

Инструкция по использованию

Требуемые компоненты

Для работы проекта требуется интерпретатор Python3 с установленным менеджером пакетов pip. Также, необходимо установить модули из файла requirements.txt командой:

```
pip install -r requirements.txt
```

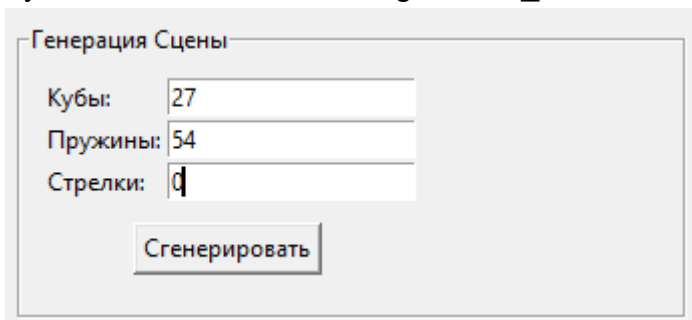
После того, как все модули установлены, можно запускать сцены и включать их работу при помощи графического интерфейса GUI.py. Запуск GUI.py выполняется командой

```
python3 GUI.py  
или  
python GUI.py
```

- Для запуска программы необходимо находиться в одной директории с файлом GUI.py
- Запуск программы можно осуществлять и с помощью IDE (например Spyder)

Создание сцены

Графический интерфейс позволяет генерировать сцены, которые будут содержать все необходимые для симуляции объекты, с уже прописанными в них параметрами плагина SimulationManager. Данная функция сделана для упрощения процесса создания сцен с большим количеством управляемых моделей. Сгенерированная сцена будет находиться в папке generate_scene - outPutFile.vrcp



Генерация Сцены

Кубы: 27

Пружины: 54

Стрелки: 0

Сгенерировать

Важно

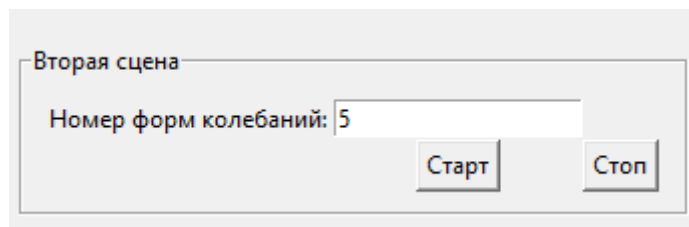
Каждая сгенерированная сцена соответствует коду, в котором задействуется определенное количество блоков, пружин и стрелок. То есть создание сцены для одной симуляции, а запуск программы другой симуляции не предусматривается.

В папке scenes находятся готовые сцены, которые рассчитаны для соответствующих программ, запускаемых через GUI.

Работа с сценами

Все сцены находятся в папке scenes. Первая сцена (1_scene) не предполагает какой-либо анимации. Ее можно запустить и просмотреть спутник и его упруго-массовую модель (УММ).

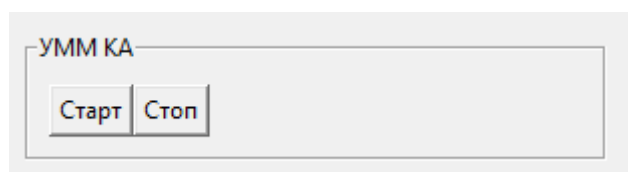
Вторая сцена (2_scene) содержит в себе УММ и возможен ее запуск для отображения форм колебаний.



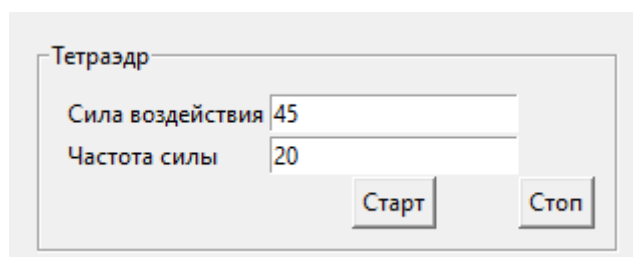
После загрузки сцены 2 необходимо выбрать форму колебаний (от 1 до 21) и нажать "Старт". УММ начнет совершать колебательные движения. Остановить работу симуляции необходимо нажатием кнопки "Стоп".

3 сцена(3_scene) состоит из трех подсцен.

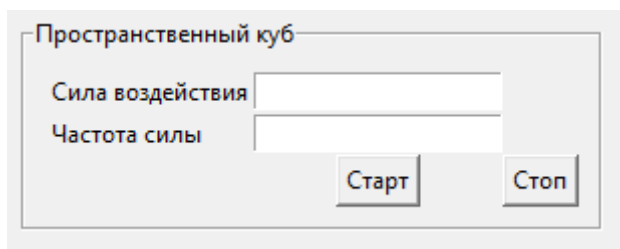
Первая подсцена (3_1scene) показывает колебания УММ рассчитанные численным методом Верле. Симуляцию можно запустить также нажатием кнопки "Старт" и необходимо остановить нажатием кнопки "Стоп".



Вторая подсцена (3_2scene) показывает колебания каркасного тетраэдра с точечными массами в узловых точках. Пользователь может задать силу и частоту воздействия, оказываемого на верхний блок.



Третья подсцена (3_3scene) показывает колебания каркасного куба, на центральный блок основания которого оказывается периодическое воздействия. Пользователь может настроить силу воздействия и его частоту.



Использование Python кода

С целью защиты интеллектуальной собственности исполняемые файлы python зашифрованы с помощью ruArmor, код программ предоставляется по согласованию с авторами проекта. Данный факт никак не мешает работе программы, лишь ограничивает попытки несанкционированного копирования и распространения.

Используемый математический аппарат

Расчет форм колебаний

Расчет происходит по следующему алгоритму:

1. Расчет матрицы жесткости и матрицы масс на основании связей между массами;
2. Нахождение собственных векторов-столбцов для каждой формы колебаний;
3. Запуск основного цикла отправки UDP пакетов на порт 6501, в которых содержатся координаты всех тел системы. Плагин SimulationManager осуществляет прием пакетов на порту 6501.

Численное интегрирование методом Верле

Расчет происходит в цикле по следующему алгоритму:

1. На каждом шаге основного цикла происходит вычисление сил, приложенных к каждому блоку.
2. На основании полученных сил находятся ускорение, скорость и следующая координата тела.
3. Отправка UDP пакетов на порт 6501, в которых содержатся координаты всех тел системы. Плагин SimulationManager осуществляет прием пакетов на порту 6501.