3A3-2

# 連続量・大規模変量を伴う分散資源割り当てのための分散制約最適化手法の検討

A Study of Distributed Constraint Optimization on Continuous and Large Domain Variables

松井 俊浩 Toshihiro Matsui　兼子 昌幸 Masayuki Kaneko　高間 有歩 Yuho Takama　松尾 啓志 Hiroshi Matsuo

名古屋工業大学
Nagoya Institute of Technology

Resource allocation problem on resource supply networks is an application of Distributed Constraint Optimization Problems. In previous studies, distributed cooperative solution methods based on feeder trees have been applied to the resource allocation problems. However, the size of variable's domains that represent the amounts of resources increases in general cases of the resource supply networks since the resource amounts originally take continuous values. That is a critical issue even if the networks are trees since it causes a number of combinations of assignments. Therefore, sampling of solutions is necessary to reduce the size of problems. In this study, we propose the methods to reduce the number of samples in solution methods for resource allocation problems on resource supply networks. To maintain the samples, boundaries for amount of resources and cost values are introduced. With the proposed methods, the size of local problems in each agent is reduced while the feasibility is hold.

## 1. はじめに

ネットワーク上に存在する共有資源の配分を決定する分散資源割り当ては，マルチエージェントシステムの重要な応用の一つである．マルチエージェントシステムにおける協調問題解決の基本的な枠組みとして，分散制約最適化問題 (DCOP) が研究されている [Mailler 04, Modi 05, Petcu 05]．DCOP の手法では，エージェントの状態とそれらの関係が，制約最適化問題として形式化され，その問題を分散アルゴリズムとして構成された最適化手法を用いて解く．これらの研究はマルチエージェントシステムの協調プロトコルに含まれる，最適化問題と分散アルゴリズムに注目している．スマートグリッドの電力供給網資源に動機づけされる資源割り当て問題に DCOP を適用する関連研究 [Miller 12, Matsui 12] では，フィーダツリー上での資源の配分を決定する問題と解法が提案されている．これらは，共有資源をエージェント間で分解可能な大域的制約として明示的に表現するように，特化された DCOP である資源制約付き DCOP(RCDCOP) [Matsui 08] と関連する．これらの解法は離散最適化問題の解法から発展した解法であり，多様な資源割り当て問題に適用できる．その一方で，連続変数を離散化する場合や変数の値域が大規模である場合は，問題の規模の増大への対処が課題となる．本研究では，ネットワークを介して共有される分散資源の割り当てのための分散制約最適化手法において，連続変数を離散化する場合や変数の値域が大規模である場合に，実行可能性を維持しつつ問題の規模を抑制する近似的手法について検討する．

## 2. 背景

### 2.1 問題の定義

本研究では，関連研究 [Miller 12, Matsui 12] と類似する資源割り当て問題を検討の対象とする．資源供給網の形状は電力網のフィーダツリーを意図した木構造である．資源供給網は次の要素から構成される．

連絡先: 松井 俊浩，名古屋工業大学，〒 466-8555 愛知県名古屋市昭和区御器所町，matsui.t@nitech.ac.jp

- ノード: ノードは資源を消費または供給する．資源の割り当て量は消費または供給の量を表し，それらには下限と上限値がある．また，資源の割り当ての量に関して選好がある．
- リンク: リンクは二つのノードを接続する．ノードを経由して，ある量の資源がノード間で移送される．リンクが移送できる資源の容量には制限がある．資源の移送における損失は十分に小さく，無視される．

上記の制限に加えて，割り当てられる資源の量に関する制限がある．すなわち，各ノードにおいて，自身および他ノード間で供給および消費される資源の量の合計はゼロでなければならない．問題の目的は，制限の下で，大域的な結合された選好の値を最適化するような資源の割り当てを求めることである．

問題は，$\langle N, L, R, F, \mathcal{L}\rangle$ により形式的に定義される．$N$ はノードの集合, $L$ はリンクの集合, $R$ はノードに割り当てられる資源の量の集合, $F$ はコスト関数, $\mathcal{L}$ はリンクを移送される資源の量の集合である．

各ノード $i \in N$ に割り当てられる資源の量およびその選好は次のように定義される．

- $R_i$: $R_i \in R$ はノード $i$ に割り当てられる資源の量を表す有限集合である．本研究では資源の量は連続的であることを前提とする．その一方で解法として離散最適化手法を用いるため，$R_i$ により資源の量の標本値の集合を表す．割り当ての量 $r \in R_i$ が負値であれば資源の供給を表し，正値であれば消費を表す．
- $f_i(r)$: $f_i(r) \in F$ は割り当てられる資源の量 $r \in R_i$ に関する選好を表す関数であり，非ゼロの値となる．ここでは最小化問題として定式化するため，選好はコスト値として表現される．

リンク $(i, j)$ はノードの組 $\langle i, j\rangle$ について定義される．各ノード $i$ とリンクにより接続する近傍のノードを集合 $Nbr_i$ により表す．リンク $(i, j) \in L$ を移送する資源の量は次のように定義される．

- $l_{i,j}$: $l_{i,j}$ はリンク $(i,j)$ を経由して移送される資源の量を表す．$l_{i,j}$ は $-l^c_{i,j} \leq l_{i,j} \leq l^c_{i,j}$ なる値を取る．ここで $l^c_{i,j}$ はリンクの容量である．$l_{i,j}$ の符号は資源を移送する方向を表す．このために，資源供給網上での資源の移送の基準となる方向 (フロー) が予め定義されているものとする．資源の移送の方向とフローが同じであれば，$l_{i,j}$ は正の値であり，異なれば負である．

各 $i \in N$ ノードでは，$r_i$ と，$i$ に接続する全てのリンク $(i,j)$ についての $l_{i,j}$ の，合計はゼロでなければならない．この制約条件は $\sum_{(i,j)\in L^{in}_i} l_{i,j} = r_i + \sum_{(i,k)\in L^{out}_i} l_{i,k}$ のように表される．ここで，$L^{in}_i$ および $L^{out}_i$ はノード $i$ への資源の入力および出力のリンクの集合を表す.

全てのノードへの資源の量の割り当て $\mathcal{R}$ について，大域的なコスト値は $f(\mathcal{R}) = \sum_{i\in N} f_i(r_i)$ のように定義される．ここで，資源の量 $r_i$ は割り当て $\mathcal{R}$ に対応する値である．問題の目的は，制約条件の下で，$f(\mathcal{R})$ を最小化する割り当て $\mathcal{R}^*$ を求めることである．

## 2.2 資源制約付き分散制約最適化問題としての表現

分散制約最適化問題 (DCOP) はマルチエージェントシステム上の協調問題解決における基本問題として研究されている．DCOP では問題は，複数のエージェントに分散して配置された変数と関数により表現される．問題はメッセージ通信を伴う分散協調型の最適化アルゴリズムにより解決される．2.1 節の問題は資源供給網上の資源割り当て問題のために拡張された資源制約付き DCOP [Miller 12, Matsui 12] により表現される．

資源供給網のための資源制約付き DCOP は $\langle A, L, X^r, D^r, F, C\rangle$ により表される． ここで，$A$ はエージェントの集合を表す．エージェント $i \in A$ は資源割り当て問題のノードに対応する．表現を簡単にするために，ノードとエージェントは必要に応じて区別せずに用いる．$L$ はリンクの集合である．$X^r$ はノードで消費または供給される資源の量を表す変数の集合である．$D^r$ は $X^r$ の値域を表す有限集合である．$F$ はコスト関数の集合であり，$C$ は制約条件の集合である．

さらに，リンクを移送する資源の量を表す変数の集合 $X^l$ およびその値域 $D^l$ が導入される．これらの変数の値域は解法において動的に計算される．

エージェントの集合における半順序関係が，擬似木 [Petcu 05] により定義される．擬似木は，資源供給網における生成木の一つに対応する．本研究ではフィーダツリーを対象とするため，擬似木はフィーダツリーに対応する．擬似木に基づいて，各エージェント $i$ について，親 $p_i$，子の集合 $Ch_i$ がそれぞれ定義される．

エージェント $i$ が消費または供給する資源の量 $r$ は変数 $x^r_i \in X^r$ により表される．また，リンク $(i,j)$ により移送される資源の量 $l_{i,j}$ は，$x^l_{i,j} \in X^l_i$ により表される．ここで $X^l_i \subset X^l$ は，ノード $i$ に接続するリンクについての変数の集合である．これらは元の資源割り当て問題と同様である．

各エージェント $i$ は $X^r_i \cup X^l_i$ に含まれる自身に関係する変数のうち，擬似木における親 $p_i$ が決定する $x^l_{p_i,i}$ 以外の変数値を決定する．

エージェント $i$ のコスト関数 $f_i(x^r_i) \in F$ は元の資源割り当て問題の $f_i(r_i)$ に対応する．同様に，$c^r_i \in C$ は，元の問題のノード $i$ における制約条件に対応し，次のように表される．

$$c^r_i:\ x^l_{p_i,i} = x^r_i + \sum_{j\in Ch_i} x^l_{i,j} \tag{1}$$

また，リンク $(i,j)$ の容量についての制約条件 $c^l_{i,j} \in C$ は次のように定義される．

$$c^l_i:\ -l^c_{i,j} \leq x^l_{i,j} \leq l^c_{i,j} \tag{2}$$

全ての変数 $X^r \cup X^l$ についての大域的な割り当て $\mathcal{X}$ に関するコスト関数 $f(\mathcal{X})$ は，$f(\mathcal{X}) = \sum_{i\in A} f_i(x^r_i)$ のように定義される．ここで，$x^r_i$ は $\mathcal{X}$ の対応する値を取る. 最適な割り当て $\mathcal{X}^*$ は制約条件のもとで $f(\mathcal{X})$ を最小化する．

## 2.3 擬似木にもとづく計算

擬似木に基づくコスト値は次のように再帰的に計算される．エージェント $i$ の親からの資源の割り当てと，$i$ を根とする部分木についての最適なコスト $g^*_i(x^l_{p_i,i})$ は次のように表される．

$$g^*_i(x^l_{p_i,i}) = \min_{\mathcal{X}_i} g_i(\{x^l_{p_i,i}\} \cup \mathcal{X}_i) \tag{3}$$

$$g_i(\{x^l_{p_i,i}\} \cup \mathcal{X}_i) = \delta_i(\{x^l_{p_i,i}\} \cup \mathcal{X}_i) + \sum_{j\in Ch_i} g^*_j(x^l_{i,j}) \tag{4}$$

$$\delta_i(\{x^l_{p_i,i}\} \cup \mathcal{X}_i) = \begin{cases} f_i(d_i) \text{ s.t. } (x^r_i, d_i) \in \mathcal{X}_i & c^r_i \wedge c^l_{p_i,i} \\ \infty & \text{otherwise} \end{cases} \tag{5}$$

ここで，$\mathcal{X}_i$ は $\{(x^r_i, d_i)\} \cup \bigcup_{j\in Ch_i}\{(x^l_{i,j}, d_{i,j})\}, d_i \in D^r_i, d_{i,j} \in D^l_{i,j}$ であるような割り当てである．$x^l_{i,j}$ は $\mathcal{X}_i$ の対応する値を取る．また，リンク $(p_i,i)$ についての変数 $x^l_{p_i,i}$ の値域 $D^l_{p_i,i}$ は，$g^*_i(x^l_{p_i,i})$ のスコープに基づいて次のように求められる．

$$D^l_{p_i,i} = \{d^l_{p_i,i} | g^*_i(d^l_{p_i,i}) \neq \infty\} \tag{6}$$

上記の式では，$g^*_i(x^l_{p_i,i}) = \infty$ であれば $g^*_i(x^l_{p_i,i})$ は除かれ，値域が制限される．

ここでは，関連研究 [Miller 12] の解法 DYDOP を基礎として用いる．この解法は DCOP の解法 DPOP [Petcu 05] を拡張したものである．DYDOP の計算は 2 段階の処理からなる．最初の段階では，擬似木の葉から根へのボトムアップな計算によりコスト値が計算される．コスト値 $g^*_i(x^l_{p_i,i})$ および値域 $D^l_{p_i,i}$ の情報は，COST メッセージにより親ノード $p_i$ へ送信される．これにより，根ノードは自身の変数についての大域的なコスト値を得る．次の段階では，トップダウンな計算により，最適な変数値が決定される．最適な変数値は子ノードに VALUE メッセージにより送信される．

計算されたコスト値の情報は表に格納される．表の行の要素は $x^l_{p_i,i}$，$g^*_i(x^l_{p_i,i})$，および $x^l_{p_i,i}$ を計算する過程で用いられた全ての子についての $x^l_{i,j}$ から構成される. ここで，表の各行は，電力値のひとつの標本に対応することから，サンプルと呼ぶ．サンプル $s$ について，その要素*を $s.*$ で表す．また，子ノード $j$ から受信した COST メッセージは，子ノードごとに $x^l_{i,j}$ および $g^*_j(x^l_{i,j})$ からなる表に格納される．

この解法は，変数の値域の規模が大きいとき，各エージェントが持つ局所的な問題およびそれに起因する表の規模が大きくなるため，実際的ではない．また，局所的な問題の規模は，ノードの近傍数に応じて指数関数的に増加する．これにより，資源供給網が木構造の簡単な問題であっても，組み合わせ爆発の影響を受ける可能性がある．そのため，解のサンプルの数をより削減するようなサンプリングの手法が問題の規模を削減するために必要である．その一方で，このような手法により解の実行可能性と精度が減少する可能性がある．

## 3. サンプル数の削減

### 3.1 サンプル数の制限のもとでの統合と置換

表に含まれるサンプルの数を削減するために，サンプル数すなわち資源の割り当てとコスト値の表に含まれる行の数を $M$ 以下に抑制する．$M$ は任意のパラメータとして与えられる．この抑制のもとで，表に新たにサンプル $s$ を追加する計算では次の場合がある．

- (a) 表に $t.x^l_{p_i,i} = s.x^l_{p_i,i}$ であるようなサンプル $t$ がある．この場合は，両方のサンプルはコスト値の最小化により統合される[*1]．従って，表に含まれるサンプルの数は増加しない．それ以外の場合は，以下のいずれかが適用される．
- (b) 表に $t.x^l_{p_i,i}$ が $s.x^l_{p_i,i}$ の「範囲内」にあるようなサンプル $t$ がある．もしも複数のサンプルが範囲内にあれば，最もコスト値が大きいサンプルが選ばれる．この場合は，両方のサンプルはコスト値の最小化により統合される．従って，表に含まれるサンプルの数は増加しない．それ以外の場合は，以下のいずれかが適用される．
- (c) 表に含まれるサンプルの数は最大数の制限値 $M$ より小さい．この場合は，新たなサンプル $s$ は表に追加される．それ以外の場合は，(d) が適用される．
- (d) 表に含まれるサンプルのうち，$t.x^l_{p_i,i}$ の値が $s.x^l_{p_i,i}$ の値に最も近い $t$ を選択する．この場合は，両方のサンプルはコスト値の最小化により統合される．従って，表に含まれるサンプルの数は増加しない．

上記 (b) の場合は，新たなサンプル $s$ について，$t.x^l_{p_i,i}$ の値が $s.x^l_{p_i,i}$ の範囲内にある $t$ が比較される．この範囲を評価するために，$t$ と $s$ の距離が次のように定義される．

$$dis(s,t) = |s.x^l_{p_i,i} - t.x^l_{p_i,i}|. \tag{7}$$

各エージェント $i$ について，サンプルの範囲は閾値 $dis_i$ を用いて定義される．$dis(s,t) \leq dis_i$ であるならば，$t$ は $s$ の範囲内である．閾値 $dis_i$ は $x^l_{p_i,i}$ が取りうる値に基づいて以下のように定められる．先ず，$D^r_i$ と各子エージェント $j$ についての $D^l_{i,j}$ の最小値および最大値をそれぞれ合計することにより，$x^l_{p_i,i}$ の最小値 $x^{lmin}_{p_i,i}$ および最大値 $x^{lmax}_{p_i,i}$ をそれぞれ求める．これらの値に基づいて $dis_i$ は以下のように計算される．

$$dis_i = (x^{lmax}_{p_i,i} - x^{lmin}_{p_i,i})/M. \tag{8}$$

新たなサンプル $s$ について，$s.g^*(x^l_{p_i,i})$ と表から選択されたサンプル $t$ の $t.g^*(x^l_{p_i,i})$ が比較される．そして，コスト値が高い方のサンプルが除去される．

また，$dis(s,t)$ は上記 (d) の場合のサンプル間の距離の評価に用いられる．

この方法は，サンプルの分布を考慮しつつ，より良いコスト値を維持することを意図した発見的手法である．しかし，サンプル数が大幅に削減された場合には，解の実行可能性と品質が減少する可能性がある．

*1 次節以降の提案手法では，境界値を含めれば実行可能だがサンプル値は実行不可能なサンプルが表に含まれる．この場合はサンプルに実行可能性があるものを優先し，さらにコスト値を最小化した．

### 3.2 実行可能性についての境界

本研究で対象とする資源割り当て問題では，解の実行可能性は重要である．サンプルの除去は実行可能性を減少させるため，残されたサンプルから実行可能な解を生成するための追加的な手法が必要である．本研究の資源割り当て問題では二つの制約条件があるが，実行不可能性は主に式 (1) で表される資源の量の合計に関する制約によって発生すると考えられる．サンプル数の削減によって，根ノードではこの制約条件を満足する解，すなわち資源の量の合計がゼロになる解が得られなくなる．そこで，資源の量が連続的であるという前提のもとで，資源の量の可能な割り当てについて，境界を計算するように解法を拡張する．

各エージェント $i$ の表の各サンプル $s$ について，$s.x_{p_i,i}$ の下界 $s.x^{\perp}_{p_i,i}$ と上界 $s.x^{\top}_{p_i,i}$ を導入する．もしも，$i$ を根とする部分木の問題について，もともとの資源の量が連続的ではなく，離散的であれば，$s.x^{\perp}_{p_i,i} = s.x_{p_i,i} = s.x^{\top}_{p_i,i}$ である．すなわち，資源の量の余剰を吸収する余地はない．しかし，連続的な資源の量の標本を表す場合には，資源の量の可能な範囲に基づいて，境界を決定できる．

資源の量 $R_i$ の範囲は，$R_i$ に含まれる最小値と最大値により，$r^{\perp}_i$ および $r^{\top}_i$ のように定義される．この範囲により，$r \in R_i$ の境界 $r^{\perp}$ および $r^{\top}$ は $r^{\perp}_i \leq r^{\perp} \leq r$ および $r \leq r^{\top} \leq r^{\top}_i$ のように定義される．さらに，リンク $(i,j)$ について，資源の量 $l_{i,j}$ の範囲は，2.1 節に示されるように，$-l^c_{i,j} \leq l_{i,j} \leq l^c_{i,j}$ により定義される．この範囲に基づいて，資源の量 $l$ の境界 $l^{\perp}$ および $l^{\top}$ は，$r^{\perp}$ および $r^{\top}$ と同様に表される．サンプル $s$ についての $s.x_{p_i,i}$ の境界は，これらの要素に基づいて計算される．この境界 $x^{l\perp}_{p_i,i}$ および $x^{l\top}_{p_i,i}$ は次のように計算される．

$$x^{l\perp}_{p_i,i} = \max\left(-l^c_{p_i,i}, x^{r\perp}_i + \sum_{j\in Ch_i} x^{l\perp}_{i,j}\right) \tag{9}$$

$$x^{l\top}_{p_i,i} = \min\left(l^c_{p_i,i}, x^{r\top}_i + \sum_{j\in Ch_i} x^{l\top}_{i,j}\right) \tag{10}$$

ここで，$x^{r\perp}_i$ と $x^{r\top}_i$ は，$x^r_i = r$ であるような，$r^{\perp}$ と $r^{\top}$ である．

上述のように，$r \in R_i$ の境界 $r^{\perp}$ および $r^{\top}$ は可能な範囲から選択されるため，その選択のための複数の戦略が考えられる．ここでは，最も広い境界を用いることとし，$r^{\perp}_i = r^{\perp}$，$r^{\top} = r^{\top}_i$ とする．

コスト値のボトムアップな計算は上記の境界を用いて拡張される．すなわち，資源の量の合計の際に，それぞれに付随する境界が同時に計算される．また，このとき，式 (1) および (2) の各制約条件は，境界による余裕を考慮するように修正される．

根ノードでは，資源の量の合計がゼロになるようなサンプルが存在しないとき，この境界を用いて資源の量の誤差を許容することにより，実行可能解を得ることを試みる．このとき，サンプル $s$ ついて，$e^l_i = 0 - s.x^l_{p_i,p}$ なる誤差を $s.x^l_{p_i,p}$ に加算することにより，$s$ は実行可能性を満足する．解を決定するトップダウンな計算では，サンプル $s$ に対応する自身および子ノードへの資源の量の境界を考慮して，誤差 $e^l_i$ を各ノードに配分する．この計算は次のように表される．

$$e^l_i = e^r_i + \sum_{j\in Ch_i} e^l_{i,j} \tag{11}$$

$$\text{where } s.x^{r\perp}_i \leq s.x^r_i + e^r_i \leq s.x^{r\top}_i \tag{12}$$

$$s.x^{l\perp}_{i,j} \leq s.x^l_{i,j} + e^l_{i,j} \leq s.x^{l\top}_{i,j} \tag{13}$$

表 1: 実行可能解の割合と最大の表のサイズ

| cost func. | alg. | capacity of links | | | |
|---|---|---|---|---|---|
| | | $[-\infty,\infty]$ | | [-500,500] | |
| | | feasibility [%] | max. sz. of tbl. | feasibility [%] | max. sz. of tbl. |
| random | exact | 100 | 6201 | 100 | 1001 |
| | lmt100rnd | 88 | 100 | 88 | 100 |
| | lmt10 | 44 | 10 | 44 | 10 |
| | lmt100 | 40 | 100 | 48 | 100 |
| | lmt10-res | 100 | 10 | 100 | 10 |
| | lmt100-res | 100 | 100 | 100 | 100 |
| linear | exact | 100 | 6201 | 100 | 1001 |
| | lmt100rnd | 40 | 100 | 40 | 100 |
| | lmt10 | 32 | 10 | 32 | 10 |
| | lmt100 | 52 | 100 | 52 | 100 |
| | lmt10-res | 100 | 10 | 100 | 10 |
| | lmt100-res | 100 | 100 | 100 | 100 |
| quadratic | exact | 100 | 6201 | 100 | 1001 |
| | lmt100rnd | 40 | 100 | 40 | 100 |
| | lmt10 | 32 | 10 | 32 | 10 |
| | lmt100 | 56 | 100 | 56 | 100 |
| | lmt10-res | 100 | 10 | 100 | 10 |
| | lmt100-res | 100 | 100 | 100 | 100 |

表 2: 解のコスト値

| cost func. | alg. | capacity of links | | | | | |
|---|---|---|---|---|---|---|---|
| | | $[-\infty,\infty]$ | | | [-500,500] | | |
| | | cost value | | | cost value | | |
| | | min | max | ave | min | max | ave |
| random | exact | 197 | 366 | 286 | 218 | 336 | 280 |
| | lmt10-res | 413 | 27331 | 13305 | 314 | 29026 | 14198 |
| | lmt100-res | 413 | 27331 | 14275 | 252 | 29026 | 13421 |
| linear | exact | 6 | 3367 | 1009 | 6 | 3367 | 1009 |
| | lmt10-res | 6 | 6455 | 2037 | 6 | 6401 | 2054 |
| | lmt100-res | 6 | 6501 | 1947 | 6 | 6449 | 1957 |
| quadratic | exact | 6 | 117390 | 20772 | 6 | 117390 | 20772 |
| | lmt10-res | 12 | 160304 | 33258 | 12 | 183161 | 40071 |
| | lmt100-res | 6 | 161127 | 40885 | 6 | 183689 | 46372 |

誤差の配分のための複数の戦略が考えられる．本研究では，初期の検討として，可能な限り誤差の範囲の広さに比例して誤差を配分する．エージェント $i$ は，子のエージェント $j$ に，解として選択した $s$ についての割り当て $s.x^l_{i,j}$ と誤差の配分 $e^l_{i,j}$ を通知する．子のエージェントは同様に資源の割り当てと誤差の配分を決定する．

上記の方法により実行可能性が増加する．その一方で，資源の割り当ての量の誤差を許容するために，解品質は減少する可能性がある．

## 4. 評価

提案手法を実験により評価した．フィーダツリーの資源供給網を動機付けとする例題を用いた．問題は [Miller 12, Matsui 12] と類似するが，本研究の評価の目的のためにパラメータを調整した．問題は 50 ノードと 49 リンクから成る．木構造は端数となるノードを除いて可能な限り 2 分木とした．簡単のために，資源の量の範囲 $R_i$ は全てのエージェントで同一とした．$R_i$ は $[-100, 100]$ の整数値の集合とした．また，リンクの容量 $l^c_{i,j}$ は全てのリンクで同一の値 $\infty$ または 500 とした．

コスト関数 $f(r_i)$ として，次の関数を用いた．

- random: $f(r_i)$ は一様分布したがって $[1, 1000]$ の範囲の整数値をランダムに取る．
- linear: 各ノード $i$ は最も好ましい資源の量 $r^*_i$ を持つ．$r^*_i$ は一様分布に従ってランダムに定めた．さらにノード $i$ は $[1, 10]$ の範囲の整数値の重み $w_i$ を持つ．これらにより $f(r_i)$ は $w_i \cdot |r^*_i - r_i|$ のように定義される．
- quadratic: linear と同様だが $f(r_i)$ は $w_i \cdot (r^*_i - r_i)^2$ のように定義される．

これらの問題をそれぞれ 25 問について評価した．次の各手法を比較した．exact: 厳密な解法．lmt100rnd: 表に含まれるサンプルの数 $M$ を 100 に制限し，表のサンプル数が $M$ に達したら，新たに追加するサンプルと表からランダムに選択したサンプルをコスト値の最小化により統合する解法．lmt10/100: 表に含まれるサンプルの数 $M$ を 10 および 100 に制限する提案手法．lmt10/100-res: 上記 lmt10/100 に加えて，各サンプルに資源の量の誤差の範囲を与える解法．

表 1 に実行可能解の割合と最大の表のサイズを示す．これらの問題では表の規模が比較的大きいため，lmt10/100 の制限の影響が大きく，実行可能解の割合が減少した．また，コスト関数が random の場合は lmt100rnd よりも実行可能解の割合が少ない傾向が見られた．その一方で，lmt10/100-res ではサンプルに与えられた誤差範囲により実行可能な解を構成できた．

表 2 に解のコスト値を示す．lmt10/100-res はサンプル数の削減の影響により，解のコスト値が増加した．これらの結果では random のコスト値の増加が他の場合よりも顕著である．提案手法では，初期の検討として，各サンプルの誤差の境界の範囲に比例して誤差を配分したことが，特にコスト値が連続的ではないランダムなコスト関数の場合に解品質が大きく低下したことの一因と考えられる．コスト関数の特徴に応じて適切に誤算を配分するための，より洗練された手法が必要である．

## 5. おわりに

本研究では，ネットワークを介して共有される分散資源の割り当てのための分散制約最適化手法において，連続変数を離散化する場合や変数の値域が大規模である場合に，実行可能性を維持しつつ問題の規模を抑制する近似的な解法について検討した．実行可能性と解の精度の双方を考慮するような，より高度な解法が今後の課題である．

## 参考文献


[Mailler 04] Mailler, R. and Lesser, V.: Solving distributed constraint optimization problems using cooperative mediation, in *3rd International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 438–445 (2004)

[Matsui 08] Matsui, T., Silaghi, M., Hirayama, K., Yokoo, M., and Matsuo, H.: Resource Constrained Distributed Constraint Optimization with Virtual Variables, in *23rd AAAI Conference on Artificial Intelligence*, pp. 120–125 (2008)

[Matsui 12] Matsui, T. and Matsuo, H.: Considering Equality on Distributed Constraint Optimization Problem for Resource Supply Network, in *2012 IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pp. 25–32 (2012)

[Miller 12] Miller, S., Ramchurn, S. D., and Rogers, A.: Optimal decentralised dispatch of embedded generation in the smart grid, in *11th International Conference on Autonomous Agents and Multiagent Systems*, Vol. 1, pp. 281–288 (2012)

[Modi 05] Modi, P. J., Shen, W., Tambe, M., and Yokoo, M.: Adopt: Asynchronous distributed constraint optimization with quality guarantees, *Artificial Intelligence*, Vol. 161, No. 1-2, pp. 149–180 (2005)

[Petcu 05] Petcu, A. and Faltings, B.: A Scalable Method for Multiagent Constraint Optimization, in *19th International Joint Conference on Artificial Intelligence*, pp. 266–271 (2005)