

User Manual 3.3 Ephemeris

De Wiki

Aller à : [navigation](#), [rechercher](#)
[User Manual 3.3 Ephemeris](#)

Introduction

Scope

This section describes the propagation of spacecraftstates based on interpolation of position velocity ephemeris.

Javadoc

The concerned classes are available in the following packages :

Library	Javadoc
Orekit	Package org.orekit.utils
Patrius	Package fr.cnes.sirius.patrius.propagation

Links

Useful Documents

None as of now.

Package Overview

See the [Propagator](#) page for more generic information about propagators.

Propagation based on interpolation of position velocity ephemeris has been design to be very simple to use:

- Ephemeris propagators extend `AbstractPropagator` class. As a result all propagators features (event detection, etc.) are available.
- Ephemeris propagators are based on a `PVCoordinateProvider` to retrieve the orbital parameters at required date.
- Implementation of `PVCoordinateProvider` based on ephemeris interpolation (Hermite, or Lagrange).
- Implementation of `AdditionalStatesProvider` based on an ephemeris of additional states.

Features Description

PVCoordinatePropagator

The PV coordinate propagator extends `AbstractPropagator` and provides a simple way to propagate an initial spacecraft state coherent with the provided `PVCoordinatesProvider` and the eventual attitude and additional states provider.

The propagation of position velocity is based on the `getPVCoordinates` of the given `PVCoordinatesProvider`. The propagation of attitude is based on the given `AttitudeProvider`. The propagation of the additional states is based on the given `AdditionalStatesProvider`.

A classical simple use case is :

- `PVCoordinatesProvider` : One of the provider based on ephemeris interpolation (Lagrange or Hermite)
- `AttitudeProvider` : Null. The attitude of the propagated spacecraft state is null.
- `AdditionalStatesProvider` : Null. The additional states of the propagated spacecraft state is null.

A classical advanced use case is to add an attitude provider and the `SimpleAdditionalStateProvider` described in next paragraph.

Warning : Events detector with `Action.RESET_STATE` should not be used with this propagator fed with a "precomputed" PV coordinate provider (such as an ephemeris or an orbit object). In particular, impulse maneuvers cannot be used. Otherwise propagation will fail. This is due to the fact that the provided PV coordinate provider is not valid any more after the `Action.RESET_STATE` action.

SimpleAdditionalStateProvider

The `SimpleAdditionalStateProvider` is an implementation of `AdditionalStateProvider` based on additional states ephemeris.

The computation of an additional state for a given date is the following:

- If the given date is found in the ephemeris, the associated additional state vector is returned
- Otherwise a linear interpolation is performed.

PVCoordinateProvider Ephemeris

Two classes provide `PVCoordinates` at a given date from an ephemeris interpolation :

- `EphemerisPvLagrange` : Computes the `PVCoordinates` using a Lagrange interpolation of the ephemeris
- `EphemerisPvHermite` : Computes the `PVCoordinates` using a Hermite interpolation of the ephemeris

Both can be built from a list of spacecraft states or from list of dates- positions - velocities expressed in a given frame.

Lagrange interpolation

Intermediate states are computed with a Lagrange interpolator (order 8 by default) which :

- expects that spacecraft states are chronologically sorted (duplicates are allowed). If not sorted, no exception will be thrown and result might be wrong.
- is based on an instance of `PolynomialFunctionLagrangeForm`, which is stored at each interpolation to be reused for same interpolation interval, and updated if the interpolation is not in the same interval $t \in [t_{i}, t_{i+1}]$, where $i \in \{0, \dots, N\}$ and N is the size of the handled ephemeris.

The interpolation order n for `LagrangeEphemeris` should be an even number, and the interpolation date should correspond to an index that belongs in $n/2 - 1 \leq i \leq N - 1 - n/2$.

Hermite interpolation

Intermediate states are computed with a Hermite interpolator which :

- expects the spacecraft states to be chronologically sorted (duplicates are allowed). If not sorted, no exception will be thrown and result might be wrong.
- is based on an instance of `HermiteInterpolator`, which is stored at each interpolation to be reused for same interpolation interval, and updated if the interpolation is not in the same interval $t \in [t_{i}, t_{i+1}]$, where $i \in \{0, \dots, N\}$ and N is the size of the handled ephemeris (in closed-open convention).

For `HermiteEphemeris`, the interpolation can use the acceleration array if it is available.

PVCoordinateProvider AlmanacPVCoordinates

The class `AlmanacPVCoordinates` extends the `PVCoordinateProvider`. It provides `PVCoordinates` (position and velocity) of GNSS almanacs. GPS and Galileo almanacs can be provided using `GPSSAlmanacParameter` or `GalileoAlmanacParameter` classes.

Getting Started

[Modèle:SpecialInclusion prefix=\\$theme sub section="GettingStarted"/](#)

Contents

Interfaces

None as of now.

Classes

Class

Summary

Javadoc

IntegratedEphemeris	This class stores sequentially generated orbital parameters for later retrieval.	...
Ephemeris	This class is designed to accept and handle tabulated orbital entries. Tabulated entries are classified and then extrapolated in way to obtain continuous output, with accuracy and computation methods configured by the user.	...
AbstractEphemeris	This abstract class extends AbstractPropagator and implements BoundedPropagator. It contains some utility methods : to get the first and the last dates from a spacecraft states array whose length is controlled (>2), to get all the PV coordinates for all the interpolation points in one matrix.	...
LagrangeEphemeris	Deprecated since 3.1	...
HermiteEphemeris	Deprecated since 3.1	...
EphemerisPvLagrange	This class handles tabulated spacecraft states entries and implements PVCoordinatesProvider interface. Tabulated entries are chronologically classified. A Lagrange interpolation is performed to compute PV coordinates at a given date.	...
EphemerisPvHermite	This class handles tabulated spacecraft states entries and implements PVCoordinatesProvider interface. Tabulated entries are chronologically classified. A Hermite interpolation is performed to compute PV coordinates at a given date.	...
PVCoordinatePropagator	This class extends AbstractPropagator. It simply implements propagate orbit method using the getPvCoordinate from the PVCoordinate provider it handles. The propagator is initialized with given initial date, frame and mu to build a coherent initial spacecraftstate. This initialisation calls the PVCoordinateprovider getPvCoordinate method to the given date which can raise an error if the date is out of bound for exemple. This class will be basically used with EphemerisPvLagrange and EphemerisPvHermite
SimpleAdditionalStateProvider	This class implements AdditionalStateProvider interface. It simply handles a list of dates associated to additional state vectors. For a given date it returns the associated additional state vector or a propagation exception if this date is not found. This class will be basically used with PVCoordinatePropagator
AlmanacPVCoordinates	This class computes PV coordinates at any date from a GPS or Galileo Almanac.	...

Tutorials

Tutorial 1

[Modèle:SpecialInclusion prefix=\\$theme sub section="Tuto1"/](#)

Tutorial 2

[Modèle:SpecialInclusion prefix=\\$theme sub section="Tuto2"/](#)

Tips & Tricks

[Modèle:SpecialInclusion prefix=\\$theme sub section="Tips"/](#)

Récupérée de « http://patrius.cnes.fr/index.php?title=User_Manual_3.3_Ephemeris&oldid=1550 »
Catégorie :

- [User Manual 3.3 Orbit Propagation](#)

Menu de navigation

Outils personnels

- [18.220.206.141](#)
- [Discussion avec cette adresse IP](#)
- [Créer un compte](#)
- [Se connecter](#)

Espaces de noms

- [Page](#)
- [Discussion](#)

Variantes

Affichages

- [Lire](#)
- [Voir le texte source](#)
- [Historique](#)
- [Exporter en PDF](#)

Plus

Rechercher

PATRIUS

- [Welcome](#)

Evolutions

- [Main differences between V4.15 and V4.14](#)
- [Main differences between V4.14 and V4.13](#)
- [Main differences between V4.13 and V4.12](#)
- [Main differences between V4.12 and V4.11](#)
- [Main differences between V4.11 and V4.10](#)
- [Main differences between V4.10 and V4.9](#)
- [Main differences between V4.9 and V4.8](#)
- [Main differences between V4.8 and V4.7](#)
- [Main differences between V4.7 and V4.6.1](#)
- [Main differences between V4.6.1 and V4.5.1](#)
- [Main differences between V4.5.1 and V4.4](#)
- [Main differences between V4.4 and V4.3](#)
- [Main differences between V4.3 and V4.2](#)
- [Main differences between V4.2 and V4.1.1](#)
- [Main differences between V4.1.1 and V4.1](#)
- [Main differences between V4.1 and V4.0](#)
- [Main differences between V4.0 and V3.4.1](#)

User Manual

- [User Manual 4.15](#)
- [User Manual 4.14](#)
- [User Manual 4.13](#)
- [User Manual 4.12](#)
- [User Manual 4.11](#)
- [User Manual 4.10](#)
- [User Manual 4.9](#)
- [User Manual 4.8](#)
- [User Manual 4.7](#)
- [User Manual 4.6.1](#)
- [User Manual 4.5.1](#)
- [User Manual 4.4](#)
- [User Manual 4.3](#)
- [User Manual 4.2](#)
- [User Manual 4.1](#)
- [User Manual 4.0](#)
- [User Manual 3.4.1](#)
- [User Manual 3.3](#)

Tutorials

- [Tutorials 4.15](#)

- [Tutorials 4.14](#)
- [Tutorials 4.13.5](#)
- [Tutorials 4.12.1](#)
- [Tutorials 4.8.1](#)
- [Tutorials 4.5.1](#)
- [Tutorials 4.4](#)
- [Tutorials 4.1](#)
- [Tutorials 4.0](#)

Links

- [CNES freeware server](#)

Navigation

- [Accueil](#)
- [Modifications récentes](#)
- [Page au hasard](#)
- [Aide](#)

Outils

- [Pages liées](#)
- [Suivi des pages liées](#)
- [Pages spéciales](#)
- [Adresse de cette version](#)
- [Information sur la page](#)
- [Citer cette page](#)

- Dernière modification de cette page le 20 mars 2018 à 16:27.
- [Politique de confidentialité](#)
- [À propos de Wiki](#)
- [Avertissements](#)
- 