

# 無線計算機群から構成される形状変化が可能なインタフェースに向けた形状推定アルゴリズム

後藤俊樹<sup>1</sup> 門本淳一郎<sup>1</sup> 入江英嗣<sup>1</sup> 坂井修一<sup>1</sup>

**概要:** 近年、小さな要素で大きな物体を構成する形状変化型のユーザインタフェースの研究が数多く行われている。これらの研究と半導体技術の微細化に伴い、無線通信で繋がる小型計算機が複数で協調しながら形状変化によりインタラクションを実現する展望が示されている。この展望の実現に向けて、複数の計算機同士の間での通信により計算機全体の形状推定を行うことが問題となる。膨大な数の計算機が通信により座標を推定するため、より近接した少数のノード同士のみで通信を行うことが望ましい。最も理想的には隣接した素子のみで形状推定を行うことが求められるが、このようなモデルでは全てが同一距離であるため、ネットワークグラフで物理位置推定を行うことに等しい。しかし、ネットワークグラフから物理配置を求める問題は NP 困難な問題であることが知られている。数 mm オーダーの素子から数 m オーダーのシステムを 3 次元的に作ることも考えられるため、これは十分ではない。本研究では物理配置に制約を加えることで、隣接するノードの情報のみを用い、初期化を  $O(n)$  追加や切断を  $O(1)$  で物理位置推定を可能にする手法を提案する。位置推定を行うことでユーザとのより直感的なインタラクションや、高スループットなネットワーク構築が実現できると期待される。提案するアルゴリズム検証のため形状自在ディスプレイをプロトタイプとして作成し、座標推定を実演した。

## 1. はじめに

物体とのより直感的なインタラクションを実現させる取り組みとして、コレクティブな物体で構成される形状変化型のユーザインタフェースの研究が数多く行われている [18]。このようなインタフェースでは data-physicalization における情報伝達 [2] のような、情報世界の可視化を物理世界で表現することを可能にする。このインタフェースを、モバイル型の端末といった持ち運ぶ類のものから、家具や置物といった一般的には静物であるようなものまで様々な種類の物体に適用することが提案されている [1, 4, 7, 10, 14, 15]。

こうした研究の中でコレクティブな計算機の形状変化によってインタラクションを実現するビジョンが示されている [11]。このビジョンの中で、コンピュータ同士は無線通信で接続された、複数の超小型計算機が協調することで全体のシステムを構成している。無線通信であるがゆえに、結合や分離、変形が容易に可能で、また細粒度かつ制約の少ない形状変更を実現している。またこれらの物体同士が相対位置を把握することで、システム構成要素の形状推定を実現する。これにより、ユーザと計算機との形状変化を介した直感的なインタラクションを実現する。

こうしたビジョン実現に向け、無線通信を行う複数の物体間でどのように形状推定を行うかが問題になる。形状推定はロボットをはじめとする多数の物体間の連携に重要な問題であるため、様々な手法が研究されてきた [3, 6, 12, 13, 16, 17]。しかし、これらの研究は、場所による制限や、誤差、実現のための回路面積の増大、外部機器の設置コスト、一つの素子あたりのコストなど小型計算機の集合体と相性が悪い。またこれらの研究は形状推定のための信号を実際には送る必要のない多数の計算機に対して送信している。しかし、小型計算機を持つ計算リソースの制限や、大量の計算機同士の衝突などの影響を踏まえると、近接した少量の計算機と無線通信することが望ましい。

この論文では新しいモデルとその方法を提案する。より少ない計算機間での通信を考えるため、隣接したノード間のみでの通信を行うモデルを考える。このモデルでは、隣接したノード間の無線通信において、どの方向から来たのかといった情報は捨象され、かつ隣接しているという条件から距離が一定であるため通信強度の情報を用いることができない。従って、物理位置推定を行う際に用いることができる情報は隣接情報のみを記録したネットワークグラフのみとなる。このため、上記手法で問題となっていた通信強度等の誤差の影響は受けず、非常に高い精度での形状推定ができることが期待される。しかし、その単純性ゆえに

<sup>1</sup> 東京大学

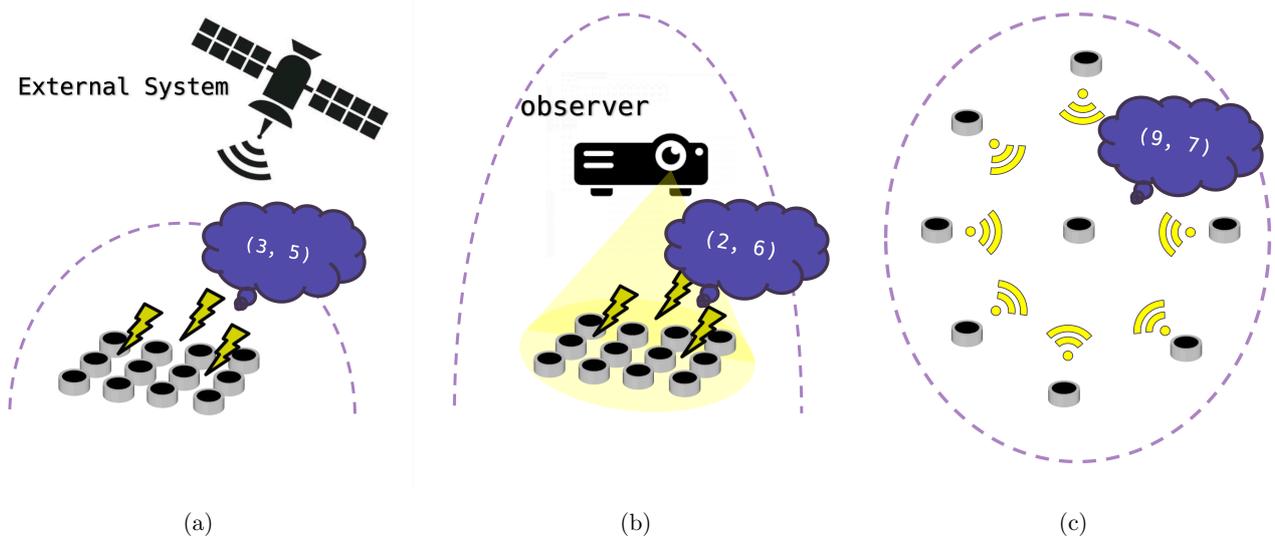


図 1 コレクティブユーザインタフェースにおける形状推定。(a) は GPS などの外部システムを利用して座標推定を行うモデル、(b) はシステム内部にカメラなどの監視者を取り入れ座標推定を行うモデル、(c) は素子間通信のみで座標推定を行うモデルを表す

ネットワークグラフから正方格子点上の物理位置を求める単位長配置問題は NP 困難のクラスに属しており、非常に困難であるに加え、ネットワークグラフから得られる物理配置が一意であるとも限らない [9]。また推定のためにネットワークグラフを全体で共有する必要があり、個々のノードが独立して計算を行うことができない。

これに対応するため、小型計算機に対して小さな制約とその制約情報を持たせる。これにより形状推定問題を、初期化の最悪計算量  $O(n)$ 、最善計算量  $O(\log n)$  で、また、計算ノードの追加や切断を  $O(1)$  で解くことができるようなアルゴリズムを紹介する。このアルゴリズムでは各計算ノードが独立に並列計算可能であり、かつ全体構成を制限することなくノイズの影響を受けない形で元の物理配置を完全に復元可能である。このアルゴリズムに対して、有線接続された形状自在ディスプレイのプロトタイプを作成し、実演することによって正当性を検証した。

## 2. 関連研究

### 2.1 形状変化が可能なインタフェースにおける形状推定

形状推定を行うモデルとして図 1 のような、システム外部の装置を用いるモデル (図 1a)、システム内部に小型デバイスの集合体の他に監視者を用いるモデル (図 1b)、小型デバイスの集合体のみで構成されるモデル (図 1c の 3 通り) が考えられる。

#### 2.1.1 外部システムを用いた位置推定

まず図 1a のようなシステム外部の装置を用いるモデルを考える。位置推定に用いることができる外部装置として、まず GPS (Global Positioning System) といったシステムが挙げられる。このような外部デバイスを用いる利点は

特別なアルゴリズムを必要とせず非常に簡単に推定することができる。しかし GPS では最高でも数センチオーダの誤差が生まれる [5] のに加えて、通信間の障害物や、太陽フレアといった気象条件に対して脆弱であり、かつ受信用回路のコストや回路面積を取るなど、小型計算機の集合体に対する位置推定として用いるのに相応しくない。

#### 2.1.2 監視者を用いた位置推定

そこで環境の影響を防ぐため、かつ、より精度よく推定するために図 1b のようなシステム内部に監視者のようなデバイスを用いる手法が提案されている。このようなシステムでは、小型計算機の中での機能は最小限にし、集合体の外側にあるカメラやプロジェクタなどの機器を用いて位置推定を行う [6]。しかしこの手法では集合体の外側に機器を設置するコストや、監視者の観測できる範囲に限定されるという制約が存在する。このため、例えば手や足で物体を操作した際、それらが監視者の障害物となり、位置推定できず、結果としてインタラクションが阻害されること等も考えられる。

#### 2.1.3 素子間通信のみでの位置推定

最後に考えられるモデルは図 1c 小型デバイスのみで構成されるモデルである。このモデルでは小型デバイス同士が協調し合うことでシステムのトポロジーを推定する。このような推定方法では大きく指向性のある通信とない通信で分けることができる。

まず指向性のある通信や他のノードの方角の取得が可能な小型デバイスを考える。このようなデバイスでは情報が来た方角を得られるため向きがわかる。加えて通信強度から距離の絶対値を計算できる。このため情報元のデバイスからのベクトルを求めることが可能になり、これを様々な

素子同士で行うことで全体のトポロジを求めることができる。このような通信を用いている例として [12,13] がある。しかし、指向性のある通信を実現するには高価なセンサが必要になり、かつそれを配置する面積も必要になる。

次に指向性のない通信を用いて位置推定を行う例を考える。指向性のない通信ではすでに座標が確定している複数点からの距離を強度情報から計算することで推定することができる。例えば2次元平面上の物体であれば3点からの距離をそれぞれ求めることで計算可能である。これを用いた例として [17] がある。しかしこの推定では3点からの情報を精度よく取得する必要がある、距離情報の誤差が問題になる。これを解決するための取り組みとして [3,16] などが挙げられる。しかしこれらは全体形状に対する制約など直感的なインタラクションを阻害する制約を入れることによって解決しようとしており望ましくない。また、これらは3点の確定したノードとの通信をする制約上、通信距離を最短でも3つの機器分取る必要がある。しかしコレクティブなインタフェースを構築するには数多くの計算素子が必要であるため3つ分の長さであっても大勢のノードと通信することになる。このため、さまざまなネットワーク情報を受け取ることになり衝突が起こることや、ノイズが乗ることが予想される。このことから、より近接した少数のノード間だけを対象とするような無線方式が望まれる。

## 2.2 グラフ理論における物理配置問題

グラフから物理配置を求める配置問題は古くから VLSI の自動レイアウト問題として関心が高かった。まず配置問題で本論文にかかわる定義、定理を紹介する。

**定義 2.1** (単位距離グラフ).  $Z^n$  上の部分集合に対応するグラフで、距離が1の点同士が辺としてつながっているグラフのことを**単位距離グラフ**という。

**定義 2.2** (単位距離配置). あるグラフがあったとき、つながっている辺のみが  $Z^n$  上の部分集合として距離1で配置されるような配置を**単位距離配置**という。特に断りがない場合は平行移動、回転、反転を同一視する。

**定義 2.3** (連続グラフ). 任意の点同士が辺をたどって到達可能なグラフのことを**連続グラフ**という。

これらの定義を用いて今回対象とするグラフを定義する。

**定義 2.4** (形状自在配置、形状自在ネットワーク).  $Z^n$  ( $n = 2, 3$ ) の部分集合  $V$  に対して、連続な単位距離グラフ  $G$  が構成できるとき、 $V$  を  **$n$ 次元形状自在配置**と呼び、 $G$  を  **$n$ 次元形状自在ネットワーク**と呼ぶ。

例えば  $n = 2$  の時、部分集合  $V$  としては図 2a のようなものがあり、これに対する単位距離グラフ  $G$  は図 2b となる。ここで [9] より以下が成り立つ。

**定理 2.1.** 子ノードの数が多くても4つの木構造グラフに対して二次元上に単位長配置があるかを判定する問題は NP 完全である。

2次元形状自在配置は正方格子にあり、そのため距離1で隣接する点は高々4つであるため、単位距離グラフに変換後、木構造グラフとして解釈すると、子ノードの数も高々4つである。このことから以下が成り立つ。

**定理 2.2.** 2次元形状自在ネットワークは子ノードが多くても4つの木構造グラフに変換できる。このことより2次元形状自在ネットワークの単位長配置問題は NP 困難である。

## 3. 方法

関連研究で見たように、制限のない形状自在ネットワークにおける単位長配置問題は計算量的な観点で非常に困難な問題である。対象となるシステムはコレクティブに多くの素子を持っていることが期待されるため、全探索で対処することは現実的ではないと考えられる。また扱う通信モデルは、通信の効率性の観点より、近接した少数の素子のみで構成できることが望ましい。

そこで今回扱うモデルは、近接素子として最も近い形である隣接素子との通信のみができるモデルとし、このモデルに対して形状推定を行うアルゴリズムを考える。隣接素子の情報に限ったことによって強度の情報を持つ必要がないため、ハードウェアやソフトウェアを単純化できる。半導体チップの小型高集積化による制限上、通信は指向性を持たないものとする。

以下必要な用語の定義を行う。まずはある単位長配置を求める問題を定義する。

**定義 3.1** (弱推定).  $n$ 次元形状自在配置  $V$  から生成された  $n$ 次元形状自在ネットワーク  $G$  から生成される単位長配置が一意に定まる時、 $V$  を**弱推定可能な配置**と呼び、その単位長配置を**弱推定された配置**と呼ぶ

**定義 3.2** (強推定). 平行移動、回転、反転を含めて単位長配列が一意に定まる時、それぞれ**強推定可能**、**強推定された配置**と呼ぶ。明らかに強推定可能ならば弱推定可能である。

**定義 3.3** (グローバルな座標). 元の配置に対応する座標を**グローバルな座標**と呼ぶ

### 3.1 単位素子の変更

制約のない形状自在ネットワークの素子は単純に1つの計算機からなる。これを複数素子まとめて一つの複合素子とするため、以下で単位素子という言葉を決める。単位素子は  $x, y$  軸方向に規則的に並べて形状自在配置を作ることができる複数素子のことである。

**定義 3.4** (単位素子). ある特定の形を成す  $Z^n$  ( $n = 2, 3$ ) の部分集合  $U$  を考える。 $n$ 次元形状自在配置  $V$  が有限個の  $U$  の平行移動の重なりのない組み合わせによって構成できるとする。この時、 $U$  を**単位素子**と呼び、この平行移動した  $U$  の集合を  $U$  と  $V$  から生成された**単位素子集合  $V'$**

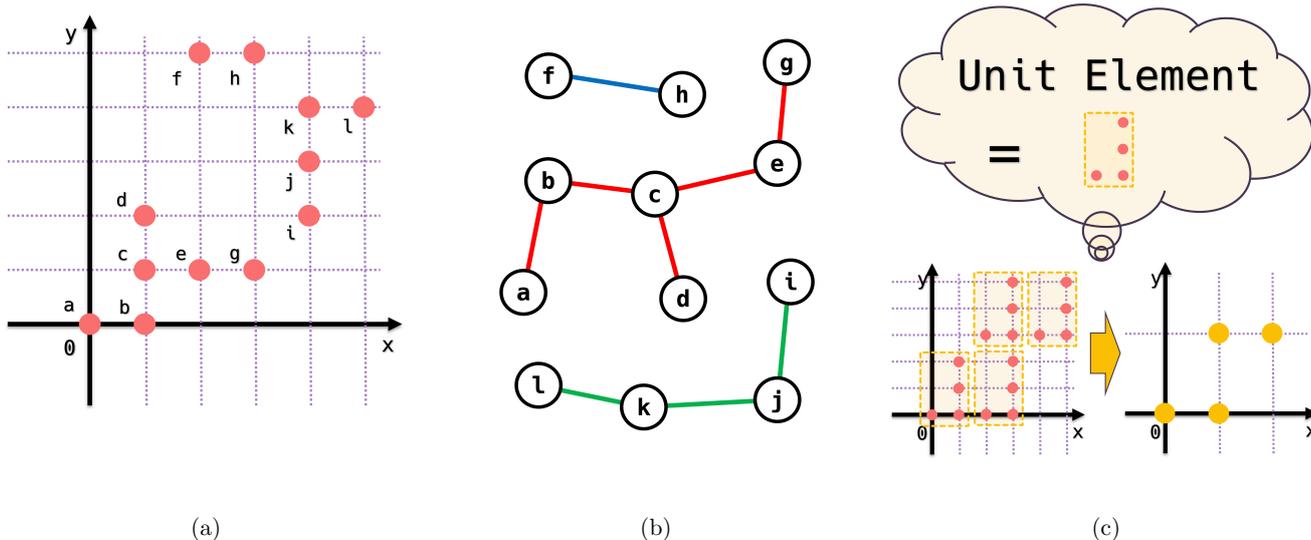


図 2 グラフ理論における用語の図的な説明。(b) は (a) から生成された単位距離グラフを表しており、(c) は 3 章で定義される単位素子の配置制約を表している。

と呼ぶ。つまり  $V = \sum_{u \in V'} u$  であり、 $u, u' \in V'$  ならば  $u \cap u' = \emptyset$  で、 $u$  は  $U$  の並行移動で得ることができる。

単位素子を用いて今回対象とする形状自在配置を定義する。これは図 2c に示すような単位素子の配置で生成された配置のことである。

**定義 3.5** (単位素子で構成される規則的な形状自在配置). ある形状自在配置  $V$  の単位素子  $U$  と、 $U$  と  $V$  から生成される単位素子集合  $V'$  を考える。この時、任意の  $u, u' \in V'$  で、 $u, u'$  の内部の点の中で距離が 1 となる座標がある時、 $n$  次元のある  $n-1$  個の座標軸に関して  $u, u'$  の要素の最小値がそれぞれ等しいとする。この時、形状自在配置  $V$  は単位素子  $U$  に対して規則的な配置であると呼ぶ。

今回用いる単位素子を定義する。これは 2 次元では正方形、3 次元では立方体のことである。

**定義 3.6** (変更された単位素子). 変更された単位素子を  $Z^n$  ( $n = 2, 3$ ) 上の格子点で長さ 1 の超立方体となるような  $2^n$  個の座標から構成された部分集合  $U$  とする。

以降の議論では、形状自在配置が変更された単位素子で規則的に構成されているとする。単位素子内部の計算機は単位素子内部のトポロジー、および計算機全体が変更された単位素子で構成されているということを情報として所持しているものとする。

### 3.2 Root 条件

root 条件、root、root 単位素子を以下のように定める。root は集合体の中で基準となる座標系を定めるものである。

**定義 3.7** (root 条件). ある計算機が root 条件を満たすとは、その計算機が先天的にグローバルな座標を知っていることとする

**定義 3.8** (root, root 単位素子). root は root 条件を満た

す計算機、root 単位素子は単位素子内の計算機すべてが root であるような単位素子とする

### 3.3 弱推定可能な十分条件

変更された単位素子に対して形状自在ネットワークを作ると、対応する形状自在配置は回転、並行移動、反転を除いて一意に定まる。

**定理 3.1.** 変更された単位素子で構成される形状自在配置から生成された形状自在ネットワークは弱推定可能であり、弱推定された配置はノード数を  $n$  とすると最悪計算量  $O(n)$  で求めることができる。

証明は付録 A.1。

### 3.4 強推定可能な十分条件

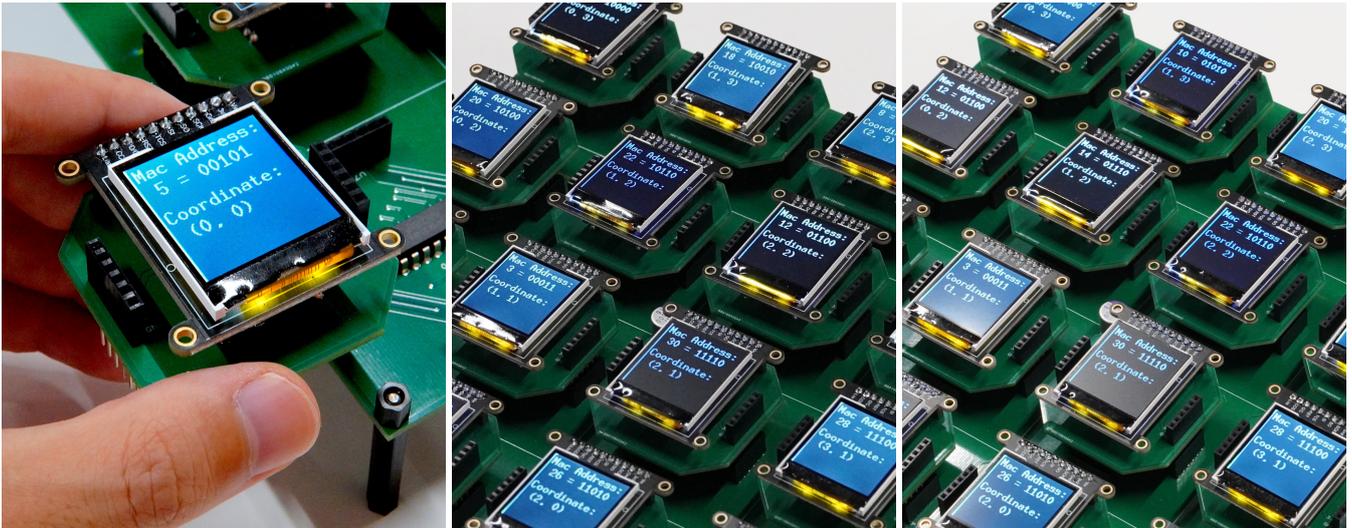
弱推定で定まった形状自在配置を固定したい。しかし固定を行うには素子間の通信だけではできない。そのため外部の座標系を導入する必要があるが、root 単位素子を定めることで解決できる。

**定理 3.2.** 変更された単位素子と少なくとも 1 つの root 単位素子で構成される形状自在ネットワークは強推定可能である。

*Proof.* root 単位素子を定めたことにより、回転、反転、平行移動を行うことができない。定理 3.1 より、変更された単位素子で構成される形状自在ネットワークからは平行移動、回転、反転を除いて一意であるため示せた。 □

## 4. 実演

応用先として挙げた形状自在ディスプレイをマイコン esp32c3 上で物理配線による検証した。単位素子内部の情



(a) (b) (c)

図 3 デモンストレーションを行った図。(a) は個々の素子を表しており、正方形の形で 4 つ繋がった形が変更された単位素子である。(b)、(c) はそれぞれ動的に入れ替える前と後の同一の場所を表している。入れ替える前と後では座標は同一であるが、MAC アドレスは異なっていることが確認できる

報は MAC アドレスを用いて認識する。物理層では、無指向性の通信を模倣するため、4 方向から来るすべての uart 通信を一つの rx,tx ポートにまとめて通信した。変化に対して座標推定ができていることを確認するため、動的に単位素子を入れ替えて検証した。図のように MAC アドレスが変わっているが、問題なく座標推定が行うことができている様子がわかる。具体的な座標構築アルゴリズムは [8] にある。

## 5. 今後の展望

インタラクションに関する制約、単位素子に関する制約、構造に関する制約に関する改善点の 3 点紹介する。

今回はアルゴリズムの検証のため物理配線でのプロトタイプ作成を行った。しかし、物理的に抜き差しする手間等がインタラクションを大きく阻害している。このため無線通信を用いたプロトタイプでのインタラクションを検証する必要がある。また目標とするビジョン [11] においては超小型計算機が想定されている。しかし今回のディスプレイは一辺約 3cm からなる巨大素子であり、小型化した際のインタラクションについてより詳細に検証する必要がある。

単位素子の配置にずれが生じない仮定を用いていた。実際のプロダクトにおいてはずれが生じるが、帰納法の証明の中でみたように、隣接する素子がそれぞれ確定していることが座標推定できる条件であるため、この条件を緩めることが可能である。また単位素子を超立方体としたが、三角形など様々な形に変換することも可能である。二次元の場合だと今回は隣り合う素子が 4 つの規則的な配列を対象にして行ったが、この条件を緩めても座標推定は可能で

ある。どのような条件が可能かさらに詳細に調べる必要がある。

形状自在配置として考えられるものは、 $Z^n$  の部分集合のみではない。わかのように循環した座標系でも配置可能である。このようなより一般的な場合での座標推定アルゴリズムなども必要になる場面も容易に考えられ、調査する必要がある。加えて、耐障害性やより高速な座標推定の要望から複数の root が存在する場合が考えられる。このような場合に対応できかつ、複数の root 間で多少の誤差があった場合、その際を吸収できるようなアルゴリズムを考える必要がある。

## 6. まとめ

この論文ではネットワークグラフから座標を効率的に生成するアルゴリズムの提案を行った。2次元、3次元問わず、緩い制約のもと高精度にかつ隣接した素子の情報のみで座標を完全に推定可能である。座標推定の正しさを検証するためにプロトタイプを作成し、任意の場所で正しく画像が生成できることを実証した。この論文により形状推定が必要な形状変化可能なインタフェースに関してより幅広い可能性が生まれた。

## 参考文献

- [1] Alexander, J., Lucero, A. and Subramanian, S.: Tilt displays: Designing display surfaces with multi-axis tilting and actuation, *MobileHCI'12 - Proceedings of the 14th International Conference on Human Computer Interaction with Mobile Devices and Services*, pp. 161-170 (online), DOI: 10.1145/2371574.2371600 (2012).

- [2] Alexander, J., Roudaut, A., Steimle, J., Hornbæk, K., Alonso, M. B., Follmer, S. and Merritt, T.: Grand challenges in Shape-changing interface research, *Conference on Human Factors in Computing Systems - Proceedings*, Vol. 2018-April (online), DOI: 10.1145/3173574.3173873 (2018).
- [3] Coppola, M., Guo, J., Gill, E. and de Croon, G. C.: Provable self-organizing pattern formation by a swarm of robots with limited knowledge, *Swarm Intelligence*, Vol. 13, pp. 59–94 (online), DOI: 10.1007/S11721-019-00163-0/FIGURES/16 (2019).
- [4] Dimitriadis, P. and Alexander, J.: Evaluating the Effectiveness of Physical Shape-Change for in-Pocket Mobile Device Notifications, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, New York, NY, USA, Association for Computing Machinery, p. 2589–2592 (online), DOI: 10.1145/2556288.2557164 (2014).
- [5] Enge, P. K.: The Global Positioning System: Signals, measurements, and performance, *International Journal of Wireless Information Networks*, Vol. 1, pp. 83–105 (online), DOI: 10.1007/BF02106512/METRICS (1994).
- [6] Goc, M. L., Kim, L. H., Parsaei, A., Fekete, J. D., Dragicevic, P. and Follmer, S.: Zoids: Building blocks for swarm user interfaces, *UIST 2016 - Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pp. 97–109 (online), DOI: 10.1145/2984511.2984547 (2016).
- [7] Goguey, A., Steer, C., Lucero, A., Nigay, L., Sahoo, D. R., Coutrix, C., Roudaut, A., Subramanian, S., Tokuda, Y., Neate, T., Pearson, J., Robinson, S. and Jones, M.: Pickcells: A Physically Reconfigurable Cell-composed Touchscreen, *Conference on Human Factors in Computing Systems - Proceedings*, (online), DOI: 10.1145/3290605.3300503 (2019).
- [8] Goto, T.: Algorithm implementation on github (2023).
- [9] Gregori, A.: Unit-length embedding of binary trees on a square grid, *Information Processing Letters*, Vol. 31, pp. 167–173 (online), DOI: 10.1016/0020-0190(89)90118-X (1989).
- [10] Grönvall, E., Kinch, S., Petersen, M. G. and Rasmussen, M. K.: Causing Commotion with a Shape-Changing Bench: Experiencing Shape-Changing Interfaces in Use, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, New York, NY, USA, Association for Computing Machinery, p. 2559–2568 (online), DOI: 10.1145/2556288.2557360 (2014).
- [11] Kadomoto, J., Sasatani, T., Narumi, K., Usami, N., Irie, H., Sakai, S. and Kawahara, Y.: Toward Wirelessly Co-operated Shape-Changing Computing Particles, *IEEE Pervasive Computing*, Vol. 20, pp. 9–17 (online), DOI: 10.1109/MPRV.2021.3086035 (2021).
- [12] Li, Y., Klingner, J. and Correll, N.: Distributed camouflage for swarm robotics and smart materials, *Autonomous Robots*, Vol. 42, pp. 1635–1650 (online), DOI: 10.1007/S10514-018-9717-6/FIGURES/12 (2018).
- [13] Mathews, N., Christensen, A. L., O'Grady, R., Mondada, F. and Dorigo, M.: Mergeable nervous systems for robots, *Nature Communications 2017 8:1*, Vol. 8, pp. 1–7 (online), DOI: 10.1038/s41467-017-00109-2 (2017).
- [14] Merrill, D., Sun, E. and Kalanithi, J.: Sifteo cubes, *Conference on Human Factors in Computing Systems - Proceedings*, pp. 1015–1018 (online), DOI: 10.1145/2212776.2212374 (2012).
- [15] Oosterhuis, K. and Bioria, N.: Interactions with Proactive Architectural Spaces: The Muscle Projects, *Commun. ACM*, Vol. 51, No. 6, p. 70–78 (online), DOI: 10.1145/1349026.1349041 (2008).
- [16] Pluhacek, M., Garnier, S., Reina, A., Pluhacek, M., Garnier, S. and Reina, A.: Decentralised construction of a global coordinate system in a large swarm of minimalistic robots, *arXiv*, (online), available from <https://arxiv.org/abs/2302.14587v1> (2023).
- [17] Rubenstein, M., Ahler, C. and Nagpal, R.: Kilobot: A low cost scalable robot system for collective behaviors, *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3293–3298 (online), DOI: 10.1109/ICRA.2012.6224638 (2012).
- [18] Suzuki, R.: Collective Shape-Changing Interfaces, *Adjunct Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, UIST '19 Adjunct, New York, NY, USA, Association for Computing Machinery, p. 154–157 (online), DOI: 10.1145/3332167.3356877 (2019).

## 付 録

### A.1 定理 3.1 の証明

*Proof.* 2次元の場合を証明する。形状自在ネットワークの仮定により、元の形状自在配置は解の一つであるため、任意の形状自在ネットワークには単位長配置が存在する。したがって計算量および一意性が問題である。まず一意性の証明を行う。変更された単位素子の個数  $n$  に関する帰納法で証明する。 $n = 0, 1$  の時、明らかに成り立つ。 $n = k$  ( $k \geq 1$ ) で成り立つと仮定し、 $n = k + 1$  を考える。形状自在配置から、取り除いた際に、取り除いた後の  $Z^2$  の部分集合が再び形状自在配置となるようなある一つの単位素子  $U$  を選ぶ。このような  $U$  は存在する。帰納法の過程により、 $U$  を除いたネットワークは一意に決まる。単位素子  $U$  に隣接するノードを単位素子  $G$  とする。回転における対称性より、 $G$  の右側のノード 2 つがネットワークグラフ上で単位素子  $U$  と接続しているものとする。この時、 $U$  を除いたネットワークから生成された唯一の配置上で単位素子  $U$  は  $G$  の右側のみに配置できる。なぜなら、接続する単位素子内部の素子は 2 つであり、どちらもユークリッド距離が 1 となる必要があることおよび、単位素子  $U$  と  $G$  が正方形であることからである。このことにより、単位素子  $U$  の追加が一意であり、 $n = k + 1$  でも成り立つ。よって数学的帰納法により示せた。ここで帰納法の議論の中で、単位素子をつ一つ増やしたときに推定に必要な計算量の増加は高々  $\mathcal{O}(1)$  であり、単位素子は 4 つで構成されるため、計算量は  $\mathcal{O}(n)$  である。3次元も同様に示すことができる。□