User manual to eop_alignment

Leonid Petrov

Abstract:

This document is a brief user manual for program eop_alignment

Table of contents:

1 Overview

```
Program  EOP_ALIGNMENT is for computing right hand side of
NO_NET_rotation/translation constraint equations of global VLBI
solutions in order to eliminate relative shift and drift of polar
motion and UT1 time series with respect to IERS C04 series.
```

2 Usage

```
Usage: eop_alignment <sol-file> <nn-cons-list> <IERS-C04-file>

<sol-file> -- generic name of the output files obtained from
              parsing spool file with using program getpar.
<nn-cons-list> -- List of NNT-POS constraints used in solution.
                  This list can be found in Spool file just before
                  the section of global parameters. Cut this list,
                  put it into the file and feed if eop_alignment.
<IERS-C04-file> -- external EOP file in IERS C04 format as it
                   was 2002.05.20
```

3 Output

```
  eop_alignment writes right hand side of constraint equations in
the screen. Just copy these lines and insert them to your Batch
control file. If your previous solution had zero net-rotation,
net-translation constraints, the new solution will have zero
shift and drift with respect to IERS C04. Station position and
velocity will be reciprocally changed with respect to the previous
```

solution, of course.

4 Algorithm

1) Compute the difference between EOP from the Solve solution with zero right part and IERS C04. Compute weighted secular drift and the shift with respect to IERS C04 at the reference epoch for which station positions in this solution were obtained.

2) Compute the right hand side of constraint equations.

Shift and drift of EOP series with respect to a reference is equivalent to a net-translation/rotation of station positions and velocity field. It can be written as:

```
   M * T = D

   where M -- matrix of dimension 3*6
   (N_pos -- total number of station position, N_vel -- total
    number of station velocity. If a station had an episodic
    motion, than its position is counted twice) in the form

    1  0  0   0    r3  -r2
    0  1  0  -r3   0    r1
    0  0  1   r2  -r1   0

 ... station 2, station 3 ... station N_pos

 0  0  0   0    0    0    1  0  0   0    r3  -r2    \
 0  0  0   0    0    0    1  0  0  -r3   0    r1     | Station 1
 0  0  0   0    0    0    1  0  0   r2  -r1   0     /

 ... station 2, station 3 ... station N_vel

  T -- 6-dimensional vector of transformation: translation and
       rotation;
  D -- 3-dimensional vector of station displacements due to
       a transformation.
```

Summing over all station (if a station had an episodic motion, than its position is counted twice) we get the system of linear algebraic equations:

```
  Sum_i ( M_i * T = D_i )
```

LSQ solution of this problem will be in the form

```
T' = ( Sum_i Sum_j M_i * M_j )^-1 * Sum_i (M_i * D_i) (j<=i)
```

We can re-write this equation as

```
Sum_i (M_i * D_i) =  ( Sum_i Sum_j M_i * M_j ) * T' (j<=i)

This is just the equation of net-rotation,net-translation
constraint used in Solve. Setting the vector T' to the shift
between the EOP series

        /  X_shift  \
  T' = |  -Y_shift   |
        \  UT1_shit /

  we compute the right hand-side of constraint equation for
net-translation/net-rotation.

Analogously we compute right hand-side of constraint equation for
net-translation/net-rotation for velocities. But in summing the
stations with episodic motion are counted only once.


Caveat: the present version assumes that all stations which
        participated in no-net-translation constraitns for
        positions participated in no-net-translation for velicity
        and no-net-rotation for both position and velocity.
```

Questions and comments about this guide should be sent to:

Leonid Petrov (sgdass@lists.nasa.gov)

Last update: 2002.05.20