

# 不動産管理業務における定期点検作業の実施計画の作成

岡本 晃昌 (沼田 一道 助教授)

## 1. はじめに

不動産業における主な仕事は物件の売買や賃貸、仲介業などである。しかし、同様に重要な仕事として、不動産管理業務があり、これを営業項目の一つとしている会社もある。管理業務の中には定期点検作業と呼ばれる、管理物件の共有部分の清掃や点検、無断駐輪自転車の処理といったものがあり、これらの作業は実際に物件まで足を運び行わなければならない。通常、物件毎に月1回行わなければならない、時間のかかる仕事の1つである。多数の管理物件を複数の社員で処理する場合、社員にどの物件を割り当て、割り当てられた物件をどのような順番で処理するか、また巡回ルートを決めるかなどによって定期点検作業にかかる時間は変わってくる。物件の割り当てや処理順番を無計画に行くと、余計な時間をかける事になる。この作業を毎月繰り返すことを考えると、移動時間を可能な限り削減するような物件の割り当て方、物件を廻る順序の決定は重要である。

本研究では、定期点検作業を限られた人数で行うという前提の下で、各社員の公平性を考慮し定期点検作業業務にかかる時間を少なくするような、実施計画の作成法を提案し、不動産会社M社のデータに適用して検証する。

## 2. 問題の概要

本研究が想定する事例のM社では、東京都を中心として多数の管理対象物件が点在しており、限られた複数の社員が仕事の一部として定期点検作業を行っている。

なお移動手段として電車を利用している。本研究では、(1)限られた時間内で、複数の社員が複数の物件を廻る時の時間を小さくする、(2)各社員が公平になるような物件集合の分割を考えるという2つの目標を置く。(1)は配送経路問題であり、(2)は配分問題である。

この問題を扱う上での前提は以下の通りである。

- ・ 各社員は会社を出発し、会社に戻る。
- ・ 各物件は1回ずつ、1人の社員が訪問する。
- ・ 物件における点検作業時間は各物件毎に異なる。(点検作業時間に最寄駅からの移動時間を含む。)
- ・ 得られたルートに含まれる物件間の移動時間と各物件の点検作業時間の総和は、社員1人の一日の点検作業時間の上限を超えない。
- ・ 移動に要する費用(電車賃)は考えないものとする。
- ・ 移動時間における、時間帯や乗り換えのタイミングによる誤差は考えないものとする。

このような条件の下で、全社員の点検作業時間の和(=全物件の点検作業時間+全社員の移動時間)を最小にするような物件の割り当て、社員が処理する最適な順番を求める。全物件の点検作業時間は一定であるので、全社員の点検作業時間の和を最小にするには全社員の移動時間を最小にすればよい。

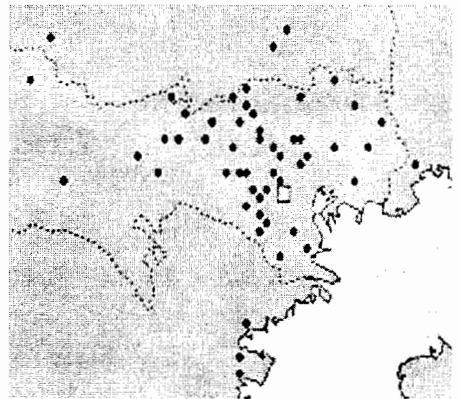


図1. 管理対象物件 (53 物件と会社)

### 3. 定式化

社員の人数を  $m$ , 会社(デポ)を含む物件数を  $n$ , 物件の集合を  $V = \{v_1, v_2, \dots, v_n\}$  ( $v_1$  は会社), 巡回路数を  $L (\geq m)$ , 巡回ルート  
の集合を  $R = \{r_1, r_2, \dots, r_L\}$ ,  $D_i$  をルート  $r_i$  に割り当てられた物件  
の集合,  $D_i$  を廻る順番を  $\sigma^{(i)}$  とする. 物件  $i$  の点検作業にかかる時  
間を  $c_i$ , 物件  $i, j$  間の移動にかかる時間を  $t_{ij}$ , 社員の一日の点検作業  
時間の上限を  $T (\leq 480[\text{分}](\text{勤務時間}))$  とする.

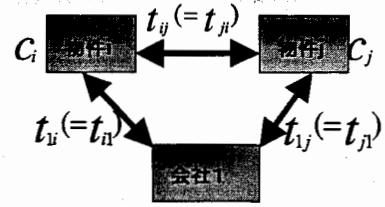


図2. 移動時間

$$\text{Minimize } z = \sum_{i,j} t_{ij} \sum_k x_{ijk} + \sum_i c_i \sum_k y_{ik} \quad (1)$$

$$\text{SubjectTo } \sum_k y_{ik} = \begin{cases} 1 & i = 2, \dots, n \\ L & i = 1 \end{cases} \quad (2)$$

$$\sum_{i,j} t_{ij} x_{ijk} + \sum_i c_i y_{ik} \leq T \quad k = 1, \dots, L \quad (3)$$

$$\sum_j x_{ijk} = \sum_j x_{jik} = y_{ik} \quad i = 1, \dots, n \quad k = 1, \dots, L \quad (4)$$

$$\sum_{i,j \in S} x_{ijk} \leq |S| - 1 \quad \text{for all } S \subseteq \{2, \dots, n\} \quad k = 1, \dots, L \quad (5)$$

$$y_{ik} \in \{0, 1\} \quad i = 1, \dots, n \quad k = 1, \dots, L$$

$$x_{ijk} \in \{0, 1\} \quad i, j = 1, \dots, n \quad k = 1, \dots, L$$

$$x_{ijk} = \begin{cases} 1 & \text{ルート } k \text{ において物件 } i \text{ の直後に物件 } j \text{ を訪れる} \\ 0 & \text{それ以外} \end{cases}$$

$$y_{ik} = \begin{cases} 1 & \text{ルート } k \text{ において物件 } i \text{ を訪れる} \\ 0 & \text{それ以外} \end{cases}$$

(1)は総点検作業時間を最小化するという目的関数である. また, 制約式(2)は, 各物件は必ずある社員が訪れることを示し, 制約式(3)において, 社員の労働可能時間制約を表す. 制約式(4)において, 社員は訪れた後, そこを去ることを示している. また, 式(5)は部分巡回路(デポ1に辺が接続していない閉路)を禁止するものである.

### 4. 解法

本研究で扱うM社の管理物件の数は 50 程度である. 物件の数をもっと少なければ全解の列挙により上記問題の厳密解を求めることができるが, 50 程度だと全列挙による求解は困難と思われる. そこで本研究では, メタ戦略の1つであるタブーサーチを用いた近似解法を提案する.

タブーサーチとは, 適当な初期解  $x$  を生成し, 現在解  $x$  の近傍を探索し, 近傍内の最良解  $x'$  を選択し,  $x'$  を次の現在解  $x$  とする. 改悪を許すことにより, 局所最適解で探索が終了してしまうことを避け, 終了条件を満たすまで探索を繰り返す. また, タブーリストを用いて近傍に含まれる解から少し前に探索した解を除くことにより直前の解に戻ってしまう循環を防いでいる. 本研究では「改悪を許し一時的に悪い解に移ってから良い解を見つける」という戦略が我々の問題に適していると考え, タブーサーチを採用した.

### 5. タブーサーチ戦略の本研究への適用

本研究において, 近傍を広くとると計算量が増大してしまうため, 近傍を図3で示すように  $v_j$  を現在の  $D_i$  から他の  $D_h (i \neq h)$  へ移動する入れ替え1回の範囲で得られる解の集合とした. 実際には近傍に含まれる解からタブー期間が残っている  $v_j$  の入れ替えで得られる解を除いたものの中で最良の解に移動する. 改悪を許し

ながら探索を繰り返しC回続けて目的関数値の更新がなければ、それ以上良い解がないとみなして終了する。解法の手順は以下の通りである。

Tabu : タブーリスト, hantei : 連続非改善回数(Cを超えたら終了),

x : 現在解, q : タブー期間

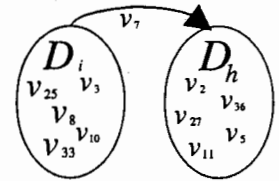


図3物件の入れ替え例

Step1. Tabu, hanteiを初期化する。  $D_i (i=1,2,\dots,L)$  にランダムに  $v_j (j=2,3,\dots,n)$  を割り当て、それを初期解  $x = \{D_1, D_2, \dots, D_L\}$  とし、準最適な  $\sigma^{(i)}$  を求めて各巡回ルートの所要時間を求め、目的関数値を計算する。それを最良目的関数値  $f^*$  とする。

Step2. Tabu [ $v_j$ ] = 0である全ての  $v_j$  を  $D_i$  から  $D_h (i \neq h)$  へ移し  $\sigma^{(i)}$  を最適化して目的関数値を計算する。それらの中で目的関数値が最小の解(値)をそれぞれ  $x', f(x')$  として、改善であっても  $x = x'$  と更新する。また、その時の移動させた物件を  $mv$  とする。

Step3.  $f(x') < f^*$  であれば  $f^* = f(x')$  とし、hantei = 0とする。そうでなければ hantei = hantei + 1 とする。

Step4. hantei > Cであれば探索を終了して結果を出力する。

そうでなければ、移動した物件を Tabu [ $mv$ ] = q, その他の物件は Tabu [ $v_j$ ] > 0ならば Tabu [ $v_j$ ] = Tabu [ $v_j$ ] - 1 と更新して Step2へ戻る。

## 6. 実験

M社のデータを用い前節で示した解法により解く。M社のデータ、物件数は54(会社含む)、定期点検作業を行う社員は5人( $n=54, m=5$ )である。巡回ルートの数や作業時間の上限を変化させ、繰り返し実験を行う。巡回ルートの数は各社員が公平になるようにするため担当社員の倍数( $L=5, 10, 15 \dots$ )とする。作業時間の上限時間  $T$  は勤務時間(480)を超えないように30分間隔( $T=480, 450, 420, \dots$ )で実行可能な限り変化させる。なお、実験プログラムはBoland社のDelphi6で作成した。

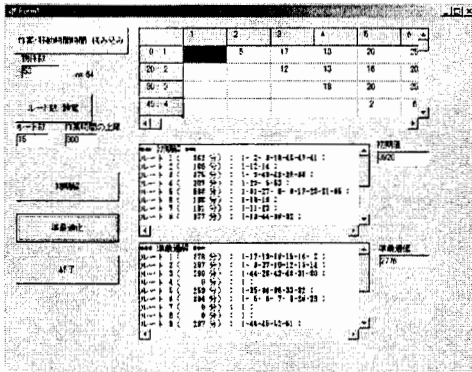


図4. 実行画面

物件	作業時間	1	2	3	4	5	6	7	8	9	10
1 渋谷(会社)	0	1	5	17	18	20	25	32	54	73	36
2 代々木	20	2	12	13	15	20	29	53	66	30	
3 西新宿	30	3		18	20	25	32	51	81	39	
4 高円寺	45	4			2	6	19	49	63	34	
5 阿佐ヶ谷	20	5				4	17	45	62	37	
6 西荻窪	25	6					9	37	57	41	
7 東小金井	35	7						33	60	56	
8 西八王子	35	8							58	65	
9 青梅	30	9								66	
10 井荻	15	10									

作業時間

移動時間

図5. 実験データ例(一部)

## 7. 結果と考察

実験の結果(総作業時間とルート数、各ルートの所要時間の上限)を表1に示した。当然であるが、各社員(各ルートの所要時間)の公平性を考慮せずに、各ルートの所要時間の上限の値を大きくすれば総作業時間は小さくなる。また、ルート数が多い場合や、各ルートの所要時間の上限に余裕があると空のルートが出現してしまうことがある。これは開始時に  $L$  を入力するが、結果的には  $L$  本未満の

ルートで作業を行う方が上限の制限を守った上で総作業時間を小さくすることができるからである。このため、実際のルート数がL本より少なくなり社員の倍数にならないことがある。これは、社員の公平性を考えると不都合である。本研究では、得られた結果のうちルート数が社員の倍数となったものが有効であるとする。有効な解となったのは、L=10の時はT=360,330、L=15の時はT=240、L=20の時はT=190（T=210では空のルートが発生し、T=180では実行不可能であった）のみであった。有効であった解に関しては、さらに各社員が公平になるようなルートの割り当てを考えた。L=10の時のT=360とT=330の結果を比較すると、T=330のときの方が各社員の作業時間のばらつきが小さく望ましい結果となった。また、ルート数が多くなると総作業時間が大きくなってしまいうので、公平性が確保されれば、ルート数は少ないほうが良い。以上のことより、ルート数が少なく各ルートの所要時間の上限が小さいL=10、T=330の時の解が良いのではないかとと思われる。

表1. 総作業時間と各ルートの所要時間の上限

T	F(L=10)	F(L=15)	F(L=20)
190			3335
210			3027
240		2955	2943
270		2838	2842
300		2746	2802
330	2716	2711	2869
360	2694	2637	2631
390	2760	2699	2596
420	2550	2620	2583
450	2580	2637	2648
480	2640	2585	2644

表2. ルートの割り当て・巡回順序

社員	物件の配分・巡回順序(L=10,T=330,F=2716)	社員の作業時間(分)
1	(1-43-42-35-34-36-33):(1-9-20-21)	589(314+275)
2	(1-10-11-12-13):(1-51-52-45-46-32)	530(201+329)
3	(1-40-22-18-19-17-15-16):(1-23-24-25-26-44)	575(314+261)
4	(1-49-48-41-39-3-27-2):(1-38-54-37-53-50-47)	577(294+283)
5	(1-8-14-7-6-5-4):(1-30-31-28-29)	445(329+116)
物件の配分・巡回順序(L=15,T=240,F=2955)		
1	(1-44-45-46):(1-8-6-5-4):(1-29-28-31-30)	584(228+240+116)
2	(1-21-20):(1-51-52):(1-35-34-36-37)	588(176+179+233)
3	(1-3-27-2):(1-40-22-18-19-17):(1-12-13-14-7)	588(124+239+225)
4	(1-49-48-26):(1-9):(1-38-54-33-32)	586(234+176+176)
5	(1-11-10-16-15):(1-39-41-53-50-47):(1-43-42-23-24-25)	609(218+218+173)
物件の配分・巡回順序(L=20,T=190,F=3335)		
1	(1-49-48):(1-17-19-15):(1-12-13-10):(1-20-21)	667(148+183+160+176)
2	(1-35-54):(1-8-7):(1-18-22-40):(1-34-36)	667(168+189+151+159)
3	(1-32-33-38-37):(1-5-6-4):(1-3-27-16-2):(1-43-42-23-24-25)	667(169+138+187+173)
4	(1-53-51):(1-26):(1-11-14):(1-45-44)	667(178+113+189+187)
5	(1-46-47-50):(1-30-31-28-29):(1-9):(1-52-41-39)	667(190+116+176+185)

## 8. まとめ

本研究では、複数の社員が複数の物件を分担して処理することを配送経路問題ととらえ、タブーサーチ法を基本とする解法を構成しプログラムを作成した。これにより、経験などに頼らない定期点検作業の実施計画を立てることができ、現状と比べても効率的であった。また、移動手段に電車を利用しており[3]により移動時間データを作成している為、所要時間は現実的で確かなものである。実験ではM社のデータをもとに行ったが、M社の管理対象物件が増加した場合や、他社の場合においても、図5のようなデータを作成することで定期点検作業の実施計画を作成することが可能である。

本研究では社員は会社を出発し会社に戻ることを前提としたが、社員が自宅から直接物件に向かう場合や、直接帰る場合などを考慮したモデルの構築が今後の課題である。また、今回は時間に注目して研究を行ったが、費用などを考慮することも重要な課題である。

## 参考文献

- [1] 柳浦睦憲, 茨木俊秀 : 「組み合わせ最適化—メタ戦略を中心として—」, 朝倉書店, 2001
- [2] 白石洋 : 「組合せ最適化アルゴリズムの最新手法」, 丸善, 1999
- [3] JRトラベルナビゲーター : <http://transit.msn.co.jp/home.asp>