

УПРАВЛЕНИЕ ОГРАНИЧЕННЫМИ РЕСУРСАМИ В ОБОБЩЕННЫХ СТОХАСТИЧЕСКИХ СЕТЕВЫХ ГРАФИКАХ

Иванов Н. Н.¹

(ФГБУН Институт проблем управления
им. В.А. Трапезникова РАН, Москва)

Рассматривается моделирование процесса управления в реальном времени вычислительной системой с ограниченным числом вычислительных ресурсов. В качестве средства моделирования используется сетевой график. Ограничение числа ресурсов может приводить к образованию очереди готовых к выполнению работ. В данной статье работа ассоциируется с соответствующей дугой сетевого графика. Предложены методы управления ресурсами вычислительной системы для различных дисциплин выполнения работ, находящихся в буфере. В частности, рассмотрены дисциплины FIFO и LIFO, а также одна из эвристических дисциплин. Основой для анализа этих методов служит понятие состояния сетевого графика. Составляющими состояний при этом считаются множество исполняемых работ и множество работ, готовых к исполнению и находящихся в буфере. Состояния сетевого графика образуют конечное дерево состояний, которое служит основой построения имитационных программ. По этим программам пользователь может производить сравнительный анализ рассмотренных алгоритмов управления ресурсами вычислительной системы. Использование предлагаемых алгоритмов моделирования позволяет также пользователю решать задачу об оптимальном выборе числа исполнителей. Решение этой задачи позволяет найти приемлемый компромисс между этим числом и временем выполнения программы в реальном времени.

Ключевые слова: обобщенный стохастический сетевой график, дерево состояний сетевого графика, буферизация работ, дисциплины FIFO и LIFO, имитационное моделирование.

1. Введение

В работе рассматриваются сетевые графики в предположении о том, что времена прохождения отдельных дуг являются случайными временами с известными распределениями [1]. Однако в отличие от традиционных сетевых графиков помимо вершин типа «И» в рассматриваемых графиках допускаются вершины типа «исключенное ИЛИ» с блокировкой дуг [4]. Эти

¹ Николай Николаевич Иванов, д.т.н., доцент (ivanov.nni@yandex.ru).

графики в работах автора получили название обобщенных стохастических сетевых графиков (ОССГ) [3].

Существенной особенностью рассматриваемых в данной работе ОССГ является то, что в них число исполнителей, осуществляющих прохождение дуг, может быть меньше степени параллелизма ОССГ [4], что может повлечь за собой наличие буфера готовых к прохождению дуг. Как следствие, это может приводить к необходимости назначать дисциплину выбора на прохождение буферизированных дуг и проводить сравнительный анализ эффективности различных дисциплин с целью выбора наиболее соответствующей используемым критериям качества (как правило, времени) выполнения рассматриваемого ОССГ.

Применительно к вычислительным процессам в управляющих параллельных вычислительных системах (ВС) ОССГ могут рассматриваться как инструмент обеспечения их временной надежности. Это понятие предполагает, что для заданных распределений времени выполнения отдельных работ предварительно может быть смоделировано поведение ОССГ и на его основе сделан вывод о возможности выполнения управляющей программы не более чем за заданное директивное время.

В работах коллектива сотрудников во главе с В.В. Игнатушенко, которые проводились с середины 90-х годов прошлого столетия, рассматривалась подобная задача [5]. Однако ее решение было ограничено предположением об экспоненциальном распределении времен прохождения дуг и способом вычисления времени выполнения сетевого графика с помощью его функции распределения. Отметим также, что во всех этих работах рассматривались сетевые графики классического типа без вершин типа «ИЛИ», что сужало круг решаемых задач.

В работах [5–7] анализ сетевых графиков осуществлялся на основе построения графа состояний [5]. При экспоненциальном распределении времен прохождения дуг этот граф превращался в граф обрывающегося марковского процесса с одним начальным и одним конечным состояниями. Это давало возможность находить функцию распределения времени его выполнения известными методами [5].

В данной работе исходным также является построение для ОССГ (с вершинами обоих типов) графа состояний, представля-

емого в виде дерева [4]. Это потребовало обоснование возможности использования понятия состояние в том виде, в котором оно рассматривалось в [5–7], применительно к ОССГ с ограниченными ресурсами (это обоснование содержится в приложении). В отличие от подхода, принятого в [5–7], последующий анализ проводится для произвольных распределений времени прохождения дуг имитационными методами. Кроме того, существует алгоритмическая возможность строить имитационные программы без построения дерева состояний, а лишь для траекторий, возникающих для конкретных наборов случайных времен прохождения дуг в виртуальном дереве состояний. При этом используется то обстоятельство, что каждому набору случайных времен прохождения дуг в дереве состояний соответствует единственная траектория.

Алгоритм построения дерева состояний учитывает тип проходимых вершин и рассматриваемые дисциплины выбора дуг из буфера. Эти дисциплины могут основываться только на апостериорных знаниях о процессе выполнения сетевого графика. Такими свойствами обладают рассматриваемые ниже алгоритмы управления буфером LIFO, FIFO и алгоритм с использованием заранее назначенных приоритетов дуг.

2. Алгоритмы управления, основанные на различных дисциплинах выбора дуги из буфера

2.1. АЛГОРИТМ НА ОСНОВЕ ДИСЦИПЛИНЫ LIFO

В результате окончания прохождения некоторой дуги в дереве состояний могут возникнуть две возможности: либо свершается некоторое событие, и тогда исходящие из вершины, соответствующей данному событию, дуги получают возможность активации, попадая в буфер, либо в противном случае освобождается один исполнитель, и для дуги, находящейся в буфере (если он не пуст) и попавшей в буфер последней, начинается процесс прохождения. В первом случае из буфера также выбирается дуга, попавшая в него последней. Если таковых дуг больше одной, то выбор дуги может осуществляться в соответствии с выбранным критерием отбора (например, в соответствии с заранее заданными приоритетами). Таким образом, процесс прохожде-

ния состояния на дереве состояний может осуществляться как бы в два этапа за нулевое время. Буфер при этом представляет собой стек.

2.2. АЛГОРИТМ НА ОСНОВЕ ДИСЦИПЛИНЫ FIFO

Алгоритм отличается от предыдущего лишь тем, что из буфера на прохождение выбирается дуга, попавшая в него по времени раньше всех остальных. При этом в случае, если таковых более одной, то выбор, как и выше, происходит в соответствии с назначенным критерием отбора. Таким образом, использование данного алгоритма предполагает фиксацию времени поступления дуг в буфер, отсчитываемого с начала выполнения ОССГ. Наиболее просто это решается соответствующим порядком размещения дуг в буфере: пришедшие раньше размещаются левее пришедших позже при графическом представлении дерева состояний.

2.3. АЛГОРИТМ НА ОСНОВЕ ПРИОРИТЕТОВ ДУГ

Выбор дуги может происходить на основе заранее назначенных абсолютных приоритетов всех дуг ОССГ. Таким образом, при наличии в буфере n дуг и k свободных исполнителей на прохождение будут выбраны m дуг, $m = \min\{k, n\}$, с наивысшими приоритетами.

Назначенные приоритеты вводят на множестве дуг линейный порядок φ , при этом ОССГ на множестве дуг определяет частичный порядок ψ ($u\psi v$, если существует цепочка π , возможно, пустая, направленных дуг такая, что конец дуги u совпадает с началом цепочки π , а конец цепочки π совпадает с началом v). В работе принято, что должно выполняться следующее правило: если две дуги u и v находятся в отношении ψ ($u\psi v$), то приоритет u должен быть выше приоритета v ($u\varphi v$). Коротко: $u\psi v \Rightarrow u\varphi v$.

Алгоритм на основе приоритетов использовался ранее в работах [5–7]. Приоритеты назначались исходя из расположения вершин сетевого графика, соответствующих некоторым работам, при графическом представлении сетевого графика. При этом более высоким приоритетом обладали вершины, находящиеся выше. Вершины, находящиеся в одном горизонтальном

уровне графика и расположенные левее, наделялись более высоким приоритетом относительно вершин, расположенных правее.

3. Имитационное моделирование алгоритмов управления ресурсами

Предпочтение, которое может отдать пользователь тому или иному алгоритму управления процессом выполнения ОССГ с ограниченными ресурсами, может быть основано на их имитационном моделировании. В рассмотренных выше случаях выбора дуг из буфера для этих целей может быть использовано дерево состояний ОССГ, вид которого зависит от выбранного алгоритма управления ресурсами и определяется общим количеством имеющихся исполнителей и дисциплиной свершения событий, сопоставленных вершинам ОССГ.

Рассматриваемый в качестве примера ОССГ (рис. 1), но с неограниченным числом исполнителей, как это показано в [4], имеет степень параллелизма, равную 4. Там же показано, что при выбранных нами распределениях времен прохождения дуг эта степень параллелизма достижима. Таким образом, для нашей задачи с двумя исполнителями размер буфера достаточно выбрать равным двум.

В качестве дерева состояний для ОССГ, показанного на рис. 1, приведено дерево состояний на рис. 2. В этом ОССГ вершины 2 и 4 имеют тип «ИЛИ» с блокированием дуг [4], а вершины 3 и 5 – тип «И». Выполнение ОССГ происходит с помощью двух исполнителей по дисциплине FIFO. В связи с этим в каждом состоянии дерева состояний присутствуют две компоненты: первая (левая) содержит номера дуг, проходимых в текущий момент времени, их число не может превосходить в данном примере двух; вторая компонента (правая) состоит из номеров дуг, образующих буфер (номера дуг в буфере обозначены жирными цифрами и расположены слева направо в порядке попадания в буфер).

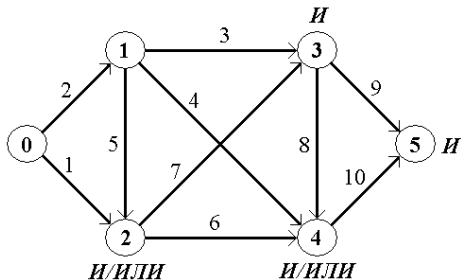


Рис. 1. ОССГ

Номера дуг в ОССГ, приведенном на рис. 1, одновременно являются их абсолютными приоритетами, в соответствии с которыми дуги последовательно размещаются в буфере в строку с возрастанием их номеров слева направо. При наличии в буфере числа дуг, большего чем число свободных исполнителей, выбор дуг на прохождение может происходить, например, по их абсолютным приоритетам (из состояния (1 3 4 5) по дуге 1 выбор на прохождение приобретает дуга 4).

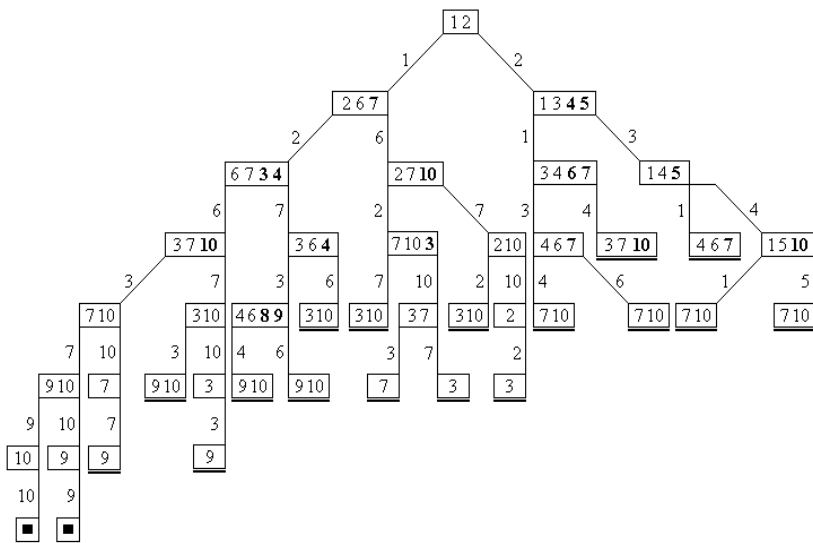


Рис. 2. Дерево состояний ОССГ по рис. 1 (FIFO)

В этом ОССГ при использовании дисциплины LIFO в дереве состояний произойдут существенные изменения, что видно на рис. 3.

Заметим, что на рис. 3 присутствуют состояния (3 6 4 7) и (3 6 7 4), которые различаются порядком прихода дуг 4 и 7 в буфер. По этой причине их продолжения в дереве не совпадают и должны быть построены исходя из того, что при выбранной дисциплине LIFO в последователях этих вершин будут назначены на прохождение либо дуга 4, либо дуга 7.

Использование алгоритма, описанного в разделе 2.3 в соответствии с приоритетами буферизированных дуг, приведет к построению дерева состояний, какое показано на рис. 4.

Дерево состояний для вершин 2 и 4 типа «И» с назначением дуг из буфера на прохождение в соответствии с их приоритетами, представлено на рис. 5.

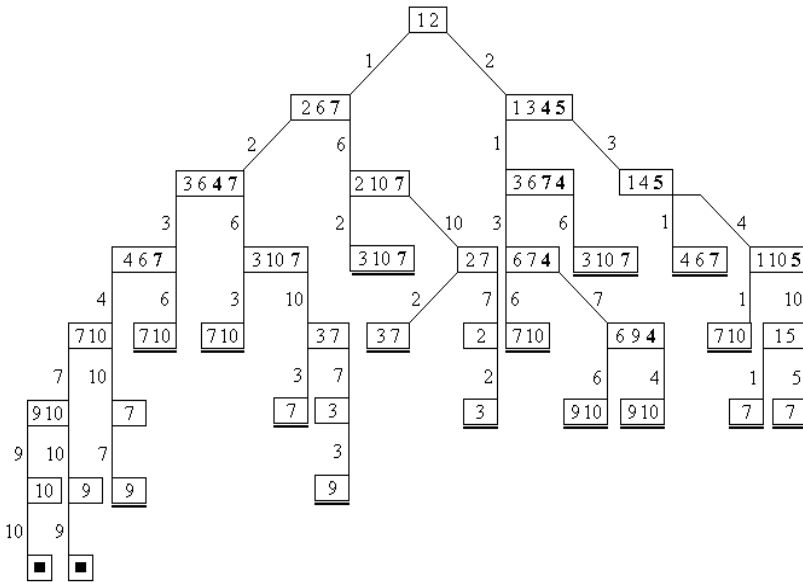


Рис. 3. Дерево состояний ОССГ по рис. 1 (LIFO)

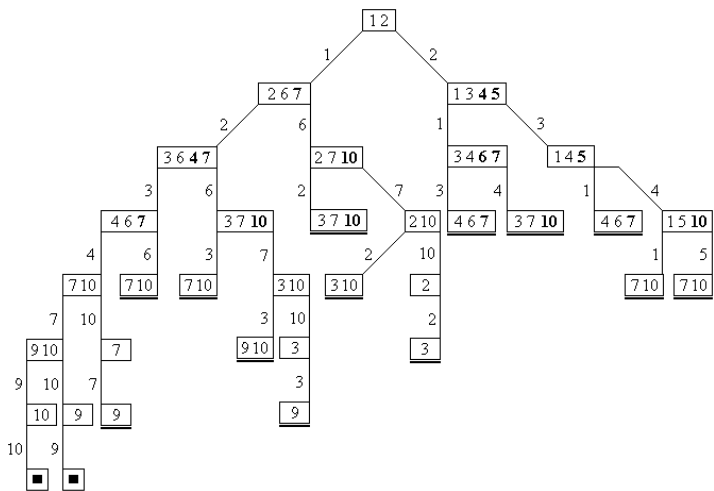


Рис. 4. Дерево состояний ОССТ по рис. 1 (приоритет)

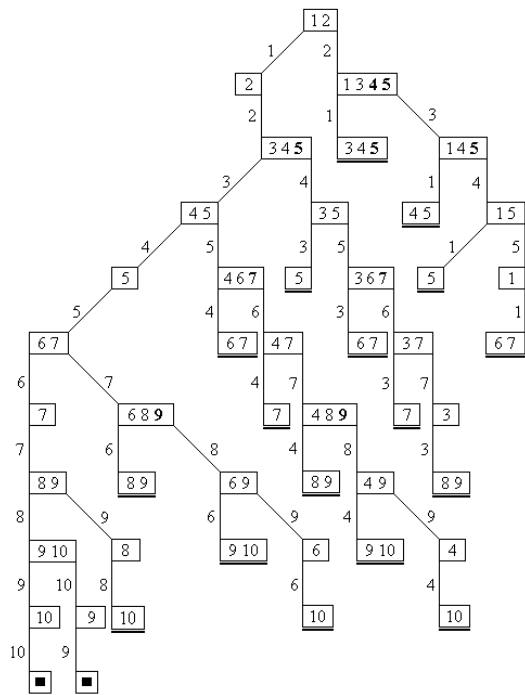


Рис. 5. Дерево состояний ОССТ по рис. 1 («И», приоритет)

Деревья состояний, вид которых не зависит от набора случайных времен прохождения дуг, могут быть названы инвариантными по отношению к этим наборам, Примерами таких инвариантных деревьев являются деревья, представленные на рис. 2–5. Однако существуют алгоритмы управления ресурсами, которым нельзя поставить в соответствие инвариантное дерево состояний. Например, невозможно построение инвариантного дерева состояний (в рамках принятого в работе определения дерева состояний) в случае, если выбор дуги производится случайным равновероятным образом.

Для алгоритмов управления ресурсами, для которых невозможно построение инвариантного дерева состояний, имитационное моделирование возможно лишь с помощью построения траекторий в виртуальном дереве состояний, соответствующем каждому случайному набору времен прохождения дуг и алгоритму управления ресурсами. Такими алгоритмами могут быть, например, алгоритм случайного равновероятного выбора дуги из буфера или какие-либо эвристические алгоритмы, основанные на апостериорной информации о буферизованных дугах.

Испытаниям подвергался ОССГ при двух исполнителях с вершинами 2 и 4 следующих типов: 1) обе вершины типа «ИЛИ», FIFO, 2) обе вершины типа «ИЛИ», LIFO, 3) обе вершины типа «ИЛИ», приоритеты дуг в соответствии с их номерами на рис. 1, 4) обе вершины типа «И», приоритеты дуг в соответствии с их номерами на рис. 1.

Вычисление оценок вероятностных параметров случайных процессов в ОССГ для всех четырех вариантов сочетаний типов вершин 2 и 4 производилось методом имитационного моделирования. Моделирующие программы строились по инвариантным деревьям состояний (рис. 2–5). Нормальные независимые распределения времен прохождения дуг (полученные по методике, содержащейся в [2]) и их обозначения принимались в соответствии с данными, приведенными ниже.

$$m_{01} = m_{12} = m_{23} = 8, \quad m_{02} = 5,5\sqrt{2}, \quad m_{13} = 6, \quad m_{14} = 6\sqrt{2}, \quad m_{24} = 10\sqrt{2}, \\ m_{34} = 7, \quad m_{35} = 3, \quad m_{45} = 9;$$

$$d_{01} = d_{14} = d_{02} = d_{12} = d_{13} = d_{23} = d_{24} = d_{34} = d_{35} = d_{45} = 1.$$

Заметим, что дерево состояний ОССГ, в котором все вершины типа «И», имеет высоту, равную числу дуг сетевого графика. Такую же высоту будет иметь дерево состояний при наличии вершин типа «ИЛИ» без блокирования входных для них дуг [4]. При наличии вершин типа «ИЛИ» с блокированием дуг высота дерева состояний станет меньше на величину, равную суммарному числу входных дуг в вершины типа «ИЛИ» за вычетом числа таких вершин. Что касается ширины (полного) дерева состояний в его нижней части, то она равна общему числу траекторий.

Результаты статистических испытаний четырех приведенных выше вариантов ОССГ, а также стандартного варианта с вершинами 2 и 4 типа «И» без ограничения ресурсов [8], приведены в таблице 1. В таблице приняты следующие обозначения: t – оценка среднего времени выполнения ОССГ, σ – среднеквадратическое отклонение этого времени, Δ – ширина интервала доверия при уровне значимости $\alpha = 0,05$ и числе испытаний 10^5 .

Из таблицы 1 видно, что выбор дисциплины назначения на прохождение дуг из буфера влияет на вероятностные характеристики моделируемого ОССГ. Таким образом, пользователь, располагая предложенным инструментом, может подбирать путем сравнительных статистических испытаний наиболее оптимальную дисциплину обслуживания буфера. Таких дисциплин может быть гораздо больше, нежели приведено в статье в качестве примеров применения имитационных средств для оценки эффективности этих дисциплин.

Таблица 1.

Параметр	Стандарт	Варианты ОССГ			
		1	2	3	4
t	40,0	30,92	32,92	32,92	46
σ	0,11	0,11	0,108	0,146	0,121
Δ	$1,36 \cdot 10^{-3}$	$1,4 \cdot 10^{-3}$	$1,34 \cdot 10^{-3}$	$1,8 \cdot 10^{-3}$	$1,5 \cdot 10^{-3}$

При сравнении столбцов «Стандарт» и 4 таблицы 1 можно отметить, что уменьшение вдвое по сравнению со степенью па-

раллелизма числа исполнителей увеличивает время выполнения ОССГ лишь на 15%. Следовательно, данный инструмент может быть использован также при решении вопроса об оптимальном выборе числа исполнителей, при котором достигается приемлемый компромисс между этим числом и временем выполнения ОССГ.

Помимо приведенных в таблице 1 вероятностных характеристик времени выполнения сетевого графика имитационное моделирование дает возможность находить интересные для пользователя величины, такие как максимальное время выполнения графика при используемом числе прогонов, относительную частоту выполнения графика за время, не превышающее заданное директивное время, оценку числа используемых исполнителей, медиану статистического ряда времен выполнения сетевого графика и т.п.

3. Заключение

Рассмотрена методика имитационного моделирования с помощью обобщенного стохастического сетевого графика процесса управления в реальном времени вычислительной системой с ограниченным числом вычислительных ресурсов. Методика имеет целью обеспечить пользователя возможностью выбирать оптимальную дисциплину обслуживания буфера готовых к прохождению дуг. Методика позволяет также решать задачу об оптимальном выборе числа исполнителей, при котором достигается приемлемый компромисс между этим числом и временем выполнения ОССГ.

Основой построения имитационной модели служит инвариантное или виртуальное дерево состояний ОССГ, вид которого зависит от выбранного алгоритма управления ресурсами и определяется общим количеством имеющихся исполнителей, а также (для неинвариантных деревьев) дисциплиной свершения событий, сопоставленных вершинам ОССГ.

Приложение

Описание эволюции ОССГ с ограниченными ресурсами в процессе его исполнения предполагает, что в этой модели используется введенное ранее понятие дерева состояний. Представляется важным выбрать такой набор параметров, который позволяет не избыточно и однозначно представлять состояние ОССГ в произвольный момент времени его выполнения.

Предварительно введем следующие обозначения:

П – множество дуг, не готовых к выполнению;

Б – множество дуг, находящихся в буфере;

И – множество дуг, проходимых в рассматриваемый момент времени;

Х – множество дуг, пройденных к рассматриваемому моменту времени (пройденными будут считаться также дуги, являющиеся входными для пройденных вершин типа «ИЛИ», см. ниже);

Е – множество всех дуг ОССГ.

Имеют место следующие соотношения между этими множествами:

1) множества П, Б, И и Х попарно не пересекаются,

2) $P \cup B \cup I \cup X = E$.

Под состоянием ОССГ в рассматриваемый момент времени будем понимать четверку, составленную из множеств П, Б, И, Х.

Примем во внимание следующие посылки:

а) знание множества Х позволяет по ОССГ однозначно найти $B \cup I$;

б) знание суммы $B \cup I \cup X$ позволяет однозначно определить $P = E \setminus (B \cup I \cup X)$.

Очевидно, что по каждому отдельно взятому множеству П, Б, И или Х не могут быть восстановлены остальные составляющие четверки. Таким образом, в качестве минимального набора, составленного из множеств П, Б, И и Х, может быть лишь набор, содержащий не менее двух из этих множеств.

Из перечисленных выше посылок следует, что пара {Х, И} является исчерпывающей для описания состояния ОССГ, по-

скольку по ней может быть однозначно определена четверка $\{\Pi, B, I, X\}$.

Однако такое не избыточное представление состояния ОССГ не единственное. Во-первых, как это следует из упомянутых посылок, таковой парой может служить также пара $\{X, B\}$. Во-вторых, покажем, что для однозначного и наименее объемного описания состояния ОССГ может быть применена пара $\{B, I\}$.

Конфигурация ОССГ, рассматриваемая ниже, будет предполагать, что из всех входных дуг каждого пройденного к рассматриваемому моменту времени состояния типа «ИЛИ» будет оставлено лишь одно, соответствующее наиболее раннему прохождению. Остальные дуги, входящие в это состояние, блокируются и удаляются из ОССГ.

Вопрос, возникающий из представления состояния ОССГ в виде пары $\{B, I\}$, заключается в том, как могут быть восстановлены составляющие X и Π .

Знание множества $B \cup I$ дает возможность однозначно построить «ядро» множества X – множество \tilde{X} , определяемое как $\tilde{X} = \bigcup_{a \in B \cup I} \{\tilde{a}\}$, где \tilde{a} – предшественник (непосредственный или более высокого ранга) дуги a . Таким образом, вопрос сводится к определению разности $X \setminus \tilde{X}$.

Рассмотрим произвольную дугу $a \notin B \cup I \cup \tilde{X}$. Если она имеет хотя бы одного предшественника из $B \cup I$, то $a \notin X \setminus \tilde{X}$ и, следовательно, $a \in \Pi$. В противном случае, если все непосредственные предшественники этой дуги содержатся в \tilde{X} , то $a \in X \setminus \tilde{X}$. Рассмотрев таким способом все дуги $a \notin B \cup I \cup \tilde{X}$, построим множество $\tilde{\tilde{X}}$, добавив к ядру все те дуги, которые содержатся в $X \setminus \tilde{X}$. Продолжим этот процесс для всех дуг $a \notin B \cup I \cup \tilde{\tilde{X}}$ до тех пор, пока $\tilde{\tilde{X}}$ не сможет быть пополнено новыми дугами. Итогом такого построения будет множество $X = \tilde{\tilde{X}}$.

Множество Π определится как $\Pi = E \setminus (B \cup I \cup X)$.

Итак, в нашем распоряжении существуют три возможности не избыточно описывать состояния ОССГ. Очевидно, предпочтение следует отдать паре $\{B, I\}$, поскольку в ней компонента B ведет себя в отличие от X более «спокойно» ($|X|$ растет по мере приближения к концу эволюции ОССГ).

Литература

1. ГОЛЕНКО-ГИНЗБУРГ Д.И. *Стохастические сетевые модели планирования и управления разработками*. – Воронеж: Научная мысль, 2010. – 283 с.
2. ЕРМАКОВ С.М., МИХАЙЛОВ Г.А. *Статистическое моделирование*. – М.: Наука, 1982. – 296 с.
3. ИВАНОВ Н.Н. *Аналитико-имитационное моделирование обобщенных стохастических сетевых графиков // Управление большими системами*. – 2015. – Вып. 53. – С. 27–44.
4. ИВАНОВ Н.Н. *Степень параллелизма обобщенных стохастических сетевых графиков // Управление большими системами*. – 2017. – Вып. 65. – С. 6–23.
5. ИВАНОВ Н.Н., ИГНАТУЩЕНКО В.В., МИХАЙЛОВ А.Ю. *Статическое прогнозирование времени выполнения комплексов взаимосвязанных работ в многопроцессорных вычислительных системах // Автоматика и телемеханика*. – 2005. – №10. – С. 89–103.
6. ИВАНОВ Н.Н., ИГНАТУЩЕНКО В.В., МИХАЙЛОВ А.Ю. *Вычисление оценок распределения времени выполнения комплексов взаимосвязанных работ в многопроцессорных вычислительных системах // Труды Института проблем управления*. – 2006. – Т. XXVII. – С. 124–135.
7. ИВАНОВ Н.Н. *Статическое прогнозирование времени выполнения комплексов взаимосвязанных работ в многопроцессорных вычислительных системах при неэкспоненциальных распределениях времени выполнения задач // Труды Института проблем управления*. – 2005. – Т. XXVI. – С. 57–72.

8. ИВАНОВ Н.Н. Алгоритмы управления ресурсами в обобщенных стохастических сетевых графиках // Управление большими системами. – 2018. – Вып. 73. – С. 95–107.

ALGORITHMS OF RESOURCE MANAGEMENT IN GENERALIZED STOCHASTIC NETWORKS

Nikolay Ivanov, V.A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences, Moscow, Doctor of Science (ivanov.nni@yandex.ru).

Abstract: The modeling of the control process in real time by a computing system with a limited number of computing resources is considered. Networking is used as a modeling tool. Limiting the number of resources can lead to the formation of queues ready to work. In this article, the work is associated with the corresponding arc of the network. Methods are proposed for managing the resources of a computing system for various disciplines of performing work that are in the buffer. In particular, the FIFO and LIFO disciplines are considered, as well as one of the heuristic disciplines. The basis for the analysis of these methods is the concept of network state. The components of the state are considered to be a set of executed works and a set of works that are ready for execution and are in the buffer. Network graphics states form a finite state tree, which serves as the basis for building simulation programs. Using these programs, the user can make a comparative analysis of the considered resource management algorithms of the computing system. Using the proposed simulation algorithms also allows the user to solve the problem of the optimal choice of the number of performers. The solution to this problem allows us to find an acceptable compromise between this number and the program execution time in real time.

Keywords: the generalized stochastic network, the state tree of the network, the execution time of the network, simulation.

УДК 519.179.2

ББК 22.176 + 65.23

Статья представлена к публикации членом редакционной коллегии А.А. Лазаревым.

Поступила в редакцию 28.03.2019.

Опубликована 31.07.2019.