

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ С ЗАКОЛЬЦОВАННОЙ ЦЕПЬЮ СОБЫТИЙ***Ю. И. Рыжиков (Санкт-Петербург)****Постановка задачи**

Проблема расчета сетей обслуживания является одной из наиболее актуальных в задачах организации производства, здравоохранения, работы аварийных служб и групп быстрого реагирования, технике связи и др. Расчеты систем и сетей обслуживания должны выполняться как на этапе проектирования соответствующих систем, так и в ходе их эксплуатации. В последнем случае необходимость расчетов может породиться изменением:

- объема нагрузки вследствие подключения дополнительных пользователей; характера нагрузки (появление новых задач, смена алгоритмов или технологии решения старых);
- организации работы (введение приоритетов, специализация каналов и узлов, перераспределение потоков в сети, смена операционной системы вычислительной машины);
- конфигурации аппаратной части (вывод в ремонт или на длительную профилактику части устройств, подключение дополнительных, модернизация или замена устаревшей техники);
- требований к качеству функционирования системы (ужесточение их, специализация по видам заявок, предъявление дополнительных требований и т. п.).

Эти расчеты в сущности являются вычислительным экспериментом над математической моделью системы (сети) обслуживания.

К сожалению, строгий расчет сетей возможен лишь при крайне стесненных условиях известной теоремы ВСМР, делающих разработанные в ее рамках методы практически бесполезными. Достаточно сослаться хотя бы на то, что она не допускает наличия канальных узлов с немарковскими (отличными от показательного) распределениями длительности обслуживания. Это обстоятельство определяет большой размах исследований по имитационному моделированию сетей преимущественно с помощью систем автоматизации моделирования (в отечественных условиях главным образом GPSS World [1, 2] и AnyLogic).

Общими недостатками систем подобного типа является скрытая от программиста встроенная логика интерпретатора, которая не может не накладывать ограничений на класс решаемых задач (см. в работе [2] далеко не полный перечень недостатков системы, выявленных при тестировании GPSS World). Кроме того, претензии на универсальность интерпретатора чрезмерно его утяжеляют и в десятки раз замедляют моделирование [2].

Упомянутые недостатки принципиально присущи любым системам автоматизированного моделирования. Отмеченные обстоятельства определяют растущий интерес к составлению программ прямого моделирования сетей на универсальных языках программирования вычислений, безусловным лидером среди которых является современный Фортран [3, 4]. *Только этот подход обеспечивает понимание логики моделирования, что делает его применение в образовательных целях незаменимым.*

Взросшее быстродействие современных ЭВМ внушает пользователям надежду достичь высокой точности результатов (иллюзорной из-за неидеальной работы датчи-

* Работа поддержана программой фундаментальных исследований ОНИТ РАН и грантом РФФИ 10-08-00906-а.

ков случайных чисел) заданием очень большого числа испытаний, измеряемого миллионами. При «лобовом» программировании цепь будущих событий (ЦБС) должна вместить соответствующее число событий, что может оказаться неэффективным или вообще недопустимым из-за ограниченного объема оперативной памяти.

Идея кольцевой ЦБС

Сеть обслуживания состоит из M рабочих узлов с n_i каналами обслуживания в каждом, $i = \overline{1, M}$, и двух вспомогательных – нулевого (источник) и $(M+1)$ -го – сток. Вероятности перемещения заявки задаются матрицей передач $\{r_{i,j}\}$, $i, j = \overline{1, M+1}$. Новая заявка рождается в источнике и адресуется в один из рабочих узлов с вероятностями $\{r_{0,j}\}$. При всех перемещениях заявку сопровождает отметка времени ее прибытия в сеть. Заявка, попавшая в сток, считается покинувшей сеть. Техника моделирования подобных процессов обычно реализуется схемой «от события к событию», которыми являются ближайшее прибытие новой заявки или завершение очередного обслуживания. Выявление таковых требует выбора в каждом из узлов ближайшего (по задействованным каналам) момента завершения обслуживания и минимального из них – по всем рабочим узлам сети. Эти действия весьма трудоемки и имеют сложную логику. Целесообразно построить ЦБС, содержащую моменты наступления запланированных событий и указания на тип события, которые обеспечат вызов специализированной подпрограммы. Именно такой подход реализован в GPSS.

Элемент ЦБС должен содержать разнотипную информацию: время завершения обслуживания, номер узла, номер канала и момент входа заявки в сеть. Следовательно, он должен быть представлен в виде *записи*. Ниже для конкретности обсуждаются фрагменты программы моделирования *замкнутой* сети обслуживания, в которой циркулирует фиксированное количество заявок K . Целью моделирования является оценка моментов распределения времени пребывания заявки в сети. Поскольку событиями в такой сети всегда являются моменты завершения обслуживания в одном из каналов, ЦБС должна содержать $N = \sum_i n_i$ записей, ключевым элементом которых являются моменты завершения обслуживания в данном канале. Для свободных каналов они принимаются равными «машинной бесконечности», например, 10^{30} . Это обстоятельство используется для выбора свободного канала при приеме на обслуживание новой заявки. Кроме того, в каждой записи должны содержаться:

- момент поступления заявки в сеть (для сопоставления в дальнейшем с моментом выхода из сети);
- номер узла сети (предполагается, что с каждым узлом связаны свое распределение трудоемкости обслуживания и количество каналов);
- номер канала в узле (для конкретизации объекта занятия или освобождения).

В динамике имитации позиция в ЦБС ближайшего события при завершении обслуживания в текущем узле, выборе следующей заявки из текущей же очереди, прибытии новой заявки извне и ее приеме на обслуживание будет постоянно перемещаться. Поскольку трудоемкость переупорядочения ЦБС пропорциональна *квадрату* числа ее элементов, имеет смысл *заменить переупорядочение выбором записи с минимальным ключевым элементом*. Предельное количество заявок в узле естественно ограничено численностью популяции K . Для простоты перенесем это ограничение на длину очереди. Позиции очереди содержат моменты прибытия в сеть ожидающих заявок и нумеруются от нуля до $K-1$. Если очередь не пуста, то при освобождении канала для голов-

ной заявки очереди генерируется время завершения ее обслуживания. Соответственно обновляется текущая запись ЦБС.

Обычного для практики имитации трудоемкого продвижения очереди можно избежать, если использовать для очередей *кольцевую адресацию*. В этом случае достаточно добавить по модулю K единицу к указателю на голову очереди. При добавлении к очереди новой заявки аналогично пересчитывается координата хвоста очереди.

Элементы Фортрана 90

Читатель даже со скромным опытом программирования без труда разберется в приводимых ниже фрагментах программы. Отметим следующие особенности современного Фортрана:

1. Комментарий начинается восклицательным знаком.
2. Включаемая в ЦБС запись определяется как объект типа event с указанными типами значений полей. При ссылке на запись имя поля указывается через знак процента. Возможно объявление массива записей. При ссылке номер элемента указывается (в скобках) сразу за именем массива.
3. Для конформных массивов разрешены групповые покомпонентные операции. Скаляр считается конформным любому массиву, что позволяет, в частности, одним оператором обнулить любой массив.
4. Вспомогательные внутренние процедуры включаются в конец охватывающей процедуры после ключевого слова contains. Объекты охватывающей процедуры для них (если имена не переопределяются) являются глобальными.

Фрагменты программы моделирования

Прежде всего приведем пример *описания типа* для элементов ЦБС:

```
type event
  real    serv, enter
  integer node, channel
end type event
```

Соответственно объявление единичной рабочей записи будет иметь вид

```
type(event) new
```

Цепь будущих событий future объявляется *динамическим массивом* завершений обслуживания:

```
type(event) future [allocatable](:)
```

Затем оператором allocate для нее в соответствии с общим числом каналов в сети выделяется память. После случайного распределения расчетной популяции K по узлам сети производится начальное заполнение ЦБС:

```
L=1                                ! Начальный элемент ЦБС
do i=1,m                            ! По узлам
  do j=1,n(i)                       ! Перебор каналов в узле
    future(L)%node = i              ! узел
    future(L)%channel= j           ! канал
    if (j<=z(i)) then
      future(L)%enter = 0          ! Время входа в сеть
      future(L)%serv = flib(i)     ! завершение обслуживания
    else
      future(L)%enter = 1e30
      future(L)%serv = 1e30
    end if
    L=L+1
    if (L>nn) goto 55              ! Выход
  end do
end do
55
```

Затем отыскивается элемент ЦБС *thead*, соответствующий самому раннему обслуживанию, и формируются начальные очереди в узлах: определяются их длины и указатели на голову и хвост каждой очереди. Как пример динамики модели опишем обработку завершения обслуживания в узле *L*:

```

z(L)=z(L)-1           ! Останется заявок в узле L
new%node=L
new%channel=j
if (z(L)<n(L)) then
  new%enter=1e30
  new%serv= 1e30      ! Освобождение канала!
  new%channel=0
else
  new%enter=q(L,hq(L)) ! Извлечение заявки из очереди
  new%serv =tt+flib(L) ! Постановка ее на обслуживание
  hq(L)=mod(hq(L)+1, kk) ! Сдвиг головы очереди в L-узле
end if
future(thead)=new    ! Обновление в обслужившем канале

```

Формируемый здесь рабочий элемент *new* заменяет собой прежнюю голову ЦБС. Далее в соответствии с матрицей передач производится выбор узла – приемника *L*-го.

Если выбирается сток сети, то заявка покидает сеть и накапливается информация для подсчета моментов распределения времени пребывания в сети. Вместо ушедшей заявки генерируется новая, определяются принимающие ее узел и (если есть свободные) канал, формируется соответствующий элемент ЦБС. Аналогично (но без генерации) обстоит дело при выборе одного из рабочих узлов. Новый элемент вставляется в соответствующую позицию ЦБС с кольцевым сдвигом хвостовой ее части, причем поиск места вставки начинается с хвоста.

Результаты расчета

Приведенная процедура была применена для моделирования сети обслуживания из четырех рабочих узлов с числом однородных каналов {2,1,1,3} соответственно. Средние длительности обслуживания по узлам составляли {2,3,3,5}, распределение его предполагалось эрланговым 3-го порядка. Моделировалось обслуживание миллиона заявок для популяции в 40 заявок. В таблице приведены результаты имитации по методике данной статьи. Они практически совпали с результатами прогона при тех же исходных данных «наивной» модели, не использовавшей кольцевую ЦБС.

Моменты распределения времени пребывания заявки в сети

Метод	Время счета, с	Порядок момента		
		1	2	3
«Наивный»	3,28	8,3836e1	1,2318e4	2,2781e6
Предлагаемый	2,67	8,3792e1	1,2310e4	2,2766e6

Заключение

Практическое совпадение результатов прогона «наивной» и усовершенствованной моделей, существенно различающихся структурно, позволяет считать их программную реализацию правильной и ниже следующие выводы – обоснованными. Трудоемкость прогона модели уменьшилась почти на 20%, что при столь большом числе испытаний свидетельствует о значимой эффективности предложенных приемов. Следует полагать, что выигрыш при моделировании более сложных сетей будет возрастать.

Эти приемы легко реализуются и для других вариантов сетей обслуживания, в частности, включением в ЦБС всех дополнительных генераторов событий с дополнительной ссылкой на обработку конкретных типов событий. Для *разомкнутой* сети, например, это будут внешние источники заявок каждого типа.

Освоение предлагаемой технологии будет способствовать росту программистской квалификации «модельеров», уменьшению их зависимости от разработчиков инструментальных систем моделирования и лучшему пониманию базовых идей имитации.

Литература

1. **Кудрявцев Е. М.** GPSS World: основы имитационного моделирования различных систем. М.: ДМК Пресс, 2004. 320 с.
2. **Рыжиков Ю. И.** Имитационное моделирование. Теория и технологии. СПб.: КОРОНА-принт, 2004. 380 с.
3. **Бартенев О. В.** Современный Фортран. М.: Диалог-МИФИ, 1998. 397 с.
4. **Немнюгин М. А., Стесик О. Л.** Современный Фортран: Самоучитель. СПб.: БХВ-Петербург, 2004. 496 с.