OPEN SOURCE ORBIT DETERMINATION WITH SEMI-ANALYTICAL THEORY

Bryan Cazabonne[*], *Luc Maisonobe*[†]

CS-SI, 5 rue Brindejonc des Moulinais, 31506 Toulouse CEDEX 5, FRANCE

ABSTRACT

Space objects catalog maintenance demands an accurate and fast Orbit Determination (OD) process to cope with the ever increasing number of observed space objects. The development of new methods, that answer the two previous problems, becomes essential.

Presented as an alternative to numerical and analytical methods, the Draper Semi-analytical Satellite Theory (DSST) is an orbit propagator based on a semi-analytical theory allowing to preserve the accuracy of a numerical method while providing the speed of an analytical method. This propagator allows computing the mean elements and the short-period effects separately. We reproduced this architecture at the OD process level in order to be able to return, as desired, the mean elements or the osculating elements. Two major use cases are thus possible: fast OD for big space objects catalog maintenance and mean elements OD for station keeping needs.

This paper presents the different steps of development of the DSST-OD included in the Orekit open-source library [1]. Integrating an orbit propagator into an OD process can be a difficult process. Computing and validating derivatives is a critical step, especially with the DSST whose equations are very complex. To cope with this constraint, we used the automatic differentiation technique. Automatic differentiation has been developed as a mathematical tool to avoid the calculations of the derivatives of long equations. This is equivalent to calculating the derivatives by applying chain rule without expressing the analytical formulas. Thus, automatic differentiation allows a simpler computation of the derivatives and a simpler validation. Automatic differentiation is also used in Orekit for the propagation of the uncertainties using the Taylor algebra.

Existing OD applications based on semi-analytical theories calculate only the derivatives of the mean elements. However, for higher accuracy or if the force models require further development, adding short-period derivatives improves the results. Therefore, our study implemented the full contribution of the short-period derivatives, for all the force models, in the OD process. Nevertheless, it is still possible to choose between using the mean elements or the osculating elements derivatives for the OD.

This paper will present how the Jacobians of the mean rates and the short-periodic terms are calculated by automatic differentiation into the DSST-specific force models. It will also present the computation of the state transition matrices during propagation. The performance of the DSST-OD is demonstrated under Lageos2 and GPS Orbit Determination conditions.

***Index Terms*—** Orbit Determination, Semi-analytical theory, Open-source, DSST, Automatic differentiation, Orekit.

[*]bryan.cazabonne@c-s.fr

[†]luc.maisonobe@c-s.fr

1. INTRODUCTION

Our needs about the space sector are always evolving. It is difficult to imagine our everyday lives without the services provided by satellites like telecommunications or navigation. However, there are some counterparts to this globalization of satellite-based systems. Between the first orbital launch of Sputnik on 4 October 1957 and 1 January 2016 there have been over 5,160 launches and approximately 200 known satellite break-ups [2]. Therefore, a particular interest must be given to the impacts of the increase in the number of space objects. Especially with the issue of space debris. Wiedemann et al [3] estimate that the number of space debris with a size of 1 cm or more have exceeded 700,000. According to a NASA study [4], the number of objects in orbit with a size of 10cm or more is greater than 17,000.

The challenge is then to track and catalog these objects in order to control the risk of collision and thus preserve the physical integrity of the satellites and their ability to fulfill their missions. Orbit Determination (OD) and space objects catalogs become indispensable tools. OD is used to estimate the observations on the position and the velocity of an object in orbit from an estimation algorithm and an orbit propagator. Setty et al [2] highlighted that the US Joint Space Operations Center (JSpOC) performed hundreds of thousands of OD per day to maintain their space objects catalog. This means that the OD process must be both fast and accurate to cope with the volume of data to be analyzed.

Three types of orbit propagators exist. First, the analytical orbit propagator. It has the advantage of being fast but it has limited accuracy. The second type of orbit propagator is the numerical. At the opposite of the analytical propagator, it presents the advantage of being highly accurate. However, its computing speed does not answer the problem of the volume of data to be analyzed. The last type of orbit propagator is the semi-analytical. It is a mix between the numerical orbit propagator and the analytical. It has the advantages of both, the accuracy of a numerical propagator and the computing speed of an analytical one. Therefore, it is natural to pay close attention to this type of propagator to cope with the important number of orbit propagations to perform each day to maintain the space objects catalogs.

The Draper Semi-analytical Satellite Theory (DSST) is an orbit propagator based on a semi-analytical theory. It is an interesting tool for a fast and accurate orbit propagation, close to current needs. Its development started in the mid-1970s at the Computer Sciences Corporation of Maryland by a team led by Paul J. Cefola [5].

The DSST is present in the Orekit open-source library [1]. It is implemented for propagating the states based on classical real numbers (i.e. double precision numbers), limiting its use. Indeed, the real numbers used for the classical computations represent only a scalar value, generally one of the components of a state vector. It is possible, without altering the already complex structure of the algorithms (i.e. without making it even more complex and thus limiting the risk of error), to perform computations on extensions of the real numbers in order to add additional information to the scalar values of the state components, typically partial derivatives. This thanks to the automatic differentiation. The aim our study is to use the automatic differentiation in order to extend the use of the Orekit implementation of the DSST. Specifically, to use it in OD processes.

The structure of this paper is as follows. In section 2, we present the theoretical elements about the DSST that are useful for the paper understanding. In section 3, we discuss the development of the DSST-OD. Section 4 gives the current numerical results obtained by the DSST-OD. Lageos 2 and GPS OD are performed. Conclusions and Future Work end the paper.

2. THE DSST

2.1. The equinoctial elements

One of the peculiarity of the DSST is the orbital parameters formalism. They are expressed in equinoctial coordinates. If a, e, i, Ω, ω, and M designate the conventional Keplerian elements set, the equinoctial elements are given by:

$$\left\{ \begin{array}{lcl} a & = & a \\ h & = & e\sin(\omega + I\Omega) \\ k & = & e\cos(\omega + I\Omega) \\ p & = & [\tan(\frac{i}{2})]^I \sin\Omega \\ q & = & [\tan(\frac{i}{2})]^I \cos\Omega \\ \lambda & = & M + \omega + I\Omega \end{array} \right. \tag{1}$$

The I element is called the retrograde factor and has two possible values:

$$I = \left\{ \begin{array}{ll} +1 & \text{for the direct equinoctial elements} \\ -1 & \text{for the retrograde equinoctial elements} \end{array} \right. \tag{2}$$

h and k represent (in the orbital frame) the components of the eccentricity vector, p and q are the components of the inclination vector and λ is the phased angle. The equinoctial elements are a rewrite of the conventional Keplerian orbital parameters. They have the advantage of avoiding singularities when the eccentricity or inclination of the orbit tend towards zero, contrary to the classical Keplerian parameters [6, 7].

2.2. DSST orbital perturbations

In Orekit, five orbital perturbations are implemented for the DSST: the Atmospheric Drag, the Third Body Attraction, the Zonal and Tesseral harmonics of the Earth's potential and the Solar Radiation Pressure (SRP). These perturbations cause the variation of the orbital element illustrated on Figure 1.

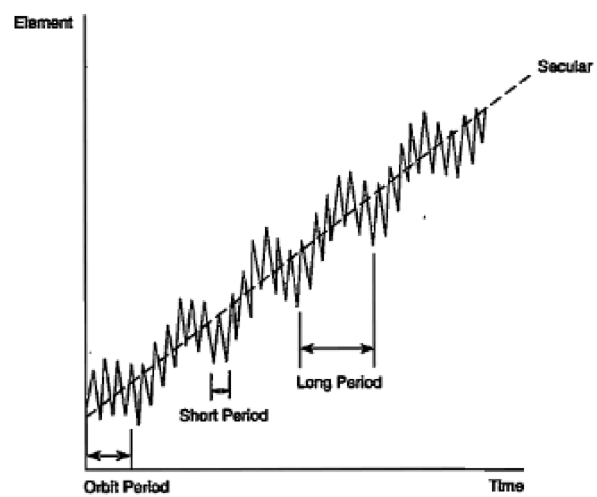

Fig. 1. Variation of an orbital parameter due to the orbital perturbations [8].

The linear variation of the element is called the secular variation. The short periods represent the periodic variations of the element whose period is lower than the orbital period. Finally, the long periods variations represent the periodic variations whose period is greater than the orbital period. All these variations are used in the mathematical model of the DSST. They are all implemented in the DSST-specific force models on Orekit.

2.3. Mathematical model of the DSST

The DSST is based on a very advanced mathematical model. This complexity makes it a powerful tool. Setty et al exposed the mathematical model of the DSST [9]. This model can be summarized by the following equation:

$$Y_i(t) = \overline{Y_i(t)} + \sum_{j=0}^{N} k_{ij}\eta_{ij}(t) \quad \text{i = 1,2,3,4,5,6} \tag{3}$$

Where,

- Y: $[a, h, k, p, q, \lambda]$ Equinoctial elements.
- $\bar{Y}$: $[\bar{a}, \bar{h}, \bar{k}, \bar{p}, \bar{q}, \bar{\lambda}]$ Mean equinoctial elements.
- $\sum_{j=0}^{N} k_{ij}\eta_{ij}(t)$: Short-periodic terms.
- k_{ij}: Multiplying factor.
- N: Degree of precision for short-periodic computation.
- $\eta_{ij}(t)$: Short-periodic functions.

Within the DSST theory, the variation of an orbital element is expressed by the sum of the mean elements and the short-periodic terms. Mean elements are computed numerically while short-periodic terms are computed analytically and that for all the DSST-specific force models.

On equation (3) the main contribution comes from the mean elements. In practice, during the computation of the partial derivatives, the main contribution also comes from the derivatives of the mean elements. In that respect, for many OD applications, computing the partial derivatives of the mean elements can be sufficient. However, for higher accuracy or if the force models require further development, adding short-periodic derivatives improves the results. Our study implemented both contributions in the OD process, while leaving the choice to the user to consider only the contribution of the mean elements derivatives or the total contribution by adding the short-periodic derivatives.

3. THE SEMI-ANALYTICAL ORBIT DETERMINATION

3.1. DSST-specific force models configuration

3.1.1. Automatic differentiation

Automatic differentiation has been developed as a mathematical tool to avoid the computation of the derivatives of long equations. All the theoretical elements about this theory are presented in Kalman's paper [10].

Let Y_i be an orbital element. Automatic differentiation allows the access to all the useful derivatives of Y_i without having to find their analytical expressions. This step is very important when we want to establish an OD process. Indeed, the computation of the derivatives is a mandatory step. The vector resulting from using automatic differentiation on equations of motion is the following one:

$$\begin{matrix} [Y_i & \partial Y_i/\partial Y_1 & \partial Y_i/\partial Y_2 & \cdots & \\ & \partial Y_i/\partial Y_6 & \partial Y_i/\partial P_1 & \cdots & \partial Y_i/\partial P_n] \end{matrix} \tag{4}$$

Where,

- P: The force model parameters (drag coefficient, etc).
- n: The number of force model parameters taken into account for the OD.

To implement an OD process with the DSST, two types of derivatives are used. First, the derivatives with respect to the orbital parameters. Then, the derivatives with respect to the force model parameters.

3.1.2. Development

In each DSST-specific force model implemented in Orekit there is a method that allows the computation of the mean elements rates. Let $\dot{Y}$ be the vector containing these terms.

$$\dot{Y} = [\dot{a}, \dot{h}, \dot{k}, \dot{p}, \dot{q}, \dot{\lambda}] \tag{5}$$

This method was implemented to provide the vector on equation (5) for the states based on classical real numbers. In order to establish a DSST-OD, this function has been also implemented to provide the Jacobians of the mean elements rates by automatic differentiation. The resulting element is then a matrix having the following form:

$$\begin{pmatrix} \dot{Y}_1 & \partial_{Y_1}\dot{Y}_1 & \partial_{Y_2}\dot{Y}_1 & \cdots & \partial_{Y_6}\dot{Y}_1 & \partial_{P_1}\dot{Y}_1 & \cdots & \partial_{P_n}\dot{Y}_1 \\ \dot{Y}_2 & \partial_{Y_1}\dot{Y}_2 & \partial_{Y_2}\dot{Y}_2 & \cdots & \partial_{Y_6}\dot{Y}_2 & \partial_{P_1}\dot{Y}_2 & \cdots & \partial_{P_n}\dot{Y}_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \dot{Y}_6 & \partial_{Y_1}\dot{Y}_6 & \partial_{Y_2}\dot{Y}_6 & \cdots & \partial_{Y_6}\dot{Y}_6 & \partial_{P_1}\dot{Y}_6 & \cdots & \partial_{P_n}\dot{Y}_6 \end{pmatrix} \tag{6}$$

This matrix can be summarized by:

$$\begin{pmatrix} \dot{Y} & \frac{\partial \dot{Y}}{\partial Y} & \frac{\partial \dot{Y}}{\partial P} \end{pmatrix} \tag{7}$$

Where,

- $\frac{\partial \dot{Y}}{\partial Y}$: Jacobian of the mean elements rates with respect to the orbital parameters.
- $\frac{\partial \dot{Y}}{\partial P}$: Jacobian of the mean elements rates with respect to the force model parameters.

These two Jacobians are used for the computation of the state transition matrices thanks to the variational equations.

3.2. The variational equations

3.2.1. The state transition matrices

The state transition matrices are used by the OD algorithm to estimate the new parameters (force models and orbitals). Consequently, they are of great importance. Computation and validation of the state transition matrices are significant steps that should not be neglected.

These matrices are given by equations (8) and (9):

$$\frac{\partial Y}{\partial Y_0} = \begin{pmatrix} \frac{\partial Y_1}{\partial Y_{0_1}} & \frac{\partial Y_1}{\partial Y_{0_2}} & \cdots & \frac{\partial Y_1}{\partial Y_{0_6}} \\ \frac{\partial Y_2}{\partial Y_{0_1}} & \frac{\partial Y_2}{\partial Y_{0_2}} & \cdots & \frac{\partial Y_2}{\partial Y_{0_6}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial Y_6}{\partial Y_{0_1}} & \frac{\partial Y_6}{\partial Y_{0_2}} & \cdots & \frac{\partial Y_6}{\partial Y_{0_6}} \end{pmatrix} \tag{8}$$

$$\frac{\partial Y}{\partial P} = \begin{pmatrix} \frac{\partial Y_1}{\partial P_1} & \frac{\partial Y_1}{\partial P_2} & \cdots & \frac{\partial Y_1}{\partial P_n} \\ \frac{\partial Y_2}{\partial P_1} & \frac{\partial Y_2}{\partial P_2} & \cdots & \frac{\partial Y_2}{\partial P_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial Y_6}{\partial P_1} & \frac{\partial Y_6}{\partial P_2} & \cdots & \frac{\partial Y_6}{\partial P_n} \end{pmatrix} \tag{9}$$

In order to obtain these matrices, we performed four steps of development:

1. Extract the Jacobians $\frac{\partial \dot{Y}}{\partial Y}$ and $\frac{\partial \dot{Y}}{\partial P}$ from the matrix on equation (7).
2. Extract the Jacobians $\frac{\partial Y}{\partial Y_0}$ and $\frac{\partial Y}{\partial P}$ from the additional state initialized by the user at the creation of the orbit propagator.
3. Apply the variational equations. These equations are a mathematical tool for expressing the variations of a parameter according to the initial state. They are given by:

$$\begin{cases} \frac{d\left(\frac{\partial Y}{\partial Y_0}\right)}{dt} &= \frac{\partial \dot{Y}}{\partial Y} \times \frac{\partial Y}{\partial Y_0} \\ \frac{d\left(\frac{\partial Y}{\partial P}\right)}{dt} &= \frac{\partial \dot{Y}}{\partial Y} \times \frac{\partial Y}{\partial P} + \frac{\partial \dot{Y}}{\partial P} \end{cases} \tag{10}$$

4. Add these differential equations to the set of equations of motion. They will be integrated simultaneously by the numerical integrator, benefiting from the additional equations mechanism of the underlying mathematical library.

After the numerical integration, a particular attention has been paid to the validation of the computation of the state transition matrices.

3.2.2. *Validation of the state transition matrices*

Validation of the state transition matrices computation, thanks to the variational equations, was done by computing the same matrices by finite differences. The goal of the validation was to compare if the values were identical. Three tests have been performed.

1. A first test to validate the computation of the matrix $\frac{\partial Y}{\partial P}$ computing only the derivatives of the central attraction coefficient.
2. A second test to validate the computation of the matrix $\frac{\partial Y}{\partial P}$ computing only the drag coefficient derivatives.
3. A third test to validate the computation of the matrix $\frac{\partial Y}{\partial Y_0}$ without computing any force model parameter derivative.

Table 1 gives the force models configuration for each test. The propagation time was 30 minutes for the three tests.

Table 1. Configuration of the force models for state transition matrices validation.

Test	Tesseral	Zonal	Drag	SRP	Moon
1	✓	✓	✓	X	X
2	✓	✓	✓	X	X
3	✓	✓	X	✓	✓

At first, these tests showed that the Newtonian attraction derivatives were not taken into account in the computation of the state transition matrices. They also revealed that the some dependencies to the central attraction coefficient were implicit and therefore not differentiated.

Therefore, we added a new force model in order to take into account the Newtonian attraction during the computation of the state transition matrices. We also implemented the DSST-specific force models to have the central attraction coefficient appear explicitly as a force model parameter.

After these improvements, we obtained consistent state transition matrices.

3.3. Short-periodic terms derivatives

3.3.1. *Development*

In addition to the mean elements derivatives, we added the contribution of the short-periodic terms derivatives into the OD process. We paid close attention to retain the possibility to choose between using the mean elements or the osculating elements derivatives for the OD. This by computing the mean elements and the short-periodic effects separately.

As for the mean elements rates, there is a method allowing the computation of the short-periodic terms. This method is implemented in each DSST-specific force model of Orekit. In order to add the contribution of the derivatives of these terms on the DSST-OD, we implemented this method to provide the Jacobians of the short-periodic terms by automatic differentiation.

The last step consisted in adding the contribution of the short-periodic terms derivatives to the state transition matrices (In the case where the user wants to add them). As we said before, mean elements are computed numerically while short periodic terms are computed analytically. In that respect, we added the contribution of the short-periodic terms derivatives after the numerical integration of the mean elements rates.

3.3.2. Validation

The validation has been performed in the same way as part 3.2.2. We computed the state transition matrices, with the contribution of the short-periodic derivatives, following the three cases of test of Table 1. We again computed the state transition matrices by finite differences in order to compare the values obtained. The similarity between both computations showed that our computation of the state transition matrices gives correct results.

3.4. Orbit Determination tools

Orekit provides two OD algorithms, which are compatible with the DSST. The first one is a Batch Least Squares algorithm and the other one is a Kalman Filter. The architecture as well as the operating principle of these algorithms in Orekit, are exposed by Maisonobe et al [11].

We paid special attention on the development of the OD process that uses the Batch Least Squares algorithm to make it compatible with the DSST. Work is under way to develop also the DSST-OD with the Kalman Filter. In the results section, the accuracy and the computation time of both methods are exposed.

4. RESULTS

4.1. Orbit Determination conditions

4.1.1. Computer characteristics

The tests were performed on a 3.20 GHz Intel Core i5-3470 laptop with a 8 GB RAM.

4.1.2. Lageos2 and GPS force models

The first step was to define the force models used for each test case. They were adapted from the ones used for the validation of the OD with the numerical propagator in Orekit. It is not yet possible, for instance, to take into account the relativity force model for the GPS OD with the Orekit implementation of the DSST. However, this force is negligible with respect to the other force models used.

Table 2 gives the configuration of the force models for each DSST-OD.

Table 2. Configuration of the force models used for the DSST Orbit Determination.

OD	Tesseral	Zonal	Drag	SRP	Moon	Sun
Lageos2	✓	✓	X	X	✓	✓
GPS	✓	✓	X	✓	✓	✓

OD	Geo-potential [degree × order]
Lageos2	8×8
GPS	20×20

4.1.3. Integration step

The second step was to define the integration step used for the DSST-OD. The DSST has a significant advantage compared to the numerical propagator in terms of computation time. For the same level of accuracy, the numerical propagator requires an integration step around one hundredth of the orbital period to achieve a correct OD while the DSST needs an integration step of the order of several orbital periods. This because the elements computed numerically by the DSST are the mean elements. Typically, a step equal to one day makes it possible to obtain satisfactory results [7].

In Orekit, the user has the possibility to not enter directly the integration step, but instead an interval and a positional tolerance. The integrator chooses the optimal intergration step (i.e. the fastest) that fulfills the tolerance requirements in position imposed by the user. This method is used for both DSST-OD tests. Table 3 gives the values of the minimum step and the maximum step used for the DSST-OD tests.

Table 3. Integration steps used for the DSST-OD tests (first line). These values are used for both Lageos2 and GPS test cases and for both Batch Least Squares and Kalman Filter algorithms. The second line gives the values used for the Numerical-OD tests, with the same configurations.

Minimum step (s)	Maximum step (s)	Tolerance (m)
6000	86400	10
0.001	300	10

4.1.4. Test cases for the short-periodic terms

The derivatives of the short-periodic terms were not added all at the same time. We gradually added them to highlight the main contributions to improving the OD accuracy (Table 4). For each test, we considered the Lageos2 OD.

Table 4. Lageos2 test cases for highlighting the contribution of the short-periodic derivatives. **Case 1** considers only the mean elements derivatives for the OD. **Case 2** considers the mean elements derivatives and the short-periodic derivatives of the Zonal harmonics in the OD process. **Case 3** adds the contribution the Tesseral short-periodic derivatives. **Case 4** considers all the derivatives.

Case	Zonal	Tesseral	Third body
1	X	X	X
2	✓	X	X
3	✓	✓	X
4	✓	✓	✓

4.2. Batch Least Squares Orbit Determination

We first performed the DSST-OD with the Batch Least Squares algorithm present in Orekit. The results obtained were compared with the Orekit numerical OD in order to demonstrate the performance of the Orekit DSST-OD.

4.2.1. Lageos2 and GPS OD with mean elements derivatives

Figure 2 and 3 present the results obtained by the Lageos2 and the GPS DSST-OD considering only the mean elements derivatives. These results were compared with those obtained by the numerical OD for the same configurations. Our study analyzed the accuracy of the DSST-OD by computing the mean relative gap between the computed state vector and the theoretical one. It is important to note that we considered all the derivatives of the mean elements for all the force models used in the simulations.

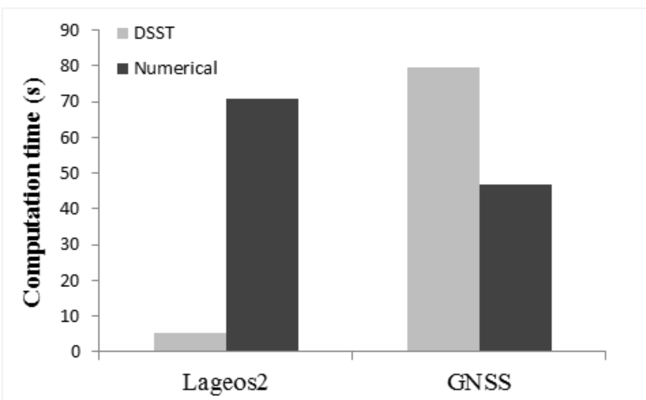

Fig. 2. Comparison between the numerical and the DSST-OD in terms of computation time. Batch Least Squares case.

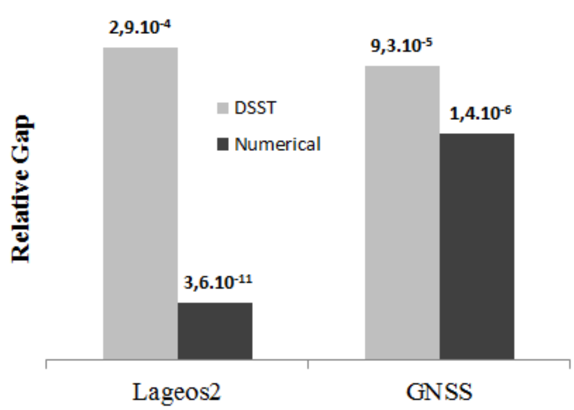

Fig. 3. Comparison between the numerical and the DSST-OD in terms of accuracy. Batch Least Squares case.

For the computation time issue, Figure 2 demonstrates that the DSST propagator is about fourteen times faster than the numerical propagator for the Lageos2 OD. However, it is about thirty seconds slower for the GPS OD. We explain this anomaly by the difference in the choice of force models used by the two Orbit Determinations. Specifically, taking into account the Solar Radiation Pressure greatly increases the computation time. We realized this thanks to a software permitting more in-depth performance testing on GPS OD.

Figure 3 shows that the accuracy of the DSST-OD is worse than the one with the numerical propagator. However, the gap between the computed and the theoretical state vector is low ($\sim 10^{-4}$) . This demonstrates that the Orekit DSST-OD gives satisfactory results. The accuracy difference between the numerical propagator and the DSST is due to the fact that the derivatives of the short-periodic terms are not considered in this tests. Therefore, we added them to the Lageos2 OD.

4.2.2. Lageos2 OD with both types of derivatives

Figure 4 shows the impact of adding the short-periodic terms derivatives to the Lageos2 OD. Several remarks can be made thanks to this figure.

First, adding the short-periodic derivatives improves the accuracy of the DSST-OD. Indeed, the addition of the Zonal short-periodic derivatives enhances significantly the relative gap between the theoretical and the computed state vector. Furthermore, the computation time have not increased significantly after adding these terms. Problems start with the Tesseral contribution for the short-periodic derivatives. The computation time was multiplied by eight without improving the accuracy. We again performed more in-depth performance testing to highlight the critical points (i.e. the ones that cause the increase of the computation time) into the Tesseral short-periodic derivatives computation.

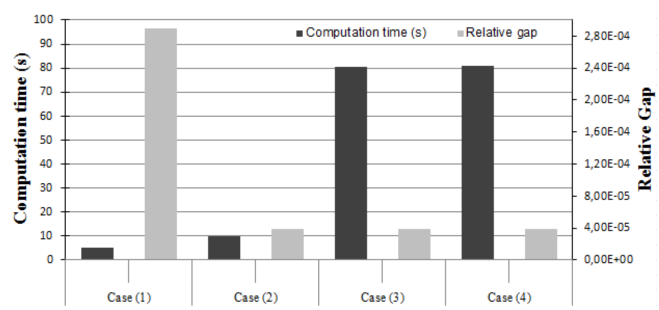

Fig. 4. Gradual addition of the short-periodic derivatives to the Lageos2 OD. Batch Least Squares case.

In conclusion, adding the Zonal short-periodic derivatives to the mean elements derivatives is sufficient to ensure the accuracy of the OD and maintain a respectable computation time. Prospects for improvement in the accuracy and in the computation time are discussed on section 6.

4.3. Kalman Filter Orbit Determination

In the same way as the Batch Least Squares algorithm, we carried out the DSST-OD with the Kalman Filter present in Orekit. Results were once again compared with the Orekit numerical OD.

Figure 5 and 6 present the results obtained by the Lageos2 DSST-OD considering only the mean elements derivatives.

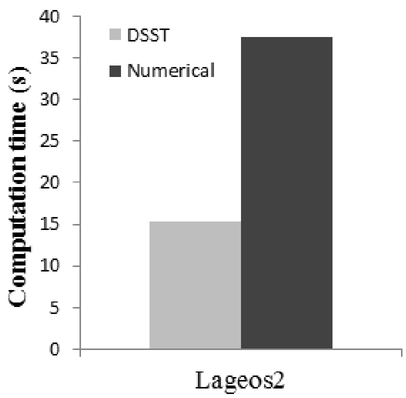

Fig. 5. Comparison between the numerical and the DSST-OD in terms of computation time. Kalman Filter case.

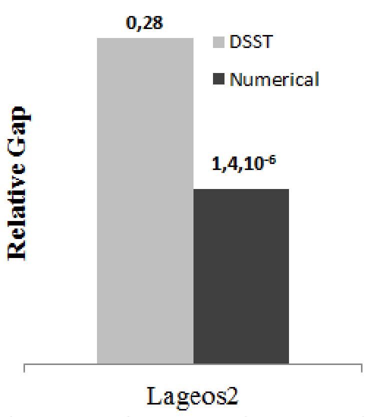

Fig. 6. Comparison between the numerical and the DSST-OD in terms of accuracy. Kalman Filter case.

Figure 7 shows the impact of adding the short-periodic terms derivatives to the Lageos2 OD.

It is important to note that the integration of the DSST in the Kalman Filter OD is under development. These results are only a first approach and improvements are still needed. However, it is possible to give a first interpretation of these results.

The conclusions in terms of accuracy and computation time are identical to section 4.2. At the moment, the results

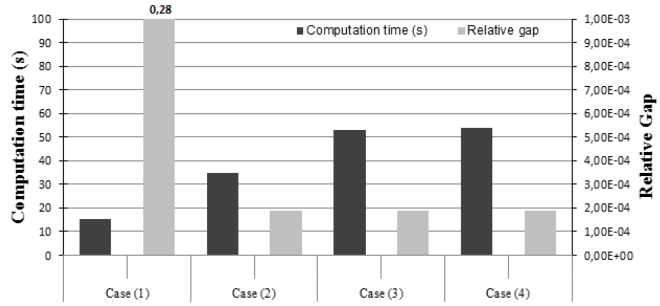

Fig. 7. Gradual addition of the short-periodic derivatives to the Lageos2 OD. Kalman Filter case.

obtained by the DSST-OD with the Batch Least Squares algorithm are better than the ones with the Kalman Filter. Indeed, computation time and accuracy are both better. Future work on the DSST-OD with the Kalman Filter will maybe reverse the current trend as show by Šegan's paper [12].

5. CONCLUSION

In this paper, we have demonstrated the performance of the Orekit implementation of the Draper Semi-analytical Theory under Lageos2 and GPS Orbit Determination conditions. The results are encouraging, they highlighted the ability of the DSST to perform Orbit Determinations with a Batch Least Squares algorithm or a Kalman Filter. The results also demonstrated that the user have the choice of using the mean elements or the osculating elements derivatives for the Orbit Determination. In addition, the results showed that, in some cases, the DSST is able to perform the Orbit Determination faster than the numerical propagator. Accuracy is better for the numerical propagator but we obtained small relative gaps with the Orekit implementation of the DSST.

All the work done in this paper will be published with release 10 of Orekit. They are nevertheless already available in a dedicated public branch of our source code management system [13]. In conclusion, open-source and semi-analytical Orbit Determination are not two opposing worlds !

6. FUTURE WORK

The authors propose to perform further implementation works in the following directions:

- Improving the performance of the DSST-OD in terms of computation time. This by optimizing the critical points highlighted in the Solar Radiation Pressure and the Tesseral force models.
- Improving the accuracy of the DSST-OD. When the user wants to initialize the Zonal and the Tesseral force

models, it must enter ten coefficients related to the short-periodic terms. However, we do not know the perfect combinations for these coefficients. We just know that they are sensible to the orbit type. Studies about the values of these coefficients must be perform in order to improve the accuracy of the DSST-OD. For our tests, we put the maximum values allowed for these coefficients. They correspond to the order and the degree of the Geo-potential.

- Improving the results of the DSST-OD with the Kalman Filter.

7. REFERENCES

[1] L. Maisonobe and V. Pommier-Maurussane, “Orekit: an Open-source Library for Operational Flight Dynamics Applications,” *4^{th} ICATT*, May 2010.

[2] S. Setty, P. J. Cefola, H. Fiedler, and J. F. San Juan Diaz, “Attributes affecting the Accuracy of a Batch Least Square Orbit Determination using Semi-analytical Satellite Theory,” *Advanced Maui Optical and Space Surveillance Technologies Conference*, January 2016.

[3] C. Wiedemann, S. K. Flegel, J. Gelhaus, H. Krag, H. Klinkrad, and P. Vrsmann, “NaK release model for MASTER-2009,” *10.1016/j.actaastro.2010.10.014*, 2011.

[4] NASA Orbital Debris Program Office, “Orbital Debris. Quarterly News,” January 2016.

[5] P. J. Cefola, Z. Folcik, R. Di-Costanzo, N. Bernard, S. Setty, and San Juan J. F., “Revisiting the DSST Standalone Orbit Propagator,” *AAS/AIAA Space Flight Mechanics Meeting*, January 2014.

[6] P. J. Cefola, “Equinoctial orbit elements Application to artificial satellite orbits,” *AIAA/AAS Conference*, September 1972.

[7] D. A. Daneilson, C. P. Sagavac, I. W. Early, and B. Neta, “Semianalytical Satellite Theory,” *Mathematical Department, Naval Postgraduate School, Monterey*, 1995.

[8] 65 Authors from the Astronautics Community, “Space Mission Analysis and Design,” *Space Technology Library*, October 1999.

[9] S. Setty, P. J. Cefola, O. Montenbruck, and H. Fiedler, “Application of Semi-analytical Satellite Theory orbit propagator to orbit determination for space object catalog maintenance,” p. 2218 to 2233, March 2016.

[10] D. Kalman, “Doubly Recursive Multivariate Automatic Differentiation,” *Mathematics Magazine*, vol. 75, pp. 187 to 202, 2002.

[11] L. Maisonobe, P. Parraud, M. Journot, and A. Alcarraz Garca, “Multi-satellites Precise Orbit Determination, an adaptable open-source implementation,” *15^{th} International Conference on Space Operations*, May 2018.

[12] S. Šegan and D. Marčeta, “Orbit Determination and Parameter Estimation: Extended Kalman Filter (EKF) Versus Least Squares Orbit Determination (LSQOD),” *Publ. Astron. Obs. Belgrade.*, vol. 86, pp. 197 to 202, September 2009.

[13] Orekit forge: `https://gitlab.orekit.org/orekit/orekit`.