



アンドロイドデバイスでのGNSS生測定値の利用



WHITEPAPER



マスマーケットでのロケーション性能向上に向けて



著者について。GSA GNSS生測定値タスクフォース

2017年6月に立ち上げられ、欧州GNSS機関（GSA）によって調整されたGNSS生測定値タスクフォース（タスクフォース）は、Android生測定値に関する知識と専門知識の共有と、大量市場でのアプリケーションに関連する高精度測位技術の可能性を含むその利用を目的としています。タスクフォースには、GNSSの専門家、科学者、GNSS市場の関係者が含まれており、全員がこれらの生測定値のより広範な利用を促進することに専念しています。

このホワイトペーパーは、以下のタスクフォースのメンバーが共同で作成したものです。

- 欧州GNSS局 (GSA)
- 欧州宇宙機関
- ノッティンガム大学ノッティンガム地理空間研究所
- エアバスディフェンス&スペース社

以下のタスクフォースメンバーからは、試験結果、修正、コメントが寄せられました。

- 欧州委員会共同研究センター (JRC)
- 欧州衛星サービスプロバイダー (ESSP)
- フランス民間航空大学 (ENAC)
- GNSSセンターオブエクセレンス (GCE)
- ろくぶん
- リエカ大学
- ザグレブ大学
- フランス交通・開発・ネットワーク科学技術研究所(IFSTTAR)
- スウェーデン研究所 (RISE)
- ウェイサー

著者は、すべてのタスクフォースメンバーに感謝の意を表したいと思います。

<https://www.gsa.europa.eu/gnss-raw-measurements-task-force>.

また、彼らが提供した貴重なコメントのためにGoogleにも特別な感謝の意を表します。

欧州連合に関する詳細な情報は、インターネット (<http://europa.eu>) で入手可能です。ルクセンブルク欧州連合出版局、2017年
ISBN 978-92-9206-033-6
doi.10.2878/449581

Copyright © European GNSS Agency, 2017

本文書および本文書に含まれる情報は、チェコ共和国およびその他の国の法律の下で適用される著作権およびその他の知的財産権の対象となります。第三者は、本文書の一部または全部を改変しないことを条件に、本文書をダウンロード、コピー、印刷し、他の第三者に提供することができます。本文書に記載されている情報は、以下のように出典および著作権者を明記することを条件に、抜粋、複写、印刷、第三者への提供を行うことができます。"ソースWhite Paper on using GNSS Raw Measurements on Android devices, copyright © European GNSS Agency, 2017".

本文書に含まれる情報の一部を含む本文書のいかなる部分も、デジタル形式であるか否かにかかわらず、<http://www.gsa.europa.eu/contact-us> の下で、要求された要素（文書および/または情報）と使用条件を明記した書面による欧州GNSS機関の事前の明示的な許可なしに、変更、編集、変更を行うことはできません。上記の利用規約に違反していることに気づいた場合は、上記の連絡先サイトを通じて、直ちに欧州GNSS機関に通知してください。これらの利用規約に違反した場合は、金銭的損害賠償および/または文書および/またはそこに含まれる情報の違法な使用を止めるための差止命令を求める法的手続きの対象となる可能性があります。

ユーザーは、この文書またはその一部をダウンロード、転送、コピーすることにより、デジタル形式であるかどうかにかかわらず、上記の利用規約を承認し、承諾したものとします。

Android ロボットは、Google が作成・共有した著作物を複製・改変したものであり、その内容は、以下に記載されている条件に従って使用されます。クリエイティブコモンズ3.0 帰属ライセンス。

目次

表目次	3
エグゼクティブサマリー	5
ドキュメント構造	6
1 必要なGNSSの基礎知識 フォー理解すること 生測定値	7
1.1 序章	7
1.2 全地球航法衛星システム	7
1.3 GNSS受信機アーキテクチャ	8
1.4 時間	8
1.4.1 序章	8
1.4.2 参考時間	9
1.5 ナビゲーションメッセージとトラッキングステータス	10
1.6 擬似レンジ生成	13
1.7 位置推定	14
1.7.1 シングルGNSS星座	14
1.7.2 マルチGNSS星座	15
2 ハウツー アクセスGNSS 生 Androidを使用した測定値 エーピーアイエス	16
2.1 序章	16
2.2 Android 7以前の位置情報API <i>Android 7以前の位置情報API</i>	16
2.3 Android 7の位置情報API	17
2.4 GNSSの生の測定値を使用する	18
2.4.1 GPS時間生成	20
2.4.2 擬似レンジ生成	20
2.4.3 搬送波位相測定	22
2.4.4 ドップラー	23
2.4.5 衛星ID	24
2.4.6 コンステレーション	24
2.5 生データアーキテクチャ	24
2.6 既存のアプリケーションとデバイス	26
3 GNSS生測定値の可能性と実用性	27
3.1 モバイルA-GNSSチップセットの概要	28
3.2 ベースライン パフォーマンス - コード ポジショニング	29
3.3 ポジションの改善	30
3.3.1 多星座	30
3.3.2 チップセット内部の情報を利用する - コード観測値のドップラー平滑化	31
3.4 電話の向こう側に持っていく - 差分観察	33
3.4.1 デューティサイクル	34
3.4.2 センサーフュージョン	37
3.5 教育と科学の応用	37
3.6 ハイインテグリティソリューション	38

4 ザ ニード AND USE ケース フォー より高い精度 イン ザ マス 市場	39
4.1 位置精度向上の恩恵を受ける主な用途	39
4.1.1 モバイルアプリケーション	40
4.1.2 安全性関連のアプリケーション	40
4.1.3 セミプロ向けアプリケーション	41
リスト オブ 図	42
リスト オブ タブレス	43
略語	44
参考文献 資料	45

仮
目
録

エグゼクティブ・サマリー

2016年5月、GoogleはAndroid 7からGNSSの生測定値が利用可能になることを発表しました。今回初めて、開発者はマスマーケットデバイスからキャリアとコードの測定値とデコードされたナビゲーションメッセージにアクセスすることができるようになりました。

スマートフォンでGNSSの生測定値を利用することには、いくつかのメリットがあります。これまではプロのGNSS受信機に限定されていた高度なGNSS処理技術への扉を開くことができるので、GNSSの性能向上につながる可能性があります。これらの利点は、コードベースポジショニング、コード支援ポジショニング、ディファレンシャルポジショニング、精密ポイントポジショニングによって実証されています。通常の状態では、GNSS生データから計算された位置は、一般的なチップセットの出力ほど最適ではないかもしれませんが、外部補正を適用する場合には、GNSS生データを使用することで、ソリューションの精度を向上させることができます。拡張現実感、ロケーションベースの広告、モバイルヘルス、資産管理など、この精度向上から利益を得ることができるアプリケーション分野がいくつかあります。生測定値により、マルチGNSSソリューションを最適化したり、性能や差別化要因に基づいて衛星を選択したりすることができます。これは、2016年12月以来、運用されており、まだ完全な展開を完了していますが、すでに今日の優れた性能と信号の利点を提供していますが、そのようなオープン信号E1と正確なポイントポジショニング上の認証など、すぐに差別化を提供しますガリレオのために特に関連しています。例えば、この論文を作成するために行われたテストでは、ガリレオのみの位置決めを得ることができました。

生測定値が利用可能なことは、技術革新の観点からも興深い。これにより、より詳細なテストや後処理が可能になり、教育用のツールとしても役立ちます。また、GNSSの生測定値は、追加の情報源を提供し、受信機の自律的な整合性モニタリングアルゴリズムを利用できるようにすることで、スマートフォンの内部整合性をサポートすることもできます。

これらの利点と機会にもかかわらず、GNSSの生測定値を使用することは、最初は思ったほど簡単ではありませんでした。Googleの発表から1年後、AndroidのGNSS生測定値を活用していたのは一部のスマホアプリだけでした。主に2つの理由で、この限られた取り込みを説明することができます。まず、GNSSの専門家は、Android上での生測定値の仕様を理解するための支援を必要とする場合があります。例えば、RINEXやNMEAなどの標準フォーマットは、Androidプラットフォームでは利用できません。第二に、Android環境に非常に精通しているJava開発者は、通常、GNSS測位の詳細を理解していません。

この知識のギャップに対処するために、ホワイトペーパーでは、GNSSの生測定値とAndroidプラットフォーム上での実装との間のリンクを作成しています¹。GNSS基準時刻の生成に関する有用な情報と、Androidからの疑似距離の導出のためのステップバイステップのアプローチを紹介しています。また、いくつかのテスト活動の結果をハイライトしており、GNSSの生測定値、特にガリレオに基づいた作業ソリューションを紹介しています。

この白書は、いくつかの大学、公的機関、GNSS会社の協力を得て作成されたものです。このグループは、欧州GNSS庁（GSA）によって設立・調整され、GNSS生計測タスクフォースとして知られています。

¹ このドキュメント内の生測定値の基本的な概念は、Frank van Diggelen ION GNSS+ 2016のAndroid 7 GNSS生測定値チュートリアルに触発されています。

文書の構造

本文書は以下のような構成になっています。

- 第1章では、GNSSの基本的な概要、すなわち受信機アーキテクチャ、GNSS時刻参照、疑似距離、ナビゲーションメッセージ、位置推定について説明しています。Androidを使ってGNSS生測定値を再構築するために必要な理論的な基礎を中心に紹介します。
- 第2章では、生測定値にアクセスする方法と、それを使用するための最初のステップ（疑似レンジやドップラーの生成など）を説明します。
- 生測定値は、RTKやPPPを含むAndroidデバイスに新たな精度技術のスペクトルをもたらします。第3章では、最も有望な技術を描き、さまざまなソースから収集した結果をハイライトしています。さらに、それぞれのテクニックのメリットや限界についても詳しく解説しています。
- 最後の章では、GNSSの生測定値を使用することで得られる精度と完全性の向上から恩恵を受ける可能性のあるユースケースを紹介します。



1 生測定値を理解するために必要なGNSSの基礎知識

1.1 序章

生測定値を理解し、利用する上で欠かせないGNSSの基本的な考え方を紹介します。これは主に既存の学術文献 [RD-03、RD-04、RD-05]の合成です。GNSSの経験が豊富な方は、直接第2章に進むことをお勧めします。

1.2 全地球航法衛星システム

GNSS宇宙セグメントは、MEO軌道で地球を周回する様々な衛星のコンステレーションで構成されており、高度は約20,000km、伝送遅延は約65msです。現在、4つのGNSS星座が運用中または展開段階にあります。GPS（アメリカ）、GLONASS（ロシア）、BeiDou（中国）、Galileo（ヨーロッパ） [RD-02]。これらは、いくつかの地域のGNSSと増強システムによって補完されています。

ユーザの位置は、ユーザの受信アンテナと少なくとも4つの衛星の位置との間の距離測定値（擬似レンジ）を用いて推定される。どちらも受信機で決定され、衛星の信号と航法メッセージをそれぞれ評価します。この情報は、地球上のどこにいてもユーザーの位置と時間を提供するPVTソリューションで必要とされています。

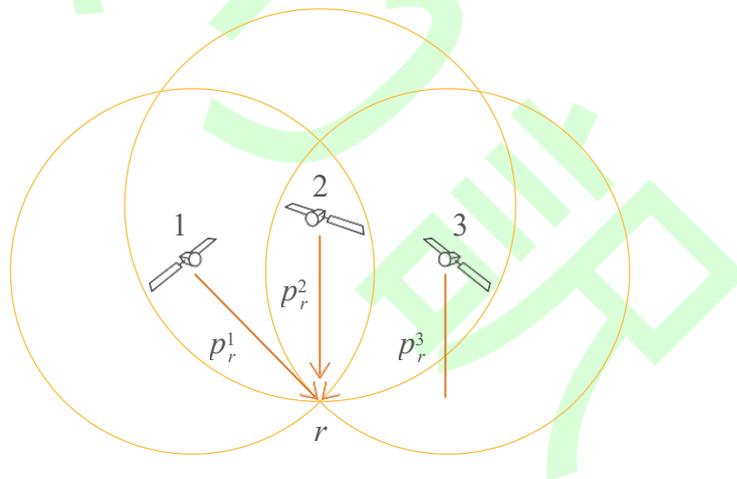


図1：交差する球体 [RD-03]

幾何学的な範囲を決定するために、すべてのGNSS信号は、衛星を一意に識別するために、特定の数ミリ秒の長さの擬似ランダムノイズ（PRN）コードで変調されています。受信機は、受信した衛星信号とPRNコードのローカルコピーを継続的に比較し、整列させます。受信したPRNコードの遅延時間の測定値は、送信機と受信機のクロックが完全に同期している場合、送信時間に等しくなります。PRNコードには、送信衛星の位置を含むナビゲーションデータビットが重畳されている。衛星の位置を1つ知っている、受信機の位置は、範囲と等しい半径を持つ衛星の周りの球体上のどこかにあります。3つの衛星の測距が可能であれば、3つの球体は2点で交わる。1点は地表に近い位置にはないので、2点目は受信機の真の位置になります。つまり、3機の衛星までの範囲を同時に計測することで、図1に示すように、3次元の固定位置を決定することが可能となる。受信時計の影響で、3つの球体が共通点で交わることはありません。これが測位問題の第4の未知数と推定されているため、少なくとも4機の可視GNSS衛星が必要となる [RD-03、RD-04、RD-05]。

1.3 GNSS受信機アーキテクチャ

GNSS受信機は信号を処理し、ユーザーに推定PVTソリューションを提供します。このPVTは、測定された擬似レンジ、ナビゲーションメッセージを通じて配信される情報、およびサードパーティによって提供されるオプションのアシスタンスとオーグメンテーションに基づいています。図2にGNSS受信機の一般的なブロック図を示す。

RFブロック（図の左側）には、アナログ信号のプロセッシングに必要なアンテナとフロントエンドが含まれています。また、ローノイズアンプ、フィルター、中間周波数ダウンコンバージョンを含んでもよい。ブロックの最後の要素は、ADC（Analogue-to-Digital Converter）です。

スマートフォンでは、ベースバンド処理ブロックとPVT処理ブロック（図の右側）は、汎用的なハードウェア上で動作するように設計されたソフトウェアベースの信号処理ユニットである。ベースバンド処理は、GNSS信号の取得と追跡、ナビゲーションメッセージのデコードを担当しています。アシストデータ（外部情報）を提供することで、固定時間を短縮することができます。

ベースバンド処理ブロックは、生データをPVTブロックに提供し、PVTブロックは受信機のPVTを計算します。このプロセスは、データ（EGNOSなど）や加速度センサ（センサ・フュージョン）を増強することで、過酷な環境での精度と可用性を向上させるというメリットがあります。

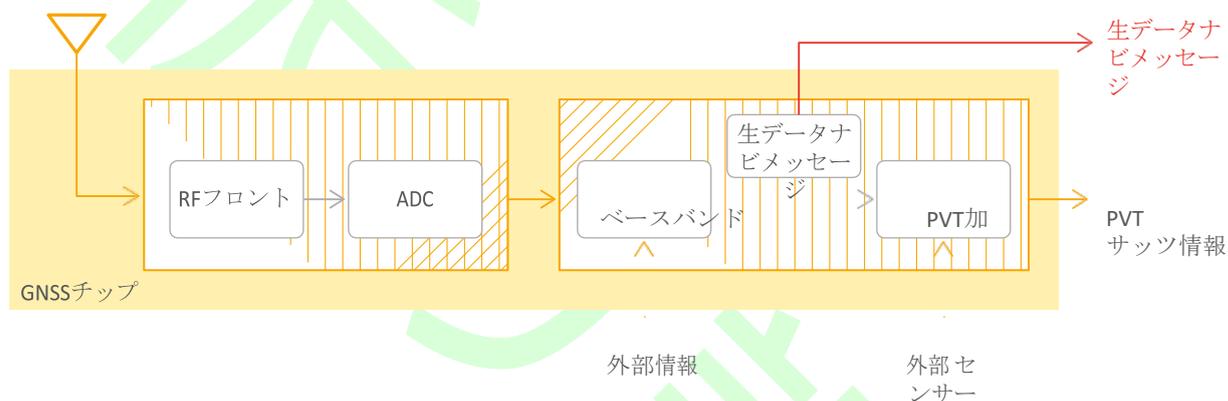


図2：GNSS受信機の一般的なブロック図

Android 7のユーザーは、`android.location`アプリケーション・プログラミング・インターフェース（API）を使用して、疑似距離の計算やナビゲーション・メッセージのデコードに必要な生データにアクセスすることができます（図2の赤い矢印で示されています）。このデータは、マスマーケットデバイスのための新しいアルゴリズムやアプリケーションを探索するために使用することができます。

スマートフォンのGNSS/ナビゲーションチップはブラックボックスの役割を果たし、追尾衛星からのPVTと限定された情報のみを出力します。高度に統合されたチップセットの中には、セルラー、WiFi、Bluetoothとの緊密な統合を利用しているものもあります。モーション/オリエンテーションセンサーをクロスアイドに使用しているものもあり、最終位置の精度と可用性を向上させることができます。ほとんどの場合、チップセットPVTを使用して独自のアルゴリズムを開始し、それに対して定期的にソリューションをテストすることをお勧めします。

1.4 時間

1.4.1 序章

GNSSはその測位能力で最もよく知られていますが、正確な時間を知る上で最も利用可能で信頼性の高い情報源の一つでもあります。

時間は、まとまりのある均一な方法で定義されなければなりません。そのため、異なるGNSS星座で使用される基準時刻間の変換が必要となります。スマートフォンでこのことを考慮しないと、マルチコンステレーションフィックスのアクセシビリティの低下につながります。GNSS星座の時刻基準とは別に、次の段落でさらに説明するように、普遍的な基準時刻系も定義されています。

1.4.2 参考時間

Temps Atomique International (TAI) の第二持続時間は、1967年に定義された。全 $9192\ 631\ 770$ セシウム133原子の基底状態の2つの超微細準位間の遷移に対応する放射線の周期

普遍的な時間 (UT) は、地球の自転の平均速度を反映した太陽時間の基準であり、その最も一般的に使用される定義はUT1として知られています。協定世界時(UTC)はTAIとUT1の妥協点です。実際には、原子時間として、UTCはTAIスケールと同じように一様である。ただし、地球の自転変動に追従するため、UT1に対して常に0.9秒以内に収まるようにしています。これは、TAIに所定のうるう秒数を加算または減算することによって達成される。

GNSS測位の主なコンセプトは、タイミング-信号の到着時刻と送信時刻-に基づいています。各GNSSは、表1に示すように、独自の基準時間を使用しています。

表1.GNSS基準時間

GNSSシステム	GNSS基準時間
GPS	GPS時刻(GPST)
ガリレオ	ガリレオシステム時
GLONASS	グローナスタイム (GLONASST)
北斗	北斗時間(BDT)

1.4.2.1 GPS時刻

GPSタイム (GPST) はうるう秒がなく連続しています。それは1980年1月5日と6日の間の真夜中 (0時間UTC) に開始されます。この時代のTAIとUTCの差は19秒であった。

GPSTは2つのパラメータを用いて送信されます。ウィークナンバー (WN) とタイムオブウィーク (TOW)。最初のカウントは、GPS時刻の開始から現在の週に発生した週です。週間は土曜日の午前0時からです。TOWは、現在の1週間に発生した秒数をカウントします。

1.4.2.2 ガリレオシステムの時間

ガリレオシステムの時間は、1999年8月^{21^日}と8月^{22^日}の間の真夜中の13秒前から始まる連続的な時間スケールであり、すなわち、GSTは1999年8月^{22^日}00:00:00 UTCで13秒に等しくなっていた。GPSTとGSTの違いが出ないようにするために行ったものです。

1.4.2.3 グローナスの時間

GLONASS時刻は、GLONASS Central Synchroniserによって生成され、ロシアとのUTC (UTC(SU)) との差が表示されます。GLONASSTは1ms+3hを超えてはいけません。

$$GLONASST = UTC(SU) + 3h + \tau,$$

ここで、 $|\text{"L1_1D70F"}| < 1ms$ 。GLONASSはUTCのように閏秒を実装しています。GLONASSTは、1年の中で2番目の日と1年の中で1日を基準にしています。

1.4.2.4 BeiDou Time

BeiDou Time (BDT) は、2006年1月1日の0^時UTCから始まる連続的な時間スケールであり、UTCに向けて舵を切っています。

1.4.2.5 GNSS基準時刻の関係

マルチGNSS受信機は、PVTソリューションのために異なるGNSSを追跡し、使用することができます。通常、受信機は1つの基準時間に基づいて1つのクロックのみを実装します。表2に、GNSS基準時刻とTAI、UTCとTAIの関係を示す。図3は、基準時間とTAIとの時間差を示す図である。

表2.リファレンスタイムの関係

システム	関係
GPST - TAI	$T_{AI} = GPST + 19s$
GST - TAI	$T_{AI} = GST + 19s$
グローナスト - 田井	$T_{AI} = GL"LU_1D442"NASST - 3h + leapsecond_{UTC-TAI}$
UTC - TAI	$UT - T_{AI} - leapsecond_{UTC-TAI}$
BDT-TAI	$T_{AI} = BDT + 33s$

UTCとTAIの間の秒単位の時差は、 $leapsecond_{UT-C-TAI}$ 。しかし、GNSSコミュニティでは、閏秒をUTCとGPSの時間差と表現しています。その差は、執筆時点では、次のように表現することができます。

$$leapsecond = leapsecond_{UT-C-TAI} - 19s$$

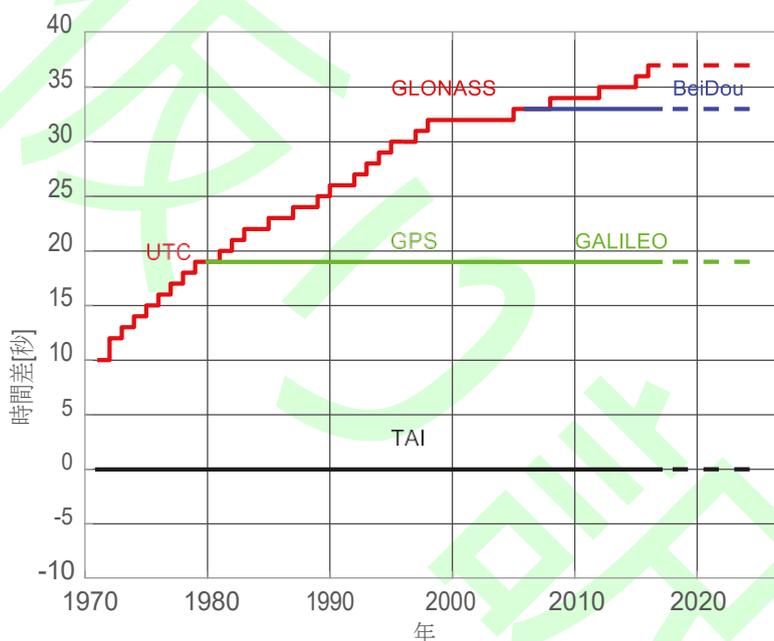


図3：基準時間の時間差

GNSS受信機は通常、GLONASS測定値の変換を必要とするGPSシステムの時間でGNSS時間を提供します。 $GPST = GL"Lu_1D442"NASST - 3h + leapsecond$ 、Androidの生データを元にGLONASS擬似レンジを生成します。

1.5 ナビゲーションメッセージとトラッキングステータス

受信機は衛星から送信された信号に同期する必要があります。現在、GPSはL1[RD-09]で4つの異なる信号を送信していますが、C/A (Coarse/Acquisition) コード信号が最も重要であり、マスマーケットデバイスで最も頻繁に使用されている信号です。PRNコードは、1msあたり1.023チップのチップングレートで1msの長さのコードに基づいています。C/Aコードのローカルレプリカは、適切な遅延とドップラーで生成されます。図4は、衛星から送信される周期的なC/Aコード（1ミリ秒ごとに繰り返される）と、受信機で生成されたローカルレプリカ（青）との間の完全な同期を示しています。

この段階では、受信機は完全な送信時間を提供することができず、小数の擬似レンジのみが利用可能である。例えば、 t_0 では、受信機はC/Aコード期間の開始までの相対的な遅延のみを提供することができる。同様に、 t_1 では、受信機は、現在のC/Aコードの先頭までの遅延を提供する。 t_0 と t_1 の間のC/Aコード全体の数は不明です。

受信機が C/A コードのみに同期している場合、送信時間の有効範囲は 0~1ms となり、移動時間の計算が曖昧になる。

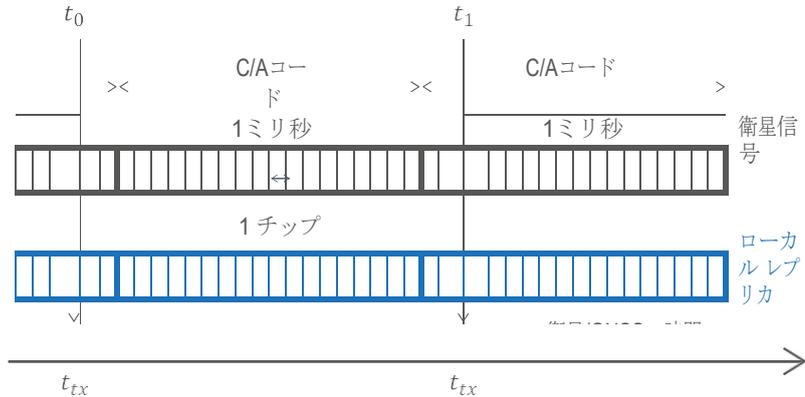


図4 : C/Aコードとの同期

この時間の曖昧さを解決するために、受信者はナビゲーションメッセージの構造を利用することができる。図5は、衛星時刻を取得するために受信機が必要とする様々な同期化の段階を示しています。

- **コードロック** : 受信機はC/Aコードにロックされています。有効範囲は0~1msです。
- **ビット同期** : 受信機はビットと同期しています。有効範囲は0~20msです。
- **サブフレーム同期** : 受信機はサブフレームと同期しています。有効範囲は0~6秒です。
- **TOW Decoded** : すべてのサブフレームにTOWが含まれています。したがって、TOWがデコードされると、有効範囲は0~1週間である。

既存の文献には、分数的なシュドランジュのみを用いてPVT解を計算することができる、より高度なアルゴリズムが記載されている。このナビゲーション解は、粗時間ナビゲーション問題[RD-10]として知られている。さらに、A-GNSS受信機は、外部システムを介してTOWを取得することも可能であり、放送されたメッセージをデコードする必要がない。

GPSの場合、連続する5個のサブフレームがフレームを構成し、25個のフレームでナビゲーションメッセージが形成される。GPSメッセージの構造に関するより多くの情報は、[RD-09]に記載されている。

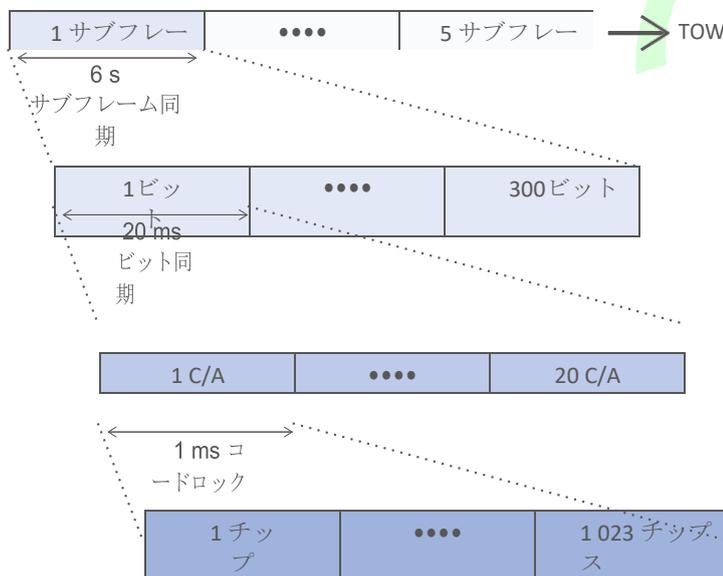


図5.GPS L1 C/Aコードとナビゲーションメッセージの構造

ガリレオは、E1オープンサービス (OS) 変調、つまり2つのコンポーネントの多重化を使用しています。

- **E1B** : E1 OSデータコンポーネント。I/NAVナビゲーションデータストリームとプライマリコード(E1BC)から生成されます。
- **E1C** : E1 OSのPilotコンポーネント。一次コード (E1BC) と二次コード (E1C_2nd) で構成されている。

図6は、構成要素E1BおよびE1Cの両方の構成を示す図である。第1は、長さ4msの一次コードE1BCに基づいており、シンボルの長さはE1BCコードと同じである。ページ (I/NAVサービス) は、TOWを含む500シンボルで構成されています。サブフレームは15ページで構成されています。フルI/NAVメッセージは34のサブフレームで構成されています。

パイロットコンポーネントは、長さ4msの一次コードと、その上に配置された二次コードE1C_2ndで構成されています。25チップの長さで、各チップの持続時間は4msです。フルセカンダリコードの長さは100msです。パイロット・コンポーネントにはナビゲーション・メッセージがないため、受信機はより長い時間、信号をコヒーレントに追跡することができ、より弱い信号を受信することができます。多くの受信機は、エフェメライドと時計が外部ソースを介して取得された場合、またはナビゲーションメッセージからデコードされた場合に、パイロットコンポーネントを追跡する。

受信機が衛星信号と同期し、GNSS時刻を取得すると、曖昧さのない送信時刻を計算することができます。GNSSの時刻は外部から取得することも可能であるが、これらの方法は本文書の範囲外である。第二のガリレオコードを追跡すると、100msの曖昧な時間が発生します。衛星から受信機までの信号の移動時間が70msであるため、GPSとは異なり、完全な疑似距離を簡単に生成することができます。

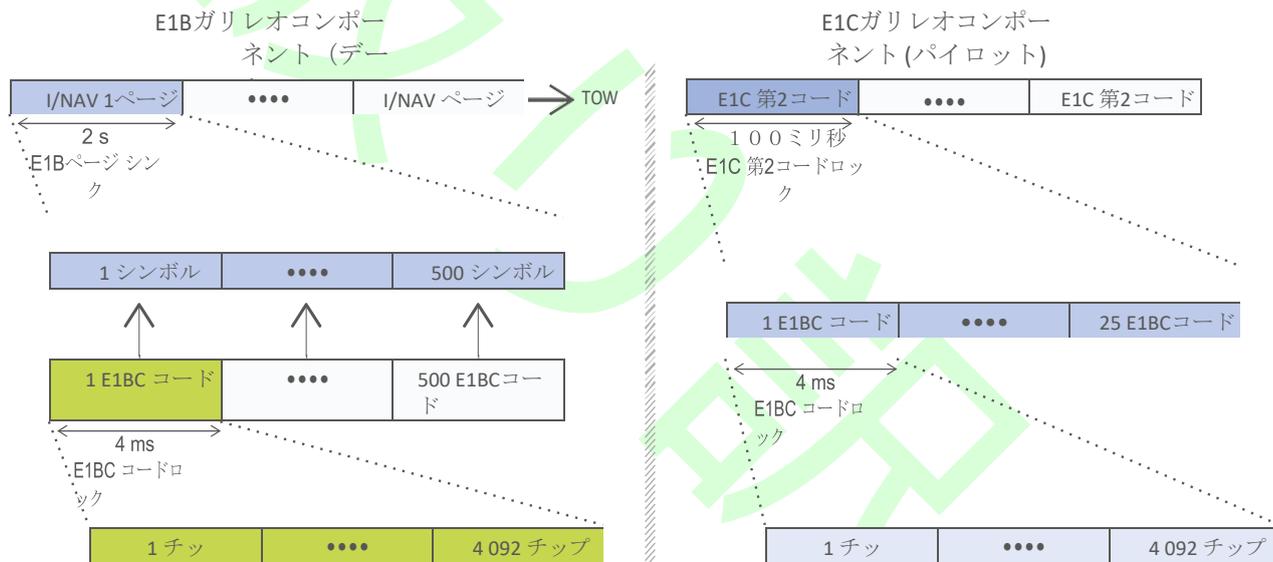


図6.E1Bガリレオ部品 (左側)、E1Cガリレオ部品 (右側)

北斗やGLONASSの航法メッセージの構造や追跡状況はGPSと似ています。表3は、4つの主要な星座の時間の曖昧さと追跡状況をまとめたものです。

表3.追跡状態に基づく時間の曖昧さ

GPS		ガリレオ		GLONASS		BeiDou	
同期状態	時間	同期状態	時間	同期状態	時間	同期状態	時間
C/Aコード	1ミリ秒	E1BCコード	4ミリ秒	C/Aコード	1ミリ秒	C/Aコード	1ミリ秒
ビット	20ミリ秒	E1C 第2コード	100ミリ秒	ビット	20ミリ秒	ビット	20ミリ秒
サブフレーム同期	6 s	E1Bページ	2 s	ひもじ	2 s	サブフレーム同期	6 s
TOW	1週間	TOW	1週間	時間帯	1日	TOW	1週間

1.6 擬似レンジ生成

擬似距離の概念は単純ですが、この距離測定は時間測定によって得られるため、その生成は単純ではありません。GNSS受信機は、受信した信号を処理して、送信時間 (t_{Tx}) と受信時間 (t_{Rx}) を得る。両者の違いは、衛星から受信機までの信号の移動時間です (電離層や対流圏などによる余分な遅延がないと仮定しています)。擬似距離は次のように計算できます。

$$\rho = (t_{Rx} - t_{Tx}) \cdot c,$$

ここで、 c は真空中での光速である。 t_{Tx} は衛星の位置を計算するためにも必要です。

擬似レンジ生成とPVT計算は、共通の受信時間と共通の送信時間の2つの異なる方法で実装することができます [RD-11]。以下の段落では、一般的な受信時間法に基づいて擬似レンジ生成を行う。

共通の受信時間は測定時間として知られています (t_{meas}^{Rx})。4つの衛星が同時に信号を送信しています。TOW 同じ時間エポック (図7の左側) ですべての衛星が送信していることを示しています。異なる伝搬経路のため、4つの信号は異なる遅延で受信機に到着します。その後、受信機は時間オフセットを計算します。の間にはTOW とエポックでの現在時刻とのこれは測定時間として知られています (t_{meas}^{Rx})。その結果、トランス 測定時の衛星時刻 (GNSSシステムの時刻で提供される) は、次のように表すことができます。

$$\begin{aligned} t_{Tx,Sat1} &= t_{TOW}^{GNSS} + \Delta_1, \\ t_{Tx,Sat2} &= t_{TOW}^{GNSS} + \Delta_2, \\ t_{Tx,Sat3} &= t_{TOW}^{GNSS} + \Delta_3, \\ t_{Tx,Sat4} &= t_{TOW}^{GNSS} + \Delta_4, \end{aligned}$$

ここで、 t_{TOW}^{GNSS} 送信 TOW と Δ_i は TOW から測定時間までの遅延時間です。

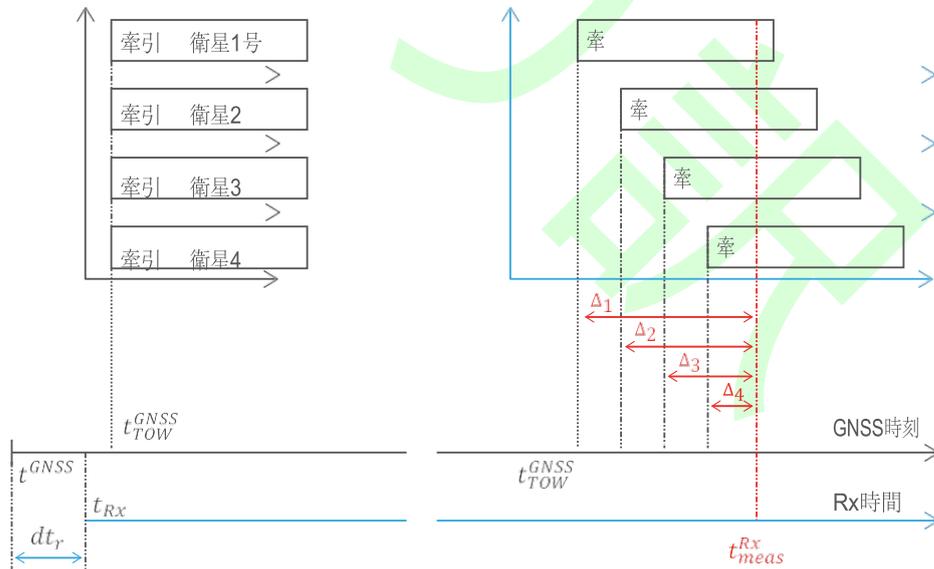


図7. 共通受信時間に基づく擬似レンジ計算

最初のPVTを計算し、デコードする前に TOW 少なくとも1つの衛星から、受信機はGNSS時刻に関する情報を持っていません。したがって、Rx時間を生成し、疑似範囲の最初のセットを計算するためには、仮定をしなければなりません。最初に到着する衛星信号を基準にしています。受信した時間 (t_{meas}^{Rx}) は送信時間と参照 伝搬時間 (t_{ref}^{path})。通常は65~85msの間の標準値を想定しています。したがって、最初に測定された 場合は次のように計算できます。

$$t_{meas}^{Rx}[1] = t_{Tx,Sat1}[1] + t_{ref}^{path}.$$

他のすべての疑似オレンジは、最初のものに対して相対的に生成されます。すべての衛星に対して一定の誤差が存在するので (t_{ref}^{path} が固定されており、推定されていない)、GNSS と受信時刻の間にバイアスが発生する ()。GNSS 基準時間での測定時間は、次のように表すことができる。

$$t_{meas}^{Rx,GNSS}[k] = t_{meas}^{Rx}[k] - dt_r[1],$$

ここで k - 番目の測定時間です。疑似レンジの最初のセットが計算され、得られた最初のPVTが決定されると、 $dt_r[1]$ 次の疑似レンジのすべてのセットに使用されます。しかし、受信機が計算して更新するのはすべてのPVTセットでクロックバイアスを設定します。したがって、次のような疑似レンジの集合が計算できる。

$$\rho_1[k] = (t_{meas}^{Rx}[k] - t_{Tx,Sat1}[k] + dt_r[1])c,$$

$$\rho_2[k] = (t_{meas}^{Rx}[k] - t_{Tx,Sat2}[k] + dt_r[1])c,$$

$$\rho_3[k] = (t_{meas}^{Rx}[k] - t_{Tx,Sat3}[k] + dt_r[1])c,$$

$$\rho_4[k] = (t_{meas}^{Rx}[k] - t_{Tx,Sat4}[k] + dt_r[1])c,$$

孰れ c は真空中での光の速さです。一部の受信機ではそのとオフ $t_{meas}^{Rx}[k]$ 測定時間の合計としてセットがあります。

1.7 位置推定

1.7.1 シングルGNSS星座

疑似距離へのさらなる影響を無視して、受信アンテナと -ばんざい 衛星は、実際の範囲で構成されています。 ρ^j と受信機のクロック誤差を乗算したものです。 掖 真空中の光の速さ

$$R^j = \rho^j + c \cdot dt_r.$$

受信アンテナ座標を推定することを目的としてい (x, y, z) と疑似レンジ測定値のセットからのクロックオフセット.送信された航法メッセージなどから得られた -番目の衛星の位置 (x^j, y^j, z^j) を知ることで、疑似距離を以下のように書き換えることができます。

$$R^j = \sqrt{(x^j - x)^2 + (y^j - y)^2 + (z^j - z)^2} + c \cdot dt_r, j = 1, 2, \dots, n (n \geq 4).$$

少なくとも、同時の範囲測定では 四つ 人工衛星が視界に入っていれば問題は十分に解決します。この方程式は非線形系を定義しているので、近似的な位置から出発して反復的な線形化で解いていきます。 (x_0, y_0, z_0) 受信機の

$$R^j - \rho_0^j = \frac{x_0 - x^j}{\rho_0^j} dx + \frac{y_0 - y^j}{\rho_0^j} dy + \frac{z_0 - z^j}{\rho_0^j} dz + c \cdot dt_r, j = 1, 2, \dots, n (n \geq 4),$$

孰れ $dx = x - x_0, dy = y - y_0, dz = z - z_0$ そし ρ_0^j j でござる その 計算範囲 (x^j, y^j, z^j) にとって えいせい (x_0, y_0, z_0) に於いて その 座標系 と 近似解 受信機の位置の方程式系は次のように表すことができます。 に於いてとしての行列記法。

$$\begin{bmatrix} R^1 - \rho_0^1 \\ \vdots \\ R^n - \rho_0^n \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{x_0 - x^1}{\rho_0^1} & \frac{y_0 - y^1}{\rho_0^1} & \frac{z_0 - z^1}{\rho_0^1} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \frac{x_0 - x^n}{\rho_0^n} & \frac{y_0 - y^n}{\rho_0^n} & \frac{z_0 - z^n}{\rho_0^n} & 1 \end{bmatrix}}_H \begin{bmatrix} dx \\ dy \\ dz \\ c \cdot dt_r \end{bmatrix}.$$

4つの未知数を解くためには、少なくとも4つの衛星が視界に入っている必要があります。疑似距離には様々な誤差の原因が含まれているため (図8)、位置を推定するために利用可能なすべての疑似距離測定値を使用することが有益です。

4つ以上の衛星 ($n > 4$) については、非線形最小二乗法 (または関連するカルマンフィルタ) の推定手順を使用して解くことができる、過決定システムが得られる。式を解いた後、更新された受信座標の推定値は

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} + \begin{bmatrix} dx \\ dy \\ dz \end{bmatrix}.$$

連続したステップでは、更新された受信位置 (x_0, y_0, z_0) を用いて、この疑似距離方程式を再び線形化することができる。これらのステップは、2つの連続する反復の間の差が所定のしきい値以下になるまで繰り返されます。位置決めの詳細については、[RD-12, RD-13, RD-14]を参照してください。

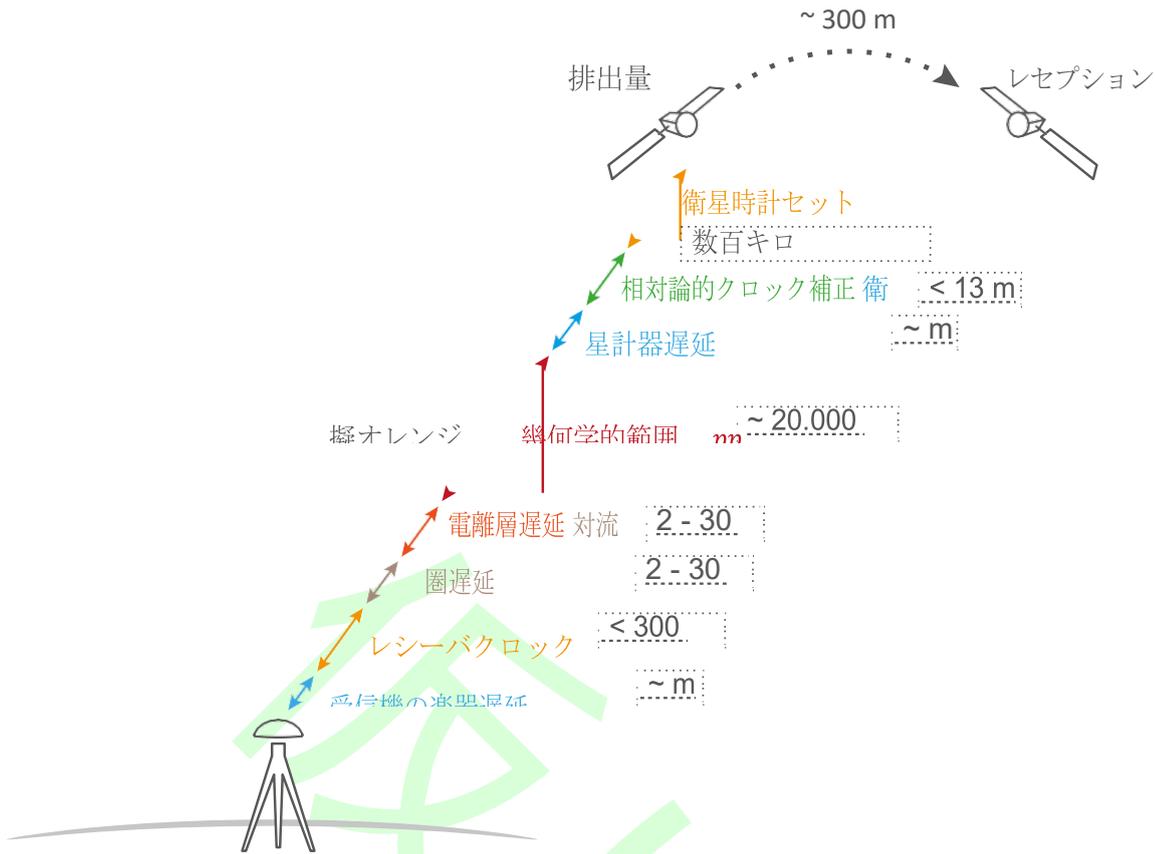


図8GNSS疑似測域誤差の発生源 [RD-05]

1.7.2 マルチGNSS星座

複数のGNSS星座を使用して位置解を解くことができます。過酷な環境下では、複数のGNSSコンステレーションを組み合わせることで、測位精度を向上させることができます。しかし、異なる星座からの信号をPVTで結合するには、システム間の時間バイアス（Inter-System Bias, ISB）を考慮する必要があります [RD-16, RD-17]。そうでなければ、範囲測定には追加の誤差が含まれており、結果的に位置情報を劣化させます。一つの戦略は、ISBのアプリオリ知識の利用である。例えば、ガリレオのGPSとのシステム間操作性のおかげで、GPSからガリレオへの時間オフセット（GGTO）がナビゲーションメッセージで送信されます。また、アシストデータを介してユーザーに提供することもできる。ISBを除去すれば、単一の衛星航法システムの位置解を計算することができる [RD-16]。

また、位置方程式に追加の未知のパラメータとして導入することで、ISBを推定することも可能である。GPS、GLONASS、Galileo、BeiDouの4つのシステムすべてが測位に使用されている場合、GPSと比較してGLONASS、Galileo、BeiDouの3つのシステム間のバイアスを決定する必要があり、その結果、拡張ナビゲーション方程式が得られます。

$$\begin{bmatrix} R^1 - \rho_0^1 \\ \vdots \\ R^n - \rho_0^n \end{bmatrix} = H \cdot \begin{bmatrix} dx \\ dy \\ dz \\ c \cdot dt_r \\ c \cdot dt_{GPS/GLONASS} \\ c \cdot dt_{GPS/Galileo} \\ c \cdot dt_{GPS/BeiDou} \end{bmatrix}$$

この場合、少なくとも7機の衛星の範囲測定が必要となる。これは、状態外定理に続いて、PVTソリューションの実際の精度の希釈（DOP）にも影響することに注意してください。最小二乗問題に余分な状態を追加することで、すべてのDOPが元の値よりも大きいか、元の値と同じになり、位置決め誤差が大きくなります。



2. GNSS RAWへのアクセス方法 Android APIsを使用した測定

2.1 序章

Androidプラットフォーム上で位置を取得する最も一般的な方法は、精度、最初の修正までの時間、可用性、または消費電力を向上させるために、複数のソース（GNSS、Wi-Fi、またはモバイルネットワーク）を組み合わせた融合型位置情報プロバイダを介して取得することです。**android.gsm.location** APIを介して利用できます。新しいAndroid 7（Nougat）版は、その後のバージョン（Android O）とともに、新しいAndroid Location API（**android.location**）を導入し、位置情報サービスに大きな革新をもたらします。

この章では、**android.location** APIと、生測定値そのものから代表的なGNSSパラメータを取得するために必要な方法論について説明します。それはGNSSの位置情報機能にのみ焦点を当てています。その他の位置情報の方法（Wi-Fi、モバイルネットワークなど）は論外です。

2.2 ^仮Android 7以前の位置情報API - **android.gsm.location**

AndroidアプリケーションとGNSSなどのさまざまなモバイルセンサーとの間の相互作用は、AndroidフレームワークのAPIを使用して行われます。Androidプラットフォームの各新バージョンは新しいAPIに関連付けられており、GNSSチップセットへの設定、相互作用、ユーザーのアクセスはすべてAPIレベルに依存しています。表4は、AndroidのバージョンとAPIレベルの関係を示しています。

表4.AndroidのバージョンとAPIレベル

バージョン	コードネーム	API
2.3.3 – ブレッド2.3.7	ジンジャー	10
4.0.3 – 4.0.4	アイスクリーム サンドイッチ	15
4.1.x 4.2.x 4.3	ゼリービーン	16 17 18
4.4	キットキャット	19
5.0 5.1	ロリポップ	21 22
6.0	マシュマロ	23
7.0 7.1	ヌガー	24 25
8.0	オレオ	26

図9は、GNSSチップセットドライバやチップセットとOS間のデータ転送を含む、Android 7以前のLocation API（APIレベル24）の構造をまとめたものです。

ユーザーアプリケーションは、フレームワークの位置情報APIを使用してGNSSデータにアクセスします。Android API 23までは、図9の下の方に記載されているデータ、即ち、データに限定されていましたが、Android API 23では、図9の下の方に記載されているデータに限定されています。GPS衛星情報（C/No、方位角、標高）、NMEA文、PVT解答（適切なタイムスタンプ付き）。ユーザーは、GNSSチップセットの再起動/起動やアシストデータのクリーニングなど、基本的な設定コマンドをチップセットに送信することができます。しかし、GNSSコンステレーションの優先順位や異なるPVTアルゴリズムを含むすべての構成設定は、チップセット自体によって駆動されます。

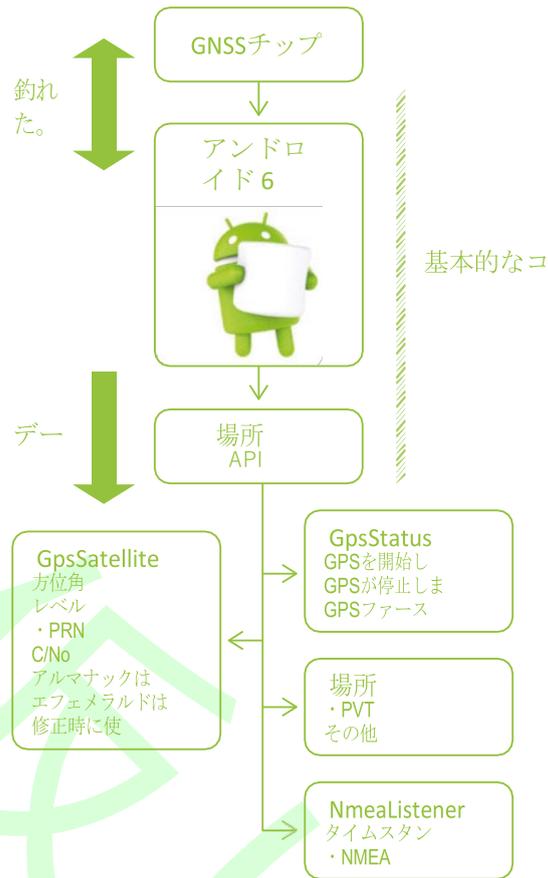


図9 Android APIレベル23の位置情報API

API 23 (Android 6) から、開発者は以下のAndroidクラスにアクセスできるようになりました。

- GPS衛星は、方位、標高、PRNとC/Noなどの基本的な衛星情報を含む。また、衛星がPVTソリューションとアルマナックとエフェメライズの可用性で使用されている場合、それはフラグを立てます。
- GPSステータスは、GNSSチップセットのステータスとソリューションに関する情報を提供します。
- 位置と時間のソリューションが提供されているかどうかを示す位置。
- NMEAリスナー、基本的なNMEAの文章を提供します。

新しいAPIレベルは、以前のバージョンと互換性があります。つまり、以前のAPIレベルのすべての機能が新しいバージョンでも有効であることを意味します。図10は、Android 7で実装されている位置情報APIレベル24を示す図である。

API 24 (Android 7)から、開発者はAndroidクラスを介して以下のGNSSの生情報と計算情報にアクセスすることができます。

- GNSS時計、それが含まれています。
 - 受信時間（疑似距離の計算に使用されます）。
 - クロックバイアス。
- を含むGNSSナビゲーションメッセージ。
 - ナビゲーションメッセージビット（すべての星座）。
 - ナビゲーションメッセージの状態です。
- 含まれているGNSS測定。
 - 受信衛星時刻（疑似距離の計算に使用）。
 - コードは？
 - キャリアフェーズ。

このデータはGNSSチップセットから直接取得したのですが、チップセット自体に直接アクセスすることはできません。

アンドロイドデバイスでのGNSS生測定値の利用

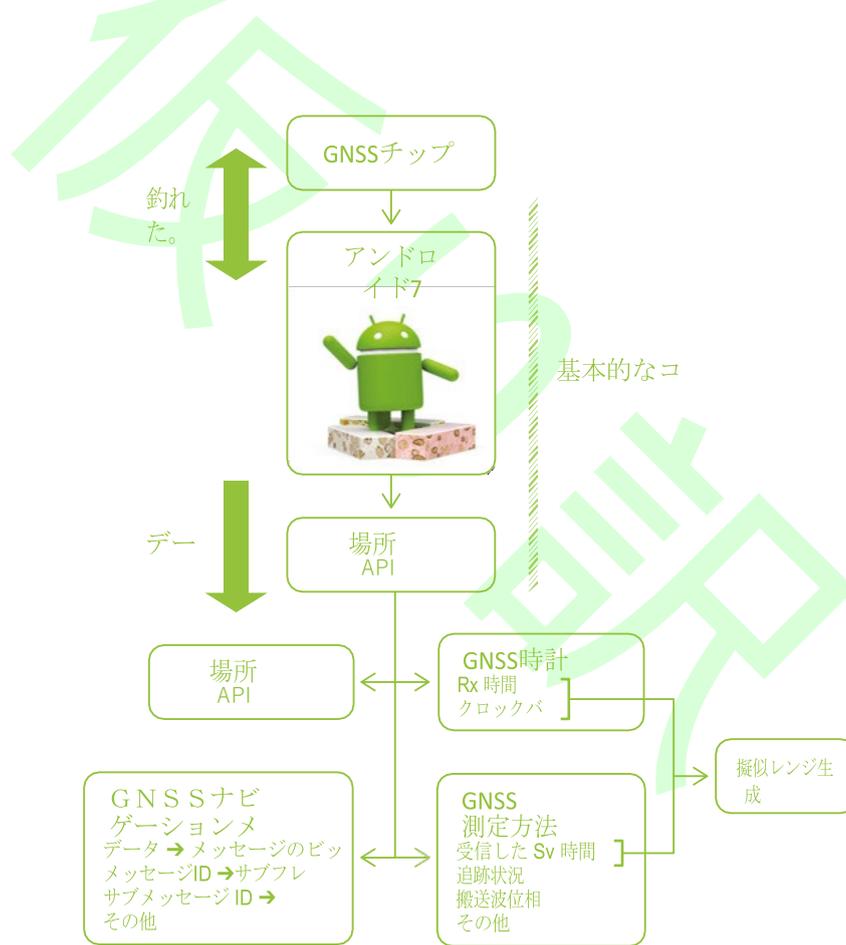


図10Androidの位置情報APIレベル24/25/26

2.4 GNSSの生測定値を使用する

この章の主な目的は、生測定値から代表的なGNSSパラメータを得る方法を説明することです。**Android APIは、素直な疑似体験を提供していません。**新しいAndroid 7 Location APIのフィールド一覧（表5）には、GNSS時刻や疑似距離は含まれていません。これらは計算する必要があります。

表5.Android 7 位置情報 API - 時計と計測フィールド

Android 7の位置情報 - 時計と計測		
アンドロイドクラス	フィールド	説明
GNSSクロック	タイムナノス	ナノ秒単位でのGNSS受信機の内部ハードウェアクロック値
GNSSクロック	<i>BiasNanos</i>	時計のサブナノ秒バイアス
GNSSクロック	<i>FullBiasNanos</i>	0000Z、1980年1月6日以来、GPS受信機内部のTimeNanosと真のGPS時間との間の差
GNSSクロック	<i>DriftNanosPerSecond</i>	時計の流れ
GNSSクロック	<i>HardwareClockDiscontinuityCount</i>	ハードウェアクロックの不連続性のカウント
GNSSクロック	リースセコンド	時計の時間に関連付けられたうるう秒
GNSS測定	コンステレーションタイプ	星座型
GNSS測定	スヴィッド	衛星ID
GNSS測定	状態	GNSSエンジンの現状

GNSS測定	<i>ReceivedSvTimeNanos</i>	測定時のGNSS衛星の受信時刻
GNSS測定	<i>AccumulatedDeltaRangeMeters</i>	最後のチャンネルリセット以降の累積デルタ範囲
GNSS測定	<i>Cn0DbHz</i>	搬送波対雑音密度
GNSS測定	<i>TimeOffsetNanos</i>	ナノ秒単位で測定された時間オフセット
GNSS測定	キャリアサイクル	衛星と受信機間のフルキャリアサイクル数
GNSS測定	<i>CarrierFrequencyHz</i>	コードとメッセージが変調されるキャリア周波数
GNSS測定	疑似オレンジ色温度計第2部	タイムスタンプにおける疑似距離のレートを取得します。

Androidのナビゲーションメッセージ情報は、表6のGPS C/A、BeiDou D1&D2、GLONASS L1 C/A、Galileo F/NAV、I/NAVに対応しています。ただし、提供される正確な情報は、GNSSチップセットとスマートフォンのメーカーの両方に依存します。

表 6.Android 7 Location API - ナビゲーションメッセージのパラメータ

Android 7 位置情報 - ナビゲーション メッセージ	
パラメーター	説明
データ	<p>報告されたGPSメッセージのデータを取得します。</p> <ul style="list-style-type: none"> GPS L1 C/A、北斗D1、北斗D2の場合、各サブフレームには10個の30ビットワードが含まれています。各ワード（30ビット）は、それぞれ6、6、0.6秒の期間をカバーする4バイトワード（B31とB32をスキップ）の最後の30ビットに収まるように、MSBを先頭にして合計40バイトに収まる必要があります。 Glonass L1 C/Aの場合、各文字列にはチェックサムを含む85ビットのデータが含まれています。これらのビットは11バイトに収まり、MSBを最初に(B86-B88をスキップして)2秒の時間をカバーします。 Galileo F/NAVの場合、各ワードは238ビット（シンクとテールシンボルを除く）で構成されています。各ワードは30バイトに収まるように、MSBを先頭にして(B239, B240をスキップ)、10秒の時間をカバーするようにします。 Galileo I/NAV の場合、各ページには偶数と奇数の 2 つのページ部分があり、合計 2 x 114 = 228 ビット（同期と尾部を除く）で 29 バイトに収まり、MSB が先になります（B229-B232 をスキップ）。
メッセージID	<ul style="list-style-type: none"> GPS L1 C/A のサブフレーム 4 と 5 の場合、この値は、1~25 の範囲のナビゲーションメッセージの「フレーム ID」に対応します（サブフレーム 1、2、3 には「フレーム ID」は含まれておらず、この値を -1 に設定することができます）。 グロナスL1 C/Aの場合は、1~5の範囲のフレームIDを指します。 北斗D 1 の場合は、1 ~ 2 4 の範囲の枠番号を指す。 北斗D 2 の場合は、1 ~ 1 2 0 の範囲のフレーム番号を指す。 ガリレオF/NAVの公称フレーム構造の場合、1 ~ 1 2 の範囲のサブフレーム番号を指す。 ガリレオI/NAVの公称フレーム構造では、1 ~ 2 4 の範囲のサブフレーム番号を指す。
ステータス	ナビゲーションメッセージのステータスを取得します。
サブメッセージID	<ul style="list-style-type: none"> GPS L1 C/A、BeiDou D1、BeiDou D2の場合は、1~5の範囲のナビゲーションメッセージのサブフレーム番号に対応しています。 グロナスL1 C/Aの場合は、1~15の範囲の文字列番号を指します。 Galileo F/NAVの場合は、1~6の範囲のページタイプを指します。 Galileo I/NAVの場合は、1~10の範囲の単語タイプを指します。 <p>特にガリレオについては、データビット内に埋め込まれた情報は、この分野で提供されている公称ページやワードタイプよりも、解釈のためにさらに有用であるかもしれない。</p>
スウィッド	衛星ID
タイプ	ナビゲーションメッセージの種類を取得します。

アンドロイドデバイスでのGNSS生測定値の利用

次の段落では、入手方法を説明しています。

- GNSS時間
- 擬オレンジ
- 搬送波位相測定
- ドップラー
- 衛星ID
- コンステレーションID

さらに、スマートフォンで消費電力を抑えるために使われているデューティサイクル技術が、測位性能にどのような影響を与えているのかが議論されています。

2.4.1 GPS時間生成

Android 7ではGNSS時間を直接提供していませんが、受信機がGNSS基準時間を推定している場合は、内部のハードウェアクロックと真のGPS時間（ナノ秒単位）へのバイアスが提供されます。受信機がGPS時間を推定した場合、それは次のように計算することができます。

$$GpsTime = TimeNanos - (FullBiasNanos + BiasNanos)[ns],$$

孰れ $FullBiasNanos$ は、受信機のクロックとGPS時間との間のバイアス（ナノ秒単位）であり、 $BiasNanos$ はサブナノ秒単位です。受信機がGPS以外のコンステレーションを使用して時間を推定した場合、推定されたGPS時間は次のように計算できます。

$$GpsTime = TimeNanos - (FullBiasNanos + BiasNanos) - InterSystemsBias,$$

孰れ $InterSystemsBias$ は、時刻推定に使用したGPS時刻とGNSS時刻のオフセットです。例えば、ガリレオシステムの時刻を使って推定した場合、 $InterSystemsBias$ GGTOに等しい。

2.4.2 擬似レンジ生成

2.4で既に述べたように、Androidシステムでは直接擬似レンジを提供していません。その代わりに、計算に必要なすべてのパラメータを提供します。述べたように、擬似レンジの構築は、時間差、すなわち、受信時刻（測定時刻）と送信時刻との時間差に基づいて行われる。

$$\rho = \frac{(t_{Rx} - t_{Tx})}{1E9} * c,$$

孰れ t_{Rx} は、測定時刻におけるGNSS衛星の受信時刻、すなわち、信号が送信されたGNSS基準時刻である。

t_{Tx} は測定時間、 c は真空中の光の速度である。としてAndroid 7のシステムで提供されています。

$$t_{Tx} = ReceivedSvTimeNanos[ns],$$

孰れ $ReceivedSvTimeNanos$ は、測定時刻におけるGNSS衛星の受信時刻をナノ秒単位で表したものです。の有効範囲は t_{Tx} はトラッキングステータスに依存しており(第1章)、もしトラッキングステータスがTOW (GPS)をデコードした場合、擬似レンジが曖昧になる、つまり擬似レンジが曖昧になる。曖昧さを利用したPVTの計算擬似レンジ。

この文書では、粗い時間のナビゲーション問題は取り上げていない(詳細は[RD-10]を参照のこと)。完全な擬似範囲（曖昧でない）を得るためには、以下の追跡状態が必要です。

- GPSだTOWデコード
- 北斗。TOWデコード
- グロナス時間帯
- ガリレオ
 - TOWデコード
 - E1C第2コード。有効な範囲は0~100msecであるにもかかわらず、これらの測定値は擬似範囲の計算に使用する必要があります。

実用的なテストでは、短い時間の後、ガリレオ測定はTOWデコードされたTOWからE1C 2ndコードの状態になり、数分間この状態のままにいらることが示されています。いくつかのガリレオレディチップは、ナビゲーションメッセージをデコードするためにデータコンポーネントを追跡します。エフェメライドとクロックが適切にデコードされると、それらはE1Cコンポーネント（パイロットコンポーネント）のトラッキングを開始し、トラッキングステータスはE1C 2ndコードステータスとしてフラグを立てられ、擬似レンジの曖昧さは100msとして、完全な擬似レンジの再構成を可能にする。著者は、E1C第2コードステータスのガリレオ測定値のみを使用することを開発者に強く推奨しています。

フルGNSS時間での測定時間は、次のように再構成することができます。

$$t_{RX_{GNSS}} = TimeNanos + TimeOffsetNanos - (FullBiasNanos(1) + BiasNanos(1))[ns],$$

孰れ *TimeOffsetNanos* は測定時の時間オフセット（ナノ秒単位）です。の最初の値(1.6節参照)のみを指定します。*FullBiasNanos* 及び *BiasNanos* は、すべての受信時間を計算するために使用されます。この動作は、内部受信時間に不連続性があるまで適用されます。これは通常、GNSSモジュールの再起動時にのみ発生します。

t_{RXGNSS} はGNSS参照システムでのみ提供されており、受信時間の計算に使用されます。ただし、GPS計測であればGPST、グロナス計測であればGLONASSなど、GNSSシステムごとに用意されています。したがって、 t_{RX} 同じ基準時間制に変換する必要があります t_{RXGNSS} 。通常、GNSS受信機は、デフォルトのGNSS基準時間としてGPSを実装しています。

疑似距離を計算する際には、両方の成分が同じ範囲でなければなりません。 t_{RXGNSS} はフルGNSS時間であり、追跡状況に応じた範囲であるため、疑似範囲を計算する際には、両方の成分が同じ範囲にある必要があります。

次の段落では、 t_{RXGNSS} 2つの異なるアプローチを用いて、GPS、Galileo、GLONASS、BeiDouの疑似距離を計算する方法を示します。 t_{Tx} 測定は、2つの変換を必要とします。

- 両方の測定値を同じGNSS時間システムに送信すること；および
- 両方の測定値を同じ有効範囲で測定します。

2.4.2.1 アプローチ1.

Google [RD-18]で提供されているMatlabコードに従って、GPSとGalileoの測定時間を計算することができます。

$$t_{Rx} = t_{RXGNSS} - weekNumberNanos[ns]$$

のためにとの北斗 TOW としてデコードされたステータス。

$$t_{Rx} = t_{RXGNSS} - weekNumberNanos[ns] - 14s$$

E1Cの第2コードステータスを持つGalileoのために。

$$t_{Rx} = t_{RXGNSS} - milliSecondsNumberNanos[ns]$$

としてGLONASS(Time of Day)に対応しています。

$$t_{Rx} = t_{RXGNSS} - DayNumberNanos + 3h - leapsecond[ns]$$

式のパラメータはすべてナノ秒単位でなければなりません。*weekNumberNanos* は、GPS時間の開始から現在のWNまでに発生したナノ秒数です。*DayNumberNanos* は、GPS時間の始まりから現在までに発生したナノ秒の数です。*milliSecondsNumberNanos* は、GPS時間の始まりから現在までに発生したミリ秒の数です。として計算することができます。

$$weekNumberNanos = \text{floor} \left(\frac{-FullBiasNanos}{NumberNanoSecondsWeek} \right) * NumberNanoSecondsWeek[ns],$$

$$DayNumberNanos = \text{floor} \left(\frac{-FullBiasNanos}{NumberNanoSecondsDay} \right) * NumberNanoSecondsDay[ns]$$

そして

$$milliSecondsNumberNanos = \text{floor} \left(\frac{-FullBiasNanos}{NumberNanoSeconds100Milli} \right) * NumberNanoSeconds100Milli[ns],$$

ここで、*NumberNanoSeconds* は、1週間以内のナノ秒の数、つまり、*NumberNanoSeconds* = 604800e9となります。*NumberNanoSecondsDay* は1日以内のナノ秒数、*NumberNanoSeconds100Milli* は100ms以内のナノ秒数です。

2.4.2.2 アプローチ2.

を使用して *mod* 演算子を用いて、TOWを復号したGPSとGalileoの測定時間を次のように求めることができる。

$$t_{Rx} = \text{mod}(t_{RXGNSS}, NumberNanoSecondsWeek)[ns],$$

TOWがデコードされたBeiDouのために。

$$t_{Rx} = \text{mod}(t_{RXGNSS}, NumberNanoSecondsWeek) + 14s[ns],$$

E1C 2ndステータスでGalileo用に。

$$t_{Rx} = \text{mod}(t_{RXGNSS}, NumberNanoSeconds100Milli)[ns]$$

そして

$$t_{Rx} = \text{mod}(t_{RXGNSS}, NumberNanoSecondsDay) + 3h - leapsecond[ns]$$

GLONASSの場合、Time of Dayのステータスがあります。

持つこと、そして、疑似オレンジの構築は簡単です。

$$\rho = \frac{(t_{Rx} - t_{Tx})}{1E9} * c.[S].$$

2.4.2.3 例としては、以下のようなものがあります。

2.4.2.3.1 例1.

以下のMatlabコードスニペットは、TOWがエンコードされている場合のGalileo、GPS、BeiDou信号の疑似距離を計算する方法を示しています。

選択 GPS + GAL TOW デコード (ステートビット 3 有効)

```
pos = find( (gnss.Const == 1 | gnss.Const == 6) & bitand(gnss.State,2^3);
測定された時間を完全なGNSS時間で生成する
tRx_GNSS = gnss.timeNano(pos) - (gnss.FullBiasNano(1) + gnss.BiasNano(1));
フルGNSSからTOWへの有効範囲を変更する tRx =
mod(tRx_GNSS(pos),WEEKSEC*1e9).
衛星時刻を生成する
tTx = gnss.ReceivedSvTime(pos) + gnss.TimeOffsetNano(pos);
% Generate the pseudorange
prMilliseconds = (tRx - tTx);
pr = prMilliseconds *Constant.C*1e-9;
```

2.4.2.3.2 例2.

以下のMatlabのコードスニペットは、Time of DayがデコードされたときのGLONASS信号の疑似距離を計算する方法を示しています。

GLONASS と Time of Day を選択してデコードします (ステートビット 7 が

有効) pos = find((gnss.Const == 3) & bitand(gnss.State,2^7).

測定された時間を完全なGNSS時間で生成する

```
tRx_GNSS = gnss.timeNano(pos) - (gnss.FullBiasNano(1) + gnss.BiasNano(1));
```

フルGNSSからTime of Dayに有効範囲を変更し、補正する。

GPSからGLONASSの時間差。3時間+リープセコン

```
tRx = mod(tRx_GNSS, DAYSEC*1e9) + (3*60*60-Constant.LeapSecond)*1e9;
```

衛星時刻を生成する

```
tTx = gnss.ReceivedSvTime(pos) + gnss.TimeOffsetNano(pos);
```

% Generate the pseudorange

```
prMilliseconds = (tRx - tTx);
```

```
pr = prMilliseconds *Constant.C*1e-9;
```

2.4.3 搬送波位相測定

キャリアの位相の測定値は、Android 7では、メートル単位のAumpumulatedDeltaRangeMetersとして提供されています。これらは時間情報を持たない曖昧なもので、受信機はエポック間に発生したサイクル数を数えることしかできないことを意味します。サイクルスリップが発生した場合、受信機はこのカウントを失う。キャリア測定の妥当性は、AccumulatedDelta-RangeStateフィールドを介して提供されます。可能なフラグを表7に示す。有効な測定値のみを使用してください。のためにの計算に使用します。

表7.AccumulatedDeltaRangeState

AccumulatedDeltaRangeState		
ステータス	値	説明
ADR_STATE_CYCLE_SLIP	4	サイクルスリップが検出されました。
ADR_STATE_RESET	2	リセットが検出されました。
ADR_STATE_VALID	1	状態が有効です。
ADR_STATE_UNKNOWN	0	状態は無効か不明です。

2.4.3.1 デューティサイクル

スマートフォンの業者は消費電力を優先しています。GNSSチップセットを継続的に動作させるとバッテリーを消耗するため、低消費電力を維持するためにいくつかの技術を実装することができます。最も一般的なものは、図11に示すようなデューティサイクルです。

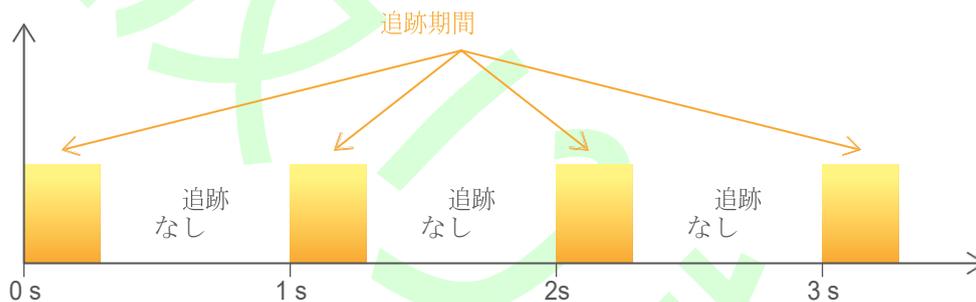


図11デューティサイクル対時間

デューティサイクルには、GNSSクロックの状態に応じて2つの異なる実装があります。

1. **TCXO** ハードウェアクロックが非トラッキング期間中は連続していない（オン/オフする）。この携帯電話には、GNSSトラッキングに使用されている高精度の温度変化補償型発振器（TCXO）と、低消費電力の水晶発振器（XO）の2つの発振器が搭載されています。チップセットがGNSSデータをトラッキングする短いトラッキング期間中は、TCXOが使用されます。この期間以外では、数百ミリ秒の間チップがシャットダウンすると、XOが時間を提供しています。

そのため、HardwareClockDiscontinuityCountが増加し、生測定値を使用してデューティサイクルを検出することができません。

2. **TCXO** ハードウェアクロックは、トラッキングされていない期間中は連続しています：GNSSチップセットがオフになると、信号がトラッキングされていなくても、TCXOクロックは動作し続けます。GNSSクロックには不連続性がなく、HardwareClockDiscontinuityCountは同じ値を維持しているため、ユーザーは受信機がデューティサイクルを使用しているかどうかを検出できません。

この技術は、新しい位置が毎秒計算されるため、ユーザーにとっては透過的ですが、搬送波の位相測定に影響を与えます。連続性をトラッキングしないと、2つの連続した測定の間数回のサイクルスリップが発生する可能性があり、リアルタイムキネマティック（RTK）や精密ポイントポジショニング（PPP）などの先進的な位相技術の使用が著しく制限されます。

2.4.4 ドップラー

衛星の移動に起因するドップラーシフトは、タイムスタンプでの疑似距離率がm/sであることを条件に、PseudorangeRateMetersPerSecondから導出することができます。この値（Hz）には、レシーバと

衛星時計の周波数誤差（未補正值）。正の「無補正」値は、衛星が受信機から遠ざかっていることを示します。補正されていない'擬似レンジ率'の符号と'ドップラーシフト'の符号との関係は式で与えられる。

$$\text{pseudorange rate} = -k * \text{dopplershift},$$

ここで k は信号の中心周波数（例： f_c (L1 = 1575.42e6 Hz)）と光速に依存する定数（ c ）は、次のように、 $k=c/f_c$ すなわち波長を表す。

2.4.5 衛星ID

衛星IDは`Svid`フィールドを使用してAndroid 7から直接提供されます。各星座の期待値は

- GPSだ1–32
- SBASです。120–151, 183–192
- GLONASS：軌道スロット番号(OSN)または周波数チャンネル番号(FCN)のいずれか+100
 - OSNとして1-24（既知であれば好ましい）
 - 93-106をFCNとして（-7→+6）+100。すなわち、-7のFCNを93、0を100、+6を106と符号化する。2つのGLONASS衛星が同じ搬送波周波数で航法信号を送信しているため、正しいOSNを識別するためにはアシストデータが必要になります。
 - QZSSです。193–200
- ガリレオ1–36
- 北斗。1–37

2.4.6 コンステレーション

星座IDは、`ConstellationType`フィールドを使用してAndroid 7から直接提供されます。各星座の値を表8に示します。

表 8.ConstellationType

ConstellationType	
ステータス	値
CONSTELLATION_BEIDOU	5
CONSTELLATION_GALILEO	6
CONSTELLATION_GLOASS	3
CONSTELLATION_GPS	1
CONSTELLATION_QZSS	4
CONSTELLATION_SBAS	2
CONSTELLATION_UNKNOWN	9

2.5 生データアーキテクチャ

ユーザーアプリケーションは、フレームワークのAPI `Location`を使用してGNSSデータにアクセスします。Android 7.0のリリース。でした Google Playサービスに限定しています(Androidプログラムの位置情報)。このAPIは、バッテリーに最適化された融合型位置情報プロバイダを使用し、GNSS、WiFi、Bluetoothなどの複数のセンサーを組み合わせることで、カプセル化された簡易な位置情報に焦点を当てています。すべてがチップ内で行われているため、ブラックボックスと見ることができます：取得ブロックとトラッキングブロックは、ナビゲーションメッセージをデコードし、疑似レンジ、位相測定、時間を生成します。これらはナビゲーションメッセージで修正します（時計誤差、電離層、対流圏など）。最後に、位置・速度・時間（PVT）を計算して出力します。掖チップセットを使用しています。

新しいAPI (`android.location`) は、生のGNSS観測とPVTソリューションへの直接アクセスを提供します。図12は、非アシスト受信機のアーキテクチャと、新旧APIの主な違いを示す図である。わかりやすくするために、新しくアクセス可能になったパラメータ（ホワイトボックス）のみを表示しています。その他のパラメータ（C/No、衛星位置など）は省略しています。Pseu-dorangesは新しいAPIでは直接提供されませんが、それらを生成するために必要なパラメータは提供されます。

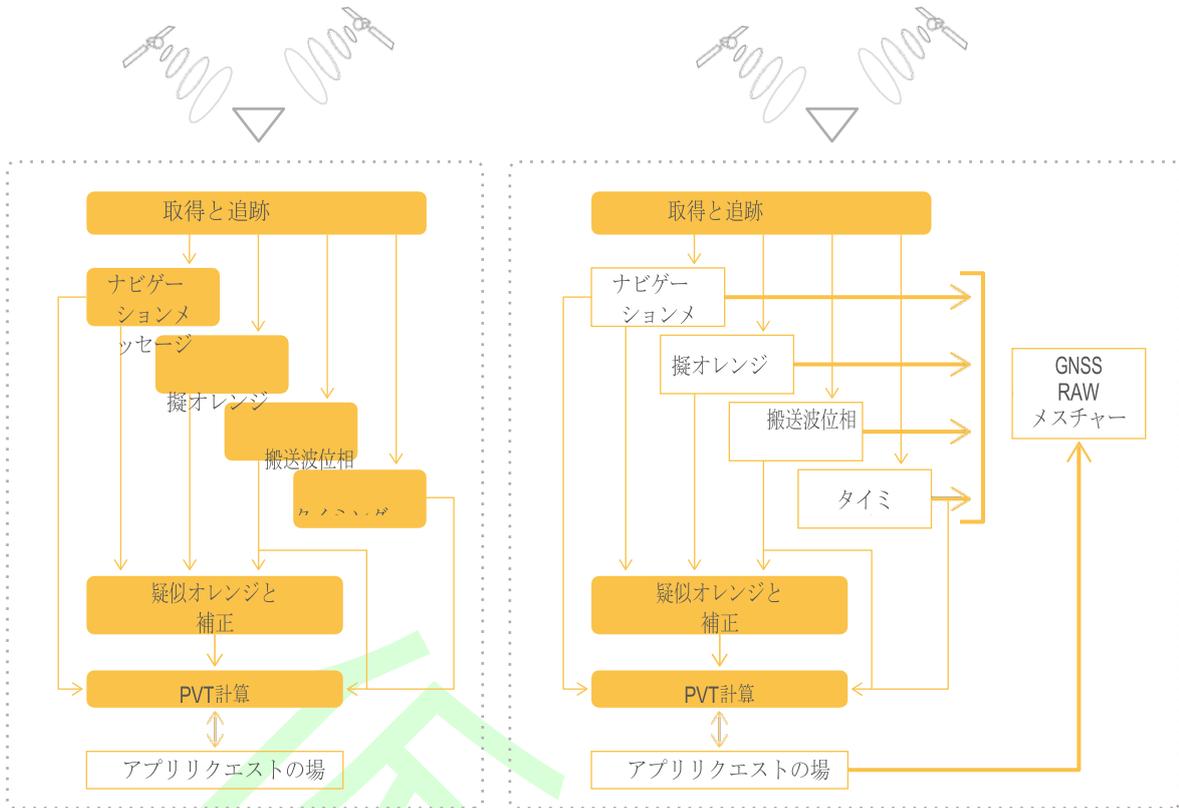


図12 : `android.gms.location`(Android 7以前) vs `android.location`(Android 7)

Androidの位置情報アプリ（LBA）は、精度や電力基準に応じて、位置情報プロバイダ（ネットワーク、Wi-Fi、またはGPS/GNSS）を指定したり、ヒューズ付きプロバイダを使用したりして、`android.location.LocationManager` クラスを介して位置情報を要求します。

生の測定値を利用する場合、このアーキテクチャをカスタムロケーションプロバイダに拡張することができ、例えば、平滑化された疑似距離や差分GNSSデータに基づくことができます。カスタムロケーションは、システム運用レベルで計算する必要があります。この新しいアーキテクチャは図13に示されており、既存のインフラストラクチャを青で、新しい機能の例をオレンジで示しています。

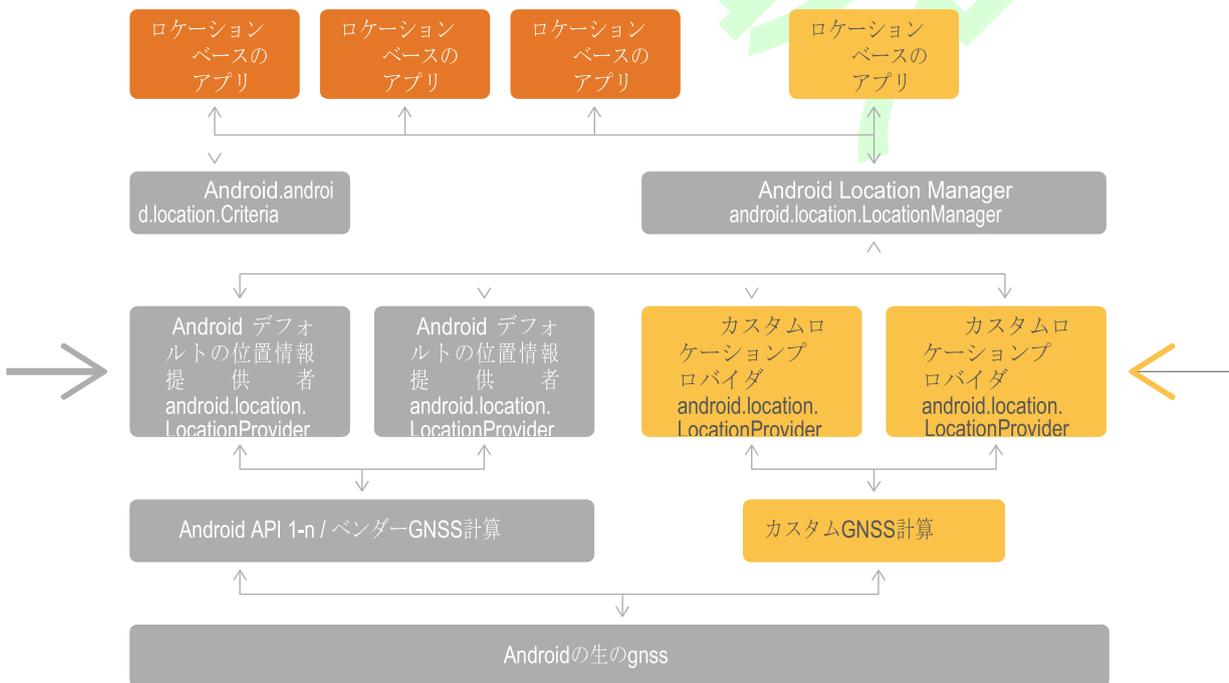


図13 アンドロイドGNSS生データアーキテクチャ

2.6 既存のアプリケーションとデバイス

少数のAndroidアプリケーションは、すでにGNSSの生測定値を利用しています。以下の各例は、何が可能であるかを示すためのデモンストレーションとして、また、更なる応用の出発点として使用することができます。

- **PPP Wizlite** (https://play.google.com/store/apps/details?id=jocs.fr.gnss_ppp)はCNES PPP-WizardプロジェクトのAndroidポートで、Nexus 5Xデバイス上のPPP位置を計算することができます。修正版をダウンロードするにはインターネット接続が必要です。
- **RTCM Converter** (https://play.google.com/store/apps/details?id=jocs.fr.gnss_tortcm)は、このようにして、あなたのAndroid携帯電話を基地局にして、広く使用されているRTCM形式にAndroidの生測定値を変換します。
- **Geo++ Rinex Logger** (<https://play.google.com/store/apps/details?id=de.geopp.rinexlogger>) は、Android Raw の測定値をRINEXファイルに記録し、測地ソフトウェアでのさらなる処理を可能にします。
- **エアバス社のGNSSデータコレクタ**は、衛星情報とPVTをリアルタイムで表示します。C/No、PVTに使用した衛星、精度などさらに、GPS、Galileo、GLONASSのRINEX観測ファイルをリアルタイムで計算して保存します。すべての生測定値を含むcsvファイルも保存されます。

生測定値を提供できる装置のリストは、[RD-25]で管理されている。情報には、星座の利用可能性や位相測定の利用可能性も含まれています。すべてのデバイスでAndroid 7以降を実行する必要があります。



3. GNSS生測定値の可能性と実用化

この章では、生測定値を提供するAndroidデバイスで利用可能な測位技術をまとめています。ハードウェアの制限と現在の性能を記述することから始まります。次に、図14および表9に要約されるように、最も有望な位置決め技術について説明する。

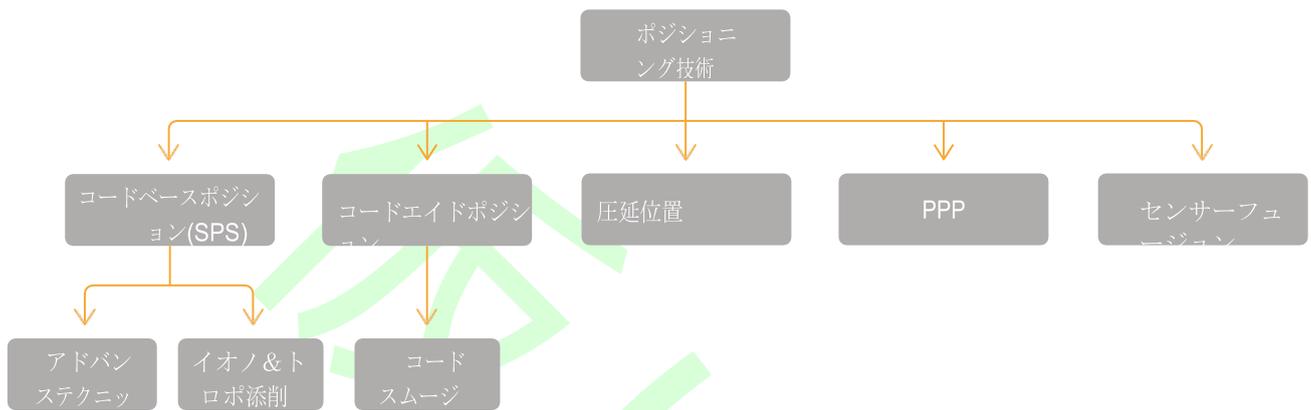


図14ポジショソニソグ技術

表9.ポジショソニソグ技術の比較

ポジショソニソグ技術	精度	空室状況	搬送波位相	外部情報	コメント
コードベースのシソグナルポイントポジシヨソニソグ	中型	高	否	オプシヨソ	局所的な大気モデルの実装が可能
コードエイド	中/高	中型	オプシヨソ	オプシヨソ	外部データは不要
差動	高	低	オプシヨソ	必要	複数のリファレンスステーション以内
PPP	高	低	必要	必要	長い収束時間が必要
センサーフュージョソ	高	高	オプシヨソ	必要	内部デバイスセンサー使える

GNSSモバイルチップセットのアーキテクチャは、TTFF（最初の固定までの時間）を秒単位で最小化し、位置の可用性と継続性を向上させることで、ユーザー体験を優先させています。最近有効になった生測定値は、**android.location** APIを介して利用可能で、図14にまとめた高度な測位技術を使用することができます。表9.この追加情報を使用して、誤差を減らすことができます（図8に概説）。また、生測定値は、携帯電話内部の他のセンサーとの統合をサポートし、追加の観測値（ドップラーやSNRなど）を使用することで、より優れた精度の推定、ナビゲーションの堅牢性、およびセキュリティの追加レイヤーを提供することができます。

3.1 モバイルA-GNSSチップセットの概要

携帯電話のデザインは、バッテリーの最適化とユーザーエクスペリエンスに焦点を当てており、ハードウェアコンポーネントの選択とソフトウェアアルゴリズムの両方に影響を与えます。GNSS/ナビゲーションチップセットは、他のGNSS機器で使用されているチップセットとは異なるA-GNSSアーキテクチャを採用しています。携帯電話は、ナビゲーションに最も重要なアンテナと発振器を内蔵した低コストのコンポーネントを展開しています。

発振器はシングルダイチップの外側に配置され、温度変化（TCXO）を補正することで周波数の安定性を向上させています。消費電力を抑えるためにGNSSチップを使用しているときだけオンになります。GNSSがオフの場合、代替の低消費電力水晶発振器（XO）が週6秒の精度低下で時刻を維持します[RD-10]。数日前から電源を切っていた機器は、起動時に時間の予測が粗くなります。

A-GNSSの主な目的は、通信機器（4GモデムやWiFi接続など）に内蔵されたアクセスを利用して、TTFFを削減することです。モバイルデバイスが取得します。

1. スマートフォンの位置の近似的な推定、例えば、Cell-ID測位を通じた位置推定、およびネットワークインフラストラクチャに依存した精度の高い初期時刻の推定。
2. 衛星エフェメリス、時計補正、電離層モデルを含むアシストデータ（アシストGNSSプロトコル）（GNSS衛星で放送されているものと同じ）[RD-10; RD-19]。拡張/代替手段として、予測GNSSエフェメライド[RD-20]や長期軌道プロダクトがあります。これは、アクティブなネットワーク接続がなくても、受信機が何日も何週間も先の未来の衛星の位置を計算できるようにします。
3. スマホの局部発振器を校正して基準周波数を向上させる。これは、周波数が一般的に50ppb以内であることが知られているセルタワーから受信した信号と、100ppb以内であることが知られている携帯電話キャリアの周波数によって校正されます[RD-10]。



図15アシストGNSS

この情報は、受信機の感度を高め、劣化した条件（都市部や明るい室内環境）でも、開始時にすべての情報が得られるため、数秒以内に最初の位置固定を提供します。また、エフェメリスデータを知ることで、過酷な環境下での位置固定の可用性と精度が向上します。デメリットとしては、スマホ内部の狭いスペースにごちゃごちゃとした隣り合う電波送信機同士が干渉して、同じチップ内でコロケーションしていることが多いため、GNSSチップのアーキテクチャが複雑になり、性能が悪化する可能性があることです。

携帯電話で使用されている代表的なアンテナは、安価なPIFA（Planar Inverted-F Antenna）で、通常はPCBピースを使用したマイクロストリップアンテナです。GPS/Galileo、GLONASS/Beidouなど、L1周波数帯のあらゆる星座に対応しています。しかし、その直線偏光（円偏光ではなく）と放射パターンの指向性により、数dBのシグナルロスが発生します。特に、指向性は、他の衛星からの信号を適切に受信しながら、いくつかの衛星を大きく減衰させることができます。アンテナの位置は、RFの制約よりもスマートフォンのデザインによって決定されます。これは、ユーザーの手との相互作用がGNSS信号の受信をさらに弱めるという、最適ではない場所での結果となる可能性がある。標準的なパッチアンテナとの相対損失は、非常に不規則な利得パターンで約11dBと推定される[RD-21; RD-22]。

この携帯電話はまた、消費電力を削減するためにデューティサイクル技術（2.4.3.1節で説明）を採用していますが、これは発振器に影響を与え、キャリア位相トラッキングの不連続性（例えば、2回の測定の間複数のサイクルスリップが発生した場合など）につながるため、リアルタイム・キネマティック（RTK）や精密点測位（PPP）のような高度な位相技術の使用が非常に困難になります。

ハードウェア設計の選択は、GNSS生データ、特に搬送波位相測定の高品質に影響を与え、その結果、測位性能に直接影響を与えます。マルチパスのレベルが上がり、フェージングの確率が高くなるため、ほとんどの携帯電話が使用されている都市環境では、この効果はさらに悪化します。

3.2 ベースラインパフォーマンス - コードポジショニング

前のセクションで説明したハードウェアの制限を可視化するために、移動中の車のダッシュボードにAndroidデバイスを置くという動的なシナリオを考えてみましょう。運動学的試験の結果は図16に示されており、太い黄色の線はタクティカルグレードのIMUによって提供される真実であり、青い線はデータ接続なしでGPSとGLONASS星座を使用してNexus 9タブレットから得られたAndroidソリューションを表している。緑の線は、マルチパス耐性のある測地アンテナを備えた低コストの単周波受信機を用いて得られたコードオンリーソリューションです。青線と緑線の違いの主な原因は、主に直線偏波アンテナであり、都市環境でのマルチパス除去が苦手なことが挙げられます（左側の画像）。晴れた田舎の環境（右側）では、マルチパスがなく、左側の建物からの迷走信号の影響を目に見えて受けている場合、Androidは低コストのGNSS受信機でも同等の性能を維持できます。



図16キネマティック試験結果

これらの結果はノッティンガム大学によって計算され、提供されています。

実証されたAndroidの性能は、車両のフロントガラスの後ろのダッシュボード上のデバイスの位置にも影響を受けています。図17は、PVTソリューションで使用する衛星の数の違いを、屋根上に設置されたアンテナと車内に設置されたアンテナとで示したものである。データは、高速道路と都市環境が混在する6時間のシナリオで収集された。ルーフソリューションは、特にデータ収集の開始時と終了時に発生するルート的高速道路区間では、ダッシュボードソリューションよりも優れた性能を発揮します。これは、主に車両構造物の妨害および受信信号の追加減衰（およびマルチパス）によるものである。興味深いことに、高速道路の橋や都市の峡谷など、限られた空のシナリオでは、スマートフォンは時折、より多くの衛星を見ることができます。これは、すでに記述されているアシスタンスデータの効果である可能性があり、より高速なTTFFと受信機レベルでの可能性のあるより高い感度につながる可能性があります。

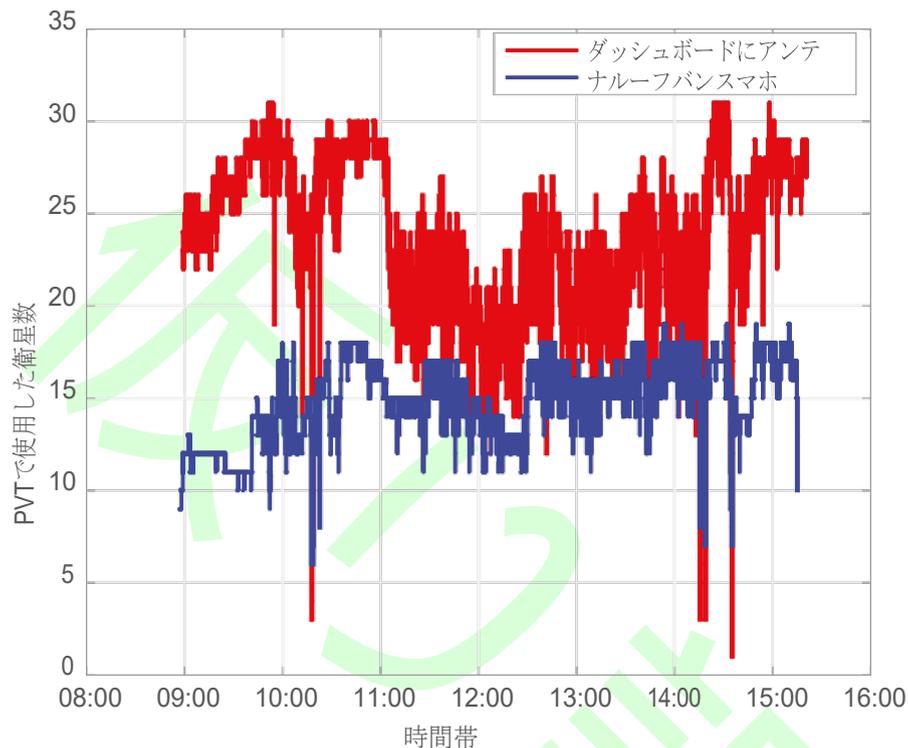


図17PVTで使った衛星数（GPS+GAL+GLO）は、バンのルーフにアンテナを接続し、バンのダッシュボードにスマートフォンを設置した量販店向けの受信機です。

これらの結果はESAが計算して提供しています。

3.3 ポジションの改善

3.3.1 多星座

ここでは、複数の星座を使用することのメリットを可視化するために、出発の高速道路区間に焦点を当ててみましょう。図18は、3つの星座（GPS, GLONASS, Galileo）を使用したPVTを用いて、最初にGPSのみを使用して計算したPVT解の3次元瞬時位置誤差（黒点）を示したものです。簡単のために、緑の点で精度の上昇（誤差の減少）を、赤の点で精度の低下を視覚化します。

最低限必要な衛星は4機ですが、信頼性の高い測位を得るためには、特にノイズや障害物の存在下では、8～10機の衛星が必要です。都市部では一般的な空の視界が限られている場合、これはマルチGNSSソリューションを使用してのみ達成することができます。これは、グラフの上部を覆う緑の円で可視化されています。

開放的な空の状況では、1つの星座から8～10個の衛星が得られる場合、その数を増やただけでは精度の向上にはつながりません。これは、グラフの下部にある赤丸のクラスターで示されるように、ノイズとマルチパス効果の蓄積によるものがほとんどです。この場合、全体の数を10以上に増やすよりも、PVTに最適な衛星を事前に選定することをお勧めします。また、システム間のISBを適切に見積もる必要があることにも注意が必要である。

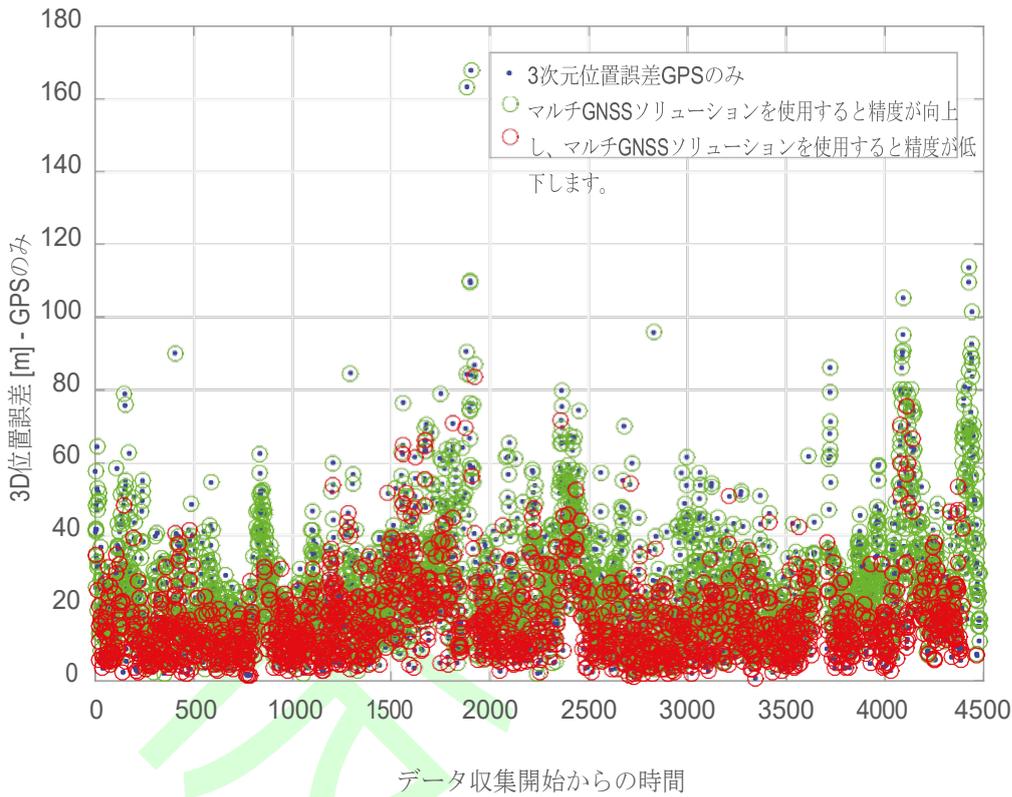


図18GPSのみとGPS+GAL+GLOでの3D位置誤差

これらの結果はESAが計算して提供しています。

表10は、すべての星座の組み合わせについて、位置の精度と利用可能性を数値化したもので、マルチGNSSの利点を示しています。すべてのデータは、オープンソースのRTKLibパッケージ[RD-36]で、kinematic Androidデータを使用して処理されています。

表10².PVTの精度と可用性におけるマルチコンスタレーションの影響

PVT構成	水平信頼度[メートル]		
	68%	95%	空室状況
GPS	13.36	25.51	97.79%
GPS + GAL	12.48	23.78	98.04%
GPS + GLO + GAL	11.24	21.57	98.30%
GPS + GLO + GAL + BEI	11.17	21.44	98.30%

これらの結果はESAが計算して提供しています。

3.3.2 チップセット内部の情報を利用する - コード観測値のドップラースムージング

静止位置精度を向上させるためには、ハッチフィルタを用いて符号観測を平滑化することができます[RD-23]。その代わりに、デューティサイクルに強いドップラー測定を利用することができます。図19は、コードのみとドップラースムージングを用いた生の測定データから得られた静的PVTを、異なる積分時間で比較したものである。20秒のスムージングウィンドウを使用することで、3Dエラー（信頼度68%）は19.7メートルから7.6メートルへとほぼ3分の1に減少しました。

2 ガリレオも北斗もまだフル運用可能な状態ではないので、完全に配備されれば、より良い性能が期待できます。

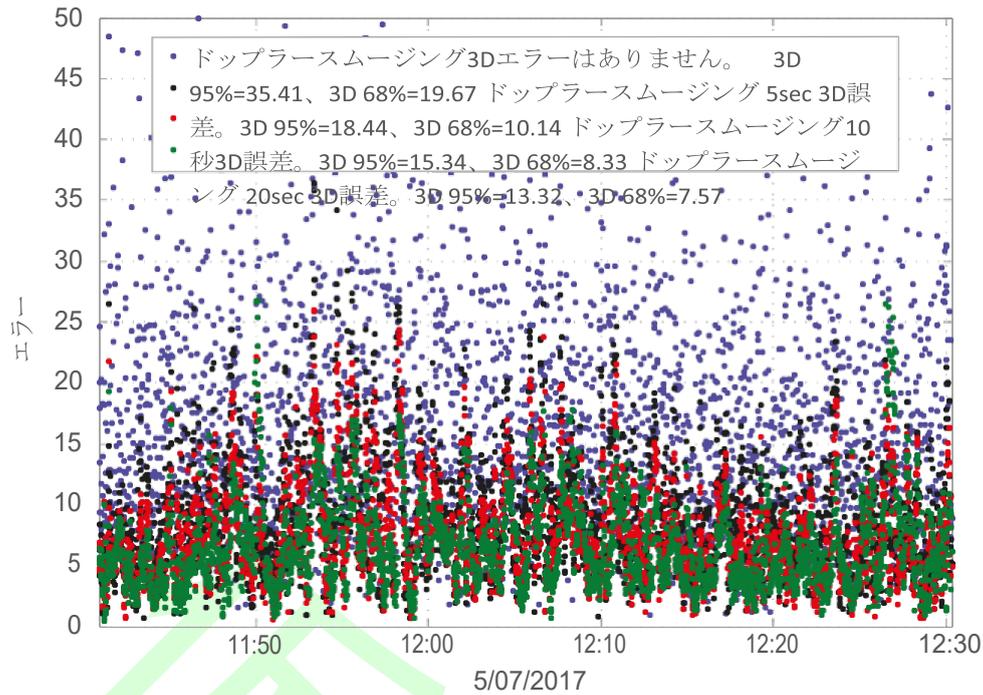


図19GPS + GLO + GAL PVTの性能比較 - ドップラースムーミング

これらの結果はESAが計算して提供しています。

この方法を運動学的なシナリオで評価するために、静的なシナリオで得られた良好な結果に基づいて選択された20秒の平滑化ウィンドウを持つ高速道路シナリオをもう一度考えてみましょう。スムージングを行うことで、静的なシナリオよりも小さいながらも精度が向上し、非常に動的な高速道路のシナリオでも、はるかに滑らかなパスを実現しました（図20）。基準軌道（緑）は、ハイエンドのGNSS受信機を用いて計算したものです。



図20PVTキネマティックソリューション - 生コード（左）アドップラー平滑化（右）測定値

表112 PVT運動性能の比較 - ドップラースムージング

マルチGNSS構成		水平信頼度[メートル]		
		68%	95%	空室状況
ドップラースムージング 20	PVT GPS	9.38	18.87	97.79%
	PVT GPS + GAL	9.21	18.63	98.04%
	PVT GPS + GLO + GAL	7.72	16.65	98.30%
	PVT GPS + GLO + GAL + BEI	7.69	16.53	98.30%
ドップラースムージング なし	PVT GPS	13.36	25.51	97.79%
	PVT GPS + GAL	12.48	23.78	98.04%
	PVT GPS + GLO + GAL	11.24	21.57	98.30%
	PVT GPS + GLO + GAL + BEI	11.17	21.44	98.30%

これらの結果はESAが計算して提供しています。

3.4 電話の向こう側に持っていく - 差分観測

位置決めに成功するための最低条件は、1つのコンステレーションから4つのGNSS衛星です（位置と内部クロックオフセットを解決する必要があるため）。主なGNSSの誤差源は、衛星ベース（軌道と時計）、大気（電離層と対流圏の遅延）、ノイズ、局所的な影響（マルチパスを含む）などですが、詳細は図8を参照してください。差動観測は、その誤差を外部から補正することで位置精度を向上させますが、通常、マルチパスや干渉などの局所的な影響を補正することはできません。これにはいくつかの技術があり、最も一般的に使用されているのは、コードソリューションのためのDGNSS（またはDGPS、差分GNSS/GPS）であり、キャリアフェーズソリューションのためのリアルタイムキネマティクス（RTK）、ネットワークRTK、および高精度ポイントポジショニング（PPP）です。RTKやPPPで搬送波位相を使用すると、最も正確な結果を得ることができます。どちらも品質の良い搬送波位相と、理想的には、曖昧さを解消して精度を高めるための静的収束時間を必要とします。コンバージェンス時間はPPPの方がかなり長いですが、データ要件はより柔軟である。

添削は無線やインターネットを介して行われますが、完全性の情報は提供されません。その代わりに、EGNOSのような大規模なSBASサービスが提供しており、ヨーロッパやアフリカ向けのクローブチャール型電離層補正モデルも提供しています。このモデルは、例えば、夏時間の静かな宇宙気象条件のためのアドリア海北部の特注電離層補正モデル[RD-24]のような、より局地的なモデルによって改良することができる。

表12と図21は、PPPアルゴリズム（青）とデバイスベースのコードベースの測位ソリューション（赤）を使用して静的な生の観測値を処理することの違いを示しています。基準となる位置は、黒い十字で示されています。

2 ガリレオ、北斗ともにまだフル運用可能な状態ではないため、本格的に運用が開始された場合には、より良い性能が期待されます。

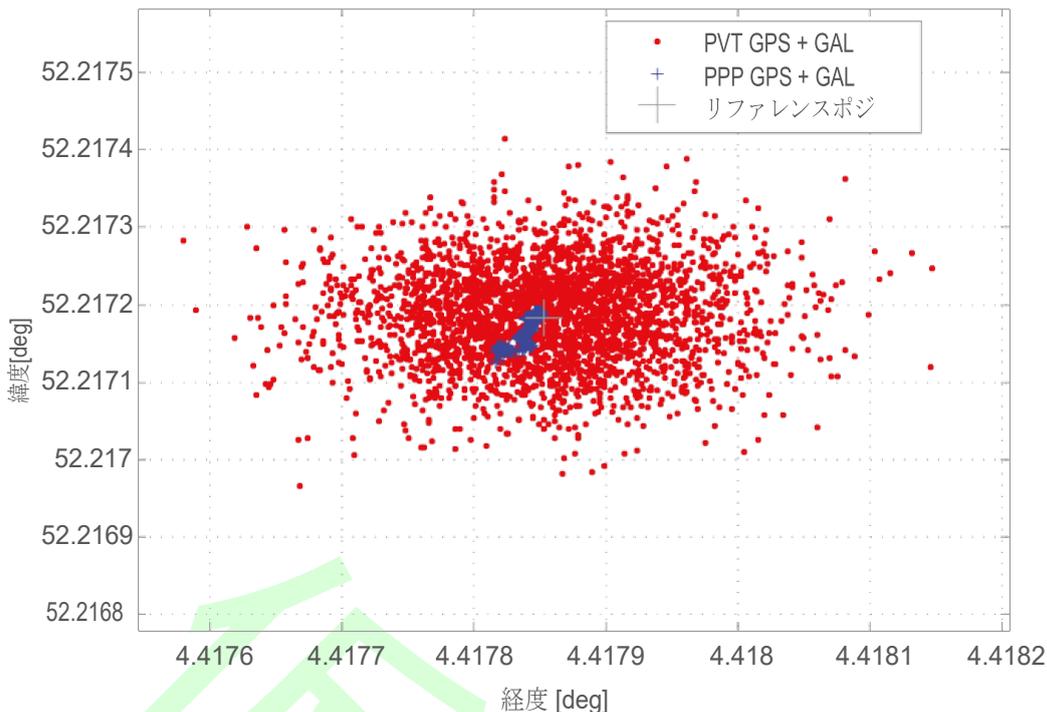


図21 PPP対PVTソリューション

これらの結果はESAが計算して提供しています。

表 122. ユーザー精度におけるPPPの影響 [m]

PPP対PVT構成	静的シナリオ				
	H68%	H95%	V68%	V95%	空室状況
PVT GPS + GAL	9.18	15.43	17.43	34.67	99.70%
PPP GPS + GAL	0.90	2.35	0.62	1.43	99.70%

これらの結果はESAが計算して提供しています。

重要な改善点は、L1/E1の上にデュアル周波数のチップセットを導入することで、マルチパスにはるかに耐性のあるGPS L5とGalileo E5信号を提供することになります。初期試験では性能が向上し、サイクルスリップの検出と修正[[RD-32]]が可能になった。

3.4.1 デューティサイクル

消費電力を削減するために携帯電話で展開されるデューティサイクルは、発振器に影響を与え、キャリアの位相追従性の不連続性を導入します。図 22 は、アクティブサイクルなし（機能はオフ）で収集した静的データを示しています。90%以上の位相測定が有効で、RTKまたはPPPアルゴリズムの使用が可能です。3つの衛星(5,21,27)ではサイクルスリップが異常に多く見られますが、これは標高の低さとノイズの組み合わせによるものと考えられます。これらは計算から削除されています。

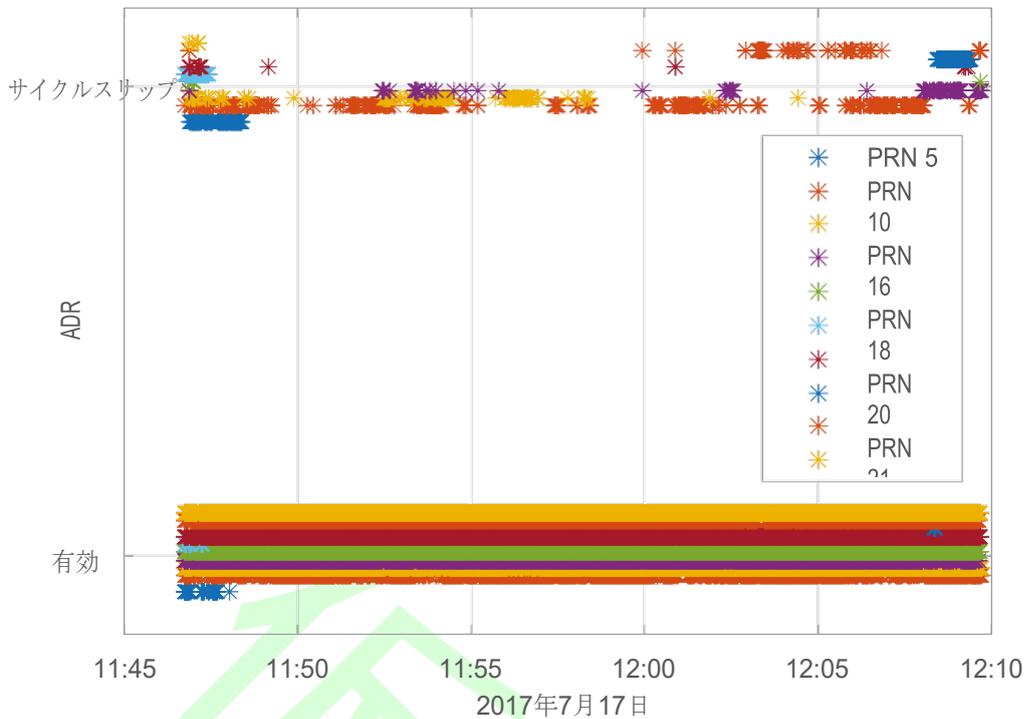


図22累積デルタレンジ有効フラグ：デューティサイクル無効

これらの結果は、Airbus Defence and Space GmbHによって計算され提供されています。

表 13.衛星ごとの位相有効性（デューティサイクルなし）

衛星	フェーズの有効性 [%]	サイクルスリップ [%]
5日(土)	26.73	73.26
10日(土)	78.75	21.24
16日(土)	92.96	7.03
18日(土)	94.92	5.07
20日(土)	99.63	0.36
21日(土)	13.95	86.04
26日(土)	98.98	1.01
27日(土)	7.69	92.30
29日(土)	96.51	3.48
土31日	99.63	0.36
GPS集計	92.86	7.13

これらの結果は、Airbus Defence and Space GmbHによって計算され提供されています。

アクティブデューティサイクルを持つスマートフォンの結果は、使用可能なデータの60%しかなく（表14）、RTKやPPPでの使用は非常に限られています。この比較から、デューティサイクルを無効にすることで性能が向上することがわかり、消費電力の増加を犠牲にしても、高精度なアプリケーションには推奨されることがわかります。

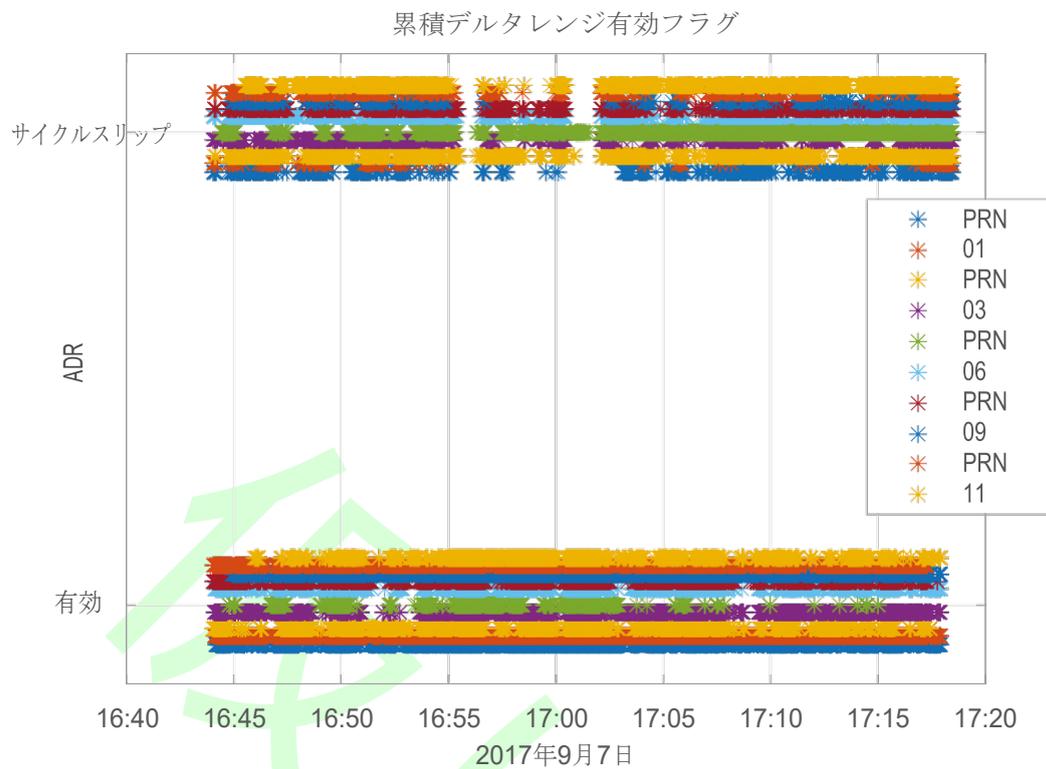


図23累積デルタレンジ有効フラグ：デューティサイクル有効

これらの結果は、Airbus Defence and Space GmbHによって計算され提供されています。

表 14.衛星ごとの位相の妥当性

衛星	フェーズの有効性 [%]	サイクルスリップ [%]
1日(土)	64.05	35.94
3日(土)	85.29	14.70
6日(土)	49.72	50.27
9日(土)	58.81	41.18
11日(土)	32.45	67.54
17日(土)	44.26	55.73
19日(土)	51.76	48.23
22日(土)	76.47	23.52
±33	85.34	14.76
±31日	46.23	53.76
GPS集計	59.44	40.55

3.4.2 センサーフュージョン

Googleの`android.gsm.position`のフェーズドロケーションプロバイダは、GNSS、ネットワークロケーション、加速度計、ジャイロスコープ、バロメーター、地磁気計を使用して位置を提供しています。を経由してGNSSにアクセスすることとは別に `AndroidPosition` を介して、以下の位置センサーにアクセスすることができます。 `AndroidSensorRD-35` です。

- GNSSチップセット
- WiFi
- ブルートゥース低エネルギー (BLE)
- ネットワークロケーション (NLP)
- 慣性計測ユニット (IMU) - 低コストMEMS加速度計、ジャイロスコープ、地磁気計
- 圧力・光センサー
- ステップカウンタ (機械式)

例として、加速度計、ジャイロスコープ、磁力計のIMUの融合を実演してみましょう。この組み合わせは、IMUの短期的な精度と高い更新率のメリットがあり、GNSS測位によって慣性ドリフトが解消されます。GNSSとIMUの間のセンサーフュージョンには、大きく分けて2つのアーキテクチャがあります。

1. GNSSの位置と速度を利用したゆるやかなアプローチ。これはAndroid API内のフュージョンされたポジショニングによって展開されています。
2. Android 7の生測定値と時間同期が可能にした密着型アプローチ。これにより、結合された位置に必要な衛星数が4機以下となり、追加情報の恩恵を受けることができるため、位置の可用性と連続性を向上させることができます。

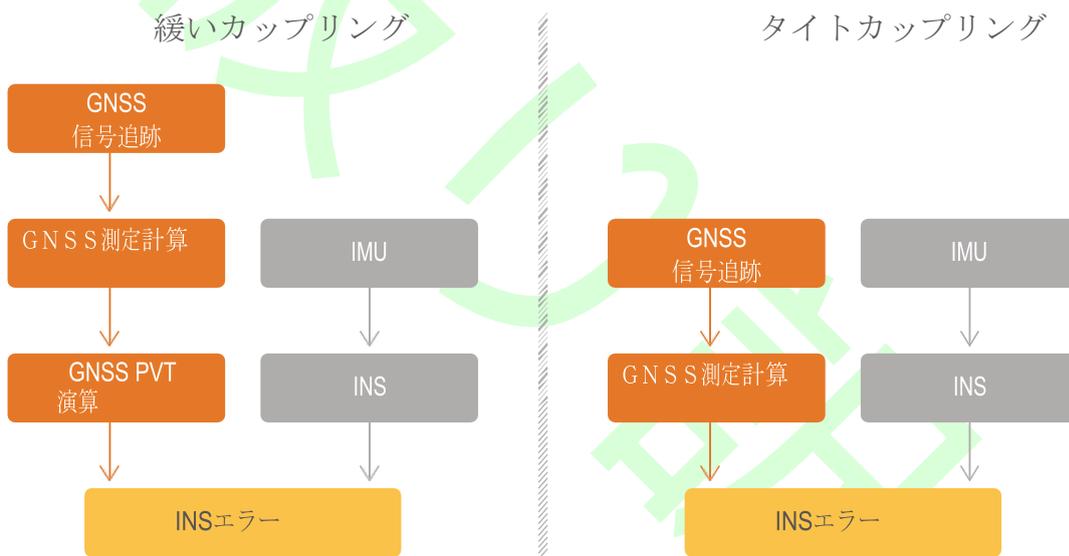


図24 センサーフュージョンアーキテクチャ

3.5 教育と科学の応用

生測定値は、科学的な観点からも非常に刺激的です。Android端末は、ハードウェアコストが合理的に低く、専用のファームウェア開発が不要であることから、センサーのグリッドとして展開することができます。ここでの科学的な使用法の一つは、1つの周波数のみを使用した詳細な大気モニタリングの提供である。その他のアプリケーションと用途としては、基地局の配備、干渉検出、スマートシティのセンサーの一部としての使用などがあります。

第二に、観測ははるかに粗い形式で提供されているように、Androidデバイスは、優れた教育ツールとして、1つは、詳細には、GNSSの測定値がどのように得られ、計算されるかを理解することができます。観測はまた、単一の星座からのソリューションを比較するために使用することができます (例えば、図25のガリレオのみの測位)、特定の衛星を排除したり、最悪のシナリオのパフォーマンスを視覚化します。教育とは別に、ハードウェアとソフトウェアのソリューションをテストし、性能レベルを観察し、チップセット自体が提供するベンチマークソリューションとこの性能を比較するために、同じアプローチを使用することもできます。

例として、Googleが提供するGNSS解析ツール[RD-18]があります。ログに記録された生測定値（静的および運動学的）を使用して重み付き最小二乗SPS PVT解を計算し、既知の基準値と比較することができます。このソフトウェアは、信号（SNR、スカイプロット）、クロック（衛星、受信機、デューティサイクル）、測定値（位置精度、擬似レンジ、キャリア残差）を可視化し、システムのダッシュボードの概要をユーザーに提供します。

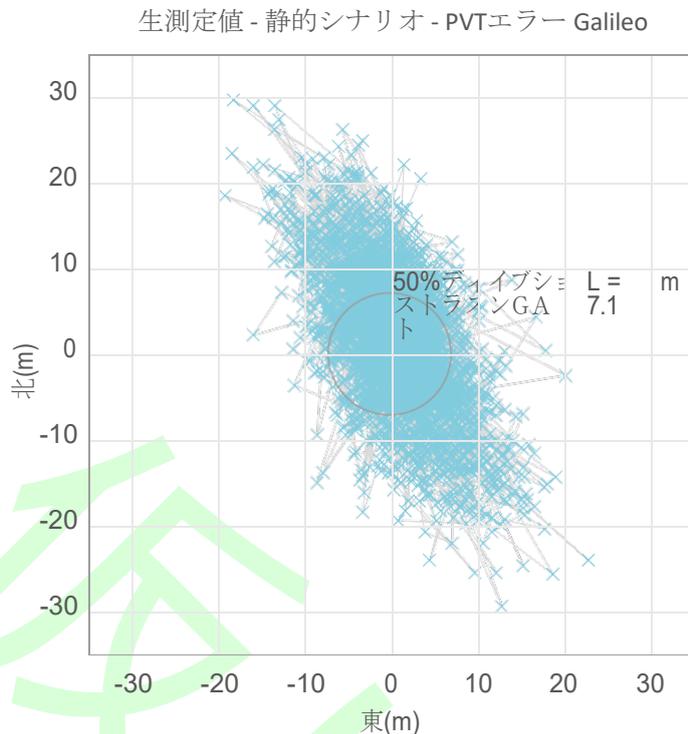


図 2⁵²を参照してください。静的ガリレオのみの位置誤差

これらの結果は、Airbus Defence and Space GmbHが計算し、提供しています。

3.6 ハイインテグリティソリューション

GNSS信号内のRF干渉の存在を検出するための低コスト能力への関心が高まっています。これは主に、劣悪なGNSS性能を調査・診断し、サービスインシデントの拒否を解決するために必要とされています。携帯電話のグリッドは、GNSS周波数内の混乱の「クラウドソース」画像を提供する可能性があります。個々の擬似レンジやC/No、AGC値などの形で生測定値にアクセスすることで、デバイス自体を使用してRF干渉を検出する新しい方法を生み出す機会を提供します。さらに、地域内の複数のAndroid端末の測定データを組み合わせることで、干渉源を特定できる可能性もあります。現在のところ、これはニッチなアプリケーションかもしれませんが、このような機能にアクセスすることで、GNSSユーザー、サービスプロバイダー、インフラ事業者、GNSSスペクトルを保護するための国家周波数当局を支援するための新しいサービスを開発する機会が生まれます。

RAIMは、GNSS信号の完全性をアルゴリズムで評価し、性能が設定されたしきい値を下回った場合にユーザーに警告を発し、別の解決策が必要な場合には、ナビゲーション性能の信頼性レベルの評価を提供します。これらは、航空、海上ナビゲーション、最近では鉄道業界など、安全性が要求されるGNSSアプリケーションを対象としています。小さな変更で、Androidプラットフォームに拡張でき、ユーザーにより良い位置の信頼性、追加の検証と警告を提供します。

これは、航空、海洋、鉄道などの安全上重要なアプリケーションで利用されているモバイルデバイスや、位置情報の取得が命に関わる場合（救急車ユニットの派遣など）に役立つでしょう。このような検証は、オンライン取引にセキュリティの別のレイヤーを提供することもでき、コミュニケーションをより明確にし、人為的なエラーを減らすことができます。これは、スカイビューなどの他の情報と組み合わせて、GNSSの性能が低下する可能性のあるエリアを予測するシステムに拡張することができます[RD-34]。RAIMはまた、ユーザが生成した地理データが位置の信頼性と収集した情報の品質管理の両方を必要とするボランティア地理情報(VGI)でも使用することができる[RD-33]。また、同じ原理を利用して、報告されたAndroidの位置が特定のタスクの精度要件を満たしていない可能性がある場合に、ユーザーに視覚的な警告を提供し、ユーザーが位置情報をより教育的に利用できるようにすることもできます。

² ガリレオ、北斗ともにまだフル運用可能な状態ではないため、本格的に運用が開始された場合には、より良い性能が期待されます。



4. マスマーケットにおける高精度化の必要性と使用例

現在、世界中で50億台以上のGNSS機器が使用されており、そのうち8割がスマートフォンです。Google PlayとAppleストアでは、50%以上のアプリが位置情報を利用しています。スマートフォンは、グランドやレジャーボートのナビゲーションから、都市公園管理のための樹木のマッピングまで、安全性を重視したものや高精度なものの境界線上にあるアプリケーションでの利用が増えています。マスマーケットで得られる位置精度の向上により、スマートフォンやウェアラブルのセミプロ用途での利用がさらに増え、現在ではまだ不可能な消費者用途の新たな範囲が可能になると考えられます。

第3章で実証されているように、今日のマスマーケットのモバイルデバイスの典型的な性能は、都市部の峡谷のような困難な条件では、数メートルから数十メートルの範囲にあります。しかし、マルチコンステレーション、デュアル周波数チップセットの使用、外部情報の提供により、近い将来、この精度をサブメートルレベルにまで高めることが約束されています。チップ上にシステムを展開している現代の携帯電話は、20年前のメインフレームコンピュータよりも高性能です。その結果、新しいアプリケーションのデプロイとテストにかかるコストを削減し、更新サイクルを短縮し、ハードウェア開発の負担を軽減することができます。Androidの生測定値は、整合性とロバストな位置の追加レイヤーを提供し、ロバストで信頼性が高く、干渉に強い位置ベースのサービスの開発を可能にします。

4.1 位置精度向上の恩恵を受ける主な用途

多くのマスマーケットアプリケーションは、精度向上の恩恵を受けることができます。例のリストは、図26を参照してください。



図26高い位置精度を活用したアプリケーション

4.1.1 モバイルアプリケーション

ロケーションベースの広告では、位置データがクリーミーな広告をカスタマイズするために使用されている場合、エンゲージメント率は最大30%に増加することができます[RD-26]。特定の企業の近隣にいるモバイルユーザーをターゲットにしている場合、位置が不正確だとキャンペーンが効果的ではないと判断されます。広告パフォーマンスを向上させるために位置情報の精度を追跡したモバイルマーケティング協会の2014年の調査では、位置情報ベースの広告リクエストの"No_2153"のみが、記載された場所の50~100メートル以内に正確であったことがわかりました。

同様の状況は、タクシー会社が車を特定のゾーンに限定したり、都市が中心部への有料アクセスを定義したり、親が子供を追跡したりするなど、すべてのジオフェンシングアプリケーションに適用されます。ジオフェンシングアプリケーションでは、絶対的な位置精度は重要ではありません。重要なのは、仮想フェンスの内外でユーザーの位置を特定できることです。

ここでは、ジャミングやスプーフィングなどの意図的な干渉が問題になります - 特に、ジオフェンシングアプリケーションが責任や支払い（位置情報に基づく駐車場など）と並んでいる場合。第3章で説明したように、生測定値の使用は、完全性チェックとなりすましの検出をサポートすることもできます。

拡張現実（AR）アプリケーションでは、位置精度も重要です。拡張現実（Augmented Reality）は、コンピュータが生成した画像を現実世界のユーザーの視界に重ねて表示することで、現実の知覚を向上させるものである。3次元での位置情報は、物理世界とデジタル世界を同期させるための鍵となりますが、精度は最終的なアプリケーションに依存します。ARアプリケーションの中には、非常に高度な同期を必要とするものがあり、より高い位置精度が必要とされます。世界の拡張現実市場は2022年までに1,080億ユーロに達すると予測されている[RD-01]。

拡張現実ツールには、拡張現実3Dビューア、拡張現実ブラウザ、没入型ゲーム体験の3つのカテゴリーがあります。第1のカテゴリーの例としては、建設業での例が挙げられる。業界では、モバイルデバイスを使って、建設現場、地下構造物、ケーブル、パイプなどのジオリファレンスモデルを視覚化するために、拡張現実感を利用する方向に進んでいます。ここでは、高い位置精度が重要な要件となります。例えば、ARシステムで地中のパイプを「見る」ことができ、そのパイプの位置が十分に正確でない場合、業者が掘り始めるときにパイプにぶつかってしまう可能性があります。建設会社は、施工管理ソフトであれ、新しい拡張現実デバイスであれ、使用しているツールに頼らなければなりません。

4.1.2 安全性関連のアプリケーション

コンシューマ機器が利用されている安全関連のアプリケーション領域の例としては、モバイルヘルス（mhealth）があります。モバイルヘルスは、通信やデータ収集など、モバイルデバイスによってサポートされる医療や公衆衛生の実践をカバーしています。このホワイトペーパーの目的のために、GNSSベースのウェアラブルを使用したすべてのウェルビーイングとフィットネスのアプリケーションも含まれています。位置精度の向上は、例えば、スポーツやフィットネスのトラッキングアプリケーション、視覚障害者や身体障害者のナビゲーション、アルツハイマー患者のためのジオフェンシングなどに役立つだろう。このカテゴリーの応用範囲はすでに非常に広いです。高齢化が進む中で、特にソーシャル・インクルージョンを中心としたイノベーションのための資金調達が可能となり、申請件数も急増しています。

位置精度の向上により、漕ぎ、ゴルフ、陸上、サッカートレーニング、スキーなどのスポーツにおいて、努力量、距離、移動速度の推定が向上し、個人のパフォーマンスを把握して改善につなげることができます。ハイエンドのGNSS受信機は、すでに様々なスポーツのプロスポーツ選手に広く利用されています。Androidの精度が向上すると、パフォーマンスを分析するための費用対効果の高い代替手段として使用される低コストのデバイスの台頭を見ることができます[RD-27]。

GNSS技術は、聴覚と触覚のインターフェースを介して、視覚障害者が周囲の環境をよりよく理解し、移動をサポートすることを可能にします。ユーザーが特定の目的地や興味のある場所を特定できるようにするためには、正確さが必要とされます。

GNSS対応のe112緊急通報は、ヨーロッパの世界共通緊急番号112の位置情報を強化したバージョンで、位置情報が自動的に通報と同時に緊急センターに提供されます。EENA [RD-30]によると、ヨーロッパの緊急電話の70~80%は携帯電話から発信されているが、緊急サービスに提供される発信者の位置情報は不正確で遅れていることが多い。現在、Cell-IDは最も広く使用されているソリューションです。GNSSを使用することで、正確な測位が可能になり、より良いアクセスとリソースの推定と割り当てが可能になるため、応答時間を短縮することができます。医学的な研究では、応答時間を1分短縮すると生存確率が24%向上することが示されています[RD-30]。

4.1.3 セミプロ向けアプリケーション

IoT (Internet of Things) では、物理的な機器や車両、建物などを相互に接続して遠隔操作することで、独自のインフラ (メッシュネットワーク) を構築することができます。IoTはネットワークインフラ全体で急速に成長し、相互に接続されたセンサーと技術のメッシュを形成し、GNSSが定位とタイミングを提供することが期待されています。それは、LBSの主要なイネーブラーとなり、携帯電話からのデータでは、ビッグデータの重要な情報源となります。これにより、非常にパーソナルなコンピュータで生成されたサービスを作成することができ、データに基づいたモデルを作成することで位置精度を向上させることができます。スマート照明やゴミ管理のようなスマートシティの資産管理は、精度向上の恩恵を受ける可能性のあるアプリケーションの一例に過ぎません。

クラウドソーシングの概念は、複数の情報源から情報を収集するという同じ原理に従っています。このアプローチは、生測定値によって提供される余分なセキュリティと完全性から利益を得ることができます。

これは一般の人々に迅速な情報収集と地図作成を提供するが、プライバシーが問題となる[RD-29]。ローカリゼーション情報は非常に機密性が高く、不当に悪用される可能性があり、個人と企業の両方にリスクをもたら

します。欧州連合 (EU) は、一般データ保護規則 (GDPR) を導入しました。これは、電子通信における個人データの利用やデータ保持など、欧州全体のデータプライバシー法を調和させる法的枠組みです[RD-31]。IoTとビッグデータの急速な発展には、制度的・技術的な統制を特定するだけでなく、関連する機会と課題について

の一般的な理解を作るための更なる議論が必要である[RD-30]。

精度向上の恩恵を受ける可能性のあるアプリケーションには、すべての労働力と艦隊管理アプリ、特に緊急通知の場合に孤独な労働者を保護するアプリが含まれています。クラウドソーシングされた地図アプリは、その目的によっては、サブメートルの精度も恩恵を受けることができます。さらに、スマートフォンレベルで利用可能なより高いレベルの精度は、メートル単位の精度で十分な場合には、デバイスをあまり要求されないマッピングタスクに使用することも可能になります。

ホワイトペーパーを読んで楽しかった？ 接触する！

コメントやフィードバックは market@gsa.europa.eu までお願いします。

GSA GNSS生測定値タスクフォースに参加しませんか？ 上記と同じ連絡先を使用してください。

図のリスト

図1 :	交差する球体 [RD-03]	7
図2 :	GNSS受信機の一般的なブロック図	8
図3 :	基準時間の時間差	10
図4 :	C/Aコードとの同期	11
図5 :	GPS L1 C/Aコードとナビゲーションメッセージの構造	11
図6 :	E1Bガリレオ部品 (左側) 、 E1Cガリレオ部品 (右側)	12
図7 :	共通受信時間に基づく擬似レンジ計算	13
図8 :	GNSS擬似測距誤差の発生源	15
図9 :	Android APIレベル23の位置情報API	17
図10 :	Androidの位置情報APIレベル24/25/26	18
図11 :	デューティサイクル対時間	23
図12 :	Androidプログラムの位置情報 (Android 7以前) vs. Androidロケーション (Android 7以降)	25
図13 :	Android GNSS生データアーキテクチャ	25
図14 :	ポジショニング技術	27
図15 :	アシストGNSS	28
図16 :	キネマティック試験結果	29
図17 :	PVT(GPS+GAL+GLO)で使用されている衛星数は、量販店向け受信機のためのものです。バンの屋根のアンテナに接続し、スマートフォンを取り付けた。バンのダッシュボード	30
図18 :	GPSのみとGPS+GAL+GLOでの3D位置誤差	31
図19 :	GPS + GLO + GAL PVTの性能比較 - ドップラースムージング	32
図20 :	PVT運動学的解 - 生のコード (左) とドップラー平滑化した測定値 (右)。	32
図21 :	PPP対PVTソリューション	34
図22 :	累積デルタレンジ有効フラグ : デューティサイクル無効	35
図23 :	累積デルタレンジ有効フラグ : デューティサイクル有効	36
図24 :	センサフュージョンアーキテクチャ	37
図25 :	静的ガリレオのみの位置誤差	38
図26 :	高い位置精度を活用したアプリケーション	39

表のリスト

表1.	GNSS基準時間	9
表2.	リファレンスタイムの関係	10
表3.	追跡状態に基づく時間の曖昧さ	12
表4.	AndroidのバージョンとAPIレベル	16
表5.	Android 7 位置情報 API - 時計と計測フィールド	18
表 6.	Android 7 Location API - ナビゲーションメッセージフィールド	19
表7.	AccumulatedDeltaRangeState	23
表 8.	ConstellationType	24
表9.	ポジショニング技術の比較	27
表10.	PVTの精度と可用性におけるマルチコンスタレーションの影響	31
表 11.	PVT運動性能の比較 - ドップラースムージング	33
表 12.	PPPがユーザーの精度に与える影響 [m]	34
表 13.	衛星ごとの位相有効性 (デューティサイクルなし)	35
表 14.	衛星ごとの位相の妥当性	36

略語

API	アプリケーションプログラミングインタフェース
AR	拡張現実
C/Aコード	粗取得コード
C/No	搬送波対雑音密度比
F/NAV	自由にアクセスできるナビゲーションメッセージ
FCN	周波数チャンネル数
GGTO	GPSからガリレオへの時間オフセット
GLONASST	グローナスの時間
GNSS	Global Navigation Satellite System
GPST	GPS時刻
GSA	European GNSS Agency
GST	ガリレオシステムの時間
IMU	慣性測定単位
I/NAV	インテグリティナビメッセージ
ISB	システム間バイアス
IoT	モノのインターネット
LBS	位置情報サービス
LNA	ローノイズアンプ
NMEA	ナショナルマリンエレクトロニクス協会
OS	オープンサービス
OSNMA	オープンサービスナビメッセージ認証
PIFA	平面倒立F型アンテナ
PPP	精密なポイント位置決め
PVT	位置速度と時間
レイム	受信機自律整合性監視
RINEX	受信側独立交換フォーマット
RTK	リアルタイムキネマティック
ソード	二日目
SPS	標準ポジショニングサービス
TAI	Temps Atomique International
TCXO	温度補償型水晶発振器
TOW	曜日の時間帯
TTFF	時間宛先ファーストフィックス
UT	ユニバーサルタイム
UTC	協定世界時
WN	週数
XO	水晶発振器

参考文献

[RD-01] GNSS Market Report, Issue 5, GSA.
https://www.gsa.europa.eu/system/files/reports/gnss_mr_2017.pdf

[RD-02] 現在、計画されている全球・地域航法衛星システムと人工衛星を利用した増強システムUN O O S Aオーストリア、ウィーン。

[RD-03] पीターJ.G.Teunissen and Oliver Montenbruck, Springer Handbook of Global Navigation Satellite Systems, Springer (2017)

[RD-04] [http://www.navipedia.net/index.php/Code_Based_Positioning_\(SPS\)](http://www.navipedia.net/index.php/Code_Based_Positioning_(SPS))

[RD-05] Sanz Subirana, J et al.(2013)."GNSSデータ処理 - Vol.I.Fundamentals and Algorithms".欧州宇宙機関 (ESA)。ノルトウェイク (オランダ)。で入手可能です。
<http://bit.ly/1QV4KAL>。2018年2月5日にアクセス

[RD-06] ペトロフスキーイワン・G(2014)."GPS、GLONASS、Galileo、Beidou for Mobile Devices"ケンブリッジ大学出版局。英国ケンブリッジ

[RD-07] リンティ、ニコラ&ロープレスティ、レティツィア&ドヴィス、ファビオ&クロスタ、パオロ。(2014).GNSSマスマーケット受信機におけるデューティサイクル省電力技術の性能解析記録 - IEEE PLANS, 位置情報とナビゲーションシンポジウムを開催しました。1096-1104. 10.1109/PLANS.2014.6851479.

[RD-08] http://www.navipedia.net/index.php/Time_References

[RD-09] インタフェース仕様書 IS-GPS-200 2013年9月号で利用可能です。 <http://www.gps.gov/technical/icwg/IS-GPS-200H.pdf>

[RD-10] ヴァン・ディグレンフランク・スティーブン・トロンプ2009.A-GPS : アシストGPS、GNSS、SBAS。ボストンダアーテックハウス。

[RD-11] コードトラッキングと擬似レンジ、インサイドGNSS、2012年1月/2月で利用可能です。 http://www.insidegnss.com/auto/IGM_janfeb12-Solutions.pdf

[RD-12] Gustafsson, F.(2010)."統計的センサフュージョン"Studentlitteratur.リンコエピン大学。リンコエピン、スウェーデン

[RD-13] Rahemi, N et al.(2014)."Accurate Solution of Navigation Equations in GPS Receivers for Very High Velocities Using Pseudorange Measurements".Adv in Aerospace Eng, 2014, Article ID 435891, 8 pages

[RD-14] ブラウン、R・G・I・ホワン、P・W・C。(2009)."ランダム信号と応用カルマンフィルタリング入門 (第3版)".
 ジョン・ワイリー&サンズニューヨーク

[RD-16] ペトロフスキーイワン・G(2014)."GPS、GLONASS、Galileo、Beidou for Mobile Devices"ケンブリッジ大学出版局。英国ケンブリッジ

[RD-17] GNSSタイムオフセット、インサイドGNSS、2007年9月/10月
 available at: http://www.insidegnss.com/auto/SepOct07-GNSS_time_offset.pdf

[RD-18] GPS測定ツール、Google
 で利用可能です。 <https://github.com/google/gps-measurement-tools>

[RD-19] Nicolas Couronneau (2013) Performance analysis of assisted-GNSS receivers, PhD.
 ケンブリッジ大学

[RD-20] <http://rxnetworks.com/location-io/predicted--gnss-ephemeris/>

[RD-21] Pesyna, K. M.; Heath, R. W. & Humphreys, T. E.スマートフォン品質のGNSSアンテナを用いたセンチメートル測位 2014 ION GNSS Conference Proceedings of 2014, 2014

- [RD-22] Humphreys, T. E.; Murrian, M.; van Diggelen, F.; Podshivalov, S. & Kenneth M. Pesyna, J. On Feasibility of cm-Accurate Positioning via a Smartphone's Antenna and GNSS Chip Proceedings of 2016 IEEE/ION PLANS Conference, 2016
- [RD-23] http://www.navigopedia.net/index.php/Carrier-smoothing_of_code_pseudoranges
- [RD-24] Brcic, D.(2015).衛星測位信号電離層遅延の非特異的日周パターンモデル (博士論文) .リエカ大学海洋学部。リエカ、クロアチア
- [RD-25] <https://developer.android.com/guide/topics/sensors/gnss.html>
- [RD-26] ロケーションスコアインデックス。2016年第2四半期、モバイル広告の位置精度ガイド、Thinknear.
- [RD-27] <https://www.peakendurancesport.com/endurance-products-and-technology/electronics-and-software/sports-equipment-gps-best-route-performance-analysis/>
- [RD-28] <http://gpsworld.com/nvs-technologies-selected-by-alberding-for-sub-meter-gnss-receiver/>
- [RD-29] ギラド・ロズナー『プライバシーとモノのインターネット』
(2017 年) 、 オ ラ イ リ ー [RD-30]
- <http://emj.bmj.com/content/early/2010/08/25/emj.2009.086363RD-31>
- <http://www.eugdpr.org/>
- [RD-32] Riley, S.; Lentz, W. & Clare, A.(2017) 精度への道を歩むために-Android GNSS での観測2017 ION GNSS カンファレンス
- [RD-33] Leibovici et al (2017) Citizen Scienceの検証のための地球観測、または、地球観測の検証のためのCitizen Science?ボランティア観測の品質保証の役割
- RD-34 ロバーツら(2017) 鉄道交通管理システムのための予測知性、ION GNSS+ 2017
- [RD-35] https://developer.android.com/guide/topics/sensors/sensors_overview.html
- [RD-36] <http://www.rtklib.com/>

欧州GNSS機関(GSA)

GSAの使命は、欧州連合の目標をサポートし、ユーザーへの利益、経済成長と競争力の観点から、欧州のGNSS投資の最高のリターンを達成することです。

- ヨーロッパのGNSSサービスとインフラを継続的に改善しながら、ユーザーのニーズに完全に答えるサービスを設計し、可能にします。
- 最も費用対効果の高い方法でユーザーの満足度を確保するための品質の高いサービスの提供を管理します。
- 市場の関係者を巻き込んで、革新的で効果的なアプリケーション、付加価値の高いサービスやユーザー技術を開発し、欧州GNSSの完全採用の達成を促進します。
- 欧州のGNSSサービスと運用が徹底的に安全、安全、アクセス可能であることを保証します。

ガリレオ

ガリレオは、正確な位置情報とタイミング情報を提供する欧州連合の全地球衛星航法システム（GNSS）です。ガリレオは民間の管理下にあるプログラムであり、そのサービスは幅広い用途に利用することができます。自律的でありながら、既存の衛星ナビゲーションシステムとの相互運用も可能です。2020年にはガリレオのFOC（Full Operational Capability）が予定されています。

2016年12月15日、初期サービスの宣言は、ガリレオのオペラ座の始まりを示した。これにより、ガリレオ対応機器をお持ちの方なら誰でも、他のGNSSシステムと組み合わせて、ガリレオの世界的な衛星コンステレーションから提供される信号を測位、ナビゲーション、タイミングのために利用できるようになりました。

完全に配備されたガリレオシステムは、4つの異なるサービスを提供します。

- マスマーケットをターゲットにしたオープンサービス（OS）。OSは、シングル（E1）またはデュアル（E1とE5）のフリークエンシーを提供し、ナビゲーションメッセージ認証（OS NMA）を通じて認証データをブロードキャストするのは初めてとなります。
- 商用サービス（CS）は、OSを補完し、認証による測位精度の向上と信号ロバスト性の向上を実現します。E6の周波数で放送されます。
- 公的規制サービス（PRS）は、政府認定の利用者専用になります。暗号化され、妨害やなりすましに対して安全に保護されます。
- 捜索救助(SAR)サービスは、ほぼリアルタイムにアラートの位置特定とメッセージの検出、より高いビーコンの位置特定の精度、高い可用性、グローバルな衛星カバレッジを可能にします。ガリレオ特有のリターンリンクを持つことになり、誤報率を下げることができます。

useGALILEO.eu

スマートフォンやカーナビなどのガリレオ対応チップセットを搭載したマスマーケット機器は、ガリレオ信号を利用して測位、ナビゲーション、タイミングを行うことができます。
www.useGALILEO.eu ツールは、ガリレオ対応チップセット、スマートフォン、ウェアラブル、トラッキングデバイスを見つけるのに役立ちます。

www.gsa.europa.eu

 @EU_GNSS

 EuropeanGnssAgency

 European-GNSS-Agency

 European GNSS Agency

 EU_GNSS



European
Global Navigation
Satellite Systems
Agency



ISBN 978-92-9206-033-6

doi:10.2878/449581