

УДК 621.372.2.052.3.018.78

Е.В. Лежнин, А.О. Мелкозеров

Улучшение пользовательского интерфейса системы TALGAT

Показано улучшение подсистемы визуализации модуля MOM2D и графического клиента QtClient системы TALGAT. Рассмотрены вопросы отображения информации в модуле MOM2D, редактирования принципиальных схем, редактирования в модуле MOM2D, доработки графического клиента, поиска ошибок.

Ключевые слова: TALGAT, Qt, VTK, визуализация.

TALGAT – система компьютерного моделирования сложных структур проводников и диэлектриков [1]. Разработка пользовательского интерфейса в этой системе ведётся уже давно, но до сих пор существует потребность в увеличении степени интерактивности графического клиента: перед командой разработчиков стоит задача сделать взаимодействие пользователя с системой как можно более удобным, чтобы у пользователя была возможность работать без ввода текста, используя для этих целей только графические средства. Для реализации графического интерфейса используются инструментарии Qt и VTK, которые работают совместно.

Qt – кроссплатформенный инструментарий для разработки элементов графического интерфейса на языке программирования C++ [2]. Он обладает обширным набором функций для создания графического интерфейса, работы с несколькими потоками исполнения, создания интернет-приложений и используется для создания графического клиента QtClient системы TALGAT. VTK – кроссплатформенный инструментарий для разработки приложений с использованием 3D-графики [3].

В данной работе показаны результаты улучшения подсистемы визуализации модуля MOM2D и графического клиента QtClient системы TALGAT. Рассмотрены вопросы отображения информации в модуле MOM2D, редактирования принципиальных схем, редактирования в модуле MOM2D, доработки графического клиента, поиска ошибок.

Отображение информации в модуле MOM2D. Задача состояла в добавлении функций отображения информации о проводниках и диэлектриках в вычислительный модуль MOM2D. Для реализации условий задачи использовался инструментарий VTK, способный отображать как объёмный трехмерный, так и плоский двумерный рисунок. Сложность задачи заключалась в том, что библиотека VTK документирована недостаточно подробно, что сделало необходимым проверку нескольких вариантов реализации функциональности. Несмотря на то, что основная часть кода – рисование границ проводников и диэлектриков – уже была реализована, потребовалось изменить и её. Также написана реализация функции отображения осей координат по причине отсутствия подходящих стандартных функций отображения осей в инструментарии VTK.

Реализовано цветное отображение плотности заряда на границах проводников при вычислении ёмкостной матрицы, которое позволяет лучше понять специфику распределения заряда в конкретных структурах и, в особых случаях, проводить контроль корректности его вычисления. К модулю добавлена следующая функциональность: отображаются значения относительной диэлектрической проницаемости для интервалов, показываются номера проводников. По отдельной команде отображаются номера и заряды подынтервалов, показываются номера интервалов, количество подынтервалов, заземлены ли проводники. Подынтервалы отображаются определённым цветом в зависимости от заряда. Есть возможность показа осей координат, шкалы заряда, инструкции по использованию всех добавленных функций.

Редактор принципиальных схем. Создание схем с помощью текстового файла является трудной задачей для обычного пользователя. Поэтому необходим редактор схем для более универсального и простого создания принципиальных схем. В результате работы создан редактор, встроенный в графический клиент QtClient.

Для реализации редактирования требовалось сделать выбор между библиотеками Qt и VTK. Инструментарий Qt позволял сделать разработку графического интерфейса для взаимодействия с пользователем значительно проще; с VTK имелся опыт работы. Выбор был сделан в пользу Qt по причине лучшей документации; также были излишни возможности трёхмерной графики, пре-

доставляемые библиотекой VTK. Так как существует большое количество неофициальных ресурсов по Qt, к ним можно было бы обращаться в случае появления сложных вопросов (во время разработки функции отображения информации о подынтервалах в MOM2D в единственный список рассылки библиотеки VTK нам пришлось написать 2 раза, но ответы так и не были получены).

В результате начального этапа разработки достигнут следующий результат: появилась возможность загрузки существующей схемы с одной линией передачи и несколькими элементами, редактирование было реализовано полностью.

Диаграмма классов для представления схемы состоит из следующих классов. Класс CanvasScheme – главный класс представления принципиальной схемы (поле, элементы, проводники). Класс WindowScheme – основанный на QWidget (окно в Qt) и CanvasScheme класс, который служит для взаимодействия с пользователем (редактирование схемы, перемещение по схеме). Реализовано множество диалоговых окон (диалоги параметров для обычных элементов, линий передачи, проводников), а также обработка событий мыши и клавиатуры. Класс Router – основан на CanvasScheme. Приложение использует объект типа Router для хранения принципиальной схемы во время размещения элементов на правильные позиции при загрузке из файла (трассировке), созданная схема затем копируется в объект WindowScheme (уже видимый для пользователя).

На поле с принципиальной схемой располагаются элементы классов, производных от класса Element (рис. 1). Все элементы SchElement (scheme element) объединяет то, что они имеют входы и могут быть соединены проводниками. При редактировании они имеют одинаковый приоритет: не могут размещаться на места, где есть проводники либо другие элементы. Nodes – целый проводник (либо группа соединяющихся узлов), имеющая один порядковый номер в представлении схемы. Node – линия или две линии, которые можно описать двумя точками: в данном редакторе линии всегда параллельны осям. Поэтому если линия, которую образуют точки узла, параллельна осям, то структура описывает одну линию, иначе – две линии. Например: точки (1; 2), (3; 4) образуют линии узла $\{(1; 2), (1; 4)\}$ и $\{(1; 4), (3; 4)\}$. TransmissionLineElement – линия передачи, отличается от обычных элементов только тем, что, в зависимости от числа проводников, может занимать больше места и иметь больше входов. Ground – заземление, относится к отдельным элементам в структуре проекта WindowScheme. Всем проводникам, подсоединённым к заземлению, присваивается нулевой номер.

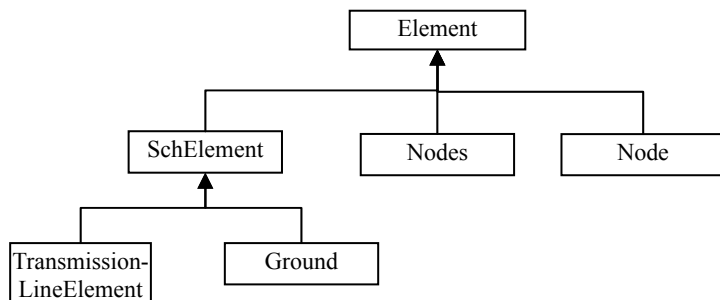


Рис. 1. Классы элементов схемы

В ходе решения задачи возникли следующие трудности:

1. Мигание экрана при рисовании. Рисунок должен выводиться на дисплей достаточно быстро, чтобы не мешать работе программы, при этом не должно происходить мигание экрана. Во время рисования изображение рисуется с помощью промежуточного объекта QPainter, связанного не с дисплеем, а с изображением (двойная буферизация), в общей сложности для этой цели используются 3 изображения: ячейка сетки, рисуемая в конструкторе; сетка, перерисовываемая каждый раз при изменении размера поля; элементы схемы, перерисовываемые каждый раз, кроме случаев, когда добавляется только один элемент.

2. Реализация объединения проводников. Ячейки на поле необходимы для удаления проводников. Если эти ячейки заняты проводниками, то они всегда хранят в себе номера тех проводников, которые были помещены на это место с самого начала. После объединения двух проводников один проводник становится главным, другой проводник (второстепенный) при этом не удаляется из памяти, но все узлы из него переносятся в главный проводник. После этого второстепенный проводник хранит в себе номер главного. Удаление проводника происходит после выбора ячейки, в которую записан номер главного или второстепенного проводника. Так что удаляться из памяти должны только узлы главного проводника, а при удалении второстепенного удалится только тот проводник, на который ссылается этот второстепенный проводник.

3. Рисование проводников, которые должны обходить элементы. Эта функция добавлена исключительно для удобства пользователя. Например, если пользователь решит провести проводник сразу через несколько элементов на одной прямой, все эти элементы соединятся. Каждая ячейка содержит

подробную информацию о себе: есть ли в ней элемент, проводник это или другой элемент, идентификатор элемента на этой ячейке, ориентация элемента, проводника (отдельно) и т.д. Такая подробная информация нужна, чтобы определить, должен ли проводник прерываться в этой ячейке, подсоединяться к выходу элемента либо объединяться с другим проводником.

4. Реализация нулевых проводников, заземления. Любой проводник может стать нулевым, когда он подсоединится к заземлению, либо будет объединён с заземлённым проводником, либо если заземление подсоединится к этому проводнику. Для того чтобы отметить, что проводник подсоединён к заземлению, используется флажок, при этом порядковый номер этого элемента никак не указывает на то, что проводник заземлён (он никогда не равен 0).

Алгоритм размещения элементов, записанных без указания координат, пока не совершенен, но всё же способен расположить на холсте схему, состоящую из одной линии передачи и нескольких обычных элементов. Алгоритм заключается в следующем: на первом этапе размещается линия передачи, все элементы, которые можно подсоединить к линии передачи напрямую, располагаются слева и справа от неё. Получается схема, к которой можно подсоединять слева и справа те элементы, для которых это возможно сделать напрямую. На втором этапе алгоритма оставшиеся элементы цепи размещаются на свободном пространстве сверху и снизу получившейся схемы. Теперь элементы размещены на схеме, но ещё не соединены между собой; чтобы их соединить, используется волновой алгоритм – алгоритм кратчайшего соединения точек с обходом препятствий [4]. Проводники могут пересекаться, пересечения проводников обозначаются дугой. Пример результата работы алгоритма показан на рис. 2.

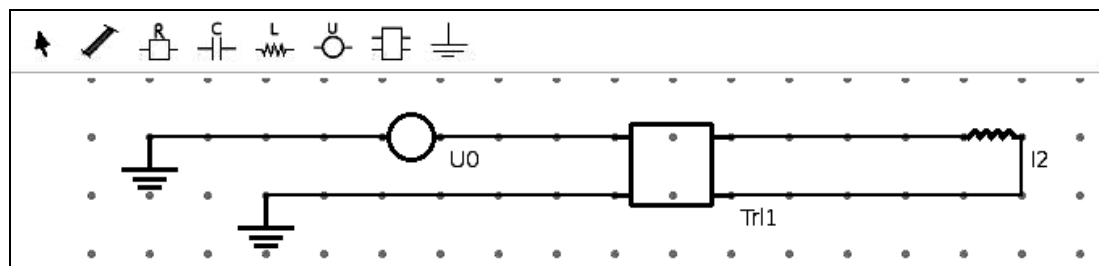


Рис. 2. Результат работы алгоритма размещения элементов на схеме

Добавление возможности редактирования в MOM2D. Задача состояла в добавлении в модуль MOM2D возможности рисовать новые интервалы и устанавливать для них значения диэлектрической проницаемости. Для реализации была выбрана библиотека VTK, так как отображение уже реализовано с её помощью, а для работы с Qt необходимо переписывать существующий код. На начальном этапе реализации возникали проблемы, связанные с редактированием (VTK – библиотека, изначально не предназначенная для редактирования рисуемых объектов), однако удалось быстро найти решение благодаря официальному списку рассылки данной библиотеки. На данном этапе можно добавлять мышью любые границы проводников и диэлектриков.

Доработка клиента. В рамках данной задачи требовалось добавить новые возможности к графическому клиенту QtClient – новому клиенту системы TALGAT на Qt. Были реализованы следующие функции:

1. Автодополнение – реализация дополнения написанного слова до ключевого при нажатии определенной комбинации клавиш. Требовалось подробно ознакомиться с кодом QtClient, дополнить текстовый редактор клиента, а также консольный редактор, код для которых должен быть общим.

2. Контекстная справка – информация о команде или ключевом слове во всплывающей подсказке при наведении указателя мыши.

3. Комментирование – пункт меню, при выборе которого комментируется или раскомментируется выбранный текст. Нужно было расширить возможности клиента и добавить доступную пользователю функцию, форматирующую выделенный текст как комментарий или убирающую все знаки комментариев (//) в выделенном тексте, если он уже закомментирован. Подобная функция доступна во многих редакторах исходного кода, таких как редакторы кода сред разработки ПО Microsoft Visual Studio и Qt Creator. Сложность задачи состояла в том, что текст нужно было форматировать с помощью средств инструментария Qt для окна редактирования, в котором этот текст находится, поэтому сделать это можно было ограниченным количеством способов.

Поиск ошибок. Выполнена проверка правильности реализации источников напряжения – проанализирован код, написанный на MATLAB, найдены несоответствия между исходными кодами на MATLAB и TALGAT. Полоса прогресса – проанализирована работа полосы прогресса при выполнении скрипта TALGAT, в том числе схема работы потоков во время работы полосы прогресса.

Заключение. Таким образом, в результате выполнения поставленных задач по улучшению пользовательского интерфейса системы TALGAT новыми функциями были дополнены графический клиент QtClient и подсистема визуализации вычислительного модуля MOM2D. Новые возможности увеличили степень интерактивности графических средств системы TALGAT и упростили работу пользователя при выполнении вычислений с помощью клиента QtClient и во время анализа результатов моделирования в модуле MOM2D.

Работа выполнена в порядке реализации постановления №218 Правительства РФ от 09.04.2010 г. «О мерах государственной поддержки развития кооперации российских высших учебных заведений и организаций, реализующих комплексные проекты по созданию высокотехнологичного производства», и договора № 13.G25.31.0017 от 07.09.2010 между ОАО «ИСС» им. акад. М. Ф. Решетнева» и Минобрнауки РФ.

Литература

1. Официальный сайт системы TALGAT [Электронный ресурс]. – Режим доступа: <http://talgat.org/>, свободный (дата обращения: 29.08.2011).
2. Сайт инструментария Qt [Электронный ресурс]. – Режим доступа: <http://qt.nokia.com/>, свободный (дата обращения: 29.08.2011).
3. VTK, библиотека для работы с 3D-графикой [Электронный ресурс]. – Режим доступа: <http://en.wikipedia.org/wiki/VTK/>, свободный (дата обращения: 29.08.2011).
4. Описание волнового алгоритма [Электронный ресурс]. – Режим доступа: <http://algotlist.manual.ru/games/wavealg.php>, свободный (дата обращения: 05.09.2011).

Лежнин Евгений Владимирович

Студент, лаборант-исследователь каф. телевидения и управления ТУСУРа
Тел.: (382-2) 41-34-39
Эл. почта: z_lezhnin@mail2000.ru

Мелкозеров Александр Олегович

Аспирант, мл. науч. сотрудник каф. телевидения и управления ТУСУРа
Тел.: (382-2) 41-34-39
Эл. почта: alexander.melkozerov@gmail.com

Lezhnin E.V., Melkozerov A.O.

Enhancing of the user interface of the TALGAT system

Improvements in the visualization module MOM2D and graphical client QtClient of the TALGAT system are presented. Visualization and editing of information in the MOM2D module, implementation of the circuit diagram editing and enhancements of the graphical client are described.

Keywords: TALGAT, Qt, VTK, visualization.