

卒業研究論文

分散制約充足/最適化による
分散センサ網の資源割当て問題の検討

指導教員 松尾 啓志 教授
津邑 公暁 准教授

名古屋工業大学 工学部 情報工学科
平成 17 年度入学 17115036 番

太田 和宏

平成 21 年 2 月 10 日

分散制約充足/最適化による 分散センサ網の資源割当て問題の検討

目次

1	はじめに	1
2	分散制約充足/最適化問題	3
2.1	概説	3
2.2	形式化	3
2.3	解法	4
2.3.1	DSA : Distributed Stochastic Search	4
2.3.2	DSTS : Distributed Stochastic Tabu Search	4
3	分散センサ網の資源割当て問題	6
3.1	概要	6
3.2	グリッドモデル	6
3.3	制約の分類	7
4	従来手法	8
4.1	STAV(Sensor-Target As Variable)	8
4.1.1	STAV おける制約	8
4.2	TAV(Target As Variable)	9
4.2.1	TAV における制約	10
4.3	STAV と TAV の問題点	10
4.3.1	STAV	10
4.3.2	TAV	11
4.4	エージェントモデル	11
5	提案手法	13
5.1	基本構造	13
5.2	リーダー選出	13
5.2.1	変数と制約	14
5.2.2	制約の種類	14
5.3	リーダーによる割当て問題	16
5.3.1	変数と制約	16

5.3.2	制約の種類	18
5.4	問題の難しさの比較	19
5.4.1	リーダー選出問題	19
5.4.2	リーダーによる割当て問題	19
5.5	局所的な同期	20
5.5.1	局所的な判定	21
5.5.2	メッセージの種類	21
6	評価	23
6.1	実験 1	23
6.1.1	問題設定	23
6.1.2	結果	23
6.2	実験 2	23
6.2.1	結果	25
6.3	実験 3	27
6.3.1	結果	27
7	まとめ	29
	謝辞	30
	参考文献	31

1 はじめに

近年，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．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研究に分散制約充足/最適化問題がある．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あらわすものとして重要である．分散制約充足/最適化問題の解法には時間がかかるが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が高速に解くことが出来る確率的な解法がある．

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も広く行われている．また，分散センサ網の観測資源の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行われている [1]．分散センサ網の目的として様々な分野が考えられるが，本研究では視野を制御可能な自律的なカメラ群による観測システムを想定する．このようなシステムでは単に画像を記録するだけではなく，各カメラエージェントが協調して観測対象を追跡するなど，高度な処理が期待される．このようなシステムを実現するためにエージェントの概念を用いた分散協調処理によるモデルが提案されている [2][3]．

本研究で想定する観測システムにおいては，観測対象の検出やカメラ間での対象同定，観測対象への資源割当てなどいくつかの重要な問題がある．その中で本研究では観測対象への資源割当て問題について検討をする．

実システムでは実時間内に処理を完了することが必要不可欠である．そのためには，少ないサイクル数で観測対象への資源割当て問題の解を見つけることが重要であるといえる．観測対象への資源割当て問題を分散制約充足/最適化問題として形式化する場合，厳密解法を適用すると問題規模を大きくした時に解探索時間が大幅に増加してしまうため，厳密解法よりも確率的な解法を適用した方が有用であると考えられる．また，エージェントの協調と資源割当てに関する問題を同時に解く事は複雑であり，探索に要する時間と解の精度には課題がある．

そこで本研究では確率的な解法を用いることを前提として，解を発見するまでのサイクル数を削減しつつ一定の解の精度を得ることを目的とする．このために問題を 2

つの階層に分割し，階層間で局所的な同期をとることを提案する．評価は分散センサ網の資源割当て問題のモデルの1つであるグリッドモデルを用いて評価を行う．

以降，第2章では分散制約充足/最適化問題について説明する．第3章では今回扱う分散センサ網の資源割当て問題について説明する．第4章では従来研究について説明する．第5章では今回提案する手法について説明をする．第6章では従来手法と提案手法の評価・比較を行う．最後に第7章で本研究についてまとめる．

2 分散制約充足/最適化問題

本章では分散制約充足/最適化問題 (Distributed constraint Satisfaction/Optimization Problems, DCSP/DCOP) の形式と解法について説明する。

2.1 概説

分散制約充足/最適化問題と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た問題である。分散制約充足/最適化問題はマルチエージェントシステムの基礎的な問題の一つとして研究されている。

2.2 形式化

制約充足問題/最適化問題 (CSP/COP) は次のように形式化される。

- n 個の変数 x_1, x_2, \dots, x_n が存在する。
- 各変数は取り得る値での集合である D_1, D_2, \dots, D_n 及び制約の集合 C を持つ。
- 制約には評価関数があり, 制約に関連する変数の組合せの違反度を計算する。
- 問題の目的は, 全ての制約を満たすような変数値の組合せ (充足解) を見つける事, または制約の違反度が最小となるような値の組合せ (最適解) を求めることである。

分散制約充足問題/最適化問題 (DCSP/DCOP) とは変数と制約が複数の異なるエージェントに分散された問題である。制約の繋がったエージェントを近傍エージェントとし, 近傍エージェントとメッセージを交換しながら解を求める。分散制約充足問題/最適化問題は次のように形式化される。

- エージェントの集合を A とする。
- 各エージェント $i \in A$ は変数 x_i を持つ。
- 変数 x_i は値域 D_i 及び制約の集合 C_i を持ち, x_i の値はエージェント i のみが決定出来る。
- 制約には評価関数があり, 制約に関連する変数の組合せの違反度を計算する。
- 問題の目的は, 全てのエージェントが全ての制約を満たすような変数値の組合せ (充足解) を見つける事, または制約の違反度が最小となるような値の組合せ (最適解) を求める事である。

2.3 解法

分散制約充足/最適化問題の解法は厳密解法と確率的な解法がある．厳密解法の実例として Adopt(Asynchronous Distributed Constraint Optimization)[4] があげられる．Adopt では二項制約網に対する深さ優先探索法を用いてサブツリーを作成し，違反度の情報を交換し最適値を求める解法である．Adopt では完全性が保証されており，充足解が存在しない場合でも最適解を発見することが出来る．しかし，問大規模が大きくなると探索時間が大幅に増加するという問題点がある．

確率的な解法の実例として DSA(Distributed Stochastic Search)[5] と DSTS(Distributed Stochastic Tabu Search)[6] があげられる．確率的な解法は解の完全性は保証されていないが，短時間で解を発見することが可能である．その点から，実システムに適用する場合，確率的な解法の方が有用であると考えられる．そこで本研究では厳密解法では無く確率的な解法を用いて検討を行う．次の節で DSA と DSTS について詳細を説明する．

2.3.1 DSA : Distributed Stochastic Search

DSA 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．変数の値をエージェント同士交換し，違反度が 0(もしくは最小)となる値を求める．DSA の手順を以下に示す．

1. 変数の初期値をランダムに決定する．
2. 終了条件を満たしているかを判定する．
3. 近傍センサからの情報を集計する．
4. 集計した情報を利用し，違反度が最小となるような新しい値 `New_Value` を決定する．
5. 改善量 > 0 , または改善量 $= 0$ でかつ自信の制約に違反が残っていれば確率 P で変数の値を `New_Value` 変更する．
6. 変数の値を変更したら近傍エージェントに情報を送信する．
7. 2に戻る．

DSA は非常にシンプルであるが，局所解に落ち入りやすく脱出しにくいという問題がある．局所解に陥るといのは，局所的に最適解となってしまう，全体の違反度を改善できなくなる現象である．

2.3.2 DSTS : Distributed Stochastic Tabu Search

DSTS は DSA にタブーサーチ機能を組み込んだアルゴリズムである．値変更時に，以前の値への遷移をタブーリストに追加し一定期間その値への遷移を禁止することで

局所解から脱出しやすくする．DSA でのパラメータは値を変更する確率 P の 1 つだけであったが，DSTS では 3 つ使用する．DSTS は基本的に DSA と同じ動作であるが，以下の点が異なる．

1. タブーリストへの書き込みにより，取り得る値が無くなった場合は自らの改善を停止する（タブー状態）．またその事を近傍へ通知する．
2. 近傍エージェントが全てタブー状態となり，自身に制約違反があるとき値を変更する．
3. 違反度が最小となるような新しい値 New_Value を求めた時，改善量 ≥ 0 であれば確率 P_1 で値を変更する．
改善量 < 0 であり自身に制約違反があれば確率 P_2 で値を変更する．
4. 値を変更した場合，タブーリストに値を書き込み，一定期間（TABU 期間）遷移を禁止する．

DSTS では選択した値が違反度を増加させるような値でも確率 P_2 で遷移を許す．このように値を遷移させることで，一時的に違反度を増加させるが，局所解から脱出しやすくする．DSTS は DSA よりも局所解から脱出しやすく，優れている事が示されている [6]．

3 分散センサ網の資源割当て問題

本章では今回扱う問題である分散センサ網の資源割当て問題について説明する。

3.1 概要

分散センサ網の資源割当て問題とは、複数のセンサと複数のターゲットが存在するような状況を想定した問題である。各センサがどのターゲットを注視するかという組合せは複数存在し、制約違反が無くなるような組合せ、つまり各変数の値を決定する。制約違反が無くならない場合は違反度が最小となるような組合せを探索する。図 1(a) のような配置があった場合、図 1(b) はその割当て例である。

3.2 グリッドモデル

本研究ではグリッドモデルを用いて評価を行った。このモデルは関連研究でも用いられている [1]。グリッドモデルでは格子状にセンサを配置する (図 2)。ターゲットはグリッドで閉じられたエリアに存在し、今回は 1 つのエリアにはターゲットは 1 つまでしか存在できないとする。また、各センサが注視できるターゲット数は 1 つが上限とし、1 つにターゲットに 3 つのセンサを割り当てることを目的とする。今回はターゲットの移動・消滅は無いものとする。

グリッドモデルでは充足解が存在する配置と存在しない配置がある。1 つのターゲットに 3 つのセンサを割り当てるにはセンサの数は最低でもターゲット数の 3 倍は必要

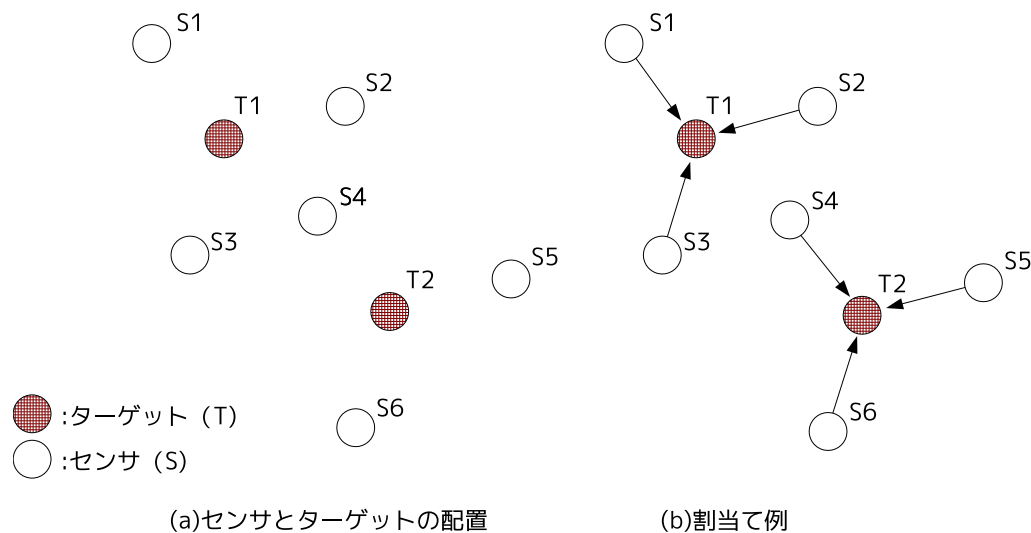


図 1: 分散センサ網の資源割当て問題

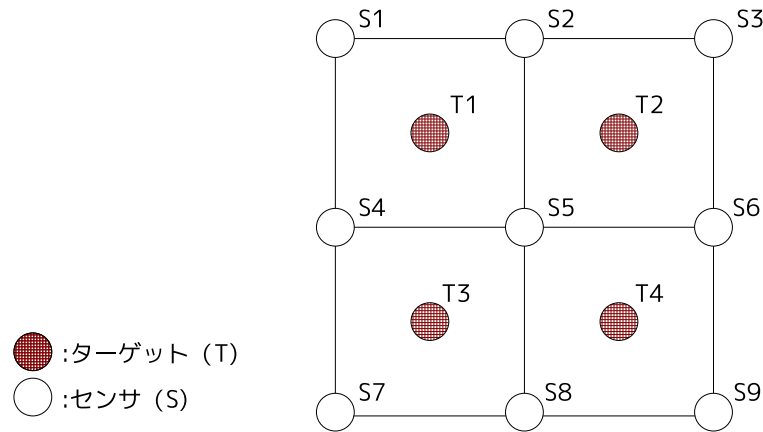


図2: グリッドモデル

である．例えば図2のような場合は充足解は存在しない．また，センサの数がターゲット数の3倍以上あっても，充足解が存在しない配置がある．例えば図2の T_4 が存在しないとすると，センサの数はターゲット数の3倍であるが，全てのターゲットに対し3つのセンサを割り当てるような組合せは存在しない．

3.3 制約の分類

分散センサ網の資源割当て問題の制約の分類について説明する．

まず，intra 制約と inter 制約に分類することが出来る．

- intra 制約

intra 制約は同一センサ内での制約である．

- inter 制約

inter 制約は異なるセンサ間での制約である．

また，制約は hard 制約と soft 制約に分類することが出来る．

- hard 制約

hard 制約は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緩和が不可能な制約である．

- soft 制約

soft 制約は目的関数であり，満たしていなくてもシステムが破綻することはなく緩和が可能な制約である．

4 従来手法

本章では従来手法について説明する。

4.1 STAV(Sensor-Target As Variable)

センサとターゲット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。

図3において S_n はセンサを示し, T_n はターゲットを示している。図3のような配置があり STAV で制約網を作成した場合, S_1 と S_2 の変数と制約を図4のようになる。 T_j^i はセンサ i が持っているターゲット j の割当て情報を示している。つまり各センサは自分の視野範囲内に存在するターゲットの数だけ変数を持つ。図3では S_1 の視野範囲内に存在しているターゲットは一つであり、 S_1 が持つ変数は一つとなる。また, S_2 の視野範囲内に存在しているターゲットは二つで、 S_2 が持つ変数の数は二つとなる。各変数の値域はあるターゲットを観測可能なセンサ $\{S_0, \dots, S_n\}$ の組合せ $\{\phi, \{S_0\}, \dots, \{S_n\}, \{S_0, S_1\}, \dots, \{S_0, \dots, S_n\}\}$ の総数である。本研究で検討するグリッドモデルでは各ターゲットに4つの割り当て候補のセンサが存在するので、各変数の値域は $2^4 = 16$ 通り存在する。

次に制約について説明する。STAV での制約は図4中 A,B,C で示した三つ制約が存在する。

4.1.1 STAV おける制約

- 各センサの資源の制限

この制約は図4中 A で示した制約である。この制約は1つのセンサは1つのターゲットまでしか注視できないという制約で、1つのセンサが2つ以上のターゲットを注視した時に違反になる。各センサーは一つのターゲットまでしか注視でき

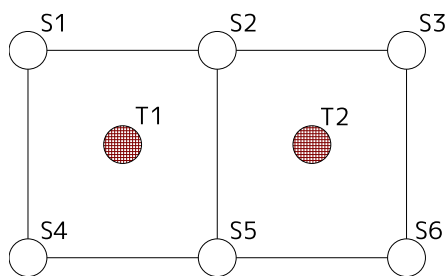


図3: グリッド状での配置例

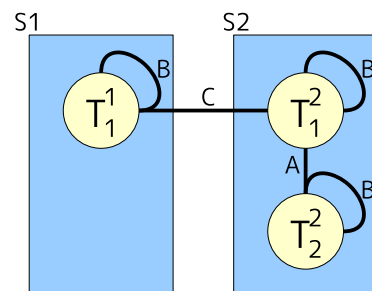


図4: STAV における変数と制約

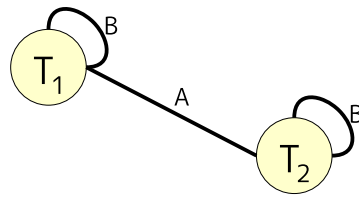


図 5: TAV における変数と制約

ないので一つのセンサが 2 つ以上のターゲットを注視するような変数値であった場合、その制限が無視されたことになりシステムは破綻する。

この制約は同一センサ内の多項制約で、intra 制約である。また、違反していた場合システムが破綻してしまうので緩和が不可能な hard 制約である。

- 各観測対象への資源割当て要求

この制約は図 4 中 B で示した制約である。この制約は一つのターゲットを 3 つ以上のセンサで注視したいという制約で、1 つのターゲットに割り当てられたセンサが 3 つ未満だった場合に違反となる。この制約が満たされていなくてもシステムは破綻しない。

この制約は同一センサ内の単項制約で、intra 制約である。また、緩和が可能な soft 制約である。

- センサ間の合意（協調）

この制約は図 4 中 C で示した制約である。この制約は異なるセンサ同士が同一ターゲットに対して同じ割り当てをとらなければならないという制約である。この制約が異なるセンサ間で協調するための制約となる。異なるセンサ間で同一ターゲットに対する割り当てが異なっていた場合システムは破綻する。

この制約は異なるセンサ間の多項制約で、inter 制約である。また、違反していた場合システムが破綻してしまうので緩和が不可能な hard 制約である。

4.2 TAV(Target As Variable)

ターゲット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。

図 3 のような配置があり TAV で制約網を作成した場合、変数と制約を図に示すと図 5 のようになる。図 5 において T_i はターゲット i に関する割当て情報で、各ターゲットに関する変数は全体で 1 つである。各変数の値域はあるターゲットを観測可能なセンサ $\{S_0, \dots, S_n\}$ の組合せ $\{\phi, \{S_0\}, \dots, \{S_n\}, \{S_0, S_1\}, \dots, \{S_0, \dots, S_n\}\}$ の総数である。つまり STAV と同様で、各変数の値域は $2^4 = 16$ 通り存在する。

次に制約について説明する．TAV での制約は図 5 中 A,B で示した二つ制約が存在する．

4.2.1 TAV における制約

- 各観測資源の制限 (協調)

この制約は図 5 中 A で示した制約である．この制約は 1 つのセンサは 1 つのターゲットしか注視できないという制約である．1 つのセンサが 2 つ以上のターゲットを注視した時に違反になる．TAV において，この制約が異なるセンサ間で協調するための制約となる．各センサは 1 つのターゲットまでしか注視できないので一つのセンサが 2 つ以上のターゲットを注視するような変数値であった場合，その制限が無視されたことになりシステムは破綻する．

この制約は異なるセンサ間の多項制約で，inter 制約である．また，違反していた場合システムが破綻してしまうので緩和が不可能な hard 制約である．

- 各観測対象への資源割当て要求

この制約は図 5 中 B で示した制約である．この制約は 1 つのターゲットを 3 つ以上のセンサで注視したいという制約で，1 つのターゲットに割り当てられたセンサが 3 つ未満だった場合に違反となる．この制約が満たされていなくてもシステムは破綻しない．

この制約は同一センサ内の単項制約で，intra 制約である．また，緩和が可能な soft 制約である．

4.3 STAV と TAV の問題点

4.3.1 STAV

STAV は視野範囲内に存在するターゲットの数だけ変数を持つため，ターゲット数が増えてもセンサ数が増えても変数の数が増加する．それに伴い制約の本数が増加する．つまり問題規模を大きくすると問題は急激に難しくなる．また，値域も問題を難しくする要因の一つである．グリッドモデルでは一つのセンサを観測できるセンサは 4 つに限られており，各変数の値域は割当て候補のセンサ四つがそれぞれ注視する・しないを選べるので $2^4 = 16$ 通りである．しかし，他のモデルや実システムの場合はそうとは限らないのは明らかである．ターゲットを観測できるセンサが 5 つ，6 つと増えていけば値域は $2^5 = 32$ 通り， $2^6 = 64$ 通りと増えていき解空間が大きくなるのも問題といえる．

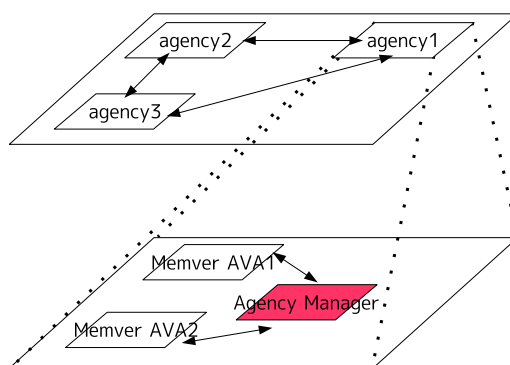


図 6: エージェンシモデル

4.3.2 TAV

TAV は STAV よりも変数も制約も少ない。また、ターゲット毎に変数を置くため、センサの数が増えても変数の数は増加しない。その分ターゲット数やセンサ数が増えても増加する変数や制約は STAV よりも抑えられる。つまり問題規模が大きくなっても、問題の難しさは STAV よりも抑えられると考えられる。しかし、本研究で想定している観測システムの場合、ターゲットには計算資源は無い。つまりターゲットに変数と制約を置く事は不可能である。TAV で制約網を表現しようとした場合、変数と制約をターゲットの代わりに持つ計算資源、つまりセンサを決定する必要が生じる。そのため TAV で制約網を作成した場合と STAV で制約網を作成した場合で解を発見するまでのサイクル数を単純に比較することには意味がない。また、各変数の値域は STAV と同様に、あるターゲットを観測可能なセンサが増えるにつれ、指数関数的に増加する。

4.4 エージェンシモデル

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シ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 [2][3]。概念図を図 6 に示す。このシステムでは自律的に動作する能動視覚エージェント (Active Vision Agent, AVA) の集合によって構成されている。AVA は制御可能なカメラセンサと通信可能な計算機で構成される。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シを作成する。
- エージェンシには一つ代表となるエージェンシマネージャが存在し、他のエージェンシメンバーはエージェンシマネージャの意志決定に従う。

- エージェンシマネージャ同士は通信により，観測資源の配分を決定する．
- 観測対象の情報はエージェンシマネージャに集約される．

5 提案手法

この章では提案手法について説明する．

5.1 基本構造

提案手法では STAV と TAV の欠点を補う事を目的とする．目的を達成することで，解を発見するまでのサイクル数を削減できると考える．

目的を達成するために，提案手法では 2 つの階層に問題を分割する．2 つの階層とは次の二つである．

- リーダー選出層
- リーダーによる割当て問題解決層

STAV の問題点である”問題規模を大きくすると問題が複雑になる”という問題点を解決するには TAV により制約網を作成すれば良い．しかし TAV には”ターゲットには計算資源が無いため，変数と制約を置くことが出来ない”という問題がある．そこで，リーダーを選出することで TAV に問題であった問題を解決する．リーダー選出層では STAV により制約網を表現せざるを得ないが，割り当て問題を STAV を用いて表現するよりも簡単であると考えられる．

またエージェントモデルの概念を分散制約充足/最適化問題の枠組に統合するために，条件を追加設定する．追加する条件は以下の 2 つである．

1. 選出されたリーダーはただの計算資源としての役割では無く，必ず自分が担当しているターゲットの注視に参加しなければならない．
2. リーダーを選出する際，各ターゲットの割当て候補数を保証するようにリーダーを選出しなければいけない．

条件 2 は条件 1 を満たすために必要な条件とも言える．

5.2 リーダー選出

リーダー選出層では各ターゲット毎に 1 つのセンサをリーダーとして選出する．制約網は STAV により作成するが，エージェントモデルの概念を分散制約充足/最適化問題の枠組に統合するための条件を満たすために，従来の STAV では 3 つであった制約に 1 つ制約を追加する．

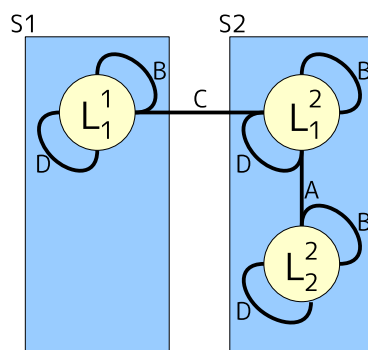


図 7: リーダー選出における変数と制約

5.2.1 変数と制約

リーダー選出における変数と制約について説明する．図 3 において S_n はセンサを示し、 T_n はターゲットを示す．図 3 のような配置があった場合、 S_1 と S_2 のリーダー選出における変数と制約を図に示すと図 7 のようになる． L_j^i がセンサ i が選出しているターゲット j のリーダーを示す．つまりセンサは自分の視野の範囲内にいるターゲットの数だけ変数を持つことになり STAV により制約網を作成したことになる．値域はリーダーが存在しない場合と、あるターゲットを観測可能なセンサーの集合 $\{S_0, \dots, S_n\}$ のうちから一つ選ぶ $n + 1$ 通りである．今回検討するグリッドモデルでは、あるターゲットを観測可能なセンサはどのターゲットでも 4 つなので値域は $4 + 1 = 5$ 通りである．

次に制約について説明する．リーダー選出での制約は図 7 中 A,B,C,D で示した 4 つ制約が存在する．

5.2.2 制約の種類

制約は図 7 中 A,B,C,D で示した 4 つ制約のうち、A,B,C が TAV の問題点である”ターゲットには計算資源が無いため、変数と制約を置くことが出来ない”という問題を解決するために変数と制約を置くセンサを決定するための制約である．D の制約がエージェントモデルの概念を統合するための条件を満たすための制約である．

- 各観測資源の制限

この制約は図 7 中 A で示した制約である．この制約は 1 つのセンサは 1 つのターゲットのリーダーにしかねないという制約である．センサが注視できるターゲットは 1 つなので、リーダーとなれるターゲットも 1 つである．この制約は、複数の変数を持っていた場合、複数のターゲットのリーダーに同じセンサをリーダーとして選出していた場合違反となる．違反していた場合、システムは破綻する．

この制約は同一センサ内の異なる変数間での多項制約で、intra 制約である。また、違反していた場合システムは破綻するので緩和が不可能な hard 制約である。

- 各観測対象への資源割当て要求

この制約は図7中Bで示した制約である。この制約は各ターゲットについてリーダーを選出したいという制約で、ターゲットにリーダーが割り当てられなかった場合、違反となる。この制約が満たされていなくてもシステムは破綻しない。

この制約は同一センサ内の単項制約で、intra 制約である。また、緩和が可能な soft 制約である。

- センサ間の合意（協調）

この制約は図7中Cで示した制約である。この制約は異なるセンサ同士が同一ターゲットに対して同じリーダーを選出しなければいけないという制約である。この制約がリーダー選出において異なるセンサ間で協調するための制約となる。異なるセンサ間で同じターゲットに対して違うリーダーを選出していた場合、システムは破綻する。

この制約は異なるセンサ間の多項制約で、inter 制約である。また、違反していた場合システムは破綻するので緩和が不可能な hard 制約である。

- 各観測対象の割当て候補の確保

この制約は図7中Dで示した制約である。エージェントモデルの概念を統合するための条件を満たすためには、上記の3つの制約だけでは不十分である。なぜならば、上記の3つの制約を満たしただけでは、エージェントモデルの概念を統合するための条件1を満たすように割り当て問題を解いた場合、局所解に陥って解が発見できなくなる場合があるからである。そのために条件2が必要となり、この制約はその条件2を満たすための制約である。

例えば図8のようにリーダーが選出されたとする。Lと書かれたセンサがリーダーに選出されたセンサで、矢印の先が担当となっているターゲットである。1つのセンサが複数のターゲットのリーダーにはなっておらず、全てのターゲットについてリーダーが選出されている。また、図では示していないが、全てのセンサが同じターゲットに対して同じリーダーを選出し協調も出来ているとする。

T_1 と T_2 に関してはリーダー以外の割当て候補が2つ以上存在し、担当しているリーダーと合わせて3つのセンサで注視することが可能である。しかし T_3 はリーダー以外の割当て候補が1つしか存在しない。そのため多くても2つのセンサからしか注視されることが出来ない。 T_3 にはもともと割当て候補は4つ存在してい

るが、そのうち二つは自分以外のターゲットのリーダーとなってしまう自分を注視することが出来なくなっている。そこで、この制約を追加し、図 8 ような状況を回避する。

この制約はリーダーとなるセンサは視野の範囲内に存在するターゲット数が少ないセンサの方が良いという制約である。リーダーの候補の中で視野の範囲内に存在するターゲット数が最小となっていないセンサがリーダーとなった時に違反となる。センサの視野の範囲内に存在するターゲット数が多いほど、多くのターゲットの割当て候補になる。リーダーとなった時、そのセンサは担当ターゲット以外のターゲットの割当て候補から除外されるので、多くのターゲットの割当て候補になっているほど多くのターゲットの割当て候補数を減らしてしまうことになる。よって、視野の範囲内に存在するターゲット数が出来るだけ少ないセンサがリーダーとなった方が各ターゲットの割当て候補数を減らさないため良いといえる。

図 9 はこの制約を追加してリーダーを選出した例である。各リーダーは視野の範囲内にいるターゲット数が最小のセンサがなっている。このようなリーダーの配置であれば、全てのターゲットを 3 つ以上のセンサで注視することが可能である。

また、リーダーとなるのは視野の範囲内にいるターゲット数が少ないセンサの方が良いというのは、グリッドモデル以外でも有用な制約である。どのようなモデル、自分がリーダーになることにより担当ターゲット以外の割当て候補を制限してしまう事には変わらないからである。

この制約は同一センサ内の単項制約で、intra 制約である。また、他の緩和可能な制約を緩和するかどうかにより緩和可能か不可能かが変わる。

5.3 リーダーによる割当て問題

選出したリーダーによって、自分の担当ターゲットをどのセンサで注視するかを決定する。制約網は TAV により作成するが、エージェントモデルの概念を分散制約充足/最適化問題の枠組に統合するための条件を満たすために、従来の TAV では 2 つであった制約に 1 つ制約を追加する。

5.3.1 変数と制約

リーダーによる割当て問題での変数と制約について説明する。図 3 において S_n はセンサを示し、 T_n はターゲットを示す。図 3 のような配置があった場合、2 つのターゲットのリーダーを S^{T_1}, S^{T_2} とし、 S^{T_1} と S^{T_2} の変数と制約を図に示すと図 10 のようになる。各ターゲットのリーダーは 1 つなので同じターゲットに関する変数は全体

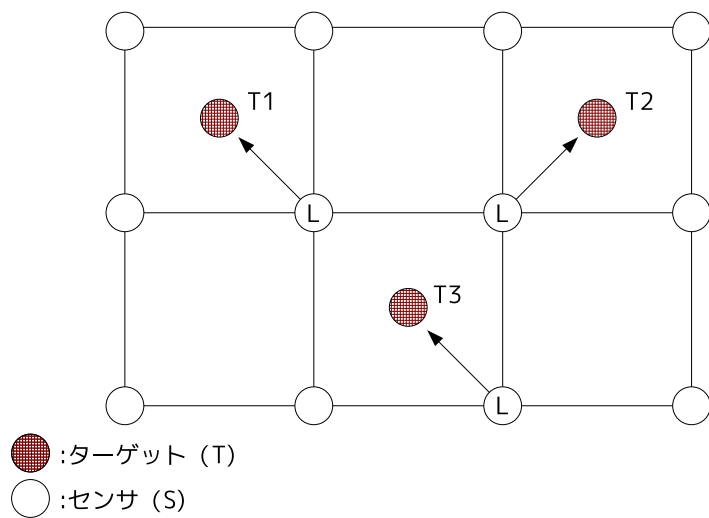


図 8: 局所解の例

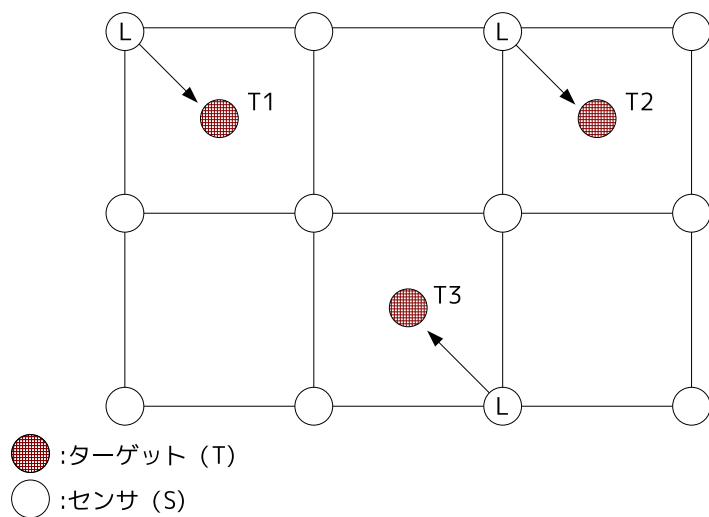


図 9: 局所解を回避した例

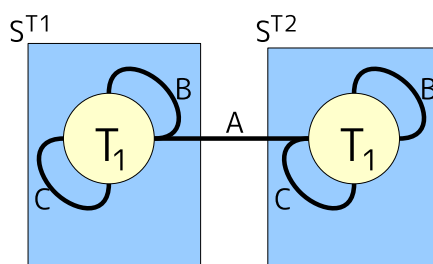


図 10: リーダーによる割当て問題における変数と制約

で1つしかない．また，1つのセンサは1つのターゲットのリーダーにしかないので1つのセンサが持つ変数は高々1つである．つまり TAV で制約網を作成したことになる．各変数の値域はあるターゲットを観測可能なセンサ $\{S_0, \dots, S_n\}$ の組合せ $\{\phi, \{S_0\}, \dots, \{S_n\}, \{S_0, S_1\}, \dots, \{S_0, \dots, S_n\}\}$ の総数である．本研究で検討するグリッドモデルでは各ターゲットに4つの割り当て候補のセンサが存在するので，各変数の値域は従来手法と同様で $2^4 = 16$ 通りである．

次に制約について説明する．リーダーによる割当て問題での制約は図 10 中 A,B,C で示した3つの制約が存在する．

5.3.2 制約の種類

制約は図 10 中 A,B,C で示した3つ制約のうち，A,B が従来の TAV と同様に各ターゲットに観測資源を割り当てるための制約である．C の制約がエージェントモデルの概念を統合するための条件を満たすための制約である．

- 各観測資源の制限 (協調)

この制約は図 10 中 A で示した制約である．この制約は1つのセンサは1つのターゲットしか注視できないという制約である．この制約が異なるリーダー間で強調するための制約である．1つのセンサが2つ以上のターゲットを注視した時に違反になる．リーダーによる割り当て問題において，この制約が異なるセンサ間で協調するための制約となる．この制約が違反していた場合，あるセンサが2つ以上のターゲットを注視していることになりシステムは破綻する．

この制約は異なるセンサ間の多項制約で，inter 制約である．また，違反していた場合システムが破綻してしまうので緩和が不可能な hard 制約である．

- 各観測対象への資源割当て要求

この制約は図 10 中 B で示した制約である．この制約は1つのターゲットを3つ以上のセンサで注視したいという制約で，1つのターゲットに割り当てられたセ

ンサが3つ未満だった場合に違反となる．この制約が満たされていなくてもシステムは破綻しない．

この制約は同一センサ内の単項制約で，intra 制約である．また，緩和が可能な soft 制約である．

- リーダーの振る舞いの制限

この制約は図 10 中 C で示した制約である．この制約はリーダーは自分の担当ターゲットを必ず注視しなければならないという制約である．リーダーが自分の担当ターゲットの注視に参加していない場合に違反となる．この制約を満たすことで，エージェントモデルの概念を統合するための条件 1 を満たす事になる．この制約が満たされていないとリーダーはリーダーとしての役割を果たしておらず，システムは破綻すると考える．

この制約は同一センサ内の単項制約で，intra 制約である．また，違反していた場合システムが破綻してしまうので緩和が不可能な hard 制約である．

5.4 問題の難しさの比較

リーダー選出問題とリーダーによる割当て問題は，従来の STAV で制約網を作成した場合よりも簡単であるといえる．つまり 1 つの難しい問題を二つの簡単な問題に分割することが出来たといえる．この節では詳細を説明する．

5.4.1 リーダー選出問題

従来手法 (STAV) と提案手法におけるリーダー選出問題を比較する．この 2 つは各センサが自分の視野の範囲内に存在しているターゲットの数だけ変数を持つ．さらに制約の本数に関してはリーダー選出問題の方が多い．しかし，各変数の値域は従来手法は 16 通り，提案手法のリーダー選出問題は 5 通りである．また各観測対象の割当て候補の確保に関する制約は各センサ間の合意に関する制約の補助にもなっていると考えられる．なぜならば，視野範囲内に存在しているターゲット数が少なければリーダーの候補になるターゲット数も少なく，他のセンサと協調しやすいからである．つまり追加した制約は解を発見するための重荷にはなっていないと考えられる．

以上より，提案手法におけるリーダー選出問題の方が簡単な問題であると言える．

5.4.2 リーダーによる割当て問題

従来手法 (STAV) とリーダーによる割当て問題を比較する．まず変数の数が違う．従来の割当て問題では STAV により制約網を作成していたため自分の視野の範囲内に存在するターゲットの数だけ変数を持っていたが，リーダーによる割当て問題では TAV

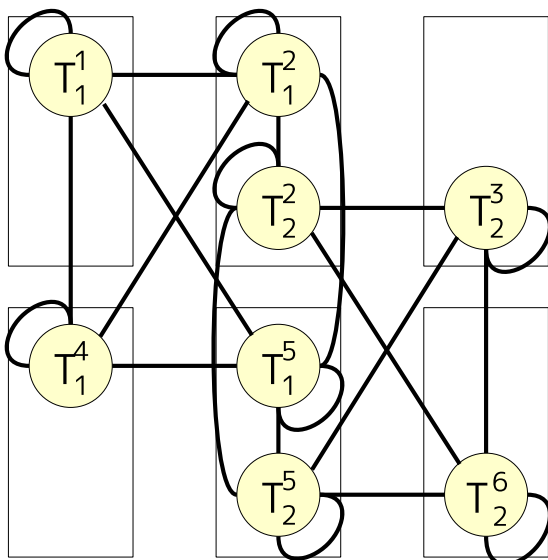


図 11: 従来手法の制約網

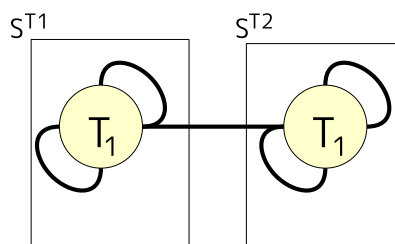


図 12: リーダーによる割当て問題の制約網

により制約網を作成するため自分がリーダーとなっているターゲットについての変数だけを持っている。また、各ターゲットについてリーダーは1つのみなので各ターゲットに関する変数は全体で1つのみである。図3のような配置があった場合、変数の数を比較すると、従来の割当て問題では全体で8つ存在するのに対しリーダーによる割当て問題では2つのみである。変数が少ないため、リーダーによる割当て問題の方が簡単な問題だと考えられる。実際に制約網を図11と図12に示す。図11が従来の割当て問題で図12がリーダーによる割当て問題である。図11で示したが従来の割当て問題の方が制約網が複雑だということがわかる。

また、リーダーの振る舞いの制限という制約は値域の制限に繋がる。この制約はすなわち、他のターゲットのリーダーは自身の担当ターゲットには割当てられないということである。この制約は割当て候補を減らし探索する解空間を小さくする役割も果たし、結果的に問題を易しくしていると考えられる。他のモデルや実システムへの適用を考えた場合、前章で解空間が大きくなる事を問題としたが、それも抑えられる。

以上より、提案手法におけるリーダーによる割当て問題の方が簡単な問題と言える。

5.5 局所的な同期

近傍というのは制約がつながっているエージェント、つまりセンサ同士であるとしたが、選出したリーダーが必ずしも制約がつながっているとは限らない。そこで、2つに問題を分割するにあたり、リーダーは互いにリーダーであるということを知る

必要がある．最も簡単な方法として，システム全体で同期をとることが考えられる．しかし，この方法ではすべてのリーダーが決定するまで待たなければいけない．分散制約充足/最適化問題の枠組を適用してリーダーを選出する場合，リーダーが決定するまでのサイクル数はターゲットによって異なる．早い段階で決定したリーダーが決定したリーダーが最後のリーダー決定するまで何もしないで待つというのは，サイクル数を削減するという観点から望ましくないと考える．また，分散アルゴリズムの性質上，システム全体の状態を知る必要はなく制約が繋がっているセンサの情報さえ知っていれば良い．そこで，局所的にリーダーが見つかった所から割当て問題を開始することにする．

5.5.1 局所的な判定

局所的にリーダーが選出された時，メッセージのやりとりでリーダーであることを伝える．そのやりとりは選出されたリーダーから始まる．局所的にリーダーに選出されたとする条件は以下の通りである．

- 自身の視野の範囲内に存在するいずれかのターゲットのリーダーに自身が選出されている．
- リーダー選出における制約に関する違反度が0である．

もっと緩い条件にすることもできるが，緩い条件にした場合リーダーが移り変わる可能性が高くなる．もちろん，違反度が0である場合でもリーダーが移り変わる可能性は残っているが，違反度が残っている場合よりは移り変わる頻度は少ない．また，自身が視野範囲内に存在するいずれかのターゲットのリーダーに選出されていたとしても，2つ以上のターゲットにリーダーに選出されていた場合はリーダーとして機能しない．その点からも，違反度が0という条件は局所的にリーダーが見つかったとする事に必要な条件である．

5.5.2 メッセージの種類

局所的な同期をとるためのメッセージについて説明する．同期をとるためのメッセージはLEADERメッセージ,RELEASEDメッセージ,HOP_Lメッセージ,HOP_Rメッセージの4種類である．

- LEADER メッセージ

このメッセージは自身がリーダーに選出された場合，近傍センサに送信する．このメッセージを受け取ったセンサはメッセージ送信元のセンサをリーダーリストに追加する．

- RELEASED メッセージ

このメッセージはリーダーとして選出されていたが、外されてしまった場合に近傍センサに送信する。このメッセージを受け取ったセンサはメッセージ送信元のセンサをリーダーリストから除外する。

- HOP_L メッセージ

このメッセージはLEADERメッセージを受け取ったセンサが近傍センサにリーダーとなったセンサの情報を送信する。このメッセージを受け取ったセンサは送られてくる情報で知らされるリーダーを、リーダーリストにはいない場合追加する。このメッセージによって、制約が繋がっていないリーダー同士を制約で繋げる。

- HOP_R メッセージ

このメッセージはRELEASEDメッセージを受け取ったセンサが近傍センサにリーダーから外されたセンサの情報を送信する。このメッセージを受け取ったセンサは送られてくる情報で知らされるセンサを、リーダーリストに入っている場合除外する。

リーダーによる割当て問題ではリーダーリストに存在するセンサとメッセージをやりとりする事になる。

6 評価

この章では提案手法と STAV で制約網を表現した従来手法を評価し、比較を行う。

6.1 実験 1

6.1.1 問題設定

実験 1 は以下のような問題設定で行った。

- センサ数：10 × 10 の 100 個。
- ターゲット数：1 ~ 20 でターゲット数は 1 ずつ増加させる。
- 配置：充足解が存在するような配置で、制約網が 1 つに繋がるように配置。
- 解法：DSTS
- 評価対象：解発見率、サイクル数。

ここで言うサイクル数は、各センサが処理を行いメッセージを送信するまでを 1 サイクルとしている。評価は 1000 サイクルを 1 試行とし 100 試行の平均をとった。また、解を発見出来なかったときは 1000 サイクルとして計算している。各パラメータは 1 番良い結果を得られた値を採用した。

この問題設定では、ターゲット数が増えても、全体として見たとき制約網が 1 つ繋がりになっているので、局所的に複数の問題が存在しているわけでは無い。つまり、ターゲット数が増えるにつれ問題は純粹に難しくなっていく。

6.1.2 結果

図 13 が解発見率、図 14 がサイクル数のグラフである。どちらの結果も提案手法の方が良い結果となっている。グラフは綺麗に増加/減少傾向を示していないが、これは確率的な解法を用いたため、どうしても結果にぶれが生じてしまうためである。

制約網が 1 つに繋がるような配置ではターゲット数が増えるにつれ問題は難しくなる。従来手法では、難しくなった問題をそのまま解いているのに対し一方提案手法では、問題を 2 つに分割することで問題が難しくなるのを抑えているため良い結果が得られたと考えられる。

6.2 実験 2

実験 2 は実験 1 と同じ問題設定で行った。ただし、緩和可能な制約は緩和し満たしていなくても良いこととした。緩和不可能な soft 制約を全て満たしてる時の解を緩和解とする。緩和できる制約は以下の通りである。

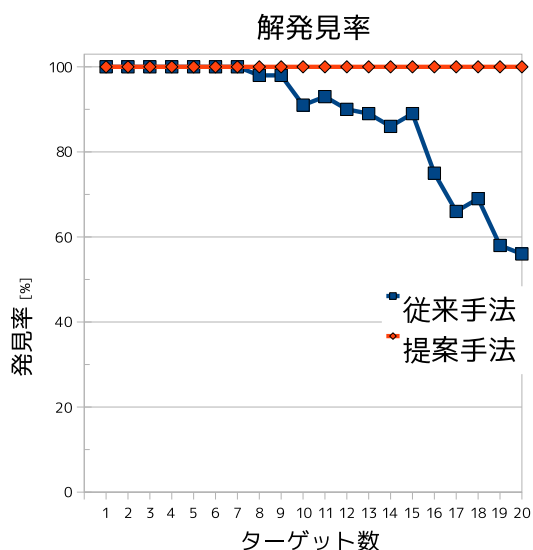


図 13: 解発見率

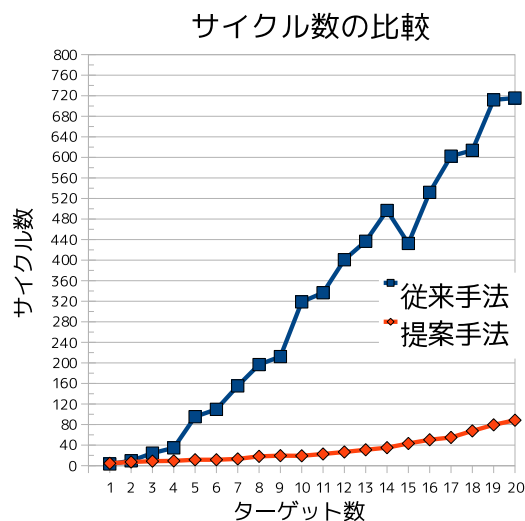


図 14: サイクル数

- リーダー選出
 - － 各観測対象への資源割当て要求
 - － 各観測対象の割当て候補の確保
- リーダーによる割当て問題
 - － 各観測対象への資源割当て要求

リーダー選出における制約の緩和は満たさなくても良いというだけであるが、リーダーによる割当て問題では3つのセンサで注視しなくても良いというだけではなく、次のような緩和する基準を設ける。

1. 出来るだけ3つのセンサで注視した方が良い。
2. 注視しているセンサが3つ未満の場合、出来るだけ多くのセンサで注視した方が良い。
3. 3つより多くのセンサで注視するのは冗長的である。

以上の3の基準により、違反度の重みは、

$$0 \gg 1 > 2 > 4 > 3 \text{ (各ターゲットに割り当てられたセンサ数)}$$

という関係にする。一つのターゲットに3つのセンサが割り当てられたときは違反度0である。また、緩和不可能な hard 制約の重みは緩和可能な soft 制約に比べ十分大きく設定した。

さらに，DSTSの動作を一部変更する．変更する箇所は確率 P_2 で値を変移する条件で，“改善量 0 でも制約違反があれば確率 P_2 で値を変更する”から“改善量 0 でも hard 制約に制約違反があれば確率 P_2 で値を変更する”に変更する．また，局所的な同期に関して，局所的にリーダーに選出されたとする条件を“リーダー選出における制約に関する違反度が 0 である”から“リーダー選出における制約のうち hard 制約に関する違反度が 0 である”に変更する．

緩和可能な制約を緩和するので，評価に“センサが割り当てられているターゲットの総数”と“注視に参加しているセンサ数”を追加する．また，サイクル数は緩和解を見つけるまでのサイクル数で，初めて緩和解を見つけたときのサイクル数とする．

6.2.1 結果

図 15 が解発見率，図 16 がサイクル数，図 17 がセンサが割り当てられているターゲットの総数，図 18 がターゲットに割り当てられたセンサの総数である．センサが割り当てられているターゲットの総数に関しては，全てのターゲットを注視している場合を 1 に正規化している．ターゲットに割り当てられたセンサの総数に関しては，充足解が存在する配置なのでターゲット数 $\times 3$ が最良の値とし 1 に正規化している．

解発見率は従来手法では若干のぶれがあり 100 % ではないが提案手法では 100 % を維持している．また，サイクル数も大きく削減できている．センサが割り当てられているターゲットの総数に関しては提案手法では極小さな差ではあるが完全に 1 を維持できていない．これは，制約を緩和したリーダー選出問題では観測可能な全てのターゲットに対してリーダーを選出しなくても緩和不可能な制約を全て満たしている状況がおきるため，全てのターゲットについてリーダーを選出していないまま選出されたリーダーの間で緩和解を見つけてしまうという状況が発生するためである．このような状況ではターゲット数だけリーダーが存在しないので，全てのターゲットについて割当てを行うことが出来ない．従来手法では全てのターゲットに対して必ず割当てを行い，ターゲットに対して割当てを行わないときの違反度が重いため，注視されていないターゲットが発生する頻度は提案手法よりも少ない．一方，注視に参加しているターゲット数に関しては提案手法の方が良い値となっている．これは，提案手法はセンサ間の協調に関する制約が従来手法にくらべ疎なので，緩和可能な制約の違反度が小さい所で協調しやすいためであると考えられる．従来手法ではセンサ間の協調に関する制約も複雑なため，緩和可能な制約の違反度が大きい所でないと協調しづらいつとえられる．

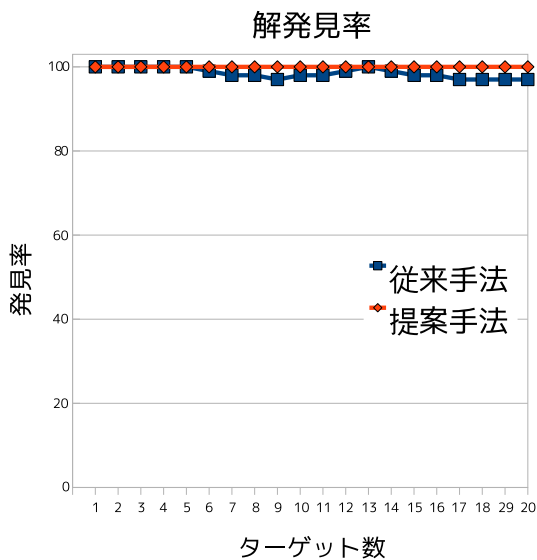


図 15: 解発見率



図 16: サイクル数

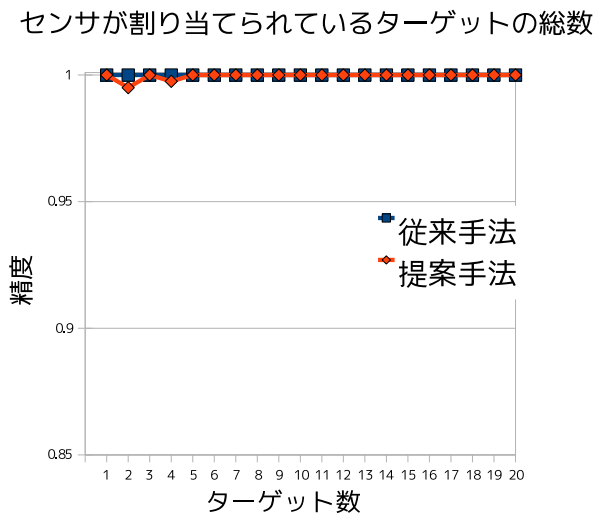


図 17: センサが割り当てられたターゲットの総数

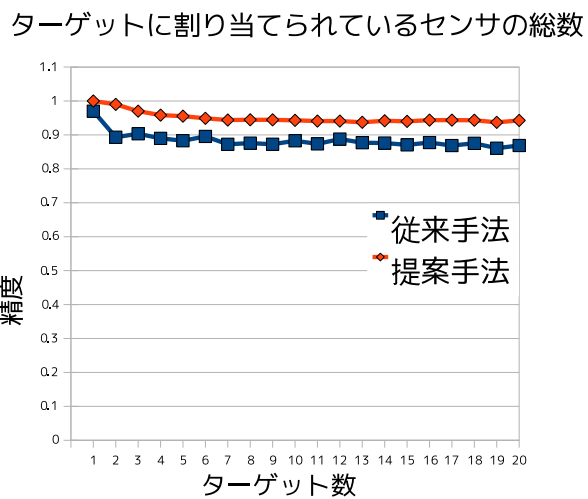


図 18: ターゲットに割り当てられたセンサの総数

図 17: センサが割り当てられたターゲットの総数 図 18: ターゲットに割り当てられたセンサの総数

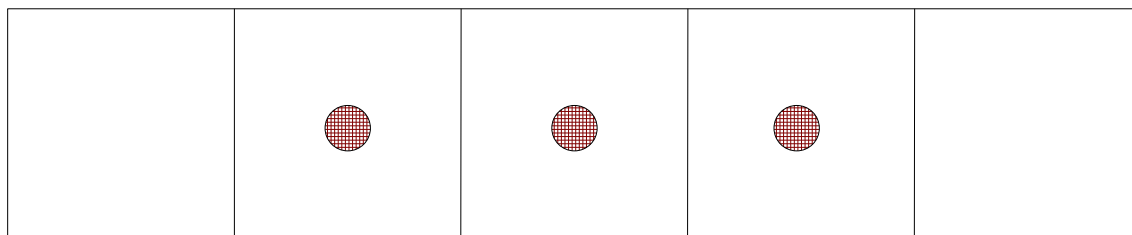


図 19: 想定する配置

6.3 実験 3

実験 3 は以下のような問題設定で行った．

- 通路を想定．
- ターゲット数：5～25 でターゲット数は 5 ずつ増加させる．
- 配置：充足解が存在しない配置で，ターゲットを 1 列に繋げて配置．

通路を想定した配置とは図 19 のような配置である．充足解が存在しない配置なので，緩和できる制約は緩和して緩和解を求める．緩和の仕方は実験 2 と同じで，各観測対象への資源割当て要求の違反度の重みに関しても実験 2 と同じである．また，評価する内容も実験 2 と同で，“解発見率”と“サイクル数”の他に“センサが割り当てられたターゲットの総数”と“ターゲットに割り当てられたセンサの総数”について評価する．

6.3.1 結果

図 20 が解発見率，図 21 がサイクル数，図 22 がセンサが割り当てられているターゲットの総数，図 23 が注視に参加しているセンサ数である．図 23 に関しては出来るだけ多くのターゲットを出来るだけ多くのセンサで注視したいと言う事で，ターゲット数 $\times 2 + 2$ が最良の値とし 1 に正規化してある．この値は全てのターゲットを出来るだけ多くのセンサで注視したときの値である．

結果から，解発見率は上がりサイクル数は削減できていることがわかる．従来手法ではそもそも解を発見できている確率が低いですが，提案手法では 100% 発見できている．センサが割り当てられたターゲットの総数とターゲットに割り当てられたセンサの総数を見ても，提案手法は従来手法のそれを維持している．

このような配置では複数のターゲットが注視できるセンサの数が多くなり制約網はより密になる．そのため問題を簡単な 2 つの問題に分割し問題の規模を小さくすることの効果により顕著に表れたと考えられる．

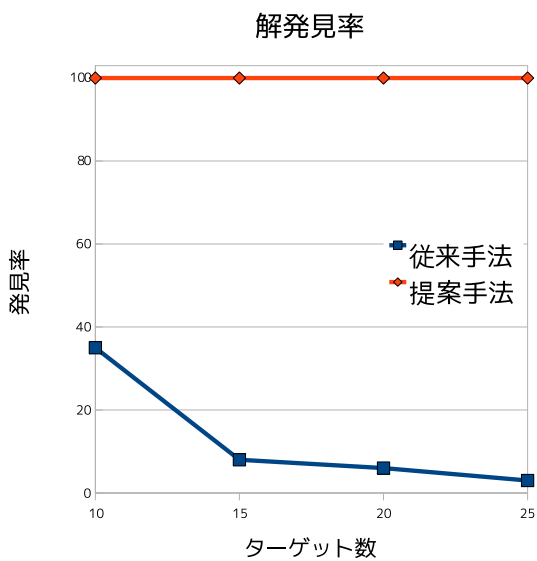


図 20: 解発見率

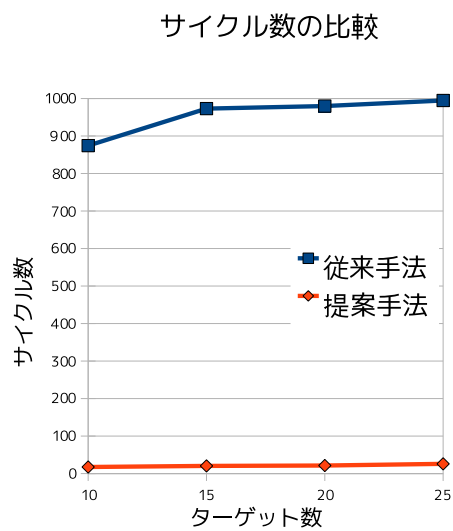


図 21: サイクル数

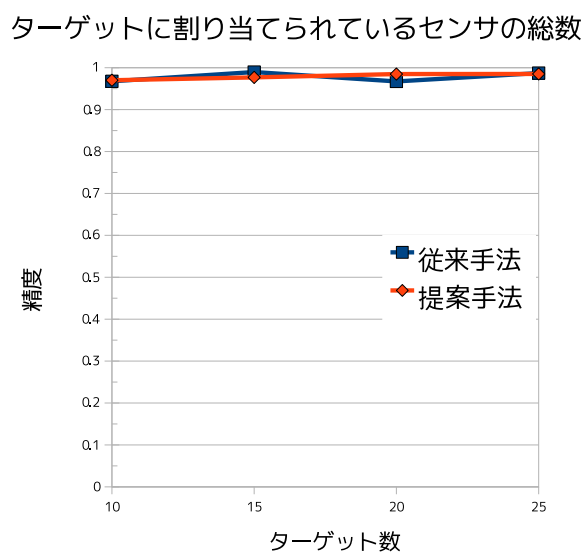
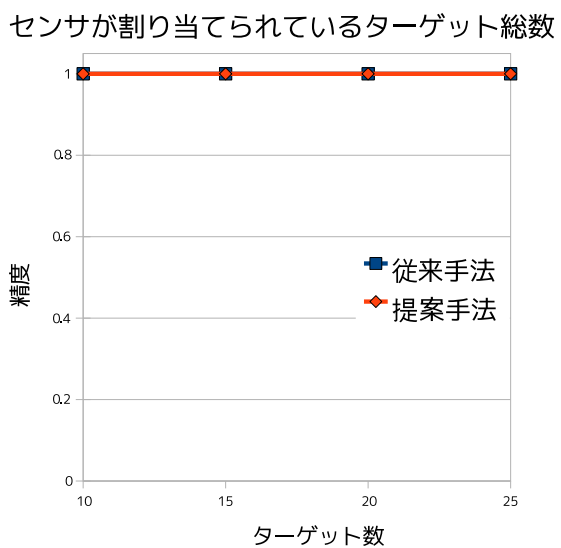


図 22: センサが割り当てられたターゲットの総数 図 23: ターゲットに割り当てられたセンサの総数

7 まとめ

本研究では、視野を制御可能な自律的なカメラ群による観測システムを想定し、重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つの階層に分割し、各層に対応する問題解決処理系を、分散処理の特徴を利用し、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、解を発見するまでのサイクル数を削減しつつ、比較的良好な解が得られる事を実験結果により示した。また、提案手法が解を発見する割合は従来手法よりも大きいことを示した。より一般的なセンサ網のモデルへの適用、複数の問題解決層の同期方法の改善、実システムにおける検証が今後の課題である。

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深く感謝致します．

参考文献

- [1] Ramon Bejar, Carmel Domshilak, Cesar Fernandez, Carla Gomes, Bhaskar Krishnamachari, Bart Selman, and Magda Valls. Sensor networks and distributed csp. *Artificial Intelligence 161*, pp. 117–147, 2005.
- [2] 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 研究会論文誌第 43 巻, 2002.
- [3] 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.
- [4] Pragnesh Jay Modi, Wei-Min Shen, Milind Tambe, and Makoto Yokoo. Asynchronous distributed constraint optimization with quality guarantees. *Autonomous Agents and Multi-Agent Systems*, 2003.
- [5] Weixiong Zhang, Guandong Wang, and Lars Wittenburg. Distributed stochastic search for constraint satisfaction and optimization. *AAAI-02 Workshop on Probabilistic Approaches in Search*, 2002.
- [6] 飯塚泰樹, 鈴木浩之. Tabu search を用いたマルチエージェント型分散制約充足手法の提. 電気情報通信学会信学技報, 2006.