

63-я научная конференция МФТИ

23–29 ноября 2020 года, Москва



Высокоуровневое моделирование управляемого орбитального движения в KIAM Astrodynamics Toolbox

М.Г. Широбоков

Институт прикладной математики им. М.В. Келдыша РАН

Содержание

- Актуальность и цель работы
- Существующее программное обеспечение
- Цели и существо KIAM Astrodynamics Toolbox
- Структура тулбокса
- Основные классы тулбокса и демонстрация их работы
- Заключение

Актуальность и цель работы

- Сложные проекты с индустрией (РКК Энергия, Beyond Atlas)
- Цель – разработать высокоуровневый программный инструментарий, который бы позволял:
 - Проводить быстрый анализ траекторий
 - Быстро строить картинки хорошего качества
 - Быстро переключаться между различными способами описания траекторий

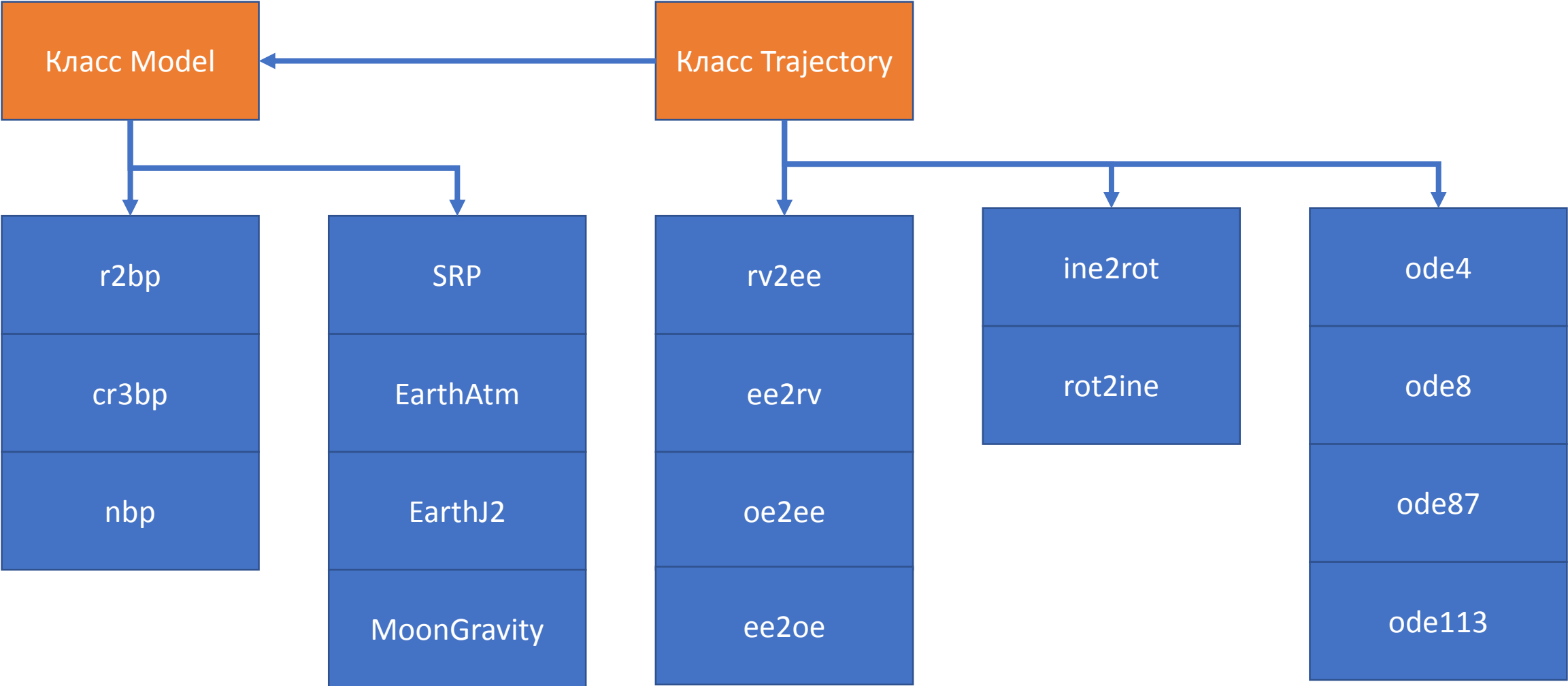
Существующее программное обеспечение

- Satellite Tool Kit (STK) компании Analytical Graphics
- FreeFlyer компании a.i. solutions
- General Mission Analysis Tool (GMAT) агентства NASA
- Basilisk лаборатории автономных транспортных систем университета Колорадо (H. Schaub)
- KIAM Satellite Modeller нашего отдела

KIAM Astrodynamics Toolbox (v. 1.0)

- Тулбокс представляет собой набор низкоуровневых и высокоуровневых инструментов для проектирования управляемого и неуправляемого орбитального движения в дальнем космосе
- Разработка высокоуровневой части ведется с конца августа 2020 г.
- Пишется на языке MATLAB, планируется также на Fortran
- Готова первая версия, скоро будет вторая дополненная

Структура KIAM Astrodynamics Toolbox



Класс Model

- Задача класса Model – подготовить для пользователя указатель на функцию правых частей как на черный ящик
- Общий синтаксис:

```
md = Model(vars, type, primary, sources_cell, JD_interval, JD_Zero)
```

- Пример использования:

```
md = Model('rv', 'cr3bp_sb', 'EarthMoon', {});  
options = odeset('AbsTol', 1e-10, 'RelTol', 1e-10);  
x0 = [-0.5; 0; 0; 0; 0.4; 0];  
[~, X] = ode113(@(t, x) md.eqs(t, x), [0, 2*pi], x0, options);  
plot(X(:, 1), X(:, 2), 'LineWidth', 2.0);  
axis equal
```

Возможности класса Model на текущий момент

- Невозмущенная задача двух тел
- Круговая ограниченная задача трех тел
- Задача n тел с выбором возмущений:
 - Сила светового давления с расчетом интенсивности света
 - Сила атмосферного сопротивления (CIRA-2012)
 - Силы гравитационного притяжения планет Солнечной системы
 - Сила гравитационного притяжения к Луне + сложное поле Луны 8×8
- Свойство `md.control` может содержать указатель на пользовательскую функцию управления

Класс Trajectory

- Задача класса Trajectory – представить траекторию аппарата как инвариантный относительно способа описания объект
- Общий синтаксис:

```
tr = Trajectory(x0, t0, jd0, vars, system, units, varargin)
```

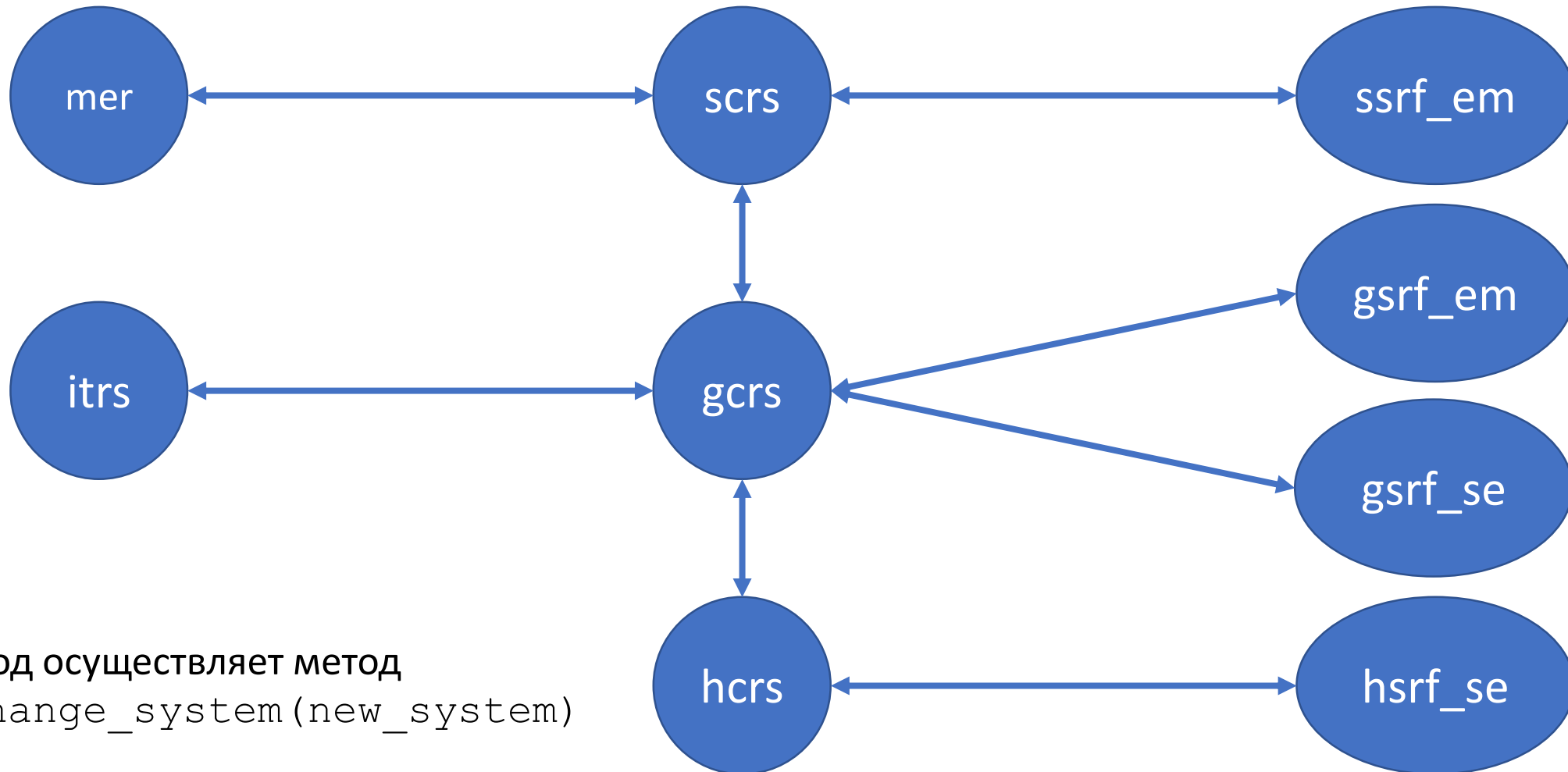
- Пример использования:

```
t0 = 0;  
x0 = [7000;0;0;0.1;8;0.2];  
jd0 = juliandate(2028,1,1);  
tr = Trajectory(x0,t0,jd0,'rv','gcrs','dim');  
tr.set_model('rv','r2bp','Earth',{});  
tr.propagate(10);  
tr.show('3d')
```

Три «взгляда» на траекторию

- Переменные:
 - Фазовые переменные (с массой или без массы аппарата)
 - Орбитальные элементы (с массой или без массы аппарата)
- Система координат
 - Международная небесная с/к с началом в Земле, Луне
 - Вращающаяся с/к с началом в Земле, Луне, Солнце
- Система единиц
 - Размерная (км, км/с, дни)
 - Безразмерная (несколько вариантов)

Граф преобразований между системами координат



Переход осуществляет метод
`tr.change_system(new_system)`

Пример использования с автоматическим преобразованием

```
t0 = 0;
oe0 = [7000;0.01;51.6/180*pi;0;0;0];
jd0 = juliandate(2028,1,1);
tr = Trajectory(oe0,t0,jd0,'oe','itrs','dim');
tr.set_model('rv','nbp','Earth',{'Sun','Moon','J2','atm','SRP'});
tr.model.data.area = 2;
tr.model.data.mass = 40;
tr.propagate(10);
tr.change_units('dim');
tr.show('xy');
axis equal
```

Заключение

- Разработана первая версия авторского тулбокса KIAM Astrodynamics Toolbox (KAT) на языке MATLAB для моделирование управляемого и неуправляемого движения с пользовательскими функциями управления
- Тулбокс состоит из двух частей – низкоуровневой и высокоуровневой части, пользователь может использовать любые функции по желанию, нет графического интерфейса
- В классе Trajectory реализована автоматическая трансляция переменных движения, системы координат и системы единиц под нужную для модели и для пользователя