《論　文》

強い強制制約式を用いた容量制約のないネットワークデザイン問題のLagrange緩和法

片　山　直　登

1　はじめに

容量制約のないネットワークデザイン問題（Uncapacitated Network Design Problem: *UND*）は，アーク上の容量の制約を考慮しないネットワークにおいて，適切なネットワークの形状と多品種のフローを求める問題であり，通信ネットワーク設計，交通ネットワーク設計や輸送・配送ネットワーク設計などに様々な応用分野が存在する基本的な問題である。

ネットワークはノードとアークで構成され，ネットワーク上に品種のフローが発生する。ノードは交換機・コンピュータ，交差点・交通需要発生吸収点や配送センター・消費地などを表し，アークは通信回線，道路や輸送経路などのノード間をつなぐものを表す。また，フローは，呼，データ，自動車や荷物などのノード間を移動するものを表す。ここでは，一般的な表現とは異なり，始点・終点の異なるものを異なる品種と定義する。本研究では，多品種を対象とする。そのため，多品種とは，始点と終点の組合せが異なる複数種類の運ぶべきモノが存在することを意味する。

ノード集合，デザイン費用とフロー費用をもつ容量制約と向きをもたないアーク集合，需要をもつ品種集合が与えられているものとする。このとき，*UND*は，フロー費用とデザイン費用の合計を最小にするアーク集合を選択し，各品種のフローを求める問題である。

*UND*に対しては，数多くの最適解法や近似解法が示されている。最適解法としては，Los-Lardinois［8］による分枝限定法，Magnanti-Mireault-Wong［9］によるベンダーズ分解原理，Holmberg-Hellstrand［3］によるLagrange緩和と分枝限定法を組み合せた解法などがある。貪欲的な近似解法としては，バックワード法が数多く提案されている。Billheimer-Gray［2］はアーク削除時の目的関数値の変化量を

事前に近似的計算する方法，Los-Lardinois［8］はアーク削除時の目的関数値の変化量を厳密に求め，単位アーク費用当たりの変化量による方法を示している。また，Minoux［11］は初期のアーク削除時の目的関数値の近似的な変化量をリストに保持し，削除対象時に再計算した変化量がリスト中で最大の場合のみ削除する方法，片山［5］は目的関数値の変化量を厳密に計算したMinouxの改良法を示している。緩和問題の解法と緩和解を用いた近似解法としては，Balakrishnan-Magnanti-Wong［1］による双対上昇法と双対ヒューリスティクス，片山ら［4］によるLagrange緩和法とLagrangeヒューリスティクスがある。また，多くのサーベイ［10］［12］［7］が提供されている。

定式化や妥当不等式の面では，Balakrishnan-Magnanti-Wong［1］やHolmberg-Hellstrand［3］は，品種毎に非集約された強制制約式を用いている。Balakrishnan-Magnanti-Wongは，さらに始点を同じ，または終点を同じにする品種が同一アーク上で逆流しないことを用いた強制制約式を示し，これを用いた双対上昇法を示唆している。片山［6］は，同一ノードを始点とする品種と終点とする品種のフローが同一のアーク上で同じ向きに流れないことを用いた強制制約式を示し，これを用いた列生成法を示している。

本研究では，*UND*に対して，Balakrishnan-Magnanti-Wongおよび片山が示した強い妥当不等式による定式化に対して，これらをLagrange緩和した問題を示し，この緩和問題の解法を提案する。また，数値実験により，提案する解法が従来の解法よりも精度が高い解を算出できることを示す。

2 *UND*の定式化

ノード集合をN，向きをもたないアーク集合をAと表す。ノードoを始点，dを終点とする品種をodと表す。ノードi，j間のアーク(i,j)を設置するか否かを表す0-1のデザイン変数をy_{ij}とし，アーク(i,j)上を$i \rightarrow j$方向に流れる品種odのフロー量を表すアークフロー変数をx_{ij}^{od}とする。アーク(i,j)を選択するときに発生するデザイン費用をf_{ij}，品種kの需要がアーク上を$i \rightarrow j$方向に通るときに発生するフロー費用をc_{ij}^{od}とする。ノードo，d間に需要があるとき1，そうでないとき0である品種odの需要をd^{od}とする。ただし，$d^{oo}=0$（$o \in N$）とする。ここでは，終点（または始点）を同じにする品種に対してフロー費用が等しいか比例などの場合を想定する。さらに，ネットワークが連結している必要があるものとし，Yをネットワークが連結するようなデザイン変数の集合とする。

このとき，アークフロー変数を用いた*UND*の定式化は，次のように表される［1］。

$$\text{最小化}\quad \sum_{(i,j)\in A}\sum_{o\in N}\sum_{d\in N}(c_{ij}^{od}x_{ij}^{od}+c_{ji}^{od}x_{ji}^{od})+\sum_{(i,j)\in A} f_{ij}y_{ij} \tag{1}$$

$$\text{条件}\quad \sum_{i\in N_n} x_{in}^{od}-\sum_{j\in N_n} x_{nj}^{od}=\begin{cases}-d^{od} & if\ n=o\\ d^{od} & if\ n=d\\ 0 & otherwise\end{cases}\quad \forall n\in N, o\in N, d\in N \tag{2}$$

$$x_{ij}^{od}+x_{ji}^{od}\le d^{od}y_{ij}\quad \forall o\in N, d\in N, (i,j)\in A \tag{3}$$

$$x_{ij}^{od}\ge 0,\ x_{ji}^{od}\ge 0\quad \forall o\in N, d\in N, (i,j)\in A \tag{4}$$

$$y_{ij}\in\{0,1\}\quad \forall (i,j)\in A \tag{5}$$

$$\boldsymbol{y}\in Y \tag{6}$$

(1)式は目的関数であり，フロー費用とデザイン費用の総和を最小化する。(2)式はフロー保存式であり，品種の需要が始点から終点に移動することを保証する。(3)式は，アークが存在するときは同じ品種のフローが同一のアーク上で互いに逆流しないことを表し，アークが存在しないときはフローが0であることを表す強制制約式である。(4)式はアークフロー変数の非負制約であり，(5)式はデザイン変数の0-1条件である。(6)式は，ネットワークが連結することを表す。

$d^{od}=0$ の場合，$x_{ij}^{od}=0$ $(i,j\in A)$ となるため，この od に関係する(2)～(4)式は不要である。しかし，ここでは記述を簡単にするために，$d^{od}=0$ の場合を含めた定式化を用いる。

問題の定式化に有効な妥当不等式を追加すると実行可能領域が狭くなり，緩和問題などを使って問題を解く場合に，妥当不等式が有効に機能する。そこで，*UND*に対する有効な妥当不等式を示す。

ここでは，終点（または始点）を同じにする品種に対して，フロー費用が等しいか比例などの場合を想定する。この場合，終点（または始点）を同じにする品種のフローが同一のアーク上で互いに逆流しない最適解が存在する。このことから，Balakrishnan-Magnanti-Wong［1］は，次の妥当な強制制約式を示している。

$$x_{ij}^{od}+x_{ji}^{on}\le d_{od}^{on}y_{ij}\quad \forall o\in N, d\in N, n\in N, (i,j)\in A \tag{7}$$

$$x_{ij}^{od}+x_{ji}^{nd}\le d_{od}^{nd}y_{ij}\quad \forall o\in N, d\in N, n\in N, (i,j)\in A \tag{8}$$

ここで，$d_{on}^{md}=\max\,(d^{on}, d^{md})$ である。また，(7)式，(8)式は(3)式を含んでいるため，(3)式は省略できる。

一方，同一ノードを始点とする品種と終点とする品種のフローが同一のアーク上で同じ向きに流れるフローが存在するとき，同じ向きに流れないような実行可能で，かつフロー費用が安いフローが存在する。このため，最適解の一つでは，同一ノードを始点とする品種と終点とする品種のフローが同一のアーク上で同じ向きに流れないことになる。

したがって，次式も妥当な強制制約式となる［6］。

$$x_{ij}^{on}+x_{ij}^{nd}\le d_{on}^{nd}y_{ij}\quad \forall o\in N,d\in N,n\in N,(i,j)\in A \tag{9}$$

$$x_{ji}^{on}+x_{ji}^{nd}\le d_{on}^{nd}y_{ij}\quad \forall o\in N,d\in N,n\in N,(i,j)\in A \tag{10}$$

3　Lagrange緩和問題

前節のUNDの定式化は膨大な数の0-1変数と制約式を含む組合せ最適化問題となるため，直接解くことは困難である。(3)式は取り除き，強制制約式としては(7)式を残し，(2)式および(8)～(10)式をLagrange緩和した問題を考え，この緩和問題を解くことにする。一般的に緩和問題の最適解は制約式を緩和しているため，UNDの実行可能解とはならない。また，緩和問題の最適値はUNDの最適値ではなく下界値となる。

(2)式に対するLagrange乗数を$\boldsymbol{v}$，(8)，(9)，(10)式に対する非負のLagrange乗数を$\boldsymbol{t}$，$\boldsymbol{u}$，$\boldsymbol{w}$として，Lagrange緩和問題LGを作成する。

はじめに，LGの目的関数zを整理する。

$$\begin{aligned}
z=&\sum_{(i,j)\in A}\sum_{o\in N}\sum_{d\in N}(c_{ij}^{od}x_{ij}^{od}+c_{ji}^{od}x_{ji}^{od})+\sum_{(i,j)\in A}f_{ij}y_{ij}\\
&+\sum_{o\in N}\sum_{d\in N}v_o^{od}\Big(-d^{od}-\sum_{i\in N_o}x_{io}^{od}+\sum_{j\in N_o}x_{oj}^{od}\Big)+\sum_{o\in N}\sum_{d\in N}v_d^{od}\Big(d^{od}-\sum_{i\in N_d}x_{id}^{od}+\sum_{j\in N_d}x_{dj}^{od}\Big)\\
&+\sum_{o\in N}\sum_{d\in N}\sum_{n\in N\backslash\{o,d\}}v_n^{od}\Big(-\sum_{i\in N_n}x_{in}^{od}+\sum_{j\in N_n}x_{nj}^{od}\Big)+\sum_{(i,j)\in A}\sum_{n\in N}\sum_{o\in N}\sum_{d\in N}t_{ij}^{ond}\Big(x_{ij}^{od}+x_{ji}^{nd}-d_{od}^{nd}y_{ij}\Big)\\
&+\sum_{(i,j)\in A}\sum_{n\in N}\sum_{o\in N}\sum_{d\in N}u_{ij}^{ond}\Big(x_{ij}^{on}+x_{ij}^{nd}-d_{on}^{nd}y_{ij}\Big)+\sum_{(i,j)\in A}\sum_{n\in N}\sum_{o\in N}\sum_{d\in N}w_{ij}^{ond}\Big(x_{ji}^{on}+x_{ji}^{nd}-d_{on}^{nd}y_{ij}\Big)
\end{aligned}$$

$$\begin{aligned}
=&\sum_{o\in N}\sum_{d\in N}d^{od}(v_d^{od}-v_o^{od})+\sum_{(i,j)\in A}\sum_{o\in N}\sum_{d\in N}\Big\{(c_{ij}^{od}-v_j^{od}+v_i^{od})x_{ij}^{od}+(c_{ji}^{od}-v_i^{od}+v_j^{od})x_{ji}^{od}\Big\}\\
&+\sum_{(i,j)\in A}\sum_{n\in N}\sum_{o\in N}\sum_{d\in N}\Big(t_{ij}^{ond}x_{ij}^{od}+t_{ij}^{nod}x_{ji}^{od}+u_{ij}^{odn}x_{ij}^{od}+u_{ij}^{nod}x_{ij}^{od}+w_{ij}^{odn}x_{ji}^{od}+w_{ij}^{nod}x_{ji}^{od}\Big)\\
&\qquad+\sum_{(i,j)\in A}\Big\{f_{ij}-\sum_{n\in N}\sum_{o\in N}\sum_{d\in N}\Big(d_{od}^{nd}t_{ij}^{ond}+d_{on}^{nd}u_{ij}^{ond}+d_{on}^{nd}w_{ij}^{ond}\Big)\Big\}y_{ij}
\end{aligned}$$

$$\begin{aligned}
=&\sum_{o\in N}\sum_{d\in N}d^{od}(v_d^{od}-v_o^{od})+\sum_{(i,j)\in A}\sum_{o\in N}\sum_{d\in N}\Big[\Big\{c_{ij}^{od}-v_j^{od}+v_i^{od}+\sum_{n\in N}\Big(t_{ij}^{ond}+u_{ij}^{odn}+u_{ij}^{nod}\Big)\Big\}x_{ij}^{od}\\
&\qquad+\Big\{c_{ji}^{od}-v_i^{od}+v_j^{od}+\sum_{n\in N}\Big(t_{ij}^{nod}+w_{ij}^{odn}+w_{ij}^{nod}\Big)\Big\}x_{ji}^{od}\Big]\\
&\qquad+\sum_{(i,j)\in A}\Big\{f_{ij}-\sum_{n\in N}\sum_{o\in N}\sum_{d\in N}\Big(d_{od}^{nd}t_{ij}^{ond}+d_{on}^{nd}u_{ij}^{ond}+d_{on}^{nd}w_{ij}^{ond}\Big)\Big\}y_{ij}
\end{aligned}$$

したがって緩和問題LGは次のように表わされる。

$$
\begin{aligned}
\text{最小化}\ & \sum_{o\in N}\sum_{d\in N} d^{od}(v_d^{od}-v_o^{od}) + \sum_{(i,j)\in A}\sum_{o\in N}\sum_{d\in N}(\alpha_{ij}^{od}x_{ij}^{od}+\beta_{ij}^{od}x_{ji}^{od}) + \sum_{(i,j)\in A}(f_{ij}-\gamma_{ij})y_{ij} \\
\text{条件}\ & x_{ij}^{od}+x_{ji}^{on} \le d_{od}^{on}y_{ij} \quad \forall o\in N, d\in N, n\in N, (i,j)\in A \\
& x_{ij}^{od}\ge 0,\ x_{ji}^{od}\ge 0 \quad \forall o\in N, d\in N, (i,j)\in A \\
& y_{ij}\in\{0,1\} \quad \forall (i,j)\in A \\
& \boldsymbol{y}\in Y
\end{aligned}
$$

ここで,

$$
\begin{aligned}
\alpha_{ij}^{od} &= c_{ij}^{od} - v_j^{od} + v_i^{od} + \sum_{n\in N}(t_{ij}^{ond}+u_{ij}^{odn}+u_{ij}^{nod}) \\
\beta_{ij}^{od} &= c_{ji}^{od} - v_i^{od} + v_j^{od} + \sum_{n\in N}(t_{ij}^{nod}+w_{ij}^{odn}+w_{ij}^{nod}) \\
\gamma_{ij} &= \sum_{n\in N}\sum_{o\in N}\sum_{d\in N}(d_{od}^{nd}t_{ij}^{ond}+d_{on}^{nd}u_{ij}^{ond}+d_{on}^{nd}w_{ij}^{ond})
\end{aligned}
$$

である。

はじめに，連結条件である(6)式を取り除いた問題LG'を考える。Lagrange乗数が与えられたとき，目的関数の第一項は定数と見なすことができるので，LG'は次のようなアーク（i，j）毎の問題LG_{ij}に分割することができる。

$$
\begin{aligned}
\text{最小化}\ & \sum_{o\in N}\sum_{d\in N}(\alpha_{ij}^{od}x_{ij}^{od}+\beta_{ij}^{od}x_{ji}^{od}) + (f_{ij}-\gamma_{ij})y_{ij} \\
\text{条件}\ & x_{ij}^{od}+x_{ji}^{on}\le d_{od}^{on}y_{ij} \quad \forall o\in N, d\in N, n\in N \\
& x_{ij}^{od}\ge 0,\ x_{ji}^{od}\ge 0 \quad \forall o\in N, d\in N \\
& y_{ij}\in\{0,1\}
\end{aligned}
$$

ここで，$y_{ij}=1$である場合と$y_{ij}=0$である場合に分けて考える。$y_{ij}=1$である場合，$f_{ij}-\gamma_{ij}$は定数項であるので，LG_{ij}は次のような品種の始点であるノードo毎の独立した問題LG_{ij}^{o}に分割できる。

$$
\begin{aligned}
\text{最小化}\ & \sum_{d\in N}(\alpha_{ij}^{od}x_{ij}^{od}+\beta_{ij}^{od}x_{ji}^{od}) \\
\text{条件}\ & x_{ij}^{od}+x_{ji}^{on}\le d_{od}^{on} \quad \forall d\in N, n\in N \\
& x_{ij}^{od}\ge 0,\ x_{ji}^{od}\ge 0 \quad \forall d\in N
\end{aligned}
$$

LG_{ij}^{o}は，上限制約のみをもつ目的関数を最小化する線形計画問題である。そのため，最適解において$x_{ij}^{od}>0$であるためには，その係数が負であることが必要である。また，制約式は$i\rightarrow j$と$j\rightarrow i$方向のどちらか高々一方向のフローの存在のみを許すことを表している。$i\rightarrow j$方向のフローが存在するためには$\sum_{d\in N}\min(0,\alpha_{ij}^{od})\le\sum_{d\in N}\min(0,\beta_{ij}^{od})$であることが必要であり，$j\rightarrow i$方向のフローが存在するためには$\sum_{d\in N}\min(0,\beta_{ij}^{od})<\sum_{d\in N}\min(0,\alpha_{ij}^{od})$であることが必要である。したがって，この問題の最適解$\bar{\boldsymbol{x}}$は，次のようになる。

$$\tilde{x}_{ij}^{od} = \begin{cases} d^{od} & if\ \alpha_{ij}^{od} < 0\ and\ \sum_{l \in N} \min(0, \alpha_{ij}^{ol}) \leq \sum_{l \in N} \min(0, \beta_{ij}^{ol}) \\ 0 & otherwise \end{cases}$$

$$\tilde{x}_{ji}^{od} = \begin{cases} d^{od} & if\ \beta_{ij}^{od} < 0\ and\ \sum_{l \in N} \min(0, \beta_{ij}^{ol}) < \sum_{l \in N} \min(0, \alpha_{ij}^{ol}) \\ 0 & otherwise \end{cases}$$

一方，$y_{ij} = 0$である場合，LG_{ij}の最適値は0である。したがって，LG_{ij}はアークフロー変数を用いない次のような問題に置き換えることができる。

$$\begin{aligned} &\text{最小化} \quad \Big[\sum_{o \in N} \Big\{ \sum_{d \in N} \min(0, \alpha_{ij}^{od}), \sum_{d \in N} \min(0, \beta_{ij}^{od}) \Big\} + f_{ij} - \gamma_{ij} \Big] y_{ij} \\ &\text{条件} \quad y_{ij} \in \{0, 1\} \end{aligned}$$

LG_{ij}の最適解$\tilde{y}_{ij}$は，次のようになる。

$$\tilde{y}_{ij} = \begin{cases} 1 & if\ \sum_{o \in N} \Big\{ \sum_{d \in N} \min(0, \alpha_{ij}^{od}), \sum_{d \in N} \min(0, \beta_{ij}^{od}) \Big\} + f_{ij} - \gamma_{ij} < 0 \\ 0 & otherwise \end{cases}$$

連結条件である(6)式も考慮すると，LGは次のような問題MTに帰着することができる。

$$\text{最小化} \quad \sum_{(i,j) \in A} \Big[\sum_{o \in N} \Big\{ \sum_{d \in N} \min(0, \alpha_{ij}^{od}), \sum_{d \in N} \min(0, \beta_{ij}^{od}) \Big\} + f_{ij} - \gamma_{ij} \Big] y_{ij} \tag{11}$$

$$\text{条件} \quad y_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \tag{12}$$

$$\boldsymbol{y} \in Y$$

MTはデザイン変数の係数をアークの重みとした最小木問題であり，クラスカル法などで解くことができる。

MTの最適解を$\hat{\boldsymbol{y}}$とすると，$\hat{\boldsymbol{y}}$はLGの最適解となる。また，LGの最適解$\hat{\boldsymbol{x}}$は次のようになる。

$$\hat{x}_{ij}^{od} = \begin{cases} \tilde{x}_{ij}^{od} & if\ \hat{y}_{ij} = 1 \\ 0 & otherwise \end{cases}, \quad \hat{x}_{ji}^{od} = \begin{cases} \tilde{x}_{ji}^{od} & if\ \hat{y}_{ij} = 1 \\ 0 & otherwise \end{cases}$$

$\hat{\boldsymbol{y}}$，$\hat{\boldsymbol{x}}$とLagrange乗数をLGの目的関数に代入することによってLGの最適値，すなわちUNDの下界値を求めることができる。

4 Lagrange乗数の設定

LGの最適値はUNDの下界値であり，この下界値を最大にするようにLagrange乗数を設定することが必要である。LGの最適解$\hat{\boldsymbol{y}}$，$\hat{\boldsymbol{x}}$が求められているとき，Lagrange乗数$\boldsymbol{v}$，$\boldsymbol{t}$，$\boldsymbol{u}$，$\boldsymbol{w}$に関する最適化問題LDは次のようになる。

$$
\begin{aligned}
\text{最小化}\ & \sum_{(i,j)\in A}\sum_{o\in N}\sum_{d\in N}(c_{ij}^{od}\hat{x}_{ij}^{od}+c_{ji}^{od}\hat{x}_{ji}^{od})+\sum_{(i,j)\in A}f_{ij}\hat{y}_{ij}\\
& \sum_{o\in N}\sum_{d\in N}v_o^{od}\Big(-d^{od}-\sum_{i\in N_o}\hat{x}_{io}^{od}+\sum_{j\in N_o}\hat{x}_{oj}^{od}\Big)+\sum_{o\in N}\sum_{d\in N}v_d^{od}\Big(d^{od}-\sum_{i\in N_d}\hat{x}_{id}^{od}+\sum_{j\in N_d}\hat{x}_{dj}^{od}\Big)\\
& +\sum_{o\in N}\sum_{d\in N}\sum_{n\in N\setminus\{o,d\}}v_n^{od}\Big(-\sum_{i\in N_n}\hat{x}_{in}^{od}+\sum_{j\in N_n}\hat{x}_{nj}^{od}\Big)+\sum_{(i,j)\in A}\sum_{n\in N}\sum_{o\in N}\sum_{d\in N}t_{ij}^{ond}\Big(\hat{x}_{ij}^{od}+\hat{x}_{ji}^{nd}-d_{od}^{nd}\hat{y}_{ij}\Big)\\
& +\sum_{(i,j)\in A}\sum_{n\in N}\sum_{o\in N}\sum_{d\in N}u_{ij}^{ond}\Big(\hat{x}_{ij}^{on}+\hat{x}_{ij}^{nd}-d_{on}^{nd}\hat{y}_{ij}\Big)+\sum_{(i,j)\in A}\sum_{n\in N}\sum_{o\in N}\sum_{d\in N}w_{ij}^{ond}\Big(\hat{x}_{ji}^{on}+\hat{x}_{ji}^{nd}-d_{on}^{nd}\hat{y}_{ij}\Big)\\
\text{条件}\ & t_{ij}^{ond}\geq 0\quad \forall o\in N,d\in N,n\in N,(i,j)\in A\\
& u_{ij}^{ond}\geq 0\quad \forall o\in N,d\in N,n\in N,(i,j)\in A\\
& w_{ij}^{ond}\geq 0\quad \forall o\in N,d\in N,n\in N,(i,j)\in A
\end{aligned}
$$

$\hat{\boldsymbol{y}}$，$\hat{\boldsymbol{x}}$は組合せ最適化問題の解であり，Lagrange乗数が変化したときに離散的に変化するため，LDの目的関数値も離散的に変化する。このためLDは微分不可能な目的関数をもつ最適化問題となり，微分不可能な問題に対する解法が必要となる。

$\boldsymbol{v}$に対しては，劣勾配法を適用する。LDの目的関数は$\boldsymbol{v}$の1次関数であることから，v_n^{od}に関する劣勾配s_n^{od}をv_n^{od}の係数，すなわち，

$$
s_n^{od}=\begin{cases}-d^{od}-\sum_{i\in N_n}\hat{x}_{in}^{od}+\sum_{j\in N_n}\hat{x}_{nj}^{od} & if\ n=o\\ d^{od}-\sum_{i\in N_n}\hat{x}_{in}^{od}+\sum_{j\in N_n}\hat{x}_{nj}^{od} & if\ n=d\\ -\sum_{i\in N_n}\hat{x}_{in}^{od}+\sum_{j\in N_n}\hat{x}_{nj}^{od} & otherwise\end{cases}
$$

とする。

一方，$\boldsymbol{t}$，$\boldsymbol{u}$，$\boldsymbol{w}$はO（$|N|^5$）個と多数存在するため，すべてを陽的に考慮することは困難である。そのため，逐次，$\hat{\boldsymbol{y}}$，$\hat{\boldsymbol{x}}$が満足しない(8)〜(10)式，すなわち，

$$\hat{x}_{ij}^{od}+\hat{x}_{ji}^{nd}>d_{od}^{nd}\hat{y}_{ij}\tag{13}$$

$$\hat{x}_{ij}^{on}+\hat{x}_{ij}^{nd}>d_{on}^{nd}\hat{y}_{ij}\tag{14}$$

$$\hat{x}_{ji}^{on}+\hat{x}_{ji}^{nd}>d_{on}^{nd}\hat{y}_{ij}\tag{15}$$

であり，かつ$d_{od}^{nd}=1$または$d_{on}^{nd}=1$である制約式に対してLagrange乗数を生成し，生成した乗数に対して劣勾配法を適用する。

(13)式が成り立つのは，$\hat{\boldsymbol{y}}_{ij}=1$かつ$\hat{\boldsymbol{x}}_{ij}^{od}=\hat{\boldsymbol{x}}_{ji}^{nd}=1$の場合に限られる。そこで，

$$
a_{ij}^o=\sum_{d\in N}\min(0,\alpha_{ij}^{od}),\quad b_{ij}^o=\sum_{d\in N}\min(0,\beta_{ij}^{od})
$$

とおくと，$\hat{\boldsymbol{x}}_{ij}^{od}=1$であるためには，$\alpha_{ij}^{od}<0$かつ$a_{ij}^o-b_{ij}^o<0$であることが必要となる。また，$\hat{\boldsymbol{x}}_{ji}^{nd}=1$であるためには，$\beta_{ij}^{nd}<0$かつ$b_{ij}^n-a_{ij}^n<0$であることが必要となる。したがって，(13)式が成り立つためには，

$$\hat{y}_{ij}=1\ and\ \min\{0,\max(\alpha_{ij}^{od},\beta_{ij}^{nd},a_{ij}^{o}-b_{ij}^{o},b_{ij}^{n}-a_{ij}^{n})\}<0 \tag{16}$$

であることが必要となる。そこで，LGの最適解$\hat{\boldsymbol{y}}$，$\hat{\boldsymbol{x}}$が求められたとき，(16)式を満足する（i, j, o, n, d）に対する強制制約式のLagrange乗数t_{ij}^{ond}を生成し，その初期値を次のように設定する。

$$t_{ij}^{ond}=-\min\{0,\max(\alpha_{ij}^{od},\beta_{ij}^{nd},a_{ij}^{o}-b_{ij}^{o},b_{ij}^{n}-a_{ij}^{n})\}$$

このとき，t_{ij}^{ond}が関係するα_{ij}^{od}，β_{ij}^{nd}，a_{ij}^{o}，b_{ij}^{o}が変化するため，

$$\alpha_{ij}^{od}:=\alpha_{ij}^{od}-t_{ij}^{ond},\ \beta_{ij}^{nd}:=\beta_{ij}^{nd}-t_{ij}^{ond}$$
$$a_{ij}^{o}:=a_{ij}^{o}-t_{ij}^{ond},\ b_{ij}^{n}:=b_{ij}^{n}-t_{ij}^{ond}$$

と更新する。

同様に，(14)式，(15)式が成り立つためには，次式が成り立つことが必要となる

$$\hat{y}_{ij}=1\ and\ \min\{0,\max(\alpha_{ij}^{od},\alpha_{ij}^{nd},a_{ij}^{o}-b_{ij}^{o},a_{ij}^{n}-b_{ij}^{n})\}<0 \tag{17}$$
$$\hat{y}_{ij}=1\ and\ \min\{0,\max(\beta_{ij}^{od},\beta_{ij}^{nd},b_{ij}^{o}-a_{ij}^{o},b_{ij}^{n}-a_{ij}^{n})\}<0 \tag{18}$$

そこで，(17)式を満足する（i, j, o, n, d）に対する強制制約式のLagrange乗数u_{ij}^{ond}，(18)式を満足する（i, j, o, n, d）に対する強制制約式のLagrange乗数w_{ij}^{ond}を生成し，これらの初期値を

$$u_{ij}^{ond}=-\min\{0,\max(\alpha_{ij}^{od},\alpha_{ij}^{nd},a_{ij}^{o}-b_{ij}^{o},a_{ij}^{n}-b_{ij}^{n})\}$$
$$w_{ij}^{ond}=-\min\{0,\max(\beta_{ij}^{od},\beta_{ij}^{nd},b_{ij}^{o}-a_{ij}^{o},b_{ij}^{n}-a_{ij}^{n})\}$$

とし，次のように更新する。

$$\alpha_{ij}^{od}:=\alpha_{ij}^{od}-u_{ij}^{ond},\alpha_{ij}^{nd}:=\alpha_{ij}^{nd}-u_{ij}^{ond}$$
$$a_{ij}^{o}:=a_{ij}^{o}-u_{ij}^{ond},\ a_{ij}^{n}:=a_{ij}^{n}-u_{ij}^{ond}$$
$$\beta_{ij}^{od}:=\beta_{ij}^{od}-w_{ij}^{ond},\ \beta_{ij}^{nd}:=\beta_{ij}^{nd}-w_{ij}^{ond}$$
$$b_{ij}^{o}:=b_{ij}^{o}-w_{ij}^{ond},\ b_{ij}^{n}:=b_{ij}^{n}-w_{ij}^{ond}$$

Lagrange乗数$\boldsymbol{t}$，$\boldsymbol{u}$，$\boldsymbol{w}$に対する劣勾配は次式となる。

$$p_{ij}^{ond}=d_{od}^{nd}y_{ij}-x_{ij}^{od}-x_{ji}^{nd}$$
$$q_{ij}^{ond}=d_{od}^{nd}y_{ij}-x_{ij}^{on}-x_{ij}^{nd}$$
$$r_{ij}^{ond}=d_{od}^{nd}y_{ij}-x_{ji}^{on}-x_{ji}^{nd}$$

LDの目的関数値を改善する可能性があるように，劣勾配を用いて次のように乗数を更新する。

$$
\begin{aligned}
v_n^{od} &:= v_n^{od} + \theta^l s_n^{od} \\
t_{ij}^{ond} &:= \max(0, t_{ij}^{ond} + \theta^l p_{ij}^{ond}) \\
u_{ij}^{ond} &:= \max(0, u_{ij}^{ond} + \theta^l q_{ij}^{ond}) \\
w_{ij}^{ond} &:= \max(0, w_{ij}^{ond} + \theta^l r_{ij}^{ond})
\end{aligned}
$$

ここで，θ^lはl回目の繰り返しにおけるステップサイズである。適当なUNDの上界値をUB，l回目の繰り返しにおけるLGの最適値であるUNDの下界値をLB^l，パラメータをρ（$0<\rho<2$）としたときに，θ^lは次式で与えられる。

$$
\theta^l = \frac{\rho(UB - LB^l)}{\sum_{o\in N}\sum_{d\in N}\sum_{n\in N}\left\{(v_n^{od})^2 + \sum_{(i,j)\in A}((t_{ij}^{ond})^2 + (u_{ij}^{ond})^2 + (w_{ij}^{ond})^2)\right\}}
$$

上界値UBが緩和問題の最適目的関数値に一致し，$\lim_{k\to\infty}\theta^l \to 0$かつ$\sum_{l=1}^{\infty}\theta^l \to \infty$であれば，Lagrange乗数は最適値に収束する。

5　Lagrange緩和法

Lagrange緩和法の流れをまとめておく。

Lagrange緩和法

［ステップ１］適当な$\boldsymbol{v}$の初期値を与える。繰返し回数の上限をl_{max}，収束判定基準をεとする。$\boldsymbol{t}$，$\boldsymbol{u}$，$\boldsymbol{w}$の生成を開始する繰返し回数をl_d，生成する周期をl_{int}とする。上界値をUB，下界値をLBとする。$UB := \infty, LB := 0$，$l := 1$とする。

［ステップ２］LGを解き，l回目の下界値LB^l，および最適解$\hat{\boldsymbol{y}}$，$\hat{\boldsymbol{x}}$を求める。$LB^l > LB$であれば$LB := LB^l$とする。

［ステップ３］$l \geq l_d$かつ$l \bmod l_{int} = 0$であれば，(16)〜(18)式を初めて満足する（i, j, o, n, d）に対する強制制約式のLagrange乗数$\boldsymbol{t}$，$\boldsymbol{u}$，$\boldsymbol{w}$を生成する。

［ステップ４］$\hat{\boldsymbol{y}}$を用いて適当なLagrangeヒューリスティックを行い，l回目の上界値UB^lを求める。$UB^l < UB$であれば$UB := UB^l$とする。

［ステップ５］$l = l_{max}$または（$UB - LB)/LB < \varepsilon$であれば終了する。

［ステップ６］$\hat{\boldsymbol{y}}$および$\hat{\boldsymbol{x}}$より劣勾配$\boldsymbol{s}$，$\boldsymbol{p}$，$\boldsymbol{q}$，$\boldsymbol{r}$を求め，劣勾配法により$\boldsymbol{v}$，$\boldsymbol{t}$，$\boldsymbol{u}$，$\boldsymbol{w}$を更新する。$l := l + 1$として，ステップ２へ戻る。

ステップ４におけるLagrangeヒューリスティックは，文献［４］［５］の方法を用いることができる。

繰返し毎に，(16)〜(18)式を満足する（i, j, o, n, d）に対する強制制約式のLagrange乗数をすべて生成してしまうと，膨大な数の乗数となる可能性がある。そこで，ステップ3において，l_dとl_{int}を用いて，新たに生成するLagrange乗数の数を制御し，時間計算量と空間計算量を抑えている。

6　数値実験

数値実験によって，従来の解法と本解法による下界値の比較を行った。アーク上のフロー費用とデザイン費用が比例するようなユークリッド平面上の問題例（同一ノード数各5問）を用いる。比較のため，Holmberg-Hellstrand［3］および片山ら［4］の解法でも計算を行った。ただし，Holmberg-Hellstrandの対象とする問題は向きをもつアークを対象としたものである。

主な条件およびパラメータは次の通りである。

・ノード数：10〜100。ノードは1000×1000のユークリッド平面上にランダムに発生させる。
・アーク候補数：ノード数の5倍。ノード間距離の短いものをアーク候補とする。ただし，2ノード間の距離を重みとした最小木に含まれるアークを含むものとする。
・品種数：ノード数の5倍。品種（始点・終点の対）をランダムに選定する。ただし，各ノードを始点または終点とする品種が少なくとも1品種は存在するものとする。
・デザイン費用：ノード間のユークリッド距離に比例
・フロー費用：ノード間のユークリッド距離に比例
・デザイン費用／フロー費用の比：10，20，30
・使用計算機：CPU Pen4 3.2GHz，OS WindowsXP
・使用言語：Intel FORTRAN Ver9.1
・劣勾配法の繰返し回数：Holmberg-Hellstrand・片山ら10000回，本解法15000回
・Lagrange乗数の生成を開始する繰返し回数l_d：1800回
・Lagrange乗数を生成する周期l_{int}：100回
・収束判定基準ε：0.01％
・下界値の誤差：（上界値－下界値）／下界値×100の平均値。上界値は片山ら［4］，片山［5］の解法，および数理計画ソフトCPLEXの分枝限定法（最大10時間）によって得られた上界値の中での最良値を用いる。なお，ノード数30まではCPLEXにより最適値が求められており，これを上界値とする。

表1〜3に，従来の解法と本研究の解法による下界値の平均誤差と平均計算時間を示す。ノード数30までの上界値は最適値であるため，平均誤差は最適値と下界値との差で

表１：下界値の比較（フロー費用／デザイン費用＝10）

ノード数	Holmberg-Hellstrand		片山ら		本研究	
	誤差（%）	計算時間（s）	誤差（%）	計算時間（s）	誤差（%）	計算時間（s）
10	0.50	1	0.16	1	0.00	2
20	0.85	8	0.48	12	0.23	31
30	1.42	21	0.97	31	0.69	99
40	1.66	43	1.21	63	0.90	189
50	1.84	72	1.40	109	1.17	296
60	2.03	113	1.66	173	1.43	457
70	2.51	167	2.22	256	1.97	668
80	2.98	235	2.68	364	2.49	867
90	2.87	319	2.58	496	2.39	1155
100	2.63	424	2.29	656	2.12	1534
平均	1.93	140	1.56	216	1.34	530

表２：下界値の比較（フロー費用／デザイン費用＝20）

ノード数	Holmberg-Hellstrand		片山ら		本研究	
	誤差（%）	計算時間（s）	誤差（%）	計算時間（s）	誤差（%）	計算時間（s）
10	2.10	1	0.27	1	0.03	2
20	2.93	8	0.34	10	0.11	26
30	3.57	21	0.76	31	0.48	106
40	3.89	43	1.63	64	1.20	193
50	4.45	72	1.99	110	1.64	309
60	4.52	113	2.13	173	1.80	473
70	4.53	167	2.43	258	2.14	667
80	5.75	235	3.36	366	3.09	882
90	5.73	319	3.48	497	3.26	1186
100	5.17	424	3.17	659	2.91	1574
平均	4.26	140	1.95	217	1.67	542

表３：下界値の比較（フロー費用／デザイン費用＝30）

ノード数	Holmberg-Hellstrand		片山ら		本研究	
	誤差（%）	計算時間（s）	誤差（%）	計算時間（s）	誤差（%）	計算時間（s）
10	4.26	2	0.06	1	0.00	2
20	5.76	8	0.20	8	0.06	25
30	6.22	21	0.56	32	0.32	109
40	6.46	42	1.49	64	1.03	206
50	7.22	72	1.82	110	1.48	320
60	7.61	111	2.35	175	1.99	499
70	7.36	164	2.77	259	2.41	704
80	8.64	231	3.49	367	3.18	920
90	8.65	314	3.72	503	3.47	1238
100	7.88	415	3.40	664	3.11	1602
平均	7.01	138	1.99	218	1.71	562

ある。また，ノード数40以上の誤差には，上界値と最適値の差，および最適値と下界値との差の両方を含んでいる。

Holmberg-Hellstrandの解法では，費用比10では0.50～2.98％，平均1.93％の誤差であるが，費用比20では2.10～5.75％，平均4.26％，費用比30では4.26～8.65％，平均7.01％となり，デザイン費用が高くなるにつれて誤差が大幅に増加している。このように，(3)式の強制制約式のみを用いた定式化はデザイン費用が相対的に高い問題では弱い定式化

となり，Holmberg-Hellstrandの解法では良い下界値を求めることができなかった。

片山らの解法では，費用比10では0.16～2.68％，平均1.56％の誤差，費用比20では0.27～3.48%，平均1.95%，費用比30では0.06～3.72%，平均1.99%と，デザイン費用が高くなるにつれて誤差が増加している。しかし，Holmberg-Hellstrandの解法に比べると増加率は小さく，誤差は費用比10で平均0.37%，費用比20で2.31%，費用比30で5.02%減少している。

一方，本解法では，費用比10では0.00～2.49％，平均1.34％の誤差であり，費用比20では0.03～3.26％，平均1.67％，費用比30では0.00～3.47％，平均1.71％となっている。Holmberg-Hellstrandの解法に比べて費用比10で平均で0.59％，費用比20で2.59％，費用比30で5.30％減少し，片山らの解法と比べて費用比10で平均で0.22％，費用比20で0.28％，費用比30で0.28％減少している。このようにすべての費用比において，従来の解法より精度が高い下界値を算出することができた。

Holmberg-Hellstrandの解法の計算時間は平均138～140秒，片山らの解法では平均216～217秒，本解法では平均530～562秒であった。計算時間においては，費用比の影響は見られなかった。３つの解法では，Holmberg-Hellstrandの解法が最も計算時間が短く，次いで片山らの解法となり，本研究の解法が最も計算時間を必要とした。本研究の解法の計算時間が大きな理由として，他の解法に比べてLagrange乗数の数が増加していること，およびこれに伴い劣勾配法における収束回数の増加することが挙げられる。

7　おわりに

本研究では，容量制約のないネットワークデザイン問題に対して，以下のような成果を示した。終点を同じにする品種のフロー，および同一ノードを始点とする品種と終点とする品種に対する強制制約式である強い妥当不等式をLagrage緩和した問題を示し，この緩和問題の解法を提案した。数値実験により，提案する解法が従来の解法よりも精度が高い下界値を算出できることを示した。

しかしながら，ノード数が70を超えると誤差が２％を超え，また，ノード数が100を超えると計算時間が膨大なものとなる。このように，規模の大きな問題における精度の改善，高速化が課題である。

参考文献

[1] A. Balakrishnan, T. L. Magnanti, and R. T. Wong. A dual-ascent procedure for large-scale uncapacitated network design. *Operations Research*, Vol. 37, pp. 716-740, 1989.

[2] J. W. Billheimer and P. Gray. Network design with fixed and variable cost elements. *Transportation Science*, Vol. 7, pp. 49-74, 1973.

[3] K. Holmberg and J. Hellstrand. Solving the uncapacitated network design problem by a Lagrangean heuristic and branch-and-bound. *Operations Research*, Vol. 46, pp. 247–259, 1998.

[4] 片山直登, 岩田実, 柳下和夫, 三原一郎, 今澤明男, ラグランジュ緩和法を用いた容量制約のないネットワーク計画問題の解法。土木計画学研究・論文集, Vol. 11, pp. 105–112, 1993.

[5] 片山直登. ネットワークデザイン問題の近似解法。流通経済大学流通情報学部開校記念論文集, pp. 171–191, 1997.

[6] 片山直登. 列性成法と行生成法を用いた容量制約のないネットワークデザイン問題の近似解法. 流通経済大学流通情報学部紀要, Vol. 12, pp. 1–15, 2007.

[7] 片山直登. ネットワーク設計問題。朝倉書店, 2008。

[8] M. Los and C. Lardinois. Combinatorial programming, statistical optimization and the optimal transportation network problem. *Transportation Research* B, Vol. 16, pp. 89–124, 1982.

[9] T. L. Magnanti, P. Mireault, and R. T. Wong. Tailoring benders decomposition for uncapacitated network design. *Mathematical Programming Study*, Vol. 26, pp. 112–155, 1986.

[10] T. L. Magnanti and R. T. Wong. Network design and transportation planning: Models and algorithms. *Transportation Science*, Vol. 18, pp. 1–55, 1984.

[11] M. Minoux. Multiflots de coût minimal avec fonctions de coût concaves. *Annales des Télécommunications*, Vol. 31, pp. 77–92, 1976.

[12] M. Minoux. Network synthesis and optimum network design problems: Models, solution methods and applications. *Networks*, Vol. 19, pp. 313–360, 1989.