

**МЕТОДИКА ИСПОЛЬЗОВАНИЯ СИСТЕМЫ ИМИТАЦИОННОГО
МОДЕЛИРОВАНИЯ РВС РВ ДИАНА, ОСНОВАННОЙ НА СТАНДАРТЕ НЛА**

**Д.Ю. Волканов, В.А. Антоненко, В.А. Герасев, А.Б. Глонина, В.А. Захаров,
Д.А. Зорин, И.В. Коннов, В.Н. Пашков, В.В. Подымов, О.К. Савенков,
Р.Л. Смелянский, Е.В. Чемерицкий (Москва)**

Распределенной вычислительной системой реального времени (РВС РВ) называют такую вычислительную систему, узлы которой распределены в пространстве, а правильность работы зависит от результатов вычислений и от продолжительности вычислений [1]. К числу РВС РВ относятся автоматизированные системы управления промышленным производством, энергетикой, транспортом, вооружением, банковскими операциями, медицинской аппаратурой, бытовая техника, электроника. Поскольку ошибки в работе таких систем могут привести к большому ущербу, к РВС РВ предъявляются повышенные требования правильности и безотказности их функционирования.

Основными методами проверки корректности работы РВС РВ являются имитационное моделирование и тестирование. На первом этапе разработки системы строится имитационная модель, которая должна позволять исследовать и оценивать наиболее важные параметры системы, такие как директивные сроки выполнения заданий, допустимые задержки, протоколы обмена данными между элементами системы. В зависимости от степени готовности аппаратной части РВС РВ, здесь можно использовать либо программную модель всей проектируемой системы, либо комплекс, состоящий из натуральных образцов аппаратуры и программных моделей приборов, собранных в единый стенд и сопряженных через аппаратные каналы бортовых интерфейсов (полунатурное моделирование). Также можно варьировать уровень детальности моделирования – от так называемых «интервальных» моделей, которые отрабатывают заданные циклограммы обменов, не выполняя вычисление передаваемых прибором данных, до полных функциональных моделей, эквивалентных реальным приборам по составу и значениям выдаваемых в бортовые каналы данных и включающих в свой состав реальное ПО РВС РВ. Затем имитационная модель заменяется собственно аппаратной частью РВС РВ и программами, работающими на этой аппаратуре, и вся система подвергается всестороннему тестированию.

Для организации описанного выше процесса проектирования инженеру-разработчику необходимо иметь в своем распоряжении систему имитационного моделирования. Данная работа посвящена методике проектирования РВС РВ и описанию системы моделирования, поддерживающей эту методику.

2. Требования к системе моделирования РВС РВ

Вначале опишем требования к системе моделирования (СМ) РВС РВ. Эти требования были сформированы на основе результатов исследований, приведенных в [2], и личном опыте авторов данной работы.

Открытость системы. При создании СМ целесообразно как можно более широко использовать разработанные в открытых проектах программные компоненты. Открытость исходных кодов и их использование позволяют повысить прозрачность функционирования СМ, а также дают большие возможности по поддержке и развитию СМ.

Организация выполнения набора моделей. Разработанную модель необходимо запустить на выполнение. В имитационном моделировании выделяют распределенное

имитационное моделирование (РИМ). РИМ имеет целый ряд достоинств над последовательным моделированием [3].

Возможность совместного использования средств моделирования и верификации. При проектировании РВС РВ важной задачей является проверка соответствия поведения системы ее спецификации. Поэтому формат описания модели должен позволять проводить не только имитационное моделирование проектируемой РВС РВ, но и ее формальную верификацию.

Возможность создания имитационных моделей различной степени сложности. Разработка РВС РВ с использованием имитационного моделирования проводится поэтапно. Вначале каждый компонент РВС РВ представляется простейшей программной моделью. Затем для более точного отражения поведение реальных устройств модели усложняются. На последних этапах создания комплекса программные модели замещаются разработанными прототипами устройств, вплоть до полного исключения программных компонент.

Сопряжение с аппаратурой и в реальном времени по натурным каналам передачи данных. СМ должна обеспечивать возможность подключения устройств с помощью подходящих натурных каналов и поддерживать скорость выполнения программных моделей на уровне, достаточном для соблюдения спецификаций используемых протоколов передачи данных. Точность привязки модельного времени к физическому должна измеряться в десятках микросекунд. Для корректного построения имитационных моделей с такой точностью необходимо разработать среду выполнения с как можно меньшим временем отклика.

Регистрация и обработка результатов моделирования, в т.ч. взаимодействие с аппаратными мониторами каналов передачи данных. Для проверки корректности функционирования РВС РВ необходимо регистрировать информацию обо всех событиях, происходящих при моделировании, в т.ч. об обменах по натурным каналам, и сохранять ее в форме, удобной для последующей обработки.

Интероперабельность системы моделирования со сторонними системами. Отдельные устройства в составе РВС РВ могут быть созданы конкурирующими организациями, отказывающимися предоставить разработчикам программную модель своего компонента из опасения утечки их технологий в процессе моделирования. Поэтому СМ должна поддерживать возможность подключения сторонних моделей.

Синхронизация времени в системе моделирования. При проведении РИМ необходимо обеспечить корректный глобальный порядок событий, происходящих во всех компонентах модели РВС РВ.

Анализ существующих СМ [2] показал, что перечисленным требованиям полностью не удовлетворяет ни одно из существующих средств, включая СМ ДИАНА [4] и стенд ПНМ [2], разработанные при участии некоторых из авторов данной работы. Начиная с 2010 года, в лаборатории вычислительных комплексов ф-та ВМК МГУ ведется разработка СМ нового поколения ДИАНА, совместимой с международными стандартами имитационного моделирования и современными методами анализа РВС РВ. В следующем разделе приводится методика работы со СМ ДИАНА.

3. Методика использования системы моделирования

При разработке моделей РВС РВ, предлагается придерживаться последовательности действий, приведенной ниже. Основная идея, заложенная в СМ ДИАНА, это использование единого описания модели для логического и количественного анализа РВС РВ. К задачам логического анализа РВС РВ относятся в задачи обнаружения

deadlockов при работе нескольких параллельных процессов, а к задачам количественного анализа относятся задачи оценки производительности проектируемой системы.

На первом этапе разработки модели проводится ее описание в виде диаграммы состояний UML. Для этого необходимо определить цель моделирования и построить диаграмму состояний UML так, чтобы конструируемая модель была корректна и адекватна целям моделирования. В диаграмму могут быть включены фрагменты кода на C++.

На следующем этапе проводится верификация, т.е. проверка того, что построенная модель удовлетворяет некоторым заданным свойствам. Этот этап необходим при исследовании сложных моделей РВС РВ, корректность которых должна быть строго доказана. Для проведения верификации необходимо преобразовать UML-диаграмму в сеть временных автоматов UPPAAL [5] с помощью разработанного нами транслятора. Транслятор может обнаруживать несколько видов ошибок в диаграммах, оценивать наихудшее время выполнения кода с помощью средства Metamoc [6], уменьшать количество получаемых временных автоматов. В случае использования средства Metamoc необходимо также задать характеристики целевой архитектуры. Если транслятор не обнаружил ошибок, то можно приступать к верификации модели с помощью средства UPPAAL. Для этого проверяемое свойство поведения диаграмм UML необходимо задать в виде формулы темпоральной логики TCTL. Верификатор проверяет выполнимость заданной формулы в дереве трасс вычислений сети временных автоматов, соответствующих диаграммам состояний UML. Если свойство не выполнено, то верификатор строит контрпример – одну из трасс, на которых проверяемое свойство нарушается. Этот контрпример можно конвертировать в трассу переходов UML с конкретными значениями параметров и таймеров.

Затем модель в несколько этапов транслируется из UML-представления в код на C++. После того как код федератов получен, запускается имитационный эксперимент в среде CERTI[7]. Результатом моделирования является трасса эксперимента, содержащая информацию о зарегистрированных событиях. Далее с помощью средства анализа и визуализации трасс пользователь может получить информацию о событиях модели и информационных обменах между ее компонентами, осуществить поиск события, навигацию по трассе, масштабирование трассы. В зависимости от результатов анализа трассы можно вернуться на один из предыдущих этапов, изменить модель и повторить эксперимент.

При проведении имитационного эксперимента в среде CERTI возможна постепенная замена моделей приборов самими приборами. В первую очередь, это актуально при моделировании работы приборов, передающих информацию по бортовому каналу передачи данных. В этом случае модели приборов обмениваются данными по аппаратному каналу передачи данных.

Архитектура СМ зависит от требований, приведенных в разделе 2, а также от методики использования СМ, описанной в данном разделе. Описание архитектуры СМ ДИАНА приведено в следующем разделе.

4. Архитектура системы моделирования ДИАНА

Архитектура разработанной системы моделирования приведена на рис. 1. Белым цветом обозначены открытые средства, используемые в проекте без модификаций, светлым – средства, модифицированные для разрабатываемой СМ, темным – средства, полностью разработанные и реализованные в рамках данной работы.

В состав среды моделирования входят следующие подсистемы:

Редактор диаграмм состояний UML; в качестве редактора используется средство ArgoUML.

Транслятор диаграмм UML в исполняемые модели, совместимые со стандартом HLA. Транслятор состоит из двух подсистем: транслятора UML в формат SCXML и генератора кода федератов. Промежуточный формат SCXML может быть использован для создания простых моделей и особенно удобен для автоматической их генерации. Также SCXML-диаграмму можно преобразовать в сеть временных автоматов UPPAAL и провести ее верификацию.



Рис. 1. Архитектура среды моделирования ДИАНА

Среда выполнения моделей на основе системы CERTI, дополненная средствами для поддержки моделирования PBC PB, интеграции с натурными каналами и интеграции с библиотекой времени компиляции Proto-X, кодирующей данные с использованием встроенных типов языка C++.

Средство трассировки, представляющее собой специальный федерат HLA; оно позволяет, используя распространенный формат OTF[8], конструировать трассу событий, возникающих в ходе моделирования.

Средство анализа и визуализации трасс, созданное на основе средства vis3, входящего в Стенд ПНМ, и обеспечивающее интеграцию с форматом описания трасс OTF.

Средство верификации UPPAAL, предназначенное для проверки свойства поведения систем реального времени, представленных в виде сетей временных автоматов. Оно снабжено транслятором, преобразующим диаграммы состояний UML в сети временных автоматов с учетом оценок наихудшего времени выполнения программы, полученных с помощью средства Metamos.

Интегрированная среда разработки, объединяющая все указанные выше средства и запускающая процедуру моделирование.

Одним из полезных свойств разработанной среды моделирования является возможность ее интеграции со средствами планирования расписаний и синтеза архитектур для оценки времени выполнения работ. Разработчику средств планирования (синтеза) предоставляется скрипт, который по расписанию заданного формата строит имитационную модель, проводит эксперимент, анализирует результаты и генерирует

файл, содержащий время выполнения всех работ расписания. Таким образом, средство планирования должно лишь в определенные моменты генерировать файл с расписанием и вызывать скрипт. Примером может служить средство решения задачи выбора механизмов обеспечения отказоустойчивости РВС РВ [9], позволяющее строить РВС РВ высокой надежности при ограничениях на стоимость и время выполнения программ.

5. Выводы

В данной работе описана архитектура и методика использования системы имитационного моделирования для решения целого комплекса задач, возникающих при проектировании РВС РВ. Система удовлетворяет приведенным в разделе 2 требованиям и готова для эксплуатации. Для ее дальнейшего совершенствования мы планируем решить следующие задачи:

Оптимизировать алгоритма трансляции диаграмм состояний UML в сети временных автоматов для преодоления эффекта «комбинаторного взрыва» в пространстве состояний автоматов.

Разработать гибридный консервативно-оптимистический алгоритм синхронизации времени в среде выполнения моделей.

Автоматизировать обработку результатов моделирования.

Литература

1. **Stankovic J.A.** Real-time Computing. // Byte Magazine – 1992. – V. 17, № 8. – P. 155–160.
2. **Балашов В.В., Бахмуrow А.Г., Волканов Д.Ю., Смелянский Р.Л., Чистолинов М.В., Ющенко Н.В.** Стенд полунатурного моделирования для разработки встроенных вычислительных систем // Методы и средства обработки информации: Третья всероссийская науч. конф. Труды конференции. – М.: Издательский отдел факультета ВМиК МГУ имени М.В. Ломоносова; МАКС Пресс, 2009. – С.16–25.
3. **Fujimoto R.D.** Parallel and Distributed Simulation Systems. WileyInterscience. 2000.
4. **Bakhmurov A., Kapitonova A., Smeliansky R.** DYANA: An Environment for Embedded System Design and Analysis // Proceedings of the 5-th International Conference TACAS'99, Lecture Notes in Computer Science. – 1999. – V. 1579. – P. 390–404.
5. **Bengtsson J., Larsen K. G., Larsson F., Pettersson P., Yi W.** UPPAAL – a Tool Suite for Automatic Verification of Real-Time Systems // Lecture Notes in Computer Science. – 1996. – P. 232–243.
6. **Dalsgaard A., Olesen M., Toft M., Hansen R., Larsen, K.** METAMOC: Modular Execution Time Analysis using Model Checking // Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, Germany. – 2010 – P. 113–123.
7. **Adelantado M., Busseno J.L., Rousselot J.Y., Siron P., Betoule M.** HP-CERTI: towards a high performance, high availability open source RTI for composable simulations // Fall simulation interoperability workshop, Orlando, USA. – 2004.
8. **Andreas Knüpfner, Holger Brunst, Allen D. Malony, Sameer S. Shende.** Open Trace Format API Specification. Version 1.1. November 13, 2006.
9. **Bakhmurov A. G., Balashov V. V., Pashkov V.N., Smeliansky R.L., Volkanov D. Yu.** Method For Choosing An Effective Set Of Fault Tolerance Mechanisms For Real-Time Embedded Systems, Based On Simulation Modeling // Problems of dependability and modelling /eds. Jacek Mazurkiewicz [i in.]. Wrocław // Oficyna Wydawnicza Politechniki Wrocławskiej. – 2011. – P. 13–26.