УДК 681.32

В. А. Кирьянчиков
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

Расчет характеристик эффективности программ на основе марковских цепей

Рассматриваются методы оценивания эффективности выполнения программных средств (ПС) с точки зрения потребления ресурсов вычислительной системы (ВС). Отмечаются преимущества аналитического подхода к расчету характеристик эффективности на ранних этапах проектирования ПС до создания готовой программы. Показываются возможности моделирования выполнения программы на заданной ВС на основе марковских цепей с дискретным и непрерывным временем. Приводятся основные преобразования операционных графовых моделей программ при расчете характеристик эффективности методом эквивалентных преобразований на основе поглощающих марковских цепей. Показываются особенности применения марковских цепей с непрерывным временем для расчета характеристик эффективности ПС, а также способы преобразования моделей на основе марковских цепей с дискретным временем в модели с непрерывным временем и обратно с целью расширения возможностей расчета характеристик эффективности как для последовательных, так и для параллельных программ.

Эффективность, параметры потребления ресурсов, операционная графовая модель программы, марковская цепь с дискретным временем, марковская цепь с непрерывным временем, метод эквивалентных преобразований, интенсивность перехода, вектор финальных вероятностей

Эффективность является одной из интересных и важных характеристик качества программного средства (ПС), имеющих количественные значения [1]. Эффективность характеризует количество ресурсов, потребляемых при выполнении ПС. Соответственно, выявление программ или их фрагментов, потребляющих чрезмерное количество ресурсов (по известному правилу: 10 % людей потребляют 90 % всего пива), так называемых узких мест выполнения программы, позволит путем исправления этих фрагментов существенно улучшить качество ПС.

Наиболее естественный способ выявления узких мест – прогон программы под управлением профилировщика – программного измерительного монитора, позволяющего измерить потребление некоторого ресурса каждым из проверяемых фрагментов программы. Основными достоинствами измерительного метода являются точность, наглядность и простота. Проблемы в применении этого метода возникают в случаях:

– когда отсутствует готовая программа (как правило, на начальных этапах разработки);

– нет свободного доступа к аппаратно-программным средствам выполнения программы (из соображений безопасности или в случае оценки эффективности для новой, еще проектируемой системы);

– нет подходящего измерительного монитора.

Наконец, при тщательных исследованиях сложных программ необходим большой объем измерительной информации, что требует существенных затрат на ее представление и обработку.

Другой метод оценки эффективности и выявления узких мест выполнения программы основан на аналитическом подходе, при котором используются математические соотношения, полученные из формального описания модели выполнения программы на заданной системе и модели обрабатываемой рабочей нагрузки. Применение аналитических моделей может заменить реальное выполнение программы, позволяет использовать аналитический или численный расчет характеристик и значительно уменьшить стоимость работ по оценке эффективности программы. При использовании аналитического подхода необходимо выполнить следующие действия:

– формализовать модель выполнения программы в виде математических соотношений, разрешимых в замкнутой форме;

– выполнить расчет характеристик эффективности программы на основе этой модели;

– проконтролировать адекватность модели реальному выполнению программы.

Основными достоинствами аналитического подхода являются:

– удобство применения на начальной стадии разработки ввиду возможности быстрой модификации и сравнение проектных вариантов программы;

– быстрота получения оценок эффективности для различных вариантов рабочей нагрузки;

– возможность соотнесения уровня детализации анализа с точностью представлений о системе и входных данных;

– минимальная стоимость выполнения анализа.

Главный недостаток аналитических методов – это необходимость разрешения математических соотношений в замкнутой форме, что обычно достигается за счет снижения точности представления модели выполнения программы. При возрастании детализации в описании модели исчезают такие преимущества, как простота модели, наглядность описания и скорость получения оценки. Тем не менее в случае необходимости быстро и недорого оценить эффективность программ на ранних стадиях проектирования аналитические методы являются наиболее приемлемыми.

Хорошими моделями выполнения программ на некоторой вычислительной системе, удобными для проведения аналитических расчетов характеристик эффективности их выполнения, являются операционные графовые модели программ (ОГМП). Они представляют собой ориентированные графы с конечным числом вершин и дуг, которым сопоставляются параметры потребления ресурсов при выполнении программы и параметры, определяющие порядок и вероятность выбора путей выполнения программы [2]. Преимуществом этих моделей является возможность использования марковского подхода к моделированию поведения программ и расчета характеристик эффективности их выполнения на основе аппарата марковских цепей с дискретным или непрерывным временем.

Общие сведения по анализу марковских цепей, используемые для расчета характеристик эффективности программ. Марковская цепь задается тройкой параметров [3], [4]:

1. Множество состояний, в которых оказывается цепь в процессе своего функционирования:

$$S = \{S_1, S_2, \ ..., \ S_n\}.$$

2. Переходная матрица, которая задает вероятности перехода из состояния S_i в состояние S_j в момент t_k:

$$P\left(t_k\right) = \left\{p_{ij}\left(t_k\right)\right\},$$

где $p_{ij}\left(t_k\right) = \text{Prob}\left\{S\left(t_{k+1}\right) = S_j \big/ S\left(t_k\right) = S_i\right\}$.

3. Вектор начальных вероятностей, который определяет вероятность нахождения цепи в одном из состояний $S_1 \ldots S_n$ в начальный момент времени t_0:

$$\Pi(t_0) = \{\pi_1(t_0), \ldots, \pi_n(t_0)\}.$$

Если переход из одного состояния в другое выполняется через равные дискретные интервалы времени, называемые шагами, то цепь называется марковская цепь с дискретным временем (МЦДВ). МЦДВ моделирует программы, реализующие последовательные и синхронные вычислительные процессы. Если переход из одного состояния в другое может произойти в любой момент времени, то имеем марковскую цепь с непрерывным временем (МЦНВ), которая используется для моделирования параллельных и асинхронных вычислительных процессов.

Если для элементов переходной матрицы МЦДВ справедливы соотношения $\{p_{ij}(t_k+1)\} = \{p_{ij}(t_k)\}$, т. е. цепь ведет себя одинаково на каждом шаге, то такая марковская цепь называется однородной. Если элементы переходной матрицы могут изменяться от шага к шагу, то такая цепь называется неоднородной.

В некоторый произвольный момент времени t_k состояние цепи определяется соотношением

$$\Pi(t_k) = \begin{cases} \Pi(t_0)\, P^k, & \text{если цепь однородная;} \\ \Pi(t_0) \prod\limits_{i=1}^{i=k} P(t_i), & \text{если цепь неоднородная.} \end{cases}$$

Все состояния марковской цепи делятся на два множества [3]:

– множество $\mathbf{T}$ невозвратных состояний – множество состояний, покинув которое, цепь больше в них не возвращается;

– множество $\mathbf{E}$ эргодических состояний – множество состояний, попав в которое, цепь не может из него выйти.

Для этих двух множеств справедливы соотношения: $\mathbf{T} \cup \mathbf{E} = \mathbf{S}$; $\mathbf{T} \cap \mathbf{E} = \varnothing$.

Обычно множество $\mathbf{T}$ используется для моделирования поведения программы до завершения и без зацикливаний. Множество $\mathbf{E}$ используется для представления зацикленных участков программы или состояний завершения программы. Если существует эргодическое множество, состоящее из единственного состояния, то оно называется поглощающим. Если все эргодические множества поглощающие, то такая цепь называется поглощающей цепью Маркова (ПЦМ).

Расчет характеристик эффективности программ методом эквивалентных преобразований ОГМП. Метод эквивалентных преобразований [5] основан на описании поведения программы как поглощающей цепи Маркова с дискретным временем (ПЦМДВ). В качестве ОГМП используется модель программы в виде графа с нагруженными дугами [2]. Нагрузка дуг задается тремя параметрами $\{p_{ij}, m_{ij}, d_{ij}\}$, где p_{ij} – вероятность выполнения процесса, соответствующего дуге ij; m_{ij} – среднее потребление ресурсов при выполнении данного процесса; d_{ij} – дисперсия потребления ресурсов при выполнении процесса.

Поведение модели соответствует чисто марковскому процессу, если в исходной модели $d_{ij} = 0$, а при $d_{ij} \neq 0$ – полумарковскому процессу.

Суть метода эквивалентных преобразований (МЭП) – последовательное применение к исходному графу программы трех типов операций:

1) исключение вершины;

2) склеивание параллельных дуг;

3) устранение цикла.

При выполнении этих операций необходимо, чтобы статистические свойства графа, определяемые средними значениями и дисперсиями потребляемых ресурсов, до и после выполнения операции были одинаковы.

В результате последовательного выполнения указанных преобразований происходит сворачивание графа до тех пор, пока не останутся начальная и конечная вершины. Дуга, соединяющая эти вершины, будет характеризоваться параметрами потребления ресурсов всей программы.

Рассмотрим каждый вид преобразования подробнее. Пусть $l_{ij} = \{p_{ij}, m_{ij}, d_{ij}\}$.

1. Исключение вершины (рис. 1).

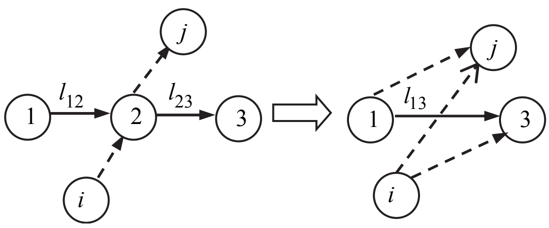

Рис. 1

Пусть t_{ij} – случайная величина, характеризующая потребление ресурсов на дуге ij. При исключении вершины 2 преобразование графа соответствует выполнению над случайными величинами операции $t_{13} = t_{12} + t_{23}$. Учитывая независимость суммируемых случайных величин, получим

$$p_{13} = p_{12} p_{23};\ m_{13} = m_{12} + m_{23};\ d_{13} = d_{12} + d_{23}.$$

2. Склеивание параллельных дуг (выполняется попарно для всех склеиваемых дуг) (рис. 2).

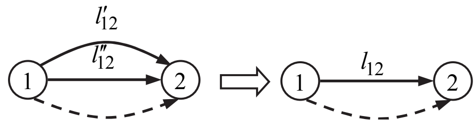

Рис. 2

Пусть U_{12} – некоторое условие, принимающее значения true или false.

Тогда рассматриваемое преобразование соответствует операции над случайными величинами

$$t_{12} = \begin{cases} t'_{12}, \text{ если } U_{12} = \text{true}, \\ t''_{12}, \text{ если } U_{12} = \text{false}. \end{cases}$$

Соответственно, для характеристик случайных величин – операндов и результата получим соотношения

$$p_{12} = p'_{12} + p'_{23};\ m_{12} = \frac{m'_{12} p'_{12} + m''_{12} p''_{12}}{p'_{12} + p'_{23}};$$

$$d_{12} = $$

$$= \frac{\left(d'_{12} p'_{12} + d''_{12} p''_{12}\right) + \left(m'_{12}\right)^2 p'_{12} + \left(m''_{12}\right)^2 p''_{12}}{p_{12}} - m_{12}^2.$$

Это преобразование может порождать дисперсию у результирующей дуги, даже если исходные дуги имели нулевые дисперсии.

3. Исключение цикла (рис. 3).

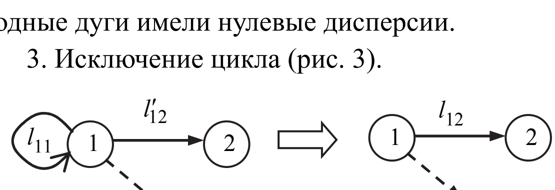

Рис. 3

Рассматриваемое преобразование соответствует операции над случайными величинами

$$t_{12} = t'_{12} + t_{11} \frac{p_{11}}{1 - p_{11}}.$$

Для характеристик случайных величин – операндов и результата получим соотношения

$$p_{12} = \frac{p'_{12}}{1 - p_{11}};\ m_{12} = m'_{12} + m_{11}\frac{p_{11}}{1 - p_{11}};$$

$$d_{12} = d'_{12} + d_{11}\frac{p_{11}}{1 - p_{11}} + \left\{ m_{11}^2 \frac{p_{11}}{\left(1 - p_{11}\right)^2} \right\}.$$

Третье слагаемое в выражении для d_{12} должно учитываться только для итерационных циклов, в которых число повторений заранее неизвестно. В этом случае будет порождаться ненулевая дисперсия для результирующей дуги даже при нулевых дисперсиях у исходных дуг.

При выполнении преобразований для минимизации числа порождаемых дуг целесообразно задать следующий ранг операций: 1) склеивание параллельных дуг (высший ранг); 2) исключение цикла; 3) исключение вершины.

Если операция «исключение вершины» может применяться к различным вершинам, то сначала исключаются вершины с наименьшим числом инцидентных дуг.

Рассмотрим пример использования МЭП для расчета среднего и дисперсии программы, ОГМП которой показан на рис. 4 (исходные дуги имеют нулевые дисперсии).

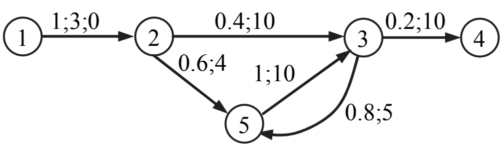

Рис. 4

Последовательность преобразований модели при сворачивании графа показана на рис. 5: *а* – исключение вершины 2; *б* – исключение вершины 5; *в* – склеивание параллельных дуг 1–3; *г* – исключение цикла у вершины 3; *д* – исключение вершины 3.

Результирующая дуга после сворачивания графа имеет нагрузочные параметры, определяющие среднее значение и дисперсию потребления рассматриваемого данной программой ресурса (например, время выполнения).

Если требуется получить характеристики потребления ресурсов для фрагмента программы, то в ее ОГМП необходимо обнулить нагрузочные значения (средние значения и дисперсии) для дуг графа, не входящих в этот фрагмент. После сворачивания графа результирующая дуга будет иметь характеристики потребления ресурсов рассматриваемым фрагментом.

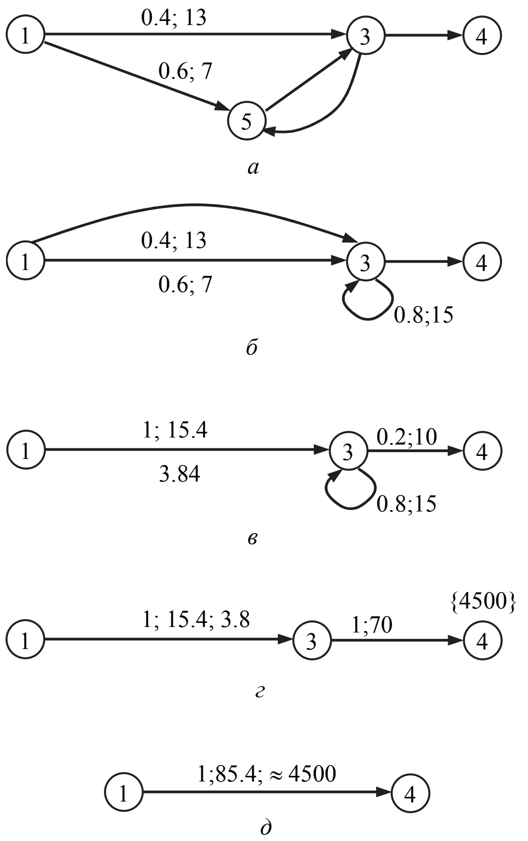

Рис. 5

Известен еще один метод расчета характеристик потребления ресурсов на основе модели программы, поведение которой соответствует ПЦМДВ, основанный на использовании фундаментальной матрицы ПЦМ [3], [4]. В данном методе для расчета характеристик потребления ресурсов необходимо выполнить обращение матрицы размерностью $n \times n$, где n – число состояний цепи. Соответственно, вычислительная (асимптотическая) сложность метода будет равна $S(n) = O(n^3)$.

Для метода эквивалентных преобразований вычислительная (асимптотическая) сложность будет определяться выражением $O(n) < S < O(n^2)$ в зависимости от способа представления графа, содержащего n дуг. Таким образом, МЭП имеет существенное преимущество по вычислительной сложности.

Использование марковских цепей с непрерывным временем для расчета характеристик эффективности программ. Марковские цепи с непрерывным временем [4] используются:

1) при анализе производительности асинхронных систем, когда запросы на ресурсы происходят в произвольные моменты времени;

2) при анализе характеристик надежности, так как отказы системы могут происходить в произвольные моменты времени [6].

Особенности переходов между состояниями цепи для МЦДВ и МЦНВ на примере линейного (рис. 6) и ветвящегося (рис. 7) процессов (*а* – МЦДВ; *б* – МЦНВ), где приняты следующие обозначения: p_{ij}, p, q – вероятности перехода; $v(t)$ – функция, удовлетворяющая соотношению $\int\limits_0^\infty v(t)dt = 1.$

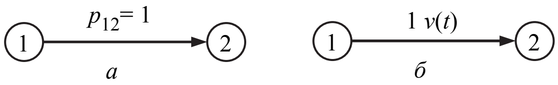

Рис. 6

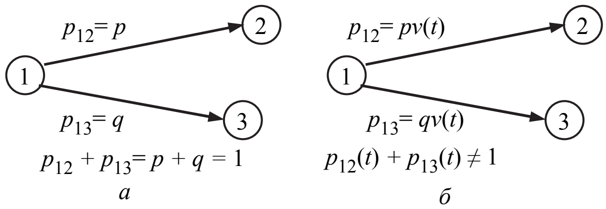

Рис. 7

Для МЦДВ переходы выполняются в фиксированные моменты времени, и для ветвящегося процесса применяется правило стохастического выбора $p_{12} + p_{13} = 1$. Для МЦНВ переходы выполняются в произвольные моменты времени, и для ветвящегося процесса правило стохастического выбора в общем случае не выполняется. Поэтому при анализе МЦНВ целесообразно вместо вероятностей использовать интенсивности переходов.

Плотностью вероятности перехода или ***интенсивностью перехода*** $\lambda_{ij}(t)$ называется [4] предел отношения вероятности перехода $p_{ij}(\Delta t)$ системы за время Δt из состояния S_i в состояние S_j к длине промежутка Δt, определяемый выражением

$$\lambda_{ij}(t) = \lim_{\Delta t \to 0} \frac{p_{ij}(\Delta t)}{\Delta t} = dp_{ij}(t)/dt\ .$$

Если для МЦДВ справедливо правило стохастического выбора $p_{ii} + \sum\limits_{i \neq j} p_{ij} = 1$, то для МЦНВ выполняется правило потоков интенсивностей $\lambda_{ii} + \sum\limits_{i \neq j} \lambda_{ij} = 0$ или $\lambda_{ii} = -\sum\limits_{i \neq j} \lambda_{ij}$.

Поведение МЦНВ полностью описывается матрицей $\mathbf{\Lambda} = \{\lambda_{ij}\}$ интенсивностей перехода из состояния S_i в состояние S_j и вектором $\Pi(0)$ начальных вероятностей состояний в момент времени $t = 0$. Марковский процесс называется однородным, если интенсивности λ_{ij} не зависят от времени, иначе имеет место неоднородный МП.

Для однородного МП действует система дифференциальных уравнений для вероятностей состояний, которая носит имя академика А. Н. Колмогорова [4]

$$d\Pi(t)/dt = \Pi(t) \times \mathbf{\Lambda}\ . \qquad (1)$$

Интегрирование этой системы по времени позволяет вычислить вероятности $\Pi_i(t)$. При этом для обеспечения линейной независимости следует рассматривать систему из $n-1$ уравнения, заменив одно из ее уравнений на уравнение

$$\sum_{i=1}^{i=n} \pi_{ij} = 1. \qquad (2)$$

Если цепь Маркова состоит только из эргодических состояний, то такая цепь называется ***эргодической***. По мере развития марковского процесса в эргодических марковских цепях (ЭМЦ) устанавливается стационарный режим, при котором вектор $\Pi(t)$ вероятностей пребывания цепи в различных состояниях не изменяется с течением времени [3]. Этот вектор не зависит от вектора $\Pi(0)$ вероятностей состояний в момент $t = 0$, называется вектором финальных вероятностей и обозначается $\Pi_ф$.

В соответствии с уравнением Колмогорова (1) и с учетом того, что в стационарном режиме $d\Pi(t)/dt = 0$, вектор финальных вероятностей (ВФВ) является решением системы уравнений

$$\Pi_ф \times \mathbf{\Lambda} = 0 \qquad (3)$$

с заменой одного из уравнений системы на уравнение (2) для обеспечения линейной независимости.

Рассмотрим пример нахождения ВФВ для простой эргодической МЦНВ, показанной на рис. 8.

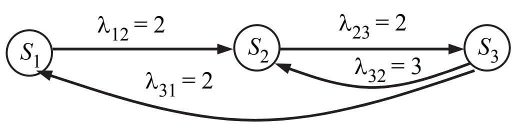

Рис. 8

Матрица интенсивностей МЦНВ имеет вид

$$\mathbf{\Lambda} = \begin{bmatrix} -2 & 2 & 0 \\ 0 & -2 & 2 \\ 2 & 3 & -5 \end{bmatrix}.$$

На основании (3) получаем систему уравнений

$$\begin{cases} -2\pi_1 + 2\pi_3 = 0, \\ \pi_1 + \pi_2 + \pi_3 = 1, \\ 2\pi_2 - 5\pi_3 = 0. \end{cases}$$

Решением системы является вектор $\Pi_ф = \{2/9, 5/9, 2/9\}$.

Для ЭМЦ существует теорема [3], в соответствии с которой среднее время возврата цепи $t_{вi}$ в состояние S_i обратно пропорционально финальной вероятности $\pi_{фi}$ пребывания в состоянии S_i

$$M\{t_{вi}\} = k / \pi_{фi},$$

$$\text{где } k = \begin{cases} 1 - \text{для равномерного критерия сложности [7];} \\ 1 \Big/ \sum\limits_{i \neq j} \lambda_{ij} - \text{для реального потребления времени.} \end{cases}$$

Соответственно, среднее время выполнения программы ($T_{пр}$) можно найти из выражения

$$M\{T_{пр}\} = \left(1 / \pi_{ф1}\right) \left(1 \Big/ \sum_{j \neq 1} \lambda_{1j}\right).$$

Для нашего примера $M\{T_{пр}\} = 9/2 \cdot 1/2 = 9/4$ (ед. времени).

При формировании моделей и расчете характеристик качества программ различного вида (последовательных или параллельных) на основе марковских цепей приходится переходить от эргодических цепей с непрерывным временем к поглощающим цепям с дискретным временем и обратно. Эти преобразования следует выполнять следующим образом.

Преобразование ЭМЦНВ в ПМЦДВ:

1) разорвать циклическую дугу цепи и направить ее в конечное (поглощающее) состояние;

2) рассчитать нагрузки каждой дуги ПМЦ по формулам

$$p_{ij} = \lambda_{ij} \Big/ \sum_{i \neq j} \lambda_{ij}; \quad m_{ij} = 1 \Big/ \sum_{i \neq j} \lambda_{ij};$$

3) выполнить анализ полученной ПМЦ, например, методом эквивалентных преобразований.

Преобразование ПМЦДВ в ЭМЦНВ.

Часто на основании исходного описания программы ее операционную модель удобно представить в виде поглощающей цепи Маркова с дискретным временем (в качестве нагрузочных параметров задать средние значения потребления ресурсов). Но затем модель должна рассматриваться как цепь с непрерывным временем, например, при анализе эффективности асинхронных программ или анализе надежности. Тогда надо исходную модель преобразовать в ЭМЦНВ.

Это преобразование выполняется более сложно, так как если вероятность p_{ij} – это относительная характеристика перехода цепи из S_i в S_j и показывает только направление перехода, а скорость перехода определяется параметром m_{ij}, то интенсивность λ_{ij} – это абсолютная характеристика перехода и показывает как направление, так и скорость перехода. Рассмотрим примеры преобразования ПМЦДВ в ЭМЦНВ для типовых процессов управления в программах (рис. 9 – линейный процесс; рис. 10 – ветвление; рис. 11 – цикл).

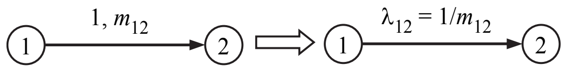

Рис. 9

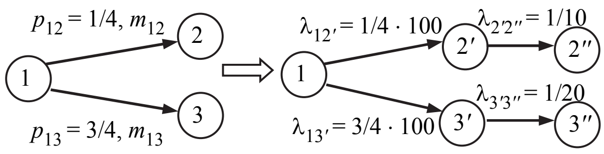

Рис. 10

Здесь $\lambda_{i'i''} = 1/m_{1i}$; $\lambda_{1j'} = p_{1j}\lambda_{гран}$; $\lambda_{гран} \gg \{1/m_{1i}; 1/m_{1j}\}$; например, $\lambda_{гран} = 100$ (или 1000).

Для раздельного учета вероятностей перехода и потребления ресурсов в ЭМЦНВ вводятся дополнительные вершины 2′ и 3′ и дуги $l_{12'}$ и $l_{13'}$, а для исключения влияния интенсивностей $\lambda_{12'}$ и $\lambda_{13'}$, определяющих направление перехода через вероятности p_{12} и p_{13}, на интенсивности, определяющие потребление ресурса в соответствующей ветви, вводится множитель $\lambda_{гран}$, представляющий большое значение интенсивности, значительно превышающее интенсивности потребления ресурса.

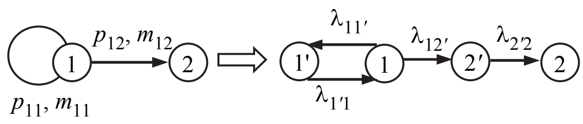

Рис. 11

Здесь $\lambda_{1'1} = 1/m_{11}$; $\lambda_{2'2} = 1/m_{12}$; $\lambda_{11'} = p_{11}\lambda_{гран}$; $\lambda_{12'} = p_{12}\lambda_{гран}$; $\lambda_{гран} >> \{1/m_{11}; 1/m_{12}\}$; например, $\lambda_{гран} = 100$ (или 1000).

Как и для ветвления, здесь для раздельного учета вероятностей перехода и потребления ресурсов в ЭМЦНВ вводятся дополнительные вершины 1′ и 2′ и дуги $l_{11'}$ и $l_{12'}$, а для исключения влияния интенсивностей $\lambda_{11'}$ и $\lambda_{12'}$, определяющих направление перехода через вероятности p_{11} и p_{12}, на интенсивности, определяющие потребление ресурса в соответствующей ветви, вводится множитель $\lambda_{гран}$, представляющий большое значение интенсивности, значительно превышающее интенсивности потребления ресурса.

После получения модели программы в виде ЭМЦНВ расчет характеристик качества следует производить путем решения системы уравнений (3).

СПИСОК ЛИТЕРАТУРЫ

1. ГОСТ ИСО/МЭК 9126–93. «Информационные технологии. Оценка программной продукции. Характеристики качества и руководства по их применению». URL: http://kinohd-online.net/serial/9538-desyat-strel-dlya-odnoy-2018.html (дата обращения: 24.03.10).

2. Кирьянчиков В. А. Методика построения операционных графовых моделей программ // Изв. СПбГЭТУ «ЛЭТИ». 2018. № 7. С. 53–58.

3. Кемени Дж., Снелл Дж. Конечные цепи Маркова. М.: Наука, 1970. 246 с.

4. Доррер Г. А. Методы анализа вычислительных систем: учеб. пособие. Красноярск: Изд-во СибГТУ, 2000. 143 с.

5. Байцер Б. Микроанализ производительности вычислительных систем. М.: Радио и связь, 1983. 360 с.

6. Sahner R., Trivedi K., Puliafito A. Performance and Reliability Analysis of Computer Systems. Kluwerr Academic Publishers, 1999. 404 с.

7. Левитин А. В. Алгоритмы: введение в разработку и анализ. М.: ИД «Вильямс», 2006. 576 с.

V. A. Kirianchikov

Saint Petersburg Electrotechnical University «LETI»

CALCULATION OF CHARACTERISTICS OF PROGRAM EFFICIENCY ON THE BASIS OF MARKOV CHAINS

Methods for evaluating the efficiency of software in terms of resource consumption of a computing system are considered. The advantages of an analytical approach to the calculation of efficiency characteristics in the early stages of program design prior to the creation of a finished program are noted. The possibilities of the program execution modeling on a given computing system based on Markov chains with discrete and continuous time are shown. The basic transformations of operational graph models of programs are given when calculating the program efficiency characteristics by the method of equivalent transformations based on absorbing Markov chains. The features of the use of continuous time Markov chains to calculate the program efficiency characteristics are shown.Also the methods for converting models based on discrete time Markov chains in continuous time modelsand vice versa are shown in order to expand the possibilities of the program efficiency characteristics calculating for both sequential and parallel programs.

Efficiency, resource consumption parameter, operational graph model of the program, discrete time Markov chains, continuous time Markov chains, method of equivalent transformations, transition intensity, final probability vector