

УДК 004.413.4; 519.252

В.В. Коробкин, А.Е. Колоденкова, В.П. Поваров

**МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ОЦЕНКИ РЕАЛИЗУЕМОСТИ СОЗДАНИЯ
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ СУПЕРКОМПЬЮТЕРОВ
НА ОСНОВЕ СИСТЕМ МАССОВОГО ОБСЛУЖИВАНИЯ**

Обсуждаются актуальные проблемы, связанные с процессом создания программного обеспечения (ПО) для суперкомпьютеров. Выявлено, что оценку реализуемости ПО необходимо проводить на ранних этапах создания ПО с целью снижения проектных рисков при его создании, а также ключевых факторов успеха и снижения возможности провала проекта по его созданию. Предлагается схема процесса проектирования ПО в терминах теории массового обслуживания без приоритетов, а также математическая модель оценки реализуемости создания ПО, под которой понимается оценка затрат времени на проектирование каждого из компонентов (подпрограмм), позволяющая оптимизировать проект по проектированию с точки зрения его затратной составляющей и сроков его выполнения, снизить возможные проектные риски при их проектировании, а также сократить управленческие ошибки, принимаемые руководителем проекта. Приводится пример оценки реализуемости создания ПО с использованием разработанной программной системы, написанной на языке C++, основным преимуществом которого является наличие большой библиотеки, позволяющей увеличить скорость разработки, а также кроссплатформенность разрабатываемой программной системы.

Оценка реализуемости создания программного обеспечения; система массового обслуживания; альтернативы программного обеспечения; время проектирования подпрограмм.

V.V. Korobkin, A.E. Kolodenkova, V.P. Povarov

**MATHEMATICAL MODEL TO ASSESS THE FEASIBILITY CREATING
SOFTWARE FOR SUPERCOMPUTERS BASED ON SYSTEMS OF MASS
SERVICE**

In the present work discusses the current problems associated with the process of building software for supercomputers. It is revealed that the assessment of the implementation of the most software should be in the early stages of creation software order to reduce the Pro-flash risks when it is created, and the key factors of success and reduce the possibilities of failure of the project to create it. Proposed scheme of the design process in terms of the theory of mass service without priorities, as well as a mathematical model to assess the feasibility of the creation, which refers to the cost estimates of time to design each of the components (subroutines) that optimizer to the project design from the point of view of its cost component and the timing of its implementation; to reduce the possible project risks in their design, as well as to reduce administrative errors made by the project Manager. Provides an example of assessing the feasibility of creating software using the developed software system, written in C++, the main advantage is the availability of a large library that allows you to increase the speed of development, and cross-platform software system.

Assessment of the feasibility of creating software; queueing system; alternative software; design time routines.

Введение. В настоящее время большую роль в техническом прогрессе играют суперкомпьютеры. Результаты высокопроизводительных вычислений на суперкомпьютерах используются во многих важнейших отраслях промышленности и медицины. Создание космических кораблей, ядерных энергетических систем, исследование ДНК, прогнозирование погоды, создание и исследование новых медицинских препаратов, изменений климата, а также проведение вычислительных экспериментов, моделирование различных процессов невозможно представить без

высокопроизводительных вычислений, которые обеспечивают получение возможности успешнее и экономически выгоднее выйти на рынок. Задачи криптографии, радиолокации, моделирования зрения и слуха, распознавания образов, математической физики, моделирования сложных систем обуславливают необходимость применения суперкомпьютеров и, что очень важно, разработку программного обеспечения (ПО) для выполнения поставленных задач [1, 2].

В связи с этим на начальных этапах создания прикладного программного обеспечения для суперкомпьютера важная роль отводится оценке реализуемости проекта по его созданию, направленной на недопущение провала проекта и снижение проектных рисков, а также на прогнозирование окончательной стоимости, сроков и качества разработки проекта ПО [3].

В настоящей работе рассматривается математическая модель оценки реализуемости создания прикладного ПО с использованием теории массового обслуживания на основе формирования и выбора его приемлемых альтернатив.

Особенности и проблемы создания программного обеспечения. Поскольку при создании ПО необходимо учитывать множество факторов, связанных с процессами его проектирования, обеспечением защиты, а также с организацией управления, то далее укажем на некоторые его особенности и проблемы [4, 5].

Проблемы организации деятельности команды исполнителей ПО обусловлены, прежде всего, нехваткой исполнителей, знающих тонкости процесса создания ПО, разработки алгоритмов и программ для конкретных прикладных систем, комплексов. Поскольку достаточно часто ошибки и дефекты в технических и программных средствах появляются из-за недостаточного понимания исполнителями процесса создания ПО, то необходимо его организовывать таким образом, чтобы это не повлияло или незначительно повлияло на весь процесс и сроки его выполнения [6].

Проблемы разработки требований создания ПО связаны прежде всего с неясностью, двусмысленностью, противоречивостью, а также частотой изменений требований заказчика. Такая ситуация приводит к разработке новых требований и внесению изменений в технические и программные средства, что, в свою очередь, ведет к задержке сдачи готового ПО. Заметим, что требования к созданию ПО, правильно сформулированные в техническом задании, обеспечивают до 40 % успеха конечного программного продукта [7].

Проблемы оценки реализуемости создания ПО напрямую связаны с процессами сбора, анализа и документирования доказательств того, что созданное ПО удовлетворяет количественным и качественным требованиям заказчика. Заметим, что оценка реализуемости создания ПО существенно усложняется наличием слабой структурированности теоретических и фактических знаний о проекте, а также сопровождением большого количества разрабатываемых вариантов создания ПО, которые сравниваются друг с другом с целью выбора из них наилучшего. Чем больше будет разработано альтернатив, тем выше будет обоснованность принимаемых решений [8, 9]. Однако *рассмотрение слишком большого числа альтернатив ведет к большим усилиям, затратам времени и путанице*. Поэтому руководители проекта, как правило, ограничиваются рассмотрением всего нескольких, наиболее приемлемых альтернатив [10, 11].

Ключевые понятия, исходные данные и допущения к оценке реализуемости создания ПО. В основе предлагаемого подхода к оценке реализуемости создания ПО лежит использование следующей обобщенной схемы, представленной на рис. 1.

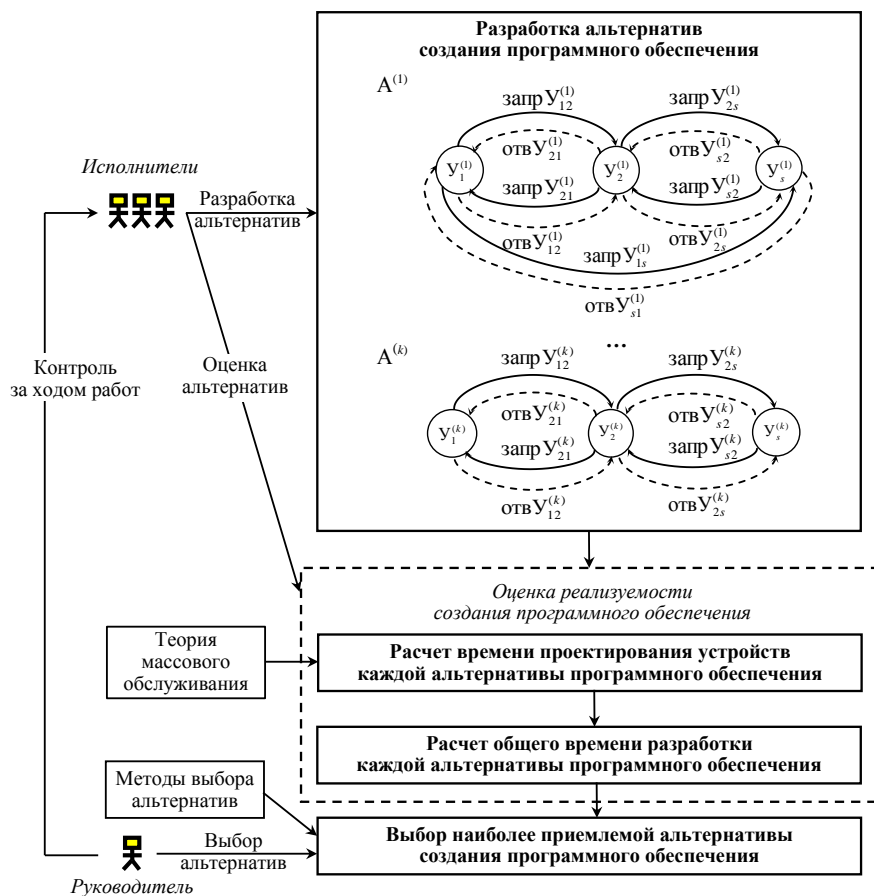


Рис. 1. Обобщенная схема оценки реализуемости создания программного обеспечения

Как видно из рис. 1, оценка реализуемости создания ПО осуществляется с использованием *теории массового обслуживания*, занимающейся анализом процессов в системах обслуживания, управления и *методов выбора альтернатив*, основанных на применении количественных и качественных методов (аппарат нечетких множеств и лингвистических переменных и др.). Под *оценкой реализуемости* понимается наличие необходимых ресурсов для осуществления и условий реализации проекта по созданию ПО.

Предполагается, что альтернативы создания ПО описываются в виде орграфа, в котором компоненты ПО изображаются в виде вершин, а запросы и ответы между компонентами, поступающие от исполнителей, в виде ориентированных ребер, при этом под ПО понимается совокупность множества программ (подпрограмм или компонентов).

В силу неоднозначности выделения количества запросов, поступающих от одного компонента ПО Y_i к другому Y_j ($i = \overline{1; s}, j = \overline{1; s}, i \neq j$), одному и тому же проекту по созданию ПО можно поставить в соответствие различные k -е альтернативы $A^{(k)}$, $k = \overline{1; m}$ (где m – число альтернатив реализации ПО).

Поскольку модели проектирования (разработки) ПО адекватны моделям массового обслуживания, то при их исследовании используются системы массового обслуживания (СМО) без приоритетов.

Схема процесса проектирования ПО без приоритетов представлена на рис. 2.

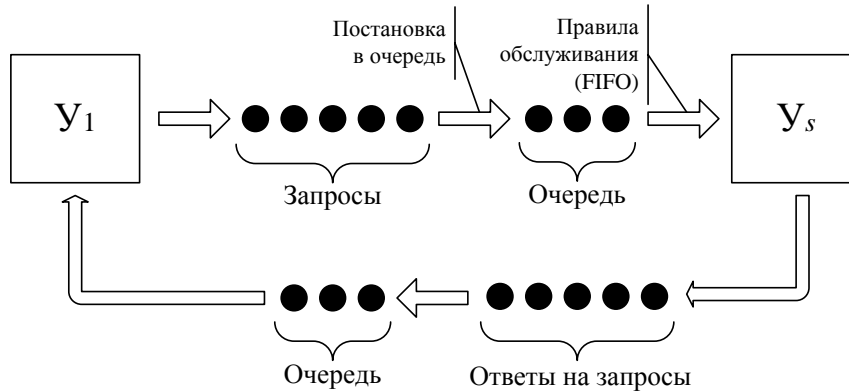


Рис. 2. Схема процесса проектирования в терминах теории массового обслуживания без приоритетов

На рис. 2 представлена СМО, моделирующая выполнение операции запроса и ответа. Поскольку потоки запросов и ответов от каждого компонента предполагаются простейшими, то данные процессы можно представить в виде простейших одноканальных СМО с неограниченной очередью $M/M/1/\infty$. Здесь каналом обслуживания является исполнитель, выполняющий ту или иную операцию. Запрос – каждое отдельное требование на выполнение какой-либо работы. Источник запросов – исполнитель, который посылает запрос другому исполнителю. Очередь – совокупность запросов, ожидающих обслуживания. Обслуживание – удовлетворение поступившего запроса на выполнение услуги. Все запросы поступают в конец очереди и первыми обслуживаются запросы, находящиеся в начале очереди (FIFO – First in-First out). Ответы на запросы – поток запросов после обслуживания.

Расчет времени, затраченного на проектирование каждого компонента $T_{Y_i}^{(k)}$

альтернативы создания ПО, и общего времени разработки каждой альтернативы $T^{(k)}$ осуществляется на основе математической модели, изложенной ниже.

Прежде, чем приступать к описанию формальной обобщенной схемы оценки реализуемости создания ПО, опишем вводимые предположения и допущения:

- ◆ альтернатива создания ПО исполнителями уже построены;
- ◆ альтернатива реализует единственное конкретное ПО;
- ◆ количество компонентов равно количеству исполнителей;
- ◆ каждый исполнитель отвечает только за проектирование одного компонента, причем исполнители не могут заменять друг друга;
- ◆ количество компонентов в каждой альтернативе создания конкретного ПО должно быть одинаковое.

Математическая модель оценки реализуемости создания ПО. Пусть имеется ограниченное количество компонентов Y_i , которые связаны между собой потоками запросов $\lambda_{\text{запр}Y_{ij}^{(k)}}$ и ответов $\lambda_{\text{отв}Y_{ji}^{(k)}}$, $i = \overline{1; s}$, $j = \overline{1; s}$, $i \neq j$. Также из-

вестно среднее время обслуживания исполнителями запросов $\bar{t}_{\text{обс.запр}Y_j}$ и среднее время обслуживания исполнителями ответов $\bar{t}_{\text{обс.отв}Y_i}$. Ставится задача нахождения времени проектирования компонентов каждой альтернативы ПО, общего времени разработки каждой альтернативы ПО, а также выбора наиболее приемлемой альтернативы создания ПО.

Для решения данной задачи предлагается алгоритм, который содержит следующие **пять этапов**.

♦ *На первом этапе* рассчитывается время, затраченное на запрос от i -го компонента к j -му $T_{\text{запр}Y_{ij}^{(k)}}$. Заметим, что для данной СМО выполняется соотношение

$$\rho_{\text{запр}Y_{ij}^{(k)}} = \lambda_{\text{запр}Y_{ij}^{(k)}} \cdot \bar{t}_{\text{обс.запр}Y_j} < 1, \quad i = \bar{1}; s, \quad j = \bar{1}; s,$$

т.е. любой запрос, который поступит, будет обслужен. Таким образом, $T_{\text{запр}Y_{ij}^{(k)}}$

может быть определено, как время пребывания заявки в системе и рассчитано по формуле Литтла [4] для СМО с неограниченной очередью:

$$T_{\text{запр}Y_{ij}^{(k)}} = \frac{\rho_{\text{запр}Y_{ij}^{(k)}}}{\lambda_{\text{запр}Y_{ij}^{(k)}} (1 - \rho_{\text{запр}Y_{ij}^{(k)}})}, \quad i = \bar{1}; s, \quad j = \bar{1}; s.$$

♦ *На втором этапе* рассчитывается время, затраченное на ответ от j -го компонента к i -му $T_{\text{отв}Y_{ji}^{(k)}}$:

$$T_{\text{отв}Y_{ji}^{(k)}} = \frac{\rho_{\text{отв}Y_{ji}^{(k)}}}{\lambda_{\text{отв}Y_{ji}^{(k)}} (1 - \rho_{\text{отв}Y_{ji}^{(k)}})}, \quad i = \bar{1}; s, \quad j = \bar{1}; s,$$

где $\rho_{\text{отв}Y_{ji}^{(k)}} = \lambda_{\text{отв}Y_{ji}^{(k)}} \cdot \bar{t}_{\text{обс.отв}Y_i} < 1$.

♦ *На третьем этапе* рассчитывается время, затрачиваемое на проектирование компонента Y_i :

$$T_{Y_i^{(k)}} = \sum_{j=1}^s T_{\text{запр}Y_{ij}^{(k)}} + \sum_{j=1}^s T_{\text{отв}Y_{ji}^{(k)}}, \quad i = \bar{1}; s, \quad j = \bar{1}; s, \quad i \neq j.$$

♦ *На четвертом этапе* рассчитывается общее время, затрачиваемое на разработку альтернативы ПО:

$$T^{(k)} = \sum_{i=1}^s T_{Y_i^{(k)}}, \quad i = \bar{1}; s.$$

♦ *На пятом этапе* осуществляется выбор приемлемой альтернативы создания ПО, для которой общее время разработки альтернативы ПО имеет наибольшее значение.

Пример оценки реализуемости создания ПО с использованием программной системы. Рассмотрим оценку реализуемости создания ПО на примере одной альтернативы. Сначала выбираются компоненты, которые необходимы для создания ПО, а затем проставляются связи между компонентами (рис. 3).

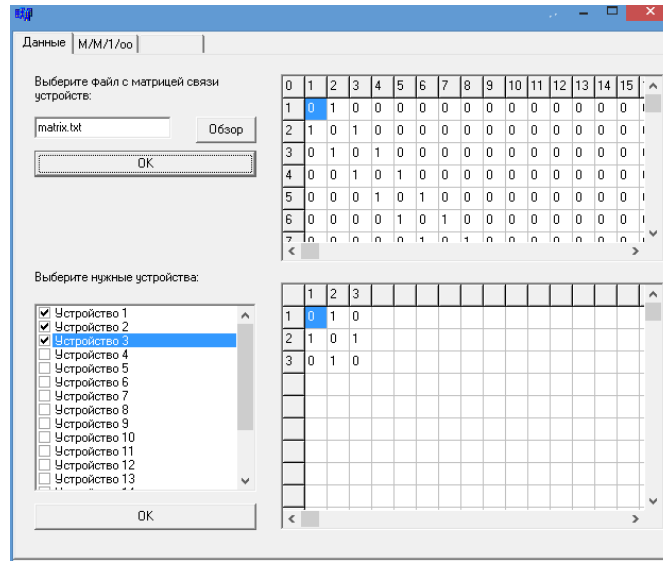


Рис. 3. Выбор компонента и связи между ними

Из рис. 3 видно, что задается файл *matrix.txt*, состоящий из единиц и нулей, где 0 обозначает отсутствие связи между двумя компонентами, а 1 – наличие связи между ними.

Далее необходимо ввести исходные данные для оценки реализуемости создания ПО (рис. 4). Здесь $L(\text{запр})[Y_{ij}]$ – интенсивность потока запросов от i -го компонента к j -му (запр/день); $L(\text{отв})[Y_{ji}]$ – интенсивность потока ответов от j -го компонента к i -му (запр/день); $T(\text{запр})[Y_j]$ – среднее время обслуживания запросов исполнителем j -го компонента (час); $T(\text{отв})[Y_i]$ – среднее время обслуживания ответов исполнителем i -го компонента (час); $T(\text{разр})[Y_{ij}]$ – время, затраченное на разработку i -го компонента, при заданных исходных данных (час).

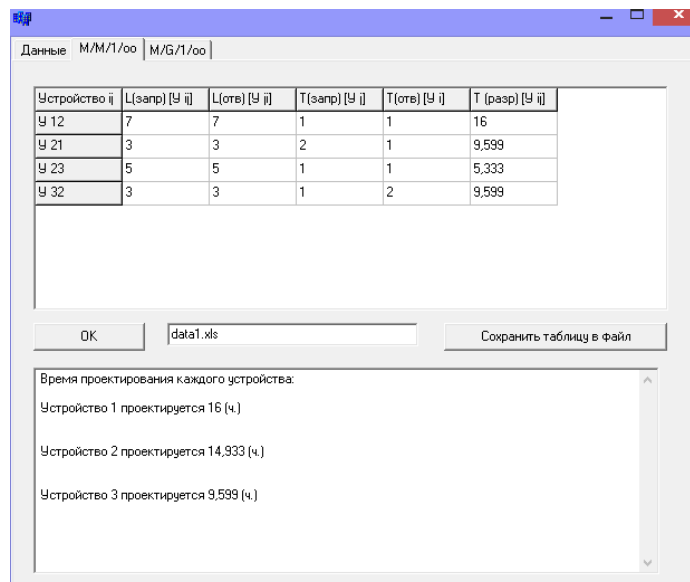


Рис. 4. Фрагмент формы ввода данных и вывода результатов

Отметим, во-первых, что у исполнителей восьмичасовой рабочий день; во-вторых, если при расчете показатель нагрузки системы для запросов и/или ответов окажется больше или равен единице, то пользователю предлагаются рекомендации по решению данной проблемы (уменьшить интенсивность потока запросов или уменьшить время обслуживания запросов).

Из рис. 4 видно, что компонент U_1 проектируется 16 часов, U_2 – 15 часов, U_3 – 10 часов, следовательно, общее время, затрачиваемое на разработку альтернативы ПО, составит 41 час.

Заключение. В настоящей работе рассмотрены особенности и основные проблемы, связанные с оценкой реализуемости создания ПО суперкомпьютеров. Предложена математическая модель оценки реализуемости создания ПО с использованием теории массового обслуживания, позволяющая повысить обоснованность принятия решения о возможности разработки нового компонента, входящего в состав ПО, и за счет этого снизить риск его неудачного завершения.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Суперкомпьютерные технологии в науке, образовании и промышленности / Под ред. академика В.А. Садовниченко, академика Г.И. Савина, чл.-корр. РАН В.В. Воеводина. – М.: Изд-во Московского университета, 2012.
2. *Каляев И.А., Левин И.И.* Многопроцессорные вычислительные системы (суперкомпьютеры): состояние и перспективы // Вестник компьютерных и информационных технологий. – 2004. – № 5. – С. 25.
3. *Дордопуло А.И., Каляев И.А., Левин И.И., Семерников Е.А.* Высокопроизводительные реконфигурируемые вычислительные системы нового поколения // В сборнике: Научный сервис в сети Интернет: экзафлопное будущее. Труды Международной суперкомпьютерной конференции. – М., 2011. – С. 42-49.
4. *Коробкин В.В., Колоденкова А.Е.* Один из подходов к оценке безопасности и рисков информационно-управляющих систем для атомных станций // XII Всероссийское совещание по проблемам управления ВСПУ-2014: труды. URL: <http://vspu2014.ipu.ru/node/8581.pdf> (дата обращения: 15.10.2014).
5. *Коробкин В.В., Колоденкова А.Е.* Детерминированный подход к оценке безопасности и рисков информационно-управляющих систем для атомных станций // Безопасность, эффективность и экономика атомной энергетики: тезисы док. Девятой Международной науч.-техн. конф. – М.: ОАО «Российский концерн по производству электрической и тепловой энергии на атомных станциях», 2014. – С. 93-94.
6. *Макконенелл С.* Сколько стоит программный проект. – М.: Русская редакция, СПб.: Питер, 2007. – 304 с.
7. *Липаев В.В.* Программная инженерия. Методологические основы: Учебник. – М.: ГУ-ВШЭ, ТЕИС. 2006. – 608 с.
8. *Беркун С.* Искусство управления IT-проектами. – СПб.: Питер, 2007. – 400 с.
9. *Вентцель Е.С.* Исследование операций. – М.: Советское радио, 1972. – 407 с.
10. *Абакаров А.Ш., Сушков Ю.А.* Программная система для выделения наилучшей альтернативы из множества имеющихся альтернатив. (MPRIORITY). Свидетельство об официальной регистрации программы для ЭВМ №2005612330 от 08 сентября 2005 г.
11. *Ногин В.Д.* Принятие решений в многокритериальной среде. Количественный подход. – М.: Физматлит, 2002. – 176 с.

REFERENCES

1. Superkomp'yuternye tekhnologii v nauke, obrazovanii i promyshlennosti [Supercomputer technologies in science, education and industry], Under the editorship of academician A. Sadovnichy, academician G. I. Savin, member of RAS Century Century Vojvodina. Moscow: Izd-vo Moskovskogo universiteta, 2012.
2. *Kalyaev I.A., Levin I.I.* Mnogoprotsessornye vychislitel'nye sistemy (superkomp'yutery): sostoyanie i perspektivy [Multiprocessor computing systems (supercomputers): status and prospects], *Vestnik komp'yuternykh i informatsionnykh tekhnologii* [Bulletin of the computer and information technology], 2004, No. 5, pp. 25.

3. *Dordopulo A.I., Kalyaev I.A., Levin I.I., Semernikov E.A.* Vysokoproizvoditel'nye rekonfiguriruemye vychislitel'nye sistemy novogo pokoleniya [High-performance reconfigurable computing system of the new generation], *V sbornike: Nauchnyy servis v seti Internet: ekzaflopsnoe budushchee. Trudy Mezhdunarodnoy superkomp'yuternoy konferentsii* [In the collection: Scientific service in the Internet: ectatomminae future. Proceedings of the International supercomputer conference]. M., 2011, pp. 42-49.
4. *Korobkin V.V., Kolodenkova A.E.* Odin iz podkhodov k otsenke bezopasnosti i riskov informatsionno-upravlyayushchikh sistem dlya atomnykh stantsiy [One of the approaches to safety assessment and risk management information systems for nuclear power plants], *XII Vserossiyskoe sove-shchanie po problemam upravleniya VSPU-2014: trudy* [XII all-Russian conference on problems of management of SPU 2014: works]. Available at: <http://vspu2014.ipu.ru/node/8581.pdf> (Accessed 15 October 2014).
5. *Korobkin V.V., Kolodenkova A.E.* Determinirovannyi podkhod k otsenke bezopasnosti i riskov informatsionno-upravlyayushchikh sistem dlya atomnykh stantsiy [A deterministic approach to safety assessment and risk management information systems for nuclear power plants], *Bezopasnost', effektivnost' i ekonomika atomnoy energetiki: tezisy dok. Devyatoy Mezhdunarodnoy nauchn.-tekhn. konf.* [Safety, efficiency and Economics of nuclear power: abstracts of the Ninth International scientific-technical conference]. Moscow: OAO «Rossiyskiy kontsern po proizvodstvu elektricheskoy i teplovoy energii na atomnykh stantsiyakh», 2014, pp. 93-94.
6. *Makkonenell S.* Skol'ko stoit programmnyy proekt [How much is a software project]. Moscow: Russkaya redaktsiya, St. Petersburg: Piter, 2007, 304 p.
7. *Lipae V.V.* Programmnyaya inzheneriya. Metodologicheskie osnovy: Uchebnik [Software engineering. Methodological framework: Tutorial]. Moscow: GU-VShE, TEIS. 2006, 608 p.
8. *Berkun S.* Iskusstvo upravleniya IT-proektami [The art of managing IT projects]. St. Petersburg: Piter, 2007, 400 p.
9. *Venttsel' E.S.* Issledovanie operatsiy [Operation research]. Moscow: Sovetskoe radio, 1972, 407 p.
10. *Abakarov A.Sh., Sushkov Yu.A.* Programmnyaya sistema dlya vydeleniya nailuchshey al'ternativy iz mnozhestva imeyushchikhsya al'ternativ [A software system for selecting the best alternative from a set of available alternatives]. (MPRIORITY). Svidetel'stvo ob ofitsi-al'noy registratsii programmy dlya EVM №2005612330 ot 08 sentyabrya 2005 g.
11. *Nogin V.D.* Prinyatie resheniy v mnogokriterial'noy srede. Kolichestvennyy podkhod [Decision making in multicriteria environment. The quantitative approach]. Moscow: Fizmatlit, 2002, 176 p.

Статью рекомендовал к опубликованию д.т.н., профессор В.В. Козляков.

Коробкин Владимир Владимирович – НИИ МВС ЮФУ; e-mail: vvk@mvs.sfedu.ru; 347928, г. Таганрог, ул. Чехова, 2; тел.: 88634311865; к.т.н.; главный конструктор по направлению; зав. лабораторией.

Колоденкова Анна Евгеньевна – Уфимский государственный авиационный технический университет; e-mail: anna82_42@mail.ru; 450000, г. Уфа, ул. К. Маркса, 12; тел.: +73472727465; кафедра технологии машиностроения; к.т.н.; доцент.

Поваров Владимир Петрович – Нововоронежская атомная станция; 396071, Воронежская область, г. Нововоронеж; к.т.н.; директор.

Korobkin Vladimir Vladimirovich – Acad. Kalyaev Scientific Research Institute of Multiprocessor Computer Systems; e-mail: vvk@mvs.sfedu.ru; 2, Chekhov street, Taganrog, 347928, Russia; phone: +78634311865; cand. of eng. sc.; chief designer in the direction; head laboratory.

Kolodenkova Anna Evgen'evna – Ufa State Aviation Technical University; e-mail: anna82_42@mail.ru; 12, K. Marksa street, Ufa, 450000, Russia; phone: +73472727465; the department of mechanical engineering; cand. of eng. sc.; associate professor.

Povarov Vladimir Petrovich – Novovoronezh nuclear power station; 396071, Voronezh region, Novovoronezh; cand. of eng. sc.; director.