

Федеральное государственное автономное образовательное учреждение
высшего образования
"Национальный исследовательский университет
"Высшая школа экономики"

Московский институт электроники и математики им. А.Н.Тихонова

Направление подготовки/специальности
«01.03.04 Прикладная математика»
Образовательная программа «Прикладная математика»

ОТЧЁТ

о прохождении
производственной практики

Студент Хлебко Никита Игоревич БПМ 174

Руководитель практики студента:

МИЭМ НИУ ВШЭ, Старший преподаватель Бобер С.А. Старший преподаватель

МИЭМ НИУ ВШЭ, Доцент Внуков А.А. к.т.н., PhD, доцент

Практика пройдена с оценкой _____

Дата _____

Москва, 2020

Содержание

1. Введение	3
2. Содержательная часть	3
2.1. Краткая характеристика организации	3
2.2. Описание профессиональных задач, решаемых студентом на практике	3
3. Исполненное индивидуальное задание	4
3.1. Обзор возможностей GMAT (General Mission Analysis Tool)	4
3.2. Необходимые критерии	4
3.3. Обзор библиотек Python для баллистического проектирования космических миссий	4
4. Моделирование полёта спутника от Земли к Марсу	7
5. Заключение	8
6. Приложение	9

1. Введение

- Сделать обзор возможностей программного продукта GMAT (General Mission Analysis Tool).
- Провести поиск и сделать обзор библиотек Python для баллистического проектирования космических миссий с аналогичным функционалом.
- Выполнение расчетов и сравнение результатов, сделанных в Python и GMAT.

2. Содержательная часть

2.1. Краткая характеристика организации

- Практика проходила в Московском институте электроники и математики имени А. Н. Тихонова Национального исследовательского университета «Высшая школа экономики», дистанционно.

2.2. Описание профессиональных задач, решаемых студентом на практике

- Во время практики были произведены сбор, систематизация и обобщение материалов, подготовлен обзор программных продуктов для проведения дальнейших исследований в рамках выпускной квалификационной работы.

3. Исполненное индивидуальное задание

3.1. Обзор возможностей GMAT (General Mission Analysis Tool)

- General Mission Analysis Tool (GMAT) – это многофункциональная система, содержащая модели космических систем высокой точности, наборы из методов оптимизации, интеграторов и симплектических интеграторов, а также инструменты для работы с графиками и данными для проведения более гибкого анализа и моделирования миссий разных сложностей. Управление GMAT осуществляется с помощью полнофункционального интерактивного графического интерфейса или программным кодом, составленным пользователем. Ключевыми особенностями General Mission Analysis Tool являются: возможность использовать высокоточные модели небесных тел и редактировать их параметры, поддержка работы со временем и координатами, в том числе включая преобразование координат из одной системы в другую и конвертацию времени, использование заготовленных или написанных пользователем интеграторов, симплектических интеграторов и методов оптимизации, построение анимированных 3D графиков миссий с возможностью их вращения и приближения, поддержка миссий с различными режимами полётов, начиная от околоземной орбиты до лунной и миссий в глубоком космосе. Также, GMAT имеет Python и Matlab-интерфейсы.

3.2. Необходимые критерии

- Поддержка стандартных эфемеридных моделей для тел Солнечной системы. Данный критерий позволяет работать с данными, которые описывают движение астрономических объектов в Солнечной системе, данные содержат набор небесных координат, вычисленных через равные промежутки времени.
- Процедуры для преобразования между различными системами координат помогают производить переходы к более простым и удобным для последующих вычислений формам моделей.
- Набор интеграторов – инструменты, необходимые для поиска решений систем дифференциальных уравнений, которые используются в моделирование космических миссий.
- Набор симплектических интеграторов – это интеграторы для поиска численного решения уравнений Гамильтона.
- Функции расчета геометрических событий – функции, позволяющие узнать, когда космический аппарат достигнет определенной точки на своей орбите, плоскости, приблизится к какому-то объекту.
- Метод пристрелки – метод (или метод аналогичный ему), который позволяет свести краевую задачу к задаче Коши для проведения последующих решений интегратором.
- Методы оптимизации – набор методов, который позволяют находить оптимальные коэффициенты для системы.

3.3. Обзор библиотек Python для баллистического проектирования космических миссий

- Plyades – библиотека, созданная на базе Poliastro. В Plyades присутствуют функции преобразования систем координат из одной в другую, но отсутствуют модули для работы со временем, интеграторами и эфемеридными моделями. Также, у этой библиотеки отсутствует обширная документация, из-за чего с библиотекой работать с ней трудно.
- Poliastro – Python библиотека с открытым программным кодом, работающая на операционных системах Linux, macOS и Windows. Библиотека имеет модуль с интегратором DOPRI (Dormand–Prince method) и модули работы с эфемеридами, временем и методами оптимизации, которые заимствуются у Astropy и SciPy соответственно. В библиотеке есть функции, позволяющие обработать геометрические события, но только для ограниченного количества для небесных тел. Также, в Poliastro присутствует Newton's method, который является аналогом метода Стрельбы. Библиотека имеет обширную документацию с большим количеством примеров и удобный API.
- Рукер - библиотека, разработанная Европейским космическим агентством, основной уклон делается на алгоритмическую эффективность. Отличительной особенностью этой библиотеки являются методы оптимизации. Рукер имеет большое количество инструментов и функций для решения задач оптимизации и реализовано большое количество алгоритмов оптимизации, такие как: PSO, SGA,

SADE. Также, осуществляется поддержка эфемеридных моделей для Солнечной системы, но библиотека не поддерживает различные преобразование координат и времени, расчеты геометрических событий. В Pykep реализованы интеграторы Рунге-Кутты и Тейлора, симплектические интеграторы и метод пристрелки отсутствуют. Библиотека имеет большую документацию с большим количеством примеров практически к каждой функции, работает на Linux, Windows, MAC.

- Astropy – инструмент для проведения расчетов в условиях моделирования космических миссий. Библиотека не имеет интеграторов и симплектических интеграторов, но в ней реализованы функции для конвертации времени и работы с ним. Также, есть модули для работы с эфемеридными моделями и координатами, в том числе включает в себя функции для преобразования координат из одной системы в другую и метод Леверье — Марквардта, являющийся альтернативой методу Ньютона. Аналогично Poliastro библиотека имеет большую документацию, работает на операционных системах Linux, macOS и Windows.
- Rebound – модуль для расчета траектории объекта. Акцент в модуле сделан на более гибкий инструмент настройки миссий и большое количество интеграторов. В Rebound присутствуют интеграторы: IAS15, MERCURIUS, Wisdom-Holman. Кроме этого, в модуле реализованы симплектические интеграторы Symplectic Epicycle Integrator и JANUS. Также, Rebound может работать с сервисом Horizons и имеет функционал для преобразования координат. Но в этом модуле отсутствует такая же гибкая работа со временем, как в Plyades, Astropy. Кроме того, в Rebound не включены модули с методами оптимизации. Модуль работает на операционных системах Linux и macOS и имеет документацию с примерами.
- Pysofa и Pysofa2 – модули созданные на базе библиотеки SOFA. Библиотеки имеют функции конвертации времени и преобразования координат. Pysofa и Pysofa2 не имеют интеграторов и симплектических интеграторов, отсутствует обработка геометрических событий. Модули различаются между собой набором требований к установке. Pysofa2 является более автономной версией Pysofa и не требует установленной библиотеки SOFA, модули имеют документацию с небольшим количеством примеров, работает на macOS, Linux и Windows.
- Orekit – JAVA библиотека, имеющая Python-интерфейс. Использует устаревшую версию библиотеки (6.1). В данной версии поддерживаются эфемеридные модели, работает с сервисом Horizons. Имеет большой набор интеграторов: Runge-Kutta method, DOPRI, Adams-Bashfort, Euler, также есть симплектический интегратор midpoint integrator. Orekit не имеет конвертеров времени, отсутствуют функции для преобразования координат, проведения геометрических расчетов и обработки геометрических событий и также отсутствуют различные методы оптимизации. Для Orekit написана только Java документация, но имеются примеры использования на Python в репозитории проекта, также библиотека не работает на Windows.

	Эфемеридные модели для СС	UTC в эфемеридное	Наличие преобразования координат
Plyades	нет	нет	да
Poliastro	да (из Astropy)	да (из Astropy)	да (из Astropy)
Pykep	да	нет	нет
Astropy	да	да	да
Rebound	да (Horizon)	нет	да
Pysofa	нет	да	да
Pysofa2	нет	да	да
Orekit	да	нет	нет

Таблица 1. Сравнение функционала библиотек и модулей Python. Часть 1.

Интеграторы	Симплектические интеграторы	Расчет геометрических событий
нет	нет	нет
DOPRI (Dormand-Prince method)	нет	да
RK, Taylor (из scipy)	нет	нет
нет	нет	нет
IAS15, MERCURIUS, Wisdom-Holman	Symplectic Epicycle Integrator, JANUS	нет
нет	нет	нет
нет	нет	нет
RK, DOPRI, Adams-Bashfort, Euler	Midpoint Integrator	нет

Таблица 1. Сравнение функционала библиотек и модулей Python. Часть 2.

Пристрелка	Методы оптимизации	Документация	ОС
нет	нет	отсутствует	нет информации
Newton's method	да (из scipy)	да	Linux, OS X, Windows
нет	да (из scipy)	да	Linux, OS X, Windows
нет	да	да	Linux, OS X, Windows
нет	нет	да	Linux, MAC
нет	нет	да	Linux, OS X, Windows
нет	нет	да	Linux, OS X, Windows
нет	нет	да (для java)	Linux, MAC

Таблица 1. Сравнение функционала библиотек и модулей Python. Часть 3.

	Источник	Научные работы
Plyades	https://clck.ru/PvoKZ	https://clck.ru/Pvojm
Poliastro	https://clck.ru/PvoMF	https://clck.ru/PvpWK , https://goo.su/1RG2 , https://goo.su/1rg4 , https://goo.su/1Rg4
Pykep	https://clck.ru/PvoNA	https://goo.su/1Rki , https://goo.su/1rkj , https://goo.su/1rKj , https://goo.su/1RKJ
Astropy	https://clck.ru/PvoP9	https://goo.su/1rkE , https://goo.su/1RkD , https://goo.su/1rKd
Rebound	https://clck.ru/PvoQe	https://goo.su/1rt5 , https://goo.su/1rt6 , https://goo.su/1Rt6 , https://goo.su/1RT6 , https://goo.su/1Rkk
Pysofa	https://clck.ru/PvoY4	нет
Pysofa2	https://clck.ru/Pvoai	нет
Orekit	https://clck.ru/PvodF	https://goo.su/1rkf , https://goo.su/1rK4 , https://goo.su/1rk5 , https://goo.su/1rk7

Таблица 2. Источники библиотек и научные работы с использованием библиотек Python.

4. Моделирование полёта спутника от Земли к Марсу

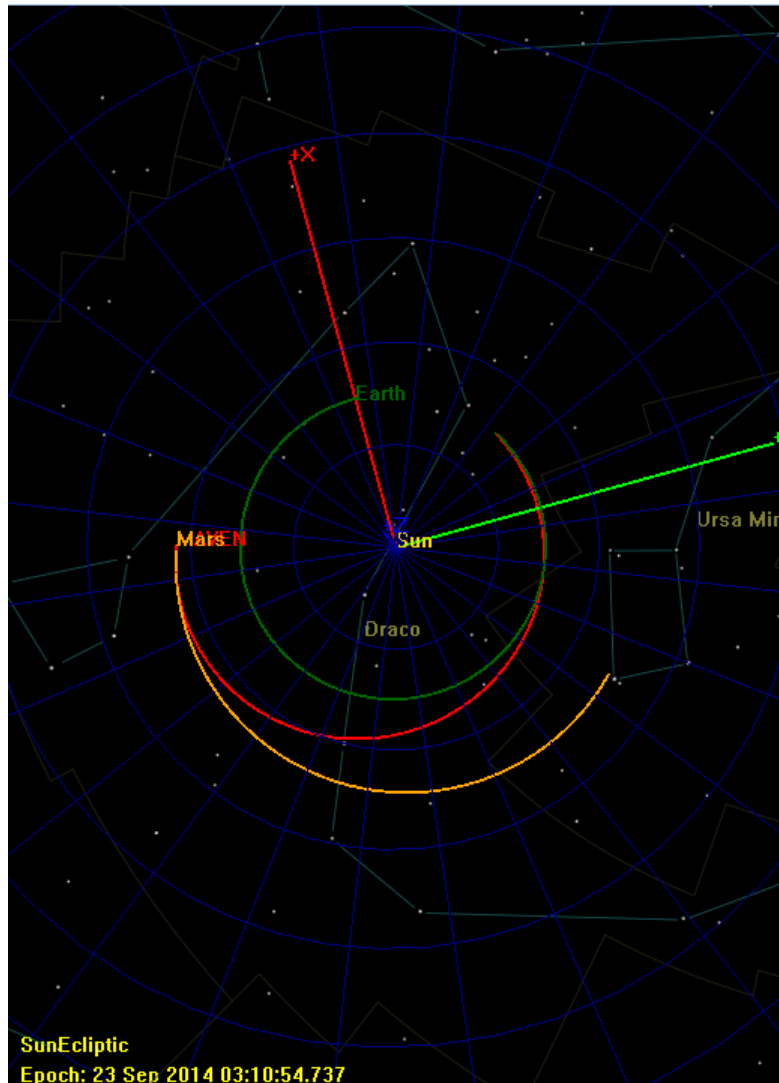


Рисунок 1. Результат моделирования миссии в GMAT

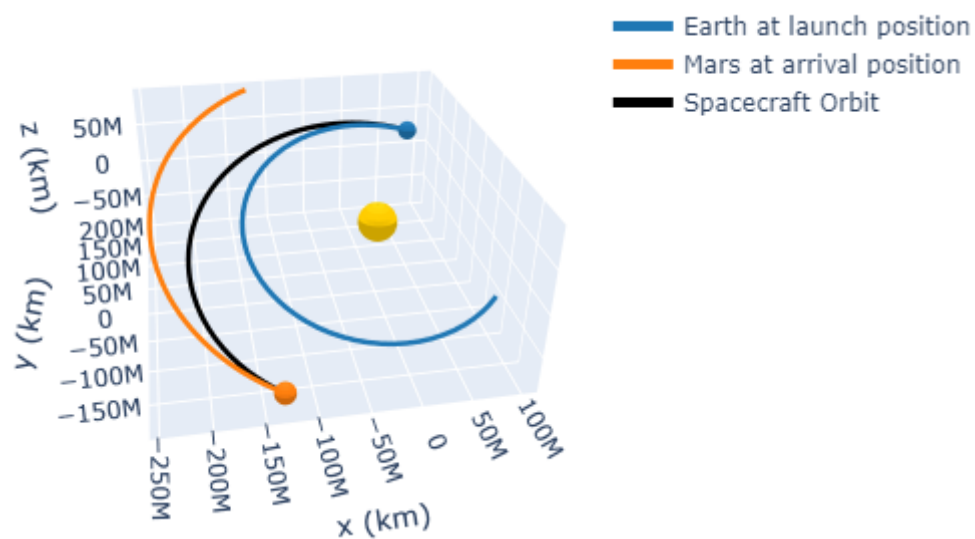


Рисунок 2. Результат моделирования миссии в Python

5. Заключение

- Анализируя произведенные вычисления, обзоры возможностей General Mission Analysis Tool и библиотек, модулей Python, которые были выполнены во время производственной практики, выявлены и выбраны для продолжения работы в рамках выпускной квалификационной работы две библиотеки: Poliastro и Astropy. Так как использование двух этих библиотек вместе делает проектирование космических миссий наиболее удобным и похожим на General Mission Analysis Tool, позволяют использовать наиболее похожий набор инструментов в Python и точность вычислений имеют минимальную погрешность. Также, во время практики, были улучшены навыки использования программных продуктов, разработки программного обеспечения в соответствии с существующими стандартами и оформления результатов работы в соответствии с существующими стандартами.

6. Приложение

	Эфемеридные модели для СС	UTC в эфемеридное	Наличие преобразования координат
Plyades	нет	нет	да
Poliastro	да (из Astropy)	да (из Astropy)	да (из Astropy)
Pykep	да	нет	нет
Astropy	да	да	да
Rebound	да (Horizon)	нет	да
Pysofa	нет	да	да
Pysofa2	нет	да	да
Orekit	да	нет	нет

Таблица 1. Сравнение функционала библиотек и модулей Python. Часть 1.

Интеграторы	Симплектические интеграторы	Расчет геометрических событий
нет	нет	нет
DOPRI (Dormand–Prince method)	нет	да
RK, Taylor (из scipy)	нет	нет
нет	нет	нет
IAS15, MERCURIUS, Wisdom-Holman	Symplectic Epicycle Integrator, JANUS	нет
нет	нет	нет
нет	нет	нет
RK, DOPRI, Adams-Bashfort, Euler	Midpoint Integrator	нет

Таблица 1. Сравнение функционала библиотек и модулей Python. Часть 2.

Пристрелка	Методы оптимизации	Документация	ОС
нет	нет	отсутствует	нет информации
Newton's method	да (из scipy)	да	Linux, OS X, Windows
нет	да (из scipy)	да	Linux, OS X, Windows
нет	да	да	Linux, OS X, Windows
нет	нет	да	Linux, MAC
нет	нет	да	Linux, OS X, Windows
нет	нет	да	Linux, OS X, Windows
нет	нет	да (для java)	Linux, MAC

Таблица 1. Сравнение функционала библиотек и модулей Python. Часть 3.

	Источник	Научные работы
Plyades	https://clck.ru/PvoKZ	https://clck.ru/Pvojm
Poliastro	https://clck.ru/PvoMF	https://clck.ru/PvpWK , https://goo.su/1RG2 , https://goo.su/1rg4 , https://goo.su/1Rg4
Pykep	https://clck.ru/PvoNA	https://goo.su/1Rki , https://goo.su/1rkj , https://goo.su/1rKj , https://goo.su/1RKJ
Astropy	https://clck.ru/PvoP9	https://goo.su/1rkE , https://goo.su/1RkD , https://goo.su/1rKd
Rebound	https://clck.ru/PvoQe	https://goo.su/1rt5 , https://goo.su/1rt6 , https://goo.su/1Rt6 , https://goo.su/1RT6 , https://goo.su/1Rkk
Pysofa	https://clck.ru/PvoY4	нет
Pysofa2	https://clck.ru/Pvoai	нет
Orekit	https://clck.ru/PvodF	https://goo.su/1rkf , https://goo.su/1rK4 , https://goo.su/1rk5 , https://goo.su/1rk7

Таблица 2. Источники библиотек и научные работы с использованием библиотек Python.

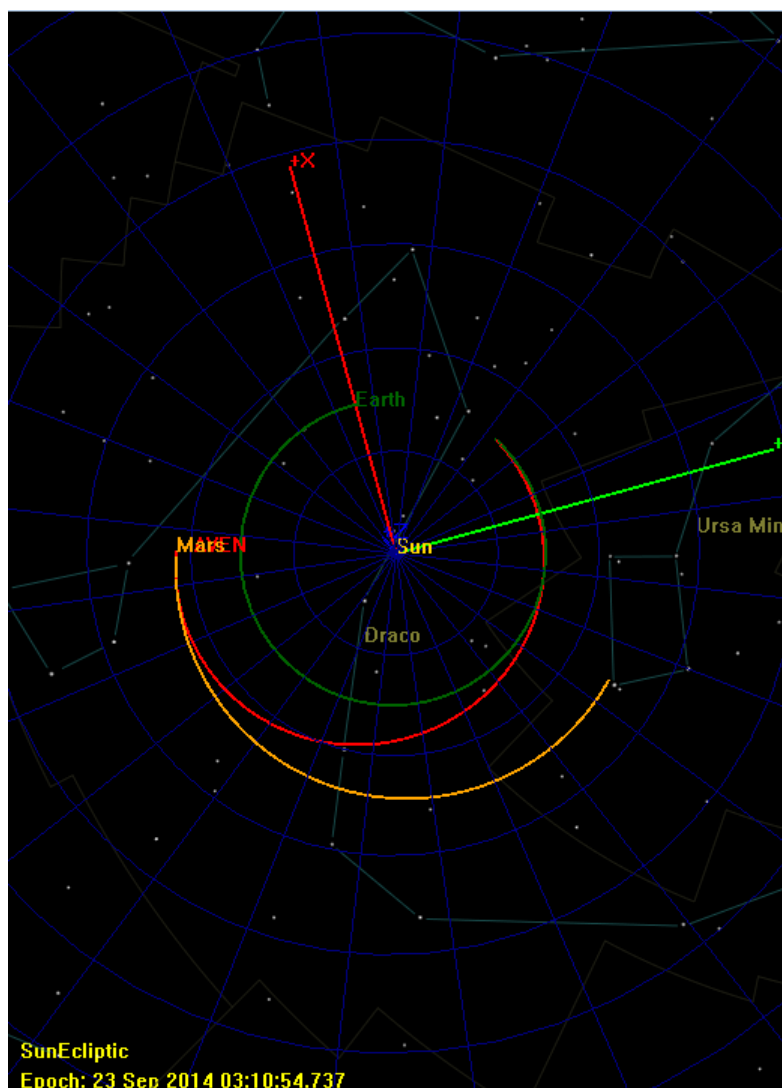


Рисунок 1. Результат моделирования миссии в GMAT

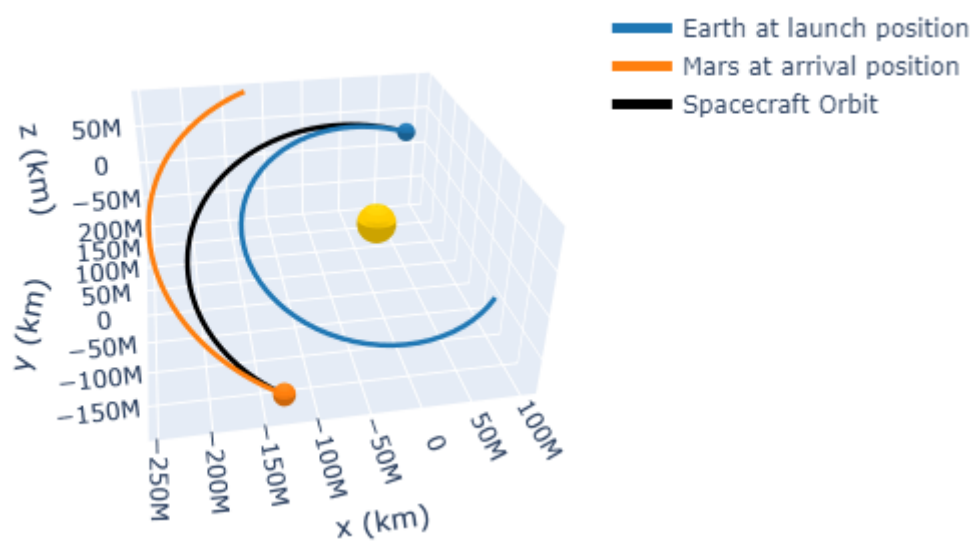


Рисунок 2. Результат моделирования миссии в Python