

УДК 621.377

О.И. Елизаров, В.Г. Литвин, Е.В. Чернышова

## ОБОСНОВАНИЕ ТРЕБОВАНИЙ КО ВРЕМЕНИ ВЫПОЛНЕНИЯ ПРОГРАММ В КОМПЬЮТЕРНЫХ СИСТЕМАХ

**Елизаров Олег Иванович**, кандидат технических наук, старший научный сотрудник, начальник отдела НИИ автоматической аппаратуры им. акад. В.С. Семенихина. Окончил МВТУ им. Н.Э. Баумана и МГУ им. М.В. Ломоносова. Имеет статьи в области проектирования и эффективной организации вычислительного процесса информационно-вычислительных систем. [e-mail: eoi251139@rambler.ru].

**Литвин Василий Григорьевич**, доктор технических наук, профессор, главный научный сотрудник НИИ автоматической аппаратуры им. акад. В.С. Семенихина. Окончил Таганрогский радиотехнический институт. Автор 115 научных публикаций в области производительности вычислительных систем. [e-mail: litvg@mail.ru].

**Чернышова Елена Викторовна**, кандидат технических наук, начальник НТЦ НИИ автоматической аппаратуры им. акад. В.С. Семенихина. Окончила Московский энергетический институт (технический университет) и Институт проблем энергетической эффективности Executive MBA ИБДА РАНХ и ГС при Президенте РФ. Имеет публикации в области проектирования и эффективного использования вычислительных систем. [e-mail: dibs@list.ru].

### Аннотация

Одной из проблем проектирования вычислительных систем является эффективное управление требованиями. При этом наряду с созданием систем документальной поддержки соответствующих процессов важную роль играют инструменты оценки требований ко времени выполнения программ. В статье рассмотрены аналитические методы оценки таких требований, базирующиеся на вероятностных моделях систем и сетей массового обслуживания (СМО и СеМО). Для моделей СМО приведены явные формулы расчета требуемого среднего времени выполнения программ и времени их выполнения с заданным уровнем доверия, а для СеМО – инженерный метод оценки, построенный на зависимостях операционного анализа СеМО, и итерационный алгоритм расчета искомого времени (метод анализа средних величин – MVA). Рассмотрены примеры, демонстрирующие использование предложенных методов оценки динамических требований ко времени выполнения программ. Точность оценивалась путем сравнения полученных результатов аналитических расчетов с результатами имитационного моделирования (Монте Карло). Предложенные в статье методы могут использоваться на всех стадиях жизненного цикла вычислительных систем.

Ключевые слова: программные требования, управление программными требованиями, производительность вычислительных систем, операционный анализ, MVA.

## FOUNDATION FOR PROGRAM RUN TIME REQUIREMENTS IN COMPUTER SYSTEMS

**Oleg Ivanovich Elizarov**, Candidate of Engineering, Senior Researcher, a head of the Department of the Research Institute for Automatic Equipment named after Academician V.S. Semikhin; graduated from the Bauman Moscow State Technical University and Lomonosov Moscow State University; an author of articles in the field of design and an effective computational process organization for information systems. e-mail: eoi251139@rambler.ru.

**Vasily Grigoryevich Litvin**, Doctor of Engineering, Professor, Chief Staff Scientist at the Research Institute for Automatic Equipment named after Academician V.S. Semikhin; graduated from the Taganrog Radio-Engineering Institute; an author of 115 scientific publications in the field of computing systems performance. e-mail: litvg@mail.ru.

**Elena Viktorovna Chernyshova**, Candidate of Engineering; a chief of Scientific and Engineering Centre at the Research Institute for Automatic Equipment named after Academician V.S. Semikhin; graduated from Moscow Power Institute (Technical University), Institute of Power Efficiency Problems and Executive MBA; an author of articles in the field of design and efficient use of computing systems. e-mail: dibs@list.ru.

### Abstract

One of the important problems in computer systems design is the efficient requirement engineering. Therefore, along with the system creation for documentary support of appropriate processes, evaluation tools for program run time requirements

play the important role. The article deals with analytical evaluation procedure for the requirements based on the stochastic models for the queueing systems and queueing networks (QS and QN). The explicit formulas for calculating mean time required for program implementation and the time required for program implementation with the set level of confidence are given for the QS-models. The engineering estimation method based on the dependences of QN operational analysis and the iteration algorithm for calculation of required time (Mean-Value Analysis – MVA) are presented for the QN-models. The examples showing the application of proposed methods for dynamic run time requirements estimation are considered. The accuracy was estimated by comparison of results of analytical calculations against results of Monte Carlo simulations. The proposed methods can be used at all stages of computer systems life cycle.

Key words: software requirements, software requirements engineering, computer system performance, operational analysis, MVA.

## ВВЕДЕНИЕ

Существует большое число различных классификаций вычислительных систем, в которых время является главным признаком. Укажем на следующие три класса систем, получивших наибольшее распространение: 1) класс систем, в которых время не является ключевым показателем оценки эффективности работы (non-real-time); 2) класс, время выполнения запросов клиентов в которых не установлено жестко и может превышать некоторую желательную для пользователя величину, не вызывая у него раздражения, если большие превышения времени случаются редко (soft-real-time); 3) и наконец, класс систем, в которых требуется жесткое выполнение ограничений на время ответа (hard-real-time) [1]. Мы будем рассматривать системы, относящиеся ко второму классу, для которых характерны либо оценки усредненных показателей времени ответа, либо вероятностно-временные характеристики. При выработке требований к показателям работы вычислительных систем необходимо проводить комплексное рассмотрение вычислительного процесса, динамические характеристики которого зависят от множества факторов, включая производительность технических средств, системное управление ресурсами, процессами и потоками, качество разработки прикладного программного обеспечения, организацию работ пользователей и др. Особую сложность в определении влияния на производительность системы и ее реактивность представляют факторы, связанные со временем выполнения программ. В связи с чем, использование методов и моделей определения динамических требований к этому времени может оказаться весьма важным инструментом обоснования решений на различных стадиях разработки и использования программного обеспечения систем [1, 2].

Обычно выделяют три обобщенных уровня формирования требований к вычислительным системам и процессам их функционирования [3, 4]:

1) пользовательский уровень; 2) уровень системных требований; 3) уровень требований к подсистемам, их компонентам и процессам. В соответствии с указанными уровнями модели, используемые для оценки показателей работы этих систем, также условно можно разделить на три следующие группы: 1) модели оценки выполнения внешних требований к информационно-вычислительным системам (ИВС); 2) функциональные модели и модели затрат; 3) модели нагрузки и производительности.

В данной статье рассматриваются модели третьей группы, базирующиеся на методах и алгоритмах расчета СМО и СеМО, которые используются для оценки требований ко времени выполнения отдельных программ при заданных внешних ограничениях на времена предоставления тех или иных сервисов клиентам.

## 1 ОЦЕНКА ТРЕБОВАНИЙ КО ВРЕМЕНИ ВЫПОЛНЕНИЯ ПРОГРАММ С ИСПОЛЬЗОВАНИЕМ МОДЕЛЕЙ СМО

Рассмотрим вначале процесс оценки требований ко времени выполнения программ, базируясь на зависимостях операционного анализа (ОА) и результатах, полученных для классических СМО [5, 6]. ОА вычислительных систем был разработан в качестве инженерного подхода к расчету усредненных показателей их функционирования. Следуя традиции, введем основные переменные, используемые как в ОА, так и в алгоритмах расчета СМО и СеМО (табл. 1).

Таблица 1  
Основные переменные, используемые в расчетах

Обозначение	Пояснение
$T$	Время наблюдения за системой
$B$	Время занятости системы выполнением запросов
$U$	Нагрузка устройства обслуживания
$S$	Среднее время выполнения программы
$X$	Интенсивность потока запросов
$M$	Количество клиентов, работающих с системой
$Z$	Среднее время обдумывания результатов клиентом
$R$	Среднее время ответа системы на запросы клиентов или время пребывания запроса в системе

Выпишем зависимости, определяющие требуемое время выполнения программы для заданных условий ее использования (табл. 2).

Таблица 2

Основные зависимости, обеспечивающие оценку предельного среднего времени выполнения программ

Определение $S$	Номер формулы	Пояснения
$S = \frac{B(R+Z)}{TM}$	(1)	Предельное среднее время выполнения программы $S$ для заданного $R$ (рис. 1)
$S = \frac{R}{XR - \ln(1 - F(R))}$	(2)	Время выполнения программы $S$ , которое не может быть превышено с вероятностью $F(R)$ (рис. 2)
$S = R(1 - U) = \frac{R}{1 + XR}$		Предельное среднее время выполнения программы $S$ при простейших предположениях о законах распределения (рис. 2)
$S = \frac{2R(1 - U)}{2 - U(1 - C^2)}$	(3)	Предельное среднее время выполнения программы $S$ для заданного $R$ при нагрузке $U$ и коэффициенте вариации времени выполнения $C$ (рис. 2)

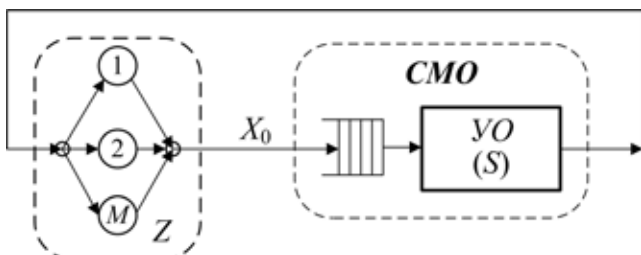


Рис. 1. Замкнутая система обслуживания

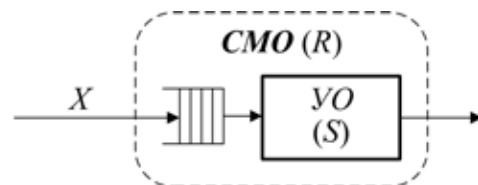


Рис. 2. Открытая система обслуживания

Рассмотрим пример оценки требования ко времени выполнения программы, когда необходимо, чтобы время ответа системы не превышало заданную величину  $R$  с вероятностью  $F(R)$ . Понятно, что помимо характеристик программы на искомое время существенное влияние оказывает организация вычислительного процесса и производительность соответствующих технических средств.

Пример 1. Необходимо определить требование ко времени выполнения программы, обрабатывающей поступающие в систему запросы с интенсивностью  $X = 0,5$  запросов в секунду, время ответа на которые не должно превышать заданную величину  $R = 15$  с с вероятностью 0,95.

В таблице 3 приведены результаты расчета требуемого предельного времени выполнения программы по формуле (2), обеспечивающего необходимую реактивность системы при различных уровнях доверия к полученным оценкам.

Таблица 3

Расчеты времени выполнения программ

Уровень доверия ( $F(R)$ )	Время выполнения программы ( $S$ ) (с)
0,99	1,24
0,95	1,43
0,9	1,53
0,8	1,65

Применение упрощенных моделей расчета требований ко времени выполнения отдельных программ в локальном случае не позволяет учесть некоторые факторы моделируемой системы, которые могут повлиять на полученные оценки. Тем не менее, приведенные формулы могут оказаться весьма полезными при оценках на начальных стадиях разработки систем и их программного обеспечения.

## 2 ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ КО ВРЕМЕНИ ВЫПОЛНЕНИЯ ПРОГРАММ С ИСПОЛЬЗОВАНИЕМ СЕТЕВЫХ МОДЕЛЕЙ МАССОВОГО ОБСЛУЖИВАНИЯ

В тех случаях, когда не представляется возможным автономное исследование узла системы и соответствующей программы, не потеряв при этом некоторых существенных особенностей их функционирования, обоснование динамических требований ко времени выполнения программы необходимо проводить с учетом существующего окружения. Для этого хорошо подходят модели ИВС, представленные в виде СеМО. Рассмотрим два метода оценок среднего времени выполнения программы выделенным модулем вычислительной сети (рис. 3).

Первый метод базируется на зависимостях ОА для СеМО [5, 7], а второй – на доработанном итерационном алгоритме MVA [7, 8]. Оценка требований будет вестись для средних значений времени выполнения программ. Более тонкие оценки, учитывающие вероятностные характеристики исследуемых систем, могут быть получены на имитационных моделях.

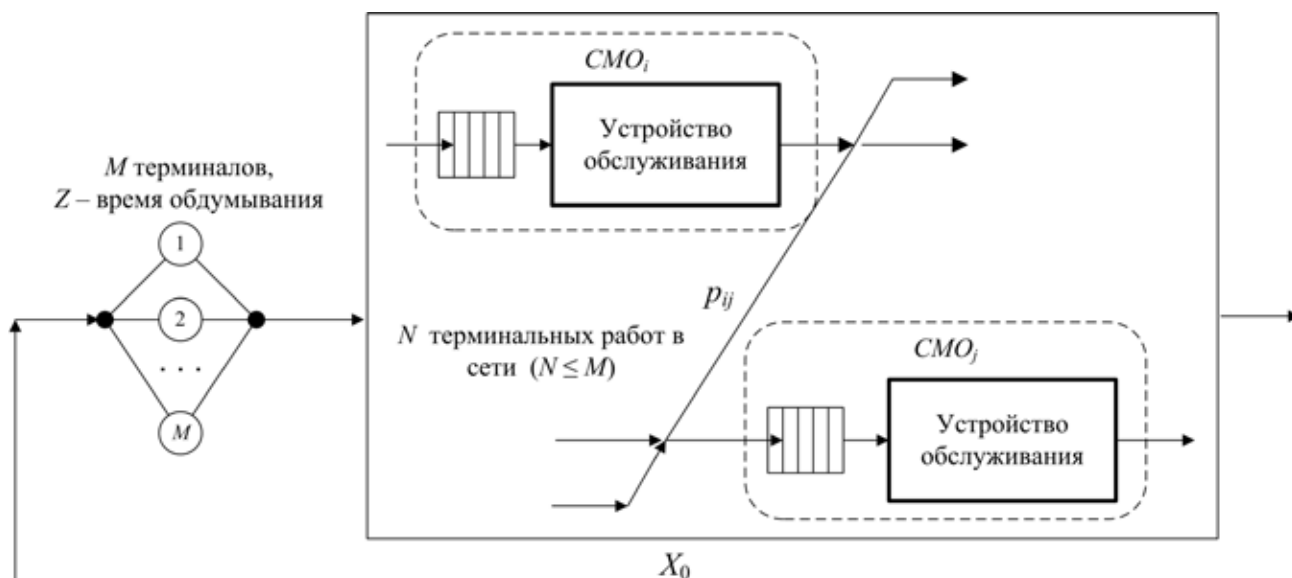


Рис. 3. Структура замкнутой СеМО

Рассмотрим более детально структуру и работу СМО (рис. 3). Сеть состоит из терминального узла ( $M$  терминалов), генерирующего запросы, и связанных между собой устройств обслуживания. Клиент, отправивший запрос с терминала, ожидает завершения его выполнения и новых запросов не создает. Получив ответ, клиент обдумывает его в течение среднего времени  $Z$ , после чего выдает новый запрос. Запросы поступают в сеть и циркулируют между узлами в соответствии с заданной вероятностной маршрутной матрицей  $P = \{p_{ij}\}$ , где  $i, j = \overline{0, K}$ , а  $0$  обозначает терминальный узел. Времена выполнения запросов в каждом узле являются случайными величинами (сл. в.) со средними значениями  $S_k, k = \overline{1, K}$ . К устройствам обслуживания образуются очереди запросов, выполняемых с дисциплиной FCFS. Так как с маршрутными матрицами работать не всегда удобно, то введем коэффициенты посещения запросами отдельных узлов. Коэффициенты посещения ( $V_k$ ) определяют среднее число раз, которое запрос, поступивший в сеть, посетит  $k$ -й узел до его выхода из сети. Коэффициенты посещения определяются через вероятностную матрицу  $P$  следующим образом:

$$V_0 = 1, \quad V_k = p_{0k} + \sum_{j=1}^K V_j p_{jk}, \quad k = \overline{1, K}. \quad (4)$$

Перейдем к рассмотрению предлагаемых методов расчета требований ко времени выполнения программ при сетевом модельном представлении вычислительных систем.

1. ОА СеМО обеспечивает усредненные оценки показателей при выполнении условий баланса потоков (считается, что процесс обслуживания в сети установился).

Используя формулу Литтла, выпишем вначале выражение для расчета среднего времени ответа на запросы, поступающие от  $M$  клиентов.

$$R = \frac{M}{X_0} - Z. \quad (5)$$

Так как все параметры в (5), кроме  $X_0$ , известны, то отсюда определяется интенсивность потока запросов, поступающего в сеть:

$$X_0 = M / (R + Z). \quad (6)$$

Если известны времена задержек запросов в каждом узле сети, то для времени ответа ( $R$ ) может быть выписана другая формула:

$$R = \sum_{j=1}^K V_j R_j, \quad (7)$$

где  $R_j$  – среднее время пребывания запроса в узле  $j$  (от момента входа в узел  $j$  до выхода из него).

Полагая, что неизвестным является время выполнения работ в узле  $i$ , определим его из формулы (7):

$$R_i = \frac{R - \sum_{j=1, j \neq i}^K V_j R_j}{V_i}, \quad (8)$$

при условии, что  $R - \sum_{j=1, j \neq i}^K V_j R_j > 0$ .

Часто на отдельные компоненты системы накладываются ограничения по нагрузке  $U_j, j = 1, 2, \dots, K$ , например,  $U_j < 75\%$ . Тогда, используя (6) и (4), можно оценить требуемое время выполнения программы:

$$S_i = U_i / (V_i X_0). \quad (9)$$

Таким образом, имея в своем распоряжении модель вычислительной системы в виде СеМО, для которой заданы отмеченные выше параметры, используя зависимости ОА, можно определить предельное среднее время выполнения программы в узле  $i$ .

Узел  $i$  может представлять собой некоторую подсеть, например, сервер баз данных с большим количеством

дисковых накопителей. И в этом случае предложенная методика оценки требований ко времени выполнения программы может быть применена.

2. Другой, более точный подход к оценке требований ко времени выполнения программы основывается на методе MVA [7, 8]. Приведем итерационный алгоритм, обеспечивающий расчет требуемого среднего времени выполнения конкретной программы.

Рассмотрим СеМО (рис. 3), для которой заданы основные параметры (коэффициенты посещения, среднее время выполнения запросов в узлах сети). Необходимо получить оценку искомого предельного времени выполнения программы в узле  $i$ .

Алгоритм.

1. Устанавливаются начальные значения.

1.1. Вводятся значения всех параметров СеМО.

1.2. Количество запросов клиентов, находящихся в очереди к устройствам обслуживания, принимается равным нулю ( $Q_k(0) = 0, k = \overline{1, K}, k \neq i$ ).

1.3. Устанавливается число клиентов, одновременно работающих с сетью ( $M$ ).

1.4. Фиксируется время ответа сети на поступающие запросы клиентов ( $R$ ).

2. Расчет показателей работы сети и требуемого времени выполнения программы узлом  $i$  путем последовательного выполнения шагов 2.1 – 2.6, начиная с  $m = 1$ , до достижения заданной величины  $M$ .

2.1. Определение времени пребывания поступающих запросов в узлах сети для текущего  $m$ :

$$R_k(m) = S_k(1 + Q_k(m)), k = \overline{1, K}, k \neq i.$$

2.2 Проверка условия:  $R - \sum_{j=1, j \neq i}^K V_j R_j(m) > 0$ .

При его нарушении осуществляется переход к шагу 3 (завершается выполнение алгоритма).

2.3. Определение требуемого времени выполнения программы для текущего  $m$ :

$$R_i(m) = \left[ R - \sum_{k=1, k \neq i}^K V_k R_k(m) \right] / V_i,$$

$$S_i(m) = \frac{R_i(m)}{(1 + Q_i(m-1))}.$$

2.4. Определение интенсивности потока запросов в сеть для текущего  $m$ :

$$X_0(m) = \frac{m}{\sum_{k=1}^K V_k R_k(m) + Z}.$$

2.5. Расчет средней очереди запросов в каждом узле сети

$$Q_k(m) = X_0(m) V_k S_k(m), k = \overline{1, K}.$$

2.6. Если  $m < M$ , то количество терминалов (клиентов) увеличивается на единицу и осуществляется переход к шагу 2.1.

3. Завершение выполнения алгоритма.

Пример 2. Рассмотрим систему «клиент-сервер», состоящую из центрального процессора (ЦП), двух дисковых накопителей ( $D_1$  и  $D_2$ ) и терминального узла ( $M = 10$ ) (рис. 4). Программа, обрабатывающая поступивший запрос на ЦП, в течение своего выполнения 10 раз обращается к  $D_1$  ( $V_2 = 10$ ) и 5 раз – к  $D_2$  ( $V_3 = 5$ ). Таким образом, коэффициент посещения ЦП  $V_1 = 16$ . Время выполнения запросов на  $D_1$  равно 300 мс, а на  $D_2$  – 200 мс. После выполнения очередного запроса клиент обдумывает результат и готовит новый запрос в течение среднего времени 25 с ( $Z = 25$  с). Необходимо определить такое требование ко времени выполнения программы на ЦП, чтобы время ответа клиентам не превышало 8 с ( $R = 8$  с).

Используя приведенный выше алгоритм, легко определить искомое требование ко времени выполнения программы в ЦП, обеспечивающее заданное время ответа

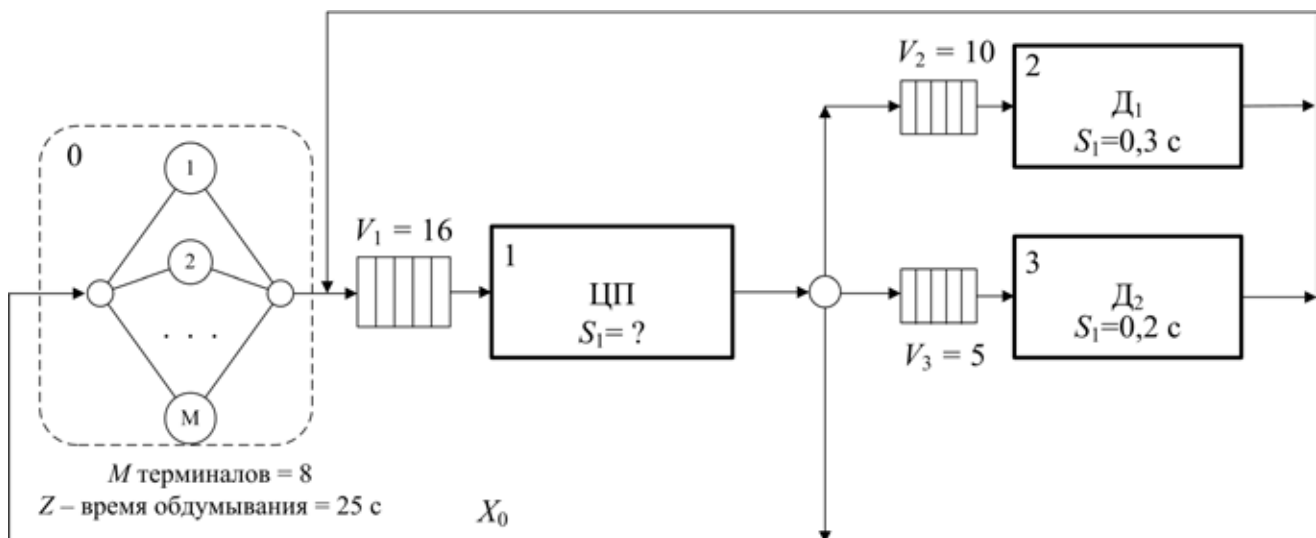


Рис. 4. Замкнутая СеМО, являющаяся моделью системы «клиент-сервер»



сервера на поступающие запросы. Приведем результаты расчетов путем последовательного увеличения числа клиентов, работающих с сервером (табл. 4).

Для приведенных исходных данных, характеризующих рассматриваемую систему «клиент-сервер», с ростом числа клиентов требуемое предельное время выполнения программы уменьшается, так как необходимо выдерживать заданное время ответа сервера. Естественно, что реальное время выполнения в каждом конкретном случае может быть меньше  $S_1$ . Оценка точности разработанного алгоритма проводилась путем сравнения результатов аналитических расчетов с результатами имитационного моделирования в среде AnyLogic. Как видно из таблицы 4, точность полученных аналитических оценок требуемого времени выполнения программы вполне приемлема.

В тех случаях, когда необходимо оценить требования ко времени выполнения нескольких программ, возникает задача перераспределения ограниченного ресурса времени между этими программами, используя при этом модельные оценки.

### ЗАКЛЮЧЕНИЕ

Важными инструментами, используемыми при создании ИВС реального времени, являются модели оценки требований ко времени выполнения входящих в них программ. Применение в качестве таких моделей СМО и СеМО позволяет получить приемлемые для практики оценки динамических требований ко времени выполнения программ. Используя методы ОА и классические методы расчета СМО, получены явные формулы для расчета искомого времени. Для случая оценки времени выполнения программы, входящей в состав ИВС, разработан алгоритм расчета модели, представленной в виде замкнутой СеМО. В основе алгоритма лежит известный метод MVA. Точность предложенных аналитических методов расчета требований ко времени выполнения программ весьма высока при различных уровнях нагрузок.

### СПИСОК ЛИТЕРАТУРЫ

1. Wilhelm R., Engblom J. The Worst-Case Execution Time Problem. In: *Overview of Methods and Survey of Tools*, 2008. – pp. 1–47.
2. Беззубцев С.О. Об одной задаче оценки времени выполнения программ на специализированном оборудовании // *Образование прикладной и промышленной математики*. – 2012. – Т. 19, вып. 3. – С. 1–3.
3. Елизабет Х., Кен Д., Джереми Д. *Разработка и управление требованиями*. – М. : Компания Telelogic, 2005. – С. 240.
4. Соммервилл И. *Инженерия программного обеспечения*. – М. : Вильямс, 2002. – С. 466.

Требования ко времени выполнения программы

Количество клиентов ( $m$ )	Поток запросов ( $X_0$ )	Требуемое время ( $S_1$ ), с – расчет	Требуемое время $m(S'_1)$ , с – имитация	Ошибка ( $S_1 - S'_1$ )
1	0,030	0,250	0,2496	0,0004
2	0,061	0,206	0,210	-0,0043
3	0,091	0,171	0,175	-0,0046
4	0,121	0,140	0,145	-0,0048
5	0,152	0,112	0,116	-0,0047
6	0,182	0,083	0,086	-0,0033
7	0,212	0,049	0,048	0,0013
8	0,242	0,005	0,001	0,0036

5. Denning P.J., Buzen J.P. Operational analysis of queueing network models // *Computing Surveys*. – 1978. – Vol. 10, No. 3. – pp. 225–261.

6. Bolch G., Greiner S., Meer H., Trivedi K. *Queueing networks and Markov chains: modeling and performance evaluation with computer science applications*. – John Wiley & Sons, Inc., 2006. – p. 877.

7. Литвин В.Г., Аладышев В.П., Винниченко А.И. Анализ производительности мультипрограммных ЭВМ. – М. : Финансы и статистика, 1984. – С. 159.

8. Reiser M., Lavenberg S. Mean-Value Analysis of closed multichain queueing networks // *Journal of the ACM*. – 1980. – Vol. 27, No 2. – pp. 313–322.

### REFERENCES

1. Wilhelm R., Engblom J. The Worst-Case Execution Time Problem. In: *Overview of Methods and Survey of Tools*, 2008, pp. 1–47.
2. Bezzubtsev S.O. Ob odnoy zadache otsenki vremeni vypolneniya programm na spetsializirovannom oborudovanii [On Problem of Program Run Time Estimation in Special-Purpose Hardware]. *Obrazovaniye prikladnoy i promyshlennoy matematiki* [Formation of Applied and Industrial Mathematics], 2012, vol. 19, no. 3, pp. 1–3.
3. Elizabeth Hull, Kenneth Jackson, Jeremy Dick. *Razrabotka i upravleniye trebovaniyami* [Requirement Engineering]. Moscow, Kompaniya Telelogic Publ., 2005. 240 p.
4. Sommervill I. *Inzheneriya programmogo obespecheniya* [Software Engineering]. Moscow, Vilyams Publ., 2002. 466 p.
5. Denning P.J., Buzen J.P. Operational analysis of queueing network models. *Computing Surveys*, 1978, vol. 10, no. 3, pp. 225–261.
6. Bolch G., Greiner S., Meer H., Trivedi K. *Queueing networks and Markov chains: modeling and performance evaluation with computer science applications*. John Wiley & Sons, Inc., 2006. 877 p.
7. Litvin V.G., Aladyshev V.P., Vinnichenko A.I. *Analiz proizvoditelnosti multiprogrammnykh EVM* [Performance Analysis of Multiprogrammed Computers]. Moscow, Finansy i statistika Publ., 1984. 159 p.
8. Reiser M., Lavenberg S. Mean-Value Analysis of closed multi chain queueing networks. *Journal of the ACM*, 1980, vol. 27, no 2, pp. 313–322.