

А.А. Шишканов, А.А. Квасников

Программный модуль для построения двумерных конфигураций типовых структур проводников и диэлектриков в системе TALGAT

Разработан программный модуль для построения двумерных конфигураций типовых структур, таких как микрополосковая линия передачи и коаксиальный кабель. Модуль обладает интуитивно понятным графическим интерфейсом пользователя, охранный механизм которого используется для защиты от ввода некорректных данных в программу.

Ключевые слова: Python, TALGAT, программный модуль, графический интерфейс пользователя, MOM2D.

TALGAT – система компьютерного моделирования электромагнитных задач, позволяющая проводить электростатический анализ сложных структур проводников и диэлектриков, а также вычислять их матрицы погонных параметров в модуле MOM2D [1]. Начиная пользователи системы сталкиваются с затруднениями при работе с этим модулем, поскольку он требует знаний синтаксиса внутреннего языка системы TALGAT_Script и его команд, а также корректного ввода данных для построения поперечных сечений анализируемых структур. Данная работа посвящена разработке программного модуля для облегченного построения двумерных конфигураций типовых структур проводников и диэлектриков, обладающего интуитивно понятным графическим интерфейсом пользователя (ГИП) и охранными механизмами полей ввода.

Инструменты разработки

В качестве языка программирования был выбран Python, поскольку этот язык лёгок в освоении, поддерживает большое число библиотек и модулей, с помощью которых можно создать ГИП, а также система TALGAT содержит его интерпретатор [2]. В качестве библиотек и модулей использовались: 1) Tkinter – библиотека, которая содержит компоненты графического интерфейса пользователя [3]; 2) Tk – модуль, с помощью которого можно изменять стиль оформления элементов интерфейса; 3) Pillow – библиотека, используемая для обработки графики в Python; 4) ImageTk – модуль, предназначенный для создания и изменения графических объектов.

Алгоритм работы программного модуля

На рис. 1 показан алгоритм работы разработанного программного модуля. На первом этапе выполняется выбор формы типовой структуры (микрополосковая линия передачи (МПЛП) или коаксиальная линия передачи). После этого пользователь задает параметры выбранной структуры. Для МПЛП задаются: толщина и ширина диэлектрика и проводника, число проводников, диэлектрическая проницаемость подложки. Для коаксиальной линии задаются: число проводников, расстояние между ними, радиусы проводников и диэлектриков. Если введенные параметры верны, то двумерная конфигурация отображается в соответствующем окне системы TALGAT, если нет, то выводится сообщение о соответствующей ошибке.



Рис. 1. Алгоритм работы программного модуля

Разработка программного модуля

Для создания ГИП использовались следующие виджеты и их методы:

Label – виджет, используемый для создания текста (подписей) на окне. ComboBox – виджет, отвечающий за создание выпадающего списка. Entry – виджет, позволяющий пользователю выполнить ввод данных в программу. Button – виджет, создающий элемент управления ГИП (кнопка). Tk – виджет для создания окна программы. Geometry – метод, служащий для задания геометрических размеров окна. Toplevel – виджет для создания окна верхнего уровня. Title – метод, с помощью которого окну можно дать название. Place – метод, используемый для размещения чего-либо на окне. Checkbutton – элемент, который представляет собой переключатель.

Основной функционал программного модуля состоит в автоматической генерации последовательности команд системы TALGAT для построения двумерной конфигурации выбранной структуры. Так, для создания МПЛП были использованы следующие команды модуля MOM2D системы TALGAT:

LINE(x, y, x1, y1) – создаёт линию, соединяющую точку с координатами x, y с точкой с координатами x1, y1; LINETO(x, y) – создает прямую линию между последней точкой и точкой с заданными ко-

ординатами x , y ; SET_SUBINTERVALS(n) – устанавливает необходимое число подынтервалов для разбиения границ проводников и диэлектриков; SET_INFINITE_GROUND(b) – устанавливает наличие бесконечной плоской земли в конфигурации ($b = 0$ – нет, $b = 1$ – есть); DRAW_CONFIGURATION(conf) – команда для отображения заданной конфигурации; SET_ER_PLUS(erp) – задает диэлектрическую проницаемость с положительной части оси (параметром является целое или дробное число); SET_ER_MINUS(erm) – задает диэлектрическую проницаемость с отрицательной частью оси; CONDUCTOR() – команда для создания проводников; DIELECTRIC() – команда для создания диэлектриков.

На рис. 2 представлен фрагмент кода функции, отвечающей за создание проводника прямоугольной формы. На рис. 3 представлен фрагмент кода функции, необходимой для создания диэлектрика прямоугольной формы.

```
if flag_value1.get() == 1:
    SET_INFINITE_GROUND(1)
    for i in range (count):
        CONDUCTOR()
        SET_SUBINTERVALS(8)
        SET_ER_PLUS(der)
        startx = dw-LR_diel_part-(cw*(i+1))-distance*i
        stopx = dw-LR_diel_part-(cw*i)-distance*i
        LINE(startx, dt, stopx, dt)
        SET_ER_PLUS(1)
        LINETO(stopx, dt+ct)
        LINETO(startx, dt+ct)
        LINETO(startx, dt)
```

Рис. 2. Фрагмент кода функции для создания проводника прямоугольной формы

```
for i in range (count-1):
    DIELECTRIC()
    SET_SUBINTERVALS(8)
    SET_ER_PLUS(1)
    SET_ER_MINUS(der)
    startx = dw-LR_diel_part-(cw*(i+1))-distance*i
    LINE(startx, dt, startx-distance, dt)
```

```
DIELECTRIC()
SET_SUBINTERVALS(8)
SET_ER_PLUS(1)
SET_ER_MINUS(der)
LINE(dw, 0., dw, dt)
LINETO(dw-LR_diel_part, dt)
LINE(LR_diel_part, dt, 0, dt)
LINETO(0,0)
```

Рис. 3. Фрагмент кода для создания диэлектрика прямоугольной формы

```
if flag1_value.get() == 1:
    for i in range (c):
        SET_INFINITE_GROUND(1)
        CONDUCTOR()
        SET_SUBINTERVALS(36)
        SET_ER_PLUS(er)
        CIRCLE(x, y, r)
        DIELECTRIC()
        SET_ER_MINUS(er)
        SET_ER_PLUS(1)
        CIRCLE(x, y, rpd)
        x += distance
    DRAW_CONFIGURATION(GET_CONFIGURATION_2D())
```

Рис. 4. Фрагмент кода функции для создания проводника и диэлектрика круглой формы

Для создания круглого проводника и диэлектрика использовалась команда CIRCLE(x , y , r). Её параметрами являются три числа, а именно координата центра x окружности по оси Ox , координата центра y окружности по оси Oy и радиус окружности r . На рис. 4 представлен фрагмент кода функций для создания проводника и диэлектрика круглой формы соответственно.

Функциональное тестирование

Выполнено тестирование функциональных возможностей программы. При запуске программного модуля пользователю доступно главное окно, в котором он может выбрать тип анализируемой структуры (рис. 5). После того, как пользователь выберет нужную ему структуру, выводится окно для задания деталей настроек ее параметров (рис. 6 и 7).

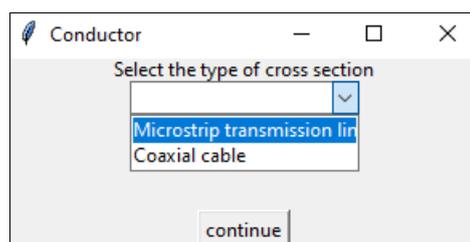


Рис. 5. Главное окно программного модуля

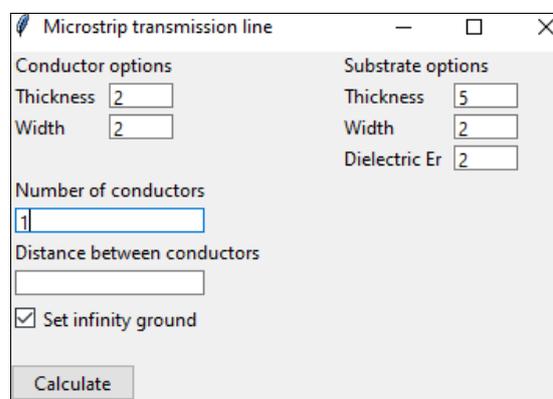


Рис. 6. Окно для задания параметров МПЛП

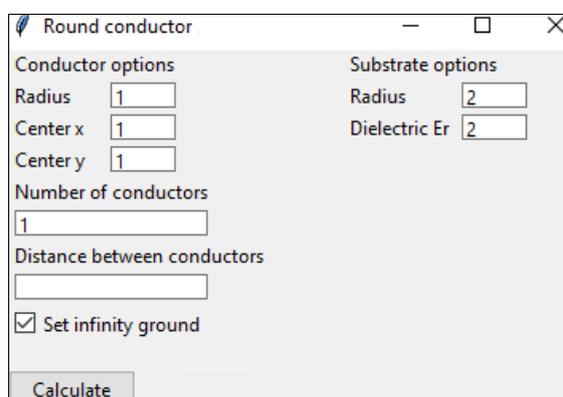


Рис. 7. Окно для задания параметров коаксиального кабеля

После ввода данных пользователем происходит проверка их корректности охранными механизмами. В случае если пользователь допустил ошибку ввода, то ему отобразится окно с информацией об ошибке

(рис. 8). В случае корректного ввода входных данных система TALGAT создает отдельное окно с двумерной конфигурацией пользовательской структуры.

Выполнен тестовый запуск программы для построения поперечных сечений типовых структур с параметрами, значения которых показаны на рис. 6 и 7. На рис. 9 приведены результаты работы программного модуля, подтверждающие корректность программной реализации.

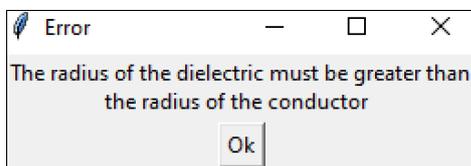


Рис. 8. Окно с информацией об ошибке

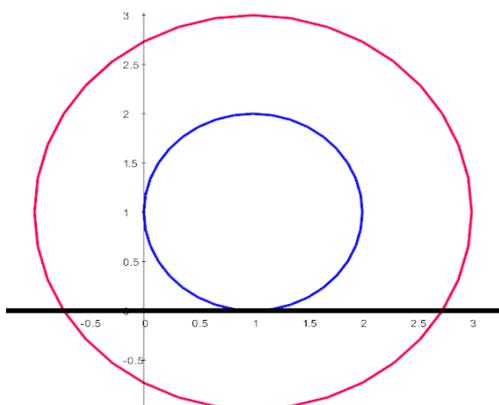
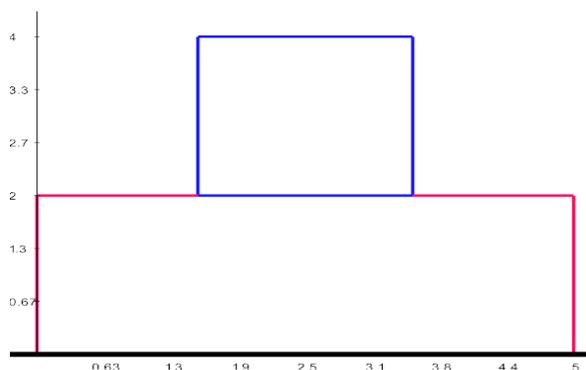


Рис. 10. Результат работы программного модуля

Заключение

Таким образом, в ходе работы был создан программный модуль для построения двумерных конфигураций типовых структур проводников и диэлектриков. Выполнено функциональное тестирование модуля, которое подтверждает корректность реализации. В дальнейшем планируется совершенствовать графическую оболочку, добавить поддержку других типовых структур, а также использовать модуль для генерации обучающих выборок при машинном обучении.

Литература

1. Квасников А.А. Программное обеспечение для проектирования оптимальной сети высоковольтного электропитания космического аппарата / А.А. Квасников, С.П. Куксенко, А.А. Иванов // Сборник трудов Всерос. науч.-техн. конф. «Проблемы разработки перспективных микро- и наноэлектронных систем». – М., 2020. – Т. 3. – С. 100–107.
2. Python [Электронный ресурс]. – Режим доступа: <https://metanit.com/python/tutorial/>, свободный (дата обращения: 15.08.2023).
3. Python Tkinter [Электронный ресурс]. – Режим доступа: <https://docs.python.org/3/library/tkinter.html>, свободный (дата обращения: 16.08.2023).

Шишканов Александр Андреевич

Студент, каф. телевидения и управления (ТУ)
Томского государственного ун-та
систем управления и радиоэлектроники (ТУСУР)
Эл. почта: Impulse1337228@mail.ru.

Квасников Алексей Андреевич

Аспирант, каф. ТУ ТУСУРа
Эл. почта: aleksejkvasnikov@tu.tusur.ru