

RTK方式を利用した 計時システムの基礎研究

海事システム工学科

学籍番号：1721063

和田山晃平

指導教員：久保信明 教授

目次

1. 研究背景
2. RTK測位方式について
3. 計時システムのタイム計測方法
4. 実験結果
5. まとめ

研究背景



赤外線センサーを利用した光電管装置
出典：SEIKO HP

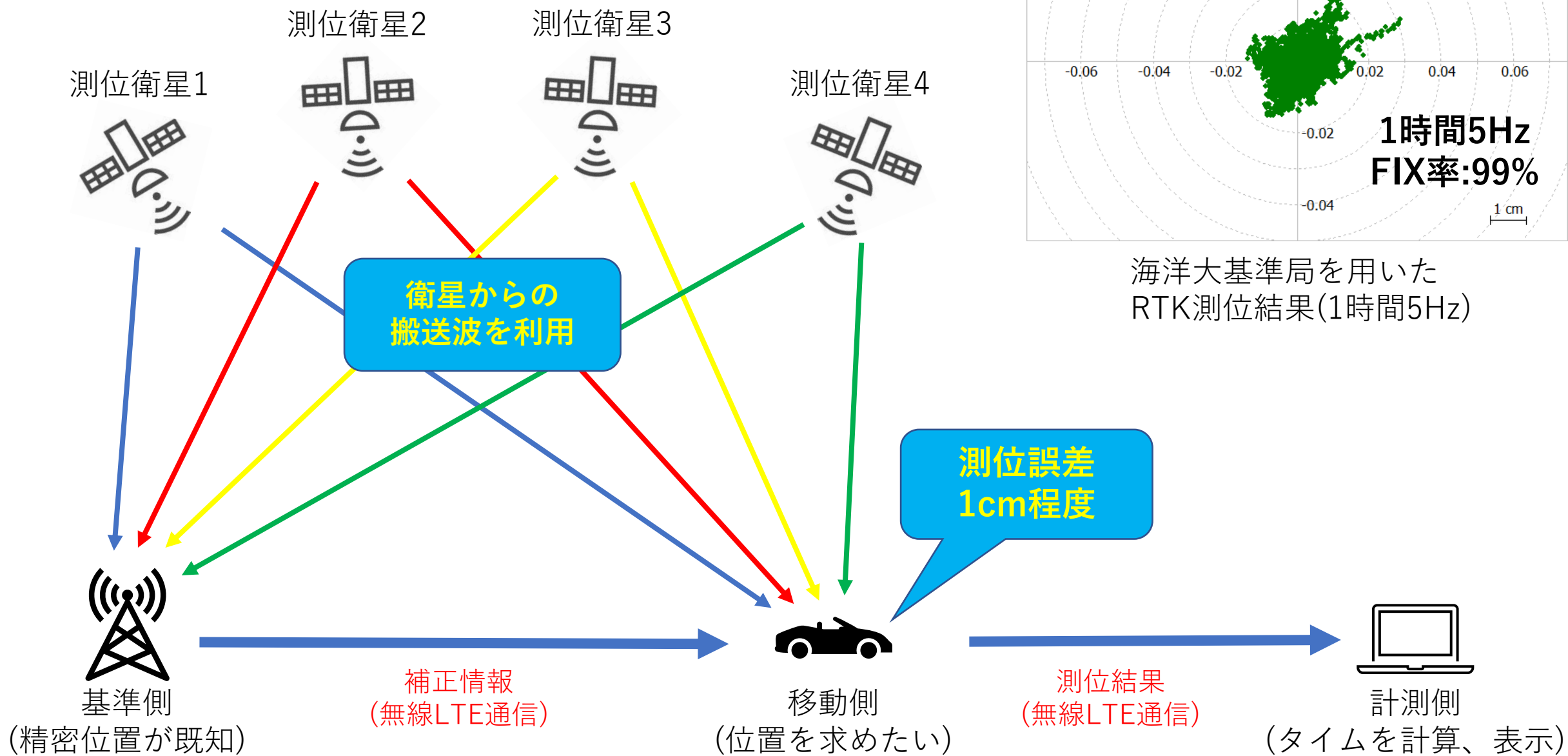


GNSS利用のドローン「AEROBO」
出典:みちびきHP



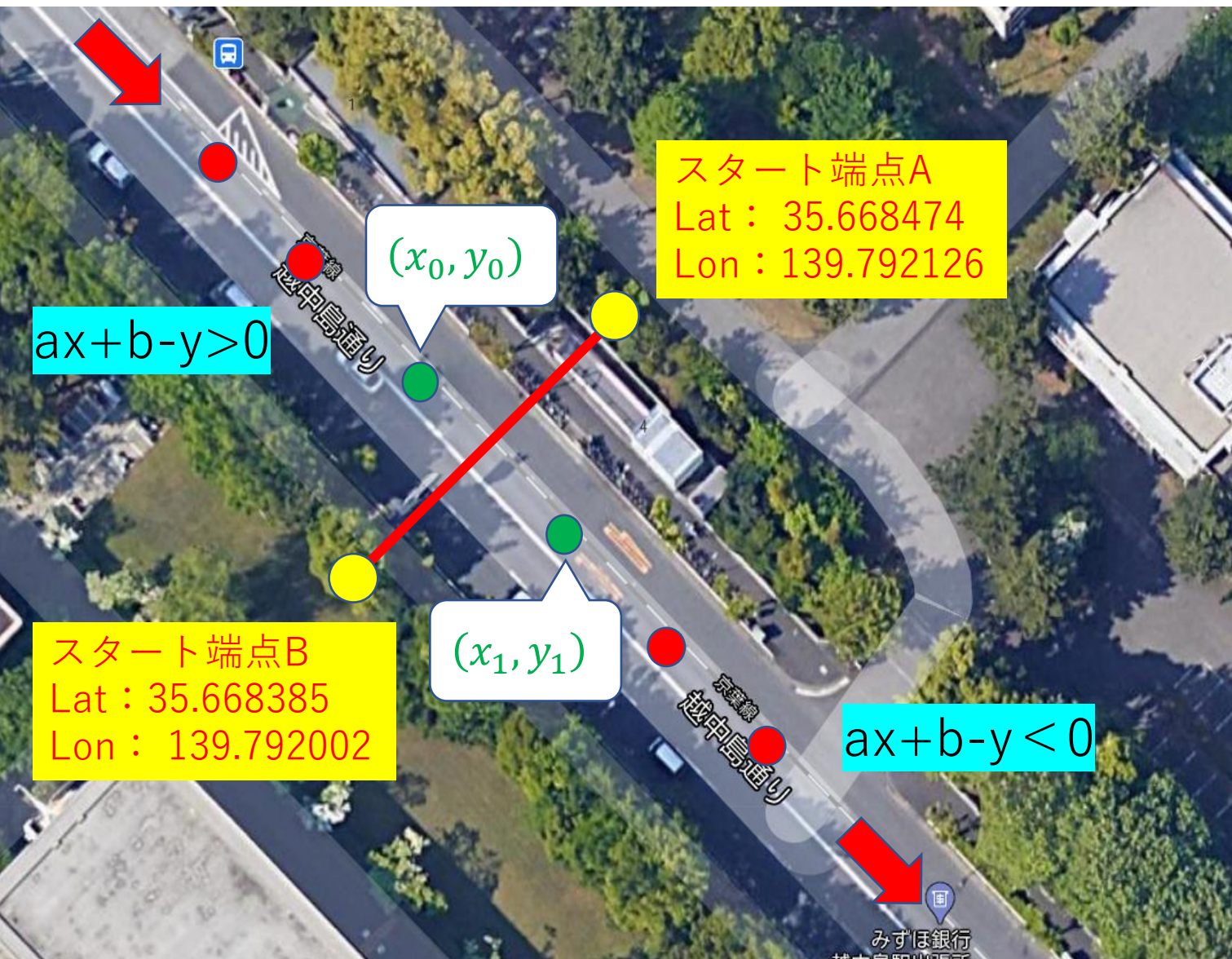
九州工業大学が作成したフォーミュラカー
出典：学生フォーミュラ公式サイト

RTK測位方式



スタート・フィニッシュライン通過判定

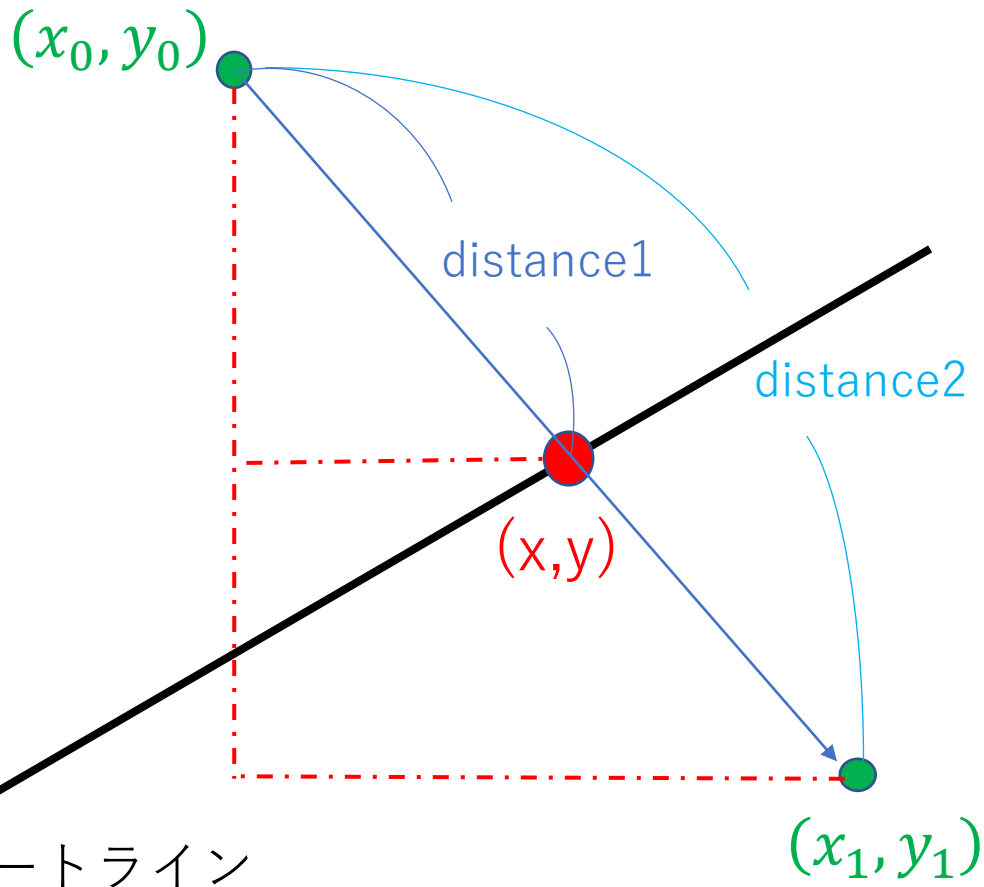




[ライン通過判定]

- ① スタートラインの2つの端点の緯度経度を設定
- ② スタートラインを1次関数化(2点を通る直線式)
- ③ 受信機の緯度経度を①式の x, y に代入していく
- ④ 通過前 (x_0, y_0) のときの①式の正負と、通過後 (x_1, y_1) のときの①式の正負が異なるときにライン通過判定

線形内挿法計算



- ① 通過前後の位置 (x_0, y_0) 、 (x_1, y_1) の2点から1次関数式を算出
- ② スタートラインの式との交点からライン通過座標 (x, y) を求める。
- ③ 三平方の定理より通過前後の距離の比を利用し通過時刻を求める。

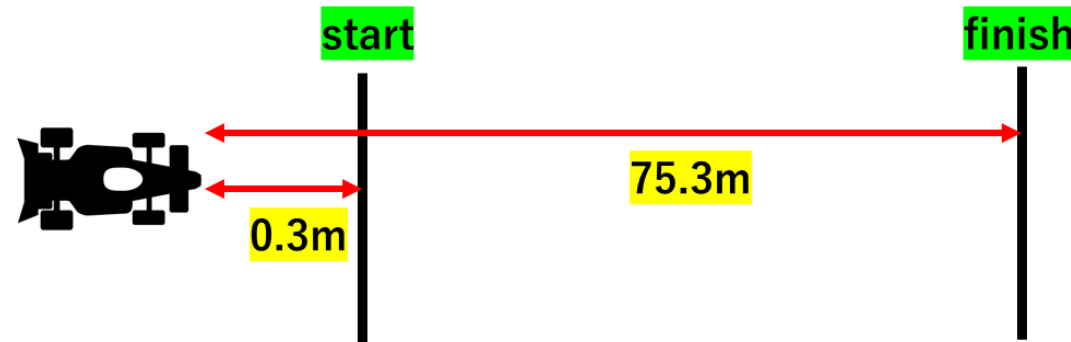
ライン通過時刻

$$= time_0 + \frac{distance1}{distance2} \times (time_1 - time_0)$$

※ $time_0$: 通過前時刻
 $time_1$: 通過後時刻

シミュレーション

- ・ 加速度は1Gを想定
- ・ 等加速度運動を仮定
- ・ 75mの直線コースを車両で走行することを仮定
- ・ 停止位置から0.3m先をスタートラインとし、停止位置から75.3m先をフィニッシュラインと設定



[スタート]

0.1秒内挿	内挿法タイム(s)	仮想真値 タイム(s)
0.01	0.242852	0.248
0.02	0.243672	0.248
0.03	0.244791	0.248
0.04	0.246178	0.248
0.05	0.246707	0.248
0.06	0.244721	0.248
0.07	0.24337	0.248
0.08	0.242572	0.248
0.09	0.242256	0.248
平均	0.244124	0.248

[フィニッシュ]

0.1秒内挿	内挿法タイム(s)	仮想真値 タイム(s)
0.01	3.91869	3.919
0.02	3.918776	3.919
0.03	3.918663	3.919
0.04	3.918577	3.919
0.05	3.918516	3.919
0.06	3.918482	3.919
0.07	3.918473	3.919
0.08	3.918489	3.919
0.09	3.918531	3.919
平均	3.918577	3.919

経過時間計測アプリ : ver. 0.9.3.2

操作 設定 バージョン情報

● 測位情報 (by NMEA via NTRIP Caster)

UTC 2020/10/26 03:25:59.900
LOCAL 2020/10/26 12:25:59.900
GPSTime 2129, 98777.900
Lat. N, 35.6664950°
Lon. E, 139.7924563°
Alt. 58.909 m
Speed 0.046 km/h
Dir. 0.000°

Flag **FIX** Age 0.900

PC情報
OS Time 2020/10/26 12:26:02.345

● NTRIP caster: ● 受信中 <-- [153.121.59.53:2101/ECJ12]

経過時間計測

計測停止 by OS Time 00:00:02.979

● Started --:--:--.--

スタートライン設定

	緯度 (deg.)	経度 (deg.)	高度 (m)
● 端点1	35.6556240	139.8048020	0.000
● 端点2	35.6556820	139.8048960	0.000

ゴールライン設定

	緯度 (deg.)	経度 (deg.)	高度 (m)
● 端点1	35.6560690	139.8043570	0.000
● 端点2	35.6561290	139.8044520	0.000

左:測位情報、経過時間

NTRIPクライアント設定

NTRIP caster 設定

IPアドレス 153 . 121 . 59 . 53 : ポート番号 2101

ユーザ名: gspase

パスワード: *****

NTRIP stream 設定

NTRIP mount point : ECJ12

詳細設定

未受信状態から再接続試行開始するまでの時間: 10 秒

OK キャンセル

右:接続設定画面

ナリタモーターランド実験(11/17)



実験環境

●天候：晴れ

●使用通信アプリ

→基準側：RTKLIB STRSVR

移動側：RTKLIB STRSVR

計測側：作成したタイム計測アプリ

●ネット環境

→基準側：ポケットWi-Fi(Docomo)

移動側：通信キット(Docomo)

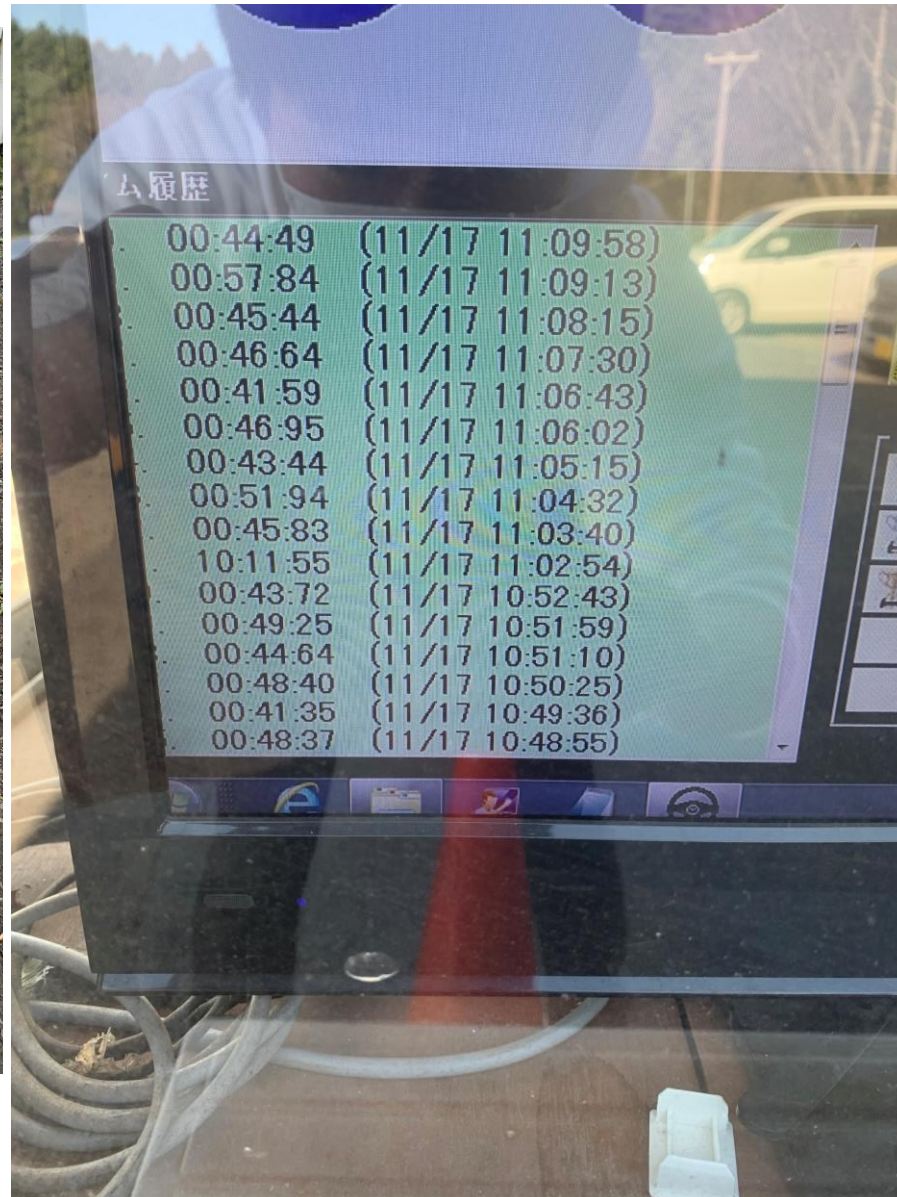
計測側：スマホのテザリング(Docomo)



引用：Google Maps



左：赤外線投光器



中央：計測結果履歴モニター



右：赤外線反射板



左：基準局アンテナ
NovAtel社
GPS-702-GG Antenna



中央：移動側アンテナ
ublox社
Multi-band GNSS antenna
ANN-MB-00

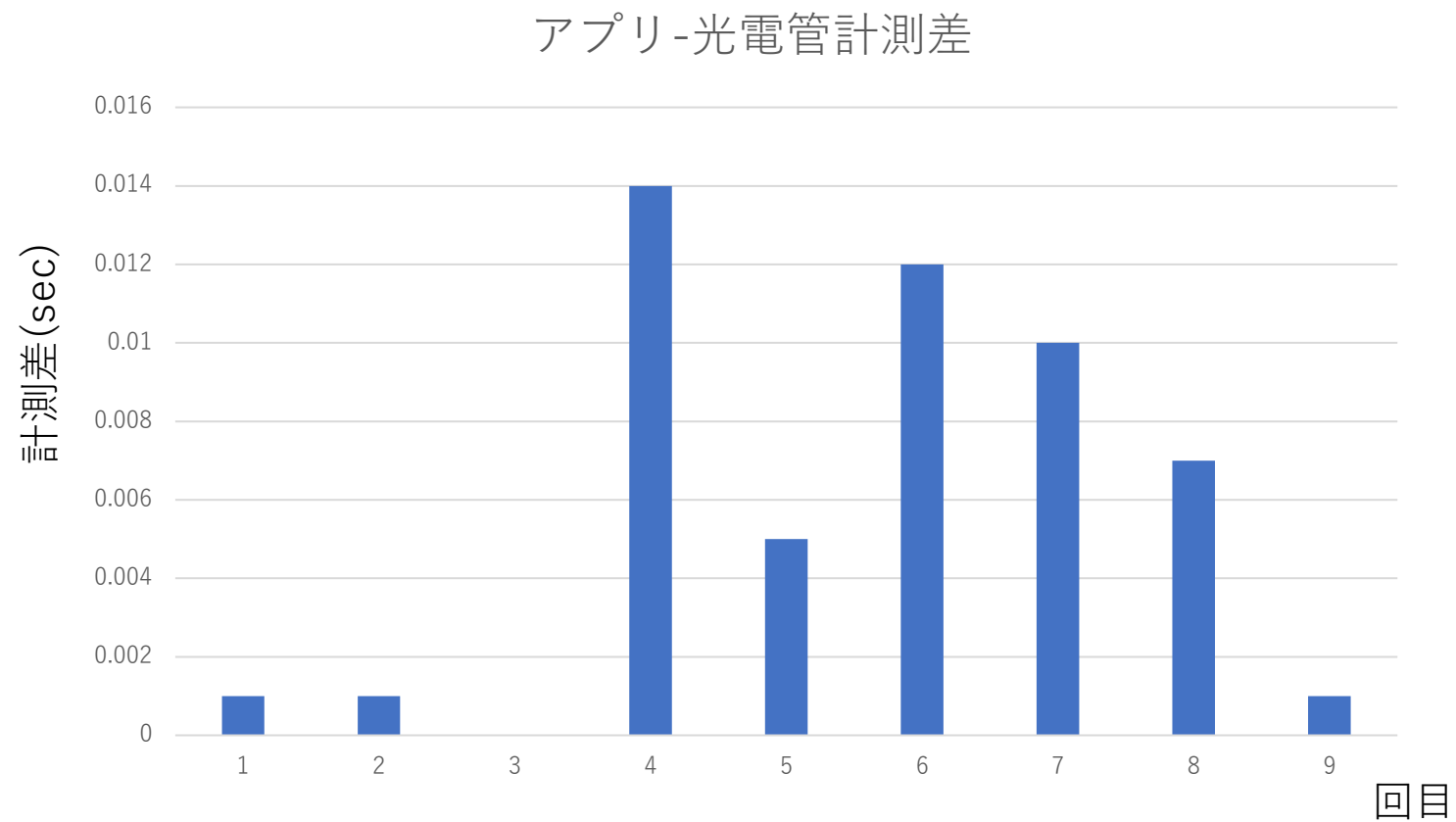


右：計測側受信機
通信キット
(ublox社 ZED-F9P)

計測結果

●定速(35km/h)

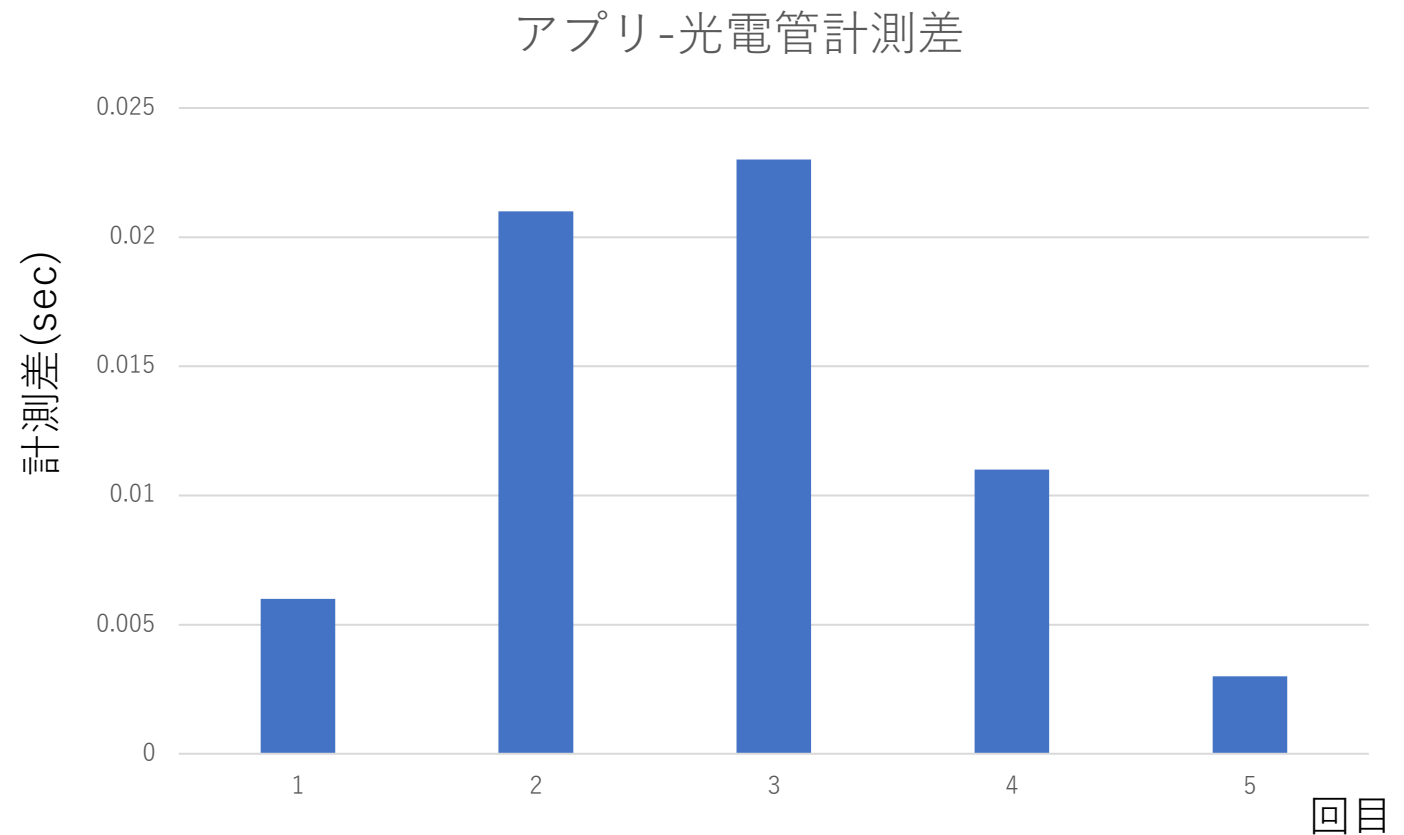
	アプリ計測タイム	光電管計測タイム
1回目	46.419	46.42
2回目	46.391	46.39
3回目	46.530	46.53
4回目	46.264	46.25
5回目	46.375	46.37
6回目	46.352	46.34
7回目	47.120	47.11
8回目	46.347	46.34
9回目	45.751	45.75



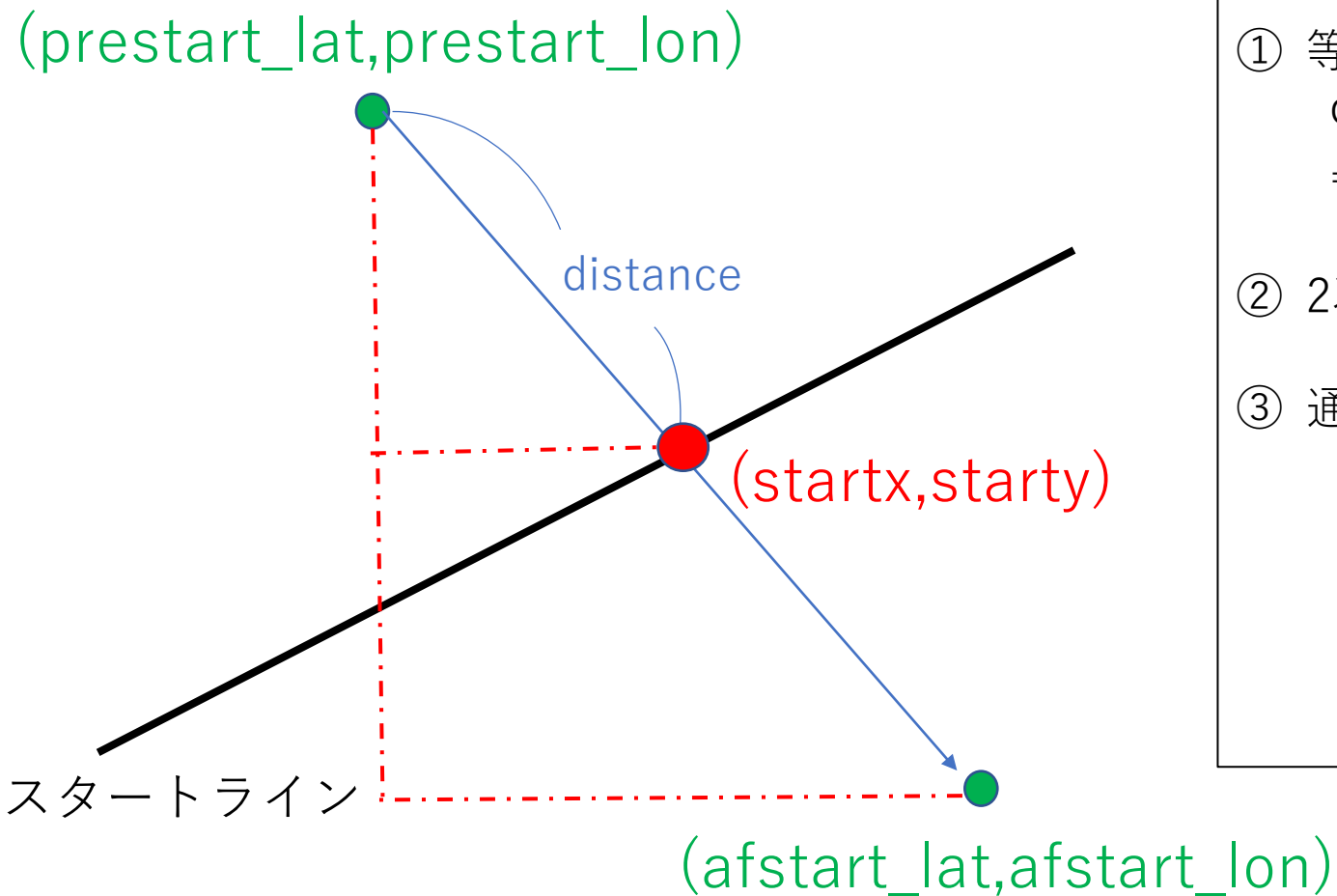
計測結果

● 急発進

	アプリ計測タイム	光電管計測タイム
1回目	48.924	48.93
2回目	42.719	42.74
3回目	41.327	41.35
4回目	44.629	44.64
5回目	43.717	43.72



速度利用メソッド

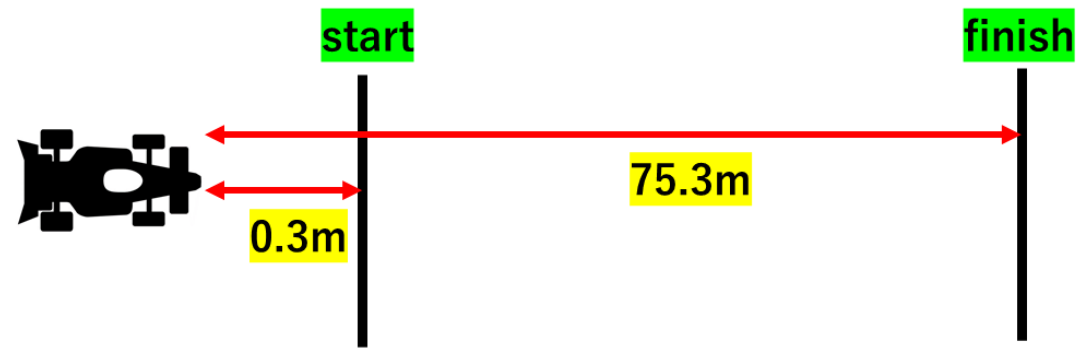


- ① 等加速度運動の公式
distance
= prespeed × (内挿秒) + $\frac{1}{2} \times acceleration \times (内挿秒)^2$
- ② 2次方程式の解の公式より(内挿秒)を算出
- ③ 通過時刻 = prestart_time + (内挿秒)

※ acceleration (m/s^2) = $\frac{|afspeed - prespeed|}{(aftime - pretime)}$

シミュレーション

- ・ 加速度は1Gを想定
- ・ 等加速度運動を仮定
- ・ 75mの直線コースを車両で走行することを仮定
- ・ 停止位置から0.3m先をスタートラインとし、75.3m先をフィニッシュラインと設定



[スタート]

0.1秒内挿	内挿法タイム(s)	仮想真値 タイム(s)
0.01	0.247352	0.248
0.02	0.247352	0.248
0.03	0.247352	0.248
0.04	0.247352	0.248
0.05	0.247352	0.248
0.06	0.247352	0.248
0.07	0.247352	0.248
0.08	0.247352	0.248
0.09	0.247352	0.248
平均	0.247352	0.248

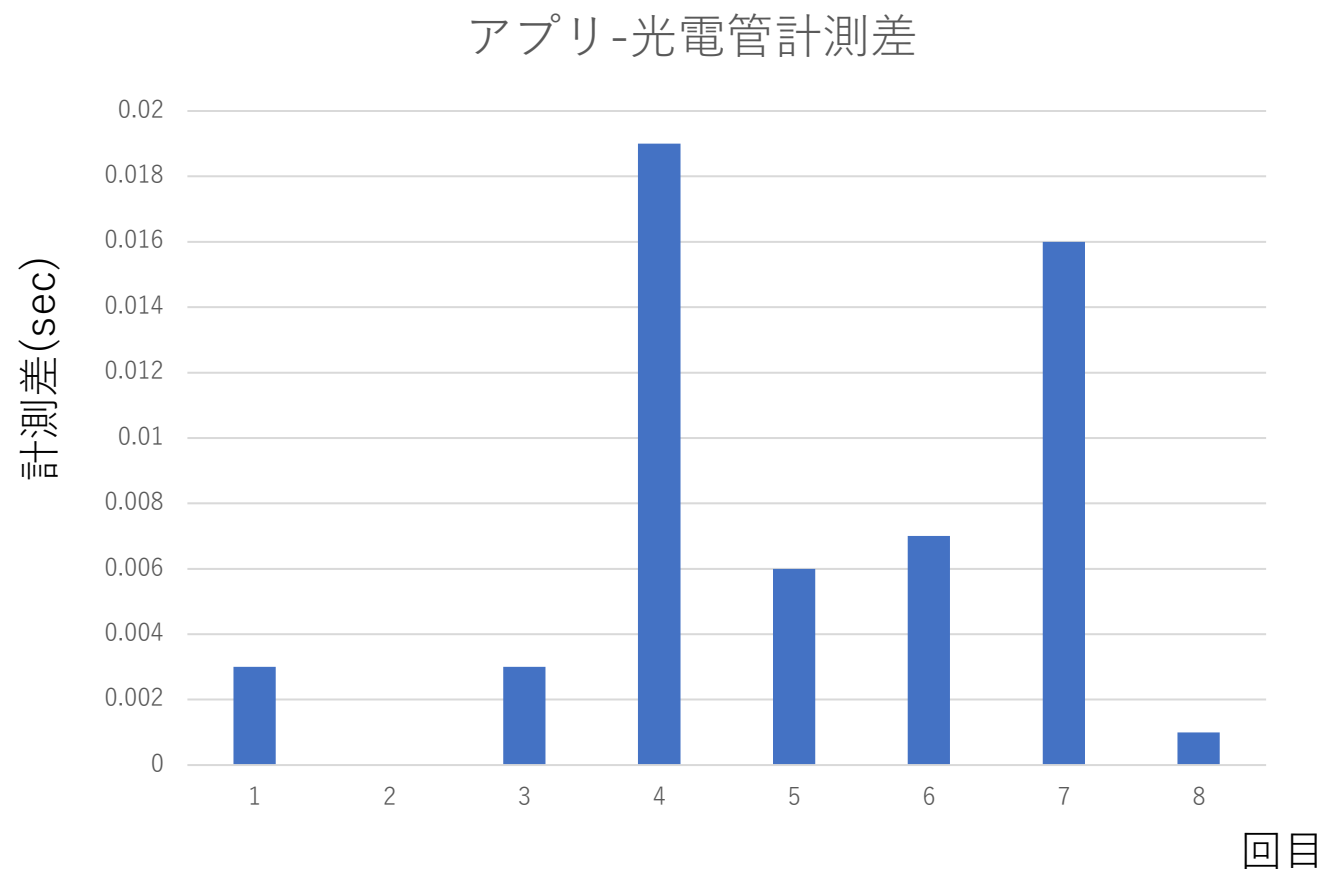
[フィニッシュ]

0.1秒内挿	内挿法タイム(s)	仮想真値 タイム(s)
0.01	3.918791	3.919
0.02	3.918791	3.919
0.03	3.918791	3.919
0.04	3.918791	3.919
0.05	3.918791	3.919
0.06	3.918791	3.919
0.07	3.918791	3.919
0.08	3.918791	3.919
0.09	3.918791	3.919
平均	3.918791	3.919

計測結果

- 前半4回 定速
- 後半4回 急発進

	アプリ計測タイム	光電管計測タイム
1回目	46.267	46.27
2回目	46.190	46.19
3回目	46.563	46.56
4回目	46.519	46.50
1回目	44.946	44.94
2回目	43.173	43.18
3回目	45.764	45.78
4回目	43.719	43.72



まとめ

- RTK測位方式における速度パラメータを利用したタイム計測メソッドではおおむね1/100秒精度でタイムを求めることができた。
- 実験中データ取得を完全に10Hzで行うことができていなかった。

今後の課題

- アプリの通信接続等、適切な通信環境の調査が必要
- 複数台での同時計測も行えると良い