

6.3 RTK-GPS 用プログラムライブラリ RTKLIB の開発・評価および応用

高須知二、久保信明、安田明生（東京海洋大学）

ttaka@gpspp.sakura.ne.jp, nkubo@e.kaiyodai.ac.jp, yasuda@kaiyodai.ac.jp

6.3.1 はじめに

RTK-GPS（Realtime Kinematic GPS）は GPS/GNSS 衛星から送信された測位信号を使って受信機位置を cm レベルの精度で測る精密測位技術である。既に RTK-GPS は精密測量、建築工事管理、地殻変動監視、地図作成、移動体精密位置決定といった広汎な分野で利用されている。今のところ RTK-GPS 用受信機が高価である等の理由で、一般測位用として普及しているという訳ではないが、今後新しい民生用測位信号を備えた近代化 GPS、Galileo、GLONASS、準天頂衛星等の利用可能な衛星や測位信号の数が増えるにつれて、その適用可能領域が拡大し、従来の測位技術では精度が十分でなかった分野を中心に、更なる応用が期待されている。

筆者らは主に RTK-GPS を中心とした精密測位技術の評価や応用研究を進める上で共通的なツールとして使用するために汎用の測位演算プログラムライブラリを開発した。このライブラリを RTKLIB と呼んでいる。本稿では RTKLIB の開発、評価および今後の拡張予定について紹介する。

6.3.2 RTKLIB の開発

6.3.2.1 背景および開発目標

通常、RTK-GPS 測位には RTK-GPS 用受信機にファームウェアとして内蔵される機能を使用することが多い。これらは一般の利用者にとっては十分な機能・性能を提供してくれるが、精密測位の研究用に利用しようとするとき使い難い点が多い。主な問題点としては、パラメータ設定の自由度が少ないこと、内部動作や技術の多くが非公開であり変更や新機能の追加が自由にできないこと、問題が発生した際にその原因を追跡するのが困難なこと、等である。すなわち RTK-GPS 用受信機は原則としてブラックボックスとしての利用が前提であり、研究開発目的で自由に改造して使用するのは困難である。以上の理由から、設

定変更、改変や機能追加が容易で、それを利用して自由に応用プログラムを開発することができる測位演算プログラムライブラリの必要性を感じていた。

既存の汎用測位演算プログラムライブラリとしては、GPSTk /1/ が著名である。GPSTk は米国テキサス大学オースチン校を中心にオープンソースとして開発が続けられているソフトウェアである。多機能で品質も高いが、その反面、内部は複雑でその全容を理解するのは簡単でない。また、C++ と STL を使って開発されており移植性や性能面の問題が生じやすいこと、RTK-GPS 測位用としては基本機能しか実装されていないため限定された観測データしか解析できないこと、等の問題があり、全くそのまま利用することは難しかった。

以上の背景から、研究開発中の精密測位アルゴリズムの検証を兼ねて、新規に RTK-GPS 測位演算機能を含めた汎用の測位演算プログラムライブラリを開発することにした。初期の目標としたのは以下の点である。

- ・ 構造、機能、API が簡潔であること。
- ・ 広汎な応用プログラムが開発可能なこと。
- ・ 組込 CPU を含め各種環境に移植容易なこと。
- ・ 測位性能、実行性能が十分であること。
- ・ 実用的な解析に利用可能なこと。

以上の目標を達成するため以下の方針を立てた。

- ・ 開発言語・API は ANSI 標準 C (89)。
- ・ 行列演算は LAPACK/BLAS を利用。
- ・ 推定手法はカルマンフィルタを利用。
- ・ OTF (on-the-fly) 整数 Ambiguity Resolution (AR) は LAMBDA /2/ およびその拡張を利用。
- ・ 後処理用応用プログラム用に RINEX ファイル操作機能を付加する。

以上で開発した測位演算用プログラムライブラリ、サンプル応用プログラム、および関連ユーティリティを総称して **RTKLIB** と呼んでおり、2007年10月現在、安定版バージョンが **ver.1.1**、開発版バージョンが **ver.2.1** となっている。

6.3.2.2 機能

RTKLIB (ver.1.1) が提供するライブラリの機能は以下の通りである。

- (1) 行列・ベクトル演算
行列生成、内積、ノルム、行列積、逆行列
- (2) 時刻・文字列処理
文字列数値間変換、時刻変換、時刻演算
- (3) 座標系変換
座標変換、緯度経度高度算出、ジオイド高
- (4) RINEX ファイル読み込み
RINEX (ver.2.10) 観測データ、航法メッセージファイル読み込み
- (5) 求解・推定
線型方程式、最小二乗法、カルマンフィルタ
- (6) 航法演算
計算、幾何学距離計算、衛星方位・仰角計算
- (7) 大気圏モデル
Saastamoinen モデル、Klobuchar モデル
- (8) OTF 整数 Ambiguity 決定
LAMBDA/MLAMBDA
- (9) 単独測位
- (10) RTK-GPS 測位

以上のうち行列・ベクトル演算ルーチン中で高速な実装が多数ある **LAPACK/BLAS** を使用している。ただし可搬性を考えてこれらを利用できない環境用に互換ルーチンも実装している。

6.3.2.3 アルゴリズム

RTKLIB の心臓部である **RTK-GPS** 測位ルーチンのフローを図 6.3-1 に示し、以下に簡単にその動作を説明する。¹²⁾

RTK-GPS 測位ルーチンでは最初に 1 エポック分の観測データおよび航法メッセージを入力し、いったん疑似距離観測値を使った単独測位により初期近似解を算出する。ここで近似解は相対測位ルーチン中でフィルタ初期化及び観測方程式線形化に使われる。

次の相対測位ルーチンでは、最初にカルマンフィルタの時間更新を行う。この中でサイクルスリ

ップや新衛星の検出を行い必要な初期化・再初期化を行っている。次に、搬送波位相及び疑似距離観測値による二重差観測値及び観測方程式から残差 (イノベーション)、観測行列を計算する。その後、観測誤差共分散行列を生成し、一般のカルマンフィルタ観測更新を行い、このエポックの推定値 (**FLOAT** 解) とその共分散行列を得る。なお、設定によって、疑似距離観測値のみでこの相対測位ルーチンを動作させることができるので **DGPS** 解を出力することも可能である。相対測位ルーチンの処理フローを図 6.3-2 に示す。

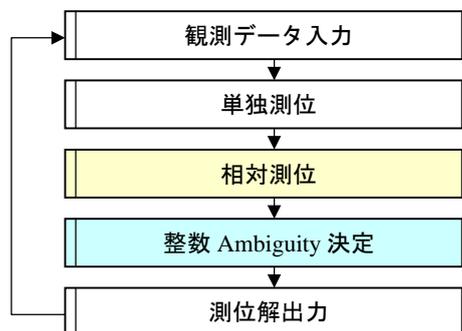


図 6.3-1 RTK-GPS 測位処理フロー

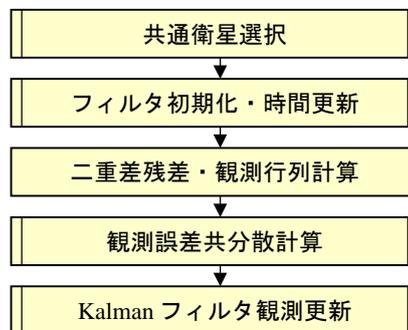


図 6.3-2 相対測位処理フロー

次に、推定値及びその共分散行列を使った整数 **Ambiguity** 決定ルーチンに入る。以上のフィルタでは衛星切り換わり操作の煩雑さを避けるため、一重差搬送波位相バイアスを推定しているため、まずこれら推定値から二重差推定値を生成する。次に生成した実数推定値から整数解を求める。整数 **Ambiguity** 決定のために今まで多数の手法が提案されているが、ここでは **OTF AR** として、現在最も一般的である整数最小二乗法の一つである **LAMBDA/3** を使用した。ただし整数解検索にはより検索効率の良い **MLAMBDA/4** を採用し

ている。整数 Ambiguity 決定処理フローを図 6.3-3 に示す。

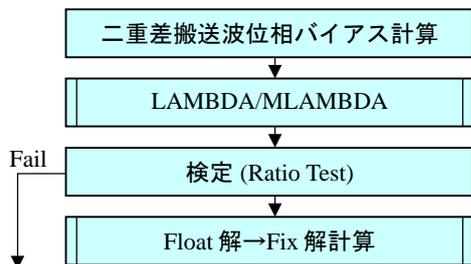


図 6.3-3 整数 Ambiguity 決定処理フロー

実装した整数 Ambiguity 決定ルーチンの性能評価も行ったが、更新レート 10Hz 程度の単一基線条件なら、組込 CPU でも十分な性能が得られることが分かった。

6.3.2.4 API (Application Program Interface)

RTKLIB では 6.3.2.2 で説明した機能にほぼ一対一に対応して、ユーザプログラムから利用することができる API が用意されている。付録 A に RTKLIB が提供する API 一覧を示す。そのうち RTK-GPS 測位演算ルーチンである `rtkpos()` は、ユーザプログラムから、引数として入力コールバック関数、出力コールバック関数を渡して呼び出ることにより 6.3.2.3 で説明した、エポック毎の測位演算処理を繰り返し実行する構造になっている。従って、ユーザが独自の観測データ・航法メッセージ入力ルーチンと測位解出力ルーチンを記述し RTKLIB とリンクすることにより、各種受信機や出力デバイスに対応した RTK-GPS 応用プログラムを開発することが可能となる。

6.3.2.5 応用プログラム

以上 RTKLIB のライブラリを利用したプログラムサンプルとして RTKLIB ver.1.1 には `rnx2rtkp` 及び `pos2kml` と呼ぶ応用プログラムが付属している。このうち `rnx2rtkp` は RINEX 形式の観測データと航法メッセージファイルを読み込んで RTK-GPS アルゴリズムにより後処理基線解析を行うコマンドラインプログラムである。付録 B に `rnx2rtkp` のコマンド仕様を示す。これらのライブラリ及びプログラムは、全て標準 C で記述されているので、ほとんどどんな環境でも構築可能である。動作確認済みの環境は Windows+msc (Visual Studio)、icc、bcc、gcc (cygwin)、Linux+gcc、Mac OS-X+gcc 等である。

6.3.3 RTKLIB の評価

以上説明した RTKLIB の動作検証を兼ねて、付属応用プログラム `rnx2rtkp` を使った後処理基線解析により RTK-GPS 性能の評価を行った。

6.3.3.1 固定点測位

国土地理院電子基準点から 0.3km~33km の基線を選択し、2004 年 1 年分の観測データを `rnx2rtkp` を使って解析し、測位解精度及び FIX 率を評価した。基準局座標は国土地理院 F2 解に固定し、ローバ測位解を同様に F2 解と比較した。表 6.3-1 に評価結果を示す。短基線条件では RTK-GPS 精度として一般的に言われる、水平 1cm+1ppm×基線長をほぼ満足しており、良好な精度を達成していると言える。

これらのうち典型的な短基線条件である 4.3km 基線での測位解精度及び FIX 率の年間推移を図 6.3-4~5 に示す。概ね安定した性能が得られていると考えられるが、特定日の精度悪化は電離層擾乱、夏場の FIX 率悪化は対流圏遅延の影響と考えられ、より実用的には何らかの対策が必要と思われる。

表 6.3-1 測位解 (FIX 解) 精度および FIX 率

基線長	FIX 解 RMS 誤差 (cm)			FIX 率
	E-W	N-S	U-D	
0.3 km	0.63	0.44	0.96	97.3 %
3.3 km	1.02	0.97	1.58	96.1 %
4.3 km	1.02	1.57	2.81	88.0 %
9.9 km	1.59	2.11	4.66	75.1 %
19.6 km	2.47	3.80	7.50	58.9 %
33.1 km	4.60	6.30	10.39	38.0 %

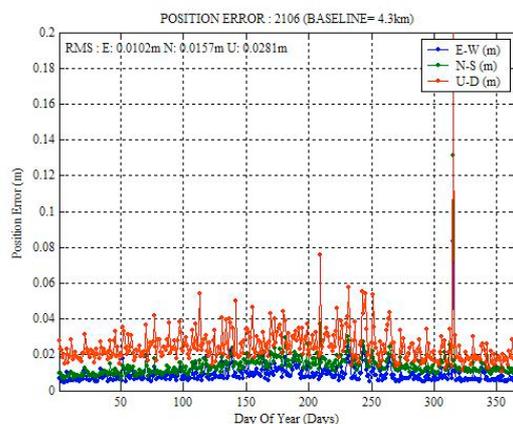


図 6.3-4 測位解精度 (基線長 4.3km, 日毎 RMS 誤差)

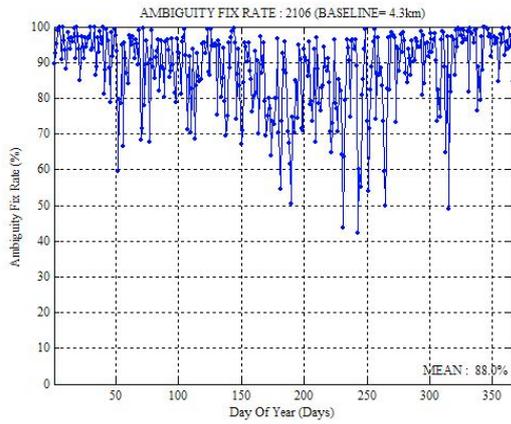


図 6.3-5 測位解 FIX 率 (基線長 4.3km、日毎)

6.3.3.2 移動体測位

次に移動体 RTK-GPS 性能評価のため、自動車ループに GPS アンテナを設置し東京海洋大近辺の道路を走行して観測データを取得した。この観測データを使って `rnx2rtkp` により解析を行い FIX 率評価を行った。データ取得日時は (1)2005/7/19 11:12~11:54、(2)2006/6/12 10:26~11:20、受信機は NovAtel 二周波受信機 OEM-4 である。基準局は東京海洋大第 4 実験棟屋上に設置し、基線長は(1) 0.2~2.7km、(2) 0.1~3.1km であった。各測位条件における測位解数を表 6.3-2 に示す。ここで Combined 解は Forward 解/Backward 解をスムーズで結合した後処理基線解析を想定した測位条件である。

表 6.3-2 測位解数及び割合

	測位条件	測位解数 (割合)		
		FIX	FLOAT	Single
(1)	連続 AR	1859 (85.6%)	252 (11.6%)	60 (2.8%)
	瞬時 AR	1738 (80.1%)	395 (18.2%)	38 (1.8%)
	Combined	2062 (95.0%)	70 (3.2%)	39 (1.9%)
(2)	連続 AR	2083 (61.9%)	1103 (32.8%)	180 (5.3%)
	瞬時 AR	1499 (44.5%)	1742 (51.8%)	125 (3.7%)
	Combined	2637 (78.3%)	590 (17.5%)	139 (4.1%)

以上のうち(1) 連続 AR 条件の測位解軌跡の拡大図例を図 6.3-6 に示す。条件の良い(1)で 86%、条件の悪い(2)で 62%程度の FIX 率が得られており、測位解軌跡を確認しても上空視界の良い区間では概ね滑らかな解が認められ実用的な性能が

得られていると考えられる。

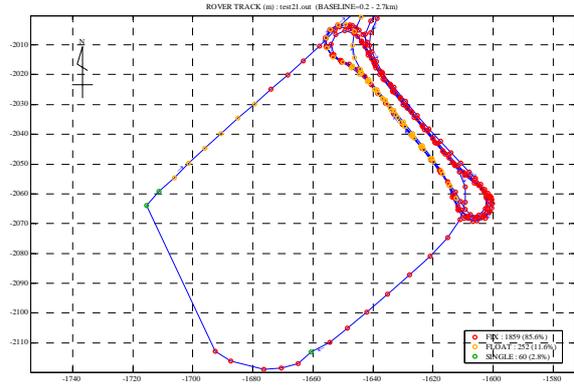


図 6.3-6 測位解軌跡 (1) 連続 AR (赤:Fix,橙:Float)

6.3.4 RTKLIB の拡張

既に関発中の RTKLIB ver.2.1 で拡張予定の機能は以下の通りである。

- ・中長基線対応、電離層・対流圏遅延推定
- ・アンテナ位相中心変動 (PCV) 補正
- ・基準局観測データ補間機能
- ・ネットワーク型 RTK 補正機能
- ・解析実行・観測データ・測位解表示 GUI
- ・SBAS (MSAS) 補正機能
- ・受信機・通信インタフェース機能

6.3.5 おわりに

精密測位研究用に開発した汎用測位演算プログラムライブラリ RTKLIB についてその開発・評価および拡張予定について簡単に紹介した。評価の結果、RTKLIB は RTK-GPS 測位用として概ね十分な性能を有していることが確認できた。今後、RTKLIB を精密測位技術の研究開発用ツールとして利用し、評価を行うと同時に、今後も機能拡張を続けていく予定である。

参考文献

- /1/ B.B.Haris et al., The GPSTk: an open source GPS toolkit, GPS Solutions, Vol.11, No.2, 2007
- /2/ 高須, 搬送波位相測定値による精密測位の理論及び解析処理, GPS/GNSS シンポジウム 2005
- /3/ P.J.G.Teunissen, The least-square ambiguity decorrelation adjustment: a method for fast GPS ambiguity estimation, J.Geodesy, Vol.70, 1995
- /4/ X.-W.Chang, X.Yang, T.Zhou, MLAMBDA: A modified LAMBDA method for integer least-squares estimation, J.Geodesy, Vol.79, 2005

付録 A RTKLIB ver.1.1 API 一覧

Function	Description	Reference *1
	Matrix and vector functions	
mat()	New matrix	A.1
zeros()	New zero matrix	A.2
eye()	New identity matrix	A.3
dot()	Inner Product	A.4
norm()	Euclid norm	A.5
matmul()	Multiply matrix	A.6
matinv()	Inverse of matrix	A.7
solve()	Solve linear equation	A.8
lsg()	Least square estimation	A.9
filter()	Kalman filter state update	A.10
matprint()	Print matrix	A.11
	Time and string functions	
str2num()	String to number	A.12
str2time()	String to time	A.13
epoch2time()	Calendar day/time to time	A.14
time2epoch()	Time to calendar day/time	A.15
gpst2time()	GPSTIME to time	A.16
time2gpst()	Time to GPSTIME	A.17
timeadd()	Add time	A.18
timediff()	Time difference	A.19
gpst2utc()	GPSTIME to UTC	A.20
utc2gpst()	UTC to GPSTIME	A.21
time2str()	Time to string	A.22
	Coordinates transformations	
ecef2pos()	ECEF to geodetic position	A.23
pos2ecef()	Geodetic to ECEF position	A.24
ecef2enu()	ECEF to local coordinates	A.25
enu2ecef()	Local to ECEF coordinates	A.26
covenu()	Covariance in local coordinates	A.27
geoidh()	Geoid height	A.28
	File Input/Output functions	
readrnx()	Read RINEX files	A.29
	OTF Integer Ambiguity resolution	
lambda()	LAMBDA/MLAMBDA integer least-square estimation	A.30
	Navigation functions	
eph2pos()	Satellite ephemeris to satellite position/clock-bias	A.31
satpos()	Satellite positions/clock-biases	A.32
geodist()	Geometric distance	A.33
satazel()	Satellite azimuth/elevation angle	A.34
	Ionospheric/Tropospheric models	
ionmodel()	Ionospheric model	A.35
tropmodel()	Tropospheric model	A.36
	Positioning solutions	
pntpos()	Single point positioning	A.37
rtkpos()	RTK positioning	A.38

*1 RTKLIB ver.1.1 マニュアル項番

付録 B RTKLIB ver.1.1 コマンド仕様

rnx2rtkp

SYNOPSIS

```
rnx2rtkp [option ...] file file [...]
```

DESCRIPTION

Read RINEX OBS/NAV files, compute receiver (rover) positions and output position solutions. The first RINEX OBS file shall contain receiver (rover) observations. For the relative mode, the second RINEX OBS file shall contain reference (base) receiver observations. At least one RINEX NAV file shall be included in input files. Command options are as follows. ([]:default).

OPTIONS

-h	Print help
-o output	Output file [STDOUT]
-p mode	Positioning Mode (0: SINGLE, 1: DGPS, 2: KINEMATIC, 3: STATIC) [2]
-m mask	Elevation mask angle (deg) [10]
-f freq	Number of frequencies for relative mode (1: L1, 2: L1+L2) [2]
-v thres	Validation threshold for Integer Ambiguity Resolution (0: No AR) [3]
-b	Backward analysis solutions [off]
-c	Forward/backward combined solutions [off]
-i	Instantaneous Integer Ambiguity Resolution [off]
-e	Output X/Y/Z-ECEF position [Latitude/Longitude/Height]
-n	Output NMEA-0183 GGA sentence [off]
-g	Output latitude/longitude in the form of ddd mm ss.ss [ddd.ddd]
-t	Output time in the form of yyyy/mm/dd hh:mm:ss.ss [sssss.ss]
-u	Output time in UTC [GPST]
-d col	Columns of time under decimal point [3]
-s sep	Field separator [' ']
-r x y z	Reference (base) receiver ECEF position (m) [Average of single pos]

EXAMPLES

Example 1. Kinematic Positioning, L1+L2, output Latitude/Longitude/Height to STDOUT.

```
> rnx2rtkp 07590920.05o 30400920.05o 30400920.05n

% program      : rnx2rtkp ver.1.0
% inputs       : 07590920.05o 30400920.05o 30400920.05n
% obs start    : 2005/04/02 00:00:00.0 GPST (gpsweek1316 518400.0s)
% obs end      : 2005/04/02 23:59:30.0 GPST (gpsweek1316 604770.0s)
% mode/obsv    : kinematic/L1+L2
% elev mask    : 10.0 deg
% ref pos      : 35.132062716 139.624305669 72.3338
%
% (time=GPST, lat/lon/height=WGS84/ellipsoidal, Q=1:fix,2:float,4:dgps,5:single, ns=# of sats)
% time         latitude(deg) longitude(deg) height(m) Q ns sdn(m) sde(m) sdu(m)
518400.000     35.160871612 139.613842087 66.8062 1 7 0.0072 0.0054 0.0164
518430.000     35.160871607 139.613842115 66.7987 1 7 0.0072 0.0054 0.0164
518460.000     35.160871593 139.613842110 66.7999 1 7 0.0072 0.0054 0.0163
...
```

Example 2. Single Point Positioning, El Mask=15deg, output NMEA GGA to file out .pos

```
> rnx2rtkp -p 0 -m 15 -n -o out.pos 07590920.05o 30400920.05n

$GPGGA,235947.00,35 9.6524150,N,13936.8296671,E,1,07,,34.318,M,36.181,M,,,*42
$GPGGA,000017.00,35 9.6525341,N,13936.8298278,E,1,07,,33.808,M,36.181,M,,,*47
$GPGGA,000047.00,35 9.6524354,N,13936.8299014,E,1,07,,33.717,M,36.181,M,,,*4F
$GPGGA,000117.00,35 9.6522549,N,13936.8298201,E,1,07,,34.418,M,36.181,M,,,*4B
...
```