

УДК [519.872+519.876.5]

ОЦЕНКА СИСТЕМЫ МОДЕЛИРОВАНИЯ GPSS WORLD

Ю. И. Рыжиков,

д-р техн. наук, профессор

Военно-космическая академия им. А. Ф. Можайского

Дан обзор проблемы математического моделирования систем и сетей с очередями. Описаны новые возможности, предоставляемые имитационной системой общего назначения GPSS World. Приводятся результаты ее тестирования. Предлагается комбинированное (в зависимости от сложности задачи, временных ресурсов и квалификации разработчика) использование численных методов теории очередей, программирования на языках типа Фортран и упомянутой системы.

It is presented a review of the mathematical modeling of the queuing systems and networks, and the description of new possibilities of the GPSS World. The results of its testing are discussed. The combined (depending on the problem complexity, needed precision, time resources and explorer qualification) using of numerical queuing theory methods, programming in Fortran and GPSS modeling is proposed.

В разных областях техники, в организации производства, в социальной сфере и военном деле постоянно возникает необходимость решения вероятностных задач, связанных с работой систем массового обслуживания разного вида требований. Термин «массовое обслуживание» предполагает многократную повторяемость ситуаций в том или ином смысле (много прибывших в систему и обслуженных заявок, большое число находящихся в эксплуатации аналогичных систем) и статистическую устойчивость картины. Выводы и рекомендации, получаемые методами теории массового обслуживания (ТМО), применимы лишь при наличии одного или обоих перечисленных факторов повторяемости.

По справедливому замечанию виднейшего советского специалиста в этой области Б. В. Гнеденко, легче указать ситуации, где не может быть использована ТМО (она же — теория очередей), чем перечислить все сферы ее потенциального применения. Подчеркнем, что задачи указанных типов приходится решать не только при проектировании вновь создаваемых систем и сетей обслуживания (СМО, СеМО), но и в процессе эксплуатации имеющих: при увеличении нагрузки, изменении трудоемкости обработки заявок, выходе из строя, деградации или модернизации техники, снижении квалификации персонала, пересмотре требований к оперативности обработки заявок и т. п.

Многообразие приложений теории массового обслуживания определяет постоянно растущий интерес к ней, а сложность возникающих задач не позволяет получить исчерпывающие решения на базе аналитических методов [1, 4] даже при чис-

ленной реализации последних. В таких ситуациях приходится прибегать к имитационному моделированию. В [3] приведен перечень заслуживающих внимания источников, дан обзор теоретических основ построения имитационных моделей и техники их реализации на алгоритмических языках высокого уровня и специальных языках моделирования.

Ни один из языков моделирования не оказал на имитацию столь большого воздействия, как GPSS (General Purpose Simulation System — система имитационного моделирования общего назначения) [6–10]. Например, сходство языков СЛЭНГ и GPSS простирается вплоть до заимствования большинства ключевых слов.

Базовые понятия GPSS

Модель GPSS состоит из сети блоков, представляющих необходимые действия или задержки транзактов (клиентов, покупателей, ремонтируемые телевизоры, телефонные звонки, электронные сигналы и т. п.), которые последовательно проходят через блоки. Например, блок GENERATE вводит в модель новые транзакты, воспроизводя рекуррентный поток заявок с требуемым распределением интервалов между смежными заявками. Транзакты перемещаются в системных времени и пространстве, переходя от одного блока модели к другому и воздействуя на них. Транзакты возникают и уничтожаются, могут расщепляться и сливаться. Входя в блок, транзакт вызывает определяемую типом блока подпрограмму, которая обрабатывает

соответствующее событие. Далее транзакт в общем случае пытается войти в следующий блок. Продвижение продолжается до тех пор, пока не окажется, что очередной блок должен выполнить одну из следующих функций:

- удалить транзакт из модели;
- временно заблокировать его в предыдущем блоке до выполнения некоторых условий;
- задержать его на определяемое моделью время.

Тогда начинается продвижение другого транзакта и т. д. — до завершения моделирования. За один шаг работы управляющей программы производится поочередный просмотр всех имеющихся процессов и имитация тех, которые могут быть запущены. При этом может освободиться ранее занятый ресурс или оказаться порожденным процесс более высокого приоритета, и просмотр списка процессов можно начать сначала.

Описание траектории транзакта содержит порядок и имена используемых им «устройств» (приборов обслуживания) или «памятей»; временные задержки; логические условия, управляющие продвижением транзактов; точки маршрута, в которых производится сбор данных об ожидании или контролируется время прохождения и т. п. Некоторые из названных объектов должны быть предварительно описаны специальными командами. Параметры транзакта могут менять свои значения на траектории, оказывать влияние на маршрут и собираемую статистику. Маршруты следования транзактов, временные характеристики, условие прекращения моделирования, требуемые показатели эффективности задаются пользователем.

При описании этих процессов и сборе итоговой статистики используются таймер модельного времени, стандартные числовые атрибуты (СЧА) и параметры транзактов, а также определяемые пользователем переменные, выражения и таблично задаваемые функции. Наиболее сложные аспекты модели — планирование предстоящих событий и очередности их обработки — автоматически реализует планировщик событий (интерпретатор) *GPSS*.

GPSS World

Система *GPSS* прошла длинный путь развития. До недавних пор ее главными недостатками были низкий уровень входного языка, невозможность реализации нестандартных вычислений, скрытость от программиста внутренней логики поведения модели. Они резко ограничивали реальную полезность системы. Радикальные изменения произошли одновременно с переходом к ее *Windows*-версиям. «Студенческая» версия *GPSS World 4.2.1* (2000 г.), далее сокращенно именуемая *GPSS/W*, бесплатно распространяется через Интернет (электронный адрес www.minutemansoftware.com); она отличается от коммерческой лишь количественными ограничениями (до 150 блоков в составе модели) и несовместимостью получаемых программ с составленными ранее на *GPSS/PC*.

Инсталляция

Для работы с *GPSS/W* необходимо иметь операционную систему *Windows 95, 98* или *2000*; ПЭВМ класса *Pentium*; оперативную память не менее 32 Мб; 10 Мб места на винчестере. От пользователя требуются опыт работы с персональной ЭВМ, навыки программирования на языке типа *Паскаль* или *Фортран*, знакомство с основными идеями имитационного моделирования.

Заархивированная система укладывается в пять дискет. Tutorial должен загружаться отдельно. Инсталляция запускается по *Setup*. Транслированные программы (Имитации) и отчеты помещаются в ту же директорию, где находятся исходные модели.

В состав электронной документации к системе входят Справочник (*GPSS World Reference Manual*), Руководство программиста (Tutorial) и подробно прокомментированные примеры тестовых программ (все — с расширением *GPS*), собранные в папке *Sample Models*. Объектами моделирования являются:

- производственные задачи (*ASSEMBLY* — процесс сборки насосов; *BICYCLE* — велосипедная фабрика; *MANUFACT* — система производства электроники; *FOUNDRY* — литейный цех; *TVREPAIR* — мастерская по ремонту телевизоров; *TEXTILE* — текстильная фабрика; *QCONTROL* — система контроля качества; *TAPEPREP* — подготовка управляющих программ для металлорежущего оборудования с ЧПУ; *FMSMODEL* — гибкое автоматизированное производство; *ROBOTFMS* — роботизированное производство);

- транспортные задачи (*TRAFFIC* — транспортные потоки на Т-образном перекрестке; *TURNSTIL* — турникет на футбольном стадионе; *SHIPPORT* — порт; *LOCKSIMN* — канал и шлюз);

- задачи управления запасами (*OILDEPOT* — нефтебаза; *ORDERPNT* — запас с критическим уровнем возобновления; *PERIODIC* — запас с периодическим контролем уровня; *STOCKCTL* — иерархическая система складов);

- проблемы торговли (*POWDER* — торговля стиральным порошком; *SUPERMRK* — универсам);

- системы и сети связи (*TELEPHON* — простая телефонная система; *EXCHANGE* — телефонный коммутатор с внутренними и внешними линиями связи; *ETHERNET* — сеть Ethernet);

- абстрактные постановки общеметодического характера (*MULTIWAY* — сеть Ethernet; *GRAECOLATIN* — планирование эксперимента с применением греко-латинских квадратов; *QTHEORY* — задача теории очередей с известным решением; *PREDATOR* — динамика популяций).

В той же папке имеются многочисленные примеры моделей из нестарейшей книги Т. Шрайбера [9], перепрограммированные с учетом новых возможностей *GPSS/W*. Соответственно в упомянутой книге можно найти развернутые пояснения к этим моделям (за ними сохранены данные Шрайбером номера). Разумеется, надо иметь в виду возможности аналогий (*Barber Shop* может служить моделью не только парикмахерской, но и космети-

ческого салона, мастерской по изготовлению ключей, ремонтных служб, служб спасения, пунктов первой медицинской помощи и т. п.). Кроме того, подходящим может оказаться только название проблемной области, но не специфика ситуации. Поэтому предпочтительны абстрактная формулировка задачи (в терминах теории очередей) и инвариантные к приложениям приемы моделирования.

Попутно отметим, что специалистам в области техники связи и компьютерных систем будут полезны примеры из учебного пособия Б. Я. Советова и С. А. Яковлева [7]. В частности, там разобраны задачи моделирования абстрактной ациклической сети обслуживания, участка транспортного цеха автомобильного завода, процесса передачи данных в информационно-вычислительной сети с коммутацией пакетов, работы концентратора сети интегрального обслуживания, испытательного участка сборочного цеха предприятия, локальной сети связи ЭВМ со случайным доступом к моноканалу, вычислительного центра коллективного пользования, а также приведены 45 разнообразных заданий на моделирование.

Принципиальные особенности *GPSS/W* можно сгруппировать в несколько разделов.

1. Каждая разработка в духе современных технологий программирования рассматривается как *проект* (эта терминология в электронной документации к системе не используется). Проект состоит из объектов четырех типов: 1) исходный модуль; 2) Имитация; 3) отчеты; 4) текстовые файлы. Модель может иметь иерархическое строение и тогда включает в себя командой *INCLUDE* ранее разработанные фрагменты. В рамках проекта может существовать множество автоматически нумерованных версий. Размеры задачи физически доступным объемом оперативной памяти не лимитируются.

Для каждого из шагов проекта есть свои *инструменты*: полноэкранный Текстовый редактор, пункт *Create Simulation* в меню *Command*, множество окон контроля и команд управления моделированием, автоматический генератор отчетов, средства настройки отчетов, статистического анализа. Команда *CONDUCT*, начиная *Эксперимент*, берет на себя управление Имитацией.

В системе реализованы элементы объектного программирования в классическом его понимании: *наследование свойств* (установок) Имитации от исходной модели и *полиморфизм* — автоматическое преобразование операндов выражений (в частности, строки символов в число и обратно) к требуемому типу.

2. Интерфейс ориентирован на пользователя. Однострочный текстовый редактор для набора модели заменен на полноэкранный с реализацией стандартных функций *Windows* — табуляция, выделение текста, работа с буфером *Clipboard*, управление типом и размером шрифтов *True Type*, операции с файлами. Есть возможность замены традиционно используемого вместо знака умножения *#* на привычную ***. Компилятор формирует

список синтаксических ошибок, обеспечивает автоматический подвод курсора к месту очередной ошибки и вывод диагностики в статусную строку.

Динамика модели может отслеживаться как в числовой, так и в графической форме с помощью окон около 20 разных типов (в зависимости от класса отображаемых сущностей), причем в двух режимах: детальном и обзорном. Размер и положение окон изменяются стандартными средствами. Степень загруженности ресурсов показывается цветом, можно наблюдать за перемещением активного транзакта. В окна можно выводить СЧА и накопленную статистику. Отображение внизу главного окна значения *таймера* облегчает диагностику зависания. В особо сложных случаях можно получить *Snapshots* — снимки мгновенного состояния *цепей событий*, используемые для отладки логики модели. Имеются средства интерактивного ввода команд, контрольных остановов и трассировки. Результаты моделирования могут быть автоматически представлены в виде гистограмм соответствующих распределений.

3. Резко расширены вычислительные возможности. Отменены обязательная целочисленность операндов и принудительное округление результатов. Максимальное число измерений матрицы увеличено до шести. Используемый при инициализации матриц атрибут *UNSPECIFIED* предупреждает работу с «неприсвоенными значениями». Язык *GPSS* дополнен языком программирования расчетов *PLUS*, близким к традиционным алгоритмическим. На нем можно записывать как выражения для операндов блоков, так и процедуры пользователя, в том числе содержащие циклы и разветвления. Это практические сняло ограничения на формируемые результаты и способы их обработки. Перечень встроенных математических функций дополнен функциями генерации случайных чисел для 24 законов распределения, что избавляет пользователя от необходимого ранее мучительно построения таблиц обратной функции распределения для каждой комбинации параметров.

Пользователь может написать собственную *PLUS*-процедуру формирования псевдослучайных чисел с требуемым законом распределения. При этом аргумент *RNj* следует умножить на 0,001. Обычно используемая в подобных программах замена $1-U$ на одинаково распределенное с ним U здесь недопустима, поскольку в среднем одно из тысячи $\{U_i\}$ окажется нулевым. Поэтому при вычислении $\ln U$ для обычной генерации показательного распределенных величин методом обратной функции гарантируется соответствующее число аварийных остановов модели.

Операции над строками позволяют редактировать стандартный отчет и применять процедуры с текстовыми аргументами. Появились операции с потоками данных (открытие и закрытие файлов, запись и считывание информации из них).

4. Система берет на себя стандартную статистическую обработку результатов моделирования. Появившаяся еще в *GPSS/PC* команда *ANOVA* однофакторного дисперсионного анализа заменена

многофакторным эквивалентом. Имеются средства автоматической разработки и проведения статистических экспериментов (включая дробные факторные) по регрессионному анализу и оптимизации моделей.

По результатам прогона программы автоматически формируется стандартный отчет. Содержание отчета зависит от состава модели и при необходимости корректируется через Установки модели. Нестандартные вычисления могут быть запрограммированы с помощью процедур встроенного языка программирования и выведены в файл.

Работа с системой

Типичный проект начинается с создания или модификации модели. Следует иметь в виду, что модель — не просто текстовый файл: она содержит также настройки Settings, закладки Bookmarks и результаты, которые не могут читаться внешним редактором.

Все настройки задаются при создании модели через меню Edit/Settings. Последующие фазы проекта эти установки только наследуют. На вкладке Simulation задаются число попыток входа в блок до прерывания и ограничение глубины ссылок (для предупреждения зацикливания). Посредством вкладки Reports можно вывести результаты в окно вместо файла, указать желаемые дополнительные подотчеты, выводить результаты с шестью знаками после десятичной точки вместо стандартных трех. Вкладка Random Numbers распределяет по типам блоков потоки случайных чисел. Вкладка Function Keys связывает операторы с функциональными клавишами. Вкладка Expressions позволяет задать список выражений, часто используемых в процессе диалога с моделью, и при необходимости выбирать их из списка через меню Select вместо повторного набора.

Этапы моделирования

Работа начинается с составления GPSS-программы. Посредством команды INCLUDE «<имя_файла>» в нее могут вставляться ранее отлаженные фрагменты (например, PLUS-процедуры). Вставки из файлов автоматически нумеруются, и диагностика ошибок в них сопровождается указанием номера вставки.

Компиляция организуется по команде Command/Create Simulation. Синтаксический контроль достаточно развит (в частности, контролируется парность скобок). При обнаружении ошибок создается их циклический список. Каждый выбор элемента списка вызывает сообщение о типе ошибки в статусную строку и помещает точку вставки точно перед ошибочным элементом. Характер «претензий» системы обычно достаточно ясен: Line 10, Col 20, Invalid Delimiter. Expecting a GPSS Verb.

Наибольшие трудности вызывает использование некорректных имен: слишком коротких или совпадающих со стандартными числовыми атрибутами. Оперативное запоминание внесенных в модель исправлений не требуется: компиляции подвергается правленный текст.

Команда START определяет окончание моделирования: ее счетчик должен уменьшиться до нуля. Любой оператор модели TERMINATE с непустым полем операнда A уменьшает счетчик при входе в него транзактов. Такой оператор в программе должен быть единственным.

Динамические окна могут быть открыты только после компиляции модели командой Window/Simulation Window/ <тип_окна>. Одновременно могут быть открыты несколько окон. Открытие online-окон замедляет моделирование, поскольку система вынуждена ждать их обновления. Эти задержки обычно недооцениваются. Для иллюстрации степени замедления приведем данные по длительности прогонов одной и той же модели:

- все окна закрыты — 4 с;
- выводится таймер — 97 с;
- открыты окно блоков и таймер — 5271 с (почти полтора часа!).

Для сравнения укажем, что та же модель на Фортране была реализована за 0,05 с. Очевидно, что открывать динамические окна следует лишь для режима пошаговой отладки.

Начатое моделирование можно прервать командой HALT, кнопка которой имеется во всех динамических окнах. После ее выполнения можно выбирать блоки и задавать контрольные остановы в них (кнопка с красным светом). При отработке очередного останова первоначальное условие останова, если оно задано по времени, удаляется, а все остальные условия сохраняются. Продолжение моделирования с использованием команды STEP позволяет автоматически остановить его точно после заданного числа входов активного транзакта в блоки модели.

Контролировать промежуточные результаты (и просматривать окончательные) можно через окна Графики (Plot) и Выражений (Expression). Окно Таблиц позволяет наблюдать сходимость исследуемых распределений в смысле гистограммы, среднего значения и среднеквадратического отклонения. В меню Windows поддерживается список всех открытых окон, включая отчеты по прогонам предыдущих версий модели.

Сбор статистики

Для типовых имитаций автоматически собираемая статистика представляется вполне достаточной. При необходимости в более подробных сведениях можно определить в командах TABLE или QTABLE таблицы (гистограммы) распределений. Фактический сбор данных в первом случае обеспечивается специальными блоками TABULATE, а во втором — при входе в блок DEPART. Упомянутые таблицы (если отметить их в установке отчета),

а также средние значения и среднеквадратические отклонения будут выводиться автоматически и могут просматриваться в окне таблиц.

Еще более детальные сведения можно собирать и обрабатывать с помощью скалярных и матричных *сохраняемых значений* и *PLUS*-процедур. Затем «историческая» база данных может быть обработана с помощью ANOVA для получения доверительных интервалов. Результаты нестандартной обработки статистики можно вывести в файл потока данных.

Эксперименты над моделями

Задача проектирования, например, сети обслуживания должна решаться с позиций системного подхода, учитывающего ее многоаспектность и иерархичность. Естественно, что те же свойства приобретает и процесс проектирования, проходящий уже на фазе моделирования несколько этапов и требующий применения разнообразного математического аппарата. В процессе имитационного *эксперимента* над моделью с учетом случайных исходов необходимо:

- классифицировать факторы на существенные и несущественные;
- разделить и оценить количественно влияния факторов и их комбинаций на целевую функцию;
- найти наивыгоднейшую комбинацию факторов.

Структура эксперимента (количество прогонов, выбор исходных данных для них и характер обработки результатов) определяется его целью. Первая проблема решается средствами дисперсионного анализа, вторая — регрессионного и факторного анализа (скрининг), третья — методами статистической оптимизации. *GPSS/W* имеет встроенные средства для решения всех трех.

Язык *PLUS* имеет особый класс процедур (*EXPERIMENT*) с широкими возможностями, включая исследование поверхности отклика. Эксперимент организуется записанной на языке *PLUS* процедурой управления прогонами, которая для каждого прогона сама готовит исходные данные и вызывает модель. Программа Эксперимента может быть составлена как вручную, так и автоматически — с помощью диалога, реализуемого через меню *Edit/Insert Experiment*. Управление прогонами реализуется с помощью встроенной процедуры *DoCommand(<строка>)*. Аргументы определяют, какая часть поверхности отклика подлежит исследованию в процессе Эксперимента. *EXPERIMENT* заполняет глобальную Матрицу Результатов, которая затем передается процедуре ANOVA.

Дисперсионный анализ

К большинству сложных систем применим принцип Парето, согласно которому 20 % факторов определяют свойства системы на 80 % (аналогичное соотношение известно для авторов научной про-

дукции коллективов [13], наименований предметов снабжения в системах управления запасами [4] и др.). Поэтому первоочередной задачей исследователя имитационной модели является отсеивание несущественных факторов, позволяющее уменьшить размерность задачи оптимизации модели.

Анализ дисперсии (*Analysis of Variation — ANOVA*) оценивает отклонение наблюдений от общего среднего. Затем вариация разбивается на части, каждая из которых имеет свою причину. Остаточная часть вариации (*residual*), которую не удается связать с условиями эксперимента, считается его случайной ошибкой. Для подтверждения значимости используется специальный тест — *F*-статистика.

Дисперсионный анализ определяет, есть ли эффект. Регрессионный анализ позволяет прогнозировать отклик (значение целевой функции) в некоторой точке пространства параметров. Ожидаемое значение отклика вычисляется по формуле регрессии с известными коэффициентами для значений факторов, лежащих внутри выбранных диапазонов. Непосредственной задачей регрессионного анализа является оценка коэффициентов регрессии.

Число уровней каждого фактора обычно ограничивается двумя (тогда объем полного эксперимента с *k* факторами составит 2^k прогонов). Кроме того, желательно уменьшить число прогонов, не жертвуя наиболее важной информацией. Последнее соображение приводит к *дробным* факторным экспериментам. Дробность плана определяется долей числа требуемых прогонов от объема полного факторного эксперимента и может быть целой отрицательной степенью двойки.

Для каждой компоненты плана с учетом расстановки плюс- и минус- единиц в его матрице записывается линейное уравнение регрессии относительно расчетных эффектов и свободного члена. Правой частью уравнения становится результат соответствующего прогона. Компоненты решения делятся на размах (произведения размахов) отображаемых факторов, что дает коэффициенты их влияния относительно середин диапазонов. На этой основе можно планировать дальнейшие действия: игнорировать незначимые факторы, более тщательно исследовать существенные, оптимизировать последние.

Оптимизирующий эксперимент для нахождения оптимума в пространстве непрерывных параметров базируется на методе скорейшего спуска (подъема). Это предполагает построение эмпирической аппроксимации поверхности отклика; вычисление градиента (его проекции являются коэффициентами чувствительности целевой функции к изменению параметров, т. е. коэффициентами регрессии); определение направления спуска и величины шага по нему — с учетом ограничений на значения параметров.

Факторные и оптимизационные эксперименты по автоматически создаваемым Имитациям реализуются очень медленно (на ПЭВМ Pentium I/166 — часами).

Тестирование GPSS/W

Общим правилом работы с новым программным обеспечением математического характера является его тестирование на задачах с известным решением (предпочтительно аналитическим). Для системы массового обслуживания *M/M/1* теоретическая средняя длина очереди $q = \rho^2 / (1 - \rho) = 0,9^2 / (1 - 0,9) = 8,1$. Среднее время ожидания w в соответствии с формулой Литтла (закон сохранения стационарной очереди) должно быть равно q/λ и в данном случае ($\lambda = 1$) численно совпадать со средней длиной очереди. Теоретическая вероятность незанятости системы равна $1 - \rho = 0,1$. Следовательно, в среднем десятая часть заявок должна получать обслуживание без ожидания. Сопоставим с этими ожиданиями результаты моделирования (табл. 1).

В составлении приведенной программы было трудно ошибиться, к тому же она отличается от обсуждаемой у Т. Шрайбера [9] (с. 57 и далее) только типом исходных распределений. Формула Литтла и доля заявок, принимаемых к обслуживанию без ожидания, подтверждаются с достаточно высокой точностью. Это дает основания доверять интерпретатору *GPSS/W*. Показатели, определяемые только средними значениями моделируемых первичных величин (среднее время обслуживания и коэффициент загрузки), также вполне приемлемы.

Время ожидания зависит уже от двух моментов распределения времени обслуживания:

$$W = \frac{\lambda b_2}{2(1-\rho)}$$

Результаты, связанные с ожиданием, оставляют желать лучшего. Причина может быть только одна: недостаточно качественный генератор псевдослучайных чисел. Увеличение числа проведенных испытаний в общем приближает результаты к ожидаемым, но даже при 500 тыс. испытаний погрешность составляет около 17%. Судя по динамике версий, *GPSS/W* проходит процесс интенсивной отладки. Можно надеяться, что встроенные в систему датчики равномерно распределенных чисел будут усовершенствованы. В этом направлении можно приложить и собственные усилия.

В дополнение к приведенным результатам отметим, что для системы *M/M/1* распределение времени пребывания заявки в системе также подчинено показательному закону со средним $1/(1-0,9 - 1/1) = 9$. При ограничителе таймера 200 000 была

■ Таблица 1. Тестирование GPSS/W на модели M/M/1

Показатель	Теория	Число испытаний		
		50 000	200 000	500 000
Коэффициент загрузки	0,900	0,896	0,897	0,899
Среднее время обслуживания	0,900	0,899	0,902	0,901
Число входов	—	49 843	199 042	499 032
из них с нулевым ожиданием	—	5 340	21 001	50 691
Средняя длина очереди	8,100	5,132	5,757	6,690
Среднее время ожидания	8,100	5,148	5,785	6,703

заказана гистограмма распределения времени пребывания; получены среднее значение 9,264 и среднеквадратическое отклонение 9,147. Требуемое для показательного распределения их равенство приближенно выполняется, однако погрешность самих значений составляет около 2%.

Выше приведены контрольные результаты для системы *M/M/1*. Сопоставим с этими ожиданиями результаты моделирования при различных способах задания показательных распределений (встроенная Exponential, табличная ДФР и теоретически точное обращение $-\ln(1 - U)$) и числе испытаний (табл. 2). Результаты, связанные с ожиданием, оставляют желать лучшего даже в случае с теоретически точным переходом от равномерного к экспоненциальному распределению. Причина та же: недостаточно качественный генератор равномерных псевдослучайных чисел. Эти ошибки усугубляются погрешностями перехода к показательному распределению в Exponential и отчасти нейтрализуются (случайный эффект) при использовании табличного преобразования. Увеличение числа проведенных испытаний в общем приближает результаты к ожидаемым.

Обсудим, наконец, временные показатели сравниваемых версий ДСЧ. Для достижения отметки таймера 500 000 при табличной реализации логарифма и при работе с Exponential потребовалось практически одинаковое время — около 135 с (процессор Pentium 1/166). По-видимому, Exponential реализована таблично. При точном обращении показательного распределения было затрачено 187 с.

Прогон аналогичной программы на *Фортране* 77 при тех же исходных данных дал среднюю длину очереди 8,683. Это заметно ближе к теоретической оценке, чем лучший из приведенных в таблице результатов. Прогон занял всего 6,26 с. Столь

■ Таблица 2. Влияние способов генерации показательных распределенных чисел

Показатель	Exponential		Табличная ДФР		$-\ln(1 - U)$	
	50 000	500 000	50 000	500 000	50 000	500 000
Коэффициент загрузки	0,896	0,899	0,900	0,900	0,909	0,902
Среднее время обслуживания	0,899	0,901	0,899	0,901	0,903	0,896
Число входов	49 843	499 032	50 048	499 390	50 320	503 176
из них с нулевым ожиданием	5 340	50 691	5 022	49 823	4 764	50 498
Средняя длина очереди	5,132	6,690	7,770	7,911	7,163	6,831
Среднее время ожидания	5,148	6,703	7,763	7,920	7,118	6,788

большая разница является естественной платой за универсальность интерпретатора *GPSS*. Окупается ли она удобством программирования — зависит от опыта и квалификации пользователя, его желания и возможностей освоить *GPSS*, наличных вычислительных мощностей, сложности моделей и частоты потребностей в моделировании.

Перед моделированием реальной ситуации желательно убедиться, что выбранная процедура действительно воспроизводит случайную величину с требуемым распределением. Гистограмма распределения, порожденная функцией *Exponential* и построенная по 200 тыс. транзактов, оказалась близкой к ожидаемой. Среднее значение вместо заказанной единицы составило 1,005, а среднеквадратическое отклонение — 1,006.

Общая оценка *GPSS World*

GPSS/W является весьма ценным инструментом имитационного моделирования, свободным от ограничений аналитических и численных методов, достаточно «прозрачным», допускающим нестандартную обработку данных и снимающим с программиста множество нетривиальных проблем программирования и отладки моделей. Тем не менее, приходится отметить наличие у нее ряда серьезных недостатков.

1. Громоздкость системы и явная перегруженность встроенными возможностями (многообразие примитивов); непомерное разнообразие графических обозначений блоков, не поддерживаемое даже новейшими версиями «чертежной» системы *Visio*.

2. Медленная работа интерпретатора.

3. Отсутствие концептуального единства. Достаточно указать, например, различие в обращении к элементам матрицы при простой ссылке и изменении значения, круглые индексные скобки в основном тексте и квадратные — в *PLUS*-выражениях; обязательность приставки *MX\$* при ссылке на глобальную матрицу в тексте модели и столь же обязательное ее отсутствие внутри процедур; контекстно зависимый вид ссылок на параметры активного транзакта; потерявшее смысл в *GPSS/W* различие переменных *VARIABLE* и *FVARIABLE*; выражение коэффициента использования устройства в тысячных долях.

4. Путаная семантика операндов: в разных операторах одно и то же значение закреплено за разными операндами. Было бы гораздо легче работать с ключевыми операндами, имеющими значимые имена.

5. Неудачные обозначения операторов отношения *L*, *G*, *E* (было бы лучше согласовать с фортрановскими *LT*, *GE*, *EQ*); арифметическое *SQR* используется для квадратного корня (в *Паскале* так обозначается квадрат); в связи с «числовым» представлением логических значений и объединением понятий числового равенства и логической эквивалентности нарушено общепринятое старшинство операций; состояние логических ключей описывается как *SET* и *RESET* (буквальный перевод «уста-

новлен» и «установлен заново») вместо *ON*, *OFF*; операнд *RE* (традиционный смысл этой приставки — повторение действия) означает удаление.

6. Однократность прерываний устройств и недопустимость прерываний для памяти, которые могут быть использованы для моделирования многоканальных устройств обслуживания и, следовательно, окажутся подвержены прерываниям.

7. Дополнительные установки действий функциональных клавиш для каждой модели приходится задавать заново; было бы удобнее иметь прямой доступ к заданию и корректировке установок по умолчанию.

8. Отсутствуют средства подбора параметров теоретических вероятностных распределений по заданным моментам, а также формулы для моментов логарифмического распределения Лапласа, логарифмически логистического, распределений крайних значений и обратного гауссова. Нет непосредственных возможностей генерации коррелированных случайных величин.

9. Использование кириллических символов даже в комментариях исключает правильную работу Имитации.

10. Невозможно непосредственно определить вектор. Нижняя граница индексов матрицы по любому измерению равна 1, что может нарушить естественность индексации (например, при расчете вероятностей состояний системы). Крайне неудобен просмотр матриц через динамическое окно: одновременно можно видеть только два столбца. Ненулевая инициализация матрицы требует отдельного оператора *INITIAL* на каждый элемент.

11. Отсутствует возможность регулировать темп автоматического моделирования; он оказывается слишком быстрым, чтобы разглядеть нужные эффекты, а пошаговое продвижение транзактов — чересчур медленным.

12. Иконки свернутых динамических окон неразличимы, и при их восстановлении трудно выбрать желаемое.

13. Нельзя менять тип шкал графиков (только линейные) и их разметку, цвет и структуру линий, что может сделать последние неотличимыми друг от друга и/или от фона при черно-белом выводе.

14. Аргументом графиков может быть только время, так что распределение вероятностей состояний системы автоматически построить нельзя. К стати, для такого графика потребовалась бы логарифмическая шкала.

15. При работе с *ANOVA* нельзя менять расчетный уровень значимости.

16. Оставляет желать лучшего электронная документация к системе. При вызове *Help*'а по ключевому слову пользователю представляется «слепое» множество альтернатив, так что приходится выполнять их перебор. Часто встречаются пережитки предыдущих версий (*GPSS/PC*). Отсутствует упоминание о наличии в системе операции отрицания 'NOT'. Заявление о числовом значении «his value has a magnitude limited to 306 decimal digits» можно понять как длину мантииссы, хотя на самом деле речь идет о модуле десятичного порядка.

Ничего (кроме факта существования) не сказано о библиотеке. В разделе определения табличных функций понятие обратной функции заменено *кумулянтной*. В заголовке определения команды MATRIX неверно указано число операндов, не выделен строкой и буквой неиспользуемый первый операнд. Этот перечень можно было бы продолжить.

17. «Универсальность» интерпретатора GPSS оказывается ограниченной. Так, обслуживание с динамическим приоритетом (линейный рост приоритетов по времени ожидания с угловыми коэффициентами, зависящими от типа заявки), в GPSS даже с помощью мощного аппарата цепей пользователя обеспечить не удастся. Дело в том, что цепь может быть переупорядочена произвольно только в момент входа в нее новой заявки, тогда как нам для реализации динамического приоритета необходимо переупорядочение в момент выборки. На такой момент максимум возможного — это инверсия, т. е. реализация дисциплины LIFO (Last In — First Out). Предложенная в книге Шрайбера ([9], с. 391) схема «динамического приоритета» на самом деле реализует другую дисциплину выбора из очереди: по возрастанию времени обработки.

Трудности возникают также при моделировании «кругового» обслуживания. Для организации обслуживания всех заявок, накопившихся к началу обслуживания данной очереди, необходима переменная, запоминающая начало обслуживания текущей очереди. Ее значение наряду со значением указателя Pointer учитывается по схеме логического умножения при допуске к серверу очередной заявки. Не пропущенные заявки приходится записывать в цепь пользователя. При завершении обслуживания текущей очереди для каждого источника анализируется суммарное число заявок в очереди и в цепи. Однако после выбора нового положения стрелки блок UNLINK требует указания имени блока, в который адресуются освобождаемые заявки. Это имя в нашей модели можно получить с помощью функции Catenate присоединением к стандартному префиксу Inc номера выбранного источника. Но по синтаксису GPSS имя блока не может быть строкой, и моделирование завершается аварийно. Безуспешными оказались и попытки использовать числовые обозначения блоков с помощью команды EQU (было получено сообщение «You can assign your own value to entity label except for block labels»).

Трудности возникают и при моделировании замкнутой системы, в которой средний интервал между смежными заявками зависит от числа заявок в ней. Отсутствие в GPSS системного атрибута, представляющего это число, обходится введением соответствующей сохраняемой величины, на которую можно сослаться в блоке GENERATE. Проблема состоит в том, что к моменту изменения числа заявок в системе в цепь будущих событий уже занесен сформированный по прежней интенсивности момент прибытия очередной заявки. Доступа же к содержимому этой цепи программист не

имеет. Значит, точное моделирование замкнутых СМО в GPSS с помощью одного GENERATE невозможно. Остается громоздкий вариант с отдельными сегментами генерации заявок по максимальному числу активных источников и циркуляцией в каждом сегменте единственной заявки.

Кстати, заметим, что Дж. Шрайбер скептически оценивает полезность цепей пользователя и в особенности их индикаторов связи ([9], с. 385). В его книге объемом в 70 авторских листов приведены лишь два не очень убедительных примера работы с цепями пользователя, а в [7] — ни одного.

18. Оптимизацию систем обслуживания иногда приходится проводить по дискретным аргументам. Выбор решения часто осуществляется из дискретных рядов устройств с фиксированными характеристиками (быстродействием, емкостью памяти, набором алгоритмов функционирования и т. п.) и состоит в назначении целого числа устройств или расширяющих блоков. Это обстоятельство определяет необходимость использования при оптимизации методов дискретного программирования вплоть до элементарного перебора сначала на редкой решетке, затем — на более частой в перспективной окрестности оптимума.

С другой стороны, при определении тенденций оптимизации часто помогает рассмотрение компонент целевой функции. Например, в задачах управления запасами [4] минимизируется сумма затрат на хранение избыточного (среднего) запаса, организацию поставок и «штрафы» за неполное или несвоевременное обеспечение спроса. Большой удельный вес одной из упомянутых составляющих указывает на необходимость соответственно уменьшения уровня заказа, предельного уровня запаса или объема заказываемой партии; увеличения максимального запаса или объема заказываемой партии, увеличения порога заказа. Оптимизирующие эксперименты GPSS/W такой возможности не предусматривают. Ясно, что никакая «универсальная» система подобную специфику учесть не может.

В общем, работа на GPSS и других подобных языках безусловно оправдана при массовом создании моделей средней сложности. Эпизодическую разработку моделей умеренной сложности и создание проекта уникальной сложности программист с опытом работы на любом языке высокого уровня может вести на этом языке, причем знакомство с основными понятиями GPSS подскажет ему ряд полезных технических приемов. Наконец, всегда остается желательным расчет упрощенных вариантов исследуемых моделей с помощью численных методов теории очередей и реализующих их пакетов прикладных программ: он необходим хотя бы для получения эталонных значений, требуемых при отладке имитационных моделей.

Таким образом, практически необходимо использовать все три упомянутых подхода. Автор подготовил монографию по численным методам и учебник — по имитационным и надеется опубликовать их в ближайшем будущем.

Литература

1. **Рыжиков Ю. И.** Машинные методы расчета систем массового обслуживания: Учеб. пособие. — Л.: ВИКИ им. А. Ф. Можайского, 1979. — 177 с.
2. **Рыжиков Ю. И.** Эффективность и эксплуатация программного обеспечения ЭЦВМ: Учеб. пособие. — МО СССР, 1985. — 263 с.
3. **Рыжиков Ю. И.** Имитационное моделирование систем массового обслуживания: Учеб. пособие. — Л.: ВИКИ им. А. Ф. Можайского, 1991. — 111 с.
4. **Рыжиков Ю. И.** Теория очередей и управление запасами: Учеб. пособие. — СПб.: Питер, 2001. — 376 с.
5. **Киндлер Е.** Языки моделирования/Пер. с чешского. — М.: Энергоатомиздат, 1985. — 288 с.
6. **Советов Б. Я., Яковлев С. А.** Моделирование систем. — М.: Высшая школа, 1998. — 320 с.
7. **Советов Б. Я., Яковлев С. А.** Моделирование систем: практикум. — М.: Высшая школа, 1999. — 224 с.
8. **Харин Ю. С.** и др. Имитационное и статистическое моделирование: Учеб. пособие. — Минск: Белорусский гос. ун-т, 1992. — 176 с.
9. **Шрайбер Т. Дж.** Моделирование на GPSS/Пер. с англ. — М.: Машиностроение, 1980. — 592 с.
10. **Gordon G.** The Application of GPSS V to Discrete System Simulation. — Prentice-Hall, Englewood Cliffs, N. J. — 1975. — 389 p.
11. **Bobiller P. A., Kahan B. S., Probst A. R.** Simulation with GPSS and GPSS V. — Prentice-Hall, Englewood Cliffs, N. J., 1976. — 495 p.
12. **Янбых Г. Ф., Столяров Б. А.** Оптимизация информационно-вычислительных сетей. — М.: Радио и связь, 1987. — 232 с.
13. **Добров Г. М.** Наука о науке. — Киев: Наукова думка, 1966. — 270 с.

МИКРОКОНТРОЛЛЕРЫ

со склада в Санкт-Петербурге

Широкий спектр микроконтроллеров для различных приложений, отладочные средства, софт со склада в Санкт-Петербурге. Поставки микроконтроллеров на основе ядра 8051, RISC, ARM. DSP процессоры, микроконтроллеры со встроенными АЦП, ЦАП. Импортные электронные компоненты двойного назначения.

UBICOM, CYGNAL, FUJITSU, GOAL, ATMEL, MOTOROLA, AMD



SX20AC, SX28AC, SX52BD, IP2022-160

ЭЛЕКТРОСНАБ

Официальный представитель
UBICOM, CYGNAL, GOAL в России

С-Петербург, наб. реки Фонтанки, д.38, тел./ф.: (812) 380 16 60
info@electrosnab.ru www.electrosnab.ru

