階層的積木法と列生成法の融合–輸送・船舶スケジューリングを例として–

久 保 幹 雄*, 小 林 和 博**

*東京海洋大学 海洋工学部 流通情報工学科 東京都江東区越中島 2-1-6
**海上技術安全研究所 物流研究センター 東京都三鷹市新川 6-38-1
* Tokyo University of Marine Science and Technology, Faculty of Marine Technology, 2-1-6 Etsuujima, Koutou, Tokyo, Japan
** National Maritime Research Institute, Center for Logistics Research, 6-38-1 Shinkawa,Mitaka, Tokyo, Japan
*E-mail: kubo@logopt.com
**E-mail: kobayashi@nmri.go.jp

キーワード：階層的積木法，列生成法，輸送スケジューリング, 船舶スケジューリング
JL 0008/03/4208–0685 ©2008 SICE

1. はじめに

本稿では，多くのメタ解法を含んだフレームワークである階層的積木法 [2] と，大規模数理計画に対するアプローチである列生成法 [1] を融合した解法を提案し，実際問題への適用例を示す．

以下の構成は次の通り．

2 節では，階層的積木法について述べる．

3 節では，列生成法について概観する．

4 節では，階層的積木法と列生成法の融合について考える．

5 節では，輸送スケジューリングに対する応用について述べる．

6 節では，船舶スケジューリングに対する適用を紹介する．

7 節では，まとめと今後の課題について述べる．

2. 階層的積木法

以下の組合せ最適化問題を考える．ある空でない有限集合 $\mathcal{U}, \mathcal{U}$ から整数（実数）への関数 $c : \mathcal{U} \rightarrow \mathrm{R}, \mathcal{U}$ の部分集合の集まり $\mathcal{F} \subseteq 2^{\mathcal{U}}$ が与えられたとき

$$\min\{f(x) = \sum_{u \in x} c(u) : x \in \mathcal{F}\} \tag{1}$$

を与える $x \in \mathcal{F}$ を求める．$\mathcal{U}$ を台集合，$\mathcal{F}$ を実行可能解の集合，その要素 $x \in \mathcal{F}$ を実行可能解または単に解とよぶ．

関数 c を台集合上の費用関数，式 (1) における関数 f を目的関数または費用関数とよぶ．目的関数を最小にする解 $x \in \mathcal{F}$ を（大域的）最適解とよび，その集合を $\mathcal{F}^*$ と記す．すなわち，$\mathcal{F}^* = \{x \in \mathcal{F} : f(x) \leq f(y), \forall y \in \mathcal{F}\}$ である．

解の集合 $\mathcal{F}$ を与えたとき，近傍 N は以下の写像と定義される．

$$N : \mathcal{F} \rightarrow 2^{\mathcal{F}}$$

近傍は何らかの意味で “近い”解の集合を表す．$x \in \mathcal{F}$ と $y \in N(x)$ の組 (x, y) を移動とよぶ．局所探索法，模擬焼き鈍し法，禁断探索法をはじめとする多くのメタ解法は近傍を基に設計される．

$\mathcal{F}$ に含まれる解 x の部分集合全体からなる集合を積木集合とよび，$\tilde{\mathcal{F}}$ と記す．また，$\tilde{\mathcal{F}}$ の要素を積木とよぶ．言い換えれば，$\tilde{\mathcal{F}}$ は $\mathcal{F}$ を単調化した集合である．ここで，$\tilde{\mathcal{F}}$ は解の集合および台集合を含む，すなわち $\mathcal{U} \subseteq \tilde{\mathcal{F}}, \mathcal{F} \subseteq \tilde{\mathcal{F}}$ が成立する．

2 つの積木 $x, y \in \tilde{\mathcal{F}}$ が $x \subset y$ を満たすとき，x は y に対して下層の積木であるといい，逆に y は x の上層の積木であるという．解 $x \in \mathcal{F}$ は，それより上層の積木をもたないので最上層の積木と定義し，逆に台集合の要素 $x \in \mathcal{U}$ は，空集合を除いてそれより下層の積木をもたないので最下層の積木と定義する．最下層と最上層の積木集合以外の積木の階層の定義は問題依存であり，同層の積木集合内での（積木集合上に拡張された）近傍が容易に定義でき，上層の積木を自然に構築できるように定義する必要がある．たとえば，配送計画問題に対しては，第 1 層（台集合 $\mathcal{U}$）は枝の集合，第 2 層はルート（巡回路）の集合，第 3 層は運搬車ルート（巡回路を組み合わせた運転手の 1 日のスケジュール）の集合，第 4 層は解集合 $\mathcal{F}$ と定義するのが有効である．

積木集合上では，複数の積木から別の積木を構成したり，1 つの積木から複数の積木を構成したりする操作が重要になる．ここでは，積木集合 $\tilde{\mathcal{F}}$ 上での移動操作として，通常の近傍探索法に対応する近傍探索移動，下層の積木から上層の積木を生成する組立移動，上層の積木から下層の積木を生成する分解移動の 3 つを導入する（**図 1**）．

近傍探索:
積木 $\beta \in \tilde{\mathcal{F}}$ を与えたとき，β と同層の積木集合上を探索する近傍探索法によって得られた積木の集合 $\mathcal{B}$ を返す．$\mathcal{F}$ 内の探索を行う通常の意味での近傍探索法は，探索を行う積木集合が最上層の特殊形と考えられる．

組立:
積木の部分集合 $\mathcal{B} = \{\beta_1, \cdots, \beta_m\} \subseteq \tilde{\mathcal{F}}$ を与えたとき，$\beta_1, \cdots, \beta_m$ を合成して得られる上層の積木を返す．

分解:

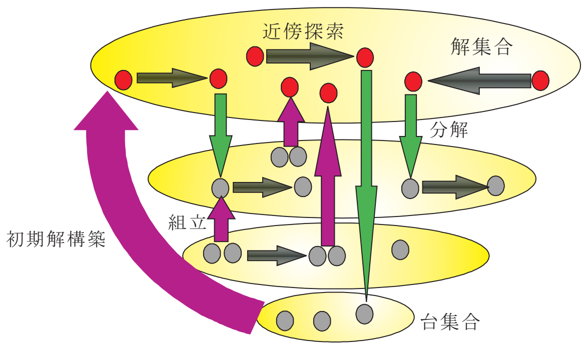

図 1　階層的積木法．

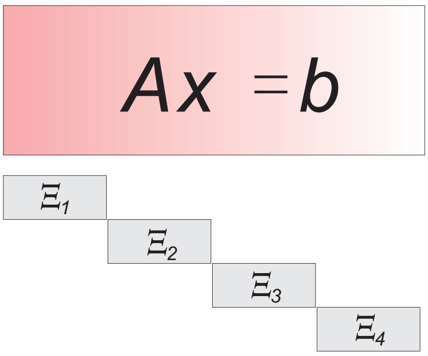

図 2　列生成法の参考図．

積木 $\beta \in \tilde{\mathcal{F}}$ を与えたとき，$\cup_{i=1}^{m} \beta_i \subseteq \beta$ を満たす下層の積木の集合 $\{\beta_1, \cdots, \beta_m\}$ を返す．

階層的積木法についての詳細は，[2] を参照されたい．

3．列生成法

列生成法は，以下のようなブロック対角構造をもった問題を対象とする（**図 2**）．

$$
\begin{aligned}
&\text{minimize} && \sum_{k \in \mathcal{K}} \boldsymbol{c}^k \boldsymbol{x}^k \\
&\text{subject to} && \sum_{k \in \mathcal{K}} \boldsymbol{A}^k \boldsymbol{x}^k = \boldsymbol{b} \\
& && \boldsymbol{x}^k \in \Xi_k \qquad \forall k \in \mathcal{K}
\end{aligned}
$$

ここで Ξ_k は特殊な構造をもった離散最適化問題に対する実行可能解の特性ベクトルの集合である．Ξ_k の凸包は有界な多面体であると仮定し，この多面体の端点を表すベクトルを $\boldsymbol{X}_p^k$ $(p \in P_k)$ と書く．

Minkowski–Weyl の定理より，解 $\boldsymbol{x}^k$ は，端点ベクトルの凸結合として，

$$
\boldsymbol{x}^k = \sum_{p \in P} \boldsymbol{X}_p^k \theta_p^k
$$

$$
\sum_{p \in P_k} \theta_p^k = 1
$$

$$
\theta_p^k \geq 0 \quad \forall p \in P_k
$$

と表すことができる．これを原問題に代入して，θ を変数とした問題に書き直し，さらに $\boldsymbol{x}$ に関する整数条件を緩和することによって，以下の問題を得る．

$$
\begin{aligned}
&\text{minimize} && \sum_{k \in \mathcal{K}} \sum_{p \in P_k} (\boldsymbol{c}^k \boldsymbol{X}_p^k) \theta_p^k \\
&\text{subject to} && \sum_{k \in \mathcal{K}} \sum_{p \in P_k} (\boldsymbol{A}^k \boldsymbol{X}_p^k) \theta_p^k = \boldsymbol{b} \\
& && \sum_{p \in P_k} \theta_p^k = 1 && \forall k \in \mathcal{K} \\
& && \theta_p^k \geq 0 && \forall p \in P_k, k \in \mathcal{K}
\end{aligned}
$$

この問題は，原問題にくらべて少ない制約をもつが，その代償として非常に多くの変数をもつ線形計画緩和問題である．これらの変数をすべて陽的に表すのは現実的ではないので，一部の変数だけを取り出した問題（制限つき主問題）を考える．制限つき主問題を解くことによって，基底解と同時に双対変数を得ることができる．最初の式に対する双対変数ベクトルを $\boldsymbol{y}$，2 番目の式に対する双対変数を r^k $(k \in \mathcal{K})$ とする．

現在，基底に入っていないすべての変数に対して，被約費用が非負であれば，現在の基底解が最適解であることが言える．すべての変数（その数は膨大である）に対して被約費用を調べるかわりに，各 $k \in \mathcal{K}$ ごとに以下の問題（子問題）を解く．

$$
\begin{aligned}
&z^k = \text{minimize} && (\boldsymbol{c}^k - \boldsymbol{y}\boldsymbol{A}^k)\boldsymbol{x}^k \\
&\text{subject to} && \boldsymbol{x}^k \in \Xi_k
\end{aligned}
$$

もし，$z^k < r^k$ であれば被約費用が負になる端点が求まったことになるので，新しい列を追加して，再び制限つき主問題を解く．この操作を，被約費用が負になる列がなくなるまで繰り返すアルゴリズムが列生成法である．

列生成法は，緩和問題を解いているだけであるので，実行可能解を得るためには，分枝限定法もしくは分枝価格法を用いるか，緩和問題から実行可能解を得るための適当なヒューリスティクス（上界導出法）を用いる必要がある．また，子問題の最適解を 1 つ求めて列を追加する方法では収束が遅いので，子問題の求解に動的計画を用い，被約費用が負になる複数の列を同時に追加する方法が有効である．

列生成法についての詳細は，[1] を参照されたい．

4．階層的積木法と列生成法の融合

列生成法は，2 段階の階層をもつ階層的積木法と解釈できる（**図 3**）．最上層は，階層的積木法と同じく解集合であるが，列生成法においては緩和解の集合が下層の積木と情報を交換する．緩和解は制限付き主問題を求解することによって得られ，同時に双対最適解を算出する．この双対情報をもとに，各 $k \in \mathcal{K}$ に対して子問題を求解することによって，新しい列に対応する下層の積木集合を得る．緩

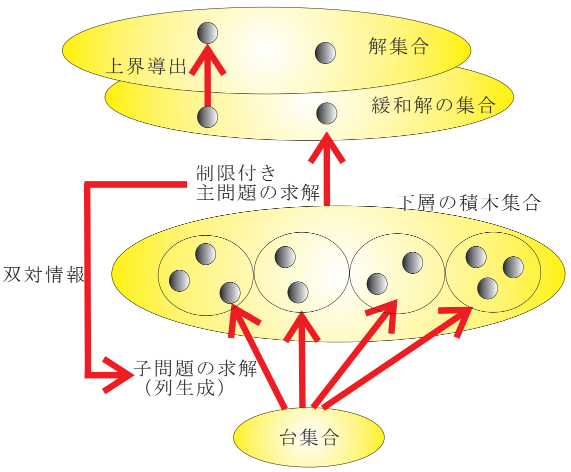

図 3　階層的積木法の観点による列生成法の解釈．

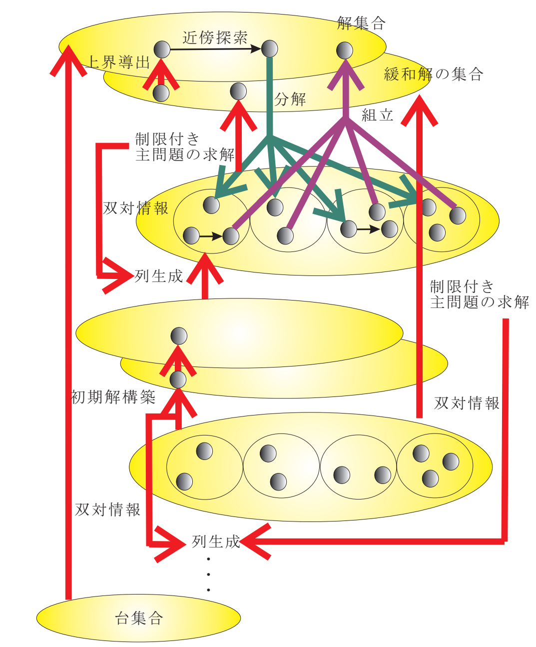

図 4　階層的積木法と列生成法の融合．

和解から解を生成するためには，何らかの上界導出ヒューリスティクス（分枝限定法を含む）を適用する．

つまり，列生成法とは，階層的積木法における各積木の階層に，緩和した積木集合を追加し，線形計画と双対情報をもとに系統的に新しい積木集合を追加するメカニズムと考えられる．このような解釈に基づくと，階層的積木法と列生成法は融合して用いることができる．**図 4** にそのフレームワークを示す．

以下では，階層的積木法と列生成法の融合の実際問題の適用例として，輸送スケジューリング問題と船舶スケジューリング問題を紹介する．

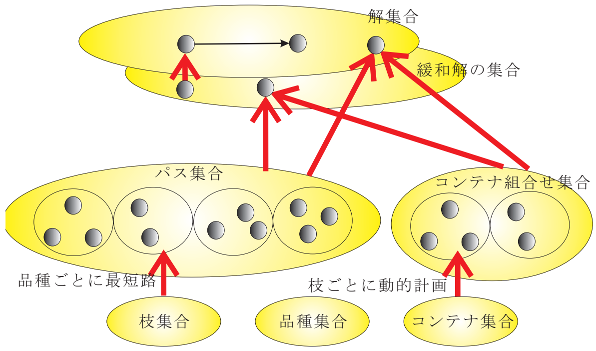

図 5　輸送スケジューリングにおける階層的積木．

5．輸送スケジューリング問題に対する応用

供給地点，中継地点，需要地点の集合が与えられている．ここで考える輸送スケジューリング問題の目的は，ある供給地点にある時刻にある商品を，ある需要地点にある時刻までに最小費用で輸送することである．同じ商品でも，異なる最早時刻や最遅時刻をもつものは，すべて異なるモノとして扱う必要がある．以下では，これを品種とよぶ．

すべての品種の集合を K とし，各 k に対して以下のデータが与えられているものとする．

s_k:　品種 k の供給地点．品種の発地を表す．

t_k:　品種 k の需要地点．品種の着地を表す．

E_k:　品種 k が供給地点 s_k を出発可能な最早時刻．

L_k:　品種 k が需要地点 t_k に到着する最遅時刻．納期を表す．

w_k:　品種 k の容積．商品マスターの 1 単位あたりの容積と需要量（輸送必要量）から算出する．重量についての制約が厳しい場合には，同様のデータを準備する．

品種を発地から着地まで分割せずに輸送する際の経路を求めるために，現在使用可能な輸送手段の下での，移動に関する費用を計算する必要がある．ここでは，地点 i,j 間の輸送は，船，航空機，トラックなどの輸送手段を用い，移動時間はダイヤによって決定される．一般には，輸送費用は輸送量に対する区分的線形な階段関数となる．特に，船舶によるコンテナ輸送では，コンテナの組合せによって費用が変化する．各枝に対して，与えられた輸送量に対して最適なコンテナの組合せを求める問題は，以下の問題になる．

(コンテナ組合せ問題)

品種のフロー量の合計 F，コンテナの集合 C，コンテナ c の容量 W_c，費用 P_c，最大数 U_c が与えられたとき，この品種をすべて積み込む最小費用のコンテナの部分集合と，積み合わせ方を求めよ．ただし，品種は複数のコンテナに分割して積み込むことができるものと仮定する．

上の問題は，使用するコンテナ c の個数を表す非負の整

数変数 x_c を用いると，以下のように定式化できる．

$$\begin{aligned}\text{minimize} \quad & \sum_{c\in C} P_c x_c \\ \text{subject to} \quad & \sum_{c\in C} W_c x_c \geq F \\ & x_c \leq U_c \qquad \forall c \in C \\ & x_c \in \mathbb{Z}_+ \qquad \forall c \in C\end{aligned}$$

これは，整数ナップサック問題と同じ構造をもつので，$\mathcal{NP}$-困難である．この問題は，整数ナップサック問題に類似のモデルだが，個数の上限がある点が特徴である．

フロー量の合計を正の整数値と仮定すると，動的計画アルゴリズムが構築できる．動的計画は，すべてのフロー量に対する最適値を一度の計算で求めることができる．これをすべての可能な（時空間ネットワークの）枝に対して「事前に」求解して保存しておくことによって，コンテナの積みあわせ相当する下層の積木が構成できる．

既にいくつかの品種の経路が決まっている状態で，新たに1つの品種を加える場合，どの経路に加えるかによってコストの増分が異なる. 時空間ネットワーク上の枝のコストを, その品種を新たに加えた場合のコストの増分に設定する. こうして時空間ネットワーク上で品種ごとの最短路問題を解くことによってパスを得ることができる．また，近傍探索は，適当な品種に対するパスを一時的に解から除き，再び（費用の差分をもとに）最短路を解くことによって解を構成することに相当する．列生成法は，動的計画アルゴリズムによって生成された区分的線形な費用関数を目的関数にもつ，多品種ネットワークフロー問題を主問題とし，時空間ネットワーク上での最短路が子問題となる．これによって，下層の積木であるパスの集合が生成される．

実証実験として，このアルゴリズムを上海から日本各地の店舗への輸送最適化に適用したところ，以下の結果を得た[4)]．CO_2 排出量を最小化する目的関数とした際には，CO_2 総排出量は 31.78% の大幅な削減となり，その際の総物流費も 2.76% の減少となった．また，費用優先の場合には，総費用は 8.39% の減少となり，その際の CO_2 総排出量は 8.06% 削減された．このように，費用最小化と CO_2 排出量は，相反するものではなく，全体最適化によって両方同時に減少させることができることが分かった．

6. 船舶スケジューリング問題に対する応用

海運会社は，荷主から輸送依頼を受けた貨物を輸送する業務を行っている. 引き受けた貨物を, 自社が管理・運用する複数の船舶（船隊とよぶ）で輸送する[3)]．全ての貨物の集合を K とし, 各 k に対して発地, 着地, 製品名, 貨物量が与えられているものとする．さらに, 発地では積み荷役開始の最早時刻と最遅時刻が, 着地では揚げ荷役開始の最遅時刻が与えられているのもとする．引き受けた貨物のことをオーダーといい, その集合のことをオーダー集合という．ここでは, 化学製品あるいは石油製品のオーダー集合を、不定期船によって輸送するための船舶スケジューリング問題を考える．5 章で述べた輸送スケジューリング問題で輸送手段として用いる船は, ダイヤによって運航時刻が決まっていた．このように，あらかじめ定められた時刻に定められた経路を繰り返し運航する船を，定期船とよぶ．これに対して，運航ごとに異なる時刻に異なる経路を運航する船を不定期船とよぶ．タンカーで化学製品や石油製品を輸送する際には, 不定期船を用いる．

これらの輸送を行う海運会社は, 荷主から一定数以上の輸送依頼を受けた時点で輸送を実施するための運航スケジュールを作成する．輸送には自社の運用する船隊を用いるが, 繁忙期にこれらのすべての貨物を処理しきれない場合には 1 航海単位で船を雇って輸送を行う．これをスポット用船という．スポット用船には費用がかかり, 1 オーダーをスポット用船で処理するための費用はあらかじめ与えらえているものとする．船舶の運航スケジュールを作成する際には, 必要に応じてスポット用船の計画も併せて作成する．海運会社が受け取る運賃はオーダーの内容によって決まっているため, 輸送コストをできるだけ低く抑えることによって, より多くの利益を確保したい．不定期船はあらかじめ定められたダイヤはもたないので, いつ, どの地点にいてもよい．したがって, 各オーダーの発地, 着地, 最早時刻, 最遅時刻, 荷量を考慮しながら運航スケジュールをうまく作成することにより, 輸送に必要な航行距離を小さくし, スポット用船の数を減らすことができる．ここで考える船舶スケジューリング問題の目的は, すべてのオーダーを, 定められた時間枠内に発地で積み, 着地に揚げることができる最小費用の船舶運航スケジュールを決めることである．

船舶による貨物輸送では, その貨物の種類によって専用の船が用いられる．例えば, コンテナの輸送にはコンテナ船が, 石油製品や化学製品の輸送にはタンカーが用いられる．それぞれの貨物の性質により, 船舶の積載方法が異なる．タンカーの貨物積載区域は, ホールドとよばれる互いに壁に隔てられた区域に分けられている．通常のタンカーには 5 つあるいは 6 つのホールドが備えられている．石油製品や化学製品は液状であり, 異なる製品は混じりあってはならず, ある貨物が積まれたホールドには, 他の貨物を積むことはできない．例えば, 容量が 300 キロリットルのホールドであれば, 積んだ貨物がわずか 50 キロリットルであったとしても, 残りの 250 キロリットル分に他の貨物を積むことはできない．したがって, ある貨物をある船舶に積むには, 貨物の積載量が船舶の最大積載量を超えないことと同時に, その貨物を割り当てることができる空きホールドが存在しなければならない．

各オーダーは, 発地から着地まで分割せずに輸送する際の経路を求めるために, そのオーダーを積むことができる

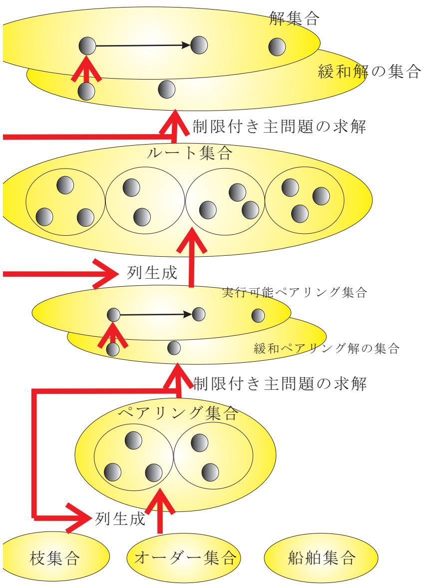

図 6　船舶スケジューリングにおける階層的積木．

船に積んだ際の移動に関する費用と時間を計算する必要がある．不定期船は固定されたダイヤは持たず，地点 i, j 間の移動時間は 2 点間の距離と船速によって決まり，輸送費用は船舶の航行距離に比例して決まる．

オーダーのうち，発地と着地，最早時刻，最遅時刻が互いに近いものは，同時に 1 隻の船に混載すると効率が良い場合がある．同時に 1 隻の船に混載するオーダーの集合をペアとよぶ．可能なペアを列挙する際に，積み付け可能性を判定するには，まず荷量の合計が船の最大積載量以下であることを調べ，次に船の各ホールドに積み付け可能であるかどうかを調べればよい．この積み付け可能性判定問題は，船のホールドの集合 H，ホールド $h \in H$ の積載量上限 U_h，オーダー $k \in K$ の荷量 D_k，オーダー k をホールド h に積載する量を表す実数変数 x_{kh}，オーダー k がホールド h に積載されるとき 1，それ以外のとき 0 の 0-1 変数 y_{kh} を用いると以下のように定式化できる．

$$
\begin{array}{lll}
\text{subject to} & \displaystyle\sum_{h \in H} x_{kh} = D_k & \forall k \in K \\
& x_{kh} \le U_h y_{kh} & \forall k \in K, h \in H \\
& \displaystyle\sum_{k \in K} y_{kh} \le 1 & \forall h \in H \\
& x_{kh} \ge 0 & \forall k \in K, h \in H \\
& y_{kh} \in \{0,1\} & \forall k \in K, h \in H
\end{array}
$$

これは実行可能性を判定するための問題なので，目的関数はどのように設定してもよい．

実行可能なペアの中から，効率の良いペアを選択する問題は，以下の問題になる．

（ペアリング問題）

ペアの集合 P，貨物の集合 K ペア p の費用 C_p，貨物 k を単独で輸送するときの費用 F_k が与えられたとき，最小費用となるペアの部分集合を求めよ．ただし，貨物は複数の船舶に分割して積み込むことはできないと仮定する．

上の問題は，ペア p を選択するとき 1，それ以外のとき 0 となる 0-1 変数 x_p，貨物 k を単独で運ぶとき 1，それ以外のとき 0 となる 0-1 変数 y_k を用いると，以下のように定式化できる．

$$
\begin{array}{lll}
\text{minimize} & \displaystyle\sum_{p \in P} C_p x_p + \sum_{k \in K} F_k y_k & \\
\text{subject to} & \displaystyle\sum_{p: k \in K_p} x_p + y_k = 1 & \forall k \in K \\
& x_p \in \{0,1\} & \forall p \in P \\
& y_k \in \{0,1\} & \forall k \in K
\end{array}
$$

これは集合分割問題であり，各 k は，選択されるペアに含まれるか，さもなければ単独で運ばれる．

実行可能なペアは多数あり得るが，その中で実際に効率的なのはその一部である．したがって，ペアの集合 P の要素は全て列挙する必要はなく，効率的なものだけを列挙すればよい．このペアリング問題を表す集合分割問題を主問題とし，以下のような子問題を考えることで，列生成法を構成できる．ペアリング問題の線形計画緩和問題を解いたときの，式

$$
\sum_{p: k \in K_p} x_p + y_k = 1
$$

に対する双対変数を λ_k とする．これは，オーダーを積み込むことに対する利益を表す．船ごとに，利益を最大にするように積み込み可能なオーダーを選択する問題を解く必要がある．そのために，オーダー k を船に積むとき 1，それ以外のとき 0 の 0-1 変数 z_k を用いると，列生成のための子問題は，以下のように書ける．

$$
\begin{array}{lll}
\text{maximize} & \displaystyle\sum_{k \in K} \lambda_k z_k & \\
\text{subject to} & \displaystyle\sum_{h \in H} x_{kh} = D_k z_k & \forall k \in K \\
& x_{kh} \le U_h y_{kh} & \forall k \in K, h \in H \\
& \displaystyle\sum_{k \in K} y_{kh} \le 1 & \forall h \in H \\
& \displaystyle\sum_{h \in H} y_{kh} \le |H| z_k & \forall k \in K \\
& x_{kh} \ge 0 & \forall k \in K, h \in H \\
& y_{kh} \in \{0,1\} & \forall k \in K, h \in H
\end{array}
$$

この子問題を解いて得られるペアの集合を保存しておくことにより，ペアの集合に相当する下層の積木が構成できる．

ペアリング問題を解くことによって，実際に用いるペアを選択したら，次にこれらのペアを用いて各船舶が運航するルートを決定する．

用いるペアの集合と，どのペアにも含まれず単独で運ぶオーダーの和集合を T と表し，その要素をタスクとよぶ．船

船が訪れるタスクの列をルートとよぶことにすると, 各船舶が運航するルートを決定する問題は, 以下の問題になる.

(ルーティング問題)

船の集合 V, タスクの集合 T, 船 $v \in V$ が実行可能なルートの集合を R_v, ルート r によって輸送されるタスクの集合 T_r, ルート r の費用 C_r, タスク $t \in T$ をスポット用船で処理するときの費用 F_t が与えられたとき, 全ての貨物を処理する最小費用の運航ルートを求めよ.

上の問題は, 船 v がルート r を選択するとき 1, それ以外のとき 0 となる 0-1 変数 x_{rv}, タスク t がスポット用船によって処理されるとき 1, それ以外のとき 0 となる 0-1 変数 y_t を用いると, 以下のように定式化できる.

$$
\begin{array}{lll}
\text{minimize} & \displaystyle\sum_{v \in V}\sum_{r \in R} C_r x_{rv} + \sum_{t \in T} F_t y_t & \\
\text{subject to} & \displaystyle\sum_{(r,v):t \in T_r} x_{rv} + y_t = 1 & \forall t \in T \\
& \displaystyle\sum_{r \in R_v} x_{rv} \le 1 & \forall v \in V \\
& x_{rv} \in \{0,1\} & \forall r \in R_v, \\
& & \forall v \in V \\
& y_t \in \{0,1\} & \forall t \in T
\end{array}
$$

スポット用船をできるだけ用いたくない場合は, F_t に大きな値を設定すればよい.

各タスクに対してノードを定義し, 船の初期位置を表すノード s と最終位置を表すノード t を定義する. このとき船の運航ルートは, 始点 s を出発し, タスクを表すいくつかのノードを訪れ終点 t に至る時空間上のパスとして表される. 船舶 v が実行可能なルートの数、すなわち R_v の要素数は膨大になるが、時空間ネットワーク上で船舶ごとの最短路問題を解くことによって効率のよいルートを得ることができる. 列生成法は, ルーティング問題、すなわち, 運航可能なルートの総航行距離に関して線形な費用関数とスポット用船費用の和を目的関数に持つ運搬スケジューリング問題を主問題とし, 時空間ネットワーク上での最短路問題を子問題とする. 子問題を解くことによって各船舶が実行可能なルートを得ることができ, これらを保存しておくことによって第 2 層の積木であるルート集合を構成することができる.

このアルゴリズムを、国内で石油製品を輸送する 2 社の運航スケジュールの最適化に適用したところ、以下の結果を得た. A 社は 7 隻のケミカルタンカーからなる船隊を運用し, 国内 25 港間で化学製品の輸送を行っている. A 社は繁忙期にはスポット用船を用いる場合がある. スポット用船で処理する貨物の数を実績以下に保つ条件を課した上で, 総航行距離を最小化する目的関数とした際には, 総航行距離は 7.9%の削減となった. また, スポット用船で処理する貨物の数を最小化する目的関数とした際には, スポット用船で処理する貨物数は 9 個から 6 個にまで減少した. この際, 総航行距離の増加は 1%未満におさまっている. どちらの目的関数を最小化した場合でも, 他方の目的関数を悪化させることなく望ましい結果が得られることが分かった. また, B 社は 14 隻からなる船隊を運用し, 国内 40 港間で, ガソリンなどの石油製品の輸送を行っている. B 社の扱う貨物では, 荷主との契約上, ペアは自明である. したがって, 第 1 層の積木であるペアリング集合の構成は必要なく, 直接第 2 層の積木であるルート集合の構成からはじめればよい. 総航行距離の最小化を目的関数としたところ, 総航行距離は 6.2%の削減となった.

なお, 船舶スケジューリングの研究の一部は、独立行政法人新エネルギー・産業技術総合開発機構 (NEDO) 先導研究開発「内航船の環境調和型運航計画支援システムの研究開発」の一部として行った.

7. おわりに

サプライ・チェインの実際問題は複雑かつ大規模であることが多い. そのため, 実用的な最適化アルゴリズムを構築する際には, 階層の概念を用いて問題を整理する方法が有効である. ここで提案した階層的積木法と列生成法の融合手法は, 整理された階層を用いて良い実行可能解を構築するものであり, 実際に 2 つの実際問題に適用され成功をおさめた. 今後は, 多くの実際問題へ適用することが課題である.

参 考 文 献

1) G. Desaulniers, J. Desrosiers, and M. M. Solomon. *Column Generation*. Gerad 25th Anniversary Series. Springer-Verlag, 2005.

2) 久保幹雄, 宮本裕一郎. メタ解法の新しいフレームワーク-階層的積木法を中心として-. 電気学会論文誌, Vol. 121-C, No. 6, pp. 976–981, 2001.

3) 小林和博, 久保幹雄, 加納敏幸. 内航ケミカルタンカーに対する配船計画問題. 平成 19 年日本船舶海洋工学会秋季講演会, pp. 7–8, 2007.

4) 大西真人. 船・飛行機のダイヤを考慮した国際物流経路最適化問題の解法. 第 19 回 RAMP シンポジウム, pp. 151–161, 2007.

(2008 年 1 月 1 日受付)