アルゴリズム 72－5
(2000. 3. 21)

配送計画問題に対する近似アルゴリズムの実際的性能評価

山下慶子　　浅野孝夫
中央大学大学院理工学研究科情報工学専攻

あらまし　配送計画問題は近年活発に研究されている問題で, 経済活動のさまざまな状況でおこる. この問題には幾つかの種類があり, 容量 k の巡回路被覆問題, 容量 k の配送巡回路問題, 電話予約送迎問題などがその代表例である. 本稿では最も一般的な枠組での配送計画問題である容量 k の配送巡回路問題の代表的なアルゴリズムを実装し, その実際的性能を評価する. さらに, 複数デポーの配送計画問題に対する近似アルゴリズムを提案する. このとき従来の目的である配送に要する総距離最小という指標から少し離れ, 現実的に乗組員の仕事量均等化を考慮に入れることも試みる. そして配達時間帯指定の配送計画問題にも取り組み, ベンチマークデータを用いて計算機実験を行い, 提案するアルゴリズムの実際的性能評価を行う.

Practical Evaluation of Approximation Algorithms for the Vehicle Routing Problem

YAMASHITA Yoshiko　　ASANO Takao
Department of Information and System Engineering, Chuo University

Abstract　The vehicle routing problem has received much attention in recent years and occurs in many practical settings. There are several variants of this problem: the k-tour cover problem, the k-delivery TSP and the dial-a-ride problem or stacker crane problem. In this paper, we try to evaluate the practical performances of representative algorithms for the k-delivery TSP. Furthermore, we propose approximation algorithms for the multi-depot vehicle routing problem and the vehicle routing problem with time windows and evaluate their practical performances through the computational experiments based on bench-mark data.

1　はじめに

配送計画問題とは, 品物の集合とそれらの供給点と需要点が与えられたときに供給点から需要点へ要求される品物を配送する問題である. もちろん, 品物は道路網等を抽象化したネットワークで車両等を利用して配送されるが, 車両の積載量 (容量) に制限がある. そこで容量付きの車両でネットワークの供給点から需要点へ要求を満たすように品物を配送する際, 配送に要する総距離を最小化する問題である. この問題には以下のようにいくつかの種類がある.

<u>容量 k の巡回路被覆 (k-tour cover) 問題</u>

入力: 原点と呼ばれる 1 つの供給点に n 個の品物, 任意の地点に配置された n 個の需要点, そして容量 k の車両.

問題: 総距離最小の容量 k の巡回路被覆を見つける. ただし, 容量 k の巡回路とは原点と高々 k 個の需要点を含む巡回路である.

<u>複数デポー (multi-depot) の配送計画問題</u>

入力: 任意の地点に配置された $m(< n)$ 個の供給点と n 個の需要点, そして容量 k の車両.

問題: 車両の容量を越えることなく, 供給点にあるすべての品物を需要点に配送する最短巡回路を見つける.

<u>容量 k の配送巡回路問題 (k-delivery TSP)</u>

入力: 任意の地点に配置された n 個の供給点と n 個の需要点, そして容量 k の車両.

問題: 車両の容量を越えることなく, 供給点にあるすべての品物を需要点に配送する最短巡回路を見つける.

<u>電話予約送迎 (dial-a-ride) 問題</u>

入力: 任意の地点に配置された n 個のラベル付けされた供給点と n 個のラベル付けされた需要点, そして容量 k の車両.

問題: 車両の容量を越えることなく, 供給点に

あるすべてのラベル付けされた品物を対応するラベル付けされた需要点に配送する最短巡回路を見つける.

配達時間帯指定の配送計画問題

入力: 原点と呼ばれる1つの供給点に n 個の品物, 任意の地点に配置された n 個の需要点と需要点への品物の配達時間帯, そして容量 k の車両.

問題: 指定の配達時間を満足し, かつ車両の容量を越えることなく, 供給点にあるすべての品物を需要点に配送する最短巡回路を見つける.

上記の問題に対する研究背景を2章で述べる. 3章では容量 k の配送巡回路問題に注目する. 4章では複数デポーの配送計画問題に注目し, 従来の配送に要する総距離最小という目的に対し, 現実的に要求される乗組員の仕事量均等化や荷量のバラツキ最適化という指標の導入を試みる. 5章では配達時間帯指定の配送計画問題について述べる.

2 配送計画の背景と従来の研究

容量 k の巡回路被覆問題に対して, $k=2$ に限定した問題は最小重み完全マッチングを求める問題に帰着でき簡単に解くことができる. しかし, $k \geq 3$ ではNP困難となる (集合のサイズが k 以下に限定された集合族の重みつき集合被覆問題になる). 1985年に3近似アルゴリズムが提案され [8], 1997年にはユークリッド距離における2次元の容量 k の巡回路被覆問題に対して $k \leq c \log n / \log\log n$ という条件で $(1+\epsilon)$ 近似アルゴリズムが示されている [3].

容量 k の配送巡回路問題については, 1992年に $k=1$ に対して2.5近似アルゴリズムが提案された [2]. 1996年には $k=1,\infty$ の場合に近似率2が達成され, さらに任意の k に対して近似率9.5のアルゴリズムが提案された [5]. またこのアルゴリズムを改良し1997年には近似率 $7-\frac{3}{k}$ が達成され [1], 1998年には近似率5が達成されている [6].

電話予約送迎問題に対して木状のネットワークにおける高性能近似アルゴリズムが1998年に Charikar-Raghavachari によって発表されている [7]. 高さ平衡木における中継点を許さない問題においては $O(\sqrt{k})$ 近似アルゴリズム, 中継点を許す問題においては $O(1)$ 近似アルゴリズムになっている. この木状のネットワークという条件は近似率は悪くなるが取り除くことができる [4].

配達時間帯指定の配送計画問題に対してこれまで発見的手法に基づく手法, 整数計画問題として定式化して線形計画問題に緩和して解く方法, 分枝限定法を用いる方法, またラグランジュ緩和に基づく方法などが研究されてきている. しかし現在, 解の精度を保証する高性能な近似アルゴリズムはまだ知られていなく, これからの研究対象になっている.

3 容量 k の配送巡回路問題に対するアルゴリズムの実際的評価

容量 k の配送巡回路問題に対していくつかアルゴリズムが提案されているが, ここでは以下の代表的な2つのアルゴリズムの実際的性能評価を行う。

3.1 CMR アルゴリズム

1996年に Chalasani-Motwani-Rao によって提案された9.5近似アルゴリズム (以下, CMR アルゴリズムと呼ぶ) [5] は, 図1のようにすべての供給点を含む巡回路とすべての需要点を含む巡回路を TSP の近似アルゴリズムを用いて求める. そして得られた供給点の巡回路から探索し始め, k 個の品物を集める. 次に需要点の巡回路に移り, k 個の品物を配達する. また供給点の巡回路に移り, 品物がなくなるまで繰り返すというものである (図1). しかしこの方法は供給点と需要点の近接関係を利用していなくて, 良い解は期待できないといわれている.

3.2 CKR アルゴリズム

1998年に Charikar-Khuller-Raghavachari によって提案された5近似アルゴリズム (以下, CKR アルゴリズムと呼ぶ) [6] では, 図2のようにすべての供給点と需要点を含む1本の巡回路を TSP の近似アルゴリズムで求める. そして巡回路上のある地点から出発し品物を拾う

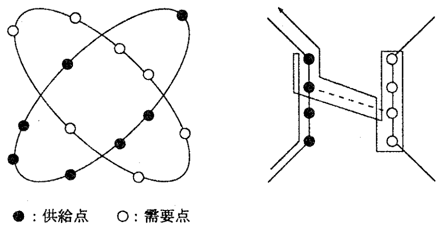


図 1: CMR アルゴリズム

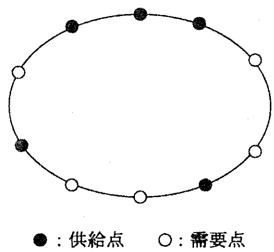


図 2: CKR アルゴリズムの基本的な考え方

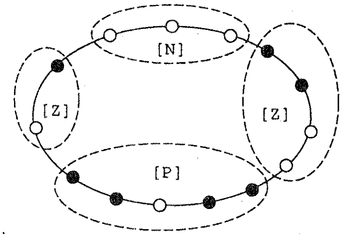


図 3: k=6 の場合の 3 つの部分 ([Z],[P],[N])

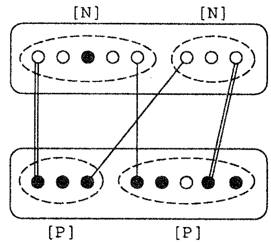


図 4: [P] と [N] のマッチング

ことと需要点に配ることをオンラインで行なう. すなわち, 供給点を訪れたならば品物を拾い, 需要点を訪れたならば品物を配達する. また車両が $\frac{k}{2}$ 個の品物を積んでいるとき満タンと考え, 残りの $\frac{k}{2}$ の領域をバッファーとして考える. アルゴリズムの最初の手順として, 1 本の巡回路を出発してから $\frac{k}{2}$ 個の品物の増減に応じて次の 3 つの部分 ([Z],[P],[N]) に分解する (図 3).

- [Z] : 供給点と需要点の数が等しい部分.
- [P] : 供給点の数が需要点の数より $\frac{k}{2}$ 個多い部分.
- [N] : 需要点の数が供給点の数より $\frac{k}{2}$ 個多い部分.

[Z] の部分の品物はその [Z] 内の需要点に配る. [P] 内では品物が $\frac{k}{2}$ 個超過してしまう. この超過分を [N] に配る. このため [P] と [N] の最小重み完全マッチングを計算し, マッチングされた辺が [P] と [N] を結ぶ配送ルート (の一部) になる (図 4).

配送ルートを考えるのに始め車両は $\frac{k}{2}$ 個の品物を積んでいるということを仮定する. この仮定は後に取り除くことができる (文献 [6] 参照). この条件で時計回り, 反時計回りのどちらかの方向に探索を始め, 上記の 3 部分によってそれぞれ以下のようにふるまう.

- [Z] に出会ったとき
  品物を拾い, 配達しながら探索する. $\frac{k}{2}$ 個の品物を積んで出発しているので, 1 回の探索で行なうことができる.
- [P] に出会ったとき
  品物を拾い, 配達しながら探索し, その [P] とマッチされている [N] とを結ぶ地点が現れたらそのマッチング辺に沿って [N] へ移動する. [N] の始めの地点まで何もしないで行き, 巡回路にそって品物を拾い配達し, またマッチング辺に沿って [P] へ戻る. 再び品物を拾い配達することを続ける.
- [N] に出会ったとき
  品物を拾うことも配達することもせずに

通りすぎる.

これで配送ルートは決定される (図 5). さらには時計回り, 反時計回りの配送ルートのうち短い方を選択する.

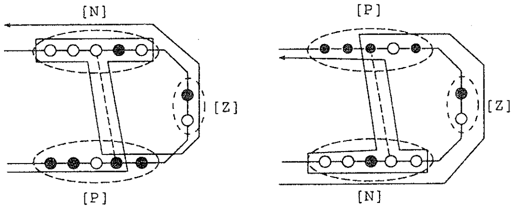


図 5: CKR アルゴリズム

3.3 計算機実験

CMR アルゴリズムと CKR アルゴリズムをプログラム化し, 実験を行なった. この際, 巡回路を MST(最小重みスパンニング木) から作るときに使用した最小重みマッチングのプログラムについては URL *ftp://ftp.zib.de/pub/Packages/mathprog/matching/weighted/* のものを使用させていただいた. なお使用言語は C 言語で, 入力データは点の座標を一様乱数を用いて発生させたものである.

図 6 には CMR アルゴリズムの解を, 図 7 には CKR のアルゴリズムの解をそれぞれ可視化して示している (実線は車両の配送経路, 破線は車両の配送する順序をそれぞれ表している). それぞれ供給点を 20 個, 車両の容量を 4 個としている. 発生させた乱数は $[0, 400]$ の整数である.

図 6 から CMR アルゴリズムは, 供給点と需要点を別々の 巡回路として考え, 車両がいっぱいもしくは空になったときにのみ, 巡回路間の移動を行なうため, 同じ所を何度も探索している部分が多く生じている. つまり非常に多くの無駄が生じているのがわかる.

それに対して, 図 7 から CKR アルゴリズムは, 供給点と需要点を 1 つの 巡回路として考え, すべての供給点と需要点の近接関係をうまく利用していることで, 多くの部分を 1 度の探索を行なうことで済んでいるということがわかる. ただここで p1, p2, s1, s2 の部分に注目すると, p1 と p2, s1 と s2 はそれぞれ 1 つの [P][N] 部分であり, p1 p2 の [P] 部分 は, s1 s2 の [N] 部分 とマッチされている. また p1 p2 の [P] 部分 と s1 s2 の [N] 部分 は 巡回路上で連続している. この部分をアルゴリズムでは次のように探索している. まず, s1, s2 に品物を配達し, p1, p2 を拾う. そして p1, s2, p1, p2 と何もせずに通りすぎている. この p1, s2, p1, p2 と何もせずに通りすぎている部分が無駄になっている. また, p3, p4, s3, s4 についても同様の無駄が生じている. このように 巡回路上で連続した 部分 がマッチされているときに無駄が生じる.

この無駄を除去するため, 次のように局所改善を行う. 巡回路上で [P] と [N] が連続しているときに, その 2 つの部分 を 1 組として扱い, マッチングをとられたときの [P], [N] の探索法ではなく, [Z] と同様に 1 度だけ探索を行なうようにする. この局所改善を組み込んだアルゴリズムの実行結果を図 8 に示している. これから無駄になっていた何もせずに通りすぎている部分が改善されていることがわかる.

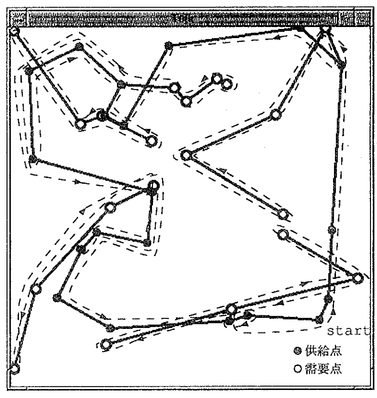


図 6: CMR アルゴリズム. 巡回路の総長 5134.

図 9 は供給点を 100 個与え, 車両の容量を変化させたときに上の 3 つのアルゴリズムの値をグラフにしたものである. 発生させた乱数は $[0, 2000]$ の整数である: 図 9 から CMR アルゴリズムは車両の容量が 2 のときに最も良

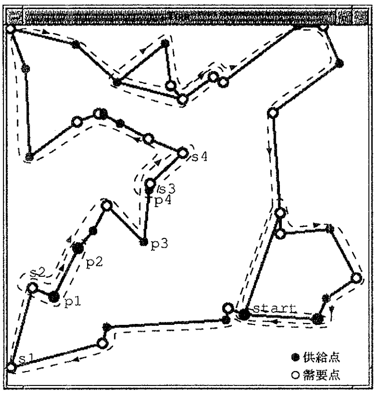


図 7: CKR アルゴリズム. 巡回路の総長 2949.

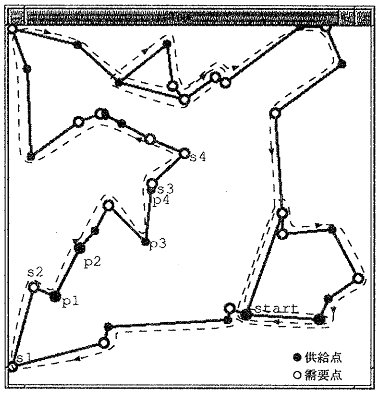


図 8: CKR アルゴリズムを局所改善したもの. 巡回路の総長 2663.

い値を出すということがわかる.

CMR アルゴリズムでは車両の容量が少ないときには供給点と需要点の間のマッチングが増えてしまうが, 図 10 の a, b 間のように需要点の経路の探索が少ない. 同様に図 10 の c, d 間のように供給点の経路の探索も少なくなる可能性もある. そして図 6 からマッチングの重みは小さいということがわかる. これらのことから CMR アルゴリズムは車両の容量が 2 のとき, 他の容量と比べて良い解が得られる. それに対して, CKR アルゴリズムは車両の容量が大きいときには, マッチングをとる部分の数が少ないので良い値が得られる (実際には, 車両の容量が 32, 64 のときは 巡回路を 1 周することで済んでいる). 車両の容量が小さいときにはマッチングが多く起こるので, あまり良い値が得られない. 特に車両の容量が 2 のときは 1 つ 1 つの点すべてが [P] または [N] になってしまうためよい解が得られない. しかし, 車両の容量が小さいときには, 巡回路上で [P] と [N] が連続していることが多くあるので, 上で述べたような局所改善を行なうことである程度改善されている.

また, この問題では最適な巡回路を求めることはできないので, 下限として MST の長さを利用すると, この実験での近似率は表 1 のようになる. この表からわかるように, CMR アルゴリズム (理論的な近似率は 9.5) は容量 16 のときに最も悪い値で, 3.3 近似となっている. また CKR アルゴリズム (理論的な近似率は 5) と それを局所改善したものはそれぞれ容量 2 のときに最も悪い値で, それぞれ 3.3 近似, 2.8 近似となっている.

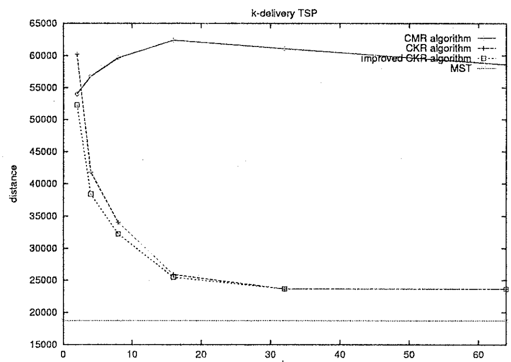


図 9: 3 つのアルゴリズムの性能比較

4 複数デポーの配送計画問題に対する提案

ここでは提案する手法および計算機実験について述べる. 従来の目的である配送に要する総距離最小を目指したもの, そして目的を拡張したものを述べる. すべての提案手法の方針は, 複数デポーの配送計画問題をいくつかの部分問

表 1: 近似率

容量	CMR	CKR	improved CKR
2	2.9	3.3	2.8
4	3.1	2.3	2.1
8	3.2	1.9	1.8
16	3.3	1.4	1.4
32	3.3	1.3	1.3
64	3.2	1.3	1.3

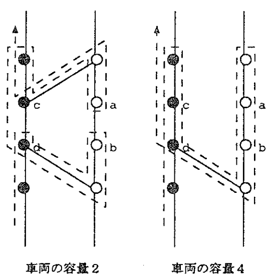


図 10: CMR アルゴリズムのルートの例

題に分割し, 供給点が 1 つである容量 k の巡回路被覆問題に帰着させるという考え方である. まず, 準備として容量 k の巡回路被覆問題を解く近似アルゴリズムを概説する.

4.1 容量 k の巡回路被覆問題に対する近似アルゴリズム

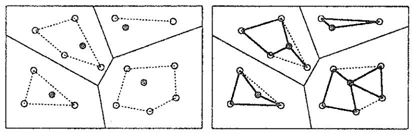
この問題に対する近似アルゴリズムで PTAS を成し遂げているものが存在するが [3], ここでは実行時間の観点から簡単なアルゴリズムを使用する.

1. すべての需要点を通る巡回路を TSP に対する Christofides の 1.5 近似アルゴリズムを用いて求める.
2. 求めた 巡回路に沿って車両容量を越えるところで辺を除去し, 要求量の和が高々車両容量となるパスに分割する.
3. パスの端点と原点を結ぶことにより巡回路被覆を構成する.

4.2 提案手法 I

ここで述べる手法は従来の目的, 配送に要する総距離最小化を目指したものである. 容量 k の巡回路被覆問題に帰着させるため, n 個の需要点を $m(< n)$ 個の供給点に割当てることを考える. 距離最小化のため各需要点を最も近い供給点に割当てる. 以下に手順を示す.

1. 供給点に対してボロノイ領域を求める.
2. 各ボロノイ領域対して容量 k の巡回路被覆問題を解く.



提案手法の解法

4.2.1 計算機実験

この方法の性能評価をするため, 計算機実験を行なった. インスタンスは表 2 にまとめたような特徴をもつ 23 個について行なった. このデータは
ftp://ftp.crt.umontreal.ca./pub/users/gilbert/tabu/
にあるベンチマークデータである. インスタンス p01-p07 は Christofides-Eilon から引用されたものである. インスタンス p08-p11 は Gillett-Johnson で述べられたものである. インスタンス p12-p23 は Chao-Golden-Wasil によって提供されたものである. 実験では, 総距離 (近似解) と近似率を求めた. 使用プログラミング言語は C 言語である. 近似率をグラフにしたのが図 11 である.

すべてのインスタンスに対して, 近似率が 1.2 から 1.4 とよい性能の解が得られた.

4.3 提案手法 II

提案手法 I は供給点の配置によりボロノイ領域を求め, 需要点をおのおのの供給点に割当てるものである. そのためこの方法は需要点の情報を無視してしまっていた. そこで需要点の情報を考慮した手法を提案する. これは各供給点にある車両が訪れる需要点の数の均等化を考慮するものである. このような割り当てを求める

表 2: インスタンス

番号	需要点数	供給点数	1ルートの最大距離	車両容量
p01	50	4	∞	80
p02	50	4	∞	160
p03	75	2	∞	140
p04	100	2	∞	100
p05	100	2	∞	200
p06	100	3	∞	100
p07	100	4	∞	100
p08	249	2	310	500
p09	249	3	310	500
p10	249	4	310	500
p11	249	5	310	500
p12	80	2	∞	60
p13	80	2	200	60
p14	80	2	180	60
p15	160	4	∞	60
p16	160	4	200	60
p17	160	4	180	60
p18	240	6	∞	60
p19	240	6	200	60
p20	240	6	180	60
p21	360	9	∞	60
p22	360	9	200	60
p23	360	9	180	60

には, 整数計画問題を解けばよい. 需要点に関する添字 i, 供給点に関する添字 j, ある需要点 i とある供給点 j 間の距離 c_{ij}, そして決定変数 x_{ij} を需要点 i が供給点 j に割り当てられるとき 1, そうでないとき 0 をとる値とすると, 以下のように定式化できる.

$$\min \quad \sum_i \sum_j c_{ij} x_{ij} \tag{1}$$

$$s.t. \quad \sum_i x_{ij} \le \left\lceil \frac{n}{m} \right\rceil \qquad j = 1, \cdots, m \tag{2}$$

$$\sum_j x_{ij} = 1 \qquad i = 1, \cdots, n \tag{3}$$

$$x_{ij} \in \{0, 1\} \tag{4}$$

式 (2) が需要点を供給点に等配分する制約である. 上の問題は整数計画問題であり, 多項式時間では解けない. この問題を難しくしているのが式 (4) である. そこでその式 (4) の整数制約を以下のように緩和する.

$$0 \le x_{ij} \le 1 \tag{5}$$

上の定式化において式 (4) を式 (5) に置き換え, 線形計画問題として定式化する. 手法 II の手順は以下のようになる.

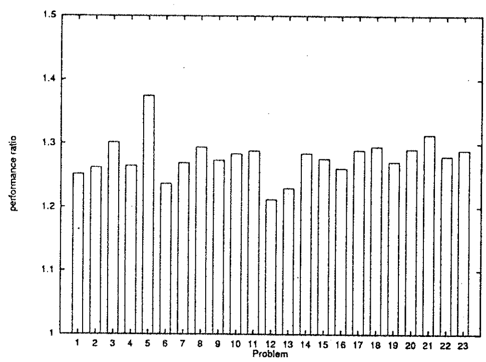


図 11: 各問題に対する近似率

1. 上の線形計画問題を解き, 分数解が得られた場合にはランダムラウンディングにより整数解を求める. それにより, すべての需要点を供給点に割り当てる.
2. 需要点が等配分された領域対して容量 k の巡回路被覆問題を解く.

4.4 提案手法 III

手法 II で供給点の情報も考慮にいれたが, まだ供給点の要求量の情報は考えられていない. そこで上と同じ変数, および需要点 i の要求量を d_i, 要求量の総和を D, 要求量の最大値を d_{max} とすると, 上で線形計画問題に定式化した式 (2) を

$$\sum_i d_i x_{ij} \le \left\lceil \frac{D}{m} \right\rceil + d_{max} \qquad j = 1, \cdots, m$$

に置き換えればよい.

5 時間帯指定配送計画問題

この問題は配送に要する総距離最小という目的でありながら, 時間帯制約と容量制約をもつ. ここでの考え方は, 目的である総距離最小化のため, ある程度距離の短い経路を作成した後, 時間帯制約と容量制約を考慮する方針である. 以下に手順の概要を示す.

1. 需要点の集合で TSP を解く.
2. ある方向 (時計回り, または反時計回り) での時間制約を守った最小車両台数本の

経路を求める．これは2部グラフのマッチングにより成し遂げる．

3. 求めた経路に対して容量 k の巡回路被覆問題を解く．

5.1 計算機実験

この問題に対して計算機実験を行なった．適用した問題例はSolomonによるベンチマーク56題である．このデータは需要点 $n = 100$ であり，特徴別に3種類に分けられている．

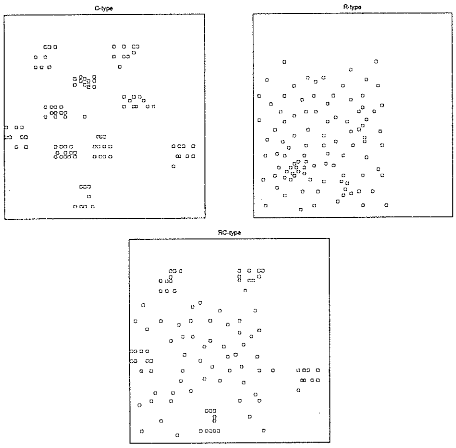


図 12: 3種類のインスタンス

各問題に対する時計回りと反時計回りそれぞれで得られた長さを図13に示す．

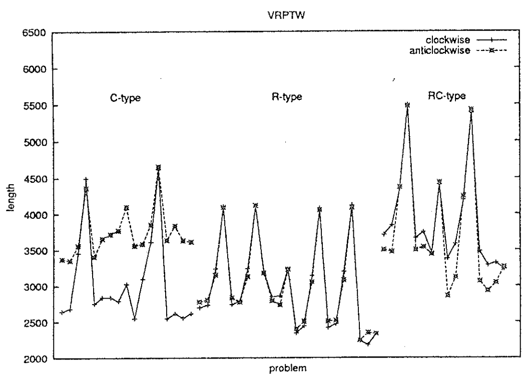


図 13: 各問題に対する総距離

6 おわりに

複数デポーの配送計画問題に対して別の指標を取り入れた．この方法は実行時間の観点からも有効性が高いと思われる．時間帯指定の配送計画問題に対しては実行時間を抑えた手法を提案できたが，近似率という面であまりよい結果は得られなかった．今後の課題としては，時間帯指定の配送計画問題に対しての新たな手法，現実データでの有効性の確認があげられる．

本研究の実験等に協力して頂いた同研究室の八巻満隆氏に感謝します．本研究は，一部，中央大学理工学研究所，同先端技術センターおよび文部省科学研究費補助金からの援助のもとで行われたものである．

参考文献

[1] S. Anily and J. Bramel: Approximation algorithms for the capacitated traveling salesman problem with pick-ups and deliveries, Manuscript, 1997.

[2] S. Anily and R. Hassin: The swapping problem, *Networks*, Vol.22 (1992), pp.419-433.

[3] T. Asano, N. Katoh, H. Tamaki and T. Tokuyama: Covering points in the plane by k-tours: towards a polynomial time approximation scheme for general k, *Proc. 29th. Annual ACM Symposium on the Theory of Computing*, 1997, pp.275-283.

[4] Y. Bartal: On approximating arbitrary metrics by tree metrics, *Proc. 30th Annual ACM Symposium on the Theory of Computing*, 1998, pp.161-168.

[5] P. Chalasani, R. Motwani and A. Rao: Algorithms for robot grasp and delivery, *2nd International Workshop on Algorithmic Foundations of Robotics*, 1996.

[6] M. Charikar, S. Khuller and B. Raghavachari: Algorithms for capacitated vehicle routing, *Proc. 30th. Annual ACM Symposium on the Thory of Computing*, 1998, pp.349-358.

[7] M. Charikar and B. Raghavachari: The finite capacity dial-a-ride problem, *Proc. 38th Annual IEEE Symposium on the Foundations of Computer Science*, 1998, pp.458-467.

[8] M. Haimovich and A.H.G. Rinnooy Kan: Bounds and heuristics for capacitated routing problems, *Math. Oper. Res.* Vol.10(1985), pp.527-542.