松 山 大 学 論 集
第 23 巻 第 3 号 抜 刷
2 0 1 1 年 8 月 発 行

お遍路における巡回セールスマン問題

鳥 居 鉱 太 郎

お遍路における巡回セールスマン問題

鳥　居　鉱太郎

1．は　じ　め　に

組合せ最適化問題の一つに，巡回セールスマン問題（TSP：Traveling Salesman Problem，以下 TSP）がある。TSP は NP 困難に属する問題で，ある都市を出発して全ての都市を1回ずつ訪問し，もとに戻るときの最適巡回路を求める問題である。最適巡回路は移動のコストが最少となる巡回路であり，n 個の都市から任意の出発地点を決めると，$(n-1)!$ 通りを調べれば最短経路が見つかることになる。この問題をたとえば力ずくで行う総当り法では，都市の数が増えれば計算量の問題から現実的な時間では困難であり，[1] 最適解の解法は未だ得られていない。そこでヒューリスティクスをはじめとした様々な近似算法が考案されている。都市間の移動コストは距離や費用などで表現でき，また様々な制約を加えることで，TSP は電子回路・基盤作成やネットワークの設計，運行・配送計画など多くの分野で応用可能である。

TSP は，グラフ理論におけるハミルトン閉路で最短のものを求める問題として帰着できる問題でもあり，都市間における移動コストの重みづけや都市の通過時間帯など，様々な制約を設けることもできる。さらに，複数のセールスマンを配置した組合せを求める並列（n 人）巡回セールスマン問題の取り組みも行われている。本論は，従来型の TSP およびその制約条件とは異なる問題設

1）都市の数を問題の大きさととらえれば，n! の指数関数的増加によってすぐ組合せ爆発となり，スーパーコンピューターや並列計算機をしても実用的な時間で最適解を得るのが困難である。その場合 n! は計算量でいえば指数時間のクラスであり，すぐに多項式時間では解けない時間量となる。

定の定義ならびに応用研究の可能性についての提案をするものである。出発点となる都市は TSP においては任意に定められるものであり，制約条件は都市間の順序や移動時の重みづけ（時間や荷物の重量など）で検討されている。そこで力ずく法では，最初に任意に決める一都市を踏まえて組合せの数が$(n-1)!$で求められることになる。本論で新たに問題設定するのは，閉路における最短経路ではなく，ある制約条件下における最適な出発地点を求める問題に結びつく，制約条件例の提案である。その具体例として，四国八十八ヶ所霊場巡り[2)]，いわゆる「お遍路」を用いて検討を行う。

お遍路の全工程は徳島県，高知県，愛媛県，香川県をまたいで千数百キロとなるが，多くの霊場は四国を一周する環状に位置しており，通過都市を霊場として考えた TSP では，移動ルートの制約から比較的簡単に最短経路が求まる。一般に制約条件を加えることは，最適解を求める上で計算量を減らす効果が期待でき，TSP においては移動コストとして距離のほか道路が工事中であったり時間帯によって通行禁止となったりする条件を付加することが考えられる。また，各訪問都市での休憩やご当地グルメ，お土産で立ち止まる，などの条件も考えられる。お遍路では訪問の順番は決められておらず，さらに何回かに分け分散して巡拝を終えることができる。以上をふまえると，お遍路では単なる最短経路問題の一具体例としては関心が低いものの，お遍路の特徴と独自の制約を生かした経路問題「お遍路問題（OP : Ohenro Problem）」の意義としては期待できるものがあるといえる。

本論文の構成はつぎのとおりである。TSP およびその解法について第 2 章ではこれまで行われている各種の算法ならびに応用事例を挙げ，第 3 章ではより具体的な解法として，制約条件および非対象性について述べる。第 4 章ではお遍路問題として新たな問題設定を提案する。さらにお遍路問題の応用例を示すと同時に，新たな問題として発展させる可能性を述べる。

2）弘法大師空海の旧跡を辿る巡礼にはじまる。

2．巡回セールスマン問題とその解法

TSP は組合せ最適化問題の中でもよく知られる問題で，様々な現実的な問題への応用でもその成果が期待される。たとえば各都市を接点とみなした基盤配線や数万点を扱う基盤穿孔，物流における運搬経路計画，スケジューリング，X 線結晶構造解析（タンパク質の構造解析）［1］などでコスト低減や生産性向上に役立っている。しかし同時に TSP は最適解を求めるのが困難であることでも知られ，いまだこの問題に対しての効率的な解法（アルゴリズム）は得られていない。ここで TSP の都市数が増えた際，力ずく法であらゆる通り方を調べて総コスト（移動距離）が小さいものを得ることを考える。単純に組合せの数を計算すると，n 個の都市に対してつぎの道順の数が存在することになる。n を 4，6，8，とすると，それぞれ $4!=24$，$6!=720$，$8!=40320$ となり，さらに都市が増えると，

$$10!=3628800$$
$$20!=2432902008176640000$$
$$30!=265252859812191058636308480000000$$

のとおり選択肢が幾何学的に増えて，あらゆる可能性を見つける必要から組み合わせ論的爆発となることが分かる。ここで，計算機で力ずくに最適解を求めようとした場合の計算時間例を示す。たとえば 100MIPS[3] で基本演算をするコンピューターを考えた場合，任意の出発都市を決めて $(n-1)$ 都市として n が 10 の場合は $(n-1)!=362880$ であり，約 0.0036 秒となる。しかし n が 20 になると $(n-1)!=121645100408832000$ と非常に大きくなり，約 39 年かかることになる。もちろん，高速化が進むコンピューターやスーパーコンピューター[4] ではこの計算時間が大きく短縮されていくことになるが，都市数が 100，500，

3）MIPS : million instructions per second 1 秒間に 100 万個の命令を実行できることを示す指標。

1,000以上と増えていけば組み合わせ数が極端に増えていくことになり，やはり同様に現実的な時間では解が得られなくなることは明らかである。[5]

以上からも導かれる通り，指数関数的に計算量が増えるアルゴリズムでは，最適解を求める困難さがTSPに存在することが分かる。これを計算量の観点からとらえると，指数時間アルゴリズムは効率の悪いアルゴリズムであると判断される [2]。すなわち，多項式関数と指数関数のふるまいを比較すれば，都市 n の数が一定数を超えると，指数関数の方は値が一気に増加する。これに対して多項式関数では，n の数にともなって増えていくものの，その振る舞いは指数関数ほどではなく，大きな値についても効率的な計算が期待できるのである。いま1秒間に1京回の命令実行が可能なスーパーコンピューターを考える。[6] このとき，n の計算として，多項式関数（アルゴリズム）n^{10} と，指数関数（アルゴリズム）10^n を計算量の観点から比較してみる。前者では $30^{10}/10^{16}$ ＝約0.06（秒），これに対し後者の指数関数（アルゴリズム）では $10^{30}/10^{16}$＝約 3×10^6（年）と，途方もない時間がかかることになる。

最良のアルゴリズムが見つからなくても，近似解を求めることにより，より現実的な解法で解かれている問題も多い。たとえばヒューリスティックによる手法の応用としてチェスのプログラムが挙げられる。これは経験則に基づいて理想的な戦略を近似的にルール化したものであるが，最短巡回経路を効率的に算出できる方法として知られる。ルールとして有効な知識として，次の3つが挙げられる [3]。

4）2011年に世界一の処理速度を達成した「京」は，浮動小数点演算を毎秒1京回実行（1京 FLOPS : floating point number operations per second）できる処理速度を達成している。京は10の16乗。

5）実際の応用では，数百万，数千万の接点最適化問題が出てくるVLSI設計などがある。こうした大規模問題に対しては，計算機速度がいくら向上しても，計算量が指数関数的に増える計算アルゴリズムでは実用的ではない。

6）スーパーコンピューター「京」では，合計864台のコンピューターを約20万本/1,000キロメートルのケーブルで接続。しかしこの並列度をさらに強化しても，TSPに代表されるNP完全問題を効率的に解く方法（アルゴリズム）が見つからなければ，最適解を求めるのは困難である。

⑴　盤上の駒を種類ごとに数え，自分と相手の形勢を相対的に推測する。

⑵　数手先に自分の形勢がもっとも有利になるように駒を動かす。

⑶　相手側が自分と同じ戦略を採用していると想定して，プレーする。

チェスの中盤では，平均して次の指し手の候補が約36ある［3］。そこでその先もまた36の候補という計算をしてしまうことは，やはり計算の爆発を起こして現実的ではなく，上記のルール（ヒューリスティックな知識）を用いて次の指し手を評価し，候補の数を絞り込むことで効率的な計算を実現させている。

TSPにおいても様々な近似アルゴリズムが考えられており，最短路より数％長い経路を近似解として求める工夫がある。生物の行動を模倣してヒューリスティックな手法で最短経路を見出すものとして，「アントコロニー最適化[4]」がある。これは多くの蟻が餌を探して巣へ持ち帰る際，次第にその最短経路をたどるようになる観察をもとにしている。この研究ではまず予備実験（コントロール）として，人工的な環境を用いた観察がなされた。巣と餌の間のルートを２本用意し，そのどちらも同じコスト（距離）に設定された。この環境のもとで，蟻は当初２つのルート双方を通ることが観察されたが，次第に一方の道を選択して用いるようになり，もう一方を通る個体はほとんどなくなっている。この実験結果から，同じ条件の道でも確率的にどちらか一方の道が選ばれることになり，それが最適ルートとして確定する様子が見られる。蟻は自分のたどった道に道標としてフェロモンを道に残し，巣へ戻る目印にしていると同時に，後に続く蟻もその目印を用いて餌にありつくことができる。このフェロモンは揮発性で短い時間で揮発するものであるが，より多くの蟻が通った道，またより短いルートを往復して再マーキングした道に多くの道標フェロモンが存在することになる。アントコロニー最適化では，最終的にこの２つの道から１つが選ばれる様子が，確率として次のとおり表現された［4]。

$$P_U(m)=\frac{(U_m+k)^h}{(U_m+k)^h+(L_m+k)^h} \tag{1}$$

ここで P_U は蟻の数 m において，2本のうちの上の道（Upper）を通る確率である。したがって，下の道（Lower）を通る確率は $P_L(m)=1-P_{U(m)}$ で表すことができる。U_m および L_m はそれぞれの道を通った蟻の数であり，$U_m+L_m=m$ となる。なお k，h は蟻の観察結果から，それぞれおおよそ20，2の値でフィットしている。式(1)は同じ長さの道において認められた観察結果を表現するものであるが，アントコロニー最適化ではこれを異なる長さの道にしてフェロモンの強さと距離に関するヒューリスティック情報を組みこみ，TSPに適用するものである。

3．制約条件の設定と非対称性

最適な巡回路を求める場合，そこに様々な制約条件を付加することにより，計算をより簡単にして効率的なアルゴリズムを得ることができる。また逆に制約条件を付加することにより，より困難な問題設定も可能である。後者の例として，中国人郵便配達問題がある。TSPに似た問題であるが，TSPがすべての都市を巡回して（各都市は一回のみ通る）出発点の都市に戻る問題であるのにたいし，「配達員」がすべての道を通って出発点に戻る必要がある。どの道も一回は通る必要がある（どの道にも配達すべき家がある）が，同じ道を通ってもかまわない。この問題では，配達エリアに一方通行と両方向通行の道が混在する場合は，最適解を求めるのがTSPと同様に困難となる。しかし制約条件を付加して一方通行の道しかない場合や両方向の道しかない場合では，実時間で解答が得られることが分かっている [2]。

TSPの制約条件には様々な設定が考えられるが，たとえばセールスマンが複数いて協調して全都市を回る場合のほか，移動時間や都市の通過時間，さらには都市滞在時間を決めておく方法もある。また都市間のコスト（距離など）が向き（移動方向）によって異なる場合も，非対称TSPとして対称TSPより解を求めやすくなる。これは中国人郵便配達問題において一方通行路の制約や移動コストを考えた制約によっても，同様に非対称が実現される。図1は100都市

についての最短経路例であるが，道の勾配（坂）を制約として付加したイメージを図２に示す。巡回路全体として，なるべく時間がかからないようにするには，徒歩などの場合は，上り坂をなるべく短くする条件が加わることになる。

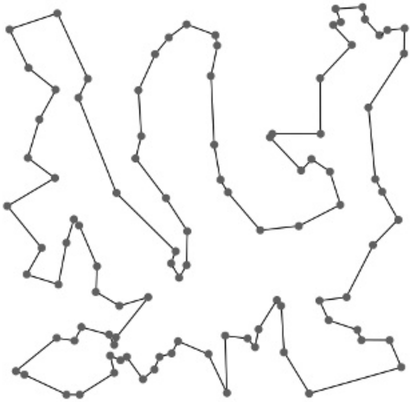

図１．100 都市の最短経路例

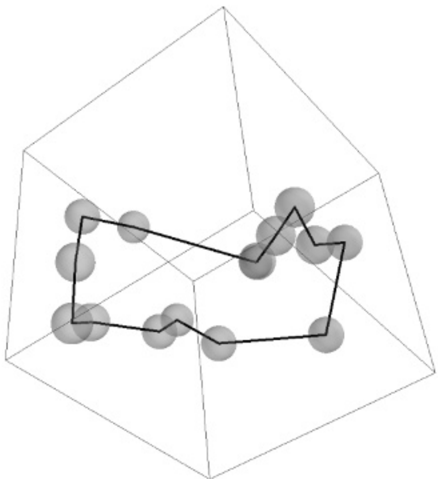

図２． 3次元で勾配を示した巡回路の例

4．お遍路における制約の設定

お遍路は四国八十八ヶ所の霊場を巡回するものであるが，その遍路道は TSP の特殊な例としてとらえることもできる。88 の霊場すべてを巡る組合せの数で考えると，出発地点を任意に決めた場合は 87! となり，つぎのとおり膨大な数であることが分かる。

87!＝2107757298379527717213600518699389595229783738061356212322972511214654115727593174080683423236414793504734471782400000000000000000000

しかし，実際のお遍路では霊場の番号順に徳島県，高知県，愛媛県，香川県の 4 県にわたって，四国の周回上に霊場が存在する。したがって，どこからでも番号順につたわっていけば最適な巡回路となるはずである。しかし，TSP としてこれをとらえた場合，制約条件の設け方によっては様々な巡回方法が考えられる。たとえば阿波国の霊場は「発心の道場」，土佐国の霊場は「修行の道場」，伊予国の霊場は「菩提の道場」，讃岐国の霊場は「涅槃の道場」と呼ばれるが，それぞれの「道場」を 5 か所ずつ訪問する（したがって，88 霊場すべてを巡回するのではなく 20 か所を巡る問題）など，すべての霊場のうちからある条件で巡回するところを選択する問題設定が考えられる。さらに次の問題(制約)設定も可能であり，これらにより TSP の実際面での応用が期待できる。

1）その土地のご当地グルメを必ず食べる。
2）その土地の有名な土産物を購入する。
3）貸自転車のあるところではそれを利用する。
4）霊場近くにバス停があるところはバスを利用する。
5）俳句甲子園や阿波踊りなどの開催時期に合わせる。
6）カマターレ讃岐や南国高知 FC などの試合を観戦する。

以上の例は，TSP における都市での滞在時間や通過時間の制約として帰着で

きるものであるが，巡回にかかわる多くの制約事例を実際面の問題から挙げていくことは，TSP 最適アルゴリズムを応用して活用するために重要である。また新たな応用研究として，たとえば複数エージェントによる協調行動をロボットによる「部品つまみあげ問題」として考えることもできる。すなわち，工場において様々な形の部品が混在する入れ物から，複数のロボットが効率的に部品をつまみあげる問題である。形状や重さ，他の部品との重なり具合をもとに，ロボットアームの初期位置からの距離とロボットアームの特性が考慮される（高速だが軽量物のみ，低速だが重量物が可能，また小さい部品のみだが小さな隙間でもつかみが可能，など)。これらは単に移動距離やエージェントの数という制約に帰着せず，より具体的な制約問題を検討してこそ応用利用可能な例であるといえる。また部品移動に限らず，ロボットを用いた瓦礫処理における最適化問題へと発展させることも可能である。以上をふまえ，より単純化した制約条件ではなく，具体例による制約付加を提案する。表 1 は四国八十八ヶ所の霊場における霊場間の，車でのおおよその移動時間（分）と距離（キロ）を示すものである。

表 1．次の霊場へ移動するためのおおよそのコスト

札所	時間	距離
第 1 番　霊山寺	5	1
第 2 番　極楽寺	10	3
第 3 番　金泉寺	15	7
第 4 番　大日寺	10	2
第 5 番　地蔵寺	15	5
第 6 番　安楽寺	5	1
第 7 番　十楽寺	10	4
第 8 番　熊谷寺	10	3
第 9 番　法輪寺	15	5
第10番　切幡寺	35	12
第11番　藤井寺	90	43

札所	時間	距離
第12番　焼山寺	60	30
第13番　大日寺	10	3
第14番　常楽寺	5	1
第15番　国分寺	10	2
第16番　観音寺	20	4
第17番　井戸寺	60	14
第18番　恩山寺	15	5
第19番　立江寺	30	14
第20番　鶴林寺	20	10
第21番　太龍寺	25	14
第22番　平等寺	30	23

第23番　薬王寺	120	75
第24番　最御崎寺	15	6
第25番　津照寺	15	5
第26番　金剛頂寺	90	33
第27番　神峯寺	60	38
第28番　大日寺	20	12
第29番　国分寺	30	11
第30番　善楽寺	30	10
第31番　竹林寺	20	8
第32番　禅師峰寺	30	11
第33番　雪蹊寺	20	8
第34番　種間寺	40	12
第35番　清滝寺	50	18
第36番　青龍寺	90	50
第37番　岩本寺	150	94
第38番　金剛福寺	120	74
第39番　延光寺	40	30
第40番　観自在寺	90	50
第41番　龍光寺	8	4
第42番　仏木寺	30	16
第43番　明石寺	120	78
第44番　大寶寺	20	13
第45番　岩屋寺	60	35
第46番　浄瑠璃寺	5	1
第47番　八坂寺	8	4
第48番　西林寺	10	3
第49番　浄土寺	5	2
第50番　繁多寺	5	1
第51番　石手寺	30	13
第52番　太山寺	10	2
第53番　円明寺	60	38
第54番　延命寺	15	4
第55番　南光坊	15	4

第56番　泰山寺	10	4
第57番　栄福寺	20	4
第58番　仙遊寺	30	8
第59番　国分寺	90	30
第60番　横峰寺	35	10
第61番　香園寺	5	2
第62番　宝寿寺	5	2
第63番　吉祥寺	10	3
第64番　前神寺	60	46
第65番　三角寺	30	23
第66番　雲辺寺	50	13
第67番　大興寺	20	10
第68番　神恵院	0	0
第69番　観音寺	10	5
第70番　本山寺	30	13
第71番　弥谷寺	10	5
第72番　曼荼羅寺	1	0.5
第73番　出釈迦寺	5	2
第74番　甲山寺	5	2
第75番　善通寺	10	5
第76番　金倉寺	10	5
第77番　道隆寺	15	8
第78番　郷照寺	15	8
第79番　天皇寺	15	7
第80番　国分寺	30	14
第81番　白峯寺	15	8
第82番　根香寺	30	15
第83番　一宮寺	40	17
第84番　屋島寺	15	8
第85番　八栗寺	15	7
第86番　志度寺	15	7
第87番　長尾寺	40	18
第88番　大窪寺	120	60

（四国八十八ヶ所霊場会の資料をもとに作成）

表１の移動時間はその札所からつぎの札所への移動にかかる時間（分）であり，たとえば第88番の大窪寺から第１番の霊前寺へは60キロで120分という車でのおおよその目安である。お遍路では番号順に回る「順打ち」と，それとは逆に回る「逆打ち」がある。場所によっては険しい山道もあり，順打ちと逆打ちでは，徒歩の場合は特に時間の差が大きくなる。この点から楽な方を選ぶのも制約の一つのパターンになり得る。さらに，通常は順打ちで回ることが多く，[7] 札所間の移動も楽なことが多い。これは順打ちに合わせた道路標識や看板が設置されているためで，逆打ちで道に迷うことも多くある。

以上をふまえてTSPの特殊な例として，ここでお遍路およびお遍路における制約条件を考慮して再度⑴式をとらえてみる。⑴式での重要性は，蟻の観察からその振る舞いをシミュレーションし，k，h として値を決められたことにある。いま k，h の代わりに時間的制約とイベント的制約をそれぞれ e，f とする。時間的制約は，たとえば上記で６つ挙げた例の制約のうち4)～6）に，イベント的制約は同1)～3）に該当すると仮定する。すると⑵式はある２地点間（霊場）の移動における２つのルートについて，より好ましい選択である通り道の確率を表現するために活用可能である。ここで m は蟻の数ではなく，お遍路に出向いた人数としてとらえれば，汎用的な最適道の解に結び付けることができる。

$$P_U(m)=\frac{(U_m+e)^f}{(U_m+e)^f+(L_m+e)^f} \tag{2}$$

5．お わ り に

本論では最適化問題における典型的な例としてTSPを挙げ，その計算アルゴリズムの困難さをふまえたうえで，その応用のためにはより具体的な事例を帰着させることなく検討する重要性を述べた。そして四国八十八ヶ所のお遍路

7）閏年の年には逆打ちのお遍路が人気となり若干増えるといわれる。

における検討例として事例を示し，TSPの特殊例として計算アルゴリズムにおける制約の要素を提案した。お遍路ではバスによるツアーやピンポイントに霊場を回るなど，必ずしも巡回という性質が保たれている訳ではないが，ロボットの「部品つまみあげ」協調作業例でもふれた通り，実際には巡回を意識せずともTSPアルゴリズムの応用範囲はかなり広いといってよい。身近な例としては，電車網を中心とした経路探索で利用者ごとの評価基準に基づいた最適な経路を探索することや，ディズニーランドなどの遊園地において，最適なアトラクション訪問ルート検索などにも応用される[5]。応用面をさらに広げれば，蟻コロニー最適化からヒントを得たことで生物学的側面でも重要な側面が出てくる。蟻コロニー最適化では，距離に関するヒューリスティック情報に道標としての「フェロモン」情報を付加したところが斬新な手法であるが，ペンローズをはじめとする量子脳理論の観点から［6,7］，脳の計算可能性について新たな知見を得ることがあるかも知れない。その発展として，脳の神経線維から構成される神経ネットワークでは，ニューロンの電気的な活動がネットワーク全般に伝播することにより，何らかの脳情報処理が行われていると考えられる。ここでTSPなどの最適化問題にたいして，ある制約条件を設けることで脳内の局所的ネットワークの振る舞いにそれを帰着できる可能性もあると考えられる。以上をふまえ，今後お遍路等を具体例にした制約条件をもとに，それを一般化した問題設定へと検討していくことが望まれる。

文　　献

[1] 山本芳嗣，久保幹雄，“巡回セールスマン問題への招待，”朝倉書店（2005）.

[2] 西野哲郎，“中国人郵便配達問題＝コンピュータサイエンス最大の問題，”講談社（1999）.

[3] ダニエル・ヒルス，倉骨彰訳，“思考するコンピュータ，”草思社（2000）.

[4] Marco Dorigo, Gianni Di Caro and Luca M. Gambardella, “Ant Algorithms for Discrete Optimization,” Journal Artificial Life, Vol. 5, Issue 2, pp. 137-172, MIT Press（1999）.

[5] 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，情報処理学会（1998）.

[6] ロジャー・ペンローズ，竹内薫・茂木健一郎訳／解説，“ペンローズの量子脳理論，”徳

間書店（1997）.
［7］守屋悦朗，“チューリングマシンと計算量の理論，”培風館（1997）.