

修士論文

分散制約最適化問題の階層化と
分散センサ網における
観測資源割り当て問題への適用

指導教員 松尾 啓志 教授
津邑 公暁 准教授

名古屋工業大学大学院工学研究科
修士課程創成シミュレーション工学専攻
平成 21 年度入学 21413516 番

太田 和宏

平成 23 年 2 月 3 日

分散制約最適化問題の階層化と 分散センサ網における 観測資源割り当て問題への適用

目次

1	はじめに	1
2	分散制約最適化問題	2
2.1	形式化	2
2.2	分散制約最適化問題を解くための解法	3
2.2.1	厳密解法	3
2.2.2	非厳密解法	4
3	分散センサ網の観測資源割り当て問題	7
3.1	グリッドモデル	7
3.2	分散制約最適化問題に基づく形式化	8
3.2.1	STAV(Sensor-TargetAsVariable)	8
3.2.2	TAV(Target As Variable)	10
3.3	エージェントによる協調モデル	11
4	提案手法: 2つの階層からなる形式化	12
4.1	リーダー選出層	13
4.2	観測資源割り当て問題解決層	16
4.3	2つの階層からなる形式化を用いた場合の解法	17
5	提案手法: 動的な問題への対応	20
5.1	センサの割り当ての状態	22
5.2	変数の生成・削除, 表の作成	22
5.2.1	変数の生成・削除	23
5.2.2	割り当て状態を保持する表の生成	25
5.3	観測対象に割り当てられたセンサの組み合わせの変化の抑制	26
5.3.1	リーダーを同じグループから再選出するための制約	27
5.3.2	観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約	28
5.4	部分的な観測対象に割り当てられたセンサの組み合わせの変化の抑制	29

6	評価	30
6.1	DSTSのパラメータ	32
6.2	DSTSの調整点および変更点	33
6.3	変数の初期値	34
6.4	例題の生成方法	35
6.5	ランダムに例題を作成し、制約違反が発生する可能性が低い変数の初期値を用いた場合の評価結果	35
6.5.1	準最適解を得るまでのサイクル数の比較	36
6.5.2	準最適解の評価値の比較	36
6.5.3	各観測対象に割り当てることが可能なセンサの数	37
6.5.4	観測対象に割り当てられたセンサの数に関する評価	38
6.5.5	観測対象に割り当てられたセンサの組み合わせの変化に関する評価	39
6.6	意図的に複雑な例題を作成し、制約違反が発生する可能性が低い変数の初期値を用いた場合の評価結果	40
6.6.1	準最適解を得るまでのサイクル数の比較	41
6.6.2	準最適解の評価値の比較	42
6.6.3	各観測対象に割り当てることが可能なセンサの数	42
6.6.4	観測対象に割り当てられたセンサの数に関する評価	43
6.6.5	観測対象に割り当てられたセンサの組み合わせの変化に関する評価	44
6.7	ランダムで例題を作成し、制約違反が発生する可能性が高い変数の初期値を用いた場合の評価結果	45
6.7.1	準最適解を得るまでのサイクル数の比較	45
6.7.2	準最適解の評価値の比較	46
6.7.3	各観測対象に割り当てることが可能なセンサの数	46
6.7.4	観測対象に割り当てられたセンサの数に関する評価	47
6.7.5	観測対象に割り当てられたセンサの組み合わせの変化に関する評価	49
6.8	意図的に複雑な例題を作成し、制約違反が発生する可能性が高い変数の初期値を用いた場合の評価結果	49
6.8.1	準最適解を得るまでのサイクル数の比較	50

6.8.2	準最適解の評価値の比較	50
6.8.3	各観測対象に割り当てることが可能なセンサの数	51
6.8.4	観測対象に割り当てられたセンサの数に関する評価	51
6.8.5	観測対象に割り当てられたセンサの組み合わせの変化に関する評価	53
6.9	総括	54
7	まとめ	54
	謝辞	55
	参考文献	55

1 はじめに

近年，分散して配置されたセンサによる協調的なセンシングのような分散システムを構成する複数の自律的な主体が分散・協調的な処理を行う枠組みである，マルチエージェントシステムの研究が行われている．そのようなマルチエージェントシステムにおける協調的問題を解決する基本的な枠組みとして分散制約最適化問題 [1][2][3][4][5][6] が研究されている．分散制約最適化問題では，エージェントの状態/意思決定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/評価関数の評価値を統合した値を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資源スケジューリングなどに含まれる，協調問題解決の本質的な問題を表すものとして重要である．

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の研究も行われている [5][6][7][8][9]．特に，分散センサ網における資源割り当て問題を分散制約最適化問題として形式化する手法の研究がされている [5][6]．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 [7][8][9] などが挙げられる．本論文では注視制御可能な複数の自律的なセンサによる観測システムを想定する．複数の対象を複数のセンサにより注視するような制御のためには，観測資源の割り当て問題の解決が必要である．実際の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．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とらえることが出来る．一方で，各時刻の問題を解決するまでの時間は限られており，解を短時間で得る必要がある．そのため比較的短時間で非厳密解を得ることが可能な確率的な解法を用いることが有効であると考えられる．このような観測資源の割り当て問題においては，準最適な観測資源の割り当てを得るうえで，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リーダーにより観測資源割り当て問題を解決する階層の2つの階層に分割し，解を得るまでの時間を削減する手法を提案する．これにより，問題の複雑さを抑制し解探索時間を削減できると期待される．また，観測対象の配置が変化するような動的な問題では，頻繁なセンサの視野方向の制御による観測情報の劣化を抑制する必要がある．この観点から，問題が変化した際に，観測対象へ割り当てられたセンサの組み合わせの変化を抑制することは重要であると

いえる．そこで，前の時刻の問題について得た解に基づいて，観測対象に割り当てられたセンサの状態に関する情報を利用し，観測対象の配置が変化した後観測対象へ割り当てられるセンサの組み合わせの変化を抑制するための制約も導入する．

本論文の構成を次に示す．第2章では分散制約最適化問題について説明し，第3章では分散センサ網の観測資源割り当て問題について説明する．また従来研究で提案された分散制約最適化問題による分散センサ網の観測資源割り当て問題の形式化とエージェントのグループであるエージェントに基いた協調的な観測システムについて説明する．第4章及び第5章では提案手法について説明する．そして，第6章で評価を行い，提案手法の有効性についての考察を示す．最後に第7章で結言を述べる．

2 分散制約最適化問題

分散制約最適化問題は，複数の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た，離散最適化問題である．分散制約最適化問題では，エージェントの状態/意思決定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/評価関数の評価値を統合した値を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資源スケジューリングなどに含まれる，協調問題解決の本質的な問題を表すものとして重要である．

本章では分散制約最適化問題 (Distributed constraint Optimization Problem, DCOP)[1][2][3][4][5][6]の基本的な形式とその解法について説明する．

2.1 形式化

基本的な分散制約最適化問題の形式化を次に示す．

- マルチエージェントシステムに含まれるエージェントの集合を A で表す．
- A に含まれるエージェント $a_i \in A$ は変数の集合を持つ． a_i が持つ変数の集合を $X_i = \{x_i^1, x_i^2, \dots, x_i^k\}$ で表す．
- エージェント a_i は X_i に含まれる変数 x_i^k の値のみ決定できる．すなわち，変数はエージェントの意思決定を表す．
- 変数間の関係は制約 c によって表す．
- 制約 c に関する評価関数 f_c を定義する．評価関数 f_c は変数値の組み合わせについての評価値を定義する．
- 制約の評価値の合計値が最適になる変数値を各変数に割り当てることが問題の目

的である。

特に本論文での形式化では，制約は次の2種類に分類される。

- 緩和不可能な制約

満たさなければシステムが破綻してしまうような制約である。

- 緩和可能な制約

満たしていなくてもシステムが破綻することはない制約である。

このように制約が分類されるため，本論文での問題の目的は，緩和不可能な制約を充足し，緩和可能な制約をできるだけ満足するような変数値の組合せを決定することである。一方で，緩和不可能な制約は全て充足しているが，緩和可能な制約の一部は満たされていない解を準最適解と呼ぶことにする。

2.2 分散制約最適化問題を解くための解法

分散制約最適化問題解の解法として解の完全性が保証されている厳密解法と完全性が保証されない非厳密解法が提案されている。

2.2.1 厳密解法

分散制約最適化問題を解くための厳密解法として，ADOPT(Distributed Stochastic Search(Asynchronous Distributed Constraint Optimization))[1][10][11] や DPOP[2] が提案されている。

ADOPT[1][10][11]

ADOPT[1][10][11] では，前処理として制約網に対する深さ優先探索木を生成する。また，深さ優先探索木に基く擬似木 (pseudo-tree) を生成する。そして，擬似木により定義される半順序関係にしたがって，メッセージ交換型の解探索により最適解を求める。解探索では ADOPT は分枝限定法/A*による探索を行う。ADOPT の問題点として，探索木の induced-width[2] にしたがって，解探索時間とメッセージ数が指数関数的に増加する。そのため，induced-width が大きい場合は，探索時間の制限のために，ADOPT を分散センサ網における観測資源割り当て問題へ適用することには課題があるといえる。

DPOP[2]

DPOP[2] も ADOPT と同様で，前処理として制約網に対する深さ優先探索木を生成する。また，深さ優先探索木に基く擬似木 (pseudo-tree) を生成する。そして，擬似木により定義される半順序関係にしたがって，メッセージ交換型の解探索により最適解を求める。DPOP では動的計画法に基づく。DPOP の問題点として，前処理で生成さ

れた擬似木の幅にメッセージサイズが依存しており，メッセージサイズやメモリ使用量が指数関数的に増加する．そのため，induced-width が大きい場合は，メモリ量の制限のために，DPOP を分散センサ網における観測資源割り当て問題へ適用することには課題があるといえる．

2.2.2 非厳密解法

一方で，分散制約最適化問題を解くための非厳密解法として，反復改善型のアルゴリズムである DBA(Distributed BreakOut Algorithm)[12]，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(Distributed Stochastic Search)[3] や DSTS(Distributed Stochastic Tabu Search)[4] が提案されている．これらの解法は厳密解法とは違い，解の完全性が保証されないが，比較的短時間で非厳密解を得ることが可能である．本論文で想定するような準時間性を要するシステムでは，解を得るまでの時間の観点から，非厳密解法が有用であると考えられる．特に，本研究では，拡張が容易な確率的解法に着目する．

DBA[12]

DBA[12] は反復改善型アルゴリズムであり，各エージェントが独立に繰り返し変数値の改善を行うことで問題を解く．DBA では，局所最適解へ陥ることを防ぐために，breakout という機構を持つ．準局所最適解を検知した場合，制約の重みを変更することにより，準局所最適解から脱出する．バックトラックなどには頼らない．DBA は次のような特徴を持つ．

- 制約で関係するエージェント間で改善可能な評価値の改善量を交換し，もっとも改善量が大きなエージェントのみ変数値を変更できる．
- 制約で関係するエージェントが改善可能な評価値の改善量が 0 であり，自身に充足されない制約が存在する場合，準局所最適解であると判断する．
- 準局所最適解に陥った場合，充足されていない制約の重みに変更を加える．

DBA は単純なバックトラックを用いた解法などより，高速に解を得ることが出来る．しかし，局所最適解から脱出するために制約の重みを変更する機構により，無限ループに陥る可能性が問題点として指摘されている．

DSA[3]

DSA[3] 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 の詳細を図 1 に示す．図 1 はエージェント a_i が主体の擬似コードである．

DSA では書くエージェントが互いに変数値を交換しつつ解を探索する．エージェント a_i は制約で関係するエージェントの変数値に基き，自身の各変数値に対する評価値

```

1 initialize V;
2 own_status leftarrow false;
3 empty tabu_list;
4 while meet the end requirement do
5     send v_message to agents related by constraints;
6     receive others' v_message;
7     call set_new_value;
8 if new values of variables is assigned then
9     generate v_message(a_i, V);
10    send v_message to agents related by constraints;
11 end if
12 end while
13 procedure set_new_value
14 if  $\Delta \geq 0$  then
15     assign new values to variables with  $p_1$ ;
16 end if

```

図 1: DSA の擬似コード

を計算する．最も評価値が良くなる値を，変数値の候補とする．エージェント a_i は，図 1 中 14-16 行における処理で，評価値の改善量 $\Delta \geq 0$ の場合，確率 p_1 に基づいて先述の変数値の候補に変数値を変更する．変数値が変更された場合，図 1 中 8-11 行における処理で，変数値を通知するためのメッセージ $v_message$ を生成し，制約で関係するエージェントに送信する．

DSA は非常にシンプルで軽量なアルゴリズムであるが，局所最適解に陥りやすく，また局所最適解から脱出しづらいという問題点がある．そのため，解の精度が求められるような場合では適用するには課題がある．

DSTS[4]

DSTS[4] は確率的なアプローチを用いた反復改善型アルゴリズムである DSA に，局所解からの脱出を目的として，タブー探索を組み込んだアルゴリズムである．DSTS の詳細を図 2 に示す．図 2 はエージェント a_i が主体の擬似コードである．

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探索する．各エージェントは制約で関係するエージェントの変数値に基き，自身の各変数値に対する評価値を計算する．そして，最も評価値が良くなる変数値を，変数値の候補とする．エージェントは，図 2 中 23-24 行における処理で，確率 p_1 に基づいて先述の変数値の候補に変数値を変更する．また，変数値を変更する際に，以前の変数値の履歴をタブーリストに追加し，一定期間 (TABU 期間) その値への遷移を禁止する．タブー探索のために，評価値の改善量 Δ が増加するような変数値しか選択出来ない可能性もある．その場合

```

1 initialize V;
2 own_status ← leftarrow false;
3 empty tabu_list;
4 while meet the end requirement do
5     send message(v_message, t_message) agents related by constraints;
6     receive others' messages;
7     call set_new_value;
8     if all values ∈ tabu_list then
9         own_status ← true;
10        generate t_message(a_i)
11        send t_message to agents related by constraints;
12    else if new values of variables is assigned then
13        generate t_message(a_i, V);
14        send v_message to agents related by constraints;
15        add new value to tabu_list;
16    end if
17 end while
18 procedure set_new_value
19     if all values ∈ tabu_list then
20         noting to do;
21     else if all conflicting agents' own_status = true then
22         assign new values to variables;
23     else if  $\Delta \geq 0$  then
24         assign new values to variables with  $p_1$ ;
25     else if current violation > 0 then
26         assign new values to variables with  $p_2$ ;
27     end if

```

図 2: DSTS の擬似コード

にも、図 2 中 25-26 行における処理で、自身の制約が満たされていないならば、確率 p_2 に基づいて変数値を変更する。これらの処理で変数値が更新された場合、図 2 中 12-15 行における処理で、自身の変数に関するメッセージ $v_message$ を生成し、制約で関係するセンサに送信する。

また、変数の値域に含まれる全ての値がタブーリストに含まれる場合、自身をタブー状態とする。その場合は、図 2 中 19-20 行における処理のように、変数値の変更は行わない。また、図 2 中 8-11 行における処理で、メッセージ $t_message$ を生成し、タブー状態であることを制約で関係するエージェントに通知する。制約で関係する全てのエージェントがタブー状態であった場合は、図 2 中 20-21 行に示すように、強制的に変数値を違う値に変更する。図 2 中、 V は変数の集合を示し、 $tabu_list$ はタブーリストを示す。また、 own_status は自身がタブー状態であるかどうかを示し、 $own_status = true$ の場合がタブー状態である。

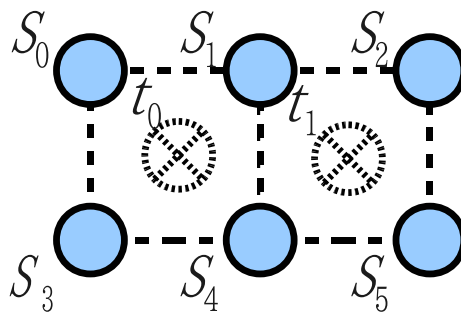


図 3: グリッドモデルでの表現

3 分散センサ網の観測資源割当て問題

分散センサ網の観測資源割り当て問題とは、複数の自律的なセンサと複数の観測対象が存在するような状況を想定した問題である。分散センサ網の観測資源割り当て問題では、各センサをどの観測対象に割り当てるかを決定することが目的である。

本章では、分散センサ網の観測資源割り当て問題を形式化する。また、本論文で想定するような観測システムを実現するための関連研究についても説明する。

3.1 グリッドモデル

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観測資源割り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、観測対象が各区画に配置されるモデルを用いる。このようなモデルは関連研究でも用いられている [5][6]。本論文では、このような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は、制御の対象となるのはセンサのみである。すなわち、センサをエージェントとみなす。図 3 では s_i がセンサを示し、 t_j が観測対象を示す。ここでは各区画には多くても 1 個の観測対象しか存在できないものとした。また、実際のシステムでは様々な観測資源に関する制限やシステムの目的がある。本論文では、次のような制限及び目的を設定する。

- センサの視野に関する制限

実際のシステムではセンサの視野には制限がある。この制限を緩和することはできない。ここでは、センサは隣接する区画のみ観測することが可能とした。

- センサの観測資源に関する制限

本論文では各センサが同時に多数の対象物を注視することは出来ないものとした。この制限を緩和することはできない。ここでは、1 個のセンサを同時に複数

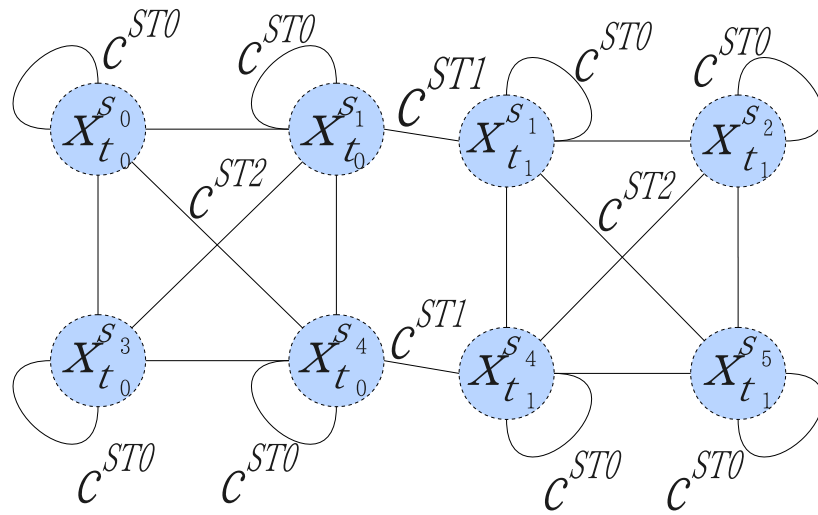


図 4: STAV による制約網の表現

の観測対象に割り当てることはできないとした。

- システムの目的

センサによる観測によって得られる情報量の向上のために、1 個の観測対象を複数のセンサで観測することが望ましい。理由として、例えば、観測対象の座標推定の精度を向上するための十分な観測情報の要求などが挙げられる。ここでは、可能な限り 3 個のセンサを 1 個の観測対象に割り当てることを目的とした。しかし、この目的は緩和可能であり、3 個未満のセンサで観測対象を観測することも可能である。

3.2 分散制約最適化問題に基づく形式化

分散制約最適化問題に基づく分散センサ網の観測資源割り当て問題の形式化について説明する。ここでは従来形式化として、センサと観測対象の組について変数を定義する表現である STAV (Sensor-Target As Variable)、および観測対象について変数を定義する表現である TAV (Target As Variable) について説明する。

3.2.1 STAV (Sensor-Target As Variable)

STAV はセンサと観測対象の組を変数として定義する表現である。この表現に類似する表現は文献 [5] で用いられている。この表現では各センサは観測可能な観測対象の数だけ変数を持つ。図 3 の配置の制約網を STAV を用いて表現すると図 4 のようになる。図 4 において $x_{t_j}^{s_i}$ はセンサ s_i が持つ観測対象 t_j に関する変数を表している。この例で

は, s_0, s_2, s_3, s_5 は観測可能な観測対象が1個なので変数を1個持つ. 一方, s_1, s_4 は観測可能な観測対象が2個なので変数を2個持つ. $x_{t_j}^{s_i}$ の値はどのセンサが観測対象 t_j に割り当てられているかを示す. 観測対象 t_j を観測可能なセンサの集合が $\{s_0, \dots, s_n\}$ ならば, センサ s_i がもつ観測対象 t_j に関する変数 $x_{t_j}^{s_i}$ は観測可能なセンサの組み合わせを表す変数値をとる. 変数 $x_{t_j}^{s_i}$ の値域は $\{\phi, \{s_0\}, \dots, \{s_n\}, \{s_0, s_1\}, \dots, \{s_0, \dots, s_n\}\}$ と表すことができる.

この表現における制約は図4中 $c^{ST0}, c^{ST1}, c^{ST2}$ で示した3種類の制約がある.

- $c^{ST0}(x_{t_j}^{s_i})$: 観測対象への観測資源割り当てに関する制約

この制約は各観測対象に可能な限り3個のセンサを割り当てるための制約である. 本論文で想定するシステムでは, 可能な限り各観測対象に3個のセンサを割り当てることを目的としている. そのため, 観測対象 t_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この制約は違反となる. この制約は緩和可能である. 制約 c^{ST0} に対する評価関数 $f_{c^{ST0}}$ は式(1)で表現される. ただし $w^{c^{ST0}}$ は制約の評価値を示す定数値である. 式(1)では n_j は観測対象 t_j に割り当てられたセンサの数を示す.

$$f_{c^{ST0}(x_{t_j}^{s_i})} = \begin{cases} w_0^{c^{ST0}} & n_j = 0 \\ w_1^{c^{ST0}} & n_j = 1 \\ w_2^{c^{ST0}} & n_j = 2 \\ 0 & otherwise \end{cases} \quad (1)$$

- $c^{ST1}(x_{t_j}^{s_i}, x_{t_{j'}}^{s_{i'}})$: 観測資源の制限に関する制約

この制約は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ための制約である. 本論文で想定するシステムでは, 各センサは同時に複数の観測対象に割り当てることはできないとした. そのため, 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この制約は違反となる. この制約は緩和不可能である.

制約 c^{ST1} に対する評価関数 $f_{c^{ST1}}$ は式(2)で表現される. ただし $w^{c^{ST1}}$ は制約の評価値を示す定数値である.

$$f_{c^{ST1}(x_{t_j}^{s_i}, x_{t_{j'}}^{s_{i'}})} = \begin{cases} w^{c^{ST1}} & x_{t_j}^{s_i} \cap x_{t_{j'}}^{s_{i'}} \neq \phi \\ 0 & otherwise \end{cases} \quad (2)$$

- $c^{ST2}(x_{t_j}^{s_i}, x_{t_j}^{s_{i'}})$: 観測資源の割り当ての整合性に関する制約

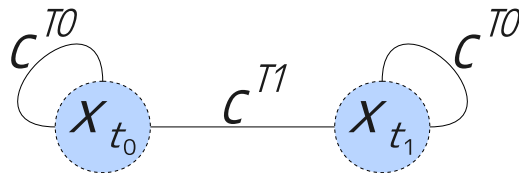


図 5: TAV による制約網の表現

この制約は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．センサ間で協調するには，同じ観測対象に割り当てられるセンサの組み合わせについて，同じ意志を示さなければならない．そのため，同じ観測対象に対する観測資源の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この制約は違反となる．この制約は緩和不可能である．

制約 c^{ST2} に対する評価関数 $f_{c^{ST2}}$ は式 (3) のように表される．ただし $w^{c^{ST2}}$ は制約の評価値を示す定数値である．

$$f_{c^{ST2}}(x_{t_j}^{s_i}, x_{t_j}^{s_{i'}}) = \begin{cases} w^{c^{ST2}} & x_{t_j}^{s_i} \neq x_{t_j}^{s_{i'}} \\ 0 & otherwise \end{cases} \quad (3)$$

STAV を用いて形式化した場合，各センサは観測可能な観測対象の数だけ変数を持つ．そのため，観測対象数が増えた場合，システム全体の変数が増加する．また，観測対象数と同じ場合でも，センサの数が増えれば変数の数が増加する．それに伴い制約の本数が増加する．

3.2.2 TAV(Target As Variable)

TAV は観測対象を変数として定義する表現である．この表現に類似する表現は文献 [6] で用いられている．この表現では各観測対象について変数を定義するので，1 個の観測対象に関する変数は全体で 1 個である．図 3 の配置の制約網を TAV を用いて表現すると図 5 のようになる．図 5 において x_{t_j} は観測対象 t_j に関する変数を表している． x_{t_j} の値はどのセンサが観測対象 t_j に割り当てられているかを示す．観測対象 t_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を表す変数値をとる．変数 x_{t_j} の値域は $\{\phi, \{s_0\}, \dots, \{s_n\}, \{s_0, s_1\}, \dots, \{s_0, \dots, s_n\}\}$ と表すことができる．

この表現における制約は図 5 中 c^{T0}, c^{T1} で示した 2 種類の制約がある．

- $c^{T0}(x_{t_j})$: 観測対象への観測資源割り当てに関する制約

各観測対象に可能な限り 3 個のセンサを割り当てるための制約である．本論文で想定するシステムでは，可能な限り各観測対象に 3 個のセンサを割り当てること

を目的としている．そのため，観測対象 t_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^{T0} に対する評価関数 $f_{c^{T0}}$ は式 (4) で表現される．ただし $w^{c^{T0}}$ は制約の評価値を示す定数値である．式 (4) では n_j は観測対象 t_j に割り当てられたセンサの数を示す．

$$f_{c^{T0}(x_{t_j})} = \begin{cases} w_0^{c^{T0}} & n_j = 0 \\ w_1^{c^{T0}} & n_j = 1 \\ w_2^{c^{T0}} & n_j = 2 \\ 0 & otherwise \end{cases} \quad (4)$$

- $c^{T1}(x_{t_j}, x_{t_{j'}})$: 観測資源の制限に関する制約

1 個の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ための制約である．本論文で想定するシステムでは，各センサは同時に複数の観測対象に割り当てることができないとした．そのため，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^{T1} に対する評価関数 $f_{c^{T1}}$ は式 (5) で表される．ただし $w^{c^{T1}}$ は制約の評価値を示す定数値である．

$$f_{c^{T1}(x_{t_j}, x_{t_{j'}})} = \begin{cases} w^{c^{T1}} & x_{t_j} \cap x_{t_{j'}} \neq \phi \\ 0 & otherwise \end{cases} \quad (5)$$

TAV を用いた形式化では STAV を用いた場合よりも変数及び制約の数が少ない．また，観測対象ごとに変数を定義するため，センサの数が増えても変数の数は増加しない．そのため，問題規模が大きくなっても，問題の複雑さは STAV よりも抑えられると考えられる．しかし，本論文で想定する観測システムでは，観測対象は制御の対象ではなく，制御の対象はセンサのみである．そのため，TAV を用いた形式化をそのまま適用することは不可能である．

3.3 エージェントによる協調モデル

分散制約最適化問題の枠組とは別にセンサ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 [7][8][9] が提案されている．ここでは，エージェントによる協調モデルと呼ぶ．エージェントによる協調モデルの概念図を図 6 に示す．このシステムでは，視野を制御可能な複数のカメラセンサと複数の観測対象からなる状況

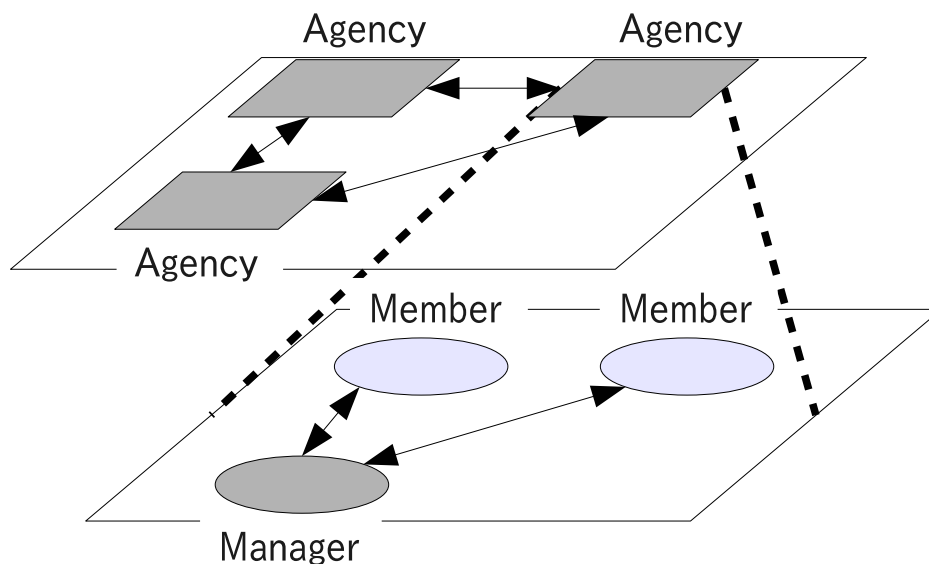


図 6: エージェンシによる協調モデル

を想定しており，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センサ（AVA: ActiveVisionAgent）の集合によって構成される．このシステムの概要は以下の通りである．

- AVA を割り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，観測対象ごとにエージェンシを作成する
- AVA 同士で情報を交換し，各観測対象に対して1つのエージェンシが存在するように調整する
- 各エージェンシには代表のAVA（エージェンシマネージャ）が1個存在する他のAVAはエージェンシマネージャの意思決定に従う
- 各エージェンシ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エージェンシマネージャに集約される

このシステムは実機を用いた小規模な実験環境で有効性が実証されている．よって，階層的な協調構造は効率的な協調動作を実現する手法として有効であると考えられる．しかし，この手法では，分散制約最適化問題のような最適化問題としての問題表現とその解法は用いられていない．

4 提案手法: 2つの階層からなる形式化

3.2.1で説明したSTAVや3.2.2で説明したTAVのような制約網による形式化に，エージェンシ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

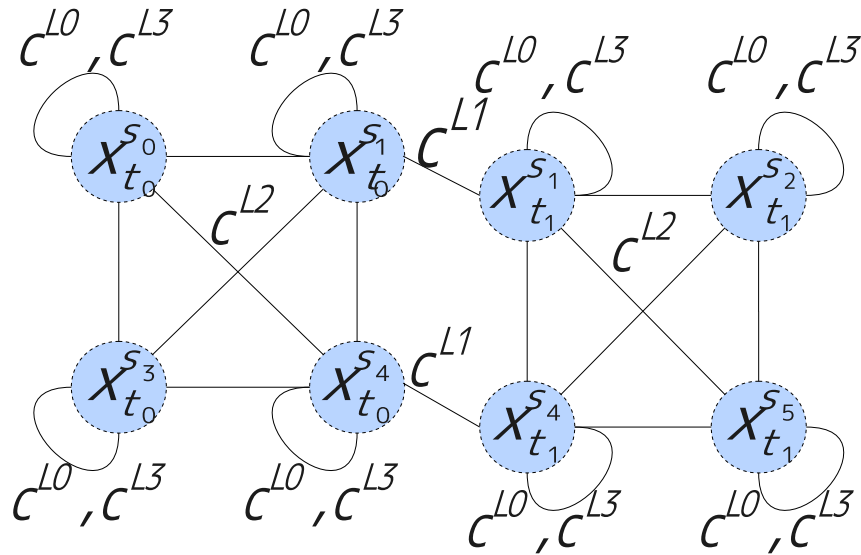


図7: リーダー選出層

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，このような形式化とその解法を提案する．すなわち，分散センサ網における観測資源割り当て問題をリーダー選出層とリーダーによる観測資源割り当て問題解決層の2つの階層に分割する．リーダー選出層では各センサが協調的に各観測対象につき1個のセンサをリーダーとして選出する．観測資源割り当て問題解決層では，選出されたリーダーによって，各観測対象に割り当てられるセンサが決定される．

4.1 リーダー選出層

リーダー選出層では，各観測対象に対して1個のセンサをリーダーとして選出する．これはエージェントによる協調モデルにおいて，各観測対象に対してエージェントを生成し，エージェント間で各観測対象につきエージェントが1個になるように調整することに相当する．

エージェントは観測対象ごとのグループに分けられる．観測対象 t_j に関するグループとは観測対象 t_j のリーダーと，観測対象 t_j に割り当てられたセンサからなる集合である．リーダーは，他の観測対象に関するリーダーと情報を交換し，同じグループに属するセンサを決める．つまり観測対象 t_j のリーダーは観測対象 t_j に割り当てられるセンサを決定する．したがって，観測対象 t_j を観測可能なセンサのいずれかが観測対象 t_j のリーダーとなることが妥当である．また，後述の観測資源割り当て問題の解探索空間を削減するために，観測対象 t_j のリーダーは必ず観測対象 t_j に割り当てられることにする．

図3のような配置の場合，リーダー選出層の制約網は図7のように表現される．図7において $x_{t_j}^{s_i}$ はセンサ s_i が持つ観測対象 t_j に関する変数である． $x_{t_j}^{s_i}$ の値は，観測対象 t_j のリーダーを示す．観測対象 t_j を観測可能なセンサの集合が $\{s_0, \dots, s_n\}$ ならば，センサ s_i がもつ観測対象 t_j に関する変数 $x_{t_j}^{s_i}$ は $\{\phi, s_0, \dots, s_n\}$ 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7中 $c^{L0}, c^{L1}, c^{L2}, c^{L3}$ で示した4種類ある．

- $c^{L0}(x_{t_j}^{s_i})$: 観測対象へのリーダーの選出に関する制約

この制約は各観測対象へ1個のセンサをリーダーとして選出するための制約である．観測対象 t_j に対してリーダーが選出されていない場合にこの制約は違反となる．この制約は緩和可能である．

制約 c^{L0} に対する評価関数 $f_{c^{L0}}$ は式(6)のように表される．ただし $w^{c^{L0}}$ は制約の評価値を示す定数値である．

$$f_{c^{L0}(x_{t_j}^{s_i})} = \begin{cases} w^{c^{L0}} & x_{t_j}^{s_i} = \phi \\ 0 & otherwise \end{cases} \quad (6)$$

- $c^{L1}(x_{t_j}^{s_i}, x_{t_{j'}}^{s_{i'}})$: リーダーであるセンサの観測資源の制限に関する制約

この制約は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ことを制限するための制約である．本論文では，センサを同時に複数の観測対象に割り当てることはできないとした．また，観測対象 t_j のリーダーになったセンサは観測対象 t_j を観測するセンサとして必ず割り当てられる．つまり，センサを同時に複数の観測対象に関するリーダーとして選出することは出来ない．そのため，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この制約は違反となる．この制約は緩和不可能である．

制約 c^{L1} に対する評価関数 $f_{c^{L1}}$ は式(7)のように表される．ただし $w^{c^{L1}}$ は制約の評価値を示す定数値である．

$$f_{c^{L1}(x_{t_j}^{s_i}, x_{t_{j'}}^{s_{i'}})} = \begin{cases} w^{c^{L1}} & x_{t_j}^{s_i} = x_{t_{j'}}^{s_{i'}} \\ 0 & otherwise \end{cases} \quad (7)$$

- $c^{L2}(x_{t_j}^{s_i}, x_{t_j}^{s_{i'}})$: 選出されたリーダーの整合性に関する制約

この制約は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．同じ観測対象に関するリーダーがセンサ間で異なっていた場合，正しく観測資源割り当て問題を解くことができない．そのため，同じ観測対象についての変数の値が

センサ間で異なっていた場合にこの制約は違反となる．この制約は緩和不可能である．

制約 c^{L2} に対する評価関数 $f_{c^{L2}}$ は式 (8) のように表される．ただし $w^{c^{L2}}$ は制約の評価値を示す定数値である．

$$f_{c^{L2}}(x_{t_j}^{s_i}, x_{t_j}^{s_{i'}}) = \begin{cases} w^{c^{L2}} & x_{t_j}^{s_i} \neq x_{t_j}^{s_{i'}} \\ 0 & otherwise \end{cases} \quad (8)$$

- $c^{L3}(x_{t_j}^{s_i})$: 観測対象に割り当て可能なセンサの確保に関する制約

この制約は各観測対象に割り当て可能なセンサ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．観測対象 t_j のリーダーは必ず観測対象 t_j を観測するセンサとして割り当てられるため，他の観測対象に割り当て可能なセンサの集合から除外される．そのため，もともと 3 個以上のセンサを割り当てることが可能な観測対象に対して 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．この様な状況を避けるために，可能な限り観測可能な観測対象の数が少ないセンサをリーダーとして選ぶことを意図して制約 c^{L3} を設ける．観測可能な観測対象の数が少ないセンサを選べば，割り当て可能なセンサの数が減少する観測対象も減少する．この制約は観測可能な観測対象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^{L3} に対する評価関数 $f_{c^{L3}}$ は式 (9) のように表される．ただし $w^{c^{L3}}$ は制約の評価値を示す定数値である．また， I_j は観測対象 t_j を観測可能なセンサの集合を示し， v_i はセンサ s_i が観測可能な観測対象の数を示す．

$$f_{c^{L3}}(x_{t_j}^{s_i}) = \begin{cases} w^{c^{L3}} & \exists s_i \in I_j, v_i < v_{x_{t_j}^{s_i}} \\ 0 & otherwise \end{cases} \quad (9)$$

リーダー選出層では 3.2.1 で説明した STAV を用いた制約網の表現と同様でセンサと観測対象の組を変数として定義するため，変数の数は STAV を用いた手法と同じであり，さらに制約が増えている．しかし，各変数の取り得る値の集合の要素数はリーダー選出層の方が少なくなる．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 個である．そのため，STAV を用いた手法では，観測対象を観測可能なセンサを観測対象に割り当てられるか割り当てられないかを選ぶので，各変数の値域は $2^4 = 16$ 通となる．一方，リーダー選出層では観測対象を観測可能な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かの 5 通りである．各変数の取り得る値の集合の

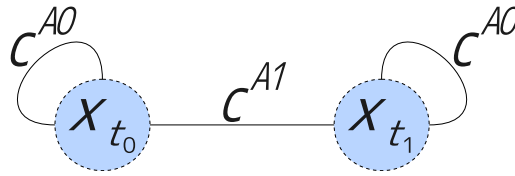


図 8: 観測資源割り当て問題解決層

要素数が少ない方が，解探索空間が小さく，比較的容易に解くことが出来ると考えられる．

4.2 観測資源割り当て問題解決層

観測資源割り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センサをどの観測対象に割り当てるかという問題を解決する．各リーダーは互いに情報を交換して問題を解き，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 3 のような配置の場合，制約網は図 8 のように表現される．図 8 において x_{t_j} は観測対象 t_j のリーダーに選出されたセンサが持つ観測対象 t_j に関する変数である． x_{t_j} の値は，観測対象 t_j を観測可能なセンサのうち，どのセンサが観測対象 t_j に割り当てられているかを示す．観測対象 t_j を観測可能なセンサの集合が $\{s_0, \dots, s_n\}$ ならば，観測対象 t_j のリーダーが持つ変数 $x_{t_j}^{s_i}$ は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，観測対象 t_j のリーダーは観測対象 t_j に必ず割り当てられるため，各観測対象 t_j 以外の観測対象を観測可能なセンサの集合から除外される．

リーダーによる観測資源割り当て問題解決層における制約は，図 8 中 c^{A0}, c^{A1} で示した 2 種類ある．

- $c^{A0}(x_{t_j})$: 観測対象への観測資源割り当ての要求に関する制約

この制約は観測対象への観測資源割り当ての要求に関する制約である．本論文で想定するシステムでは，可能な限り各観測対象に 3 個のセンサを割り当てることを目的としている．そのため，観測対象 t_j に割り当てられているセンサの数が 3 個に満たない場合にこの制約は違反となる．この制約は緩和可能である．

制約 c^{A0} に対する評価関数 $f_{c^{A0}}$ は式 (10) のように表される．ただし $w^{c^{A0}}$ は制約の評価値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の数を

示す．

$$f_{c^{A0}}(x_{t_j}) = \begin{cases} w_0^{c^{A0}} & n_j = 0 \\ w_1^{c^{A0}} & n_j = 1 \\ w_2^{c^{A0}} & n_j = 2 \\ 0 & otherwise \end{cases} \quad (10)$$

- $c^{A1}(x_{t_j}, x_{t_{j'}})$: 観測資源の制限に関する制約

この制約は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ための制約である．リーダーによる観測資源割り当て問題解決層では，各観測対象につき 1 個の変数が存在し，観測対象 t_j にどのセンサが割り当てられているかを示す．そのため，複数の変数に同じセンサが割り当てられていた場合，そのセンサは複数の観測対象に割り当てられたことになる．そのため，1 個のセンサが同時に 2 個以上の変数値の要素である場合にこの制約は違反となる．この制約は緩和不可能である．

制約 c^{A1} に対する評価関数 $f_{c^{A1}}$ は式 (11) のように表される．ただし $w^{c^{A1}}$ は制約の評価値を示す定数値である．

$$f_{c^{A1}}(x_{t_j}, x_{t_{j'}}) = \begin{cases} w^{c^{A1}} & x_{t_j} \cap x_{t_{j'}} \neq \phi \\ 0 & otherwise \end{cases} \quad (11)$$

観測資源割り当て問題解決層の問題の表現は TAV と同様であるため，変数と制約の数が STAV で制約網を表現する場合よりも少ない．また，観測対象 t_j に関するリーダーは必ず観測対象 t_j 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．つまり，解探索空間が削減される．この 2 点から，観測資源割り当て問題解決は STAV を用いた手法よりも比較的容易に問題を解くことが出来ると考えられる．

4.3 2つの階層からなる形式化を用いた場合の解法

図 10 に 2 つの階層からなる形式化を用いた場合の解法の概念図を示す．2 つの階層には，それぞれ独立した分散制約最適化問題の解法を適用する．それぞれの階層では自身の変数値と制約で関係するセンサから送られてきた変数値に基づいて独立に問題を解く（図 10 中の a, b）．リーダーによる観測資源割り当て問題解決層での変数は，リーダー選出層での変数の 1 個と観測対象に割り当てることが可能なセンサの集合や観測対象の情報と同じであるため，リーダー選出層の変数の情報に基づき作成したもので

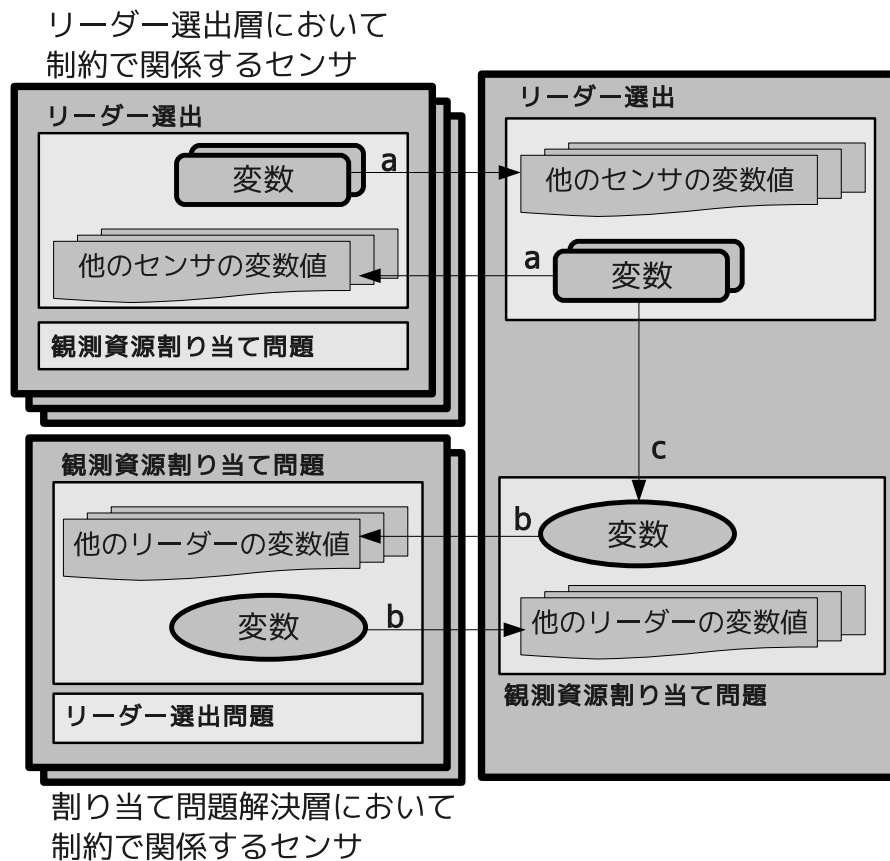


図 9: 階層化による形式化の概念図

ある (図 10 中 c) .

2 つの階層からなる形式化を用いた場合の解法の擬似コードを図 11 に示す . 図 11 はセンサ s_i を主体とした擬似コードである . 図 11 中の `my_status` は自分がリーダーに選出されているかどうかを示す . `my_status = true` であれば , センサ s_i はリーダーに選出されている . また , `leader_list` はリーダーであるセンサのリストである .

観測資源割り当て問題解決層では , リーダー選出層の解に基づき問題を解く必要がある . 一方で , 各センサは制約で関係するリーダーの観測資源割り当て問題解決層における変数値に基づいて自身の割り当て問題を解決する . すなわち , 観測資源割り当て問題解決層において制約で関係するリーダーからの変数値を得ることができれば , 自身に関する部分問題を解くことができる . したがって , 自身がリーダーに選出されたと判断したとき , 観測資源割り当て問題解決層において制約で関係する可能性のあるセンサの集合 S_a に含まれるセンサにリーダーであることを通知する . 自身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 種類である .

- 観測可能な観測対象のいずれかのリーダーに自身が選出されている．この条件はセンサ s_i が観測可能な観測対象の集合を J とすると，式 (12) のように表すことができる．

$$l_1 = \begin{cases} true & \exists t_j \in J, x_{t_j}^{s_i} = i \\ false & otherwise \end{cases} \quad (12)$$

- リーダー選出層における緩和しない制約がすべて満たされている．この条件は緩和しない制約の集合を C とすると，式 (13) のように表すことができる．

$$l_2 = \begin{cases} true & \forall c \in C, f_c = 0 \\ false & otherwise \end{cases} \quad (13)$$

$l_1 \wedge l_2 = true$ であれば，自身がリーダーであると判断する．そして，図 11 中 18-22 行目における処理で，メッセージ *is_leader* を生成する．メッセージ *is_leader* は自身がリーダーであることを通知するためのメッセージである．生成したメッセージ *is_leader* は S_a に含まれるセンサに送信される． S_a に含まれるセンサは，観測可能な観測対象に割り当てることが可能なセンサ，すなわち観測対象を観測可能なセンサの一部がリーダーとなったセンサが担当する観測対象に割り当てることが可能なセンサと競合しているセンサである．観測対象 t_j に割り当てることが可能なセンサの集合を S_{t_j} ，それ以外の観測対象 $t_{j'}$ が観測可能な観測対象に割り当てることが可能なセンサの集合を $S_{t_{j'}}$ とすると，観測対象 t_j のリーダーに選出されたセンサ s_i がリーダーであることを通知するセンサの集合 S_a に含まれるセンサは式 (14) を満たす観測対象 $t_{j'}$ を観測可能なセンサである．

$$l_3 = \begin{cases} true & S_{s_i} \cap S_{s_x} \neq \phi \\ false & otherwise \end{cases} \quad (14)$$

$l_3 = true$ である観測対象 t_j を観測可能であるセンサは S_a に含まれるセンサである．グリッドモデルにおいて， S_a に含まれる可能性のあるセンサは，グリッド上で近傍であるセンサと，近傍であるセンサのさらに近傍であるセンサである．そのため，メッセージ *is_leader* はまずグリッド上で近傍であるセンサに送信される．そして，図 11 中 8-11 行目における処理で，メッセージ *is_leader* を受信したセンサは同じメッセージを生成しグリッド上で近傍であるセンサに伝搬させる．

リーダー選出層において全てのセンサが解を得ていない状況では，リーダーとなっ

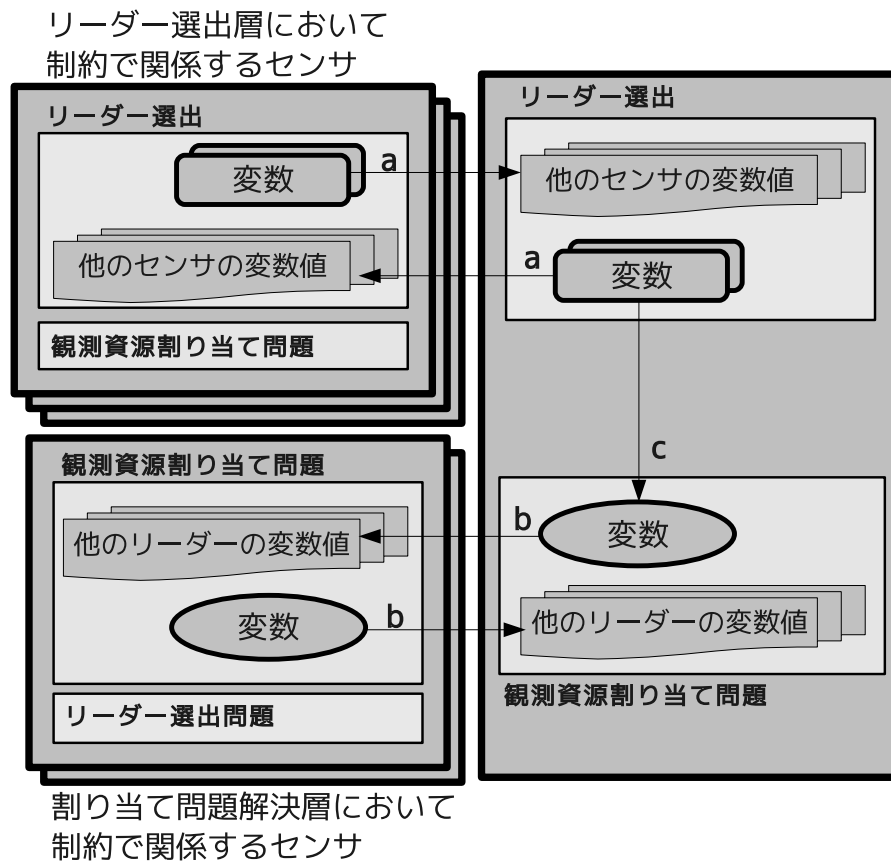


図 10: 階層化による形式化の概念図

ていたセンサがリーダーである条件を満たせなくなる可能性もある。その場合には、図 11 中 23-28 行目における処理で、メッセージ *not_leader* を生成し、 S_a に含まれるセンサに送信する。メッセージ *not_leader* は自身がリーダーでなくなったことを通知するためのメッセージである。メッセージ *not_leader* を受信したセンサは、図 11 中 12-15 行目における処理で、同じメッセージを生成しグリッド上で近傍であるセンサに送信する。

図 11 中 V は 2 つの階層における変数の集合を示す。*is_leader*, *not_leader* 及び変数値は図 11 中 32 行目における処理で一括で送信する。

5 提案手法: 動的な問題への対応

階層化された問題構造を分散制約最適化問題の枠組みに導入するということは、エージェントのグループを形成しグループ間での交渉を行うことを分散制約最適化問題の枠組みに導入することを志向している。グループを形成する事で、観測対象 t_j のリー

```

1 initialize V;
2 my_status ← false;
3 empty leader_list;
4 while
5 previous_status ← my_status;
6 receive others' messages And add others' messages to message_list;
7 foreach message_list do
8   if receive is_leader( $s_x$ ) then
9     add  $s_x$  to leader_list;
10    generate is_leader( $s_x$ );
11   end if
12   if receive not_leader( $s_x$ ) then
13     remove  $s_x$  from leader_list;
14     generate not_leader( $s_x$ );
15   end if
16 end
17 DSTS for leader election layer;
18 if Previous_status  $\neq$  leader then
19   if  $l_1 \wedge l_2 = true$  then
20     my_status ← leader;
21     generate is_leader( $s_i$ );
22   end if
23 if previous_status = leader then
24   if  $l_1 \wedge l_2 = false$  then
25     my_status ← notleader;
26     generate not_leader( $s_i$ );
27   end if
28 end if
29 if my_status == leader then
30   DSTS for resource allocation layer
31 end if
32 send messages(is_leader, not_leader, V) to others.
33 end while

```

図 11: 2つの階層からなる形式化を用いた場合の解法

ダーには観測対象 t_j に割り当て可能なセンサ，すなわち観測対象 t_j を観測可能なセンサがどの観測対象に割り当てられたかという情報が集約される．その情報を用いることにより動的な問題，すなわち時刻とともに観測対象の配置が変化する問題における解の安定性を考慮した要求に対応する制約を表現する枠組みも導入する．解を安定させるとは，問題が変化した際に観測対象に割り当てられたセンサの組み合わせの変化を抑制することである．

基本的に動的な問題への追従は，ある時刻のスナップショットを反復的に解くこととしてとらえることが出来る．また，想定するシステムでは，頻繁なセンサの視野方向の制御による観測情報の劣化を抑制する観点から，観測対象に割り当てられたセン

サの組み合わせが頻繁に変化することは望ましくないと考えられる．そこで，観測対象に割り当てられたセンサの組み合わせの変化を少なくするために，センサの割り当ての状態を考慮し，その状態を交換する前処理を導入する．また，センサの割り当ての状態を観測対象ごとに表として管理し，それを用いて観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約を設ける．また，本論文では，観測対象の配置が変化する前の状態を問題変化前と表記し，観測対象の配置が変化した後の状態を問題変化後と表記する．

5.1 センサの割り当ての状態

観測対象の配置が変化した後に観測対象に割り当てられたセンサの組み合わせの変化を抑制するためには，各センサがどの観測対象に割り当てられているかという情報が必要である．そのために，観測対象 t_j に割り当て可能なセンサの割り当て問題解決層での割り当ての状態を次のように分類する．この分類は，観測対象 t_j に関するグループと観測対象 t_j に割り当て可能なセンサの関係を表す．ここでの観測対象 t_j に関するグループとは，4 で述べたとおり，観測対象 t_j に関するリーダーと観測対象 t_j に割り当てられたセンサの集合である．

- Inside_group : 観測対象 t_j に関するグループに属するセンサ
- Outside_group : 観測対象 t_j に関するグループに属さないセンサ
- New : 観測対象 t_j に新しく割り当てることが可能になったセンサ
- Other_Leader : 観測対象 t_j 以外の観測対象のリーダー
- Free : 上記以外のセンサ

先述のとおり，この分類は観測対象 t_j に関するグループと観測対象 t_j に割り当て可能なセンサの関係を表すものである．そのため，観測対象 t_j 以外の観測対象にも割り当て可能なセンサは，別の観測対象に関する割り当ての状態は異なった状態を取る可能性がある．例えば，観測対象 t_j に関する割り当ての状態が Inside_group であるセンサは，それ以外の観測対象については割り当ての状態が Outside_group となる．また，これらの状態を保持する表を変数ごとに生成する．

5.2 変数の生成・削除，表の作成

動的な問題では，観測対象の配置が変化し，あるセンサが観測できる観測対象が変化した場合，センサが持つ変数に変化が生じる．自身が持つ変数に変化が生じたセンサは変数の生成・削除を行い制約で関係する可能性のあるセンサに対し変数の生成・

削除に関して通知するためのメッセージを送信する．また，観測対象に割り当てられたセンサの組み合わせの変化を抑制するために，先述のセンサの割り当て状態を保持した表を生成する必要がある．その際の擬似コードを図 12 に示す．図 12 はセンサ s_i が主体の処理を示す擬似コードである．図 12 において， V はセンサ s_i が持つリーダー選出層における変数の集合を示す． my_status はセンサ s_i がリーダーであるかどうかを示し， $my_status = true$ であればセンサ s_i はリーダーに選出されている． v_{t_j} はリーダー選出層における観測対象 t_j に関する変数であり， $v_{t_j}^l$ は観測対象 t_j のリーダーが持つ割り当て問題解決層における変数である． $v_{t_j}.s_list$ は観測対象 t_j を観測可能なセンサ，すなわち観測対象 t_j に割り当てることが可能なセンサの集合を示し， $v_{t_j}.table$ は観測対象 t_j に関する上述のセンサの割り当ての状態を保持する表を示す．また，この表は状態表と表記する．

5.2.1 変数の生成・削除

変数の生成

センサ s_i が新しく観測対象 t_j を観測可能になった場合，図 12 中 5-9 行目における処理で，観測対象 t_j に関する変数を生成し観測対象 t_j に関するメッセージ new_target を生成する．観測対象 t_j に関するメッセージ new_target には観測対象 t_j に関する変数と観測対象 t_j に対するセンサ s_i の割り当ての状態が含まれる．センサ s_i の割り当ての状態に関する処理の詳細は後述する．観測対象 t_j に関するメッセージ new_target は観測対象 t_j に関する変数についてリーダー選出層において制約で関係する可能性のあるセンサの集合 S_l に通知する．新しく観測対象 t_j に関する変数を生成したセンサ s_i が観測可能な領域の集合を A_{s_i} ，それ以外のセンサ s_x が観測可能な領域の集合を A_{s_x} とすると，式 (15) を満たすセンサ s_x が S_l に含まれる．

$$A_{s_i} \cap A_{s_x} \neq \phi \quad (15)$$

グリッドモデルにおいて， S_l に含まれるセンサはグリッド上でセンサ s_i の近傍であるセンサである．観測対象 t_j に関するメッセージ new_target を受信したセンサ s_x は，図 12 中 16-22 行目における処理で，自身が同じ観測対象 t_j に関する変数を持っていた場合，センサ s_i を観測対象 t_j に割り当てることが可能なセンサの集合に追加しメッセージ $own_message$ を生成する． $own_message$ には観測対象 t_j に対するセンサ s_x の割り当ての状態が含まれる．観測対象 t_j に対するセンサ s_x の割り当ての状態に関する処理の詳細は後述する．メッセージ own_status は観測対象 t_j に関するメッセージ new_target の送信元であるセンサ s_i に送信される．メッセージ $own_message$ を受信

```

1 my_status ← false;
2 initialize V;
3 while
4 detect targets;
5 if new target  $t_j$  is detected then
6     generate new variable  $v_{t_j}$ ;
7     add  $s_i$  to  $v_{t_j}.s\_list$ ;
8     generate  $new\_target(s_i, v_{t_j}, status_{s_i}^{t_j})$ ;
9 end if
10 if target  $t_j$  is lost then
11     generate  $lost\_target(s_i, v_{t_j})$ ;
12     remove variable  $v_{t_j}$ ;
13 end if
14 receive others' messages And add others' messages to message_list ;
15 foreach message_list do
16     if receive  $new\_target(s_x, v_{t_k}, status_{s_x}^{t_k})$  then
17         if there is  $v_{t_k} \wedge s_x \notin v_{t_k}.s\_list$  then
18             add  $s_x$  to  $v_{t_k}.s\_list$ ;
19             add  $s_x$  to  $v_{t_k}.table(s_x, status_{s_x}^{t_k})$ ;
20             generate  $own\_status(s_i, status_{s_x}^{t_k})$ ;
21         end if
22     end if
23     if receive  $lost\_target(s_x, v_{t_k})$  then
24         if there is  $v_{t_k}$  then
25             remove  $s_x$  from  $v_{t_k}.s\_list$ ;
26             remove  $s_x$  from  $v_{t_k}.table$ ;
27         end if
28     end if
29     if receive  $own\_status(s_x, status_{s_x}^{t_k})$  then
30         if there is  $v_{t_k}$  then
31             add  $s_x$  to  $v_{t_k}.s\_list$ ;
32             add  $s_x$  to  $v_{t_k}.table(s_x, status_{s_x}^{t_k})$ ;
33         end if
34     end if
35     if receive  $own\_table(s_x, v_{t_k}^l.table)$  then
36         if there is  $v_{t_k}$  then
37              $v_{t_k}.table \leftarrow v_{t_k}^l.table$ ;
38         end if
39     end if
40 end
41 local calculation for leader election layer and resource allocation layer;
42 if my_status == true then
43     if suboptimal solution is found then
44         generate  $v_{t_{own}}^l.table$ ;
45         generate  $own\_table(s_i, v_{t_{own}}^l.table)$ ;
46          $v_{t_{own}}.table \leftarrow v_{t_{own}}^l.table$ 
47     end if
48 end if
49 send messages( $new\_target, lost\_target, own\_table$ ) to others.
50 end while

```

図 12: 問題が変化した場合の通知に関する擬似コード

したセンサ s_i は、図 12 中 29-34 行目における処理で、送信元である s_x を観測対象 t_j に割り当てることが可能なセンサの集合に追加する。

変数の削除

センサ s_i が観測対象 t_j を観測不可能になった場合は、図 12 中 10-13 行目における処理で、観測対象 t_j に関する変数を削除し観測対象 t_j に関するメッセージ *lost_target* を生成する。観測対象 t_j に関するメッセージ *lost_target* は、 $v_{t_j.s_list}$ に含まれるセンサ、すなわち観測対象 t_j を観測可能なセンサに送信される。観測対象メッセージ *lost_target* を受信したセンサは、図 12 中 23-28 行目における処理で、観測対象 t_j に関する変数を持っていた場合、 $v_{t_j.s_list}$ 、すなわち観測対象 t_j に割り当てることが可能なセンサの集合から送信元であるセンサ s_i を除外する。

5.2.2 割り当て状態を保持する表の生成

観測対象に割り当てられたセンサの変化を抑制するための制約を表現するために、各センサの割り当ての状態を保持する状態表を生成する。各センサは観測対象に割り当てることが可能なセンサの割り当ての状態に関する状態表を各観測対象につき 1 つ保持する。この状態表に関する概念図を図 13,14 に示す。この状態表は各観測対象のリーダーに選出されたセンサによって生成される場合と、各観測対象に割り当てられることが可能なセンサが生成する場合がある。

リーダーが作成する場合

ある時刻の問題の解を得た時点では観測対象のリーダーに選出されたリーダーが状態表を生成する。図 12 中 42-48 行目における処理で、観測対象 t_j に関するリーダーであるセンサ s_i は観測対象 t_j に関する状態表を生成しメッセージ *own_table* を生成する。メッセージ *own_table* には、生成した観測対象 t_j に関する状態表の情報が含まれる。観測対象 t_j のリーダーは他のリーダーと観測対象の割り当ての情報を交換し、観測対象 t_j にどのセンサを割り当ててかを決定する。そのため、観測対象 t_j のリーダーは観測対象 t_j に割り当てることが可能なセンサが他の観測対象に割り当てられているかどうかを知る。その情報を用いて担当する観測対象 t_j に関する状態表を生成する。つまり、リーダーが生成する場合、他のリーダーから得られた変数値と自身の変数値に基いて生成する（図 13 中 a）。そして、メッセージ *own_table* は観測対象 t_j に割り当てることが可能なセンサ、すなわち観測対象 t_j を観測可能なセンサに送信される（図 13 中の b）。メッセージ *own_table* を受信したセンサは、図 12 中 35-39 行目における処理で、同じ観測対象に関する状態表を更新する。

各センサが作成する場合

問題が変化し、観測対象を観測可能なセンサの集合に変化が生じた場合は、図 14 に示すように、他のセンサと情報を交換しつつ各センサが状態表を生成する。問題が変化し、新しく観測対象 t_j に割り当てられることが可能になったセンサ s_i は、図 12 中 5-9 行目における処理で、観測対象 t_j に関する変数を生成し観測対象 t_j に関するメッセージ new_target を生成する。観測対象 t_j に関するメッセージ new_target には観測対象 t_j に対するセンサ s_i の割り当ての状態 $status_{s_i}^{t_j}$ が含まれる。すなわち、観測対象 t_j を観測可能なセンサに、自身の観測対象 t_j に関する割り当ての状態を通知する（図 14 中 a）。メッセージ new_target は、先述のとおり、リーダー選出層において制約で関係する可能性のあるセンサの集合 S_l に通知する。観測対象 t_j に関するメッセージ new_target を受信したセンサ s_x は、図 12 中 16-22 行目における処理で、観測対象 t_j に関する状態表にセンサ s_i を追加し、メッセージ own_status を生成する。メッセージ own_status には観測対象 t_j に関するセンサ s_x の割り当ての状態 $status_{s_x}^{t_j}$ が含まれる。メッセージ own_status はメッセージ new_target の送信元であるセンサ s_i に送信される（図 14 中 b）。センサ s_i が観測対象 t_j を観測不可能になり、観測対象 t_j に割り当てることが可能なセンサの集合からセンサ s_i が除外される場合、図 12 中 10-13 行目における処理で、観測対象 t_j に関するメッセージ $lost_target$ を生成する。観測対象 t_j に関するメッセージ $lost_target$ は $v_{t_j, s_i, ist}$ に含まれるセンサ、すなわち観測対象 t_j に割り当てることが可能なセンサに送信される。観測対象 t_j に関するメッセージ $lost_target$ を受信したセンサは、図 12 中 23-28 行目における処理で、観測対象 t_j に関する状態表からセンサ s_i の割り当ての状態に関する情報を除外する。各観測対象に割り当てることが可能なセンサは作成する場合、状態表に関してメッセージを送信する対象は、変数の生成・削除の際にメッセージを送信するセンサと同じである。また、これらの処理は変数の生成・削除と同じタイミングで行われる。そのため、変数の生成・削除に関する処理に付随して処理を行う。

5.3 観測対象に割り当てられたセンサの組み合わせの変化の抑制

観測対象に割り当てられたセンサの組み合わせの変化を抑制するために、リーダー選出に 2 種類の制約を設ける。また、観測資源割り当て問題解決層には 1 種類の制約を設ける。これらの制約では、先述の状態表を用いる。

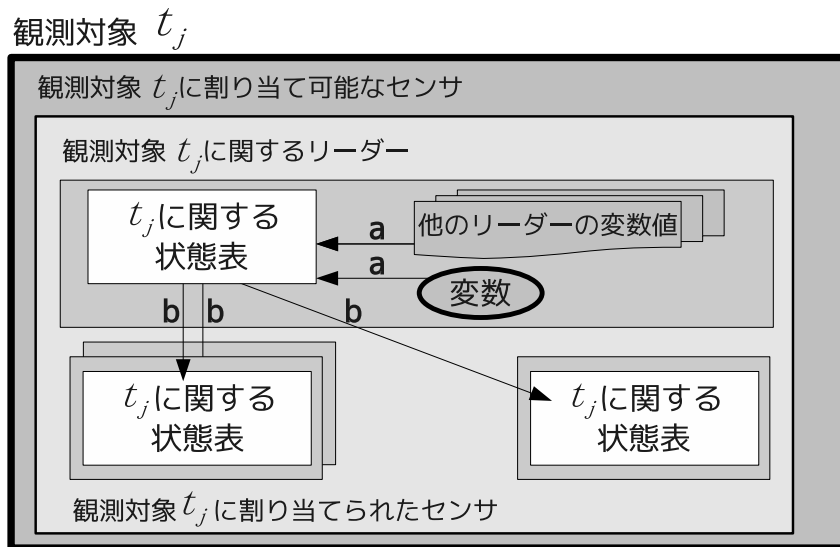


図 13: リーダーによる状態表の作成

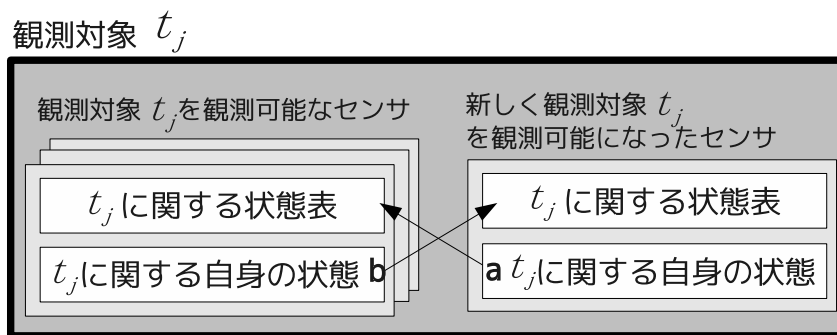


図 14: 各センサによる状態表の生成

5.3.1 リーダーを同じグループから再選出するための制約

前述の状態表と問題変化前の変数値を用いて、リーダーを同じグループから再選出するための制約を設ける。観測対象に対するグループの構造と観測対象に割り当てられたセンサの組み合わせの変化を抑制するためには、問題変化後に、問題変化前と同じセンサをリーダーとして選出する必要がある。問題が変化した際に観測対象 t_j に関するリーダーとなっていたセンサが観測対象 t_j に割り当てることが可能なセンサではなくなった場合は、同じグループ内からリーダーを選出する。

- c^{la0} : リーダーの変化を抑制するための制約

問題変化前に観測対象 t_j に関するリーダーに選出されていたセンサと同じセンサを観測対象 t_j に関するリーダーとして選出するための制約である。問題変化前のリーダーと現在のリーダー選出層における変数値が異なっていた場合この制約は

違反となる．制約 c^{la0} に対する評価関数 $f_{c^{la0}}$ は式 (16) のように表される．ただし $w^{c^{la0}}$ は制約の評価値を示す定数値である．また， pre_{t_j} は観測対象 t_j の配置が変わる前のリーダーを示している．つまり問題変化前のリーダーである．

$$f_{c^{la0}(pre_{t_j}, x_{t_j}^{s_i})} = \begin{cases} w^{c^{la0}} & pre_{t_j} \neq x_{t_j}^{s_i} \\ 0 & otherwise \end{cases} \quad (16)$$

ただし，問題変化前に観測対象 t_j にリーダーが選出されなかった場合と問題変化前のリーダーが観測対象 t_j に割り当てることが可能なセンサではない場合は制約違反とはならない．

- c^{la1} : リーダーを同じグループから選出するための制約

観測対象 t_j のリーダーが観測対象 t_j を観測不可能になり，リーダーを変えなければならない場合，観測対象 t_j に関するグループ内のセンサの中からリーダーを選出するための制約である．観測対象 t_j に関するグループ内からリーダーが選出されなかった場合に，この制約は違反となる．制約 c^{la1} に対する評価関数 $f_{c^{la1}}$ は式 (17) のように表される．ただし $w^{c^{la1}}$ は制約の評価値を示す定数値である．

$$f_{c^{la1}(x_{t_j}^{s_i})} = \begin{cases} w_0^{c^{la1}} & x_{t_j}^{s_i} \text{ is } New \text{ or } Free \\ w_1^{c^{la1}} & x_{t_j}^{s_i} \text{ is } Inside_group \\ w_2^{c^{la1}} & x_{t_j}^{s_i} \text{ is } Outside_group \\ w_3^{c^{la1}} & x_{t_j}^{s_i} \text{ is } Other_Leader \end{cases} \quad (17)$$

1 度作られたグループ構造を可能な限り維持するには，他の観測対象に割り当てられているセンサはリーダーとして適切でなく，割り当てられているセンサは適切であるため，各定数値の大小関係は式 (18) で表される．

$$w_3^{c^{la1}} > w_2^{c^{la1}} > w_0^{c^{la1}} > w_1^{c^{la1}} = 0 \quad (18)$$

5.3.2 観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約

リーダーだけでなく，リーダーが担当する観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約を割り当て問題解決層に設ける．この制約では，状態表を用いる．

- c^{aa0} : 観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約
この制約では状態表を利用し評価値を計算する．この制約は次の 2 つの場合にこの制約は違反となる．

- 観測対象 t_j に割り当てられていたセンサが観測対象 t_j に割り当てられなかった場合
- 観測対象 t_j に観測対象 t_j 以外のセンサに割り当てられていたセンサを割り当てた場合

制約 c^{aa0} に対する評価関数 $f_{c^{aa0}}$ は式 (19) のように表される．式 (19) では観測対象 t_j に割り当てることが可能な k 番目のセンサを $s_{t_j}^k$ と示す．また， $w^{c^{aa0}}$ は制約の評価値を示す定数値である．

$$f_{c^{aa0}(x_{t_j})} = \begin{cases} w_0^{c^{aa0}} & s_{t_j}^k \text{ is } Inside_Group \wedge s_{t_j}^k \notin x_{t_j} \\ w_1^{c^{aa0}} & s_{t_j}^k \text{ is } Outside_Group \wedge s_{t_j}^k \in x_{t_j} \end{cases} \quad (19)$$

これを全ての割り当て可能なセンサについて計算する．

5.4 部分的な観測対象に割り当てられたセンサの組み合わせの変化の抑制

解の精度の向上，すなわち評価値を小さくするためには，観測対象に割り当てられたセンサの組み合わせの変化を抑制しないほうがよいと考えられる．なぜならば，観測対象に割り当てられたセンサの組み合わせの変化を抑制した場合，各観測対象に割り当てられたセンサの数に不均衡が生じる可能性があるためである．

例として，観測対象に割り当てられたセンサの組み合わせの変化を完全に抑制したと仮定する．その場合，問題変化後に各観測対象に割り当てられるのは `Inside_group, Nerw, free` であるセンサのみである．これらの状態であるセンサは限られる．つまり，1つの観測対象に割り当てることが可能なセンサの数が減少する観測対象が増加すると考えられる．

観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約は緩和可能であるが，評価した場合は上記のような状況が起こるケースがあると考えられる．そのため，観測対象に割り当てられたセンサの組み合わせの変化を抑制しないほうが，解の精度が良くなると考えられる．

そこで，観測資源の割り当てを決定するリーダーを次の2種類に分類する．

- 観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} について評価するリーダー
- 観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} について評価しないリーダー

ある時刻の問題を解きはじめた際に，観測対象 t_j に関するリーダー s_i が担当する観

測対象 t_j に割り当てられているセンサの数を num_{t_j} とすると、観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} について評価するかどうかは式 (20) で判断する。

$$l_{co} = \begin{cases} true & num_{t_j} = 3 \\ false & otherwise \end{cases} \quad (20)$$

$l_{co} = true$ であればリーダーは観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} について評価する。つまり、担当する観測対象に十分な数のセンサが割り当てられており、かつ割り当てられたセンサの数が過剰でないリーダーが観測対象に割り当てられたセンサの変化を抑制するための制約 c^{aa0} について評価する。一方、担当する観測対象に割り当てられたセンサの数が不十分か、過剰に割り当てられているリーダーはこの制約を評価しない。

この様に観測資源の割り当てを決定するリーダーを分類することで、部分的に観測対象に割り当てられたセンサの組み合わせの変化を抑制する作用が働き、必要な部分で観測対象に割り当てられたセンサの数に関して最適化が行われる効果が期待できる。

6 評価

3.2.1 で説明した STAV を用いた手法と 4 で説明した 2 つの階層からなる形式化を用いた提案手法を比較し、提案手法の有効性を評価した。また、提案手法においては問題が変化する際に観測対象に割り当てられたセンサの組み合わせの変化を抑制する制約を追加した手法も評価した。STAV を用いた手法では解法として DSTS を用い、2 つの階層からなる形式化を用いた手法では図 11 で示したように 2 つの階層について DSTS を用いた。しかし、明確な終了条件の無い DSTS では準最適解が 1 つに決定しない。そこで、DSTS に簡単なルールを加え、準最適解が 1 つに決まるように調整した。評価した手法は次のように表記する。

- STAV: 3.2.1 で説明した形式化を用いた手法
- LYR: 4 で説明した 2 つの階層からなる形式化を用いた提案手法
- LYR+: LYR におけるリーダー選出問題にのみ観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約を追加する手法
- LYR++: LYR における 2 つの階層に観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約を追加する手法

- LYR_co: LYR++をベースに観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} を評価するリーダーと、観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} を評価しないリーダーに分類する手法

評価を行うためにシミュレーションによる実験を行った．シミュレーションでは，各エージェントは次の処理を繰り返す．1回のこの処理を1サイクルとする．

1. 例題の設定に沿って観測対象の配置を変更する
2. 各センサはメッセージを受信して，局所処理を行う
3. 各センサは必要に応じてメッセージを他のエージェントに送信する

評価した項目は次のとおりである．

- シミュレータが全体の評価値を集計し，準最適解を得たことを検知するまでに要したサイクル数
- 得た準最適解の評価値
- 観測対象に割り当てることが可能なセンサの数
- 1つの観測対象に割り当てられたセンサの数
- 観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} の制約違反の個数

評価値は各観測対象に割り当てられているセンサの数によって評価したものである．観測対象 t_j に関する評価値 $eval_{t_j}$ は，式 (21) で計算される．式 (21) において， num_{t_j} は観測対象 t_j に割り当てられたセンサの数を示す．式 (21) の定数値は，後述の制約の重み準じて設定したものである．すなわち，評価値が小さくなる解の方が最適解に近い準最適解である．準最適解を得たとシミュレータが検知した時点で，すべての観測対象についての $eval_{t_j}$ を計算し，その合計値が準最適解の評価値となる．

$$eval_{t_j} = \begin{cases} 15 & n_j = 0 \\ 5 & n_j = 1 \\ 1 & n_j = 2 \\ 0 & otherwise \end{cases} \quad (21)$$

1000 サイクルを1試行とし，500 サイクルの時点で観測対象の1つの配置が変化する例題を評価に用いた．問題が変化した際に，問題が変化する前の問題での解を保持した状態で問題を解き始めることで動的な問題を模倣した．評価は問題変化前と問題変化後でそれぞれ行い，それぞれ500 サイクルで解を得られない場合は，その試行に

表 1: DSTS のパラメータ

	p_1	p_2	TABU 期間
STAV	0.8	0.4	1
LYR			
Layer1	0.9	0.3	2
Layer2	0.7	0.2	1

表 2: 制約の重み 1

STAV		LYR			
		リーダー選出		割り当て問題	
c^{ST0}	$w_0^{c^{ST0}} = 15$	c^{L0}	$w^{c^{L0}} = 10$	c^{A0}	$w_0^{c^{A0}} = 15$
	$w_1^{c^{ST0}} = 5$	c^{L1}	$w^{c^{L1}} = 200$		$w_1^{c^{A0}} = 5$
	$w_2^{c^{ST0}} = 1$	c^{L2}	$w^{c^{L2}} = 100$		$w_2^{c^{A0}} = 1$
c^{ST1}	$w^{c^{ST1}} = 200$	c^{L3}	$w^{c^{L3}} = 1$	c^{A1}	$w^{c^{A1}} = 200$
c^{ST2}	$w^{c^{ST2}} = 100$	c^{L4}	$w^{c^{L4}} = 1$		

表 3: 制約の重み 2

リーダー選出		割り当て問題	
c^{la0}	$w^{la0} = 200$	c^{aa0}	$w_0^{c^{aa0}} = 10$
c^{la1}	$w_0^{c^{la1}} = 5$		$w_1^{c^{aa0}} = 10$
	$w_1^{c^{la1}} = 0$		
	$w_2^{c^{la1}} = 100$		
	$w_3^{c^{la1}} = 200$		

おけるサイクル数を 500 サイクルとし，その他の評価にはその試行は含めない．評価には 20 個の例題を用意した．評価結果は全ての例題の結果の平均である．各例題の評価は 1000 回試行の平均を用いた．

6.1 DSTS のパラメータ

DSTS のパラメータを表 1 に示す．予備実験により，解の評価値が小さくなるような値をパラメータに設定した．また，各制約の重みを表 2 に示す．緩和不可能な制約の

重みは比較的大きくしてある．また，制約の数を考慮して，表2で示したように緩和不可能な制約の重みに違いをつけた．STAVと2つの階層からなる形式化を用いた手法における観測資源割り当て問題の観測対象に割り当てられたセンサの数に関する制約の重みは，観測対象に割り当てられたセンサの数が少なくなるほど大きくなるように設定した．

2つの階層からなる形式化を用いた手法におけるリーダー選出の緩和不可能な制約の重みは，各観測対象の割り当てることが可能なセンサの数の確保よりも，できるだけ全ての観測対象にリーダーを選出することを意図して，表に示す値を設定した．

観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約の重みを表3に示す．これら制約は緩和可能であるが，予備実験の結果，小さな値では効果がなかったため，表のように一部大きな値を用いた．

6.2 DSTSの調整点および変更点

本論文では，2つの階層からなる形式化を用いた提案手法とSTAVによる形式化を用いた手法が最善の結果を得るようにDSTSを調整した．また，先述のように解を1個に決定するため簡単なルールをDSTSに加えた．調整点と変更点を次に示す．

- 探索する解の範囲に関する調整

確率 p_1 に基づき値を変更する条件を各手法，及び各階層で異なる条件を用いる．提案手法におけるリーダー選出層では，“改善量 $\Delta > 0$ ”の場合に確率 p_1 に基づき値を変更する．STAVによる形式化を用いた手法と提案手法における観測資源割り当て問題解決層では“改善量 $\Delta \geq 0$ ”の場合に確率 p_1 に基づき値を変更する．この値の変更基準に関する違いは，比較的值域の広い問題では解を探索する範囲を広くすることにより，局所解から抜けやすくするためのものである．

- 短時間で準最適解を得るための変更点

確率 p_2 に基づき値を変更する条件“自身に制約違反が残っていた場合”を変更する．この変更点は提案手法の2つの階層とSTAVによる形式化を用いた手法の両方に適用する．変更後の条件は“自身の緩和不可能な制約に違反が残っていた場合”という条件である．この変更により，緩和不可能な制約を全て充足している場合，改善量が負である場合には値を変更しなくなる．よって，緩和不可能な制約を全て充足した解の範囲に素早く収束すると考えられる．

- 解を1つに決定するためのルール

確率 p_1 に基づきで値を変更する条件を“改善量 $\Delta \geq 0$ ”にした場合，解が1個に決

定しない．同じ評価値の変数値が複数存在していた場合，どちらかに決定する基準が DSTS には無いためである．そこで，STAV による形式化を用いた手法と提案手法における観測資源割り当て問題解決層において，緩和不可能な制約を全て充足していた場合は”改善量 $\Delta > 0$ ”の場合に確率 p_1 に基づき値を変更することにする．この様にすれば，緩和不可能な制約が全て充足しているとき，同じ評価値の解が存在していても，変数値が変更される事はなく解が 1 個に収束する．

6.3 変数の初期値

各センサは観測対象 t_j に関する変数を生成した時点で知っている観測対象 t_j に割り当てることが可能なセンサは自身のみである．そのため，STAV を用いた手法と提案手法におけるリーダー選出層の変数の初期値は，限られた情報に基づき決定する．変数の初期値を決定する方針は次の 2 つがある．

- 制約違反を起こす可能性が低い初期値

変数の初期値を解探索により決定する場合，比較的制約違反が起こる可能性は低くなる．観測対象 t_j に関する変数を生成した時点では知っている観測対象 t_j に割り当てることが可能なセンサは自身のみである．そのため，自身を観測対象 t_j に割り当てるか何も割り当てないかのうち，評価値が良い方を初期値として選択する．

例えば，センサ s_i は観測対象 t_j に関する変数を持っている状態で，観測対象 t_j に関する変数を生成したとする．また，観測対象 t_j にセンサ s_i が割り当てられていたとする．その状態で観測対象 t_j にセンサ s_i を割り当てたような値をとった場合，各センサは同時に複数の観測対象に割り当てられており，緩和不可能な制約が違反となる．その場合の評価値は大きいため，センサ s_i は解探索により，初期値としてセンサ s_i を割り当てない値をとる．

一方で，センサ s_i が観測対象 t_j に関する変数のみ持っている場合は，何らかのセンサを割り当てた方が評価値が良い．そのため，観測対象 t_j に関する変数の初期値として，センサ s_i を割り当てた値をとる．この場合，同じ観測対象に関する変数値が他のセンサと異なり，制約違反となる場合がある．

- 制約違反を起こす可能性が高い初期値

新しく観測可能になった変数の初期値として必ず自身を割り当てた値をとる場合，比較的制約違反が起こる可能性は高くなる．

例えば，センサ s_i は観測対象 $t_{j'}$ に関する変数を持っている状態で，観測対象 t_j に関する変数を生成したとする．また，観測対象 $t_{j'}$ にセンサ s_i が割り当てられていたとする．その状態であっても，必ず観測対象 t_j に関する変数の初期値としてセンサ s_i を観測対象 t_j に割り当てる値をとる．つまり，制約違反が生じる場合でも，制約違反が生じるような初期値をとる．このような初期値をとることで，制約違反が発生する変数が増加する．そのため，問題が変化した際に，最適化が行われる範囲が広がると考えられる．

一方で，センサ s_i が観測対象 t_j に関する変数のみ持っている場合は，制約違反を起こす可能性が低い初期値をとる場合と同様に制約違反が発生する可能性がある．

6.4 例題の生成方法

評価に使う例題は制約網が単一の連結成分からなる例題を用いた．この条件の元で観測対象数 n と観測対象の近傍数 t をパラメータとし， $n = 5, t = 2$ の例題を用意した．例題の作成手順は以下の通りである．

1. 1 個目の観測対象をランダムに選択した区画に配置する．
2. 1 番新しく配置した観測対象に隣接する区画のうち，観測対象がない区画を次の観測対象を配置する候補とする．
3. 候補のうち，そこに新しい観測対象を配置した場合，新しい観測対象に隣接する観測対象の数が t 以下になる区画を，最終的な候補とする．
4. 最終的な候補数が 0 であれば，初めからやりなおす．
5. 最終的な候補からランダムに 1 つ選び，新しい観測対象を配置する．
6. 2~5 を観測対象の数が n になるまで繰り返す．

また，各例題には制約網が単一の連結成分からなるという条件とパラメータが同じになるよう観測対象のうち 1 個の配置を変更した例題も用意した．観測対象の配置が変更された例題が問題変化後の問題となる．

6.5 ランダムに例題を作成し，制約違反が発生する可能性が低い変数の初期値を用いた場合の評価結果

例題を 6.4 で示した生成方法にしたがってランダムに生成した例題を用いて各手法を評価した．また，変数の初期値は，制約違反が発生する可能性が低い方針を全ての手法において用いた．

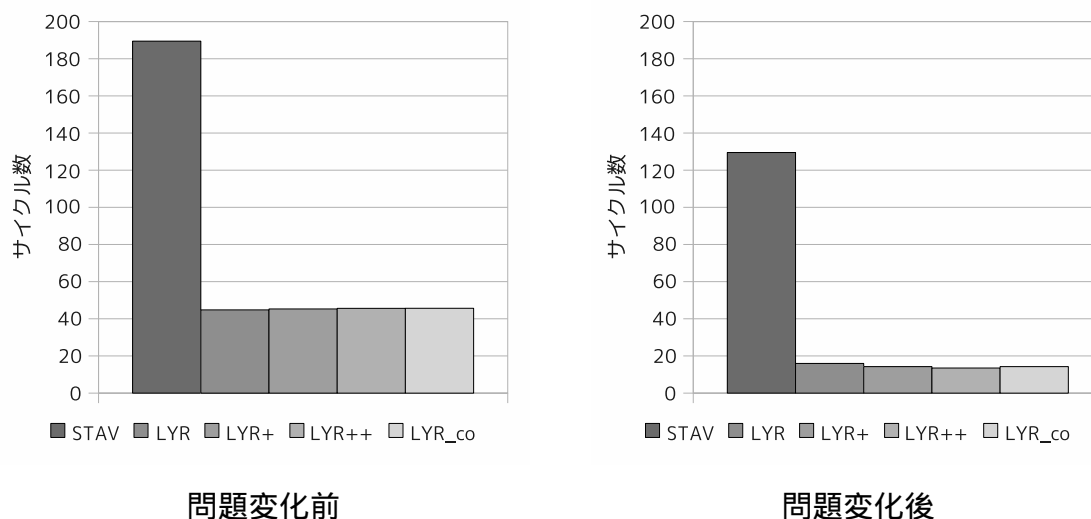


図 15: サイクル数の比較

6.5.1 準最適解を得るまでのサイクル数の比較

図 15 に準最適解を得るまでに要したサイクル数の評価を示す。提案手法では STAV を用いた手法と比較して少ないサイクル数で準最適解を得た。これは問題を 2 つの階層に分割したことで、問題の複雑さが緩和された結果であると考えられる。また、問題変化前と問題変化後を比較すると、問題変化後の方が少ないサイクル数で準最適解を得た。これは、部分的に解を得た状態、すなわち解探索がある程度進んでいる状態から解探索を始めるからであると考えられる。

提案手法においては、観測対象に割り当てられたセンサの組み合わせの変化を抑制する制約を追加した場合においても、準最適解を得るまでに要するサイクル数に関しては差がでていない。この結果は問題を簡略化されたグリッドモデルに限定しているため、観測対象に割り当てられたセンサの組み合わせの変化を抑制した場合も、観測対象に割り当てられたセンサの数に不均衡が生じなかったと考えられる。

6.5.2 準最適解の評価値の比較

図 16 に評価値に関する評価として準最適解の評価値を示す。この結果は、提案手法では準最適解を得るまでのサイクル数は削減できているが評価値に関しては大きい。このような結果になった理由は、提案手法は問題を 2 つの階層に分割したことで制約の緩和も 2 つの階層において行われているためである。リーダー選出層において制約を緩和した場合、リーダーが選出されない観測対象が存在するということが発生する可能性がある。リーダーが選出されないということは、観測対象に割り当てられるセンサの数は 0 ということになる。観測対象に割り当てられたセンサの数が 0 の時の評

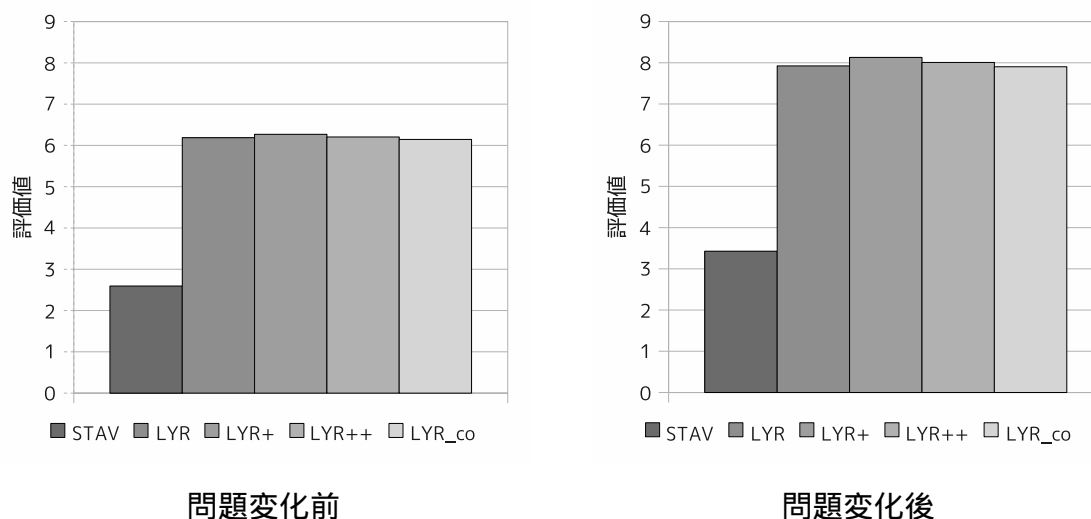


図 16: 評価値の比較

表 4: 各観測対象に割り当てることが可能なセンサの数 (問題変化前)

センサの数	0	1	2	3	4
STAV	0.00	0.00	0.00	0.00	5.00
LYR	0.16	0.00	0.04	1.15	3.65
LYR+	0.17	0.00	0.04	1.14	3.66
LYR++	0.16	0.00	0.04	1.14	3.66
LYR_co	0.16	0.00	0.04	1.15	3.66

価値は最も大きいため評価値に大きく影響している。また、各観測対象に割り当てられているセンサの数に不均衡が生じているのも、評価値が大きい原因である。この原因については6.5.3で詳細を述べる。

また、問題変化前と問題変化後を比較すると、問題変化後の方が評価値が大きい。これは、解探索が進んでいる状態から問題を解きはじめているため、局所解に陥るケースが増えたためであると考えられる。

6.5.3 各観測対象に割り当てることが可能なセンサの数

表 4.5 に各観測対象に割り当てることが可能なセンサの数を示す。もともとの問題では全ての観測対象は4個のセンサによって観測可能である。そのため、STAVを用いた手法では、全ての観測対象に対して4個の割り当てることが可能なセンサが存在する。一方、提案手法では、観測可能なセンサの数が4個である観測対象が減少して

表 5: 各観測対象に割り当てることが可能なセンサの数 (問題変化後)

センサの数	0	1	2	3	4
STAV	0.00	0.00	0.00	0.00	5.00
LYR	0.27	0.00	0.05	1.40	3.27
LYR+	0.29	0.00	0.05	1.31	3.35
LYR++	0.29	0.00	0.05	1.31	3.36
LYR_co	0.28	0.00	0.05	1.32	3.34

表 6: 各観測対象に割り当てられたセンサの数 (問題変化前)

センサの数	0	1	2	3	4
STAV	0	0	2.57	2.40	0.03
LYR	0.16	0.44	1.58	2.06	0.76
LYR+	0.17	0.44	1.57	2.06	0.76
LYR++	0.16	0.44	1.58	2.06	0.75
LYR_co	0.16	0.42	1.56	2.07	0.75

いる。これは、観測対象 t_j のリーダーは必ず観測対象 t_j に割り当てられ、他の観測対象に割り当てることが可能なセンサの集合から除外されるためである。つまり、リーダーの配置によっては、観測対象に割り当てることが可能なセンサの数に不均衡が生じる。また、リーダー選出層において制約が緩和され、リーダーが選出されないケースが生じる。その結果、割り当てることが可能なセンサの数が 0 である観測対象が発生している。

問題変化前と問題変化後を比較すると、問題変化後の方が 4 個のセンサを割り当て可能な観測対象数が減少している。これは、観測対象の配置が変化した場合、観測対象の配置が変化しない部分では、問題変化前の解を保持した状態で解探索を始めるため、リーダー選出層における局所解に陥るケース、つまり複数の観測対象に割り当てることが可能なセンサがリーダーとして選出されるケースが増加しているためであると考えられる。

6.5.4 観測対象に割り当てられたセンサの数に関する評価

表 6,7 に何個のセンサが各観測対象に割り当てられかを示す。提案手法では STAV を用いた手法と比較して、割り当てられたセンサの数が多い観測対象と少ない観測対象

表 7: 各観測対象に割り当てられたセンサの数 (問題変化後)

センサの数	0	1	2	3	4
STAV	0.01	0.10	2.70	2.15	0.04
LYR	0.27	0.45	1.60	2.04	0.64
LYR+	0.29	0.44	1.55	2.04	0.68
LYR++	0.29	0.41	1.65	1.99	0.66
LYR_co	0.29	0.42	1.56	2.14	0.60

に比較的偏っている。

これは、各観測対象に割り当てることが可能なセンサの数に不均衡が生じたためである。表 4,5 で示したように、提案手法では各観測対象に割り当てられるセンサの数に偏りが生じている。そのため、観測対象に割り当てられたセンサの数にも不均衡が生じ、割り当てられたセンサの数が少ない観測対象と多い観測対象が増加したといえる。

また、リーダーが選出されるタイミングはそれぞれ違うため、比較的早い段階で選出されたリーダーは割り当てることが可能なセンサが競合するリーダーが存在しない期間があると考えられる。その場合、早い段階で選出されたリーダーは貪欲にセンサを担当する観測対象に割り当てる。その結果、遅いタイミングで選出されたリーダーが多くセンサを担当する観測対象に割り当てるためには、既存のリーダーの制約違反を発生させる必要がある。そのため、比較的遅いタイミングで選出されたリーダーは、多くのセンサを担当する観測対象に割り当てづらくなる。これもまた、割り当て数に不均衡が生じている理由である。一方、いずれかの観測対象に割り当てられているセンサの数の合計を比較した場合、提案手法と STAV を用いた手法に大きな差はない。

6.5.5 観測対象に割り当てられたセンサの組み合わせの変化に関する評価

図 17 に観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} の違反の個数を示す。この結果より、2つの階層において観測対象に割り当てられたセンサの変化を抑制するための制約を追加した LYR++ では、違反の個数が少く観測対象に割り当てられたセンサの組み合わせの変化を抑制しているといえる。

また、リーダー選出層にのみ観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約を追加した LYR+ と観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約を追加しない LYR を比較した場合、LYR+ の方が違反の個数が少くなっている。これは、LYR+ ではリーダーに選出されたセンサの変化

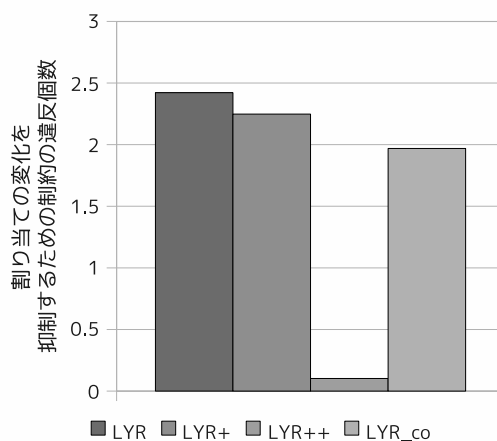


図 17: 観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} の違反の個数

を抑制する作用が働くためである。問題変化後のリーダーであるセンサが問題変化前と同じであるか、同じグループからリーダーが選出された場合、Inside_group であったセンサの少なくとも 1 個は、同じ観測対象に割り当てられることになる。そのため、リーダー選出層にのみ割り当ての変化を抑制するための制約を追加した LYR+ も観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} の違反の個数が LYR と比較して少くなる。

リーダーを観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} を評価するリーダーと評価しないリーダーに分類する LYR_co に関しては、LYR と LYR+ と比較して、違反の個数は少ない。この結果は、LYR_co でも、観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} を評価するリーダーは存在し、観測対象に割り当てられたセンサの組み合わせの変化を抑制する働きが部分的に働くためである。

6.6 意図的に複雑な例題を作成し、制約違反が発生する可能性が低い変数の初期値を用いた場合の評価結果

6.4 で示した生成方法に従って問題変化前の例題をランダムに生成し、問題変化後の例題は意図的に問題の複雑さが増すように生成した例題を用いて各手法を評価した。複雑な問題とは、1 個のセンサが持つ変数の数の平均が多い問題である。また、変数の初期値は、制約違反が発生する可能性が低い方針を全ての手法において用いた。

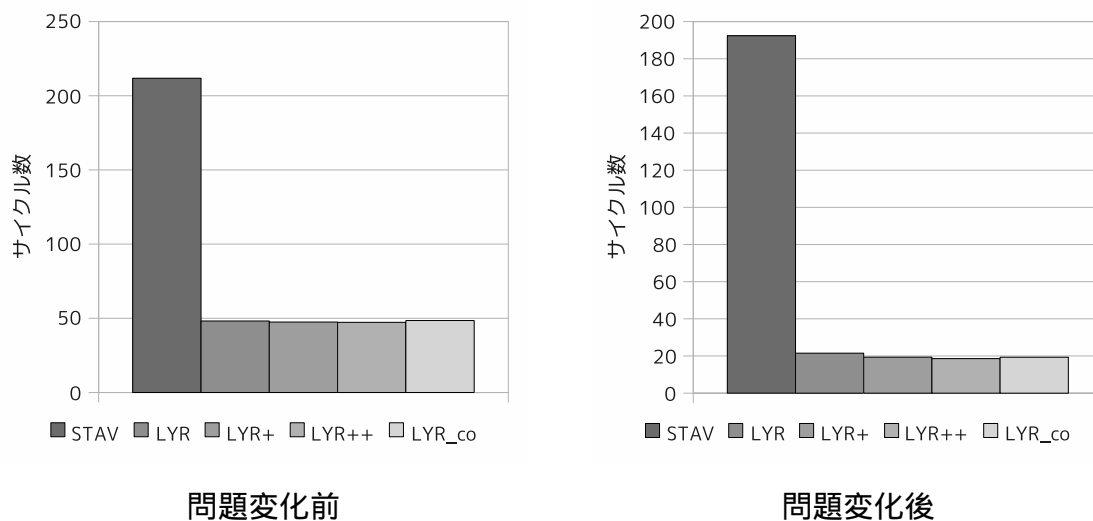


図 18: サイクル数の比較

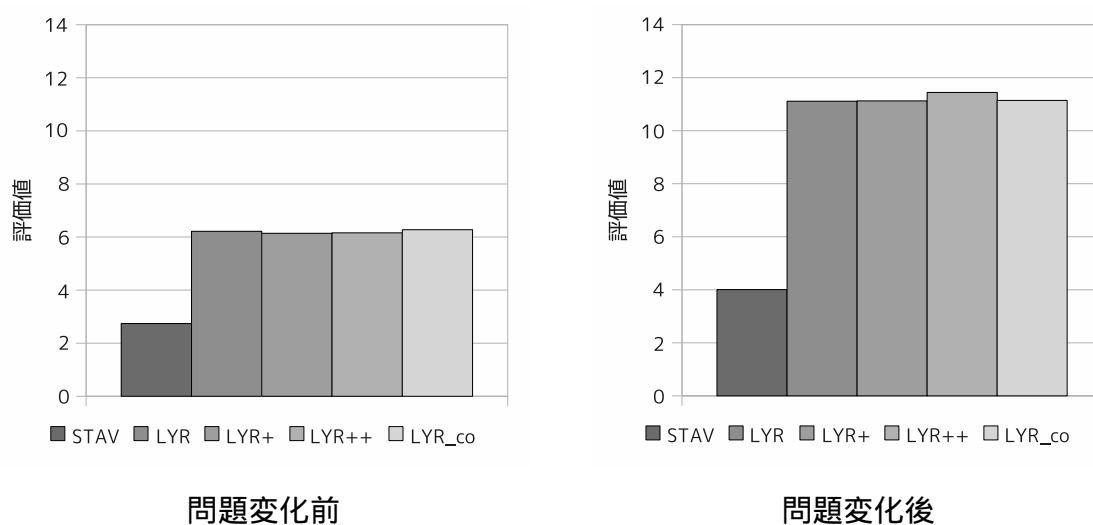


図 19: 評価値の比較

6.6.1 準最適解を得るまでのサイクル数の比較

図 18 に準最適解を得るまでに要したサイクル数の評価を示す．6.5 と同様に提案手法では STAV を用いた手法と比較して少ないサイクル数で準最適解を得た．また，問題変化前と比較して，問題変化後では比較的少ないサイクル数で準最適解を得た．

6.5 での結果と比較すると，総じて準最適解を得るまでのサイクル数が増加している．これは意図的に複雑な例題を生成したため，解いている問題が複雑であるためである．

6.6.2 準最適解の評価値の比較

図 19 に評価値に関する評価として準最適解の評価値を示す．6.5 で示した結果と同様，提案手法では準最適解を得るまでのサイクル数は削減できているが評価値に関しては大きい．また，問題変化前と比較すると，問題変化後の方が準最適解の評価値が大きい．

複雑な例題を用いて評価した結果では，観測対象に割り当てられたセンサの組み合わせの変化を抑制する制約について評価する LYR++ の評価値が大きい．これは，観測対象に割り当てられたセンサの組み合わせの変化を抑制するため，他の手法に比べ観測対象に割り当てられたセンサの数に不均衡が生じたためである．複雑な問題では，リーダー選出層において，1 個のセンサが持つ変数の数が多い．つまり，1 個のセンサが観測可能な観測対象が多くなる．これは，あるセンサがリーダーに選出された場合，観測対象 t_j のリーダーとなったセンサは観測対象 t_j 以外の観測対象の割り当てることが可能なセンサの集合から除外されるケースが増加する原因である．そのため，Outside_group であるセンサを割り当てると制約違反になる LYR++ では制約違反なしで割り当てられるセンサの数が少なくなる．そのため，観測対象に割り当てられたセンサの数に不均衡が生じ，LYR++ の評価値が大きい値となった．

一方，LYR++ をベースに観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} を評価するリーダーと，観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} を評価しないリーダーに分類する手法である LYR_co では，評価値は LYR++ と比較して小さい．これは，観測対象に割り当てたセンサの数が過剰なリーダーと不十分なリーダーは，観測対象に割り当てられたセンサの組み合わせの変化を抑制する制約 c^{aa0} に関しては評価しないためである．観測対象に割り当てられたセンサの組み合わせの変化を抑制する制約 c^{aa0} に関して評価しない観測対象 t_j に関するリーダーは Inside_group であるセンサを観測対象 t_j 以外の観測対象に割り当てられたり，Outside_group であるセンサを観測対象 t_j に割り当てたとしても制約違反ではない．そのため，LYR_co では必要な箇所では観測対象に割り当てられたセンサの数に関する最適化が行われ，観測対象に割り当てられたセンサの数の不均衡を解消することが出来る．そのため，LYR_co の評価値は LYR++ と比較して小さい値になっている．

6.6.3 各観測対象に割り当てるとが可能なセンサの数

表 8,9 に各観測対象に割り当てることが可能なセンサの数を示す．6.5 で示した結果と比較して，提案手法では割り当て可能なセンサの数が 4 個である観測対象が減少し

表 8: 各観測対象に割り当てることが可能なセンサの数 (問題変化前)

センサの数	0	1	2	3	4
STAV	0.00	0.00	0.00	0.00	5.00
LYR	0.16	0.00	0.04	1.33	3.47
LYR+	0.15	0.00	0.04	1.34	3.47
LYR++	0.15	0.00	0.04	1.34	3.47
LYR_co	0.16	0.00	0.04	1.33	3.47

表 9: 各観測対象に割り当てることが可能なセンサの数 (問題変化後)

センサの数	0	1	2	3	4
STAV	0.00	0.00	0.00	0.00	5.00
LYR	0.55	0.00	0.09	1.22	3.14
LYR+	0.55	0.00	0.10	1.27	3.08
LYR++	0.55	0.00	0.10	1.26	3.08
LYR_co	0.56	0.00	0.09	1.25	3.09

ている。これは、1個のセンサあたりが持つ変数の数が増加しているため、あるセンサがリーダーに選出された場合に、割り当てることが可能なセンサの集合からリーダーであるセンサが除外されるケースが増加したためである。この原因は問題変化後の結果においても同様のことがいえる。

またリーダー選出層においてリーダーの変化を抑制する手法と LYR を比較した場合、割り当てることが可能なセンサの数が4個である観測対象がリーダー選出層においてリーダーの変化を抑制する手法の方が少ない。そのため、割り当て問題解決層で観測対象に割り当てられたセンサの組み合わせの変化を抑制する手法である LYR++では、割り当て問題解決層において観測対象に割り当てられたセンサの数に関して最適化が行われにくいため、割り当て数の不均衡が解消できず、評価値が大きいといえる。

6.6.4 観測対象に割り当てられたセンサの数に関する評価

表 10,11 に何個のセンサが各観測対象に割り当てられたかを示す。6.5 で示した結果と同様で、提案手法では評価値の小さい対象と大きい対象に比較的偏っている。また、表 11 をより、観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約を評価する LYR++では、より割り当てられたセンサの数が少ない観測対象が多い

表 10: 各観測対象に割り当てられたセンサの数 (問題変化前)

センサの数	0	1	2	3	4
STAV	0.00	0.00	2.71	2.26	0.03
LYR	0.16	0.45	1.67	2.05	0.67
LYR+	0.15	0.45	1.68	2.05	0.67
LYR++	0.15	0.45	1.67	2.06	0.67
LYR_co	0.16	0.45	1.67	2.05	0.67

表 11: 各観測対象に割り当てられたセンサの数 (問題変化後)

センサの数	0	1	2	3	4
STAV	0.06	0.06	2.85	2.01	0.02
LYR	0.55	0.28	1.51	2.06	0.60
LYR+	0.55	0.27	1.53	2.06	0.59
LYR++	0.55	0.33	1.52	2.04	0.56
LYR_co	0.56	0.24	1.49	2.23	0.47

ことが確認できる．それ以外の LYR では観測対象に割り当てられたセンサの数に関して再度最適化が行われやすいため，割り当て数の少ない観測対象への偏りが小さいといえる．

6.6.5 観測対象に割り当てられたセンサの組み合わせの変化に関する評価

図 20 に観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} の違反の個数を示す．6.5 で示した結果と同様で，観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} を評価する LYR++ では制約 c^{aa0} の違反の個数が少く，観測対象に割り当てられたセンサの組み合わせの変化が抑制されているといえる．観測対象に割り当てられたセンサの組み合わせの変化を抑制する制約 c^{aaa} を評価するリーダーと評価しないリーダーに分類する LYR_co では，LYR 及び LYR+ よりも違反の個数が少い．一方で，評価値に関しては LYR や LYR+ と同程度になっている．評価値との結果と統合し，LYR_co では，リーダーを 2 種類に分類し部分的に観測対象に割り当てられたセンサの組み合わせの変化を抑制しつつ必要な部分で観測対象に割り当てられたセンサの数に関して最適化が行われる期待通りの効果が得られているといえる．

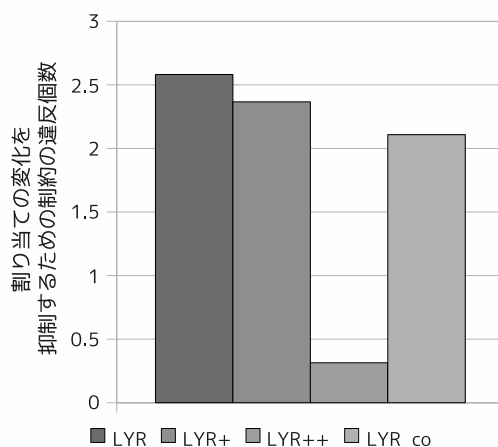


図 20: 観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} の違反の個数

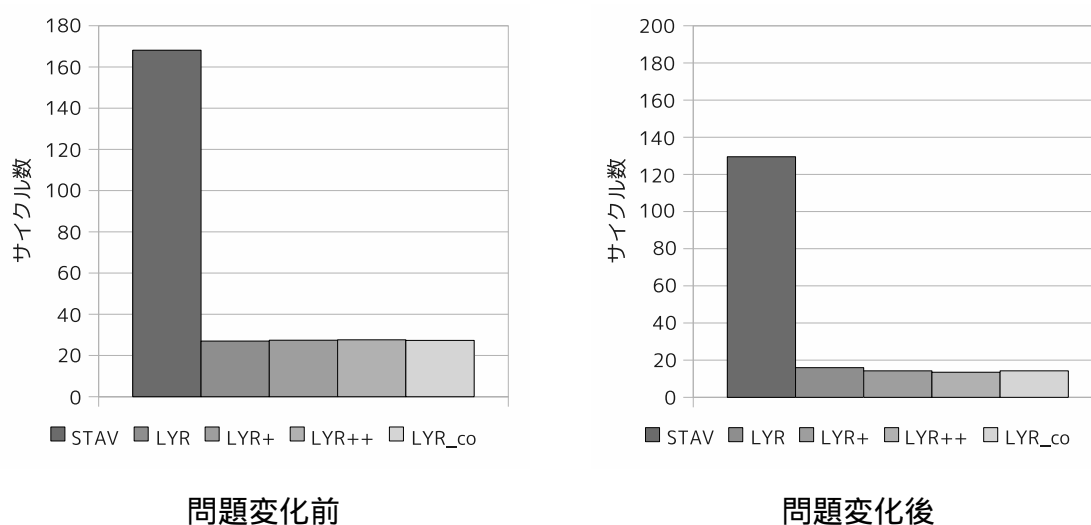


図 21: サイクル数の比較

6.7 ランダムで例題を作成し、制約違反が発生する可能性が高い変数の初期値を用いた場合の評価結果

例題を 6.4 で示した生成方法にしたがって、ランダムに生成した例題を用いて各手法を評価した。また、変数の初期値は、制約違反が発生する可能性が高い方針を全ての手法において用いた。

6.7.1 準最適解を得るまでのサイクル数の比較

図 21 に準最適解を得るまでに要したサイクル数の評価を示す。この結果は、制約違反が発生する可能性が低い初期値を用いた場合と同様で、提案手法では少ないサイク

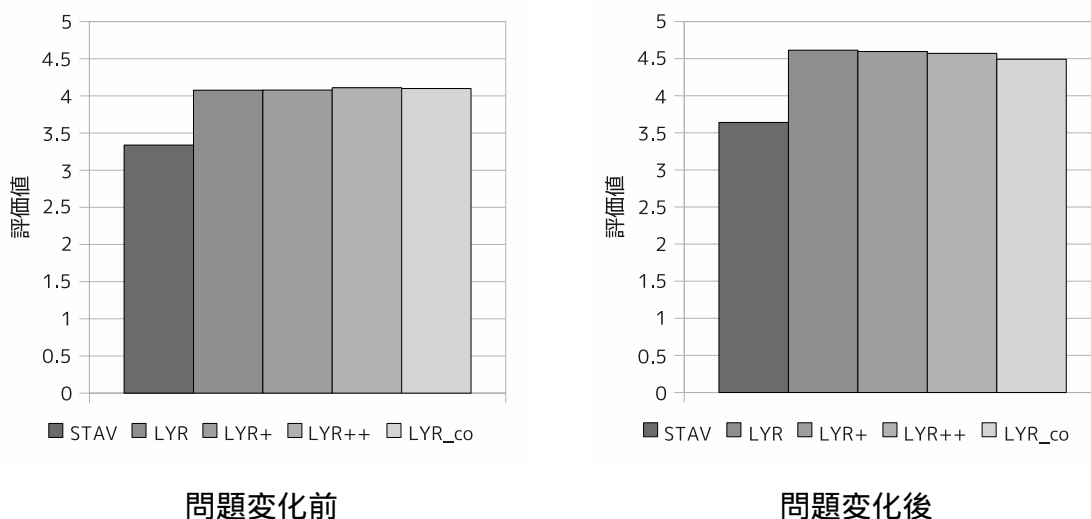


図 22: 評価値の比較

ル数で準最適解を得たことを示している。これは問題を2つの階層に分割した事で、問題の複雑さが緩和された結果であると考えられる。また、部分的に解探索がある程度進んでいる状態から解探索をはじめている結果、問題変化後の方が解を得るまでに要するサイクル数を削減できている。

6.7.2 準最適解の評価値の比較

図 22 に評価値に関する評価として準最適解の評価値を示している。この結果は、制約違反が発生する可能性が低い初期値を用いた場合と同様で、提案手法では準最適解を得るまでのサイクル数は削減できているが評価値に関しては大きいことを示している。

6.5 で示した変数の初期値を制約違反が発生する可能性が低い値を用いた場合と比較すると、評価値は小さいことが確認できる。この結果の原因として2つの理由が考えられる。1つは制約違反が発生し、最適化が行われる範囲が広がっているためである。最適化が行われる変数が多いため、より最適解に近い準最適解を得たと考えられる。もう1つは、変数の初期値として、必ず自身を割り当てるような値をとるためである。そのため、リーダー選出層においてリーダーが選出されないケースが減少し、評価値が小さい値を示し。

6.7.3 各観測対象に割り当てることが可能なセンサの数

表 12,13 に各センサに割り当てることが可能なセンサの数を示す。制約違反が発生する可能性が低い初期値を用いた場合と同様で、提案手法では観測可能なセンサが4個ある観測対象が減少している。これは、観測対象 t_j のリーダーは必ず観測対象 t_j に割り当てられ、他の観測対象に割り当て可能なセンサの集合から除外されるためである。

表 12: 各観測対象に割り当てることが可能なセンサの数 (問題変化前)

センサの数	0	1	2	3	4
STAV	0.00	0.00	0.00	0.00	5.00
LYR	0.00	0.00	0.05	1.26	3.69
LYR+	0.00	0.00	0.05	1.26	3.70
LYR++	0.00	0.00	0.05	1.28	3.68
LYR_co	0.00	0.00	0.05	1.27	3.69

表 13: 各観測対象に割り当てることが可能なセンサの数 (問題変化後)

センサの数	0	1	2	3	4
STAV	0.00	0.00	0.00	0.00	5.00
LYR	0.01	0.05	0.07	1.57	3.30
LYR+	0.01	0.05	0.07	1.50	3.37
LYR++	0.01	0.05	0.06	1.52	3.35
LYR_co	0.01	0.05	0.06	1.51	3.36

しかし、制約違反が発生する可能性が低い初期値を用いた場合と比較して、観測対象に割り当てることが可能なセンサの数が少ない観測対象は減少している。また、観測対象に対してリーダーが選出されないケースも減少している。これは、初期値として制約違反が発生する可能性が高いため、リーダー選出において最適化が行われる変数が多くなるためである。そのため、リーダーの配置が最適解に近くなり、割り当てることが可能なセンサの数が減少する観測対象数が減少したといえる。また、必ず割り当てるとような初期値をとるため、リーダーが選出されない観測対象が減少する。そのため、割り当てることが可能なセンサの数が0である観測対象数は減少している。

6.7.4 観測対象に割り当てられたセンサの数に関する評価

表 14,15 に何個のセンサが各観測対象に割り当てられかを示す。提案手法ではSTAVを用いた手法と比較して、評価値の小さい対象と大きい観測対象に比較的偏っている。制約違反が発生する可能性が低い初期値を用いた場合と同様の傾向である。

しかし、制約違反が発生する可能性が低い初期値を用いた場合と比較して、リーダーが選出されない観測対象が減少しているため、割り当てられたセンサの数が0である観測対象の数は大きく減少している。そのため、観測対象に割り当てられたセンサの

表 14: 各観測対象に割り当てられたセンサの数 (問題変化前)

センサの数	0	1	2	3	4
STAV	0.00	0.20	2.32	2.43	0.04
LYR	0.00	0.48	1.69	2.14	0.70
LYR+	0.00	0.48	1.69	2.14	0.70
LYR++	0.00	0.48	1.68	2.13	0.70
LYR_co	0.00	0.48	1.67	2.15	0.70

表 15: 各観測対象に割り当てられたセンサの数 (問題変化後)

センサの数	0	1	2	3	4
STAV	0.00	0.21	2.58	2.17	0.04
LYR	0.01	0.55	1.75	2.12	0.57
LYR+	0.01	0.54	1.71	2.15	0.56
LYR++	0.01	0.52	1.80	2.07	0.60
LYR_co	0.01	0.52	1.71	2.23	0.53

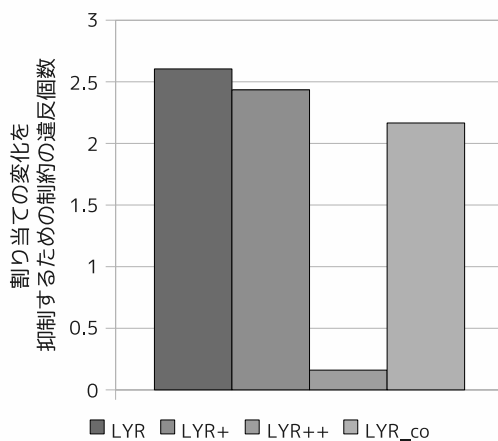


図 23: 観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} の違反の個数

数の合計は、提案手法と STAV を用いた手法を比較すると、ほぼ差がない。

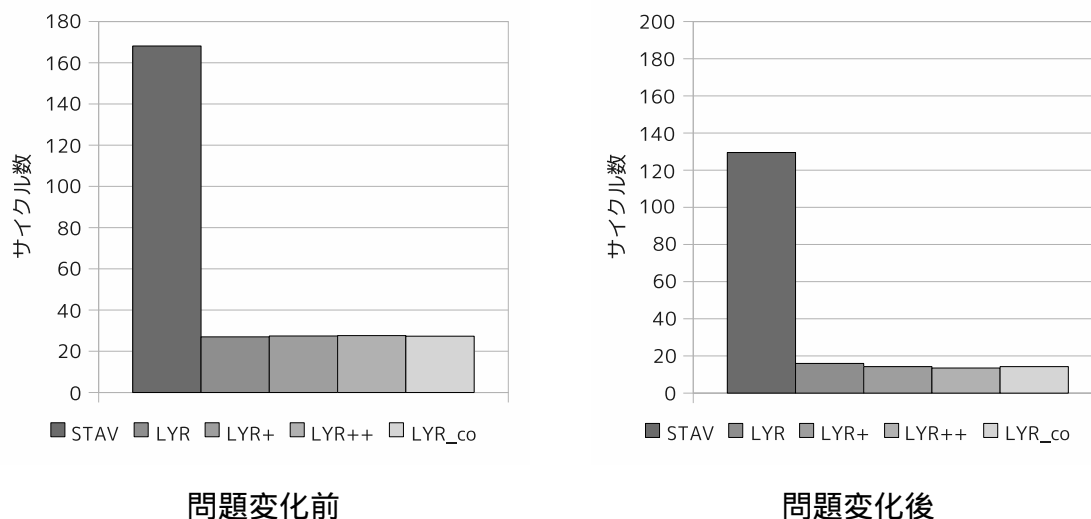


図 24: サイクル数の比較

6.7.5 観測対象に割り当てられたセンサの組み合わせの変化に関する評価

図 23 に観測対象に割り当てられたセンサの組み合わせの変化を抑制するため制約 c^{aa0} の違反の個数を示す．制約違反が発生する可能性が低い初期値を用いた場合と同様で，観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約を追加した LYR++ では，違反の個数が少く観測対象に割り当てられたセンサの組み合わせの変化を抑制しているといえる．また，リーダー選出層にのみ観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約を追加した LYR+ と観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約を追加しない LYR を比較した場合，LYR+ の方が違反の個数が少くなっている．また，リーダーを観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} を評価するリーダーと評価しないリーダーに分類する LYR_co は，LYR と LYR+ と比較して，違反の個数は少ない．

6.8 意図的に複雑な例題を作成し，制約違反が発生する可能性が高い変数の初期値を用いた場合の評価結果

6.4 で示した生成方法に従って，問題変化前の例題をランダムに生成し，問題変化後の例題は意図的に問題の複雑さが増すように生成した例題を用いて各手法を評価した．複雑な問題とは，1 個のセンサが持つ変数の数の平均が多い問題である．また，変数の初期値は，制約違反が発生する可能性が高い方針を全ての手法において用いた．

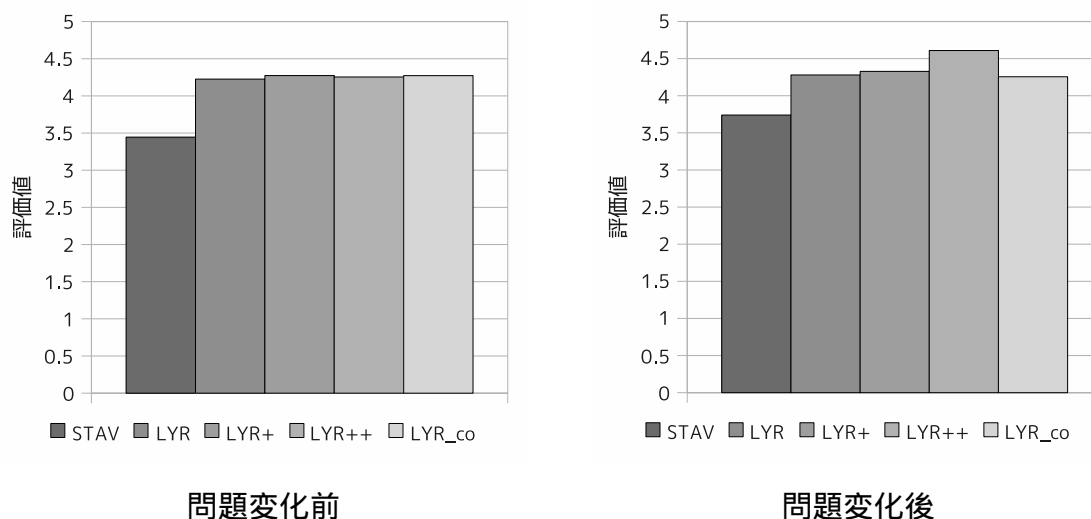


図 25: 評価値の比較

6.8.1 準最適解を得るまでのサイクル数の比較

図 24 に準最適解を得るまでに要したサイクル数の評価を示す。制約違反が発生する可能性が低い初期値を用いた場合及び 6.7 で示した結果と同様で提案手法では少ないサイクル数で準最適解を得た。また、問題変化前と比較して、問題変化後では比較的少ないサイクル数で準最適解を得ている。

6.7 での結果と比較すると、総じて準最適解を得るまでのサイクル数が増加している。これは意図的に複雑な例題を生成したため、解いている例題が複雑であるためである。

6.8.2 準最適解の評価値の比較

図 25 に評価値に関する評価として準最適解の評価値を示している。制約違反が発生する可能性が低い初期値を用いた場合及び 6.7 で示した結果と同様で、提案手法では準最適解を得るまでのサイクル数は削減できているが評価値に関しては大きいことを示している。また、問題変化前と比較すると、問題変化後の方が準最適解の評価値が大きい。

観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} を評価する LYR++ では評価値が他の提案手法と比較して大きく、その差は制約違反が発生する可能性が低い初期値を用いた場合よりも顕著に現れている。

また 6.7 での結果と同様で、制約違反が発生する可能性が低い初期値を用いた場合と比較すると、評価値は小さい。

表 16: 各観測対象に割り当てることが可能なセンサの数 (問題変化前)

センサの数	0	1	2	3	4
STAV	0.00	0.00	0.00	0.00	5.00
LYR	0.00	0.00	0.05	1.47	3.48
LYR+	0.00	0.00	0.05	1.45	3.50
LYR++	0.00	0.00	0.06	1.46	3.49
LYR_co	0.00	0.00	0.06	1.46	3.48

表 17: 各観測対象に割り当てることが可能なセンサの数 (問題変化後)

センサの数	0	1	2	3	4
STAV	0.00	0.00	0.00	0.00	5.00
LYR	0.01	0.05	0.16	1.66	3.12
LYR+	0.01	0.05	0.17	1.71	3.05
LYR++	0.01	0.05	0.17	1.71	3.05
LYR_co	0.01	0.05	0.18	1.72	3.04

6.8.3 各観測対象に割り当てることが可能なセンサの数

表 16,17 に各センサに割り当てることが可能なセンサの数を示す。6.7 で示した結果と比較して、提案手法では割り当てることが可能なセンサの数が 4 個である観測対象が減少している。これは、1 個のセンサあたりが持つ変数の数が増加しているため、あるセンサがリーダーに選出された場合に、割り当てることが可能なセンサの集合からリーダーであるセンサが除外されるケースが増加したためである。

またリーダー選出層においてリーダーの変化を抑制する手法と LYR を比較した場合、割り当てることが可能なセンサの数が 4 個である観測対象がリーダー選出層においてリーダーの変化を抑制する手法の方が少い。そのため、割り当て問題解決層で観測対象に割り当てられたセンサの組み合わせの変化を抑制する手法である LYR++ では、割り当て数の不均衡が解消できず、評価値が大きい。

6.8.4 観測対象に割り当てられたセンサの数に関する評価

表 18,19 に何個のセンサが各観測対象に割り当てられかを示す。6.7 で示した結果と同様で、提案手法では評価値の小さい対象と大きい対象に比較的偏っている。また、観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} を評価

表 18: 各観測対象に割り当てられたセンサの数 (問題変化前)

センサの数	0	1	2	3	4
STAV	0.00	0.20	2.45	2.32	0.03
LYR	0.00	0.49	1.78	2.11	0.62
LYR+	0.00	0.50	1.77	2.10	0.63
LYR++	0.00	0.49	1.78	2.10	0.63
LYR_co	0.00	0.50	1.77	2.10	0.63

表 19: 各観測対象に割り当てられたセンサの数 (問題変化後)

センサの数	0	1	2	3	4
STAV	0.00	0.19	2.77	2.02	0.02
LYR	0.01	0.45	1.96	2.16	0.43
LYR+	0.01	0.44	1.96	2.17	0.42
LYR++	0.01	0.51	1.91	2.11	0.46
LYR_co	0.01	0.43	1.93	2.29	0.34

する LYR++ では、他の提案手法と比較して、観測対象に割り当てられたセンサの数が少ない観測対象が増加している。これは、観測対象に割り当てられたセンサの数に不均衡が生じ、さらに観測対象に割り当てられたセンサの組み合わせの変化を抑制したため、観測対象に割り当てられたセンサの数の不均衡を解消できなかったためである。リーダーを 2 種類に分類する LYR_co ではセンサが 3 個よりも多く割り当てられた観測対象が減少し、3 個割り当てられた観測対象が増加している。これは、観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} を評価しないリーダーである条件に担当する観測対象に割り当てられたセンサの数が 3 個よりも多いという条件があるためである。この条件により、過剰にセンサを割り当てた観測対象から割り当てられていたセンサの一部が他の観測対象に分配されたといえる。一方、6.7 と同様で、制約違反が発生する可能性が低い初期値を用いた場合と比較して、観測対象に割り当てられたセンサの数の合計は提案手法と STAV を用いた手法とを比較して差は小さい。

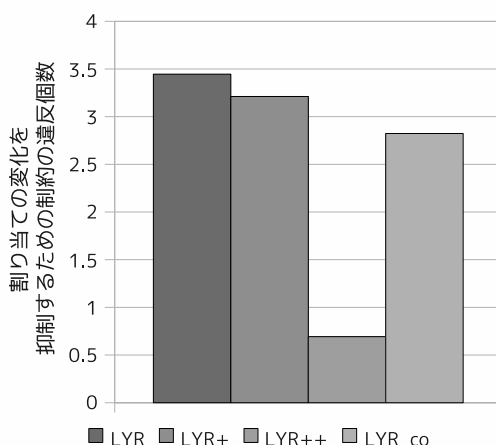


図 26: 観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} の違反の個数

6.8.5 観測対象に割り当てられたセンサの組み合わせの変化に関する評価

図 23 に観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} の違反の個数を示す．制約違反が発生する可能性が低い初期値を用いた場合及び 6.7 で示した結果と同様，観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} を評価する LYR++ では，違反の個数が少く観測対象に割り当てられたセンサの組み合わせの変化が抑制されているといえる．また，観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} を評価するリーダーと評価しないリーダーに分類する LYR_co では，LYR 及び LYR+ よりも違反の個数が少い．

一方，制約違反が発生する可能性が低い初期値を用いた場合及び 6.7 で示した結果と比較して観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約 c^{aa0} の違反の個数は全体的に増加している．この原因として，2つの理由が考えられる．1つは問題が複雑な問題のため，複数の観測対象に割り当てることが可能なセンサの数が多く，観測対象に割り当てられたセンサの組み合わせの変化を抑制しない場合は，より観測対象に割り当てられたセンサの組み合わせの変化が発生しやすいためである．もう1つは，制約違反が発生し問題変化後に再度最適化が行われる変数が増加するため，観測対象に割り当てられたセンサの組み合わせに変化が生じやすくなっているためである．

6.9 総括

観測対象の配置が変化する動的な問題において、どの手法も問題変化後の方が解を得るまでの平均サイクル数が少くなっている。これは、部分的に解探索が進んでいる状態から問題を解きはじめているためであると考えられる。一方、評価値は問題変化後の方が大きい。この原因は、解探索が進んでいる状態から問題を解きはじめているため、局所解に陥り易くなっているためであると考えられる。

2つの階層からなる形式化を用いた提案手法では準最適解を得るまでのサイクル数は大きく削減できているが、解の質に関しては各観測対象に割り当てられたセンサの数に不均衡が生じている。しかし、短時間で解を得ることが必要であるリアルタイム性を重視するようなシステムでは、この程度のサイクル数に対する解の質のトレードオフを許容出来るようなケースがあると考えられる。このような解の質の悪化も、変数の初期値として制約違反が発生する可能性が高い値を取ることによって、改善することが出来る。また、動的な問題において観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約についても効果が得られた。特にLYR_coでは観測対象に割り当てられたセンサの組み合わせの変化を抑制しつつ準最適解を得るまでのサイクル数及び評価値の増加を抑制している。

7 まとめ

本論文では、複数の自律的なセンサによる観測システムを対象として、その観測資源割り当て問題に分散制約最適化問題の枠組を適用した。また、確率的な反復改善型アルゴリズムであるDSTSを観測資源割り当て問題に適用した。特に、問題をリーダー選出層と割り当て問題解決層の2つの比較的簡単な問題の階層に分割し、解探索時間を抑制する手法を提案した。

提案手法におけるリーダー選出層では各観測対象に対して1個のリーダーを選出する。そして、選出されたリーダーは、観測資源割り当て問題解決層で各観測対象に割り当てるセンサの組み合わせを決定する。観測資源割り当て問題解決層ではリーダー選出層の解に基づき問題を解く必要がある。そのため、各階層の処理系の間では部分的なメッセージ交換を行う。また、観測対象の配置が変化した後、観測対象に割り当てられたセンサの組み合わせの変化を抑制するために、センサの割り当ての状態を考慮し、その状態を交換する前処理を導入した。また、そのセンサの割り当ての状態を利用し、観測対象に割り当てられたセンサの組み合わせの変化を抑制するための制約を導入した。提案手法により、準最適解を得るまでに要するサイクル数を削減でき

ることを確認した。また、観測対象の配置が変化した後、観測対象に割り当てられたセンサの組み合わせの変化を少なく抑える効果も得られることを確認した。一方で、提案手法では観測対象に割り当てられたセンサの数に不均衡が生じている。しかし、変数の初期値として制約違反が発生する可能性が高い値をとることにより、観測対象に割り当てられたセンサの数の不均衡を改善することができることを確認した。また、観測対象に割り当てられたセンサの組み合わせの変化を抑制するセンサを限定することで、観測対象に割り当てられたセンサの数の不均衡を改善するモデルを提案し、部分的に観測対象に割り当てられたセンサの組み合わせの変化を抑制しつつ観測対象に割り当てられたセンサの数の不均衡を改善することができることを確認した。これらの結果により、準最適解を得るまでのサイクル数の削減のためのトレードオフを小さく抑えることができた。

本論文では、分散センサ網における観測資源割り当て問題の表現としてグリッドモデルを用いた。このような例題に基づく検討は、分散センサ網において本質的な問題とその解法を明らかにする上で、一定の重要性があると考えられる。より一般的なセンサ網のモデルへの適用、実システムにおける検証は今後の課題である。

謝辞

本研究のために多大な御尽力を頂き、日頃から熱心な御指導を賜った名古屋工業大学の松尾啓志教授、津邑公暁准教授、齋藤彰一准教授、松井俊浩助教に深く感謝致します。また、本研究の際に多くの助言、協力をして頂いた松尾・津邑研究室、齋藤研究室の皆様にも深く感謝致します。

参考文献

- [1] Modi, P. J., Shen, W.-M., Tambe, M. and Yokoo, M.: ADOPT: Asynchronous Distributed Constraint Optimization with Quality Guarantees, *Artificial Intelligence*, Vol. 161, pp. 149–180 (2005).
- [2] Petcu, A. and Faltings, B.: A Scalable Method for Multiagent Constraint Optimization, *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 266–271 (2005).
- [3] Zhang, W., Wang, O. and Wittenburg, L.: Distributed stochastic search for constraint satisfaction and optimization: Parallelism, phase transitions and performance, *AAAI Workshop on Probabilistic Approaches in Search*, pp. 53–59 (2002).

- [4] 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, 電子情報通信学会論文誌, Vol. J90-D, No. 9, pp. 2302–2313 (2007).
- [5] Modi, P. J. and Ali, S. M.: Distributed Constraint Reasoning under Unreliable Communication, *In Proceedings of Distributed Constraint Reasoning Workshop at Second International Joint Conference on Autonomous Agents and MultiAgent Systems* (2003).
- [6] Bejar, R., Domshilak, C., Fernandez, C., Gomes, C., Krishnamachari, B., Selman, B. and Valls, M.: Sensor networks and distributed CSP, *Artificial Intelligence*, Vol. 161, pp. 117–147 (2005).
- [7] 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, 情報処理学会論文誌コンピュータビジョンとイメージメディア, Vol. 43, pp. 64–79 (2002).
- [8] 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, 電気情報通信学会論文誌, Vol. J88-D-i, No. 9, pp. 1438–1447 (2005).
- [9] Ukita, N.: Real-time Dense Communication among Agents for Active Tracking, *4th International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pp. 1335–1336 (2005).
- [10] Modi, P. J., Shen, W.-M., Tambe, M. and Yokoo, M.: An Asynchronous Complete Method for Distributed Constraint Optimization, *2nd International joint Conference on Autonomous Agents and Multiagent Systems*, pp. 161–168 (2003).
- [11] Ali, S., Koenig, S. and Tambe, M.: Preprocessing techniques for accelerating the DCOP algorithm ADOPT, *4th International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 1041–1048 (2005).
- [12] 横尾真, 平山勝敏: 分散 breakout : 反復改善型分散制約充足アルゴリズム (特集, 並列処理), 情報処理学会論文誌, Vol. 39, No. 6, pp. 1889–1897 (1998-06-15).