SOFTWARE FOR MODELING AND SIMULATION OF INERTIAL SENSORS AND DESIGN OF FILTERS FOR INITIAL ALIGNMENT OF NAVIGATION SYSTEMS.

J. Aranda, J.M. de la Cruz, P. Ruipérez and S. Dormido
Dpto. Informática y Automática
Facultad de Ciencias. UNED
Madrid. Spain

The scope of this paper is to describe the SEINAL software package for modeling and simulation of inertial sensors and for designing filters for the initial alignment. This software is made in FORTRAN on a HP370 computer with the HP-UX operating system. The software package is formed by modules. There is one module for each command and each command runs its respective module. Functionally, the software has five blocks: three blocks for modeling and simulation of inertial sensors and two blocks for studying the alignment filters. Each block allows two methods of performance: a sequential program and a translator of commands. The package is completed by a module for plotting. The results of analysis and simulation may be plotted, printed or stored for further processing.

The paper is completed with the modeling of a ring laser gyro by this program and the analysis of a filter for alignment of a strapdown system.

1.INTRODUCTION.-

The primary sensor in an Inertial Navigation System (INS) is the accelerometer. This instrument produces a precise output, which is proportional to the acceleration applied along the input axis of the sensor. To accomplish the orientation control of the accelerometers, an inertial sensor, the gyroscope, is used.

The initialization of an INS or alignment process (for both gimbal and strapdown systems) is defined as the determination of the angular relationship between a vehicle-fixed set of axes and a reference or navigation coordinate frame. The alignment is a critical process. The performance of an INS can only be as good as the accuracy to which it is initially aligned.

The scope of this paper is to describe the SEINAL software package for modeling and simulation of inertial sensors and for designing filters for the initial alignment. This software is made in FORTRAN on a HP-370 computer with the HP-UX operating system (the HP´s UNIX system V) and it uses the UNIX shell features.

This package software do the following jobs:

- Calibration: Parameter estimation of the deterministic model of the sensor. Linear and no-linear least square algorithms are used [1,2].

- Analysis and modeling of the stochastic noise. They are made by: test for stationarity and randomness, DC component and trend removal, time plots, spectral analysis, covariance analysis, sigma plot and Time Series Analysis (Box-Jenkins [3] and Pandit-Wu [4] techniques).

- Model simulation for each sensor. The whole model of a sensor can be simulated and their characteristics can be liken to the characteristics of real data.

- Analysis and design of filters applied to the alignment. This part enables: observability and covariance analysis, error budget tables, proposed low-order filters and analysis of the proposed low-order filters under parameter uncertainties [5,6].

- Simulation of the initial alignment process. The obtained filter can be simulated and it can be liken to other methods of alignment [6].

Each job can be made by a sequential program or by commands.

A brief description of the SEINAL software is giving in section 2 and performances examples are showing in section 3.

2. DESCRIPTION OF THE SEINAL SOFTWARE PACKAGE.-

The SEINAL software package is formed by modules. There is one module for each command and each command runs its respective module. Table-1 has the modules description. The commands are runed from the UNIX prompt. The call is alike to UNIX commands calls. It provides the user with a number of features that are present in the UNIX shell (the commands standard input/output can be redirected and the commands can be connected by pipelines). The call has the form:

$ *command {parameters} {options}*

where *command* is the command's name, *parameters* are the input to the command (i.e.: files name, data values), and *options* are literal, usually introduced by a minus sign, what modifies the action of the command (see table-2). If the parameters are not specified when invoking the command, they are assumed by defect or requested at the run time.

COMMAND	DESCRIPTION
MINCUA	Least square
MICUNL	No-linear least square
MICUNLR	No-linear least square whit constrained
CONSTR	Constrained apply to estimates parameters
COMSDE	Compesation scale factor and cosines
COMBDE	Compesation bias and cosines
AJPOL	Polinomial adjust
TRDACP	Data conversion
ACOR	Autocorrelation function
BOX-JEN	TSA by Box-Jenkins approach
CMPSE	Correlation method for power spectrum estimation.
COMTRA	Switch-on transients removal
DIFST	Time serie differencing
ELTEN	DC and tren removal
HISTG	Histogram
INTESPOT	Power spectrum integration
PAN-WU	TSA by Pandir-wu approach
PAREST	Statistical parameter
PMPSE	Peridiogram method for power spectrum estimation
SPLOT	Sigma plot
TSTAC	Run test
TSX2	Chi-square test
ERBUACEL	Accelerometer error table
ERBUGIR	Gyros errors table
INPSYS	Store systems matrices in a file
SIMAC	Simulation of one accelerometer
SIMEAC	Simulation of one accelerometer on the test table
SIMEGI	Simulation of one gyro on the test table
SIMGI	Simulation of one gyro
SIMSEN	Simulation of 3-gyros and 3-accelerometers
ANACOV	Covariance analysis
FILKAL	Kalman filter
MOCAFIL	Monte Carlo analysis
OBSER	Observability matrix
ALIGIR	Alignment by gyrocompassing
ALIKAL	Alignment by Kalman filter
ALISTI	Alignment by Stieler method
PLOT	Plot
PRINT	Print data files

TABLE-1: Command description

OPTION	ACTION
- p	Change standard output format to beautiful format
- i	Interactive execution
-t *file*	Print run trace to *file* file
-r *file*	Print result to *file* file

TABLE-2: Main options

The command use the following kinds of files:

Printer files (.i suffix), ASCII files ready for printer.
Data files (.r suffix), sequential data files, they usually have time series data or values of one function (i.e.: file with the gyro output or the values of the autocorrelation function).
Matrices files (.s suffix), files with system's matrices.
Graphics files (.g suffix), files with HPGL command, they are ready to be send to a HP plotter. Those files are made by the plot module.

The output produced by the commands can be send to four different devices:

Screen (standard output), it can be redirected and its format can be modified by options; it usually produce a results list.
Printer files with command results; they are selected by -r option.
Data files with data results.
Graphics, that can also be send to three different devices: screen, plotter or graphics files. Graphics are usually maded by PLOT command.

Commands are input either from the keyboard (standard input) or from data files. The keyboard input can be redirected from a file or from other command by pipelines. A pipeline have the form:

$ command1 | command2

the command1 output format must be equal to the command2 input format.

New commands can be easily appended. The new command can be written in FORTRAN or it can be a shell program containing previous commands.

Functionally, the software has five blocks (one block for each job). Three main blocks for modeling and simulation of inertial sensors: calibration (CALIB), analysis and modeling of the stochastic noise (ANAMOD), simulation of the model for each sensor (SIMSEN). Two blocks to study the filters for alignment: analysis and design of filters applied to the alignment (ADESFIL) and simulation of the initial alignment process (SIMALI). There are a shell program for each block, which can be easily modified. The blocks shell programs usually are a logical sequence of commands, which execute the block job. Some command parameters are assumed by defect, but other parameters can not be assumed and they are request at run time.

3. EXAMPLES.-

We can better understand the performance of a software package with some examples. In this section, we make use of the SEINAL software for calibrating and modeling a laser-gyro and for studying the alignment of strapdown systems with laser-gyros. We show the command sequence and the shell programs for those examples.

CALIBRATION OF A RING LASER GYRO.-

The calibration process consists in the parameter estimation of the deterministic error model of the sensor. Figure-1 has examples of the procedure for calibrating a laser-gyro. The procedure consist of two parts. In the first part, the fixed drift, the direction cosines and a value for the scale factor are estimated. In the second part the scale factor $S(w_{in})$ is identified as a function of the input rate. A three-axis test-table is considered as the main test-equipment for data acquisition. Table-3 has a commands typical sequence for the laser-gyro calibration.

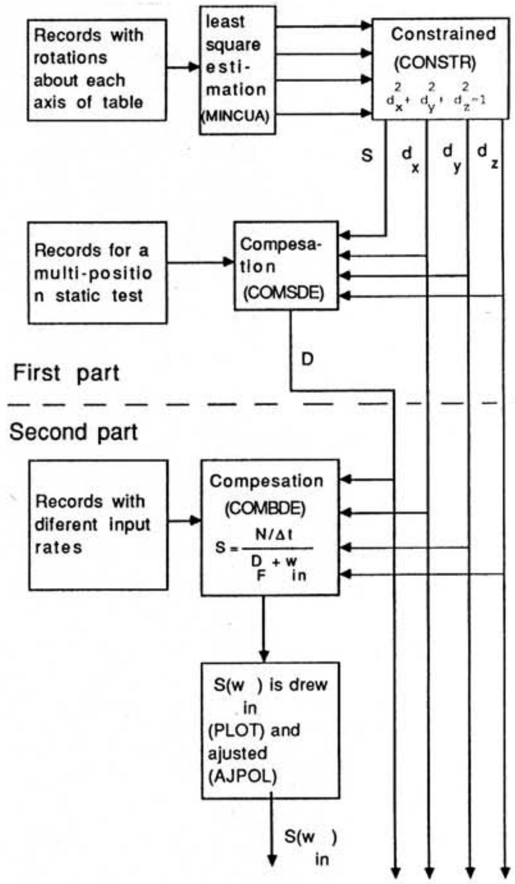

FIGURE 1: Calibration of a ring laser gyro

```
$ mincua 4 dat1giro.r > salida
$ constr < salida
131045  0.00355  0.00376
0.3535535  4.89475E-9  2.51833E-8
0.3535534  4.39892E-9  2.46875E-8
0.8660254  5.1517E-9  4.4545E-8
0.2931117E-5  1.4245E-8  1.42451E-8
$ comsde dat2giro.r bias.r 131045 0.35355535
0.3535534  0.8660254
$ parest bias.r > salida
$ combde dat3giro.r facesca.r 0.2931117E-5
0.35355535  0.3535534  0.8660254
$ plot facesca.r
$ ajpol facesca.r >> salida
```

TABLE-3: Typical commands sequence for the laser-gyro calibration

First, we assume that the laser gyro calibration model is [2]:

$$\frac{N}{\Delta t} = S*(\omega_{in} + D)$$

$$S = S_0 + S(\omega_{in})$$

where:

N: number pulses received in a Δt period of time.

D: fixed drift.

w_{in}: input angular rate = $d_x w_x + d_y w_y + d_z w_z$, where ($w_x$, w_y, w_z) are the three rate in the input frame and (d_x, d_y, d_z) are the misalignment angle.

S: scale factor.

S_0: nominal scale factor.

$S(w_{in})$: scale factor error (no-linearities), it is function of w_{in}.

The four parameters S^*d_x, S^*d_y, S^*d_z, S^*D are determined in a least square sense. To do this the MINCUA command is applied to the data in the file *dat1giro.r.* The command output is redirected to the file salida. The command MINCUA applies a linear regression technique to the input data.

We use the constrained:

$$d_x^2 + d_y^2 + d_z^2 = 1$$

(CONSTR command). The file *salida* is the input to the command; the output is the values of S,D, d_x, d_y, d_z together with an optimistic and pessimistic standard deviation for each of them.

The fixed drift is reestimated by a multi-position static test (file *dat2giro.r*). The gyro's measure is compensated for the scale factor and for the direction cosines (COMSDE). The final estimate of D is obtained by averaging of the fixed drift calculated for every position (PAREST).

In order to determine possible non-linearities of the scale factor, test are performed at various rotation rates of the test-table (file *dat3giro.r*). For each rate the scale factor is calculated by the COMDDE command. The scale factor versus the input rate is plotted (PLOT) and adjusted (AJPOL).

ANALYSIS AND MODELING OF THE STOCHASTIC NOISE.-

The modeling of the stochastic noise can be made by using two different techniques: a description of the in-run drift and a Time Series Analysis (TSA). Commands for both techniques are provided.

Table-4 shows a commands sequence for the stochastic modeling of a laser-gyro. The explanation of the table is as follows.

```
$ trdacp giro.dat giro.r
$ tstac giro.r -p > resanamod.i
$ tsx2 giro.r -p >> resanamod.i
$ parest giro.r -p >> resanamod.i
$ tail 3 resanamod.i
Mean value:              6.344
Standard deviation:      0.539
Average slope:           -0.18E-6
$ elten giro.r girocomp.r 6.344 0.539 -0.18E-6
$ plot giro.r
$ pmpse girocom.r 1024 2 512
$ plot espectro.r
$ acor girocomp.r 100
$ plot autocor.r
$ splot giromp.r 10
$ plot sigmaplot.r
$ box-jen girocomp.r >> resanamod.i
```

TABLE-4: Commands sequence for description of in-run drift.

The gyro's output are recorded in the file *giro.dat* with ASCII format. This file is converted into a form suitable for further processing (file *giro.r*) by the command TRDACP.

Then, they are preprocessing by: A test for stationarity and randomness (TSTAC and TSX2), calculation of statistical parameters (PAREST) and DC component and trend removal (ELTEN). The values showed in table-4 correspond to a run of 15 hours sampled every seconds. The mean value is 6.344 pulses, the standard deviation is 0.539 pulses and the average slope is -0.18E-6 pul/sec. It is noticed a little ramp with a slope of about 10^{-7}.

With plot *giro.r* description of the switch-on transients by the study of time plots is made.

Afterwards, follows a data analysis made by three different methods: Spectral analysis (PMPSE); covariance analysis (ACOR); and standard deviation versus time analysis (SPLOT). Each command makes a file with the values of the respective function, which are plotted. Figures 2 and 3 show power spectrum and sigma plot. The noise observed are: a rate white noise, an angle quatization, and a pink rate noise.

We can assumed that the gyro drift is a linear combination of a random bias, a rate white noise, a random ramp and a first-order Markov process (pink rate noise).

Finally, a TSA is made by the command BOX-JEN.

The commands standard output are redirected to the file resanamod.i.

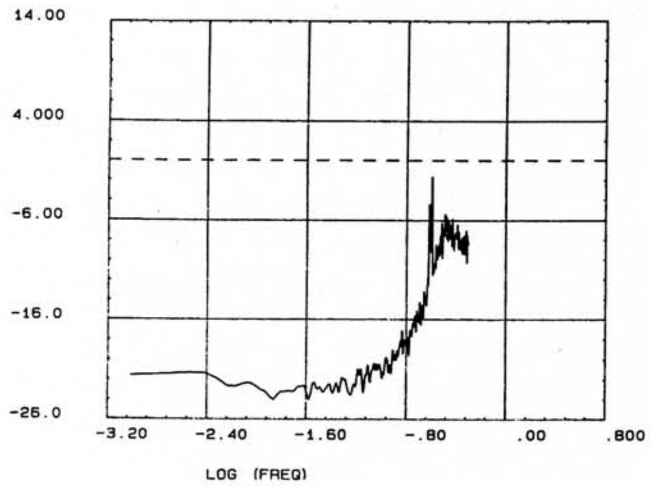

FIGURE-2: Power spectrum

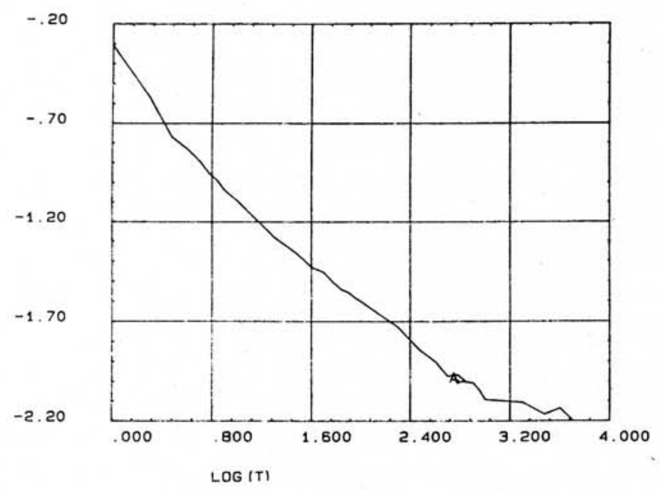

FIGURE-3: Sigma plot

STUDY OF STRAPDOWN SYSTEMS ALIGNMENT BY THE SEINAL SOFTWARE.-

The process of alignment is defined as the determination of the angular relationship between the body frame and a computational frame. A Kalman filter can be used for the angular misalignment estimation. The ADESFIL block helps to the analysis and design of filters. Table-5 has the shell program for the ADESFIL block.

The analysis and design of Kalman and low-order filters require the following steps [5,6]:

```
#
# Shell program of the ADESFIL block
#
echo "---------------- ADESFIL ----------------"
echo "Analysis and DESign of FILters for the alignment"
echo "Considered an whole model with error sensor shaping filters"
# -------------------------------- Input truth model matrices.
inpsys sisreal.s
# ------------------------------------ Observability analysis.
obser sisreal.s -p > resadesfil.i
# -------------------------------------- Covariance analysis.
# -------- ANACOV replies covarian.r file with states covariances.
anacov sisreal.s sisreal.s
plot covarian.r
# -------------------------------------- Error budget.
echo "Do you want make error budget (y/n)?:"
read resp
while test "$resp" = "y"
do
      inpsys sisdis.s                    # Input design model matrices with
                                         # only one error source.
      anacov sisreal.s sisdis.s
      print covarian.r >> resadesfil.i # print command request the
                                         # values to printer and they
                                         # are printer in resadesfil.i
      echo "Do you want continue error budget (y/n)?:"
      read resp
done
# -------------------- Analysis under parameters uncertainties.
echo "Do you want analysis under parameters uncertainties(y/n)?:"
read resp
while test "$resp" = "y"
do
      inpsys sisdis.s
      anacov sisreal.s sisdis.s
      print covarian.r >> resadesfil.i # print command request the
                                         # values to printer and they
                                         # are printer in resadesfil.i
      echo "Do you want continue analysis under uncertainties
(y/n)?:"
      read resp
done
# ------------------------------------ Monte Carlo analysis.
echo "Do you want a Monte Carlo simulation (y/n)?:"
read resp
if test "$resp" = "y"
then mocafil
fi
```

TABLE-5: ADESFIL shell program

1) Modeling of inertial sensors errors (this is maded by ANAMOD block).

2) Obtaining of alignment equations with sensor errors shaping filters. For a strapdown systems with laser-gyros, the misalignment error equations with the shaping filters have a state vector with 23 components. However, this would not be economic, because some of the terms may be taken into account, whose influence on the system behavior is not significant, and some terms may not be observable and there may be, therefore, an estimator of lower order yielding the same accuracy as the full-order Kalman filter.

3) Observability analysis. The observability is studied by the OBSER command. OBSER requests the system matrices file and replies the observability matrix and its rank.

4) Covariance analysis and error budget table [8]. They are made by the ANACOV command. Developing an error budget involves determining the individual effects of a single error source, or group of error sources. ANACOV requests the files with the system matrices for the truth model and the design model, and replies the covariances of the states. First, the design model is equal to the truth model, then the covariance of the truth states model is obtained. For each error source, the design model is considered with only this error source, and the states covariance is obtained. In this way, the error budget table is constructed.

5) From the covariance analysis, low-order filters are obtained. After the observability and covariance analysis, we arrive at a model with the following state vector: $(\varepsilon_N, \varepsilon_E, \varepsilon_D, B_N, B_D, x_D)$.

$(\varepsilon_N, \varepsilon_E, \varepsilon_D)$ are misalignment angles, (B_N, B_D) are north and down components of gyro bias and x_D is the down component of the accelerometers correlated-noises.

6) Analysis of the proposed filter under parameter uncertainties. This is also maded by the ANACOV command. To find the sensitive of the filter with respect to some parameters uncertainties, only those parameters are changed.

7) Monte Carlo analysis (MOCAFIL). Finally, a Monte Carlo simulation can be made. MOCAFIL run the commands FILKAL and ALIKAL.

Other two methods for alignment may be used: gyrocompassing (ALIGIR) and an open loop estimation procedure of the azimuth misalignment angle and of the north gyro drift (ALISTI), Stieler method. Those methods are described in reference [7].

4. CONCLUSIONS.-

In summary, SEINAL is a software package for modeling and simulation of inertial sensors and for designing filters for the initial alignment. It is an open package, which can be easily modified and expanded. The commands format are alike to the UNIX commands format. That provides it with the flexibility of the UNIX shell features. It is a useful tool for the experts.

Our primary design goal was to develop tools for people expert in UNIX systems. A secondary goal is to made the tools useful for novices and also to extend the software to other systems.

REFERENCES.-

[1] - K. K. Joos and U.K. Krogmann. "Identification and Determination of Strapdown Error-Parameters by Laboratory Testing". AGARDograph No. 254. Edited by H. Sorg. April 1.981. pp.7.1-7.37.

[2] - J. Aranda, J.M. de la Cruz, H. Ruiperez and S. Dormido. "Calibration and Stochastic Modeling of a Laser-Gyro by Laboratory Testing". Seventh International Conference on Mathematical and Computer Modeling. 2-5 August 1.989. Chicago (U.S.A).

[3] - G.E.P. Box and G.M. Jenkins. Time Series Analysis: forecasting and control. Holden-Day. 1.976.

[4] - S.M. Pandit and S.M. Wu. Time Series and System Analysis with Applications. John Wiley & Sons, Inc., 1.983.

[5] - W. Kortüm. "Design and Analysis of Low-Order Filters Applied to the Alignment of Inertial Platforms". AGARD Lectures Series, No. 82, 1.976, pp. 7.1-7.23.

[6] - J. Aranda. Modelización y calibración de giros-laser y alineamiento de sistemas ligados (strapdown). Thesis. UNED. Madrid. 1.989.

[7] - B. Stieler and H. Winter. "Gyroscopic instruments and their application to flight testing". AGARDograph, No. 160, vol. 15, September 1.982.

[8] - Applied Optimal Estimation. Edited by Arthur Gelb. MIT Press, 1.974.

IMACS - MCTS Symposium

MAY 7-10, 1991

CASABLANCA - MOROCCO

IMACS MCTS 91 CASABLANCA

Pr J. ARANDA
Dpto. Informatica y Automatica
Facultad de Ciencias
UNED
MADRID
SPAIN

138

PAPER TITLE : Software for Modelling and Simulation of inertial sensors and design of filters for initial alignment of navigation systems

Villeneuve d'Ascq, October 30, 1990

Dear Professor,

I have the pleasure to inform you that your paper indicated in reference has been accepted for presentation at the IMACS-MCTS Symposium to be held in Casablanca in May 7-10, 1991.

This paper has been assigned the code number ...138..... Please, use it in any further correspondence related to your paper.

You will receive the author's kit including special sheets and instructions for authors under separate cover by December 1990.

Sincerely Yours,

Professor P. BORNE
Co-Chairman of the IMACS-MCTS Symposium

Submissions : IMACS-MCTS Symposium, I.D.N., B.P. 48, 59651 VILLENEUVE D'ASCQ CEDEX, FRANCE
IDN, Telephone 33 20 91 01 15, Telefax 33 20 67 18 44, Telex : EUNOR 131 339 Code 350
Correspondences and camera ready papers : IMACS - MCTS Symposium, L.A.I.C., Faculté des Sciences I, B.P. 5366, MAARIF, CASABLANCA, MOROCCO - Telex FAC SCI 1 - 45937 M