

Uターンを行わない配達経路最適化の一方法

A Delivery Routing Problem without U-turns

流通経済大学 流通情報学部

Ryutsukeizai University

Faculty of Distribution and Logistics Systems

倉田 是

Tadashi Kurata

最近、生鮮や冷凍の食料品などを定期的に配達する企業が増加している。多くの配達箇所は住宅街にあり、そのアクセス道路の道幅は狭い場合が多く、配達車両Uターンができるほどの空き地がない。住宅地域に複数の配達箇所がある場所で、地図データを使った最短距離で配達する経路探索のアルゴリズムの作成とプログラミング開発を行った。デポと配達箇所及び配達箇所間の距離は両木探索法によって求めた。デポから出発しデポに戻る経路は、Uターン禁止であるから、配達箇所の前の道路に進入した進行方向のまま道路を進行して、次の配達箇所またはデポに進む経路である。単に次の最短距離の配達箇所を選択するのではなく、乱数による揺らぎを利用して、第2、第3の近距離の配達箇所を求める方法を採用した。これを10,000回以上繰り返して、その試行の中から最短経路を求める。このアルゴリズムを実現するプログラムを開発した。デポと50配達箇所を巡回する問題をシミュレーションして、最短経路が得られることを確認した。

Companies delivering fresh or frozen food products are becoming increasingly common recently. There are numerous points of delivery in a residential area, access roads are often narrow, and there are few vacant lots which can be used by drivers to turn their vehicles. I developed an algorithm for determining the shortest route in a residential area with many points of delivery, and incorporated it into some programs.

1. はじめに

商店が住宅区域になくなり、共稼ぎや子育て、老人世帯など買い物に不便を来すようになった。そのために、戸別に生鮮・冷凍の食料品を配達してもらう傾向が増加している。配達箇所は住宅街にあり、その前の道路は狭い場合が多く、Uターンができるほどの空き地がない。このような地域に複数の配達箇所がある住宅地域を例題として、地図データを使った巡回配達時間を最少にする経路探索問題を扱った。経路探索には、探索木の高さを限った部分木を用いて計算量を減少させて、Uターンをせずに最短時間で配達するアルゴ

リズムを考案し、そのプログラムを開発し、実行して確認した結果を報告である。

前回までの報告は⁽¹⁾、一方通行路や交差点の信号、道路の走行速度、交差点のロス時間などを考慮して、最短距離よりも最短時間を求めることであった。時間に関しては本プログラムに容易に付加できることから、今回は時間でなく距離として、アルゴリズムの構成に重点を置き、その実現のためのプログラムの開発を行ったものである。なお、この方法と手法の同一で、Uターン可能な配達経路最適化との比較して、Uターン禁止と差の少ないことを報告した⁽²⁾。

2. 本報告の特徴

本報告も最短経路探索問題の一種である。この分野では言うまでもなく、

- 巡回セールスマン問題
- カーナビゲータの経路探索問題
- 輸送計画問題

等で広くまた深く扱われている。本報告も巡回セールスマン問題と同様な巡回経路探索の分野ではあるが、実際の道路における経路探索の問題である。この点では、本報告はむしろカーナビゲータの経路探索の問題に近い。一般にカーナビゲータでは、ある地点から他の地点への2点間移動のみ扱っている。また、輸送計画問題は、主要道路を利用して多量に荷物を効率よく配送する問題であり、住宅への配達問題異なる点が多い。

本報告の上記諸問題と異なる第1の点は、道路幅員が3~4mの住宅街の狭い道路を1~2トン積みトラックで配送するので、車のすれ違いもままならない場合が多い。原則としてUターンを行なわない方が安全である。また、文献⁽³⁾によれば、交通事故の恐れがあるので、原則としてUターンを禁止している状況である。道路の広い米国においても、危険の起きやすいUターンと左折を最小化する研究がなされている⁽⁴⁾。巡回セールスマン問題では、Uターン禁止は一切考慮しない。カーナビゲータは2点間の最短経路を求めるものであるから、到着地点から次の地点への探索を行って次々と巡回問題を解くことができる。しかし、到着地点でのUターン禁止は考慮されていない。輸送計画問題は、配達地点には荷扱い用の空き地があり、Uターンが可能である。よって、他の3種の問題は本報告と異なる問題と考えられる。

第2の相違点は道路データを用いている点である。カーナビゲータも同様に道路データを用いている。すでに多くの実用機が広く用いられ、ソフトウェアパッケージも市販されている⁽⁵⁾。ナビゲータ以外の探索問題は実際

の道路データによる探索ではなく、むしろ探索のアルゴリズムを主として扱う問題である。

第3にカーナビゲータと異なる点を挙げると、探索経路が主要道路か住宅街の狭い道路かの違いが大きい。カーナビゲータでは、主要道路の探索に加えて、道路の渋滞による迂回路探索問題が多く扱われている⁽⁶⁾。

道路データを用いる長距離の経路探索では、探索データを少なくするために、探索範囲を狭く限る方策などが採られている⁽⁷⁾。本報告では、比較的狭い地域に配達箇所がまとまっているので、特別な方策を必要としない。

3. アルゴリズム

図1の地図を例題としてアルゴリズムを示す。デポ(depot)から約1km×1.4kmの地域にある、配達箇所20~50を巡る最短時間の経路を求める問題を取り扱う。交差点の数は230、交差点間の道路区間数は337である。

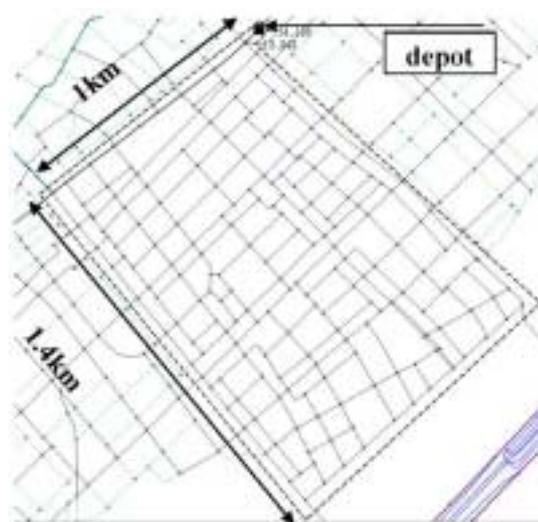


図1 例題に扱う地域と道路図

Fig.1 Map of delivery area

北側に走る幹線道路の北東の先に本来のデポがあり、往復とも図のデポと示した道路地点を通らなければならない。したがってシミュレーションで扱う地図の範囲を狭める都合上で、この地点を仮想のデポとした。

MapInfo の全国版道路データ⁽⁸⁾を利用する。道路点データは、都道府県コード・道路点ID・隣接道路点ID・接続道路数・区間(1)ID・区間(2)ID・・・等よりなる。道路区間データは区間ID・区間長(単位m)・路線番号コード・車線数コード・接続点ID・・・等よりなる。各ID(たまたまこの地域では5~6桁の整数値)はMapInfoのデータ中でユニークなものである。MapInfoの道路データのうち、利用したデータは次の2種類である。

- 道路点データ
- 道路区間データ

この2種類のデータを合体させ、表1の属性を持つ、図2に示す交差点の基本有向木を作る。木の各ノードは交差点(道路点ID)であり、木の幹の属性は区間距離である。



図2 基本有向交差点木構造

Fig.1 Structure of basic directed tree of road crossing

デポと配達箇所間の距離、及び配達箇所間の距離は両木探索法によって求めた。すなわち、デポと配達箇所の前方木と後方木を求めて重なる枝で双方の根からの距離を求めた。なお、木の枝の長さはどちらも8とした。したがって、マハッタン距離16以上の距離にある地点間の距離は求められない。この求めた距離の中、最短となる距離を最短距離表とした。なお、この前方木と後方木は基本有向交差点木構造を次々に連結することにより得られる。

次にこの最短距離表を使って、デポから出発してデポに戻るまでの最短距離となる経路

表1 基本有向交差点木構造

Table 1 Structure of basic directed tree of road crossing

項目	内容
出発交差点	ID
進路変更交差点	ID
区間距離	m
交差点分岐数	2~4
進路変更先交差点1	ID
進路変更先交差点2	ID
進路変更先交差点3	ID
進路変更先交差点4	ID

を捜し出す。Uターンを行わないために、ある配達箇所への進入方向をそのまま進出方向として、次の配達箇所を探さなければならない。このために、最短距離表には、相互間の距離とそれぞれの進行方向を記述してある。

この配達経路は、デポから出発してデポに戻らなければならない。ある配達箇所から次の最短距離にある配達箇所を探す方法を採用することも考えられるが、必ずしもデポに戻ることができない。最終配達箇所から、まっすぐに配達せずにデポに戻るという方法も考えられるが、本報告では、配達をしながらデポに戻るアルゴリズムを考えた。

一般に、巡回セールスマン問題では、配達箇所の交換を繰り返すことにより最適化をすることをやっている。本報告ではUターンを禁止することを原則としているために、次のように考えなければならない。

配達箇所の交換の際に配達箇所に着し、出発する際の道路の進行方向が同一であることから、ある配達箇所の交換した場合には、その前の配達箇所を出発する進行方向をそのまま保持する必要がある。一方、交換した配達箇所への進入経路は2方向ある。最初選択した経路と同じ方向ならば、そのまま交換することが可能であるが、もし異なっていれば、

その次の配達箇所への経路も変えなければならない。経路の変更することにより、さらに後まで変更が続く恐れもある。つまり、1箇所変えたことによる、複数の配達箇所を変更しなければならないという玉突き現象も考えられる。

したがって、単なる配達箇所の交換を進めていって、最適化を図ることは複雑なアルゴ必要とする。

本報告のアルゴリズムを説明するために、例を挙げて説明すると次のようになる。釘をたくさんに打ち込んである板があり、この釘を交差点と見なし、釘と釘の間に配達箇所が散在している。輪になったゴム紐を、端の釘をデポと見なし、そこから釘の間にある配達箇所を通るようにゴム紐を掛け回し、全ての配達箇所を巡って、距離が最短になるようにすることになる。最短距離を求めるには、ゴムひもを別の釘に掛け替えることになる。

本報告では、このゴムひもを掛ける作業にきわめて単純な方法を採用した。ある配達箇所から、次の配達箇所を探して、ゴムひもが繋がっているように、配達箇所の進入と進出の経路を同一の経路となるようにして、次々と配達箇所を探す手順の繰り返しを行う。

一度全ての配達箇所にゴム紐を掛けてから、最短経路を求めてゴム紐を掛けなおす代わりに、次の配達箇所を探すときに、最短距離の箇所ばかりでなく、第2に近い箇所、第3に近い箇所を候補に挙げて、乱数によりそのうちの1箇所を次の配達箇所として選ぶ方法を使った。このことにより、探索に揺らぎを与えて、最適解を求めることが可能となると考えた。

なおこの際に、図3に示すように、もし第2、第3の配達箇所の間に第1の配達箇所があり、第2または第3の配達箇所が選択されたときに、その経路上に第1の箇所があれば、第1の配達箇所を次の配達箇所とする。また、第3の箇所が選ばれて、その経路上に第2の

配達箇所があれば、第2の箇所を次の配達箇所とする。

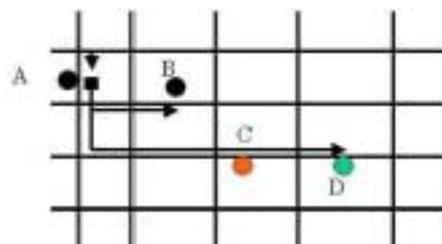


図3 次の配達箇所の選択

Fig.3 Selection of next destination

なお、Uターンも可能であり、袋小路では自動的に、Uターンをする。

図4はプログラムの流れ図である。10,000回乱数初期値を変えて試行する。各試行ので、デポに戻れない、次の配達箇所が最短距離表にないなどの理由でエラーになったものは、ファイルに格納しない。エラー率はおおよそ30%である。

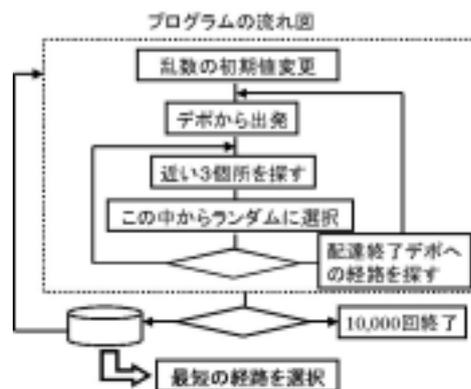


図4 処理の流れ図

Fig.4 Flow chart

プログラムに必要なデータファイルを図5に示す。これらのデータの主な役目は次のとおりである。デポ・配達箇所データは各デポと配達過疎の面する道路区間とその両端の道路点ID、それからの距離を表したもので、デポあるいは配達箇所からの出発と到着を処理中に使用する。前方木、後方木はそれぞれの

木の枝の場所を経路に記録しておき、経路を図に矢印の線で示すときに用いる役目のためである。最短距離表は、次の配達個所を探し、累積経路長を計算するためである。



図5 プログラムに必要なデータファイル
Fig.5 Data files for execution program

4. アルゴリズムのプログラミング化と結果

基本交差点木構造の作成までのアルゴリズム実現のために、プログラミング言語に Map Basic を用いた。Map Basic は独立した Basic 言語ではなく、MapInfo データ処理のためにある言語である。したがって、MapInfo からデータを引き出し、使いやすいように加工するには便利な言語である。また、最後に地図上に経路を引くプログラムもこの言語を利用した。

前方木及び後方木の作成以後のプログラミングはすべて C 言語で行った。

経路選択のプログラムの実行経過時間は 1 ~ 2 min である (CPU は Pentium 1.5GHz)。他のプログラムはほとんど即座に終了する。

配達経路を矢印線で図示するプログラムでは、線の幅、線の色を変更できるようにしてある。また、下敷きとなる道路の線 (一般に複数の座標点を使う折れ線) に重ねるために、道路区間のデータを利用している。オプションで配達個所を通過するたびに矢印線の色を変えることもできる。

配達個所 50 と 20 の最短経路を図 6 と図 7

に示す。

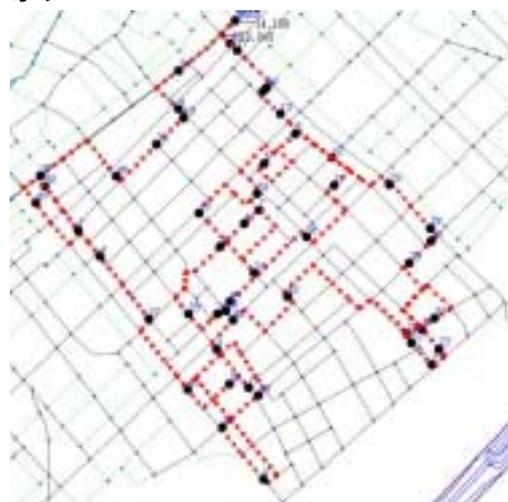
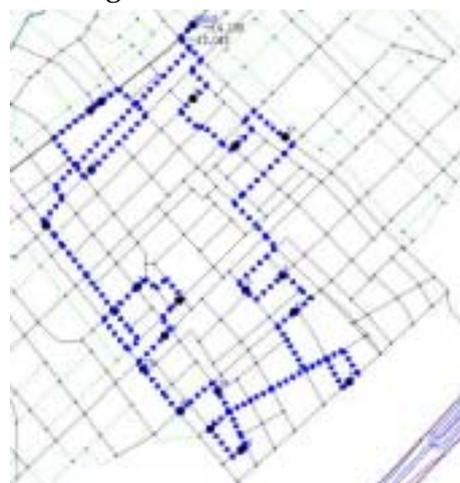


図6 最短巡回経路:配達個所=50
Fig.6 Routes of shortest route



Destinations=50

図7 最短巡回経路:配達個所=20
Fig.7 Route of shortest route
Destinations=20

5. 考察と今後の課題

MapInfo の地図データを利用した配達の最短距離経路を求めるアルゴリズムはほぼ成功したと考えられる。

本アルゴリズムを実現したプログラムは、実用にするにはさらにヒューマンインターフェースを充実しなければならない。特に、配達先を地番の指定で求められる工夫をしなけ

ればならない。

一般に、配達個所は一つのデポで 1,000 以上となり、複数の配達担当者がいる。この担当者ごとの配達経路を作成する作業は、管理者にとって簡単ではない。配送問題として、複数の経路を地図の地域割りや方角で選択するアルゴリズムが考えられているが、住宅地域では必ずしもすっきりと分けることができない場合が多い。これも残された課題である。

今回は、道路・交差点の状況に応じた走行速度、交差点のロス時間を考慮した、最適配達時間を求めなかったが、これも残された課題である。

また、中高層階への荷物の持ち上げの労働の問題も、配達担当者の健康などの面で無視できない状況にある。この問題も課題である。

6. むすび

MapInfo の地図データをもとにして、U ターンをせずに戸別配達を行う最短経路最適化問題をアルゴリズム化し、これをコンピュータのプログラミングによって実現した。

【文献】

(1) 倉田是 “ 地図データを使った戸別配達経路最適化 ”, 情報処理学会, 高度交通システム研究会, 9月(2000)

倉田是 “ 道路の渋滞に対応して経路を変更することが可能な戸別配達経路の最適化 ”, 日本シミュレーション&ゲーミング学会, 第12回全国大会, 10月(2000)

(2) 倉田是 “ U ターンをしない配達経路探索問題 ”, 日本シミュレーション&ゲーミング学会, 第13回全国大会, 10月(2001)

(3) 特集生協の総合力で実現する車両事故削減 ”, 生協運動, No.581, No.8, pp. 5-17(2000)

(4) Boldin L., Golden B., Assad A., Ball B. “Routing and Scheduling of Vehicles and

Crews”, Comput. & Ops Res., Vol.10, No.2, pp. 63-211(1983)

(5) A C T 距離計算パッケージシリーズ”, アドバンスド・コア・テクノロジー株式会社

(6) 加藤誠巳 “ 経路探索問題とその応用 ”, 情報処理, Vol.39, No.6 pp. ¥ 552-557(1998)

(7) 大西啓介, 加藤誠巳 “ 交差点内コストを考慮した道路網における経路探索の手法とそのマルチメディア型経路探索システムへの応用 ”, 情報処理学会論文誌, Vol.33, No.7, pp. 970-979(1992)

飯村伊智郎, 加藤誠巳 “ ルックアップ・テーブルにより探索領域を限定した日本全国道路網における経路探索手法 ”, 情報処理学会論文誌, Vo.35, No.12, pp. ¥ 2831-2841(1994)

杉本克行, 加藤誠巳 “ 有向ネットワークにおいて閉路を含まない k 個の最短経路を求めるための手法 ”, 情報処理学会論文誌, Vol.26, No.2, pp. 356-364(1985)

内村圭一, 神吉健一郎 “ 道路ネットワークにおける巡回配送経路探索 ”, 電気学会論文誌, Vo.114, No4, pp. ¥ 456-461(1994)

(8) “MapInfo” 三井造船システム技研株式会社