

Набор утилит чтения входных данных и печати выходных данных программы моделирования

- [View Source](#)

```
def read_input(filename: str) -> dict[str, float]:
```

- [View Source](#)

Считывает входные данные из csv-файла по указанному пути `filename` и возвращает словарь, соответствующий таблице исходных данных::

```
{имя параметра: значение}
```

Прим.: В качестве разделителя в csv-файле используется точка с запятой

Parameters

- **filename**: Путь до csv-файла с входными данными

Returns

Словарь, где ключ – имя входного параметра, а значение – его величина

```
def print_results(filename: str, results: list[dict]):
```

- [View Source](#)

Печатает результаты моделирования, собранные в списке словарей `results` в csv-файл по пути, указанному в `filename`

В качестве шапки таблицы используется множество ключей словаря. Для каждого элемента списка `results` набор ключей одинаков. Каждый элемент списка `results` – это отдельная строка csv-таблицы.

Прим.: В качестве разделителя в csv-файле используется точка с запятой

Parameters

- **filename**: Путь до печатаемого csv-файла с результатами моделирования
- **results**: Список словарей с результатами моделирования

```
def plot_vector(
    time: numpy.ndarray,
    vector_data,
    plot_name: str,
    titles: list[str],
    y_labels: list[str],
    subplot_order: tuple[int, int] = None,
    single_scale_y=True,
    add_plotting: Callable = None,
    args=None
):
```

- [View Source](#)

Выводит окно названия `plot_name` с графиками функций проекций векторов `vector_data` от времени `time`

Параметр `vector_data` принимает как матрицу 3xN, где N - кол-во моментов времени, для которых строится график проекций (каждый столбец матрицы соответствует значению вектора в момент времени), так и список таких матриц 3xN. В случае списка матриц, каждый элемент этого списка строится в окне графиков в отдельном столбце/строке.

Каждая проекция вектора строится на отдельном подграфике. Параметр `subplot_order` определяет количество строк и столбцов подграфиков в окне. По умолчанию `subplot_order` задан так, чтобы каждый элемент списка `vector_data` выводился тремя проекциями в одной строке. Заголовки подграфиков указываются в параметре `titles`. Подписи осей ординат подграфиков задаются параметром `y_labels`. Размерность, определяемая `subplot_order` должна совпадать с кол-вом элементов в `titles` и `y_labels`.

Режим единого масштаба осей для проекций одного вектора определяется булевым параметром `single_scale_y` (по умолчанию включен). Помимо единого масштаба, центр оси ординат задается так, чтобы графики проекций были отцентрованы по горизонтали.

Параметр `add_plotting` принимает функцию вида::

```
add_plotting(axes: Axes, i: int, j: int, *args)
```

которая выполняется после построения каждого подграфика проекции `axes` и используется для дополнительных операций отрисовки на подграфике `j`-ой проекции `i`-го вектора.

Parameters

- **time**: Массив моментов времени функций проекций векторов
- **vector_data**: Список матриц 3xN или одна матрица 3xN со значениями проекций вектора
- **plot_name**: Название графика виде строки, выводится как заголовок окна
- **titles**: Список строк заголовков подграфиков
- **y_labels**: Список строк подписей осей ординат подграфиков
- **subplot_order**: Количество строк и столбцов подграфиков в окне
- **single_scale_y**: (по умолчанию True) Флаг включения режима единого масштаба осей
- **add_plotting**: (по умолчанию None) Функция дополнительных построений на подграфике
- **args**: (по умолчанию None) Варьируемые аргументы функции дополнительных построений

`def plot_vector_with_ref_line(`

```
time: numpy.ndarray,  
vector_data,  
ref_t: numpy.ndarray,  
ref_y,  
plot_name: str,  
titles: list[str],  
y_labels: list[str]  
):
```

• [View Source](#)

Вариация функции::

```
plot_vector(time, vector_data, plot_name, titles, y_labels, subplot_order, single_scale_y, add_plotting, args)
```

которая дополнительно для каждого подграфика строит опорную прямую для явной демонстрации, что выводимая на подграфике функция не является линеной

Параметры `ref_t` и `ref_y` определяют график функции опорной прямой для каждого подграфика. Координаты опорной прямой для одной проекции определяются двумя значениями.

Parameters

- **time**: Массив моментов времени функций проекций векторов
- **vector_data**: Список матриц 3xN или одна матрица 3xN со значениями проекций вектора
- **ref_t**: Массив из двух элементов моментов времени функций опорных прямых
- **ref_y**: Список матриц 3x2 или матрица 3x2, где каждая строка – две точки, определяющие опорную прямую
- **plot_name**: Название графика виде строки, выводится как заголовок окна

- **titles:** Список строк заголовков подграфиков
- **y_labels:** Список строк подписей осей ординат подграфиков

```
def plot_bias(  
    axes: mpl_toolkits.mplot3d.axes3d.Axes3D,  
    point: numpy.ndarray,  
    labels: tuple[str, str, str] = ('$x_I$', '$y_I$', '$z_I$'),  
    mult: float = 1.0  
):
```

• [View Source](#)

Выводит графическое представление ортонормированного базиса на пространственном графике **axes** с началом координат в точке **point**, подписями на концах осей базиса **labels** и длиной осей **mult**

Parameters

- **axes:** Объект пространственного графика
- **point:** Координата начала координат базиса на пространственном графике
- **labels:** (по умолчанию задано) Подписи концов осей базиса
- **mult:** (по умолчанию 1) Длина осей базиса

```
def plot_vehicles_trajectory(  
    figure: matplotlib.figure.Figure,  
    t: numpy.ndarray,  
    vehicles_vectors: numpy.ndarray,  
    frame: int = None  
):
```

• [View Source](#)

Строит пространственный график траектории движения РН и КА в окне **figure** по массиву координат аппаратов **vehicles_vectors**

График траектории строится в виде двух подграфиков. Масштабы всех осей на каждом подграфике одинаковы для лучшей демонстрации физического положения аппаратов друг относительно друга.

На левом подграфике строится общий вид траекторий движения двух тел. На него выводятся траектории тел в виде пространственных кривых и точки, соответствующие текущему местоположению аппаратов на траекториях. Этот график имеет статичные пределы осей, не меняющиеся от местоположения РН и КА.

На правом подграфике строится увеличенный вид траектории. Он используется для демонстрации относительного расстояния между РН и КА и ориентации их векторов скоростей в СК. Дополнительно на этом графике приводится абсолютная скорость каждого аппарата в применяемой СК. Пределы осей на этом графике меняются согласно текущему местоположению РН и КА. Пределы подобраны таким образом, чтобы для любых координат аппаратов в один момент времени их точки оставались в пределах подграфика.

Параметры **t** и **frame** по большей части нужны для анимации графика. Массив моментов времени **t** используется для печати в заголовок на правом подграфике времени, соответствующего местоположению РН и КА. Параметр **frame** используется для перестроения текущего вида траектории, согласно порядковому номеру кадра в анимации.

Parameters

- **figure:** Объект окна для вывода графиков
- **t:** Массив из N моментов времени
- **vehicles_vectors:** Матрица 12xN, где первые 6 строк – радиус-вектор и вектор скорости РН, а вторые – то же самое для КА
- **frame:** (по умолчанию не используется) Порядковый номер кадра анимации

```
def show_anim(  
    figure: matplotlib.figure.Figure,  
    func_draw: Callable,  
    frame_freq: float,  
    t: numpy.ndarray,  
    *args  
):
```

• [View Source](#)

Настраивает анимацию и проигрыватель для нее в окне графиков `figure`, используя для отрисовки кадра анимации функцию `func_draw` вида::

```
func_draw(figure, t, *args, frame)
```

Функция создает проигрыватель анимации, который предоставляет возможность в произвольный момент времени анимации: - Запускать/останавливать анимацию - Переходить к ближайшему предыдущему/следующему кадру анимации - Переходить в начало конец анимации

Параметр `frame_freq` определяет количество кадров, выводящихся на секунду времени, используя в качестве шкалы массив моментов времени `t`. Таким образом, частота `frame_freq` не может быть выше частоты печати временной шкалы `t`. Параметр `args` определяет варьируемый список параметров функции отрисовки кадра.

Parameters

- **figure**: Объект окна для вывода графиков
- **func_draw**: Функция отрисовки кадра анимации
- **frame_freq**: Количество кадров, выводящихся на секунду временной шкалы `t`
- **t**: Массив моментов времени (временная шкала)
- **args**: Варьируемые аргументы функции отрисовки