

# 複数のウェアラブルデバイスによる 時刻同期センシングに関する検討

幸島 明男<sup>†</sup>

<sup>†</sup> 国立研究開発法人 産業技術総合研究所 人間拡張研究センター

## 1 はじめに

近年、一人の利用者が、腕時計型や携帯型など、複数のウェアラブルデバイスを日常的に利用するようになりつつある。このような複数のウェアラブルデバイスから得られた多様なデータを統一的に扱う際の問題として、センシングデータの時刻同期がある。身体の動作に伴い発生する物理的事象を多次元の時系列データとして扱うには、複数のセンサから得られたデータのタイムスタンプが同期していなければならない。

本論文では、無線 LAN アクセスポイントを用いたスマートウォッチの時刻同期に関する検討 [1] を踏まえて、モバイル環境における時刻同期について検討する。具体的には、携帯可能な時刻サーバと無線通信するウェアラブルデバイスのネットワークを構築し、それらの間の時刻同期を行う手法について実験的に検討する。

## 2 モバイル時刻同期センシング

本研究で提案するモバイル時刻同期センシングシステムの概要を図 1 として示す。本システムは、GPS モジュールを接続した Linux デバイス (Raspberry Pi)、モバイルバッテリー、スマートウォッチ (Sony SmartWatch3) から構成される。

Linux デバイスはモバイルバッテリーを電源として動作し、GPS モジュールと接続している。この GPS モジュールから得られる GPS 信号を用いて日本標準時に正確に時刻同期させることで時刻サーバを実現する。次に、内蔵 Wi-Fi モジュールを用いて、時刻サーバを無線 LAN アクセスポイントとして動作させ、そこに各スマートウォッチを接続することでマスター・スレーブ型の通信ネットワークを構築する。マスターとなる時刻サーバとスレーブとなる各スマートウォッチは、次節で述べる方法で時刻同期を行う。

**A study of time-synchronized sensing for multiple wearable devices**

Akio SASHIMA<sup>†</sup>

<sup>†</sup>Human Augmentation Research Center, National Institute of Advanced Industrial Science and Technology (AIST)

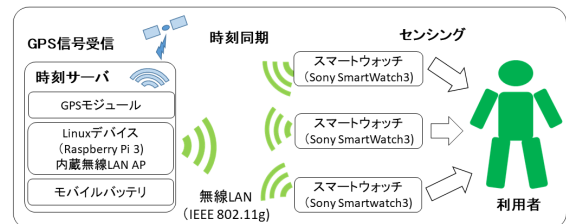


図 1: モバイル時刻同期センシングシステム

## 3 時刻同期

ある時点  $t_0$  におけるマスターの時刻  $T_{M0}$  とスレーブの時刻  $T_{S0}$  とし、両者の時刻差を  $offset$  とすると、 $T_{M0} = T_{S0} + offset$  となる。そして、その後のある時点  $t$  におけるマスターの時刻を  $T_M$ 、スレーブ上の  $T_{S0}$  からの経過時間を  $T_S$  とすると、その関係は  $T_M = skew * T_S + offset$  と表せる。ここで、 $skew$  は、スレーブのマスターに対する単位あたりの時間の進みや遅れの度合いである。各スレーブが、この  $offset$  と  $skew$  を推定することで時刻同期が可能となる。

しかしながら、マスターとスレーブの時刻を直接観測することは困難なため、 $offset$  と  $skew$  の推定は、いくつかの仮定のもとで統計的に行われることが多い。ここでは、時刻同期プロトコル: NTP [3] と同様に通信による方法を用いる。具体的には、送受信時間が等しいという仮定のもと、マスター・スレーブ間で双方向通信を行う。 $i$  回目の双方向通信における、スレーブ側のパケット送信時刻:  $T_{1,i}$ 、マスター側のパケット受信時刻:  $T_{2,i}$ 、マスター側の返信パケット送信時間:  $T_{3,i}$ 、スレーブ側の返信パケット受信時間:  $T_{4,i}$  とすると、(1),(2) 式により、マスター時刻とスレーブ時刻の観測値:  $T_{S,i}$  および  $T_{M,i}$  が得られる。

$$T_{S,i} = (T_1 + T_4)/2 \quad (1)$$

$$T_{M,i} = (T_2 + T_3)/2 \quad (2)$$

この値に基づいて、マスター時刻とスレーブ時刻の差分:  $T_{delta,i}$  を計算し、(4) 式を出力方程式とするカルマンフィルタ [2] を適用することで、 $offset$  と  $skew$

を逐次的に推定する.

$$T_{\text{delta},i} = T_{M,i} - T_{S,i} \quad (3)$$

$$T_{\text{delta},i} = \begin{bmatrix} T_{S,i} & 1 \end{bmatrix} \begin{bmatrix} \text{skew}_{\text{delta}} \\ \text{offset} \end{bmatrix} + w \quad (4)$$

$$\text{skew} = \text{skew}_{\text{delta}} + 1 \quad (5)$$

ここで,  $w$  は観測雑音,  $\text{skew}_{\text{delta}}$  は差分に対する遅れ・進み度合いであり,  $\text{skew}$  は (5) 式により計算する.

#### 4 実験結果と考察

観測値  $T_{S,i}, T_{M,i}$  は, 送信と受信に要する時間が等しいという仮定の下で得られる観測値である. しかしながら, 実際の観測値は, 一方の通信遅延などにより, この前提が満たされない場合, 異常値として観測され, 時刻同期の性能を悪化させる. そこで, 本システムでは, (6) 式により計算される遅延時間:  $T_{\text{delay}}$  の値が一定範囲 ( $5.5\text{msec} < T_{\text{delay}} < 7.5\text{msec}$ ) から外れたとき, 異常値として観測値から除去することとした.

$$T_{\text{delay},i} = ((T_4 - T_1) - (T_3 - T_2))/2 \quad (6)$$

図2として, 遅延時間による除去の有無による予測誤差を比較した箱ひげ図を示す. ここで予測誤差はスレーブが予測した時刻  $\hat{T}_{M,i}$  と観測された時間:  $T_{M,i}$  の差である. 集計は, 同期されていない状態から約10分間, 時刻同期を行い, 同期直後の過大値を除いたデータを用いた. 図2より, 異常値の影響が遅延時間による除去により緩和されていることがわかる.

次に, 遅延時間による除去を行った上で, 加速度センサと回転ベクトルセンサによるセンシングと記録を時刻同期と同時に行い, その影響を比較した. 図3として, センシング無し, 1台だけでセンシング, 複数同時にセンシングした場合の予測誤差を箱ひげ図で示す. 図3より, 台数が増えると同期性能は低下する傾向が見えるが, 加速度センサのセンシングレートを考慮すると, 大きな変化ではなく, おおむね妥当な同期性能が得られている.

#### 5 まとめ

本論文では, 携帯可能な時刻サーバと複数のウェアラブルデバイスを用いた時刻同期センシングのプロトタイプを提案した. 通信遅延に由来する異常値を排除し, カルマンフィルタによる推定を行うことで, おおむね妥当な時刻同期性能が得られることを確認した. 今後は, 異常値の排除を適応的に行いながら時刻同期を行う方法について検討したい. また, 本システムを実際

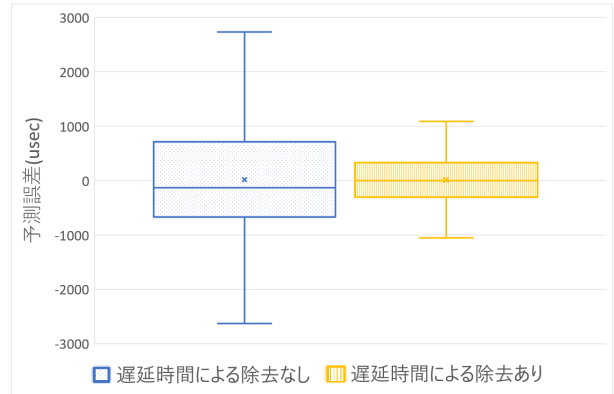


図2: 推定時刻の誤差の比較 (遅延時間による異常値の除去)

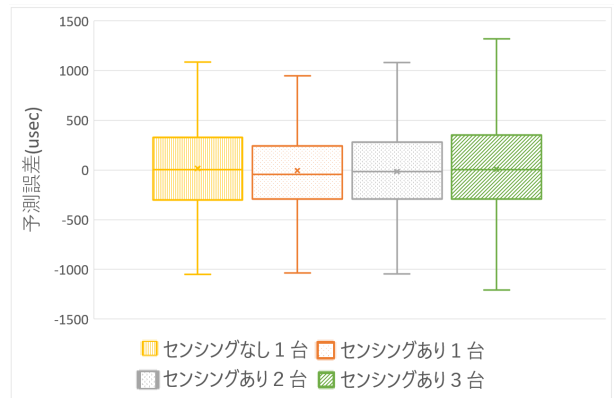


図3: 推定時刻の誤差の比較 (複数台のセンシングの影響)

の行動計測に適用することで, ウェアラブルデバイスによる時刻同期センシングの可能性について検討を進めていきたい.

#### 謝辞

本研究の一部は JSPS 科研費 17K00145, JST, CREST, JPMJCR18A4 の支援を受けて実施されました.

#### 参考文献

- [1] モバイル生体センシングにおける時刻同期機構に関する一検討, 幸島 明男, 信学技報, vol. 118, no. 54, MICT 2018-7, pp. 31-34, 2018.
- [2] 線形カルマンフィルタの基礎, 足立 修一, 計測と制御, 56 巻, 9 号, pp 632-637, 2017.
- [3] Network Time Protocol Version 4: Protocol and Algorithms Specification, <http://www.ietf.org/rfc/rfc5905.txt>