geq p\right\}, \tag{28}$$

and compute $d(u^k, v^k) = (v^{k+1})^T\lambda_v^k - (u^{k+1})^T\lambda_u^k$.

Step 4. **Termination Condition**. If $d(u^k, v^k) = \bar{d}(u^k, v^k)$ then stop; otherwise set $J_{k+1} = J_k \cup (k+1)$, $k = k+1$ and goto Step 1.

The subproblem in Step 3 has a nonlinear constraint $g(u, v) \geq p$. When demand at each station is assumed to be independent, this constraint can be written as

$$\ln(g(u^k, v^k)) = \sum_{i=1}^{n} \ln(g_i(u_i^k, v_i^k)) \geq \ln p \tag{29}$$

If each component of ξ takes values between l_i and b_i, the subproblem can formulated as an MIP. Denote y_{ijk} as a binary decision variable that is one if for the i-th dimension of ξ, $u_i = j$ and $v_u = k$ and zero otherwise. For a given set of multipliers $(\lambda_u^k, \lambda_v^k)$ and a desired probability level p, the subproblem can be written as

$$\min \sum_{i=1}^{n} \sum_{j=l_i}^{b_i} \sum_{k=l_i}^{b_i} (\lambda_i^v k - \lambda_i^u j)\, y_{ijk} \tag{30}$$

subject to

$$\sum_{i=1}^{n} \sum_{j=l_i}^{b_i} \sum_{k=l_i}^{b_i} \ln(q_{ijk}) y_{ijk} \geq \ln(p) \tag{31}$$

$$\sum_{j=l_i}^{b_i} \sum_{k=l_i}^{b_i} y_{ijk} = 1 \qquad i = 1, \ldots, n \tag{32}$$

$$y_{ijk} \in [0, 1], \tag{33}$$

where $q_{ijk} = g_i(j, k) = \mathbb{P}(j \leq \xi_i \leq k)$.

4. A Failure Apportionment Bound

If the systemwide reliability level can be translated to a component-level measure, the joint chance constraint can be decoupled to give linear constraints. These constraints provide a bound on the original problem. The VSP stations can be viewed as being ‘in series’, since the unserviced demand at any station implies lower reliability. A system that is p-reliable has an acceptable failure rate of at most $1-p$. The Boole-Bonferroni inequality (Prékopa 1995) implies that the sum of the station failure rates cannot exceed the systemwide failure rate.

$$\sum_{i=1}^{n} (1 - p_i) \leq 1 - p. \tag{34}$$

Under an equal apportionment of failure we have $1 - p_i = (1-p)/n$. Decoupling the joint chance constraint (6) results in n joint constraints:

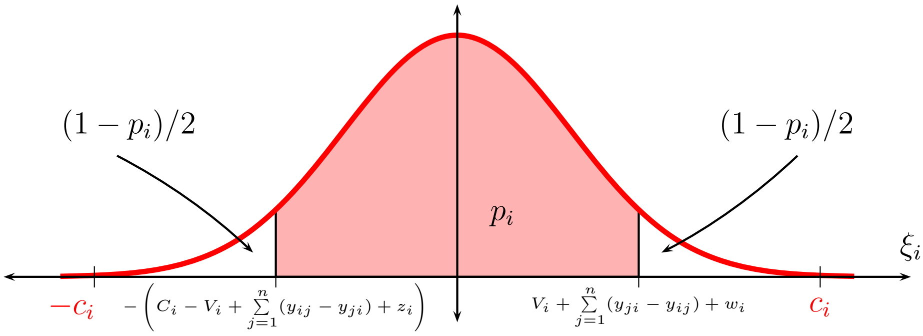


Figure 3 **Demand scenarios covered by the chance constraint at one station**

$$\mathbb{P}\begin{pmatrix} -\left[C_i - V_i + \sum_{j=1}^{n}(y_{ij}-y_{ji}) + z_i\right] \leq \xi_i \\ V_i + \sum_{j=1}^{n}(y_{ji}-y_{ij}) + w_i \geq \xi_i \end{pmatrix} \geq p_i \qquad i = 1,\ldots,n, \tag{35}$$

where $p_i = (n-1+p)/n$. This set of n joint chance constraints can be further reduced by allowing the acceptable failure rate to be divided amongst unserviced demand for vehicles and unserviced demand for spaces. This is shown in Figure 3.

This results in $2n$ chance constraints:

$$\mathbb{P}\left(V_i + \sum_{j=1}^{n}(y_{ji}-y_{ij}) + w_i \geq \xi_i\right) \leq \frac{1-p_i}{2} \qquad i = 1,\ldots,n, \tag{36}$$

$$\mathbb{P}\left(-\left[C_i - V_i + \sum_{j=1}^{n}(y_{ij}-y_{ji}) + z_i\right] \leq \xi_i\right) \leq \frac{1-p_i}{2} \qquad i = 1,\ldots,n. \tag{37}$$

In terms of the inverse marginal distribution, the constraints (36) and (37) can be derived as

$$V_i + \sum_{j=1}^{n}(y_{ji}-y_{ij}) + w_i \geq F_{\xi_i}^{-1}\left(\frac{1+p_i}{2}\right) \qquad i = 1,\ldots,n, \tag{38}$$

$$-\left[C_i - V_i + \sum_{j=1}^{n}(y_{ij}-y_{ji}) + z_i\right] \leq F_{\xi_i}^{-1}\left(\frac{1-p_i}{2}\right) \qquad i = 1,\ldots,n. \tag{39}$$

The solution to the MIP defined by the objective (5) and constraints (7), (8), (9), (10), (11), (38), (39) provides a bound on the optimal solution.

5. Application

The Intelligent Community Vehicle System (ICVS) operated by the Honda Motor company in Singapore City, Singapore was a car-sharing system with 14 stations mainly in the downtown region and one at the Changi Airport. The system was also studied by Kek et al. (2009). The program is no longer operational. Data from the program available from March 2003 through January 2006 documents 45,570 trips. Across the 14 stations, the system had an assumed capacity of 202 spaces, with 94 vehicles spread around the network. The characteristics of the fleet are not known and are assumed to be homogeneous. The trip characteristics are summarized in Figures 4 and 5.

The realized demand process is assumed to be the true demand process. This implies that extreme demand scenarios are not represented in the inputs (since they are never observed). The demand-supply interaction is ignored and treated as exogenous and inelastic. Each day is divided into four time periods when redistribution is considered. During each time period at each station, the number of vehicles checked out and the number returned were found to be Poisson distributed. Of all 112 input distributions (one for each station and each period during the day for checkouts and returns), 28 failed the χ^2 test due to the low number of observations. Sample distributions for two stations are shown in Figure 6.

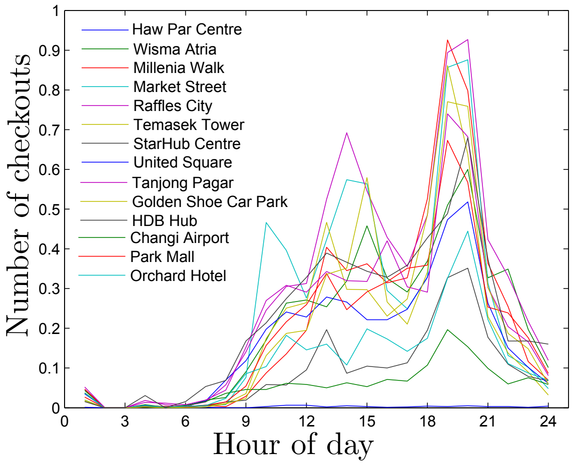
For each station j and time period t, the Poisson vehicle checkout rate (λ^1_{tj}) and the Poisson vehicle return rate (λ^2_{tj}) are determined. Since the random variable ξ_i in program CCM-p is the difference of the two, the distribution of the difference is needed. When two random variates are Poisson distributed with means λ^1_{tj} and λ^2_{tj}, their difference ξ_{tj} is Skellam distributed with pmf

$$\mathbb{P}(\xi_{tj} = k) = e^{-(\lambda^1_{tj}+\lambda^2_{tj})} \left(\frac{\lambda^1_{tj}}{\lambda^2_{tj}}\right)^{k/2} I_k(2\sqrt{\lambda^1_{tj}\lambda^2_{tj}}), \tag{40}$$

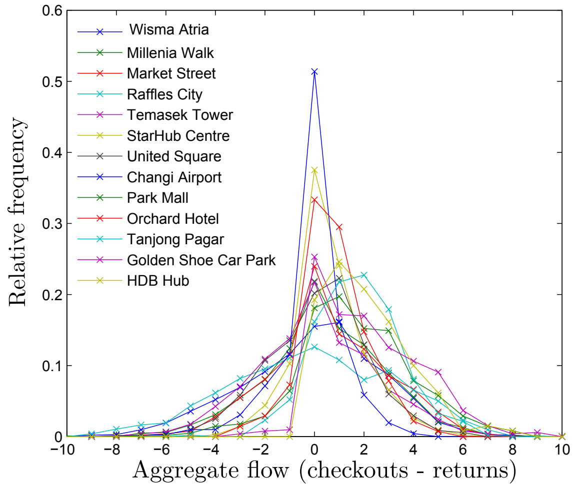
where $I_k(z)$ is the modified Bessel function of the first kind. Since this is a discrete distribution, $F^{-1}_{\xi}(p)$ exists when we define the inverse function as the infimum and when $0 < p < 1$. Figure 7 shows the Skellam distribution for some sample values.

5.1. Computational Experiments

Nine strategies for redistribution are tested. A do-nothing (DN) approach is when operators relocate only once before the start of each day but do not redistribute during the day. An expected value approach (AVG) generates redistribution based on the expected value of demand. The PEP enumeration based solution approach (ENUM-p) is used to generate strategies for three values of p (0.8, 0.9, and 1.0). Since the support of the Skellam distribution is $(-\infty, \ldots, -1, 0, 1, \ldots, \infty)$, for the last case p is very close to, but not equal, to 1.0. The cone generation method (CGM-p) is run for $p = 0.8, 0.9$. The failure apportionment bounds (FAB-p) presented in Section 4 are computed for two values of p: 0.8 and 0.9. The proposed procedures (DN, AVG, ENUM-p, CGM-p, FAB-p) were implemented in MATLAB 2009a, java, and CPLEX 11.2. All experiments were run on an Intel Xeon processor running at 3.00 GHz with 16GB of RAM. Since for ENUM-p, the PEP enumeration is needed only once, the PEP enumeration procedure was allowed to run for 24 hours for each stage

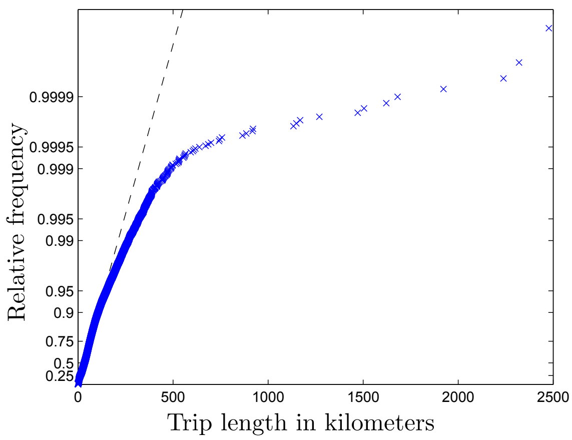


(a)Average Number of Checkouts

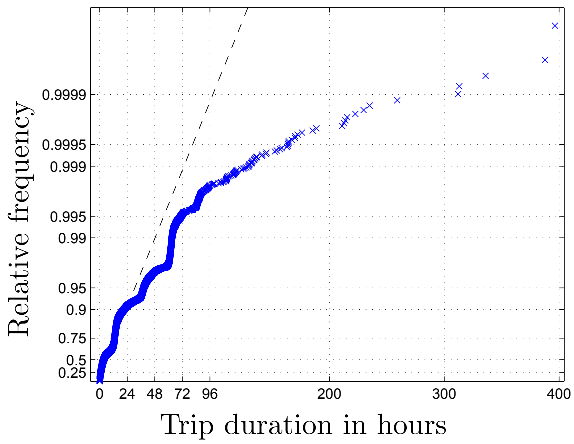


(b)Distribution of Daily Imbalance

Figure 4 **Characteristics of the IVCS Singapore System**

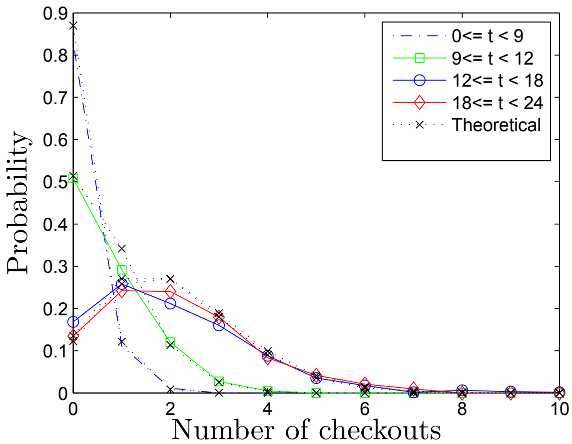


(a)Distribution of Trip Distance

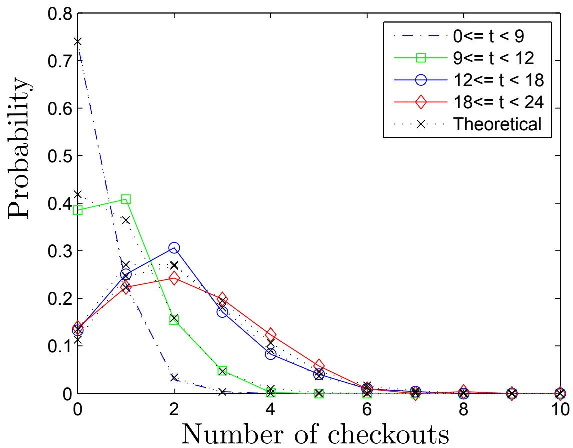


(b)Distribution of Trip Duration

Figure 5 **Trip Characteristics**



(a)Station 10



(b)Station 12

Figure 6 **Actual and Theoretical Demand Distributions for 2 Stations**

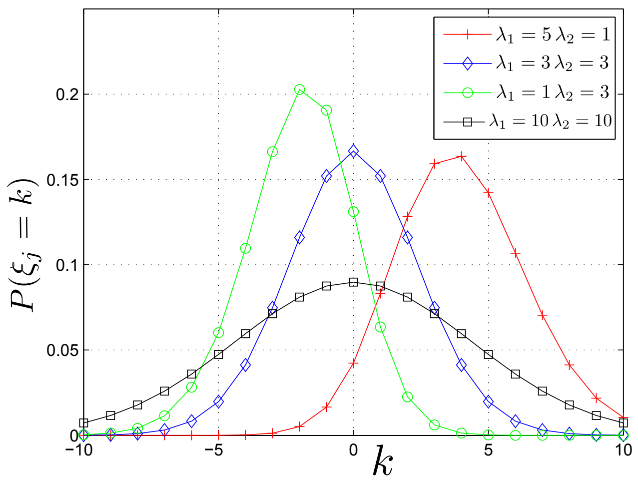


Figure 7 The Skellam distribution function for sample λ combinations.

State	Time Period (hrs)	# PEPs $p=0.8$	# PEPs $p=0.9$
1	$0 \leq t < 9$	50.10	9.41
2	$9 \leq t < 12$	37.60	22.68
3	$12 \leq t < 18$	64.60	173.80
4	$18 \leq t < 24$	77.20	202.00

Table 1 Number of PEPs for each stage (in millions)

using $p = 0.8$ and 0.9 (for $p = 1.0$, there is only one PEP). The number of PEPs generated for each period are shown in Table 1.

Redistribution is performed at the start of each time period. A demand scenario is randomly generated and the fleet inventories are adjusted to serve as the start point for the next time period. Since the set of PEPs is extremely large, the run time for ENUM-p experiments at each time period was restricted to one hour. Consequently, the solutions presented next are not guaranteed to be optimal. Since demand at each station is assumed to be independent, the results from CGM-p (where independence is forced) can be directly compared with other strategies. The main measures of effectiveness are actualized or true reliability, relocation cost (without the penalties for phantom resources, since this is what operators will experience) and the number of dropped demands.

Results are presented for a 2-day period. Figures 8 and 9 show the actualized reliability $\hat{p}$ versus the relocation costs for the various time periods and various strategies. The relocation costs are computed as the value of the objective function minus the penalties for using the phantom resources. For ENUM-p methods, the family of solutions is also depicted.

When resources are adequate (for example, Figures 8(a) and 8(b)), the CGM-p methods with $p =$ 0.8 or 0.9 provide the best actualized reliability through the use of redistribution. When inventories are low (Figure 8(d)), a complete redistribution plan is only achieved by the CGM-0.8 strategy.

Under these circumstances, all other solutions show drops in true reliability and represent partial redistribution plans. For the same reliability level p, CGM methods outperform ENUM since all PEPs are not explored for ENUM and optimality not guaranteed. When the plans are complete, the FAB-p solutions cost more than the ENUM-p and CGM-p strategies (for a comparable p). When the plans are partial, this need not be the case, since the penalty costs are not included.

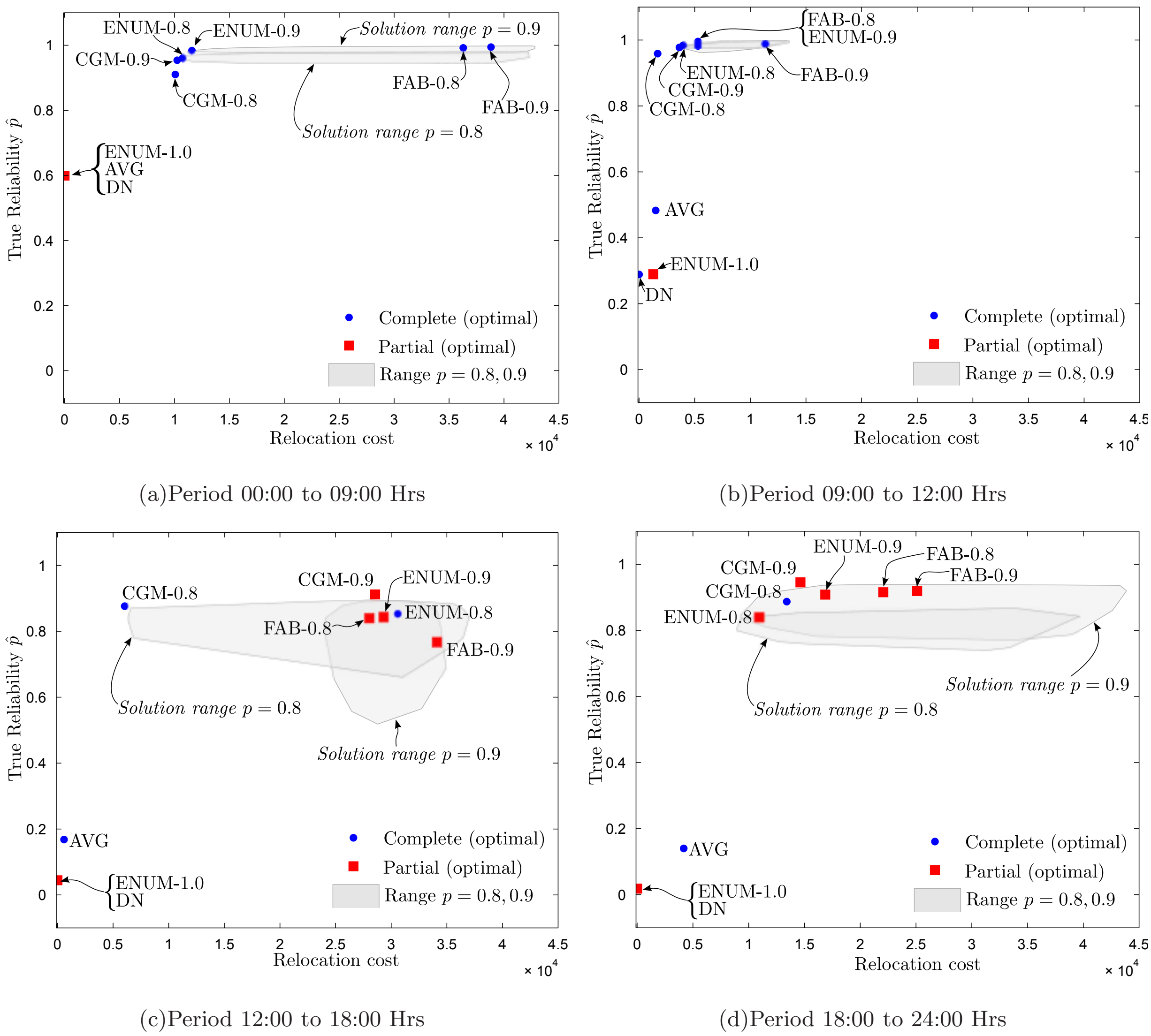
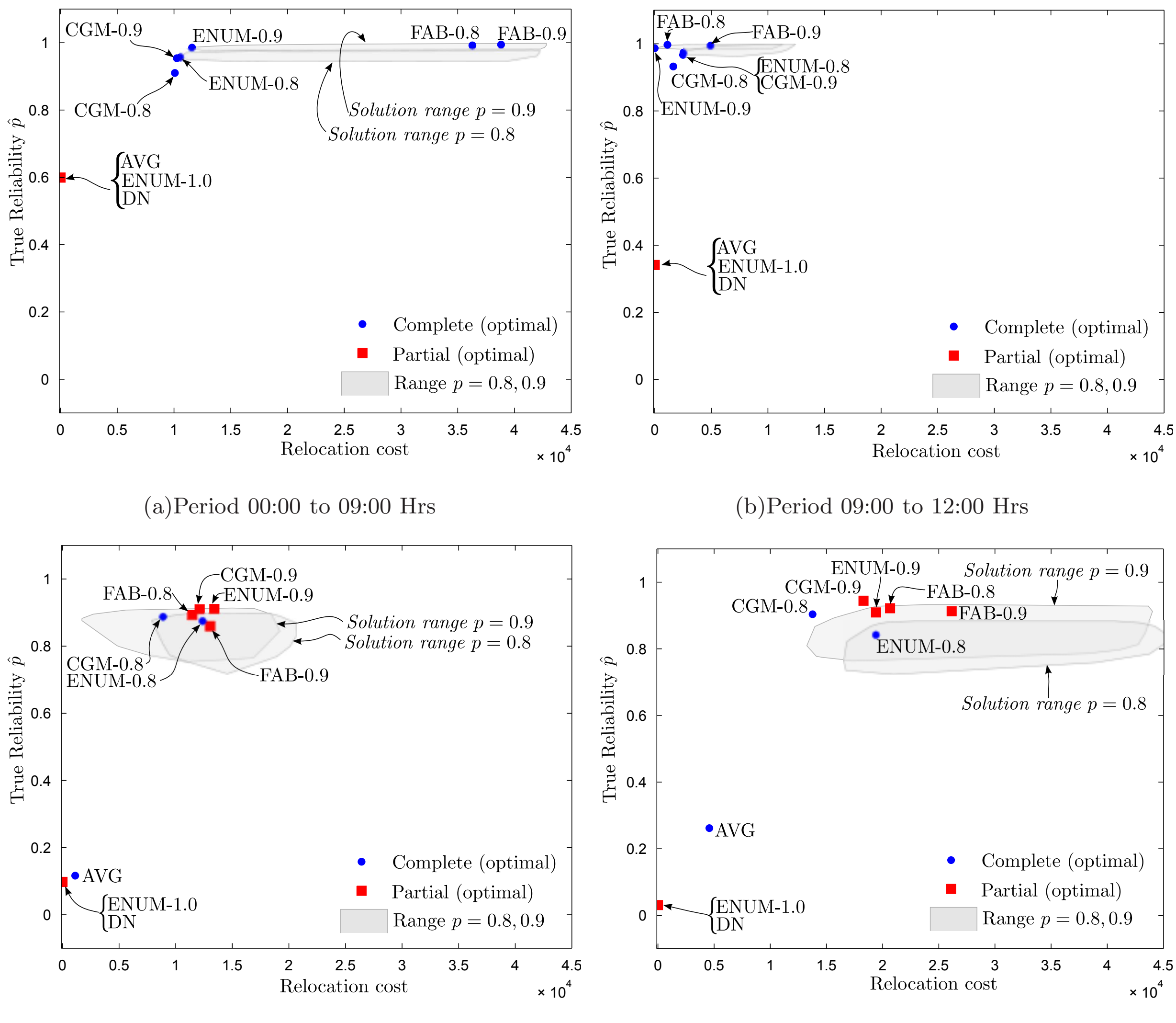


Figure 8 **Relocation strategies for various states: Day 1**

Figure 10 shows the number of vehicles redistributed for the same two-day period for various states. The FAB strategies are generally more expensive to implement, since they require a greater number of relocations.

To determine the value of considering stochasticity in generating redistribution plans, simulation is employed. The strength of a particular system configuration can be tested over a range of demand



(a)Period 00:00 to 09:00 Hrs (b)Period 09:00 to 12:00 Hrs

(c)Period 12:00 to 18:00 Hrs (d)Period 18:00 to 24:00 Hrs

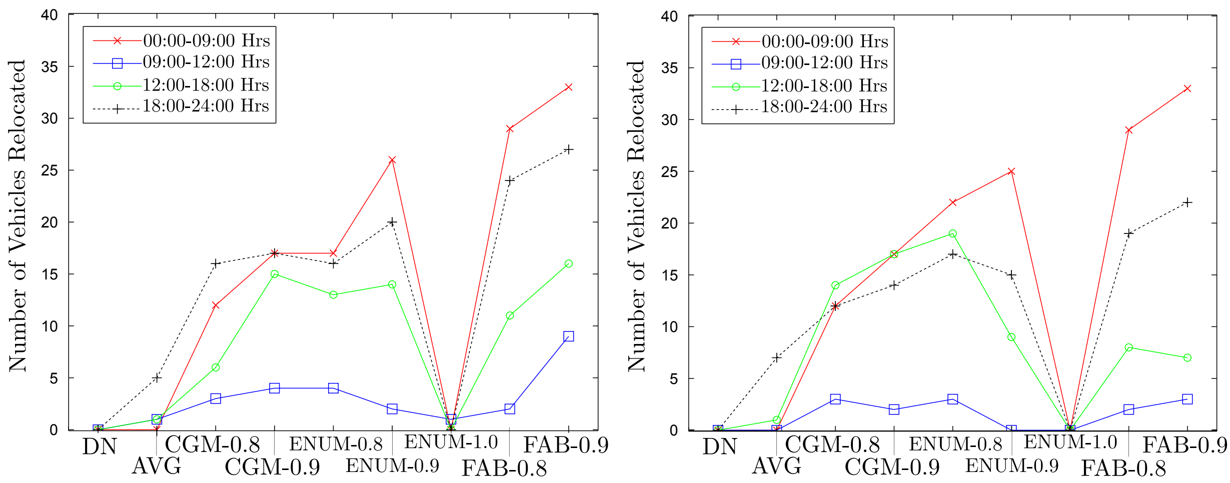
Figure 9 Relocation strategies for various states: Day 2

realizations. The number of unserviced users, or dropped demand, is measured for each realization. Given a random realization of demand $\bar{\xi}$, the dropped demand for vehicles d_v and spaces d_s can be computed as

$$d_v = \sum_{i=1}^{n} \max(0, \bar{\xi}_i - V_i), \tag{41}$$

$$d_s = -\sum_{i=1}^{n} \min(0, \bar{\xi}_i + C_i - V_i). \tag{42}$$

These quantities are computed for all the strategies and time periods for two days over 100,000 realizations. Results are summarized in Tables 2 and 3. The ENUM-0.8, ENUM-0.9, and CGM-p strategies do not drop any demand for a high proportion of realizations regardless of resource



(a)Day 1 (b)Day 2

Figure 10 Number of Vehicles relocated

availability. When resources are adequate, at the start of the day, all strategies do well. During later time periods, the AVG, ENUM-1.0, and DN approaches are more likely to leave demand unserviced. The FAB strategies perform as well as ENUM, but these are more expensive to implement. Tables 2 and 3 also shows the worst-case demand realization for each algorithm. The ENUM-0.8 and ENUM-0.9 consistently drop fewer number of demand requests for vehicles and spaces compared to other methods even in the worst-case realization, indicative of robustness of the redistribution strategy.

The ENUM-1.0 performs poorly in most scenarios. Since the resource requirements to satisfy such a high level-of-service are extraordinary, the redistribution plan that the model achieves in this case is always partial, since phantom variables must be utilized. The actualized reliability achieved for this case is very low as demonstrated by the simulation experiments. The solutions yield additional insights on system characteristics. Since the PEPs have a physical interpretation, for the PEP that resulted in the best known solution, the system resource requirements can be computed (see Table 4). These numbers directly relate the desired level-of-service with the resources. For example, in Day 1, to achieve a systemwide reliability of 0.9 requires 77 vehicles during the 18:00-24:00 Hours (Hrs) time period for the CGM and ENUM methods. A reliability of 0.8 requires eight fewer vehicles for the same period. These values are contingent on starting inventories, thus, their purpose is illustrative. In a similar vein, the stations which have frequent local infeasibilities can be the target of capacity improvements, since local infeasibilities indicate recurrent imbalance in flows. Since only a subset of PEPs were solved to determine the ENUM-p strategies, the solutions for the ENUM-0.8 and ENUM-0.9 strategies are not guaranteed to be optimal in these experiments.

Time (Hrs)	Algorithm	Obj. Value	Partial Redistribution	Probability of no dropped vehicle demands	Average number of dropped vehicle demands	Number of dropped vehicle demands for worst-case	Probability of no dropped spaces	Average number of dropped space demands	Number of dropped space demands for worst-case
0-9	DN	0.00	-	0.998	0	2	1	0	3
	AVG	0.00	-	0.998	0	2	1	0	3
	FAB-0.9	38817.33	No	0.998	0	3	1	0	2
	FAB-0.8	36291.29	No	1	0	2	1	0	2
	ENUM-1.0	11340000	Yes	0.998	0	2	1	0	3
	ENUM-0.9	11590.26	No	1	0	3	1	0	3
	ENUM-0.8	10749.17	No	1	0	2	1	0	3
	CGM-0.9	10250.17	No	0.999	0	2	1	0	3
	CGM-0.8	10077.12	No	0.999	0	2	0.999	0	3
9-12	DN	0.00	-	0.487	0.8	8	0.391	1.16	9
	AVG	1495.01	-	0.754	0.34	7	0.423	1.06	9
	FAB-0.9	11335.09	No	0.988	0.01	4	0.99	0.01	4
	FAB-0.8	5294.02	No	0.997	0	3	0.986	0.02	4
	ENUM-1.0	10431279.00	Yes	0.487	0.8	8	0.391	1.16	9
	ENUM-0.9	3626.02	No	0.985	0.02	4	0.97	0.04	5
	ENUM-0.8	3959.04	No	0.998	0	3	0.822	0.24	6
	CGM-0.9	5294.04	No	0.99	0.01	4	0.944	0.07	5
	CGM-0.8	1668.03	No	0.989	0.01	4	0.817	0.24	6
12-18	DN	0.00	-	0.121	2.94	16	0.219	2.42	16
	AVG	592.01	-	0.323	1.66	14	0.328	1.75	14
	FAB-0.9	1294089.00	Yes	0.773	0.4	11	0.882	0.18	8
	FAB-0.8	728025.10	Yes	0.854	0.24	10	0.829	0.27	9
	ENUM-1.0	23240000.00	Yes	0.121	2.94	16	0.226	2.38	16
	ENUM-0.9	519310.10	Yes	0.852	0.24	10	0.773	0.37	9
	ENUM-0.8	30574.13	No	0.866	0.21	7	0.835	0.26	9
	CGM-0.9	2028554.00	Yes	0.954	0.06	7	0.71	0.5	10
	CGM-0.8	6043.06	No	0.934	0.09	7	0.69	0.53	10
18-24	DN	0.00	-	0.022	4.86	18	0.024	4.88	20
	AVG	4175.05	-	0.156	2.68	15	0.22	2.34	17
	FAB-0.9	795102.30	Yes	0.924	0.11	7	0.477	1.12	13
	FAB-0.8	372069.20	Yes	0.919	0.12	7	0.48	1.1	13
	ENUM-1.0	21350000.00	Yes	0.022	4.86	18	0.026	4.78	20
	ENUM-0.9	226865.20	Yes	0.912	0.13	7	0.473	1.13	13
	ENUM-0.8	80984.16	Yes	0.841	0.25	9	0.602	0.75	11
	CGM-0.9	1014639.00	Yes	0.950	0.07	7	0.398	1.39	14
	CGM-0.8	13413.16	No	0.906	0.13	8	0.458	1.14	12

Table 2 Dropped demand for spaces and vehicles in 100,000 simulation runs for Day 1

CGM-p strategies on the other hand are optimal since demand at each station is assumed to be independent.

In summary, a comparison of fleet management strategies based on stochastic assumptions offers greater reliability than plans based on static methods. Redistribution strategies based on the proposed stochastic MIP also weather scenarios in which demand outstrips supply. In simulation studies, the CGM-p, ENUM-0.9 and ENUM-0.8 strategies demonstrate robustness over all sampled demand scenarios.

Time (Hrs)	Algorithm	Obj. Value	Partial Redistribution	Probability of no dropped vehicle demands	Average number of dropped vehicle demands	Number of dropped vehicle demands for worst-case	Probability of no dropped spaces	Average number of dropped space demands	Number of dropped space demands for worst-case
0-9	DN	0.00	-	0.998	0	3	1	0	2
	AVG	0.00	-	0.998	0	3	1	0	2
	FAB-0.9	38817.33	No	0.998	0	3	1	0	1
	FAB-0.8	36291.29	No	1	0	1	1	0	1
	ENUM-1.0	11340000.00	Yes	0.998	0	3	1	0	2
	ENUM-0.9	11590.25	No	1	0	1	1	0	2
	ENUM-0.8	10575.22	No	0.999	0	3	1	0	2
	CGM-0.9	10250.17	No	0.999	0	3	1	0	2
	CGM-0.8	10077.12	No	0.999	0	3	0.999	0	2
9-12	DN	0.00	-	0.965	0.04	4	0.332	1.33	11
	AVG	0.00	-	0.965	0.04	4	0.332	1.33	11
	FAB-0.9	4950.03	No	0.995	0.01	3	0.998	0	3
	FAB-0.8	1125.02	No	0.999	0	3	0.989	0.01	4
	ENUM-1.0	10430000.00	Yes	0.965	0.04	4	0.332	1.33	11
	ENUM-0.9	0.00	No	0.999	0	2	0.975	0.03	5
	ENUM-0.8	2542.03	No	0.999	0	2	0.979	0.02	4
	CGM-0.9	2501.02	No	0.991	0.01	4	0.958	0.05	5
	CGM-0.8	1641.03	No	0.989	0.01	4	0.908	0.12	6
12-18	DN	0.00	-	0.494	1.05	11	0.089	3.95	21
	AVG	1125.01	-	0.495	1.05	11	0.115	3.5	20
	FAB-0.9	1273059	Yes	0.896	0.16	7	0.846	0.24	9
	FAB-0.8	711480.10	Yes	0.947	0.08	7	0.78	0.36	10
	ENUM-1.0	23240000.00	Yes	0.494	1.05	11	0.089	3.95	21
	ENUM-0.9	573418.10	Yes	0.946	0.08	6	0.789	0.35	10
	ENUM-0.8	12392.19	No	0.888	0.17	8	0.841	0.25	10
	CGM-0.9	3012119	Yes	0.963	0.05	6	0.671	0.59	11
	CGM-0.8	8908.14	No	0.955	0.06	6	0.632	0.68	11
18-24	DN	0.00	-	0.052	3.76	20	0.005	7.28	27
	AVG	4577.07	-	0.33	1.6	16	0.083	3.8	23
	FAB-0.9	656147.20	Yes	0.917	0.12	10	0.485	1.09	15
	FAB-0.8	300664.20	Yes	0.927	0.11	9	0.4	1.39	17
	ENUM-1.0	21350000.00	Yes	0.052	3.76	20	0.005	7.28	27
	ENUM-0.9	229417.20	Yes	0.914	0.12	10	0.536	0.94	15
	ENUM-0.8	19441.17	No	0.844	0.24	11	0.629	0.69	13
	CGM-0.9	1018299.00	Yes	0.952	0.07	9	0.395	1.42	17
	CGM-0.8	13766.12	No	0.91	0.13	10	0.501	1.04	16

Table 3 Dropped demand for spaces and vehicles in 100,000 simulation runs for Day 2

6. Conclusions

Fleet management strategies that explicitly consider demand stochasticity are developed for vehicle sharing systems. In developing these management strategies, no assumptions are made on the specific operational characteristics and demand processes of a particular system. However, system specific attributes can be incorporated with relative ease. For example, if a sharing system allows advance reservations of vehicles, then the demand process splits into a static known component and an uncertain one. By adjusting start inventories at each stage, the static portion can be guaranteed

Day 1		Required Vehicles				Required Spaces				Local Infeasibility			
Strategy	p	0-9	9-12	12-18	18-24	0-9	9-12	12-18	18-24	0-9	9-12	12-18	18-24
AVG	-	2.49	8.97	25.73	27.15	14.55	6.72	17.79	10.47	0	0	0	0
FAB	0.8	22	47	76	84	74	51	69	54	0	0	8	6
FAB	0.9	19	41	71	79	68	48	63	48	0	0	5	2
ENUM	1.0	137	149	245	240	161	141	228	207	13	13	13	13
ENUM	0.9	21	45	71	77	67	46	66	54	0	0	5	2
ENUM	0.8	20	41	62	69	61	48	62	51	0	0	0	0
CGM	0.9	17	39	77	77	53	32	51	34	0	0	0	0
CGM	0.8	14	36	63	69	46	26	41	28	0	0	0	0
Day 2		Required Vehicles				Required Spaces				Local Infeasibility			
Strategy	p	0-9	9-12	12-18	18-24	0-9	9-12	12-18	18-24	0-9	9-12	12-18	18-24
AVG	-	2.49	8.97	25.73	27.15	14.55	6.72	17.79	10.47	0	0	0	0
FAB	0.8	22	47	76	84	74	51	69	54	0	0	8	6
FAB	0.9	19	41	71	79	68	48	63	48	0	0	5	2
ENUM	1.0	137	149	245	240	161	141	228	207	13	13	13	13
ENUM	0.9	22	48	71	76	65	46	68	57	0	0	5	3
ENUM	0.8	21	45	63	70	60	44	63	52	0	0	0	0
CGM	0.9	17	39	76	77	53	32	53	34	0	0	0	0
CGM	0.8	14	36	63	69	46	26	41	28	0	0	0	0

Table 4 **Systemwide resource needs based on various strategies with the number of stations with local capacity infeasibilities**

service.

The main contributions of this work are in formulating the VSP fleet management problem as a stochastic MIP. The approach taken herein overcomes the limitations of prior works that assume static or known demand. The proposed framework quantifies the systemwide level-of-service offered based on a probabilistic characterization of demand. Two solution techniques, one based on enumeration and the other on cone generation, are presented. For the enumeration based technique, the PEP enumeration algorithm improves on existing tools by using a divide-and-conquer paradigm that is able to quickly eliminate areas of the search space that are guaranteed not to contain PEPs. Our technique has a smaller memory and computational footprint than previously proposed methods. Additionally, the concept of PEP is extended to include dual-bounded chance constraints. The second solution technique assumes independence of the random vector. An equal-failure apportionment bound is also derived. Under these limiting assumptions (independence or equal failure probability), exact solutions can be quickly obtained. In an application of the proposed framework to a system in Singapore, the operational strategies were found to be robust in simulation studies. Additionally, trade-offs between redistribution costs and level-of-service were explored.

Future work along this direction could relax some assumptions, namely immediate fleet relocation, incorporate staff availability to perform redistribution, and tackle heterogeneous fleets. One might also study the assignment and routing of relocation teams to carry out fleet redistribution.

Acknowledgments

The authors would like to thank Dr. Kevin Chou for graciously sharing data on the Singapore system. This research was partially funded through the Mid-Atlantic University Transportation Center (MAUTC). The first author was partially supported by the I-95 Corridor Coalition Fellowship. Comments from two anonymous referees have improved the quality of the report and are acknowledged.

References

Barnhart, Cynthia, Runa R. Schneur. 1996. Air network design for express shipment service. *Operations Research* **44**(6) 852–863.

Barth, M., M. Todd. 1999. Simulation model performance analysis of a multiple station shared vehicle system. *Transportation Research Part C* **7**(4) 237–259.

Barth, M., M. Todd, L. Xue. 2004. User-Based Vehicle Relocation Techniques for Multiple-Station Shared-Use Vehicle Systems. *Transportation Research Board Annual Conference CD-ROM, Washington, DC*.

Beraldi, P., ME Bruni, D. Conforti. 2004. Designing robust emergency medical service via stochastic programming. *European Journal of Operational Research* **158**(1) 183–193.

Beraldi, P., A. Ruszczyński. 2002. The probabilistic set-covering problem. *Operations Research* 956–967.

Britton, F. 2008. Public bike system inventory. URL `http://www.ecoplan.org/`.

Cervero, Robert, Nina Creedman, Madhav Pai, Muhammad Pohan. 2002. City carshare: Assessment of short-term travel-behavior impacts. Tech. rep., Institute of Urban and Regional Development, University of California, Berkely.

Dawn Haines, Ian Skinner. 2005. The marketing of mobility services. Tech. rep., Institute for European Environmental Policy.

DeMaio, P., J. Gifford. 2004. Will smart bikes succeed as public transportation in the United States? *Journal of Public Transportation* **7**(2) 1–15.

Dentcheva, D., A. Prékopa, A. Ruszczynski. 2000. Concavity and efficient points of discrete distributions in probabilistic programming. *Mathematical Programming* **89**(1) 55–77.

Dentcheva, D., A. Prekopa, A. Ruszczyński. 2002. Bounds for probabilistic integer programming problems. *Discrete Applied Mathematics* **124**(1-3) 55–65.

Erlanger, Steven. 2008. A new fashion catches on in Paris: Cheap bicycle rentals. The New York Times. URL `http://www.nytimes.com/2008/07/13/world/europe/13paris.html`.

Fusco, G., L. Tatarelli, M.P. Valentini. 2003. Last-mile, a procedure to set-up an optimized delivery scheme. Eiichi Taniguchi, Russell G. Thompson, eds., *Logistics Systems for Sustainable Cities: Proceedings of the 3rd International Conference on City Logistics*. Elsevier, 147–161.

Katzev, R. 2003. Car sharing: A new approach to urban transportation problems. *Analyses of Social Issues and Public Policy* **3**(1) 65–86.

Kek, A.G.H., R.L. Cheu, Q. Meng, C.H. Fung. 2009. A decision support system for vehicle relocation operations in carsharing systems. *Transportation Research Part E* **45** 149–158.

Kim, Daeki, Cynthia Barnhart, Keith Ware, Gregory Reinhardt. 1999. Multimodal express package delivery: A service network design application. *Transportation Science* **33**(4) 391–407.

Kuby, Michael J., Robert Gordon Gray. 1993. The hub network design problem with stopovers and feeders: The case of Federal Express. *Transportation Research Part A* **27A**(1) 1–12.

Matsuura, Masahiro. 2003. Bicycle sharing. Presentation for International Workshop on Social Decision Making Process for Energy Technology Introduction.

Prékopa, A. 1990. Dual method for the solution of a one-stage stochastic programming problem with random rhs obeying a discrete probability distribution. *Mathematical Methods of Operations Research* **34**(6) 441–461.

Prékopa, A. 1995. *Stochastic programming*. Kluwer.

Prékopa, A., B. Vizvári, T. Badics. 1998. Programming under probabilistic constraint with discrete random variable. *New trends in mathematical programming: homage to Steven Vajda* 235–256.

Prékopa, András. 2003. *Probabilistic Programming*, *Handbooks in Operations Research & Management Science*, vol. 10, chap. 5. Elsevier Science, 267–352.

Rodier, CJ, SA Shaheen, L. Novick. 2004. Improving bay area rapid transit (BART) district connectivity and access with the segway human transporter and other low speed mode devices research report. Tech. rep., UCB-ITS-PRR-2004-27. California PATH, University of California, Berkeley.

Rydén, Christian, Emma Morin. 2005. Mobility services for urban sustainability: Environmental assessment, moses deliverable d6.2. Tech. rep., Trivector Traffic, Sweden.

Saxena, A., V. Goyal, M.A. Lejeune. 2008. MIP reformulations of the probabilistic set covering problem. *Mathematical Programming* **121**(1) 1–31.

Shaheen, S.A., A.P. Cohen. 2007. Growth in worldwide carsharing: An international comparison. *Transportation Research Record* **1992**(-1) 81–89.

Shaheen, S.A., M. Meyn, K. Wipyewski. 2003. US Shared-Use Vehicle Survey Findings on Carsharing and Station Car Growth: Obstacles and Opportunities. *Transportation Research Record* **1841** 90–98.

Shaheen, S.A., C.J. Rodier. 2006. Easyconnect: Low-speed modes linked to transit planning project. Working Paper UCB-ITS-PWP-2006-7, California PATH, Institute of Transportation Studies, University of California, Berkeley. URL `http://database.path.berkeley.edu/imr/papers/PWP-2006-07.pdf`.

Shaheen, S.A., C.J. Rodier, J. Seelig. 2006. Easyconnect ii: Integrating transportation, information, and energy technologies at transit oriented developments. *Proceedings the Transportation Research Board Annual 2006 Meeting* .

Shaheen, S.A., A. Schwartz, K. Wipyewski. 2004. Policy considerations for carsharing and station cars: Monitoring growth, trends, and overall impacts. *Transportation Research Record* **1887** 128–136.

Shaheen, Susan A., Caroline J. Rodier, Joshua Seelig. 2005. Easyconnect II: Integrating transportation, information and energy technologies at transit oriented developments. Tech. Rep. UCD-ITS-RR-05-14, Institute of Transportation Studies, University of California, Davis.

Taniguchi, Eiichi, Michihiko Noritake, Tadashi Yamada, Toru Izumitani. 1999. Optimal size and location planning of public logistics terminals. *Transportation Research Part E* **35** 207–222.

TRB. 2005. Transit cooperative research program synthesis 108: Car sharing: Where and how it succeeds. Tech. rep., Transportation Research Board.

USDOT. 2001. 2001 national household travel survey. Tech. rep., U.S. Department of Transportation.

Wray, J. Harry. 2008. *Pedal Power: The Quiet Rise of the Bicycle in American Public Life*. Paradigm Publishers, Boulder.