

OrbisNet Sigma - 基線解析・平均化ソフトウェア仕様書

2026年5月 シーラス&株式会社リットー

概要

OrbisNet は、GNSS（衛星測位）による基準点測量を行うための2つのソフトウェアから構成される統合システムです。

- OrbisNet Sigma : 基線解析・平均化ソフトウェア（個別点計算・厳密網平均）
- OrbisNet Logos : 手簿・記簿出力ソフトウェア

OrbisNet Sigma - 基線解析・平均化ソフトウェア

全体フロー



主要機能

1. GNSS受信データの入力

対応形式：

- **.ubx** ファイル (u-blox GNSS受信機の出力)
- **.obs/.nav** ファイル (RINEX形式)

処理：

- **convbin.exe** で UBX → RINEX 変換
- 複数セッション対応

2. 基線解析 (rnx2rtkp)

入力：

- 移動局RINEX：**rover.obs**
- 移動局ナビゲーションメッセージ：**rover.nav**
- 基準局RINEX：**base.obs** (固定基準点方式)
- または RTCMストリーム (RTK方式)

出力：

- **.pos** ファイル (解析結果)

フォーマット (.posファイル)：

```
% ref pos : 31.824063027 130.599590529 314.7612
% GPST      latitude(deg) longitude(deg) height(m) Q ns sdn(m) sde(m) sdu(m) ...
2026/04/13 04:59:53.000 31.738699233 130.617284488 59.9833 1 17 0.0022 0.0024
0.0064 ...
```

各フィールド意味：

フィールド	説明
Q	解の種類 (1=Fix, 2=Float, 5=Single等)
ns	使用衛星数
sdn, sde, sdu	標準偏差 (北・東・上方向、メートル)

3. フィルタリング・平均化 (ParsePosRecordsWithFilter)

目的：ノイズを除去し、信頼度の高い解を抽出

処理フロー：

.posファイル読込

↓

1. 品質フィルター適用

- Q値フィルター：Fix解 (Q=1) のみを抽出
- 時間フィルター：固定時間範囲内のデータを抽出
- 精度フィルター： σ が閾値以下のみを抽出

↓

2. 有効データ抽出

- 品質基準を満たすレコードのみを選択

↓

3. パラメータ計算

- 観測時刻 (開始・終了)
- 衛星数 (系別：GPS/GLONASS/Galileo/Beidou)
- 精度 (σ)
- 基準点抽出 (.posヘッダから自動取得)

主要な計算式：

① 基準点の自動抽出

```
% ref pos : latitude longitude ellipsoid_height
```

↓

```
refLat = 31.824063027
```

```
refLon = 130.599590529
```

```
refHeight = 314.7612
```

② 基線ベクトル成分の計算

メートル毎緯度 ($m/^\circ$):

$$\begin{aligned}\text{meterPerDegreeLat} &= \text{EARTH_RADIUS} \times \pi / 180 \\ &= 6371000 \times \pi / 180 \\ &\approx 111,195 \text{ m/}^\circ\end{aligned}$$

メートル毎経度 ($m/^\circ$):

$$\begin{aligned}\text{meterPerDegreeLon} &= \text{EARTH_RADIUS} \times \cos(\text{refLat}) \times \pi / 180 \\ &= 6371000 \times \cos(31.824^\circ) \times \pi / 180 \\ &\approx 94,871 \text{ m/}^\circ\end{aligned}$$

基線ベクトル成分:

$$\Delta X = (\text{lat} - \text{refLat}) \times \text{meterPerDegreeLat}$$

$$\Delta Y = (\text{lon} - \text{refLon}) \times \text{meterPerDegreeLon}$$

$$\Delta Z = \text{height} - \text{refHeight}$$

例（実測値）：

観測点：lat = 31.738699, lon = 130.617284, height = 59.9833

基準点：lat = 31.824063, lon = 130.599590, height = 314.7612

$$\Delta X = (31.738699 - 31.824063) \times 111195 \approx -9492.02 \text{ m}$$

$$\Delta Y = (130.617284 - 130.599590) \times 94871 \approx +1671.71 \text{ m}$$

$$\Delta Z = 59.9833 - 314.7612 \approx -254.78 \text{ m}$$

③ 基線距離の計算

基線距離:

$$S = \sqrt{(\Delta X^2 + \Delta Y^2 + \Delta Z^2)}$$

例:

$$\begin{aligned} S &= \sqrt{(9492.02^2 + 1671.71^2 + 254.78^2)} \\ &= \sqrt{(90,097,420 + 2,794,614 + 64,914)} \\ &= \sqrt{92,956,948} \\ &\approx 9641.47 \text{ m} \end{aligned}$$

④ 精度 (σ) の計算

水平 σ :

$$\begin{aligned} \sigma_{\text{Horz}} &= \sqrt{(\sigma_N^2 + \sigma_E^2)} \\ &= \sqrt{(0.0022^2 + 0.0024^2)} \\ &= \sqrt{(0.00000484 + 0.00000576)} \\ &= \sqrt{(0.0000106)} \\ &\approx 0.00326 \text{ m} \end{aligned}$$

鉛直 σ :

$$\sigma_{\text{Vert}} = \sigma_U = 0.0064 \text{ m}$$

平均化方法:

複数レコードがある場合、全レコードの値から中央値を計算:

$$\begin{aligned} \sigma_{\text{Horz_avg}} &= \text{median}(\sqrt{(\sigma_{N_1}^2 + \sigma_{E_1}^2)}, \sqrt{(\sigma_{N_2}^2 + \sigma_{E_2}^2)}, \dots) \\ &\approx 0.000608 \text{ m} \end{aligned}$$

$$\begin{aligned} \sigma_{\text{Vert_avg}} &= \text{median}(\sigma_{U_1}, \sigma_{U_2}, \dots) \\ &\approx 0.001162 \text{ m} \end{aligned}$$

⑤ 衛星数の推定

.posファイルの "ns" から総衛星数を推定：

衛星数比率（実績データより）：

GPS (G) : 50%

GLONASS (R) : 25%

Galileo (E) : 20%

Beidou (B) : 5%

例：ns=17の場合

SatG = $17 \times 0.50 = 9$ (GPS)

SatR = $17 \times 0.25 = 4$ (GLONASS)

SatE = $17 \times 0.20 = 3$ (Galileo)

SatB = $17 \times 0.05 = 1$ (Beidou)

⑥ 標高の計算

楕円体高 - ジオイド高 = 標高（正規高）：

```
altitude = ellipsoid_height - geoid_height
           = 59.9833 - 31.8345
           = 28.1488 m
```

ジオイド高：

```
geoid_height = GeoidCorrection.CalculateGeoidHeight(lat, lon)
              （日本のジオイド2011 V2.1のテーブルから補間）
```

4. 座標系の自動判定

全国対応：19座標系（ゾーン1～19）

経度 → 座標系判定：

ゾーン1：130° 00'E ~ 136° 00'E（鹿児島県・沖縄県）

ゾーン2：126° 00'E ~ 130° 00'E（九州西部）

...

ゾーン8：139° 00'E ~ 142° 00'E（関東地方）

ゾーン9：142° 00'E ~ 145° 00'E（東北地方）

...

判定アルゴリズム：

$zoneNumber = \text{Math.Floor}((referencePointLon + 0.0000001) / 1.5) + 1$

5. 平面直角座標の計算（BL2XY変換）

入力：緯度経度（world Geodetic System 84）、座標系ゾーン番号 出力：平面直角座標（X, Y）

計算手順：

① 緯度経度 → 測地座標へ変換

- 楕円体パラメータ（GRS80）を使用

② 測地座標 → 横メルカトル投影

- 中央子午線での投影

- スケール係数 $m_0 = 0.9999$

③ 横メルカトル座標 → 平面直角座標

- $X_{offset} = 0.0$ m（北）

- $Y_{offset} = -referenceX$ （東、参考値からのシフト）

実装：`CoordinateConverter.ConvertToPlaneRectangular(lat, lon, zoneNumber)`


6. 時刻処理 (UTC → JST)

入力: .posファイルの時刻 (UTC) 出力: 日本標準時 (JST = UTC+9)

```
// UTC → JST 変換
```

```
TimeZoneInfo jstZone = TimeZoneInfo.FindSystemTimeZoneById("Tokyo Standard  
Time");
```

```
DateTime jstTime = TimeZoneInfo.ConvertTime(utcTime, TimeZoneInfo.Utc, jstZone)
```

 出力フォーマット (OrbisNet JSON)
ファイル形式: JSON (.orbisnet 拡張子)

```
{
  "orbisnet_format": "1.0",
  "generated_by": "OrbisNet Sigma",
  "generated_at": "2026-04-27T13:18:06Z",
  "project": {
    "project_name": "令和〇年度 GNSS基準点測量業務",
    "observation_date": "2026-04-13",
    "method": "RTK",
    "reference_point": "TS7788",
    "fixed_coord_x": "24600.523",
    "fixed_coord_y": "-48176.791",
    "fixed_height": "592.660",
    "fixed_ellipsoid_h": "634.914",
    "fixed_geoid_h": "42.254",
    "coord_sys_no": "2 系",
    "geoid_name": "日本のジオイド2011 V2.1"
  },
  "records": [
    {
      "point_name": "1",
      "point_no": "1",
      "start_time": "04:59:53",
      "end_time": "05:05:19",
      "antenna_height": 1.65,
      "sat_g": 10,
      "sat_r": 4,
      "sat_e": 3,
      "sat_b": 0,
      "solution_type": "Fix",
      "delta_x": -9492.023,
      "delta_y": 1671.710,
      "delta_z": -254.776,
      "dist_s": 9641.475,
      "ellipsoid_h": 59.986,
      "geoid_h": 31.835,
      "altitude": 28.151,
      "sigma_horz": 0.000608,
      "sigma_vert": 0.001162,
      "plane_x": -139809.756,
      "plane_y": -36262.624
    }
  ]
}
```

OrbisNet Logos - 手簿・記簿出力ソフトウェア

機能

1.Sigma からの JSON インポート

1. 自動でプロジェクト情報と観測データを読み込
2. 固定点座標を動的に設定

2.手簿（観測手簿）の出力

1. 観測日時、セッション名、天候
2. 捕捉衛星数（G/R/E/B別）
3. アンテナ高、観測開始・終了時刻

3.記簿（観測記簿）の出力

1. 解析ソフト、軌道情報、楕円体
2. 座標系番号、ジオイド名
3. 固定観測点の座標
4. 基線ベクトル成分（ $\Delta X/\Delta Y/\Delta Z$ 、距離S）
5. 平面直角座標（X/Y）
6. 高さ（楕円体高、ジオイド高、標高）
7. 精度（ σ_{Horz} , σ_{Vert} ）

4.PDF/印刷出力

1. FlowDocument を使用した帳票レイアウト
2. 複数ページ対応

データフロー

OrbisNet JSON

↓

ImportOrbisNetFile()

↓

ApplyProjectInfo() / ImportRecords()

↓

Project プロパティ / Records コレクション

↓

BuildPreviewDocument() / BuildKisboDocument()

↓

FlowDocument (プレビュー表示)

↓

印刷 / PDF 出力

計算精度

固定点座標の精度

基線ベクトル成分の誤差：

$\Delta(\Delta X) \approx \pm 2 \text{ mm}$ (衛星測位精度)

$\Delta(\Delta Y) \approx \pm 2 \text{ mm}$

$\Delta(\Delta Z) \approx \pm 3 \text{ mm}$

平面直角座標への変換誤差：

$\Delta(X) \approx \pm 1 \text{ cm}$ (投影歪みが小さいため)

$\Delta(Y) \approx \pm 1 \text{ cm}$

高さの精度

標高の誤差：

$\Delta(\text{高さ}) \approx \pm 3 \text{ mm}$ (ジオイド高テーブルの精度)

ジオイド高の精度：

$\sigma(\text{geoid_h}) \approx \pm 5 \text{ mm}$ (日本のジオイド2011 V2.1)

全国対応と自動化

自動判定される項目

1.座標系 (ゾーン番号)

1. 固定点の経度から自動計算
2. ゾーン1~19に対応

2.基準点座標

1. .posファイルのヘッダから自動抽出
2. 「% ref pos」セクションから読込

3.ジオイド高

1. 観測点の緯度経度から自動補間
2. 日本のジオイド2011 V2.1を使用

4.衛星数 (系別)

1. .pos ファイルの **ns** から推定
2. GPS/GLONASS/Galileo/Beidouの比率を計算

5.時刻 (JST)

1. .pos ファイルの UTC を JST に変換
2. TimeZoneInfo を使用した正確な変換

項目	仕様
言語	C# (.NET Framework 4.7+)
UI	WPF (Windows Presentation Foundation)
データ形式	JSON (RFC 8259)
文字エンコーディング	UTF-8
出力形式	PDF、印刷、FlowDocument
座標系	日本平面直角座標系 (19ゾーン)
楕円体	GRS80 (世界測地系)
ジオイド	「ジオイド2024日本とその周辺」

外部ツール

ツール	用途	バージョン
convbin	RINEX変換	RTKLIB v2.4.2+
rnx2rtkp	基線解析	RTKLIB v2.4.2+
GeoidCorrection	ジオイド高計算	カスタム実装
CoordinateConverter	座標変換	カスタム実装

参考資料

- RTKLIB: <https://github.com/tomojitakasu/RTKLIB>
- ジオイド2024日本とその周辺: <https://www.gsi.go.jp/>
- GNSS測量技術基準: 国土交通省