

## 階層化された分散制約充足/最適化手法を用いた 分散センサ網における観測資源割り当ての検討

太田和宏 † 松井俊浩 † 松尾啓志 †

マルチエージェントシステムの応用分野の1つとして分散センサ網に関する研究が行われている。本論文では、分散センサ網の目的として注視制御が可能なセンサ群で対象の追跡を行う分散センサ網を想定する。特に、その重要な要素である観測資源割り当て問題に分散制約充足/最適化問題の枠組を適用する。従来の形式化では、問題の規模を大きくすると問題が複雑になり、解探索時間が大きくなるという問題があった。そこで、問題を2つの階層に分割し、それらの階層に対する解法を連携させることで、解に到達するまでのメッセージ交換サイクル数を削減する手法を提案する。シミュレーションにより提案手法を評価した結果、提案手法では少ないメッセージ交換サイクル数で解を発見できる事を示した。

### Hierarchized Distributed Constraint Satisfaction/Optimization for resource allocation problem in distributed sensor network

KAZUHIRO OTA, † TOSHIHIRO MATSUI † and HIROSHI MATSUO †

Distributed sensor network is an important research area of multi agent systems. We focus on a type of distributed sensor network systems that cooperatively observes multiple objects using multiple autonomous gaze control sensors. The sensor resource allocation problems of the distributed sensor network can be formalized as distributed constraint satisfaction/optimization problems. However, in the previous works, computation cost to solve the resource allocation problems highly increases with scale/density of the problems. In this work, we divide the problem into two layers of problems. Then two layered cooperative solvers are applied to those problems. The result of the experiment shows that proposing method reduces number of message cycles.

#### 1. はじめに

マルチエージェントシステムの協調問題解決の基礎的な領域として、分散制約充足/最適化問題が研究されている<sup>3)5)6)</sup>。分散制約充足/最適化問題では、エージェントの状態が変数として表現され、エージェント間の関係が制約/評価関数として表現される。各エージェントは互いに情報を交換しつつ自身の変数値を決定し、制約/目的関数を大域的に充足/最適化する変数値の割り当てを得る。このような表現は、分散システムにおける協調的な資源スケジューリングの本質的な問題を表すものとして重要である。また、マルチエージェントシステムの応用分野として分散センサ網に関する研究も広く行われている。また、分散センサ網の資源割り当て問題を分散制約充足/最適化問題として形式化する手法の研究もされている<sup>2)</sup>。分散センサ網には様々な目的があり、広域の観測情報の収集や移動ロボットによる協調的なセンシングなどが挙げられる。本論文では注視制御可能な自律的なセンサ群による観

測システムを想定する<sup>8)7)</sup>。複数の対象を複数のセンサにより注視するための制御は、観測資源の割り当て問題として捉えることができる。実際的な観測システムでは観測対象が移動するなど、環境は動的に変化する。そのため、変動する環境に追従して観測資源を割り当てる必要がある。しかし、このような動的な環境への追従は動的に変化する環境を表す時系列的な割り当て問題を反復的に解くこととして捉えることができる。そこで本論文では、初期の検討として、ある時刻の問題に対する解法を検討する。ある時刻の問題に対して割ける時間は限られており、解を高速に発見する必要がある。そのため比較的高速に解を見つけることが可能な確率的な解法を用いることが有用であるといえる。このような観測資源の割り当て問題においては、最適な観測資源の割り当てを決定するだけでなく、各エージェントが矛盾無く意思決定を行う必要がある。しかし、エージェントの協調と資源割り当てを含む問題は複雑であり、探索に要する時間の抑制には課題がある。そこで本論文では、問題をリーダーを選出する階層と割り当て問題を解決する階層の2つの階層に分割し、解を発見するまでの時間を削減する手法を提案する。これに

† 名古屋工業大学  
Nagoya Institute of Technology

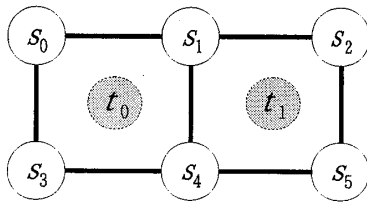


図1 グリッドモデルでの表現

より、問題が複雑になる事を抑制し解探索時間を削減できると期待される。以降、第2章では分散センサ網の資源割当て問題について説明し、第3章では分散制約充足/最適化問題について説明する。第4章では従来の形式化について説明し、第5章では提案手法について説明する。そして、第6章で評価を行う。

## 2. 分散センサ網の資源割当て問題

本論文では、センサの持つ観測資源を観測対象に割り当てる資源割当ての基礎的な表現として、センサがグリッド状に配置されたモデルを用いる。このようなモデルは関連研究<sup>1)2)</sup>でも用いられている。以後、このモデルをグリッドモデルと表記する。本論文で想定するシステムで計算資源をもつのはセンサのみで観測対象は制御の対象ではない。グリッドモデルで表現されたセンサ網の例を図1に示す。図1で  $s_i$  はセンサを示し、 $t_j$  は観測対象を示す。ここでは1つのグリッドに存在する観測対象は1個とした。各センサの観測範囲、すなわち視野には制限がある。その制限は妥協することは出来ない。ここでは、各センサは隣接するグリッドのみ観測可能とした。また、各センサが同時に観測できる範囲には制限があり、多数の対象物を注視することは出来ない。ここでは1個のセンサを同時に複数の観測対象に割り当てられないものとした。一方で、観測情報の精度や情報量の向上のために各観測対象には一定の観測資源が要求される。そこで、1個の観測対象に可能な限り3個のセンサを割り当てることを目的とする。しかし、この目的は妥協可能であり、3個未満のセンサで観測することが出来る。

## 3. 分散制約充足/最適化問題

### 3.1 形式化

分散制約充足/最適化問題 (DCSP/DCOP) は各エージェントに変数と制約、評価関数が分散して配置された問題として形式化される。変数の集合  $x_1, x_2, \dots, x_n$  があり、それぞれの変数は離散的な値の集合  $D_1, D_2, \dots, D_n$  に含まれるいずれかの値をとる。変数は複数のエージェントに分散して配置される。各変数の値を決定するのは変数をもつそのエージェントのみである。すなわち、変数はエージェントの状態、あるいは意思決定を表す。変数間の関係は制約として表現される。DCSPにおいては制約は満足すべき関係であり、DCOPでは各制約  $c$  に対応する評価関数  $f_c$  により、変数値の組み合わせについて評価値、すなわち制約に対する違反度が評価される。問題の目的は、すべての制約を満足する変数値の組み合わせ(充足解)を求めること、または評価関数の合計値が最小となる変数値の組み合わせ(最適解)を求めることである。

## 3.2 厳密解法

分散制約充足/最適化問題の最適解を求める厳密解法としてADOPT<sup>3)</sup>、DPOP<sup>4)</sup>が提案されている。ADOPTは深さ優先探索に基づいて分枝限定法/A\*に基づく探索を行う解法である。また、DPOPは動的計画法に基づく。厳密解法の問題点として、深さ優先探索木の induced-width<sup>4)</sup>にしたがって探索時間、記憶領域が指数関数的に増加することが挙げられる。このため、短時間かつ少ないメモリで解を見つけることが重要なシステムでは厳密解法を適用することには課題があるといえる。

## 3.3 確率的解法

一方、比較的短時間で準最適解を得ることを目的として、確率的解法 DSA<sup>5)</sup> や DSTS<sup>6)</sup> が提案されている。解探索に要する計算時間の点で、実システムでは、確率的解法が有用であると考えられる。本論文では特に DSTS をセンサの資源割当て問題の解法として適用する。DSTSは確率的なアプローチを用いた反復改善型のアルゴリズムである DSA にタブーサーチを組み込んだアルゴリズムである。DSTSでは各エージェントが互いの変数値を交換しつつ解を求める。各エージェントは、近傍のエージェントの変数値に基づいて、自身の各変数値に対する評価値を計算する。そして、ある確率  $p_1$  に基づいて変数値を最良の評価値を持つものに変更する。また、局所解から脱出するためにタブーサーチを用いる。変数値を変更する際に、以前の変数値への遷移をタブーリストに追加し一定期間(TABU期間)禁止する。タブーサーチのために、評価値の改善量  $\Delta$  が増加するような変数値しか選択出来ない場合がある。そのとき、自身の制約が充足されていないければ、確率  $p_2$  でその値に変数値を変更する。DSTSの詳細を図2に示す。図2中、variablesは変数の集合を示し、tabu\_listはタブーリストを示す。

## 4. 従来手法

この章では従来の形式化について説明する。従来の形式化には、センサと観測対象の組について変数を定義する STAV<sup>3)</sup> と観測対象について変数を定義する TAV<sup>2)</sup> がある。また、分散制約充足の枠組みとは別の協調モデルについても説明する。

### 4.1 STAV(Sensor-TargetAsVariable)

センサと観測対象の組について変数を定義する制約網の表現(STAV)について説明する。センサと観測対象の組を変数とするので、各センサは観測可能な観測対象の数だけ変数を持つ。図1の配置の制約網をSTAVで表現すると図3のようになる。図3において  $x_{t_j}^{s_i}$  はセンサ  $i$  が持つ観測対象  $j$  についての変数である。 $s_0, s_2, s_3, s_5$  は観測可能な観測対象が1個なので変数は1個である。 $s_1, s_4$  は観測可能な観測対象が2個なので持つ変数も2個である。 $x_{t_j}^{s_i}$  の値は、観測対象  $j$  を観測可能なセンサのうち、どのセンサが観測対象  $j$  に割り当てられているかを示す。観測対象  $j$  を観測可能なセンサの集合が  $\{s_0, \dots, s_n\}$  ならば、観測可能なセンサの組み合わせ  $\{\phi, \{s_0\}, \dots, \{s_n\}, \{s_0, s_1\}, \dots, \{s_0, \dots, s_n\}\}$  を

```

1 initialize own variables;
2 empty tabu_list
3 send values of variables to neighborhood agents;
4 until meet the end requirement do
5     receive other agents values;
6     call set_new_value;
7 if all values are in tabu_list then
8     send status that all values are in tabu_list to
        neighborhood agents;
9 else if new values of variables is assigned then
10    send own values of variables to neighborhood
        agents;
11    add new value to tabu_list;
12 end if
13 end while
14 procedure set_new_value
15 if all values are in tabu_list then
16    nothing to do;
17 else if all conflicting agents are in tabu state then
18    assign new values to variables;
19 else if  $\Delta \geq 0$  then
20    assign new values to variables with  $p_1$ ;
21 else if current violation > 0 then
22    assign new values to variables with  $p_2$ ;
23 end if
    
```

図2 DSTSの詳細

表す変数値をとる。制約は図3中  $c^0, c^1, c^2$  で示した3種類の制約がある。

- $c^0(x_{t_j}^{s_i})$ : 観測対象への観測資源割り当ての要求  
観測対象への観測資源割り当ての要求に関する単項制約である。観測対象  $j$  に割り当てられたセンサの数が3個に満たない場合に違反となる。この制約は緩和可能である。制約  $c^0$  に対する評価関数  $f_{c^0}$  は式(1)のように表される。ただし  $w^0$  は制約の違反度を示す定数値である。また  $n_j$  は観測対象  $t_j$  に割り当てられたセンサ数を示す。

$$f_{c^0}(x_{t_j}^{s_i}) = \begin{cases} w_0^0 & n_j = 0 \\ w_1^0 & n_j = 1 \\ w_2^0 & n_j = 2 \\ 0 & otherwise \end{cases} \quad (1)$$

- $c^1(x_{t_j}^{s_i}, x_{t_{j'}}^{s_{i'}})$ : 観測資源の制限  
各センサが同時に複数の観測対象に割り当てられることを制限する制約である。1個のセンサが同時に2個以上の観測対象に割り当てられた場合に違反となる。この制約は緩和不可能である。制約  $c^1$  に対する評価関数  $f_{c^1}$  は式(2)のように表される。ただし  $w^1$  は制約の違反度を示す定数値である。

$$f_{c^1}(x_{t_j}^{s_i}, x_{t_{j'}}^{s_{i'}}) = \begin{cases} w^1 & x_{t_j}^{s_i} \cap x_{t_{j'}}^{s_{i'}} \neq \phi \\ 0 & otherwise \end{cases} \quad (2)$$

- $c^2(x_{t_j}^{s_i}, x_{t_{j'}}^{s_{i'}})$ : 観測資源の割り当ての整合性  
同じ観測対象を観測可能なセンサが協調するための制約である。同じ観測対象に対する割り当ての組み合わせがセンサ間で異なっていた場合に違反となる。この制約は緩和不可能である。制約  $c^2$  に

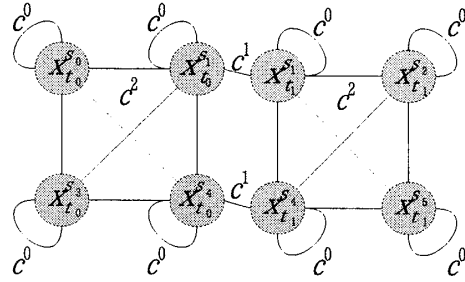


図3 STAVによる制約網の表現

対する評価関数  $f_{c^2}$  は式(3)のように表される。ただし  $w^{c^2}$  は制約の違反度を示す定数値である。

$$f_{c^2}(x_{t_j}^{s_i}, x_{t_{j'}}^{s_{i'}}) = \begin{cases} w^{c^2} & x_{t_j}^{s_i} \neq x_{t_{j'}}^{s_{i'}} \\ 0 & otherwise \end{cases} \quad (3)$$

#### 4.2 TAV(TargetAsVariable)

観測対象について変数を定義する制約網の表現(TAV)について説明する。観測対象について変数を定義するので各観測対象に関する変数は全体で1個である。図1の配置の制約網をTAVで表現すると図4のようになる。図4において  $x_{t_j}$  は観測対象  $j$  についての変数である。  $x_{t_j}$  の値は、観測対象  $j$  を観測可能なセンサが観測対象  $j$  に割り当てられているかを示す。観測対象  $j$  を観測可能なセンサの集合が  $\{s_0, \dots, s_n\}$  ならば、観測可能なセンサの組み合わせ  $\{\phi, \{s_0\}, \dots, \{s_n\}, \{s_0, s_1\}, \dots, \{s_0, \dots, s_n\}\}$  を表す変数値をとる。制約は図4中  $c^0, c^1$  で示した2種類ある。

- $c^0(x_{t_j})$ : 観測対象への観測資源割り当ての要求  
観測対象への観測資源割り当ての要求に関する単項制約である。観測対象  $j$  に割り当てられたセンサの数が3個に満たない場合に違反となる。この制約は緩和可能である。制約  $c^0$  に対する評価関数  $f_{c^0}$  は式(4)のように表される。ただし  $w^0$  は制約の違反度を示す定数値である。また  $n_j$  は観測対象  $t_j$  に割り当てられたセンサ数を示す。

$$f_{c^0}(x_{t_j}) = \begin{cases} w_0^0 & n_j = 0 \\ w_1^0 & n_j = 1 \\ w_2^0 & n_j = 2 \\ 0 & otherwise \end{cases} \quad (4)$$

- $c^1(x_{t_j}, x_{t_{j'}})$ : 観測資源の制限  
各センサが同時に複数の観測対象に割り当てられることを制限する制約である。また、この制約が変数間で協調するための制約である。1個のセンサが同時に2個以上の観測対象に割り当てられた場合に違反となる。この制約は緩和不可能である。制約  $c^1$  に対する評価関数  $f_{c^1}$  は式(5)のように表される。ただし  $w^1$  は制約の違反度を示す定数値である。

$$f_{c^1}(x_{t_j}, x_{t_{j'}}) = \begin{cases} w^1 & x_{t_j} \cap x_{t_{j'}} \neq \phi \\ 0 & otherwise \end{cases} \quad (5)$$

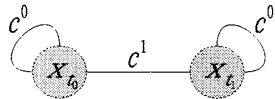


図4 TAVによる制約網の表現

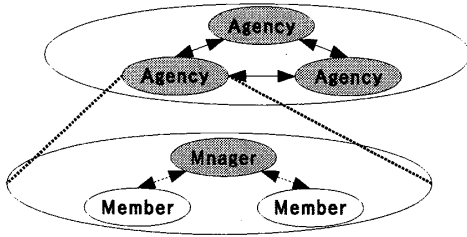


図5 エージェントによる協調モデル

TAVでは各観測対象について変数を定義するため、変数は観測対象の上に置かれる形になる。しかし、本論文で想定するシステムでは観測対象は計算資源を持たないため、観測対象は変数を持たず、TAVを実現することは出来ない。

#### 4.3 エージェントによる協調モデル

分散制約充足/最適化問題の枠組とは別にエージェントのグループであるエージェントを用いた分散協調処理による観測システムも提案されている<sup>8)7)</sup>。エージェントによる協調モデルの概念図を図5に示す。このシステムは視野制御可能なカメラエージェントと観測対象からなり、本論文で想定するシステムの条件と同じである。システムは自律的に動作するカメラエージェント(AVA: ActiveVisionAgent)の集合によって構成される。このシステムは実機を用いた小規模な実験環境で実証されている。このシステムの概要は以下の通りである。

- AVAを割当て可能な観測資源として扱う。
- AVAは観測対象を検出すると、同一観測対象毎にエージェントを作成する。
- 各エージェントには代表のAVA(マネージャ)が1個存在する。他のAVAはマネージャの意思決定に従う。
- 各マネージャは互いに情報を交換し、観測資源の配分を決定する。
- 観測対象の情報は各マネージャに集約される。

#### 5. 提案手法

STAVやTAVのような制約網による形式化に、エージェントによる協調モデルの持つ階層的な構造を導入することで、制約網による柔軟な問題の記述と、エージェントによる効率的な協調動作を統合することができると考えられる。本論文では、この様な形式化に基づく解法を提案する。すなわち、リーダー選出層と割当て問題解決層の、2つの比較的解くことが容易な問題の階層からなる制約網を用いた形式化を導入する。

##### 5.1 リーダー選出層

リーダー選出層では、各観測対象につき1個のセンサをリーダーとして選出する。エージェントの協調構造を各対象物に対して局所的に集約するためには、その観測対象を観測可能なセンサのいずれかがリーダーとなることが妥当である。リーダーはエージェントに

よる協調モデルのマネージャにあたり、リーダーは必ず注視に参加することとする。図1のような配置の場合、リーダー選出層の制約網は図6(a)のように表現される。図6(a)において $x_{t_j}^{s_i}$ はセンサ $i$ が持つ観測対象 $j$ についての変数である。 $x_{t_j}^{s_i}$ の値は、観測対象 $j$ のリーダーを示す。観測対象 $j$ を観測可能なセンサの集合が $\{s_0, \dots, s_n\}$ ならば、 $\{\phi, s_0, \dots, s_n\}$ のいずれかの値を表す変数値をとる。リーダー選出層における制約は、図6(a)中 $c^0, c^1, c^2, c^3, c^4$ で示した5種類の制約がある。

- $c^0(x_{t_j}^{s_i})$ :リーダーの選出

観測対象への観測資源割り当ての要求に関する単項制約である。観測対象 $j$ に対してリーダーが選出されていない場合に違反となる。この制約は緩和可能である。制約 $c^0$ に対する評価関数 $f_{c^0}$ は式(6)のように表される。ただし $w^{c^0}$ は制約の違反度を示す定数値である。

$$f_{c^0}(x_{t_j}^{s_i}) = \begin{cases} w_0^{c^0} & x_{t_j}^{s_i} = \phi \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

- $c^1(x_{t_j}^{s_i}, x_{t_j}^{s_{i'}})$ :観測資源の制限

各センサが同時に複数の観測対象のリーダーに選出される事を制限する制約である。1個のセンサが同時に2個以上の観測対象のリーダーに選出された場合に違反となる。この制約は緩和不可能である。制約 $c^1$ に対する評価関数 $f_{c^1}$ は式(7)のように表される。ただし $w^{c^1}$ は制約の違反度を示す定数値である。

$$f_{c^1}(x_{t_j}^{s_i}, x_{t_j}^{s_{i'}}) = \begin{cases} w^{c^1} & x_{t_j}^{s_i} = x_{t_j}^{s_{i'}} \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

- $c^2(x_{t_j}^{s_i}, x_{t_j}^{s_{i'}})$ :観測資源の割り当ての整合性

同じ観測対象を観測可能なセンサが協調するための制約である。同じ観測対象についてのリーダーがセンサ間で異なっていた場合に違反となる。この制約は緩和不可能である。制約 $c^2$ に対する評価関数 $f_{c^2}$ は式(8)のように表される。ただし $w^{c^2}$ は制約の違反度を示す定数値である。

$$f_{c^2}(x_{t_j}^{s_i}, x_{t_j}^{s_{i'}}) = \begin{cases} w^{c^2} & x_{t_j}^{s_i} \neq x_{t_j}^{s_{i'}} \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

- $c^3(x_{t_j}^{s_i})$ :割当て候補の確保1

各観測対象の割当て候補の数が十分な数だけ確保されるようにリーダーを選出するための単項制約である。リーダーは必ず担当の観測対象に割り当てられるため、他の観測対象の割当て候補から除外される。そのため、もともと3個以上のセンサを割り当てられる観測対象に対して3個のセンサを割り当てられなくなる状況に陥る可能性がある。その状況を避けるために、可能な限り観測可能な対象数が少ないセンサを選ぶようこの制約を設ける。観測可能なセンサが一番少ないセンサ以外をリーダーに選出した場合に違反となる。この

制約は緩和可能である。制約  $c^3$  に対する評価関数  $f_{c^3}$  は式 (9) のように表される。ただし  $w^{c^3}$  は制約の違反度を示す定数値である。また、 $I$  は  $t_j$  を観測可能なセンサの集合を示し、 $v_i$  はセンサ  $i$  が観測可能な観測対象数を示す。

$$f_{c^3}(x_{t_j}^{s_i}) = \begin{cases} w^{c^3} & \exists i \in I, v_i < v_{x_{t_j}^{s_i}} \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

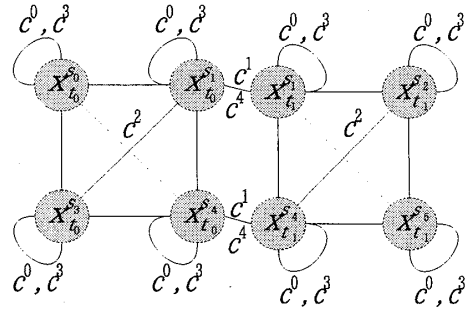
- $c^4(x_{t_j}^{s_i}, x_{t_{j'}}^{s_{i'}})$ : 割当て候補の確保 2  
 $c^3$  と同様に各観測対象の割当て候補の数が十分な数だけ確保されるようにリーダーを選出するための制約である。リーダーの観測可能な領域が重なっているということは、同じ観測対象を観測可能であり、割当て候補の制限につながる。そこで、可能な限り観測可能なエリアが重ならないようにリーダーを選出する。グリッドモデルでは観測可能な領域を4つに分割しており、重なっている観測可能領域の数が2つ以上である場合に違反となる。この制約は緩和可能である。制約  $c^4$  に対する評価関数  $f_{c^4}$  は式 (10) のように表される。ただし  $w^{c^4}$  は制約の違反度を示す定数値である。また、 $A_i$  はセンサ  $i$  が観測可能な領域の集合を示す。

$$f_{c^4}(x_{t_j}^{s_i}, x_{t_{j'}}^{s_{i'}}) = \begin{cases} w^{c^4} & |A_{x_{t_j}^{s_i}} \cap A_{x_{t_{j'}}^{s_{i'}}}| > 1 \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

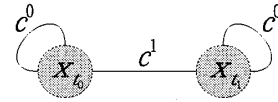
リーダー選出層では 4.1 と同様の制約網で表現されるため、変数の数は STAV を用いた従来手法と同じである上に制約が2つ増えている。しかし、各変数の取り得る値の集合の要素数は提案手法の方が少なくなる。グリッドモデルの場合、各観測対象を観測可能なセンサは4個なので従来手法では各変数の値域は  $2^4 = 16$  通りだが、リーダー選出では割当て候補センサのうちいずれかがリーダーになるかリーダーを選出しないの5通りである。各変数の取り得る値の集合の要素数が少ない方が比較的容易に解くことが出来ると考えられる。

### 5.2 割当て問題解決層

割当て問題解決層では、リーダー選出層で選出されたリーダーによって割当て問題を解決する。リーダー同士で情報を交換し問題を解き、リーダー以外のセンサはリーダーの決定に従う。リーダー選出層で各観測対象に1個のリーダーが選出されているため、図1のような配置の場合、制約網は図6(b)のように表現される。図6(b)において  $x_{t_j}$  は観測対象  $j$  のリーダーに選出されたセンサが持つ変数である。 $x_{t_j}$  の値は、観測対象  $j$  を観測可能なセンサが観測対象  $j$  に割り当てられているかを示す。観測対象  $j$  を観測可能なセンサの集合が  $\{s_0, \dots, s_n\}$  ならば、観測可能なセンサの組み合わせ  $\{\phi, \{s_0\}, \dots, \{s_n\}, \{s_0, s_1\}, \dots, \{s_0, \dots, s_n\}\}$  を表す変数値をとる。しかし、リーダーは自分が担当している観測対象に必ず割り当てられるため、各観測対象を観測可能なセンサの集合から除外される。リーダーによる割当て問題解決層における制約は、図6(b)



(a) リーダー選出層



(b) 割当て問題解決層

図6 提案手法における制約網

中  $c^0, c^1$  で示した2種類の制約がある。

- $c^0(x_{t_j})$ : 観測対象への観測資源割り当ての要求  
 観測対象への観測資源割り当ての要求に関する単項制約である。観測対象  $j$  に割り当てられたセンサの数が3個に満たない場合違反となる。この制約は緩和可能である。制約  $c^0$  に対する評価関数  $f_{c^0}$  は式 (11) のように表される。ただし  $w^{c^0}$  は制約の違反度を示す定数値である。また  $n_j$  は観測対象  $t_j$  に割り当てられたセンサ数を示す。

$$f_{c^0}(x_{t_j}) = \begin{cases} w_0^{c^0} & n_j = 0 \\ w_1^{c^0} & n_j = 1 \\ w_2^{c^0} & n_j = 2 \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

- $c^1(x_{t_j}, x_{t_{j'}})$ : 観測資源の制限  
 各センサが同時に複数の観測対象に割当てられることを制限する制約である。この制約が変数間で協調するための制約である。1個のセンサが同時に2個以上の観測対象に割り当てられた場合違反となる。この制約は緩和不可能である。制約  $c^1$  に対する評価関数  $f_{c^1}$  は式 (12) のように表される。ただし  $w^{c^1}$  は制約の違反度を示す定数値である。

$$f_{c^1}(x_{t_j}, x_{t_{j'}}) = \begin{cases} w^{c^1} & x_{t_j} \cap x_{t_{j'}} \neq \phi \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

割当て問題解決層の問題の表現は 4.2 の TAV と同様であるため、変数と制約の数が STAV で制約網を表現した場合よりも少ない。また、リーダーは必ず担当している観測対象に割り当てられるため、変数の取り得る値の集合も制限することになる。この2点から、割当て問題解決は STAV を用いた従来手法よりも比較的容易に解くことが出来ると考えられる。

### 5.3 層間の同期

提案手法では層間で同期をとる必要がある。一方で、各センサが自身の割り当て問題を解決するために必要なものは制約で関係しているセンサの情報のみである。すなわち、制約で関係するリーダーからの変数値を受

け取ることが出来れば、自身に関係する部分問題を解くことが出来る。したがって、自分がリーダーに選出されたと判断したとき、制約で関係する可能性のあるセンサに通知し、リーダー同士で変数値を含めたメッセージを交換すれば、割り当て問題に移行することができる。自分がリーダーに選出されたと判断するための条件は次の2つである。

- 観測対象のいずれかのリーダーに自分が選出されている。センサ  $i$  が観測可能な観測対象の集合を  $J$  とすると、式 (13) のように表すことが出来る。

$$l_1 = \begin{cases} true & \exists j \in J, x_{t_j}^{s_i} = i \\ false & otherwise \end{cases} \quad (13)$$

- リーダー選出における緩和しない制約がすべて満たされている。緩和しない制約の集合を  $C$  とすると、式 (14) のように表すことが出来る。

$$l_2 = \begin{cases} true & \forall c \in C, f_c = 0 \\ false & otherwise \end{cases} \quad (14)$$

$l_1 \wedge l_2 = true$  であれば、自分がリーダーであると判断し、自分がリーダーであることを他のセンサに通知する。大域的に問題が解決されていない状態では、リーダーである条件を満たせなくなる可能性もある。その場合は、リーダーでなくなった事を他のセンサに通知する。本論文では全体に通知することは通信コストが高いと考え、制約が関係するセンサにのみ通知する。ただし、センサと観測対象の配置により間接的に制約が関係する場合がある。その場合は通知メッセージを中継させる。グリッドモデルでは、この中継が必要な回数は高々1回である。シミュレータでの動作を図7に示す。図7中の `my_status` は自分がリーダーであるかどうかを示し、`leader_list` はリーダーであるセンサのリストである。また、`message` にはリーダーに関する情報が含まれ、`variables` は2つの階層の問題の変数であり、それぞれの階層において制約で関係しているセンサと情報を交換する。メッセージ及び変数値は図7の31行に示したように一括して送信する。

## 6. 評価

STAV を用いた従来手法と提案手法を比較し、問題を比較的簡単に解くことが出来る問題に分割することの有用性を評価した。従来手法では解法としてDSTSを用い、提案手法では図7で示したように2つの階層についてDSTSを用いた。しかし、最適解に対する終了条件の無いDSTSでは解が1つに決定しない。したがって、解を決定する方法が必要であるが、本論文では初期の検討として、特にDSTSには変更を加えず評価を行った。ここでは、各センサが同期的にメッセージ交換を行い、シミュレータが全体の評価値を集計し解を到達したことを検知するまでのメッセージ交換サイクル数と解に到達する頻度を評価した。各センサが処理をしメッセージを送信するまでを1サイクルとする。1000サイクルを1試行とし、1000サイクルで解に到達できない場合は、その試行におけるサ

```

1 initialize variables;
2 my_status ← false;
3 empty leader_list;
4 while
5   previous_status ← my_status;
6   receive others messages;
7   check messages;
8   if new leader is elected then
9     add new leader to leader_list;
10    make message about information of new leader;
11  end if
12  if elected leader came off then
13    remove old leader from leader_list;
14    make message about information of old leader;
15  end if
16  DSTS for solving leader election;
17  if Previous status ≠ leader then
18    if  $l_1 \wedge l_2 = true$  then
19      my_status ← leader;
20      make message about my_status;
21    end if
22  if previous_status = leader then
23    if  $l_1 \wedge l_2 = false$  then
24      my_status ← notleader;
25      make message about my_status;
26    end if
27  end if
28  if my_status = leader then
29    DSTS for allocation problem
30  end if
31  send messages and values of variables to agents
   connected by the constraint.
32 end while

```

図7 シミュレータでの動作

イクル数の評価を1000サイクルとした。また、評価は1000回試行の平均を用いた。

### 6.1 最適解の違反度が0である配置での実験

実験で用いた例題は、最適解の違反度が0であり、制約網は単一の連結成分からなる、という制限のもとで作成した。センサと観測対象はグリッド状に配置されるが、問題の難易度を比較的均等にするを意図して、制約が単一の連結成分からなるものとした。そのためグリッドの大きさは重要な指標ではなく、観測に参加可能なセンサ数が重要な指標となる。他の指標が同じであれば、観測に参加可能なセンサ数が少ないほど制約網は密になる。この設定に基づいて、観測対象数5の問題と観測対象数10の問題を各3つずつ作成した。比較的スケールが大きい問題でも、このような問題のインスタンスは鏡像等を除外すると少ない。そこで本論文では初期の検討として3つのインスタンスを用いて評価を行った。各問題の観測に参加可能なセンサ数  $n_s$ 、変数の数  $n_v$ 、全体の制約の数  $n_c$  を表1に示す。実験におけるDSTSの各パラメータの値を表2に示す。この値は観測対象数5の `problem1` で解に到達するまでのサイクル数が最も少なかったパラメータを採用したものである。また、各制約の重みの値を表3に示す。この値は緩和不可能な制約の重みを緩和可能な制約の重みが越えないように設定した。緩和不可

表1 問題のパラメータ

最適解の違反度が0である配置

観測対象数 5	problem1	problem2	problem3
従来手法	$n_s = 16, n_v = 20$ $n_{c^0} = 20, n_{c^1} = 4, n_{c^2} = 60$	$n_s = 16, n_v = 20$ $n_{c^0} = 20, n_{c^1} = 4, n_{c^2} = 60$	$n_s = 16, n_v = 20$ $n_{c^0} = 20, n_{c^1} = 5, n_{c^2} = 60$
提案手法 (リーダー選出)	$n_s = 16, n_v = 20$ $n_{c^0}, n_{c^3} = 20, n_{c^1}, n_{c^4} = 4, n_{c^2} = 60$	$n_s = 16, n_v = 20$ $n_{c^0}, n_{c^3} = 20, n_{c^1}, n_{c^4} = 4, n_{c^2} = 60$	$n_s = 16, n_v = 20$ $n_{c^0}, n_{c^3} = 20, n_{c^1}, n_{c^4} = 5, n_{c^2} = 60$
提案手法 (割当て問題)	$n_v = 5$ $n_{c^0} = 5, n_{c^1} = 8$	$n_v = 5$ $n_{c^0} = 5, n_{c^1} = 8$	$n_v = 5$ $n_{c^0} = 5, n_{c^1} = 8$
観測対象数 10	problem1	problem2	problem3
従来手法	$n_s = 30, n_v = 40$ $n_{c^0} = 40, n_{c^1} = 12, n_{c^2} = 120$	$n_s = 31, n_v = 40$ $n_{c^0} = 40, n_{c^1} = 9, n_{c^2} = 120$	$n_s = 30, n_v = 40$ $n_{c^0} = 40, n_{c^1} = 10, n_{c^2} = 120$
提案手法 (リーダー選出)	$n_s = 30, n_v = 40$ $n_{c^0}, n_{c^3} = 40, n_{c^1}, n_{c^4} = 12, n_{c^2} = 120$	$n_s = 31, n_v = 40$ $n_{c^0}, n_{c^3} = 40, n_{c^1}, n_{c^4} = 9, n_{c^2} = 120$	$n_s = 30, n_v = 40$ $n_{c^0}, n_{c^3} = 40, n_{c^1}, n_{c^4} = 10, n_{c^2} = 120$
提案手法 (割当て問題)	$n_v = 10$ $n_{c^0} = 10, n_{c^1} = 20$	$n_v = 10$ $n_{c^0} = 10, n_{c^1} = 19$	$n_v = 10$ $n_{c^0} = 10, n_{c^1} = 18$

最適解の違反度が0でない (リーダー選出における最適解の違反度は0)

観測対象数 5	problem1	problem2	problem3
従来手法	$n_s = 12, n_v = 20$ $n_{c^0} = 20, n_{c^1} = 8, n_{c^2} = 60$	$n_s = 14, n_v = 20$ $n_{c^0} = 20, n_{c^1} = 6, n_{c^2} = 60$	$n_s = 11, n_v = 20$ $n_{c^0} = 20, n_{c^1} = 13, n_{c^2} = 60$
提案手法 (リーダー選出)	$n_s = 12, n_v = 20$ $n_{c^0}, n_{c^3} = 20, n_{c^1}, n_{c^4} = 8, n_{c^2} = 60$	$n_s = 14, n_v = 20$ $n_{c^0}, n_{c^3} = 20, n_{c^1}, n_{c^4} = 6, n_{c^2} = 60$	$n_s = 11, n_v = 20$ $n_{c^0}, n_{c^3} = 20, n_{c^1}, n_{c^4} = 13, n_{c^2} = 60$
提案手法 (割当て問題)	$n_v = 5$ $n_{c^0} = 5, n_{c^1} = 8$	$n_v = 5$ $n_{c^0} = 5, n_{c^1} = 8$	$n_v = 5$ $n_{c^0} = 5, n_{c^1} = 13$
観測対象数 10	problem1	problem2	problem3
従来手法	$n_s = 22, n_v = 40$ $n_{c^0} = 40, n_{c^1} = 18, n_{c^2} = 120$	$n_s = 24, n_v = 40$ $n_{c^0} = 40, n_{c^1} = 16, n_{c^2} = 120$	$n_s = 26, n_v = 40$ $n_{c^0} = 40, n_{c^1} = 16, n_{c^2} = 120$
提案手法 (リーダー選出)	$n_s = 22, n_v = 40$ $n_{c^0}, n_{c^3} = 40, n_{c^1}, n_{c^4} = 18, n_{c^2} = 120$	$n_s = 24, n_v = 40$ $n_{c^0}, n_{c^3} = 40, n_{c^1}, n_{c^4} = 16, n_{c^2} = 120$	$n_s = 26, n_v = 40$ $n_{c^0}, n_{c^3} = 40, n_{c^1}, n_{c^4} = 16, n_{c^2} = 120$
提案手法 (割当て問題)	$n_v = 10$ $n_{c^0} = 10, n_{c^1} = 18$	$n_v = 10$ $n_{c^0} = 10, n_{c^1} = 18$	$n_v = 10$ $n_{c^0} = 10, n_{c^1} = 18$

表2 解法のパラメータ

	p1	p2	tabu 期間
STAV	0.8	0.4	1
提案手法			
リーダー選出	0.9	0.3	2
割当て問題	0.7	0.2	1

表3 制約の重み

従来手法	リーダー選出	割当て問題
$w_0^{c^0} = 15$	$c^0, w^{c^0} = 1$	$w_0^{c^0} = 15$
$w_1^{c^0} = 5$	$c^1, w^{c^1} = w^{c^2} \times 2$	$w_1^{c^0} = 5$
$w_2^{c^0} = 1$	$c^2, w^{c^2} > w^{c^4} \times 4$	$w_2^{c^0} = 1$
$w^{c^1} = w^{c^2} \times 2$	$c^3, w_0^{c^3} = 1$	$w^{c^1} > w_0^{c^0} \times 8$
$w^{c^2} > w_0^{c^0} \times 4$	$c^4, w^{c^4} = 10$	

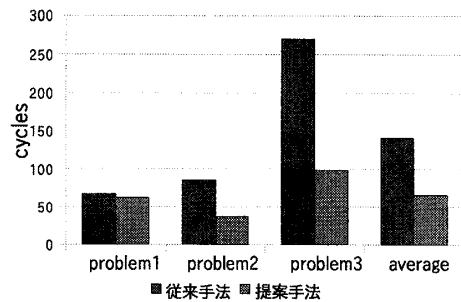
表4 最適解の違反度が0である配置での解に到達する頻度

観測対象数 5				
	problem1	problem2	problem3	average
従来手法	1000	999	970	989.67
提案手法	1000	1000	1000	1000
観測対象数 10				
	problem1	problem2	problem3	average
従来手法	599	931	539	689.67
提案手法	999	1000	997	998.67

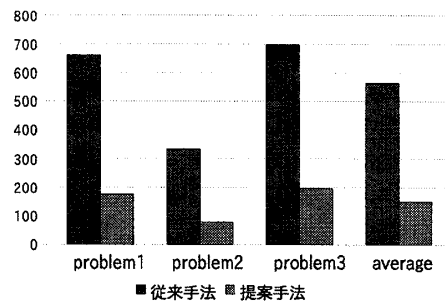
能な制約の重みは  $\infty$  とするべきだが、ここでは緩和不可能な制約の間にも優先順位をつけることにより、タイブレークのバランスをとることを意図し有限の値で表現した。

6.1.1 最適解に到達した頻度・サイクル数

最適解に到達した頻度を表4に示す。数値は1000回中1000サイクル以内で最適解に到達した回数である。また、サイクル数を比較したグラフを図8に示す。この結果は、提案手法によって最適解に到達する頻度が高くなり、最適解に到達するまでのサイクル数を削



(a) 観測対象数 5



(b) 観測対象数 10

図8 最適解の違反度が0である配置でのサイクル数

減できていることを示している。

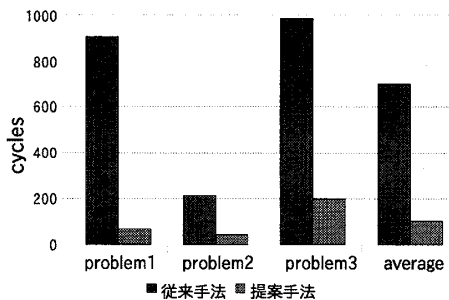
6.2 最適解の違反度が0でない配置での実験

この実験では、割当て問題の最適解の違反度は0でないがリーダー選出における最適解の違反度が0である例題を作成し、提案手法との比較を行った。問題は6.1と同様に制約網が単一の連結成分からなる問題を用いた。各問題のパラメータを表1に示す。最適解の違反度が0でない配置の場合、観測対象を観測可能な

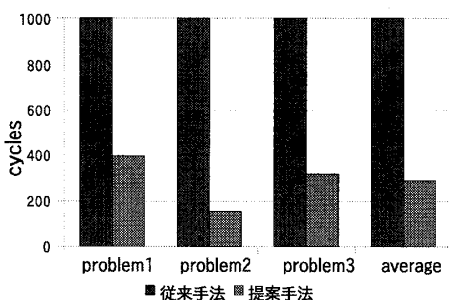


表5 最適解の違反度が0でない配置での解に到達する頻度

観測対象数 5				
	problem1	problem2	problem3	average
従来手法	198	996	29	407.67
提案手法	1000	1000	997	999
観測対象数 10				
	problem1	problem2	problem3	average
従来手法	0	0	0	0
提案手法	868	1000	971	946.33



(a) 観測対象数 5



(b) 観測対象数 10

図9 最適解の違反度が0でない配置でのサイクル数  
センサ数は少なくなり、変数の数は最適解の違反度が0である配置と同じでも、1つのセンサあたりの変数の数が増える。このため、制約網が密な問題となる。DSTSのパラメータは最適解の違反度が0である配置での実験と同じに設定した。最適解の違反度が0ではないので、評価は緩和不可能な制約を全て満たした解に到達する頻度とサイクル数を比較する。ここでは、緩和不可能な制約は全て充足したが、緩和可能な制約の一部は充足していない解を「実行可能解」と表記する。違反度が異なる実行可能解は複数存在するが、ここでは実行可能解に最初に到達したときの結果を評価する。

### 6.2.1 実行可能解に到達した頻度・サイクル数

実行可能解に到達した頻度を表5に示す。数値は1000回中1000サイクル以内で実行可能解に到達した回数である。また、サイクル数を比較したグラフを図9に示す。この結果は、提案手法によって実行可能解に到達する頻度が高くなり、実行可能解に到達するまでのサイクル数が削減できていることを示している。提案手法では実行可能解とは2つの階層について制約を緩和した解である。また、リーダーが選出されるタイミングはそれぞれ異なり、先に選出されたリーダーは貪欲に自身の変数値を決定する。上述の2点から提

案手法では解の質が劣ると考えられる。しかし、それは即応性とのトレードオフとして許容できると考えられる。解を決定する規則の追加および、解の質に関する評価は今後の検討課題である。

## 7. まとめ

本論文では、注視制御可能な自律的なセンサ群による観測システムを想定し、その主要な要素の一つである資源割当て問題に対して、分散制約充足/最適化問題の枠組を適用した。特に、問題を2つの比較的簡単な問題に分割し、各層に対応する問題解決処理系を、部分的に同期しつつ解を得る手法を提案した。これにより、解に到達するまでのサイクル数を削減できていることを実験により示した。この結果から、複雑な問題を比較的簡単な複数の問題に分割することは有用であると考えられる。問題を分割する手法として、地域ごとに分割することも考えられる。そのような形式化との比較は今後の課題である。また、より一般的なセンサ網のモデルへの適用、解を安定させる手法の提案、解の質に関する詳細な評価、実システムにおける検証も今後の課題として挙げられる。

## 参考文献

- 1) Syed Ali, Sven Koenig, and Milind Tambe. Preprocessing techniques for accelerating the dcopt algorithm adopt. In *AAMAS*, pp. 1041–1048, Jul. 2005.
- 2) Ramon Bejar, Carmel Domshilak, Cesar Fernandez, Carla Gomes, Bhaskar Krishnamachari, Bart Selman, and Magda Valls. Sensor networks and distributed csp. *Artif Intell*, Vol. 161, pp. 117–147, 2005.
- 3) Pragnesh Jay Modi, Wei-Min Shen, Milind Tambe, and Makoto Yokoo. Adopt: Asynchronous distributed constraint optimization with quality guarantees. *Artif Intell*, Vol. 161, pp. 149–180, 2005.
- 4) Adrian Petcu and Boi Faltings. A scalable method for multiagent constraint optimization. In *IJCAI*, pp. 266–271, Aug 2005.
- 5) Weixiong Zhang, Ong Wang, and Lars Wittenburg. Distributed stochastic search for constraint satisfaction and optimization: Parallelism, phase transitions and performance. In *PAS*, pp. 53–59, 2002.
- 6) 飯塚泰樹, 鈴木浩之, 竹内郁雄. 分散制約充足問題のための multi-agent tabu search 手法の効果. *信学論*, Vol. J90-D, No. 9, pp. 2302–2313, 2007.
- 7) 浮田宗伯. 能動視覚エージェント群の密な情報交換による多数対象の実時間協調追跡. *信学論*, Vol. J88-D-i, No. 9, pp. 1438–1447, 2005.
- 8) 浮田宗伯, 松山隆司. 能動視覚エージェント群の情報交換による複数対象の実時間協調追跡. *情報処理学会 CVIM 研究会*, Vol. 43, pp. 64–79, 2002.