1L1-R-7-1

エージェント間のコスト配分の均衡を考慮する分散制約最適化手法の検討

Distributed Constraint Optimization to Allocate Cost Values based on Fairness

松井俊浩　松尾啓志
Toshihiro Matsui　Hiroshi Matsuo

名古屋工業大学
Nagoya Institute of Technology

Distributed constraint optimization problem is a fundamental problem in multi-agent systems. Scheduling and allocating distributed resources including sensor network and power supply network can be modeled with DCOPs. While previous studies mainly address the optimization of globally total cost/utility values, the cost/utility values have to be evenly allocated to all agents in several practical problems in which agents want to divide shared resource. To represent that class of problems, we introduce a criterion that minimizes difference of the allocated cost values. We also present solution methods that resemble algorithms for multi-objective DCOPs. The proposed methods are experimentally evaluated with resource constrained DCOPs for shared resources.

1. はじめに

マルチエージェントシステムの基礎的研究である分散制約最適化問題は，分散資源のスケジューリングや割り当て問題への応用が期待される．特に本研究では，ネットワーク上の複数のノードに分散して存在する共有資源を，各ノードに適切に配分するような資源割り当て問題を対象とする．このような複数のノードに存在する共有資源を適切に再配分する問題は，分散電源を含む電力スマートグリッドにおける資源の需要と供給などに類似する点で重要であると考えられる．

マルチエージェントシステムにおける基本的な問題である分散制約最適化問題 (DCOP) に共有資源の概念を導入した資源制約付き DCOP(RCDCOP)[1] が提案されている．DCOP では，マルチエージェントシステムの協調問題解決を，エージェントに分散して配置された，組み合わせ最適化問題として表現する．そして，評価関数を結合した大域的な評価値を最適化するような変数値の組み合わせを，エージェント間のメッセージ交換を伴う分散アルゴリズムを用いて探索する．さらに RCDCOP では，各エージェントは自身の変数の選択に関して，ある量の資源を要求する．利用可能な資源の総量は大域的制約により制限される．上述の共有資源の割り当て問題は，RCDCOP として定式化できる．

従来の (RC)DCOP は，評価値の大域的な合計を最小化または最大化することが目的であり，個々のエージェントの評価値の公平性は考慮されていない．その一方で，資源の配分においては，各エージェントの要求に対する不均衡は減少されるべきである．[2] では，ネットワーク上の共有資源割り当て問題のための RCDCOP の解法に，エージェント間の評価値の不均衡を軽減する発見的手法を導入したが，不均衡さの軽減は最適化の指標としては扱われていない．そこで，本研究では，系全体のエージェントの局所的なコスト値の，最大値と最小値の差分を，最小化する最適化の指標を解法に導入し，従来手法と比較する．

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

2. 準備

2.1 分散資源の割り当てのモデル

資源供給ネットワーク上の複数のノードに分散して存在する資源を配分する，分散資源の割り当てのモデル [2] を示す．ここでは，電力網におけるフィーダツリーのような木構造のネットワークを対象とする．ネットワークは，次の要素により構成される．

- ノード： 資源の供給または需要の要求を持つ．要求される資源の量に関して評価値が与えられる．
- リンク：資源を移送する経路である．移送可能な資源の量には制限がある．

各ノード i における，資源についての需要と供給の要求は，次の要素により表される．

- R_i: 要求される資源の量の離散集合．資源の量 $r \in R_i$ が負値であれば供給を表し，正値であれば需要を表す．ノード i は R_i の要素のいずれかを選択する．
- $f_i(r)$: 資源の量 $r \in R_i$ のコスト値の関数．$f_i(r)$ は非負の離散値である．

リンク j による資源の移送は次の要素により表される．

- L_j: リンクを移送される資源の量の離散集合. 資源の量 $l \in L_j$ は，リンクの容量 l^c について $-l^c \leq l \leq l^c$ なる離散値であり，その値の組み合わせは原点に対して対称である．また，l の値の正負は，資源の移送の方向を表す．その値は，上述の木構造のネットワークにおける根からのノード順序に従うとき正である．

各ノード i では，i が要求する資源の量 r_i およびノード i と接続する各リンク j の資源の移送量 l_j の総和は，常に 0 でなければならない．この制約条件は，i と接続するリンクの集合 E_i を用いて次式のように表される．

$$r_i + \sum_{j \in E_i, l_j \in L_j} l_j = 0 \quad (1)$$

この制約条件のもとで，資源の量のコスト値 $f_i(r)$ を大域的に結合した値を最適化する．

2.2 RCDCOP としての表現および解法

前節のモデルにおける各ノードは，RCDCOP におけるエージェントとして表現される．各エージェントは，木構造のネットワークにより順位づけされた親子関係を持つ．エージェント i の資源の要求の量 r_i を，i の変数 x_i^r を用いて表す．また，i は自身の子 j それぞれへの資源の配分を決定する．子 j への資源の配分の量を変数 $x_{i,j}^l$ を用いて表す．親 p_i からの資源の配分の量を変数 $x_{p_i,i}^l$ を用いて表す．$x_{i,j}^l, x_{p_i,i}^l$ の値は，子または親との間のリンクを移送される資源の量と対応する．

上記の問題に対して，分枝限定法と動的計画法にもとづく，RCDCOP の従来解法 [1] を適用する*1．この解法は制約網についての疑似木による変数順序に依存する．また，複数の異なる種類の資源に関して，要求量の総和それぞれを上限値以下とする制約条件を扱う．本研究で扱う問題は，木構造のネットワークと，1 種類の資源の要求量の総和を 0 とする資源制約を用いるため，従来解法で対象とされる問題の部分クラスであるといえる．厳密には，資源の量が正負値である点や，ノードの資源の要求量およびコスト関数の表現が異なるが，本質は同様である．

解法において集計される，親からの資源の配分の量および，ノード i を根とする部分木に関する最適コスト $g_i^*(\{(x_{p_i,i}^l, d_{p_i,i})\})$ は次のように定義される．

$$
\begin{aligned}
& g_i^*(\{(x_{p_i,i}^l, d_{p_i,i})\}) \\
& = \min_{d_i \in R_i, d_{i,j} \in L_{i,j}} g_i(\{(x_{p_i,i}^l, d_{p_i,i}), (x_i^r, d_i)\} \\
& \cup \bigcup_{j \in C_i} \{(x_{i,j}^l, d_{i,j})\}) \qquad (2)
\end{aligned}
$$

$$
\begin{aligned}
& g_i(\{(x_{p_i,i}^l, d_{p_i,i}), (x_i^r, d_i)\} \cup \bigcup_{j \in C_i} \{(x_{i,j}^l, d_{i,j})\}) \\
& = \delta_i(\{(x_{p_i,i}^l, d_{p_i,i}), (x_i^r, d_i)\} \cup \bigcup_{j \in C_i, d'_{i,j} = d_{i,j}} \{(x_{i,j}^l, d'_{i,j})\}) \\
& \oplus \bigoplus_{j \in C_i, d'_{i,j} = d_{i,j}} g_i^*(\{(x_{i,j}^l, d'_{i,j})\}) \qquad (3)
\end{aligned}
$$

$$
\begin{aligned}
& \delta_i(\{(x_{p_i,i}^l, d_{p_i,i}), (x_i^r, d_i)\} \cup \bigcup_{j \in C_i} \{(x_{i,j}^l, d_{i,j})\}) \\
& = \begin{cases} f_i(d_i) & \text{式 (1) の資源制約を満足する} \\ \infty & \text{otherwize} \end{cases} \qquad (4)
\end{aligned}
$$

ここで，C_i は i の子ノードの集合である．実際の計算では，探索の過程で未知のコスト値を含む部分問題を評価するために，下限値 0 と上限値 ∞ にもとづく上下界値が用いられる．

解法は，コストの集計と，最適解の決定の 2 つの段階からなる．コストの集計では，木に従って，ボトムアップに大域的な最適コスト値が計算される．分枝限定法を併用する手法では，コストの集計は木探索により時分割的に実行される．いずれ，根ノードでは，大域的最適コストが得られる．その後，木に従ってトップダウンに，最適な変数値が決定される．解法の詳細は [1, 3, 4] などを参考にされたい．

*1 ただし，本問題に適合するように再構成した．また，簡単のために動的計画法と木探索の機構のみを用い，大域的コストの上界値に基づく枝刈りは除去した．

2.3 エージェント間のコスト配分の均衡を考慮する最適化

従来の (RC)DCOP では，大域的なコストの合計を最適化するため，コストの結合演算 $\oplus$ は加算である．しかし，不均衡性の軽減を重視する場合は別の結合演算を用いることも可能である．[2] では，最大化関数 (max) を用い，全エージェントのコスト値の最大値を最小化する手法が評価され，コスト値の合計を最小化するよりも探索空間が削減されることが示されている．

その一方で，コスト値の合計または最大値を最小化する指標のみでは，各エージェントの，局所コストの不均衡を抑制できない．この問題点への対処として，最適解を決定する処理に発見的手法が導入されている．この手法では，まず，解法が木に従ってボトムアップにコスト値を集計する際に，各部分木ごとのコスト値を記憶しておく．コスト値の合計を最小化する場合は，その計算結果を保持しておけばよい．コスト値の最大値を最小化する場合は，本来必要な計算と同時に集計することができる．そして，いずれは根ノードで最適なコスト値が得られ，各エージェントは木に従ってトップダウンに最適な変数値を選択する．この際に複数の解の候補がある場合は，各候補について，自身および各子以下のサブツリーそれぞれに配分されるコスト値からなるベクトルを計算する．そして，このようなベクトル全ての中間値を要素とするベクトルを構成する．解の候補についてのベクトルが，中間値からなるベクトルに最も近いものを選択し，解を決定することによりコスト値の均一化を図る．

3. 提案手法

3.1 コスト値ベクトルの導入

従来研究では，ネットワーク上の共有資源割り当て問題のための RCDCOP の解法に，エージェント間の評価値の不均衡を軽減する発見的手法を導入したが，不均衡静さの軽減は最適化の指標としては扱われてはいない．すなわち，不均衡さに関する最適性の保証は無い．このような不均衡さの軽減を最適化の目的とすることは比較的難しい課題であると考えられる．たとえば，分散などの二次的な統計量を最適化するためには，コストの集計に先だって大域的なコストの合計をあらかじめ知る必要があるため，部分問題に分解された計算においては直接評価することができない．本研究では，系全体のエージェントの局所的なコスト値の最大値と最小値の差分を最小化する最適化の指標を解法に導入し，従来手法と比較する．この手法は，コストの最小値と最大値をともに集計する手法であり，多目的分散制約最適化問題 [5] と関連するといえる．

提案手法では，コスト値は，最小値と最大値をあらわすベクトルとして表現される．すなわち，式 (3) で定義されるコスト値は，ベクトル

$$\mathbf{g}_i = [\underline{g}_i, \overline{g}_i] \qquad (5)$$

に拡張される．$\underline{g}_i$ では，コストの結合演算 $\oplus$ を最小化関数 (min) として計算する．$\overline{g}_i$ では，$\oplus$ を最大化関数 (max) として計算する．

もはや，このようなベクトルの集合から唯一の「最小な」ベクトルを選択することは一般にはできない．したがって，式 (2) の最適コスト値は，コスト値ベクトルの集合 G_i^* に拡張される．これに伴って，式 (2) の最小化は，ベクトルの集合から冗長なベクトルのみを除去するような適切なフィルタに置き換えられる．

表 1: 線形なネットワークの結果

prb.	(a) $n^c = 10, r^{all} = 20$				(b) $n^c = 11, r^{all} = 20$							
alg.	itr.	cost			itr.	cost						
		ave.	dif.	var.		ave.	dif.	var.				
sum., greedy	168.5	0	0	0	173.8	0.182	2	0.331				
sum., center	168.5	0	0	0	173.8	0.182	1	0.149				
max., greedy	176.8	0	0	0	154.9	0.182	1	0.149				
max., center	176.8	0	0	0	158.0	0.182	1	0.149				
diff.	185.7	0	0	0	202.8	0.196	1	0.155				
diff + max.	163.9	0	0	0	174.6	0.182	1	0.149				
diff + sum.	163.9	0	0	0	174.1	0.182	1	0.149				
prb.	(c) $n^c = 10, r^{all} = 10$				(d) $n^c = 11, r^{all} = 10$				(e) $n^c = 10, r^{all} = 8$			
alg.	itr.	cost			itr.	cost			itr.	cost		
		ave.	dif.	var.		ave.	dif.	var.		ave.	dif.	var.
sum., greedy	254.4	0.909	2	0.992	275.7	1.091	2	0.992	200.7	1.091	2	0.992
sum., center	254.4	0.909	1	0.083	275.7	1.091	1	0.083	200.7	1.091	2	0.264
max., greedy	96.5	0.909	1	0.083	146.6	1.091	2	0.992	135.6	1.091	2	0.992
max., center	96.5	0.909	1	0.083	146.6	1.091	1.2	0.119	135.6	1.091	2	0.264
diff.	124.5	1.069	1	0.092	198.3	1.149	1	0.109	162.4	1.091	2	0.264
diff + max.	120.4	0.909	1	0.083	183.5	1.091	1	0.083	158.2	1.091	2	0.264
diff + sum.	151.1	0.909	1	0.083	200.3	1.091	1	0.083	158.7	1.091	2	0.264

式 (2), (3) は次のように置き換えられる．

$$
\begin{aligned}
&G_i^*(\{(x^l_{p_i,i}, d_{p_i,i})\}) \\
&= filter(\bigcup_{d_i \in R_i, d_{i,j} \in L_{i,j}} G_i(\{(x^l_{p_i,i}, d_{p_i,i}), (x^r_i, d_i)\} \\
&\cup \bigcup_{j \in C_i} \{(x^l_{i,j}, d_{i,j})\})) \qquad (6)
\end{aligned}
$$

$$
\begin{aligned}
&G_i(\{(x^l_{p_i,i}, d_{p_i,i}), (x^r_i, d_i)\} \cup \bigcup_{j \in C_i} \{(x^l_{i,j}, d_{i,j})\}) \\
&= \{\boldsymbol{\delta}_i(\{(x^l_{p_i,i}, d_{p_i,i}), (x^r_i, d_i)\} \cup \bigcup_{j \in C_i, d'_{i,j} = d_{i,j}} \{(x^l_{i,j}, d'_{i,j})\})\} \\
&\oplus \bigoplus_{j \in C_i, d'_{i,j} = d_{i,j}} G_i^*(\{(x^l_{i,j}, d'_{i,j})\}) \qquad (7)
\end{aligned}
$$

ただし，式 (7) の $\boldsymbol{\delta}_i$ は，$\boldsymbol{\delta}_i = [\delta_i, \delta_i]$ なるベクトルである．また，$\oplus$ は，被演算要素の 2 集合の直積に含まれる，コスト値ベクトルの組それぞれを結合した要素からなる，ベクトルの集合を返すように拡張される．

実際には，コスト値ベクトルにおいても，探索における未知のコスト値について上下界値が導入される．そのため，コスト値ベクトルの集計の演算およびフィルタは，上下界値について一般化される．

解法の処理は，基本的にはスカラ値の場合と同様である．ただし，最適性の判定は根ノードのエージェントのみが行うことができる．コスト値ベクトルの集合に含まれる各要素の境界値が収束したとき，根ノードは，コストの最大値と最小値の差分値が最も小さい解を選択する．同様に，最適解が木に従ってトップダウンに決定される．このとき，根ノードが選択したコストの最小値と最大値も伝搬され，各ノードは，自身以下の部分木のコスト値がこれらを超えないように解を決定する．

3.2 コスト値ベクトルの集合に対するフィルタ

式 (6) の $filter$ は冗長なベクトルを除去する．ここでは，最大値と最小値の差分の最小化という目的を考慮し，2 つのベクトル間に内包関係がある場合に，外側のベクトルを除去する．すなわち，

$$\mathbf{g} = [\underline{g}, \overline{g}], \mathbf{g}' = [\underline{g}', \overline{g}'], \underline{g} \leq \underline{g}' \wedge \overline{g}' \leq \overline{g} \qquad (8)$$

であるとき，$\mathbf{g}$ は除去される．また，これに先だって実行不可能解を表す要素 $[\infty, \infty]$ は集合から除去する．根のエージェントにおいて $G_i^* = \phi$ であれば，問題は大域的に実行不能である．

3.3 追加的な手法

上述の提案手法に加えて，2 つの手法を提案する．一つは，フィルタにおいて，コストの最大値がより大きいベクトルを除去する方法である．すなわち，

$$\mathbf{g} = [\underline{g}, \overline{g}], \mathbf{g}' = [\underline{g}', \overline{g}'], \overline{g}' \leq \overline{g} \qquad (9)$$

であるとき，$\mathbf{g}$ は除去される．この条件は式 (8) の条件を含む．最大のコスト値を最小化する指標を反映するものといえる．

もう一つの手法では，コスト値ベクトルの要素である，最小値と最大値に加えて，従来のコスト値の合計を追加し，3 次元ベクトルとする．また，フィルタにおいては，コストの合計値が大きい要素を除去する条件を追加する．合計コスト値を最小化する指標を反映するものといえる．

4. 評価

提案手法の有効性を実験により評価した．例題として，ノード数 11 の問題において，共有資源のパラメータの設定を 5 種類用いた．全ノードのうち 1 つは供給可能な資源の容量 r^{all} を持ち，他は常に資源を消費するものとした．資源を持つノードについては，資源の消費の有無を選択し，資源を消費する場合は，他のエージェントと同等の要求を持つものとした．資源を要求するエージェントの数を n^c で表す．各ノード i が資源

表 2: 二分木のネットワークの結果

prb.	(a) $n^c = 10, r^{all} = 20$				(b) $n^c = 11, r^{all} = 20$			
alg.	itr.	cost			itr.	cost		
		ave.	dif.	var.		ave.	dif.	var.
sum., greedy	182.8	0	0	0	221.2	0.182	2	0.331
sum., center	182.8	0	0	0	221.2	0.182	1	0.149
max., greedy	161.9	0	0	0	208.2	0.262	1	0.185
max., center	161.9	0	0	0	208.2	0.182	1	0.149
diff.	238.9	0	0	0	239.5	0.313	1	0.208
diff + max.	209.0	0	0	0	221.0	0.313	1	0.208
diff + sum.	175.1	0	0	0	205.1	0.284	1	0.195

prb.	(c) $n^c = 10, r^{all} = 10$				(d) $n^c = 11, r^{all} = 10$				(e) $n^c = 10, r^{all} = 8$			
alg.	itr.	cost			itr.	cost			itr.	cost		
		ave.	dif.	var.		ave.	dif.	var.		ave.	dif.	var.
sum., greedy	249.3	0.909	2	0.992	252.6	1.091	2	0.992	232.8	1.091	2	0.992
sum., center	249.3	0.909	1	0.083	252.6	1.091	1	0.083	232.8	1.091	2	0.264
max., greedy	194.2	0.909	1	0.083	212.0	1.164	2	0.952	211.9	1.091	2	0.963
max., center	194.2	0.909	1	0.083	212.0	1.091	1.2	0.119	211.9	1.091	2	0.264
diff.	249.4	1.164	1	0.152	254.6	1.200	1	0.139	242.4	1.091	2	0.774
diff + max.	229.5	0.909	1	0.083	246.6	1.164	1	0.129	244.9	1.091	2	0.774
diff + sum.	226.0	0.909	1	0.083	232.7	1.098	1	0.087	239.4	1.091	2	0.774

を要求する量は 2 から 0 であり，対応するコスト関数 f_i の値はそれぞれ 0 から 2 である．資源を伝達するリンクの容量は十分な大きさとした．

従来手法と提案手法を比較した．従来手法は最適化の指標として合計コスト (sum.) か最大コスト (max.) のいずれかを用いる．また，最適解の決定において，各エージェント貪欲的に自変数値を決定する (greedy) か，中間値ベクトル (center) を用いてコストを配分する．コストの最大値と最小値の差分を最小化する提案手法 (dif.) では，さらに 3.3 節で述べた，最大のコスト値を最小化 (+min.) または合計コスト値を最小化 (+sum.) する指標を併用する場合も評価した．

表 1 は，線形なネットワークにおける結果を表す．(a) は資源の過不足が無い場合であり，いずれの解法でもコスト値は 0 である．(b) は資源が不足する場合であり，コストの最大値と最小値の差分は 1 以上になる．コストの配分を考慮しない (sum, greedy) ではコスト値の分散が比較的大きい．また，コストの最小化を考慮しない dif. では，平均コスト値が比較的大きい．(c), (d), (e) の順に，さらに資源が不足する場合となる．コストの配分を考慮する各手法は，コスト値の分散を比較的抑制している．特に資源が不足する問題では，コストの結合が加算である sum よりも最大化である max の方が，コスト空間の規模が小さいために反復回数が少ないと考えられる．

表 2 は，二分木のネットワークにおける結果を表す．表 1 と，ほぼ同様の傾向がみられるものの，提案手法 (dif. ほか) ではコスト値の平均と分散が比較的大きい．これは，木の場合には，複数の子へのコストの配分に自由度があるためと考えられる．提案手法は，系全体の局所コスト値の最大値と最小値の差分をのみを考慮するため，解の決定において均衡な配分 (center) を行う手法の方が分散を小さくする場合がある．

5. まとめ

ネットワーク上の共有資源割り当て問題のための RCDCOP の解法において，エージェント間の評価値の不均衡を軽減する手法を検討した．特に，系全体のエージェントの局所的なコスト値の最大値と最小値の差分を最小化する最適化の指標を解法に導入し，従来手法と比較しておおむね同様の効果を得ることを実験により示した．提案手法は，多目的最適化の領域を活用する解法の検討としても意義があると考える．今後の課題として，異なる公平性の指標の導入と解法の開発，エージェントのグループごとに個別に公平性を保証するなどの実際的な指標の導入，探索の効率化が挙げられる．

謝辞

本研究の一部は，科研費 若手 (B)22700144 および平成 23 年度人工知能研究振興財団研究助成による．

参考文献

[1] T. Matsui, M. Silaghi, K. Hirayama, M. Yokoo, and H. Matsuo. Resource constrained distributed constraint optimization with virtual variables. In *AAAI 2008*, pp. 120–125, 2008.

[2] 松井, 松尾. 公平性を考慮した動的な分散資源配分のための協調問題解決手法の検討. 第 74 回情報処理学会全国大会, 2012.

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

[4] A. Petcu and B. Faltings. A scalable method for multiagent constraint optimization. In *IJCAI 2005*, pp. 266–271, 2005.

[5] F. M. Delle Fave, R. Stranders, A. Rogers, and N. R. Jennings. Bounded decentralised coordination over multiple objectives. In *AAMAS 11*, pp. 371–378, 2011.