

コストマップを用いた移動型センサノードの経路探索手法

中 宮 正 樹^{†1} 岸 野 泰 恵^{†2}
寺 田 努^{†1} 西 尾 章 治 郎^{†1}

本研究では、コストマップと呼ぶ移動コストを表す地図を用いた移動型センサノードの経路探索手法を提案する。提案手法を用いることにより、従来の研究では扱われていなかったセンサのセンシング範囲やノード移動時の障害物、ノードの移動特性といった実用上の問題を考慮した経路探索が可能となる。提案手法では、センシング領域を4つのパラメータを用いて定義し、コストマップを用いた最小コスト経路探索アルゴリズムを提案する。さらに、経路探索に広く用いられているA*アルゴリズムと提案手法をシミュレーションにより比較し、提案手法の有効性を確認した。また、提案手法を実機のセンサノードに実装し、実環境で正しく動作することを確認した。

A Route Planning Method Using *Cost Map* for Mobile Sensor Nodes

MASAKI NAKAMIYA,^{†1} YASUE KISHINO,^{†2} TSUTOMU TERADA^{†1}
and SHOJIRO NISHIO^{†1}

In this research, we propose a route planning method for mobile sensor nodes using *cost map*. The proposed method achieves a novel path planning that can solve several practical problems in previous works: the limitations of sensing area, barricades on nodes' path and restrictions on nodes' movements. We propose a method to define the sensing area as four parameters to deal with many kinds of sensors and a route planning method using *cost map*. The method can find the path that has the lowest energy consumption. Furthermore, we compared the proposed method to A*algorithm which is one well-known route planning algorithm. We also implemented prototypes of sensor nodes to verify our algorithm in the real environment.

1. はじめに

近年のネットワーク技術や無線通信技術の発展、センサデバイスの小型化および高性能化により、センサデバイス自体に無線通信機能を持たせたセンサノードによってネットワークが形成される、センサネットワークの研究がさかんに行われている¹¹⁾。最近では、センサノードにアクチュエータを搭載し、自由に移動できるようにした移動型センサノードに関する研究が注目されている⁶⁾。ノードに移動性を持たせることで、センシングデータ要求の動的な変化に対応でき、人が近づけない危険地帯や人の手が届きにくい場所などへのノード配置も可能となる。

しかしこれまで行われてきた移動型センサノードを用いた研究では、実用上起こりうる問題を扱うものは数少ない。具体的には、まずセンサのセンシング範囲が考慮されていない点あげられる。センサネットワークでは様々なセンサの利用が考えられるが、センサの種類によりセンシング可能な方向や範囲が制限される場合がある。次に、ノード移動時の障害物や地形の影響が想定されていない点である。障害物によって移動を妨げられたり、凹凸の多い道を移動することで電力消費が大きくなるが、従来の経路探索手法ではこれらの要素が考慮されていないため、障害物をうまく避けて通るような移動コストが小さい経路を得ることが難しい。さらに、これまでの研究ではノードの移動特性を考慮したものが少ない。移動型センサネットワークでは、車型や2足歩行型など様々な種類のセンサノードの使用が考えられるが、従来研究では移動特性が考慮されていないため、実際には動けない方向への移動が指示されたり、移動に要する消費電力がより低く抑えられる経路が他に存在するにもかかわらず、

†1 大阪大学大学院情報科学研究科
Graduate School of Information Science and Technology, Osaka University

†2 日本電信電話株式会社コミュニケーション科学基礎研究所
Communication Science Laboratories, Nippon Telegraph and Telephone Corporation

別の移動経路が指定されたりする可能性がある。

そこで本研究では、これらの実用上の問題を解決する手法として、コストマップを用いた移動型センサノードの経路探索手法を提案する。本研究では、実際のセンサのセンシング範囲を想定したセンシングモデルを定義し、さらに、ノードの移動特性および障害物や地形の変化を考慮した最小コスト経路探索アルゴリズムを提案する。

以下、2章で想定する環境について述べ、3章で提案手法について説明し、4章でシミュレーションによる性能評価について述べ、5章で実装とその評価について説明する。さらに6章で考察を行い、7章で関連研究を紹介する。最後に8章で結論と今後の課題を述べる。

2. 想定環境

本研究では、ビル内や地下鉄の構内などの地形が変化しない静的な環境におけるセンシングを想定し、地形は図1に示すような2次元格子状に区切られたフィールドマップとしてあらかじめ与えられているものとする。フィールドマップの1つの格子を本研究ではセルと呼び、セルごとに、凹凸の多い地面や障害物といったノードの移動に影響を及ぼす情報が、地形コストとして設定されている。提案手法は、ノード移動時における消費電力の削減を目的としており、消費電力をコストと定義し、単位はジュールである。ここで、地形コストはノードが凹凸の多い地面を通過することで何も無い地面を通過する場合よりも余分に消費する電力を示し、提案手法においては、地形コストが設定されたセルをノードが通過するとコストが加算される。特に、ノードが通過できない障害物の地形コストを無限大とする。このような状況において、1台のサーバが地形コストを含めたフィールドマップの情報およびノードに関する情報を保持しているとする。このサーバに対してユーザから単一ノードではまかなえないような複数箇所での広範囲にわたるセンシングデータの要求があり、サーバはノードごとにセンシング対象ま

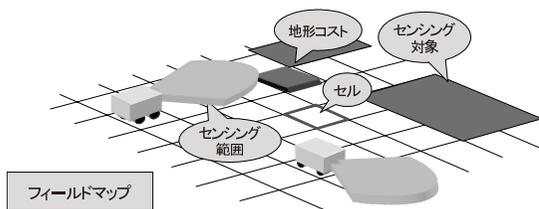


図1 想定環境
Fig.1 Assumed environment.

での経路を探索してノードに移動を指示する。

移動型センサネットワークで使用するノードの種類としては、車型ノードや2足歩行型ノードなど様々な種類が考えられる。それぞれ移動方法に特性を持ち有利な環境が異なるため、移動型センサネットワークでは同じ種類のノードを複数利用せず、様々な種類のノードを併用してセンシングすることが望ましい。また、最近では様々なセンサが日常生活で利用されており、センサネットワークにおいても、ユーザが知りたい情報により使用するセンサが異なるため、ノードに多種のセンサを搭載することが望ましい。このような観点から、本研究では様々なセンサを搭載した多種のノードを併用したセンシングを行う環境を想定する。

3. 提案手法

3.1 センシングモデル

センサは種類によってセンシング可能な範囲や方向が異なる。表1に示すように、たとえば温度や湿度はどの方向を向けて測定しても値は変わらないが、焦電型赤外線センサや距離センサは、特定方向の決まった角度の範囲内ではしか測定できない。そこで実際のセンサのセンシング範囲を、図2に示すように4つのパラメータを用いて表す。このようにセンシングモデルを定義することでセンシング可能な方向が制限されるセンサの使用に対応できる。

3.2 経路探索アルゴリズム

本研究で提案するコストマップを用いた経路探索アルゴリズムは、経路探索に広く用いられているA*アルゴリズムを基にしている。まずA*アルゴリズムについて説明し、その後提案アルゴリズムを詳細に解説する。

表1 センサ別センシング方向制限の有無
Table 1 The limits of the sensing direction of the sensors.

方向の制限	センサの種類
有	焦電型赤外線センサ, 距離センサ 照度センサ, タッチセンサ
無	温度センサ, 湿度センサ 音感センサ, 風力センサ

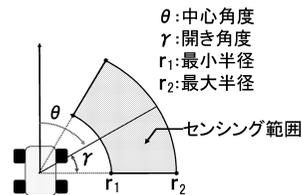


図2 センシングモデル
Fig.2 Model of sensing area.

3.2.1 A*アルゴリズム

A*アルゴリズムは、途中で通過する経路のコストを考慮して最小コスト経路の探索を行うためのアルゴリズムで、人工知能やゲームなどの分野で広く用いられている。A*アルゴリズムでは、最終状態に近づく経路を優先的に探索していくことで計算量を軽減し、解となる経路がある場合には必ず最小コストの経路が得られることが証明されている¹⁾。

A*アルゴリズム計算手順

A*アルゴリズムの計算手順について説明するために、図3に示すように、スタート地点とゴール地点が設定された2次元格子状のマップを用いる。以降の説明ではセルを特定するために、フィールドマップの列番号 x 、および行番号 y を用いてセルの位置を (x, y) のように表す。たとえば、図3に示すフィールドマップにおいてスタート地点の位置は $(3, 5)$ 、ゴール地点の位置は $(5, 2)$ となる。フィールドマップ上における複数のセルにコスト無限大または3の地形コストを設定し、ノードが地形コストの設定されたセルを通過すると移動コストに地形コストが加算される。A*アルゴリズムでは、最短経路探索を行うためにセルごとに以下の簡単な計算によりスコア (Score) を求める。

$$Score = C + H$$

C は、スタート地点から注目するセルまでの移動に要する総移動コスト、 H は、注目するセルからゴール地点まで地形コストがないと仮定した場合の移動コストである。ここで、 C, H および $Score$ はノードの消費電力を示し、単位はすべてジュールである。また経路探索の過程ではセルは以下の3つの状態に分類される。

未計算: $Score$ の計算が行われていないセル

OPEN: $Score$ の計算が行われたセル

CLOSE: $Score$ の計算が行われ、そのセルまでの最短経路が確定しているセル

次に、最短経路を探索する計算手順を図4にそって説明する。なお、この例では説明を簡単化するため、縦方向への移動と斜め方向への移動に要する移動コストを両方とも1としている。

Step1 スタート地点に隣接するセルの $Score$ を計算して OPEN 状態にする。セル $(2, 4)$ のように地形コストが設定されたセルへ移動する場合は $Score$ に地形コストを加算する。

Step2 OPEN 状態のセルの中で $Score$ が最小、かつゴール地点により近いセル $(4, 4)$ に注目して CLOSE 状態にする。

Step3 注目したセルに隣接するセルの $Score$ を計算

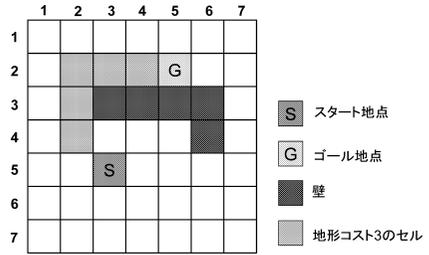


図3 2次元格子状マップ

Fig. 3 An example of 2-dimensional gridironed map.

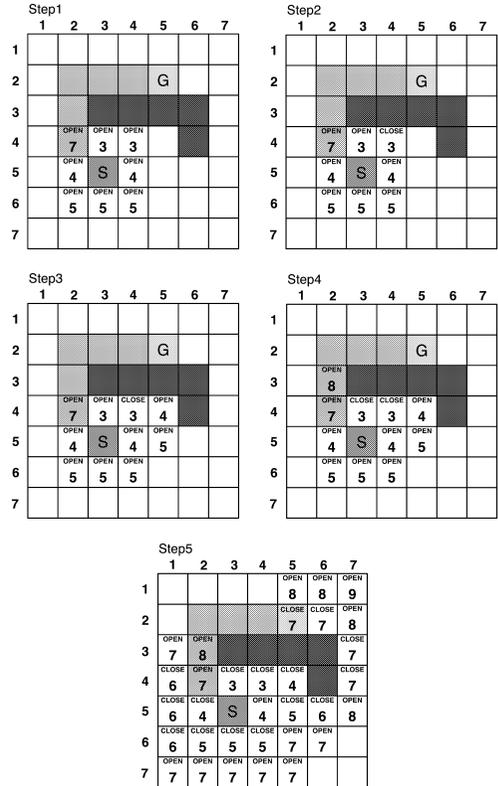


図4 A*アルゴリズム適用例

Fig. 4 An example of A* algorithm.

して OPEN 状態にする。このとき OPEN 状態にされたセルは、注目しているセルの位置を記憶する。また、すでに計算されていた $Score$ より新しく計算された $Score$ の方が小さい場合、 $Score$ と直前に注目されていたセルの位置を更新する。

Step4 OPEN 状態であるすべてのセルの中で $Score$ が最小、かつゴール地点により近いセル $(3, 4)$ に注目して CLOSE 状態とし、隣接するセルの $Score$ を計算して OPEN 状態にする。

Step5 ゴール地点が注目され CLOSE 状態となるまで $Score$ の計算を繰り返す。セルは直前に注目さ

れていたセルの位置を記憶しているため、ゴール地点からスタート地点までをたどると最短経路が発見できる。

3.2.2 コストマップを用いた経路探索アルゴリズム

本研究で提案するアルゴリズムでは、さらにノードの移動特性を考慮した最小コスト経路探索を行う。ベースコスト

経路探索を行う前にベースコストを測定する。ベースコストとは、地形の影響がまったくない状態で周辺のセルへの移動に要する実際の消費電力で、単位はジュールである。たとえば図 5 (a) に示すように、周囲 2 セルの範囲内にあるセルへ移動するのに必要な実際の消費電力を、実機のノードごとにあらかじめ測定する。なお、ノードが途中通過する経路はセルにとらわれないものとし、通過するすべてのセル座標を記憶しておく。また、図 5 (b) に示すように、ノードが斜め方向を向く状態からのベースコストも測定する。本研究では移動前、移動後の各セルにおけるノードの向きを上、下、左、右、右上、右下、左上、左下の 8 方向とする。同じ座標のセルへの移動でも移動後のノードの向きによりコストが異なるため、ベースコストは各セルごとに 8 方向ずつ測定する。各セルに到達したときにノードは 90° 方向および 45° 方向のいずれかを向くため、Score 計算には図 5 (a), (b) に示す 2 種類のベースコストをノードの向きに合わせて使い分ける。具体的には、移動前にノードが上、下、左、右方向を向いている場合は図 5 (a)、それ以外の方向を向いている場合は図 5 (b) を用いる。

コストマップ

センシングに使用するノードごとにあらかじめ測定したベースコストと、地形コストの情報を含むフィールドマップから、提案する経路探索アルゴリズムを用いてコストマップを作成し、経路の探索を行う。コストマップとは経路探索中におけるセルごとの Score を示すマップである。なお Score の算出式は A*アルゴリズムと同様であり、Score の単位はジュールである。また、提案アルゴリズムの計算過程において A*アル

ゴリズムと以下の点が異なる。本研究では、各セルにおけるノードの向きを 8 方向で区別する。つまり、経路探索の過程においてあるセルに注目するときにノードの向きも決まっており、その位置および向きからの移動に要するコストが計算される。さらに移動後のノードの向きも考慮し、1 つのセルに対して 8 方向別に Score を計算する。また、A*アルゴリズムでは注目したセルに隣接するセルの Score のみを計算するが、提案するアルゴリズムではベースコストの範囲内に含まれるすべてのセルの Score を同時に計算する。Score 計算の際、ノードの向きに応じて図 5 (a) に示す 90° 方向、または図 5 (b) に示す 45° 方向のベースコストを使い分ける。ノードはセルを移動してゴール地点に到達するが、各セル間を移動する途中の経路はセルにとらわれない。

経路探索アルゴリズム

図 6 に示すフィールドマップと、図 5 に示すベースコストを用いて、提案アルゴリズムによりコストマップを作成する手順を図 7 を用いて説明する。以降の説明では A*アルゴリズムの説明と同様にフィールドマップの列番号 x 、および行番号 y を用いてセルの位置を (x, y) のように特定する。たとえば、図 6 に示すフィールドマップにおいてスタート地点の位置は $(4, 7)$ 、ゴール地点の位置は $(9, 1)$ となる。なお、この例では説明の簡単化のため移動後のノードの向きによるコストの違いを考慮しない。

Step1 スタート地点から、ベースコストの範囲に含まれるセルの Score を計算して OPEN 状態にする。ここでスタート地点のセルを、OPEN 状態にされたセルの親セルと呼ぶ。スタート地点でのノードの向きが上向きであるため Score の計算にはベースコスト A を用いる。たとえば、セル $(6, 5)$ の Score は C がベースコストによるコスト 3 に途中通過する地形コスト 5 を加えた 8 となり、 H が 4 となるため Score は 12 となる。なお、周辺のセルへはベースコストを測定したときの経路

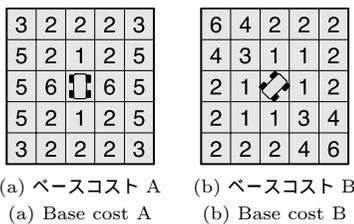


図 5 ベースコスト

Fig. 5 Base cost that direction of node is not considered.

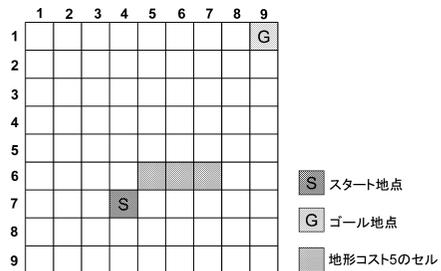


図 6 フィールドマップ

Fig. 6 An example of field map.

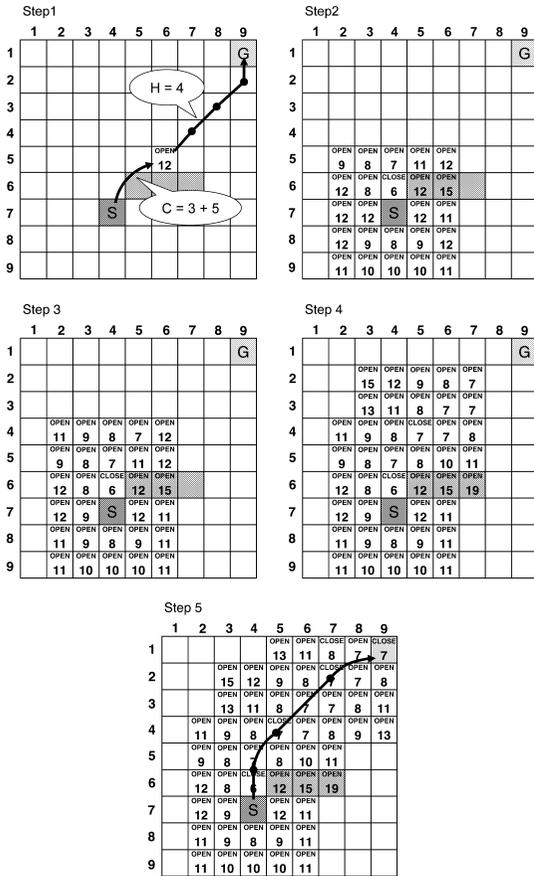


図 7 経路探索アルゴリズム適用例

Fig. 7 An example of the proposed algorithm.

を移動する。

Step2 OPEN 状態のセルの中で *Score* が最小、かつゴール地点に最も近いセル (4, 6) に注目し、CLOSE 状態にする。

Step3 注目したセル (4, 6) から、ベースコストの範囲に含まれるセルの *Score* を計算して OPEN 状態にする。スタート地点からセル (4, 6) へ移動後ノードは上方向を向いており、*Score* の計算にはベースコスト A を用いる。ここで、OPEN 状態にされたセルは、注目しているセル (4, 6) の位置とセル (4, 6) からの移動経路を記録しておく。さらに、セル (2, 8), (3, 7), (6, 8) ではセル (4, 6) から計算された *Score* の方が小さいため、*Score* と移動経路および直前に注目しているセルの位置を更新する。また、セル (4, 6) からセル (6, 4) へ移動する途中、ノードは地形コストが設定されたセル (5, 6) を通過するため、地形コストがセル (6, 4) の *Score* に加算されている。

Step4 OPEN 状態の全セルの中で *Score* が最小、かつゴール地点に最も近いセル (5, 4) に注目して CLOSE 状態にし、ベースコストの範囲に含まれるセルの *Score* を計算して OPEN 状態にする。このセルへ移動後ノードは右上方向を向いており、*Score* の計算にはベースコスト B を用いる。OPEN 状態にされたセルは注目しているセル (6, 4) の位置とセル (6, 4) からの移動経路を記録しておく。

Step5 ゴール地点が注目されて CLOSE 状態となるまで *Score* の計算を繰り返し行う。セルには、自身の *Score* が計算される直前に注目されていたセルの位置と、そのセルからの移動経路が記録されているため、ゴール地点からスタート地点までをたどると最小コスト経路が得られる。

実際の経路探索では 1 つのセルごとに 8 方向別の *Score* を計算する。たとえば Step4 においては、OPEN 状態の方向別の *Score* の中で最小となるセルの方向に注目して、そのセルからベースコストの範囲に含まれるセルの 8 方向別の *Score* を計算する。また、ユーザが要求したセンサのセンシング範囲がセンシング対象の方向を向く状態がゴールであり、ゴールが複数ある場合がある。複数あるゴールの中で最小コストになるものを配置場所とする。

4. 性能評価

提案するアルゴリズムの性能を評価するため、シミュレーションによる A*アルゴリズムとの比較を行った。なお地形コストは、壁などのノードが通過できない障害物を想定し、コストが無限大のもののみを配置した。ノードは、前進、後退、旋回を行う戦車型ノードと、前輪で角度をつけて前進、後退を行う車型ノードの 2 種類を考える。実際に LEGO 社 MindStorm¹²⁾ を用いてこれら 2 種類のノードを製作し、それぞれの 3×3 セル、5×5 セルのベースコストをあらかじめ測定し、経路探索を行った。

本研究では複数のセンサを搭載した多種ノードを複数台併用してセンシングを行う環境を想定しているが、本論文では、提案手法の有効性を明確に示すため、単一のセンサを搭載した 1 台のノードを用いてセンシングを行うものとする。指向性を有する焦電型赤外線センサを用いたセンシングを想定し、センシングモデルのパラメータを表 2 に示すように定義する。センシングモデルをもとにノードを静止させる位置およびノードの向きを決定する。ユーザは、全範囲のセンシング、および一部センシングのいずれかを指定する。たとえ

表 2 センシングモデルのパラメータ
Table 2 Parameter of sensing model.

θ	γ	r_1	r_2
0°	45°	0 [cm]	40 [cm]

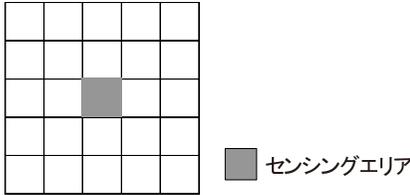


図 8 フィールドマップの一例
Fig. 8 Assumption of fieldmap.

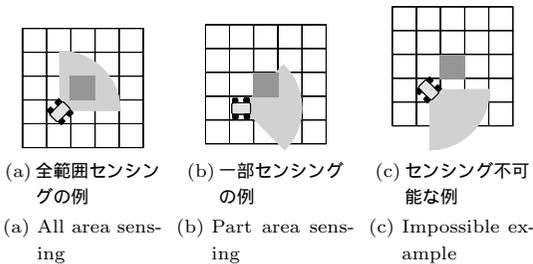


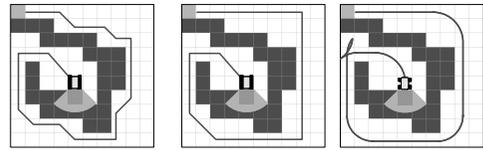
図 9 センシング例

Fig. 9 Example of sensing.

ば、図 8 に示すような範囲のセンシングを想定する。全範囲センシングが指定された場合、センシングを行うためには図 9 (a) に示すようにセンサのセンシング範囲がセンシング対象を包含する位置にノードを配置する必要がある。一部センシングが指定された場合、図 9 (b) に示すようにセンサのセンシング範囲の一部がセンシング対象にかかる位置でもセンシング可能と見なす。ノードが同一セルで静止していても、ノードの向きにより図 9 (c) に示すようにセンサのセンシング範囲がセンシング対象にまったくかからない場合はセンシングできないとする。以上のような状況を考慮したうえで提案手法では、センシング対象ごとにノードが静止するセル、およびノードの方向を最初に列挙し、それぞれをゴール地点として経路探索を行い最小コストの経路を決定する。

4.1 経路探索結果の比較

図 10 に示すフィールドマップにおいて中央の 1 セルの全範囲センシング要求を想定し、A*アルゴリズム、戦車型ノードを想定した提案アルゴリズム、および車型ノードを想定した提案アルゴリズムによる経路探索結果の一例を図 10 に示す。センサの種類として指向性を持つ焦電型赤外線センサを想定し、センシ



(a) A*アルゴリズム (b) 戦車型 (c) 車型
(a) A*algorithm (b) Tank type (c) Car type

図 10 経路探索結果例

Fig. 10 A result of path planning.

ングモデルのパラメータを表 2 に示すように設定する。フィールドの最左上のセルをスタート地点としてノードの向きは右向きとし、センサのセンシング範囲が中央のセンシング対象セルをすべて包含する位置およびノードの向きをゴール地点とする。

A*アルゴリズム (図 10 (a)) はゴール地点までの最短経路を求めるが、最短経路が複数ある場合、ゴール地点により早く近づく経路を優先的に選択するため、このように方向転換の回数が多い移動経路が得られることがある。しかし戦車型ノードは方向転換することに停車、旋回、前進を行うため、方向転換が少ない場合に比べて消費電力が増加したり、ノードの移動誤差により指定された経路から外れやすくなったりする。提案アルゴリズムで戦車型ノードを想定した結果 (図 10 (b)) では A*アルゴリズムの結果と比べて、方向転換の回数が少ない経路が探索されていることが分かる。これは、提案アルゴリズムでは経路探索前に実際のベースコストを測定しており、方向転換に消費する電力を考慮した経路探索を行うためである。車型ノードを想定した場合 (図 10 (c))、広い道での曲がり角では、緩やかな曲線を描きながら 90° 右方向へ曲がっているが、狭い道では、いったん曲がり角を通過して停車後、後退しながら進行方向に向き、前進するという経路が選択されている。これは、車型ノードが進行方向から見て 90° 方向へは移動できないという移動特性が考慮されていることを意味する。

4.2 フィールドマップの広さの変化による消費電力の比較

正方形のフィールドマップの広さを 60 セル四方から 100 セル四方まで 10 セル間隔で変化させてシミュレーションを行った。各フィールドマップの広さごとに 10 回ずつシミュレーションを行い、平均値について比較評価を行った。地形コストは、5×5 セルおよび 10×10 セルの正方形の地形コストを図 11 に示すようにランダムに配置した。フィールドの最左上のセルをスタート地点としてノードの向きは右下向きとし、最右下セルの全範囲センシングが要求されている

とした．センサの種類は前節と同様に焦電型赤外線センサを想定し，センシングモデルのパラメータを表 2 に示すように設定した．図 12 に戦車型ノードを想定したときのシミュレーション結果を示す．

提案手法はベースコストの広さにかかわらずつねに A*アルゴリズムの結果より小さい消費電力でセンサノードがゴール地点に到達することが分かる．これは，提案アルゴリズムで得られた経路が A*アルゴリズムで得られる経路よりも方向転換の回数が少ないためである．

図 13 に車型ノードを想定したときのシミュレーション結果を示す．戦車型を想定したときと同様，提案アルゴリズムを用いた場合，フィールドの広さにかかわ

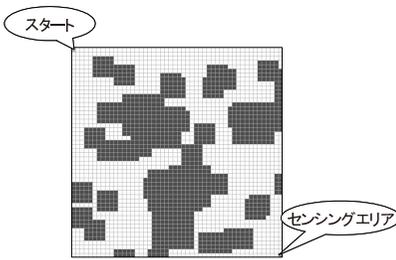


図 11 ランダム配置例

Fig. 11 An example of geographical feature arranged at random.

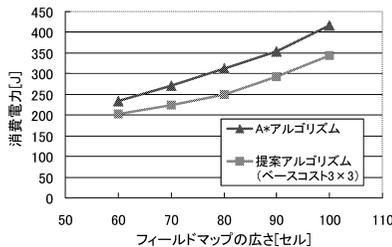


図 12 フィールドマップの広さ変化による目的地までの消費電力 (戦車型ノード)

Fig. 12 Power consumption vs. width of field maps (Tank type node).

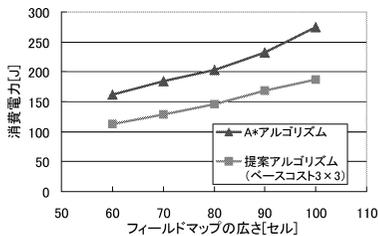


図 13 フィールドマップの広さ変化による目的地までの消費電力 (車型ノード)

Fig. 13 Power consumption vs. width of field maps (Car type node).

らず A*アルゴリズムよりも小さい消費電力でゴール地点に到達することが分かる．また，戦車型ノードを想定したときのシミュレーション結果に比べ，A*アルゴリズムとの差が大きい．これは，前節で述べたように，車型ノードは進行方向から見て 90° の方向へ移動することは車輪の構造上不可能なためである．A*アルゴリズムでは移動制約が考慮されていないためにそのような経路が選択される場合が生じ，この場合車型ノードは 1 度離れた位置へ移動してから 90° の方向へ移動するため移動コストが増加する．

4.3 地形コスト配置の変化による消費電力の比較
次に，地形コスト配置の変化による消費電力の違いを調べた．ノードは戦車型ノードと車型ノードの両方を用いることを想定し，以下に示す 3 通りの地形コスト配置に対して A*アルゴリズムと提案手法を用いた経路探索を行った．

- 正方形の地形コストをランダムに配置 (図 11)
- 迷路 (図 14)
- 筆者らの所属する研究室内 (図 15)

図 11, 14, 15 に示すように，フィールドの最左上をスタート地点としてノードの向きは下向きとし，最右下セルの全範囲センシングが要求されているとした．センサの種類は前節と同様に焦電型赤外線センサを想定し，センシングモデルのパラメータを表 2 に示すように設定した．

表 3 は戦車型ノード，車型ノードを想定したときの

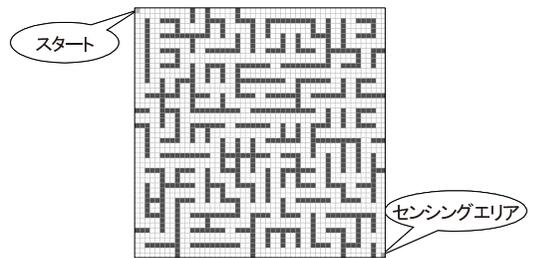


図 14 迷路状配置例

Fig. 14 An example of field map assumed our laboratory.

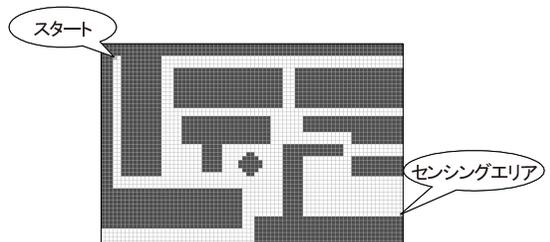


図 15 研究室内

Fig. 15 An example of field map assumed our laboratory.

表 3 地形コスト配置の変化による消費電力
Table 3 Power consumption vs. barricade arrangements.

		ランダム	迷路	研究室
戦車型	A*	416.8 [J]	493.9 [J]	250.6 [J]
	3 × 3	343.7 [J] (18%)	416.4 [J] (16%)	228.4 [J] (8.9%)
	5 × 5	334.1 [J] (20%)	412.6 [J] (17%)	227.4 [J] (9.3%)
車型	A*	251.1 [J]	366.0 [J]	148.1 [J]
	3 × 3	164.1 [J] (35%)	225.9 [J] (38%)	117.0 [J] (21%)
	5 × 5	151.1 [J] (40%)	194.8 [J] (47%)	108.1 [J] (27%)

表 4 ベースコストサイズの変化による消費電力改善率
Table 4 Power consumption vs. barricade arrangements.

	ランダム	迷路	研究室
戦車型	2.8%	0.9%	0.4%
車型	7.7%	14%	7.6%

経路探索結果を示し、ベースコストサイズ 3 × 3 セル、5 × 5 セルの結果における括弧内の値は A*アルゴリズムの結果に対する改善率を示す。

ノードの種類にかかわらず、3 通りすべての地形コストの配置において A*アルゴリズムの結果より小さい消費電力でセンサノードがゴール地点に到達しており、センシングを行う環境にかかわらず提案手法が有効に動作することが分かる。また、車型ノードの結果は、戦車型ノードの結果よりも地形コストの配置にかかわらず改善率が高い。前節で述べたように、A*アルゴリズムを用いて経路探索を行う場合、方向転換の回数が多い移動経路が得られる場合がある。その際、戦車型ノードは 1 回の旋回で方向転換ができるのに対し、車型ノードは前進、後退を数回繰り返す必要がある場合があり、戦車型ノードよりも 1 回の方向転換で余分に消費する電力が大きいためである。

4.4 ベースコストサイズの検討

さらに、ベースコストサイズの変化による目的地までの消費電力の変化を調べた。表 4 はベースコストサイズを 3 × 3 セルから 5 × 5 セルへ変化させたときの目的地までの消費電力の改善率を示す。

車型ノードでは、3 通りすべての地形コストの配置においてベースコストサイズが大きくなることで消費電力が改善されている。これは、ノードが移動しながら方向を変更する際に、5 × 5 セルのベースコストを利用して得られた経路が、3 × 3 セルのベースコストを利用して得られた経路と比べてより滑らかな経路が選択されるためである。さらに車型ノードでは、迷路状に地形コストを配置した場合、他の配置と比べて改善率がより大きくなっている。これは、この配置では他の配置よりもノードが曲がる回数が増加するためである。

戦車型ノードでは、地形コストがランダムに配置さ

れている場合 2.8%改善されるが、車型ノードの結果と比較して、ベースコストサイズを大きくすることによる消費電力の改善はあまりみられない。これは、戦車型ノードは旋回を行うことでノードの向きを変更できるため全方向への移動が容易なためである。

5. 実装

5.1 移動型ノード

実証実験には米 LEGO 社 Mind Storm RCX2 を使用した。RCX2 は 8 bit CPU を内蔵したコントローラであり、PC 上で実装したプログラムを赤外線通信により送信できる。製作した戦車型ノードは左右のキャタピラをそれぞれ別のモータで制御し、プログラミングにより前進、後退、旋回の動作と継続時間を指定できる。図 16 に、製作した戦車型ノードを示す。ノードの大きさは縦 16 cm × 横 15 cm である。ベースコストのセルの 1 辺の長さを 20 cm とし、あらかじめベースコストを測定した。

5.2 実証実験

製作した戦車型ノードを用いて、屋内での実証実験を行った。ノードの移動を妨げる机や椅子、壁などの障害物をコスト無限大の地形コストとして設定してフィールドマップを作成した。フィールドマップ上のスタート地点のノードの向きを上向きとし、最右上セルの全範囲センシングが要求されているとした。また焦電型赤外線センサを想定し、センシングモデルのパラメータを表 2 に示すように設定した。ベースコストを用いて提案アルゴリズムと A*アルゴリズムによる経路探索を行い、得られた経路に沿ってノードを移動させた。

経路探索の結果と、ノードが実際に通過した経路を図 17、図 18 に示す。点線が経路探索の結果、実線がノードの移動した経路である。経路探索結果において提案アルゴリズムと A*アルゴリズムにより得られた経路は、移動が隣接するセルに限定されている場合の最短経路であるが、提案アルゴリズムにより得られた経路はゴール地点に到達するまでに旋回を 4 回行うのに対して、A*アルゴリズムにより得られた経路は旋

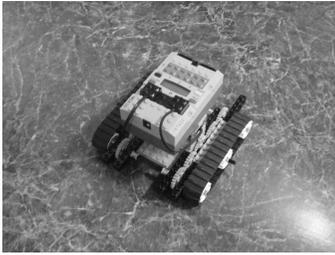


図 16 戦車型移動ノード
Fig. 16 Tank type node.

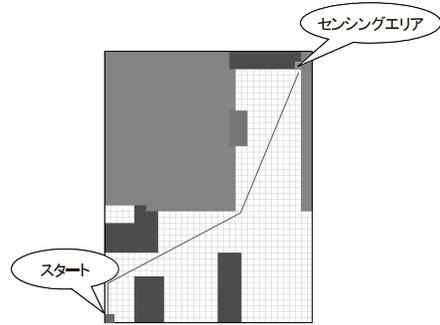


図 19 実装結果 (A*アルゴリズム)
Fig. 19 The shortest path.

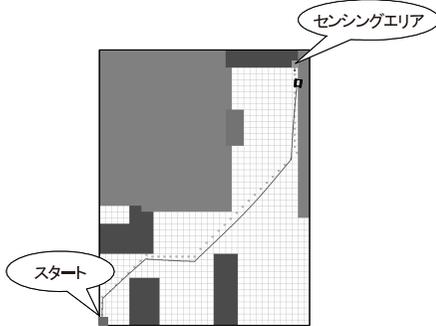


図 17 実装結果 (提案アルゴリズム)
Fig. 17 Mounting result by the proposed algorithm.

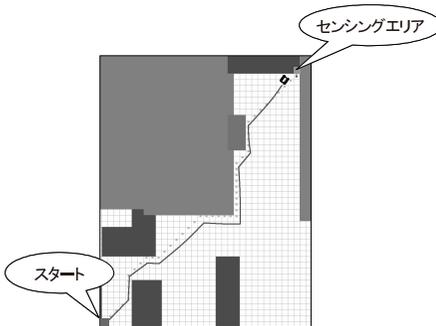


図 18 実装結果 (A*アルゴリズム)
Fig. 18 Mounting result by A* algorithm.

回を 9 回行う。これは、A*アルゴリズムではノードが旋回するのに要する消費電力を考慮せず、単に最も最初にゴール地点に到達した経路を出力するため、ノードが旋回する回数が多い経路が選択される可能性があるためである。それに対して提案アルゴリズムでは、ノードの旋回による電力消費を考慮するため、より旋回する回数の少ない経路が選択される。また、図 19 はスタートからゴールまでの最短経路を示している。提案アルゴリズムよりもさらに旋回数が少ない経路であるが、最短経路を見出すには人が作図して求めなければならない、労力がかかる。

表 5 は、シミュレーションと実証実験から得られ

表 5 消費電力比較

Table 5 Comparison of power consumption.

	実装	シミュレーション	誤差
提案アルゴリズム	115.5 [J]	114.9 [J]	0.6 [J]
A*アルゴリズム	133.4 [J]	133.1 [J]	0.3 [J]
最短経路	108.7 [J]	102.7 [J]	6.0 [J]

た、ゴール地点までの移動に要する A*アルゴリズムと提案アルゴリズム別の消費電力、およびノードを手動で操作して最短経路を移動させたときの消費電力を示している。実験で使用した戦車型ノードの移動誤差により、図 17、図 18 に示すように指定した経路から多少の位置的なずれは生じたが、シミュレーション結果と実装結果の誤差が、提案アルゴリズムでは約 0.5%、A*アルゴリズムでは約 0.2%とごくわずかであり、実環境においても提案するアルゴリズムが有効に動作することが分かる。

6. 考 察

6.1 ベースコストの測定

現在ベースコストは、周辺のセルへ手動でノードを移動させて測定している。今後より容易にベースコストを測定する方法を検討する予定である。具体的には、ノードのベースコストを自動的に測定できるプログラムの作成を検討している。

移動型センサネットワークでは多種のノードを複数台併用してセンシングすることが望ましい。現在経路探索に使用するベースコストのセルの大きさは、ノードのサイズや形状、タイヤが装着されている位置などから判断して、ノードごとに最適な大きさが異なる。そのため、ベースコストのセルの大きさが異なるノードを併用しなければならない状況が起こりうる。今後はベースコストのセルの大きさを小さく区切り、ノードの細かい動きに対応できるようにする予定である。

6.2 ベースコストサイズの検討

ベースコストは、ベースコストサイズを広げるにつれてより多くの時間と手間を要する。また、提案手法において *Score* を計算する際、ベースコストの範囲に含まれるすべてのセルの *Score* を同時に計算するため、ベースコストサイズを大きくすると計算量が指数的に増加する。ノードの種類によっては戦車型ノードのようにベースコストサイズの変化による影響が少ない場合もあるため、センシング対象や目的に応じてベースコストサイズを十分に検討する必要がある。

6.3 地形コストの設定

本研究ではノードの移動の妨げとなる障害物や地形の変化を地形コストとして数値化している。具体的には、平坦な地面を通過する際と凹凸などの障害がある地面を通過する際の消費電力量の差分を地形コストとして設定する。多種のノードを併用してセンシングを行う場合、実際にはノードの種類によって地形コストがどの程度影響を与えるかが異なる。たとえば戦車型ノードは地面の凹凸や段差の多い環境下でも他のノードに比べて地形の影響が少ない。そこで地形の種類として設定しておき、経路探索の際に使用するノードによって地形コストの値を変化させることが有効である。

さらに、現在のシミュレーションにおいてはノードが通過できない障害物のみを想定しているが、今後凹凸のある地面などノードが通過できる障害物の地形コストを測定し、評価を行う予定である。

また、現在は移動コストに単純に地形コストを加算するアプローチをとっているが、実際の環境では凹凸のある地面などを通過することで消費電力が増大するだけでなく、ノードの向きが変わるなどの影響も考えられる。さらに傾斜のある地面の場合、ノードが通過する方向によりコストが変化するため、方向別にコストを設定することを考えている。

6.4 計算量オーバーヘッド

A*アルゴリズムおよび提案手法の計算量はセルの *Score* を計算する回数に集約される。ここで、A*アルゴリズムと 3×3 セルのベースコストを用いた提案手法では、計算量に関して以下の2点が異なる。両手法では、*Score* 計算済みのセルの中から最小 *Score* のセルに注目して CLOSE 状態とし、注目したセルを中心としたベースコストの範囲内に含まれるセルの *Score* 計算を行う。提案手法は1セルをさらに8方向別に区別するため、ゴール地点が注目されるまでにA*アルゴリズムと比較して最大8倍のセルに注目する。次に、A*アルゴリズムは周囲8セル分の *Score* 計算を行うのに対し、提案手法は各セルをさらに8方向で区別す

表6 フィールドマップサイズによる *Score* 計算回数
Table 6 *Score* calculation steps vs. width of field maps.

マップサイズ	提案手法	A*アルゴリズム	増加率
100	1,328,854 [回]	23,295 [回]	57 [倍]
90	1,077,826 [回]	16,789 [回]	64 [倍]
80	751,943 [回]	13,778 [回]	55 [倍]
70	624,694 [回]	10,767 [回]	58 [倍]
60	485,511 [回]	7,991 [回]	61 [倍]
50	305,305 [回]	5,338 [回]	57 [倍]

表7 マシンスペック
Table 7 Machine spec.

マシン名	DELL PowerEdge 1950
CPU	Intel Xeon 5140 (2.33 GHz)
メモリ	DDR2 4 GB

るため最大64セル分の *Score* を計算する。以上の2点から、提案手法を適用した場合A*アルゴリズムと比較して最大64倍の計算量を要すると考えられる。

計算量について評価を行うため、迷路状に地形コストが配置された100~60セル四方のフィールドにおいて、1台の車型ノードでセンシングを行う状況を想定したシミュレーションを行った。経路探索の過程で要した *Score* 計算回数の結果を表6に示す。提案手法はA*アルゴリズムと比較して55~64倍の計算量となり理論上の64倍より小さい。これは、あるセルへ移動する際、移動後のノードの向きによって移動経路上に障害物が存在する場合があります、*Score* 計算を行わないためである。また、物理的な計算時間について評価するため、表7に示す端末を用いて、同様に迷路状に地形コストを配置した100セル四方のフィールドを想定したシミュレーションを行い、要する実時間を比較した。A*アルゴリズムは0.7秒なのに対して提案手法では約5秒を要する。ここで、提案手法では1カ所のセンシングエリアに対しノードがセンシングを行う位置および方向の候補は複数あり、最小コストでセンシング領域に到達する経路を求めするために、すべての候補セルをゴール地点として経路探索を行う必要がある。たとえば、ノードの向きにかかわらず全方位をセンシングでき、センシング可能な範囲が半径3セルの温度センサを用いてフィールド最右下の2セル四方の領域を全範囲センシングする場合、センシング領域の周囲5セルであればどの向きでノードが静止しても全範囲センシングできるため、周囲5セルがゴール地点の候補となる。この場合、経路探索が終了するまでの計算時間は25秒となるが、シミュレーションの結果からセンシング対象までのノードの移動時間が280秒であることから、実際のアプリケーションへの影響は十分に小さいといえる。

6.5 実機の移動誤差

実証実験では、地形コストが設定されたセルを通らないにもかかわらず、指定した移動経路と実際にノードが移動した経路の間に誤差が生じた。今後は GPS や加速度センサ、ジャイロセンサを用いてノードの位置情報を取得し、指定した経路から一定距離離れば、その位置から再度経路探索を行い、経路を修正しながら移動を行うといった手法を検討している。今回の実証実験では、実際にノードが移動した経路と指定した経路のずれが、A*アルゴリズムと提案アルゴリズム間でほとんど差がなかったが、方向転換の回数が増えるにつれて移動誤差が大きくなると考えられ、より方向転換の回数が少ない経路を発見する提案アルゴリズムがさらに有効であるといえる。

6.6 複数ノードの配置

本研究では複数のノードを併用してセンシングを行うことを想定しているが、今回提案した手法では、1つのノードの配置場所の決定と移動経路の探索にとどまっている。しかし、複数箇所のセンシング要求に対して複数ノードを用いて強制的にセンシングする場合、まずセンサの種類やノードに搭載されたセンサのセンシングモデルを考慮したノード配置候補を複数パターン列挙し、その配置候補ごとに提案アルゴリズムによる経路探索を行う。その際、ノード間の衝突を考慮した経路探索手法が必要となる。そこで、セルごとに *Score* を計算すると同時に、スタート地点から要する移動時間も計算し、衝突を検出する。衝突を回避する手段として、どちらかのノードをそのセルの直前で一定時間停止させるか、次に最小コストとなる経路を探索する方法を検討している。

6.7 データ転送

センサネットワークでは、得られたセンサデータをデータセンタへ転送する必要があるが、今回の提案手法ではデータ転送については考慮していない。特に移動型ノードを用いたセンサネットワークでは、ノードが移動することでアドホックネットワークを形成しマルチホップ通信を利用して転送する手法が研究されている。我々の研究グループでは現在、最も簡単にデータ転送を実現する手法として、データセンタを移動目的の1つと設定することを検討している。

6.8 動的な環境への対応

本研究では、ビルの屋内や地下鉄構内など静的な環境を想定している。しかし、災害時の利用を想定する場合、屋内であっても環境が動的に変化する。このような動的な環境に対応する方法として、障害物を検出する赤外線センサおよび小型カメラをノードに搭載し、

経路上に障害物を発見した場合フィールドマップの地形コストの情報を更新して再度経路探索を行う手法を検討している。時間帯によって障害物の移動パターンが決まっている場合には、経路探索の過程でノードが移動する時間を同時に計算することで対応できる。さらに、センシング対象が移動する場合も、赤外線センサおよびカメラなどにより追跡し、新たにセンシング対象を設定する必要がある。

7. 関連研究

これまでの移動型センサネットワークに関する研究としては、複数のノードが協調して様々な状況に応じたルーティングを行うことを目的とした RAMOS⁷⁾ や、ノードを障害物が存在する未知の空間に配置することを目的とした研究³⁾ が行われている。また、固定型ノードと移動型ノードを併用し、移動型ノードがデータセンタまでの経路を動的に構築する研究⁸⁾ や、固定型ノードが移動型ノードの道案内の役割を果たしてセンシングを行う研究⁹⁾ がある。さらに、センシングカバレッジの拡大を目的とした研究¹⁰⁾ や、ユーザのセンシング要求の内容を重視し、アプリケーションの期待するセンサノード配置を実現する研究⁵⁾ も行われている。

しかしこれらの研究では、実際に移動型ノードを移動させる方法について言及しているものは少なく、RAMOS など実機を開発している研究においても、複数の移動モデルへの対応や地形コストの影響を考慮しているものはない。さらに、実際のセンサのセンシング範囲を考慮したノード配置が実現されている研究も数少ない。したがって、移動型センサネットワークが実用化される際には、提案方式のように複数種類の移動型ノードを統一的に取り扱え、さらにセンサの種類によるセンシング可能な範囲の違いを考慮できる手法が必要となる。

Chang²⁾ らの提案した手法では、モバイルアドホックネットワークにおいてビルなどの物理的な障害物の転送妨害によるメッセージ転送ロスを軽減する通信経路の探索を目的とし、フィールドを六角形のセルに区切り、目的地までの通信経路を探索する。しかし、Chang らの手法では障害物の形状によっては経路が見つからない場合があるのに対し、提案手法は A*アルゴリズムの特性から実現可能解が存在すれば必ず見つかることが保証されている。仮に、フィールドマップを六角形をベースにする場合も、ベースコストを六角形の形状に合わせて測定することで対応可能である。また、提案手法は各セルごとにノードの向きを区別し

てコストを計算することでセルを四角形にすることによる制限が緩和できている。

さらに、ノードが人や動物に寄生して移動するという新しい移動手法を提案し実際に実機の製作を行っている例として、Parasitic Mobility⁴⁾があげられる。しかし Parasitic Mobility は寄生対象がつねに存在する環境を想定しており、人や動物が近づけない危険地帯へのノードの配置は不可能であるため、移動型センサネットワークの特徴を活かしきれていない。

8. 結 論

本研究では、従来の移動型センサネットワークにおける実用上の問題を解決するために、コストマップを用いた移動型センサノードの最小コスト経路探索手法を提案した。提案手法では、センサのセンシング範囲を想定したセンシングモデルを定義することでセンシング可能な方向の制限に対処する。また、ノードが移動するときの障害物の存在や地形の変化、およびノードの移動特性を考慮した経路探索アルゴリズムを用いているため、現実世界におけるセンシングに適した経路探索が行える。本研究では、経路探索に広く用いられている A*アルゴリズムとの比較を行い、提案手法の有効性を確認した。さらに提案手法を実機の移動型センサノードに実装し、実環境で正しく動作することを確認した。

今後は、衝突などを考慮した複数ノードの効率的な配置と、センサデータの条件を細かく指定できる移動型センサネットワークにおける問合せ言語の作成を行う予定である。

謝辞 本研究の一部は、文部科学省科学研究費補助金基盤研究 (A ¥17200006) の研究助成によるものである。ここに記して謝意を表す。

参 考 文 献

- 1) Bourq, D.M. and Seemann, G.: *AI for Game Developers*, O'Reilly (July 2004).
- 2) Chang, C. and Tu, S.: Obstacle-Free Geocasting Protocols for Single/Multi-Destination Short Message Service in Ad Hoc Networks, *Wireless Networks*, Vol.9, No.2, pp.143–155 (2003).
- 3) Howard, A., Mataric, M.J. and Sukhatme, G.S.: An incremental self-deployment algorithm for mobile sensor networks, *Autonomous Robots*, Vol.13, No.2, pp.113–126 (2002).
- 4) Laibowitz, M. and Paradiso, J.A.: Parasitic mobility for pervasive sensor networks, *Proc. 3rd International Conf. on Pervasive Comput-*

ing (PERVASIVE 2005), Munich, pp.255–278 (May 2005).

- 5) 村瀬正名, 西尾信彦, 徳田英幸: 引力・斥力モデルに基づいたセンサノードの動的再配置手法, 情報処理学会システムソフトウェアとオペレーティングシステム研究会論文集, Vol.92, pp.31–38 (Feb. 2003).
- 6) Sibley, G.T., Rahimi, M.H. and Sukhatme, G.S.: Robomote: A tiny mobile robot platform for large-scale ad-hoc sensor networks, *Proc. IEEE International Conf. on Robotics and Automation (ICRA 2002)*, Washington, pp.1143–1148 (May 2002).
- 7) Tobe, Y. and Suzuki, T.: WISER: Cooperative Sensing Using Mobile Robots, *Proc. International Workshop on Heterogeneous Wireless Sensor Networks (HWISE 2005)*, Fukuoka, Japan, Vol.2, pp.388–392 (July 2005).
- 8) Treeprapin, K., Kanzaki, A., Hara, T. and Nishio, S.: A Mobile Sensor Control Method for Sparse Sensor Networks, *Proc. ACM Symposium on Applied Computing (SAC 2007)*, Seoul, pp.891–895 (Mar. 2007).
- 9) Verma, A., Sawant, H. and Tan, J.: Selection and Navigation of Mobile Sensor Nodes Using a Sensor Network, *Proc. 3rd International Conf. on Pervasive Computing and Communications (PERCOM 2005)*, Kauai island, pp.41–50 (Mar. 2005).
- 10) Wang, G., Cao, G. and Porta, T.L.: Movement-assisted sensor deployment, *Proc. Conf. on Computer and Communications Societies (INFOCOM 2004)*, Hong Kong, Vol.23, No.1, pp.2469–2479 (Mar. 2004).
- 11) Crossbow. <http://www.xbow.jp/>
- 12) MindStorms. <http://mindstorms.lego.com/>

(平成 19 年 4 月 4 日受付)

(平成 19 年 12 月 4 日採録)



中宮 正樹 (学生会員)

2006 年大阪大学工学部電子情報エネルギー工学科情報システム工学科目卒業。現在、同大学院情報科学研究科マルチメディア工学専攻博士前期課程に在籍。センサネットワークに興味を持つ。

クに興味を持つ。



岸野 泰恵 (正会員)

2002年大阪大学工学部卒業。2004年同大学院情報科学研究科博士前期課程修了。2007年同研究科博士後期課程修了、日本電信電話株式会社入社。博士(情報科学)。ユビキタスコンピューティング、ヒューマンインタフェースに関する研究に従事。



寺田 努 (正会員)

1997年大阪大学工学部情報システム工学科卒業。1999年同大学院工学研究科博士前期課程修了。2000年同大学院工学研究科博士後期課程退学。同年より大阪大学サイバーメディアセンター助手。2005年より同講師。2007年神戸大学大学院工学研究科准教授。現在に至る。2002年より大阪大学院情報科学研究科マルチメディア工学専攻助手、2005年より同講師を併任。2004年より特定非営利活動法人ウェアラブルコンピュータ研究開発機構理事、2005年には同機構事務局長を兼務。工学博士。アクティブデータベース、ウェアラブルコンピューティング、ユビキタスコンピューティングの研究に従事。IEEE、電子情報通信学会、日本データベース学会、ヒューマンインタフェース学会の各会員。



西尾章治郎 (正会員)

1975年京都大学工学部数理工学科卒業。1980年同大学院工学研究科博士後期課程修了。工学博士。京都大学工学部助手、大阪大学基礎工学部および情報処理教育センター助教授、大阪大学大学院工学研究科情報システム工学専攻教授を経て、2002年より大阪大学大学院情報科学研究科マルチメディア工学専攻教授となり、現在に至る。2000年より大阪大学サイバーメディアセンター長、2003年より大阪大学大学院情報科学研究科長、その後2007年より大阪大学理事・副学長に就任。この間、カナダ・ウォータールー大学、ビクトリア大学客員。データベース、マルチメディアシステムの研究に従事。現在、Data & Knowledge Engineering等の論文誌編集委員。本会理事を歴任。電子情報通信学会フェローを含め、ACM、IEEE等8学会の各会員。