

# Разработка программного обеспечения для модельно-ориентированного проектирования тросовой системы переноса груза квадрокоптером

Рыжевнин Максим, Бутковский Владислав  
Научный руководитель - Валерий Чернов,  
Место прохождения - Геоскан



## Содержание

1. Введение
2. Постановка задач
3. Методика
4. Результат
5. Благодарности и список литературы

# Введение



## Цели:

- Составить математическую модель дрона с грузом на тросе
- Написать программу, рассчитывающую оптимальный путь

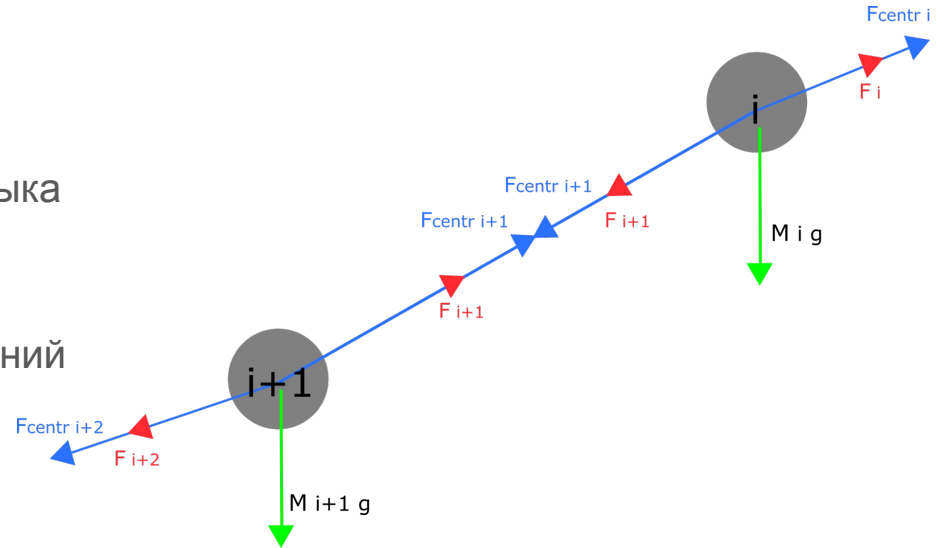
## Актуальность:

Возможны сценарии использования квадрокоптеров, в которых:

- Нет возможности использовать каркасные конструкции
- Требуется точность и быстрое выполнение задачи без шанса на ошибку
- Встраиваемых библиотек, позволяющих проводить расчеты движения тросов мало

## Постановка задач

1. Составить мат.модель троса.
2. Решить эту систему с помощью языка *python*.
3. Создать модель дрона с тросом.
4. “Научить” дрон избегать столкновений троса с объектами окружения.

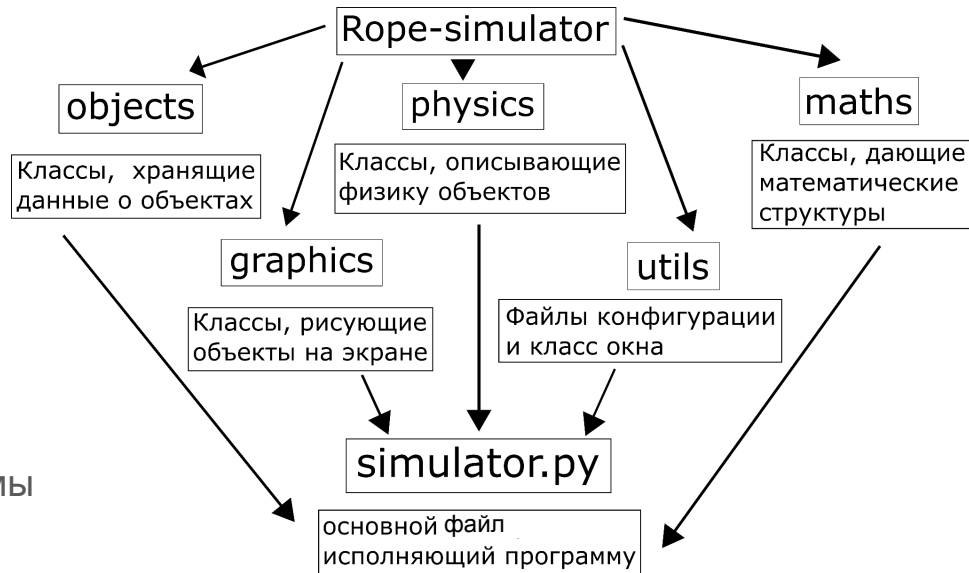


# Структура программы

Мы используем язык *python* для программы.

Наша программа состоит из различных классов, представляющих нам функциональность.

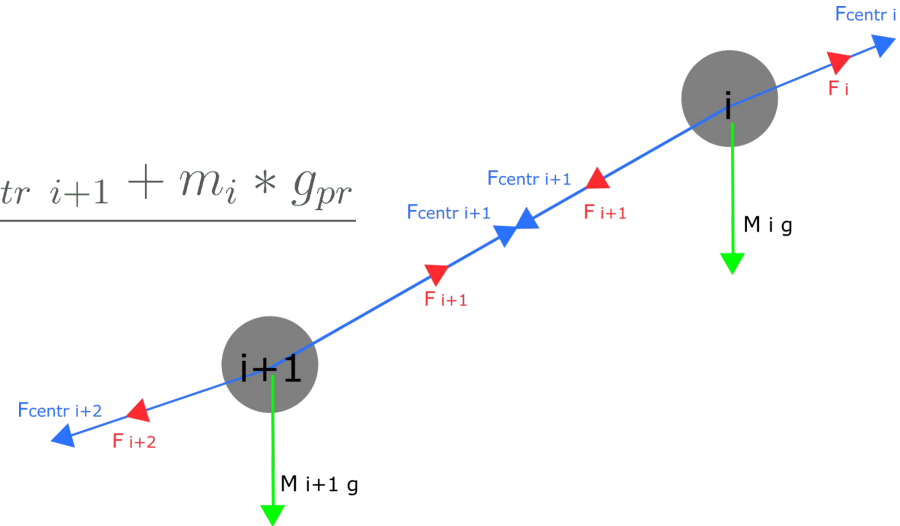
С помощью всего этого в файле *simulator.py* мы создаем симуляцию.



# Методика. Создание мат. модели

$$a_{i_{pr}} = a_{i+1_{pr}}$$

$$a_{i_{pr}} = \frac{F_{i_{pr}} - F_{i+1} + F_{centr\ i_{pr}} - F_{centr\ i+1} + m_i * g_{pr}}{m_i}$$



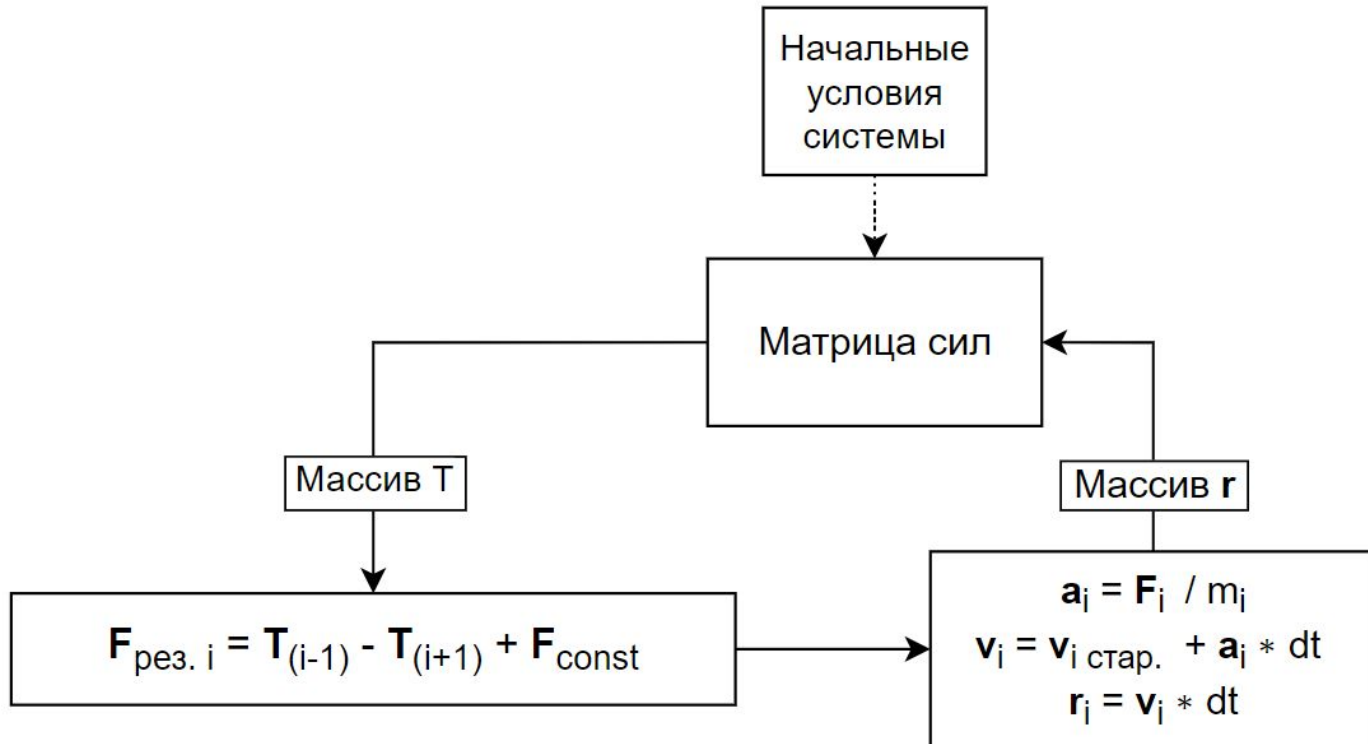
$$\begin{aligned} F_{i_{pr}} * m_{i+1} - F_{i+1} * (m_i + m_{i+1}) + F_{i+2_{pr}} * m_i = \\ = -F_{centr\ i_{pr}} * m_{i+1} + F_{centr\ i+1} * (m_i + m_{i+1}) - F_{centr\ i+2_{pr}} * m_i \end{aligned}$$

# Методика. Решение системы уравнений

$$\begin{bmatrix} -m_0 - m_1 & m_0 \cos \alpha_1 & 0 & 0 & \dots & 0 & 0 & 0 \\ m_2 \cos \alpha_0 & -m_1 - m_2 & m_1 \cos \alpha_2 & 0 & \dots & 0 & 0 & 0 \\ 0 & m_3 \cos \alpha_1 & -m_2 - m_3 & m_2 \cos \alpha_3 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & -m_{n-4} - m_{n-3} & m_{n-4} \cos \alpha_{n-2} & 0 \\ \dots & \dots & \dots & \dots & \dots & m_{n-2} \cos \alpha_{n-4} & -m_{n-3} - m_{n-2} & m_{n-3} \cos \alpha_{n-1} \\ \dots & \dots & \dots & \dots & \dots & 0 & m_{n-1} \cos \alpha_{n-3} & -m_{n-2} - m_{n-1} \end{bmatrix}$$

$$\begin{pmatrix} F_{centr\ 1} * (m_0 + m_1) - F_{centr\ 2_{pr}} * m_1 \\ \dots \\ -F_{centr\ i_{pr}} * m_{i+1} + F_{centr\ i+1} * (m_i + m_{i+1}) - F_{centr\ i+2_{pr}} * m_i \\ \dots \\ -F_{centr\ n-2_{pr}} * m_{n-1} + F_{centr\ n-1} * (m_{n-2} + m_{n-1}) \end{pmatrix}$$

# Методика. Структура мат. модели



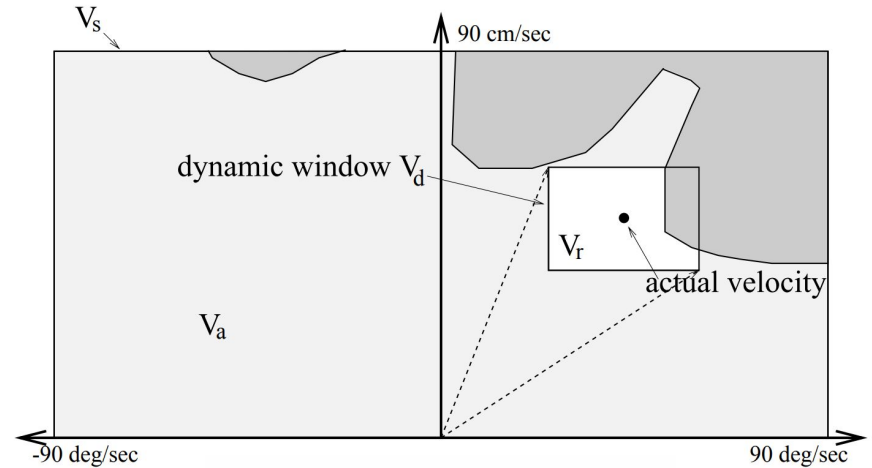


# Методика. Метод динамического окна

В методе дин.окна мы рассматриваем какие скорости мы можем достичь.

Проверяем, какие из них безопасны.

Выбираем ту из достижимых и безопасных, которая даст наибольшее преимущество.



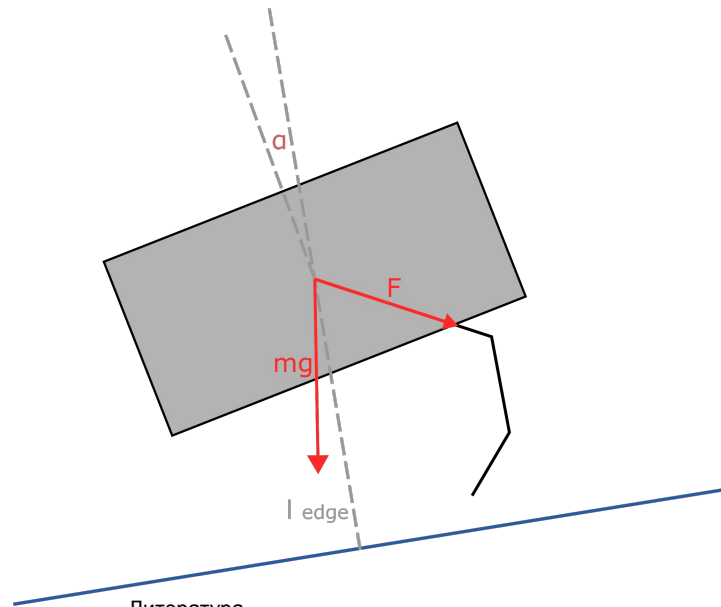
## Методика. Метод динамического окна

Скорость достижима, если:

$$|V_{now} - V_{check}| \leq a_{max} * dt$$

Скорость безопасна, если:

$$\frac{l_{end}}{v_{perp}} \geq - \frac{l_{full} - l_{edge}}{v_{drone}}$$



Литература

The Dynamic Window Approach to Collision Avoidance - Dieter Fox, Wolfram Burgard, Sebastian Thrun



## Методика. Метод динамического окна

Уравнения полезности позволяют выбрать лучший вариант из всех возможных и безопасных.

$$benefit_{speed} = \frac{1}{1 + (v_{drone} - v_{need})^2}$$

$$benefit_{dist} = \frac{1}{1 + (l_{edge} - l_{need})^2}$$

$$benefit_{heading} = \cos(\text{angle}(\text{vec}_{goal}, v_{drone})) = \frac{vec_{goal} * v_{drone}}{|v_{drone}| * |vec_{goal}|}$$

# Полученный результат и его анализ

## Выполнено:

- Получена программа, способная рассчитывать и визуализировать перемещение троса
- Реализована мат. модель дрона и метод динамического окна


## Осталось выполнить:


- Реализовать новый метод решения мат. модели троса, совместимый с мат. моделью дрона
- Смоделировать полет дрона с подвешенным грузом при заданной 2D модели окружения



# Возможные пути развития проекта



1. Увеличение скорости вычислений, проверка их точности и влияния ошибки на работу программы.
  2. Перейти от модели троса на плоскости к пространственной модели.
  3. Добавление сценариев и их выполнение.
  4. Реализация взаимодействия дрона и троса с объектами.
- 



## Благодарности и список литературы

Выражаем благодарность своему научному руководителю Чернову Валерию Андреевичу за ценные советы при планировании проекта, помощь при его выполнении и рекомендации по оформлению отчета.

### **Литература:**

The Dynamic Window Approach to Collision Avoidance - Dieter Fox, Wolfram Burgard, Sebastian Thrun

**Спасибо за внимание!**