

# すれ違い通信を利用した GPS の精度向上手法の提案

荻野正<sup>†1</sup> 北上真二<sup>†2</sup> 大岸智彦<sup>†3</sup> 宮西洋太郎<sup>†4</sup> 浦野義頼<sup>†2</sup> 白鳥則郎<sup>†2</sup>

**概要**：現在多くの携帯端末には GPS センサが搭載され、センサが取得した位置情報をベースにしたサービスが提供されている。GPS センサが取得する位置情報は、電波条件が良い場合には数 m 以下の精度を実現可能であるが、市街地等高層の建物に囲まれた場所では数十 m 以上の誤差が発生する。本論文では、センサを搭載した端末間で近距離無線通信を利用して位置情報を交換し、統計的な処理を行うことで位置情報の精度を高める手法について提案する。提案手法の実現可能性について、近距離通信及び GPS データに関する予備実験を行い、フィージビリティを確認した。

## 1. はじめに

現在、多くの携帯電話、携帯端末には GPS センサが搭載されており、位置情報が地図アプリケーションを含む多くのアプリケーションに利用されている。GPS センサは、条件の良い場合は数 m 以下の精度を実現可能であるが、街中など電波条件が良くない場合には、数十 m 以上の誤差が発生する可能性がある。本論文では、携帯端末間の近距離通信を利用して、端末間で誤差情報を共有することで、より精度の高い位置情報を得るための手法について提案する。

## 2. IoT アーキテクチャ

本章では、今回の提案システムと IoT アーキテクチャの関係について簡単に説明する。

携帯端末だけでなく、家電、自動車、産業機器など様々なモノ（デバイス）をインターネットに接続し、統合的に監視・運用・制御しようという IoT システムについては様々な新しい技術・応用が提案されている。IoT システムは、通常は、デバイスをインターネット経由でサーバに接続し、サーバ側でデータ処理を行う形態が多い（図 1（1））[1]。しかし、デバイスの数が増え扱うデータ量が増加すると、

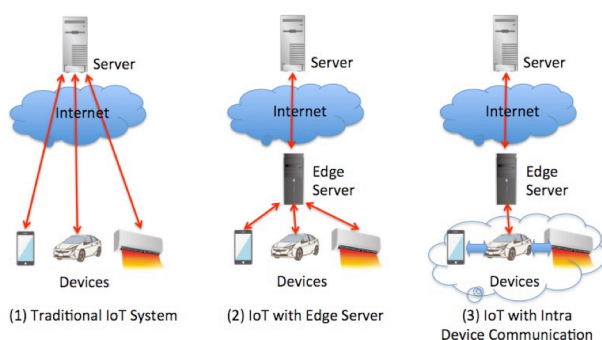


図 1. IoT システムのアーキテクチャ

ネットワークのトラフィックの低減や反応時間の削減等のため、処理の一部をデバイスに近いエッジサーバで扱うシステム形態が提案されている（図 1（2））[2]。今後、IoT デバイスの数がさらに増え、処理能力も増加すると、デバイス同士で通信を行う形態も現れてくることが予想される（図 1（3））。今回提案する手法は、このデバイス間通信を利用して GPS 情報を交換し、位置情報の精度を向上させるものである。本研究は、デバイス同士の通信を利用した IoT アーキテクチャの一応用として、GPS 情報の高精度化を目指すものであり、今後さらに本アーキテクチャの別アプリケーションへの適用のベースとすることを考えている。

## 3. GPS の誤差と精度向上

GPS システムは、複数の測位用衛星からの電波を受信して、地上での位置を特定するシステムである[3]。GPS システムでの誤差には、衛星のクロック誤差、衛星の位置情報の誤差、電離層での電波の遅延等様々な要因があるが、大きく分類するとバイアス誤差とランダム誤差がある。バイアス誤差は、主に衛星や電離層などの外部要因で発生する誤差で、精度の高い基準局で誤差情報を計算し配布することで誤差の補正を行う。ランダム誤差は、平均値が 0 になる事を利用し、長時間測定することで誤差を縮小する。従って、比較の見晴らしの良い場所に専用アンテナを設置し、長時間固定されているような場合は、1 m 以下の誤差を実現することも可能である。逆に、以上のような理想的な条件にない、例えば移動する小型の携帯端末のような場合には、精度はかなり劣化するのが実態である。

GPS システムの精度は、通常は平均誤差半径（Circular Error Probability, 以下 CEP と表記）で示される。CEP は、50% 確率円の半径のことで、その半径の円の中に正解値が存在する確率が 50% になる値である。誤差が正規分布であると仮定すると、標準偏差を  $\sigma$  とすると  $CEP = 1.1774 \sigma$  である。

現在市販されている GPS チップ単体の精度(CEP)は例えば u-blox 社の UBX-G7020 では、2m から 2.5m 程度である[4]。携帯電話に搭載されている GPS チップの仕様について

†1 明星大学 Meisei University  
†2 早稲田大学 Waseda University  
†3 KDDI 総合研究所 KDDI Research, Inc.  
†4 株式会社アイエスイーエム ISEM, Inc.

は公開されていないが、ほぼ同程度であると予想される。しかし、携帯電話のGPSでの位置測位は、街中等電波条件の良くない場合には、10m以上の誤差が発生することもある。簡単にGPS情報を測定した結果を図2に示す。



図2. 携帯電話でのGPS測定実験

本データは、図の右側から左方向へ、真ん中の広い道路の左側(図では下側)を歩きながら、アンドロイド端末のGPS装置の返す位置情報をプロットしたものである。アンドロイド端末は、wifiやキャリアとの通信機能はオフにして、GPSセンサのみからの位置情報を記録した。

図で分かるように、本来道路の左側を歩いていたものが、GPS情報からは道路の右側に近い位置を返してくる場所もある。この誤差が、この周辺のGPS装置で共有できる補正量であれば、その値を共有することでより正確な位置情報を得ることができる(図3)。



図3. GPS誤差の補正

神谷、荻野の実験によれば[5]、2台のGPSセンサの情報を使って、センサからの位置情報を補正することで、より正確な位置情報が得られることが分かっている(図4)。

[5]では、位置情報の補正方式に関する基本的な検討結果のみの報告であり、位置情報の共有方式については検討していない。

本論文では、多数の携帯端末間で近距離通信を用いることでGPSの誤差情報を共有し、高精度の位置情報を得る手法について提案する。

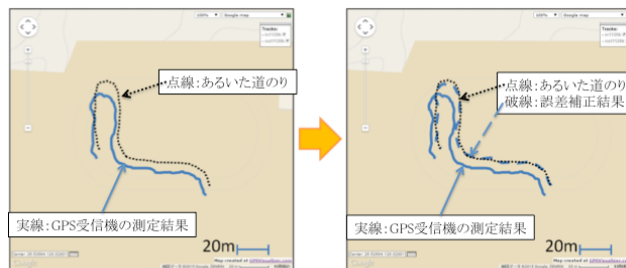


図4. GPSデータの補正実験

## 4. 提案方式

### 4.1 基本的な考え方

提案方式では、デバイス間で、近距離通信を使ってお互いが保持している誤差情報を交換し、交換した誤差情報に基づいて誤差情報を更新する。以下に提案方式の基本的なアイデアを説明する。説明を簡単にするために、支障がない限り1次元上の位置で説明する。

デバイスの存在する位置を  $x_{actual}$  とする。GPSセンサの返す値  $x_{measurement}(t)$  には、誤差  $E(t)$  が含まれる。

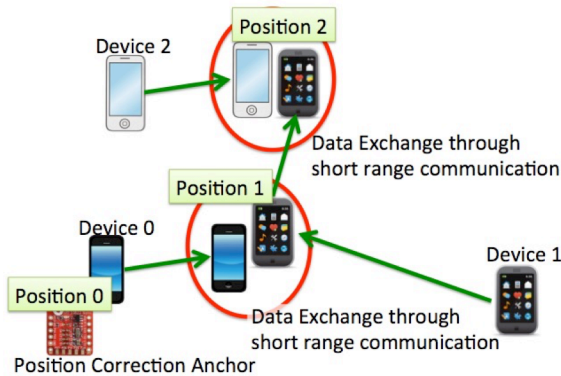
$$x_{measurement}(t) = x_{actual} + E(t) \quad (1)$$

複数のデバイスが、同じ位置  $x_{actual}$  に存在する場合、これら複数のデバイスが持つGPSセンサの返す値のデータの加算平均を計算することができる。

$$average(x_{measurement}^i(t)) = x_{actual} + average(E^i(t)) \quad (2)$$

GPSセンサの返す値を正規分布と仮定すると、複数の測定値の平均を計算することで、標準偏差の値を元のGPSデータ単独の値よりも小さくすることができる。これは、GPSデータの場合には、より小さいCEPの値を得ることができるということになる。

実際には、多数のデバイスが同時に同じ場所に存在する



Overview of GPS improvement system

図5. 提案方式の概要

可能性は少ないため、2つのデバイスが近距離に存在した時に、誤差情報の交換を行う方式を提案する。

また、精度を高めるために、位置情報が予め分かっている位置補正アンカ(Position Correction Anchor)を導入する。位置補正アンカは、位置が固定されたデバイスで、移動するデバイスがアンカの近傍に来た時にその場所の位置情報を知ることができ、自分の誤差情報を更新することができる。(図5)

#### 4.2 位置の推定計算

提案方式での、位置の推定方法について以下に説明する。

GPSの誤差情報は正規分布であると仮定する。デバイス0のGPSセンサの値が $x_0$ 、GPSデータの標準偏差が $\sigma_0$ の時、デバイス0の存在位置 $x$ を推定するための確率密度関数は、正規分布の確率密度関数から以下の式となる。

$$f(x; x_0, \sigma_0^2) = \frac{1}{\sigma_0 \sqrt{2\pi}} \exp \left[ -\frac{1}{2} \left( \frac{x - x_0}{\sigma_0} \right)^2 \right] \quad (3)$$

デバイス1についても、同様に以下の式で求められる。

$$f(x; x_1, \sigma_1^2) = \frac{1}{\sigma_1 \sqrt{2\pi}} \exp \left[ -\frac{1}{2} \left( \frac{x - x_1}{\sigma_1} \right)^2 \right] \quad (4)$$

デバイス0とデバイス1が同じ位置に存在する事が分かっている場合、その位置の確率密度関数は以下の式で求められる。

$$\begin{aligned} f(x; x_A, \sigma_A^2) &= f(x; x_0, \sigma_0^2) \times f(x; x_1, \sigma_1^2) \\ &= \frac{1}{\sigma_A \sqrt{2\pi}} \exp \left[ -\frac{1}{2} \left( \frac{x - \frac{\sigma_1^2 x_0 + \sigma_0^2 x_1}{\sigma_0^2 + \sigma_1^2}}{\sqrt{\frac{\sigma_0^2 \sigma_1^2}{\sigma_0^2 + \sigma_1^2}}} \right)^2 \right] \end{aligned} \quad (5)$$

但し、

$$\sigma_A = \sqrt{\frac{\sigma_0^2 \sigma_1^2}{\sigma_0^2 + \sigma_1^2}} \quad (6)$$

$R_0 = 1/\sigma_0^2$ ,  $R_1 = 1/\sigma_1^2$ ,  $R_A = 1/\sigma_A^2$  という表記を使うと、

$$x_A = \frac{R_0 x_0 + R_1 x_1}{R_0 + R_1} \quad (7)$$

$$R_A = R_0 + R_1 \quad (8)$$

となる。すなわち、 $1/(\text{分散})$ という変数を導入することで、デバイス0とデバイス1の存在位置を推定することが出来る。式(7)の $x_A$ が、デバイス0及びデバイス1の推定位置(期待値)である。 $R (=1/(\text{分散}))$ を本システムでの信頼度と呼ぶことにする。 $R$ は常に正の値であり、式(8)で分かるように、推定された位置の $R_A$ は、 $R_0$ ,  $R_1$ のいずれよりも大きくなる。

GPSの誤差のうち、デバイス依存のランダム誤差については、上記の方法で標準偏差の値を減少させることが出来るが、電離層でのずれやマルチパス等、デバイス外の要因によるバイアス誤差は、該当箇所のすべてのデバイスに同等の誤差を発生させるため、データ交換のみでは削減することができない。この誤差を削減させるために、位置補正アンカを導入する。このアンカは、自分の正確な位置情報を保持し、GPS情報によらずにその情報を周辺のデバイスに知らせるものである。周辺のデバイスは、アンカの情報と自分のGPSデータから誤差情報を計算し、自らの位置推定に利用すると共に、データ交換を通して他のデバイスに伝える。

アンカで位置情報を修正した直後は、デバイスの推定位置はアンカの位置であり、信頼度 $R$ は最大値に設定する。本システムでは、該当デバイスのGPSセンサチップのCEP値に相当する値とする。

式(8)より分かるように、データ交換後は $R$ の値はデータ交換以前より増加する。その後、時間の経過と共に衛星の位置関係が変化し、またデバイスの移動と共に電波の受信状況が変わることで信頼度 $R$ は減少する。本システムでは、以下の式に従い信頼度 $R$ が減少するものと仮定する。

$$R(\Delta x, \Delta t) = R_0 \exp(-\alpha \Delta x) \exp(-\beta \Delta t) \quad (9)$$

$\alpha$ と $\beta$ は、実際のデータから求めるものとする。

信頼度 $R$ の計算においては、最小値 $R_{min}$ と最大値 $R_{max}$ を設定し、その範囲に収まるように調整する。最小値 $R_{min}$ は、通常のGPSデータの測定値に相当する値とする。最大値 $R_{max}$ は、利用しているGPSセンサのCEPの仕様に相当する値とするが、搭載デバイスやセンサの仕様が未公開な場合も多く、以下の実験では公開されている何社かのチップベンダの公開値(2.5m)を用いた。

#### 4.3 具体的な計算方式

デバイスの位置推定は、以下の計算方法に従う。

デバイスは、誤差情報 $xdiff$ と信頼度 $R$ の2つの変数を持つ。初期値はそれぞれ、0と $R_{min}$ である。デバイスの推定位置 $x$ は、GPSセンサの値 $x_{GPS}$ と誤差情報 $xdiff$ から以下の式で求める。

$$x_{estimated} = x_{GPS} - xdiff \quad (10)$$

デバイスは、A)近傍デバイスとのデータ交換時、B)位置補正アンカとのデータ交換時、C)データ交換せずに一定の時間経過あるいは一定距離移動した時に $xdiff$ と $R$ の値を以下の通り更新する。

##### A) 近傍デバイスとのデータ交換時

デバイスが近傍に他のデバイスを発見した場合は、実際

の推定位置  $x_A$  を式 (7) に従って計算する.  $xdiff$  は,  $x_{GPS} - x_A$  に,  $R$  は式 (8) を使って計算される.

### B) 位置補正アンカとのデータ交換時

デバイスが近傍に位置補正アンカを発見した場合は, 以下の式に従い内部変数の値を更新する

$$xdiff = x_{GPS} - x_{ANCHOR} \quad (11)$$

$$R = 1/\sigma_{CEP}^2 = R_{max}. \quad (12)$$

$xdiff$  は GPS データから計算されるため, 信頼度  $R$  は, GPS チップの信頼度 ( $R_{max}$ ) を超えることはないとしている.

### C) データ交換が行われず一定時間経過あるいは一定距離移動した時

近傍に他のデバイスや位置補正アンカが発見できない場合でも, デバイスの移動や衛星の移動によって  $xdiff$  は以下の式に従って,  $R$  は式 (9) に従って更新する.

$$xdiff(\Delta x, \Delta t) = xdiff_0 \exp(-\alpha \Delta x) \exp(-\beta \Delta t), \quad (13)$$

新しい  $xdiff$ ,  $R$  は最後のデータ交換/位置補正によって更新される. 計算の結果,  $R$  の値が  $R_{min}$  より小さくなった場合には, 自分の GPS データのみに従うということになり,  $xdiff$  は 0 に,  $R$  は  $R_{min}$  に設定される.

## 5. 予備実験

提案手法のフィジビリティを確認するために, 以下の2つの予備実験を行った.

- 近距離デバイスの検出
- GPS データの測定

本章及び次章でそれぞれの予備実験について説明する.

### 5.1 近距離デバイスの検出

今回の提案手法では, 近傍のデバイス間で GPS データの情報を交換することで, 位置情報の精度を向上させる. 近傍のデバイス間の通信 (以降「すれ違い通信」) を行うためには, 赤外線通信, 近距離無線通信などの様々な手法が利用できる. 提案手法では, 2つのデバイスが標準の CEP (約 2.5m) 以内に存在する事を検出する必要がある. 現在多くの携帯端末で利用できる技術で, 提案手法で利用できる技術として Bluetooth Low Energy (以下 BLE) を利用することとした. 予備実験では, 現在の BLE の HW 及び SW の組み合わせを用いて近距離に存在するデバイスが検出できるかどうかを確認する. なお, BLE での近接検知機構については, 田畑らによる先行研究[6]があるが, 端末間の距離として比較的長い距離 (~100m) である等, 本提案方式では利用できないため, 本実験を行っている.

### 5.2 距離の検出方法

理論上は, 障害のない自由空間において, 電波の送信電力 (TXPower) と受信信号強度 (RSSI) が分かれば, 送信

機と受信機の距離は, 以下の式で求めることができる.

$$distance(m) = 10^{\frac{TXPower - RSSI}{20}} \quad (13)$$

但し, TX Powerは1m地点でのdBm, RSSIは dBm

しかし, 実際の利用状況では, 周囲に存在する物で電波の反射や吸収が発生するため, 受信信号強度は周辺環境に大きく左右される. また, 受信信号の RSSI については公開されている API で知ることができるが, 送信電力については, iOS の公開 API では知ることができない. 従って, デバイスが iOS を含む携帯端末の場合は, RSSI のみを用いて相手のデバイスとの距離を推定する必要がある. そこで, デバイス間の距離を RSSI のみから推定できるかどうかを確認する実験を行った.

### 5.3 実験概要

今回の予備実験では, 送信機として2台の携帯電話 (iOS 端末と android 端末) を準備し, 受信機 (iOS 端末) で受信電波の RSSI を測定する. 送信機と受信機の距離を変化させて, RSSI の変化を確認する.

#### 5.3.1 詳細

- 送信側 2機種
  - 1台目 iPhone6S Plus (iOS9)
  - 2台目 ASUS zenfone2 (android5.0)
- 受信側 1機種
  - iPad mini 4 (iOS9)

#### 5.3.2 実験内容

送信側の機器は, BLE の Peripheral 機能を実装し, advertising パケットを連続で送出する.

受信側は, BLE の Central 機能を実装し, 周辺デバイスの検索を行う. advertising パケットを受信すると, 標準 API にてその時の RSSI を記録する.

送信側と受信側の距離は, 0.5m から 5m まで変化させた. 測定は, 同じ距離の元で 10秒間隔で 10回行った.

### 5.4 結果

図6と図7に, それぞれ iPhone と android 端末での結果を示す.

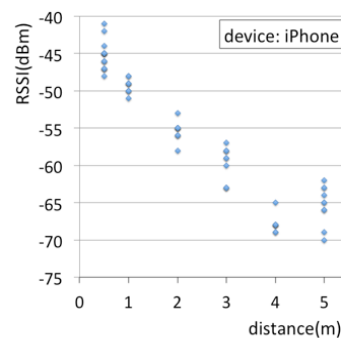


図6. BLE 通信の距離と電波強度実験結果 (iPhone)

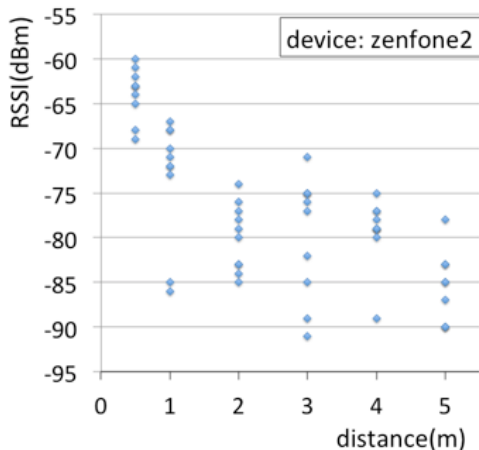


図 7. BLE 通信の距離と電波強度実験結果 (zenfone2)

iPhone の場合は、距離と RSSI の値の関係が比較的安定しており、この場合は RSSI の閾値を約  $-55\text{dbm}$  と設定することで、デバイスが近傍に存在することを検出できると思われる。

一方、android 端末の場合は、距離と RSSI 値の関係が安定せず、この RSSI 値だけから近傍に存在することを推定するのは難しい。そこで、追加的に以下の実験を行った。

### 追加実験

「RSSI の測定を 3 回連続で行い、その平均値を求める。」

### 追加実験の結果

図 8 に追加実験の結果を示す。3 回の平均を取ることで、RSSI 値が安定し、RSSI の閾値を約  $-72\text{dbm}$  と設定することで、近傍のデバイスを検出できると思われる。

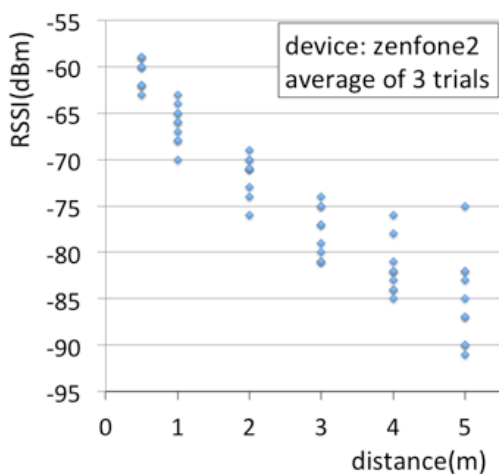


図 8. BLE 通信の距離と電波強度追実験結果

今回の実験では、相手端末の種類によって閾値が異なって

いる。実際の利用場面では、相手の端末の種類は事前にはわからないため、advertising パケット受信後に相手の端末の情報を得た後で、近傍に存在するデバイスかどうかを判断する必要がある。

また、今回の実験では 2 種類のデバイスについて、1 台ずつの実験をしたのみであり、端末の機種や、同じ機種でも異なるデバイスでどうなるかを確認していく必要がある。

## 6. GPS データに関する予備実験

本実験では、実際に携帯端末の GPS 機能を用いて、どの程度の誤差が発生するかとの測定と、提案手法によってどの程度の改善ができるかの確認を行う。

### 6.1 GPS データの誤差確認実験

現在の携帯端末で、どの程度の GPS 測定誤差が出るかの確認実験を行う。実験は 8 台の携帯端末 (iPhone 5 台、android 端末 3 台) を用い、明星大学日野キャンパス内の 8 ヶ所で GPS データを測定した (図 9)。



図 9. GPS データの測定ポイント

GPS データは各測定ポイントで 8 台同時に測定する。測定は、同一ポイントでは 1 分間隔で 2 回測定して、次のポイントに移動する。8 ヶ所の測定で 1 周分の測定とし、実験は 2 周行う。測定データは、8 台  $\times$  8 ポイント  $\times$  2 回 / ポイント  $\times$  2 周で、256 個のデータとなる。

測定結果を図 10 に示す。

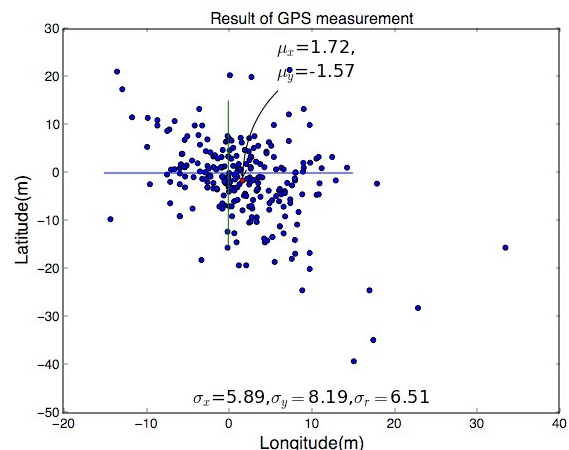


図 10. GPS データの測定結果

各ポイントにおいて、正解値を Google Map を用いて求め、そこからの誤差についてプロットする。誤差は、正解値と測定値の距離を緯度と経度から計算し、その標準偏差を求めた。緯度と経度に分けて別々に平均  $\mu_x$ 、 $\mu_y$  と標準偏差  $\sigma_x$ 、 $\sigma_y$  を求めた。また、各ポイントで、測定データの

平均値からの距離の標準偏差  $\sigma_r$  を求めた。

各場所別のデータを図 1 1 に示す。誤差、標準偏差ともに場所によって大きくばらついていることが分かる。場所 6、7、8 はほぼ四方を建物で囲まれている場所であり、電波の反射の影響を受けている事が分かる

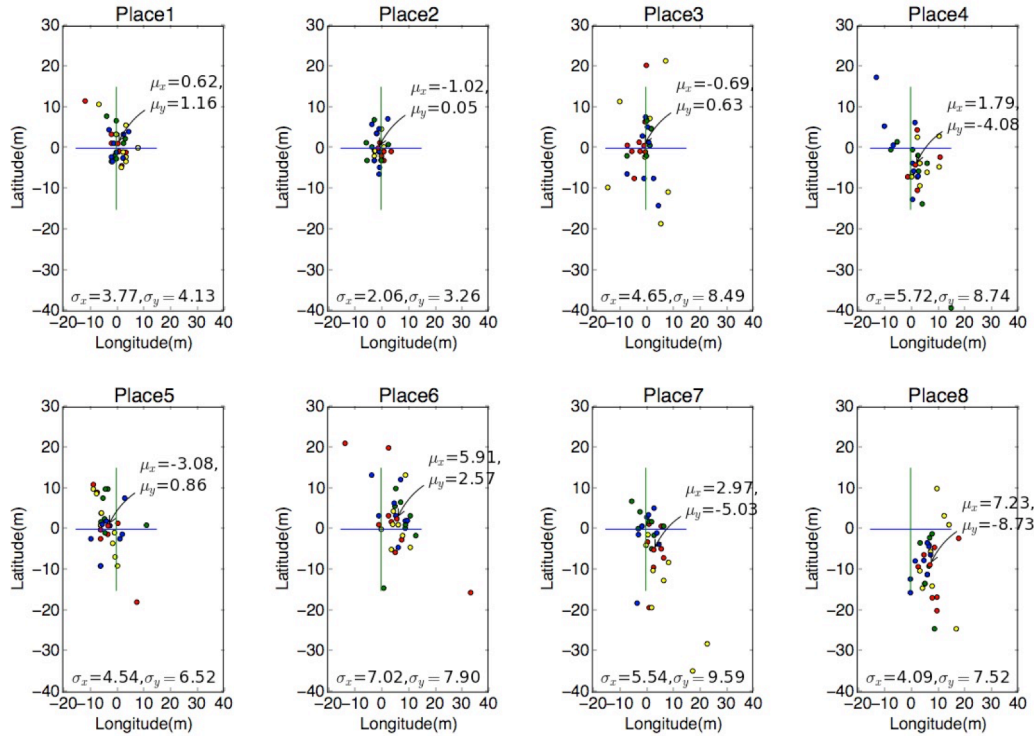


図 1 1. GPS データの測定結果 (場所別)

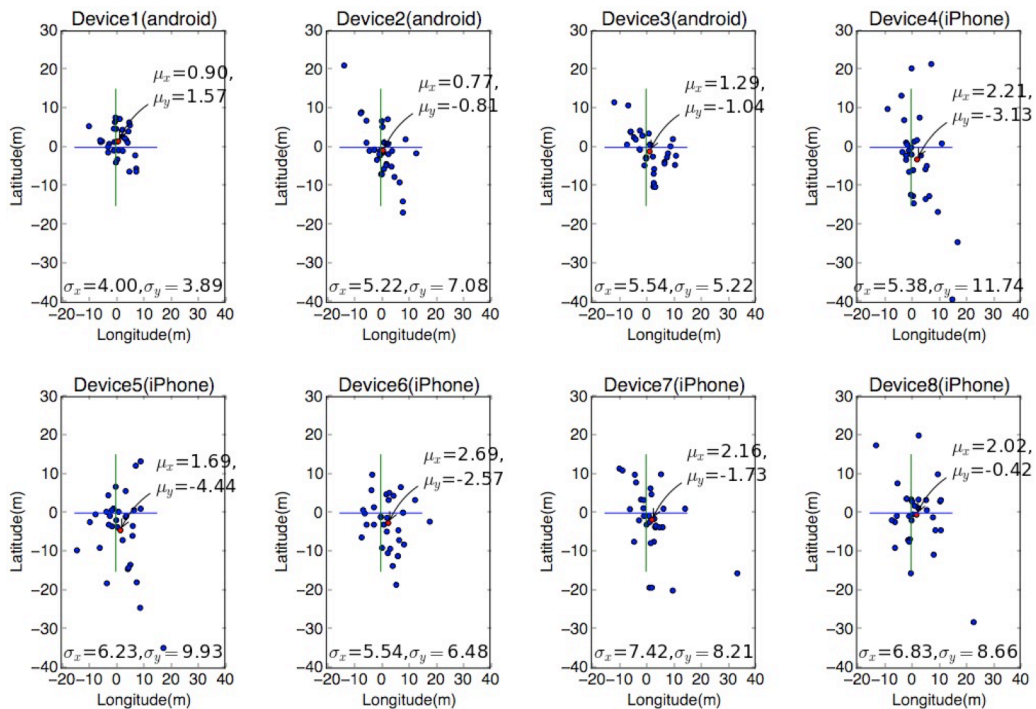


図 1 2. GPS データの測定結果 (デバイス別)

図 1 2 にデバイス別のデータを示す。本実験結果の範囲では、android 端末(デバイス 1 ~ デバイス 3)よりも iPhone 端末(デバイス 4 ~ デバイス 8)の方が誤差、標準偏差ともにやや大きい傾向が見られる。

## 6.2 簡易シミュレーション

上記実験のデータを用いて、提案手法での GPS 情報の精度向上の簡易的なシミュレーションを行った。シミュレーションの手順は、以下の通りである。

1. 同一ポイントの同一時刻の 8 個の観測データを集める。
2. Google Map から求めた正解値を位置補正アンカのデータと仮定し、0 または 1 つの観測データを選択して位置補正を行う。すなわち、式(11)、式(12)を用いて、 $xdiff$  と  $R$  の値を更新する。(0 の場合は、位置補正アンカでの更新を行わない事になる。)
3. 1 組または 4 組の観測データの組を選択し、データ交換を行う。すなわち、式(7)、式(8)を用いて  $xdiff$  と  $R$  の値を更新する。
4. 次の時刻データに進む。この時、 $R$  の値は仮に 1/2 になるとした。 $R_{min}$  の値 (CEP 2.5m と仮定) より小さくなった場合には、 $R_{min}$  とする。
5. 次の地点に移動した場合は、 $\Delta x$  が CEP の値より大きくなるため、 $xdiff$  と  $R$  の値を初期化する。

シミュレーションは以下のパラメータの組み合わせで実施した。

- 位置補正アンカによる補正をする、または、しない
- 各時刻でデータ交換をするデータ組数、1 組または 4 組

図 1 3 に、位置補正アンカあり、データ交換 4 組のときの結果を示す。すべてのシミュレーションの結果(誤差平均と標準偏差)について、表 1 に示す。

## 6.3 結果

4 通りのシミュレーションのいずれにおいても、誤差の標準偏差の値が、オリジナルの値(図 10)よりも削減されていることが分かる。また、4 通りの組み合わせでの結果を比較すると「アンカなし」より「あり」の方が、またデータ交換 1 組より 4 組の方が、精度が高くなっていることが確認できる。誤差の平均値は、位置補正アンカを使うことで改善されている。

## 7. まとめ

本論文では、近距離通信を用いてデバイス間で GPS 情報

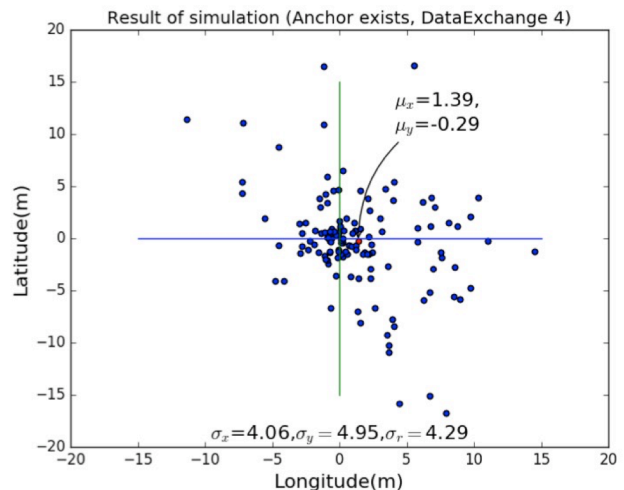


図 1 3. シミュレーション結果  
(アンカーあり, データ交換 4 組)

表 1. シミュレーション結果まとめ

	unit (m)	
	No Anchor	Anchor Exists
Data Exchange: 1	$\mu_x=1.74, \mu_y=-1.67$ $\sigma_x=5.70, \sigma_y=7.80, \sigma_r=6.41$	$\mu_x=1.29, \mu_y=-1.14$ $\sigma_x=5.20, \sigma_y=6.89, \sigma_r=6.50$
Data Exchange: 4	$\mu_x=1.72, \mu_y=-1.71$ $\sigma_x=4.89, \sigma_y=6.42, \sigma_r=5.20$	$\mu_x=1.39, \mu_y=-0.29$ $\sigma_x=4.06, \sigma_y=4.95, \sigma_r=4.29$

を交換することで、位置情報の精度を向上させる方式について提案した。提案方式で必要な、近傍デバイスを検出する方式については、現在普及している BLE 通信を用いて、信号強度から検出が可能であることを確認した。また、携帯電話の GPS 機能を用いて GPS データの測定とシミュレーションを行い、単体では GPS チップの CEP の値より精度は悪いが、提案方式により GPS データの誤差、標準偏差を削減できる事を確認した。

提案方式について、現段階では、デバイスや端末の種類・台数などにおいて、まだ限定された範囲でのフィージビリティの確認にとどまっているため、今後デバイスの種類や台数を増やしての測定、すれ違いアプリケーションの実装、より実動作環境に近い環境での実験などを通じて、本方式の検証作業を行っていく。

また、本提案システムは、IoT アーキテクチャとして、デバイス間通信も活用することで、システムとしての適用範囲が増えていく事を想定している。今後、さらに新しい応用分野についても検討を行っていく。

## 参考文献

- [1] Ala Al-Fuqaha, Mohsen Guizani, Mehdi Mohammadi, Mohammed Aldehari, Moussa Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols and Applications", IEEE Communications Surveys & Tutorials, Volume: 17, Issue: 4, Fourthquarter 2015
- [2] D.Okanohara, S.Hido, N.Kubota, Y.Unno, H.Maruyama, Krill: An

Architecture for Edge-Heavy Data. Third Workshop on Architectures and Systems for Big Data, Tel Aviv, June, 2013.

- [3] 杉本末雄,柴崎亮介, GPS ハンドブック.朝倉書店,2010,pp.1
- [4] “u-blox 社ホームページ UBX-G7020 製品情報”.  
[https://www.u-blox.com/sites/default/files/products/documents/UBX-G7020\\_ProductSummary\\_%28UBX-13003349%29.pdf](https://www.u-blox.com/sites/default/files/products/documents/UBX-G7020_ProductSummary_%28UBX-13003349%29.pdf) (参照 2016-11-23)
- [5] 神谷潤, 荻野正, .M2M システムでの位置情報検出手法に関する研究. 信学技報, vol. 114, no. 468, SWIM2014-31, pp. 25-29, 2015 年 2 月.
- [6] 田畑亮馬,林亜梨沙,徳永清輝,佐伯幸郎,まつ本真佑,中村匡秀. すれちがいフレームワークのための BLE を用いた近接検知機構の実装と評価. 信学技報 115(486), 19-24, 2016-03-03