

# 検知領域交点を考慮した連結センサカバーアルゴリズム

高田 亮 藤原 暁宏  
九州工業大学 情報工学部 電子情報工学科

概要：近年、特定範囲の検知機能と無線通信による通信機能を備えた小型センサによるセンサネットワークが注目を集めている。このセンサネットワークにおける通信コストの削減手法の一つとして、連結かつ要求領域を検知領域で被覆することのできるセンサの集合である連結センサカバーを求めてネットワークを構築する手法が提案されている。本研究では、与えられたセンサ集合と要求領域に対して、少ない個数のセンサで構成される連結センサカバーを求めるアルゴリズムを提案する。まず最初に、最適化手法の一つである局所探索を用い、連結センサカバーにおけるセンサ数の検証を行う。次に、既存の手法により得られる連結センサカバーのセンサ数を検証するとともに、検知領域交点を考慮した連結センサカバーを求める手法を提案する。最後に、既存のアルゴリズムと提案アルゴリズムをシミュレーション環境において検証し、提案手法により既存手法よりも少ないセンサ数の連結センサカバーが得られることを示す。

## A connected sensor cover algorithm considering intersection points

Ryo Takada, Akihiro Fujiwara  
Faculty of Computer Science and Systems Engineering, Kyushu Institute of Technology

**Abstract:** Recently, sensor networks, which consists of small sensors that have sensing and wireless communication functions, has considerable attention. In the sensor network, the connected sensor cover is widely considered as one of technique that reduce the communication costs. The connected sensor cover is a problem that compute a set of sensors that are connected and cover a given region. In the paper, we propose an algorithm for computing the connected sensor cover consisting of a small set of sensors. We first propose a centralized algorithm using the local search, which is a well-known optimization technique, and verify the number of sensors in the obtained connected sensor cover. We next propose a distributed algorithm for computing the connected sensor cover for considering intersection points of sensing regions. In addition, we implement a known algorithm and the proposed algorithm in simulation environment and compare the number of sensors in the obtained connected sensor covers. The results show that the number of sensors obtained by the proposed algorithm is constantly less than the number obtained by the known algorithm.

## 1 はじめに

近年、特定範囲の検知機能と無線通信による通信機能を備えた小型センサによるセンサネットワークが注目を集めている。このセンサネットワークは、観測対象とする環境に複数のセンサノードを分布させ、各センサノードが取得した環境データを基地局に収集するためのネットワークである。

このセンサネットワークでは、特定の領域全体をセンサ群の検知領域で被覆することがしばしば求められており、加えて、各センサが得た情報は1台の計算機で収集されることも必要である。この操作を一般にセンサネットワークにおける照会と呼ぶが、この照会を行うためには、任意のセンサ間は連結である必要がある。

一方で、各センサは容量の小さな充電できないバッテリーで駆動されることが一般的であり、センサネットワークの可用性を高めるためには、通信コストの削減はセンサネットワークにおける重要なテーマである。この通信コストの削減手法の一つとして、照会に必要なセンサの部分集合を求める手法が研究されている。特に、照会に最低限必要なセンサ数よりも遥かに多いセンサでセンサネットワークを構築している場合、この手法は省電力化に非常に効果があると考えられる。この様な連結かつ要求領域を検知領域で被覆することのできるセンサの集合

を、連結センサカバーと呼ぶ。

この連結センサカバーに対して、センサ数が最小となる連結センサカバーを求めることは  $NP$  困難であることが知られている [2]。また、少数のセンサで構成される連結センサカバーを求める手法はいくつか提案されている [1, 2, 3, 4]。既存の手法の一つである Gupta らの手法 [2] ではグリーディー法を基にしており、連結センサカバーとして選出されているセンサの周辺にセンサから要求領域を効果的に被覆するセンサを選出する動作を繰り返すことで、少数の連結センサカバーを求めている。しかし、効果的に被覆するセンサを選出する基準が、その時の最も多くの領域を被覆することの可能なセンサであり、得られた解は改善の余地があるものである。また、森川らの手法 [4] は、複数のセンサから並行してアルゴリズムを開始することによって、連結センサカバーを求める時間の短縮と通信コストの削減を実現している。しかし、Gupta らのアルゴリズムを基に作られているため、得られる解には Gupta らのアルゴリズムと同じ傾向がある。

本研究では、与えられたセンサ集合と要求領域に対して、少ない個数のセンサで構成される連結センサカバーを求めるアルゴリズムを提案する。まず最初に、最適化手法の一つである局所探索を用い、連結センサカバーにおけるセンサ数の検証を行う。次に、既存の手法により得られる連結センサカバーの

センサ数を検証するとともに、分散環境において連結センサカバーを求める新たな手法を提案する。提案アルゴリズムでは、各センサの検知領域が交差する点である検知領域交点を考慮し、この検知領域交点を重点的に被覆するように連結センサカバーを構築するものである。最後に、既存のアルゴリズムと提案アルゴリズムをシミュレーション環境において検証し、提案手法により既存手法よりも少ないセンサ数の連結センサカバーが得られることを示す。

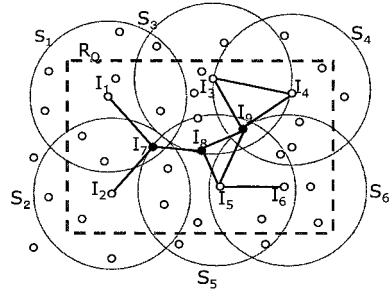


図1: 連結センサカバー

## 2 準備

### 2.1 システムモデル

本研究で対象とするセンサネットワークは、実際のセンサネットワークをグラフを用いてモデル化したものである。センサネットワーク  $G = (V, E)$  は、 $n$  個のセンサの集合  $V = \{I_0, I_1, \dots, I_{n-1}\}$  とリンク集合  $E$  で定義される。各センサ  $I_i$  にはそれぞれ固有の識別番号  $ID_i$  が割りふられている。また各センサ  $I_i$  は二次元平面  $R$  上に配置されており、自身の地理的位置を知っているものとする。

各センサ  $I_i$  は自身を内部に含む通信領域  $T_i$  を持ち、その領域内に存在する他のセンサとのみ無線通信可能である。センサ  $I_i, I_j$  が互いに通信可能であるとき、 $G$  上の  $I_i, I_j$  間に双方向通信リンク  $e_{ij} \in E$  が存在し、 $I_i, I_j$  は隣接するという。

本研究では、各センサの通信領域は、センサを中心とする半径  $t$  の円であると仮定する。また、センサネットワーク中のセンサは全て起動しており、センサ  $I_i$  がセンサ  $I_j$  に送信したメッセージはセンサ  $I_j$  が必ず受信する。このとき、通信の衝突等によるメッセージの消失、及び、メッセージの変更は起こらないとする。また、センサ  $I_i, I_j$  間はメッセージの交換によるのみ情報交換を行う。

センサ  $I_i$  の検知領域  $S_i$  とは、センサ  $I_i$  が平面  $R$  上での検知を保証できる領域であり、検知領域  $S_i$  はセンサ  $I_i$  を内部に含む。本研究では、各センサの検知領域も、センサを中心とした半径  $s$  の円であると仮定する。

センサネットワーク  $G$  において、互いの検知領域に共通領域が存在する任意のセンサ  $I_i, I_j$  の最大通信距離をリンク半径  $r_G$  とよぶ。このリンク半径とは、ネットワーク中で検知領域が重なり合う2つのセンサ間で必要になる最大のホップ数のことである。半径  $s$  の検知領域が重なり合う2つのセンサ間の距離は  $2s$  以内であり、2つのセンサを結ぶ直線上に十分なセンサが存在するならば、通信半径  $t$  に対して、リンク半径は  $\frac{2t}{s} + 1$  以下となる。

### 2.2 連結センサカバー問題

連結センサカバー問題とは、センサネットワーク  $G$ 、及び、被覆すべき領域が与えられているとき、連結かつ領域全体をセンサの検知領域で被覆するセンサの部分集合を求める問題である。なお、被覆すべき領域を要求領域  $R_Q$  とよび、要求領域  $R_Q$  は

センサネットワーク  $G$  が存在する2次元平面  $R$  上の平面領域である。また、連結センサカバーにおいては、センサ数が少なければ少ない程、使用するセンサを減らすことができ、よい連結センサカバーであると考えられる。

以下にこの連結センサカバーを更に厳密に定義する。

**定義1** センサネットワーク  $G = (V, E)$  に存在する各センサ  $I_i$  の検知領域を  $S_i$  とする。センサネットワーク  $G$  に対して要求領域  $R_Q$  が与えられたとき、以下の条件を満たすセンサの部分集合  $M = \{I_{i_0}, I_{i_1}, \dots, I_{i_{n-1}}\} (\subseteq V)$  を連結センサカバーとよぶ。

1.  $R_Q \subset (S_{i_1} \cup S_{i_2} \cup \dots \cup S_{i_m})$

2.  $G$  の  $M$  による部分グラフは連結である。□

ここで、線分を用いた点被覆問題は  $NP$  困難であることが知られており [2]、この問題を更に限定した問題である、センサ数最小の連結センサカバー問題もまた  $NP$  困難問題である。

本稿では、センサ数  $n$  は連結センサカバーを構築するのに十分に大きいとする。すなわち、要求領域  $R_Q$  を被覆可能な連結センサカバー  $M$  は必ず存在するものとする。また、各センサ  $I_i$  には要求領域  $R_Q$  の情報が与えられており、要求領域  $R_Q$  は矩形領域だと仮定する。図1に連結センサカバーの例を示す。この例では、センサ集合  $M = \{I_1, I_2, \dots, I_9\}$  の各センサの検知領域が円で表されており、円集合の和が要求領域  $R_Q$  を被覆しているため、センサ集合  $M$  は連結センサカバーである。

また、この連結センサカバーにおける重要な概念であるサブエレメントについて説明する。サブエレメントとは複数のセンサの検知領域が交差することで区切られる最小の領域である。図2は円で表された3つのセンサの検知領域が交差している例であるが、この状況において、各領域1~7がサブエレメントである。

### 2.3 既存の連結センサカバーアルゴリズム

本節では、既存の連結センサカバーアルゴリズムである Gupta らの手法 [2] について概説する。

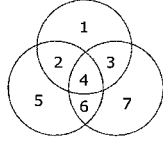


図 2: サブエレメント

まず, Gupta らの集中型アルゴリズム [2] の概略を以下に述べる. Gupta らのアルゴリズムでは, 要求領域内の任意の 1 センサからグリーディー法を元にしたアルゴリズムを実行する.

Gupta らのアルゴリズムは, いくつかのラウンドで構成され, 各ラウンド  $k$  で選ばれたセンサ集合を  $M^k$  と表記する. ラウンド  $k$  では以下の操作を実行する. まず最初に,  $M^k$  に含まれないセンサの中から,  $M^k$  に含まれるセンサと定数個のセンサを介して通信可能なセンサを選出する. このセンサを候補センサと呼び, 候補センサまでの通信可能なセンサの経路を候補パスと呼ぶ. 次に, この候補センサの中から, ある条件により一つのセンサを選び, そのセンサとそのセンサまでの候補パスを連結センサカバーの集合に加えて  $M^{k+1}$  とする.

アルゴリズムは, センサの集合  $M^k$  に含まれるセンサによって要求領域  $R_Q$  を完全に被覆し終えたとき終了する. アルゴリズムの各ラウンドで計算される集合の詳細を以下に示す.

**候補センサ:**すでに連結センサカバーの集合として選出されたセンサ集合  $M^k$  の検知領域を  $S(M^k)$  とする. このとき, 以下の条件を満たすセンサ  $I_i$  のことを候補センサ  $C_i$  とする.

- 1: センサ  $I_i$  は  $M^k$  に含まれていない
- 2: センサ  $I_i$  の検知領域  $S_i$  が  $S(M^k)$  の検知領域と重なりを持つ.

**候補パス:** 候補センサ  $C$  からセンサ集合  $M^k$  に含まれるセンサまでの最短通信経路のことを候補パス  $P_i$  とする.

具体的な Gupta らのアルゴリズムについて以下に示す.

**入力:** 要求領域  $R_Q$ , 及び, センサの集合  $\{I_0, I_1, \dots, I_{n-1}\}$ . ただし, 各センサ  $I_i$  について, 検知領域  $S_i$  と通信領域  $T_i$  が定義されている.

**出力:** 連結センサカバー  $M$ .

### 集中環境における Gupta らのアルゴリズム

アルゴリズムの開始時では  $k = 0$  とし, センサ集合  $M^0$  は空集合である.

**Step 1:** 要求領域  $R_Q$  内の任意のセンサ  $I_i$  を  $M^k$  に加え,  $k = 1$  とする.

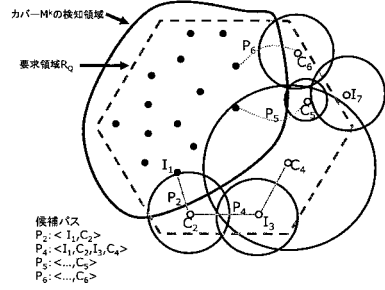


図 3: Gupta らのアルゴリズムの動作例

**Step 2:** カバー  $M^k$  の検知領域  $S(M^k)$  と重なりを持つ候補センサ  $C_i$  を求める.

**Step 3:** 求めた候補センサ  $C_i$  に対し, 候補パス  $P_i$  を決定する.

**Step 4:** 全ての候補パスに対し, 以下の式 (1) で表わされる評価値を計算する.

$$\text{評価値} = \frac{\text{候補パス上のセンサで被覆される未被覆のサブエレメント数}}{\text{候補パス上の } M^k \text{ に含まれていないノード数}} \quad (1)$$

**Step 5:** 全ての候補パスの中から評価値が最大となる最終候補パス  $P$  を選び, 最終候補パス  $P$  上のセンサを  $M^k$  に加える. また,  $k = k + 1$  とする.

**Step 6:** 要求領域  $R_Q$  が完全に被覆されるまで Step 2 から Step 5 を繰り返す.

アルゴリズムの動作例を図 3 に示す. 図 3 では, 候補センサとして  $C_2, C_4, C_5, C_6$  があり, 候補パスとして  $P_2, P_4, P_5, P_6$  がある. 候補パス  $P_5$  と  $P_6$  は図の簡略化のために詳しい経路は省略しており, 候補パス  $P_2, P_4$  について考える. 候補パス  $P_4$  の評価値が最大であるとすると, 候補パス  $P_4$  上のセンサである  $C_2, I_3, C_4$  がセンサ集合  $M^{k+1}$  に加えられる.

次に, Gupta らの分散型アルゴリズムの概略を以下に述べる. Gupta らのアルゴリズムはラウンドで構成されており, 各ラウンドで選ばれたセンサ集合を  $M^k$  と表記する.

Gupta らのアルゴリズムでは, ラウンド  $k$  において以下の操作を実行する. まず最初に,  $M^k$  に含まれないセンサの中から,  $M^k$  に含まれるセンサと定数個のセンサを介して通信可能なセンサを選出する. どの周辺センサを連結センサカバーに加えるかは,  $M^k$  に含まれる 1 センサが決定する. このセンサを  $M^k$  のコーディネータ  $C^k$  とよぶ. アルゴリズムの開始直後では,  $M^0$  に含まれるただ 1 つのセンサがコーディネータ  $C^0$  となる. コーディネータとなるセンサは, カバーにセンサが追加される毎に変わり,  $C^{k_1} \neq C^{k_2} (k_1 \neq k_2)$  である. 次に, コーディネータはある条件により一つのセンサを選び, そのセンサとそのセンサまでの通信経路上のセンサ

を連結センサカバーに加えて  $M^{k+1}$  とする。このとき、コーディネータによって選ばれたセンサがラウンド  $k+1$  のコーディネータ  $C_{new}^{k+1}$  となり、アルゴリズムは、センサの集合  $M^k$  に含まれるセンサによって要求領域  $R_Q$  を完全に被覆し終えたとき終了する。

## 2.4 局所探索

本節では、集中型のアルゴリズムで用いる局所探索と呼ばれる手法について概説する。局所探索とは、なんらかの方法で得られた可能解  $M$  を暫定解とし、この暫定解に対して、その近傍  $N(M)$  (暫定解に変更を加えることによって得られる解の集合) を定義し、 $N(M)$  中の可能解の中で  $M$  より良い目的関数値を持つ解があれば、その解を  $M$  に代入していく方法である [5]。

本研究では、 $N(M)$  は、連結センサカバーを構築するセンサの集合から選んだ連結な  $k$  個のセンサを除外し、新たに  $k$  個未満のセンサを連結センサカバーに加えた解の集合とする。目的関数値は、連結センサカバーのセンサ数とする。また、 $k$  の数に応じて  $k$ -OPT ( $k$ -オプト) と定義する。

## 3 既知の手法に対するシミュレーション

本節では、2章で述べた Gupta らのアルゴリズム [2] をシミュレーション環境に実装し、得られる解の検証と考察を行う。以下にその内容を示す。

本研究のシミュレーションでは、一辺の長さが 100 の正方形を要求領域  $R_Q$  とする。また、入力 of センサ集合は、センサ数を 2000, 3000, 4000 とし、要求領域  $R_Q$  内にランダムに配置する。各センサ  $I_i$  の持つ検知領域  $S_i$  は検知半径  $s = 4$ 、通信領域  $T_i$  は通信半径  $t$  の単一円とし、通信半径  $t$  は 4, 5, 6, 7, 8, 9 に対して計測を行う。各センサはラウンド方式で動作し、1 回のラウンドで各センサはメッセージを受信し、メッセージに対する適切な計算を行い、計算結果としてメッセージの送信を行うものとする。また、評価指標として連結センサカバーを構成するセンサ数  $m$  を用いる。

図 4, 図 5, 及び、図 6 に、センサ数が 2000, 3000, 及び、4000 の場合の Gupta らの集中型、分散型アルゴリズムの結果、及び、集中型アルゴリズムに対する局所探索の結果を示す。縦軸はセンサ数  $m$  を表しており、横軸はセンサの通信半径  $t$  を表している。なお、局所探索は 3-OPT で実行している。また、集中型アルゴリズムと分散型アルゴリズムでは、同じセンサの配置に対して起動センサを 20 と 100 回変更して行っており、グラフの値は 20 と 100 回の実験結果の平均値を表している。

これらの結果より、どのセンサ数においても、Gupta らのアルゴリズムでは集中型、分散型ともに改良の余地があり、特に分散型アルゴリズムでは通信半径が小さい場合において改善の余地があることを示している。

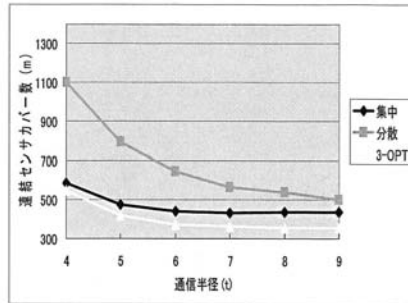


図 4: Gupta らのアルゴリズムによる連結センサカバーのセンサ数 (センサ数 2000)

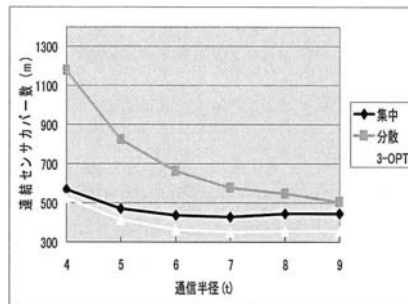


図 5: Gupta らのアルゴリズムによる連結センサカバーのセンサ数 (センサ数 3000)

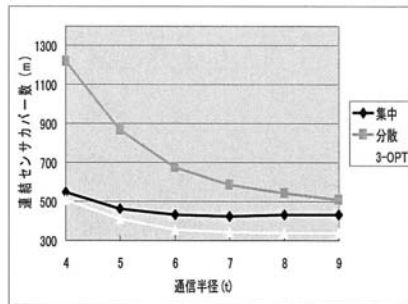


図 6: Gupta らのアルゴリズムによる連結センサカバーのセンサ数 (センサ数 4000)

## 4 検知領域交点を考慮した連結センサカバーアルゴリズム

### 4.1 提案手法のアイデア

本節では、まず提案アルゴリズムのアイデアについて説明する。Gupta らのアルゴリズムでは前述の評価値に従って候補センサを加えることで連結センサカバーを構築するため、検知円同士の重なりが少ないセンサが選出される傾向がある。このため、ど

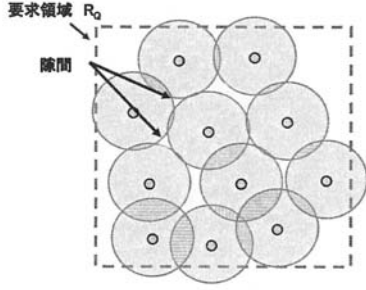


図 7: Gupta の手法によるカバー  $M^k$  の傾向

うしても既に連結センサカバーに選出されているセンサの集合  $M^k$  の検知領域に、図 7 の様な複数の検知円が重なることで生じるセンサ集合  $M^k$  に含まれるセンサで被覆することのできない小さな隙間が残ってしまう。その結果、隙間を埋めるためのセンサが必要となり、最終的な連結センサカバーのセンサ数が大きくなってしまふことがシミュレーション結果から推測される。

そこで提案手法では、円被覆問題における最適配置 [3] に近づく様に、図 8 の様に候補センサを  $M^k$  に含まれる各センサの検知領域同士によってできる交点 (検知領域交点) を被覆するセンサに変更する。ここで検知領域交点とは、共通領域を持つセンサ  $I_i$  と  $I_j$  の共通領域において、センサ  $I_i$  と  $I_j$  の検知領域で被覆することのできない交点のことである。この様に候補センサを選ぶことにより、できるだけすき間を作らないようにし、連結センサカバーのセンサ数を少なくすることができるものと考えられる。

## 4.2 提案集中型アルゴリズム

提案集中型アルゴリズムの概略を以下に述べる。提案アルゴリズムでは、要求領域内の任意の 1 センサからアルゴリズムを開始し、各ラウンド  $k$  で以下の値を計算する。

候補センサ：以下の条件を満たすセンサ  $I_i$  を候補センサ  $C1_i$  とする。

1. センサ  $I_i$  は  $M^k$  に含まれていない
2. センサ  $I_i$  の検知領域  $S_i$  が  $M^k$  に含まれるセンサの検知領域交点を被覆することができる。

候補センサ：以下の条件を満たすセンサ  $I_i$  を候補センサ  $C2_i$  とする。

1. センサ  $I_i$  は  $M^k$  に含まれていない
2. センサ  $I_i$  は  $C1_i$  に選ばれていない。
3. センサ  $I_i$  の検知領域  $S_i$  が  $S(M^k)$  の検知領域と重なりを持つ。

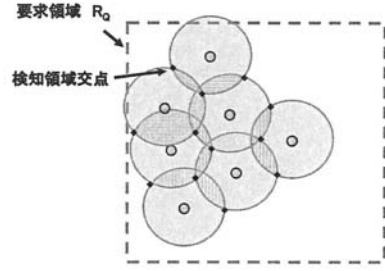


図 8: 提案手法によるカバー  $M^k$  の傾向

候補パス：候補センサ  $C1_i \cup C2_i$  からカバー  $M^k$  に含まれるセンサまでの最短通信経路を候補パス  $P_i$  とする。

具体的な提案アルゴリズムについて以下に示す。

入力：要求領域  $R_Q$ 、及び、センサの集合  $\{I_0, I_1, \dots, I_{n-1}\}$ 。ただし、各センサ  $I_i$  について、検知領域  $S_i$  と通信領域  $T_i$  が定義されている。

出力：連結センサカバー  $M$ 。

提案集中型アルゴリズム

アルゴリズム開始時は  $k = 0$  とし、センサ集合  $M^0$  は空集合である。

**Step 1:** 要求領域  $R_Q$  内の任意のセンサ  $I_i$  をカバー  $M^0$  に加える。

**Step 2:** カバー  $M^k$  に含まれるセンサの検知領域交点を被覆する候補センサ  $C1_i$  と、カバー  $M^k$  の検知領域  $S(M^k)$  と重なりを持つ候補センサ  $C2_i$  を求める。

**Step 3:** 探索した候補センサ  $C1_i$  と  $C2_i$  に対し、候補パス  $P_i$  を決定する。

**Step 4:** 全ての候補パスに対し、以下の式 (2) で表わされる評価値を計算する。

$$\text{評価値} = \frac{N_{SE}}{N_N} \quad (2)$$

ここで、 $N_{SE}$  は、“候補パス上のセンサで被覆する未被覆のサブエレメント数”であり、 $N_N$  は、“候補パス上の  $M^k$  に含まれていないノード数”である。

**Step 5:** 候補センサ  $C1_i$  の中に、評価値が最大となる最終候補パス  $P$  が存在する場合は、その最終候補パス  $P$  上のセンサを  $M^k$  に加える。候補センサ  $C1_i$  の中に候補パスが存在しない場合は、候補センサ  $C2_i$  の中から評価値が最大となる最終候補パス  $P$  を選び、最終候補パス  $P$  上のセンサを  $M^k$  に加える。また、 $k = k + 1$  とする。

**Step 6:** 要求領域  $R_Q$  が完全に被覆されるまで Step 2 から Step 5 を繰り返す。

### 4.3 提案分散型アルゴリズム

提案分散型アルゴリズムの概略を以下に述べる。提案アルゴリズムでは、集中型アルゴリズムと同様に要求領域内の任意の1センサからアルゴリズムを開始する。連結センサカバーを構築し終えるまで、センサネットワークでは以下の値を保持する。

**連結センサカバーの集合:** ラウンド  $k$  終了時に、アルゴリズムによってすでに連結センサカバーに含まれているセンサの集合  $M^k$ 。

**候補パス集合:** 以下の条件を満たすセンサからの候補パスの集合  $SP1^k$ 。

1. センサ集合  $M^k$  に含まれていない。
2.  $M^k$  に含まれるセンサの検知領域交点を被覆する。

**候補パス集合:** 以下の条件を満たすセンサからの候補パスの集合  $SP2^k$ 。

1. センサ集合  $M^k$  に含まれていない。
2. 候補パス集合  $SP1^k$  に含まれていない。
3.  $M^k$  に含まれるセンサの検知領域とセンサが共通領域を持つ。

**候補パス:** 最も新しく連結センサカバーに加えられた候補パス  $P$ 。

**コーディネータ:** 最も新しく連結センサカバーに加えられたセンサ  $C^k$ 。

提案分散型アルゴリズムは以下の4つのフェーズからなる。以下では各フェーズの内容を説明する。

#### 提案分散型アルゴリズム

**1. 探索フェーズ:** 探索フェーズでは、コーディネータ  $C^k$  が周辺センサへ探索メッセージを送信する。探索メッセージにはカバー  $M^k$  に含まれるセンサの座標、及び、コーディネータ  $C^k$  が送信者である情報を付加する。探索メッセージはブロードキャストによってコーディネータ  $C^k$  から  $2r_G (= \frac{4r}{\lambda} + 1)$  ホップ内に存在する全てのセンサに転送される。すなわち、コーディネータ  $C^k$  の検知領域  $S_i$  と共通の検知領域をもつセンサ  $I_j$ 、及び、センサ  $I_j$  の検知領域  $S_j$  と共通の検知領域をもつセンサまで探索メッセージが届くようになっている。探索メッセージを受信した各センサは、最初に受信した探索メッセージのみを受信し、その探索メッセージが経由した経路をセンサ  $I_i$  の候補パス  $P_i$  とする。

**2. 応答フェーズ:** 探索メッセージを受信したセンサ  $I_i$  はコーディネータ  $C^k$  へ応答メッセージを返す。ただし、応答メッセージを返すのは自身の検知領域  $S_i$  が  $M^k$  の検知領域  $S(M^k)$  の検知領域交点を被覆する場合、もしくは、共通領域を持つ場合のみである。この応答メッセージは、探索メッセージが経由した経路、すなわち、候補パスを逆に辿ってコーディネータ  $C^k$  まで伝えられる。応答メッセージには、候補パス (パス上に存在するセンサ ID の列) と、パス上の各センサの検知領域の情報が付加される。

**3. 決定フェーズ:** 決定フェーズでは、コーディネータが応答メッセージによって得られた各候補パス  $P_i$  を候補パス集合  $SP1^k$  と  $SP2^k$  に追加、もしくは、更新し、さらに各候補パス  $P_i$  の検知領域  $S(P_i)$  を保持しておく。候補パス  $P_i$  の検知領域  $S(P_i)$  は以下の式 (3) で求められる。

$$S(P_i) = \cup_{I_j \in P_i} (S_j \cap R_Q) \quad (3)$$

応答フェーズで得られた各候補パスの情報からコーディネータ  $C^k$  は  $M^k$  に追加すべき候補パス  $P_{new}$  を候補パス集合  $SP1^k$  の中から選出し、選出された候補パス  $P_{new}$  上の全てのセンサを  $M^k$  に追加する。

候補パス  $P_{new}$  の選出方法は以下の通りである。コーディネータ  $C^k$  は、すべての応答メッセージを受信すると、候補パス集合  $SP1^k$  に含まれる候補パスの中から以下の式 (4) で表わされる候補パスの表価値が最大となるパスを選出する。このとき、 $SP1^k$  に候補パスがない場合、 $SP2^k$  から選出する。

$$\text{評価値} = \frac{N_{SE}}{N_N} \quad (4)$$

ここで、 $N_{SE}$  は、“候補パス上のセンサで被覆する未被覆のサブエレメント数”であり、 $N_N$  は、“候補パス上の  $M^k$  に含まれていないノード数”である。

評価値が最大となったパスを  $P_i$  とする。コーディネータ  $C^k$  は決定メッセージを  $P_i$  上のセンサに送信し、 $P_i$  上のセンサは新たに連結センサカバーを構成するセンサになる。すなわち、 $M^{k+1}$  は  $M^k$  に含まれるセンサと、新たに追加した  $P_i$  上のセンサで構成される。また、センサ  $I_i$  がカバー  $M^{k+1}$  のコーディネータ  $C^{k+1}$  になる。決定メッセージには候補パス集合  $SP1^k$  と  $SP2^k$ 、 $SP1^k$  と  $SP2^k$  に含まれる各候補パスの検知領域の情報、及び、 $M^k$  の情報を付加し、コーディネータ  $C^{k+1}$  は自身の候補パス  $P_{new}$  を除いた候補パス集合  $SP1^k$  と  $SP2^k$  を自身の候補パス集合  $SP1^{k+1}$  と  $SP2^{k+1}$  の初期値とする。

**4. 繰り返し:** 以上の動作を  $M^{k+1}$  の検知領域が要求領域  $R_Q$  を完全に被覆するまで繰り返す。

#### 4.4 分散型アルゴリズムにおける改良

分散型アルゴリズムでは候補パス集合  $SP1^k$ 、及び、 $SP2^k$  の更新が非常に重要である。メッセージ数の削減のために、コーディネータ  $C^k$  から送られた探索メッセージを受信したセンサは、自身が既に連結センサカバー  $M^k$  に含まれているならばブロードキャストしないようになっていた。しかし、応答メッセージを返したセンサの候補パスは前回の評価値に関係なく更新されてしまう。

そこで、探索メッセージを受信したセンサは、自身が既にセンサ集合  $M^k$  に含まれている場合でも探索メッセージをブロードキャストするように変更する。また、候補パス上の途中で既にセンサ集合  $M^k$  に含まれるセンサが存在する場合、候補パスの更新は候補センサから  $C^k$  までの経路ではなく、候補センサから候補パス上のセンサ集合  $M^k$  までの経路として更新する。

なお、本改良は、Gupta らのアルゴリズムにも適用可能である。そこで、本改良を加えた Gupta らのアルゴリズムを以下では「改良した Gupta らのアルゴリズム」とし、加えて提案アルゴリズムも同様に「改良した提案アルゴリズム」として 5 章で合わせて実験を行うものとする。

### 5 提案手法に対するシミュレーション

本節では、まず最初に、Gupta らの集中型アルゴリズムと提案アルゴリズムについてシミュレーションを行い、その結果を比較する。次に、分散型アルゴリズムに関して 4.4 節で述べた改良を加えた Gupta らのアルゴリズム、提案アルゴリズム、及び、改良を加えた提案アルゴリズムについてシミュレーションを行い、その結果を比較する。

#### 5.1 集中型アルゴリズムに関するシミュレーション

まず、Gupta らのアルゴリズムと提案アルゴリズムについて、集中型アルゴリズムに関するシミュレーションを行い、センサ数に対して評価を行う。

図 9、図 10、及び、図 11 は、センサ数が 2000、3000、及び、4000 における Gupta らのアルゴリズム、提案アルゴリズム、及び、Gupta らのアルゴリズムに対する局所探索により得られた連結センサカバーのセンサ数の結果を表している。このグラフにおいて、縦軸は連結センサカバーを構成するセンサ数  $m$  を表し、横軸はセンサの通信半径  $t$  を表している。なお、実験は同じセンサの配置に対して起動センサを 20 回変更して行っており、グラフの値は 20 回の平均値を表している。また、局所探索は、3-OPT で実行している。

各図ともに通信半径が小さい場合では、提案手法は Gupta らの手法と変わらないが、通信半径が大きい場合では連結センサカバーを構成するセンサ数

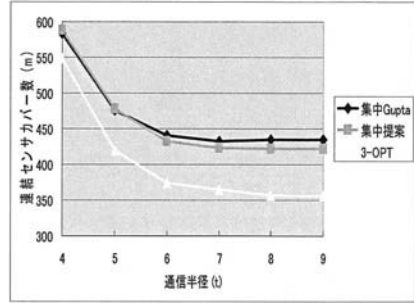


図 9: 集中型アルゴリズムの比較 (センサ数 2000)

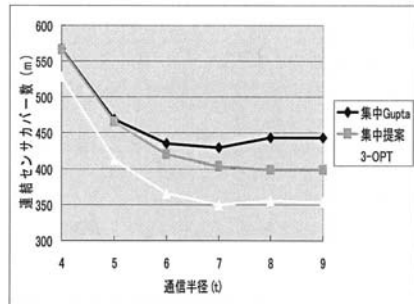


図 10: 集中型アルゴリズムの比較 (センサ数 3000)

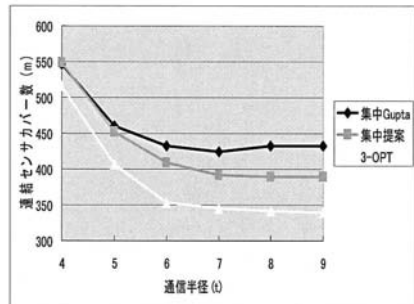


図 11: 集中型アルゴリズムの比較 (センサ数 4000)

が改善されており、局所探索によって求めた値に近づいていることがわかる。

#### 5.2 分散型アルゴリズムに関するシミュレーション

以下では、分散型アルゴリズムに関して改良を加えた Gupta らのアルゴリズム、提案アルゴリズム、及び、改良を加えた提案アルゴリズムのシミュレーションを行い、Gupta らのアルゴリズムの結果と比較を行う。

図 12、図 13、及び、図 14 は、センサ数が 2000、3000、及び、4000 の場合に Gupta らのアルゴリ

ズム, 改良を加えた Gupta らのアルゴリズム, 提案アルゴリズム, 及び, 改良を加えた提案アルゴリズムにより得られた連結センサカバーのセンサ数を表している。縦軸は連結センサカバーを構成するセンサ数  $m$  を表し, 横軸はセンサの通信半径  $t$  を表している。なお, 実験は同じセンサの配置に対して起動センサを 100 回変更して行っており, グラフの値は 100 回の実験結果の平均値である。

本結果を見ると, Gupta らのアルゴリズムよりも提案アルゴリズムの結果のほうが連結センサカバー数  $m$  は改善されている。また, 改良を加えた Gupta らのアルゴリズムと改良を加えた提案アルゴリズムで通信半径  $t$  が小さいときに, 連結センサカバー数を大幅に改善することができている。これは通信半径  $t$  が小さい程最大ホップ数が大きくなるので, 改良を加えた候補パスの更新が増えたためであると考えられる。

## 6 まとめ

本研究では, 検知領域交点を考慮した連結センサカバーアルゴリズムを提案した。また, 提案アルゴリズムをシミュレーション環境に実装し, 集中型と分散型それぞれにおいて, 既知のアルゴリズムとの比較および検証を行った。

集中型アルゴリズムでは, 提案手法によって得られる連結センサカバーは, センサ数が少ない場合や通信半径が小さい場合, Gupta 手法の結果と変わらないが, センサ数が多い場合や通信半径が大きい場合, 連結センサカバー数は改善されており, 近似解に近い値を示した。

分散型アルゴリズムでは, 提案手法により得られる連結センサカバーは通信半径が小さい場合は Gupta らの手法の場合とそれほど変わらない結果であるが, 通信半径が大きい場合は, 大きく改善した。

今後の課題としては, 検知領域の異なるセンサネットワークへの対応できるように改良することや, 動的なセンサ環境に対して連結センサカバーを求める問題などが挙げられる。

## 参考文献

- [1] H. Brönnimann and M. T. Goodrich. Almost optimal set covers in finite VC-dimension. *Discrete Comput. Geom.*, 14, 1995.
- [2] H. Gupta, S. R. Das, and Q. Gu. Connected sensor cover:self-organization of sensor networks for efficient query execution. *Proc of MobiHoc 03*, 2003.
- [3] S. Slijepcevic and M. Potkonjak. Power efficient organization of wireless sensor networks. *ICC 2001. IEEE International Conference*, 2, 2001.
- [4] 森川 雅和, 鈴木 明子, 大下 福仁, 角川 裕次, and 増澤 利光. 領域被覆のためのセンサネットワークアルゴリズム. 第 2 回情報科学ワークショップ, 2006.
- [5] 茨木 俊秀. 離散最適化法とアルゴリズム. 岩波出版, 1993.
- [6] 小牧 省三. 無線 LAN とユビキタスネットワーク. 丸善株式会社, 2004.

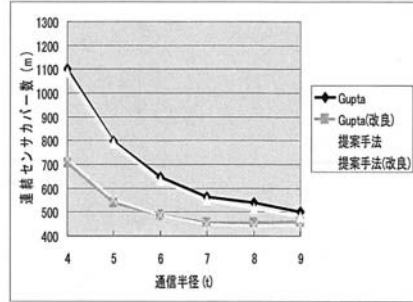


図 12: 分散型アルゴリズムの比較 (センサ数 2000)

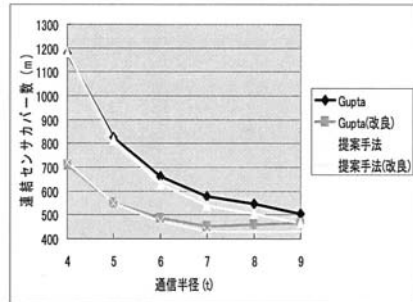


図 13: 分散型アルゴリズムの比較 (センサ数 3000)

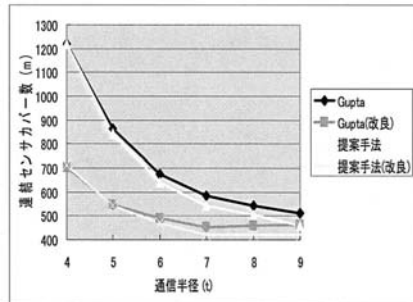


図 14: 分散型アルゴリズムの比較 (センサ数 4000)