《論　文》

非分割フローを考慮した容量制約をもつネットワーク設計問題の MIP 近傍探索法

片　山　直　登

1　はじめに

ネットワーク設計問題は，アークにかかる固定費用とものの移動にかかる変動費用を考慮して，ネットワークを適切な形状と異なる始点と終点をもつ複数のものの適切な移動経路を決める問題である．通信ネットワーク上のパケット通信では同一の始点と終点をもつパケットは必ずしも単一の経路上で送信されるとは限らず，トラフィックの状態に応じて複数の経路上で分割されて送信されることが一般的である．一方，ロジスティクスネットワーク上では，同一の発地と着地をもつ同一の品種の荷物が複数に分割され，別々の経路で輸送されることはまれであり，単一の経路上で輸送される．

このような同一の品種の経路が分割されないフローを非分割フローとよび，この非分割フローを考慮した問題を非分割フローを考慮した容量制約をもつネットワーク設計問題（*UCND*: Unsplittable Capacitated Network Design Problem）とよぶ．

この問題では，アークのデザイン変数のみならずパスやアーク上のフロー変数も0−1変数となり，すべての変数が変数である大規模な組合せ最適化問題となるため，最適に解くことが困難な問題となる．

近年，非分割フローを考慮した容量制約をもつネットワーク設計問題に関する研究が行われ始めており，Hewitt et al.（2010）の IP 探索法，Yaghini and Kazemzadeh（2012）のシミュレーテッドアニーリング法，Hewitt et al.（2013）の分枝価格法とガイドつき探索法，片山（2013）の容量スケーリング法と局所分枝法・パス再結合法，片山

（2016）の容量スケーリング法と貪欲法が開発されている．一方，容量制約をもつ設計問題に対する高速な近似解法として，片山（2017）は容量スケーリング法と MIP 近傍探索法を組み合わせた高速解法を提案しており，精度の高い近似解を短時間で生成することに成功している．

本研究では，非分割フローを考慮した容量制約をもつ設計問題に対して，容量スケーリング法と MIP 近傍探索法を組み合わせた高速解法を提案する．

2　問題の定式化

N をノード集合，A を向きをもつアーク集合，K をネットワーク上で移動する品種集合とする．また，f_{ij} をアーク（i, j）上においてアークを選択したときに発生する非負のアーク費用，c_{ij}^{k} をアーク（i, j）上を移動する品種 k の需要量に対する非負のフロー費用，C_{ij} をアーク（i, j）上のアーク容量，d^{k} を品種 k の始点 O^{k} と終点 D^{k} 間を流れる品種 k とする．x_{ij}^{k} をアーク（i, j）上を移動する品種 k のフローが存在するとき１，そうでないとき０を表す0−1変数であるアークフロー変数とし，y_{ij} をアーク（i, j）上のアークを選択するとき１，そうでないとき０である0−1変数であるデザイン変数とする．このとき，アーク集合 A に関する *UCND* のアークフローによる定式化 *UCNDA* は次のように表すことができる．

（*UCNDA*）

$$\Phi = \min \sum_{(i,j)\in A}\sum_{k\in K} c_{ij}^{k} x_{ij}^{k} + \sum_{(i,j)\in A} f_{ij} y_{ij} \tag{1}$$

subject to

$$\sum_{i\in N_n^{+}} x_{in}^{k} - \sum_{j\in N_n^{-}} x_{nj}^{k} = \begin{cases} -1 & if\ n = O^{k} \\ 1 & if\ n = D^{k} \\ 0 & otherwise \end{cases} \quad \forall n \in N,\ k \in K \tag{2}$$

$$\sum_{k\in K} d^{k} x_{ij}^{k} \le C_{ij} y_{ij} \quad \forall (i,j) \in A \tag{3}$$

$$x_{ij}^{k} \le y_{ij} \quad \forall k \in K, (i,j) \in A \tag{4}$$

$$x_{ij}^{k} \in \{0,1\} \quad \forall k \in K, (i,j) \in A \tag{5}$$

$$y_{ij} \in \{0,1\} \quad \forall (i,j) \in A \tag{6}$$

(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．なお，Φ

は目的関数の最適値である．(2)式はフロー保存式であり，ノードに流入するフロー変数値と流出するフロー変数値の差が，品種 k の始点であれば -1，終点であれば 1，その他のノードであれば 0 であることを表す．(3)式は，容量制約式である．アーク（i, j）が選択されるとき，アーク上のフロー量の合計がアーク容量以下であることを表す．また，アークが選択されないときはフロー量の合計が 0 であることを表す．(4)式は，アーク（i, j）における品種 k に関する強制制約式である．アーク（i, j）が選択されるときには品種 k のアークフロー変数値が最大で 1 であり，アーク（i, j）が選択されないときには 0 となることを表す．(5)式はアークフロー変数の0－1条件，(6)式はデザイン変数の0－1条件である．

0－1条件である (5)式と(6)式を線形緩和した問題を *UCNDAL* とする．*UCNDAL* は線形緩和問題であるために，MIP ソルバーにより容易に最適解を求めることができる．

P^k を品種 k の取りうるパス集合，z_p^k を品種 k のフローがパス p 上を移動するとき 1，そうでないと 0 を表す0－1変数であるパスフロー変数，δ_{ij}^p をパス p にアーク（i, j）が含まれるとき 1，そうでないとき 0 を表す定数とする．このとき，アーク集合 A，パス集合 P，アーク容量 C に対する *UCND* のパスフローによる定式化 *UCNDP*（A, P, C）は次のように表すことができる．

（*UCNDP*（A, P, C））

$$\min \quad \sum_{(i,j)\in A}\sum_{k\in K} c_{ij}^k \sum_{p\in P^k} \delta_{ij}^p z_p^k + \sum_{(i,j)\in A} f_{ij} y_{ij} \tag{7}$$

subject to

$$\sum_{p\in P^k} z_p^k = 1 \quad \forall k\in K \tag{8}$$

$$\sum_{k\in K}\sum_{p\in P^k} \delta_{ij}^p d^k z_p^k \le C_{ij} y_{ij} \quad \forall (i,j)\in A \tag{9}$$

$$\sum_{p\in P^k} \delta_{ij}^p z_p^k \le y_{ij} \quad \forall k\in K, (i,j)\in A \tag{10}$$

$$z_p^k \in \{0,1\} \quad \forall p\in P^k, k\in K \tag{11}$$

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

(7)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．(8)式は，品種 k のパスフロー変数値の合計が 1，すなわち単一のパス上のみにフローが流れることを表す．(9)式は，アーク（i, j）が選択されるときはアーク上を移動するフロー量の

合計がアーク容量以下であり，アークが選択されないときは0であることを表す容量制約式である．(10)式は，アーク（i, j）における品種kに関する強制制約式である．(11)式はパスフロー変数の0−1条件，(12)式はデザイン変数の0−1条件である．

0−1条件である(11)式と(12)式を線形緩和した問題を*UCNDPL*（*A, P, C*）とする．*UCNDPL*（*A, P, C*）は線形緩和問題であるために，MIPソルバーにより容易に最適解を求めることができる．

アークフロー変数は$|A||K|$個，パスフロー変数は指数個と膨大な数であり，*UCNDA*や*UCNDP*（*A, P, C*）は膨大な数の0−1変数を含む大規模な組合最適化問題となる．一方，アークフロー変数およびパスフロー変数がの0−1条件を緩和すれば，*UCND*は容量制約をもつネットワーク設計問題に帰着される．そこで，容量制約をもつネットワーク設計問題に対する近似解法である容量スケーリング法とMIP近傍探索法を適用する．

3　近似解法

0−1条件を線形緩和した問題に対して容量スケーリング法を適用し，デザイン変数解を求める．正であるデザイン変数解をもとにフロー変数解の0−1条件を満たす実行可能なデザイン変数解を探索し，この解をもとに限定した分枝限定法とMIP近傍探索法を適用する．

3．1　容量スケーリング法

はじめに，パスフローによる定式化*UCNDP*（*A, P, C*）を線形緩和した問題*UCNDPL*（*A, P, C*）に対して容量スケーリング法を適用する．容量スケーリング法は，線形緩和問題の解をもとに，アーク容量を変更して繰り返し線形緩和問題を解き，デザイン変数の0−1変数解を導く近似解法である．解法の中では，列生成法を実施する．列生成法は限定主問題を解き，被約費用が負となる変数を生成する方法である．

容量スケーリング法では，繰り返しともに多くのデザイン変数解が0または1に収束する一方で，いくつかの変数は0または1には収束しないことが知られている．そこで，0または1に収束しないデザイン変数の数が一定値α以下となったときに，容量スケーリングを終了する．なお，パスフロー変数も線形緩和しているため，得られるパスフロー変数は0または1になるとは限らない．

容量スケーリング法のアルゴリズムをAlgorithm1に示す．容量スケーリング法および列生成法については，片山（2013）に詳細が記載されている．

3．2　アーク付加と限定分枝限定法

容量スケーリング法ではデザイン変数およびフロー変数を線形緩和した問題を対象と

しているため，得られたデザイン変数解は $UCNDP$（A, P, C）や $UCNDA$ の実行可能解とは限らない．加えて，容量スケーリング法から得られるフロー変数解も 0 または 1 となることは期待できない．そこで，1 に収束したデザイン変数解からなるアーク集合から始め，フロー変数の0－1条件のもとで，このアーク集合に 0 または 1 に収束しなかった変数に対応するアークを付加していき，実行可能なアークを含むアーク集合を選定する．これによってアークを選別し，問題規模の縮小を図る．

容量スケーリング法により得られたデザイン解を $\hat{y}$ とし，1 に収束した $\hat{y}_{ij}=1$ であるアークからなるアーク集合を $\bar{A}$ とする．アークフローによる定式化を線形緩和した $UCNDAL$ において，$\bar{A}$ に含まれるアークのデザイン変数を 1 に固定する次の条件を付加した問題 $UCNDALF$（$\bar{A}$）を解く．

$$y_{ij}=1 \quad \forall (i,j)\in \bar{A} \tag{13}$$

$UCNDALF$（$\bar{A}$）は線形計画問題であるので，容易に解くことができる．$UCBDALF$（$\bar{A}$）の最適なデザイン解を $\tilde{y}$ とおく．

アーク集合 $\bar{A}$ に含まれるアーク（i, j）については 1 に固定されているため，$\tilde{y}_{ij}=1$ となる．続いて，フロー変数の0－1条件のもとで実行可能解を含むアーク集合を得るために，$\bar{A}$ に含まれていない $\tilde{y}_{ij}>0$であるアークを $\bar{A}$ に付加していく．$\tilde{y}_{ij}$ の降順に該当するアークをソートする．$\tilde{y}_{ij}$ が 1 に近いアークは解に含まれる可能性が高く，0 に近いアークは解に含まれる可能性が低いと考えることができる．

$UCNDA$ のデザイン変数の0－1条件である (6)式のみを線形緩和し，アーク集合 $\bar{A}$ に対する (13)式を付加した問題を $UCNDALY$（$\bar{A}$）とする．$UCNDALY$（$\bar{A}$）を MIP ソルバーを用いて解く．しかし，この問題には多数の0－1変数であるアークフロー変数が含まれるため，最適に解くことは困難である．そこで，最適に解くのではなく，実行可能解が存在するか否かのみを判定する．1 つの実行可能解を探索するのみであるので，相対的に短い時間で判定することができる．

適当な整数を H とし，デザイン変数 $\tilde{y}_{ij}$ の値の降順に上位の H 本のアークを選択し，アーク集合 $\bar{A}$ に付加して，$\bar{A}$ に対する $UCNDALY$（$\bar{A}$）を解く．$\bar{A}$ に対して，実行可能解が存在すると判定された場合は，手順を終了する．そうでない場合は，$\tilde{y}_{ij}>0$ であるすべてのアークを付加するまで，順次，次の H 本のアークを選択して，同様な手順を繰り返す．

列生成法にて容量スケーリング法を解いた場合に生成されたパス集合を $\bar{P}$ とする．設定した計算時間 T のもとで，実行可能解を含むアーク集合 $\bar{A}$ と $\bar{P}$ に限定された $UCNDP$（$\bar{A}$, $\bar{P}$, C）を MIP ソルバーを用いて解く．この問題はアークとパス集合が限定されているため，比較的容易に解を求めることができる．この方法を限定分枝限定法とよぶ．設定した計算時間 T 以内で実行可能解解を算出できた場合は，この解を $\bar{y}$ と

する．実行可能解が算出できない場合は，$\bar{A}$ に含まれるすべてのアークを $y_{ij}=1$ とした *UCNDA* を解く．実行可能解が得られた場合は，得られた解を $\bar{y}$ とする．それでも実行可能解が求まらない場合は，$\hat{y}$ を切り上げたものを $\bar{y}$ とする．

アーク付加と限定分枝限定法のアルゴリズムを Algoritm2に示す．

3．3　MIP 近傍探索法

前節の手順で求めた解 $\bar{y}$ を初期の暫定解とし，*UCNDA* において，その近傍を MIP ソルバーを用いて探索し，近傍解を算出していく．

本研究では，適時，暫定解を改善するように探索範囲を限定する条件を付加する MIP 近傍探索法を適用する．始めに，*UCNDA* に次の制約式を追加する．

$$\sum_{(i,j)\in A|\bar{y}_{ij}=1} y_{ij} \leq L-1 \tag{14}$$

ここで，L は暫定解において $\bar{y}_{ij}=1$ であるアーク数である．(14)式は，暫定解で選択されたアークから少なくとも 1 本のアークはネットワークから取り除くことを表しており，実行可能領域から暫定解を排除することができる．また，現在までの最良の上界値を *UB* とおき，(14)式を付加した *UCNDA* の最適値が現在の上界値 *UB* 未満となる条件である次の式も追加する．

$$\Phi < UB \tag{15}$$

(14)式と(15)式により暫定解と探索済み解を排除できることから，解の循環を防ぐことができる．なお，現在までの最良値よりも良い解が存在しなければ問題は実行不可能となる．

計算時間に T の制限を設け，MIP ソルバーを用いて(14)式と(15)式を付加した問題を解く。実行可能解が得られた場合は，改善された解が探索されたことになる．その場合には，得られた解を暫定解 $\bar{y}$ として，更新する．そうでない場合は，解を更新しない．

続いて，(14)式と(15)式を取り除き，暫定解 $\bar{y}$ に対する(14)式と(15)式を加え，さらに次の制約式を追加する．

$$\sum_{(i,j)\in A|\bar{y}_{ij}=1} y_{ij} \geq L-M \tag{16}$$

(16)式は，暫定解で選択されたアークを高々 M 本のアークをネットワークから取り除くことを表す．なお，M は正の整数であり，近傍の探索範囲である．M が大きければ，条件を付加した *UCNDA* の実行可能領域は広くなる．一方，M が小さければ実行可能領域は狭くなるため，相対的に短時間で実行可能解を探索できる可能性が高まることになる。

UCNDA に(14)式，(15)式および(16)式を追加し，一定時間内で MIP ソルバーを用

いて実行可能解 $\tilde{y}$ を求め，$\bar{y}=\tilde{y}$ とし，暫定解を更新する．続いて，追加した 3 本の制約式を削除して，更新された解に対応した 3 本の制約式を追加し，近傍探索を繰り返す．

この近傍探索において，実行不可能または暫定解よりも良い解が無いと判断できた場合は，探索を終了する．また，一定時間内に暫定解より良い解を算出することができない場合は，$M:=\lfloor M/\beta \rfloor$ として，M を減少させ，実行可能領域を縮小する．$M>0$ である間，同様の探索を繰り返す．ここで，$\beta\,(>1)$ は M の変更基準であり，M は単調減少することになる．

このような探索法を MIP 近傍探索法とよぶ．従来の局所分枝法の制約と比べると，削除するアーク数には制約があるが，追加するアークには制限が無いのが特徴である．また，局所分枝法では領域を二分割しながら分枝・限定操作を行っていくが，この近傍探索法は単に限定された近傍探索と解の移動を繰り返すのみである．(14)式と(15)式により解の循環を防ぐことができ，得られる目的間数値は単調減少であり貪欲的な探索法となるため，必要な計算時間を抑制することができる．

容量スケーリング法と MIP 近傍探索法を組み合わせた解法のアルゴリズムを Algoritm3に示す．

4　数値実験

容量制約をもつネットワーク設計問題で用いられるベンチマーク問題である C 問題の内，$UCND$ で実行可能な31問（Hewitt et al. 2010）を用いて，数値実験を行った．

数値実験で使用した設定した機器等は以下の通りである．

・使用 OS および言語：UBUNTU 17.04，C++

・最適化ソルバー：Gurobi 7.5

・CPU AMD Ryzen7 1800X 3.6GHz 8Cores，RAM16GByte

・使用コア数：容量スケーリング 1 コア，MIP 近傍探索 8 コア

また，数値実験で使用した設定したパラメータは以下の通りである．

・スケーリングパラメータ：0～1

・スケーリング法の終了判定アーク数 α：200

・近傍 M：20

・近傍 M の変更基準 β：2

・アークの付加本数 H：10

・MIP ソルバー計算時間の制限時間 T：60秒

近似解の誤差を算出するために，MIP ソルバーである Gurobi により，定式化 $UCNDA$ を30時間解き，下界値を算出した．同時に上界値も算出した．

C 問題に対しては多くの研究が行われ，その結果が公開されている．ここでは，IP 探

索法（IPS）（Hewitt et al. 2010），シミュレーテッドアニーリング法（SA）（Yaghini and Kazemzadeh 2012），分枝価格・ガイドつき探索法（BPGS）（Hewitt et al. 2013），容量スケーリング・限定分枝限定法（CSRB），容量スケーリング・局所分枝法（CSLB），容量スケーリング・パス再結合法（CSPR）（片山2013），容量スケーリング・貪欲解法（CSGR）（片山2017）の結果を示す．これらに加え，本研究である容量スケーリング・MIP 近傍探索法（CSMNS），およびパラメータチューニングをした容量スケーリング・MIP 近傍探索法（CSMNST）の結果を示す．前者では，Algorithm1に記載しているスケーリングパラメータ λ を0.65と設定した．後者は，問題ごとにスケーリングパラメータを変化させた中の最良値である．なお，誤差（Gaps）は「(各解法の上界値－下界値)/ 下界値」とし，平均誤差はこれらの平均値である．

表１にＣ問題に対する上界値の平均誤差を示す．従来の研究では，IP 探索法は平均誤差2.22%，シミュレーテッドアニーリング法は14.42%，分枝価格・ガイド付き局所探索法は1.26%，容量スケーリング・貪欲法は3.71% である．容量スケーリング・限定分枝限定法は1.11%，容量スケーリング・局所分枝法は1.00%，容量スケーリング・パス再結合法は0.90% である．従来の研究では，容量スケーリング・パス再結合法の誤差が最小であり，１% 以下となっている．なお，MIP ソルバーである Gurobi は平均誤差0.89%であり，最も誤差が小さいが，後述のように膨大な計算時間を必要とする．

一方，本研究である容量スケーリング・MIP 近傍探索法は平均誤差1.24% であり，スケーリングラメータをチューニングした容量スケーリング・MIP 近傍探索法は0.92%であり，１% 以下となった．容量スケーリング・MIP 近傍探索法は，容量スケーリング・局所分枝法や容量スケーリング・パス再結合法よりやや劣っているが，分枝価格・ガイド付き局所探索法と同程度の誤差である．また，チューニングした容量スケーリング・MIP 近傍探索法は，最も良い容量スケーリング・パス再結合法より0.02% 劣っているが，その他の近似解法よりも良い解を算出している．

表２に，個別問題の上界値を示す．なお，LB は下界値または最適値であり，太字は最適値，斜体文字は最良値を表している．分枝価格・ガイドつき探索法は最適値を９問を算出している．容量スケーリング・局所分枝法は，８問の最適値，最適値を除く６問の最良値を算出している．容量スケーリング・パス再結合法は，15問の最適値，２問の最良値を算出している．一方，容量スケーリング・MIP 近傍探索法は11問の最適値を算出し，パラメータをチューニングした容量スケーリング・MIP 近傍探索法は14問の最適値，４問の最良値を算出している．なお，Gurobi は16問の最適値，６問の最良値を算出し，31問の内の23問の最良値を算出している．

表３にＣ問題に対する個々の計算時間と平均計算時間を示す．従来の研究の計算時間は，各論文に掲載しているものであり，使用しているコンピュータが異なっているため，計算時間を直接比較することはできない．使用している主なコンピュータは以下の

通りである．

・LB，Giurobi：AMD Ryzen 1800x（8-Core, 3.6GHz）
・IP 探索法：Intel Xeon CPUs（8-Core, 2.66GHz）
・シミュレーテッドアニーリング法：記載なし
・分枝価格・ガイドつき探索法：Intel Xeon CPUs（8-Core 2.66GHz）
・容量スケーリング・限定分枝限定法，容量スケーリング・局所分枝法，容量スケーリング・パス再結合法：Intel i7 2600（4-Core, 3.4GHz）
・容量スケーリング・貪欲解法：Intel i7 3770K（4-Core, 3.5GHz）

平均誤差の最も小さい Gurobi では30時間の制限を付けているため，最適解を求められない場合は108000秒となり，平均計算時間は59252.3秒で，16時間を越えている．タブー探索法は900秒，分枝価格・ガイド付き局所探索法は1800秒の計算時間を設定し，シミュレーテッドアニーリング法は1200秒の計算を10回繰り返しており，良好な解を算出するためには大きな計算時間が必要としている．小さな誤差を算出できている容量スケーリング・限定分枝限定法は5491.2秒，容量スケーリング・局所分枝法は7566.3秒，容量スケーリング・パス再結合法は14503.2秒といずれも非常に大きな計算時間が必要としている．また，容量スケーリング・貪欲解法の平均計算時間は87.8秒と大変短いが，平均誤差は大きくなっている．

一方，本研究である容量スケーリング・MIP 近傍探索法の平均計算時間は451.2秒，パラメータをチューニングした容量スケーリング・MIP 近傍探索法の平均計算時間は435.3秒であり，他の解法に比べて，大幅に計算時間を短縮することができている．なお，後者は，実際にはパラメータ選定のための事前の計算時間が必要である．使用しているコンピュータが異っているにしても，従来に研究に比べ，精度を保ちながら大幅に計算時間が短縮できることが分かる．

5　おわりに

本研究では，非分割フローを考慮した容量制約をもつネットワーク設計問題に対して，容量スケーリング法と MIP ソルバーによる近傍探索を組み合わせた高速で精度の高い近似解法を提案した．また，ベンチマーク問題である C 問題に対して，数値実験を行い，従来の研究との比較を行った．

本研究で提案した解法は，最良の解法の一つであり，容量スケーリング・パス再結合法と比較すると誤差はわずかながら大きいが，計算時間は1/30以下と大幅に短縮することができた．また，容量スケーリング・パス再結合法以外のいずれの近似解法よりも，短い計算時間で精度の高い近似解を算出することができた．

本研究は科学研究費基盤研究 C（課題番号17K01268）による成果の一部である．

参考文献

M. Hewitt, G. L. Nemhauser, and M. W. P. Savelsbergh. Combining exact and heuristics approaches for the capacitated fixed charge network flow problem. *Journal on Computing,* 22: 314-325, 2010.

M. Hewitt, G. L. Nemhauser, and M. W. P. Savelsbergh. Branch-and-price guided search for integer programs with an application to the multicommodity fixed charge network flow problem. *INFORMS Journal on Computing*, 25（2）: 302-316, 2013.

M. Yaghini and M. R. A Kazemzadeh. A simulated annealing algorithm for unsplittable capacitated network design. *International Journall of Industrial Engineering & Production Research*, 23（2）: 91-100, 2012.

片山直登．非分割フローを考慮した容量制約をもつネットワーク設計問題．流通経済大学流通情報学部紀要，20 (1): 1-19, 2013.

片山直登．容量制約をもつネットワークデザイン問題の貪欲解法．流通経済大学流通情報学部紀要，20 (1): 1-24, 2015.

片山直登．容量制約をもつネットワーク設計問題に対する MIP 近傍探索法．流通経済大学流通情報学部紀要，22 (1): 1-18, 2017.

Algorithm 1: Capacity Scaling

Set $\bar{P}$, α, λ, ITE_{min}, ITE_{max};
$l \leftarrow 1$;
$C^1 \leftarrow C$;
repeat
    Solve $UCNDPL(A, \bar{P}, C^l)$;
    Get the solution $\hat{y}$ of $UCNDPL(A, \bar{P}, C^l)$;
    Add paths to $\bar{P}$ by Column Genaration;
    $n \leftarrow 0$;
    for $(i, j) \in A$ **do**
        $C_{ij}^{l+1} \leftarrow \lambda C_{ij}^l \hat{y}_{ij} + (1 - \lambda) C_{ij}^l$;
        if $0 < \hat{y}_{ij} < 1$ **then**
            $n \leftarrow n + 1$;
        end
    end
until $l \geq ITE_{min}$ *and* $n \leq \alpha$, *or* $l \geq ITE_{max}$;
$\bar{A} \leftarrow \varnothing$;
for $(i, j) \in A$ **do**
    if $\hat{y}_{ij} = 1$ **then**
        $\bar{A} \leftarrow \bar{A} \cup \{(i, j)\}$;
    end
end
Return $\hat{y}$,$\bar{A}$,$\bar{P}$;

Algorithm 2: Add Arcs and Restricted Branch-and-Bound($\bar{A}$,$\bar{P}$,$\hat{y}$)

Set H and T;
Solve $UCNDALF(\bar{A})$ with equation (13) and get the solution $\tilde{y}$;
Sort the the solution $\tilde{y}_{ij}$,$\forall(i,j) \in A\backslash\bar{A}$ in descending order;
Set $1, \cdots, |A\backslash\bar{A}|$ be the arc indexs in ascending order of $\tilde{y}$;
for $l = 1$ *to* $|A\backslash\bar{A}|$ **do**
    if $\tilde{y}_l = 0$ **then**
        break;
    end
    $\bar{A} := \bar{A} \cup arc_l$;
    if $l\ Mod\ H = 0$ **then**
        Solve $UCNDALY(\bar{A})$ within time T;
        if *a feasible solution of* $UCNDALY(\bar{A})$ *is found* **then**
            break;
        end
    end
end
Solve $UCNDP(\bar{A}, \bar{P}, C)$ within time T;
if *the feasible solution y' of* $UCNDP(\bar{A}, \bar{P}, C)$ *is found* **then**
    Get the upper bound UB of $UCNDP(\bar{A}, \bar{P}, C)$;
    $\bar{y} = y$';
else
    Add equations (13) to $UCNDA$ associated with $\bar{A}$;
    Solve $UCNDA$ within time T;
    if *the solution y' of* $UCNDA$ *is found* **then**
        Get the upper bound UB of $UCNDA$;
        $\bar{y} = y$';
    else
        $\bar{y} = \lfloor \hat{y} \rfloor$;
    end
end
return $\bar{y}$,UB;

Algorithm 3: MIP Neighborhood Search($\bar{y}$, UB)

Set M, β, T;
Add equations (14) and (15) to $UCNDA$ associated with the solution $\bar{y}$;
Solve $UCNDA$ within time T;
if *the solution* $\tilde{y}$ *of* $UCNDA$ *is found* **then**
    Get the upper bound UB_{neigh} of $UCNDA$;
    $\bar{y} \leftarrow \tilde{y}$;
    $UB \leftarrow UB_{neigh}$;
end
Delete equations (14) and (15) from $UCNDA$;
repeat
    Add equations (14), (15) and (16) to $UCNDA$ associated with the solution $\bar{y}$;
    Solve $UCNDA$ within time T;
    if $UCNDA$ *has no feasible solution* **then**
        break;
    else
        if *the solution* $\tilde{y}$ *of* $UCNDA$ *is found* **then**
            Get the upper bound UB_{neigh} of $UCNDA$;
            $\bar{y} \leftarrow \tilde{y}$;
            $UB \leftarrow UB_{neigh}$;
        else
            $M \leftarrow \lfloor M/\beta \rfloor$;
        end
    end
    Delete equations (14), (15) and (16) from $UCNDA$;
until $M = 0$;
Return $\bar{y}, UB$;

表１：Average Gaps for C-Category Problems（%）

Gurobi	IPS	SA	BPGS	CSRB	CSLB	CSPR	CSGR	CSMIP	CSMIPT
0.89	2.22	14.42	1.26	1.11	1.00	0.90	3.71	1.24	0.92

表 2 ：Results for C-Category Problems

Problem	LB	Gurobi	IPS	SA	BPGS	CSRB	CSLB	CSPR	CSGR	CSMIP	CSMIPT
20/230/040/V/L	423933	**423933**	**423933**	-	**423933**	424075	424075	**423933**	427814	**423933**	**423933**
20/230/040/V/T	398870	**398870**	**398870**	-	**398870**	400380	400380	**398870**	400933	**398870**	**398870**
20/230/040/F/T	668699	**668699**	**668699**	670928	**668699**	669642	669642	**668699**	**668699**	**668699**	**668699**
20/230/200/V/L	94644	**94644**	95695	118785	94843	**94644**	**94644**	**94644**	98725	95770	94765
20/230/200/F/L	138084	**138084**	141700	-	138476	**138084**	**138084**	**138084**	145339	139394	**138084**
20/230/200/V/T	98610	**98610**	100884	112792	98947	98674	**98610**	98674	100728	98754	**98610**
20/230/200/F/T	135505	*137499*	141734	167006	139889	137618	137594	137618	146213	138517	137709
20/300/040/V/L	430253	**430253**	**430253**	433929	430314	**430253**	**430253**	**430253**	431583	**430253**	**430253**
20/300/040/F/L	597059	**597059**	**597059**	-	**597059**	597170	**597059**	**597059**	615270	**597059**	**597059**
20/300/040/V/T	501766	**501766**	**501766**	511384	501889	511019	511019	**501766**	501784	**501766**	**501766**
20/300/040/F/T	643395	**643395**	**643395**	656412	**643395**	651667	644453	**643395**	644816	**643395**	**643395**
20/300/200/V/L	74907	*75786*	76946	91702	76333	75820	75802	75820	77680	75864	75864
20/300/200/F/L	114144	*117539*	119590	151513	118204	117814	117617	117814	123947	118100	117570
20/300/200/V/T	76035	76381	77055	88550	76986	76549	*76357*	76483	78043	76531	*76357*
20/300/200/F/T	106915	109698	110516	-	109776	109875	*109385*	109785	120076	110302	109538
30/520/100/V/L	54387	**54387**	54427	57422	**54387**	54432	**54387**	**54387**	55474	**54387**	**54387**
30/520/100/F/L	94791	*95877*	97199	-	96277	96357	95947	96134	101353	96137	*95877*
30/520/100/V/T	53812	**53812**	**53812**	-	**53812**	53971	53961	**53812**	55080	**53812**	**53812**
30/520/100/F/T	98921	*99195*	100391	-	99355	99355	99204	99204	105476	99355	99204
30/520/400/V/L	112688	114454	116465	134587	114867	114405	*114295*	114405	115816	114729	114355
30/520/400/F/L	148188	*150875*	156433	-	152837	150909	150965	150909	154430	152654	151365
30/520/400/V/T	115289	116307	118193	-	116730	116318	*116306*	116318	117735	116884	116428
30/520/400/F/T	150896	155324	161513	-	155982	155820	155720	155249	160872	155913	*155039*
30/700/100/V/L	47883	**47883**	**47883**	51505	**47883**	**47883**	**47883**	**47883**	48494	**47883**	**47883**
30/700/100/F/L	60384	**60384**	61254	68179	**60384**	60573	**60384**	**60384**	62232	60829	60688
30/700/100/V/T	47670	**47670**	47736	-	47686	47766	47733	**47670**	48434	**47670**	**47670**
30/700/100/F/T	56686	**56686**	56931	-	56809	56942	56934	**56686**	58021	56898	**56686**
30/700/400/V/L	97429	98777	100368	-	98850	98538	*98535*	98538	100611	98689	98559
30/700/400/F/L	131500	136952	144077	-	138376	137468	137337	137017	141304	138186	*136812*
30/700/400/V/T	94914	96526	98040	-	97352	96468	96475	*96366*	98288	97260	96688
30/700/400/F/T	128298	132180	135512	168070	133759	*132112*	*132112*	*132112*	137226	133307	132316

表 3 ：Computation Time for C-Category Problems（Seconds）

Problem	LB/Gurobi	IPS	SA	BPGS	CSRB	CSLB	CSPR	CSGR	CSMIP	CSMIPT
20/230/040/V/L	0.3	900	12000	1800	0.3	1.6	2.9	0.4	0.7	0.7
20/230/040/V/T	0.3	900	12000	1800	2.3	2.3	2.5	1.2	0.7	0.7
20/230/040/F/T	0.4	900	12000	1800	2.4	2.2	3.7	1.0	1.4	1.4
20/230/200/V/L	51424.4	900	12000	1800	6387.1	6230.0	12633.9	75.2	642.2	672.1
20/230/200/F/L	104375.0	900	12000	1800	8444.1	7894.7	18309.0	74.2	537.1	555.1
20/230/200/V/T	43441.6	900	12000	1800	7193.8	7317.8	13721.4	68.6	662.5	609.0
20/230/200/F/T	108000.0	900	12000	1800	8406.0	8821.9	23224.7	75.5	728.9	730.7
20/300/040/V/L	0.2	900	12000	1800	0.3	1.9	3.0	0.3	0.5	0.5
20/300/040/F/L	0.8	900	12000	1800	1.4	3.1	6.3	1.4	2.5	2.4
20/300/040/V/T	0.6	900	12000	1800	0.5	0.5	2.5	1.4	1.5	1.4
20/300/040/F/T	0.5	900	12000	1800	0.9	2.6	3.2	0.7	1.2	1.2
20/300/200/V/L	108000.0	900	12000	1800	8057.3	14376.2	24778.3	77.1	633.8	628.4
20/300/200/F/L	108000.0	900	12000	1800	10223.2	11707.4	26117.9	76.5	700.6	815.3
20/300/200/V/T	108000.0	900	12000	1800	12236.3	15756.4	22231.7	75.1	688.2	708.0
20/300/200/F/T	108000.0	900	12000	1800	14141.9	11642.5	24045.8	77.3	770.6	788.3
30/520/100/V/L	129.0	900	12000	1800	28.4	69.1	1438.6	8.7	222.4	208.3
30/520/100/F/L	108000.0	900	12000	1800	6591.0	6610.9	15656.3	67.3	891.3	489.2
30/520/100/V/T	23.6	900	12000	1800	7.9	18.8	120.7	10.9	46.7	42.6
30/520/100/F/T	108000.0	900	12000	1800	5373.8	3946.1	8074.3	66.3	663.9	510.2
30/520/400/V/L	108000.0	900	12000	1800	7588.5	13609.0	26018.1	372.6	792.6	785.6
30/520/400/F/L	108000.0	900	12000	1800	7353.9	22043.1	34450.8	163.0	705.4	630.7
30/520/400/V/T	108000.0	900	12000	1800	7552.4	32353.1	44761.0	189.2	694.6	767.4
30/520/400/F/T	108000.0	900	12000	1800	8156.1	8180.4	31161.2	220.9	671.4	749.4
30/700/100/V/L	23.9	900	12000	1800	17.4	17.4	324.1	4.4	40.3	40.2
30/700/100/F/L	15729.0	900	12000	1800	2581.2	1014.9	6419.0	18.8	473.9	480.4
30/700/100/V/T	92.5	900	12000	1800	36.3	37.9	845.0	11.7	293.7	223.6
30/700/100/F/T	1576.3	900	12000	1800	568.6	407.9	8344.5	26.3	546.5	425.8
30/700/400/V/L	108000.0	900	12000	1800	7436.4	12658.6	25126.7	114.9	622.4	629.2
30/700/400/F/L	108000.0	900	12000	1800	22397.1	18378.7	29991.3	372.1	647.6	640.2
30/700/400/V/T	108000.0	900	12000	1800	7099.1	19107.2	27030.8	176.4	642.3	706.1
30/700/400/F/T	108000.0	900	12000	1800	12340.3	12340.3	24750.6	292.1	660.7	651.7
Average	59252.3	900	12000	1800	5491.2	7566.3	14503.2	87.8	451.2	435.3