

УДК 681.31

И. В. Рудakov

ПРОГРАММНЫЙ КОМПЛЕКС ИЕРАРХИЧЕСКОГО ИССЛЕДОВАНИЯ СЛОЖНЫХ ДИСКРЕТНЫХ СТРУКТУР

Рассмотрена проблема создания программного обеспечения САПР технических систем на базе блочно-иерархического исследования сложных дискретных структур. Описан разработанный язык моделирования, ориентированный на транзакты, предназначенный для иерархического проектирования структур с очередями. Подробно приведен механизм управления, включающий в себя методику хранения и индексирования событий, а также сами алгоритмы управления. Описан синтаксис языка моделирования и особенности использования разработанного программного обеспечения.

E-mail: irudakov@yandex.ru

Ключевые слова: анализ сложных дискретных систем, язык моделирования, транзакты.

В настоящее время задача создания программного обеспечения САПР анализа сложных дискретных устройств является актуальной и находит широкое применение [1, 2] (робототехника, устройства управления техническими системами, информационные системы, микропроцессорная техника и т.д.). Однако в силу исключительного разнообразия встречающихся на практике задач и недостаточной изученности их математического описания арсенал формализации и моделирования непрерывно пополняется. Поэтому порядок структуризации, разбиения на подсистемы и элементы объектов и отбор математических схем для описания элементов нельзя считать окончательно сложившимся.

При анализе и проектировании структур сложных систем с очередями используется блочно-иерархический метод [1], который предусматривает расчленение процесса проектирования на ряд последовательных уровней и сведение задачи большей размерности к совокупности задач значительно меньшей размерности.

Одной из задач при разработке структур сложных систем является задача анализа и контроля правильности функционирования системы на ранних этапах ее разработки, в результате чего определяется состав функциональных блоков и их взаимодействие. На последующих этапах анализируются отдельные блоки. Предлагаемый подход позволяет анализировать функционирование сложной структуры, используя методы макро моделирования и декомпозиции на базе языка моделирования, ориентированного на транзакты.

Для реализации правильности функционирования сложных систем в качестве элементов моделирования предлагается использовать макро модели функциональных блоков сложной системы на основе формализации их функционирования в виде логической сети [3].

Метод анализа сложных структур только на базе макромоделей функциональных блоков недостаточен, так как не позволяет учитывать такие характеристики системы, как например, временные параметры элементов, входящих в макромоделю, а следовательно, выполнять надежную верификацию проекта. В целях уточнения характеристик сложной системы используется принцип “увеличительного стекла”, когда различные участки структуры системы анализируются с разной степенью детализации. В отличие от традиционного подхода, при котором переход с одного иерархического уровня на другой выполняется разработчиком системы, предлагаемый метод декомпозиции, реализованный в виде программной подсистемы, позволяет выполнить автоматический переход от макромоделей функционального блока к моделям элементов структуры при выполнении смешанного моделирования сложных систем.

Практическая реализация иерархического исследования сложных дискретных структур опирается на технологию объектно-ориентированного программирования (ООП) [4, 5]. В качестве инструментального средства анализа и верификации структуры разработан язык моделирования на базе Visual C++, ориентированный на транзакты. Предлагаемый программный комплекс основан на методологии общецелевой системы моделирования GPSS и является расширением ее в плане реализации исследования сложных дискретных структур на уровне функциональных блоков (макромоделей). Интерфейсная часть комплекса включает возможность визуального графического построения и отображения моделируемой системы, что облегчает создание модели и ее анализ.

Программное обеспечение иерархического проектирования структур с очередями функционирует в рамках ОС Windows XP и более.

Для формализации процесса функционирования сложного дискретного устройства в виде логической сети в графическом виде введена следующая классификация блоков:

- блок GENERATE — создание транзактов;
- блок TERMINATE — уничтожение транзактов;
- блок DEVICE — обслуживание транзактов;
- блок QUEUE — организация очередей транзактов;
- блок BRANCH — ветвление маршрута движения транзактов;
- блок COLLECT — объединение маршрутов движения транзактов.

Сеть моделируется при использовании механизма планирования событий (событийный подход). Считается, что изменения состояния системы происходят в некоторые дискретные моменты времени – моменты наступления событий. Изменение состояния происходит мгновенно. Состояние системы характеризуется совокупностью переменных. Начальное состояние устанавливается путем задания значений этих переменных. События, возникающие в модели, связаны с каждым блоком отдельно. В некоторых случаях возникновение события характеризуется интервалом времени, через которое оно должно произойти, и планируется заранее (в моменты определения интервала времени). К такому виду событий относятся события создания (блок GENERATE) и обслуживания (блок DEVICE) транзактов. В остальных случаях явно задать время возникновения событий невозможно, поэтому их наступление описывается с помощью условий (выход транзакта из блока QUEUE). Наличие условных событий приводит к тому, что помимо использования техники планирования событий необходимо использовать технику

сканирования активностей. Учитывая, что условные события связаны с состоянием системы (блок свободен/занят и т.д.), проверять условия срабатывания нужно только в моменты изменения состояния.

Для управления процессом моделирования с учетом наличия двух разнородных групп событий разработан механизм управления, включающий в себя методику хранения и индексирования событий, а также сами алгоритмы управления.

Управляющие алгоритмы опираются на специальные структуры: списки будущих и условных событий. Данные списки хранят информацию о планируемых событиях в системе. Управляющие алгоритмы основываются на добавлении и выталкивании событий из этих списков.

Список будущих событий представляет собой набор списков событий, каждый элемент которого связан с конкретным временем моделирования. Это время характеризует моменты времени, в которые должны произойти события, описанные в связанном списке.

Список будущих событий постоянно поддерживается в отсортированном по возрастанию времени виде. Первый связанный список описывает ближайшие по времени события. В моменты протяжки модельного времени он связан с моментом времени, на которое в следующий момент переведется модельное. В остальных случаях он связан с событиями, которые необходимо исполнить в текущий момент времени.

Список условных событий — это набор пар вида событие и условие его срабатывания. Условие характеризует моменты наступления события. Как только выполняется условие, возникает соответствующее событие.

Для оптимизации доступа к этому списку вводится дополнительный массив. С каждым элементом этого массива связан определенный параметр моделируемой системы, от которого может зависеть условие возникновения события. С каждым таким элементом связан список индексов. Элементы такого списка являются ссылками на события, на которые может повлиять изменение связанного параметра.

Массив параметров формируется перед началом моделирования, во время моделирования этот массив не меняется. Изменения происходят в связанных индексных списках при добавлении и выборке событий из списка условных событий. В массив параметров первым элементом добавляется индексный список событий, условия срабатывания которых не удалось разбить по зависящим параметрам. Этот список хранит ссылки на события, условия которых проверяются каждый раз.

Для быстрого поиска индексов, которые нужно удалить при выборке событий из списка условных событий, к паре событие и условие добавляется третий элемент, который ссылается на элементы индексных списков.

Таким образом, планируемые события (привязанные к времени срабатывания) группируются по времени возникновения и упорядочиваются во времени в специальном списке будущих событий. Модельное время изменяется согласно обработке событий из этого списка. Событие на обработку берется из начала списка. После обработки каждого такого события, состояние системы может измениться, и поэтому необходимо проверить условия возникновения непланируемых событий. При срабатывании условия выполняется соответствующая обработка. Алгоритм функционирования заключается в последовательной обработке событий с попеременным сочетанием обработки событий из списка будущих событий и проверки условий срабатывания непланируемых событий.

При разработке программного комплекса выделен ряд основных объектов (блок, транзакт, переменная, значение, выражение оператор, таблица, библиотека), которые в дальнейшем были взяты за основу при построении структуры классов.

Модульная структура программного комплекса состоит из трех частей:

— статическая библиотека (Values.lib), которая содержит описание функциональных блоков модели;

— системная динамическая библиотека (SysLib.dll), которая содержит описание базовых функций для работы с блоками (к таким функциям относятся функции сравнения значений, базовые арифметические функции (сложение, вычитание, умножение, деление), функции преобразования и функции генерации случайных величин);

— основной модуль (NetMassServ.exe), который несет всю функциональную нагрузку, содержит описание объектов сети, алгоритмов функционирования этих объектов, поведение комплекса при редактировании и моделировании сети.

Данный программный комплекс имеет возможность подключения внешних динамических библиотек.

Созданное программное обеспечение иерархического метода исследования дискретных структур позволяет эффективно выполнять анализ правильности функционирования структур данного класса и практически применяется при моделировании систем управления пожаробезопасности в транспортных туннелях (третье транспортное кольцо).

СПИСОК ЛИТЕРАТУРЫ

1. Н о р е н к о в И. П. Основы автоматизированного проектирования: Учебник для вузов. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2000. – 360 с.
2. В о р о т н и к о в С. А. Информационные устройства робототехнических систем: Учеб. пособие. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2005. – 384 с.
3. Н о в о с е л ь ц е в В. Н. Математическое моделирование в век компьютеров. http://lgkb.kazan.ru/00_1_4/
4. В а n k s J., С h a i r P. Simulation in the future // Proc. of the 2000 Winter Simulation Conference.
5. S o l i m a n H. M. On the selection of the state saving strategy in time warp parallel simulations // Transactions. – 1999. – Vol. 16. No. 1. – P. 32–36.

Статья поступила в редакцию 25.04.2011

Игорь Владимирович Рудаков — канд. техн. наук, доцент кафедры “Программное обеспечение ЭВМ и информационные технологии” МГТУ им. Н.Э. Баумана. Автор ряда научных работ в области имитационного моделирования.

I.V. Rudakov — Ph. D. (Eng.), assoc. professor of “Computer Software and Information Technologies” department of the Bauman Moscow State Technical University. Author of a number of publications in the field of imitating simulation.