

配送路決定問題における

データ依存性

沼田研究室

4496017 太田 佳孝

発表構成

1. はじめに
2. 本研究の目的
3. 配送路決定問題とは
4. 本研究で用いる解法
5. 扱うデータについて
6. 実験および結果
7. 考察

参考文献

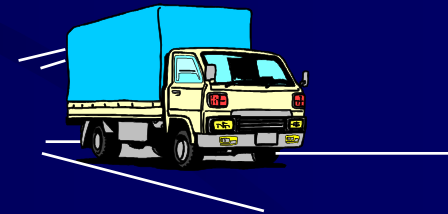
1. はじめに

..各事業体に配送業務の実際について..

インターネットなどの通信手段の発達

事務書類等の物理的交換

配送業務の費用の抑制



1. はじめに

・・配送費用を抑えるために・・

支店



グループに分け

周る順序を決める

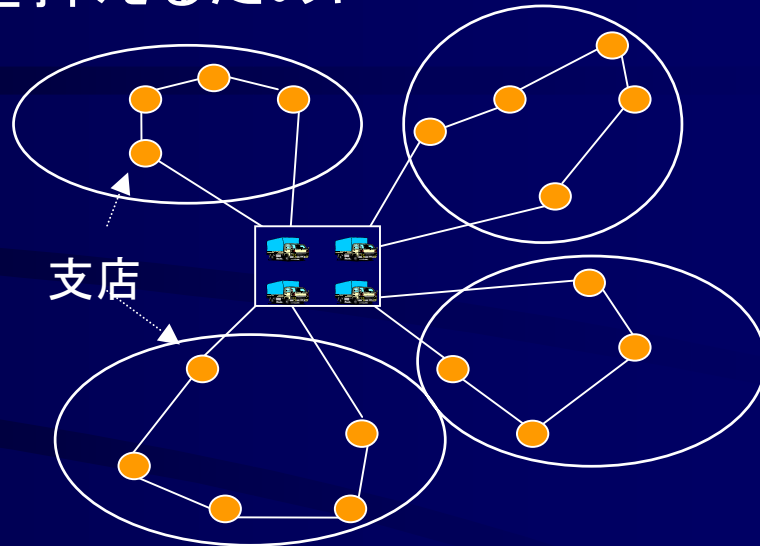


図2: 配送計画問題の概念図

走行距離の最小化

配送路決定問題 (Vehicle Routing Problem: VRP) と呼ぶ。

1. はじめに

・・配送路決定問題(VRP)を解くためには・・

道のリデータが必要

道のリデータを得るために

・・問題点1・・

・道路状況を熟知している必要がある。

・・問題点2・・

取得時間がかかり、大変である。

2. 本研究の目的

ユークリッド距離なら容易に取得できる

そこで..

道のりを推定

VRPを解く

実測の道のり

VRPを解く

両者の最適解にどの程度の違いが生ずるかを観察する。

ユークリッド距離から推定したデータは妥当か？

3. 配送路決定問題とは

1. 組み合わせ最適化問題である

2. 巡回セールスマン問題に様々な制約を加えたもの

・・・巡回セールスマン問題 (Traveling Salesman Problem: TSP) ・・・

無向グラフ $G1 = (V1, E1)$ (i, j) 間の費用 $t_{ij} (= t_{ji})$ が与えられたとき、 $G1$ の全ての点を一度だけ通過し、最少費用の巡回路を求める問題

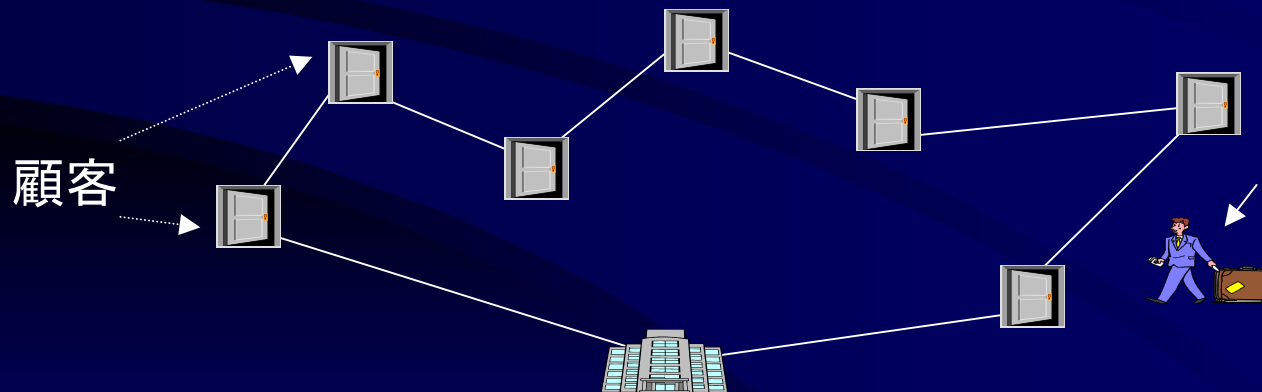


図1: TSPの概念図

3. 配送路決定問題とは

・・配送計画問題(VRP)の特徴・・

数ある組み合わせ最適化問題の中でも、特に難しく、適切なアルゴリズムの設計が求められる

組み合わせ最適化問題に対する研究が実務的に役に立つことを示すには、良い問題である

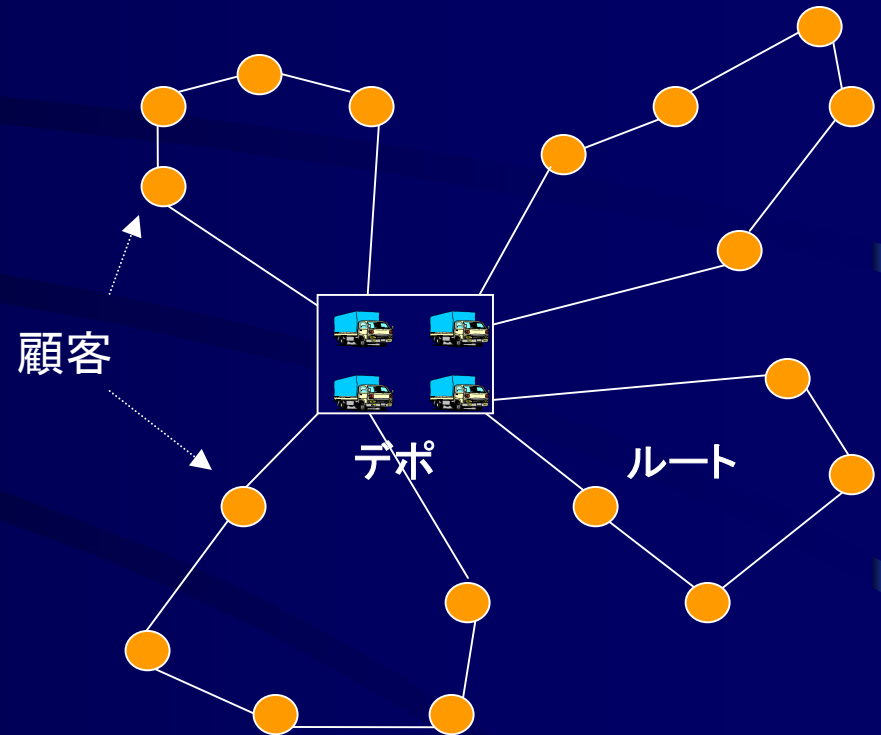


図2: 配送計画問題の概念図

3. 配送路決定問題とは

以下の仮定を持つ

- デポと呼ばれる特定の地点を出発した運搬車が、顧客を經由し再びデポに戻る。このときの顧客の通過順をルートと呼ぶ。
- デポに待機している運搬車の種類および走行可能距離は既知である。
- 顧客の位置は既知である。
- 地点間の距離は既知である。
- 各運搬車について、得られたルートに含まれる配送地点間の距離の総和は車両の走行可能上限を超えない。

配送計画問題

複数台の運搬車がデポから得意先へ配送する上で、その配送コストの総計を最少にするルートを求めることに他ならない

3. 配送路決定問題とは

・・VRPの定式化・・

走行可能上限Dをもったm台の車両があり、n個のデポを含む配送地点とij間の移動距離 $t_{ij}(=t_{ji})$ が与えられたとき

$x_{ijk}=1$ (車両kが配送地iの直後にjを訪れる),
0(それ以外)

$y_{ik}=1$ (車両kが配送地iを訪れる),
0(それ以外)

とすると

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

ルート^①の総距離を最小化

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

各配送地は必ずある車両が訪れる

3. 配送路決定問題とは

・・VRPの定式化・・

走行可能上限 D をもった m 台の車両があり、 n 個のデポを含む配送地点と ij 間の移動距離 $t_{ij}(=t_{ji})$ が与えられたとき

$$\sum_{i,j} t_{ij} x_{ijk} \leq D \quad k = 1, \dots, m, \quad (3)$$

車両の走行可能距離制約

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

車両は各配送地を訪れた後、そこを去る

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

部分巡回路(デポに辺が接続していない閉路)を禁止

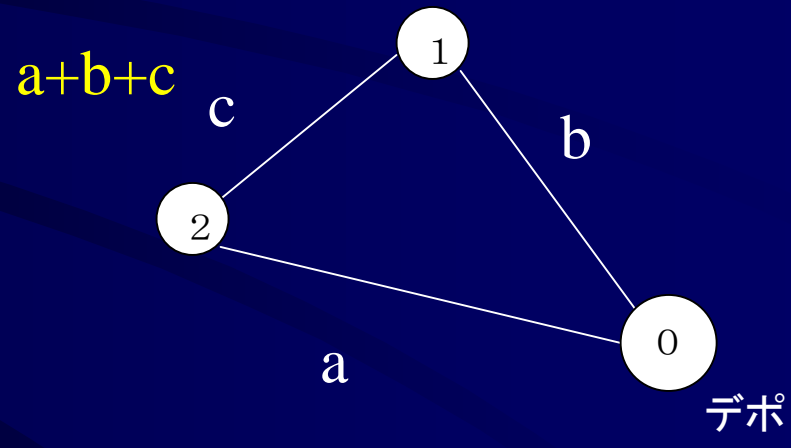
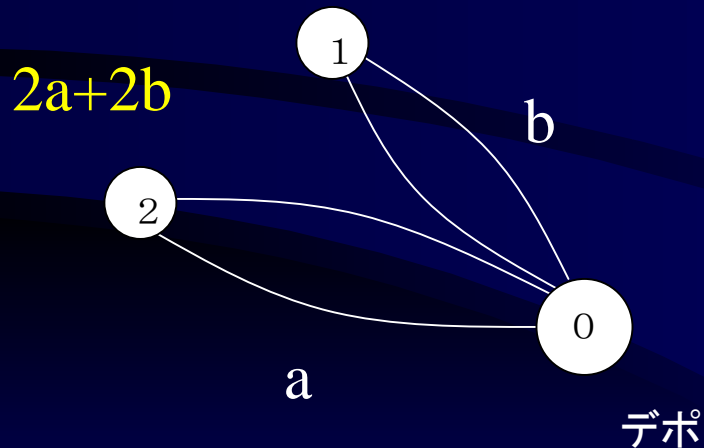
$$x_{ijk} \in \{0,1\}, \quad i, j = 1, \dots, n, \quad k = 1, \dots, m, \quad (6b)$$

$$y_{ik} \in \{0,1\}, \quad i = 1, \dots, n, \quad k = 1, \dots, m, \quad (6a)$$

4. 本研究で用いる解法

・・Saving法の適用・・

セービング値 S_{ij} という指標を導入し、車両の走行可能距離制約を満たすようにセービング値の大きい配送地の組み合わせの順に結んでいき、ルートを形成していく方法



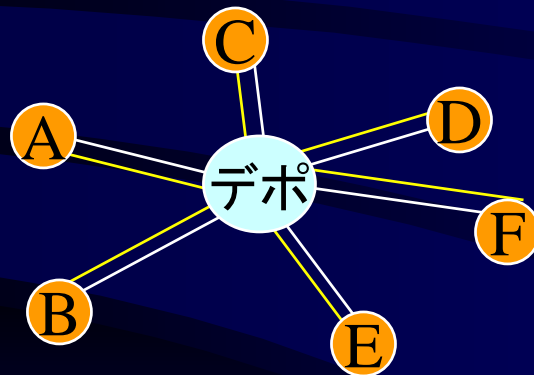
$$\text{セービング値} = 2a+2b-(a+b+c)$$

4. 本研究で用いる解法

・・Saving法のアルゴリズム・・

Step 1

初期解として、デポと各配送地を往復するルートを設定する。そして、すべての配送地の組み合わせに対するセービング値を求める。



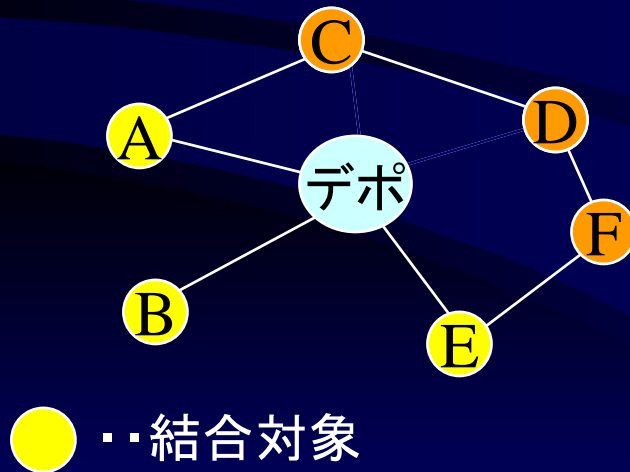
+ 全ての組み合わせに
対するセービング値

4. 本研究で用いる解法

・・Saving法のアルゴリズム・・

Step2

現在のすべてのルートにおいて直接デポにつながっている配送地を対象として、最もセービング値の大きい組み合わせ(同じルートに存在しない)のルートを結合したとき、距離制約を満足するかどうかをチェックする。



最大のセービング値 = $C-D$

← 距離制約を超えないか？

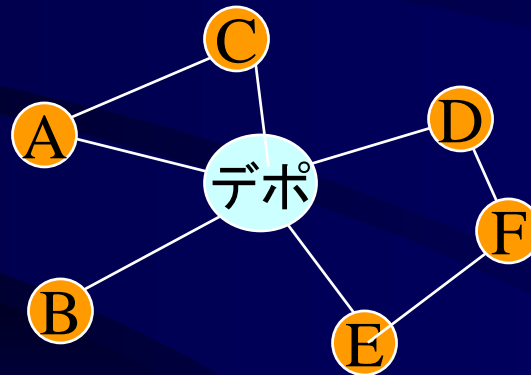
距離制約をオーバーしたら
結合しない

4. 本研究で用いる解法

・・Saving法のアルゴリズム・・

Step3

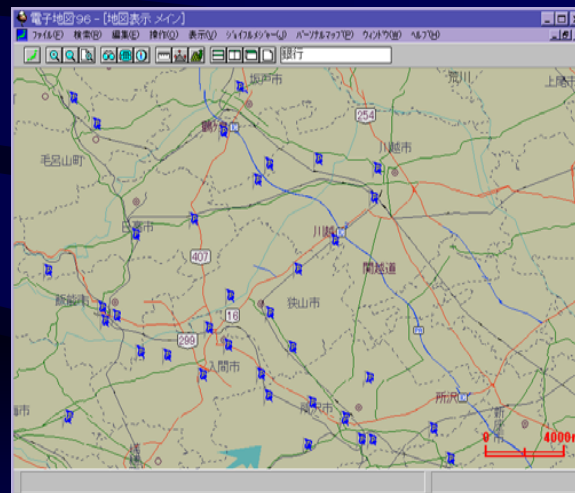
距離制約を満たすルートが存在しなくなったところで、アルゴリズムを終了。そのときのルートが解となる。



5. 扱うデータについて

- 埼玉県南西部を中心に事業を展開しているH信用金庫の全36支店データを用いた。
- 本研究では、デジタル地図上に全36支店をプロットし、マウスをクリックすることでユークリッド距離と道のりを測った。

ユークリッド距離、道のり合わせて計1406の区間全てにおいてこの作業を繰り返し、データを取得した。



拡大
→



図3. 道のり、ユークリッド距離の取得

5. 扱うデータについて

・・・ユークリッド距離を回帰直線で補正する・・・

- 本研究では、ユークリッド距離と道のりの相関を調べ、ユークリッド距離を回帰直線で補正したデータを算出する。

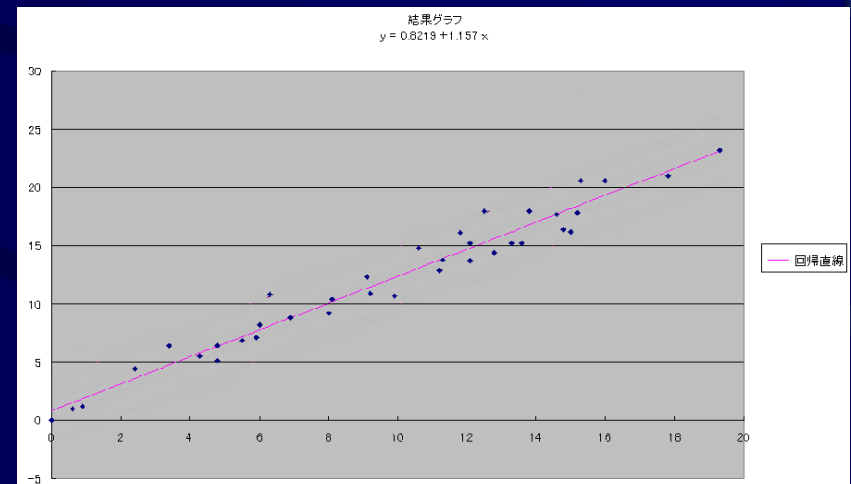
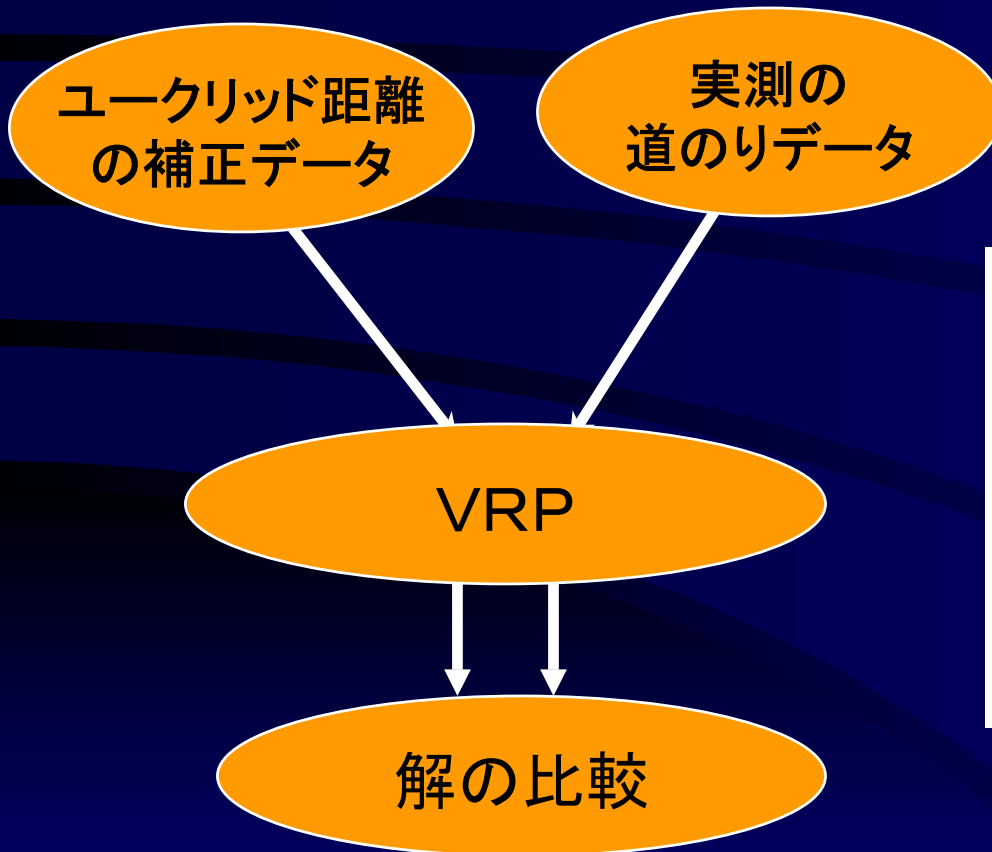


図4. 回帰直線

6. 実験および結果

・・計算機によるセービング法の実行・・

・プログラムは、Delphi3.1で作成。

一車両の走行距離制約を入力

Savingボタン

- ・総走行距離表示
- ・各ルート内の巡回順表示
- ・各ルートの距離表示

exit

終了

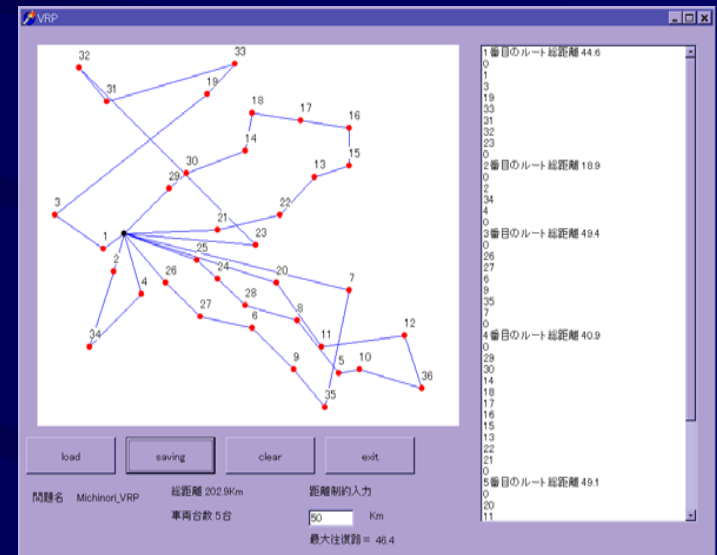


図5. 実行画面

6. 実験および結果

・・実験結果・・

•結果は、制約距離の変化に対する、車両台数、総走行距離である。

•なお、制約距離は5kmずつ、作成ルートが一つになるまで増やしていった。(制約距離140kmでルートが実測データ、推定データともに一つになった)

•ルート数と車両台数は同じである

表1. 実験結果

(制約距離) (Km)	実測データ		推定データ	
	車両台数	総運行距離(Km)	車両台数	総運行距離(Km)
なし	1	139.7	1	136.5
5	35	890.9	35	831
10	34	888.5	34	824.1
15	33	880.2	31	804.5
20	29	828.7	27	756.3
25	24	742	24	707.9
30	20	656.7	19	605.9
35	14	493.9	13	438.9
40	9	337	8	297.5
45	7	277.4	6	227.7
50	5	202.9	4	188.8
55	4	187.6	4	172.9
60	4	186.9	4	156.7
65	3	169.9	3	156.7
70	3	163.9	3	154.6
75	3	161.6	3	150.6
80	2	159.9	2	145.7
85	2	154.3	2	145.7
90	2	154.3	2	145.7
95	2	154.3	2	145.7
100	2	154.3	2	145.7
105	2	154.3	2	145.7
110	2	152.2	2	145.7
115	2	152.2	2	145.7
120	2	151.4	2	145.6
125	2	148.1	2	142.8
130	2	148.1	2	140.6
135	2	148.1	2	140.6
140	1	139.7	1	136.5

7. 考察

・・・実験を終えて・・・

- 車両台数(ルート数)においては、両者にさほどの差異は見られない。
- 制約距離の変化に対する両者の変化は同様の傾向を示した。
- 作成ルートやルート内の巡回順は両者は大きく異なった。
- それに伴い、総走行距離については、補正データで求まる解の方が、常に短かった。

以上から・・・

7. 考察

・・総括的結論・・

必要車両台数と大まかな総走行距離を知るという観点



補正したデータを用いても十分であるといえる。

実際の運行計画を立てるには



道のリデータ、さらには所要時間データが大切である。

実際の業務では、さらに「時間」に対する制約も生じてくる。

実際に車を走らせ、走行時間データを扱ってVRPを解くことが考えられる。



参考文献

1. 福島雅夫:数理計画入門

朝倉書店 1996年

2. 園川隆男、伊東謙治:生産マネジメントの手法

朝倉書店 1996年

3. 村上宣寛:“やさしいDelphi”日刊工業 1997年

4. 高橋 剛:“Delphi Object Pascal リファレンスブック” 1999年

5. 田村 直:「時刻制約及び容量制約付き配送路決定

問題—パラメータ最適化による解法」

東京理科大学工学部経営工学科卒業論文(1990)