

近隣ノードの移動履歴情報を用いた位置推定手法のシミュレーションによる性能解析

藤井 彩恵^{†1} 野村 崇志^{†1} 梅津 高朗^{†1,†2}
山口 弘純^{†1,†2} 東野 輝夫^{†1,†2}

我々はこれまでに、無線端末のアドホック通信を利用して近隣端末との移動軌跡情報を交換することにより、各端末が自身の位置を推定する手法について検討している。本稿では、これまでに提案した手法のアルゴリズム特性を明らかにし、シミュレーションにより解析する。また、交換する軌跡情報の量や頻度などが精度や通信量に与える影響を評価し、合理的な通信量で一定の精度の位置を得るためのアルゴリズムの最適なプロトコル設計について述べる。

Performance Analysis of Localization using Trajectory Information from Neighbor Nodes

SAE FUJII,^{†1} TAKASHI NOMURA,^{†1}
TAKAAKI UMEDU,^{†1,†2} HIROZUMI YAMAGUCHI^{†1,†2}
and TERUO HIGASHINO^{†1,†2}

We have designed a range-free position estimation algorithm for mobile nodes using trajectory information from neighbor nodes. In this paper, we analyze the property of the proposed estimation algorithm, and validate the analysis by simulation. Also, we evaluate the impact of several factors such as the amount of trajectory for exchange and sending interval on position error and communication traffic, and give the optimal design of the protocol to estimate the position with a certain error range and reasonable communication traffic.

^{†1} 大阪大学大学院情報科学研究科

Graduate School of Information Science and Technology, Osaka University, Japan

^{†2} 独立行政法人 科学技術振興機構, CREST

Japan Science and Technology Agency, CREST

1. はじめに

ユビキタスコンピューティングを実現する機能を持ったデバイスの普及にともない、それらのデバイスの位置情報をリアルタイムに測定し、ナビゲーションやロケーションに依存した情報提供などのサービスに利用したり、それらの位置情報の履歴から人間の行動解析を実現することが期待されている。近年、多くの携帯電話やカーナビゲーションシステムにおいてGPS受信機が装備されているが、屋内や地下街、多くの建造物が複雑に入り組んだ都市環境などでは、GPS衛星からの電波が受信できないため、測位できない可能性がある。このため、GPSで測位できない場所でも上記のようなサービスを実現するために、端末の位置をリアルタイムに測位できる技術が必要とされている。

端末の位置を推定する手法は様々な分野において提案されている。例えば文献1), 2)では、都市部における歩行者の行動解析やスーパーマーケットにおけるマーケティングのために、カメラを用いて移動体を追跡する手法が提案されている。これらの手法では主に解析により対象を認識、識別することを目的としているため、リアルタイムに端末の位置を必要とするサービスには適さない。また、文献3)のように、RFIDタグを用いた移動体の追跡手法も提案されているが、RFIDタグを配置、管理するのに多くのコストを要する。さらに、無線アドホック通信機能を用いた位置推定手法も数多く提案されている。アドホック通信機能を用いた位置推定手法は、ランドマークとの距離や角度などを専用ハードウェアを用いて測定し、位置を推定するレンジベース手法^{4),5)}と、そのようなハードウェアは用いず、端末間の接続情報のみを利用して推定を行うレンジフリー手法に分類される。レンジベースは専用ハードウェアに別途コストが必要となるため、低コストで実現するためにはレンジフリーがより適切である。また、位置推定手法は集中型アルゴリズム⁶⁾⁻⁸⁾と分散型アルゴリズム⁹⁾⁻¹³⁾に分類することができる。集中型アルゴリズムでは、端末間の通信に関する情報をすべて用いるため高精度な推定が可能であるが、各端末が自身の位置をリアルタイムに推定することができない。一方、分散型アルゴリズムでは端末自身もしくは他端末と協調することにより位置を推定するが、それらの手法の多くはセンサネットワークにおけるセンサ端末の推定を対象としており、端末が常に移動するモバイルネットワークに単純には適用できない。

そこで、我々は文献14)において、無線端末のアドホック通信のみを利用して近隣端末との移動軌跡情報を交換することにより、各端末が位置を推定する手法を提案している。同手法では近接端末間で移動軌跡情報を交換することにより、低コストで実現可能で、高精度に

位置を推定することができる。また、文献 15) において、実際に使用される環境や利用例について検討するとともに、それらの使用環境を想定したシミュレーションにおいて、提案手法の性能ならびに実用性について評価を行っている。

本稿では、我々が文献 14), 15) で検討してきた位置推定手法について、そのアルゴリズム特性を明らかにし、シミュレーションにより解析する。シミュレーション結果より、得られる移動軌跡情報が多いほど、推定に利用できる制約が増え、精度が良いことが分かった。各端末が対象領域内のすべての端末の移動軌跡情報を保持した場合、提案手法において最も高精度に位置を推定できると考えられるが、その一方で莫大な通信量が必要となる。そこで、交換する軌跡情報の量や頻度などが精度や通信量に与える影響を評価し、合理的な通信量で一定の精度の位置を得るためのアルゴリズムの最適なプロトコル設計を考える。さらにシミュレーションによる性能評価により既存の分散型リアルタイム位置推定手法よりも精度が良く、パケットロスや無線の揺らぎによる影響に対してもロバストであることを示す。

2. 提案手法の概要

2.1 想定環境

提案アルゴリズムは、IEEE802.15.4 のような PAN 通信デバイスを装備した移動ノード (ノード) と呼ばれる移動端末上で実行されるものとし、ノードは正確な座標が既知であるランドマークが配置されている環境を移動するものとする。移動ノードおよびランドマークは、MAC アドレスまたはランダムな値で生成される固有の ID を持つものとする。提案アルゴリズムはレンジフリー手法であるため、ノード間の距離を測定する必要がない代わりに、ノードの最大無線到達距離と最大移動速度を仮定し、それぞれ R と V で表す。各ノードは整数値で表されるタイマを保持しているものし、タイマは周期 T ごとに増加し、 T の値はすべてのノードで共通であるものとする。ここでノード u の k 番目のタイムスロットを $\phi_u^{[k]}$ で表す。タイマは図 1 のようにノード間で同期する必要はない。以下、 $\phi_u^{[k]} - \phi_v^{[h]}$ を $\phi_u^{[k]}$ と $\phi_v^{[h]}$ の間のタイムスロットの数 (ただし、 $\phi_u^{[k]}$ が $\phi_v^{[h]}$ よりも後の時刻である) と定義する。図 1 の例の場合、 $\phi_u^{[k]} - \phi_v^{[h]} = 1$, $\phi_u^{[k]} - \phi_v^{[h+1]} = -1$, $\phi_u^{[k+1]} - \phi_u^{[k-1]} = 2$ となる。

2.2 各ノードの基本動作

ここで、 $p_u^{[k]}$ をノード u の $\phi_u^{[k]}$ における推定位置とする。また、 $P_u^{[k,k']}$ をノード u の $\phi_u^{[k]}$ から $\phi_u^{[k']}$ までの推定移動軌跡とする ($k \leq k'$)。よって、 $P_u^{[k,k']}$ は推定位置 $p_u^{[k]}, p_u^{[k+1]}, \dots, p_u^{[k']}$ の並びで表される。本稿における表記の概要は表 1 に示す。

メッセージ交換 各ノード u は自身の推定軌跡を Hello メッセージとして F タイムスロット

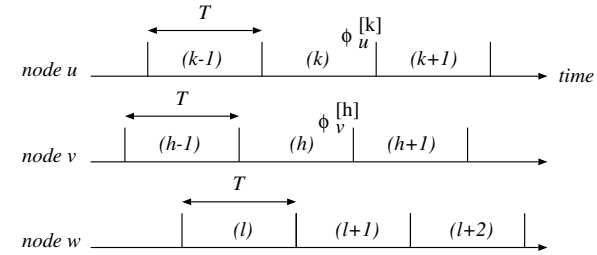


図 1 ノードの周期
Fig. 1 Time intervals of nodes.

表 1 表記
Table 1 Notations.

記号	定義
T	タイムスロットの間隔
V	ノードの最大速度
$\phi_u^{[k]}$	ノード u の k 番目のタイムスロット
$p_u^{[k]}$	$\phi_u^{[k]}$ におけるノード u の推定位置
$P_u^{[k,k']}$	$\phi_u^{[k]}$ から $\phi_u^{[k']}$ までのノード u の推定軌跡 ($p_u^{[k]}, p_u^{[k+1]}, \dots, p_u^{[k']}$)
F	1 ホップ送信間隔 (Hello メッセージとして自身の情報を送信する間隔)
F^+	2 ホップ送信間隔 (Hello メッセージに 1 ホップ隣接ノードの推定軌跡を含める間隔)
W	現在位置を推定するのに利用する軌跡の長さ
L	更新の対象となる軌跡の長さ

トに 1 回、定期的に周辺ノードに対しブロードキャストを行う。 F を 1 ホップ送信間隔と呼ぶ。さらに F^+ ごとに送信する Hello メッセージには、自身の推定軌跡に加え、1 ホップ以内に存在する隣接ノードが送信した推定軌跡も含める。 F^+ は 2 ホップ送信間隔と呼ぶ。通信量を減少させるためには F^+ はある程度大きい値が望ましいと考えられる。ノード u が $\phi_u^{[h]}$ に送信した Hello メッセージにはノード u の推定軌跡 $P_u^{[h',h]}$ ($h' = h - L + 1$) が含まれる。ここで L は軌跡長と呼ぶ。また F^+ に 1 回ノード u が送信する Hello メッセージには、ノード v の推定軌跡 $P_v^{[l',l]}$ ($\phi_u^{[h']} - \phi_v^{[l]} \leq 1$, $h' \leq h'' < h$ かつ $l' = l - h'' + h'$) が含まれる。ただし、ノード v は $\phi_u^{[h']}$ においてノード u の 1 ホップ隣接ノードであるとする。もし、ノード u が $\phi_u^{[k]}$ にノード v の Hello メッセージを受信した場合、ノード u はノード v の推定軌跡 $P_v^{[h',h]}$ と、もしあれば、2 ホップ隣接ノードであるノード w の推定軌跡 $P_w^{[l',l]}$ を得る。各ランドマークが送信する Hello メッセージには、正確な位置情報が含まれるものとする。

現在位置の推定 ノード u の現在のタイムスロットを $\phi_u^{[k]}$ で表す。 $p_u^{[k]}$ の推定は式 (1)

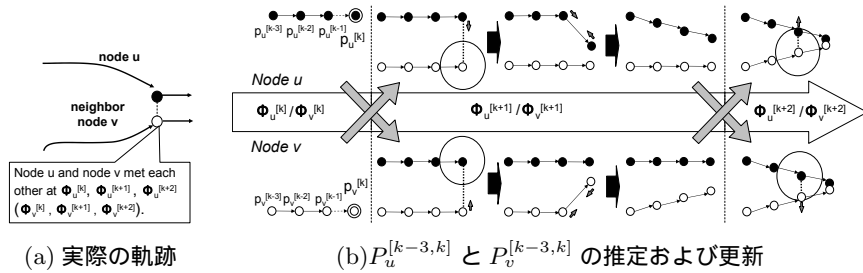


図 2 アルゴリズムの動作例
Fig.2 Snapshot from algorithm.

で定義されるように、直近の W 個 (W は定数とする) のタイムスロット $\phi_u^{[\kappa-k]}$ ごとに $p_u^{[\kappa-k]} - p_u^{[\kappa-(k+1)]}$ で定義されるベクトルを求め、 $p_u^{[\kappa-W]}$ にそれらの和の平均を加えることによって求める。

$$\begin{aligned}
 p_u^{[\kappa]} &= p_u^{[\kappa-W]} + \frac{1}{W} \sum_{k=1}^W (p_u^{[\kappa-k]} - p_u^{[\kappa-(k+1)]}) \\
 &= p_u^{[\kappa-W]} + \frac{1}{W} (p_u^{[\kappa-1]} - p_u^{[\kappa-(W-1)]}) \quad (1)
 \end{aligned}$$

軌跡の更新 次に、Hello メッセージに含まれる 1 ホップまたは 2 ホップ隣接ノードの推定軌跡と、各タイムスロットにおける 1 ホップおよび 2 ホップ隣接ノードを表す通信履歴情報を用いて、直近の L 個のタイムスロットにおける推定軌跡 $P_u^{[\kappa-(L-1),\kappa]}$ を更新する。推定軌跡の更新には、同時刻における 2 ノード間の位置関係または同じノードにおける前後の時刻間の位置関係から得られる下記の 3 種類の制約を考える。

- 通信制約 (positive radio-range constraint): 任意のタイムスロット $\phi_u^{[k]}$ とノード v に対し、 $\phi_u^{[k]}$ においてノード v がノード u の隣接ノードであった場合、 $p_u^{[k]}$ と $p_v^{[k]}$ 間の距離は無線到達距離 R 以内でなければならない (ただし、 $\phi_u^{[k]} - \phi_v^{[k]} = 1$) 。
- 非通信制約 (negative radio-range constraint): 任意のタイムスロット $\phi_u^{[k]}$ とノード v に対し、 $\phi_u^{[k]}$ においてノード v がノード u の隣接ノードではない場合、 $p_u^{[k]}$ と $p_v^{[h]}$ 間の距離は無線到達距離 R よりも大きくななければならない (ただし、 $\phi_u^{[k]} - \phi_v^{[h]} = 1$) 。
- 移動制約 (movable-range constraint): 任意のタイムスロット $\phi_u^{[k]}$ において、 $p_u^{[k]}$ と $p_u^{[k-1]}$ 間の距離は時間 T 内にノードが移動できる距離 $V \cdot T$ 以内でなければならない。

ノードの位置は時間的、空間的に相互に関係するため、ある位置の更新が同じノードの異なるタイムスロットの位置における制約や同じ時刻の異なるノードの位置における制約を満たさなくなる可能性がある。例えば、移動制約を満たす位置 $p_u^{[k]}$ を通信制約により更新した場合、 $p_u^{[k-1]}$ や $p_u^{[k+1]}$ との間における移動制約が満たさなくなる可能性がある。そこで、可能な限り多くの制約を満たすために $P_u^{[\kappa-(L-1),\kappa]}$ に含まれるそれぞれの位置について $p_u^{[\kappa-(L-1)]}$ から順に複数回更新を行う。例えば図 2(a) に示すようにノード u とノード v が移動した場合を考える。はじめに、ノード u は図 2(b) のように、現在位置 $p_u^{[k]}$ を推定する。ノード u は $\phi_u^{[k]}$ にノード v から受信した Hello メッセージによって得られる通信制約を満たすように $p_u^{[k]}$ を $\phi_u^{[k+1]}$ で修正する。この修正により、移動制約が破られるため、 $P_u^{[k-2,k]}$ が移動制約を満たすように再び修正される。さらに、 $\phi_u^{[k+1]}$ にノード v から受信した Hello メッセージに基づき、非通信制約が満たされるよう $p_u^{[k-1]}$ を $\phi_u^{[k+2]}$ で修正する。このように、ノード u の更新された移動軌跡は、通信制約や非通信制約を介して隣接ノードの移動軌跡に影響を与える。そして、各ノードが局所的に移動軌跡を更新し、更新された移動軌跡を交換し続けることにより、すべてのノードの移動軌跡が徐々に改善される。

なお、アルゴリズムの詳細は文献 [14], [15] で述べてあるため、本稿では割愛する。

3. アルゴリズムの特性解析

3.1 制約が精度に及ぼす影響

本節では、提案手法において各制約がどのように軌跡の改善に寄与するかを典型的な例を用いて説明する。ノード a, b, c, d が図 3(a) のような軌跡を辿った場合のノード a における推定軌跡の修正に関して考える。タイムスロット $\phi_a^{[3]}$ において b から a へ、 d から c へ、 c から a へとそれぞれ Hello メッセージが送られる。ここで、各ノード n の $\phi_n^{[t]}$ における正確な位置を $q_n^{[t]}$ で表す。また、説明の簡単のため、周辺のノードは自身の軌跡を十分に正確に推定できており、Hello メッセージにはほぼ正確な位置が含まれているものとする ($n \in \{b, c, d\} : p_n^{[t]} \simeq q_n^{[t]}$)。また、すべてのノードのタイムは同期しているものとする。

提案手法では、各ノードは直前までの軌跡からの線形補完により現在位置を推定し、その後、受信した Hello メッセージによりその位置を更新する。例えば $\phi_a^{[3]}$ において、ノード a は現在位置 $p_a^{[3]}$ を推定し、隣接ノードからのメッセージに含まれる推定軌跡情報に基づき、自身の軌跡 $p_a^{[2]}, p_a^{[3]}$ を修正する。

はじめに、隣接ノードの推定軌跡を考慮せず、最新の推定位置のみを用いて位置を求めた場合の推定例を図 3(b) に示す。この場合、 $\phi_a^{[3]}$ における自身の推定位置 $p_a^{[3]}$ と受信した b

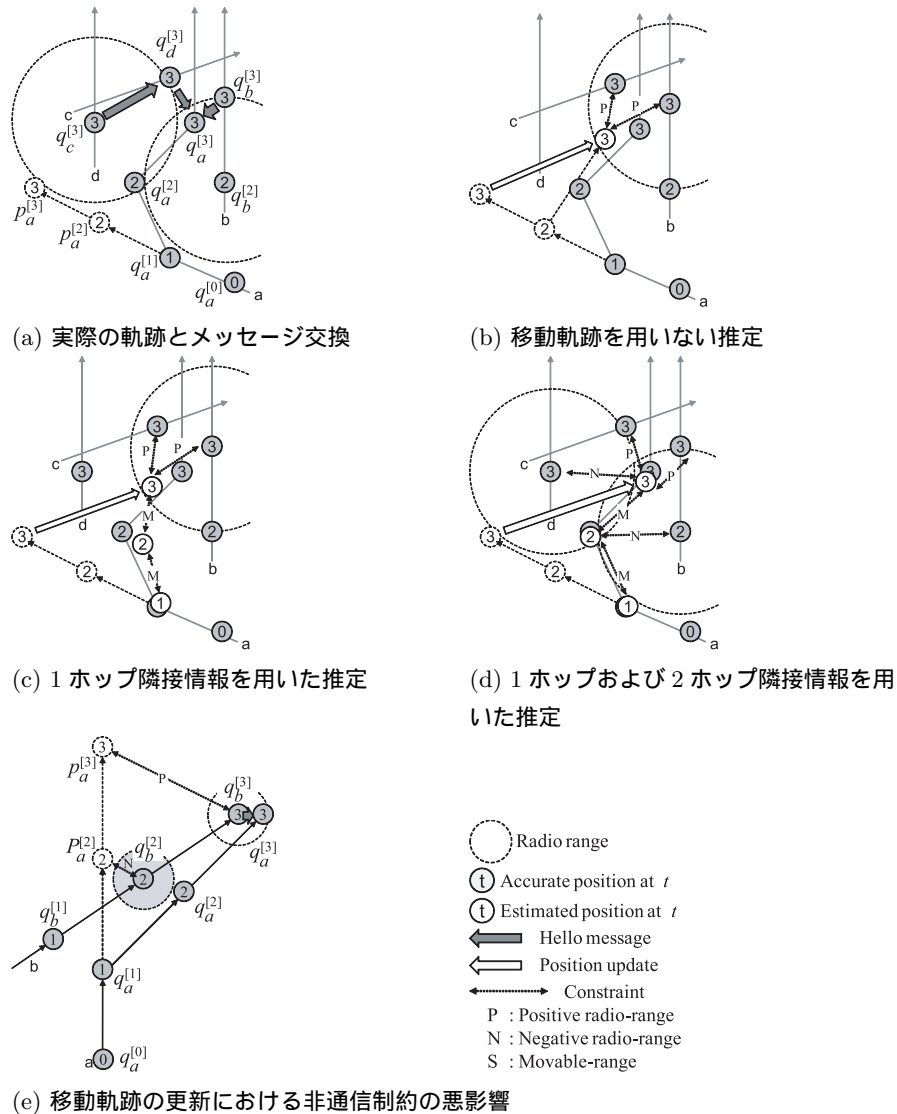


図3 推定過程の概要

Fig. 3 Overview of estimation process.

の推定位置 $p_b^{[3]}$ の距離が、Hello メッセージを受信できたにも関わらず、無線到達距離以上あり、通信制約を満たさない。このため、 $p_a^{[3]}$ には $p_b^{[3]}$ へと近づく方向へ位置が修正される。所定回数の位置の修正の結果、 $p_a^{[3]}$ は制約を満たす位置へと修正される。

次に、1 ホップ隣接ノードの推定軌跡を考慮に入れた場合を図 3(c) に示す。この場合も先ほどと同様に $p_a^{[3]}$ には $p_b^{[3]}$ へと近づく方向へ修正されるが、軌跡を考慮した場合には、現在的位置に加え、それまでにノード a が辿った軌跡も考慮に入れた修正が行なわれる。この場合、 $p_a^{[1]}$ と $p_a^{[2]}$ 間、 $p_a^{[2]}$ と $p_a^{[3]}$ 間の移動制約により、 $p_a^{[1]}$ と $p_a^{[2]}$ も修正される。

さらに、ノードが自身の軌跡情報を送信するのみならず隣接ノードの軌跡情報も転送する場合、ノード a は、ノード c 経由で d の位置 $p_d^{[3]}$ も取得できる。図 3(d) に 1 ホップ隣接ノードと 2 ホップ隣接ノードの推定軌跡から得られる非通信制約も考慮した場合の推定結果を示す。時刻 $\phi_a^{[2]}$ で、 a は b から Hello メッセージを受信していないため、 $p_a^{[2]}$ が $p_b^{[2]}$ から遠ざかる方向へ修正される。同様に $\phi_a^{[3]}$ で a は d からの Hello メッセージを受信していないため、 $p_a^{[3]}$ が $p_d^{[3]}$ から遠ざかる方向へ修正される。結果として、(c) の場合と比較してもより精度の高い軌跡が推定できる。 $\phi_a^{[4]}$ 以降も受信した情報に従って修正が繰り返されることにより、最終的には誤差が大きい位置に関しても徐々に正しい位置へ改善されていく。

ここで、非通信制約は、特定の範囲内に軌跡が修正されることを禁ずる制約であるため、修正前の座標が正しい座標とかけ離れている場合などには、有効には働かない可能性がある。図 3(e) に非通信制約が解の修正を阻害する例を与える。ノード a は、 $\phi_a^{[1]}$ において進路を変え、その後 $\phi_a^{[3]}$ においてノード b から Hello メッセージを受信している。その際、 a は直前までの軌跡からの線形補完により $\phi_a^{[2]}$ および $\phi_a^{[3]}$ における自身の推定を、 $p_a^{[2]}$ 、 $p_a^{[3]}$ と推定し、Hello メッセージを受信した際、通信制約により、 $p_a^{[3]}$ を $p_b^{[3]}$ の方へと修正する。しかし、 $\phi_a^{[2]}$ において、 a は b からの Hello メッセージを受信していないため非通信制約が働く。このため図のように、非通信制約が $p_a^{[2]}$ が正しい座標 $q_a^{[2]}$ への移動を邪魔するように働いてしまう。このような問題は、非通信制約を適用前の位置が正しい場合には発生しにくい。推定誤差が比較的大きいと想定される現在時刻に近い推定位置に関しては非通信制約を適用せず、現在時刻の推定位置がある程度修正されてから適用することで回避できると思われる。4.1 節でその効果を示す。

3.2 アルゴリズムの特性の検証

3.1 節で述べたプロトコルの特性解析を証明するために、提案手法のシミュレーションを行った。100m 四方の自由空間に 9 つのランドマークを配置した領域にノードが Random

Waypoint (RWP) モビリティモデルに従って移動するシナリオを想定する。シミュレーション時間 S は 300 秒とし、ノードの移動速度は $[1.0, 2.0]$ m/s の一様分布に従うものとする。RWP における停止時間は 0 とする。アルゴリズムの特性を調べるために、無線の揺らぎやパケットロスによる影響は排除する。このため、ある 2 ノード間の距離が無線到達距離 R 以内であるとき、必ず通信できるような理想的な無線通信モデルを考える。その他のパラメータは下記の通りである。タイムスロットの間隔 T は 1.0 秒とする。各タイムスロットで更新される軌跡長 L は 10 とする。現在位置 $p_i^{[k]}$ を推定するために用いられる軌跡の長さ W も同様に 10 とする。無線到達距離 R は 10m とする。ノード数 n は 200 とする。また、1 ホップ送信間隔 F は 1、2 ホップ送信間隔 F^+ は ∞ とする（すなわち、今回のシミュレーションでは 2 ホップ送信を行わない）。

シミュレーションでは式 (2) および (3) で定義される位置誤差を評価する。式中の $p_u^{[k]}$ は、タイムスロット $\phi_u^{[k]}$ ($k \leq k'$) における $p_u^{[k]}$ の値とする。また、 $q_u^{[k]}$ はタイムスロット $\phi_u^{[k]}$ におけるノード u の正しい位置とする。以降、 l を k からの経過時間 (elapsed time slots) と呼ぶ。直感的に $Error(k, l)$ はすべてのノードの k 番目のタイムスロットにおける位置を $(k+l)$ 番目のタイムスロットで参照されたときの平均誤差を表し、 $Error(l)$ はシミュレーション全体を通じた $Error(k, l)$ の l に対する平均を表す。ここで、 $Error(0)$ を *realtime error*、 $Error(L-1)$ を *final error* と呼ぶ。ただし、 $L \leq l$ において位置の更新は行われない。

$$Error(k, l) = \frac{1}{n} \sum_{u=1}^n \|q_u^{[k]} - p_u^{[k+l]}\| \quad (2)$$

$$Error(l) = \frac{1}{S/T} \sum_{k=1}^{S/T} Error(kT, l) \quad (3)$$

はじめに、軌跡を交換することにより、非通信制約が増加し、その結果位置の誤差が改善されることを示す。提案手法において、ノード u は $\phi_u^{[k]}$ においてそのタイムスロットにおける位置 $p_u^{[k]}$ を推定し、続く $L-1$ 個のタイムスロット ($\phi_u^{[k+1]}, \phi_u^{[k+2]}, \dots, \phi_u^{[k+L-1]}$) で $p_u^{[k]}$ を更新する。図 4 に $Error(l)$ ($0 \leq l < L$ かつ $L=10$) の値と、 $\phi_u^{[k+1]}, \phi_u^{[k+2]}, \dots, \phi_u^{[k+L-1]}$ ($0 \leq k \leq S/T-L$) において位置を更新するのに使用された通信制約および非通信制約の数を示す。図 4 より、時間が経過するにつれ、新しい非通信制約を得ることができるため、位置誤差が減少していることが分かる。

次に、1 ホップ隣接ノードの軌跡情報を交換することにより、最終的には 2 ホップ隣接

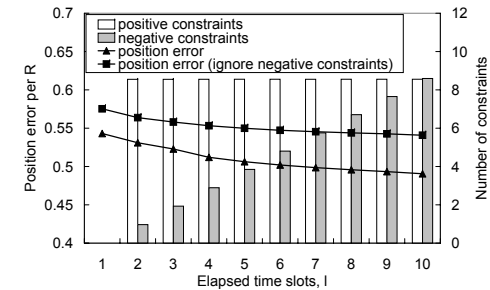


図 4 経過時間 l に対する通信制約と非通信制約の数および位置誤差
Fig. 4 Number of positive/negative constraints and position error (vs. elapsed time slots l).

ノードの位置情報と同数の制約を得ることができ、2 ホップ隣接ノードの位置情報から推定するよりも推定結果が良くなることを示す。2 ホップで隣接ノードの位置情報を交換する場合に得られる非通信制約数と位置誤差を評価するため、 $L=2, F^+=1$ とした場合のシミュレーションを行った。 $L=2, F^+=1$ における $p_u^{[k]}$ の推定では、 $\phi_u^{[k]}$ で得られる通信制約数は 8.55、非通信制約は 9.16 であった。これは図 4 より、 $L=10, F^+=\infty$ とした場合に $\phi_u^{[k+L-1]}$ で得られる制約数とほぼ同数である。これより、ノード自身の推定軌跡情報を交換することにより、2 ホップで位置情報を送信した場合と同様の数の制約数を少し遅れて得ることができることを示している。また、 $L=2, F^+=1$ における final error は $0.81R$ であるが、 $L=10, F^+=\infty$ における final error はほぼ同数の制約を用いているにもかかわらず $0.50R$ と良い。2 つのケースの違いは、非通信制約を得るタイミングである。前者では、ほぼリアルタイムに非通信制約を得ることができるが、これにより 3.1 章の図 3(e) で述べたような非通信制約が位置の更新を妨害してしまうケースが増え、短期間で最適な解を見つけるのが難しくなると考えられる。一方で、後者では遅れて非通信制約を得ることができるため、通信制約により十分実際の位置と近い解を推定したのち非通信制約を用いた解の修正を行うことができ、推定結果が良くなっていると考えられる。このように提案手法では、軌跡情報の交換により、通信制約をリアルタイムで、非通信制約を時間をかけて得ることができ、その結果誤差が良くなることが分かる。

4. 性能解析

本章では、精度に影響を与えるいくつかの要因についてシミュレーションを通してその影響を解析する。また、提案手法の通信量についても解析を行う。

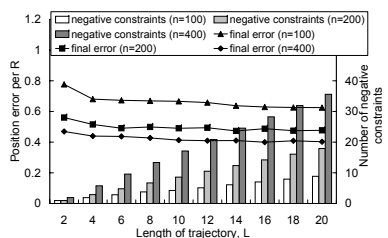


図5 軌跡長 L に対する通信制約と非通信制約の数と final error

Fig.5 Number of positive/negative constraints and final error (vs. trajectory length L).

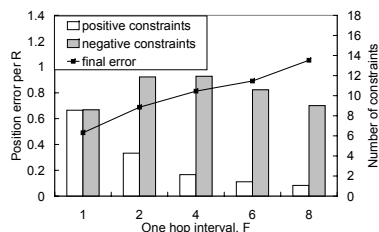


図6 1 ホップ送信間隔 F に対する通信制約と非通信制約の数および final error

Fig.6 Number of positive/negative constraints and final error (vs. 1-hop forwarding interval F).

4.1 推定精度

軌跡長 はじめに軌跡長 L が final error に与える影響を評価する。図5に軌跡長 L に対する final error と非通信制約数を示す。なお、このシミュレーションでは、隣接ノード数による影響を排除するため、ノード数 n を 100, 200 および 400 とした場合それぞれについて評価する。結果より、 L の値が大きいほど final error は低減することが分かる。これは、 L が長いほどより多くの非通信制約をもたらす、その結果誤差が良くなるためである。また、 $n=100, 200, 400$ の場合の平均通信制約数はそれぞれ 4.27, 8.55, 16.99 である。以降、非通信制約が通信制約と同数以上得られれば十分と考え、 $L=10$ と決定する。

1 ホップ送信間隔 図6に、1 ホップ送信間隔 F が final error に与える影響を示す。結果より、1 ホップ送信間隔が長くなるほど、得られる通信制約数が減少するため誤差が大きくなる。これに対し、非通信制約の数は $F=2$ および $F=4$ において最大となる。この理由として、 F が 6 以上の場合、あるノードの無線範囲に滞在する時間を考慮すると隣接ノードからの Hello メッセージを受信できない可能性があり、通信制約と非通信制約の和は減少する。一方、 $F=2$ または 4 の場合、無線範囲内に存在する隣接ノードから必ず 1 回は Hello メッセージを受信する可能性が高いため、通信制約と非通信制約の和は $F=1$ の場合とほとんど変わらない。よって $F=2$ または 4 の場合、非通信制約の数が増加する。

2 ホップ送信間隔 各ノードは F^+ ごとに送信される Hello メッセージにおいて、自身の軌跡に加えて 1 ホップ隣接ノードの軌跡情報も含める。図7(a)にさまざまな F^+ での時間経過とともに誤差の変化を示す。またその際、推定に利用された非通信制約数を図7(b)に

示す。なお、通信制約数は 8.55 である。結果より、2 ホップ送信間隔が短いほど誤差が大きいたことが分かった。この理由として、繰り返し位置を修正するというアルゴリズムの特性から、2 ホップ送信間隔が短いと制約数が増加し、短期間で数多くの制約を満たすような解を見つけることが難しく、これにより次のタイムスロットにおける位置の初期解を推定する際に、誤差が伝播してしまうためと考えられる。この理由を裏付けるために、図8にリアルタイムからある一定期間の間、非通信制約を利用しない(通信制約と移動制約のみを利用して推定する)場合の位置誤差の値を示す。結果より、初期のスロット、特に最初の 2 スロット目まで非通信制約を用いなかった場合、realtime error ならびに final error が大きく向上しているのが分かる。また、final error の誤差が約 $0.45R$ であり、 $F^+=\infty$ (2 ホップ送信を行わない) 場合よりも良くなっている。一方で、最初の 2 スロット以上で非通信制約を用いなかった場合、推定に利用した非通信制約数が減少するため、final error が悪化する。よって、2 ホップ送信を行う場合には、最初の 2 スロットまでの非通信制約を利用しないのが望ましい。

ノード密度と無線到達距離 異なるノード数 n や無線到達距離 R でのノード密度による影響を評価する。ここで、 d_n は隣接ノード数の平均とする。はじめに図9より様々な R や d_n による final error を示す。結果より、ノード密度が高いほど推定結果が良くなるのが分かる。特に、 n がいずれの場合でも d_n が 20 以上である場合、final error は $0.2R$ 以下である。また、ノード数 n が小さい場合、同じノード密度でも誤差が良くなる。これは、同じノード密度においてノード数 n が小さいということは、無線到達距離 R が大きいことを意味し、ランドマークと通信する機会が増加するためである。次に図10に様々な n での final error の値を示す。結果より、無線到達距離が 15m よりも小さい場合、無線到達距離が長いほどランドマークと通信できる機会が増えるため誤差が小さくなると考えられる。一方で、無線到達距離が 20m 以上である場合、通信制約が緩和されるため誤差が大きくなる。

4.2 通信量

本節では、提案手法の通信量について考察する。ある時刻における位置座標は絶対位置座標を用いた場合でも、 $4\text{byte} \times 2 = 8\text{byte}$ で表現することができる。このため、一つのメッセージサイズは $8L\text{byte}$ である。また、 F^+ サイクルごとに送信される 2 ホップ情報を含むメッセージのサイズは各ノードの平均リンク数を n_l とすると、平均 $8Ln_l$ である。すなわち、毎秒の平均メッセージサイズは $8L + (8Ln_l / F^+)$ となる。 $n=200, R=10$ の場合、 $n_l=8.54$ であるため、 $L=10$ の場合の平均メッセージサイズは 250byte であり、各ノードが消費する帯域幅は 2kbps である。これは IEEE802.15.4 など十分実装できる値である。

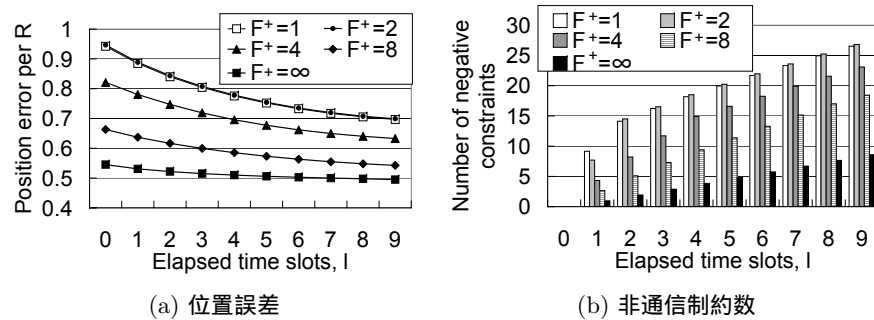


図7 さまざまな2ホップ送信間隔 F^+ における経過時間 l に対する通信制約と非通信制約数および位置誤差
Fig.7 Number of positive/negative constraints and position error (vs. elapsed time slots l) for various values of 2-hop forwarding interval F^+ .

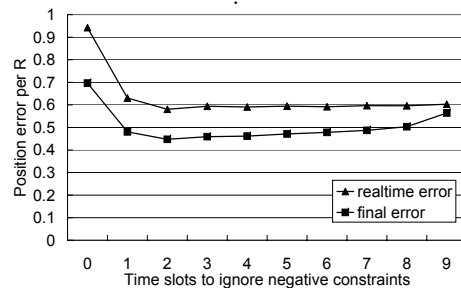


図8 非通信制約を使用しない期間に対する位置誤差 ($F^+ = 1$)
Fig.8 Position error (vs. time slots to ignore negative constraints). $F^+ = 1$

5. 性能評価

本章では、提案手法の性能を評価するために2種類のシミュレーションを行う。1つ目の性能評価では、実環境において提案手法の精度に影響を与える要因について評価を行う。前章では主に理想的な環境における提案手法の誤差や通信量について解析を行ったが、実際の環境ではパケットロスや無線の揺らぎなどが影響を与えとされる。2つ目の性能評価は、既存のリアルタイム位置推定手法との比較である。提案手法の軌跡交換による効果により既存手法よりも高精度な推定が実現できていることを示す。なお、シミュレーション環境は、前章で用いたシナリオおよびパラメータを想定する。ただし $F^+ = 4$ とする。

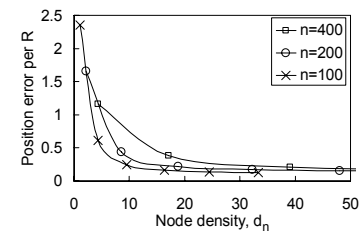


図9 ノード密度に対する final error
Fig.9 Final error (vs. node density).

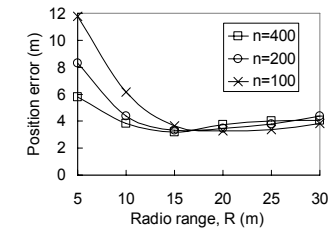


図10 無線到達距離に対する final error
Fig.10 Final error (vs. radio range).

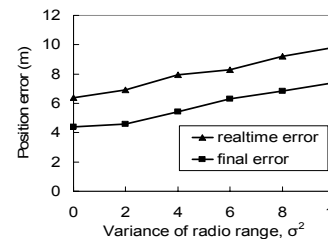


図11 無線到達距離の分散に対する位置誤差
Fig.11 Position error (vs. variance of radio range).

表2 メッセージロスモデル下での position error
Table 2 Position error (with channel contention model).

n	100	200	300
パケットロス率	0.01	0.05	0.17
Final error (モデル有)	0.62m	0.46m	0.37m
Final error (モデル無)	0.60m	0.46m	0.38m

5.1 実環境への適用

無線伝播距離の不均一性 実際の環境では電波の物理的な特性により無線到達距離は不均一な値をとる。このような不均一な無線到達距離が精度に与える影響を観察するため、通信ごとに無線到達距離を正規分布 $R = N(10m, \sigma^2)$ に基づくランダム値で与え、評価した。図11のように様々な分散 σ^2 による realtime error と final error を示す。結果より無線到達距離の分散値が大きいくほど、通信制約が不正確となり誤差が悪化することが分かった。パケットロス ここまでのシミュレーションではパケット衝突等によるメッセージのロスを考慮していなかった。前章での結果より、ノード密度が高いほどよい精度が得られることを示したが、実際にはノード密度が高くなるほど、Hello メッセージのサイズが大きくなり、パケットの衝突などによりメッセージ交換が正常に行えない可能性が高まる。パケットロスによる影響を観察するために、 $P = (1-p)^k$ で定義される衝突モデルを仮定する。なお、 P を隣接ノードからのパケット受信成功確率、 p を帯域占有率とする。ここで、IEEE802.15.4.などの PAN 通信デバイスを想定し、帯域容量は 250kbps とする。表2にモデルの有無に

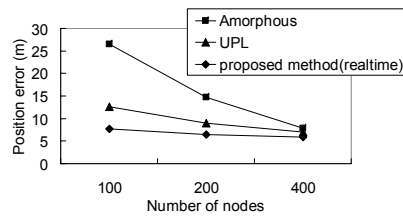


図 12 他手法との比較

Fig. 12 Comparison with other algorithms.

よる final error の値を示す．結果より，ノード数が多いほど，パケットロス率が悪化し，final error も悪化することが分かる．しかし，帯域容量を考慮しても提案手法は十分な精度を実現可能であるといえる．

5.2 既存手法との比較

提案手法の性能を分散型リアルタイム推定手法である，Amorphous¹⁰⁾ と UPL¹³⁾ と比較した．Amorphous は，ランドマークからマルチホップで情報を伝播させ，ホップ数からランドマークからの距離を推定し，位置を推定する手法であり，UPL¹³⁾ では，端末の移動速度の最大値を仮定し，移動端末の存在範囲を推定する手法である．提案手法の realtime error と Amorphous および UPL の推定誤差を図 12 に示す．結果より，提案手法ではノード密度が低い状況でも軌跡の交換により他手法よりも誤差を抑制できることが分かった．

6. ま と め

本稿では，我々が検討してきた位置推定手法についてアルゴリズムの特性解析とそれに基づくプロトコルの検討および性能評価を行った．シミュレーション結果から，提案手法では移動軌跡情報の交換により，通信制約をリアルタイムに，非通信制約を徐々に得ることができ，これにより既存手法よりも高精度な分散位置推定を実現できていることを示した．今後，実際の無線端末を用いた実験により，手法の評価を行う予定である．

参 考 文 献

- 1) Fleuret, F., Berclaz, J., Lengagne, R. and Fua, P.: Multicamera People Tracking with a Probabilistic Occupancy Map, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.30, No.2, pp.267–282 (2008).
- 2) Morris, B. and Trivedi, M.: A Survey of Vision-Based Trajectory Learning and Analysis for Surveillance, *IEEE Transactions on Circuits and Systems for Video*

Technology, Vol.18, No.8, pp.1114–1127 (2008).

- 3) Kleiner, A. and Sun, D.: Decentralized SLAM for pedestrians without direct communication, *Proc. of International Conference on Intelligent Robots and Systems, IROS 2007*, pp.1461–1466 (2007).
- 4) Bahl, P. and Padmanabhan, V.N.: RADAR: an in-building RF-based user location and tracking system, *Proc. of INFOCOM 2000*, pp.775–784 (2000).
- 5) Priyantha, N.B., Chakraborty, A. and Balakrishnan, H.: The Cricket Location-Support System, *Proc. of MobiCom 2000*, pp.32–43 (2000).
- 6) Shang, Y., Rml, W., Zhang, Y. and Fromherz, M.: Localization from Connectivity in Sensor Networks, *IEEE Transaction on Parallel and Distributed Systems*, Vol.15, No.11, pp.961–974 (2004).
- 7) Cabero, J.M., la Torre, F.D., Sanchez, A. and Arizaga, I.: Indoor people tracking based on dynamic weighted multidimensional scaling, *Proc. of MSWiM 2007*, pp.328–335 (2007).
- 8) Fujii, S., Uchiyama, A., Umedu, T., Yamaguchi, H. and Higashino, T.: An Off-line Algorithm to Estimate Trajectories of Mobile Nodes Using Ad-hoc Communication, *Proc. of PerCom 2008*, pp.117–124 (2008).
- 9) He, T., Huang, C., Blum, B.M., Stankovic, J.A. and Abdelzaher, T.: Range-Free Localization Schemes for Large Scale Sensor Networks, *Proc. of MobiCom 2003*, pp.81–95 (2003).
- 10) Nagpal, R., Shrobe, H. and Bachrach, J.: Organizing a global coordinate system from local information on an ad hoc sensor network, *Proc. of IPSN 2003*, pp.333–348 (2003).
- 11) Li, M. and Liu, Y.: Rendered path: range-free localization in anisotropic sensor networks with holes, *Proc. of MobiCom 2007*, pp.51–62 (2007).
- 12) Hu, L. and Evans, D.: Localization for Mobile Sensor Networks, *Proc. of MobiCom 2004*, pp.45–57 (2004).
- 13) Uchiyama, A., Fujii, S., Maeda, K., Umedu, T., Yamaguchi, H. and Higashino, T.: Ad-hoc Localization in Urban District, *Proc. of INFOCOM 2007 Mini-Symposium*, pp.2306–2310 (2007).
- 14) 野村崇志, 梅津高朗, 山口弘純, 東野輝夫: 近隣ノードの移動履歴情報を用いたアドホックネットワーク上の位置推定手法, 電子情報通信学会技術研究報告, Vol.107, No.517, pp.37–42 (2008).
- 15) 野村崇志, 内山彰, 梅津高朗, 山口弘純, 東野輝夫: 無線端末間の通信情報を利用した移動軌跡推定手法の実環境を想定した評価事例および実機による性能評価, マルチメディア, 分散, 協調とモバイル (DICOMO2008) シンポジウム論文集, pp.1066–1074 (2008).