Solving Min-Max Multi-Depot Vehicle Routing Problem*

John Carlsson[†], Dongdong Ge[‡], Arjun Subramaniam[§], Amy Wu[¶] and Yinyu Ye[‖]

1st May 2007

Abstract

The Multi-Depot Vehicle Routing Problem (MDVRP) is a generalization of the Single-Depot Vehicle Routing Problem (SDVRP) in which vehicle(s) start from multiple depots and return to their depots of origin at the end of their assigned tours. The traditional objective in MDVRP is to minimize the sum of all tour lengths, and existing literature handles this problem with a variety of assumptions and constraints. In this paper, we explore the notion of minimizing the maximal length of a tour in MDVRP ("min-max MDVRP"). We present two heuristics in the paper. The first heuristic is a linear programming-based approach with global improvement. The second one, the region partition heuristic, is proved to be asymptotically optimal and is potentially useful for general network applications. A comparison of the computational implementations for different heuristics is presented.

Key words: vehicle routing problem; region partition; heuristic.

1 Introduction

The Vehicle Routing Problem (VRP) has been one of the central topics in optimization since Dantzig proposed the problem in 1959 [7]. A simple general model of VRP can be described as follows: a set of service vehicles need to visit all customers in a geographical region with the minimum cost.

In the Single-Depot Vehicle Routing Problem (SDVRP), multiple vehicles leave from a single location (a "depot") and must return to that location after completing their assigned tours. The Multi-Depot Vehicle Routing Problem (MDVRP) is a generalization of SDVRP in which multiple vehicles start from multiple depots and return to their original depots at the end of their assigned tours. The traditional objective in MDVRP is to minimize the sum

*This research is supported in part by the Boeing Company.

[†]Institute of Computational and Mathematical Engineering.

[‡]Department of Management Science and Engineering.

[§]Department of Management Science and Engineering.

[¶]Department of Computer Science.

[‖]Department of Management Science and Engineering and, by courtesy, Electrical Engineering, E–mail: yinyu-ye@stanford.edu; Stanford University, Stanford, CA 94305, USA.

of all tour lengths, and existing literature handles this problem with a variety of assumptions and constraints.

In this paper, we explore the notion of minimizing the maximal length of a tour in a MDVRP ("min-max MDVRP") using both theoretical analysis and heuristic design. To the best of our knowledge, no prior exploration of min-max MDVRP has been published, but this formulation is advantageous for a number of applications. For example, consider a network model in which depots represent servers and customers represent clients. A network routing topology generated by solving min-max MDVRP results in a set of daisy-chain network configurations that minimize the maximum latency between a server and client. This can be advantageous in situations in which the server-client connection cost is high but the client-client connection cost is low.

The formulation of min-max MDVRP is as follows, with the assumption that all points are randomly and uniformly distributed in a Euclidean plane:

$$\begin{array}{ll} \text{minimize} & \lambda \\ \text{subject to} & TSP(S_i) \leq \lambda, \ \forall i \\ & \cup S_i = \mathcal{N}, \end{array}$$

where $\mathcal{N}$ is the set of all customers, $S_i(\subset \mathcal{N})$ is the subset of customers assigned to vehicle i, and $TSP(S)$ is the minimal TSP tour length that visits all customers in set S.

The implementation results demonstrate the efficiency of the heuristic methods developed in this paper, as they reduce MDVRP to a sequence of smaller-scale TSP problems in polynomial time. In addition, the sum of all tour lengths that our methods generate is highly comparable to the minimized total lengths produced by traditional approaches. And these methods are also capable of quickly processing tens of thousands of customers, a scalability property which is becoming increasingly important as networks expand in size.

In this paper we first give a theoretical analysis for the optimal solution of min-max MDVRP by developing lower and upper bounds. We show that the optimal solution to min-max MDVRP is asymptotically close to the optimal TSP tour of all customers divided by the number of depots when the ratio of customers to depots is high. The same conclusion applies to the solution generated by a region partition algorithm when the number of depots is fixed.

Then we develop two heuristics to min-max MDVRP. The first heuristic is a linear programming-based heuristic with a global load balancing technique. By starting from a simple linear programming-based algorithm, we rapidly assign customers to depots and generate TSP routes for each vehicle. The solution is further improved in a post refinement period in which we keep changing certain parameters globally. Next, noticing that a convex equitable region partition yields an even division of points (i.e., if we divide the service region into a set of subregions with equal area, then each subregion will contain – asymptotically – the same number of points), we propose a fast approximation algorithm to generate good initial solutions for min-max MDVRP. The solution can then be improved with common local improvement procedures. Moreover, this region partition method, for which we demonstrate both theory and implementation, is potentially useful in many network design problems with regular topology structures, such as the Steiner tree and the minimum spanning forest problem.

The simulation results of these two methods and a routine heuristic are compared and analyzed in the performance analysis section. Finally, we summarize our results and address future research issues.

2 Asymptotic Bounds on Min-Max MDVRP

2.1 Analysis of Optimal Solutions

Since the MDVRP is NP-hard, researchers have made compelling efforts to design heuristics; in addition, exploring the theoretical bounds of this problem has been an intriguing topic as well. Baltz et al. proved that the sum of all tour lengths asymptotically approaches $\alpha_k n^{\frac{d-1}{d}}$ with the uniform distribution, and α_k depends on the number of depots. In this section, we provide lower and upper bounds for min-max MDVRP in a planar graph and prove its asymptotic convergence for a broad class as the problem size expands. To provide context, we will first review known results for the probabilistic TSP. The most celebrated discovery is the *BHH* theorem (1959) by Beardwood, Halton and Hammersley.

Theorem 1. *(BHH Theorem) Suppose X_i's, $1 \leq i \leq n$, are independent and identically distributed random points uniformly distributed in a unit cube $[0,1]^d, d \geq 2$. Then with probability one, the length $TSP(\{X_1, X_2, \cdots X_n\})$ of an optimal TSP tour traversing all points satisfies:*

$$\lim_{n\to\infty} TSP(\{X_1, X_2, \cdots, X_n\})/n^{\frac{d-1}{d}} = \alpha(d),$$

where $\alpha(d)$ is a positive constant depending on d.

Considering that we are only interested in planar graphs, we assume all customers and depots are uniformly and randomly distributed in a unit square, and define $\alpha \equiv \alpha(2)$ in our discussion. Denote the set of depots by $\mathcal{D} = \{D_1, D_2, \cdots, D_m\}$, $|\mathcal{D}| = m$. Denote the set of vehicles by $\mathcal{V} = \{V_1, V_2, \cdots, V_k\}$, $|\mathcal{V}| = k \geq m$. In the case $k = m$, we always assume each depot has one single vehicle. Denote the set of nodes(or customers) by $\mathcal{N} = \{N_1, N_2, \cdots, N_n\}$, $|\mathcal{N}| = n$. We will use node and customer interchangeably in this paper. Denote the value of an optimal solution to min-max MDVRP by L, i.e., L is the length of the longest tour. Denote the largest distance between an arbitrary pair of points in two different sets A and B by $d(A, B)$, i.e., $d(A, B) = \max_{x\in A, y\in B} ||x - y||$.

First we have the following observation:

Lemma 2. *For a general planar graph (points do not necessarily follow any distribution),*

$$\frac{TSP(\mathcal{D} \cup \mathcal{N}) - TSP(\mathcal{D})}{k} \leq L \leq \frac{TSP(\mathcal{N})}{k} + 2 * d(\mathcal{D}, \mathcal{N}).$$

Proof. Consider the lower bound first.

In an optimal pattern of min-max MDVRP, assume the sum of all tour lengths is S_{opt}. Note that L is the longest tour, we have

$$k * L \geq S_{opt}.$$

Add an optimal TSP tour T for all the *depots* to this pattern. Now each point(node or depot) in the graph has an even degree, which implies an Euler tour. This Euler tour can be reduced to a feasible TSP tour for the set of all the depots and nodes. Then,

$$S_{opt} + TSP(\mathcal{D}) \geq TSP(\mathcal{D} \cup \mathcal{N}).$$

Therefore,

$$L \geq \frac{TSP(\mathcal{D} \cup \mathcal{N}) - TSP(\mathcal{D})}{k}.$$

For the upper bound, consider a tour partition heuristic.

Heuristic 1 Tour Partition Heuristic

1. Generate an optimal TSP tour for all the nodes.
2. Partition this tour into k equal subtours $t_1, t_2, \cdots, t_k$.
3. Connect both the starting and ending nodes of subtour t_i to the depot where vehicle i stays.

This heuristic generates a feasible solution for the MDVRP. The maximal tour length, which is an upper bound of L, is at most

$$\frac{TSP(\mathcal{N})}{k} + 2 * d(\mathcal{D}, \mathcal{N}).$$

□

The proof implies that the lower bound holds even when L is the average length of all the tours, so it is not tight. However, when the ratio of nodes to depots is high, we can derive the asymptotic bounds for L.

Corollary 3. *(a) When $k = o(\sqrt{n})$,*

$$\lim_{n\to\infty} \frac{L}{\sqrt{n}/k} = \alpha.$$

(b) For non-fixed MDVRP, i.e., a vehicle can start from and return to arbitrary different depots, if $n = \Omega(k \log^2 k)$ and $k = m$,

$$\lim_{n\to\infty} \frac{L}{\sqrt{n}/k} \leq \alpha.$$

Proof. (a) According to the BHH theorem, for arbitrarily small ϵ, when n, m are sufficiently large,

$$\frac{\alpha((1-\epsilon)\sqrt{m+n} - (1+\epsilon)\sqrt{m})}{k} \leq L \leq \frac{\alpha(1+\epsilon)\sqrt{n}}{k} + 2d(\mathcal{D}, \mathcal{N}) + \epsilon. \quad (1)$$

Noticing that $m \leq k$ and $k = o(\sqrt{n})$, we know $\sqrt{m}/k$ is less than 1, while $\frac{\sqrt{m+n}}{k}$ goes to infinity as n goes to infinity. On the other side, $d(\mathcal{D}, \mathcal{N})$ is bounded by a constant while $\sqrt{n}/k$ goes to infinity as n goes to infinity.

(b) Consider the tour partition heuristic. Then at Step 3 in Heuristic 1, for each subtour t_i, instead of connecting its starting and ending nodes to the depot where vehicle i stays, we consider a perfect matching between depots and starting points of subtours, and another perfect matching between depots and ending points of subtours.

The min-max perfect matching for uniformly distributed points has the bound provided by Leighton and Shor in 1989 [13]: if X_i's and Y_i's are independent and uniformly distributed points on a unit square for $1 \leq i \leq r$, then there exists a constant C, such that

$$\min_{\sigma \in \mathcal{P}} \max_{1 \leq i \leq r} ||X_i - Y_{\sigma(i)}|| \leq C r^{-\frac{1}{2}} (\log r)^{\frac{3}{4}} \tag{2}$$

with probability one as $r \to \infty$, where $\mathcal{P}$ is the set of all permutations of $\{1, 2, \cdots, r\}$.

First apply the bound (2) to depots and all starting points of subtours. Then apply it again to depots and all ending points of subtours. We get $L \leq \frac{\alpha(1+\epsilon)\sqrt{n}}{k} + 2Ck^{-\frac{1}{2}}(\log k)^{\frac{3}{4}}$. With the assumption $n = \Omega(k \log^2 k)$, we complete the proof.

□

For the general case $k = \lambda n + o(n)$, $\lambda \geq 0$, Baltz et al. [3] proved that the sum of all tour lengths asymptotically approaches $\alpha' \sqrt{n}$ as $n \to \infty$, where the constant α' is equal to the TSP constant α for the case $\lambda = 0$ and depends on λ for the case $\lambda > 0$. Therefore, we can obtain a direct lower bound according to their results.

Lemma 4. *When $k = \lambda n + o(n)$, $\lambda \geq 0$,*

$$\lim_{n \to \infty} \frac{L}{\sqrt{n}/k} \geq \alpha',$$

where the constant α' is defined as above.

2.2 Bounds by Region Partition Heuristics

Based on the discussion above, we can conclude that the optimal solution to min-max MDVRP with uniform distributed points will numerically approach $\alpha\sqrt{n}/k$, the value of the optimal TSP tour of all customers split by the number of vehicles, under the constraint $k = o(\sqrt{n})$. This matches our intuition, although the constraint is restrictive. We would like to relax this constraint to a broader class and derive some nontrivial bounds by analyzing local performance in large size problems. In this section, we will analyze a region partition heuristic with theoretically good performances, and an estimation on upper bounds for all cases $k = \Omega(\sqrt{n})$ derived from a grid region partition.

Before presenting the heuristics, we need several lemmas to review some facts:

Theorem 5. *(Convex Region Partition Theorem) Given k points in a convex bounded planar polygon, it is always possible to find a partition of the domain into k equal-area convex polygons, with exactly one point in each face.*

This convex region partition theorem was proved in [4] for both continuous and discrete versions, and an algorithm for discrete version was also given. An algorithm for the continuous version, i.e. the convex region partition described in the theorem was recently revealed by Carlsson and Armbruster [1].

For the time being, we assume an equitable partition exists and develop a theoretical rationale for the desirability of such a partition. The next two lemmas will prepare us some important backgrounds for analyzing region partition approaches.

Lemma 6. *(Occupancy Lemma) Randomly put n balls into m bins with equal probability. If $n = \Omega(m \log^2 m)$, then the number of balls in the bin holding the least balls has an asymptotic performance as $\frac{n}{m}$. So does the number of balls in the bin holding the most balls.*

Proof. This fact can be proved by Chernoff bounds. We will give a sketch of the proof here.

We only consider the case $n = m \log^2 m$ here. For an arbitrary bin, define x_i is 1 if the ith ball falls into this bin, and 0 otherwise. Let $S_i = x_1 + x_2 + \cdots + x_i$. Note that the expected number of balls falling into a bin is $\log^2 m$. From the Chernoff bounds [12],

$$P = Prob(S_n \leq \frac{n}{m}(1 - 2\sqrt{\frac{2}{\log m}})) \leq e^{-nD(p-px||p)},$$

where $p = \frac{1}{m}$, $x = 2\sqrt{\frac{2}{\log m}}$, $D(a||b) = a \log \frac{a}{b} + (1-a) \log \frac{1-a}{1-b}$.

When m is sufficiently large, we have

$$e^{-nD(p-px||p)} \leq e^{-n(px^2/4)} = \frac{1}{m^2}.$$

Therefore, the probability that one bin has less than $\frac{n}{m}(1 - 2\sqrt{\frac{2}{\log m}})$ balls is at most $mP \leq \frac{1}{m}$.

Similarly, by applying another Chernoff bound, we can also prove that the probability that one bin has more than $\frac{n}{m}(1 + 2\sqrt{\frac{2}{\log m}})$ balls is at most $\frac{1}{m}$. □

Simple scaling arguments show that the BHH theorem still holds even if the unit cube is replaced by an arbitrary compact subset K in R^d in [18]. This fact suggests that the limit is independent of the shape of the compact set K.

Theorem 7. *(generalized BHH Theorem [18]) In particular, if for $i \geq 1$, X_i's are i.i.d with uniform distribution on a compact set K of Lebesgue measure one, the BHH theorem still holds.*

We consider a region partition heuristic for min-max MDVRP for the case $k = m$.

Heuristic 2 Region Partition Heuristic

---

1. Divide the region into k equitable convex polygons such that each polygon has exactly one vehicle inside.

2. Find an optimal TSP tour for each vehicle and the nodes in the same subregion.

With the fixed number of depots, we have

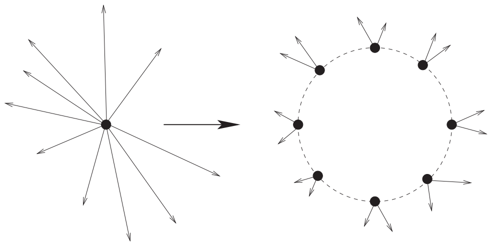

Figure 1: The perturbation method, and its subsequent effects on assignment and routes generation. Here the routes are suggested by arrows emanating from each depot node.

Lemma 8. *If k=O(1), the solution L generated by Heuristic 2 satisfies:*

$$\lim_{n\to\infty} \frac{L}{\sqrt{n}/k} = \alpha.$$

Proof. Assume the region with the most nodes has $N_{max} = n/k + t$ nodes. From Occupancy lemma, we know $\lim_{n\to\infty} \frac{t}{n/k} = 0$. Therefore, from the generalized BHH theorem, we know:

$$L \leq \alpha(1+o(1))\sqrt{N_{max}}\sqrt{\frac{1}{k}} \leq \alpha(1+o(1))\sqrt{(1+o(1))\frac{n}{k}}\sqrt{\frac{1}{k}} \leq \alpha(1+o(1))\frac{\sqrt{n}}{k}.$$

On the other hand, since the region with the most nodes has at least n/k nodes, the claim is true:

$$\lim_{n\to\infty} \frac{L}{\sqrt{n}/k} = \alpha.$$

□

The lemma implies that the length of the longest tour generated by the region partition heuristic is asymptotically close to a lower bound when the size of the problem expands. Thus this heuristic is asymptotically optimal in this case.

If vehicles are more than depots available ($k > m$), we can use the perturbing idea to generate their routes. In this procedure, if there are two or more vehicles on the same depot, we only need to slightly relocate vehicles evenly distributed on a small circle encircling that depot (Figure 1). For a sufficiently small perturbation, the algorithm remains feasible, and the optimal value only changes marginally.

If the ratio of n to k is smaller, for example, $n = O(k \log k)$, it is harder to predict the optimal solution. Motivated by the Occupancy lemma, we still can derive an asymptotic upper bound. The following theorem illustrates an extreme case:

Theorem 9. *Assume $k = m$, i.e., all vehicles are i.i.d. in the square. If the number of vehicles is proportional to the number of nodes, i.e., $k = \lambda n$, $0 < \lambda < 1$, then with probability one, as $k \to \infty$,*

$$L \leq (\frac{\alpha}{\lambda} + 2\sqrt{2}\log n)\frac{1}{\sqrt{n}}.$$

Proof. We divide the unit square into $n/\log^2 n$ smaller squares each of which has the side length $\log n/\sqrt{n}$. Then, from the Occupancy lemma, we know that any small square has at most $\log^2 n + O(\log n)$ nodes and at least $\lambda(\log^2 n - O(\log n))$ vehicles with probability one as $n \to \infty$. Apply the tour partition heuristic to the square with the most customers. According to the generalized BHH theorem and Lemma 2, we have

$$L \leq \frac{\alpha\sqrt{\log^2 n + O(\log n)}}{\lambda(\log^2 n + O(\log n))}\frac{\log n}{\sqrt{n}} + 2\sqrt{2}\frac{\log n}{\sqrt{n}} \approx (\frac{\alpha}{\lambda} + 2\sqrt{2}\log n)\frac{1}{\sqrt{n}}.$$

□

Applying the similar argument, we obtain an upper bound for the case $k = \Omega(\sqrt{n})$:

Corollary 10. *Assume all vehicles are uniformly distributed in the cube and $k = \Omega(\sqrt{n})$, then with high probability, as $k \to \infty$,*

$$L \leq \frac{\alpha\sqrt{n}}{k} + 2\sqrt{2}\frac{\log k}{\sqrt{k}}.$$

3 An LP-based Load Balancing Heuristic

We present a linear programming-based heuristic in this section, which uses a *global* load balancing technique during post-refinement periods.

Although many heuristics have been applied to solve the SDVRP and MDVRP, we are unaware of any existing specific heuristic or theoretical results for min-max MDVRP, especially in the large-size case. In general literature, one idea to solve the MDVRP is a two-stage heuristic(for example: Wren and Holliday [17]): construct an initial solution, then apply a number of local improvements. The initial solutions in common heuristics are usually constructed by the nearest neighbor assignment. Gillet and Johnson [9] proposed a clustering heuristic which used a sweep heuristic at each depot. Golden et al. [11] presented two heuristics for MDVRP; the second one specifically solved the large-size problem by a two-stage algorithm which assigns nodes to depots first and then built TSP tours for each vehicle. Chao et al. [5] proposed a simple initialization heuristic followed by a refinement, which gained the best performance in many benchmark problems. In 2002, Giosa et al. [10] demonstrated a series of 2-stage heuristics to solve MDVRPTW. Recently, Lim and Wang [14] gave a one-stage approach to MDVRP with the constraint each depot only has a fixed number of vehicles. Baltz et al. [2, 3] presented a probabilistic analysis of the optimal solution for the problem. Tansini [16] proposed a polynomial-time approximation scheme (PTAS) for the MDVRP similar to Arora's PTAS algorithm for TSP.

Our heuristic is based on a *load balancing* idea: the load for each vehicle must be almost balanced in an optimal plan, so each vehicle is assigned the same working load in terms of the number of nodes to serve. This intuition is plausible considering that the distribution of nodes in our model is uniform. We carefully balance the number of nodes assigned to each depot by linear programming at the first step, then generate a near-optimal route for each vehicle by using the Concorde TSP solver [6]. The solution is further improved by global

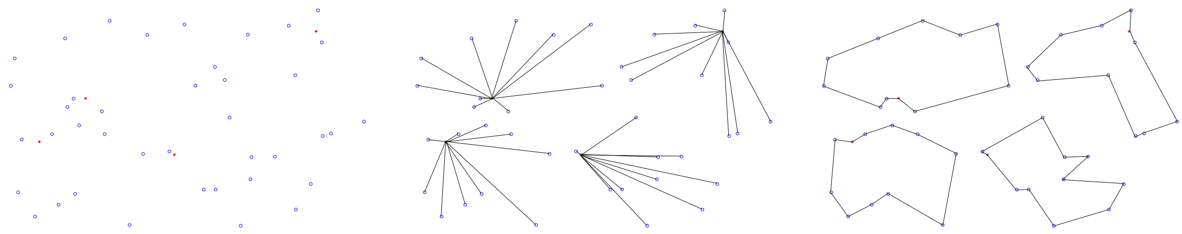

Figure 2: The initial input (a), the assignment (b), and the final construction of TSP tours (c).

adjustment of the number of nodes assigned to vehicles, rather than traditional local search procedures.

Assume all customers are uniformly distributed on a square and the locations of depots are randomly distributed. A depot is allowed to have more than one vehicle. Each vehicle must return to the depot from which it originates. First we assume that each depot has exactly one vehicle, and no two depots share a location (this assumption will be generalized later). Suppose each node $X_j, 1 \le j \le n$, is located at (x_j, y_j), and each vehicle $V_i, 1 \le i \le k$, is located at (v_i, w_i). The distance c_{ij} between X_j and V_i is $||(x_j, y_j) - (v_i, w_i)||$.

Define x_{ij} to be a binary variable indicating if node X_j is assigned to vehicle V_i or not. We build an integer program for clustering:

$$
\begin{aligned}
(LP) : \text{minimize} \quad & \sum_{1 \le i \le k, 1 \le j \le n} c_{ij} x_{ij} \\
\text{subject to} \quad & \sum_{1 \le j \le n} x_{ij} = \frac{n}{k}, \ \forall i, \\
& \sum_{1 \le i \le k} x_{ij} = 1, \ \forall j, \\
& x_{ij} \in \{0, 1\}, \ \forall i, j,
\end{aligned}
$$

where we assume that n/k is an integer(the fractional case will be discussed later).

Note that its linear program relaxation (LPR) is a typical network flow model, so any vertex solution to LPR will be integral. Therefore, we can directly solve LPR for the initial assignment.

A sketch of the heuristic is as follows:

Heuristic 3 LP-based Load Balancing Heuristic

1. Build LPR and solve it.

2. Build a TSP tour for each vehicle by the Concorde solver. Assume that their lengths are sorted in a descending order $\{L_1, \cdots, L_k\}$.

3. If $(L_1 - L_k)/L_k \le r$, with r a small constant, or if the linear program has been run many times, stop and output a current best solution. Otherwise, decrease the number of nodes assigned to vehicles having longer tours and increase the number of nodes assigned to vehicles having shorter tours (we will explain details later). This way, a new revised LP is generated. Return to step 1 and repeat the procedure.

A sketch of the heuristic in one loop is shown in figure 2.

There are several details we need to explain in our algorithm.

Note 1: If n/k is fractional. Assume $n = pk + q$. p and q are integers and q is the residue. Then we assign the first q vehicles $p + 1$ nodes and assign each of the remaining vehicles p nodes. Thus the number of nodes assigned to each vehicle is always an integer.

Note 2: Constant r in step 3 is a ratio threshold to measure the output, which can be adjusted according to the need for accuracy. In most cases, setting $r = 25\%$ is enough to achieve a satisfactory result.

Note 3: At step 3, if $(L_1 - L_K)/L_K \le r$, we adjust the number of nodes assigned to each depot by the following procedure:

- Build the sets $L^+ = \{i : L_i - \bar{L} > 0\}$, $L^- = \{i : L_i - \bar{L} < 0\}$, where $\bar{L}$ is the average of tour length.

- Without loss of generality, assume $s = |L^+| \ge |L^-|$. For any $i \in L^+$, define the *relative drift* $d_i^+ = \frac{L_i - \bar{L}}{\bar{L}}$. Decrease the number of nodes assigned to depot i by $[c * \frac{d_i}{r}]$(c is a constant empirically decided. We set $c = 2$ in implementation). Sort elements in L^- in the descending order of their tour lengths. Then define $d_i^- = \frac{\bar{L} - L_i}{\bar{L}}$ for the first s elements in L^-. Increase the number of nodes in the same way.

- If the number of nodes removed from L^+ is more than the number of nodes added to L^-, add one node to each element in L^- sequentially starting from the depot which has the fewest nodes, keep doing that until nodes are balanced. Repeat the similar procedure for the contrary case.

3.1 Multiple Vehicles on a Depot

We have assumed that each depot includes exactly one vehicle in all the heuristics aforementioned. However, in a practical situation, multiple vehicles may lie on the same depot. We need to extend our heuristics to this general case. If we still follow the original procedures and assign the same number of nodes to each vehicle instead of each depot, the solution is poor, because nodes are assigned to each such vehicle randomly – therefore, our initial geometric intuition for node assignment is lost (Figure 3).

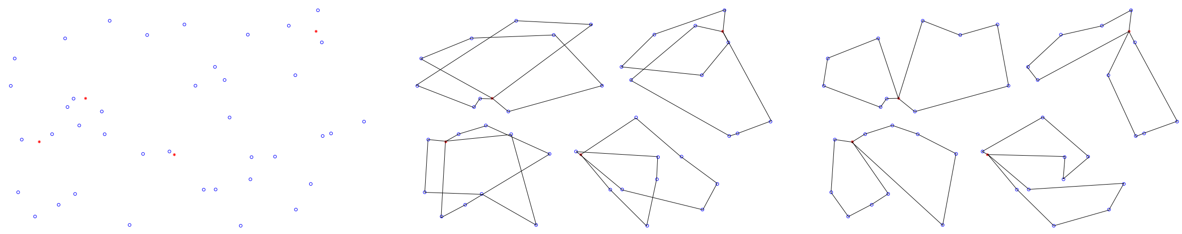

Figure 3: Problems encountered when multiple vehicles are assigned to a single depot. (a) is the initial node set given to the problem (we have 8 vehicles and 4 depots) and (b) represents a solution without the perturbation applied. (c) represents a solution with the perturbation applied; it is clearly preferable, as vehicle routes do not intersect.

In order to work around this difficulty for vehicles in a same depot, we perturb the locations of vehicles located at the same depot before generating c_{ij}'s. Each vehicle is relocated to a new location lying in a small circle around that depot, keeping them evenly distributed on that circle. This small change will dramatically improve the final solution (Figure 1). We can generate good solutions with this technique.

4 A Region Partition Heuristic

The second heuristic we present is the region partition heuristic (Heuristic 2) based on the algorithm in [1] that takes as input a convex planar region C and a set of m depots $\mathcal{D} = \{D_1, \ldots, D_m\}$, and outputs a partition of C into m convex subregions satisfying the following properties:

- Each subregion is convex.
- Each subregion contains one depot D_i.
- Each subregion has equal area.

Note that a Voronoi diagram satisfies the first two properties, but not the third. An example of such a partition is shown in figure 4.

As mentioned previously, the requirement that each subregion have equal area ensures that the lengths of the TSP tours of each region be equal asymptotically [18]. This result applies to any compact subset of $\mathbb{R}^2$ with Lebesgue measure one, but we require that these regions be convex for practical purposes. The shortest path between any two cities in a subregion R_i is a straight line, and we want to ensure that the route taken from these one city to the next lie in the same service region and therefore be feasible.

Many network structures, like minimum spanning trees and Voronoi graphs, have conclusions similar to that of the travelling salesman tour, i.e., their lengths converge asymptotically in a manner similar to the BHH theorem. For example, [19] describes a similar result for Minimum Spanning Tree (MST). Region partition is therefore a potentially powerful tool not

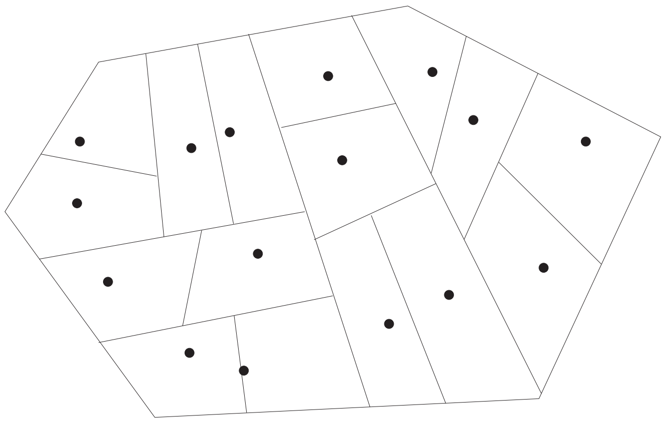

Figure 4: An equitable partition of a convex polygon.

only in vehicle routing but also in many network applications, such as finding the minimum spanning forest of a node set with fixed roots. It is worth extra effort to investigate how to implement partition quickly. Bespamyatnikh et al. [4] gave the proof for the existence of a convex region partition and an algorithm with runtime $O(N^{\frac{4}{3}} \log N)$ for the discrete version in which the dividing subjects are red and blue points. And N is the sum of the number of nodes and sides of the initial polygon. Given that an area can be approximated within any ϵ factor by taking enough sampling points, we can implement their algorithm to obtain a near-optimal convex subdivision. We have also been searching for a quick method to realize the algorithm. Our best result is the realization of an equitable partition of a polygon and a set of n points in $O(Nn \log N)$ time.

5 Performance and Analysis

No specific benchmark problems for min-max MDVRP are available in the literature. Most available MDVRP benchmarks have time window constraints, and their sizes are generally small, varying between $50 \sim 1000$(see [15]). Thus, we generated 13 MDVRP benchmark sets ourselves and the results are reported in this section. These sets vary in size with respect to the number of depots (2 to 20) and number of nodes (100 to 2048), and the depot and node locations. Each set contains 4 instances, each with the same number of depots and nodes, but randomly generated depot and node locations. The data presented represent the average values obtained over the 4 instances for each benchmark set. Tests were conducted on a Pentium 4 1.8GHz/512MB notebook. The implementation was done in Matlab with the COPL LP Solver [20] and the Concorde TSP solver [6].

In Table 1 and 2, we compare the initial solutions provided by three heuristics: the LP-based heuristic, the region partition heuristic and a traditional nearest-neighbor approach which can be stated as follows:

Heuristic 4 Nearest Neighbor Heuristic

1. Assign each node to its nearest depot.

2. Build a TSP route for each vehicle accordingly; get an initial solution for the problem.

3. Run the local improvement method to improve the solution.

We have also implemented a modified version of a local improvement procedure proposed by Chao et al. [5] called "1-point movement". In Table 3 and 4, we present the final outputs of four heuristics. The first heuristic uses the LP-based load balancing heuristic (Heuristic 3) to distribute the load evenly among vehicles. The other three heuristics all use the modified 1-point movement improvement method in their post optimization periods although they start with different initial assignments. The second heuristic uses the same initial assignment as the first heuristic. The third heuristic starts with the region partition method, and the last one starts with the nearest neighbor assignment.

We have applied different heuristics on each instance and presented in the tables the value of the longest tour("Max"), the value of the shortest tour("Min"), the average value of all tours("Mean") and the CPU time("Time"). The primary goal of these heuristics is to minimize the value of the longest tour. The running time of the LP-based load balancing heuristic is primarily comprised of the execution time of the LP and TSP solvers. The running time of the region partition with local improvement method (Heuristic 2) mainly consists of the running time of the region partition heuristic and TSP solvers. For small to medium-sized benchmarks, the main bottleneck for all heuristics is finding a near optimal TSP tour for each depot using the Concorde solver.

We have made the following observations based on the simulation results presented in the tables.

First, according to Table 1 and 2, both LP-based and region partition heuristics produce notably better initial routes than the traditional nearest neighbor approach. They are efficient and effective candidates for the initial assignment.

Second, the region partition heuristic with local improvement produced the best solutions("Max") for 7 out of 13 instances, and the LP-based load balancing heuristic generated the best solutions for 5 out of 13 instances. They both maintained a competitive running time, as seen in Table 3 and 4.

Third, when the number of depots is small, the region partition heuristic tends to outperform the other three heuristics, and its running time remains low. The LP-based load balancing heuristic performs better as the number of depots is increased.

Fourth, the region partition heuristic with local improvement and the LP-based load balancing heuristic not only improve the maximal tour length as compared to the initial solution, but they also do not significantly increase the sum of all tour lengths, suggesting that the adjustment will not lead to a major global increase in cost.

Finally the region partition heuristic with local improvement generates a higher total cost than the LP-based load balancing heuristic in most instances, though it obtains the best results for min-max MDVRP in more instances than the latter. This fact demonstrates that the difference of objective functions between usual MDVRP and min-max MDVRP

Table 1: Summary of First Iteration Solutions for Three Heuristics

			LP-TSP				Region Partition			
Dataset	Depots	Nodes	Max	Min	Mean	Time	Max	Min	Mean	Time
d2n100	2	100	472.61	391.12	431.87	1.07	429.22	399.90	414.56	1.10
d2n500	2	500	870.76	847.10	858.93	41.70	875.66	843.35	861.00	28.34
d5n100	5	100	242.12	159.50	197.15	1.13	209.24	162.26	190.24	2.20
d5n500	5	500	378.95	332.04	356.64	6.43	377.97	334.34	356.03	9.06
d5n1000	5	1000	542.37	467.60	503.22	35.65	522.34	473.58	497.20	36.73
d10n1000	10	1000	283.32	225.20	252.86	11.03	277.63	233.97	255.43	41.11
d10n2000	10	2000	380.10	331.38	355.08	115.34	378.11	317.47	353.58	180.76
d16n256	16	256	161.22	75.21	106.87	12.16	140.74	70.26	105.51	21.47
d16n512	16	512	182.11	106.46	132.74	10.95	172.55	106.58	131.59	22.32
d16n1024	16	1024	206.95	149.55	173.64	17.31	196.16	145.83	169.19	57.66
d16n2048	16	2048	263.61	207.90	231.87	91.03	253.85	206.56	229.62	230.51
d20n1000	20	1000	172.29	117.07	139.24	24.95	170.34	118.31	138.16	84.88
d20n2000	20	2000	207.23	167.49	184.83	83.02	222.11	160.57	185.62	177.91

Table 2: Continued: Summary of First Iteration Solutions for Three Heuristics

			Nearest Neighbor			
Dataset	Depots	Nodes	Max	Min	Mean	Time
d2n100	2	100	574.85	260.88	417.85	0.35
d2n500	2	500	1097.33	715.18	906.25	48.33
d5n100	5	100	268.58	95.23	186.30	0.20
d5n500	5	500	519.58	223.48	369.35	5.00
d5n1000	5	1000	1101.95	163.28	511.98	295.65
d10n1000	10	1000	451.18	83.50	257.03	11.68
d10n2000	10	2000	688.58	178.15	361.55	351.40
d16n256	16	256	186.23	23.03	91.40	1.38
d16n512	16	512	265.48	32.13	122.93	2.33
d16n1024	16	1024	419.10	50.58	167.35	12.55
d16n2048	16	2048	496.25	72.25	230.63	243.13
d20n1000	20	1000	278.38	36.50	134.63	8.75
d20n2000	20	2000	368.88	57.25	183.35	49.10

Table 3: Summary for Improved Heuristics

			LP-TSP + Load Balancing				LP-TSP + Local Improvement			
Dataset	Depots	Nodes	Max	Min	Mean	Time	Max	Min	Mean	Time
d2n100	2	100	458.35	403.23	430.80	9.18	448.93	443.28	446.10	6.65
d2n500	2	500	868.60	846.50	859.53	192.35	865.00	863.63	864.33	187.33
d5n100	5	100	207.08	172.08	190.58	8.85	229.45	174.68	215.50	7.83
d5n500	5	500	377.28	332.20	357.08	57.90	374.65	357.33	368.03	60.85
d5n1000	5	1000	531.85	467.73	501.05	192.58	534.73	488.08	520.13	202.33
d10n1000	10	1000	277.83	229.28	252.93	84.88	279.90	224.80	262.23	86.10
d10n2000	10	2000	378.65	331.60	355.43	875.45	379.70	335.68	365.88	1098.43
d16n256	16	256	114.63	88.08	101.98	68.60	142.63	75.55	112.75	61.23
d16n512	16	512	142.15	118.70	131.55	78.75	171.95	108.08	144.38	73.60
d16n1024	16	1024	187.98	155.00	171.98	153.88	201.05	150.58	178.10	134.05
d16n2048	16	2048	254.75	209.05	231.55	870.18	261.83	207.18	235.10	860.70
d20n1000	20	1000	151.05	124.20	138.20	192.15	166.33	117.25	145.65	166.93
d20n2000	20	2000	201.43	168.25	184.80	601.60	205.03	168.75	189.13	612.70

Table 4: Continued: Summary for Improved Heuristics

			Region Partition+Local Imp				Nearest Neighbor+Local Imp			
Dataset	Depots	Nodes	Max	Min	Mean	Time	Max	Min	Mean	Time
d2n100	2	100	416.70	414.03	415.38	5.73	528.75	313.88	421.30	6.60
d2n500	2	500	873.15	869.18	871.15	152.38	913.35	851.85	882.60	135.03
d5n100	5	100	204.80	187.00	199.40	8.98	235.20	141.35	204.73	7.58
d5n500	5	500	371.88	339.00	359.00	62.18	447.48	261.73	381.85	72.90
d5n1000	5	1000	518.33	499.83	511.43	247.50	858.75	253.08	548.78	321.43
d10n1000	10	1000	270.50	238.30	260.30	165.03	391.08	124.85	282.80	150.03
d10n2000	10	2000	375.50	335.80	363.25	923.58	574.48	173.93	362.43	1206.25
d16n256	16	256	132.33	80.08	111.85	74.18	156.93	23.68	102.60	58.80
d16n512	16	512	168.15	106.93	143.68	75.03	204.70	35.38	136.63	64.08
d16n1024	16	1024	191.68	151.65	176.75	182.20	321.70	57.45	184.38	188.48
d16n2048	16	2048	250.95	207.40	233.63	1062.70	414.53	90.43	238.30	1011.68
d20n1000	20	1000	163.70	118.13	143.43	229.15	244.88	41.20	144.58	242.35
d20n2000	20	2000	219.38	162.65	191.60	543.58	335.48	59.93	209.18	617.83

motivates different types of algorithms.

6 Conclusion

This paper has presented the first study to min-max MDVRP. We explored some theoretical asymptotic properties of the optimal solution, which gave rise to the region partition heuristic. The region partition idea may have applications beyond the vehicle routing problem. An LP-based heuristic with global improvement was also presented to solve very large-sized problems effectively. We concluded by reporting our simulation results.

Many interesting topics remain after this first attempt in min-max MDVRP. For example, from a theoretical point of view, accurate estimation of the optimal solution in large-scale problems for the general case is still open. Our theoretical and implemented work can serve as starting points for further exploration as well.

The techniques used in our algorithms provide advantages over traditional local search methods. For example, since the LP-based load balancing heuristic does not inherently require that the points be scattered in a Euclidean space, it can be adapted as a backend to existing map software to solve VRPs with actual geographic locations. In addition, the region partition algorithm can be applied to problems outside the scope of VRP. An exact convex region partition algorithm will be helpful not only for high-quality initial solutions to MDVRPs, but also for solving a class of network problems with similar structures and a min-max objective function.

References

[1] B. Armbruster, J. Carlsson. Finding equitable convex partitions of points in a polygon efficiently. Submitted for publication, 2006.

[2] A. Baltz, D. P. Dubhashi, A. Srivastav, L. Tansini, S. Werth. Probabilistic Analysis for a Multiple Depot Vehicle Routing Problem. *Foundations of Software Technology and Theoretical Computer Science, 25th International Conference(FSTTCS)*, pp 360-371, 2005

[3] A. Baltz, D. P. Dubhashi, A. Srivastav, L. Tansini, S. Werth. Probabilistic Analysis for a Multiple Depot Vehicle Routing Problem. *Random Structure and Algorithms*, pp 206-225, 2007.

[4] S. N. Bespamyatnikh, D. G. Kirkpatrick, J. Snoeyink. Generalizing Ham Sandwich Cuts to Equitable Subdivisions. *Symposium on Computational Geometry (SoCG)*, pp 49-58, 1999.

[5] I.-M. Chao, B. Golden, E. Wasil. A new heuristic for the multidepot vehicle routing problem that improves upon best-known solutions. *American Journal mathematical management Science*, Vol. 13, No 3, pp 371-406, 1993.

[6] W. Cook, CONCORDE solver for the TSP, Version 1.1. Computer software. `http://www.tsp.gatech.edu/concorde.html`

[7] G.B. Dantzig, J.H. Ramser. The truck dispatching problem. *Management Science*, Vol. 6:60, 1959.

[8] J.F. Soumis, M. Desrochers A New Optimization Algorithm for the Vehicle Routing Problem With Time Windows by Column generation. *Networks*, Vol. 14, pp 545-565,1984

[9] B. Gillet, J. Johnson Multi-terminal vehicle-dispatching algorithm. *Omega*, Vol. 4, pp 711-718, 1976.

[10] I. Giosa, I. Tansini, I. Viera. New assignment algorithm for the multi-depot vehicle routing problem. *Journal of Operations Research Society*, Vol. 53, pp 283-292, 1984.

[11] B.L. Golden, EL. Magnanti, H.Q. Nguyen. Implementing vehicle routing algorithms. *Networks*, Vol. 7, pp 113-148, 1977.

[12] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association.* 58 (301): 13-30, Mar. 1963.

[13] T. Leighton, P.W. Shor. Tight Bounds for minimax grid matching with applications to the average case analysis of algorithms. *Combinatorica,* Vol. 9, pp 161-187, 1989

[14] A. Lim, F. Wang Multi-Depot Routing Problems: A one-Stage Approach. *IEEE transactions on automation science and engineering*, Vol. 2, No. 4, Oct. 2005.

[15] http://www.top.sintef.no/vrp/benchmarks.html

[16] L. Tansini. Department of Computer Science, Chalmers Univeristy, *PhD Thesis.* To appear.

[17] A. Wren, A. Holliday. Computer scheduling of vehicles from one or more depots to a number of delivery points. *Operations Research Quarterly,* Vol. 23, pp 333-344, 1972.

[18] J. E. Yukich. Probability Theory of Classical Euclidean Optimization Problems. *Lecture Notes in Mathematics*, Vol. 1675, 1998.

[19] J. M. Steele. Growth rates of Euclidean minimal spanning trees with power weighted edges. *Annals of Probability*, Vol. 16, pp 1767-1787, 1988.

[20] X. Zhang, Y. Ye. COPL Computational Programming Optimization Library : Linear Programming. Computer software. `http://www.stanford.edu/ yyye/Col.html`