

# 分散制約最適化手法を適用した協調カメラ網アルゴリズムと実装

加藤 伸哉<sup>†</sup> 松井 俊浩<sup>†</sup> 松尾 啓志<sup>†</sup>

<sup>†</sup> 名古屋工業大学

あらまし マルチエージェントシステムの応用分野の一つとして分散センサ網に関する研究が行われている。本研究では、分散センサ網の一応用として、注視制御ができるセンサ群を用いて観測対象物の追跡を行う観測システムを対象とし、その重要な要素である観測資源割り当て問題に対して、分散制約最適化手法を適用する。従来の分散制約最適化問題を用いた形式化では、解の精度と解探索時間の両立に課題があり、また実際のセンサ網への適用例は少ない。本研究では、比較的小規模な部分問題への適用を前提として、実際のセンサ網における割り当て問題に含まれる、センサ間の合意を簡略した形式化を提案する。また、一度決定した割り当てを次の割り当て問題の解として可能な限り保持するため、直前の問題の解と、現在の問題の解を同一に保つことを目的とする動的な問題のための制約を導入した。提案手法をシミュレーションにより評価するとともに、観測システムを実装し検証実験をおこなった。

キーワード 協調センサ網, 分散制約最適化問題, 資源割り当て問題

## Algorithms and Implementation of Cooperative Camera Network with DCOP

Shinya KATO<sup>†</sup>, Toshihiro MATSUI<sup>†</sup>, and Hiroshi MATSUO<sup>†</sup>

<sup>†</sup> Nagoya Institute of Technology

**Abstract** 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 problem of the distributed sensor network can be formalized as distributed constraint optimization problem(DCOP). However, balances of quality of solution and limitation of execution time of the solver should be evaluated in actual cooperative sensor networks. In this work, we formalize the resource allocation problem as DCOP that employ simple representation of agreement between agents. The proposed model is implemented as an actual cooperative sensor network. And the performance of the system is evaluated by experiments.

**Key words** sensor network, DCOP, resource allocation problem

### 1. はじめに

近年、マルチエージェントシステムの応用分野として、分散センサ網が注目されている [1] [2] [3]。分散センサ網は、自律的なセンサ群が相互に接続されたシステムであり、各センサが協調して情報収集やセンサ機器などを制御する。分散センサ網の応用例として、ロボットサッカー [2] や協調的な監視カメラ網 [3] などが提案されている。その一方で、マルチエージェントシステムの協調問題解決における基本的な枠組みとして、分散制約最適化問題が研究されている [4] [5] [6] [7]。分散制約最適化問題では、エージェントの状態が変数として表現され、エージェント間の協調が変数間の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、評価関数を大域的に最適

化する変数値を決定する。マルチエージェントシステムの目的を分散制約最適化問題として形式化することにより、そのシステムで解決すべき問題を明示的に記述することができる。特に分散センサ網においては、観測資源を観測対象物にどのように割り当てるかを分散制約最適化問題へ形式化する手法の研究が行われている [8] [9]。実際的な観測システムでは、センサの注視方向の変化や観測対象物の移動など、環境は時刻とともに変化する。環境の変化に観測システム全体が追従するために、各センサに対する観測資源の割り当ても環境の変化にともなって変更する必要がある。時刻とともに変化する環境における観測資源割り当て問題を分散制約最適化問題へ形式化する場合、ある時刻での大域的なスナップショットに対する割り当て問題を反復して解く問題として考えることができる。各時刻におけ

る問題解決に利用可能な時間は限られているので、短時間で解を発見することができる確率的解法を用いることが有効である。しかし、各エージェントの協調と資源割り当てをおこなう問題は複雑であり、解の精度の改善と探索時間の削減の両立には課題がある。また、実際の観測システムにおいては、時系列的な割り当てについて、あるエージェントの動作が頻繁に切り替わることは好ましくない。一度決定した割り当てを次の割り当て問題の解としてできる限り保持するため、直前の問題の解と、現在の問題の解を同一に保つことを目的とする「弱い」制約を導入する必要がある。

本研究は、協調カメラ網に含まれる比較的小規模な部分問題への適用を目的とする。短時間で解を得るために、従来手法より簡単な分散制約最適化問題への形式化を用いる。そして、提案手法を実際の分散協調カメラ網へ適用し、解の精度、実行時間などの評価をおこなった。

## 2. 協調センサ網における観測資源割り当て問題

### 2.1 協調センサ網

協調センサ網とは、自律的な複数のセンサが協調して観測対象物の情報収集をおこなう観測システムである。本研究では、協調センサ網のうち、次の特徴を持つものを対象とする。

- 観測対象物は計算資源を持たない
- 各センサが観測できる範囲に制限がある
- 各センサが同時に観測できる観測対象物の数に制限がある

これらの特徴を持つ協調センサ網には監視カメラ網や協調的な天文観測などがあげられる。その一方で、ロボットサッカーや協調的なカーナビゲーションシステムなど、観測対象物が計算・通信資源を持ち、協調的な処理の構成要素に含まれるものは、本研究の対象としない。以降では、協調センサ網の資源割り当て問題解決について、エージェントを用いた協調モデルと、分散制約最適化問題への形式化による協調モデルについて述べる。

### 2.2 エージェントを用いた協調モデル

本研究で対象とする協調センサ網について、グループ化されたエージェントであるエージェントを用いた協調モデルが提案されている [3]。エージェントを用いた協調モデルでは、エージェント内のエージェントマネージャと呼ばれるエージェントがメンバとして参加するエージェントを制御する。エージェントを用いた協調モデルでは、観測システムは自律的に動作する能動カメラエージェント (Active Vision Agent, AVA) の集合として構成される。各 AVA は、観測対象物を検出すると観測対象物ごとにエージェントを生成する。各エージェントについて一つのエージェントマネージャが含まれる。各エージェントマネージャは互いに情報を交換して観測資源の割り当て、すなわち自身のエージェントのメンバの数を調整し、他の AVA はエージェントマネージャの決定に従う。エージェントを用いた協調手法は実機を用いた小規模な環境で実証されているが、その協調プロトコルを最適化問題として考えた場合の解の精度については明確ではない点があると考えられる。

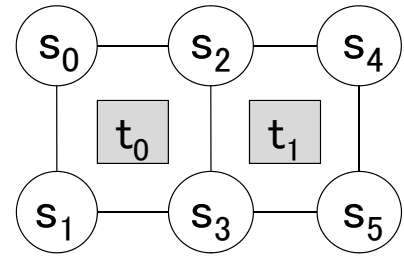


図1 グリッドモデルによる表現

### 2.3 分散制約最適化問題

分散制約最適化問題は、マルチエージェントシステムにおける協調問題解決のための基礎的な枠組みとして研究されている。分散制約最適化問題では、各エージェントは自身が管理する変数と制約で関係する変数を管理する他のエージェントと通信し、メッセージを交換することで解の探索をおこなう。メッセージ交換をとまなう分散処理により解を求めるためのアルゴリズムが提案されている [4] [5] [6] [7]。

既存研究では、協調センサ網における資源割り当て問題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ている。このモデルでは、観測対象物の検出や同定処理などを簡略化している。グリッドモデルで用いるセンサ網を、センサを  $s_i$ 、観測対象物を  $t_j$  として表した図を図1に示す。図1で、センサの可視範囲はセンサが隣接するブロックまでであり、同一グリッド内に存在する観測対象物の数は  $m_{max}$  以下である。関連研究では、 $m_{max}$  を1としている。既存の研究では、各センサ  $s_i$  が知る各観測対象物  $t_j$  に対する割り当てを変数  $x_{t_j}^{s_i}$  として形式化する手法 [9] や、観測資源の割り当て問題とリーダー選出問題へ問題の階層を分割し、解を発見するまでの時間を探索する時間を削減する方法 [8] が提案されている。しかし、本研究で対象とするような協調センサ網では、各エージェントの協調と資源割り当てをおこなう問題は複雑であり、解の精度の改善と探索時間を削減することの両立には課題がある。

## 3. 提案手法

本研究では、協調センサ網における観測資源割り当て問題のうち、比較的小規模な部分問題を対象とし、各エージェント間の観測対象物に対する観測割り当ての合意決定を簡略化するような、分散制約最適化問題への形式化を提案する。また、時系列的に変化する環境において、各センサが観測対象物への観測資源の割り当てを変更することを抑制するために、観測資源の割り当てを変更することを制限するための制約を追加する。

### 3.1 観測対象物の注視に関する制約

提案手法では、各観測対象物に対して割り当てを決定する問題を、センサ  $s_i$  に対して、センサが注視する観測対象物の集合を変数  $x_{s_i}$  として形式化する。すなわち、センサ  $s_i$  はその変数の要素  $t_k \in x_{s_i}$  を注視する。各センサが一度に注視できる観測対象物の数には制限があるため、本研究では、各センサが一度に注視できる観測対象物の最大数を一つとする。各変数  $x_{s_i}$  の値域  $d_{s_i}$  はセンサ  $s_i$  が注視できる観測対象物の集合を  $\{t_0, \dots, t_m\}$  とすると  $\{\emptyset, \{t_0\}, \dots, \{t_m\}\}$  であり、値域の総数

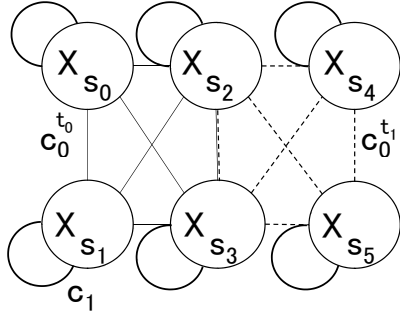


図2 提案手法による図1の形式化

は  $m+1$  となる．観測資源の要求と制限に関する制約を、「追跡制約」として次のように形式化する．

追跡制約  $c_0^{t_j}$

観測対象物  $t_j$  に十分な数のセンサが割り当てられるという制約である．本研究では，十分な数のセンサを  $n_e$  とし， $t_j$  に割り当てられたセンサ数が  $n_e$  に満たない場合に違反となる．この制約は多項制約であり，制約  $c_0^{t_j}$  に対する評価関数  $f_{c_0^{t_j}}$  は以下の式として与えられる． $n_{t_j}$  は  $t_j$  に割り当てられたセンサ数を表す．また， $X_{t_j}$  は観測対象物  $t_j$  を注視できるセンサ集合を表す． $w_n^{c_0}$  ( $0 \leq n \leq n_e - 1$ ) は制約の違反度を示す定数値である．

$$f_{c_0^{t_j}}(X_{t_j}) = \begin{cases} w_0^{c_0} & n_{t_j} = 0 \\ \dots & \\ w_{n_e-1}^{c_0} & n_{t_j} = n_e - 1 \\ 0 & otherwise \end{cases} \quad (1)$$

制約  $c_0^{t_j}$  は 1 以上のセンサ数で追跡するという制約と，2 以上のセンサ数で追跡するという制約に分解することができる．以降では，特に必要がある場合には制約  $c_0^{t_j}$  の条件のうち， $n$  以上のセンサ数で追跡するという条件を  $c_{0n}^{t_j}$  と記述する．

この形式化では視野を共有するセンサを表す変数全てに制約辺が存在する．図1を本提案手法を用いて形式化した場合の制約網を図2に示す．この場合，たとえば，変数  $x_{s_0}$  と変数  $x_{s_4}$  との間には制約辺が存在しない．また，変数  $x_{s_2}$  と変数  $x_{s_3}$  は 2 つの観測対象物に対する制約でつながっている．各観測対象物  $t_j$  に対する制約  $c_0^{t_j}$  は，完全グラフとなるような部分的な制約網を構成し，全体の制約網はそれらを連結した構造となる．各部分の完全グラフの次数は，その観測対象物  $t_j$  を注視できるセンサ数である．

### 3.2 センサの割り当て保持に関する制約

連続したスナップショット間において，可能な限り同一の観測対象物を割り当てるために，直前のスナップショットにおけるカメラ  $s_i$  の割り当て  $a_{s_i}$  と変数  $x_{s_i}$  の間に制約を追加する．あるセンサ  $s_i$  の直前のスナップショットにおける割り当てと，現在のスナップショットにおける割り当てが同一であることが好ましい場合というのは，以下の条件を両方とも満たす場合である．

- 直前のスナップショットでカメラ  $s_i$  へ割り当てられた観測対象物  $a_{s_i}$  は現在のスナップショットにおいてもカメラ  $s_i$  が注視できる観測対象物である
- 直前のスナップショットでカメラ  $s_i$  に観測対象物が割り

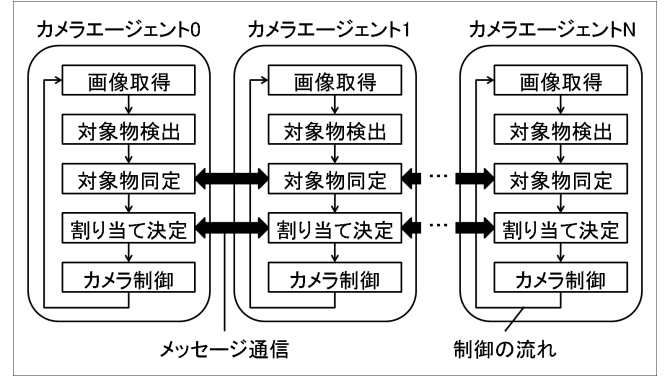


図3 実装した観測システムの処理フロー

当てられていた

そこで，安定的な資源割り当てを保持するための制約を、「保持制約」として次のように形式化する．

保持制約  $c_1$

一度，あるスナップショットで決定した割り当てである  $a_{s_i}$  を次以降のスナップショットでも保持するための制約である．センサ  $s_i$  に割り当てられた観測対象物  $t_j$  を変更する場合に違反となる．この制約は単項制約であり，制約  $c_1$  に対する評価関数  $f_{c_1^{s_i}}$  は以下の式として与えられる． $w^{c_1}$  は制約の違反度を示す定数値である．

$$f_{c_1}(x_{s_i}) = \begin{cases} w^{c_1} & (x_{s_i} \neq a_{s_i}) \wedge (\{a_{s_i}\} \in (d_{s_i} \setminus \{\emptyset\})) \\ 0 & otherwise \end{cases} \quad (2)$$

図2に，全てのセンサがいずれかの観測対象物を注視している場合の保持制約を示す．

### 3.3 評価関数の間の優先関係

追跡制約と保持制約について，満足できない場合の重大さは，対象に関する情報そのものが得られない追跡制約違反の方が保持制約よりも大きい． $A \succ B$  を B よりも A の制約を優先する記号として，各制約の優先度は  $c_{00}^{t_1} \succ \dots \succ c_{0(n_e-1)}^{t_1} \succ c_1$  とする．制約間の優先度を満足させるためには，優先度の高い制約の一つ違反した場合の違反度よりも優先度の低い制約全てに違反した場合の違反度が下回る必要がある．観測システム中のカメラ数を  $N$ ，観測システムが注視できる最大の観測対象物の数を  $M$  として，違反度の値は各  $0 \leq n \leq n_e - 1$  に対して  $w_n^{c_0} \geq w_{n-1}^{c_0} * (M+1)$  かつ  $w_{n_e-1}^{c_0} \geq w^{c_1} * (N+1)$  の関係を満たすように決定した．

## 4. 提案手法を適用した観測システムの実装

観測資源の割り当て解決に本提案手法を用いた，カメラによる観測システムの実装について述べる．

### 4.1 全体の処理

本研究で実装する観測システムは，複数のカメラエージェントによって構成される．各カメラエージェントはそれぞれ1台のパン・チルト制御が可能なカメラを制御し，図3に示すように，対象物検出，位置推定，対象物同定，割り当て決定，カメラ制御を反復する．観測対象物の検出には，色検出を用い，同一の色を持つ，最大の連結領域を観測対象物の領域とする．そ

の領域に対して、画像中の位置とカメラのパン・チルト値から、観測対象物の床での位置を計算する。続いて行う観測対象物の同定では、これまでに得られた観測対象物の色の情報を用い、各カメラエージェントの間で矛盾がないように、観測対象物のIDを決定する。また、各カメラエージェントにおける直前のスナップショットで割り当てられた観測対象物と現在注視できる観測対象物間の対応も決定する。各カメラエージェント同士の情報交換はメッセージ交換網として実装した。以降では、センサ割り当ての決定に関する処理の実装について記述する。観測対象物の同定処理までが完了しているため、各観測対象物について、その位置はただ一つに決定しており、またそれぞれ矛盾なくID付けがされている。

#### 4.2 制約網の生成

対象の割り当てをおこなう前に、対象同定までの処理が実行され、各カメラについて、そのパン・チルト制御を考慮して注視できる観測対象物が決定している。それらの情報を元に、各カメラエージェントは、自身の割り当てを示す変数を作成する。

各カメラエージェントは、注視できる観測対象物リストのうち、自身が注視できる観測対象物をリストアップし、それらの観測対象物IDと、なにも割り当てないことを示す空集合を示す定数を値域としてリストに登録する。

各カメラエージェントは、自身が注視できる対象  $t_i$  について評価関数  $f_{c_0 t_j}$  を計算できるようにするための情報を他カメラエージェントから得る。また、同様に評価関数  $f_{c_1}$  を計算できるようにするための情報を、直前のスナップショットにおける割り当てから得る。それぞれの評価関数の計算に必要な情報を以下に示す。

- 追跡制約の評価関数  $f_{c_0 t_j}$  の計算

自センサが注視できる観測対象物に対して、その観測対象物を注視できる他センサの変数値が必要である。追跡制約の評価関数を各自計算するためには、自センサが注視できる観測対象物  $t_j$  を同様に注視できるセンサについてのみ、メッセージの交換を行えばよい。提案システムでは、各センサの視野については既知で、その視野の一部を共有しており、同一の観測対象物を注視できるセンサは各自のセンサで計算できるものとする。

- 保持制約の評価関数  $f_{c_1}$  の計算

現在のスナップショットにおいて注視できる観測対象物と、前問題で割り当てられた観測対象物との対応付けが必要である。これは、対象物の同定処理において、すでに完了している。

#### 4.3 解の探索

本提案システムでは、解の探索アルゴリズムとして DSTS [7] を用いた。DSTS は、以前に遷移した値へ遷移しない期間であるタブー期間をゼロとすることで、DSA [6] と同様の動作となる。実装した解の探索アルゴリズムを図4に示す。ただし、 $P(x)$  を確率  $x$  で *true*、確率  $1-x$  で *false* となる関数とする。また、 $F^{global}$  は大域的な評価関数とする。初期値としては、前問題の割り当てを示す値と同一の値を使用する。

また、DSTS は確率的に解を変更していくため、最終的なサイクルでの各変数値の組み合わせが、それまでに探索していた組み合わせのうちの最良のものである保証がない。実際には、

```

 $x_{s_i} \leftarrow a_{s_i}$ 
while  $k \leq DSA\_LIMIT$  do
  メッセージを交換し、message_list を生成
   $V_{s_i} \leftarrow F^{s_i}(message\_list, x_{s_i})$ 
  if  $s_i = leader$  then
    if  $global\_min > F^{global}(message\_list)$  then
       $global\_min\_set \leftarrow message\_list$ 
    end if
  end if
   $min \leftarrow INT\_MAX$ 
  for all  $j \in A_{s_i} \wedge j \notin TABU\_LIST$  do
    if  $min > F^{s_i}(message\_list, j)$  then
       $min\_alloc \leftarrow j$ 
       $Min_{s_i} \leftarrow F^{s_i}(message\_list, j)$ 
    end if
  end for
  TABU_LIST の更新
   $\Delta = V_{s_i} - Min_{s_i}$ 
  if  $\Delta \geq 0 \wedge P(p_1)$  then
     $x_{s_i} \leftarrow min\_alloc$ 
  else if  $\Delta < 0 \wedge now\_val > 0 \wedge P(p_2)$  then
     $x_{s_i} \leftarrow min\_alloc$ 
  end if
   $k \leftarrow k + 1$ 
end while

```

図4 DSTS を応用した解の探索アルゴリズム

これまでに探索してきた組み合わせのうち、最良のものを用いて観測システムを制御すべきである。この点については、あらかじめ決定しておいた最良解検出エージェントが探索中の最良解を保持し、一定回数の探索後に放送するものとする。このために、最良解検出エージェントは各エージェントの解の情報を収集する。

## 5. 評価

提案手法を実装した観測システムを用い、実験による評価をおこなった。使用したカメラの数  $N$  は6、観測対象物の数  $M$  を2とし、観測対象物を注視するのに十分なカメラの数  $n_e$  を2とした。また、各カメラを図5のように部屋の端から床を見下ろすように配置した。

#### 5.1 シミュレーションによる評価

シミュレーションによる評価を行い、最適解を求めるために必要なサイクル数、およびそれを最小にする解の探索パラメータを決定する。まず、解の探索に用いるアルゴリズムである DSTS のパラメータを予備実験により決定した。図6に示す制約網に対して、確率  $p_1, p_2$  を変化させて最適解への収束サイクルを計測したグラフを図7に示す。図7(a)は確率  $p_2$  を固定して確率  $p_1$  を変化させた場合のグラフであり、図7(b)は確率  $p_1$  を固定して確率  $p_2$  を変化させた場合のグラフである。図7の結果から、解の探索手法における確率  $p_1$ 、確率  $p_2$  はそれぞれ  $p_1=0.6$ 、 $p_2=0.2$  とした。

また、提案手法における形式化では、最適解が解の全体に定める割合と、問題の難しさにある程度の相関があるため、問題

表 1 使用した機材

CPU	Core 2 Duo 2.66GHz
メモリ	2GB
カメラ	SONY EVI G-20
カメラサーバ	中央電子社製 VW-14

表 2 同一の観測対象物を連続して注視したスナップショットの期間

カメラ ID	パン・チルト制御	最長	最短	平均
0	可能	100	100	100
1	可能	97	97	97
2	不可	17	2	9.5
3	不可	13	7	9.7
4	可能	49	23	36
5	可能	96	3	49.5

の難しさを最適解の割合で見積もることとした。最適解の割合が異なる複数の問題で計測した、DSA と DSTS の最適解に到達したサイクル数を図 8 に示す。図 8 から、DSTS よりも DSA を用いた方が高速に最適解を発見できているため、解の探索手法として DSA を用いた。また、最も難しいと推定される問題において最適解の探索に必要なサイクル数から、最適解検出エージェントが最良解を放送するタイミングを、解の探索を開始してから 230 サイクル経過した後とした。

## 5.2 実観測システムによる評価

実装に用いた機材を表 1 に示す。カメラサーバは、各カメラと直接接続し、カメラからの画像のバッファリングや、カメラ制御などを行う。カメラサーバとカメラエージェントが動作するマシンは、LAN を用いて相互に接続した。また、実装の都合上、カメラ 2、3 はパン・チルト制御を行わないこととした。

### 5.2.1 単独のスナップショットに対するシステムの挙動

実装した観測システムを動作させた際の、各カメラの入力画像を図 9 に示す。各センサの検出結果を総合し、各観測対象物は表 4 に示す位置と推定された。

実装した観測システムでは、観測対象物の画像中位置や大きさは考慮していないので、カメラ 3 のように端に映るだけでも注視可能とされた。割り当て探索の結果、カメラ 0、カメラ 2 が対象  $t_0$  を、カメラ 1、カメラ 3 が対象  $t_1$  を注視する割り当てを決定した。解を用いてカメラを制御した結果が図 10 である。パン・チルトできるカメラはそれぞれ、割り当てられた観測対象物を画像中心へ捉える位置に視線方向を制御した。

### 5.2.2 連続したスナップショットに対するシステムの挙動

観測対象物を部屋中央付近でランダムに移動させ、提案手法を実装した観測システムを 100 回のスナップショットに対する割り当てを決定するまで動作させた。このときの使用カメラ数を図 11 に示す。また、同一の観測対象物を注視し続けたスナップショットの期間を表 2 に示す。パン・チルトにより、多くの範囲を観測できるカメラは長く、そうでないカメラは短い期間となった。また、カメラ 0、1 では、注視する観測対象物の変更されず、安定的に観測対象物を注視しつづけた。

### 5.2.3 実行時間

提案手法を実装した観測システムを、異なる 1000 スナップ

表 3 各処理の実行時間

処理	最悪実行時間 (ms)	平均実行時間 (ms)
画像取得	230	53
対象物検出	36	17
対象物同定	132	50
割り当て決定	238	195
カメラ制御	803	797

ショットに対して実行した場合の、図 3 に示した 1 ループあたりの平均実行時間および最悪の実行時間を表 3 に示す。実装の都合上、各カメラエージェントとカメラサーバを LAN で接続したことにより、画像取得およびカメラ制御の処理で長い実行時間が必要となった。割り当ての決定には 1 サイクルあたり 1ms、総合して 200 ms 程度しか必要としないため、十分高速に解を発見できたといえる。

## 6. まとめ

本研究では、協調センサ網を分散制約最適化問題へ形式化する一手法を提案した。特に現実の観測システムの実装による検証を重視し、比較的小規模な部分問題への適用を前提として、センサ間の合意を得るための形式化や合意を得る問題を簡略した形式化を適用した。これにより問題の解空間の規模を削減した。また、複数のスナップショット間で観測資源の割り当ての変化を抑制するための制約を追加した。提案手法をシミュレーションにより評価するとともに、観測システムを実装し検証実験をおこなった。これらの結果により提案手法の有効性を示した。今後の課題としては、システム規模の拡大のための検討、およびより自律・分散・協調的な観測資源割り当て処理を指向した問題の形式化と解の探索手法の改良があげられる。

謝辞 本研究の一部は、科学研究費補助金(基盤研究 C, 課題番号 21500073)による。

## 文 献

- [1] 末永, 吉岡, 本位田: “無線センサネットワークにおける複数プログラムの動的配備”, 情報処理学会, **50**, 1 (2009).
- [2] RSJ2009: “外部カメラを用いたヒト型ロボットによるサッカー競技 RoboCup SSL Humanoid の提案と現状” (2009).
- [3] 浮田: “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, **J88-D-i**, 9, pp. 1438-1447 (2005).
- [4] P. J. Modi, W. Shen, M. Tambe and M. Yokoo: “ADOPT: Asynchronous distributed constraint optimization with quality guarantees”, *Artif Intell*, **161**, pp. 149-180 (2005).
- [5] A. Petcu and B. Faltings: “DPOP: A scalable method for multiagent constraint optimization”, *IJCAI 05*, Edinburgh, Scotland, pp. 266-271 (2005).
- [6] W. Zhang, O. Wang and L. Wittenburg: “Distributed stochastic search for constraint satisfaction and optimization: Parallelism phase transitions and performance”, *PAS*, pp. 53-59 (2002).
- [7] 飯塚, 鈴木, 竹内: “分散制約充足問題のための multi-agent tabu search 手法の効果”, 信学論, **J90-D**, 9, pp. 2302-2313 (2007).
- [8] 太田, 松井, 松尾: “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”, *FIT2009*, **2**, pp. 67-74 (2009).
- [9] 貝嶋, 松井, 松尾: “確率的な分散制約最適化手法を用いたセンサ網の資源割り当て手法の提案”, 情報科学技術レターズ (FIT2007), **43**, pp. 173-177 (2007).

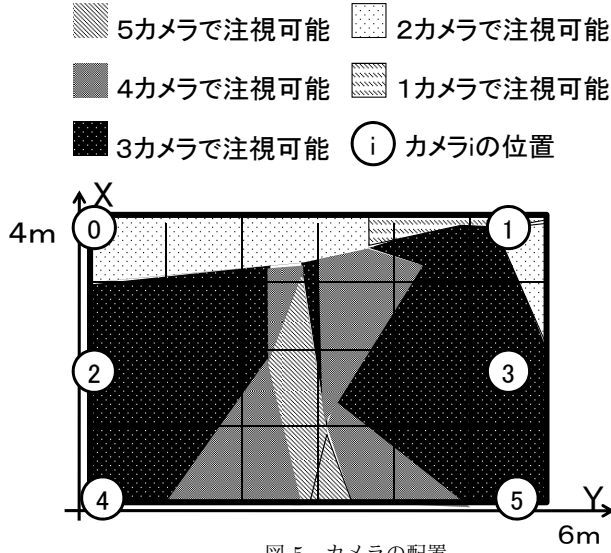


図5 カメラの配置

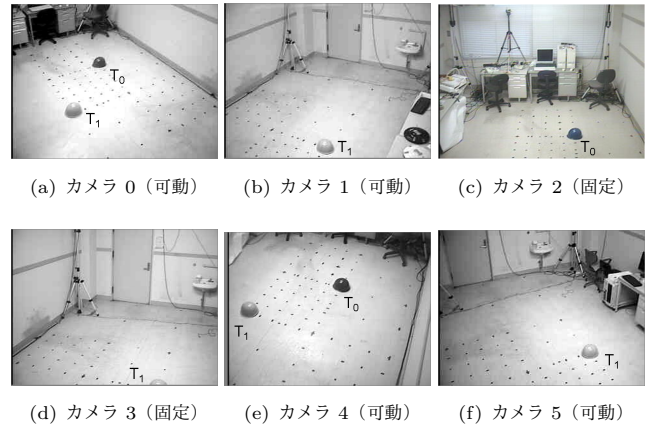


図9 入力画像

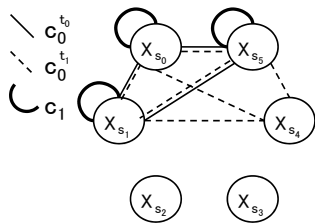


図6 パラメータ決定用の制約網

表4 各観測対象物ごとの位置と注視できるカメラ

観測対象物のID	色	X座標	Y座標	注視できるカメラID
$t_0$	青	1.0m	4.2m	0,2,4
$t_1$	黄	2.1m	3.0m	0,1,3,4,5

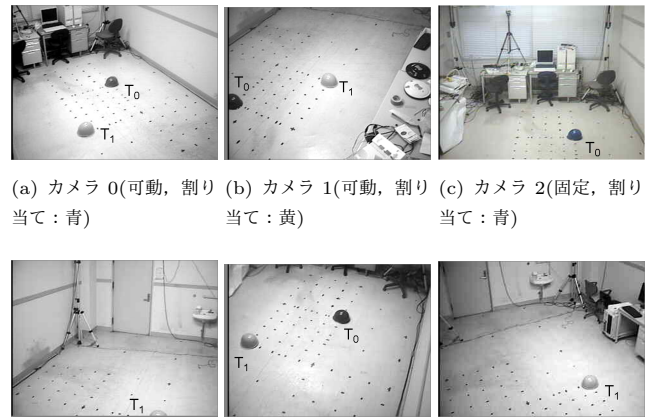
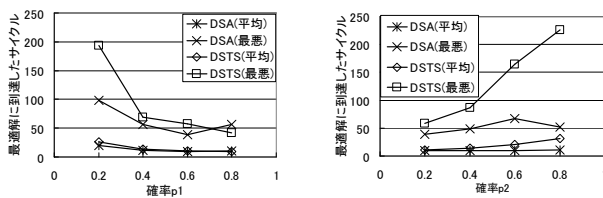


図10 パン・チルト制御後の画像



(a) 確率  $p_1$  による解の探索サイクルの変化 (b) 確率  $p_2$  による解の探索サイクルの変化

図7 パラメータによる解の探索サイクルの変化

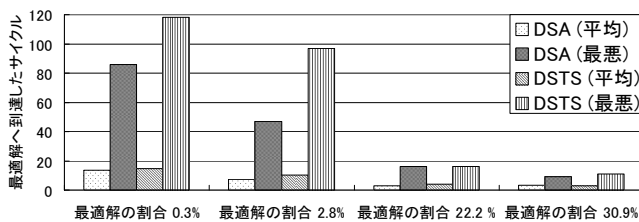


図8 問題ごとの最適解到達サイクル数

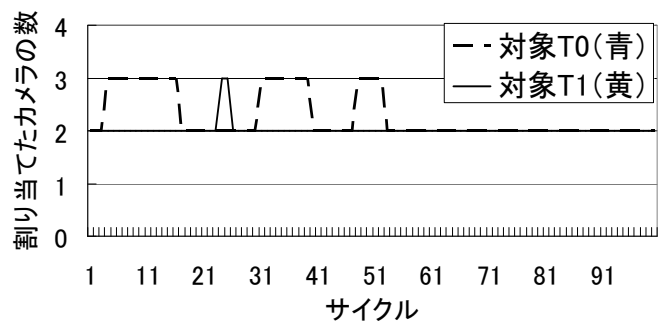


図11 注視に使用したカメラ数の推移