

# PRSolve ver. 4.0 User's Guide

## July, 2011

*Dr. Brian W. Tolman  
Applied Research Laboratories  
The University of Texas at Austin*

**Introduction.** PRSolve is an application of the GPS Toolkit ([www.gpsstk.org](http://www.gpsstk.org)) that computes a navigation (position and clock) solution from the pseudorange data of several satellites contained in RINEX (version 2 or 3) format observation files. A Receiver Autonomous Integrity Monitoring or RAIM algorithm is included in the processing but may optionally be omitted. The solution is independently computed at each time tag; there is no filtering or other connection between solutions at different times.

**Basics, and the syntax page.** When PRSolve is run without any command line arguments, or with the --help option, it produces a "help" or "syntax page" (Appendix A). Almost all the information needed to run PRSolve is found on the syntax page. **This documentation is meant as a supplement to, not a replacement for, the syntax page; it assumes you are familiar with the syntax page.** (Note that the syntax page is likely to be updated more frequently than this document.)

The syntax page begins with a brief description of the program, then describes the records found in its output, then finally lists each option and its argument (if it has one) followed by a brief description, with the default value in parentheses. Most options have no default and so the parentheses are empty "()". Also "[repeat]" may appear, meaning the option is allowed to appear more than once on the command line (otherwise an error is generated). Options may appear in any order on the command line. Also, options may be placed in a text file, and then applied together with the single option '--file <filename>' (comments beginning with # are also allowed in this file).

The *minimum* required input consists of a RINEX observation file (--obs) and corresponding ephemeris input (--eph or --nav). All other input is optional, however note that specifying a "solution descriptor" (see below) appropriate for the input data, is probably a good idea. Input RINEX clock files (--clk) will override the clock information in the ephemeris input. The path of the input files may be specified separately from the file name; that is, the following lines of input

```
--eph /home/user/eph/igs15505.sp3 --eph /home/user/eph/igs15506.sp3
--eph igs15505.sp3 --eph igs15506.sp3 --ephpath /home/user/eph
--eph igs15505.sp3,igs15506.sp3 --ephpath /home/user/eph
```

are all equivalent. Note that the two file names may appear either with their own option (--eph), as in the second line, or separated by a comma following a single option, as in the third line. If the data in these files does not match the time tags in the RINEX observation file, a Warning or Error will be issued in the log file. It is the user's responsibility to ensure that these input files are consistent, i.e. that they are in the same frame (ITRF, WGS84, etc.) and they are appropriate to the data.

**Solution Descriptors.** PRSolve can simultaneously compute more than one solution at each time tag. The satellite systems and data types used in each solution are determined by the "solution descriptor," and are input with the --sol <descriptor> option (see syntax page); this string also labels the output. The solution descriptor is a string of the form "GPS:12:WC" consisting of three parts, system(s),

frequency(ies) and code(s), separated by colons (:). The data used to compute the solution consists of all the pseudoranges in the data from the given system, frequency and code.

"System" means satellite system or GNSS; namely GPS, GLONASS, Galileo, etc.; see --sys on the syntax page for supported systems. There may be more than one system separated by a '+', meaning data from all satellites in these systems will be combined. If more than one frequency appears, then an ionosphere-free linear combination of data from the two frequencies is formed; specifically frequency "12" implies the usual linear combination of L1 and L2 that eliminates the ionospheric delay. (Currently only the frequencies 1,2,5,12,15,25 are supported.)

The codes in the descriptor consist of characters which give RINEX tracking codes (cf. e.g. the log file produced by PRSolve --verbose, or RinSum output, for your data file). The codes input is *ordered*, meaning that PRSolve will choose the *first* code in the --code string (moving left to right) that is found in the data. If there is more than one system (separated by a '+') then there must be corresponding code strings also separated by a '+'.

Multiple solution descriptors are given by multiple --sol options (--sol is repeatable), or as a single argument, separated by commas. For example, input of either

```
--sol GPS+GLO:12:PYXWC+PC --sol GPS:15:WXL
```

or

```
--sol GPS+GLO:12:PYXWC+PC,GPS:15:WXL
```

will cause PRSolve to compute two solutions, the first being a combination of GPS and GLONASS data, using an ionosphere-free combinations of L1 and L2, with tracking codes P,Y,X,W, or C for GPS, and codes P or C for GLONASS. The second solution produced by PRSolve will be GPS-only, L1/L5 ionosphere-free combination, using whichever pseudoranges with codes W,X,L or C (in that order) are found in the data. (Note that the actual code chosen at each epoch is given in the DAT output line; see below.)

An alternate way of specifying the solution descriptors is to NOT input any --sol options (note that this is the PRSolve default, since --sol has no default) and use --sys, --code and --freq instead. This causes ALL solution descriptors built from combinations of --sys, --freq and --codes input to be used. Use this technique with care; it is easy to generate a very large output and a long processing time this way. The defaults are (see the syntax page)

```
--sys GPS,GLO,GPS+GLO
```

```
--freq 12
```

The option --code has no default, however default codes are built into PRSolve and will be provided to a solution descriptor, where necessary, based on the systems involved; these default codes for each system are given on the syntax page. Thus the default set of solution descriptors is the same as

```
--sol GPS:12:PYMIQSLXWCN
```

```
--sol GLO:12:PC
```

```
--sol GPS+GLO:12:PYMNIQSLXWC+PC
```

**Configuring the algorithm.** PRSolve uses an iterated linearized least squares algorithm to compute the navigation (position + clocks) solution. The user is able to configure this algorithm using command line input as follows (see the syntax page starting with # Solution Algorithm). The defaults for these options are carefully chosen based on a lot of experience, and should be changed only with great care.

The option --wt causes the pseudorange measurements to be weighted based on their elevation angle; the weight is defined as

wt = 1 (elev >= 30 degrees)  
wt = [sin(elev)/sin(30)]^2 (elev < 30 deg)

Of course elevations below zero are excluded. This option is quite expensive in terms of computation time.

The options --niter and --conv control the iteration loop; --niter gives the maximum number of iterations allowed (default 10), and --conv gives a limit on the RMS change in the position solution. If the number of iterations is exceeded then iteration stops and the solution is marked bad. If the change is less than the --conv limit then iteration stops and the solution is marked good. (The RMS output shows the actual values produced by the algorithm; see below.)

The RAIM algorithm used by PRSolve ("A Baseline GPS RAIM Scheme and a Note on the Equivalence of Three RAIM Methods," by R. Grover Brown, Journal of the Institute of Navigation, Vol. 39, No. 3, Fall 1992, pg 301) operates simply by computing many different solutions with different satellites until a good solution is found. "Good solution" is defined as one with post fit RMS residual less than the limit given by the --rms option. If all the data does not produce a good solution, then the algorithm computes solutions with rejected satellites, first each satellite rejected in turn, and then with pairs of satellites rejected in turn, and so on, until either a good solution is found, a given maximum number of satellites is rejected, or there is not enough data to proceed. This limit on the number of satellites that may be rejected by the RAIM algorithm is given by the --nrej option; if this limit is -1, then no limit is applied. (The RMS output gives the number of rejected satellites and which satellites were rejected, in the final RAIM solution; see below.)

The least squares algorithm yields an RMS post-fit pseudorange residual; if this value exceeds the limit given by --rms then the solution is marked suspect. The RAIM algorithm also produces a quantity called 'slope'; if this value exceeds the limit given by --slope then the solution is marked suspect.

**Tropospheric model and weather input.** PRSolve corrects the measured pseudoranges for tropospheric delay using a standard model and weather parameters. The tropospheric model and default weather parameters may be input using the --Trop option. Weather information in the form of RINEX meteorological files may be input with the --met option. If no meteorological files are input, the default weather parameters are used.

The argument of the --Trop option is a string of 4 fields separated by commas: a key giving the model name, and three numbers for the temperature (degrees C), pressure (millibars) and relative humidity (%). The default for this option is 'NewB,20.0,1013.0,50.0'. The accepted keys with the corresponding model are (also see syntax page)

Zero	This is a "dummy" model that always returns a zero correction; used for testing
Black	A simple Black model
Saas	The Saastamoinen model
NewB	New Brunswick UNB3 model
Neill	Neill model
GG	The model of Goad and Goodman
GGHt	The GG model with an explicit height dependence

Some of these models (Saas, Neill, NewB) require the height (altitude) and latitude of the receiver and

the day of year; PRSolve will give it this information when it has a reference position (--ref) and a start or stop time (--start or --stop). Note that the NewB model computes its own weather parameters given the latitude and day of year. For more information on these tropospheric models, see class TropModel in the GPS Toolkit.

**Reference position input.** The --ref option gives PRSolve a "reference" position for the solution; this should be the known "correct" position, or at least a good guess. The reference position is not used in the computation and does NOT affect the computed solution, but it does allow PRSolve to compute residuals (the difference between the solution position and this reference position), and to rotate the solution residuals into local topocentric (North-East-Up) coordinates (see RPR and RNE output records, below). The argument for --ref is just the position coordinates separated by commas; thus for example

```
--ref -740311.8581,-5457066.4731,3207249.3343
```

is acceptable; this assumes the coordinates are Earth-centered, Earth-fixed Cartesian (ECEF XYZ) with units meters. The --ref argument must specify a *complete* geodetic position.

Other coordinates and/or units may be used, by supplying a format description, after a colon (:), along with the corresponding coordinates (--ref <data:format>, see the syntax page); the meanings of the format descriptors come from class Position in the GPS Toolkit; they are as follows. Note that the default format is "%x,%y,%z".

```
%x  Position::X() (meters)
%y  Position::Y() (meters)
%z  Position::Z() (meters)
%X  Position::X()/1000 (kilometers)
%Y  Position::Y()/1000 (kilometers)
%Z  Position::Z()/1000 (kilometers)
%A  Position::geodeticLatitude() (degrees North)
%a  Position::geocentricLatitude() (degrees North)
%L  Position::longitude() (degrees East)
%l  Position::longitude() (degrees East)
%W  Position::longitude() (degrees West)
%w  Position::longitude() (degrees West)
%t  Position::theta() (degrees)
%T  Position::theta() (radians)
%p  Position::phi() (degrees)
%P  Position::phi() (radians)
%r  Position::radius() meters
%R  Position::radius()/1000 kilometers
%h  Position::height() meters
%H  Position::height()/1000 kilometers
```

(On Windows, '%' is a special character on the command line; make it an ordinary character by entering it twice, e.g. "%x,%y,%z".)

Thus, other possibilities for the --ref argument above would be

```
--ref -740311.8581,-5457066.4731,3207249.3343:%x,%y,%z # the default format
--ref -740311.8581,-5457066.4731,3207249.3343:%x,%y,%z # Windows version of above
--ref 30.216754900,97.726379400,6372918.624:%a,%w,%r # geocentric LL(W)H
--ref 30.384393671,262.273620600,218.412:%A,%l,%h # geodetic LL(E)H
--ref 59.783245100,262.273620600,6372918.624:%t,%p,%r # spherical th,phi,r
```

**Other optional input.** PRSolve accepts some options that edit the data in various ways. PRSolve will ignore data in the input RINEX observation file that are outside time limits given by --start and --stop, each followed by a time. The time argument may be given as "week,seconds-of-week" or as "year,month,day,hour,minute,seconds", or as another form given by a format included after a colon (:) in the argument. The format descriptors are taken from class Epoch in the GPS Toolkit; the defaults are '%F,%g' and '%Y,%m,%d,%H,%M,%S'. Here are some examples.

```
--stop 2009,9,23,18,1,2 --start 1550,280800
--start 1550,299999.000 --stop 2009,9,23,12,0,1
--start 1550,280800.:%F,%g --stop 55098.23487024:%Q
--stop "2009 9 23 18 1 2:%Y %m %d %H %M %S"
```

Note that there must be quotes around any argument that include spaces. Also note that the format does not have to include commas; however the data before the colon must fit into the format after the colon. Again, on Windows you must double up the %, for example "%F %g".

The input observation data can be decimated down to time spacing dt by specifying --decimate dt. For example, if the RINEX observation file has an interval of 30 seconds but you want to compute solutions only every 5 minutes (300 seconds), the input '--decimate 300' would cause PRSolve to do that. Note that the start time of the decimation is determined by the --start option, or if that is not available, then seconds-of-week that are even multiples of dt are used. In other words,

```
--decimate 300
```

alone produces results at seconds-of-week 0, 300, 600, ... 345000, 345300, 345600, ... 604500 (as they exist in the data); but if you had

```
--decimate 300 --start 1550,345030.0
```

then PRSolve would produce solutions at seconds-of-week 30, 330, ..., 345030, 345330, 345630, ... (as they exist in the data).

The option --elev <deg> forces satellites below elevation angle <deg> to be eliminated. This option requires that a reference position (--ref) be given; then the elevation angle is computed, independent of, and before, the solution algorithm, using the reference position for the receiver. If --ref is not given but --forceElev is given, then PRSolve uses the latest computed position to compute the elevation angle, and then applies the mask. Note that with --forceElev there may be some time tags initially when there is no solution and so the mask cannot be applied; also if a solution is very poor then the masking may consequently be inappropriate, but this should not happen with a good amount of data.

Individual satellites may be excluded from the data stream using the option --exSat. The argument of this option is a RINEX-style satellite identifier; for example

```
--exSat R9 --exSat G01      # option is repeatable
--exSat R9,G01,G17          # comma-separated list works
--exSat R                    # exclude entire system R (GLONASS)
```

The last example excludes ALL GLONASS (R) satellites (although this command is redundant if the solution descriptors do not contain the system GLO). [Note that the supported systems and their

RINEX identifiers are GPS:G GLONASS:R Galileo:E SBAS or Geocentric:S and Compass:C.]

**Computation of ORDs.** An Observed Range Deviations (ORD) is simply the difference of the measured pseudorange and the pseudorange computed using the position+clock solution. PRSolve will compute ORDs and write them to a file named in the argument of the option --ORDs if such an option is input. This requires that a reference position (--ref) be given. The format of the output file is as follows (there were three solutions computed, hence three sets of ORDs).

```
ORD sat week  sec-of-wk  elev  iono      ORD1      ORD2      ORD Solution_descriptor
ORD G01 1550 259200.000 18.795 8.022    8.928    14.118    0.906 GPS:12:WC
ORD G02 1550 259200.000 71.229 4.297    9.568    12.348    5.271 GPS:12:WC
ORD G04 1550 259200.000 46.999 9.135    10.853   16.763    1.718 GPS:12:WC
(... the rest of the satellites at this time, solution)
ORD R07 1550 259200.000 20.925 18.298   36.885   48.835   18.587 GLO:12:PC
ORD R10 1550 259200.000  8.977 35.663   46.160   69.450   10.497 GLO:12:PC
ORD R11 1550 259200.000 27.755 25.970   40.286   57.246   14.316 GLO:12:PC
(... the rest of the satellites at this time, solution)
ORD G01 1550 259200.000 18.795 8.022     9.281    14.471    1.258 GPS+GLO:12:WC+PC
ORD G02 1550 259200.000 71.229 4.297     9.921    12.701    5.624 GPS+GLO:12:WC+PC
ORD G04 1550 259200.000 46.999 9.135    11.206   17.116    2.071 GPS+GLO:12:WC+PC
ORD R07 1550 259200.000 20.925 18.298   24.070   36.020    5.771 GPS+GLO:12:WC+PC
ORD R10 1550 259200.000  8.977 35.663   33.344   56.634   -2.319 GPS+GLO:12:WC+PC
ORD R11 1550 259200.000 27.755 25.970   27.471   44.431    1.501 GPS+GLO:12:WC+PC
(... the rest of the satellites at this time, solution)
ORD G01 1550 259210.000 18.859 7.899     7.496    12.606   -0.403 GPS:12:WC
ORD G02 1550 259210.000 71.194 5.024     6.431     9.681    1.407 GPS:12:WC
ORD G04 1550 259210.000 46.932 9.104     7.740    13.630   -1.364 GPS:12:WC
ORD G05 1550 259210.000 34.404 7.960    10.758   15.908    2.797 GPS:12:WC
ORD G09 1550 259210.000 17.533 12.227    12.650   20.560    0.423 GPS:12:WC
(...)
```

The first line in the file is a header that identifies the columns. Note that all lines start with the tag 'ORD', the satellite ID (RINEX style) and the time in week and seconds-of-week. This is followed by the elevation angle of the satellite in degrees, the ionospheric delay (at L1, the first frequency) in meters. There are three ORDs computed, the first, ORD1, is computed using the pseudorange at the first frequency (L1 here – note the solution descriptor) while ORD2 uses the second. The third one, labelled ORD, uses the ionosphere-free combination of the two measured pseudoranges. Finally the line includes the solution descriptor.

**Output to a RINEX file.** The option --out <filename> causes PRSolve to write an output RINEX (ver. 3.01) observation file. This file will be the same as the input file except for small changes in the header and the addition of auxiliary header information, at each time tag, in the form of comments that contain the PRSolve navigation solution(s). That is, the output RINEX file will include, at each epoch, a block like the following.

```
> 2009 09 23 00 02  0.0000000  4  9
XYZ  -740312.118 -5457067.840 3207248.661 GPS:12:WC      COMMENT
CLK GPS  -19598.442 GPS:12:WC      COMMENT
DIA11 2.57 3.25      2.07 GPS:12:WC      COMMENT
XYZ  -740313.050 -5457053.843 3207245.768 GLO:12:PC      COMMENT
CLK GLO  -19654.477 GLO:12:PC      COMMENT
DIA 5 35.21 43.21      1.14 GLO:12:PC      COMMENT
XYZ  -740312.884 -5457067.601 3207248.781 GPS+GLO:12:WC+PC COMMENT
CLK GPS  -19598.289 GLO  -19648.020 GPS+GLO:12:WC+PC      COMMENT
DIA16 2.18 3.40      2.05 GPS+GLO:12:WC+PC      COMMENT
```

This is a standard RINEX epoch line (> 2009 ... 4 9) with epoch flag 4 followed by 9 comment lines. Note that there are three lines for each solution descriptor. The XYZ line gives the ECEF XYZ solution in meters, followed by the descriptor. The CLK line gives the clock solution in meters, preceded by the system to which it applies, for each system included in the solution, followed by the descriptor again. Finally the DIA line includes the number of satellites, followed by the PDOP, GDOP and the RMS residual of fit and the descriptor. Note that RINEX comments are limited to 60 characters, and so it may happen that very long solution descriptors will get truncated.

**PRsolve log file output.** PRsolve produces a large amount of output to a log file (--log with default prs.log). Anything unexpected or anomalous will be noted with a line beginning with "Warning" or (if the problem is critical) "Error"; *whenever PRsolve does not give good results, the log file should be searched for these two words.*

Otherwise, there are three types of output

- 1) a dump of the input configuration and input file content, at the top of the file
- 2) solutions and diagnostic information at each time tag
- 3) statistics and a weighted average solution at the bottom of the file

The log file ends with a line starting with "PRsolve timing:" and including both the processing (CPU) time used and the "wall clock" timing; the weighted average and timing information are also printed on the screen.

The configuration is dumped to the log file in a format similar to the syntax page; this tells you exactly what PRsolve found the input to be. This is followed by various summaries and dumps of the content of the input files. The verbose option (--verbose) causes more of this information to be produced, including file headers and summaries of the ephemeris input, etc. A dump of the header of the input RINEX observation file is here; it gives the RINEX observation types for all the data in the file, which may be useful in providing codes for --sol or --code input options.

Before processing (between outputs (1) and (2)), each solution descriptor is printed in the log file after the tag 'SOLN'; if the solution descriptor is invalid for any reason, this will be indicated at this point and the solution descriptor will then be ignored in the processing.

**Output per time tag.** Each solution descriptor yields several lines of output, each with a different label, at each time tag in the input RINEX observation file. The lines are of the following form (this output is described on the syntax page; the comments below are meant to augment that description).

TAG descriptor LABEL week sec-of-week CONTENT (code) [N]V

At minimum you will get output for TAG RPF and LABELs DAT, NAV and RMS. Tag "RPF" means the RAIM final solution, which is the solution produced by the least squares algorithm. The NAV content consists of the position, in Earth-centered, Earth-fixed Cartesian coordinates (ECEF XYZ), and clock biases, with units meters. [Note that the frame (ITRF, WGS84, etc.) of the solution is determined by the frame of the input satellite ephemeris (--eph or --nav).]

For example (the line is necessarily wrapped here, even though it is a single line in the log file):

```
RPF GPS:12:WC NAV 1550 259300.000 -740311.906314 -5457065.109818 3207248.819548 GPS
-19600.074 (0 ok) V
```

This is a GPS ionosphere-free L1/L2 solution with (X,Y,Z = -740311.906314, -5457065., 3207248.819548) meters and a GPS clock bias of -19600.074 meters. There were no problems with this solution "(0 ok)" and it is valid "V". The clock bias is preceded by the system to which it applies; if more than one system (GNSS) was used in the solution, then there will be more than one clock bias; for example

```
RPF GPS+GLO:12:WC+PC NAV 1550 259300.000 -740312.591550 -5457064.512332 3207247.908676 GPS
-19600.297 GLO -19650.258 (0 ok) V
```

This is a combined GPS and GLONASS solution similar to the previous one. [The solution for two clock biases here is equivalent to solving for the GPS-GLONASS system time offset, which must be done whenever GPS and GLONASS data is mixed.]

POS records are very similar to NAV records; they simply omit the clock solution(s). These are used to output residuals (RPR or RNE, see below). CLK records output only the clock solution(s) [they may not appear in the current version].

The record labelled DAT indicates which satellites were present in the data, which were used, and which RINEX tracking code applies to the pseudorange that was actually used in the algorithm. Thus

```
RPF GPS:12:WC DAT 1550 259300.000 11 11 G01:1W2W G02:1W2W G04:1W2W G05:1W2W G09:1W2W G10:1W2W
G12:1W2W G17:1W2W G24:1W2W G27:1W2W G30:1W2W
```

shows that 11 satellites had sufficient data, and that there are 11 satellites present; on frequency L1 code W was used (1W) and on L2 it was also W (2W). If, say, GPS satellite PRN 17 had been present but code W was not available on L1 while code C was, then you would see 'G17:1C2W'. If in addition there were NO pseudorange on L2 for G17, you would see instead '-G17:1C2-' and the satellite G17 would not be part of the solution (-G17 means G17 was not used; 2- means there were no acceptable codes on L2). Also in this case '11 11' would be replaced with '10 11'.

The output labelled with RMS contains a lot of information about the operation of the solution algorithm; a labelled example is provided here (again wrapping is necessary but a bother here):

```
RPF descriptor      week sec.of.week n N      RMSresid TDOP      PDOP      GDOP      slope niter conv  sats
('-' means rejected) ... (code) Valid
RPF GPS:12:WC RMS 1550 259300.000 1 10 1.839 0.68 2.57 3.25 5.6 4 8.69e-10 G01
G02 G04 G05 G09 G10 G12 -G17 G24 G27 G30 (0 ok) V
```

Note that satellite G17 was rejected; 'n' is the number of rejected satellites (1) and 'N' is the number used (10, and there are 11=1+10 satellites listed).

Additional output is generated for each solution descriptor when the user provides the --ref option. The reference position yields two additional records with LABELs RPR and RNE. "RPR" is the RAIM solution residuals, or the difference between the solution (RPF) and the reference position; again in ECEF XYZ (meters). The RNE record is these residuals rotated into the local North-East-Up directions (again in meters).



If the option --SPSout is present, there will also be a record with TAG SPS similar to the RPF record, and if a reference position is given there will be SPR and SNE records similar to RPR and RNE. "SPS" denotes the straightforward pseudorange solution without the RAIM algorithm; this means all satellites are included and no attempt is made to edit out bad data.

The end of each record contains two items that indicate how the algorithm finished; the "return code" of the RAIM algorithm appears in parentheses, and if --verbose is present there will also be words giving its meaning. Finally each record ends with either "V" or "NV", indicating the solution is Valid or Not Valid. **RECORDS THAT END IN NV SHOULD BE IGNORED** as the solution algorithm failed in some way; they are included in the output file for diagnostic purposes. The return codes and their meanings are as follows.

- 4 PRSolve failed to find ANY ephemeris
- 3 PRSolve failed to find enough satellites with data
- 2 PRSolve failed because the problem is singular
- 1 PRSolve failed because the algorithm failed to converge
- 0 Success
- 1 The algorithm succeeded but the solution might be degraded because
  - a) a tropospheric correction could not be applied
  - b) the RMS residual is high, or
  - c) the RAIM slope is high

Code -4 suggests that you check that the ephemeris input covers the timespan of the data (see --eph or --nav). A -3 code just means there were not enough satellites at this time tag to get a good solution; this is probably the most common non-zero code. Codes -2 and -1 indicate that the algorithm failed and should be extremely rare. Code 1 means the solution may be an outlier, but it might also be good.

**Output Statistics and the Weighted Average Solution.** PRSolve keeps statistics on the residuals it computes, as well as a weighted average solution, for each solution descriptor. "Weighted average solution" here means the valid solutions at each time tag, weighted by the least squares covariance matrix, are combined to form a single solution and covariance. This yields a single "best" solution for the entire dataset; of course for a moving receiver this probably is not useful information. This solution is printed at the bottom of the log file and to the screen when PRSolve is finished. If there is a reference position given, then simple statistics on each component of the residuals (both RPR and RNE) are also given in the log file. RNE output also includes the covariance matrix, in the NEU frame.

**Examples of running PRSolve.** The following is an example command line for running PRSolve, along with some of the output (file prs.log). The operating system here is Linux; that explains the continuation lines (\) and the symbol (~). Note that there is --eph and --clk input for both GPS (igs\*.sp3 and igs\*.clk\_30s) and GLONASS (igl\*.sp3 and esa\*.clk).

```
PRSolve --verbose \  
  --obspath ~/Data/obs/Javad/R301 --obs ARL82660.09o \  
  --ephpath ~/Data/eph/igs/ \  
  --eph igs15502.sp3,igs15503.sp3,igs15504.sp3 \  
  --eph igl15502.sp3,igl15503.sp3,igl15504.sp3 \  
  --clkpath ~/Data/clk/ \  
  --clk igs15502.clk_30s,igs15503.clk_30s,igs15504.clk_30s \  
  --clk esa15502.clk,esa15503.clk,esa15504.clk \  
  --ref -740311.8581,-5457066.4731,3207249.3343:%x,%y,%z \  
  --sol GPS:12:WC,GLO:12:PC,GPS+GLO:12:WC+PC
```

Exactly the same run would result if we created a file prs.cfg, as follows, and then ran the command

```
PRsolve --file prs.cfg
```

```
# file prs.cfg, configuration file for test run of PRsolve
--verbose
--obspath ~/Data/obs/Javad/R301 --obs ARL82660.09o
--ephpath ~/Data/eph/igs/
--eph igs15502.sp3,igs15503.sp3,igs15504.sp3
--eph igl15502.sp3,igl15503.sp3,igl15504.sp3      # IGS GLO ephemeris is in ITRF
# 30-second clocks
--clkpath ~/Data/clk/
--clk igs15502.clk_30s,igs15503.clk_30s,igs15504.clk_30s
--clk esa15502.clk,esa15503.clk,esa15504.clk      # these are 5 minutes!
--ref -740311.8581,-5457066.4731,3207249.3343:%x,%y,%z  # known position
--sol GPS:12:WC,GLO:12:PC,GPS+GLO:12:WC+PC
# end prs.cfg
```

Excerpts from the resulting log file prs.log follow. Note that at the bottom of the output there is an epoch in which ALL the data is missing; this is because the Javad receiver seems to like to write blank lines at the bottom of the RINEX observation file...PRsolve handles it.

```
...
Solutions to be computed for this file:
SOLN GPS:12:WC GPS:L1:C1W,C1C GPS:L2:C2W
SOLN GLO:12:PC GLO:L1:C1P,C1C GLO:L2:C2P,C2C
SOLN GPS+GLO:12:WC+PC GPS:L1:C1W,C1C GPS:L2:C2W GLO:L1:C1P,C1C GLO:L2:C2P,C2C

RPF GPS:12:WC DAT 1550 259200.000 11 11 G01:1W2W G02:1W2W G04:1W2W G05:1W2W G09:1W2W G10:1W2W G12:1W2W
G17:1W2W G24:1W2W G27:1W2W G30:1W2W
RPF GPS:12:WC NAV 1550 259200.000 -740311.068003 -5457060.903092 3207247.627015 GPS -19600.788 (0
ok) V
RPF GPS:12:WC RMS 1550 259200.000 0 11 1.251 0.68 2.58 3.26 5.6 6 1.39e-09 G01 G02
G04 G05 G09 G10 G12 G17 G24 G27 G30 (0 ok) V
RPR GPS:12:WC POS 1550 259200.000 0.790097 5.570008 -1.707285 (0 ok) V
RNE GPS:12:WC POS 1550 259200.000 1.372593 0.034151 -5.716545 (0 ok) V
RPF GLO:12:PC DAT 1550 259200.000 5 5 R07:1P2P R10:1P2P R11:1P2P R20:1P2P R21:1P2P
RPF GLO:12:PC NAV 1550 259200.000 -740309.974450 -5457033.276262 3207238.303764 GLO -19665.710 (0
ok) V
RPF GLO:12:PC RMS 1550 259200.000 0 5 0.267 9.93 43.73 53.66 33.7 6 9.00e-09 R07 R10
R11 R20 R21 (0 ok) V
RPR GLO:12:PC POS 1550 259200.000 1.883650 33.196838 -11.030536 (0 ok) V
RNE GLO:12:PC POS 1550 259200.000 7.250670 -2.596091 -34.175122 (0 ok) V
RPF GPS+GLO:12:WC+PC DAT 1550 259200.000 16 16 G01:1W2W G02:1W2W G04:1W2W G05:1W2W G09:1W2W G10:1W2W
G12:1W2W G17:1W2W G24:1W2W G27:1W2W G30:1W2W R07:1P2P R10:1P2P R11:1P2P R20:1P2P R21:1P2P
RPF GPS+GLO:12:WC+PC NAV 1550 259200.000 -740311.741093 -5457060.030858 3207246.564689 GPS
-19601.158 GLO -19652.625 (0 ok) V
RPF GPS+GLO:12:WC+PC RMS 1550 259200.000 0 16 1.763 1.23 2.19 3.42 4.3 6 2.08e-09 G01
G02 G04 G05 G09 G10 G12 G17 G24 G27 G30 R07 R10 R11 R20 R21 (0 ok) V
RPR GPS+GLO:12:WC+PC POS 1550 259200.000 0.117007 6.442242 -2.769611 (0 ok) V
RNE GPS+GLO:12:WC+PC POS 1550 259200.000 0.847565 -0.750084 -6.921411 (0 ok) V

RPF GPS:12:WC DAT 1550 259210.000 11 11 G01:1W2W G02:1W2W G04:1W2W G05:1W2W G09:1W2W G10:1W2W G12:1W2W
G17:1W2W G24:1W2W G27:1W2W G30:1W2W
RPF GPS:12:WC NAV 1550 259210.000 -740313.047661 -5457064.945300 3207249.264493 GPS -19598.848 (0
ok) V
RPF GPS:12:WC RMS 1550 259210.000 0 11 1.246 0.68 2.58 3.26 5.6 4 3.17e-09 G01 G02
G04 G05 G09 G10 G12 G17 G24 G27 G30 (0 ok) V
RPF GPS:12:WC PFR 1550 259210.000 2.120 11 0.432 2.009 5.432 1.395 0.845 4.743 1.250 2.481 1.573 3.321
1.205
RPR GPS:12:WC POS 1550 259210.000 -1.189561 1.527800 -0.069807 (0 ok) V
RNE GPS:12:WC POS 1550 259210.000 0.624628 -1.384145 -1.203363 (0 ok) V
RPF GLO:12:PC DAT 1550 259210.000 5 5 R07:1P2P R10:1P2P R11:1P2P R20:1P2P R21:1P2P
RPF GLO:12:PC NAV 1550 259210.000 -740314.662170 -5457046.313830 3207245.846773 GLO -19657.920 (0
```

```

ok) V
RPF GLO:12:PC RMS 1550 259210.000 0 5 1.898 9.74 42.90 52.65 33.8 4 1.01e-08 R07 R10
R11 R20 R21 (0 ok) V
RPF GLO:12:PC PFR 1550 259210.000 26.079 5 20.329 13.084 15.527 21.498 12.805
RPR GLO:12:PC POS 1550 259210.000 -2.804070 20.159270 -3.487527 (0 ok) V
RNE GLO:12:PC POS 1550 259210.000 6.904575 -5.488623 -18.671466 (0 ok) V
RPF GPS+GLO:12:WC+PC DAT 1550 259210.000 16 16 G01:1W2W G02:1W2W G04:1W2W G05:1W2W G09:1W2W G10:1W2W
G12:1W2W G17:1W2W G24:1W2W G27:1W2W G30:1W2W R07:1P2P R10:1P2P R11:1P2P R20:1P2P R21:1P2P
RPF GPS+GLO:12:WC+PC NAV 1550 259210.000 -740314.204350 -5457064.927306 3207249.694315 GPS
-19598.419 GLO -19649.276 (0 ok) V
RPF GPS+GLO:12:WC+PC RMS 1550 259210.000 0 16 1.926 1.23 2.19 3.42 4.3 4 3.81e-09 G01
G02 G04 G05 G09 G10 G12 G17 G24 G27 G30 R07 R10 R11 R20 R21 (0 ok) V
RPF GPS+GLO:12:WC+PC PFR 1550 259210.000 2.805 16 -0.414 2.139 6.761 2.733 0.467 6.216 0.680 5.683
0.960 3.209 0.325 0.655 3.957 6.051 5.973 1.181
RPR GPS+GLO:12:WC+PC POS 1550 259210.000 -2.346250 1.545794 0.360015 (0 ok) V
RNE GPS+GLO:12:WC+PC POS 1550 259210.000 0.925789 -2.532754 -0.867208 (0 ok) V

```

...

```

RPF GPS:12:WC DAT 1550 345580.000 11 11 G01:1W2W G02:1W2W G04:1W2W G05:1W2W G09:1W2W G10:1W2W G12:1W2W
G17:1W2W G24:1W2W G27:1W2W G30:1W2W
RPF GPS:12:WC NAV 1550 345580.000 -740311.517695 -5457065.304631 3207250.568202 GPS -19602.528 (0
ok) V
RPF GPS:12:WC RMS 1550 345580.000 0 11 0.924 0.68 2.55 3.23 5.6 4 3.87e-09 G01 G02
G04 G05 G09 G10 G12 G17 G24 G27 G30 (0 ok) V
RPF GPS:12:WC PFR 1550 345580.000 2.028 11 -0.293 2.153 1.291 1.271 3.363 4.750 1.157 1.867 3.200
3.241 1.768
RPR GPS:12:WC POS 1550 345580.000 0.340405 1.168469 1.233902 (0 ok) V
RNE GPS:12:WC POS 1550 345580.000 1.673214 0.180239 -0.414222 (0 ok) V
RPF GLO:12:PC DAT 1550 345580.000 6 6 R07:1P2P R08:1P2P R11:1P2P R13:1P2P R21:1P2P R23:1P2P
RPF GLO:12:PC NAV 1550 345580.000 -740313.825477 -5457065.733134 3207249.969644 GLO -19651.961 (0
ok) V
RPF GLO:12:PC RMS 1550 345580.000 0 6 1.060 1.87 10.39 12.26 26.9 4 9.81e-09 R07 R08
R11 R13 R21 R23 (0 ok) V
RPF GLO:12:PC PFR 1550 345580.000 3.950 6 3.256 1.479 2.950 1.060 3.418 -1.674
RPR GLO:12:PC POS 1550 345580.000 -1.967377 0.739966 0.635344 (0 ok) V
RNE GLO:12:PC POS 1550 345580.000 0.785185 -2.048993 -0.083043 (0 ok) V
RPF GPS+GLO:12:WC+PC DAT 1550 345580.000 17 17 G01:1W2W G02:1W2W G04:1W2W G05:1W2W G09:1W2W G10:1W2W
G12:1W2W G17:1W2W G24:1W2W G27:1W2W G30:1W2W R07:1P2P R08:1P2P R11:1P2P R13:1P2P R21:1P2P R23:1P2P
RPF GPS+GLO:12:WC+PC NAV 1550 345580.000 -740312.082564 -5457064.790136 3207250.122495 GPS
-19602.667 GLO -19652.588 (0 ok) V
RPF GPS+GLO:12:WC+PC RMS 1550 345580.000 0 17 1.138 0.99 1.86 2.85 5.7 4 2.73e-09 G01
G02 G04 G05 G09 G10 G12 G17 G24 G27 G30 R07 R08 R11 R13 R21 R23 (0 ok) V
RPF GPS+GLO:12:WC+PC PFR 1550 345580.000 2.475 17 -0.306 3.377 2.982 1.827 2.678 5.502 1.818 3.533
2.360 2.556 1.816 2.358 1.872 4.467 1.518 0.062 -0.930
RPR GPS+GLO:12:WC+PC POS 1550 345580.000 -0.224464 1.682964 0.788195 (0 ok) V
RNE GPS+GLO:12:WC+PC POS 1550 345580.000 1.508178 -0.448668 -1.013954 (0 ok) V

```

```

RPF GPS:12:WC DAT 1550 345590.000 0 11 -G01:1-2- -G02:1-2- -G04:1-2- -G05:1-2- -G09:1-2- -G10:1-2-
-G12:1-2- -G17:1-2- -G24:1-2- -G27:1-2- -G30:1-2-
RAIMCompute failed to find enough satellites with data for GPS:12:WC at time 2009/09/23 23:59:50 = 1550
345590.000
RPF GLO:12:PC DAT 1550 345590.000 0 6 -R07:1-2- -R08:1-2- -R11:1-2- -R13:1-2- -R21:1-2- -R23:1-2-
RAIMCompute failed to find enough satellites with data for GLO:12:PC at time 2009/09/23 23:59:50 = 1550
345590.000
RPF GPS+GLO:12:WC+PC DAT 1550 345590.000 0 17 -G01:1-2- -G02:1-2- -G04:1-2- -G05:1-2- -G09:1-2-
-G10:1-2- -G12:1-2- -G17:1-2- -G24:1-2- -G27:1-2- -G30:1-2- -R07:1-2- -R08:1-2- -R11:1-2- -R13:1-2-
-R21:1-2- -R23:1-2-
RAIMCompute failed to find enough satellites with data for GPS+GLO:12:WC+PC at time 2009/09/23 23:59:50
= 1550 345590.000

```

```

----- Final output GPS:12:WC -----
Simple statistics on GPS:12:WC RAIM solution
ECEF_X N: 8639 Ave: -740311.9398 Std: 1.2427 Min: -740317.7395 Max: -740305.2473
ECEF_Y N: 8639 Ave: -5457067.2085 Std: 2.4240 Min: -5457086.9883 Max: -5457044.9484
ECEF_Z N: 8639 Ave: 3207250.0099 Std: 1.7892 Min: 3207243.5695 Max: 3207267.4687
Weighted average GPS:12:WC RAIM solution
-740311.9488 -5457067.4067 3207250.0661 8639
Covariance: GPS:12:WC RAIM solution
ECEF_X ECEF_Y ECEF_Z

```

ECEF_X	6.663e-05	3.418e-05	-1.398e-05
ECEF_Y	3.418e-05	2.508e-04	-8.128e-05
ECEF_Z	-1.398e-05	-8.128e-05	1.115e-04

APV: GPS:12:WC RAIM solution sigma = 2.028 meters with 94066 degrees of freedom.

Simple statistics on GPS:12:WC RAIM XYZ position residuals (m)

ECEF\_X N: 8639 Ave: -0.0817 Std: 1.2427 Min: -5.8814 Max: 6.6108

ECEF\_Y N: 8639 Ave: -0.7354 Std: 2.4240 Min: -20.5152 Max: 21.5247

ECEF\_Z N: 8639 Ave: 0.6756 Std: 1.7892 Min: -5.7648 Max: 18.1344

Weighted average GPS:12:WC RAIM XYZ position residuals (m)

-0.0907	-0.9336	0.7318	8639
---------	---------	--------	------

Simple statistics on GPS:12:WC RAIM NEU position residuals (m)

North N: 8639 Ave: 0.2087 Std: 1.4603 Min: -8.8120 Max: 11.4800

East N: 8639 Ave: 0.0179 Std: 1.1747 Min: -5.0229 Max: 5.5095

Up N: 8639 Ave: 0.9798 Std: 2.6663 Min: -19.9129 Max: 20.9591

Weighted average GPS:12:WC RAIM NEU position residuals (m)

0.1572	0.0357	1.1787	8639
--------	--------	--------	------

Covariance of GPS:12:WC RAIM NEU position residuals (m)

	North	East	Up
North	7.669e-05	1.735e-06	-2.302e-05
East	1.735e-06	6.085e-05	-8.738e-06
Up	-2.302e-05	-8.738e-06	2.914e-04

----- Final output GLO:12:PC -----

Simple statistics on GLO:12:PC RAIM solution

ECEF\_X N: 8490 Ave: -740312.1445 Std: 5.9541 Min: -740374.1379 Max: -740128.9950

ECEF\_Y N: 8490 Ave: -5457068.3142 Std: 26.3454 Min: -5457830.2649 Max: -5455731.5328

ECEF\_Z N: 8490 Ave: 3207251.1083 Std: 12.9146 Min: 3207166.4586 Max: 3207446.2792

Weighted average GLO:12:PC RAIM solution

-740311.8559	-5457067.4204	3207250.1055	8490
--------------	---------------	--------------	------

Covariance: GLO:12:PC RAIM solution

	ECEF_X	ECEF_Y	ECEF_Z
ECEF_X	3.506e-04	1.082e-04	-8.020e-05
ECEF_Y	1.082e-04	1.130e-03	-4.770e-04
ECEF_Z	-8.020e-05	-4.770e-04	5.448e-04

APV: GLO:12:PC RAIM solution sigma = 3.950 meters with 44056 degrees of freedom.

Simple statistics on GLO:12:PC RAIM XYZ position residuals (m)

ECEF\_X N: 8490 Ave: -0.2864 Std: 5.9541 Min: -62.2798 Max: 182.8631

ECEF\_Y N: 8490 Ave: -1.8411 Std: 26.3454 Min: -763.7918 Max: 1334.9403

ECEF\_Z N: 8490 Ave: 1.7740 Std: 12.9146 Min: -82.8757 Max: 196.9449

Weighted average GLO:12:PC RAIM XYZ position residuals (m)

0.0022	-0.9473	0.7712	8490
--------	---------	--------	------

Simple statistics on GLO:12:PC RAIM NEU position residuals (m)

North N: 8490 Ave: 0.5881 Std: 10.5373 Min: -320.1389 Max: 688.5441

East N: 8490 Ave: -0.0363 Std: 6.4160 Min: -62.6684 Max: 168.4972

Up N: 8490 Ave: 2.5044 Std: 27.2785 Min: -1148.0907 Max: 699.3606

Weighted average GLO:12:PC RAIM NEU position residuals (m)

0.1907	0.1295	1.1996	8490
--------	--------	--------	------

Covariance of GLO:12:PC RAIM NEU position residuals (m)

	North	East	Up
North	2.765e-04	-1.303e-05	-2.577e-05
East	-1.303e-05	3.359e-04	-8.129e-06
Up	-2.577e-05	-8.129e-06	1.413e-03

----- Final output GPS+GLO:12:WC+PC -----

Simple statistics on GPS+GLO:12:WC+PC RAIM solution

ECEF\_X N: 8639 Ave: -740311.9184 Std: 1.2048 Min: -740320.8280 Max: -740306.0475

ECEF\_Y N: 8639 Ave: -5457067.1221 Std: 2.3078 Min: -5457083.5895 Max: -5457050.9971

ECEF\_Z N: 8639 Ave: 3207249.9581 Std: 1.6746 Min: 3207241.4046 Max: 3207261.6197

Weighted average GPS+GLO:12:WC+PC RAIM solution

-740311.9242	-5457067.3740	3207250.0795	8639
--------------	---------------	--------------	------

Covariance: GPS+GLO:12:WC+PC RAIM solution

	ECEF_X	ECEF_Y	ECEF_Z
ECEF_X	5.909e-05	2.642e-05	-1.309e-05
ECEF_Y	2.642e-05	2.110e-04	-7.548e-05
ECEF_Z	-1.309e-05	-7.548e-05	9.686e-05

APV: GPS+GLO:12:WC+PC RAIM solution sigma = 2.475 meters with 138592 degrees of freedom.

Simple statistics on GPS+GL0:12:WC+PC RAIM XYZ position residuals (m)  
 ECEF\_X N: 8639 Ave: -0.0603 Std: 1.2048 Min: -8.9699 Max: 5.8106  
 ECEF\_Y N: 8639 Ave: -0.6490 Std: 2.3078 Min: -17.1164 Max: 15.4760  
 ECEF\_Z N: 8639 Ave: 0.6238 Std: 1.6746 Min: -7.9297 Max: 12.2854  
 Weighted average GPS+GL0:12:WC+PC RAIM XYZ position residuals (m)  
                   -0.0661                  -0.9009                  0.7452                  8639

Simple statistics on GPS+GL0:12:WC+PC RAIM NEU position residuals (m)  
 North N: 8639 Ave: 0.2088 Std: 1.2589 Min: -6.3141 Max: 7.1115  
 East N: 8639 Ave: 0.0275 Std: 1.1573 Min: -7.1202 Max: 4.9938  
 Up N: 8639 Ave: 0.8773 Std: 2.5802 Min: -13.9709 Max: 18.4990  
 Weighted average GPS+GL0:12:WC+PC RAIM NEU position residuals (m)  
                   0.1868                  0.0556                  1.1547                  8639

Covariance of GPS+GL0:12:WC+PC RAIM NEU position residuals (m)  
                                   North                                  East                                  Up  
       North          6.034e-05          2.125e-07          -1.427e-05  
       East          2.125e-07          5.479e-05          -5.940e-06  
       Up           -1.427e-05          -5.940e-06          2.518e-04

PRSolve timing: processing 135.200 sec, wallclock: 138 sec.

## PRSplot User's Guide

### July, 2011

**Introduction.** PRSplot.pl is a script, written in perl, which reads the PRSolve output log file and generates plots of the results using gnuplot. Perl and gnuplot are freely available for all the platforms on which PRSolve is available, namely Windows, Linux, Unix, Mac, Solaris, etc. PRSplot will generate plots, versus time, of position residuals, clocks, RMS residual, DOPs, RAIM slope and number of satellites, as well as scatter plots of position residuals.

**Installing perl and gnuplot.** Perl is free and already installed on many systems, including all Unix-based ones. It is easily installed on Windows; consider the ActiveState distribution ([www.activestate.com](http://www.activestate.com)). Gnuplot must be installed on your system for PRSplot to work; gnuplot is available at [www.gnuplot.info](http://www.gnuplot.info). Be sure to install version 4.4 or higher, and include the wx terminal. Windows users can get a binary install, and on other platforms the build is straightforward. PRSplot has an option (`--gnuplot`) to specify the name of the gnuplot executable (including path if necessary); also in PRSplot.pl the defaults are defined at the top of "sub Clear" and may be changed if you like. gnuplot includes documentation, and there are many things you could do with PRSplot or its `--dirty` (see below) output using other gnuplot commands.

Running PRSplot is as simple as typing PRSplot.pl (perl PRSplot.pl in Windows) at the command line. You could also create a script PRSplot (or batch file PRSplot.bat on Windows) that contains the single command `perl /path/PRSplot.pl "$@"` (or `perl \path\PRSplot.pl %*` in Windows) and then the script can be run simply as PRSplot (this is assumed in the following).

When PRSplot is run, if there is no log file with the default name (prs.log) in the current directory, all you get is an error message. You can specify the PRSolve log file to use with the option `--file <logfile>`. To see the "syntax page" for PRSplot, enter `PRSplot --help`; the output produced by this is also found in Appendix B.

**Which solution to plot.** PRSplot plots data for only one of the PRSolve TAGs; the tag may be specified in the `--tag` option. Thus if you want to plot North-East-Up residuals you would enter `--tag RNE`. The default tag is RPY (which is NOT a PRSolve TAG); in this case PRSplot uses the RPY tag but with the weighted average RPY solution subtracted out, i.e. RPY residuals from the *average* solution (not quite the same as RPY residuals). This is particularly useful when there is no reference position given to PRSolve.

The solution descriptor to plot may be given to PRSplot with the `--desc` option; the default is the *last* descriptor listed in the log file. The first thing PRSplot does is find the solution descriptor (if no `--desc`) and the "Weighted average RAIM solution" written by PRSolve at the bottom of the log file for this descriptor. It also counts how many time tags there are at which the solution was marked bad. This information is printed before the plot(s) are generated.

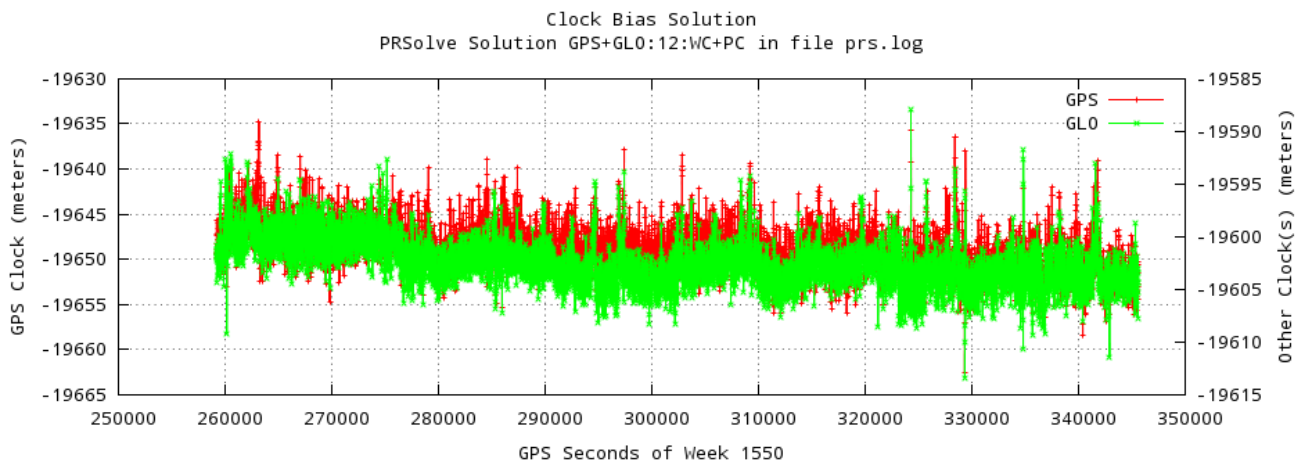
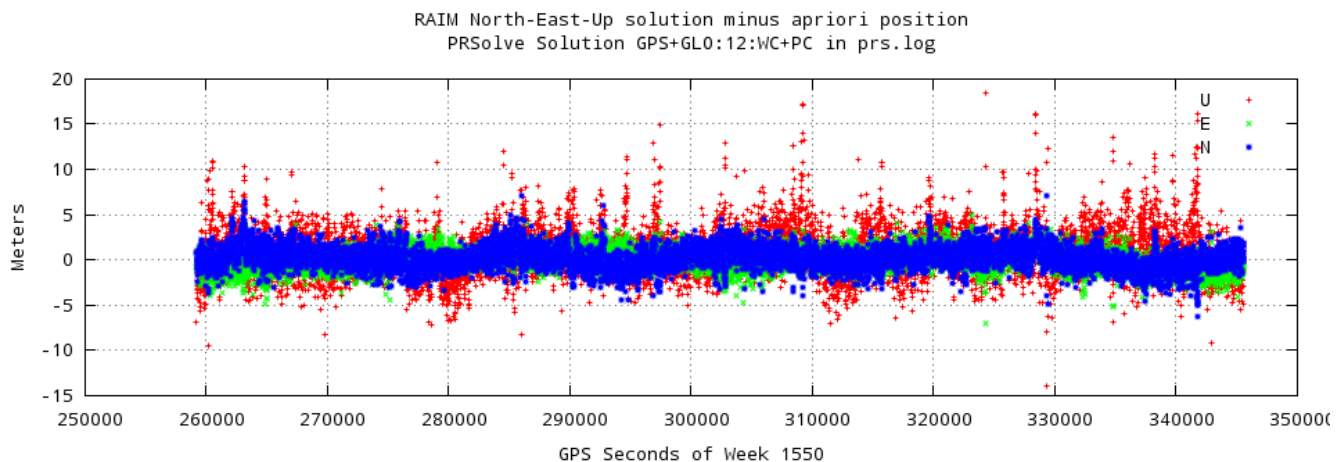
Thus if, after running PRSolve (with no --log option), you simply run PRSplot you get output like this.

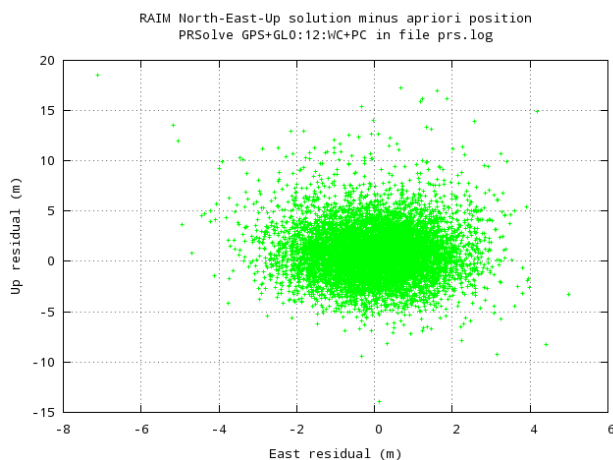
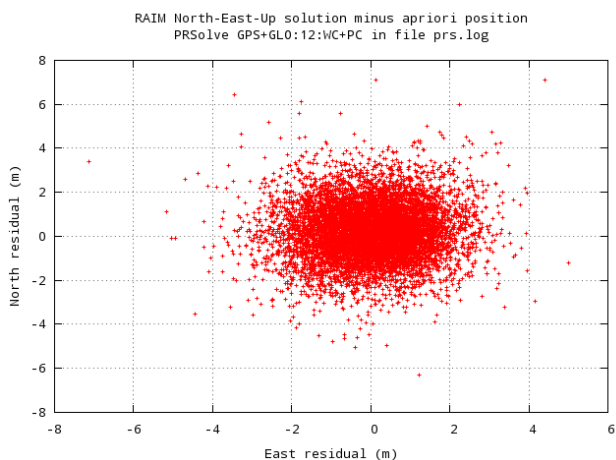
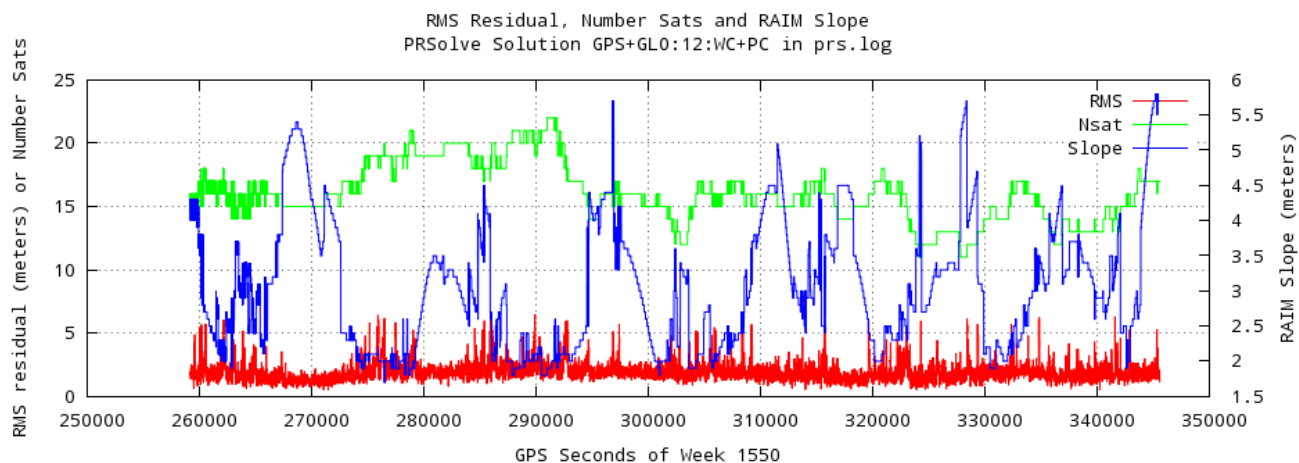
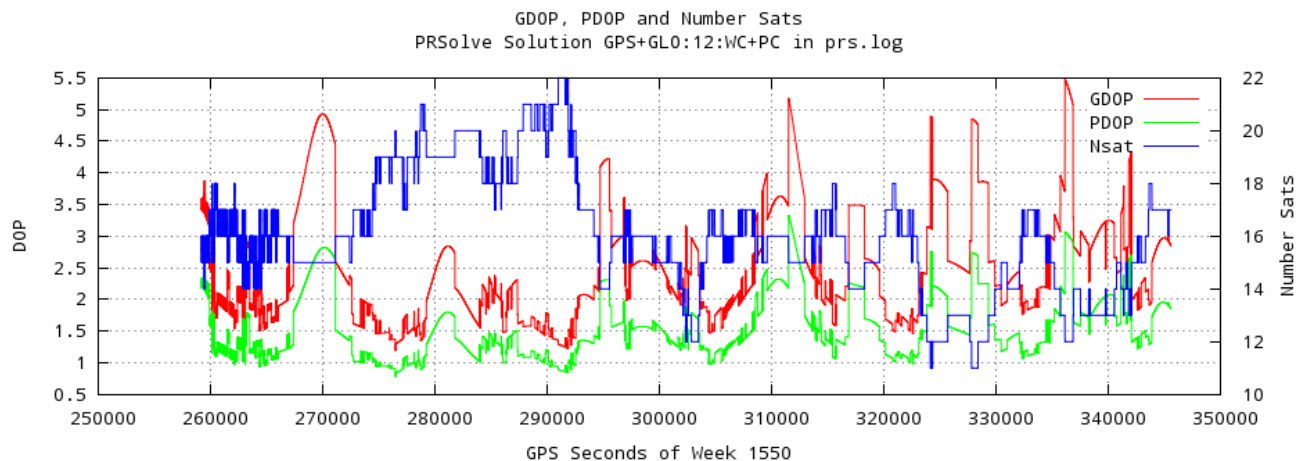
```
Solution description is GPS+GL0:12:WC+PC
Weighted average position solution: (8639 epochs)
X   -740311.9242 +/- 0.7145 m
Y   -5457067.3740 +/- 1.3501 m
Z    3207250.0795 +/- 0.9148 m
There were NO rejected epochs
Warning - no plots selected (--pos|clk|rms|dop|scatter).
```

This gives you the solution descriptor being plotted, the average solution with the number of epochs, and how many rejected epochs there were. It does not plot anything because, as it says, no plots were specified.

**What to plot.** Plots are specified with one or more of the options --pos, --clk, --rms, --dop, and --scatter. The title of each plot contains the solution descriptor and the PRSolve log file name. The plots are of timeseries (horizontal axis is GPS seconds of week) except the scatter plots, which show the position residuals as X versus Y (or East versus North) and Y versus Z (or East versus Up) on square plots. Here are examples; they were generated by commands such as

```
PRSplot --tag RNE --pos --png pos.png
```





Note that while the scatter plots are approximately square, the horizontal and vertical scales are not necessarily the same.



**Data options.** Time limits may be applied to the data that PRSplot takes out of the PRSolve log file, using options `--beg` and `--end`. The argument here is always "week,seconds-of-week". The `--no4` option tells PRSplot NOT to plot solutions where there are only 4 satellites' data used; this is useful when plotting "Simple" solutions (tag S\*) because often a 4-satellite solution is an outlier (RAIM cannot operate with only 4 satellites).

**Plot options.** The range of the vertical or 'Y' axes may be adjusted by the options `--yrange` (left) and `--y2range` (right); these are passed directly onto gnuplot as 'set yrange [<arg>]' (see gnuplot documentation). Also, the size of the plot in pixels may be determined by the `--geometry` (and `--long`) option; the argument is width:height where width and height are numbers of pixels. Note that the default geometry for the scatter plots is 640x480. The geometry option is possible because PRSplot uses gnuplot's wx terminal. [Changing the script to use another terminal is beyond the scope of this document.]

**Output options. PNG file.** Output may be to an image file in "png" format; this is accomplished with the `--png` option. No plot is shown on the screen in this case. The file name need not have the .png extension; PRSplot will add it. Often the .png output isn't quite as good as the plot on the screen because of the font size; with the `--font` option you can specify the font to use in the .png. The format of the argument here is that of gnuplot (see the gnuplot doc); the default is 'Droid Sans Mono, 10'. (The Droid fonts are freely available over the internet.)

The option `--quiet` causes nothing to be written to the screen.

**Gnuplot command file.** Finally, the `--dirty` option causes PRSplot to leave behind a file of the gnuplot commands that generated the plot. The plot may be reproduced on the screen by running the command 'gnuplot file' where gnuplot is the gnuplot command on your system, and file is the dirty filename. The dirty file is named 'prsplotxxx.gp' where xxx is the plot type (pos, clk, rms, dop) or prsplotXY.gp and prsplotYZ.gp for the scatter plots. This file allows you to customize the plot directly by modifying or adding gnuplot commands; refer to the gnuplot documentation for details. Note that PRSplot uses 'set terminal wxt enhanced' for plots and 'set term pngcairo enhanced' for .png images.

**Important notes for Windows users.** 1) Most important is that when gnuplot puts up a plot window, it also puts up a little window with an 'Ok' button and the label 'Close xxx window' that destroys the plot when the button is clicked. **It is important that you click the 'Ok' to close**, rather than closing the plot window itself. If you close the plot window with alt-F4 or with the 'X' button in the upper right, Windows destroys the plot and the 'Ok' button, however it leaves an 'orphan' gnuplot executable running, waiting for each 'Ok' to be clicked. Eventually all these gnuplot.exes will eat up the memory in your system; you'll need to kill them with something like Process Explorer. 2) Gnuplot geometry (and colors, etc) in Windows is controlled by the little text file wgunuplot.ini found (probably) in the same directory as the gnuplot executable (wgunuplot.exe). The PRSplot script, if you give it a `--geometry` option, reads this file, backs it up as wgunuplot.ini.bak, and rewrites it with the chosen geometry. If this does not work it may be because PRSplot cannot find the right directory; you should define an environment variable GNUPLOT to contain the path to the gnuplot directory. For example at the command line run

```
export GNUPLOT=C:\gnuplot\bin
```

using the correct path, before you run PRSplot.

## Appendix A. PRSolve syntax page.

PRSolve, part of the GPS Toolkit, Ver 4.0 6/1/11, Run 2011/09/16 08:56:07

Usage: PRSolve [options]

Program PRSolve reads one or more RINEX (v.2+) observation files, plus one or more ephemeris (RINEX nav or SP3) files, and computes a pseudorange position-and-clock solution, using a RAIM algorithm to eliminate outliers. Either single- or mixed-system (GNSSs) processing may be selected; input data is determined by, and solutions are labelled with, the 'solution descriptor' (see below). Output is to a log file, and also optionally to a RINEX observation file with the position solutions in comments in auxiliary header blocks. A final solution, covariance and statistics are given at the bottom of the log file.

In the log file, results at each time tag appear in lines with the format:

"TAG descriptor LABEL week sec.of.week CONTENT (code) [N]V"

where TAG denotes the type of solution or solution residuals:

RPF RAIM ECEF XYZ solution  
RPR RAIM ECEF XYZ solution residuals [only if --ref given]  
RNE RAIM North-East-Up solution residuals [only if --ref given]  
SPS Simple ECEF XYZ solution [only if --SPSout given]  
SPR Simple ECEF XYZ solution residuals [only if both SPS & ref given]  
SNE Simple North-East-Up solution residuals [only if SPS & ref given]

and LABEL followed by CONTENT is:

NAV X Y Z SYS clock\_bias [SYS clock\_bias ...]  
POS X Y Z  
CLK SYS clock\_bias [SYS clock\_bias ...]  
RMS Nrej Ngood RMS TDOP PDOP GDOP Slope niter conv SAT [SAT ...]  
DAT Ngood Nsats <SAT>:<freq><code> ... (list of sats with freq+code found)

and where

X Y Z = position solution, or solution residuals, depending on TAG;  
RNE and SNE yield North-East-Up residuals, at --ref position  
SYS = system or GNSS, e.g. GPS GLO GAL ... (identifies system of clock bias)  
Nsats = number of satellites in the RINEX file at this time  
Ngood = number of satellites used in the solution algorithm  
Nrej = number of satellites rejected by the RAIM algorithm  
RMS = RMS residual of fit (meters)  
Slope = RAIM 'slope' value  
xDOP = Dilution of precision (T=time, P=position, G=geometric=T+P)  
niter = number of iterations performed by the solution algorithm  
conv = final convergence value (delta RMS position) of the solution algorithm  
SAT = satellite identifier (e.g. G10, R07); minus sign means rejected  
CODE = return value from solution algorithm (with words if --verbose)  
[N]V = V for valid solution, NV for not valid (don't use!)

Default values appear in () after options below.

# Input via configuration file:

--file <fn> Name of file with more options [#->EOL = comment] [repeat] ()

# Required input data and ephemeris files:

--obs <fn> RINEX observation file name(s) [repeat] ()

--eph <fn> Input Ephemeris+clock (SP3 format) file name(s) [repeat] ()

--nav <fn> Input RINEX nav file name(s) [repeat] ()

# Other (optional) input files

--clk <fn> Input clock (RINEX format) file name(s) [repeat] ()

--met <fn> Input RINEX meteorological file name(s) [repeat] ()

--dcb <fn> Input differential code bias (P1-C1) file name(s) [repeat] ()

# Paths of input files:

--obspath <p> Path of input RINEX observation file(s) ()

--ephpath <p> Path of input ephemeris+clock file(s) ()

--navpath <p> Path of input RINEX navigation file(s) ()

--clkpath <p> Path of input RINEX clock file(s) ()

```

--metpath <p>      Path of input RINEX meteorological file(s) ()
--dcbpath <p>      Path of input DCB (P1-C1) bias file(s) ()
# Editing [t(time),f(format) = strings; default wk,sec.of.wk OR YYYY,mon,d,h,min,s]
--start <t[:f]>    Start processing data at this epoch ([Beginning of dataset])
--stop <t[:f]>     Stop processing data at this epoch ([End of dataset])
--decimate <dt>   Decimate data to time interval dt (0: no decimation) (0.00)
--elev <deg>      Minimum elevation angle (deg) [--ref or --forceElev req'd] (0.00)
--forceElev       Apply elev mask (--elev, w/o --ref) using sol. at prev. time tag (don't)
--exSat <sat>     Exclude this satellite [eg. G24 | R | R23,G31] [repeat] ()
# Solution Descriptors [NB. --sol causes --sys, --code and --freq to be ignored]
--sol <s:f:c>     Explicit descriptor <sys:freq:code> e.g. GPS+GLO:12:PWC+PC [repeat] ()
--sys <s>         Compute solutions for system(s) (GNSSs) <s>=S[,S,S+S], etc. [repeat]
(GPS,GLO,GPS+GLO)
                    Allowed systems s: GPS,GLO,GAL,GEO(SBAS),COM
--code <s:c>      System <s> preferred tracking codes <c>, in order [cf RINEX] [repeat] ()
                    Defaults: GPS:PYMIQSLXWCN, GLO:PC, GAL:ABCIQXZ, GEO:CIQX, COM:IQX
--freq <f>       Frequencies (L<f>) to use in solution [e.g. 1 12 12+15] [repeat] (12)
# Solution Algorithm:
--wt             Weight the measurements using elevation [--ref req'd] (don't)
--rms <lim>      Upper limit on RMS post-fit residual (m) (6.50)
--slope <lim>    Upper limit on maximum RAIM 'slope' (1000.00)
--nrej <n>        Maximum number of satellites to reject [-1 for no limit] (-1)
--niter <lim>    Maximum iteration count in linearized LS (10)
--conv <lim>     Maximum convergence criterion in estimation in meters (3.00e-07)
--Trop <m,T,P,H> Trop model <m> [one of Zero,Black,Saas,NewB,Neill,GG,GGHt
                    with optional weather T(C),P(mb),RH(%) (NewB,20.0,1013.0,50.0)
# Output [for formats see GPSTK::Position (--ref) and GPSTK::Epoch (--timefmt)] :
--log <fn>       Output log file name (prs.log)
--out <fn>       Output RINEX observations (with position solution in comments) ()
--ver2          Write out (--obs) version 2 of RINEX (don't)
--ref <p[:f]>     Known position p in fmt f (def. '%x,%y,%z'), for resids, elev and ORDs ()
--SPSout        Output autonomous pseudorange solution [tag SPS, no RAIM] (don't)
--ORDs <fn>     Write ORDs (Observed Range Deviations) to file <fn> [--ref req'd] ()
--timefmt <f>   Format for time tags in output (%4F %10.3g)
# Diagnostic output:
--verbose       Print extended output information (don't)
--debug        Print debug output at level 0 [debug<n> for level n=1-7] (-1)
--help         Print this and quit (don't)

```

## Appendix B. PRSplot syntax page.

Usage: PRSplot.pl (or ./PRSplot.pl) [options]

Plot the position solutions in a PRSolve (pseudorange solution) output file; the plot may include the solution residuals (XYZ or NEU), the clock solution, the number of satellites, DOPs and the RMS residual. Scatter plots (X vs Y and Y vs Z) may also be produced.

The default plot uses tag RPV and includes XYZ position residuals (vs average), clock(s), RMS residual of fit and number of satellites.

Options (default):

```
# Input data
--file <f>      input PRSolve log file <f> [-f] (prs.log)
--desc <d>      use solution descriptor [defaults to last in PRSolve file][-d] ( )
--tag <t>       choices as follows [-t] (RPV)
                  RPF  RAIM ECEF XYZ solution
                  RPV  RAIM ECEF XYZ solution minus average position
                  RPR  RAIM ECEF XYZ solution minus apriori position
                  RNE  RAIM North-East-Up solution minus apriori position
                  SPS  Simple ECEF XYZ solution
                  SPV  Simple ECEF XYZ solution minus average position
                  SPR  Simple ECEF XYZ solution minus apriori position
                  SNE  Simple North-East-Up solution minus apriori position

# Edit data
--beg <[w,s]>   ignore data before this GPS <week,sow> time [-b] (0,0)
--end <[w,s]>   ignore data after this GPS <week,sow> time [-e] (9999,604800)
--no4          reject solutions using only 4 satellites

# What to plot [one or more must be given]
--scatter      plot 'X vs Y' and 'Y vs Z' [or 'E vs N' and 'E vs U']
--pos          plot position [residuals]
--clk          plot clock bias for each system
--rms          plot RMS residual, Nsats, and RAIM slope
--dop          plot GDOP, PDOP, and Nsats

# Details of plot(s)
--gnuplot <g>  name, with path, of the gnuplot executable (gnuplot)
--yrange <l:h> set range of left axis on position plots to l(low)-h(high) ( )
--y2range <l:h> set range of right axis on position plots to l(low)-h(high) ( )
--psize <s>    point size (0.5)
--geometry <g> plot geometry (size) in pixels [-g] (960x350)
--long        plot with 'long' geometry 1200x400
--nogrid      plot without grid

# Output options (MSWin is always dirty)
--png <f>     output png to file <f>.png [choose only one plot] ( )
--font <f>    use font <f> on the .png output (Droid Sans Mono, 10)
--dirty       leave behind gnuplot file prsplotpos.gp, etc.
--quiet       don't print anything to screen [-q]
--help        print this message and quit [-h]
```

PRSplot configuration:

```
--beg 0,0
--psize 0.5
--file prs.log
--font "Droid Sans Mono, 10"
--end 9999,604800
--gnuplot gnuplot
--geometry 960x350
--tag RPV
--help
```