配送問題の研究

2003MM011 後藤壮智　2003MM028 石井大輔
指導教員: 鈴木敦夫

1　はじめに

1.1　研究目的

我々は, 配送に関する問題の最適化について研究する. 研究の対象としているある運送会社は, 1日に多くの顧客先を周り部品などを配送している. そこで問題になっている事は, 顧客を周るルートが果たして最適なルート（最短距離）なのかということである. またこの会社では多くの顧客を抱えているために, それぞれの地区に分けて配送を行っている. そして, 新しい顧客が加わった時にはその地区内の配送ルートに単純に組み込んでいる. そのため, 地区間で顧客数のばらつきがあり, それを解消するために地区の再編成を行って, 最適なルートを求める必要がある.
そこで, ここではすべての顧客をそれぞれ必ず1回だけ訪問して, 最短距離で訪問する巡回路を算出する問題と, 隣接する地区を併合して, 新たに最適な2ルートの算出する問題に取り組む. ここでは, この問題を巡回セールスマン問題（Traveling Salesman Problem）として定式化し, 最適化ソフトウェア(Xpress-MP)を用いて解を求める.

1.2　過去の研究

巡回セールスマン問題(Traveling Salesman Problem TSPと略記)とは, 多くの都市と各都市間の移動コストが与えられたとき, 全ての市を1度だけ周って戻ってくるルートのうちコスト最小のものを求める（セールスマンが所定の数の都市を1回だけ巡回する場合の最短経路を求める）問題である. この問題は, ナップサック問題などと同様にNP困難であることが知られている. 巡回セールスマン問題はNP困難ではあるものの実質, 分枝限定法や, 切除平面法により, 約2000都市以内ならば普通のパソコンでも, およそ1日以内で厳密解を得られることが多い.

また巡回セールスマン問題として過去には配送問題だけに留まらず, 小学校, 中学校, 高校, 大学などのスクールバスの効率的な巡回路の計画, 郵便や新聞などの配達経路, ゴミの収集経路の計画など近年配送というものをいかに効率よくできるかという事に, 注目され研究されている. また配送のみに留まらず荷物の集荷の分野にも使うことが可能であるために, 現在は運搬経路問題(vehicle routing problem)と呼ばれている.

1.3　問題へのアプローチ

本研究では, 配送ルートの距離を最短とすることを目的関数として, 問題を解く. ただし, ここで用いた距離は道路距離ではなく, 配送先の2点間の直線距離を使う事にする. 道路距離ではないので, 実際のデータとは異なるが, すべてのデータにおいて直線距離を用いることで, 条件をそろえた. ここでは, Xpress-MPを用いてその得られた結果を最適解とする.

2　データの整備

ある会社の配送に関するデータを頂き現状のルートの分析を行う.

2.1　データの説明

頂いたデータは以下のように記されている. 例として「市内便」の1日分のデータを記載する.

2.2　使用した道路距離のデータ

ここで使用した街区レベル位置参照情報[1]とは, 全国の都市計画区域を対象に, 街区単位(「○○町△丁目□番」)の位置座標(代表点の緯度・経度、平面直角座標)を整備したデータである. そして, その個々の住所の座標から 直線距離を求めた.

表 1: 座標データの例

住所	X軸	Y軸
名古屋市港区竜宮町	-100245	-24379.9
名古屋市港区遠若町	-100131	-27349.7
名古屋市港区宝神＊丁目	-100584	-28655.2
名古屋市港区本宮町＊丁目	-99109.4	-27856.8
名古屋市中川区松年町	-98587.6	-26629.3
名古屋市中川区三ツ屋町	-97545.5	-26677.6
名古屋市中川区清川町＊丁目	-95504.6	-26707.8
名古屋市中川区福川町	-96386	-26331
名古屋市中川区＊番町	-97506	-26089.7
東海市元浜町	-109126	-27000.8
知多市八幡字西前田	-111514	-26592.1
東海市荒尾町南脇	-107137	-24433
東海市荒尾町ワノ割	-106178	-24293.7
名古屋市緑区大高町	-103755	-22704
名古屋市港区竜宮町	-100245	-24379.9

(単位：m)

このデータを基に, 次の事が分かる.

1. この会社は分けられた区間によって配送が行われている. 現在,「北方便」,「刈谷便」,「豊田便」,「春日井便」,「瀬戸便」,「稲沢便」,「市内便」,「大垣便」,「新三重便」,「三重便」,「東南便」,「小牧便」の計12便が配送を行っている.
2. 各便には, 日にちごとの配送順が記されている.
3. それぞれの配送を行なう会社の情報は, 住所だけである.
4. 配送は毎日行われるのではなく, 顧客先によっては週に2日, 3日というところもある.

ここでは, 現在行われている配送の順番にそれぞれ自然数番号を付けた.

2.3 現状ルート分析

ここでは, 「市内便」について各日ごとのルートを分析して, その後最適なルートと比較する. その他の地域は付録に記載する
ここでは, 現状の「市内便」ルートの分析をする.

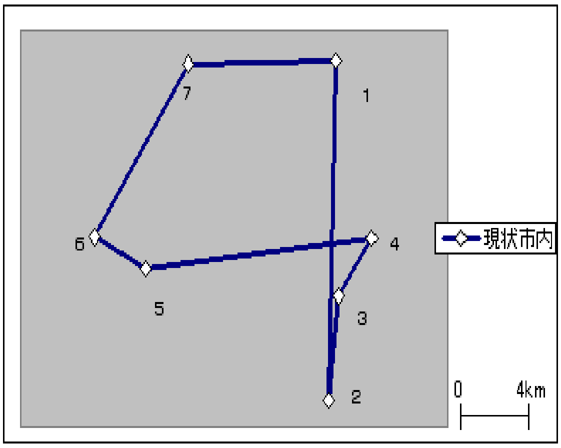

図 1: 現状の「市内便」5日目のルート（1月28日）

表2: 改善前の「市内便」各ルートの直線距離

日付	距離	日付	距離
24日	39.6	27日	38.5
25日	35.9	28日	32.1
26日	40.7	30日	39.8

(単位：km)

図1より, 交差しているポイントがあるため最適なルートとは考えにくい. そのため, このようなポイントを改善し, 最適なルートを算出する.

3 1区間内における最適巡回路の算出

3.1 記号の定義

まず定式化にあたり, 記号を定義する.

初めに, 添字の設定を行う.

i, j, h : 各顧客の番号

$(i, j) \in A$: 枝

I : iの添字集合

J : jの添字集合

H : hの添字集合

次に, 変数の定義を行う.

$$x_{ij} = \begin{cases} 1 \cdots & \text{顧客}i\text{から}j\text{へ配送するとき} \\ 0 \cdots & \text{それ以外のとき} \end{cases}$$

u_i : 排除制約のためのダミー変数

最後に, 定数の設定を行う.

c_{ij} : 顧客iと顧客jの間の距離

N : 顧客の数

3.2 定式化

3.2.1 目的関数

$$\sum_{(i,j) \in A} c_{ij} x_{ij} \to \min \qquad (1)$$

この目的関数は, 顧客iと顧客jの距離行列と0-1変数を使い, 運搬車の移動距離を最小化するために設定する.

3.2.2 制約条件

$$\sum_{j \in J} x_{ij} = 1 \qquad (i = 1, ..., N) \qquad (2)$$

$$\sum_{h \in H} x_{jh} = 1 \qquad (j = 1, ..., N) \qquad (3)$$

$$u_i - u_j + N x_{ij} \leq N - 1 \quad (i,\ j = 2, ..., N) \qquad (4)$$

$$u_i,\ v_j \geq 0 \qquad (i,\ j = 1, ..., N) \qquad (5)$$

$$I,\ J,\ H = \{1, ..., N\} \qquad (6)$$

変数制約

$$x_{ij} \in \{0,\ 1\} \quad (i \in I,\quad j \in J) \qquad (7)$$

制約条件の説明

制約条件(2)：運搬車は顧客iからいずれかの顧客jに出発しなければならないとする.（出発地制約）
制約条件(3)：運搬車は顧客hへどこかの顧客jから入らなければならないとする.（目的地制約）
制約条件(4)：閉路排除制約式である. 顧客を出発したらまた元の場所に戻ってくる事はできないようにする. これを初めに設定せずに解かせたところ, ルートが複数個の解として求められてしまった. この閉路排除制約を加えることによって, データに対して1つのルートを作成する事ができるようになった.
制約条件(5)：非負制約である.
制約条件(6)：I, J, Hはそれぞれ1からNまでの自然数とする.

3.3 1区間内における最適巡回路の実行結果

表3: 改善前後の「市内便」各ルートの直線距離比較

日付	改善前の距離	改善後の距離	短縮率
24日	39.6	36.4	8.1
25日	35.9	32.1	10.6
26日	40.7	35.5	12.8
27日	38.5	36.7	4.7
28日	32.1	28.7	10.6
30日	39.8	37.1	6.8

(単位：km km %)

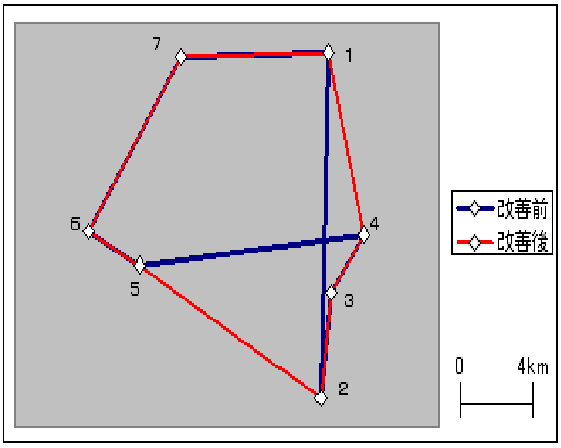

図 2: 改善前後の「市内便」5日目のルート（1月28日）

「市内便」５日目の改善ルート
「１－７－６－５－２－３－４－１」

3.4 考察

以上のように, 各便,各日にちとも距離の短縮に成功した. 図をみると分かるが, 初期状態でみられた交差する点が改善され, 距離が短縮された.

4 ２区間内における最適巡回路の算出

次にデータの中から隣接している「瀬戸便」, 「豊田便」を選択し, 現在行われているそれぞれの配送を組み合わせ, 新たに効率的な２区間での最適巡回路を生成できないかを考える. 先ほどと同様に配送先の企業の住所データより各顧客先の住所を調べ, その顧客間の直線距離を求めて最適化を図る.
今回新しく求めるルートをそれぞれルート1, ルート2とする.

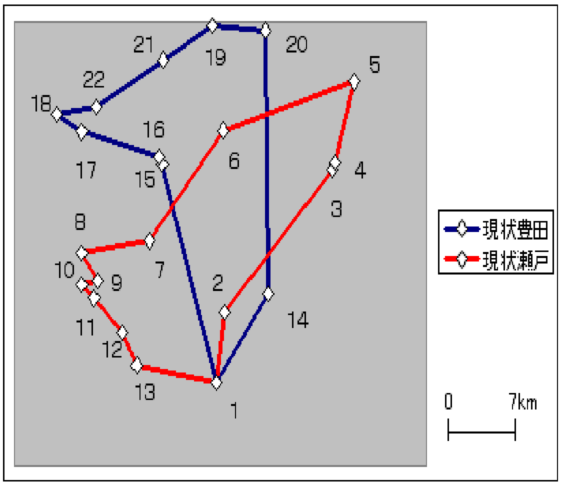

図 3: 現状の「瀬戸豊田便」1日目のルート（1月24日）

4.1 記号の定義

まず定式化にあたり, 記号を定義する.

初めに, 添字の設定を行う.

i, j, h : 各顧客の番号
$(i, j) \in A$: 枝
I : iの添字集合
J : jの添字集合
H : hの添字集合

次に, 変数の定義を行う.

$$x_{ij} = \begin{cases} 1 \cdots & \text{ルート1で顧客}i\text{から}j\text{へ配送するとき} \\ 0 \cdots & \text{それ以外のとき} \end{cases}$$

$$y_{ij} = \begin{cases} 1 \cdots & \text{ルート2で顧客}i\text{から}j\text{へ配送するとき} \\ 0 \cdots & \text{それ以外のとき} \end{cases}$$

u_i : ルート１での排除制約のためのダミー変数
v_i : ルート２での排除制約のためのダミー変数

最後に, 定数の設定を行う.

c_{ij} : 顧客iと顧客jの間の距離
N, N_1, N_2 : 顧客の数　$(N = N_1 + N_2)$

4.2 定式化

4.2.1 目的関数

$$\sum_{(i,j)\in A} c_{ij}x_{ij} + \sum_{(i,j)\in A} c_{ij}y_{ij} \rightarrow \min \tag{8}$$

この目的関数は, 顧客iと顧客jの距離行列と0-1変数を使い, 運搬車の移動距離を最小化するために設定する.

4.2.2 制約条件

$$\sum_{(i,j)\in A} x_{ij} = N_1 \tag{9}$$

$$\sum_{(i,j)\in A} y_{ij} = N_2 \tag{10}$$

$$\sum_{i\in I} x_{ij} = \sum_{h\in H} x_{jh} \qquad (j = 2, ..., N) \tag{11}$$

$$\sum_{i\in I} y_{ij} = \sum_{h\in H} y_{jh} \qquad (j = 2, ..., N) \tag{12}$$

$$\sum_{j\in J} x_{ij} + \sum_{j\in J} y_{ij} = 1 \qquad (i = 2, ..., N) \tag{13}$$

$$\sum_{i\in I} x_{ij} + \sum_{i\in I} y_{ij} = 1 \qquad (j = 2, ..., N) \tag{14}$$

$$\sum_{i\in I} x_{1j} = 1 \qquad (i = 2, ..., N) \tag{15}$$

$$\sum_{j\in J} x_{i1} = 1 \qquad (j = 2, ..., N) \tag{16}$$

$$\sum_{j\in J} y_{1j} = 1 \qquad (i = 2, ..., N) \tag{17}$$

$$\sum_{j\in J} y_{i1} = 1 \qquad (j = 2, ..., N) \tag{18}$$

$$u_i - u_j + N_1 x_{ij} \leq N_1 - 1 \quad (i,\ j = 2, ..., N) \tag{19}$$

$$v_i - v_j + N_2 y_{ij} \leq N_2 - 1 \quad (i,\ j = 2, ..., N) \tag{20}$$

$$x_{ij} + y_{ij} \leq 1 \qquad (i,\ j = 1, ..., N) \tag{21}$$

$$I,\ J,\ H = \{1, ..., N\} \tag{22}$$

$$u_i,\ u_j,\ v_i,\ v_j \geq 0 \qquad (i,\ j = 1, ..., N) \tag{23}$$

変数制約

$$x_{ij} \in \{0,\ 1\} \ \ (i \in I,\quad j \in J) \tag{24}$$

$$y_{ij} \in \{0,\ 1\} \ \ (i \in I,\quad j \in J) \tag{25}$$

制約条件の説明

制約条件(9)(10)：ルート1の顧客の数をN_1，ルート2の顧客の数をN_2と表す．
制約条件(11)(12)：顧客iから顧客jに向かう枝と，顧客jから顧客hに向かう枝の数は同じでなければならない．
制約条件(13)(14)：ルート1とルート2で同じ顧客を通らないようにする．
制約条件(15)(16)(17)(17)：X, Yはある基準点（スタート地点）から出発してまたその地点に戻ってこなければならない．
制約条件(19)(20)：閉路排除制約式である．ある顧客先を出発したらまた元の場所に戻ってくる事はできないようにする．
制約条件(21)：x, yで同じ枝は通れないものとする．
制約条件(22)：$I,\ J,\ H$はそれぞれ1からNまでの自然数とする．
制約条件(23)：非負制約である．

4.3 ２区間内における最適巡回路の実行結果

表4: 改善前後の「瀬戸豊田便」各ルートの直線距離比較

日付	改善前の距離	改善後の距離	短縮率
24日	111.5	100.5	9.9
26日	115.1	100.7	12.5
28日	73.8	73.8	0
30日	115.1	101.2	12.0

(単位：km km %)

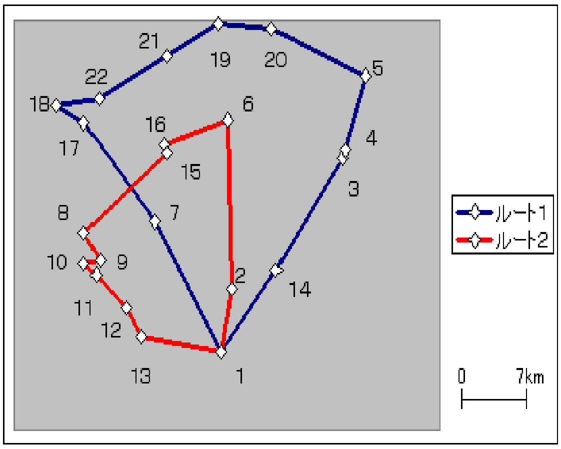

図 4: 改善された「瀬戸豊田便」1日目のルート（1月24日）

4.4 考察

改善前の２区間内で用いているルートは１区間内で最適化したルートを用いているが，改善後のルート距離は短縮されている．以上の事より，3, 4区間などで最適化した場合もさらに効率の良いルートが作成されることが予想される．またこの問題を解くにあたって3時間48分要した．

また例として，制約式に $\sum_{(i,j)\in A} C_{ij}X_{ij} \leq 50$ という式を加えることによって，運搬車の距離50km以下という制限を持たせてルートを周らせることが可能になる．これによって緊急時に対処するための待機用運搬車にすることが可能になる．

5 おわりに

配送計画の研究を進めるにあたって，初めに簡単な巡回セールスマン問題の定式化し最適解を求めた．その後隣接する地区で再編成を行い，最適なルートを求めた．

またすべての距離データは，直線距離を用いたため，道路距離を用いることによって，より実用性の高いものになる．

参考文献

[1] 国土地理院情報整備室HP: http://www.gsi.go.jp.