

РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ МОДЕЛИРОВАНИЯ СЛОЖНЫХ СИСТЕМ СЕТЯМИ ПЕТРИ И РАСЧЁТА ПОКАЗАТЕЛЕЙ ИХ НАДЁЖНОСТИ И ПРОИЗВОДИТЕЛЬНОСТИ

А.В. Васяшин¹, А.Р. Теплякова¹

¹ Обнинский институт атомной энергетики – филиал федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский ядерный университет «МИФИ», г. Обнинск

Сети Петри являются эффективным инструментом, применяемым для моделирования сложных дискретных систем. При помощи моделей, представляемых сетями Петри, можно не только понять, как ведёт себя та или иная система, но и определить показатели её надёжности и производительности. Отсюда возникает потребность в создании удобного средства для решения подобного рода задач. Реализуется это за счёт разработки и объединения в одном программном комплексе расчётных модулей и редактора сетей Петри. В статье приводится краткое описание создания графического редактора сетей Петри, являющегося основным модулем программного комплекса, описываются теоретические основы, заложенные в основу разработанных расчётных библиотек, а также созданное программное средство в целом. Также рассматривается приложение полученного средства к моделированию и расчёту показателей надёжности и производительности многопроцессорной отказоустойчивой системы. Разработка является довольно гибкой, может быть дополнена другими расчётными модулями, её использование может быть полезно не только специалистами, занимающимся расчётом показателей надёжности систем, но и преподавателями и студентами вузов в рамках выполнения лабораторных работ по курсу "Моделирование систем".

Ключевые слова: сети Петри, моделирование, надёжность, производительность.

ВВЕДЕНИЕ

Обеспечение надёжности сложных систем является важной задачей, оказывающей влияние на многие сферы деятельности человека. Надёжность представляет собой одну из самых важных характеристик любой технической системы, учитываемых практически на всех этапах её жизненного цикла: от разработки спецификации требований до эксплуатации. С течением времени техника развивается и, как следствие, усложняется, что делает проблему её надёжности ещё более значимой. Для уменьшения количества случайных факторов, которые непредсказуемым образом влияют на надёжность системы, необходимы знания о протекании процессов, которые происходят в её составных элементах. Стоит отметить, что прогнозировать надёжность количественно можно только при наличии модели системы, описывающей отказы и восстановления, происходящие в ней. Так как отказы в сложной системе, как правило, происходят в случайные моменты времени, то для её описания требуется вероятностная, или стохастическая, модель. Для стохастических процессов определяются вероятности тех или иных вариантов протекания. Такой случайный процесс, вероятностные характеристики которого в будущем для любого момента времени t_0 зависят только от

того, в каком состоянии находится процесс в данный момент, и не зависят от того, когда и каким образом система пришла к этому состоянию, называется марковским процессом.

Основной характеристикой однородных марковских процессов является отсутствие зависимости значений вероятностей переходов от времени, то есть предполагается экспоненциальный характер распределения времени безотказной работы. Состояние системы, описываемой однородным марковским процессом, в любой момент времени t_0 однозначно определяет её случайное (стохастическое) поведение в будущем.

Что касается моделирования марковских процессов, то оно осуществляется при помощи стохастических сетей Петри. В стохастических сетях Петри вводится дополнительный тип узлов – стохастические переходы. Для моделирования некоторых процессов необходимо количественно рассматривать время, поэтому немедленных переходов, входящих в стандарт сетей Петри, недостаточно. Стохастические же сети Петри как расширение стандарта могут рассматривать время [1-3].

В ходе исследования были изучены имеющиеся программные средства для моделирования систем сетями Петри. Были сформулированы критерии для оценки этих программных средств в плане

пригодности для решения задач, заключающихся в построении модели систем и оценке их надёжности. Оказалось, что большинство из рассмотренных средств представляют собой просто редакторы сетей Петри, а те, которые содержат функции для оценки надёжности, являются достаточно ограниченными и не могут быть дополнены расчетными модулями. Был сделан вывод о необходимости разработки собственного программного средства, подходящего для решения описанных задач.

ОСНОВНАЯ ЧАСТЬ

Главным компонентом программного решения является графический редактор для работы с сетями Петри. При выборе инструмента для разработки учитывалась необходимость реализации взаимодействия пользователя с графическими объектами, представляющими элементы модели, интуитивно понятным и максимально удобным образом (перетаскивание с зажатой кнопкой мыши и прочее), а также необходимость реализации эффектов анимации (для эмуляции выполнения сетей). Из сказанного был сделан вывод о необходимости поиска оптимальной библиотеки для создания графического пользовательского интерфейса приложения. Был произведен обзор кроссплатформенных инструментов для разработки пользовательских интерфейсов (Qt, wxWidgets, JUCE, CEGUI). В качестве инструментария для разработки программного средства для работы с сетями Петри была выбрана библиотека Qt. Она отличается от остальных рассмотренных тем, что для работы с ней могут быть использованы удобные среды разработки Qt Creator и Qt Designer, а также имеется много документации, содержащей информацию о компонентах, входящих в состав Qt. Также нельзя не упомянуть кроссплатформенность Qt (библиотека доступна как на Windows, Linux и Mac, так и на мобильных платформах — Windows Mobile, Android и iOS), являющуюся существенным достоинством данного инструмента [4, 5].

Разработанное программное средство состоит из семи модулей:

1. Графический редактор;
2. Модуль для построения графа достижимых маркировок;
3. Модуль для формирования файлов, необходимых для дальнейших расчётов;
4. Модуль для расчёта средней производительности;
5. Модуль для расчёта моментов меры затрат;
6. Модуль для оценки распределения затрат (доходов);
7. Модуль для построения графиков верхних и нижних границ распределения.

Основной модуль (графический редактор) реализует возможности добавления элементов сети

(позиций, переходов и дуг), их изменения и рисования, а также моделирует выполнение сети. Созданные в редакторе сети Петри могут быть сохранены в файлы формата xml, а впоследствии могут быть воспроизведены из этих файлов. Также реализована возможность имитации выполнения сети, сопровождающегося перемещением фишек из одних позиций в другие.

Второй модуль позволяет построить граф достижимых маркировок, на основании которого могут быть рассчитаны показатели надёжности технической системы, моделируемой данной сетью Петри.

После того, как пользователь построил графическое представление графа достижимых маркировок, ему предоставляется возможность сформировать три файла, соответствующих этому графу и являющихся входными для четвёртого и пятого модулей. За их формирование отвечает третий модуль. Файл «q.m», помимо базового описания констант и перечисления маркировок, содержит матрицу переходов для моделируемой сетью Петри марковского процесса. Если говорить в терминах сетей Петри, в данном файле содержатся сведения о том, из какой маркировки в какую можно перейти в результате срабатывания того или иного перехода. Файл «g.v», помимо базового описания констант и перечисления маркировок, содержит вектор темпов накопления затрат, размер которого равен количеству маркировок. Файл «p0.v», помимо базового описания констант и перечисления маркировок, содержит вектор вероятностей начального состояния.

Четвёртый модуль представляет собой динамически подключаемую библиотеку brunoshin.dll, которая была собрана в Visual Studio 2015, а затем подключена к проекту приложения в среде Qt Creator. Она предназначена для расчёта вероятности того, что приведённый накопленный доход за время превысит заданное значение. В библиотеке brunoshin.dll применяется алгоритм вычисления распределения производительности, предложенный Серикола и Набли в работе [6]. Его вычислительная сложность является полиномиальной, и он оперирует только неотрицательными числами не большими единицы. Это важное свойство позволяет определить границы усечения бесконечных сумм, для того чтобы улучшить время выполнения алгоритма. Главной функцией этой библиотеки является brunoFrm, которая принимает 11 параметров, наиболее важными из которых являются те, которые задают время, для которого производится расчёт, время начала и окончания расчёта, количество шагов.

Пятый модуль, библиотека mrmsolve.dll, применяется для расчёта моментов накопленных затрат. В её основе лежит следующая теорема:

n -ый момент ($n \geq 1$) накопленных затрат есть

$$m^{(n)}(t) = (-1)^n \sum_{i=0}^{\infty} \frac{t^i}{i!} N^{(n)}(i) \cdot h,$$

где $N^{(n)}(i)$ определяется следующим образом

$$N^{(n)}(i) = \begin{cases} I, & (1) \\ 0, & (2) \\ \Lambda^i, & (3) \\ \Lambda \cdot N^{(n)}(i-1) - nV \cdot N^{(n-1)}(i-1), & (4) \end{cases}$$

(1) – если $i = 0, n = 0$;

(2) – если $i = 0, n \geq 1$;

(3) – если $i \geq 1, n = 0$;

(4) – если $i \geq 1, n \geq 1$.

Выходной файл состоит из строк, в которых содержатся следующие значения: время, соответствующие ему моменты, минимально возможные накопленные затраты за это время (\min), максимально возможные накопленные затраты за это время (\max). Строка файла, другими словами, сообщает следующее: случайная величина "накопленные затраты за указанное время" распределена на интервале $[\min, \max]$ и имеет следующие моменты.

Библиотека `distributionestimation.dll`, являющаяся шестым модулем, используется для оценки распределения накопленных затрат. Метод оценки распределения функции, применяемый в разработанной библиотеке, даёт область, содержащую все возможные функции распределения. Это практически означает, что метод даёт верхнюю и нижнюю границы возможных функций распределения (например, если единственной информацией о неотрицательной случайной величине является ее среднее значение (μ_1), то идеальная

оценка даёт область $1 \geq F(x) \geq 1 - \frac{\mu_1}{x}$ на основе

неравенства Маркова). Этот метод оценки даёт лучшие и худшие случаи для исследуемой меры. В качестве входного файла для главной функции используется выходной файл, формируемый пятым модулем. Выходной файл содержит в себе верхние и нижние оценки распределения накопленных затрат.

Для наглядности представления оценок вероятностей достижения определенных затрат на основании данных, сформированных шестым модулем, седьмой модуль строит графики на базе виджета `QCustomPlot`.

Далее приводится модель архитектуры отказоустойчивой, мультипроцессорной системы с разделяемой памятью, которая в качестве примера была смоделирована и проанализирована в разработанном приложении. Вышеупомянутая система состоит из n ЦПУ, шины и восстанавливаемой разделяемой памяти. Каждое ЦПУ

состоит из двух процессоров, что можно рассматривать как нагруженный резерв, так как их выходы сравниваются для обнаружения отказа. Каждое ЦПУ получает доступ к разделяемой памяти посредством отдельного (личного) КЭШа, который содержит самые последние данные, используемые процессором. Эта архитектура была разработана таким образом для того, чтобы восстанавливаемая, разделяемая память была единственным специализированным аппаратным обеспечением и могла, следовательно, использовать стандартные процессоры, КЭШ и протоколы связи с КЭШами. Протокол обратного восстановления, который используется в этой архитектуре, для того чтобы переносить некоторые отказы процессоров, реализуется средствами восстанавливаемой, разделяемой памяти. Для того чтобы обеспечить обратное восстановление результатов, базовый механизм разделяемой памяти имеет для поддержки по две копии для каждого места хранения памяти, которыми являются, соответственно, одна текущей копией, доступной для ЦПУ, а другая восстановительной копией, соответствующей предыдущей точке восстановления. Когда устанавливается новая точка восстановления, текущая копия воспроизводится на восстановительной копии, так что они обе содержат идентичные данные. Здесь предполагается, что как шина, так и память совершенно надёжны. С другой стороны, отказ процессора может быть всегда обнаружен, будь он постоянным или временным отказом, например, путём проведения диагностических проверок повреждения ЦПУ. Повреждённое ЦПУ будет использоваться в системе, если повреждение временное, и не будет использоваться, если повреждение является постоянным. Интенсивность отказа каждого ЦПУ задаётся параметром β . Временный отказ обнаруживается с вероятностью d , а постоянный отказ с вероятностью $1-d$. После такого события выполняется протокол обратного восстановления в течение времени, имеющего экспоненциальное распределение с интенсивностью μ . Более того, предполагается, что протокол обратного восстановления переконфигурирует систему корректно с вероятностью c и неудачно с вероятностью $1-c$. Всё это определяет марковский процесс, граф переходов которого показан на рис. 1, для системы с количеством ЦПУ $n = 3$. Состояние i ($1 \leq i \leq n$) соответствует состоянию системы, в котором i ЦПУ работоспособны. В таком состоянии отказ обнаруживается с интенсивностью $i\beta$. Состояние $n+1$ ($1 \leq i \leq n$), соответствует состоянию системы, в котором протокол обратного восстановления пытается реконфигурировать систему с i ЦПУ, когда отказ является временным, и с $i-1$

ЦПУ, $i > 1$, когда отказ является постоянным. Состояние 0 представляет собой ситуацию, в которой работа системы прекращена. Стоимость обеспечения отказоустойчивости в этой архитектуре в основном определяется установлением точек восстановления. Она оценивается в 30% от мощности системы, в крайнем случае. Например, для стандартной архитектуры с n работающими процессорами и без устойчивости к отказам, которая имеет мощность равную n , с состоянием i ассоциируется темп накопления затрат r_i , $1 \leq i \leq n$, а в рассматриваемой модели – $0.7i$. Темпы накопления затрат, связанные с другими состояниями, равны нулю. При такой структуре затрат, связанной с моделью,

распределение производительности $P\{\frac{\mu_i}{\tau} > r\}$ представляет собой вероятность того, что мощность архитектуры в течение отрезка времени $[0, t]$, усредненная по времени t , больше чем r , с r , выбранным из интервала $[0, r_n]$.

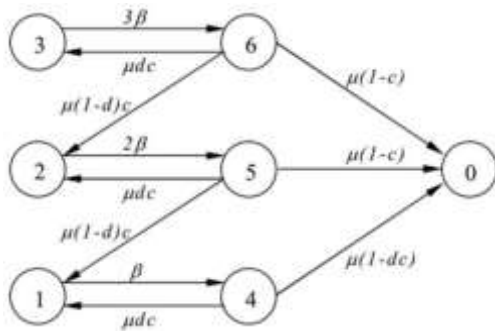


Рис. 1. Граф переходов для архитектуры с тремя ЦПУ

Сеть Петри, соответствующая описанной модели системы. Эта сеть представлена на рис. 2.

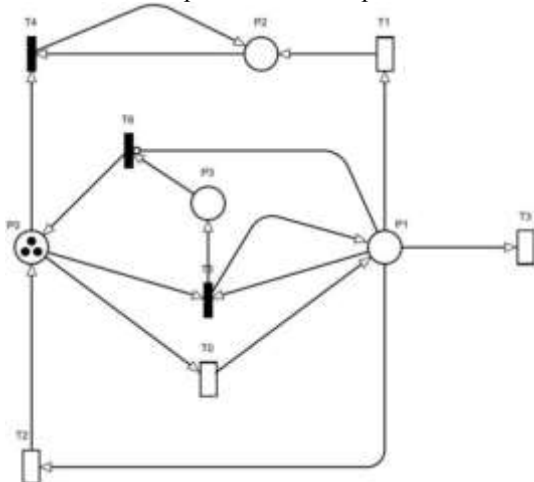


Рис. 2. Сеть Петри для моделирования многопроцессорной системы

Модель будет проиллюстрирована со следующими параметрами:

- интенсивность отказов ЦПУ $\beta = 10^{-5}$;

- вероятность обнаружения временного отказа $d = 0.9$;

- интенсивность восстановления $\mu = 1$;

- вероятность корректного переконфигурирования системы $c = 0.95$.

Для стохастических переходов получены вероятности активации и временные задержки, представленные в табл. 1.

Табл. 1. Параметры стохастических переходов

Переход	Вероятность активации, 1/с	Временная задержка, с
T0	$\mu\beta = 3 \cdot 10^{-5}$	$3.3 \cdot 10^4$
T1	$\mu(1-c) = 0.05$	20
T2	$\mu(1-d)c = 1 \cdot 0.1 \cdot 0.95 = 9.5 \cdot 10^{-2}$	10.526
T3	$\mu(1-d)c = 1 \cdot 0.1 \cdot 0.95 = 9.5 \cdot 10^{-2}$	10.526

Описанная выше сеть была составлена в разработанном приложении (рис. 3), для описания стохастических переходов были объявлены константы n, c, d, μ, β .

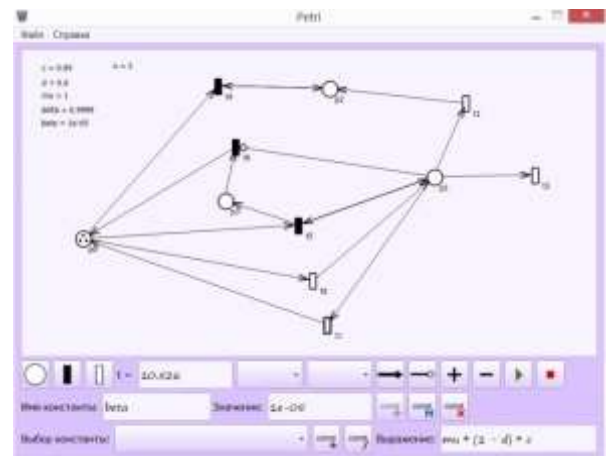


Рис. 3. Сеть Петри для моделирования многопроцессорной системы

Для вышеупомянутой сети в разработанном приложении был построен граф достижимых маркировок (рис. 4).

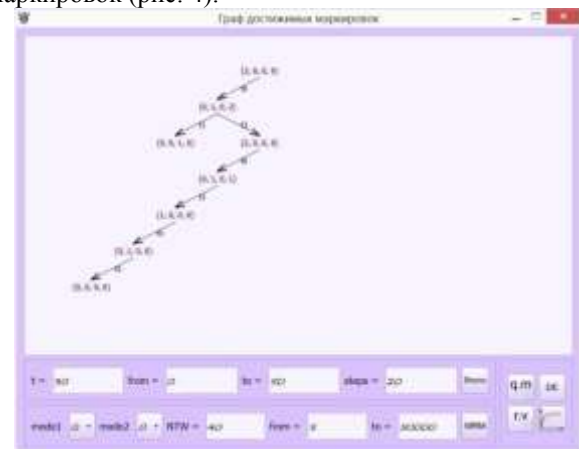


Рис. 4. Граф достижимых маркировок для моделируемой системы

Далее при помощи библиотеки `brunoshin.dll`, подключение которой описано ранее, был произведен расчёт. Для времени $t = 3000$ в выходной файл было записано число 0.913931, обозначающее вероятность того, что приведённый накопленный доход $\frac{\mu_t}{t}$ за время t превысит r . Приведение производится делением на расчётное время. Эту величину можно трактовать как среднюю производительность системы на интервале времени от 0 до t . Она ограничена и не может превышать $r_n = 2.1$.

Далее был произведен расчёт моментов с помощью библиотеки `mrmsolve.dll` в разработанном программном средстве. На основе файла, содержащего моменты меры затрат, при помощи библиотеки `distributionestimation.dll` была произведена оценка нижних и верхних границ распределения доходов. Результатом этой оценки является файл, на основании данных которого были построены графики, показанные на рис. 5.

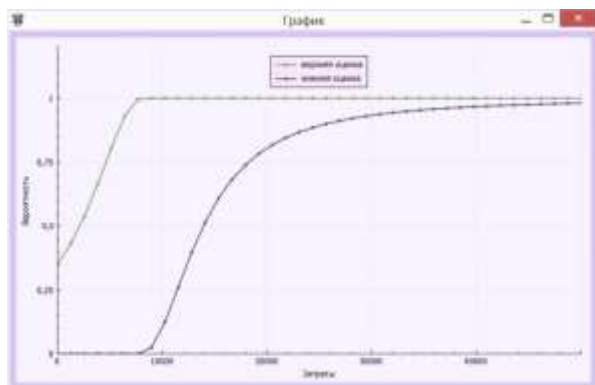


Рис. 5. Графики, отражающие нижнюю и верхнюю границы распределения доходов

ЗАКЛЮЧЕНИЕ

Разработан редактор сетей Петри с возможностями редактирования сети, эмуляции её выполнения, сохранения в файл (с возможностью последующего воспроизведения), построения соответствующих графов достижимых маркировок. С целью использования построенной модели для расчёта показателей надёжности и производительности в приложении реализованы возможности объявления констант модели, формул для расчёта интенсивностей стохастических переходов и производительности системы в возможных маркировках. Для расчета показателей надёжности и производительности моделируемых систем разработаны и оформлены в виде подключаемых динамических библиотек процедуры их расчёта.

Разработанное программное средство на данный момент стабильно работает с относительно небольшими системами, в дальнейшем оно может быть усовершенствовано для более крупных систем, а

также дополнено новыми расчётными модулями. Приложение планируется протестировать в рамках курса "Моделирование систем": предполагается разработка вариантов лабораторных работ, рассчитанных на выполнение студентами в программном средстве.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Котов В.Е. Сети Петри [Текст] / В.Е. Котов. – М.: Наука, 1984. – 160 с.
2. Лескин А.А. Сети Петри в моделировании и управлении [Текст] / А.А. Лескин, П.А. Мальцев, А.М. Спиридонов. – Л.: Наука, 1989. – 135 с.
3. Питерсон Д. Теория сетей Петри и моделирование систем [Текст] / Д. Питерсон. – науч. ред. В.А.Горбатов; пер. с англ. М.В.Горбатовой и др. – М.: Мир, 1984. – 264 с.
4. Бланшет Ж. Qt 4. Программирование GUI на C++ [Текст] / Ж. Бланшет, М. Саммерфилд – СПб.: КУДИЦ-Пресс, 2008. – 718 с.
5. Шлее М. Qt 5.10. Профессиональное программирование на C++ [Текст] / М. Шлее. – СПб.: БХВ-Петербург, 2018. – 1072 с.
6. N. Nabli and B. Sericola, "Performability Analysis: A New Algorithm", IEEE Trans. Computers, vol. 45, no. 4, pp. 491-494, Apr. 1996.

Васяшин Андрей Владимирович – старший преподаватель Отделения Интеллектуальных Кибернетических Систем, Обнинский институт атомной энергетики – филиал федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский ядерный университет «МИФИ», тел. +7(910)916-60-92, e-mail: vasiashin.andrei@yandex.ru.

Теплякова Анастасия Романовна – студентка магистратуры (направление 09.04.01 – «Информатика и вычислительная техника»), Обнинский институт атомной энергетики – филиал федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский ядерный университет «МИФИ», тел. +7(910)597-13-43, e-mail: anastasija-t23@mail.ru.

DEVELOPMENT OF AN APPLICATION FOR MODELING COMPLEX SYSTEMS WITH PETRI NETS AND CALCULATION OF THEIR RELIABILITY AND PERFORMANCE INDICATORS

A.V. Vasyashin¹, A.R. Teplyakova¹

¹ *Obninsk Institute for Nuclear Power Engineering, Obninsk*

Petri nets are an effective tool used to model complex discrete systems. Using the models represented by Petri nets, it is possible not only to understand how a particular system behaves, but also to determine the indicators of its reliability and performance. Hence, there is a need to create a convenient tool for solving such problems. This is realized through the development and integration in one software package of calculation modules and Petri nets editor. The article provides a brief description of the creation of a graphical editor of Petri nets, which is the main module of the software complex, describes the theoretical foundations laid in the basis of the developed calculation libraries, as well as the created software tool as a whole. Application of the received means to modeling and calculation of indicators of reliability and productivity of multiprocessor fault-tolerant system is also considered. The development is quite flexible, can be supplemented by other calculation modules, its use can be useful not only by specialists engaged in the calculation of reliability indicators of systems, but also by teachers and students of universities in the framework of laboratory work on the course "Modeling systems".

Index terms: Petri nets, modeling, reliability, performance.

REFERENCES

1. Kotov V.E. *Seti Petri* / V.E. Kotov. – M.: Nauka, 1984. – 160 p.
2. Leskin A.A. *Seti Petri v modelirovanii i upravlenii* / A.A. Leskin, P.A. Mal'cev, A.M. Spiridonov. – L.: Nauka, 1989. – 135 p.
3. Piterson D. *Teoriya setej Petri i modelirovanie sistem* / D. Piterson. – nauch. red. V.A.Gorbatov; per. s angl. M.V.Gorbatovoj i dr. – M.: Mir, 1984. – 264 p.
4. Blanshet Zh. Qt 4. *Programmirovaniye GUI na C++* / ZH. Blanshet, M. Sammerfeld – SPb.: KUDIC-Press, 2008. – 718 p.
5. SHlee M. Qt 5.10. *Professional'noe programmirovaniye na C++* / M. SHlee. – SPb.: BHV-Peterburg, 2018. – 1072 p.
6. N. Nabli and B. Sericola, "Performability Analysis: A New Algorithm", *IEEE Trans. Computers*, vol. 45, no. 4, pp. 491-494, Apr. 1996.

Vasyashin Andrey Vladimirovich – senior lecturer of the Department of Intelligent Cybernetic Systems, Obninsk Institute for Nuclear Power Engineering, Tel.: +7(910)916-60-92, e-mail: vasiashin.andrei@yandex.ru.

Teplyakova Anastasia Romanovna – graduate student (direction 09.04.01 - "Computer Science and Computer Engineering"), Obninsk Institute for Nuclear Power Engineering, Tel.: +7(910)597-13-43, e-mail: anastasija-t23@mail.ru.