

輸送システムに関する最適化技術

小林 和博*

Mathematical Optimization techniques in transportation system management

by

Kazuhiro KOBAYASHI

Abstract

We describe the mathematical optimization techniques in transportation system design in supply chain management. Especially, we describe the mathematical optimization techniques for tramp ship routing and scheduling, route and speed decision problem in ship navigation, and the risk management in supply chain management.

In recent years, there has been a significant improvement in the software to solve mathematical optimization problems. By utilizing the software, it becomes possible to solve planning problems with which the players in the supply chain confront. However, for nonlinear optimization problem with integer variables, it is sometimes difficult to solve large-scale planning problem. In order to overcome this difficulty, we developed some efficient computation approaches to solve these problems. In Section 2, for tramp ship routing and scheduling problem, we show the column generation approach in which the planning problem is formulated as the set partitioning problem and the mixed-integer linear optimization solver is utilized. We also show the nonlinear programming formulation for the ship navigation problem in which the ship route and the speed between the consecutive visiting ports are optimized. In Section 3, we show the mathematical optimization approach for the risk management in the supply chain management. In this approach, probabilistic aspects have been incorporated in the mathematical optimization approaches.

* 運航・物流系

原稿受付 平成27年1月30日

審査日 平成27年2月20日

目 次

1. サプライ・チェーンに	
おける数理最適化	46
1.1 数理最適化と	
サプライ・チェーン	46
1.2 数理最適化問題	46
1.2.1 線形最適化問題	47
1.2.2 非線形最適化問題	47
1.2.3 整数最適化問題	47
1.3 サプライ・チェーン	
最適化問題の例	48
1.3.1 ロジスティクス	
ネットワーク設計問題	48
1.3.2 安全在庫配置問題	49
2. 運航計画	49
2.1 船舶スケジューリング	49
2.1.1 問題設定	50
2.1.2 集合分割問題による定式化	51
2.1.3 モデリング言語による記述	52
2.2 船速および航路計画	53
2.2.1 問題設定	53
2.2.2 数理最適化モデル	54
2.2.3 航路生成アルゴリズム	55
2.3 船速を考慮した	
船舶スケジューリング	57
3. 不確実性を考慮した	
サプライ・チェーンモデル	57
3.1 リスクを考慮した	
最適化モデル	57
3.1.1 2段階の確率最適化	57
3.1.2 リスク中立化モデル	58
3.1.3 リスク回避モデル	58
3.1.4 リスクに対する対処法と戦略	59
3.1.5 意思決定レベル	59
3.2 リスクを考慮した	
最適化モデルの例	60
3.2.1 確率的在庫モデル	60
3.2.2 安全在庫配置モデル	60
4. まとめ	60
参考文献	61

1. サプライ・チェーンにおける数理最適化

輸送はサプライ・チェーンの一部である。サプライ・チェーンの最適化とは、物の流れの最適化に他ならない。サプライ・チェーンでは受発注、資材調達、在庫管理、製品発送を含んだ物の流れの構造を表す。海運を含んだ輸送は、このような一連の流れの中に位置づけて考察する必要がある。

る。したがって、輸送システムの最適化にあたっては、連携する要素、すなわち在庫管理や調達との関連性を考慮することも重要である。

数理最適化は、対象とする問題を数式を用いて表す点に特徴がある。決めたいもの（選択肢）を変数として表し、それらの変数に対していくつかの制約条件が与えられているときに、変数を用いて表現された目的関数を最大または最小にする変数の値を求める数学的方法論のことである。数理最適化は英語で **Mathematical Optimization** という。数年前までは、数理計画 (**Mathematical Programming**) とよばれていたが、**Programming** という用語はコンピュータ上での計算プログラムを作成する意に用いられることが多く、その意味に誤解されることが多かった。そこで、国際数理計画学会は、**Mathematical Programming** のかわりに **Mathematical Optimization** を用いることとした。**Mathematical Programming** と同様に、**Linear Programming**（日本語では線形計画）という語も **Linear Optimization** といわれることが多くなっている。

数理最適化は、その数式の形式によって、線形最適化、非線形最適化、など、さまざまなクラスに分類される。

1.1 数理最適化とサプライ・チェーン

数理最適化の発展はめざましく、輸送システムを含むサプライ・チェーンの効率化に寄与する（図-1）。輸送システムにおける意思決定とは、運用する船舶の隻数、寄港する港の場所・数の決定、倉庫の場所の決定など、さまざまなものがある。このように、決定したいものを変数として表し、それらに課される制約条件を数式として表現する。そして、それらの変数を用いて目的関数を最大化/最小化するような数理最適化問題として定式化できる。目的関数としてよく用いられるものに、利潤の最大化、コストの最小化、環境負荷物質排出量の最小化、などがある。

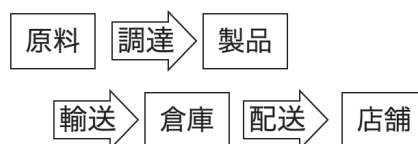


図-1 サプライ・チェーンの概念図

1.2 数理最適化問題

数理最適化問題は、実際に解きたい問題を数式として書き表すことによって最適解（数学的にそれ以上よいものがないことが保証されている解）、

またはそれに近い解（近似最適解とよぶ）を得るための問題である。数理最適化の理論では、最適解とは、上記のようにそれ以上よいものがないことが数学的に保証されている解をさし、近似最適解と明確に区別される。最適解ではなく近似最適解を用いるのは、計算時間・データ量の観点から最適解を求めることが難しいときである。これに対して実際の応用を扱った文献では最適解と近似最適解は区別せずに用いられることが多いので注意が必要である。

現実問題を数式で表すには具体的には次のようにする。まず、意思決定したいものの選択肢を変数 x で定義する。そして、実務上の制約を数式であらわし、変数 x が取り得る集合 F を定める。さらに、最大化または最小化したいものを変数 x で表した目的関数 $f(x)$ を定義する。このとき、

$$\min\{f(x)|x \in F\}$$

を数理最適化問題とよぶ。 $x \in F$ を満たす x のことを実行可能解とよぶ。数理最適化問題は、 f および F を記述する数式のタイプにより、複数のクラスに分類される（図-2）。

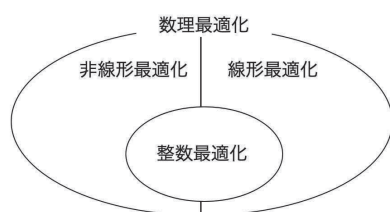


図-2 数理最適化問題の分類

1.2.1 線形最適化問題

目的関数や制約条件がすべてアフィン関数として記述されている問題を、線形最適化問題とよぶ。線形最適化問題は理論面、計算面からさまざまな研究成果が得られている。また、それらの成果を反映したソフトウェアとして、商用・非商用ともに高性能なものが入手できる。これらのソフトウェアを使えば、かなり大規模な問題であっても実

用的な時間で解くことができると考えてよい。

1.2.2 非線形最適化問題

線形最適化でない問題、すなわち、目的関数か制約条件を表す数式に線形でないものが含まれている数理最適化問題を、非線形最適化問題とよぶ。非線形最適化問題に対しては、1次の最適性条件や2次の最適性条件を用いて、勾配ベクトルなどの情報を用いて局所的な解を計算するのが標準的な方法である。反復的な求解方法として、最急降下法、ニュートン法、準ニュートン法などがよく知られている¹⁾。

1.2.3 整数最適化問題

含まれる変数が連続値をとる最適化問題は、連続最適化問題(continuous optimization problem)、離散値をとる問題は、離散最適化問題(discrete optimization problem)とよばれる。離散最適化問題は、その表現方法により、整数最適化問題(integer optimization problem)と組合せ最適化問題(combinatorial optimization problem)の2つに区別される。整数最適化問題は、整数という数値を含んだ制約式(数式)によって表現されるものである。一方で、組合せ最適化問題は、数値を用いた制約式による表現ではなく、その問題の組合せ構造によって表現されるものである。ここで、組合せ構造は適切な定式化によって整数を用いた数式表現が可能であり、逆に整数を用いた制約式を満たす点は組合せ構造として表現できるので、両者は同値であるとみなしてよい。

変数の一部が整数のみを値にとるという制約を課した数理最適化問題を、実数変数と整数変数が混じっているという意味で、混合整数最適化問題(mixed-integer optimization problem)とよぶ。一般の整数最適化問題を解くことはいわゆる「組合せ爆発」のために難しい。ところが、目的関数と制約関数が線形である整数線形最適化問題(integer linear optimization problem)に対しては、効率的なソフトウェアが提供されている(ソルバとよぶ)。これらのソルバは分枝限定法を基礎とし、問題の構造を検出・活用しながら可能な限り計算を効率的に行う工夫が取り入れられている。問題の構造によっては、非常に長い時間がかかる可能性はあるが、定式化を工夫することによってサプライ・チェーンの実務規模のサイズの問題であっても実用的な時間で解けることが期待できる。

最近は、目的関数、制約条件に非線形な数式が含まれている混合整数最適化問題 (mixed-integer nonlinear optimization problem, MINLP) を扱うことのできるソフトウェアも登場してきた。ただし発展途上の段階であり、整数線形最適化問題ほどには安定した動作は期待できない。

整数最適化問題 (組合せ最適化問題) の解を得るために頻繁に用いられる方法のひとつに、遺伝アルゴリズムがある。線形最適化、非線形最適化が問題のクラスを表していたのに対して、遺伝アルゴリズムは数理最適化問題の近似最適解を求める計算手順のことを指す。遺伝アルゴリズムはもともと生物進化のシミュレーションのために起こったものであり、さまざまな観点からみることができるが、ここでは近似最適解を求める一手法としてみる。これは、メタ解法とよばれる戦略の中のひとつと位置づけることができる²⁾。遺伝アルゴリズムは、あらゆるクラスの数理最適化問題に適用することができる。しかし、性能を出す、すなわち現実的な時間で十分によい質の近似最適解を得るには、職人芸が必要である。そして、効率のよい解法がすでに知られている問題に対して遺伝アルゴリズムを開発・適用することは、望ましくない。たとえば、線形最適化問題に遺伝アルゴリズムを適用して近似最適解を求めることはできるが、迂遠な方法である。素直に線形最適化ソルバを適用することで、厳密な最適解を圧倒的に速く得ることができる。

遺伝アルゴリズムを最適化のツールとして用いる場合は、局所探索法と組み合わせて用いられることが多い。局所探索を基礎として数理最適化、とくに組合せ最適化問題の近似最適解を得る計算方法はメタ解法と総称される。メタ解法には GRASP 法、反復局所探索法、アント法、アニーリング法、タブー探索法、など、独自の名称をもった計算方法が非常に多くある²⁾。したがって遺伝アルゴリズム自体の研究ではなく問題を解くことが目的であるならば、広くいろんな方法を候補とすることが合理的である。

整数最適化の場合、実行可能解 (制約条件を満たす変数の値) の数は有限個であり、原理的には列挙することができる。しかし、実行可能解の数は陽に列挙するには数が多すぎる。そこで、これらを全て調べるのではなく、よい目的関数値を与えそうな部分だけを調べようとするのがメタ解法の基本的な考

え方である (図-3)。

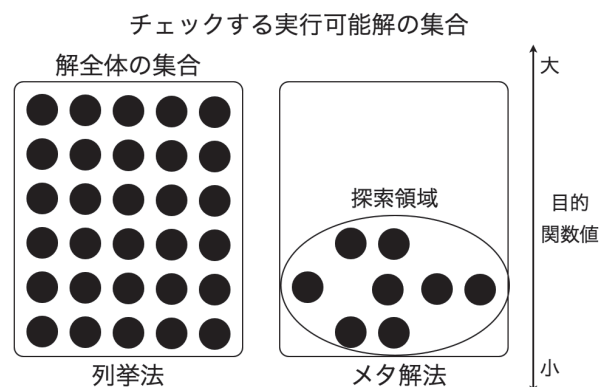


図-3 メタ解法での探索領域の概念

1.3 サプライ・チェーン最適化問題の例

サプライ・チェーンの最適化のためには、その各要素に対して様々な最適化問題を考慮する必要がある。

サプライ・チェーンにおける意思決定は、期間や影響範囲によって分類することが便利である。古くから経営における意思決定で用いられている分類にならい、ストラテジック (strategic, 戦略的) レベル、タクティカル (tactical, 戦術的) レベル、オペレーショナル (operational, 運用的) レベルに分類する。ストラテジックレベルは長期 (1年から数年/数十年)、タクティカルレベルは中期 (1週間から数ヶ月、もしくは数年)、オペレーショナルレベルは短期 (リアルタイムから日、週) の意思決定を表す (図-4)。

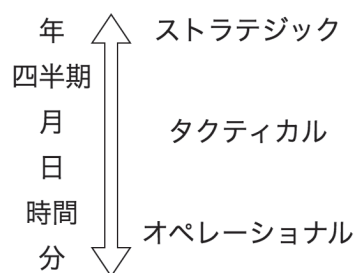


図-4 サプライ・チェーンにおける意思決定レベル

ここでは輸送最適化に限らず、サプライ・チェーンにおけるいくつかの最適化問題をあげる。

1.3.1 ロジスティクスネットワーク設計問題

一般にサプライ・チェーンは、物の供給地点か

ら需要地点までの流れを表す。ロジスティクスネットワーク設計問題は、中・長期的視点に立って、サプライ・チェーン全体における物の流れを最適化する施設配置とそれらの間の輸送構造を定めるための問題である(図-5)。これは、工場や倉庫などの物の流れの起点/通過点/終点となるものを「ノード」として、それらの間の接続関係を「アーク」として表し、物は「アーク」上を移動するとしてモデル化するとよい。このようなノードとアークを用いたモデルは、グラフまたはネットワークと呼ばれる。このモデルでは、物の空間上の移動は実数変数で表される。このように、モデル上の物の移動は、「フロー」とよばれる。物の輸送には手段が必要であるが、その調達のために固定費用が必要とするモデルでは、整数を用いることとなる。したがって、混合整数最適化問題となり、求解可能な問題の大きさが制限される。これを克服するには、データの集約などのモデル化の工夫によって解けるサイズの問題に帰着する必要がある。一方で、固定費用をモデル化に含まない問題は、フローを表す変数を実数変数としてモデル化することにより、線形最適化問題として定めることができる。したがって、超大規模な問題でも容易に最適解を求めることができる。

数理最適化モデルで求めることができるのは、どこから原材料/部品を調達するか、どの工場のどの生産ラインで生産するか、どの地点からどの地点にどのような輸送モードで輸送を行うか、どこに工場/倉庫を新設するかなどがある。工場の設置、生産ラインの設置、輸送システムの設置などは数年、数十年にわたって影響を及ぼすものである。で、ストラテジックレベルの意思決定である。

また、ロジスティクスネットワーク設計の最適化では、いつ、どこから原材料を調達するか、どの工場のどの生産ラインでどれだけ生産を行うか、どのようなタイミングで倉庫を賃貸するか、などのタクティカルレベルの意思決定にも用いることができる。

1.3.2 安全在庫配置問題

在庫をどこに配置するかは、サプライ・チェーン全体を設計する際に重要な意思決定である。需要のスピードが一定でない場合、輸送や生産の頻度がまちまちになる。安全在庫とは、需要の不確実性に対処するために余分に保持する在庫のことであり、特に需要の不確実性が正規分布に従うと仮定して用いられることが多い。一般に、予測とは「必ず」といっていいほど外れる。これはサプライ・チェーンにおける将来の顧客需要の予測についてもいえる。そのため、品切れをなくし、かつ顧客のニーズに迅速に対応するためには、あ

る程度の在庫を抱えておく必要がある。安全在庫の意思決定にあたっては、顧客サービスと在庫コストとのトレードオフを考える必要がある。このようなトレードオフ関係を数式として適切に記述することにより、定めた制約と目的のもとで最もよい意思決定を得ることができる。

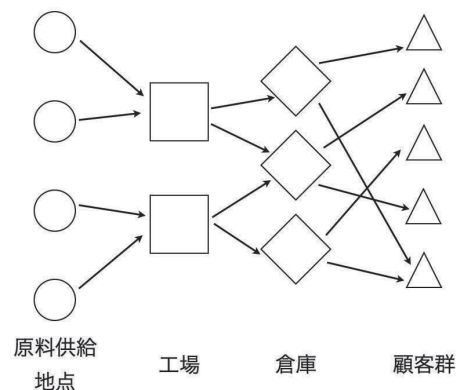


図-5 ロジスティクスネットワーク設計の概念
図

2. 運航計画

サプライ・チェーンを構成する要素の中から、輸送システムについての具体的な数理最適化モデルを述べる。輸送システムの最適化においても、ストラテジックレベル、タクティカルレベル、オペレーショナルレベルの意思決定を区別して扱うことが有用である。海上輸送におけるストラテジックレベルの意思決定として代表的なものとしては、新船の建造計画、新港の設置計画などがある。タクティカルレベルの意思決定としては、定期船航路の計画、用船の計画などがある。オペレーショナルレベルの意思決定としては、不定期船の配船計画、港における在庫管理などがある。

2.1 船舶スケジューリング

サプライ・チェーンにおいては、部品や原料などを地点間で効率的に輸送する必要がある。このとき、大量に安価に運ぶには船舶を用いることが有用である。ここでは船舶による貨物の輸送のうち、不定期船からなる船隊で貨物の輸送を効率的に行う計画を作成する問題を扱う。これはオペレーショナルレベルの意思決定ととらえることができる。国内の実務では、配船計画と呼ばれることが多

い。

2.1.1 問題設定

船舶スケジューリング(Ship routing and scheduling)は、荷主から輸送依頼を受けた貨物を効率的に輸送するために、各船舶の処理する貨物および寄港順序を定める問題である。これは、オペレータが直面する問題である。

積荷、積日、揚荷、揚日、量によって指定される(表-1)。

表-1 輸送依頼の例

ID	積港	積日	揚港	揚日	量
C1	東京	1	大阪	3	300
C2	名古屋	4	福岡	6	900

輸送する貨物は、積港、積日、揚港、揚日、量によって指定される。貨物の種類により輸送形態が異なる場合は、貨物の種類も明示する。貨物は積日に積港で積み、揚港まで運び、揚日までに揚げる必要がある。このような時間の制限のことを、時間枠制約とよぶ。積荷の種類によって、船舶の積載条件は個別に扱う必要がある。たとえば、バラ積貨物であれば積荷の総量が船舶の積載容量より大きくなければよいが、石油製品であれば異なる製品は互いに分離されたコンパートメントへ独占的に積む必要がある。

船舶スケジューリングはオペレータの解くべき問題と考えるとよい。すなわち、複数の荷主から輸送依頼を受けた複数の貨物を、自らの運航する複数の船舶からなら船隊でできるだけ効率的に運ぶ運航計画を作成する問題である。各船の運航計画は運航上のルール・制約条件を満たす必要があり、そのなかでできるだけ目的にあった計画がほしい。不定期船の運航計画は比較的短期(数日から数週間)であるため、長期の固定費用、すなわち用船費、保険料、船員費はモデル化のコストには含めないのが通常である。

この船舶スケジューリングは、数理的な構造は配送計画問題(Vehicle Routing Problem)と同じである³⁾。配送計画問題は、組合せ最適化において最もよく研究されているもののひとつである。1980年代から理論的な研究が進んでいるとともに、実務的な問題を解くための計算技術・ソフトウェアの開発が進んでいる。

船舶スケジューリングの数理最適化モデルを構築するには、配送計画問題で得られた豊かな研究成果を用いることができる。ただし、この際には陸上輸送と海上輸送の違いに注意する必要がある⁴⁾。たとえば、陸上輸送で用いられる運搬車(vehicle)は、互いに性能が同一であると考える場

合が多い。というのも、トラックはある規格にしたがって大量生産されるからである。このような同一性能の運搬車を用いる問題を、等質(homogeneous)な問題という。これに対して、海上輸送で用いられる運搬車、すなわち船舶は、性能が互いに少しずつ違っていると考える。このように同一でない性能の運搬車を用いる問題を、非等質(nonhomogeneous)な問題という。さらに陸上輸送では数百台規模の多数の運搬車を用いる場合があるのに対し、海上輸送では、日本国内の問題に限ると用いられる運搬車(船舶)はせいぜい百隻である。

また、陸上輸送では計画期間の最初に「デポ」とよばれる拠点で貨物を積載したあと各顧客地点を訪問し、その後再びデポにもどってくる計画を作成することが多い。これに対して海上輸送ではデポに相当する拠点は設定しない場合が多い。すなわち、ある計画期間内での最後の貨物を揚げた船舶は、特定の場所に移動するのではなくその場にとどまり、次に計画期間での最初の積港に直接移動するとする。

このような運用上のちがいをよく意識して用いる手法・モデルを選べば、配送計画問題に対して得られた豊富な研究成果を用いることによって、船舶スケジューリングに対して有効な数理最適化モデルを導くことができる。

ここで、船舶スケジューリングに対して提案されてきた数理的手法のひとつを述べる。まず、混合整数線形最適化モデルが挙げられる。制約条件や目的関数によって様々なバリエーションがあるが、典型的には港 i の訪問直後に港 j を訪問するとき 1、それ以外るとき 0 をとる 0-1 変数 x_{ij} を用意し、それ以外の運航に関する様々の情報(例: 港 i への到着時刻)を連続変数として定式化する⁴⁾。このアプローチは原理的にはソルバの入力データ形式に沿ってデータを用意さえすれば答えを得ることができる利点がある。しかし実際には、変数の数や制約式の数が計算機に格納できないほどに膨大になってしまうので、陽に入力データを用意することはできない。そこで、分枝限定法の途中の計算過程で問題の構造を利用したさまざまな工夫を施すことによって最適解または近似最適解を得る工夫が開発されている。

もうひとつの方法は、数理最適化モデル、すなわち問題を数式として表現するのではなく、局所探索に基づいて近似最適解を直接求める方法である。これはメタ解法と総称される。しばしば目にする遺伝アルゴリズムもこのメタ解法のひとつである。メタ解法の多くは局所探索に基づいている。局所探索とは、現在得られている実行可能解

$x_1 \in F$ に若干の変更を加える(近傍操作とよぶ)

ことで得られる他の実行可能解 $x_2 \in F$ とを比べ、

$f(x_2) < f(x_1)$ であれば x_2 に移動する、という

操作を繰り返すものである(最小化の場合)。この操作はやがて自分の周りに自分よりも目的関数が小さい実行可能解が見つからなくなった段階で終わる。つまり、局所的な最適解にたどりついて終わる。こうして見つかる局所最適解は一般には全体での最適解(大域最適解)からはほど遠いため、局所探索にさまざまな工夫をほどこしてできるだけ大域最適解に近い実行可能解を得ようとするのがメタ解法の考え方である。

万能のメタ解法は存在せず、問題ごとに個別の手順を設計する必要があるが、汎用的に有効と考えられる方法はある。例としては、反復局所探索法、タブー探索法などが挙げられる。

2.1.2 集合分割問題による定式化

船舶スケジューリングに対する強力な数理最適化モデルとして、集合分割問題による定式化が広く知られている。集合分割問題とは、代表的な組合せ最適化問題のひとつで、広く研究されている⁵⁾。この問題の入力として与えられるのは、集合と、その各部分集合に対する費用である。この与えられた集合をその部分集合に重複なく漏れなく分割するのであるが、その際、選ばれた部分集合の費用の和を最小にすることを目的とする。集合分割問題は整数線形最適化問題として定式化することができるが、その制約行列の要素は全て0または1、さらに制約式の右辺のベクトルの要素は全て1という著しい性質をもつ。多くのソルバにはこれらの性質を生かして効率的に解を求める仕組みが備えられている。したがって、大規模な問題でも実用的な時間で解くことができると考えてよい。

船舶スケジューリングを集合分割問題として定式化する際の考え方を述べる。船舶スケジューリングでは、問題の入力として荷主から引き受けた複数の輸送依頼が与えられる。これを、分割すべき集合とみなす。これらの輸送依頼を適切に分割し、各船舶に割り当てる、と考えるのである。そして、各船舶は割り当てられた輸送依頼を処理する。そして、割り当てられた依頼の処理には港の訪問順序もあわせて決める必要がある。集合分割問題の言葉でいえば、各船舶に割り当てられた輸送依頼が「部分集合」にあたり、船舶に割り当てられた輸送依頼を順に処理するのにかかる運航費用が「部分集合に対する費用」である(図-6)。

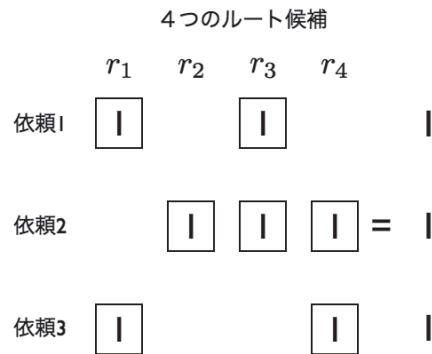


図-6 集合分割問題による定式化の概念図

集合分割問題への定式化を述べる際に必要なのが、ルート候補(または実行可能ルート)とよぶものである。これは、ある船舶に割り当てられる輸送依頼とその処理の為の寄港順序を表す。これは、運航上の様々な制約条件を満たしたものではなくてはならない。

これを集合分割問題の制約式として表すには、制約行列の行 i に輸送依頼 i を、列 j に船舶のルート候補 j を対応させる。そして、ルート候補 j で輸送依頼 i を処理するときには定数1、しないときには定数0で制約行列の要素 a_{ij} を定義する(図-6)。

ルート候補 j に対しては0-1変数 x_j を定義するが、これはルート候補 j を採用するとき1、採用しないとき0となる。目的関数は、 $c_j x_j$ の和として定義される。ここで、 c_j はルート候補 j の運航

コストである。制約行列の第 j 行の要素(0か1か)をみると、そのルート候補で処理する輸送依頼はわかるが、そのルートの運航に必要な情報(例:各港への訪問時刻)はわからない。このようルート候補に付随する情報は、別途保持しておく。船舶スケジューリングは、このようにして集合分割問題として定式化できる。集合分割問題の整数線形最適化問題としての定式化は、次のように表される。

$$\begin{aligned}
& \min \sum_{j \in R} c_j x_j \\
& \text{subject to } \sum_{j \in R} a_{ij} x_j = 1 \quad \forall i \in C \\
& \quad x_j \in \{0,1\} \quad \forall j \in R
\end{aligned}$$

ここで、 R はルート候補の集合、 C は輸送依頼の集合を表す。

集合分割問題としての定式化を記述するには、各船舶のルート候補を全て生成する必要がある。実務規模の問題だと、このルート候補の総数が膨大なものになる。具体的には、貨物が n 個あるとすると、その順列のなかから運航上の制約条件を満たしたものがその総数になる。制約条件がある程度厳しい場合でも、 n が数十になれば膨大な数になり、行列の要素として陽に入力データとして用いることはできない。

この課題を克服する方法として、全ての列を陽に生成するのではなく、効率の良い、すなわち最適解に含まれそうなルート候補のみを反復的に生成する計算方法がある。これを、列生成法⁶⁾とよぶ。集合分割問題における制約行列の列を生成するからこの名前である。理論的な保証はないものの、経験的には、実務規模の問題でもせいぜい 1000 程度の列を生成すれば十分である。

列を生成する方法はいろいろなものが考えられる。ひとつには、熟練者が輸送依頼のデータと各船舶の位置を見ながら 1 つずつルート候補を入力していく方法が挙げられる。この方法を用いると熟練者のスキルの高さにより非常によい計画が得られるだろう。しかし、作業量は多く、制約条件のチェック過程でミスがでないとも限らない。

計算機を用いて列を生成する方法として、ルート候補をネットワーク上の s - t パスとして表現する方法が挙げられる。実務のネットワークによる表現は、船舶スケジューリングに限らずサプライ・チェーンにおける最適化モデルでよく用いられ、さまざまなモデル化手法が提案されている。ここで、ネットワークとは、ノード（点）と、それらをつなぐアーク（向きのある枝）からなる数学モデルである。ネットワークを定義するノード集合の中に、2 つの特別な点を定義する。1 つは始点 s で、もう一つは終点 t である。ネットワーク上で、 s から始めてアークを順に辿っていき t に至るノードとアークの並びを、 s - t パスという。各アークに通過するためのコストを定義し、含まれるアークのコストの和が最小となるような 2 点

間のパスを求める問題は最短路問題と呼ばれ、ネットワーク上の最適化問題として広く知られている。大学初年度のデータ構造とアルゴリズムの講義で学ぶことも多い。船舶のルート候補をネットワーク上の s - t パスで表すには次のようにする。まず、各輸送依頼に対してノードを定義する。そして、ノード i を処理した直後にノード j を処理することが可能なとき、 i から j へのアークを定義する。始点 s はその計画期間内の船舶の初期状態を表し、終点 t は同じく最終状態を表す。こうすると、ネットワーク上の s - t パスは、その計画期間内の運航ルート（候補）を表す。アーク (i, j) のコストとしては、 i の揚げ港から j の積港までの移動コストを定義すればよい（図-7）。

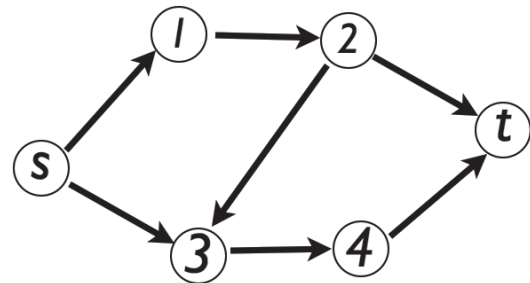


図-7 運航ルートを表すネットワーク

これらの変数を用いて、ネットワーク上でコストが最小になる s - t パスを求める数理最適化問題を定義する。これは、「制約付き最短路問題」として定式化することができる。ここで制約付きにするのは、運航上の制約条件を表現するためである。制約付き最短路問題は、最短路問題と異なりダイクストラ法のような効率的な計算方法は一般には知られていない。しかし、整数線形最適化問題として定式化できるため、ソルバを用いて解くことができる。特に、船舶スケジューリングにおいては、制約条件が厳しいために短い実行時間で解を得ることが期待できる。こうしてコストの小さい s - t パスを求めて、それらを集合分割問題の制約行列の列として加える。この行程を繰り返し、「被約費用」が負の s - t パスがなくなったら列生成を終了する⁷⁾。

2.1.3 モデリング言語による記述

整数線形計画問題を計算機プログラムへの入力として記述するには、「数理最適化モデリング言語」による記述が便利である。途中で複雑な処理をする場合は C 言語や Python 内部で用いることのできる API を用いることも多いが、ここでは簡単

に手軽に用いることのできる、モデリング言語による記述方法を述べる。モデリング言語を用いると、人間に読みやすい形式で数理最適化問題を書くことができる。そして、それを数理最適化ソフトウェアの入力形式に変換する仕組みも整備されている。モデリング言語にはさまざまなものがあるが、ここではZuse Institute Berlinの提供するZIMPLによって説明する⁸⁾。モデリング言語によって多少の書き方のちがいがあがあるが、基本的な考え方は同じである。

モデリング言語では、変数、制約式、集合などを、数式に近い人間に読みやすい形で記述することができる。例えば、集合Cの要素をその部分集合に分割する集合分割問題をモデリング言語で記述するには、次のようにする。まず、集合Cを定義する。例として、CはC1、C2、C3、C4、C5の5つの要素からなるとする。Cの部分集合としてS0、S1、S2の3つを考え、要素S0、S1、S2からなる集合をSとする。Sの定義は、次のように書くことができる。

```
set S:={"S0", "S1", "S2"};
```

さらに、Sの各要素に対して0-1変数xを定義するには、次のように書く。

```
var x[S] binary;
```

集合Sの各要素に対してその費用cを定義するには、次のように書く。

```
param c[S]:=
    <"S0">10,<"S1">20,<"S2">15;
```

目的関数を、選ばれた要素の費用の和の最小化として定義するには、次のように書く。

```
minimize cost:
    sum <j> in S : c[j]*x[j];
```

ここで、costは目的関数につける名前であって、どのようなものでもよい。Sの各要素にCのどの要素が含まれるかを表す上述の a_{ij} を、パラメータ $a[i, j]$ として定義しておく。Cの要素が選ばれた何れかの部分集合に含まれるという制約条件は、次のように書く。

```
subto assign:
    forall <i> in C do
```

```
    sum<i, j> in C cross S :
        a[i, j]*x[j] == 1;
```

こうしてZIMPLのモデルとして記述できたら、それを数理最適化ソルバの入力形式に変換してくれる処理系に渡す。その結果を入力として数理最適化ソルバを実行すると、集合分割問題の解が得られる。

2.2 船速および航路計画

2.2.1 問題設定

輸送システム最適化におけるオペレーショナルレベルの意思決定として、船速および航路計画を取り上げる。先に述べた船舶スケジューリングでは、港の訪問順序とその訪問時刻を計画するのに対して、この船速及び航路計画では、ある港から他の港に運航する際の航路と船速を計画するものである。物理的な最短距離で航行しようすると大圏航路を辿ればおおむねよいが、ここでは航行で用いる燃料消費量を最小化するような船速と航路計画を求める数理最適化モデルを述べる。

ここでは、対地船速 v で運航する船舶の単位時間あたりの燃料消費量は、船速の n 乗に比例すると仮定する。この燃料消費量を船速 v の関数として $f(v)$ と表すことにする。 n は荷の種類、船舶の仕様、気象・海象によって定まる。

対地船速 v で距離 d を移動する時間は d/v であるから距離 d を対地船速 v で運航するときの船舶の燃料消費量は、 $f(v) \times (d/v)$ で与えられる。

いま、距離 d を移動する際に、気象・海象その他の影響により、対地船速が r だけ低下することがわかっているとすると、このとき、距離 d を移動する

のにかかる時間は $d/(v-r)$ で得られる。したがって、距離 d を移動するときの燃料消費量は、

$f(v) \times (d/(v-r))$ である。

2点間を航行する際に、コストをできるだけ小さくするような航路と船速を求める手法はこれまでにいくつか提案されている。その代表的なものがLo and McCordのモデルである⁹⁾。このモデルは、始点と終点を結ぶ直線を均等に分割し、その各分割点での方位と出力を定めることを目的としている。数理モデルとしては、船首方位と出力を変数として、これらを定める計算手順を提示している。Lo and McCordはこの問題を数理モデルに翻訳する段階では方位と出力の2種類の意思決定を同時に行うことを目的としている。しかし、実際にそのモデルを解く段階になると、2種類の決

定を同時に行うことは難しいとして、一方の変数を固定して他方のみを最適化する、という計算方法を採用している。すなわち、彼らは近似最適解を求めている。Lo and McCord は論文中にこの近似最適解が真の最適解とどの程度離れているかを数値的に評価している。そこでは最適解の乖離は小さく、実用上は十分な精度であるとしている。

2.2.2 数理最適化モデル

ここで述べる船速および航路計画のための数理最適化モデルは、船速と航路を同時に最適化するものである¹⁰⁾。この最適化モデルでは、始点から終点への移動を、ネットワーク上の s - t パスとして表現する。まず、船舶の航行海域内から船舶の通過するポイントを選ぶ。これら各ポイントに対してノードをひとつ定義する。船舶スケジューリングで用いたネットワークではノードは各輸送依頼を抽象的に表していたのに対して、ここではノードは船舶の航行する海域内の具体的な地点を表していることに注意する。ノードを定義する地点は、等間隔である必要はなく、また相互の位置関係も特に制限はない。たとえば、細かい操舵が必要と考えられる比較的沿岸に近い海域では狭い間隔でポイントを取り、必要でない海域では広い間隔でポイントをとればよい。そして、あるノード i を通過後、次にノード j を通過する可能性があるときに、 i から j にアーク (i, j) を定義する。こうして定義したアークの集合を A と表す。運航の出発位置をネットワーク上の始点 s と定義し、目的位置を終点 t と定義すると、出発位置から目的位置までの船舶の航路はネットワーク上の s - t パスとして表される。

アーク (i, j) にはその上を航行する費用を定義するが、この費用は i から j への航行の際の燃料消費量で定義する。燃料消費量は、航行距離 d_{ij} と、

その間の船速 v_{ij} との関数として定義される。船速 v_{ij} はエンジン仕様によって定まる最小値 $v_{\min}$ 以

上、最大値 $v_{\max}$ 以下の値をとる。

この船速および航路計画を数理最適化問題として定式化する方法を述べる。まず、アーク (i, j) に対して 0-1 変数 x_{ij} を定義する。これは航路にアーク (i, j) が含まれるときに 1、含まれないときに 0 をとる。また、アーク (i, j) には x_{ij} に加えてそこを航行する船速を表す連続変数 v_{ij} を定義する。

v_{ij} は、 x_{ij} が 0 のときは 0 とし、 x_{ij} が 1 のときの

み $v_{\min}$ 以上、 $v_{\max}$ 以下の値をとる（図-8）。

通常、航海では、目的地には定められた時刻までに到着する必要がある。言い換えると、出発位置から到着位置までの航海時間は、ある与えられた時間 T 以下である必要がある。この制約のことを、総航行時間制約と呼ぶことにする。ここで、アーク (i, j) 上の航行時間を t_{ij} と表すことにすると、航海にかかる時間は

$$\sum_{(i,j) \in A} t_{ij} x_{ij}$$

として表される。総航行時間制約は、この和が T 以下であるということなので、数式では

$$\sum_{(i,j) \in A} t_{ij} x_{ij} \leq T$$

と書くことができる。

アーク (i, j) の費用は、その航行の際の燃料消費量として定義する。船速 v_{ij} のときの単位時間当た

りの燃料消費量を $f(v_{ij})$ と表すと、アーク (i, j) の移動時間は $d_{ij} / (v_{ij} - r_{ij})$ であることからアーク (i, j) 上の燃料消費量は

$f(v_{ij}) \times (d_{ij} / (v_{ij} - r_{ij}))$ として表される。ここで、

r_{ij} はアーク (i, j) 上の船速低下を表す。これらの

記号を用いると、目的関数は

$$\sum_{(i,j) \in A} f(v_{ij}) \times (d_{ij} / (v_{ij} - r_{ij}))$$

と表される。つまり、目的関数は v_{ij} の非線形な関数となる。 $f(v)$ は一般には n 次の多項式として表すが¹¹⁾、以下では特に 3 次の多項式と仮定する、すなわち、

$$f(v) = \alpha v^3 + \beta v^2 + \gamma v$$

とする。

アーク (i, j) 上の移動時間を t_{ij} と表したが、これは $t_{ij} = d_{ij} / (v_{ij} - r_{ij})$ によって定まる。これを、総航行時間制約に代入すると、

$$\sum_{(i,j) \in A} x_{ij} d_{ij} / (v_{ij} - r_{ij}) \leq T$$

という制約式が得られる。これは非線形な制約式である。

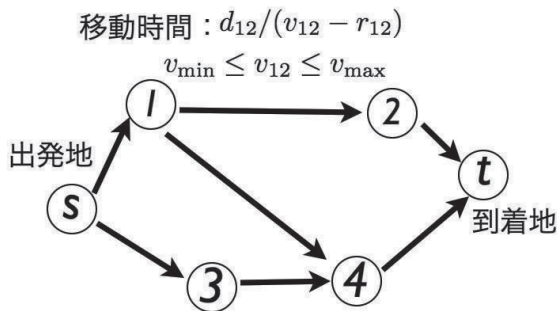


図-8 船速および航路計画で用いるネットワーク

一般に非線形制約は、線形制約に比べて扱いにくい。ところが、 $f(v)$ が3次の多項式であれば、この制約式は二次錐制約として表すことができる。二次錐制約は非線形制約ではあるが、特別な望ましい構造を持っているため、効率的な計算が可能となる。二次錐制約と線形制約で表された数理最適化問題は、二次錐計画問題と呼ばれる¹²⁾。

線形最適化は、変数が非負であるという制約をもっており、この非負象限の数理的構造を活かすことによって効率的な求解が可能となるのであった。そして、二次錐は、非負象限の数理的構造を一般化したものである¹²⁾。

総航行時間制約は二次錐制約として表すことができるが、同様に、目的関数も二次錐制約を用いて表すことができる。結局、この船速および航路計画は、整数変数(0-1変数)を含んだ二次錐最適化問題(mixed-integer second-order cone optimization problem, MISOCP)として表すことが

できる。二次錐最適化問題は、整数変数を含まなければ相当程度大規模な問題でも内点法によって解くことができる。一方、MISOCPを解くことは難しかったが、近年理論・計算手法の両面から研究がすすみ、小規模な問題であれば実用的な時間で解くことができるソルバが登場してきた。ただし、ある程度の成熟段階にある整数線形最適化ソルバと異なり、MISOCPソルバは未だ発展途上の段階である。したがって、中規模以上の問題に対しても安定して解を求めるには、問題の構造を活かした計算方法を構築することが望ましい。

2.2.3 航路生成アルゴリズム

船速および航路計画に対する問題の構造を活かした計算方法として、「航路生成アルゴリズム(route generation algorithm)」とよぶものを述べる¹⁰⁾。これは、整数を含まない二次錐最適化問題は効率的に解けること、および、制約の少ない最短路問題は整数最適化問題としてソルバで解けることを利用した計算方法である。

航路生成アルゴリズムでは、「緩和問題」を利用している。整数変数を含む最適化問題(整数最適化問題)は、難しい数理最適化問題とされている。正確な最適解を現実的な計算時間で求めるための万能手法はなく、効率的な計算のために個々の問題の構造を活かしたさまざまな手法が提案されている。特に規模の大きな整数最適化問題では、厳密な最適解ではなく、近似最適解で満足する必要のある状況も多い。いま、ある近似最適解 x が得られているとき、これが厳密な最適解からどの程度離れているかを評価できると有用である(もしかしたら、 x よりもはるかによい実行可能解があるかもしれない)。このようなとき、近似最適解のよさを見積もる仕組みが必要となる。このための仕組みの1つが緩和問題である。

いま、考えている数理最適化問題 P_0 の実行可能領域(制約条件を満たす解の集合)を S 、目的関数を $f(x)$ ($x \in S$) とする。このとき、 S を含むより大きな領域 $S' \supseteq S$ と、すべての $x \in S$ において $g(x) < f(x)$ を満たす関数 $g(x)$ を定める。これらを用いて定義する数理最適化問題 $\tilde{P}_0$ を、次で定義する。

$$\begin{array}{ll} \min & g(x) \\ \text{subject to} & x \in S' \end{array}$$

この $\tilde{P}_0$ を、元の問題の緩和問題と呼ぶ。いま、 P_0 は (大規模な整数最適化問題のように) 最小値 $f^* = f(x^*)$ を求めることが難しいとする (ただし、 $x^* \in S$)。一方、 $\tilde{P}_0$ は (線形最適化問題のように) P_0 に比べて解くことが容易で、その最小値 $g^* = g(y^*)$ も容易に求まるとする (ただし、 $y^* \in S'$)。このとき、

$$g(x) \leq f(x) \quad \forall x \in S$$

であることから、 $g^* < f^*$ であることがわかる。

すなわち、緩和問題の最小値 g^* を知ることで、元の最適化問題の最小値 f^* を評価することができるのである。具体的には、 P_0 に対して求めた近似最適値 f' の値が g^* に十分に近ければ、 f' は質のよい近似最適値として安心して使える (図-9)。

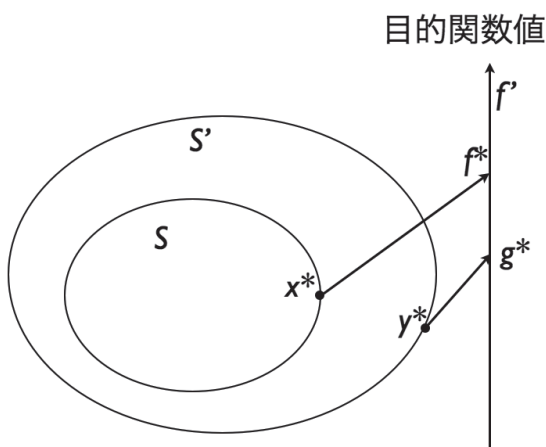


図-9 緩和問題の概念図

緩和問題は、整数最適化問題を解く計算のために効果的に用いることができる。整数最適化問題

ソルバの中にも巧みに組み込まれている。整数最適化問題ソルバの多くは、分枝限定法とよばれる計算を実行している。この計算においては、分枝操作と限定操作を繰り返し行う。そして、限定操作において、緩和問題が重要な役割を果たす。分枝操作とは、問題の実行可能領域 S を、複数の重なりあわない領域に分割する操作である。例として、 S を $S1$, $S2$, $S3$ に分割するとする。ここで、 $S1$, $S2$, $S3$ における最小値をそれぞれ評価したい。いま、 $S3$ における最小値 f_3 がわかっているとすると、このとき、 $S1$ における最小値 f_1 , $S2$ における最小値 f_2 がいずれも f_3 より大きければ、 f_3 が S 全体における最小値であることがわかる。さて、元々の問題が難しければ、 f_1 , f_2 を求めるのも相変わらず難しい。 f_1 , f_2 の代わりに、 $S2$, $S3$ に対する緩和問題の最小値 f_1^* , f_2^* を求める。 $f_1 > f_1^*$, $f_2 > f_2^*$ を満たしているから、 $f_3 < f_1^*$, $f_3 < f_2^*$ を満たしていれば、 f_3 は S 全体でも最小値であることがわかる。このように、緩和問題に情報を用いることで、もとの実行可能領域 S のなかで最適解を含む領域を効果的に絞ることができる。

この緩和問題による限定操作を用いることで、船速および航路計画に対する航路生成アルゴリズムを構築する。

船速および航路計画は、MISOCP として定式化できた。これを直接 MISOCP を解くソルバに直接入力すると、データが大きい場合などは実行時間が長くなる可能性がある。そこで、MISOCP からより解きやすい2つの問題を定義し、それらの結果を用いて MISOCP の最適解を得る方法を述べる。

MISOCP において、 x をある値に固定する。この x は 1 つの $s-t$ パスを表している。このとき、MISOCP は整数変数を含まない SOCP となり、効率的に解くことができる。これは、固定した $s-t$ パス上での各アーク上の最適船速を求めていることにあたるので、この SOCP を船速最適化問題とよぶことにする。 $s-t$ パスの数は有限であるので、原理的にはすべての $s-t$ パスに対して船速最適化問題を解き、その中で費用が最小のものを選べば MISOCP の最適解が求まる。しかし、 x の数は膨大であるので、これを実行することは不可能である。そこで、緩和問題の考え方を用いて、最適船速を求める必要のないパスを取り除くしくみを導入する。

MISOCP による定式化において、 v を固定すると、総航行時間制約のみをもつ最短路問題となる。これは整数線形最適化問題であり、ソフトウェアによって高速に解ける。この解として、 $s-t$ パスがひとつ得られるので、航路探索問題とよぶことに

する。ここで、制約条件においては船速 v は $v_{\max}$ に固定し、目的関数においては $f(v_{ij}) \times (d_{ij} / (v_{ij} - r_{ij}))$ を最小化する v_{ij}^* に固定する。これは、目的関数はもとの問題のものよりも小さく、制約条件は緩くなっているのもとの問題の緩和問題になっているのである。

航路生成アルゴリズムは、航路探索問題と船速最適化問題とを繰り返し解くものである。まず、航路探索問題を解き、 s - t パス $p1$ をひとつ得る。このパス $p1$ に対して船速最適化問題を解く。これで、もとの MISOCP に対する実行可能解 (x_j, v_j) がひとつ得られる。そして、 (x_j, v_j) のときの燃料消費量を f_1 と表す。この f_1 を暫定最適解 f^* として記憶しておく。次に、再び航路探索問題を解き ($p1$ の生成をはばむような線形制約を加えておく)、 $p1$ の次に目的関数値の小さい s - t パス $p2$ を得る。このときの航路探索問題の目的関数値が f_1 よりも大きければ、緩和問題の性質より (x_j, v_j) が最適値であることがわかる。一方、 f_1 以下であれば (x_j, v_j) よりもよい解がある可能性があるので、 $p2$ に対して船速最適化問題を解く。これにより、もとの MISOCP の実行可能解 (x_2, v_2) を得る。このときのコスト f_2 と f_1 を比較し、 $f_2 < f_1$ であれば暫定最適解 f^* として f_2 を記憶しておく。これをくりかえすのが航路生成アルゴリズムである。

このアルゴリズムの性能を、数値計算により検証した。MISOCP を最適化ソフトウェアに入力した場合の計算と比較したところ、計算効率が相当程度よくなることがわかった¹⁰⁾。

2.3 船速を考慮した船舶スケジューリング

上述した船舶スケジューリングでは、計画期間(例：2週間)全体で、船舶の運航速度は一定速度であると仮定していた。しかし、実際には時間枠に余裕があるときには低速で運航することができる。単位時間あたりの燃料消費量は一般に3次以上の多項式として近似できるとされているので、低速で運航することによって燃料消費量をかなりの程度小さくすることができる。

そこで、船舶スケジューリングにあたって、港の訪問順序に加えて2港間の運航速度をも決定変数とすることにより、船隊全体の燃料消費量を定速運航のときよりも小さくすることができる。この設定での最適化モデルにおいて追加で与える入力データは、各船舶の最大船速と最小船速である。この最適化モデルの答えとして、総燃料消費量を最小化するような運航ルートを得ることが出来るが、ここでは運航ルートは港の訪問順序に加えて

それに含まれる2港の運航速度も得られる。

この最適化モデルを解く際に、上述の一定速度での最適化モデルと異なるところは、ルート候補を生成する際に解くネットワーク上の最適化問題である。一定速度の場合は、ネットワーク内のアークの費用は定数として定義されるが、速度をも決定変数とする場合は、アークの費用は船速の非線形関数として定義される。これにより、ネットワーク上の最適化問題は整数非線形最適化問題(MINLP)となり、ネットワーク構造を活かした効率的な計算手法を構築することは容易ではない。そこで、MINLP を解くことのできる最適化ソフトウェアによって求解する。ただし、MINLP を解くソフトウェアはいまだ発展途上の段階であり、常に安定して解ける訳ではない。そこで、問題のネットワーク構造とMINLPの機能を併せて活用することによる効率的な計算手法を開発し¹³⁾、現在数値検証を実施しているところである。

3. 不確実性を考慮したサプライ・チェーンモデル

3.1 リスクを考慮した最適化モデル

サプライ・チェーンにおいては効率化が重要であるか、同時に機能の継続性も重要である。輸送を含めたサプライ・チェーン内の事象には不確実性がある。たとえば、需要には不確実性があるため、品切れの確率を一定以下に押さえるには、この不確実性に対処するための方策をとる必要がある。需要に限らず、予測は必ず外れるものであり、予測に基づいて計画を立て、実行することは現実的な効果は薄い。できることは、起こり得るケースを挙げ、それらが実現したときの対策を決めておくことである。数理最適化モデルにおいては、不確実性に対処するための方法論が複数提案されているが、そのうちでサプライ・チェーンにおいて有効なものを述べる¹⁴⁾。

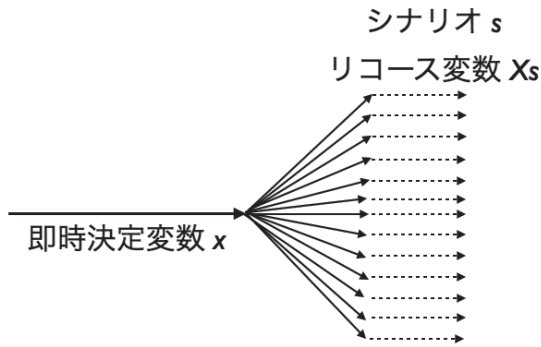
3.1.1 2段階の確率最適化

定常的なリスクに対応するための数理最適化の手法として、確率最適化がある。一般には多段階の確率最適化の枠組みが提案されているが、サプライ・チェーンの運用と関連づけた場合には、2段階の確率最適化がわかりやすいため、ここでは、2段階の確率最適化を述べる。

確率最適化においては、将来における不確実な事象は、シナリオの集合 S として表される。すなわち、将来に起こる可能性のあるケースを挙げていき、それらを要素として S を定義する。さらに、

あるシナリオ $s \in S$ の起こる確率を p_s と表すこ

とにする。



図－10 シナリオを用いた確率的最適化の概念図

2段階の確率最適化といった場合の「2段階」とは、意思決定を2段階で行うということを意味する。まず、第1段階、すなわち今すぐに決定しなければならない意思決定を表す変数を、即時決定 (here and now) 変数とよぶ。ここでは、即時決定変数をベクトル x で表す。これは、将来的に起こる可能性のある事象に起因するリスクに対する予防行動をモデル化したものです。一方で、第1段階の予防行動を決定後、第2段階で行う意思決定を表す変数をリコース (recourse: 償還請求) 変数とよぶ。リコース変数はシナリオごとに定義される。シナリオ s に対してここでは、リコース変数をシナリオ s ごとに定義されたベクトル X_s で表す。これは、シナリオ s が実現した際、それに対する応答行動をモデル化したものである (図－10)。

即時決定変数 x に対する費用関数を $f(x)$ とし、 x のとり得る値の集合 (実行可能解の集合) を F と表す。また、シナリオ s に対するリコース変数 X_s に関する費用関数を $g_s(X_s)$ とする。リコース変数 X_s は、シナリオ s に対する応答行動を表しており、即時決定変数 x に依存する。したがって、 X_s のとりうる値の集合を即時決定変数 x の関数として $G_s(x)$ と表す。ただし、 $G_s(x)$ が空にならないように実行可能解を緩和し、元の実行可能解の集合からの逸脱をペナルティとして加えるものと仮定する。

これらを用いると、2段階の確率最適化の数理最適化モデルは、次のように表される。

$$\begin{aligned} \min \quad & f(x) + \sum_{s \in S} p_s g_s(X_s) \\ \text{subject to} \quad & x \in F \\ & X_s \in G_s(x) \quad \forall s \in S \end{aligned}$$

3.1.2 リスク中立モデル

上のモデルの目的関数は、期待値の最小化として定義している。このように、期待値のみを評価基準とするモデルは、リスク中立 (risk neutral) モデルと呼ばれる。期待値の最小化を目的とするということは、同様の意思決定を繰り返す状況を想定していることになる。

3.1.3 リスク回避モデル

短期の繰り返し行う事項であれば、この立場のモデルでよいが、長期の課題などで1度だけ意思決定を行う場合には期待値だけでは不十分なケースもある。このように、期待値を評価基準としない立場の代表的な例が保険の導入である。期待値だけを評価基準とすると、保険をかける理由はないが、1度だけであっても破壊的な損失を被る可能性のあるリスクに対して、その対応のために保険をかけるという立場がある。このようにリスクを避ける行動をとる立場を、リスク回避 (risk averse) とよぶ。

リスク回避の立場に立った数理モデルとして、VaR (Value at Risk) モデルがある。VaR モデルでは以下で述べるような評価基準が用いられる。まず、費用が連続な確率変数であると仮定する。意思決定者は、費用が閾値以下になる確率が β 以下になるようにしたい。ここで用いる閾値を VaR とよぶ。VaR は設定確率 β の値によって変わるので、 β -VaR と表されることもある。たとえば、 β を 0.95 とすると、95% の確率で費用が VaR 以下であることが保証されるような変数を求めるモデルとなる。数理モデルとしては、事象の正規確率が対称な分布をもつ場合は、分散を最小化するような古典的なモデル (例: 金融におけるポートフォリオ理論) も考えられる。

VaR モデルの問題点はその数理最適化モデルが解きにくいことである。そこで、VaR モデルの代わりに用いられるのが、CVaR (条件付き VaR, Conditional Value at Risk) モデルである。CVaR は、費用が VaR を超えるときの期待値のことである。CVaR は VaR の上界を与える。これも確率 β に依存するので、 β -CVaR と書かれることもある。

CVaR は以下の最適化問題を解くことで求めることができる。

$$\begin{aligned}
& \min \quad y + \frac{1}{1-\beta} \sum_{s \in S} z_s \\
& \text{subject to} \quad z_s \geq f(x) + \sum_{s \in S} p_s g_s(X_s) - y \quad \forall s \in S \\
& \quad \quad \quad z_s \geq 0 \quad \forall s \in S \\
& \quad \quad \quad x \in F \\
& \quad \quad \quad X_s \in G_s(x) \quad \forall s \in S
\end{aligned}$$

3.1.4 リスクに対する対処法と戦略

リスクに対処する方法は、大きく予防と応答に分けられる。予防は、サプライ・チェーンが内部機能の一部が途絶することにより機能不全に陥ることを防ぐための行動を指し、応答は、機能不全に陥ったときにそれに対してとる対応のことを指す。応答は、途絶に伴う被害をできるだけ小さく抑えることを目的とする。リスクへの対処法は、その活動が継続する期間や、途絶からの経過時間・状態、すなわちフェーズによって分類することができる。前述のように、途絶を考慮しない通常のサプライ・チェーンにおける意志決定は、ストラテジック、タクティカル、オペレーショナルと各レベルに区別することが有用であった。リスクへの対処法にもこの3つのレベルへの分類を用いることが有用である。

予防とは、途絶によるサプライ・チェーン機能の変化に備えてとる行動である。サプライ・チェーンにおいて途絶が発生した場合、内部のプレーヤー（例：小売店、生産者）は自らの被害を小さくする目的で、普段とは違う行動を取ることが考えられる。典型的な例としては、小売店からの需要の急激な増加が挙げられる。このような変化に対応しなければ、サプライ・チェーンが機能不全に陥ってしまう。変化に俊敏に対応するためには、あらかじめ予防行動をとっておくことが必要である。

途絶には、事前に予兆のあるものとそうでないものがある。事前に予兆のあるものに対する備えは、効果的なものを行うことができる。このように、事前にある程度具体的・定量的な情報が得られている状態でとる予防行動のことを、事前準備という。事前準備の例としては、台風に対するものが挙げられる。台風の進路と規模、そしてそれに伴う特定の地域の被害は、ある程度の確度で知ることができる。そして、その被害を軽減するための具体的な対策を知ることができる。例えば、住民の避難所への誘導にかかる時間、避難所で必要な生活用品の内容・量、避難期間などが対策の例として挙げられる。事前準備は数日～数週間の

比較的短い期間で行う行動であり、オペレーショナル、またはタクティカルレベルの意志決定に分類できる。

事前準備が数日から数週間の比較的短い時間で行う行動であるのに対し、事前準備でない予防は、数ヶ月から数年単位で行う行動、すなわちストラテジックレベルな行動である。実際に途絶が発生したときに迅速に対応するには、そのためのインフラ整備が必要である。具体的な例としては、建物の耐震補強、避難場所の選定が挙げられる。人員の訓練、情報処理機器の設置なども事前準備でない予防に含まれる。

予防行動においては、予想される途絶の種類、場所、規模などの適切な評価が重要である。そのためには、地理的データ、人口データ、気象データ、地質学的データなど、広範なものを組み合わせる必要がある。場合によってはそれらによるコンピュータ・シミュレーションも有用である。

応答行動は、大きく初期応答と復旧に分けられる。初期応答は、途絶発生直後に行う被害軽減を目的とした活動を指す。初期応答で重要なのは、迅速な情報収集である。途絶発生時には、通常の状態で行っていた連絡手段が機能しない可能性がある。このようなときに、被害状況などを迅速に収集するための方法を用意しておくことが重要である。たとえば、都市部における災害発生時の情報収集のための1つの方法が、Savvaidis らによって提案されている¹⁵⁾。Savvaidis らが扱った問題は、都市部での災害時の初期応答における緊急車両の最適な管理と道路途絶状況の把握である。このとき、道路の途絶状況を把握するためのセンサとして利用するシステムが提案されている。このシステムでは都市部にすでに存在している車両から各道路セグメントの移動にかかった時間をGPS データに基づいて集める。このことにより、災害発生地域の道路ネットワーク全体の情報を収集することができる。こうして収集したデータは、緊急車両の配置などの初期応答行動のために用いることができる。

途絶発生直後に行う初期応答に対して、途絶発生直後ある程度の期間が経ったのち、サプライ・チェーンの機能を元に戻すための行動を、復旧とよぶ。復旧で重要なのは、関与するプレーヤー間の関係である。具体的には、被害地域の行政機関、企業、国の期間、支援団体などがそれぞれ異なる目的を持って関与するため、それらの目的を鑑みながら適切な復旧活動につながるようにすることが望ましい。

3.1.5 意思決定レベル

ここでは、確率計画の数理モデルと意志決定レ

ベルとの関係を述べる。

ストラテジックレベルの意思決定は、数年から数十年にわたって影響を及ぼすものである。このレベルの意思決定は、従来は即時決定変数のみを用いてモデルを組み立てることが多かった。ところがこのレベルの意思決定にも2段階の確率最適化によるモデルを与えることができる。これには、途絶発生後に変更可能な変数を、リコース変数としてモデル化を行うとよい。また、第1段階での意思決定を表す即時決定変数は、途絶発生後にペナルティを支払うことによって変更可能な変数として扱う場合もある。このような変数は準リコース変数と呼ぶ。準リコース変数でモデル化できるものの例は、船舶やトラックの輸送手段が挙げられる。これらの輸送手段を長期的な契約に基づいて借りている場合、短期的には超過料金を支払って使用できる権利を変更することができるような状態を想定している。

タクティカルレベルの意思決定では、ストラテジックレベルと同様に、途絶発生後に変更可能な変数は、リコース変数としてモデル化を行う。

タクティカルモデルは、ストラテジックレベルと同様に、途絶発生後に変更可能な変数は、リコース変数としてモデル化を行う。

オペレーショナルレベルの意思決定において、応答行動をモデル化する際には、従来はリコース変数のみを用いた最適化が行われてきたが、予防に相当する変数を取り入れたモデルを構築することもできる。

3.2 リスクを考慮した最適化モデルの例

リスクを考慮したサプライ・チェーン最適化モデルの例として、確率的在庫モデル、安全在庫配置、動的ロットサイズのモデルを述べる。これは直に海上輸送のシステムを扱うものではない。しかし、海上輸送システムの動きは、運ぶものがあるてはじめて定まるものである。したがって、港間での船舶による輸送だけ取り出して考察の対象とすることは望ましくない。むしろ、運ぶものは（陸上含めて）どこから、いつ、どれだけ、どのように発生するのか、も考察の対象とする方が自然である。

3.2.1 確率的在庫モデル

確率的在庫モデルでは、途絶のリスクを考慮する。

サプライ・チェーンにおいて途絶が発生した場合の機能不全として代表的なものに、供給の途絶がある。このモデルは、供給の途絶をモデルに加味したものを中心として、理論的な研究が進んでいる。確率的在庫モデルの古典的モデルでは、動

的計画が用いられてきた。この古典的モデルを拡張したものや、さまざまな条件下でのシステムの振る舞いに関する洞察を得る理論的モデルが提案されている。

途絶を考慮した多期間の確率的在庫モデルとしては、即時決定変数を用いた静的発注量モデルが有用である。このモデルは、発注量を過去の情報に対するアフィン関数として定義した変種も有用である。これを、適応型モデルとよぶ。

確率的在庫モデルに対しては従来は理論的モデルによる解析が盛んであるが、リスクを考慮した確率最適化モデルを構築することができ、これらを数値実験により解析することで、途絶発生時の応答行動を定めることができる。

また、確率的在庫モデルでは、部品/原料の調達元を複数としたモデルを構築することができる。このモデルでは、ある製品を製造するための部品の調達元として、2つ以上を設定する。こうしておくことで、1つの調達元が途絶による機能しなくなったシナリオでも、他の調達元から部品を調達することによって製品の製造を継続することができる。

3.2.2 安全在庫配置モデル

安全在庫とは、需要の変動があった場合にも品切れをある確率で防ぐためにもっておく在庫のことである。品切れが起きる確率によって、もつべき安全在庫の量も変わる。安全在庫配置モデルでは、正味補充時間という量が重要な機能を果たす^{16, 17)}。これは、大まかには、ある製品を注文してから実際に補充されるまでの時間を表す。サプライ・チェーン内の各拠点での在庫費用は正味補充時間に依存する。そこで、安全在庫問題配置モデルでの目的は、品切れを防ぎつつ在庫の総費用が最小となる正味補充時間を求めることとなる。

この安全在庫配置モデルにリスクを加味する場合、正味補充時間は途絶の前に決定しておく即時決定変数としてモデル化する。これは、安全在庫量は途絶の前に決定しておく必要があり、途絶後も急には変更できないことを反映している。

4. まとめ

本稿では、輸送システムに関する数理最適化の手法を述べた。輸送システムは、サプライ・チェーンの一構成要素であり、在庫管理や施設配置と完全に切り離すことはできず、それらの接続部分もある程度考察の対象に入れる必要がある。そこで、輸送システムの他にも、ロジスティクスネットワーク設計問題と安全在庫配置問題について述

べた。輸送システムにおける数理最適化モデルとして、船舶スケジューリング(配船計画)の最適化モデルならびに船速と航路を同時に最適化するモデルを述べた。これらは筆者らがこれまで研究を進めてきたものであり、近年の数理最適化およびその応用分野での最先端の知見を反映したものである。

また、サプライ・チェーンにおいては情報の不確実性がつきものである。それらを扱うことは難しい点が多いが、近年そのための様々なフレームワークが開発されつつある。筆者らは東京海洋大学などと共同でそのフレームワークの開発をすすめてきたが、その一部を紹介した。予測は「常に外れる」という前提で、不確実性に向き合う方法である。

数理最適化の分野は、理論・モデル化・ソフトウェアが年々進歩しており、10年前には解くことが難しかったような大規模実務問題も扱えるようになった。ただし、そのためには「適切なモデル化」が必要である。適切なモデル化には、対象と数理最適化の両方についての洞察が必要であるが、そのような機能が提供可能なソフトウェア企業、コンサルティング企業も複数存在する。弊所やこのような企業との情報交換を通じて、輸送システムの効率化を実現する具体的な手順が明らかになる可能性は高い。

謝 辞

本研究の一部は、科研費 23510162 の助成を受けたものである。

参考文献

- 1) 田村明久, 村松正和: 最適化法, 共立出版, 2002
- 2) 柳浦睦憲, 茨木俊秀: 組合せ最適化-メタ戦略を中心として-, 朝倉書店, 2001
- 3) Golden, B. L., Raghavan, S., and Wasil, E. A. (eds.), The Vehicle Routing Problem: Latest Advances and New Challenges, Springer, 2008
- 4) Christiansen, M., Fagerholt, K., Nygreen, B. and Ronen, D.: Ship routing and scheduling in a new millennium, European Journal of Operational Research, 228(2012), pp.467-483
- 5) 藤澤克樹, 梅谷俊治: 応用に役立つ50の最適化問題, 朝倉書店, 2009
- 6) Desaulniers, G., Desrosiers, J., and Solomon,

M. (eds.): Column Generation, Springer, 2005.

- 7) Kobayashi, K. and Kubo, M.: Optimization of oil tanker schedules by decomposition, column generation, and time-space network techniques, Japan Journal of Industrial and Applied Mathematics, 27(2010), pp.161-173
- 8) Koch, T.: Rapid Mathematical Programming, Technische Universität Berlin, ZIB-Report 04-58(2004)
- 9) Lo, H. K. and McCord, M. R.: Adaptive ship routing through stochastic ocean current: general formulations and empirical results, Transportation Research Part A: Policy and Practice 32(2003), pp.138-156.
- 10) Tanaka, M. and Kobayashi, K.: MISOCP formulation and route generation algorithm for ship navigation problem, Department of Industrial Engineering and Management, Tokyo Institute of Technology, Technical Report 2013-8(2013)
- 11) Psaraftis, H. N. and Kontavas, C. A.: Speed models for energy-efficient maritime transportation: A taxonomy and survey, Transportation Research Part C, 26(2013), pp. 331-351
- 12) Alizadeh, F. and Goldfarb, D.: Second-order cone programming, Mathematical Programming, 95(2003), pp.3-51
- 13) 小林和博, 非線形な目的関数をもつ船舶スケジューリングに対するルート列挙解法, 日本オペレーションズ・リサーチ学会 2015 年春季研究発表会アブストラクト集, 2015
- 14) 久保幹雄: SCRM における数理最適化モデルと実験的解析, サプライ・チェーン・リスク管理と人道支援ロジスティクス第三回セミナー, 2014
- 15) Savvaiddis, P., Lakakis, K., and Ifadis, I.: Organization of emergency response after a major disaster event in an urban area with the help of an automatic vehicle location and control system, GPS Solutions, 5(2002), pp.70-79
- 16) 久保幹雄, 小林和博, 武田朗子, 田中未来, 村松正和, サプライ・チェーン最適化における2次錐最適化の応用, オペレーションズ・リサーチ, 59(2014), pp.739-747
- 17) 田中未来, 久保幹雄, 小林和博, 武田朗子, 村松正和, 不確実性を考慮したサプライ・チェ

イン最適化モデル，スケジューリングシンポジウム 2014，G5-1, 2014