Theoretical Maximum Capacity as a Benchmark for Empty Vehicle Redistribution in Personal Rapid Transit

John D. Lees-Miller
Department of Engineering Mathematics,
University of Bristol, Queen's Building, University Walk,
Bristol, BS8 1TR, United Kingdom
enjdlm@bristol.ac.uk
(corresponding author)

John C. Hammersley
Advanced Transport Systems Ltd.
Unit B3, Ashville Park, Short Way
Thornbury, Bristol, BS35 3UU, United Kingdom
johnhammersley@atsltd.co.uk
Tel: +44(0) 1454 414700, Fax: +44(0) 1454 414770

R. Eddie Wilson
Department of Engineering Mathematics,
University of Bristol, Queen's Building, University Walk,
Bristol, BS8 1TR, United Kingdom
RE.Wilson@bristol.ac.uk
Tel.: +44(0) 117 331 5627, Fax.: +44(0) 117 331 5606

5673 words + 6 figures

ABSTRACT
A Personal Rapid Transit (PRT) system uses compact, computer-guided vehicles running on dedicated guideways to carry individuals or small groups directly between pairs of stations. Vehicles move *on demand* when a passenger requests service at his/her origin station. Because the number of trips requested from a station need not equal the number of trips ending there, some vehicles must run empty to balance the flows. The *empty vehicle redistribution* (EVR) problem is to decide which empty vehicles to move, and when and where to move them; an *EVR algorithm* makes these decisions in real time, as passengers arrive and request service. This paper describes a method for finding the theoretical maximum demand (with a given spatial distribution) that a given system could serve with *any* EVR algorithm, which provides a benchmark against which particular EVR algorithms can be compared. The maximum passenger demand that a particular EVR algorithm can serve can be determined by simulation and then compared to the benchmark. The method is applied to two simple EVR heuristics on two example systems, and the results suggest that this is a useful method for determining the strengths and weaknesses of a variety of EVR heuristics across a range of networks, passenger demands and fleet sizes.

1. INTRODUCTION

A Personal Rapid Transit (PRT) system provides on-demand, non-stop travel with compact, computer-guided vehicles running on a dedicated network of guideways. Each PRT vehicle carries either an individual or a small party traveling together by choice. The vehicle begins its trip *on demand*, when a passenger arrives at a PRT station (Figure 1). The network that connects these stations is usually built by connecting many short, one-way loops (as will be seen later in Figure 2). Once the passenger is ready to depart, his vehicle takes the quickest path to the chosen destination station, and it does not stop at intermediate stations to let other passengers on or off. Hence, a PRT system is similar to a taxi system, except that PRT vehicles are constrained to start and end their journeys at stations.

Because the number of trips requested from a particular station need not equal (on average or instantaneously) the number of trips ending at that station, a PRT system must be able to move empty vehicles to balance the movement of occupied ones. The question of which vehicles to move, and when and where to move them, is known as the *empty vehicle redistribution* (EVR) problem. The optimal solution of the EVR problem allows a given passenger demand to be met with the minimum number of vehicles (important to the PRT system designer) or, conversely, the maximum level of service to be provided with a given number of vehicles (important to the PRT system operator). Here, *level of service* refers to passenger waiting time as measured at its mean or at some percentile. Note that as passenger demand increases, service levels can generally be maintained by using more vehicles, but this increases cost, both directly through the provision of vehicles and indirectly through the extra station and guideway infrastructure required to accommodate them.

While a PRT system is operating, an EVR algorithm must move individual empty vehicles in real time to serve waiting passengers, or in anticipation of future passenger arrivals. It is important to emphasize that the particular passenger arrival times and locations are not known in advance; historical demand averages may be known, but the actual arrivals are revealed as the system operates. EVR algorithms often involve decision rules (*1*, ch. 5.7; *2*; *3*; *4*) to be executed when a new passenger arrives or a vehicle finishes service. For example, if a station is short of empty vehicles, a vehicle is chosen according to the rules and sent there; this may simply be the nearest empty vehicle at a station that is not itself short of vehicles (*2*), or it may be determined by solving an assignment problem (*4*). Similar methods using repeated assignment problems also appear in the literature for *full truckload motor carriers* (*5*) and related *dynamic pickup and delivery problems*; see (*6*) for a recent survey. In the context of taxi operations, (*7*) describes algorithms based on dynamic programming, one of which is detailed in section 5. An alternative approach is to work in the *fluid limit*, at the level of long-run averages of vehicle flows rather than individual vehicle movements (*1*, ch. 5.6; *8*, pp. 67–76). The fluid limit problem is tractable, but the resulting average flows do not prescribe an algorithm for solving the real-time problem. While several EVR algorithms have been proposed, it appears that little is known about their merits, either relative to one another or in absolute terms.

In this paper, the fluid limit is used to analyze the *capacity region* of a PRT system. This analysis is motivated by queueing theory, where the capacity region is also known as the *stability region*. The capacity region is defined to be the set of passenger demands that a system can serve whilst keeping the long-run average number of waiting passengers (and their waiting times) finite. When the passenger demand is outside of the capacity region, both the number of waiting passengers and their waiting times grow indefinitely – that is, passengers arrive more quickly

than they can be served. Necessary conditions for a demand to be in the capacity region are described in section 3, under the modeling assumptions in section 2; it is not yet known whether these conditions are sufficient, however.

The main focus of this paper is on using the capacity region to benchmark EVR algorithms. The capacity region describes the maximum possible demand (section 4) that a particular system could serve with any EVR algorithm. The algorithm to be benchmarked is first implemented in simulation. For any fixed demand, the long run average number of waiting passengers in the simulation will either converge to a finite number or grow indefinitely; so, the maximum demand that the system can serve is measured by running simulations with progressively more intense demands, until the number of waiting passengers begins to diverge. An algorithm can thus be evaluated by the fraction of the theoretical maximum demand that it can actually serve. To illustrate this approach, two simple heuristics (section 5) are simulated. The results for two example systems (section 6) are then presented in section 7. In short, the key innovation is that the capacity region is used to benchmark EVR heuristics; comparison to this benchmark determines whether an EVR algorithm delivers optimal throughput, and, if not, it quantifies the potential for improvement.

2. MODELLING ASSUMPTIONS

The focus of this paper is on capacity constraints due to the size of the vehicle fleet. In light of this, other mechanisms that limit capacity are neglected. The consequent simplifications come in two distinct classes:

1. Neglect constraints due to *line capacity*.

To clarify: in practice, a minimum headway (e.g. 3 seconds) must be maintained between vehicles, thus limiting the vehicle flow on each section of guideway. Attention is restricted to networks and demand patterns where these constraints are not active – that is, it is assumed that as passenger demand increases, the vehicle fleet becomes limiting before line capacity does. (Of course, it is possible to construct networks and demand patterns where the contrary is true, and their analysis is the subject of ongoing research.) As a consequence of this assumption, it is assumed further that all vehicles may be routed via the quickest possible paths.

2. Neglect processes at stations which result in delay to vehicles.

To clarify, the following delays are neglected: passenger boarding and deboarding times; delays at the start of journeys, when the central controller (in synchronous control) books a vehicle's path through the network; delays due to limited station throughput; delays due to interference between vehicles leaving or entering a station. Furthermore it is assumed that each station has sufficient berths so that arriving vehicles are never delayed or waved off (*1*, ch. 3.2.3) due to insufficient space in their destination station. The consequence of this second class of approximations is that at the capacity limit, all vehicles are continuously employed either in passenger journeys or empty vehicle redistribution.

Because of these simplifications, there is a tendency for this analysis to over-estimate capacity, but as a first step section 7 compares theory with more detailed simulations in which line capacity is limited. Finally, it is assumed that passenger demand at stations is given in terms of pre-formed parties traveling together by choice, and *ride sharing* is neglected – that is, it is assumed for simplicity that parties are unwilling to combine further and share a vehicle even when a substantial queue has formed at a station. In practice, ride sharing may increase the capacity of a PRT system substantially (*9*).

3. THE FLUID LIMIT AND THE CAPACITY REGION

This section reviews well-known (*1*, ch. 5.6; *8*, pp. 67–76) results that hold in the fluid limit; that is, when occupied and empty vehicle flows are expressed as time-averaged continuous variables. These results are then interpreted in a new way, as necessary conditions for a given demand to be in a system's capacity region. It is not yet known whether they are also sufficient conditions, but they are still useful approximations.

Under the assumptions in section 2, the particular network topology and station characteristics are not important. The relevant system parameters are the number of stations, N ($N \geq 2$), the quickest path travel times, T_{ij} ($1 \leq i \neq j \leq N$; $T_{ij} > 0$; circular trips are not allowed), between each pair of stations i and j, and the vehicle fleet size, $C_{\max}$ ($C_{\max} \geq 1$). It is assumed that the demand is specified in an origin-destination (OD) demand matrix with entries D_{ij} ($1 \leq i \neq j \leq N$; $D_{ij} \geq 0$) in parties per unit time. In order for all of the arriving parties to be served (and neglecting ride sharing), the occupied vehicle flow from i to j must be identical to the party flow, D_{ij}. These occupied vehicle flows must then be balanced by empty vehicle flows, X_{ij} ($1 \leq i \neq j \leq N$; $X_{ij} \geq 0$), such that the total (occupied and empty) vehicle flow is conserved at every station:

$$\sum_{\substack{j \\ j \neq i}} (D_{ij} + X_{ij}) = \sum_{\substack{j \\ j \neq i}} (D_{ji} + X_{ji}) \tag{1}$$

for each station i. That is, the total flow into each station must equal the total flow out. The number of concurrently moving vehicles, C, required to serve the demand is then given by the sum-product of total flows and journey times,

$$C = \sum_{\substack{i,j \\ i \neq j}} (D_{ij} + X_{ij}) T_{ij} \ . \tag{2}$$

Equations (1) and (2) can be written more succinctly in vector form by introducing the column vectors $\mathbf{t}$, $\mathbf{d}$ and $\mathbf{x}$ that respectively list the elements T_{ij}, D_{ij} and X_{ij} in order. (For example, $\mathbf{t} = (T_{1,2}, T_{1,3}, \ldots, T_{N,N-1})'$, where $'$ (prime) denotes the transpose.) They become

$$\mathbf{A}(\mathbf{d} + \mathbf{x}) = \mathbf{0} \quad \text{and} \quad C = \mathbf{t}'(\mathbf{d} + \mathbf{x}) \tag{3}$$

where $\mathbf{A}$ is a matrix with N rows and $N^2 - N$ columns that encodes the constraints from (1); all of its entries are either ±1 or 0.

Because the system has only $C_{\max}$ vehicles in total, it can serve demand $\mathbf{d}$ only if $C \leq C_{\max}$. This inequality and (3) give a necessary condition for demand $\mathbf{d}$ to be in the system's capacity region: there must exist empty vehicle flows $\mathbf{x_d}$ for $\mathbf{d}$ such that

$$\mathbf{A}(\mathbf{d} + \mathbf{x_d}) = \mathbf{0} \quad \text{and} \quad \mathbf{t}'(\mathbf{d} + \mathbf{x_d}) \leq C_{\max} \ . \tag{4}$$

If any such empty vehicle flows $\mathbf{x_d}$ exist, it is clear from (4) that any flows $\mathbf{x_d^*}$ (not necessarily unique) that solve the linear program

$$\begin{array}{ll} \min & \mathbf{t}'\mathbf{x} \\ \text{s.t.} & \mathbf{A}(\mathbf{d} + \mathbf{x}) = \mathbf{0} \\ & \mathbf{x} \geq \mathbf{0} \end{array} \tag{5}$$

satisfy the conditions in (4). That is, $\mathbf{x_d^*}$ is chosen to minimize the number of concurrent empty vehicles, subject to flow conservation. The linear program (5) can be solved with standard techniques, such as the simplex method. In summary,

$$\mathbf{t}'(\mathbf{d} + \mathbf{x_d^*}) \leq C_{\max} \tag{6}$$

is a necessary condition for the demand $\mathbf{d}$ to be in the capacity region. Geometrically, the constraints on $\mathbf{d}$ due to (6) describe a convex polytope in $N^2 - N$ dimensions.

It is not known whether (6) is also a sufficient condition; that is, there may be systems and demands for which *no* EVR algorithm can prevent the number of waiting passengers from diverging, even though (6) is satisfied. This is a subject of ongoing research. However, simulation results indicate that for some networks, demands and EVR algorithms, the bound (6) is very nearly attained.

4. BENCHMARKING EVR ALGORITHMS

Once an EVR algorithm is implemented in simulation, the aim is to measure the `maximum demand' that it can serve. However, the demand is described by the $N^2$ - $N$ entries of an OD matrix, so the maximum is not well-defined. The approach here is to choose a *demand pattern*, **r**, that fixes the proportions of the demand. The `benchmark' for this demand pattern is set by the largest s such that demand $s\mathbf{r}$ satisfies (6); this s is denoted $s_{\max}$. Geometrically, **r** defines the direction of a ray in N^2 - N dimensions, and $s_{\max}/\|\mathbf{r}\|$ is the distance from the origin to the boundary of the capacity region in the direction of **r**. In practical terms, **r** is chosen to represent a particular demand scenario, like an AM peak. The scalar s varies the total demand without changing its distribution between pairs of stations. To understand the performance of an EVR algorithm over a variety of demand scenarios, several such demand patterns must be analyzed.

To find $s_{\max}$, one can substitute $s\mathbf{r}$ for **d** in (6) and observe that the linear program in (5) satisfies $\mathbf{x}^*_{s\mathbf{r}} = s\mathbf{x}^*_{\mathbf{r}}$; that is, the required empty vehicle flows scale linearly with s. The inequality (6) may then be written as

$$\mathbf{t}'(s\mathbf{r}) + \mathbf{t}'\mathbf{x}^*_{s\mathbf{r}} = s(\mathbf{t}'\mathbf{r} + \mathbf{t}'\mathbf{x}^*_{\mathbf{r}}) \le C_{\max},$$

and the largest s occurs when this is satisfied at equality, so

$$s_{\max} = \frac{C_{\max}}{\mathbf{t}'(\mathbf{r} + \mathbf{x}^*_{\mathbf{r}})}. \tag{7}$$

It is now useful to define the *intensity* of the demand $s\mathbf{r}$ as $s / s_{\max}$; the intensity is 1.0 when the demand $s\mathbf{r}$ reaches the boundary of the capacity region. An EVR algorithm is evaluated by the actual intensity at which the number of waiting passengers begins to diverge, as determined by simulation. These *saturation intensities* can be used to compare an EVR algorithm across demand patterns and networks, as is demonstrated in section 7.

5. STOCHASTIC SIMULATIONS FOR EVR HEURISTICS

The preceding analysis of the fluid limit does not include an explicit EVR algorithm, so it must be combined with other tools if it is to be useful for comparing EVR algorithms. In this paper, these tools are stochastic simulations.

Several EVR algorithms (*1*; *2*; *3*; *4*; *7*) have been described in the literature. The methods developed in this paper are suitable for studying all of these, but for illustrative purposes, the focus here is on comparing two fairly simple heuristics. Each heuristic requires its own simulation structure. Both simulations have the same inputs, namely the demand **d**, the travel times **t**, and the fleet size $C_{\max}$, and both are implemented in discrete time, with one second time steps. For convenience, it is also assumed that the demands D_{ij} are in parties per second with all $D_{ij} \ll 1$, and that the travel times T_{ij} are rounded to the nearest second. In each time step t, and for each pair of stations i and j with $i \neq j$, a party is generated with probability D_{ij}; over many time steps, this is a discrete-time approximation to the Poisson process with mean rate D_{ij}. The

next action to take with this passenger party depends on which of the EVR heuristics described below is in use.

Bell and Wong Nearest Neighbors (BWNN)

BWNN is the simplest of several heuristics explored by (*7*) in the context of taxi operations. It assumes that each vehicle has a list of passengers that it must serve. The origins and destinations of these passengers are known, so the heuristic can calculate when and where (at which station) each vehicle will finish serving all of the passengers in its list. When a new passenger arrives, the heuristic immediately selects a vehicle to serve him and adds him to the end of that vehicle's list. Precisely, it selects a vehicle k^* that minimizes the new passenger's waiting time; that is,

$$k^* = \underset{k}{\operatorname{argmin}} \max(0, a_k - t) + T_{d_k i},$$

where t is the current time, d_k is vehicle k's final destination (either the last in its list or the station at which it is idle), and a_k is k's arrival time at d_k. If there is a tie, the vehicle with lowest k is selected. Once a passenger is added to a vehicle's list, the BWNN heuristic never moves him to another list, even though this might reduce overall waiting times.

Longest-Waiting Passenger First (LWPF)

In contrast with BWNN, the LWPF heuristic requires that each vehicle store only its next destination; once it reaches its destination, the system decides where it should go next. When a vehicle becomes idle at its destination, it is dispatched to the longest-waiting passenger. Precisely, the following steps are carried out at each time step t:

1. Each generated passenger (if there are any) joins the queue at his origin station.
2. For each station i (in order of index, as order does not matter here):
    2.1. All vehicles finishing their trips to i at time t become idle at i.
    2.2. If there are both waiting passengers and idle vehicles at i, the first passenger is removed from the queue and a vehicle becomes inbound to his destination, j, with arrival time $t + T_{ij}$. This step repeats until there are either no waiting passengers or no idle vehicles at i.
3. For each station i with waiting passengers, let h_i be the arrival time of the longest-waiting passenger.
4. For each station i with more waiting passengers than parked plus inbound vehicles, in ascending order by h_i, consider the stations $j \neq i$ in ascending order by T_{ji} (breaking any ties randomly); choose the first station (if any) that has more parked empty vehicles than waiting passengers, and send an empty vehicle from this station to station i.

Compared to BWNN, LWPF tends to wait longer before assigning a vehicle to a passenger. In the intervening time, more passengers may arrive, in which case LWPF performs its optimization with more information and so can, in principle, make better decisions. This principle is further exploited in (*4*), which demonstrates improved service levels when empty vehicles can be reassigned even later, while en route. Another notable difference is that BWNN allows a vehicle to leave waiting passengers at a station and proceed empty to serve the next passenger on its list, whereas LWPF does not. This is not an issue for taxis, because taxi drop-off locations are not concentrated at stations, but it could be an issue when using BWNN for PRT, because passengers may not like being left behind.

6. TEST STUDIES

Two representative scenarios are used in this study; each scenario consists of a network, OD matrix and fleet size. The first network (Figure 2(a)) is taken from the Corby case study (*10*).

The network layout and demand used in this study are both publicly available as part of the *ATS/CityMobil* PRT simulator (*11*). The demand matrix represents the AM peak for phase 1 of the proposed system. There are 15 stations. The fleet size is set to 200 vehicles, as is estimated in the case study.

The second network (Figure 2(b)) is a regular grid of one-directional guideways with 24 stations located at the line midpoints; this idealized topology appears several times in the PRT literature (*1*, ch. 2, for example). Lines are spaced at 800m (0.5mi) to provide 400m (0.25mi) maximum walk distances. Assuming 10m/s (22mph) average speed, adjacent stations are 80s apart, and the maximum station-to-station travel time is 12 minutes (e.g. from *B* to *G*). The OD matrix for the grid network is obtained from a standard gravity model with

$$D_{ij} = \begin{cases} A_i B_j O_i D_j \exp(-\theta T_{ij}) & i \neq j \\ 0 & i = j \end{cases} \tag{8}$$

where $O_i = \sum_j D_{ij}$ and $D_j = \sum_i D_{ij}$ are the desired total origin and destination flows, θ is the dispersion parameter and T_{ij} is the travel time, in seconds, from i to j through the network (not Euclidean distance). The O_i and D_j are chosen to represent an AM peak, with the demand distribution given in Figure 3. The A_i and B_j coefficients are computed by fixed-point iteration. The θ parameter is initially set to 0.01; other values of θ, which generate different demand patterns, are considered later. The fleet size is set at 200 vehicles.

7. RESULTS

For each scenario, $s_{\max}$ is computed with (7) by setting **r** according to the scenario's demand matrix. Each simulation run uses a stationary (constant) demand s**r** for a different value of s (that is, a different intensity for demand pattern **r**). The intensity at which the number of waiting passengers begins to diverge is most easily measured by observing the mean number of concurrent moving vehicles; as suggested in (6), this quantity *saturates* at the fleet size, $C_{\max}$, when the demand approaches the boundary of the capacity region.

Simulation results are presented in Figure 4. Intensity is increased in increments of 0.01, and each point is based on data from 10 independent trials. Each trial consists of a 10 hour warm up period, in which no statistics are collected, followed by 80 hours of statistics collection. Running the simulation for a long time makes the saturation intensity easier to identify, because the observed queue length increases in proportion to the running time when the queue is diverging.

Figure 4(a) shows that both EVR algorithms saturate very close to the predicted intensity on the Corby network, because the number of concurrent vehicles reaches the fleet size near intensity 1.0. Figure 4(b) shows the same measure for the Grid network; in this case, the EVR algorithms both saturate at intensities less than 1.0 (LWPF at 0.85 and BWNN at 0.95). This is also apparent in Figures 4(d) and 4(f), where the mean number of waiting passengers and their waiting times diverge at roughly the same intensities. It appears that neither LWPF nor BWNN attains the theoretical maximum throughput for all networks and demands; it is not yet known whether there is any practical algorithm that does.

One notable feature of Figure 4(b) is that for LWPF the number of concurrent empty vehicles increases suddenly at intensity 0.80. This increase in concurrent empty vehicles prevents an increase in the number of concurrent occupied vehicles, since at intensities above 0.80, the system is serving the same number of passengers with more empty vehicle movement. The reason is that, when a vehicle becomes idle, it must serve the longest-waiting passenger,

regardless of his location in the network. When there are standing queues at many of the stations, the average empty vehicle trip may be significantly longer for LWPF than for BWNN.

Figures 4(c–f) show that, while BWNN may saturate at higher intensity, LWPF may give lower queue lengths and waiting times at lower intensities. So, it is not necessarily true that the EVR algorithm with the highest saturation intensity also provides the best level of service at lower intensity. There may be a trade-off between throughput at high intensities and waiting times at low intensities, or there may be an algorithm that can perform well in both regards; this is not yet known.

Figures 4(e) and 4(f) show long waiting times even when intensity is near zero, and they increase only slowly with intensity. This is because the EVR algorithms used here do not move vehicles in anticipation of future passengers. For example, even if there is tidal flow from an origin *i* to a destination *j*, vehicles stay at *j* until a passenger arrives at *i* and requests a vehicle, so all passengers wait at least T_{ji}, regardless of the intensity. In this case, it is clear that the system should move vehicles back to *i*. However, it is not so simple when flow is not tidal, there are multiple origins and destinations, or the demands are uncertain.

The Effects of Line Congestion

The analysis and simulation done so far has assumed that line capacity is infinite. There are certainly networks and demands for which this is a poor assumption. So, it is prudent to check these results against a more detailed simulator that includes line congestion. Here, results are from a proprietary simulator developed by Advanced Transport Systems Ltd.. It is configured to use simple synchronous control (*1*, pp. 92–94), and vehicles are restricted to the path with the smallest free-flow time; this gives a pessimistic estimate of the line throughput that is realistically achievable. The proprietary EVR algorithm has been configured to closely (but not exactly) match the LWPF algorithm described here.

The curves in Figure 5(a) and 5(b) are very similar to those in Figures 4(c) and 4(d) when the minimum headway is 1s; in particular, the saturation intensities are roughly the same. When the minimum headway is increased, the line capacity is decreased, so delays due to line congestion become more likely; these delays contribute to the overall trip times (effectively increasing the T_{ij}), which causes the number of waiting passengers to diverge at lower intensity. In this case, reducing line capacity by a factor of 2 (or more, in the Grid network) produces only small changes in Figure 5. Of course, this might not be the case with larger fleet sizes or smaller travel times.

The Effects of Different Demand Patterns

The proposed method works with only one demand pattern, **r**, at a time. To evaluate the performance of an EVR algorithm for a given system, several demand patterns must be investigated. As a first step, Figure 6 shows the effect of varying the dispersion parameter, θ, in the gravity model (8). For $\theta = 0.01$, Figure 6(a) shows that mean queue lengths are shorter for LWPF than for BWNN at intensities below 0.82. However, for $\theta = 0.005$, the two heuristics give similar mean queue lengths until LWPF diverges (Figure 6(c)), and, for $\theta = 0.001$, mean queue lengths for LWPF are larger than for BWNN at all intensities (Figure 6(e)). The corresponding OD matrices in Figures 6(b, d and f) suggest a reason: as θ decreases, the demand is spread over more origins (indicated by an increase in the number of dark cells), and these are further from the main destinations. This means that the longest-waiting passenger is typically further away for smaller θ, which leads to longer empty vehicle trips. This example illustrates the importance of

considering different demand patterns. It also shows how the intensity measure defined here is useful for comparing results for these different demand patterns.

8. CONCLUSIONS

This paper demonstrates a new method for the evaluation of empty vehicle redistribution (EVR) algorithms, providing an absolute measure of their performance according to a metric based on the capacity region for a given network. The capacity region is defined as the set of OD matrices which are feasible in the sense that their demands can be met without passenger queues growing indefinitely. It describes the maximum possible demand that a particular system could serve with an ideal EVR algorithm, and hence acts as an absolute benchmark against which different EVR algorithms can be compared.

The ability to compare and evaluate EVR algorithms is important for the successful operation of highly-connected PRT systems, like the grid network in Figure 2(b). In normal PRT operation, the minimization of passenger waiting time is usually the priority, and hence one could expect an EVR heuristic which prioritizes this (e.g. LWPF) to be in operation. At times of high demand, however, when the vehicle fleet is stretched and there are passengers waiting at numerous stations across the network, the LWPF algorithm often moves vehicles too far. One would instead prefer an algorithm which prioritizes the efficient use of the vehicle fleet (e.g. the BWNN heuristic). Thus the central controller should at some point switch from one algorithm to another (or indeed choose from a selection of many others), and the methods described here provide a rigorous basis for this decision. Furthermore, the absolute benchmark indicates when an EVR algorithm is near-optimal in the sense of capacity (such as the BWNN heuristic on both networks studied here), so that no other algorithm need be considered.

This analysis also shows how both the network topology and the spatial distribution of the demand can affect EVR performance, even when line congestion is ignored. For the Corby network, the LWPF heuristic consistently outperforms the BWNN alternative (Figure 4(c, e)). For the Grid network, BWNN consistently performs better in terms of throughput, but, in terms of queue lengths and passenger waiting times, the relative performance of these heuristics depends on the spatial dispersion of the demand.

The proposed method allows for the absolute assessment of EVR algorithms in terms of throughput, subject to the modeling assumptions in section 2. For other performance measures, such as those based on passenger waiting times, only relative performance can be assessed. In general, conclusions about the relative merits of EVR algorithms must be based on the analysis of many networks, demands and fleet sizes, and, at present, this requires detailed simulation. The results presented here show that the capacity region formalism is essential for comparing and interpreting these simulation results.

As mentioned earlier, there are a number of alternative heuristics already present in the literature (*1*; *2*; *3*; *4*; *7*), and an analysis of these algorithms using this evaluation tool is a natural next step. It would also be desirable to include other limiting effects (such as line congestion, see section 7) in the capacity region calculations. This would enable the EVR evaluation to be confidently applied to all networks, not just those in which the vehicle fleet is the limiting factor. This extension to the analysis presented here is currently ongoing.

ACKNOWLEDGEMENTS
The authors thank Prof. Frank P. Kelly (Cambridge), Prof. Martin V. Lowson (Advanced Transport Systems Ltd.) and Dr. Philip Bly for ideas, discussion and review. JDLM acknowledges the support of an Overseas Research Scholarship from the University of Bristol. REW acknowledges the support of an EPSRC Advanced Fellowship EP/E055567/1. This work was partly funded by the CityMobil Sixth Framework Programme for DG Research Thematic Priority 1.6, Sustainable Development, Global Change and Ecosystems, Integrated Project, Contract Number TIP5-CT-2006-031315.

REFERENCES

1. Irving, J. H., H. Bernstein, C. L. Olson, and J. Buyan. *Fundamentals of Personal Rapid Transit*. Lexington Books, D.C. Heath and Company, 1978.

2. Andréasson, I. Vehicle Distribution in Large Personal Rapid Transit Systems. In *Transportation Research Record: Journal of the Transportation Research Board*, No. 1451, Transportation Research Board of the National Academies, Washington, D.C., 1994, pp. 95–99.

3. Anderson, J. E. Control of Personal Rapid Transit Systems. *Journal of Advanced Transportation*, Vol. 32, No. 1, 1998.

4. Andréasson, I. Reallocation of Empty Personal Rapid Transit Vehicles en Route. In *Transportation Research Record: Journal of the Transportation Research Board*, No. 1838, Transportation Research Board of the National Academies, Washington, D.C., 2003, pp. 36–41.

5. Powell, W. B. A stochastic formulation of the dynamic assignment problem, with an application to truckload motor carriers. *Transportation Science*, Vol. 30, No. 3, 1996, pp. 195–219.

6. Berbeglia, G., J.-F. Cordeau, and G. Laporte. Dynamic pickup and delivery problems. *European Journal of Operational Research*, Vol. 202, No. 1, 2010, pp. 8–15.

7. Bell, M. G. H., and K. I. Wong. A Rolling Horizon Approach to the Optimal Dispatching of Taxis. In *Transportation and Traffic Theory: Flow, Dynamics and Human Interaction*, H. S. Mahmassani Ed., 2005, pp. 629–648.

8. Anderson, J. E. *Transit Systems Theory*. D. C. Heath and Co., Lexington Mass., 1978.

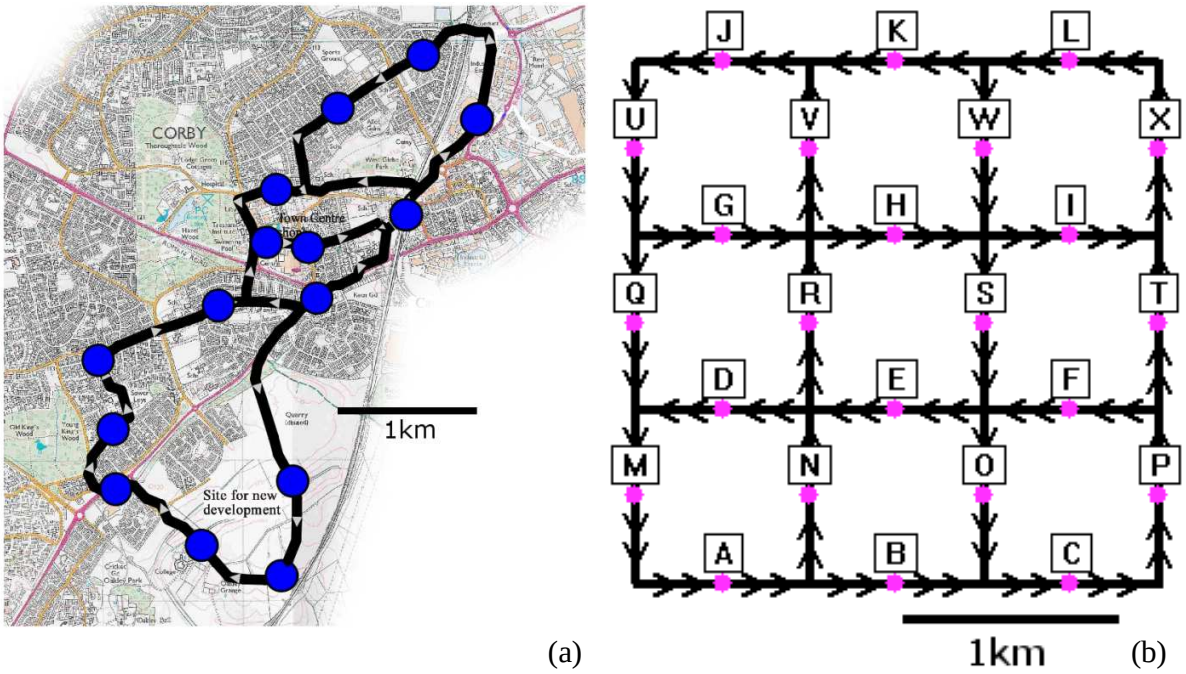
9. Lees-Miller, J., J. Hammersley, and N. Davenport. Ride sharing in personal rapid transit capacity planning. In *Automated People Movers 2009*, American Society of Civil Engineers, 2009.

10. Bly, P. H., and R. Teychenne. Three financial and socio-economic assessments of a personal rapid transit system. In *Automated People Movers 2005*, American Society of Civil Engineers, 2005.

11. CityMobil. *PRT simulation tool*. http://www.citymobil-project.eu/site/en/documenten.php. Accessed March 1, 2010.

LIST OF FIGURES

FIGURE 1 PRT vehicle and at-grade station at London Heathrow Airport. PRT vehicles, stations and infrastructure are smaller than typical Automated People Mover and urban rail systems. Vehicle length, width and height are 3.7m, 1.4m and 1.8m (12ft, 4.6ft and 5.9ft), respectively. Photo courtesy of Advanced Transport Systems Ltd.



(a) (b)

FIGURE 2 Network layouts used for stochastic simulation of the Corby (a) and Grid (b) networks. Guideways (black lines) are one-way in the direction indicated; circles represent stations in (a), and letters represent stations in (b).

	5.0	5.0	5.0	
5.0	3.8	3.8	5.0	
	3.8	2.5	3.8	
5.0	2.5	2.5	5.0	
	3.8	2.5	3.8	
5.0	3.8	3.8	5.0	
	5.0	5.0	5.0	

(a)

	0.8	0.8	0.8	
0.8	3.8	3.8	0.8	
	3.8	15.0	3.8	
0.8	15.0	15.0	0.8	
	3.8	15.0	3.8	
0.8	3.8	3.8	0.8	
	0.8	0.8	0.8	

(b)

FIGURE 3 Total flows for the grid network gravity model. Table layouts correspond to the station layout in Figure 2(b). For example, the top left station (labeled J) is the origin of 5.0% of passenger parties and the destination for 0.8%.

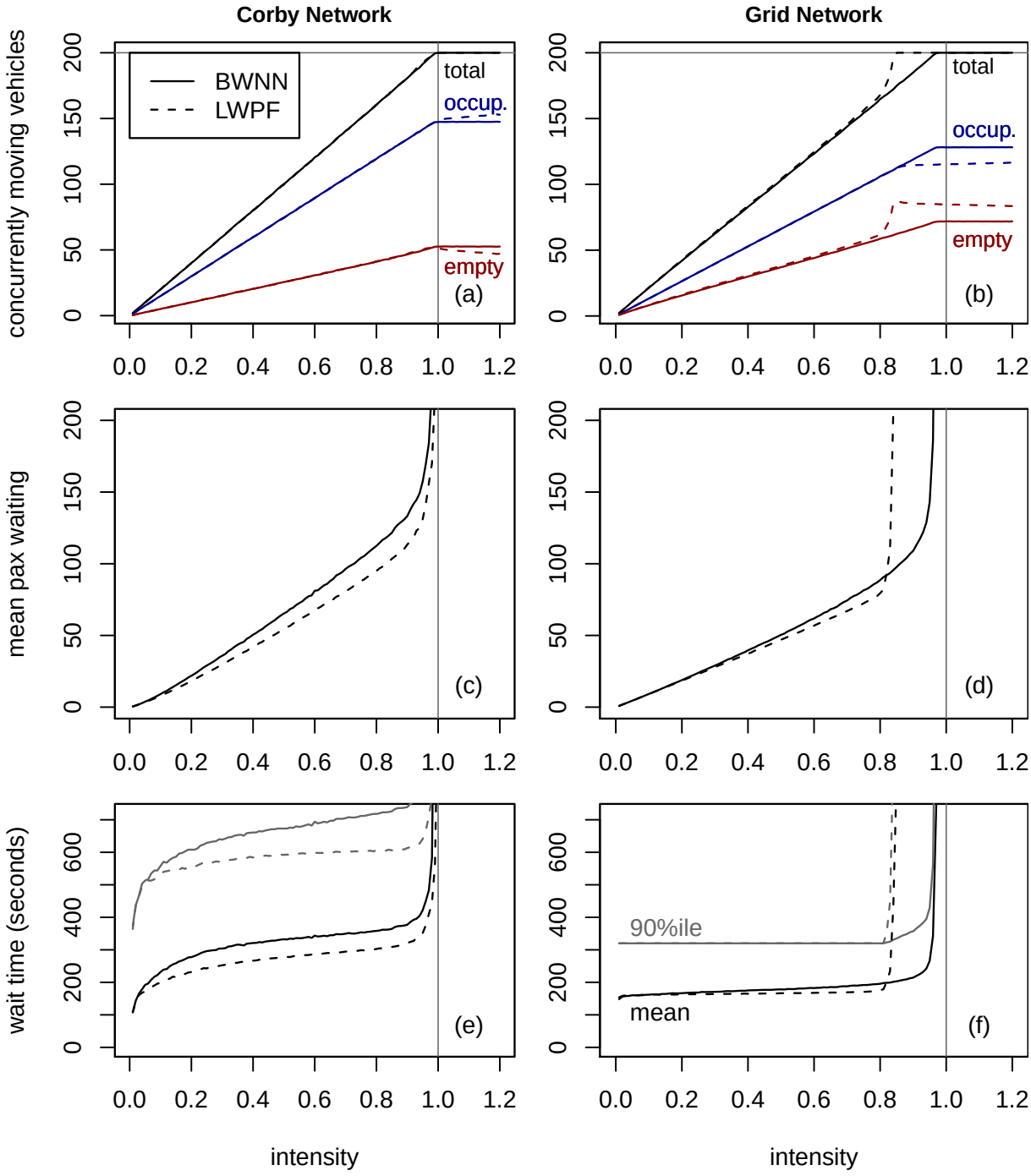


FIGURE 4 Simulation results for the BWNN and LWPF EVR algorithms. Their saturation intensities are similar for the Corby network but different for the Grid network; LWPF shows higher empty vehicle use when there are passengers waiting at many stations. Until divergence, LWPF gives lower waiting times and queue lengths. Intensity 1.0 corresponds to 1414 parties/hr on Corby and 2035 parties/hr on Grid; normalizing using theoretical capacity (section 4) helps comparison between different networks.

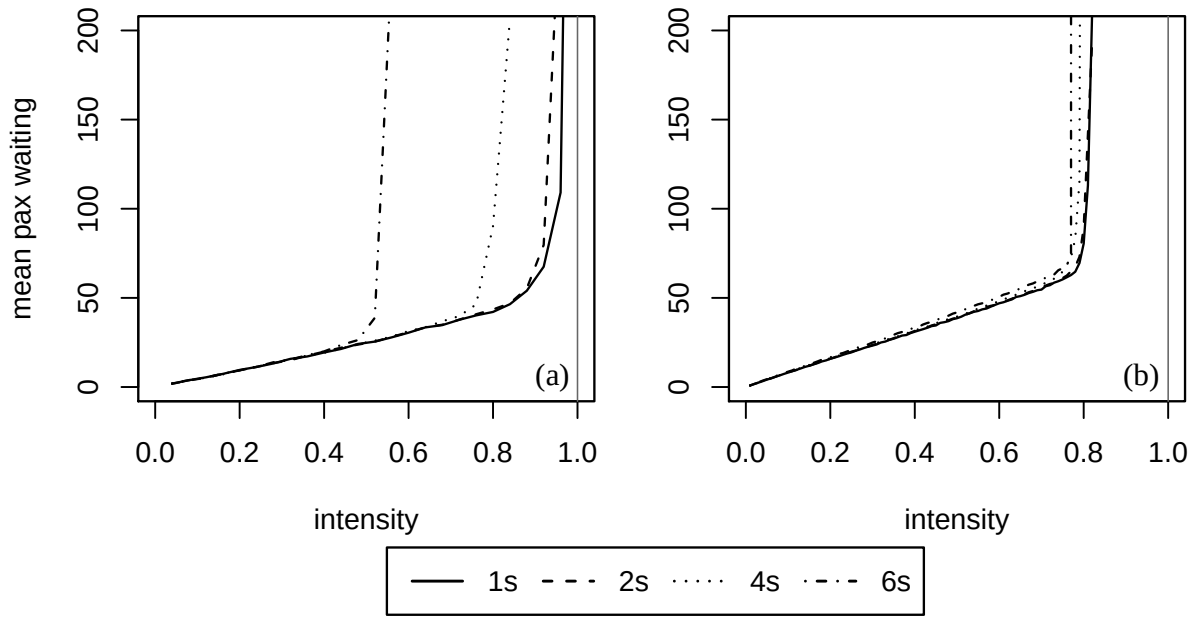


FIGURE 5 Effect of line congestion on mean queue length. Queues diverge at the same intensities as in Figure 4, validating model assumptions. The Grid network is operating far below maximum line capacity with 200 vehicles; more could be added. The Corby system is closer to line capacity: if the minimum headway exceeds 2s, delays due to line congestion reduce throughput.

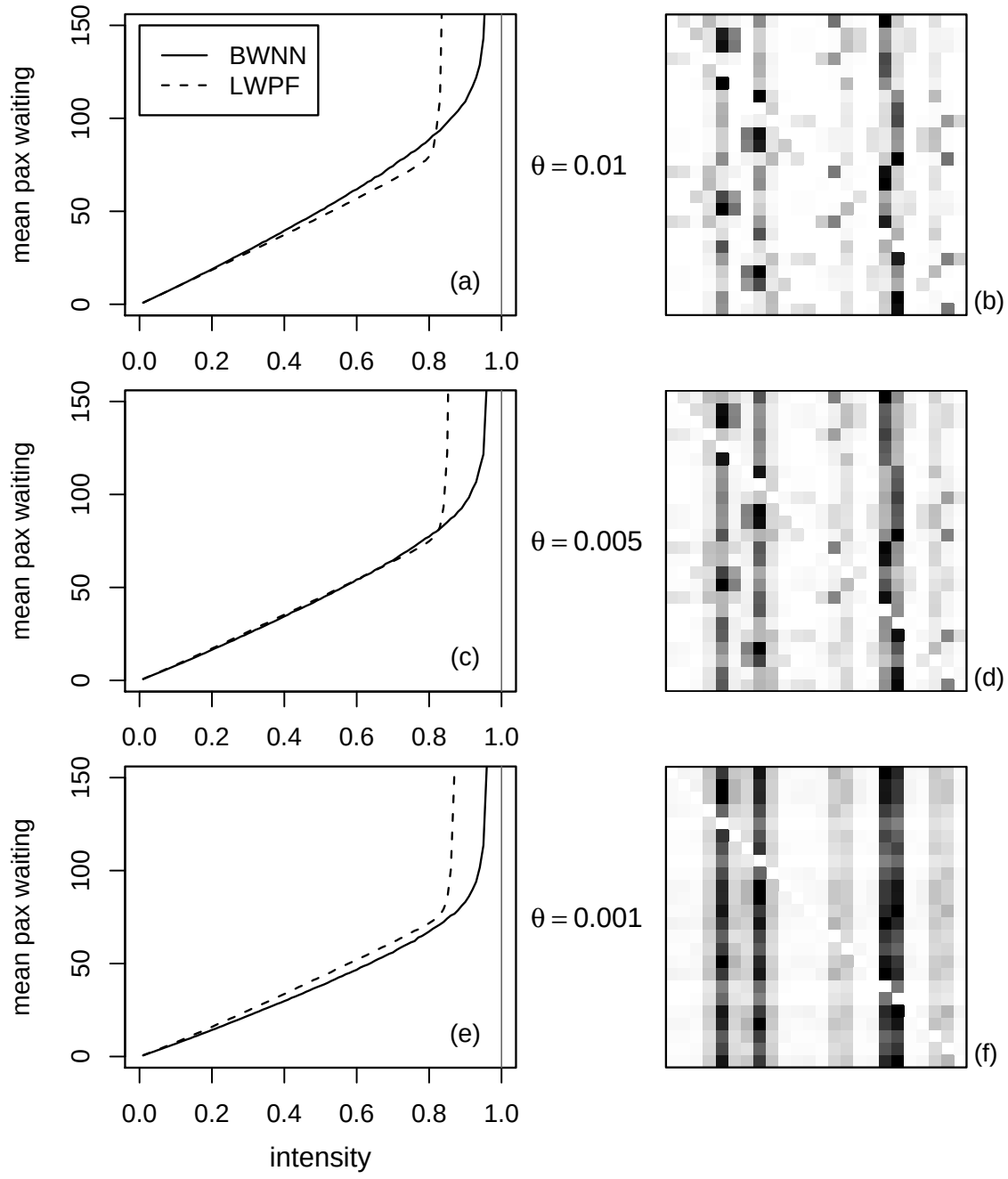


FIGURE 6 Effect of dispersion, θ, on mean queue length for the Grid Network. For larger θ, mean queue lengths are shorter for LWPF than for BWNN at low intensity, but this is not true for smaller θ. The OD matrices from the gravity model (8) are shown in (b), (d) and (f); each cell represents one OD pair (one row per origin), and darker cells represent a larger share of the total demand. Demand is more uniformly distributed for smaller θ.