Probabilistic Analyses and Practical Algorithms for the Vehicle Routing Problem with Time Windows*

Julien Bramel† and David Simchi-Levi‡

September 1992 Revised: April 1993, July 1993

Abstract

In the Vehicle Routing Problem with Time Windows, a set of customers are served by a fleet of vehicles of limited capacity, initially located at a central depot. Each customer provides a period of time in which they require service, which may consist of repair work or loading/unloading the vehicle. The objective is to find tours for the vehicles, such that each customer is served in its time window, the total load on any vehicle is no more than the vehicle capacity, and the total distance traveled is as small as possible. In this paper, we present a characterization of the asymptotic optimal solution value for general distributions of service times, time windows, customer loads and locations. This characterization leads to the development of a new algorithm based on formulating the problem as a stylized location problem. Computational results show that the algorithm is very effective on a set of standard test problems.

*Research supported in part by ONR contract N00014-90-J-1649 and NSF contract DDM-8922712.

†Graduate School of Business, Columbia University, NY, NY, 10027.

‡Dept. of Industrial Engineering and Operations Research, Columbia University, NY, NY, 10027 and Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL 60208.

1 Introduction

The Vehicle Routing Problem with Time Windows (VRPTW) can be stated as follows: a set of customers dispersed in a geographic region has to be served by a fleet of vehicles initially located at a given depot. Each customer has a load that must be picked up and the customer specifies a period of time, called a *time window*, in which this pick up must occur. The customers are served by vehicles of limited capacity, that is, total load carried by each vehicle can be no more than the vehicle capacity. The objective is to find a set of routes for the vehicles, where each route begins and ends at the depot, serves a subset of the customers without violating the capacity and time window constraints, while minimizing the total length of the routes.

Due to the wide applicability and the economic importance of the problem in the service industry, variants of it have been extensively studied in the vehicle routing literature; for a review see Solomon and Desrosiers (1988). Most of the work has focused on *empirical analysis*, where heuristics are implemented on standard test problems and their performance is compared to other heuristics, see for example Solomon (1986). Desrochers, Desrosiers and Solomon (1992) use a column generation approach to solve the problem when only a fraction of the customers have time windows.

By contrast, very few papers have studied the problem from an analytical point of view in an attempt to characterize the theoretical behavior of heuristics. Included here is Daganzo (1987a,b) who obtained an approximation for the total distance traveled by all vehicles under assumptions on customer locations and service times. Bramel, Li and Simchi-Levi (1991) performed an average-case analysis of a restricted version of the model. Federgruen and van Ryzin (1992) recently analyzed a general model which includes the VRPTW as a special case. Their analysis however does not lend itself to a computationally attractive algorithm.

In this paper we perform a probabilistic analysis of the VRPTW in an attempt to characterize the structure of the asymptotic optimal solution. Interestingly enough, this optimal solution is directly related to an optimal solution of a stylized machine scheduling problem. That is we show that asymptotically the VRPTW can be solved by solving a related scheduling problem. This insight is used to develop a new algorithm which is computationally attractive and theoretically interesting.

In the next section we provide a detailed description of the model analyzed together with our main result.

2 The Model

To formally describe the model we analyze here, let the index set of the n customers be denoted $N = \{1, 2, \ldots, n\}$. Let $y_k \in \Re^2$ be the location of customer $k \in N$. Assume, without loss of generality, that the depot is at the origin and, by rescaling, that the vehicle capacity is 1 and that the length of the working day is 1. We assume vehicles can leave and return to the depot at any time. Associated with customer k is a quadruplet (w_k, e_k, s_k, l_k), called the customer *parameters*, which represents, respectively, the load that must be picked up, the earliest starting time for service, the time required to complete the service, called the *service time*, and the latest time service can end. Clearly, feasibility requires that $e_k + s_k \le l_k$ and $w_k, e_k, l_k \in [0, 1]$, for each $k \in N$.

For any point $y \in \Re^2$, let $||y||$ denote the Euclidean distance between y and the depot. Let $d_k \equiv ||y_k||$ be the distance between customer k and the depot. Also, let $d_{jk} \equiv ||y_j - y_k||$ be the distance between customer j and customer k. Let Z_n^* be the total distance traveled in an optimal solution to the VRPTW, and let Z_n^{H} be the total distance traveled in the solution provided by a heuristic H.

Consider the customer locations to be distributed according to a distribution μ with compact support in $\Re^2$. Let the customer parameters $\{(w_k, e_k, s_k, l_k) : k \in N\}$ be drawn from a joint distribution Φ with a continuous density ϕ. Let C be the support of ϕ, i.e., C is a subset of $\{(x_1, x_2, x_3, x_4) \in [0,1]^4 : x_2 + x_3 \leq x_4\}$. Each customer is therefore represented by its location in the Euclidean plane along with a point in C. Finally, we assume that a customer's location and its parameters are *independent* of each other.

In our analysis we associate a *job* with each customer. The parameters of job k are the parameters of customer k, that is, (w_k, e_k, s_k, l_k), where w_k is referred to as the *load* of job k and, using standard scheduling terminology, e_k represents the earliest time job k can begin processing, s_k represents the processing time, and l_k denotes the latest time the processing of the job can end. The value of e_k can be thought of as the *release time* of job k, that is, the time it is available for processing. The value of l_k represents the *due date* for the job. Each job can abstractly be viewed as simply a point in C. Occasionally, we will refer to customers and jobs interchangeably; this convenience should cause no confusion.

For a given set of customers $T \subseteq N$ with parameters $\{(w_k, e_k, s_k, l_k) : k \in T\}$, we associate a corresponding *machine scheduling problem* as follows: Consider the set of jobs T and an infinite sequence of *parallel* machines. Job k becomes available for processing at time e_k and must be finished processing by time l_k. The objective in this scheduling problem is to assign each job to a machine such that (i) each machine has at most one job being processed on it at a given time, (ii) the processing time of each job starts no earlier than its release time and ends no later than its due date, (iii) the total load of all jobs assigned to a machine is no more than 1, and the number of machines used is minimized. In our discussion we refer to (ii) as the *job time window constraint* and to (iii) as the *machine load constraint.*

Scheduling problems have been widely studied in the operations research literature; see Lawler, Lenstra, Rinnooy Kan and Shmoys (1989). Unfortunately, no paper has considered the scheduling problem in its general form with the objective function of minimizing the number of machines.

Let $M^*(S)$ be the minimum number of machines needed to schedule a set S of jobs. It is clear that this machine scheduling problem possesses the property known as *subadditivity*, which states:

$$\forall S, T, \quad M^*(S \cup T) \leq M^*(S) + M^*(T).$$

The theory of subadditive processes (see Kingman (1976)) implies that if M_n^* is the minimum number of machines needed to schedule a set of n jobs whose parameters are drawn independently from a distribution Φ, then there exists a constant $\gamma > 0$ (depending only on Φ) such that $\lim_{n\to\infty} M_n^*/n = \gamma$ (a.s.).

In this paper we relate the solution to the VRPTW to the solution to the scheduling problem defined by the customers parameters. That is, we show that asymptotically the VRPTW is no more difficult to solve than the corresponding scheduling problem. Our main result is the following:

Theorem 2.1 *Let $y_1, y_2, \ldots, y_n$ be independently and identically distributed according to a distribution μ with compact support in $\Re^2$, and let*

$$E(d) = \int_{\Re^2} \|y\| d\mu(y).$$

Let the customer parameters $\{(w_k, e_k, s_k, l_k) : k \in N\}$ be drawn independently from Φ. Let M_n^* be the minimum number of machines needed to feasibly schedule the n jobs corresponding to these parameters, and let $\lim_{n\to\infty} M_n^*/n = \gamma$ (a.s.). Then

$$\lim_{n\to\infty} \frac{1}{n} Z_n^* = 2\gamma E(d) \quad \text{(a.s.)}.$$

We prove this Theorem in Section 3 by introducing a lower bound on the optimal solution value and then developing an upper bound that converges to the same value. The lower bound uses a similar technique to the one developed in Simchi-Levi and Bramel (1990), see also Bramel at el. (1991) and Bienstock, Bramel and Simchi-Levi (1993). The upper bound can be viewed as a randomized algorithm that is guaranteed to generate a feasible solution to the problem. That is, different runs of the algorithm on the same data may generate different feasible solutions. In Section 4, we show that the analysis leads, in a natural way, to the development of a new deterministic algorithm which is asymptotically optimal for the VRPTW. Computational evidence shows that our algorithm works very well on a set of standard test problems.

3 Proof of Theorem 2.1

We start the analysis by introducing a lower bound on the optimal objective function value Z_n^*. First, let A be the compact support of μ, and define $d_{\max} = \sup\{\|y\| : y \in A\}$. Pick a fixed integer $r > 1$, and define $\underline{d}_j = (j-1)d_{\max}/r$, for $j = 1, 2, \ldots, r$. Now define the sets:

$$F_j = \left\{ k \in N \,\middle|\, d_k > \underline{d}_j \right\} \qquad \text{for } j = 1, 2, \ldots, r.$$

Let $M^*(T)$ be the minimum number of machines needed to schedule the jobs $\{(w_k, e_k, s_k, l_k) : k \in T\}$, for any $T \subseteq N$. The next Lemma provides a deterministic lower bound on Z_n^*.

Lemma 3.1

$$Z_n^* > 2\frac{d_{\max}}{r} \sum_{j=2}^{r} M^*(F_j).$$

Proof. Let V_j^* be the number of vehicles in an optimal solution to the VRPTW that serve a customer from F_j, for $j = 1, 2, \ldots, r$. By this definition, V_r^* is exactly the number of vehicles whose furthest customer visited is in F_r, and $V_j^* - V_{j+1}^*$ is exactly the number of vehicles whose furthest customer visited is in $F_j \setminus F_{j+1}$. Observe that if $V_j^* = V_{j+1}^*$, then there are no vehicles whose furthest customer visited is in $F_j \setminus F_{j+1}$. Consequently,

$$\begin{aligned} Z_n^* &> 2\underline{d}_r V_r^* + \sum_{j=1}^{r-1} 2\underline{d}_j (V_j^* - V_{j+1}^*) \\ &= 2\underline{d}_1 V_1^* + \sum_{j=2}^{r} 2(\underline{d}_j - \underline{d}_{j-1}) V_j^* \\ &= 2\frac{d_{\max}}{r} \sum_{j=2}^{r} V_j^*. \end{aligned}$$

We now claim that for each $j = 1, 2, \ldots, r$, $V_j^* \geq M^*(F_j)$. This should be clear from the fact that the set of jobs in F_j can be feasibly scheduled on V_j^* machines by scheduling the jobs at the times they are served in the VRPTW solution. ∎

We can now determine the asymptotic value of this lower bound. This can be done in a similar manner as in Simchi-Levi and Bramel (1990), Bramel et al. (1991) and Bienstock, Bramel and Simchi-Levi (1993), and hence we omit the proof here.

Lemma 3.2 *Under the conditions of Theorem 2.1*

$$\underline{\lim}_{n \to \infty} \frac{1}{n} Z_n^* \geq 2\gamma E(d) \quad \text{(a.s.)}.$$

We prove Theorem 2.1 by approximating the optimal cost from above by that of the following four-step heuristic. In the first step, we partition the region where the customers are distributed into subregions. In the second step, we randomly separate the customers of each subregion into two sets. Then for each subregion, we solve a machine scheduling problem defined on the customers in one of these sets. Finally, we use this schedule to specify how to serve all the customers in the subregion.

Pick an $\epsilon > 0$, and let δ be given by the definition of continuity of ϕ, that is, $\delta > 0$ is such that for all $x, y \in C$ with $||x - y|| < \delta$, we have $|\phi(x) - \phi(y)| < \epsilon$. Finally, pick a $\Delta < \min\{\delta/\sqrt{2}, \epsilon\}$.

Let $G(\Delta)$ be an infinite grid of squares of *diagonal* Δ, that is, of side $\Delta/\sqrt{2}$, with edges parallel to the system coordinates. Recall that A is the compact support of μ and let $A_1, A_2, \ldots, A_{t(\Delta)}$ be the subregions of $G(\Delta)$ that intersect A and have $\int_{A_i} d\mu > 0$.

Let $N(i)$ be the indices of the customers located in subregion A_i, and define $n(i) = |N(i)|$. For each customer $k \in N(i)$, with parameters (w_k, e_k, s_k, l_k), we associate a job with parameters $(w_k, e_k, s_k + \Delta, l_k + \Delta)$. For any set $T \subseteq N$ of customers, let $M^*_\Delta(T)$ be the minimum number of machines needed to feasibly schedule the set of jobs $\{(w_k, e_k, s_k + \Delta, l_k + \Delta) : k \in T\}$. In addition, for any set T of customers, let $T(i) = N(i) \cap T$, for $i = 1, 2, \ldots, t(\Delta)$.

For the given grid partition and for any set $T \subseteq N$ of customers, the following is a feasible way to serve the customers in N. All subregions are served separately, that is no customers from different subregions are served by the same vehicle. In subregion A_i, we solve the machine scheduling problem defined by the jobs $\{(w_k, e_k, s_k + \Delta, l_k + \Delta) : k \in T(i)\}$. Then, for each machine in this scheduling solution we associate a vehicle that serves the customers corresponding to the jobs on that machine. The customers are visited in the exact order they are processed on the machine, and they are served in exactly the same interval of time as they are processed. This is repeated for each machine of the scheduling solution. The customers of the set $N(i) \setminus T(i)$ are served one vehicle per customer. This strategy is repeated for every subregion, thus providing a solution to the VRPTW.

We will show that for a suitable choice of the set T, this routing strategy is asymptotically optimal for the VRPTW. The interesting fact about the set T is that it is a <u>randomly generated</u> set, that is, running the algorithm twice can result in <u>two different</u> sets

T, and hence in two different routing solutions.

The first step is to show that, for any set $T \subseteq N$ (even possibly empty), the solution produced by the above mentioned strategy provides a feasible solution to the VRPTW. This should be clear from the fact that having an extra Δ units of time to travel between customers in a subregion is enough since all subregions have diagonal Δ. Therefore any sets of customers scheduled on a machine together can be served together by one vehicle. Customers of $N(i) \setminus T$ can clearly be served within their time windows since they are served alone.

We now proceed to find an upper bound on the value of this solution. For each subregion A_i, let $n_j(i)$ be the number of jobs on the j^{th} machine in the optimal schedule of the jobs in $T(i)$, for each $j = 1, 2, \ldots, M_\Delta^*(T(i))$. Let $\underline{d}(i)$ be the distance from the depot to the closest customer in $N(i)$, i.e., in subregion A_i. Using the routing strategy described above, the distance traveled by the vehicle serving the customers whose job was assigned to the j^{th} machine of subregion A_i is no more than

$$2\underline{d}(i) + \Delta(n_j(i) + 1).$$

Therefore,

$$\begin{aligned}
Z_n^* &\le \sum_{i=1}^{t(\Delta)} \sum_{j=1}^{M_\Delta^*(T(i))} \left[2\underline{d}(i) + \Delta(n_j(i) + 1) \right] + \sum_{k \notin T} 2d_k \\
&\le 2 \sum_{i=1}^{t(\Delta)} M_\Delta^*(T(i)) \underline{d}(i) + 2n\Delta + \sum_{k \notin T} 2d_k.
\end{aligned}$$

Dividing by n and taking the limit we have

$$\begin{aligned}
\overline{\lim_{n\to\infty}} \frac{1}{n} Z_n^* &\le 2 \sum_{i=1}^{t(\Delta)} \overline{\lim_{n\to\infty}} \frac{1}{n} M_\Delta^*(T(i)) \underline{d}(i) + 2\Delta + \overline{\lim_{n\to\infty}} \frac{1}{n} \sum_{k \notin T} 2d_k \\
&= 2 \sum_{i=1}^{t(\Delta)} \overline{\lim_{n\to\infty}} \frac{n(i)}{n} \frac{M_\Delta^*(T(i))}{n(i)} \underline{d}(i) + 2\Delta + \overline{\lim_{n\to\infty}} \frac{1}{n} \sum_{k \notin T} 2d_k
\end{aligned}$$

$$\leq 2\sum_{i=1}^{t(\Delta)} \overline{\lim_{n\to\infty}} \frac{n(i)}{n} \overline{\lim_{n\to\infty}} \frac{M^*_\Delta(T(i))}{n(i)} \underline{d}(i) + 2\Delta + \overline{\lim_{n\to\infty}} \frac{1}{n} \sum_{k\notin T} 2d_k \tag{1}$$

In order to relate this quantity to the lower bound of Lemma 3.2, we must choose the set T appropriately. For this purpose, we make the following observation. Recall that ϕ is the continuous density associated with the distribution Φ. The customer parameters (w_k, e_k, s_k, l_k) of each of the customers of N are drawn randomly from the density ϕ. Associated with each customer is a job whose parameters are perturbed by Δ in the third and fourth coordinates, that is, $(w_k, e_k, s_k + \Delta, l_k + \Delta)$. This is equivalent to randomly drawing the job parameters from a density which we call ϕ'. The density ϕ' can be found simply by translating ϕ by Δ in the third and fourth coordinates, that is, for each $x = (\theta_1, \theta_2, \theta_3, \theta_4) \in \Re^4$, $\phi'(x) = \phi'(\theta_1, \theta_2, \theta_3, \theta_4) = \phi(\theta_1, \theta_2, \theta_3 - \Delta, \theta_4 - \Delta)$. Finally, for each $x \in \Re^4$, define $\psi(x) \equiv \min\{\phi(x), \phi'(x)\}$ and let $q \equiv \int_{\Re^4} \psi < 1$.

The n jobs (or customer parameters) $\{x_k \equiv (w_k, e_k, s_k + \Delta, l_k + \Delta) : k \in N\}$ are drawn randomly from the density ϕ' and our task is to select the set $T \subseteq N$. To simplify presentation, we refer interchangeably to the index set of jobs and to the set of jobs itself, i.e, $k \in N$ will have the same interpretation as $x_k \in N$ where $x_k = (w_k, e_k, s_k + \Delta, l_k + \Delta)$.

For each job x_k, generate a random value, call it u_k, uniformly in $[0, \phi'(x_k)]$. The point $(x_k, u_k) \in \Re^5$ is a point below the *graph* of ϕ', i.e., $u_k \leq \phi'(x_k)$. Define T as the set of indices of jobs whose u_k value falls below the graph of ϕ, that is $T = \{k \in N : u_k \leq \phi(x_k)\}$. Then the set of jobs $\{x_k : k \in T\}$ can be viewed as a random sample of $|T|$ jobs drawn randomly from the density ψ/q.

In order to relate this upper bound to the lower bound we need to present the following lemma.

Lemma 3.3 *For T generated as above and for each subregion A_i, $i = 1, 2, \ldots, t(\Delta)$,*

$$\overline{\lim_{n\to\infty}} \frac{M^*_\Delta(T(i))}{n(i)} \le \gamma, \text{ (a.s.)}.$$

Proof. To prove the result for a given subregion A_i, we construct a feasible schedule for the set of jobs $\{x_k = (w_k, e_k, s_k + \Delta, l_k + \Delta) : k \in T(i)\}$. Generate $n(i) - |T(i)|$ jobs randomly from the density

$$\frac{1}{1-q}[\phi - \psi].$$

Call this set of jobs D, for *dummy* jobs. From the construction of the sets D and $T(i)$, it is a simple exercise to show that the parameters of the jobs in $D \cup T(i)$ are distributed like ϕ.

A feasible schedule of the jobs in $T(i)$ is obtained by optimally scheduling the jobs in $D \cup T(i)$, using say M_i machines. The number of machines needed to schedule the jobs in $T(i)$ is obviously no more than M_i, since the jobs in D can simply be ignored. Thus we have the bound

$$M^*_\Delta(T(i)) \le M_i.$$

Now dividing by $n(i)$ and taking the limits, we get

$$\overline{\lim_{n\to\infty}} \frac{M^*_\Delta(T(i))}{n(i)} \le \overline{\lim_{n\to\infty}} \frac{M_i}{n(i)} = \gamma, \text{ (a.s.)},$$

since the set of jobs $D \cup T(i)$ is just a set of $n(i)$ jobs whose parameters are drawn independently from the density ϕ. ∎

Lemma 3.3 thus reduces equation (1) to

$$\begin{aligned}
\overline{\lim_{n\to\infty}} \frac{1}{n} Z_n^* &\le 2\sum_{i=1}^{t(\Delta)} \gamma \overline{\lim_{n\to\infty}} \frac{n(i)}{n} \underline{d}(i) + 2\Delta + \overline{\lim_{n\to\infty}} \frac{1}{n} \sum_{k\notin T} 2d_k \\
&= 2\gamma \overline{\lim_{n\to\infty}} \frac{1}{n} \sum_{i=1}^{t(\Delta)} n(i)\underline{d}(i) + 2\Delta + \overline{\lim_{n\to\infty}} \frac{1}{n} \sum_{k\notin T} 2d_k \\
&\le 2\gamma \overline{\lim_{n\to\infty}} \frac{1}{n} \sum_{k\in N} d_k + 2\Delta + \overline{\lim_{n\to\infty}} \frac{1}{n} \sum_{k\notin T} 2d_k \\
&= 2\gamma E(d) + 2\Delta + \overline{\lim_{n\to\infty}} \frac{1}{n} \sum_{k\notin T} 2d_k \\
&\le 2\gamma E(d) + 2\Delta + 2d_{\max} \overline{\lim_{n\to\infty}} \frac{1}{n} |N \setminus T|
\end{aligned}$$

The next lemma determines an upper bound on $\overline{\lim}_{n\to\infty} \frac{1}{n}|N \setminus T|$.

Lemma 3.4 *Given T and $\epsilon > 0$ generated as above,*

$$\overline{\lim_{n\to\infty}} \frac{1}{n}|N \setminus T| < (1+\epsilon)^2\epsilon, \text{ (a.s.).}$$

Proof. By the Strong Law of Large Numbers, the limit is equal to the probability that a job of N is not in the set T. The probability of a particular job x_k *not* being in T is simply

$$\begin{cases} \frac{\phi'(x_k)-\phi(x_k)}{\phi'(x_k)}, & \text{if } \phi'(x_k) \ge \phi(x_k), \\ 0, & \text{otherwise.} \end{cases}$$

Hence, almost surely

$$
\begin{aligned}
\overline{\lim_{n\to\infty}} \frac{1}{n}|N \setminus T| &= \int_{\Re^4} \max\Big\{\frac{\phi'(x)-\phi(x)}{\phi'(x)}, 0\Big\}\phi'(x)dx \\
&\le \int_{\Re^4} \Big|\frac{\phi'(x)-\phi(x)}{\phi'(x)}\Big|\phi'(x)dx \\
&= \int_{\Re^4} |\phi'(x)-\phi(x)|dx \\
&= \int_{\Re^4} |\phi'(\theta_1,\theta_2,\theta_3,\theta_4)-\phi(\theta_1,\theta_2,\theta_3,\theta_4)|d(\theta_1,\theta_2,\theta_3,\theta_4) \\
&= \int_{\Re^4} |\phi(\theta_1,\theta_2,\theta_3-\Delta,\theta_4-\Delta)-\phi(\theta_1,\theta_2,\theta_3,\theta_4)|d(\theta_1,\theta_2,\theta_3,\theta_4) \\
&< (1+\Delta)^2\epsilon \\
&< (1+\epsilon)^2\epsilon
\end{aligned}
$$

where the second to last inequality follows from $\|(\theta_1,\theta_2,\theta_3-\Delta,\theta_4-\Delta)-(\theta_1,\theta_2,\theta_3,\theta_4)\| \le \Delta\sqrt{2} < \delta$ and the continuity of ϕ. ∎

We now have all the necessary ingredients to finish the proof of Theorem 2.1.

$$
\overline{\lim_{n\to\infty}} \frac{1}{n} Z_n^* \le 2\gamma E(d) + 2d_{\max}(1+\epsilon)^2\epsilon + 2\Delta \qquad \text{(a.s.)}.
$$

Recalling that $\Delta < \epsilon$ and since ϵ was arbitrary, we have

$$
\overline{\lim_{n\to\infty}} \frac{1}{n} Z_n^* \le 2\gamma E(d), \qquad \text{(a.s.)}.
$$

This upper bound combined with the lower bound proves Theorem 2.1.

Having determined the asymptotic limit of Z_n^*/n, an interesting question is how fast does Z_n^*/n approach this limit? In other words, what is the rate of convergence of the optimal solution to its asymptotic value? Unfortunately, this is often very difficult to determine. For example, in the case of the CVRP, which is a special case of the VRPTW where no time windows are present, the rate of convergence of the optimal solution value to the asymptotic value is only known in some simple special cases of distributions. Results of Rhee (1991) and Bramel et al. (1992) show that under some specific assumptions on the distribution of the data in the CVRP, the expected difference between the optimal solution and its asymptotic limit is $O(n^{2/3})$. Since the CVRP is a special case of the VRPTW, the convergence in the VRPTW cannot be better than this.

4 An Asymptotically Optimal Heuristic

In this section we present a heuristic for the VRPTW called the Location Based Heuristic (LBH). The heuristic is based on work by Bramel and Simchi-Levi (1991) who developed a general framework for solving vehicle routing problems with capacity constraints, such as the Capacitated Vehicle Routing Problem (CVRP).

The CVRP is a special case of the VRPTW in which no time window constraints are present. For this problem, Bramel and Simchi-Levi (1991) prove that a version of the LBH is asymptotically optimal and provide empirical evidence that it works well on problems of moderate size.

In this section we generalize the LBH to handle time window constraints as well as capacity constraints and prove that the heuristic is asymptotically optimal for this general model also. We should emphasize that the introduction of time window constraints makes the analysis more difficult and requires major changes, relative to the capacitated models analyzed in Bramel and Simchi-Levi (1991).

4.1 The Location Based Heuristic

Before describing the heuristic, we present the intuition behind it. The upper bound developed in Section 3 is not constructive in the sense that it does not lend itself to an efficient algorithm for the VRPTW. It has however several important properties. The grid partitioning scheme breaks up the length of every tour in the upper bound in a natural way into two parts. The first is the length of a *simple tour* that brings the vehicle from the depot to the subregion and back. The second is the additional distance accrued by *inserting* customers into these simple tours. Therefore, if a heuristic assigns customers to vehicles so as to minimize the sum of the length of all simple tours plus the sum of all costs associated with inserting customers into each simple tour, then the solution provided by this heuristic will have the same structure as an asymptotically optimal solution.

The LBH has this property. It can be viewed as a three-step algorithm. In the first step, the parameters of the VRPTW are transformed into data for a location problem called the Capacitated Location Problem with Time Windows (CLPTW), described in Subsection 4.1.1. This location problem is solved in the second step. In the final step, we transform the solution to the CLPTW into a feasible solution to the VRPTW.

4.1.1 The Capacitated Location Problem with Time Windows

The Capacitated Location Problem with Time Windows (CLPTW) can be described as follows: we are given m possible sites to locate vehicles of capacity Q. There are n customers geographically dispersed in a given region, where customer i has w_i units of product that must be picked up by a vehicle. The pick-up of customer i takes s_i units of time and must occur in the time window between times e_i and l_i, that is, the service of customer i can start at any time $t \in [e_i, l_i - s_i]$. The objective is to select a subset of the possible sites, to locate one vehicle at each site, and to assign the customers to the vehicles. Each vehicle must leave its site, pick-up the load of customers assigned to it in such a way that the vehicle capacity is not exceeded and all pick-ups occur within the customer's time window, and then return to its site. The costs are as follows: a site-dependent cost is incurred for locating each vehicle; that is, if a vehicle is located at site j, the *set-up* cost is v_j, for $j = 1, 2, \ldots, m$.

The cost of assigning customer i to the vehicle at site j is c_{ij} (the *assignment* cost), for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, m$. We assume that there are enough vehicles and sites so that a feasible solution exists.

The CLPTW can be formulated as the following mathematical program. Let

$$y_j = \begin{cases} 1, & \text{if a vehicle is located at site } j, \\ 0, & \text{otherwise,} \end{cases}$$

and let

$$x_{ij} = \begin{cases} 1, & \text{if customer } i \text{ is assigned to the vehicle at site } j, \\ 0, & \text{otherwise.} \end{cases}$$

For any set S, let $f_j(S) = 1$ if the set of customers S can be feasibly served in their time windows by one vehicle that starts and ends at site j (disregarding the capacity constraint), and 0 otherwise.

$$\text{Problem } P : Min \qquad \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} x_{ij} + \sum_{j=1}^{m} v_j y_j$$

$$s.t. \qquad \sum_{j=1}^{m} x_{ij} = 1 \qquad \forall i, \tag{2}$$

$$\sum_{i=1}^{n} w_i x_{ij} \leq Q \qquad \forall j, \tag{3}$$

$$x_{ij} \leq y_j \qquad \forall i, j, \tag{4}$$

$$f_j(\{i : x_{ij} = 1\}) = 1 \qquad \forall j, \tag{5}$$

$$x_{ij} \in \{0, 1\} \ \forall i, j \text{ and } y_j \in \{0, 1\} \ \forall j \tag{6}$$

Constraints (2) ensure that each customer is assigned to exactly one vehicle, and constraints (3) ensure that the vehicle's capacity constraint is not violated. Constraints (4) guarantee that if a customer is assigned to the vehicle at site j, then a vehicle is located at that site. Constraints (5) ensure that the time window constraints are not violated. Constraints (6) ensure the integrality of the variables.

4.1.2 The Heuristic

To relate the CLPTW to the VRPTW, consider each customer in the VRPTW to be a potential site for a vehicle, that is, the set of potential sites is exactly the set of customers, and therefore $m = n$. Picking a subset of the sites in the CLPTW corresponds to picking a subset of the customers in the VRPTW; we call this set of selected customers the *seed* customers. These customers are those that will form simple tours with the depot.

In order for the LBH to perform well, the costs of the CLPTW should approximate the costs of the VRPTW. The set-up cost for locating a vehicle at site j (v_j), or, in other words, of picking customer j as a seed customer, should be the cost of sending a vehicle from the depot to customer j and back (i.e., the length of the simple tour). Hence, we set $v_j = 2d_j$, for each $j \in N$. The assignment cost c_{ij} is the cost of assigning customer i to the vehicle at site j. Therefore, this cost should represent the added cost of inserting customer i into the simple tour through the depot and customer j. Consequently, when i is added to a tour with j, the added cost is $c_{ij} = d_i + d_{ij} - d_j$; so that $v_j + c_{ij} = d_i + d_{ij} + d_j$. However, when a third customer is added, the calculation is not so simple, and therefore the values of c_{ij} should in fact represent an *approximation* to the cost of adding i to a tour that goes through customer j and the depot. In Section 4.3 we present values of c_{ij} that we have found to work well in practice.

Once these costs are determined the second step of the LBH consists of solving CLPTW. The solution provided is a set of sites (seed customers) and a set of customers assigned to each of these sites (to each seed). This solution can then be easily transformed into a solution to the VRPTW, since a set of customers that can be feasibly served starting from site j can also be feasibly served starting from the depot.

4.2 A Solution Method for CLPTW

The computational efficiency of the LBH depends on the efficiency in which CLPTW can be solved. We therefore present a method to solve the CLPTW. The CLPTW without constraints (5) is simply the Capacitated Concentrator Location Problem (CCLP) for which efficient solution methods exist based on the celebrated Lagrangian relaxation technique; see Pirkul (1987) or Bramel and Simchi-Levi (1991). For the CLPTW, we use a similar method, though the specifics are much more complex in view of the existence of the time window constraints.

In this technique, the problem is relaxed by omitting a set of constraints and including them in the objective function with a multiplier vector. In this case, for a given multiplier vector $\lambda \in \Re^n$, constraints (2) are relaxed and put into the objective function with the multiplier vector. The resulting problem can be separated into n subproblems (one for each of the n sites), since constraints (2) are the only constraints that relate the sites to one another. The subproblem for site j is:

$$
\begin{aligned}
\text{Problem } P_j : Min \quad & \sum_{i=1}^{n} \bar{c}_{ij} x_{ij} + v_j y_j \\
s.t. \quad & \sum_{i=1}^{n} w_i x_{ij} \le Q \\
& x_{ij} \le y_j \qquad \forall i \\
& f_j(\{i : x_{ij} = 1\}) = 1 \\
& x_{ij} \in \{0,1\} \; \forall i \text{ and } y_j \in \{0,1\}
\end{aligned}
$$

where $\bar{c}_{ij} \equiv c_{ij} + \lambda_i$, for each $i \in N$.

In the optimal solution to problem P_j, y_j is either 0 or 1. If $y_j = 0$ then $x_{ij} = 0$ for all $i \in N$, and the objective function value is 0. If $y_j = 1$ then the problem reduces to a different, but simpler, routing problem. Consider a vehicle of capacity Q initially located at site j. The driver gets a profit of $p_{ij} \equiv -\overline{c}_{ij}$ for picking up the w_i items at customer i in the time window (e_i, l_i). The pick-up operation takes s_i units of time. The objective is to choose a subset of the customers, to pick up their loads in their time windows, without violating the capacity constraint, using a vehicle which must begin and end at site j, while maximizing the driver's profit. Let G_j^* be the maximum profit attainable at site j, that is, G_j^* is the optimal solution to the problem just described for site j. This implies that $v_j - G_j^*$ is the optimal solution value of Problem P_j given that $y_j = 1$. Therefore, we can write the optimal solution to Problem P_j as simply $\min\{0, v_j - G_j^*\}$.

Unfortunately, in general determining the values G_j^* for $j \in N$ is NP-hard. We can however determine upper bounds on G_j^*, call them G_j. This provides a *lower bound* on the optimal solution to problem P_j which is equal to $\min\{0, v_j - G_j\}$. We use the bound given by $G_j \equiv \sum_{\{i:p_{ij}>0\}} p_{ij}$. Consequently, $\sum_{j=1}^{n} \min\{0, v_j - G_j\} - \sum_{i=1}^{n} \lambda_i$, is a lower bound on the optimal solution to the CLPTW.

To generate a feasible solution to the VRPTW at each iteration of the procedure, we use information from the upper bounds on profit, G_j, for $j \in N$. After every iteration of the lower bound (for each multiplier) we renumber the sites so that $G_1 \geq G_2 \geq \cdots \geq G_n$. The upper bounds on profit are used as an estimate of the profitability of placing a vehicle at a particular site. For example, site 1 is considered to be a "good" site (or seed customer), since a large profit is possible there. A large profit for site j corresponds to a seed customer where neighboring customers can be feasibly served from it at *low cost*. Therefore, a site with large profit is selected as a seed customer since it will tend to have neighboring customers around it that can be feasibly served by a vehicle starting at that site.

To generate a feasible solution to CLPTW, we do the following: starting with $j = 1$ in the new ordering of the sites (customers), we locate a vehicle at site j. For every customer still not assigned to a site, we first determine if this customer can be feasibly served with the customers that are currently assigned to site j. Then, of the customers that can be served from this site, we determine the one which will cause the least increase in cost, that is, the one with minimum c_{ij} over all customers i that can be served from this site. We then assign this customer to the site. We continue until no more customers can be assigned to site j, due to capacity or time constraints. We then increment j to 2 and continue with site 2. After all customers have been feasibly assigned to a site, we obtain a feasible solution whose cost is compared to the cost of the current best solution.

As we find solutions to the CLPTW, we also generate feasible solutions to the VRPTW, using the information from the lower bound to CLPTW. Starting with $j = 1$, pick customer j as a seed customer. Then, for every customer that can be feasibly served with this seed, we determine the added distance this would entail, that is, we determine the best place to insert the customer into the current tour through the customers assigned to seed j. We choose the customer that causes the least increase in distance traveled as the one to assign to seed j. This idea is similar to the Nearest Insertion heuristic of Rosenkrantz, Stearns and Lewis (1977). We then continue trying to add customers in this way to seed j. Once no more can be added to this tour (due to capacity or time constraints), we increment j to 2, select seed customer 2 and continue. Once every customer appears in a tour, that is, every customer is assigned to a seed, we have a feasible solution to the VRPTW corresponding to the current set of multipliers. The cost of this solution is compared to the cost of the current best solution.

Multipliers are updated using the well-known formula of Held, Wolfe and Crowder (1974). The step size is initially set to 2 and halved after the lower bound has not improved in a series of 30 iterations. After the step size has reached a preset minimum (0.05), we terminate the heuristic.

4.3 Implementation

It is clear that many possible variations of the LBH can be implemented depending on the type of assignment costs (c_{ij}) used. In our computational work, we have implemented the following:

$$\textit{direct cost}: \ c_{ij} = 2d_{ij}, \quad \text{and}$$

$$\textit{nearest insertion cost}: \ c_{ij} = d_i + d_{ij} - d_j.$$

Direct cost c_{ij} has the advantage that, when several customers are added to the seed, the resulting cost, which is the sum of the set-up costs and these direct costs, is an upper bound on the length of any efficient route through the customers. On the other hand, the nearest insertion cost works well because it is accurate at least for tours through two customers, and often for tours through three customers as well.

We have implemented several versions of the LBH. In the first, the Star-Tours (ST) heuristic, we use the direct assignment cost, while in the second, the Seed-Insertion (SI) heuristic, we use the nearest insertion assignment cost. We remark that the LBH is not a polynomial-time heuristic. However, as we shall shortly demonstrate, the running times reported on standard test problems are very reasonable and is comparable to the running time of many heuristics for the vehicle routing problem.

The Star-Tours (ST) heuristic is of particular interest because it is asymptotically optimal as demonstrated in the following lemma. The proof is similar to the previous proofs and is therefore omitted.

Lemma 4.1 *Let n customers, indexed by N, be independently and identically distributed according to a distribution μ with compact support in $\Re^2$. Let the customer parameters $\{(w_k, e_k, s_k, l_k) : k \in N\}$ be jointly distributed like Φ. In addition, let M_n^* be the minimum number of machines needed to feasibly schedule the jobs $\{(w_k, e_k, s_k, l_k) : k \in N\}$ and let*

$\lim_{n\to\infty} M_n^*/n = \gamma$, (a.s.). *Then,*

$$\lim_{n\to\infty} \frac{1}{n} Z_n^{\mathrm{ST}} = \lim_{n\to\infty} \frac{1}{n} Z_n^* = 2\gamma E(d) \quad \text{(a.s.)}.$$

4.4 Numerical Study

In Tables 1 and 2 we summarize our computational experiments with the standard test problems of Solomon (1986). The problem set consists of 56 problems of various types. All problems consist of 100 customers and one depot, and the distances are Euclidean. Problems with the "R" prefix are problems where the customer locations are randomly generated according to a uniform distribution. Problems with the "C" prefix are problems where the customer locations are clustered. Problems with the "RC" prefix are a mixture of both random and clustered. In addition, all the problems have a constraint on the latest time, T_0, at which a vehicle can return to the depot. For a full description of these problems we refer the reader to Solomon (1986).

As far as we know, the only methods that have been implemented (and reported in the literature) on these test problems are the heuristics of Solomon (1986) and the column generation approach of Desrochers, Desrosiers and Solomon (1992). The latter method was only able to solve 7 of the 56 test problems.

To compare the LBH to these solution methods, we have implemented a time window reduction phase before the start of the heuristic. Here, the earliest time for service, e_k, is replaced by $\max\{e_k, d_k\}$, in that way, vehicles leave the depot no earlier than time 0. In addition, the latest time service can end, l_k, is replaced by $\min\{l_k, T_0 - d_k\}$. The LBH can then be run as it is described in Section 4.1.2.

We implemented both the Star-Tours (ST) and the Seed-Insertion (SI) heuristics. CPU times are in seconds on a Sun SPARC Station II. In Tables 1 and 2, the column "Solomon's

Best Solution" corresponds to the best solution found by Solomon (1986). Solomon tested eight different heuristics on problem sets R1 and C1, and six heuristics on problems RC1, R2, C2 and RC2. We see that the ST heuristic provides a better solution than Solomon's heuristics in 25 of the 56 problems, while the SI heuristic provides a better solution in 21 of the 56 problems. In Table 2, the column "DDS Solution Value" corresponds to the value of the solution found using the column generation approach of Desrochers, Desrosiers and Solomon.

We remark that the running time of the LBH is comparable to the running time of many heuristics designed for vehicle routing problems. We observe that the method of Desrochers, Desrosiers and Solomon took on average 15 times longer than the LBH on comparable machines.

Table 1

Problem	Alg. ST	CPU Time	Alg. SI	CPU Time	Solomon's Best Solution
C201	591.6	245.9s	591.6	260.5s	591
C202	* 652.8	276.1s	* 640.8	262.7s	731
C203	* 692.2	309.2s	* 741.1	308.9s	786
C204	* 721.6	335.9s	782.3	340.6s	758
C205	713.8	250.8s	699.9	258.8s	606
C206	770.8	257.3s	* 722.8	283.3s	730
C207	767.2	265.7s	708.9	275.8s	680
C208	736.2	287.7s	660.2	272.4s	607
R201	*1665.3	207.1s	*1533.4	209.6s	1741
R202	*1485.3	276.4s	*1484.3	248.5s	1730
R203	*1371.5	406.5s	*1349.3	389.0s	1567
R204	1096.7	532.0s	1077.0	538.2s	1059
R205	1472.3	287.0s	*1329.4	312.6s	1471
R206	*1237.0	412.2s	*1283.7	374.2s	1405
R207	*1217.7	484.8s	*1162.9	453.9s	1241
R208	* 966.1	587.8s	* 959.9	612.6s	1046
R209	*1276.1	394.8s	*1262.8	355.7s	1418
R210	*1312.5	380.7s	*1340.6	388.6s	1425
R211	1080.9	474.7s	1141.3	488.7s	1016
RC201	*1873.8	203.5s	*1841.7	185.8s	1880
RC202	*1742.1	227.8s	*1705.1	241.0s	1799
RC203	*1417.5	331.5s	*1471.1	300.1s	1550
RC204	*1139.6	437.7s	*1190.3	411.5s	1208
RC205	*1830.5	233.0s	*1878.9	214.0s	2080
RC206	1640.1	259.0s	1607.5	248.2s	1582
RC207	*1566.4	294.2s	*1557.3	272.3s	1632
RC208	1254.8	345.7s	1298.7	317.3s	1194

(* indicates that the LBH improves upon the best solution known.)

Table 2

Problem	Alg. ST	CPU Time	Alg. SI	CPU Time	Solomon's Best Solution	DDS Solution Value
C101	828.9	74.1s	828.9	67.0s	829	827.3
C102	982.8	82.9s	1043.4	73.1s	968	827.3
C103	*1015.1	95.9s	1232.9	88.4s	1026	
C104	* 980.9	105.4s	* 976.1	114.5s	1053	
C105	* 828.9	79.7s	860.8	67.3s	829	
C106	852.9	82.8s	880.1	66.7s	834	827.3
C107	828.9	83.1s	841.2	74.7s	829	827.3
C108	852.9	88.6s	853.6	80.9s	829	827.3
C109	991.0	88.6s	1014.5	83.1s	829	
R101	1983.7	57.2s	2071.2	39.9s	1873	1607.7
R102	1789.0	70.8s	1821.4	57.4s	1843	1434.0
R103	1594.5	88.6s	1599.1	67.9s	1484	
R104	1242.0	106.2s	1237.3	81.0s	1188	
R105	1604.4	67.0s	1696.2	52.0s	1502	
R106	1606.9	78.0s	1589.2	70.0s	1460	
R107	*1324.9	92.4s	1361.2	70.4s	1353	
R108	1202.6	107.5s	1205.5	101.1s	1134	
R109	1504.7	78.5s	1491.8	69.6s	1412	
R110	1380.9	92.0s	1434.4	69.4s	1211	
R111	1422.1	91.7s	1432.4	69.5s	1202	
R112	1248.1	105.2s	1284.6	79.4s	1086	
RC101	2045.1	60.6s	2014.4	45.0s	1867	
RC102	1806.6	68.7s	1969.5	52.2s	1760	
RC103	1708.9	81.7s	1716.3	69.6s	1641	
RC104	1372.1	93.5s	1458.8	79.5s	1301	
RC105	*1826.3	68.9s	2036.8	51.3s	1922	
RC106	1710.8	68.0s	1804.8	50.5s	1611	
RC107	1593.2	76.4s	1630.9	64.9s	1385	
RC108	1421.0	84.7s	1493.8	65.5s	1253	

(* indicates that the LBH improves upon the best solution known.)

Acknowledgements

We would like to thank Prof. Marius Solomon for providing the data for the computational tests.

5 References

Bienstock, D. , J. Bramel and D. Simchi-Levi (1993), A Probabilistic Analysis of Tour Partitioning Heuristics for the Capacitated Vehicle Routing Problem with Unsplit Demands. *Mathematics of Operations Research* **18**, pp. 786–802.

Bramel, J. , Coffman, E. G. Jr. , Shor, P. and D. Simchi-Levi (1991), Probabilistic Analysis of Algorithms for the Capacitated Vehicle Routing Problem with Unsplit Demands. *Operations Research* **40**, pp. 1095–1106.

Bramel, J., C-L. Li and D. Simchi-Levi (1991), Probabilistic Analysis of the Vehicle Routing Problem with Time Windows. Working Paper, Dept. of IE&OR, Columbia University.

Bramel, J. and D. Simchi-Levi (1991), A Location Based Heuristic for General Routing Problems. Working Paper, Dept. of IE&OR, Columbia University.

Christofides, N. and S. Eilon (1969), An Algorithm for the Vehicle Dispatching Problem. *Operations Research Quarterly* **20**, pp. 309–318.

Christofides, N. , A. Mingozzi and P. Toth (1981a), State-Space Relaxation Procedures for the Computations of Bounds to Routing Problems. *Networks* **11**, pp. 145–164.

Christofides, N., A. Mingozzi and P. Toth (1981b), Exact Algorithms for the Vehicle Routing Problem Based on Spanning Tree and Shortest Path Relaxations. *Mathematical Programming* **20**, pp. 255–282.

Daganzo, C. F. (1987a), Modeling Distribution Problems with Time Windows: Part I. *Transportation Science* **21**, pp. 171–179.

Daganzo, C. F. (1987b), Modeling Distribution Problems with Time Windows: Part II: Two Customer Types. *Transportation Science* **21**, pp. 180–187.

Desrochers, M. , J. Desrosiers and M. Solomon (1992), A New Optimization Algorithm for the Vehicle Routing Problem with Time Windows. *Operations Research* **40**, pp. 342–354.

Federgruen, A. and G. van Ryzin (1992), Probabilistic Analysis of a Generalized Bin Packing Problem with Applications to Vehicle Routing and Scheduling Problems. Working Paper, Graduate School of Business, Columbia University.

Held, M. , P. Wolfe and H. P. Crowder (1974), Validation of Subgradient Optimization. *Mathematical Programming* **6**, pp. 62–88.

Kingman, J. F. C. (1976), Subadditive Processes. *Lecture Notes in Math. 539*, Springer-Verlag, Berlin, pp. 168–222.

Lawler, E. L., J. K. Lenstra, A. H. G. Rinnooy Kan and D. B. Shmoys (1989), Sequencing and Scheduling: Algorithms and Complexity. Report BS-R89xx, Center for Mathematics and Computer Science, Amsterdam, The Netherlands.

Pirkul, H. (1987), Efficient Algorithms for the Capacitated Concentrator Location Problem. *Computers in Operations Research* **14**, pp. 197–208.

Rhee, W. T. (1991), An Asymptotic Analysis of Capacitated Vehicle Routing. Working Paper, The Ohio State University, Columbus, OH.

Rosenkrantz, D. J., R. E. Stearns and P. M. Lewis II (1977), An Analysis of Several Heuristics for the Traveling Salesman Problem. *SIAM J. on Computing* **6**, pp. 563–581.

Simchi-Levi, D. and J. Bramel (1990), On the Optimal Solution Value of the Capacitated Vehicle Routing Problem with Unsplit Demands. Working Paper, Department of IE&OR, Columbia University, New York.

Solomon, M. M. (1986), On the Worst-Case Performance of Some Heuristics for the Vehicle

Routing and Scheduling Problem with Time Window Constraints. *Networks* **16**, pp. 161–174.

Solomon, M. M. and J. Desrosiers (1988), Time Window Constrained Routing and Scheduling Problems: A Survey. *Transportation Science* **22**, pp. 1–13.