

12. Кабанихин С.И. Обратные и некорректные задачи. Новосибирск: Сибирское научное издательство, 2009. – 457 с.
13. Тихонов А.Н., Арсенин В.Я. Методы решения некорректных задач. М.: Наука, 1979. – 284 с.
14. Баскаков С.И. Радиотехнические цепи и сигналы. М.: Высшая школа, 1983. – 536 с.
15. Анго А. Математика для электро- и радиоинженеров. Пер. с франц. М.: Наука, 1964. – 772 с.
16. Сергиенко А.Б. Цифровая обработка сигналов. Спб.: «Питер», 2002. – 608 с.
17. Simon M.K., Omura J.K., Scholtz R.A., Levitt B.K. Spread Spectrum Communications Handbook. McGraw-Hill, Inc., New York, 2002. – 1229 p.
18. Pickholtz R.L., Schilling D.L., Milstein L.B. Theory of Spread-Spectrum Communications – A Tutorial// IEEE Transactions on Communications. Vol. COM-30, № 5, May 1982. – P. 855-884.
19. Теория электрической связи: Учебник для вузов. Под ред. Д.Д.Кловского. М.: Радио и связь, 1999. – 432 с.

METHOD OF COMBINED APPLICATION OF FRACTIONALLY-SPACED PREEQUALIZATION AND SPREAD-SPECTRUM MODULATION FOR PLC-NETWORKS AND WIRELESS CHANNELS

Bagmanov V.Kh., Lyubopytov V.S., Sultanov A.Kh., Tlyavlin A.Z.

In the article a method of signal processing on the transmitter side is described, utilizing jointly fractionally-spaced equalization and spread-spectrum modulation. Thus two crucial problems of digital signal reliable transmission over the distorting channel with narrowband noise are solved: fractionally-spaced preequalization compensates for intersymbol interference caused by signal distortion, and spread spectrum allows transmission system to be robust against electromagnetic interference. Method efficiency in the sense of error probability and transmit power minimizing is considered.

***Keywords:** intersymbol interference, linear distortions, signal equalization, digital signal processing, spread spectrum systems.*

Багманов Валерий Хусаинович, д.т.н., профессор Кафедры «Телекоммуникационные системы» (ТС) Уфимского государственного авиационного технического университета (УГАТУ). Тел. (8-347) 273-06-89. E-mail: tks@ugatu.ac.ru.

Любопытов Владимир Сергеевич, аспирант кафедры ТС УГАТУ. Тел. 8-927 944-88-99. E-mail: untrivial@rambler.ru.

Султанов Альберт Ханович, д.т.н., профессор, заведующий Кафедрой ТС УГАТУ. Тел. (8-347) 273-06-89. E-mail: tks@ugatu.ac.ru.

Тлявлин Анвар Зуфарович, к.т.н., начальник Учебно-методического управления УГАТУ. Тел. (8-347) 273-79-38. E-mail: azt@ugatu.su

УДК 004.057.4

СИМУЛЯТОРЫ БЕСПРОВОДНЫХ MANET-СЕТЕЙ

Жолобов А.Н., Прозоров Д.Е., Романов С.В

В обзоре рассмотрены системы имитационного моделирования беспроводных сетей связи, позволяющие исследовать и разрабатывать собственные протоколы маршрутизации. Приведены критерии оценки систем сетевого моделирования, отмечены достоинства и недостатки.

Ключевые слова: сетевой симулятор, самоорганизующаяся сеть, беспроводная связь, моделирование.

Введение

Современный этап развития человеческого общества характеризуется усложнением структур инфокоммуникационных сетей,

расширением типов и увеличением объема передаваемого трафика. Активно развивающейся в настоящий момент областью беспроводных инфокоммуникационных сетей являются MANET – мобильные самоорганизующиеся сети (Mobile Ad-hoc NETworks). В отличие от фиксированной, иерархической структуры традиционных сотовых сетей связи с выделенными управляющими центрами, MANET используют распределенные принципы управления сетью с возможностью самоорганизации и самоуправления узлов сети. Подобные сети могут применяться во время военных действий, в структурах МЧС,

в транспортных системах и различных силовых структурах [1-2].

Одноранговые мобильные ad-hoc сети характеризуются динамически изменяющейся топологией и отсутствием жесткой инфраструктуры. Кроме того, взаимодействие узлов беспроводных ad-hoc сетей ограничено выделенными частотными ресурсами, энергоемкостью узлов, условиями распространения радиосигнала и т.п. Отдельные абонентские терминалы могут передвигаться, появляться в зоне действия сети и уходить из ее влияния. Узлы сети взаимодействуют случайным образом, при этом связь между парой узлов может осуществляться через цепочку посредников.

К функциям сетевых узлов относятся не только прием и обработка данных, но и прокладка маршрута к конечному адресату. Описанные выше особенности MANET приводят к тому, что протоколы и методы передачи данных классических проводных сетей с явно выделенными маршрутизаторами в ad-hoc сетях оказываются неэффективными [1-2; 15]. Маршрутизация пакетов данных с учетом мобильности узла-отправителя, узла-приемника и транзитных узлов при масштабировании сети до единиц и десятков тысяч абонентов в таких условиях становится чрезвычайно сложной. Таким образом, задача исследования и разработки протоколов маршрутизации, быстро адаптирующихся к изменениям топологии сети и имеющих небольшие накладные расходы на поиск и поддержку маршрутов, требует тщательной проработки и исследования моделей MANET-сетей в системах имитационного моделирования сетей связи.

Системы имитационного моделирования сетей связи

Основной особенностью программных пакетов, позволяющих имитировать работу сетей связи, является возможность отслеживания (трассировки) событий, происходящих на различных уровнях (физическом, канальном, сетевом и т.д.) модели сети. Это программное обеспечение должно достаточно точно воспроизводить все основные особенности и параметры реального оборудования.

Разработку имитационных моделей стимулирует значительное сокращение времени исследования моделируемых систем за счет подмены процесса, происходящего в реальном масштабе времени, на ускоренный или замедленный процесс смены событий в симуляторе

[3]. Кроме того, симуляторы позволяют оценить работу сети в значительно более широком диапазоне изменения ее параметров, таких как число сетевых узлов, типы используемых протоколов, характеристики среды распространения радиосигналов и многих других. Имитационное моделирование также незаменимо для анализа работоспособности сложных стохастических систем, какими и являются мобильные самоорганизующиеся сети связи.

В отличие от математической модели, имитационная модель не всегда содержит в себе строгие математические описания всех зависимостей между параметрами моделируемых процессов. Напротив, имитационное моделирование используется тогда, когда строгое математическое описание процессов в моделируемой системе невозможно или крайне затруднено, когда речь идет о сложной системе, зависимости между параметрами которой не известны в необходимой степени, а полное воспроизведение их приводит к излишне громоздким системам уравнений. В таких системах подробное математическое описание используется лишь для некоторых уровней моделирования. Например, физический слой, описывающий среду передачи, можно описать как имитационной моделью, что будет весьма ресурсоемко, так и аналитическим выражением [3].

Сетевые симуляторы в ходе своей работы могут собирать статистику о наиболее важных событиях, происходящих в модели: количестве потерянных пакетов данных, времени задержки распространения пакетов, коэффициентах использования каналов и узлов и т.п.

На данный момент рынок программных пакетов для имитационного моделирования сетей наполнен множеством решений. По области решаемых задач эти решения варьируются от небольших узкоспециализированных утилит, нацеленных на моделирование конкретного типа сетей (например только беспроводных сенсорных сетей) до крупных программных пакетов, стремящихся охватить все многообразие телекоммуникационных сетей.

С позиции разработки моделей возникает дилемма. С одной стороны, необходимо использовать низкоуровневые языки. Как правило, это Си или Си++. Программный код на них хорошо оптимизируется и требует малых вычислительных затрат. Проблемы же низкоуровневых языков заключаются в сложности разработки и необходимости времени для компилирования модели. Это заметно снижает темпы разработки. С дру-

гой стороны, использование высокоуровневых языков позволяет упростить программный код и вносить поправки без перекомпилирования, однако скорость работы таких языков ощутимо ниже. В качестве таких языков могут использоваться Tcl, Python, Java.

Различные разработчики программных продуктов решают эту дилемму по-своему. Некоторые используют связку из двух языков: один, низкоуровневый, они отводят для описания ресурсоемких частей модели, которые не требуют постоянного редактирования (узлы, физические модели), другой, высокоуровневый – для часто изменяемых частей модели (топология сети). Сложность такого решения заключается в необходимости специальных средств для сопряжения этих языков. Второй способ решения – использование средств, ускоряющих работу высокоуровневых языков, таких как «байт-код». Таким образом, за счет времени, потраченного на компиляцию, повышается общая скорость работы модели. Третий способ – надстройка над существующими низкоуровневыми языками специализированных фреймворков, упрощающих разработку различных частей модели. Из-за повышения уровня абстракции несколько снижается скорость модели, однако ощутимо упрощается разработка. Также выбором языка программирования определяются такие параметры, как мультиплатформенность и возможность параллельного вычисления на кластерах.

Существует ряд систем имитационного моделирования, которые не дают пользователю доступа к уровню программирования. Такие системы обладают малой гибкостью и применимы для очень узкого круга задач.

С точки зрения политики лицензирования, можно выделить платные пакеты, пакеты бесплатные для академического использования и пакеты, распространяющиеся свободно. Платные пакеты являются собственностью компаний-разработчиков, которые позволяют пользоваться своим продуктом за деньги. Стоимость такого пакета может достигать нескольких десятков тысяч долларов США.

Существует множество программных продуктов, разрешающих бесплатное использование в некоммерческих целях для частных лиц, некоммерческих организаций, учебных заведений и т.д. Однако они требуют оплаты в случае использования программного продукта с целью извлечения прибыли. Такие ограничения оговариваются в лицензии продукта. Исходный код находится либо в свободном доступе, либо

высылается по запросу. Цена, как правило, не указана и является предметом договора пользователя с разработчиками. Наиболее предпочтительные, с этой точки зрения, так называемые открытые программные пакеты. Исходный код таких программ доступен для просмотра, изучения и изменения, что позволяет пользователю принять участие в доработке самой программы, использовать код для создания новых программ и исправления в них ошибок через заимствование или через изучение использованных алгоритмов и методов. Рассмотрим некоторые наиболее популярные пакеты.

Программный пакет Network Simulator 2 (NS-2) – система имитационного моделирования дискретных событий. Этот пакет является программным обеспечением с открытым исходным кодом. Разработка сетевого симулятора NS-2 началась в 1995 г. в рамках проекта VINT (Virtual InterNetwork Testbed), организованного DARPA (Defense Research Projects Agency) и реализуемого под руководством целого ряда научных организаций и центров: USC/ISI (University of Southern California / Information Sciences Institute), Xerox PARC, LBNL (Lawrence Berkley National Laboratory) и UCB (UC Berkley) [3]. За основу был взят существующий в то время открытый пакет REAL network simulator, разрабатываемый с 1989 года в Калифорнийском университете. На сегодня основными спонсорами проекта являются DARPA, NSF и ACIRI (AT&T Center for Internet Research at ICSI).

NS-2 предлагает для написания модели низкоуровневый язык Си++ и высокоуровневый – Tcl. В качестве связующего языка используется объектная модификация Tcl – OTcl. Использование языка Си++ позволяет описывать объекты сложными математическими моделями. Благодаря этому в пакете есть множество библиотек для математического моделирования физической среды. Являясь открытым программным обеспечением, его исходный код может быть скомпилирован для использования в операционных системах UNIX, GNU/Linux, Mac OS, семейство MS Windows с использованием UNIX-подобной среды Cigwin. Актуальная на данный момент рабочая версия ns-2.34 датирована июнем 2009 г. [4].

В 2005 г. возникло предложение в корне пересмотреть текущую архитектуру NS-2. В результате дальнейшей дискуссии произошло формирование основных принципов новой системы имитационного моделирования, названной Network Simulator 3 (NS-3). В первую очередь

разработчики решили отказаться от языка программирования Tcl в пользу Си++. Возможность высокоуровневого программирования оставили, реализовав ее при помощи языка Python. Переработка ядра позволила как увеличить производительность и масштабируемость моделирования, так и производить вычисления на кластерах. Из-за смены высокоуровневого языка программирования NS-3 обладает лишь частичной обратной совместимостью. Многие библиотеки NS-2 были портированы для использования в NS-3.

NS-3 является свободным программным обеспечением, распространяемым под лицензией GNU GPLv2, и ориентирован на исследовательское применение, а также применение в образовательных целях. Исходные коды NS-3 открыты для исследования, модификации и использования и доступны на сайте проекта <http://www.nsnam.org> [5-6].

Основная цель проекта NS-3 – создание симулятора для IP-сетей, поддерживающего работу на уровнях 2-4 стека сетевых протоколов с дискретным механизмом обработки событий [7-8]. Разработчикам предоставляется возможность построения как собственных моделей любой сложности, так и, благодаря используемой лицензии GNU GPLv2, изменение и дополнение уже существующих моделей, входящих в комплект программного обеспечения.

В NS-3 разработаны модели беспроводных сетевых технологий, включая большое число вариантов стандарта IEEE. Работы над созданием симулятора NS-3 продолжают и в настоящий момент. Установка NS-3 проводится из исходного кода или готовых дистрибутивов для большинства популярных платформ. Он может запускаться в таких операционных системах как UNIX, GNU/Linux, Mac OS, семейство MS Windows. Последняя актуальная версия – NS-3.12, выпущена в августе 2011 г.

Система OPNET Modeler – система имитационного моделирования и анализа сетей связи. Разрабатывается компанией OPNET Technologies, Inc с 1986 г. Входит в пакет программ для проектирования локальных и глобальных сетей. Имеет множество готовых модулей для моделирования протоколов и устройств. Поддерживает три режима имитации: имитация дискретных событий, имитация в аналитической форме, смешанная имитация. Обеспечивает параллельные вычисления на кластере. Последняя актуальная версия продукта – 17.1. Операционной системой для работы является MS Windows, так же есть порты для дистрибутивов

GNU/Linux, но гарантированная работа заявлена только с ядром 2.6.18 [9].

OMNeT++ – это фреймворк с открытым исходным кодом для разработки приложений имитационного моделирования. Является бесплатной академической версией программного продукта OMNEST, разрабатываемой компанией Simulcraft, Inc. с 1992 г. Основным языком разработки является язык Си++. Для описания топологии сетей используется особый высокоуровневый язык NED с Си-подобным синтаксисом. В пакет включена интегрированная среда разработки, основанная на Eclipse IDE. Существует ряд свободно распространяемых библиотек, основанных на OMNeT++, для работы с беспроводными сетями (MiXiM [10], Mobility Framework [11]) для работы с готовыми протоколами (INET Framework [12]). OMNeT++ – кроссплатформенный пакет и может работать в операционных системах MS Windows и GNU/Linux. Текущая актуальная версия номер 4.2 за ноябрь 2011 г. доступна для скачивания на сайте без ограничений [13].

GloMoSim (Global Mobile Information System Simulator) – система имитационного моделирования, рассчитанная на широкую масштабируемость и упрощенную компоновку, включает в модель множество слоев, связанных между собой API-интерфейсами [14]. GloMoSim – бесплатная академическая версия пакета QalNET, разрабатываемого компанией Scalable Network Technologies. Версия GloMoSim не включает в себя большинство современных протоколов и предназначена исключительно для обучения. GloMoSim написана на языке программирования PARSEC – специальной версии языка Си, рассчитанной на параллельные вычисления. PARSEC разработан в Parallel Computing Laboratory at UCLA. Крупноформатность этого языка обеспечивает работу GloMoSim в разных операционных системах: Solaris, GNU/Linux, FreeBSD, MS Windows [15]. Последняя версия 2.03, датированная декабрем 2001 г., и доступна для скачивания с сайта [16].

JiST (Java in Simulation Time) – высокопроизводительная система имитационного моделирования, использующая так называемый принцип моделирования в виртуальной машине. Этот метод позволяет достичь максимальной, на данный момент, производительности модели [17]. SWANS (Scalable Wireless Ad hoc Network Simulator) – система моделирования беспроводных сетей, основанная на виртуальной машине JiST. Отличительной особенностью данной

системы является высокая масштабируемость и производительность. JiST/SWANS разработана на языке Java, что обеспечивает ее кроссплатформенность. Компания Cornell Research Foundation, Inc. начала разработку этой системы в 2004 г. В соответствии с лицензионным договором бесплатное использование разрешено только в некоммерческих и академических целях. Текущая версия продукта – 1.0.6 датирована сентябрем 2008 г. доступна для свободного скачивания с сайта [18].

SimPy (Simulation in Python) – язык программирования, основанный на Python, рассчитанный на имитационное моделирование дискретных событий. Включает в себя графический интерфейс, систему визуализации [19]. Основным достоинством SymPy является простота синтаксиса и скорость реализации модели, что также является и достижением языка Python. К сожалению, в состав SimPy не входят библиотеки, описывающие реальные протоколы, поэтому система представляет собой лишь чистые API-функции для прототипирования [17]. Язык программирования Python обеспечивает приложению мультиплатформенность. SymPy является программным обеспечением с открытым исходным кодом, выложенным в свободный доступ под лицензией LGPL. Последняя актуальная версия SimPy 2.2 вышла в сентябре 2011 г.

Выводы

Рассмотренные системы имитационного моделирования можно классифицировать по следующим признакам: открытость исходного кода (существуют системы с открытым и закрытым исходным кодом); политика лицензирования (существуют свободные системы, системы бесплатные для академического использования и платные коммерческие системы); поддерживаемые платформы (системы могут работать под управлением таких платформ, как GNU/Linux, Mac OS, Windows, Solaris, FreeBSD). Если система поддерживает больше одной платформы, то она называется кроссплатформенной. Такие опции, как дополнительные библиотеки, системы визуализации и поддержка распределенных вычислений, могут быть встроены в базовую систему, поддерживаться при помощи сторонних средств либо не поддерживаться вообще.

Выделим наиболее важные критерии отбора сетевого симулятора:

- программное обеспечение с открытым исходным кодом;
- политика лицензирования;
- кроссплатформенность;
- поддержка распределенных вычислений;
- поддержка беспроводных стандартов связи.

Сравнение сетевых симуляторов по этим критериям приведено в таблице 1.

Таблица 1. Сравнение систем имитационного моделирования

	Исходный код	Лицензия	Кроссплатформенность	Поддержка распределенных вычислений	Поддержка беспроводных стандартов связи
NS-2	Открыт	Свободная	GNU/Linux, Mac OS, Windows	Сторонние библиотеки	Встроенная
NS-3	Открыт	Свободная	GNU/Linux, Mac OS, Windows	Встроенная	Встроенная
OPNET	Закрыт	Коммерческая	GNU/Linux (ограниченная), Windows	Встроенная	Встроенная
OMNet++	Открыт	Бесплатная для академического использования	GNU/Linux, Windows	Сторонние библиотеки	Встроенная
GloMoSim	Открыт	Бесплатная для академического использования	GNU/Linux, Mac OS, Windows, Solaris, FreeBSD	Встроенная	Встроенная
JiST / SWANS	Открыт	Бесплатная для академического использования	GNU/Linux, Windows, Solaris, FreeBSD	Отсутствует	Встроенная
SimPy	Открыт	Свободная	GNU/Linux, Windows, Solaris, FreeBSD	Отсутствует	Отсутствует

Литература

1. Hong X. et al. Scalable routing protocols for mobile ad hoc networks // IEEE Network. Vol. 16, №4, 2002. – P. 11-21.
2. Винокуров В.М. и др. Маршрутизация в беспроводных мобильных Ad hoc-сетях // Доклады ТУСУРа. Ч. 1, № 2 (22), 2010. – С. 288-292.
3. Галкин А.М. и др. Пакет моделирования NS-2 // СПб.: Изд. СПбГУТ, 2007. – 64 с.
4. ns: Change History // URL: <http://isi.edu/nsnam/ns/CHANGES.html> – 25.10.2011.
5. Балашов В. Обзор сетевого симулятора NS-3. LVEE 2011. The 7th International conference of developers and users of free open source software // URL: http://lvee.org/en/reports/LVEE_2010_31 – 25.10.2011.
6. NS-3 // URL: <http://www.nsnam.org> – 25.10.2011.
7. Henderson T.R., Floyd S., Riley G.F. NS-3 Project Goals // URL: <http://www.nsnam.org/docs/meetings/wns2/wns2-ns3.pdf> – 25.10.2011.
8. Ерыгина Т. Проект NS-3 // URL: <http://masters.donntu.edu.ua/2007/fvti/erygina/library/lib9.htm> – 25.10.2011.
9. Modeler 17.1 System Requirements // URL: http://www.opnet.com/solutions/system_requirements/modeler/index.html – 25.10.2011.
10. MiXiM project // URL: <http://mixim.sourceforge.net/> – 25.10.2011.
11. Mobility Framework for OMNeT++ 4 // URL: <https://github.com/mobility-fw/mf-opp4/wiki> – 25.10.2011.
12. INET Framework // URL: <http://inet.omnetpp.org/> – 25.10.2011).
13. OMNeT++ Downloads // URL: http://www.omnetpp.org/component/docman/cat_view/17-ownloads/1-omnet-releases – 25.10.2011.
14. Xiang Zeng, Bagrodia R., Gerla M. GloMoSim: A Library for Parallel Simulation of Large-scale Wireless Networks // Department of Computer Science, University of California, Los Angeles, CA 90095.
15. UCLA Parsec Programming Language // URL: <http://pcl.cs.ucla.edu/projects/parsec/> – 25.10.2011.
16. GloMoSim Registration Receipt // URL: <http://pcl.cs.ucla.edu/projects/gloMosim/academic/download.html> – 25.10.2011.
17. Weingärtner E., Von Lehn H., Wehrle K. A performance comparison of recent network simulators // IEEE International Conference on Communications, ICC. Dresden, June 2009. – P.1-5.
18. JiST – Java in Simulation Time / SWANS – Scalable Wireless Ad hoc Network Simulator // URL: <http://jist.ece.cornell.edu/> – 25.10.2011.
19. SimPy Overview // URL: http://simpy.sourceforge.net/simpy_overview.htm – 25.10.2011.

SIMULATORS OF WIRELESS MANET-NETWORKS**Zholobov A.N., Prozorov D.E., Romanov S.V.**

Simulators of wireless networks, allowing to research and develop own routing protocols are considered in the article. Criteria of an estimation of network simulators and their advantages and limitations are denoted.

Keywords: *network simulator, ad-hoc, wireless network, modeling.*

Жолобов Алексей Николаевич, аспирант Кафедры «Радиоэлектронные средства» (РС) Вятского государственного университета (ВГУ). раб. Тел. (8-833) 235-72-59; 8-909-140-11-91. E-mail: zholobovan@gmail.com

Прозоров Дмитрий Евгеньевич, д.т.н., профессор Кафедры РС ВГУ. Тел. (8- 833) 235-72-59; 8-912-711-21-11. E-mail: prozorov.de@gmail.com

Романов Сергей Владимирович, научный сотрудник ОАО «НИИ средств вычислительной техники» (г. Киров). Тел. (8- 833) 267-96-29; 8-922-668-06-79. E-mail: romanov@niisvt.kirov.ru