Cube Users Guide ver. 1.0

Shuai Liu, Yunbin Yuan

E-mail: lsnav@foxmail.com

yybgps@asch.whigg.ac.cn

October 15, 2021

Contents

1 Overview

The Cube Software was designed and developed at Innovation Academy for Precision Measurement Science and Technology, Chinese Academy of Sciences for precise satellite clock estimation and PPP-AR. Cube is a secondary development based on RTKLIB and is designed for satellite clock estimation and PPP-AR.

The main features of Cube are as follows:

- Network process
  - Satellite clock estimation based on the ionospheric-free (IF) model
  - Satellite clock estimation based on the IF model with satellite code bias (SCB) extraction
  - Satellite clock estimation based on the decoupled clock model
- PPP process
  - PPP based on the IF model
  - PPP-AR based on the integer-recovered clock model
  - PPP-AR based on the decoupled clock model

2 Requirements

2.1 Supported platforms

Cube is developed based on Microsoft Visual Studio Community 2019, and the computer configuration requirement are as follows:

Operating system: Windows

System type: 32 or 64 bit (64 bit for network process)

Memory: at least 8G

Dependency: Intel MKL for network process

2.2 License

The Cube software is a secondary development based on RTKLIB and inherits the license of RTKLIB.

License of Cube:

and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

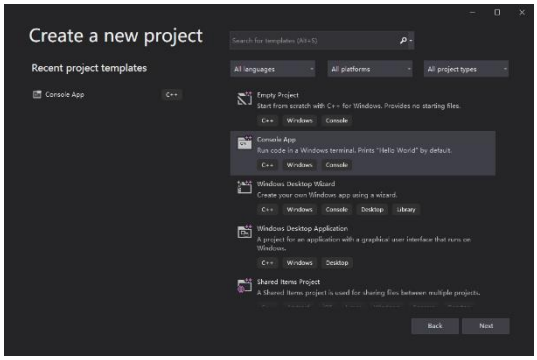
The software package includes some companion executive binaries or shared libraries necessary to execute APs on Windows. These licenses succeed to the original ones of these software.

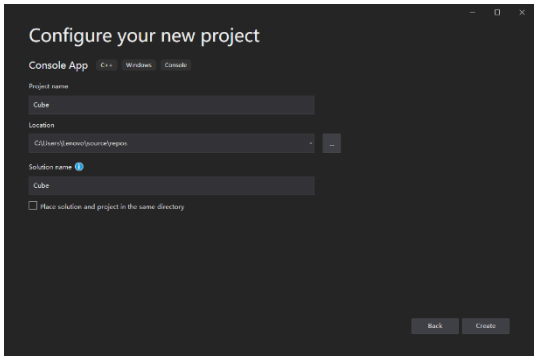
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

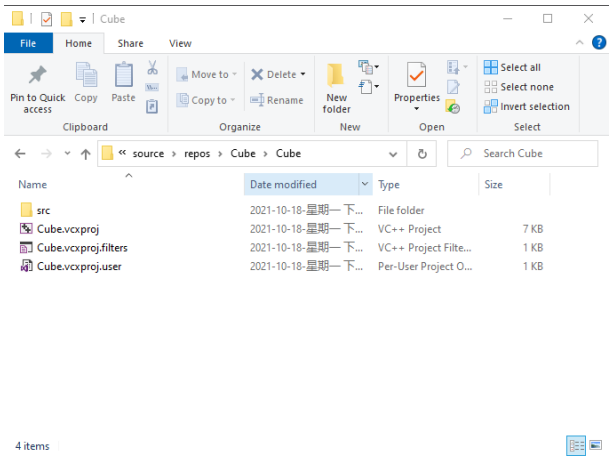
Notes:
Previous versions of RTKLIB until ver. 2.4.1 had been distributed under GPLv3 license.

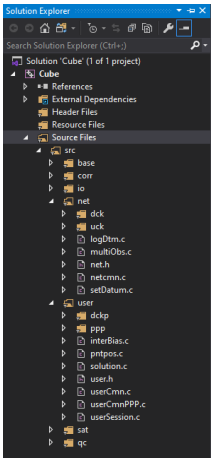
3 Installation

1. Create a new project, copy the source files to the specified directory and create folders in the project

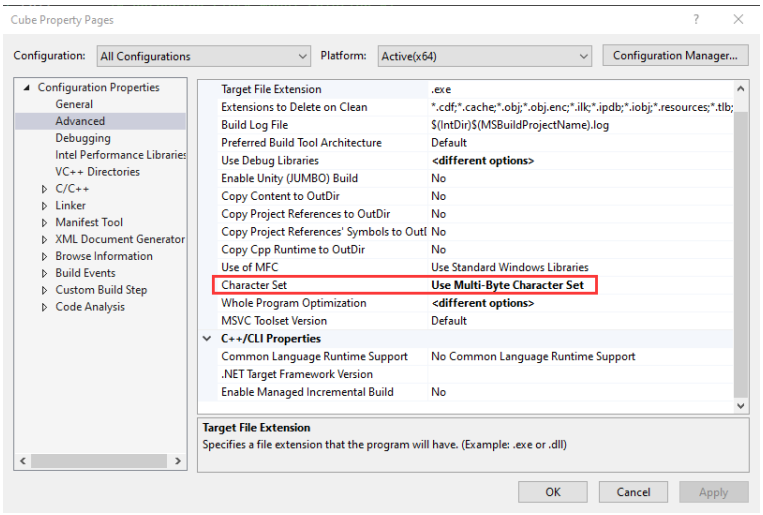






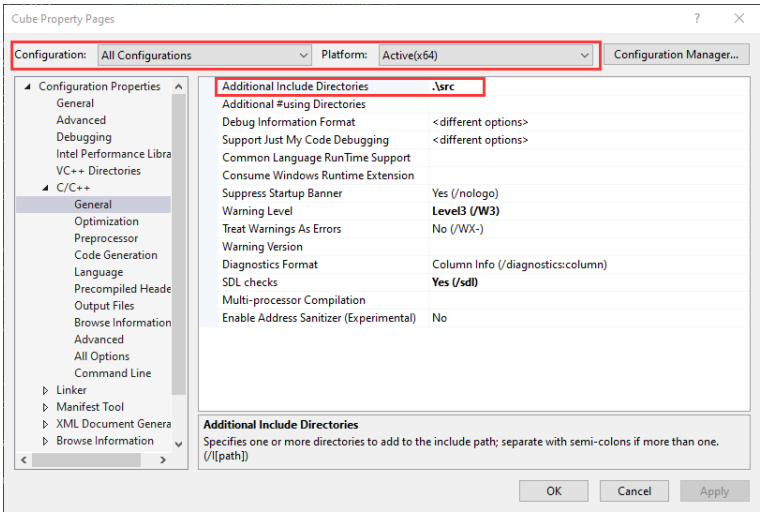


2. Configuration Properties -> Advanced, set Character Set to "**Use Multi-Byte Character Set**"



3. Add "**.\src**" to Configuration Properties -> C/C++ -> General -> Additional Include

Directories



4. Add commands to

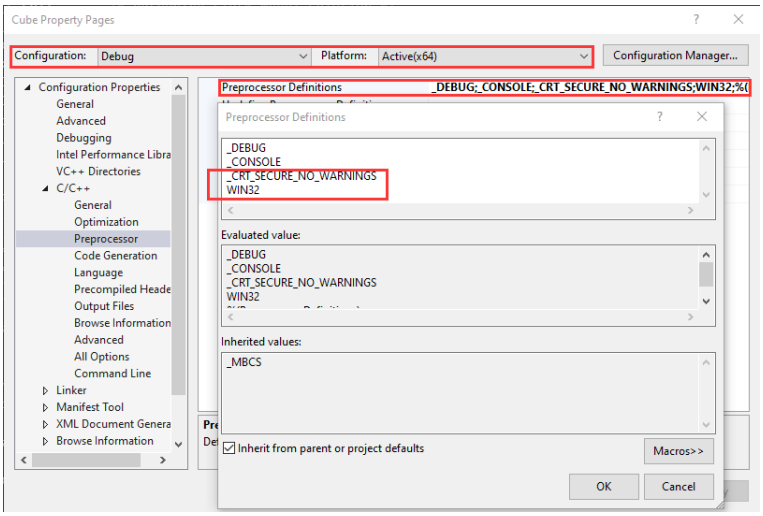
Configuration Properties -> C/C++ -> Preprocessor -> Preprocessor Definitions

Commands:

_CRT_SECURE_NO_WARNINGS

WIN32

MKL (if applied)



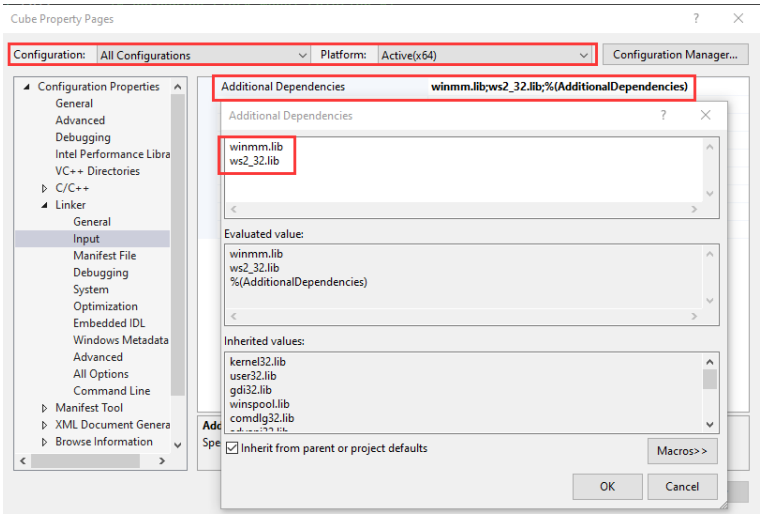
5. Add dependencies to

Configuration Properties -> Linker -> Input -> Additional Dependencies

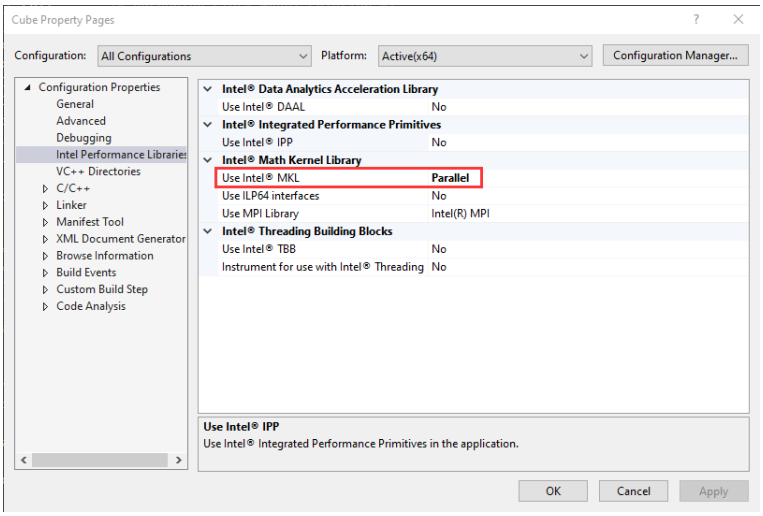
Dependencies:

winmm.lib

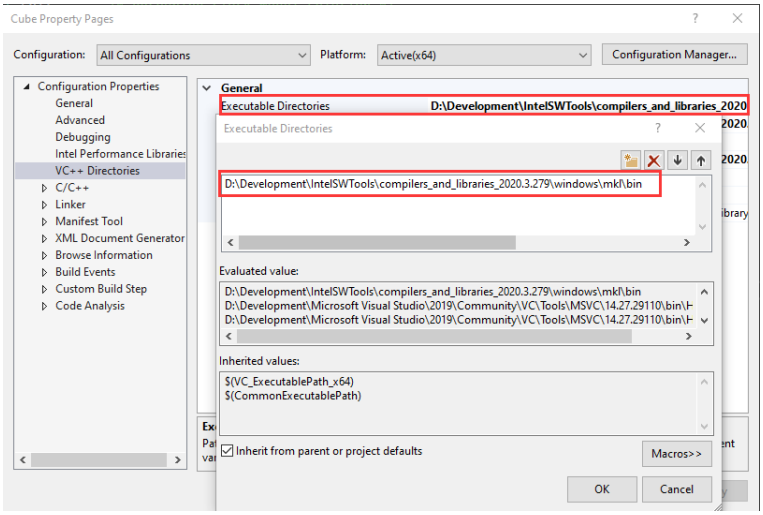
ws2_32.lib



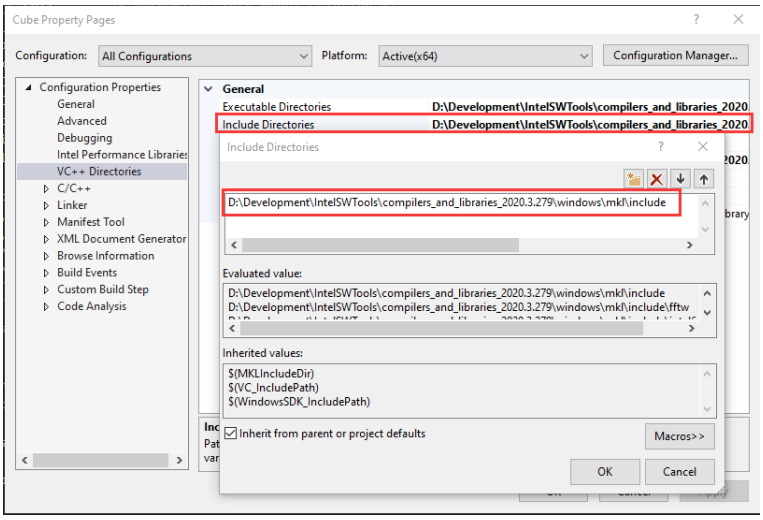
6. Install intel MKL, Configuration Properties -> Intel Performance Libraries, set Use Intel MKL to "**Parallel**"



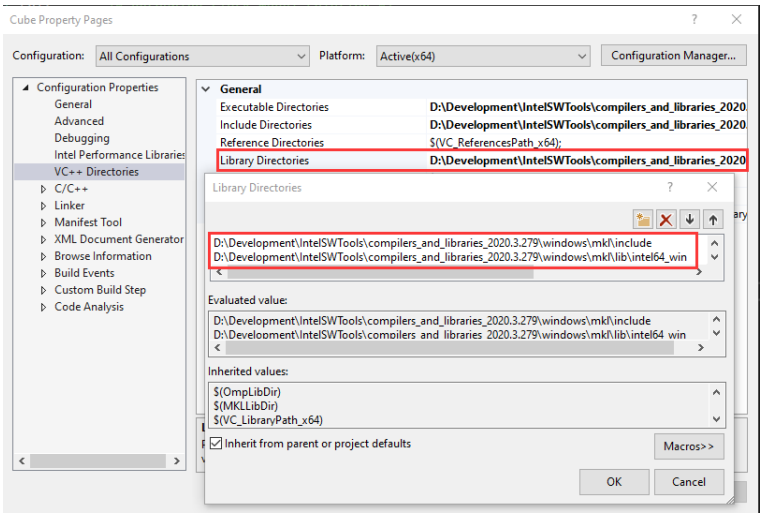
7. Add directory to Configuration Properties -> VC++ Directories -> Executable Directories (The directory added depend on your MKL installation location)



8. Add directory to Configuration Properties -> VC++ Directories -> Include Directories (The directory added depend on your MKL installation location)



9. Add directory to Configuration Properties -> VC++ Directories -> Library Directories (The directory added depend on your MKL installation location)



4 Data Processing

4.1 Structure, Strategy and Configuration

Figure 1 shows the overall structure of Cube software, and the processing strategy of Cube is summarized in Table 1. We also explain the detailed configuration information of Cube, as shown in Table 2. Then, we would like to introduce how to run the software through examples.

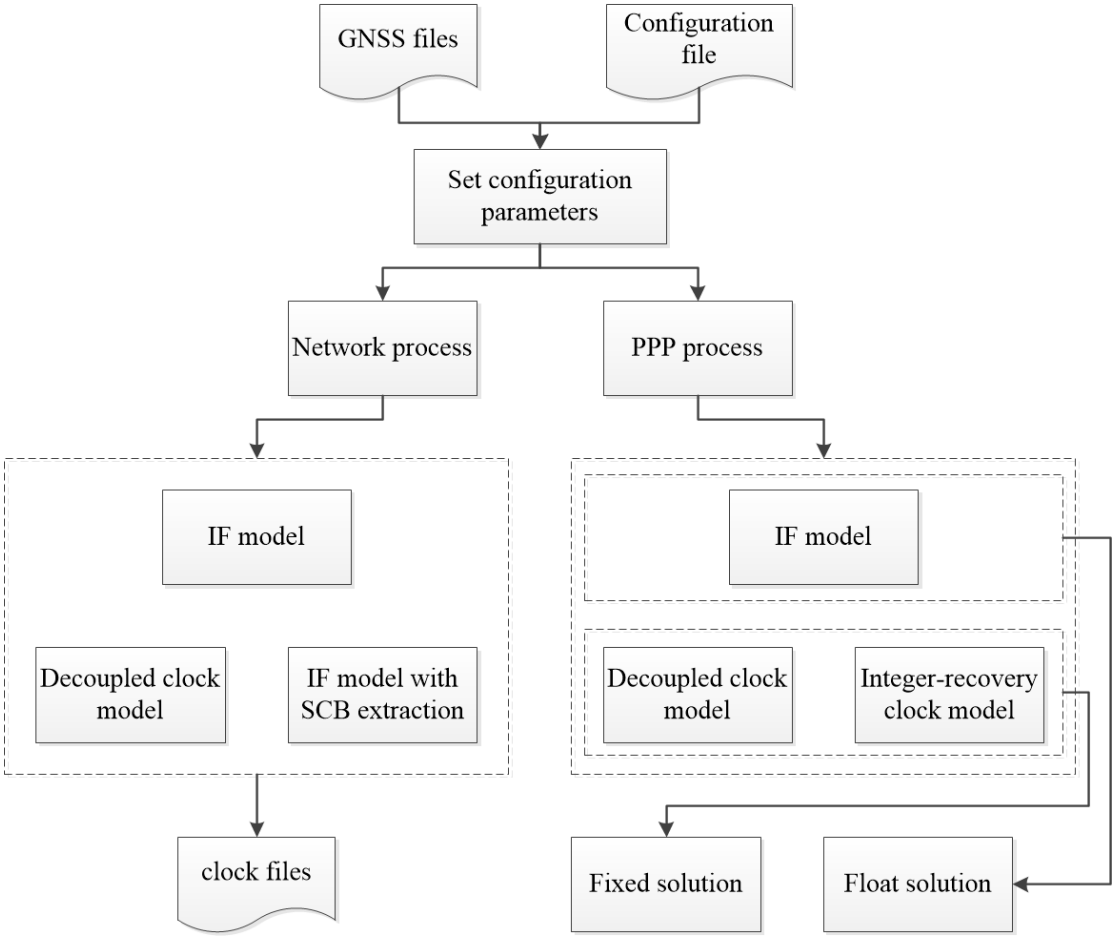


Fig 1 Overall structure of Cube software

Table 1 Summary of the processing strategy of Cube

Items	Strategies
Observations	GPS pseudorange and phase

A priori noise	Configurable (pseudorange: 0.3m, phase: 0.003m)
Station datum	Configurable (NRC1)
Cut-off elevation	Configurable (10°)
Cut-off elevation of ambiguity datum	30°
Differential code bias	CODE P1-C1 products
Relativistic effect	Corrected
Phase wind-up	Corrected
Phase center offset and variations	IGS ATX files
Tidal displacements	Solid earth tide, ocean tide loading and pole tide
Station coordinates	Fixed to IGS weekly solutions at the server end, and estimated at the user end
Satellite orbits	Configurable (IGS final products)
Earth rotation parameters	Configurable (IGS final products)
Receiver clocks	Estimated as white noises
Satellite clocks	Estimated as white noises at the server end, and fixed to IGS final products, CNES/CLS final products or the estimated products in this study at the user end
Zenith troposphere delays	Estimated as random-walk noises (configurable, $10^{-7}\ m^2/s$) with respect to Saastamoinen model (Saastamoinen 1972) and NMF is used
Horizontal troposphere gradients	Estimated as random-walk noises (configurable, $10^{-9}\ m^2/s$) or ignored
Ambiguities	Estimated as constants over each continuous session
Integer ambiguity fixing	Rounding directly at the server end (Dong and Bock 1989), rounding and LAMBDA are applied to fix ambiguity at the user end
Estimator	Least square filter

Table 2 Cube configuration options

cmn-ts	Start time, replace time of file in PPP
cmn-te	End time
cmn-dck	Use decoupled clock model or not
ppp-ti	PPP processing time interval
ppp-tu	PPP processing time unit
net-week	Network processing GPS week, replace time of file in Network
net-dow	Network processing GPS day of week, replace time of file in Network
net-netdays	The number of consecutive days to estimate the satellite clock
net-estclk	Estimate satellite clock or not in Network processing
net-inh	Use the ".ind" and ".var" files of previous day or not
net-rcb	Estimate receiver code bias or not
net-scb	Estimate satellite code bias or not
net-datum	Datum type
pos1-posmode	Processing mode
pos1-soltype	Solution type, only forward now
pos1-frequency	Frequency, only L1 and L2 now
pos1-elmask	Elevation mask
pos1-ionoopt	Ionosphere options, dual-freq now
pos1-tropopt	Troposphere options, est-ztd and est-ztdgrad now
pos1-exclsats	Excluded satellite
pos2-armode	AR or not
pos2-arelmask	AR elevation mask
pos2-rejgdop	Rejected dop
pos2-arthres	AR ratio
pos2-arthres1	AR success rate
pos2-arthres2	AR fractional part

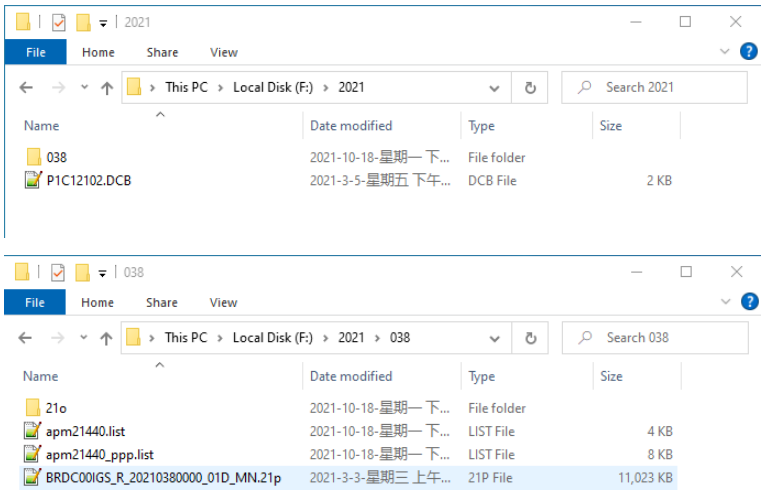
out-outstat	Output status or not
out-outtrace	Output log file
stats-eratio1	Error ratio
stats-eratio2	Error ratio
stats-errphase	Carrier-phase error
stats-errphaseel	Carrier-phase error related to elevation
stats-prnpos	Position random noise variance
stats-prntrp	Tropospheric delay random noise variance
stats-prnion	Ionospheric delay random noise variance, estimated as white noise if prn<0.0
stats-prnamb	Ambiguity random noise variance
stats-prnrpb	Receiver carrier-phase bias random noise variance
stats-prnspb	Satellite carrier-phase bias random noise variance
stats-prnrcb	Receiver code bias random noise variance, estimated as white noise if prn<0.0
stats-prnscb	Satellite code bias random noise variance, estimated as white noise if prn<0.0
ant1-postype	Position type
ant1-pos1	Antenna position
ant1-pos2	Antenna position
ant1-pos3	Antenna position
ant1-antdele	Antenna eastward offset
ant1-antdeln	Antenna northward offset
ant1-antdelu	Antenna upward offset
misc-rnxopt1	Rinex options
misc-pppopt	PPP options
file-obsfile	Observation files, support *.list (obs list file)
file-navfile	Navigation file

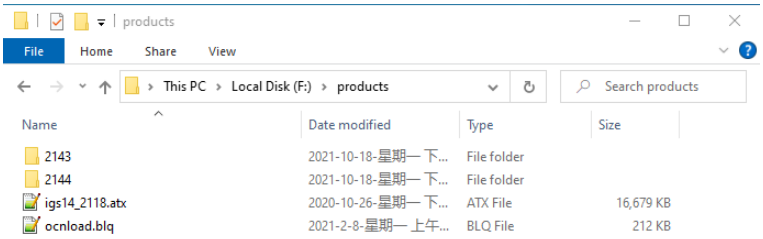
file-dcbfile	Dcb file
file-clkfile	Precise clock file
file-sp3file	Precise orbit file
file-erpfile	Erp file
file-snxfile	Snx file
file-atxfile	Atx file
file-blqfile	Blq file
file-indfile	Ind file
file-varfile	Var file
file-cobfile	Cob file
file-bisfile	Bis file

4.2 Decoupled clock estimation

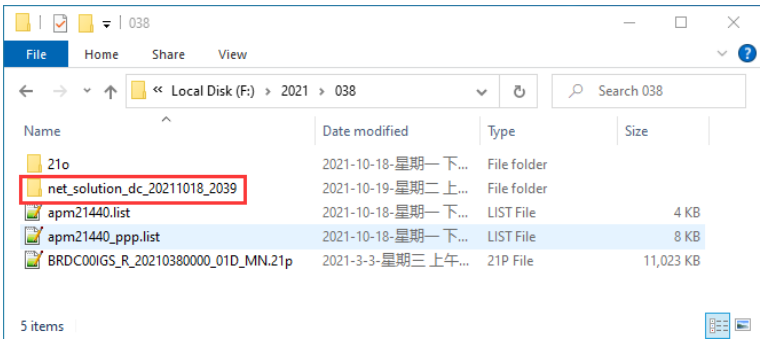
Suppose we want to generate the decoupled clock on DOY 038 in 2021, we can do it through the following steps:

1. Data preparation. We only provide two observation station list files (apm21440.list for network and apm21440_ppp.list for PPP), and other files need to be downloaded by readers.

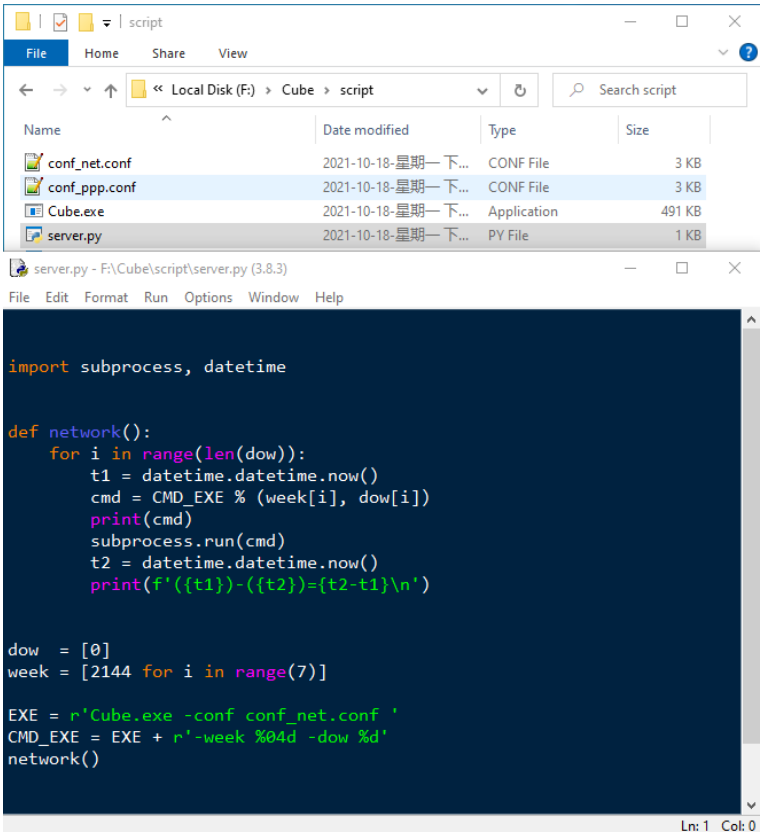




2. Read the configuration file “conf_net.conf” (Cube\script\conf_net.conf) and run the software in VS 2019.
3. Obtain network solution.

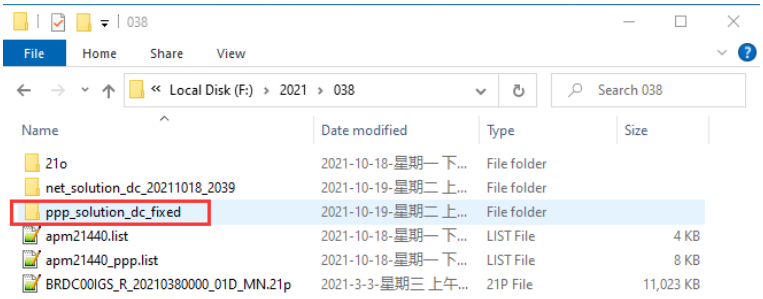


In addition, readers are also allowed to use Python scripts to directly run Cube executable file (release by yourself) for network solution.

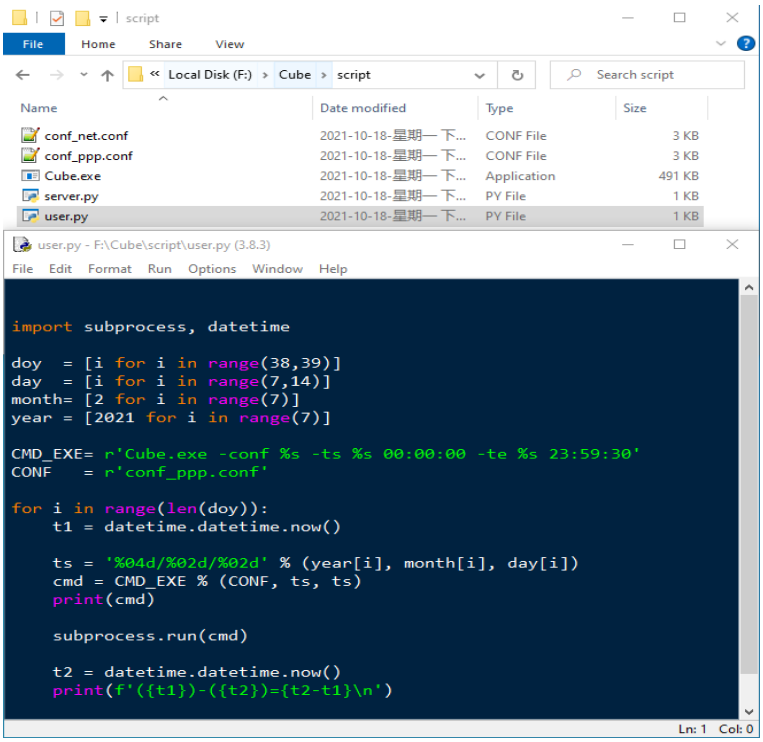


4.3 PPP Processing

1. Data preparation. We only provide two observation station list files (apm21440.list for network and apm21440_ppp.list for PPP), and other files need to be downloaded by readers. In particular, decoupled clock products are available from https://github.com/liush18/DCK.
2. Read the configuration file “conf_ppp.conf” (Cube\script\conf_ppp.conf) and run the software in VS 2019.
3. Obtain PPP solution.



In addition, readers are also allowed to use Python scripts to directly run Cube executable file (release by yourself) for PPP solution.



5 File Formats

5.1 Network solution

5.1.1 Decoupled clock file

The format of the decoupled clock file is basically the same as that of the IGS clock file, and only slightly different in the data part. The format of the data part is as follows:

Description	Format
Clock data type (AR, AS)	A2, 1X
Receiver or satellite name	A4, 1X
GPS time, year, month, day, hour, min second	I4, 4I3, F10.6
Number of data values to follow	I3, 3X
Pseudorange/Phase/Wide-lane clock	E19.12, X, E19.12, X, E19.12, X
Pseudorange/Phase/Wide-lane clock sigma	E19.12, X, E19.12, X, E19.12

5.1.2 Zenith troposphere delay file

The format of zenith troposphere delay file (.ztd) format is as follows:

TIME	GPS time, Year/month/day hour:min:second
RCV	Receiver name
ZTD	Zenith troposphere delay (m)
GRAD_N	North gradient, optional
GRAD_E	East gradient, optional
STD_ZTD	Zenith troposphere delay sigma
STD_GN	North gradient sigma, optional

STD_GE	East gradient sigma, optional

5.1.3 Ambiguity file

The format of ambiguity file (.amb) format for decoupled clock model is as follows:

TIME	GPS time, Year/month/day hour:min:second
RCV	Receiver name
SAT	Satellite ID
NARR	Narrow-lane ambiguity (cycle)
WIDE	Wide-lane ambiguity (cycle)
STD_N	Narrow-lane ambiguity sigma (cycle)
STD_W	Wide-lane ambiguity sigma (cycle)
DTM	Ambiguity datum flag, 0: No observation 1: Excluded 2: Estimated 3: Ambiguity datum for separating satellites' clocks 4: Ambiguity datum for separating receivers' clocks 5: Transmitted fixed ambiguity
OUTC	Receiver out of lock count

5.1.4 Ambiguity datum file

The ambiguity datum file (.dtm) gives the position of the station, the satellite position of each epoch, and the ambiguity datum flag formed by the satellite and station. To avoid repetition, it's not described here.

5.1.5 Residual file

The residual file has the extension ". res". The file format is as follows:

TIME	GPS time
RCV	Station name
SAT	Satellite ID
CODE	Pseudorange residuals
PHASE	Carrier-phase residuals
WIDE_LANE	MW equation residuals
AZ	Azimuth
EL	Elevation
DTM	Datum flag
OUTC	Satellite out of lock count

5.1.6 Estimated parameter state and variance files

These two files are used to record the state parameters of the last epoch of the network solution to constrain the subsequent network solution. The file with extension ".ind" records the status parameters, and the file with extension ".var" records the variances of parameters. The variance file corresponds to the parameters of the status file one by one. The format of the status file is as follows:

#TIME [time]

#TRP [station name] [type] [value]

#AMB [station name] [satellite id] [frequency number] [datum flag] [value]

The [type] in #TRP means:

0	Tropospheric zenith total delay
1	East gradient
2	North gradient

5.2 PPP solution

5.2.1 Position file

This file (.pos) has the same format as the file of RTKLIB. To avoid repetition, it's not described here.

5.2.2 Receiver clock file

The format of decoupled receiver clock file (.rclk) format is as follows:

TIME	GPS time
SOL	Solution type
RCLK_CODE	Receiver pseudorange clock (s)
RCLK_PHASE	Receiver carrier-phase clock (s)
RCLK_WL	Receiver wide-lane bias (cycle)
CODE_STD	Receiver pseudorange clock sigma (s)
PHASE_STD	Receiver carrier-phase clock sigma (s)
WL_STD	Receiver wide-lane bias sigma (cycle)
RCLK_CODE_FIX	Receiver fixed pseudorange clock (s)
RCLK_PHASE_FIX	Receiver fixed carrier-phase clock (s)
RCLK_WL_FIX	Receiver fixed wide-lane bias (cycle)
CODE_STD_FIX	Receiver fixed pseudorange clock sigma (s)
PHASE_STD_FIX	Receiver fixed carrier-phase clock sigma (s)
WL_STD_FIX	Receiver fixed wide-lane bias sigma (cycle)

The format of receiver clock file (.rclk) format is as follows:

TIME	GPS time
SOL	Solution type

RCLK	Receiver clock (s)
RCLK_STD	Receiver clock sigma (s)
RCLK_FIX	Receiver fixed clock (s)
RCLK_FIX_STD	Receiver fixed clock sigma (s)

5.2.3 Zenith troposphere delay file

The format of Zenith troposphere delay file (.ztd) format is as follows:

TIME	GPS time
SOL	Solution type
ZTD	Zenith troposphere delay (m)
GN	North gradient, optional
GE	East gradient, optional
ZTD_STD	Zenith troposphere delay sigma
GN_STD	North gradient sigma, optional
GE_STD	East gradient sigma, optional
ZTD_FIX	Fixed zenith troposphere delay (m)
GN_FIX	Fixed north gradient, optional
GE_FIX	Fixed east gradient, optional
ZTD_STD	Fixed zenith troposphere delay sigma
GN_STD	Fixed north gradient sigma, optional
GE_STD	Fixed east gradient sigma, optional

5.2.4 Ambiguity file

The format of ambiguity file (.amb) format for decoupled clock model is as follows:

GPS time	
SAT	Satellite ID
AR	Wide-lane ambiguity AR flag
WL_AMB	Wide-lane ambiguity (cycle)
STD	Wide-lane ambiguity sigma (cycle)
WL_AMB_FIX	Wide-lane fixed ambiguity (cycle)
AR	Narrow-lane ambiguity AR flag
NL_AMB	Narrow-lane ambiguity (cycle)
STD	Narrow-lane ambiguity sigma (cycle)
NL_AMB_FIX	Narrow-lane fixed ambiguity (cycle)
AZIMUTH	Azimuth (degree)
ELEVATION	Elevation (degree)

The format of ambiguity file (.amb) format for integer-recovered clock model is as follows:

GPS time	
SAT	Satellite ID
WL_AMB(c)	Wide-lane ambiguity (cycle)
WL_N	Number of wide-lane ambiguity
WL_SDAMB(c)	Single difference wide-lane ambiguity between satellites (cycle)
STD	Single difference wide-lane ambiguity between satellites sigma (cycle)
WL_SDAMB_FIX	Fixed single difference wide-lane ambiguity between satellites (cycle)
NL_SDAMB(c)	Single difference narrow-lane ambiguity between satellites (cycle)
STD	Single difference narrow-lane ambiguity between satellites

	sigma (cycle)
NL_SDAMB_FIX	Fixed single difference narrow-lane ambiguity between satellites (cycle)
IF_AMB(m)	Ionosphere free ambiguity (m)
STD	Ionosphere free ambiguity sigma (m)
IF_AMB_FIX	Fixed Ionosphere free ambiguity (m)
AZI(deg)	Azimuth (degree)
ELE(deg)	Elevation (degree)

5.2.5 Residual file

The format of residual file (.res) format is as follows:

GPS time	
SAT	Satellite ID
P_IF(m)	Pseudorange residuals
L_IF(m)	Carrier-phase residuals
WL(c)	MW equation residuals, ignored for the conventional model
AZI(deg)	Azimuth
ELE(deg)	Elevation
OUT	Satellite out of lock count

6 Support

We would like to acknowledge T. Takasu for his open-source software RTKLIB.

RTKLIB: www.rtklib.com

Any bugs and suggestions about Cube are sincerely welcomed and could be sent to:

Shuai Liu

E-mail: lsnav@foxmail.com

Github: https://github.com/liush18

微信公众号：魔方的块

CSDN Blog：学问思辨行_魔方的块（https://blog.csdn.net/Pro2015）