

## **Post OCT99 scheme of computation of correlations between adjustments in geodetic VLBI processing software SOLVE.**

---

*Leonid Petrov*

### *Abstract:*

*This document describes Solve interface for generating correlations between the parameters estimates and describes input and output formats.*

### *Table of contents:*

- 1 [Overview](#)
  - 2 [Input interface.](#)
  - 3 [Output ASCII CRL\\_SPOOL format](#)
  - 4 [Output Binary CRL\\_SPOOL format](#)
  - 5 [Notes](#)
- 

## **1 Overview**

Serious analysis of VLBI results requires knowledge of a covariance matrix between the adjustments of interest. Program COREL which did this task in SOLVE is not obsolescent. New post OCT99 scheme of computation of correlations is developed.

## **2 Input interface.**

Four different cases in computation of correlations between adjustments in global solution occur:

1. GLO\_GLO: correlations between global parameters;
2. GLO\_LOC: correlations between global and local parameters;
3. LOC\_LOC: correlations between local parameters within the same session;
4. CROSS\_LOC: correlations between local parameters of different sessions.

Global parameters by definition are parameters which are estimated using entire data set. Example: station velocities.

## guide correlations

Segmented parameters are parameters modeled by linear spline (or the same: continuous piece-wise function) and are estimated for each session independently. Example: atmosphere path delay.

Local parameters are parameters which are estimated for each session independently and which are not segmented parameters. Example: nutation angle Delta Psi.

S-SOLVE treats segmented and local parameters by the same manner and calls them "arc-parameters". F-SOLVE treats them differently. No global parameters: only local and segmented are estimated in independent solution. In principle it is possible to compute correlations between segmented parameters also but it is not planned in visible future.

Correlations are computed in BATCH mode only. A keyword CORRELATIONS in section \$OUTPUT controls whether to compute correlations between adjustments and if yes, how.

```
{CORRELATIONS [NO or YES FORMAT [ASCII or BINARY]
  [{ GLO_GLO  PARAM_INCLUDE <par_filename>
    {PARAM_EXCLUDE <par_filename>}}
  { GLO_LOC  PARAM_INCLUDE <par_filename>
    {PARAM_EXCLUDE <par_filename>}}
  { LOC_LOC  PARAM_INCLUDE <par_filename>
    {PARAM_EXCLUDE <par_filename>}}
  { CROSS_LOC PARAM_INCLUDE <par_filename>
    {PARAM_EXCLUDE <par_filename>}}
    SES_INCLUDE <ses_filename>
    {SES_EXCLUDE <ses_filename>}}
  ]
}
```

A construction PARAM\_INCLUDE <filename> PARAM\_EXCLUDE <filename> provides a flexible way for specification of the parameter names. Correlations between parameters listed in PARAM\_INCLUDE definition files are computed except the parameters listed in PARAM\_EXCLUDE files. Solve "doesn't see" any other parameters when it computes correlations. Parameters definition files <filename> specify the list of 20-letters long parameters names. Wild-card symbols \* ? are allowed!. If no path to a parameters definition file is specified then SOLVE searches for them in \$SAVE\_DIR directory. Format of parameters definition file:

\* Lines with ## symbols in positions 1-2 and empty lines are considered as comments and ignored.

## guide correlations

\* Each line (which is not a comment) contains a 20-symbols long string with parameter name in internal SOLVE format. Parameter name is generated in `./src/solve/cutil/get_names.f` for pre-defined parameters or in a user program for user partials. Symbols `* ?` are considered as wild-cards and expanded in according rules of filename expansion in operating system UNIX.

Examples:

\* PARAM\_INLCUDE all.cdf (to compute correlations between all parameters)  
all.cdf:

```
##  
## Correlations definition file  
##  
## All parameters  
##  
*
```

\* PARAM\_INLCUDE sta.cdf (to compute correlations between components of all station coordinates)  
sta.cdf:

```
*COMPONENT
```

\* PARAM\_INLCUDE vel.cdf (to compute correlations between components of velocities of all stations)  
vel.cdf:

```
*VELOCITY
```

\* PARAM\_INLCUDE my1.cdf (to compute correlations between coordinates and velocities of NRAO20 station and components of nutation)  
my1.cdf:

```
NRAO20 * VELOCITY  
NRAO20 * COMPONENT  
LONGITUDE NUTATION  
OBLIQUITY NUTATION
```

\* PARAM\_INLCUDE my2.cdf PARAM\_EXCLUDE my3.cdf (to compute correlations

## guide correlations

between all source positions except some sources)

my2.cdf:

\* RT. ASC.

\* DEC.

my3.cdf:

0059+581 RIGHT ASCEN

0059+581 DECLINATION

CTA26 RIGHT ASCEN

CTA26 DECLINATION

0454-234 RIGHT ASCEN

0454-234 DECLINATION

0458-020 RIGHT ASCEN

0458-020 DECLINATION

2234+282 RIGHT ASCEN

2234+282 DECLINATION

2255-282 RIGHT ASCEN

2255-282 DECLINATION

There is no distinction between user parameters and built-in parameters in this scheme. When new built-in parameters will be added (why not?) no modification in source code will be needed. Analogously, construction SES\_INCLUDE, SES\_EXCLUDE allows to specify a list of sessions for computation of CROSS\_LOC correlations. Wild-card symbols \* ? are allowed!

### 3 Output ASCII CRL\_SPOOL format

Correlations, unlike covariances, are dimensionless coefficients in the range [-1.0, 1.0] . Example of the output file:

```
# ASCII CRL_SPOOL Format. Revision 2001.05.18
# Analysis center: GSF -- NASA Goddard Space Flight Center
# Analyst:      Leonid Petrov ( xxx )
# Machine:      leo 9000/735 HP-UX B.10.20
# Executables:  /data11/mk4/bin
# Solve initials: PE
# Solution ID:   test_job1 (test of correlations)
```

## guide correlations

```
# Spool format:  PRE_OCT2000
# Local time:   2001.05.31-19:35:07
#
# Type:        GLO_GLO Correlations
#
*
* This line is comment.
*
4  5 "GILCREEK X COMPONENT" "GILCREEK Y COMPONENT" -0.556215603
4  6 "GILCREEK X COMPONENT" "GILCREEK Z COMPONENT"  0.432941964
4  7 "GILCREEK X COMPONENT" "HARTRAO X COMPONENT" -0.217073754
4  8 "GILCREEK X COMPONENT" "HARTRAO Y COMPONENT" -0.549527974
4  9 "GILCREEK X COMPONENT" "HARTRAO Z COMPONENT"  0.639315466
4 10 "GILCREEK X COMPONENT" "HOBART26 X COMPONENT" -0.695608370
#
# Type:        GLO_LOC Correlations
# Database:     $99MAR15XH <5>
# Start_date:   2451253.08330000  1999.03.15-13:59:57.120
# Stop_date:    2451254.08120000  1999.03.16-13:56:55.679
#
1 389 "FORTLEZA X COMPONENT" "X WOBBLE 09903151400" -0.117249278
1 390 "FORTLEZA X COMPONENT" "Y WOBBLE 09903151400" -0.080747136
1 393 "FORTLEZA X COMPONENT" "LONGITUDE NUTATION "  0.021860446
1 394 "FORTLEZA X COMPONENT" "OBLIQUITY NUTATION "  0.033797252
2 389 "FORTLEZA Y COMPONENT" "X WOBBLE 09903151400" -0.136677586
2 390 "FORTLEZA Y COMPONENT" "Y WOBBLE 09903151400"  0.061179534
2 393 "FORTLEZA Y COMPONENT" "LONGITUDE NUTATION " -0.053816501
```

Output file consists of a) header -- lines which start from #; b) comments -- lines which start from \*; c) body. The body has records of fixed length. Fields:

1. 1-5 -- index of the first parameter of the pair;
2. 7-11 -- index of the second parameter of the pair;
3. 15-34 -- name of the first parameter of the pair;
4. 39-58 -- name of the second parameter of the pair;
5. 61-73 -- correlation coefficient. Format: F12.9;

(Of course, correlation cannot be obtained with precision better than 0.001, but more digits may appear useful for diagnostic purposes.)

Output file name is CORLxx where xx are SOLVE user initials and it is located in SOLVE scratch directory. The file is purged at the start of BATCH. New correlations are appended to the end of the file in order of their computation.

## guide correlations

Correlation definitions files all.cdf, sta.cdf, src.cdf, eop.cdf are available in SAVE\_DIR directory.

Correlations are computed in all modes: NON-FAST and FAST. Parameter FAST\_COV should be set appropriately in the latter case. Only LOC\_LOC correlations can be computed in independent mode.

### 4 Output Binary CRL\_SPOOL format

Binary format is much more efficient and 8 times terse. ASCII correlation file with correlation matrix 4096x4096 would occupy approximately 1 Gb disk space.

Binary correlation file consists of physical records. Each physical record has the format:

4 bytes

| descriptor | logical\_record |

Descriptor has 4 bytes length and contains the length of logical record in bytes in binary format long\_signed. Logical record can be one of the types:

- comment
- sections delimiter
- parameter name
- correlation coefficients buffer

Logical record of the same type, except section delimiter, are grouped in sections.

Each comment line contains # character as the first character of the line. The first logical record is always a comment line which keeps the format identifier:

# Binary CRL\_SPOOL Format. Revision 2001.05.18

Other comments are the same as in ASCII format. Logical records with comments are grouped in the section which is the first section of the file.

Logical record "section delimiter" has the format

## guide correlations

4 bytes, char | 4 bytes  
| delimiter\_type | number\_of\_elements |

delimiter\_type is one of

"\$ CH" -- characters section. It points out the the the next section of logical records will be the section with parameters names.  
The field number\_of\_elements has the meaning of the number of parameters for which the correlations are computed. It will be exactly number\_of\_elements logical record in the next section.

"\$ CR" -- correlation sections. It points out the the the next section of logical records will be the section with correlations coefficients. The field number\_of\_elements has the meaning of the number of correlation coefficients which this file contains.

Logical record "parameter name" has the format

5 bytes, char | 2 bytes | 20 bytes, char | 2 bytes  
| parameter\_index | filler\_1 | parameter\_name | filler\_2

parameter\_index keeps the index of this parameter in Solve parameters list.  
parameter\_index is of character type, in the format I5  
Parameter\_name keeps 20-bytes long character string of solve parameter name.

Logical record "correlation coefficients buffer" keeps N segments:

8 bytes   8 bytes   8 bytes   8 bytes   8 bytes  
| segment\_1 | segment\_2 | segment\_3 | ...   | segment\_N |

N can be arbitrary in range [1, 4096]

Each segment has the format:

2 bytes | 2 bytes | 4 bytes  
| index\_1 | index\_2 | correlation |  
short   short   float

index\_1 -- index in the Solve list of the first parameter between which correlation is computed.

index\_2 -- index in the Solve list of the second parameter between which correlation is computed.

correlation -- correlation coefficients as 4-bytes long float number

## guide correlations

value.

NB: it is possible that  $\text{index\_1} > \text{index\_2}$  and  $\text{index\_1} < \text{index\_2}$ . Usually correlations are put in the ascending order of their indices, but software for reading correlation file should not rely upon it.

Solve provides the set of subroutines for reading binary correlation file and the program for conversion binary correlation file to ascii:

`$MK4_ROOT/utls/read_crl/`

## 5 Notes

Current (04-JUN-2001) status.

Not implemented:

CROSS\_LOC correlation mode;

BINARY output format for GLO\_LOC and LOC\_LOC correlations.

---

Questions and comments about this guide should be sent to:

Leonid Petrov ( [sgdass@lists.nasa.gov](mailto:sgdass@lists.nasa.gov) )

*Last update: 2001.06.04*