

TLIFES におけるスマートフォンアプリケーションの消費電力低減の検討

坪井 俊也*, 旭 健作, 渡邊 晃(名城大学)

Study of the Reduction of Power Consumption of Smartphone Application in TLIFES

Toshiya Tsuboi, Kensaku Asahi, Akira Watanabe (Meijo University)

1 はじめに

少子高齢化と核家族化の進行により高齢者の徘徊行動や孤独死などが問題視されている。我々は、スマートフォンの通信機能とセンサ機能を活用し、見守る側（家族や地域の人など）と見守られる側（高齢者や子どもなど）で、ユーザの位置情報や行動状態などの情報を共有することにより、住民が安心して生活できるシステムとして統合生活支援システム TLIFES (Total LIFE Support system) [1] を提案している。

TLIFES で使用するスマートフォン側アプリケーションは、その消費電力が稼働時間に大きな影響を与える。これまでの省電力化の取り組みにより消費電力は小さくなりつつあるが、位置情報取得や行動状態の判定は、依然として消費電力が大きく、見守りシステムとして大きな課題となっている。そこで本稿では、消費電力の大きい処理を調査し、その解決方法を検討した。

2 TLIFES の概要と課題

TLIFES では、すべてのユーザがスマートフォンを所持していることを前提とする。スマートフォンの通信機能と、GPS や加速度センサ等のセンサ機能を利用し、ユーザの位置情報や行動状態を判定し、共有できるシステムを実現している。

TLIFES では、スマートフォン側アプリケーションの省電力のために、これまでに以下の取り組み [1] がなされてきた。GPS は、屋内のような GPS 衛星の電波が届かない場所においても、必要以上に位置情報を取得しようとして電力を消費する可能性がある。また、衛星を 4 機以上捕捉しないと正確な位置情報を取得できない。捕捉衛星数は比較的早く検出できるため、測位開始 10 秒後に衛星数を 4 機以上補足できなかった場合、位置情報が取得できないと判定し、GPS 測位を中断するようにした。また、自宅や職場などで長時間移動しない場合は、GPS を起動する必要がない。そのため、ユーザの行動状態を判定し、移動中と判定したときのみ GPS を起動させることとした。

以上の取り組みにより、GPS による電力消費は大きく削減された。しかし、地下鉄乗車時や、地下街歩行中など、GPS 電波が届かないところでは、依然として 10 秒間の無駄な GPS 起動時間がある。また、行動状態の判定をするために、常に加速度センサを利用しており、CPU の動作に対する消費電力の割合が依然として大きい点が課題となっており、これらの電力を減らす必要がある。

3 電力消費低減の検討

CPU の消費電力の要因を調査するため、Android アプリ解析ツールの LAP (Log analyzer for Android Platform) を用いて、TLIFES アプリケーションの JAVA メソッド毎の CPU 処理時間を測定した。測定方法は、端末に Samsung Galaxy Nexus (AndroidVer4.2.2) を用いて、室内で 5 分間、歩行した。TLIFES の動作としては、歩行していることから移動していると判断

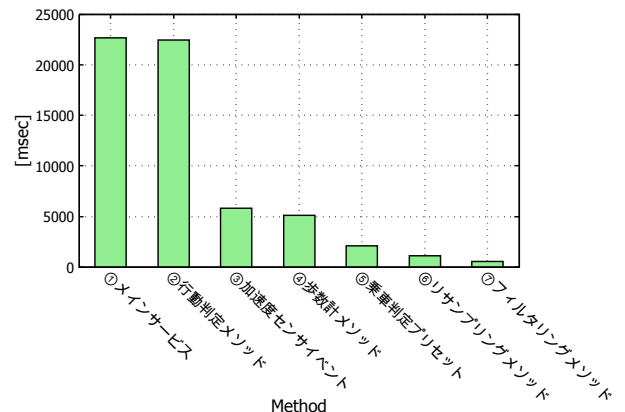


Fig. 1 CPU processing time for each method

し、2分に1回GPS測位を開始する。しかし、室内であるため位置情報を取得できず、10秒後にGPS測位を中断する。以上の動作時に取得した、メソッド毎のCPU処理時間の上位7項目をグラフにしたものをFig. 1に示す。

Fig. 1の中で、処理①②は、GPSの位置測位の終了待ちによる時間がほとんど占めている。処理③④は、20ms間隔で加速度センサの値を取得し、歩数計の計算を行うメソッドである。このことから、GPSの10秒間の測位にかかる処理が非常に大きく、次に20ms間隔の定期処理が、消費電力の大きな要因となっていることがわかる。

そこで、GPSの補足衛星数だけでなく、衛星の受信強度(信号雑音比:SNR)も用いることで、GPSの位置取得可否を短時間に判定でき、消費電力を低減できると考えられる。また、Androidにおける消費電力の既存研究[2]によると、2000×2000の行列の乗算時に、JNIを用いたC言語による実装の消費電力は、JAVA言語による実装の消費電力の69.7%になるという報告がある。処理③④は信号処理が主であり、この部分をC言語による実装とすれば、歩数計部分の消費電力を最大で70%にできると考えられる。

以上により、TLIFESの消費電力の多くを占めるGPSとCPUの消費電力を削減できると考えられる。

4 まとめ

本稿では、TLIFESにおける消費電力削減の手法について検討した。今後は提案内容を実装に反映し、評価を行っていく。

文 献

- [1] 加藤大智, 他: TLIFES における省電力化を目的とした位置測位手法の提案と実装, 研究報告コンシューマ・デバイス&システム (CDS), Vol.2013-CDS-6, No.13, pp.1-6, 2013.
- [2] Ramirez, R.I., et al.: Differences of energetic consumption between Java and JNI Android apps, Integrated Circuits (ISIC), 2014 14th International Symposium on, pp.348-351, 2014

TLIFESにおける スマートフォンアプリケーションの 消費電力低減の検討

名城大学 理工学部

坪井 俊也 旭 健作 渡邊 晃

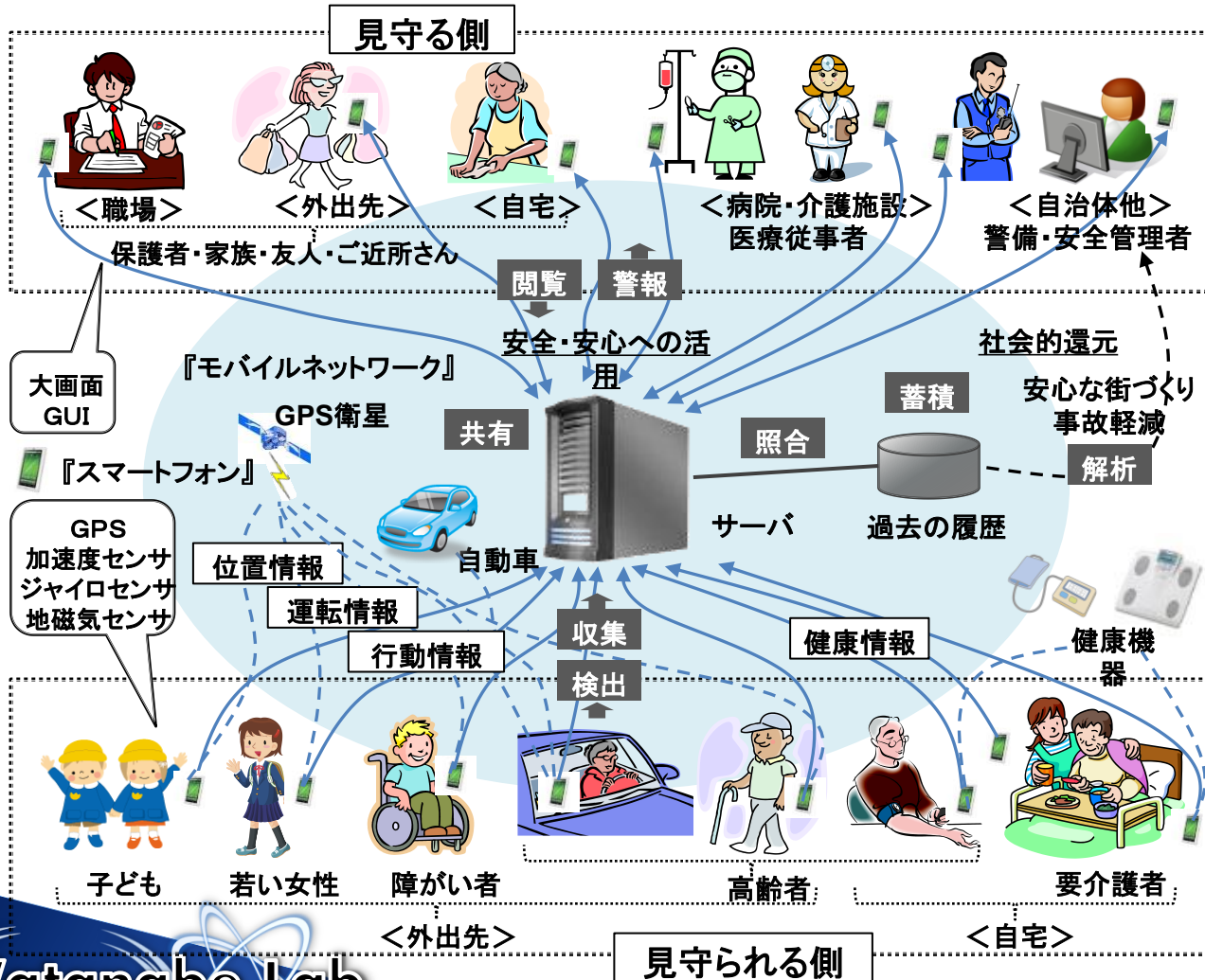
研究背景

- ▶ 高齢化社会，核家族化の進行
 - 一人暮らしの高齢者の増加
 - 高齢者の徘徊行動
 - 認知症の行方不明者は年間1万人超
- ▶ スマートフォンの普及
 - GPSやセンサ，無線通信など多くの機能を搭載



スマートフォンを用いた見守りシステム
TLIFES(Total LIFE Support system)を提案

TLIFESの概要

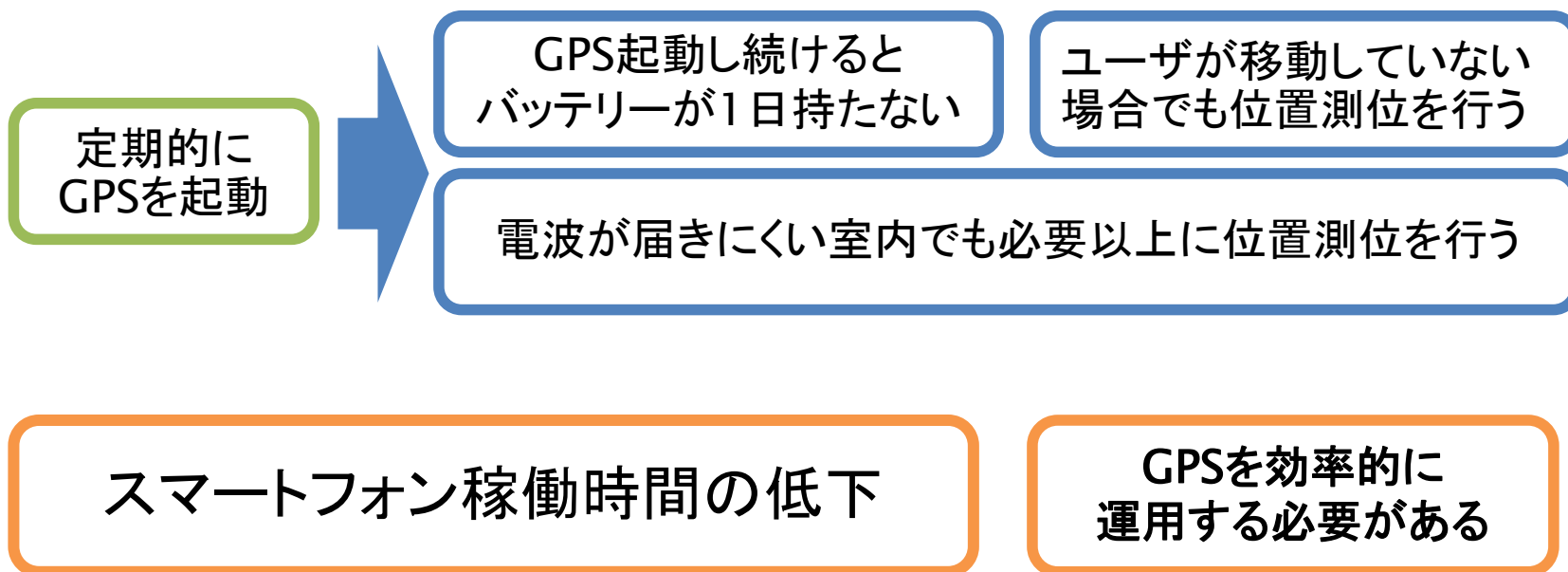


徘徊を検出
 アラーム
 メール送信

✓ 位置情報
 ✓ 行動情報

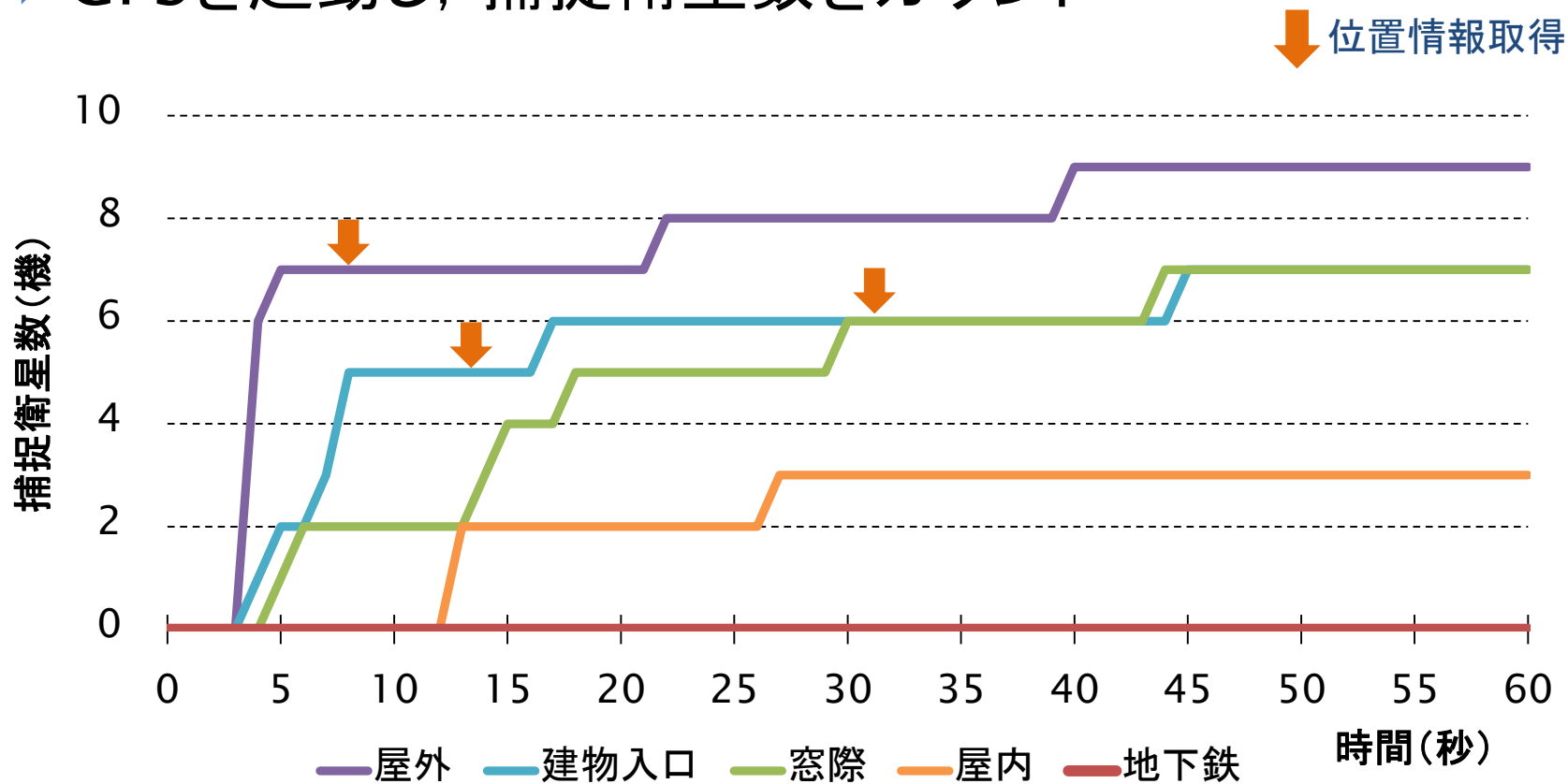
初期からのTLIFESの課題

- ▶ スマートフォンアプリケーションの消費電力量
 - GPSはセンサの中でも消費電力が大きい
 - TLIFESでは、定期的にGPSを起動



GPSの時間経過による特性

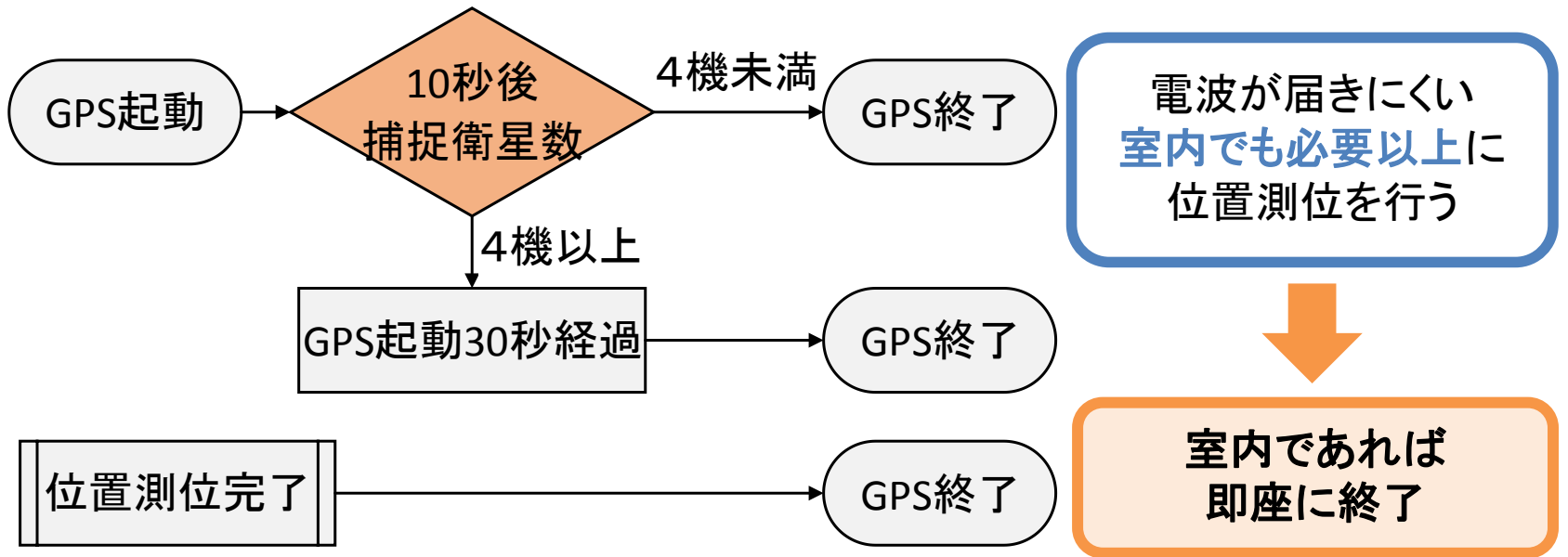
▶ GPSを起動し、捕捉衛星数をカウント



TLIFESにおける消費電力削減—GPS制御1

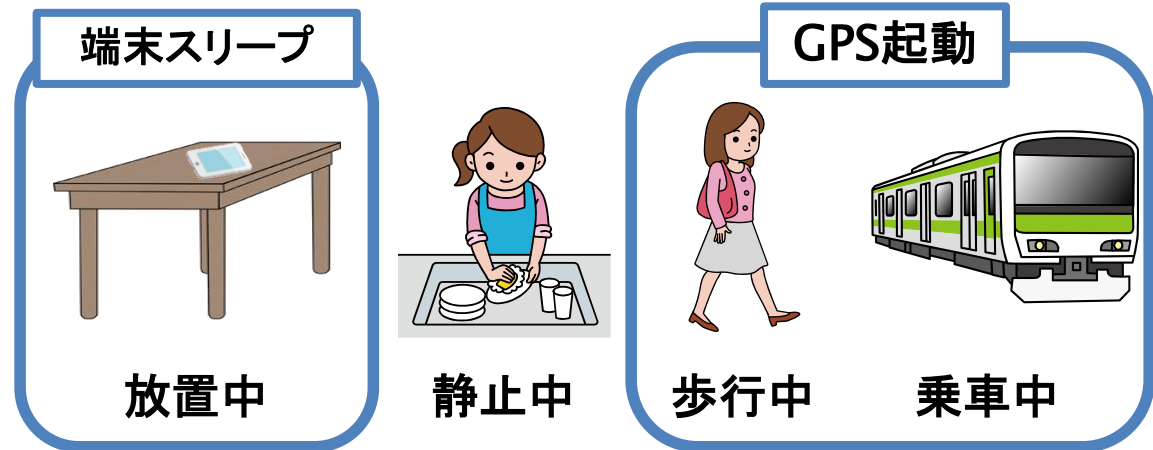
▶ 捕捉衛星数を用いたGPS制御

- GPSは衛星数が4機以上で、正確に位置計算可能
- 起動10秒後、4機未満なら**室内**と判定し、測位停止



TLIFESにおける消費電力削減—GPS制御2

- ▶ 利用者の**行動判定**を行いGPS制御
 - 低消費電力の加速度センサのみ使用
 - 2分に1回判定
- ▶ 判定する行動
 - 放置中
 - 静止中
 - 歩行中
 - 乗り物乗車中



2分に1回GPS起動 → 2分に1回の移動時のみGPS起動
GPSによる消費電力削減

本研究の目的

- ▶ 高消費電力のGPSの処理時間をさらに低減
 - 現状, 地下街で歩行すると最低10秒間は起動



- ▶ GPS位置取得可否を短時間に判定
 - GPS衛星の受信強度（信号雑音比:SNR）も利用
- ▶ 判定方式決定のため, GPSデータの計測を行う

GPS取得可否判定—データ収集

- ▶ GPSデータ計測方法
 - GPSを30秒間起動
 - 計測する情報
 - 位置測位完了したか
 - 位置測位完了に要した時間
 - 1秒毎に計測する情報
 - 捕捉衛星数
 - 捕捉衛星の受信強度
 - Androidアプリケーションを作成
 - 計3000回以上の計測データ

GPS取得可否判定一方針

▶ 位置測位可能のときを重視

- 再現率 (Recall) 80%以上

▶ 現在の判定方法と比較

- 平均GPS起動時間を短縮

		実際の結果	
		取得可	取得不可
予測結果	取得可	TP	FP
	取得不可	FN	TN

$$\text{適合率 (Precision)} = \frac{TP}{TP + FP}$$

取得できると予測されたもののうち、
実際に取得できたものの割合
例: 取得可能と予測した中の精度は高い
そのかわり、取得できたものを見逃すことは多い

$$\text{再現率 (Recall)} = \frac{TP}{TP + FN}$$

実際に取得できたもののうち、
取得できると予測されたものの割合
例: 実際に取得できるのは網羅している
そのかわり、取得できないものを誤警報することは多い

GPS取得可否判定—方式決定(1)

- ▶ 開始5秒後に着目。
 - 取得可能と判定する条件は,
“5秒目の**捕捉衛星数** > 閾値 χ [機]”

閾値 χ [機]	0	1	2
適合率[%]	85.7%	94.5%	98.4%
再現率[%]	84.8%	72.3%	63.5%

捕捉衛星数では、細かな閾値調整ができないため、
5秒目の判定に適さない

GPS取得可否判定—方式決定(2)

- ▶ 開始5秒後に着目。
 - 取得可能と判定する条件は,
“5秒目の捕捉衛星の受信強度の最大値 > 閾値 χ [dB]”

閾値 χ [dB]	19.0	19.1	19.2	19.3	19.4
適合率[%]	89.53%	89.75%	89.76%	89.84%	89.95%
再現率[%]	80.79%	80.53%	80.17%	79.92%	79.51%

再現率が80%以上, 適合率が最大である
“5秒目の捕捉衛星の受信強度の最大値 > 19.2[dB]”に決定

GPS取得可否判定—既存との比較

- ▶ 受信強度を利用して判定
 - 再現率80%以上となり、見守りに必要な位置情報の取りこぼしがより少なくなる
 - 平均GPS起動時間が85.6%になり、GPSにおける消費電力が約14.4%低減可能

	既存方式	新方式	倍率
判定する時間	開始10秒後	開始5秒後	
判定項目	捕捉衛星数	捕捉衛星の受信強度	
適合率(Precision)	96.2%	89.8%	0.93
再現率(Recall)	75.8%	80.2%	1.06
1測定あたりの平均GPS起動時間	9,798ミリ秒	8,387ミリ秒	0.856

まとめ

- ▶ 本研究の目的
 - 高消費電力のGPSの処理時間をさらに低減
- ▶ 消費電力低減の検討
 - GPS位置取得可否を衛星の受信強度を用い短時間に判定
 - GPSにおける消費電力が約14.4%低減可能
- ▶ 今後の方針
 - 更なる消費電力低減
 - 検討内容の実装と評価

補足資料

- ▶ 以下, 補足資料

TLIFESの概要

- ▶ スマートフォンの通信機能とセンサ機能を活用し、ユーザが情報を共有できるシステム
- ▶ ユーザ全員がスマートフォンを所持
- ▶ ユーザの行動情報、位置情報、歩数などを収集
- ▶ 情報は定期的にサーバへ送信しデータを蓄積
- ▶ 許可されたメンバはデータの閲覧可能
- ▶ 過去の履歴と比較し、ユーザの異常があると判断したらアラームを送信

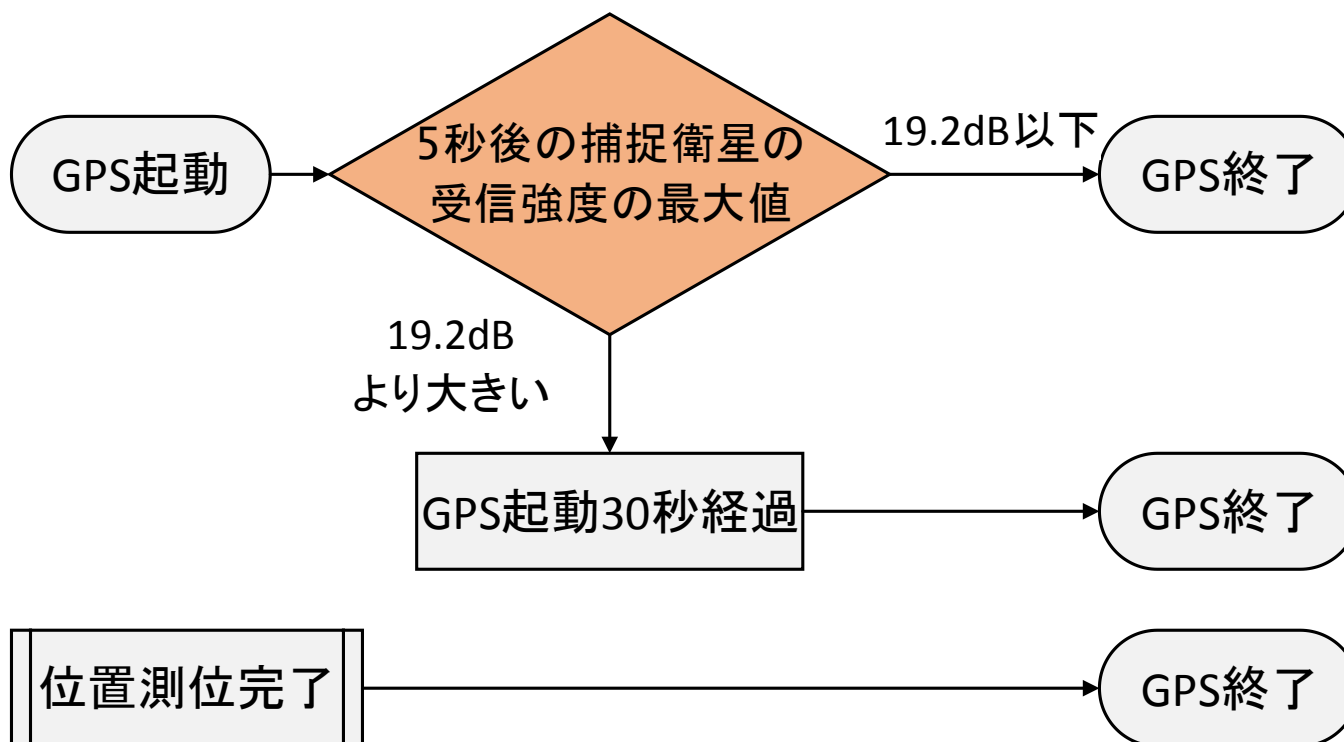
GPS終了時間30秒の検討

- ▶ 現在のGPS制御において、取得可能と判定されて、位置測位完了するまで、GPS開始後30秒間起動する
 - この30秒間をTLIFESに適切な時間に変更する
 - 30秒間を20秒間にした場合の平均GPS起動時間を試算
 - 既存方式(30秒間)→新方式(20秒間)で、約26%削減

	既存方式	新方式
1測定あたりの平均GPS起動時間(30秒間)	9,798ミリ秒	8,387ミリ秒
1測定あたりの平均GPS起動時間(20秒間)	9,567ミリ秒	7,255ミリ秒
倍率	0.976	0.865

GPS取得可否判定—フローチャート

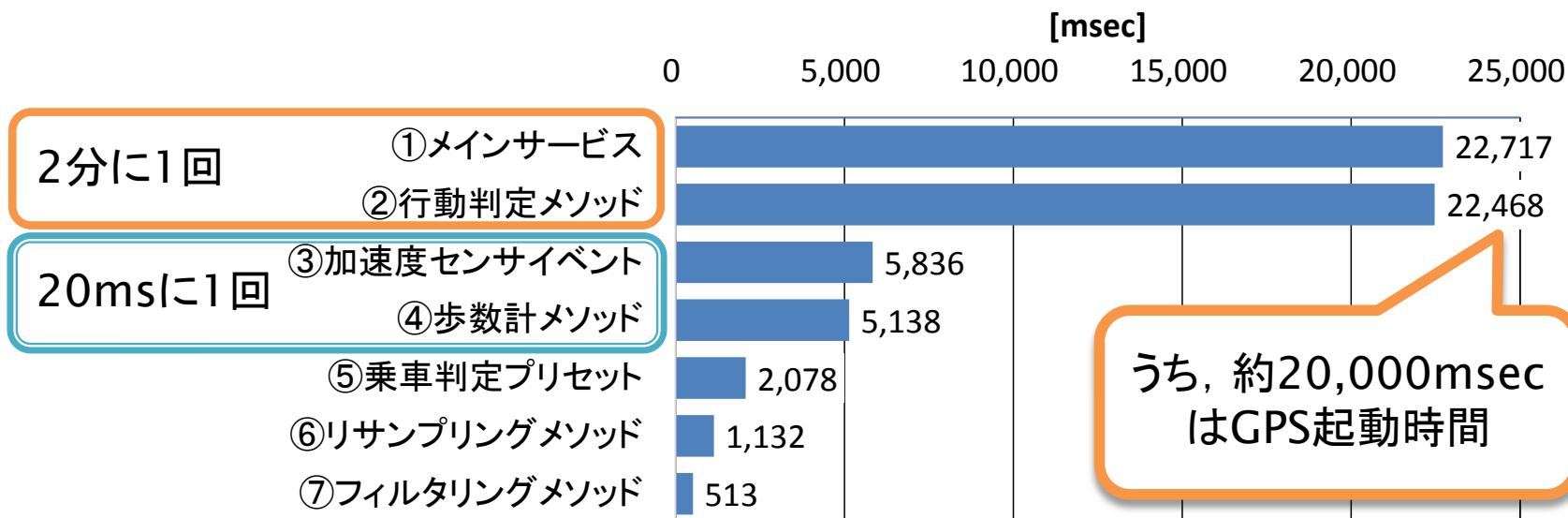
▶ 取得可否判定のフローチャート



消費電力低減の検討

- ▶ アプリケーションのJAVAメソッド毎のCPU処理時間を測定
 - Androidアプリ解析ツール「LAP^[1]」を使用
 - 5分間測定. 室内で歩行, 行動判定は2回「歩行中」

移動時のメソッド毎のCPU処理時間



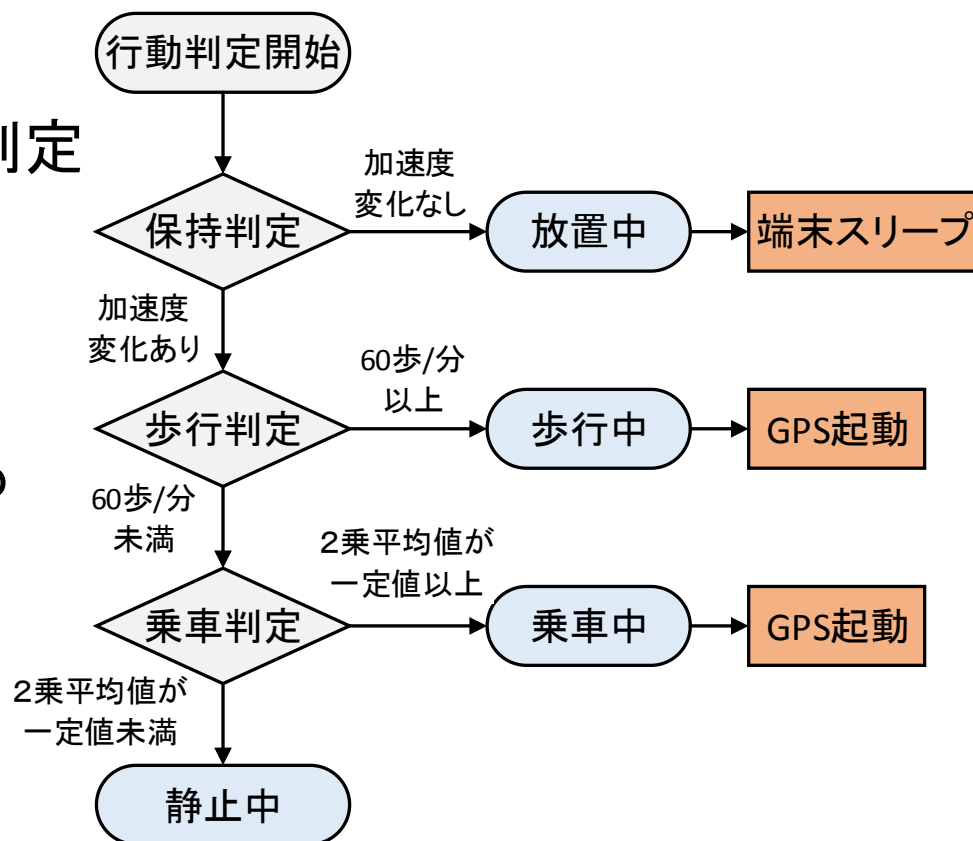
GPS取得可否判定—既存との比較

- ▶ 再現率を既存と同等まで低下させる
 - 平均GPS起動時間が80.1%になり、GPSにおける消費電力が**約20%低減可能**

	既存方式	新方式	倍率
判定する時間	開始10秒後	開始5秒後	
判定項目	捕捉衛星数	捕捉衛星の受信強度	
適合率(Precision)	96.2%	91.4%	0.95
再現率(Recall)	75.8%	76.0%	1.00
1測定あたりの平均GPS起動時間	9,798ミリ秒	7,849ミリ秒	0.801

行動判定フロー

- ▶ 2分毎に判定開始
- ▶ 加速度値を用いる
 - スマートフォンの保持判定
 - 歩数による歩行判定
 - 加速度値の処理による乗り物乗車判定

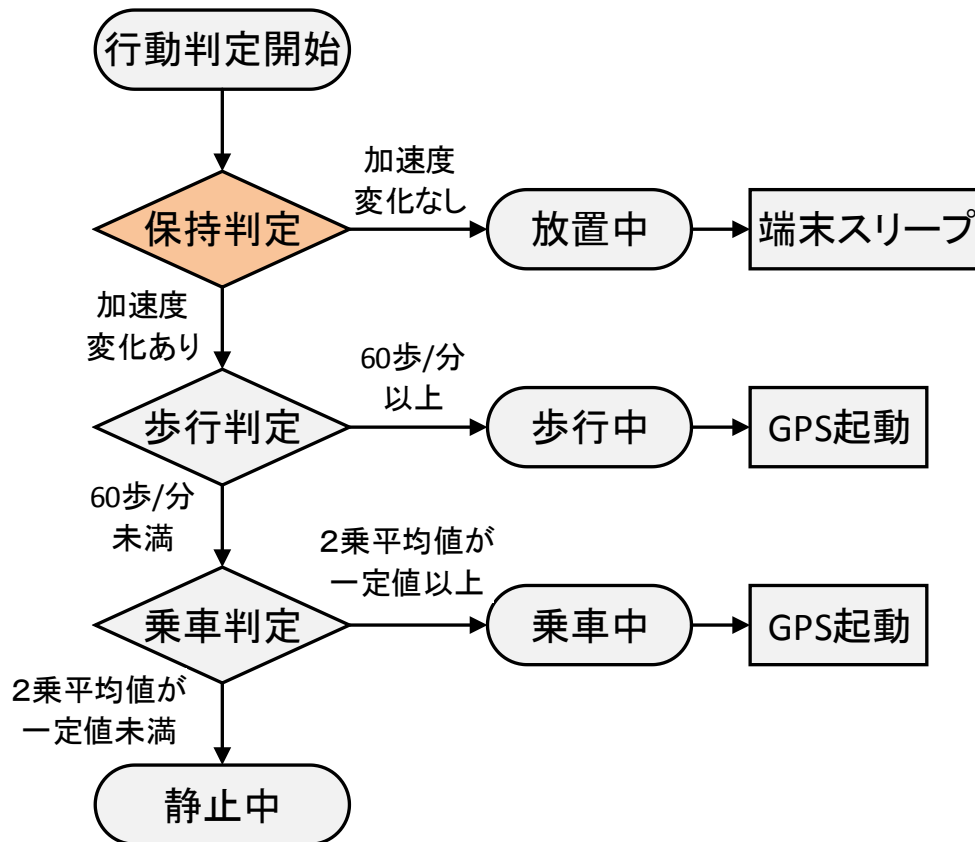


行動判定フロー(保持判定)

▶ スマートフォンの保持判定

- 2分間スマートフォンの加速度に変化なし
⇒「放置中」

- ▶ 加速度に変化あり
⇒ 歩行判定を行う

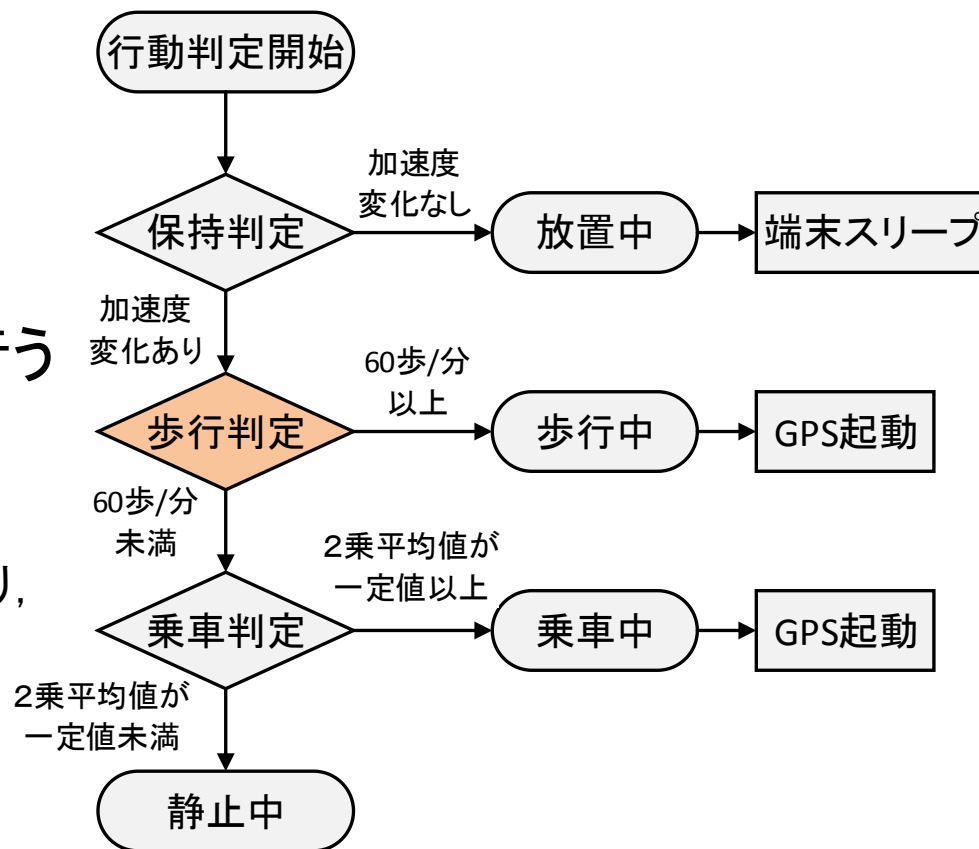


行動判定フロー(歩行判定)

▶ 歩数による歩行判定

- 毎分60歩以上
⇒「歩行中」
⇒GPS起動
- 毎分60歩未満
⇒ 乗り物乗車判定を行う

※閾値毎分60歩について
人の平均歩数が毎分120歩であり、
判定する時間の半分以上歩行で
歩行中と判定するため



行動判定フロー(乗車判定)

▶ 乗り物乗車判定

- 加速度データを利用
- 乗車時の特徴的な高周波ノイズを利用
- 加速度の2乗平均値
 - 一定値以上
⇒「乗車中」
⇒GPSを起動
 - 一定値未満
⇒「静止中」

