注文順を考慮した分割配送問題

込江　良仁（沼田　一道助教授）

1. はじめに

日常品を販売する小売店が、サービスとして商品の配達を行うということはよく見られる。小規模店の場合、１台の配送車で何回かに分けて配送している。早い回から注文先着順に商品を積み込み、積み込んだものを走行距離の短くなるような順に配達していくというのが一般的であるが、発想の転換を伴う改善の余地がかなりあると思われる。本研究では、この問題を「配送問題」として定式化し、「先着順」という制約を緩めることがどの程度の効果を生むか考察する。

2. 現状と問題点

（１）**現状**…対象とする酒類小売店は、東京都北区においてお酒の宅配サービスを行っている。配達方法は、次のような手順になっている。

①顧客の注文を配達日の前日までに受注し、１トン積みの車一台を使い配達している。

②１日を４つの時間帯に分け、それぞれ１０時～１２時、１３時～１５時、１５時～１８時、１８時～２０時としている。

③それぞれの時間帯の平均的な配達件数は、１０時～１２時、１３時～１５時、１８時～２０時の時間帯では８件程度、１５時～１８時では１２件程度である。

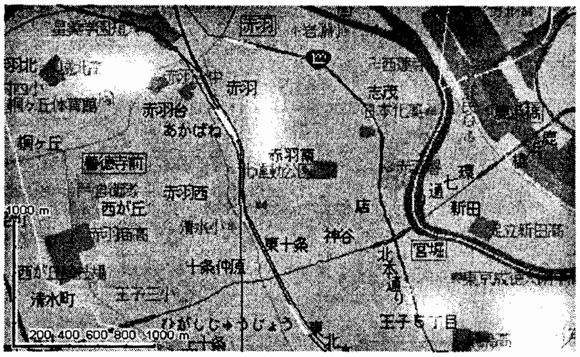

図１．配達地区

④各時間帯での配達先は、早い時間帯から遅い時間帯へ注文の先着順に顧客を割り当てしたものである。

⑤各時間帯内の配達順は、必ずしも先着順にならなくてもよい。

（２）**問題点**…各時間帯の標準的な配達件数は、ある程度余裕を持たせたものである。これは、配達先がバラバラに散らばっている場合でも、時間内に配達しきれる件数を考慮している。例えば、配達地区を端から端に移動するような場合、移動の時間がかかり、時間をフルに使わなければ全配達先を廻りきれない。逆に、ある地域に配達先が集中している場合は、時間を持て余してしまう。

また、現在の配達方法では各時間帯ごとに同じ地区や、マンション、団地を２度３度訪問する事があり効率が悪い。

3.改善目標

現在の先着順に配達先を各時間帯に割り当て、各時間帯ごとに配送順を決定していく方法では効率が悪い。そこで、先着順という制約を少し緩め、時間帯と時間帯で顧客を入れ替える事で次の３つの目標をできるだけ満足させるようにし問題点を改善したい。

①各時間帯の中で、店を出発し全配達先を廻り再び店に帰るまでの移動距離ができるだけ短くなるように４つの時間帯に分割する。

②各時間帯ごとにかかる総走行距離の比ができるだけ（２：２：３：２）となるようにする。

③配送順を入れ替える際に、顧客が不満を持たないように、早めに受注した注文は早い時間帯になるようにする。

4.定式化

$o(i=1,\cdots,n)$を先着順に並べた顧客とし、時間帯をj $(j=1,\cdots,4)$とする。顧客o_iが時間帯jに属するかそうでないかは$0-1$変数x_{ij}で表す（(4）式)。時間帯jに属する顧客の集合S_jを（2）式で表す。すべての顧客は時間帯jのいずれかに必ず含まれ、2 つ以上に含まれる事はない、これを（3）式で表す。店を出発しS_jに属するすべての顧客を廻り再び店に戻る最短巡廻路の距離を$L(S_j)$, 時間帯jの長さを$\alpha_j(\alpha_1=\alpha_2=\alpha_4=2,\alpha_3=3)$とすると、時間帯$j$において時間の長さ$\alpha_j$に対する総移動距離のバラツキをできるだけ小さくするという目標は（1）のようにかける。

$$\textit{Minimize} \quad \left(\max_j \frac{L(S_j)}{\alpha_j}\right) \qquad (1)$$

$$\textit{sub.to} \quad S_j=\left\{o_i \,\middle|\, \underset{(1\le i\le n)}{x_{ij}}=1\right\} \quad (2),\quad \sum_{j=1}^{4} x_{ij}=1 \quad (3),\quad x_{ij}\in\{0,1\}(i=1,\cdots,n;j=1,\cdots,4) \quad (4)$$

5．入れ替えモデル

各時間帯の間で顧客を入れ替えるときに、制約の緩め方によって異なるモデルが考えられる。

制約条件：$^\forall o_i \in S_j$, $^\forall o_{i'} \in S_{j+K}$, $i<i'$ （k…時間帯の飛び数）

※入れ替え後kだけ離れた時間帯の任意の要素を比較したとき、必ず早い時間帯の顧客の番号iは遅い時間帯の顧客の番号i'よりも小さくなければならない。

モデル1…$^\forall o_i \in S_1, {^\forall o_{i'}} \in S_3, i<i'$, $^\forall o_i \in S_1, {^\forall o_{i'}} \in S_4, i<i'$, $^\forall o_i \in S_2, {^\forall o_{i'}} \in S_4, i<i'$, $(k=2,3)$

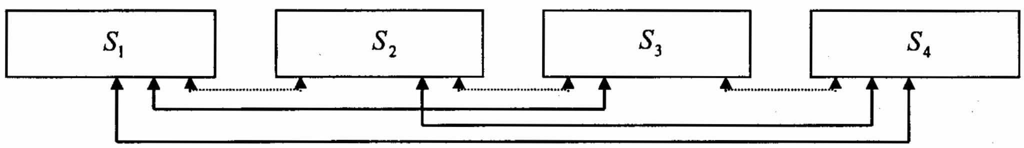

図2．入れ替えモデル1

モデル2…$^\forall o_i \in S_1, {^\forall o_{i'}} \in S_4, i<i'$, $(k=3)$

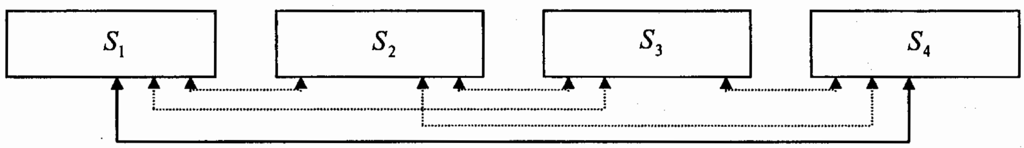

図3．入れ替えモデル2

モデル３…制約条件なし

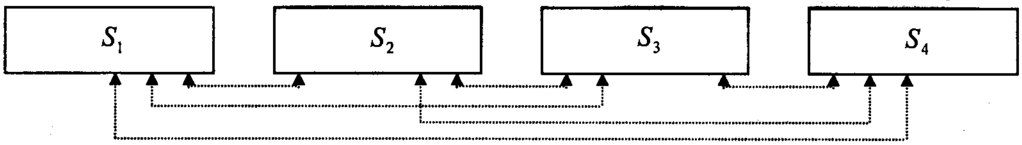

図４．入れ替えモデル３

６．解法

S_jが与えられたとき、最適道順を求める問題は巡回セールスマン問題（ＴＳＰ）となる。このこ回セールスマン問題の解法としては、法を用いる。

＜全体の解法＞

Step0　各時間帯の客数n_1,n_2,n_3,n_4を設定する。$f^* \leftarrow \infty$。

Step1　全顧客を受注順（先着順）に時間帯S_1,S_2S_3,S_4にS_iの客数がn_iとなるように分割する。

Step2　S_1,S_2S_3,S_4からS_y,S_zを選び、$o_p \in S_y, o_q \in S_z$を選ぶ。

（但し、入れ替えても順序制約条件を侵さないもの）

Step3　$S_y' = S_y - \{o_p\} + \{o_q\} S_z' = S_z - \{o_q\} + \{o_p\}$について店を入れたTSPを解く。

$L(S_y') + L(S_z') < L(S_y) + L(S_z)$ならば$o_p$と$o_q$を入れ替え、新たに$S_y' \rightarrow S_y, S_z' \rightarrow S_z$

とし入れ替えができなくなるまでStep2～Step3を繰り返す。

Step4　$\dfrac{L(S_j)}{\alpha_j}$が最大の時間帯を$j_{\max}$、最小の時間帯を$j_{\min}$とする。$f = \dfrac{L(S_{j_{\max}})}{\alpha_{j_{\max}}}$

もし$f < f^*$ならば$f^* \leftarrow f$とし$n_{j_{\max}} \leftarrow n_{j_{\max}} - 1$，$n_{j_{\min}} \leftarrow n_{j_{\min}} + 1$としStep1へ、さもなければ終り。

７．計算機を使った実験

本研究では、制約の違いによりどのような最適解が得られるか、また、どのように異なるのか検討するために、プログラムを作成し計算機実験を行った。データは１９９９年１１月６日のものを使用した。

表１．注文順

1：王子５	10：王子５	19：岩淵	28：東十条５
2：王子５	11：神谷２	20：東十条２	29：王子５
3：王子５	12：赤羽西５	21：新田３	30：志茂４
4：神谷３	13：志茂３	22：東十条４	31：赤羽１
5：志茂４	14：岩淵	23：新田３	32：赤羽西１
6：赤羽２	15：志茂４	24：志茂３	33：新田３
7：東十条３	16：赤羽西２	25：志茂２	34：神谷３
8：新田１	17：東十条２	26：東十条５	35：神谷３
9：新田１	18：新田３	27：東十条３	

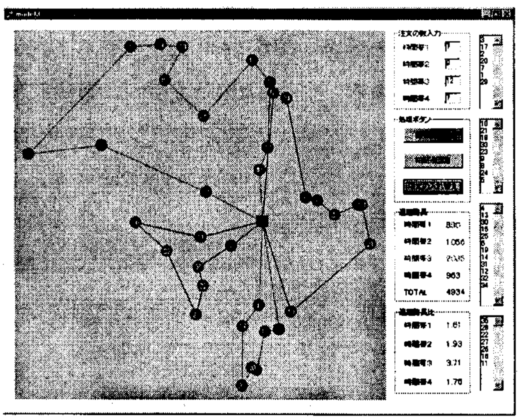

図５．実行画面

8．実験結果

以下に、各モデルの分割結果および巡廻路長を示す。

表2．先着順の分割

10時～12時	13時～15時	15時～18時	18時～20時
1：王子5	9：新田1	17：東十条2	28：東十条5
2：王子5	10：王子5	18：新田2	29：王子5
3：王子5	11：神谷2	19：岩淵	30：志茂4
4：神谷3	12：赤羽西5	20：東十条2	31：赤羽1
5：志茂4	13：志茂3	21：新田3	32：赤羽西1
6：赤羽2	14：岩淵	22：東十条4	33：新田3
7：東十条3	15：志茂4	23：新田2	34：神谷2
8：新田1	16：赤羽西2	24：志茂3	35：神谷3
		25：志茂2	
		26：東十条5	
		27：東十条3	

表3．モデル1の分割結果

10時～12時	13時～15時	15時～18時	18時～20時
1：王子5	4：神谷3	19：岩淵	17：東十条2
2：王子5	5：志茂4	24：志茂3	18：新田2
3：王子5	6：赤羽2	25：志茂2	20：東十条2
7：東十条3	8：新田1	26：東十条5	21：新田3
10：王子5	9：新田1	28：東十条5	22：東十条4
11：神谷2	12：赤羽西5	30：志茂4	23：新田2
16：赤羽西2	13：志茂3	31：赤羽1	27：東十条3
	14：岩淵	32：赤羽西1	29：王子5
	15：志茂4	33：新田3	
		34：神谷2	
		35：神谷3	

表4．モデル2の分割結果

10時～12時	13時～15時	15時～18時	18時～20時
1：王子5	5：志茂4	3：王子5	17：東十条2
2：王子5	13：志茂3	6：赤羽2	20：東十条2
4：神谷3	15：志茂4	11：神谷2	22：東十条4
7：東十条3	18：新田2	12：赤羽西5	26：東十条5
8：新田1	21：新田3	14：岩淵	27：東十条3
9：新田1	23：新田2	16：赤羽西2	28：東十条5
10：王子5	24：志茂3	19：岩淵	29：王子5
	30：志茂4	25：志茂2	35：神谷3
	33：新田3	31：赤羽1	
		32：赤羽西1	
		34：神谷2	

表5．モデル3の分割結果

10時～12時	13時～15時	15時～18時	18時～20時
2：王子5	1：王子5	4：神谷3	8：新田1
7：東十条3	3：王子5	5：志茂4	9：新田1
10：王子5	11：神谷2	6：赤羽2	18：新田2
17：東十条2	20：東十条2	12：赤羽西5	24：志茂3
21：新田3	22：東十条4	13：志茂3	30：志茂4
23：新田2	26：東十条5	14：岩淵	33：新田3
29：王子5	27：東十条3	15：志茂4	34：神谷2
	28：東十条5	16：赤羽西2	
	35：神谷3	19：岩淵	
		25：志茂2	
		31：赤羽1	
		32：赤羽西1	

9．考察

計算機実験により、先着順の制約が緩くなるほど巡廻路長は短縮できるといえる。また、先着順の分割方法では、時間帯の持ち時間の比＝巡廻路長ではなかったものを、持ち時間に比例した移動距離を各時間帯に配分する事ができた。

表6．巡廻路長

	先着順の分割	モデル1の分割	モデル2の分割	モデル3の分割
10時～12時	1974（8件）	1410（7件）	1213（7件）	1211（7件）
13時～15時	2152（8件）	1876（9件）	1107（9件）	861（9件）
15時～18時	2451（11件）	2234（11件）	2124（11件）	2111（12件）
18時～20時	2437（8件）	1342（8件）	1108（8件）	1157（7件）
総巡廻路長	9014	6862	5552	5340

10．おわりに

本研究では、酒類小売店の商品配達を注文順に行う配送問題を「順序制約」を段階的に緩和した3つのモデルを提案し、その違いについて考察した。実際に、本研究を適用する事を考えると、どのモデルを用いても効果的であると考えられるが、顧客満足と業務の効率化を考えた場合、モデル1を用いるのが無難である。また、現状のプログラムでは、簡単な最適化手法を用いているが、解法の高精度化等は今後の課題である。

【参考文献】

[1] 岩倉　行信：“巡廻セールスマン問題に対する発見的解法の高速化”，東京理科大学工学部第一部経営工学科卒業論文，（１９９８）

[2] 小出　俊夫：“Borland　Delphi3.1　リファレンスガイド”，秀和システム，（１９９８）