

ОПРЕДЕЛЕНИЕ НАДЕЖНОСТИ СИСТЕМЫ ЭЛЕКТРОННОГО ДОКУМЕНТООБОРОТА DIRECTUM ПО РЕЗУЛЬТАТАМ ЭКСПЛУАТАЦИИ

Мостовой Я.А., Бурзуева Н.Н.

Поволжский государственный университет телекоммуникаций и информатики, Самара, РФ

E-mail: nburzueva@gmail.com

В статье рассмотрены вопросы расчета показателей надежности программного обеспечения по результатам статистической обработки наблюдений за числом проявившихся ошибок для системы электронного документооборота Directum, дорабатываемой по требованиям заказчика. Предложены и рассмотрены две дополняющие друг друга модели проявления ошибок – дискретная и непрерывная и рассчитаны методом наименьших квадратов параметры Джелинского-Моранды. Достигнутые в результате отладки системы Directum показатели надежности оценены и могут уточняться при эксплуатации. Полученные результаты позволяют прогнозировать изменения надежности системы Directum.

Ключевые слова: надежность программного обеспечения, Directum, модель Джелинского-Моранды, метод наименьших квадратов

Постановка задачи

При разработке сложных технических систем широко используются количественные показатели надежности. Эти показатели наряду с другими показателями эффективности задаются при проектировании систем и позволяют выбирать оптимальные или компромиссные проектные решения с учетом их надежности.

При разработке программного обеспечения (ПО), определяющего в настоящее время концептуальный уровень систем, широко используются методы его верификации, направленные на повышение его надежности, но расчет достигнутой надежности на практике не проводится. В работах [1-4] рассмотрены различные показатели надежности ПО и подходы к их определению и отмечается, что ни один из подходов не получил широкого распространения. Одна из причин этого связана с неразвитостью методов расчета, другая с тем, что факты и причины проявления ошибок в ПО часто не указываются разработчиком или маскируются им под проведение улучшений по требованиям заказчика, что затрудняет проведение анализа. В этой ситуации предлагается считать, что любое проведенное разработчиком формально зарегистрированное исправление кода является устранением ошибки своей или заказчика, который может устранять свою ошибку на стадии определения требований.

В то же самое время анализ достигнутых показателей надежности позволяет оценить правильность технологии разработки ПО, наметить и разработать мероприятия, позволяющие уменьшать количество ошибок, передаваемых на этап эксплуатации ПО. Это делает необходимым создание общей модели проявления ошибок в ПО, модели изменения надежности ПО с течением

времени. Эта необходимость очевидна еще и потому, что обеспечить высокую надежность ПО можно, только научившись ее измерять. При создании общей математической модели проявления ошибок в ПО необходимо учесть и придать ей конкретную индивидуальную применимость, чтобы модель содержала согласующие параметры, которые должны определяться для каждого ПО индивидуально по наблюдениям за тем, как часто проявляются ошибки в данном конкретном ПО. Вопросы расчета показателей надежности программного обеспечения рассмотрены применительно к системе электронного документооборота Directum. При этом предложено использовать две модели проявления ошибок – непрерывную и дискретную. Обе модели опираются на гипотезу Джелинского-Моранды [1-2]. Непрерывная модель «заточена» на наблюдаемом на некотором интервале времени количестве ошибок в ПО, то есть непосредственно на результаты наблюдений. По ней удобно определять параметры Джелинского-Моранды. Однако на выходе у непрерывной модели – число оставшихся в ПО ошибок, которое принимает с течением времени все меньшее и меньшее значение, но никогда не достигает нуля. Это затрудняет интерпретацию результатов расчетов.

Поэтому рассмотрена также дискретная модель проявления ошибок программного обеспечения, опирающаяся на марковский случайный процесс типа Юла-Фарри с переменной интенсивностью переходов между состояниями процесса с учетом тех же параметров Джелинского-Моранды. Эта дискретная модель позволяет, рассматривая случайный динамический процесс проявления ошибок при отладке или эксплуатации ПО, определить показатели его надежности

за требуемое время. При этом обработка статистики по проявившимся с течением времени ошибкам проводится процедурой метода наименьших квадратов [5-6] в рамках непрерывной модели.

Directum – это система электронного документооборота и управления взаимодействием производства компании Directum, нацеленная на повышение эффективности работы сотрудников организации в разных областях их совместной деятельности. Система относится к классу ЕСМ-систем (Enterprise Content Management) и поддерживает полный жизненный цикл управления документами, при этом традиционное «бумажное» делопроизводство органично вписывается в электронный документооборот. Directum обеспечивает организацию и контроль деловых процессов на основе технологии Workflow: согласование документов, обработка сложных заказов, подготовка и проведение совещаний, поддержка цикла продаж и других процессов взаимодействия [10].

Ряд организаций в процессе официального сопровождения ПО на основе платформы Directum проводят самостоятельные его доработки под потребности пользователей. Периодически данные доработки приводят к ошибкам.

С целью улучшения работы данной системы электронного документооборота была собрана статистика проявления ошибок в ПО за период времени 8 мес., проведены расчеты для определения количества оставшихся ошибок на конечный момент и для определения времени полного устранения ошибок с вероятностью 0,99.

Методы исследования

Свяжем состояния дискретного марковского случайного процесса проявлений ошибок в ПО с количеством обнаруженных ошибок [5]. Состоянию S_0 соответствует ноль обнаруженных ошибок – это начальное состояние нашего дискретного марковского процесса. Состоянию S_1 соответствует одна проявившаяся ошибка, S_2 – две; S_3 – три; S_n – n ошибок и состоянию S_{N_0} соответствует N_0 ошибок, причем всего в ПО N_0 ошибок. С течением времени все они проявятся и процесс завершится (см. рис. 1).

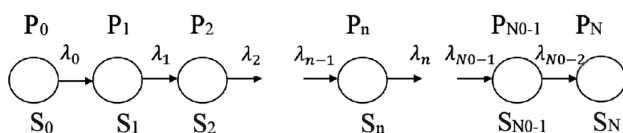


Рис. 1. Граф состояний процесса «гибели»

С каждым состоянием процесса свяжем вероятность пребывания в этом состоянии $P_i(t)$. Далее t в обозначениях вероятностей мы будем опускать, подразумевая, что P – это функции времени. Данный случайный процесс с «линейчатым» графом характерного вида носит название процесса «гибели или размножения» и широко применяется в теории надежности, теории массового обслуживания [8]. Имея граф состояний процесса, можно расписать систему уравнений Колмогорова для данного случайного процесса, которая имеет вид

$$\begin{aligned} \frac{dP_0}{dt} &= -\lambda_0 P_0 + P_{0(0)}; \\ \frac{dP_1}{dt} &= -\lambda_1 P_1 + \lambda_0 P_{0(0)}; \\ &\dots \\ \frac{dP_n}{dt} &= -\lambda_n P_n + \lambda_{n-1} P_{n-1}; \\ &\dots \\ \frac{dP_{N_0}}{dt} &= -\lambda_{N_0-1} P_{N_0-1} + P_{N_0-1}. \end{aligned} \quad (1)$$

Начальные условия для этой системы:

$$P_{0(0)} = 1, P_1 = P_2 = \dots = P_{N_0} = 0. \quad (2)$$

Нормирующее условие имеет вид

$$\sum_{n=0}^{N_0} P_n = 1. \quad (3)$$

Однако для случайного процесса проявления ошибок в чистом виде процесс «гибели» или размножения с постоянной интенсивностью проявления ошибок не совсем подходит. Практика использования ПО и физические соображения говорят, что с течением времени по мере обнаружения все большего числа ошибок в ПО и их исправления интенсивность проявления оставшихся ошибок уменьшается, то есть λ с течением времени уменьшается также. Гипотеза Желлинского-Моранды позволяет учесть этот факт, связывая, однако, переменность интенсивности проявления ошибок не со временем, а с состояниями процесса – числом оставшихся ошибок. Это позволяет нам оставаться в рамках марковских случайных процессов. Согласно указанной гипотезе [1], интенсивность проявления ошибок пропорциональна числу оставшихся ошибок:

$$\lambda_n = (N_0 - n) k, \quad (4)$$

где k – коэффициент пропорциональности; n – число проявившихся ошибок; N_0 – начальное число ошибок; $(N_0 - n)$ – число оставшихся ошибок. Запишем уравнения Колмогорова для процесса проявления ошибок и подставим в них значения интенсивности проявления ошибок:

$$\begin{aligned}\lambda_0 &= N_0 k, \lambda_1 = (N_0 - 1)k, \dots \lambda_n = \\ &= (N_0 - n)k.\end{aligned}\quad (5)$$

Процесс с зависящей от состояния процесса интенсивностью перехода λ_n , рассмотренный нами, называется случайным процессом типа Юла-Фарри [5]. Применение преобразования Лапласа для данного процесса позволяет получить изображения вероятностей его пребывания в интересующих состояниях (в нашем случае это вероятность состояния, когда все ошибки проявились). Применим последовательно преобразование Лапласа к системе уравнений Колмогорова с учетом начальных условий:

$$\begin{aligned}P_0(S) &= \frac{1}{S + k N_0}; \\ P_1(S) &= \frac{P_{n-1}(S) k (N_0 - n + 1)}{S + k (N_0 - n)}; \\ &\dots \\ P_n(S) &= \frac{P_{n-1}(S) k (N_0 - n + 1)}{S + k (N_0 - n)}; \\ &\dots \\ P_{N_0}(S) &= \frac{P_{N_0-1}(S) k}{S}.\end{aligned}\quad (6)$$

Последовательно подставляя в изображение для каждого $P_n(s)$ выражение для $P_{n-1}(s)$, имеем

$$\begin{aligned}P_n(S) &= \\ &= \frac{k^n N_0 (N_0 - 1)}{(S + k N_0) ((S + k (N_0 - 1)))} \times \\ &\times \frac{(N_0 - 2) \dots (N_0 - n + 1)}{(S + k N_0 - 2) \dots (S + k (N_0 - n))}.\end{aligned}\quad (7)$$

Разделим и числитель, и знаменатель на k^n . В случае, когда $n = N_0$, то есть когда все ошибки обнаружены, имеем изображение вида:

$$\begin{aligned}P_{N_0}(S) &= \\ &= \frac{N_0!}{s(\frac{1}{k}s + 1)(\frac{1}{k}s + 2)(\frac{1}{k}s + 3) \dots (\frac{1}{k}s + N_0)}.\end{aligned}\quad (8)$$

В достаточно полных таблицах преобразования Лапласа есть подобные изображения и оригиналы. Последнее изображение приведено к табличному виду [5]. Применим обратное преобразование Лапласа:

$$P_{N_0}(t) = (1 - e^{-kt})^{N_0}.\quad (9)$$

Соотношение (9) определяет вероятность обнаружения (проявления) всех N_0 ошибок за время t . В данной формуле N_0 и k неизвестны и, более того, они индивидуальны для каждого ПО, для каждого программиста и не могут иметь общего значения. Модель изменения надежности программного обеспечения, основанная на гипотезе Джелинского-Моранды, позволяет, основываясь на наблюдениях за процессом проявления ошибок в ПО, определить характеристику его надежности – интенсивность проявления ошибок λ , а точнее, начальное число ошибок N_0 и параметр k , определяющий скорость проявления ошибок. Решение дифференциального уравнения первого порядка относительно числа проявившихся ошибок n вытекает из записи (4):

$$n = N_0(1 - e^{-kt}).\quad (10)$$

Запишем выражение для числа ошибок, проявившихся на интервале Δn_i , полагая шаг наблюдений за проявлениями ошибок Δt равномерным

$$\begin{aligned}\Delta n_i &= N_0(1 - e^{-k t_i}) - N_0(1 - e^{-k t_{i-1}}) = \\ &= -N_0 e^{-k t_i} + N_0 e^{-k (t_i - \Delta t)} = \\ &= N_0 e^{-k t_i}(-1 + e^{k \Delta t}).\end{aligned}\quad (11)$$

Здесь Δn_i – приращение числа обнаруженных ошибок на интервале Δt . Пусть шаг Δt можно выбрать так, чтобы произведение $k \Delta t < 1$. Тогда, разлагая экспоненциальную функцию в ряд и пренебрегая членами выше первой степени $k \Delta t$, как малыми, имеем

$$\begin{aligned}\exp(x) &= 1 + \frac{x}{1!} + \frac{x^2}{2!} + \dots; \\ \Delta n_i &\approx N_0 e^{-k t_i} (1 + k \Delta t - 1) = \\ &= N_0 k \Delta t e^{-k t_i}.\end{aligned}\quad (12)$$

Для определения методом наименьших квадратов параметров N_0 и k запишем сумму квадратов невязок

$$S = \sum_{i=1}^r (\Delta n_i - N_0 k \Delta t e^{-k t_i})^2.\quad (13)$$

Найдем значение N_0 и k , при которых эта сумма имеет минимум. Для этого приравняем нулю производные суммы по искомым параметрам:

$$\frac{\partial S}{\partial N_0} = \sum_{i=1}^r 2 (\Delta n_i - N_0 k \Delta t e^{-kt_i}) \times (-k \Delta t e^{-kt_i}) = 0; \quad (14)$$

$$\sum_{i=1}^r (\Delta n_i - N_0 k \Delta t e^{-kt_i}) e^{-kt_i} = 0;$$

$$\frac{\partial S}{\partial k} = \sum_{i=1}^r 2 (\Delta n_i - N_0 k \Delta t e^{-kt_i}) \times (-N_0 \Delta t e^{-kt_i} + N_0 k t_i \Delta t e^{-kt_i}) = 0;$$

$$\sum_{i=1}^r (\Delta n_i - N_0 k \Delta t e^{-kt_i}) \times e^{-kt_i} (k t_i - 1) = 0.$$

Окончательно запишем полученную систему двух уравнений с двумя неизвестными N_0 и k .

$$\sum_{i=1}^r (\Delta n_i - N_0 k \Delta t e^{-kt_i}) t_i e^{-kt_i} = 0; \quad (7)$$

$$\sum_{i=1}^r (\Delta n_i - N_0 k \Delta t e^{-kt_i}) e^{-kt_i} = 0. \quad (8)$$

Выразим из первого уравнения N_0 , раскрывая скобки и вынося постоянные, не зависящие от индекса i , за знак суммы:

$$\sum_{i=1}^r \Delta n_i t_i e^{-k t_i} - N_0 k \Delta t \sum_{i=1}^r (e^{-k t_i} t_i) = 0; \quad (15)$$

$$N_0 = \frac{\sum_{i=1}^r (\Delta n_i t_i e^{-k t_i})}{k \Delta t \sum_{i=1}^r (e^{-k t_i} t_i)}. \quad (16)$$

Подставляя это выражение во второе уравнение, после преобразований имеем

$$\sum_{i=1}^r (\Delta n_i e^{-k t_i}) = \frac{\sum_{i=1}^r \Delta n_i t_i e^{-k t_i}}{\sum_{i=1}^r t_i e^{-2k t_i}} \times \sum_{i=1}^r e^{-2k t_i}. \quad (17)$$

Мы получили уравнение с одним неизвестным k , которое может быть решено численно.

Отладка ПО должна проводиться по плану и завершаться по выполнению намеченного плана. Модель Джелинского-Моранды может независимо и достаточно достоверно оценивать ход процесса отладки, который целесообразно прекращать, когда интенсивность проявления ошибок достигла через определенное время очень низкого значения.

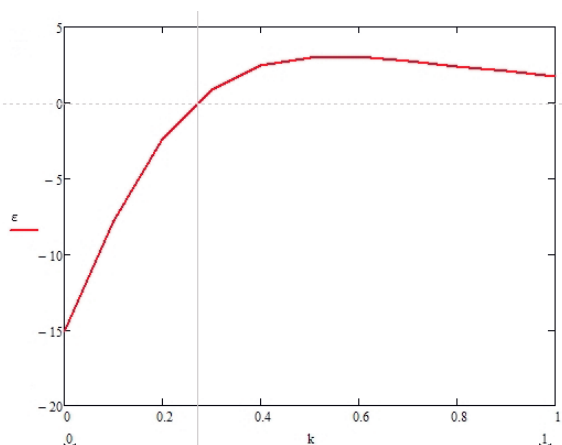
Однако есть два момента, которые надо рассмотреть, чтобы получить ответ на вопрос о том, когда можно прекращать отладку. Во-первых, рассмотренная модель аппроксимации ошибок экспонентой является непрерывной и может показывать дробное число ошибок на заданный момент времени, в том числе и значение числа оставшихся ошибок меньше единицы. Во-вторых, модель никогда не покажет отсутствие оставшихся ошибок – их число будет приближаться к нулю бесконечно долго. Это не соответствует физическому смыслу и может считаться недостатком рассмотренной непрерывной модели надежности ПО. Выше нами рассмотрена дискретная модель с непрерывным временем для определения вероятности проявления n ошибок, построенная на базе случайного процесса «гибели», также с использованием параметров Джелинского-Моранды, и была получена формула (9), определяющая вероятность обнаружения всех ошибок из N_0 , имеющихся в ПО.

Задавшись вероятностью обнаружения всех ошибок в ПО, мы можем определить время, для которого данная вероятность будет достигнута. Это позволяет ответить на вопрос, сколько времени надо продолжать отладку – столько, чтобы вероятность отсутствия ошибок была, например, 0,999 или любым другим заданным числом.

Результаты исследования

Для данной работы была собрана статистика возникновения ошибок для доработок программного обеспечения системы Directum за восемь месяцев.

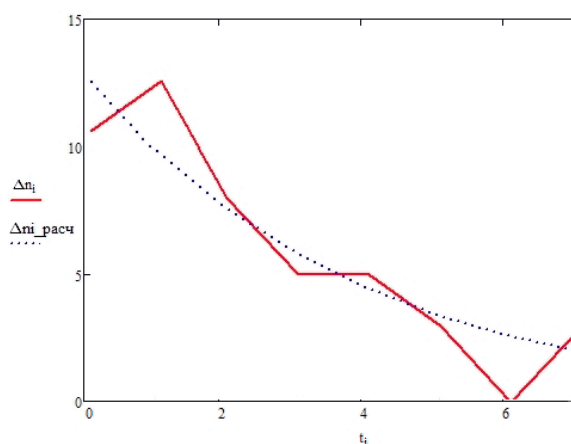
Для нахождения коэффициента пропорциональности была организована итерационная численная процедура вычисления левой и правой частей равенства (17) при различных значениях k . Процедура прекращается и k считается найденным, как только разница между вычисленной левой и правой частями равенства, которую обозначим ε , станет меньше заданного малого числа, например, $\varepsilon = 10^{-6}$.

Рис. 2. График изменения функции $\varepsilon(k)$

На рис. 2 представлен график функции

$$\varepsilon(k) = \left(\sum_{i=1}^r e^{-2kt_i} \right) \frac{\sum_{i=1}^r \Delta n_i e^{-kt_i} t_i}{\sum_{i=1}^r e^{-2kt_i} t_i} - \sum_{i=1}^r \Delta n_i e^{-kt_i},$$

которая для $k = 0,26$ практически равна 0. Полученное значение k подставим в (15) и получим результат $N_0 = 195,994$. Проведен расчет сглаженных методом наименьших квадратов значений Δn_i по определенным N_0 и k и формуле (13) и построен совместный график изменений Δn_i по результатам наблюдений и сглаженных методом наименьших квадратов в функции t_i (см. рис. 3).

Рис. 3. Аппроксимация методом наименьших квадратов исходных данных Δn_i расчетными данными $\Delta n_{расч}$

Задавшись вероятностью обнаружения всех ошибок в ПО $P_{N_0} = 0,99$ по формуле (9) определим время, за которое ПО Directum достигнет заданной надежности: это время составляет $t = 23,857$ месяца.

Выводы

Рассмотрены две модели для оценки надежности ПО: непрерывная относительно числа проявившихся ошибок за время t и дискретная относительно вероятности проявления n ошибок на момент времени t . Обе модели опираются на одинаковые параметры Джелинского-Моранды, которые определены по непрерывной модели методом наименьших квадратов. Проведены расчеты по собранным статистическим данным и формально зарегистрированным изменениям кода. Изменение по времени количества проявившихся ошибок позволяет сделать вывод, что все ошибки, внесенные при доработках системы электронного документооборота Directum, будут устранены через $23,857 \approx 24$ месяца с вероятностью $P_{N_0} = 0,99$. Необходимы тщательный подход к проведению отладки доработок ПО Directum с использованием моделей операционной среды и более тщательный и формальный подход к регистрации изменений кода ПО Directum.

Литература

1. Майерс Г. Надежность программного обеспечения. М.: Мир, 1980. – 360с.
2. Благодатских В.А., Волнин В.А., Посакалов К.Ф. Стандартизация разработки программных средств. М.: Финансы и статистика, 2005. – 288 с.
3. Романюк С.Г. Оценка надежности ПО // Открытые системы. №4, 1994. – С. 95-104.
4. Василенко И.В., Макаров В.А. Модели оценки надежности программного обеспечения // Вестник НовГУ. №28, 2004. – С. 126-132.
5. Козлов Д.И., Аншаков Г.П., Мостовой Я.А., Соллогуб А.В. Управление космическими аппаратами зондирования Земли. Компьютерные технологии. М.: Машиностроение, 1998. – 368 с.
6. Мостовой Я.А. Лекции по технологии разработки программного обеспечения. Самара: Изд-во ПГУТИ, 2014. – 178 с.
7. Вентцель Е.С. Исследование операций. Задачи, принципы, методология. М.: Дрофа, 2004. – 208 с.
8. Сандлер Дж. Техника надежности систем. М.: Наука, 1966. – 300 с.
9. Дейч Г. Руководство к практическому применению преобразования Лапласа. М.: Наука, 1965. – 288 с.
10. Directum. Wikipedia.org // URL: <https://ru.wikipedia.org/wiki/Directum> (д.о. 07.03.2016)

11. О системе Directum. Электронный документооборот и управление взаимодействием // URL: <http://www.directum.ru/system> (д.о.10.05.2016). tadviser.ru/index.php/Продукт: Directum_ (СЭД/ЕСМ-система) (д.о. 10.05.2016).
12. Directum (СЭД/ЕСМ-система). Tadviser. Государство. Бизнес. ИТ // URL: [http://www.tadviser.ru/index.php/Продукт: Directum_ \(СЭД/ЕСМ-система\) \(д.о. 10.05.2016\).](http://www.tadviser.ru/index.php/Продукт: Directum_ (СЭД/ЕСМ-система) (д.о. 10.05.2016).) Получено 20.05.2016

Мостовой Яков Анатольевич, д.т.н., профессор, лауреат Государственной премии СССР, профессор Кафедры программного обеспечения и управления в технических системах (ПОУТС) Поволжского государственного университета телекоммуникаций и информатики (ПГУТИ). Тел. 8-917-114-78-85. E-mail: jakob.mostovoi@yandex.ru

Бурзуева Наталья Новрузовна, аспирант Кафедры ПОУТС ПГУТИ. Тел. 8-917-957-29-10. E-mail: nburzueva@gmail.com

EVALUATION OF RELIABILITY OF ENTERPRISE CONTENT MANAGEMENT SYSTEM DIRECTUM BY ANALYSIS OF OPERATION RESULTS

Mostovoi J.A., Burzueva N.N.

Povolzhsky State University of Telecommunication and Informatics, Samara, Russian Federation

E-mail: nburzueva@gmail.com

Modern reliability theory widely uses quantitative indicators. Those indicators together with other efficiency indexes are applied during system design processes. They provide to choose optimal or compromise design solutions under taking into account their reliability. During software development that defines system concept level a number of methods for its verification are used. These methods are focused on reliability improvement. However on practice, estimation of achieved reliability is not performed. This work is concerned with estimation of software reliability indicators by analysis of results of appeared error statistical processing for enterprise content management system Directum updated according to customer requirements. We propose two complementary models of error appearing: discrete and continues under computing of Jelinski-Moranda parameters by the least square method. Reliability indicators been achieved because of Directum system debugging were evaluated and may be refined during its operation. Presented results provide prediction of Directum system reliability indicator variations.

Keywords: software reliability, Directum, Jelinski-Moranda model, least square method

DOI: 10.18469/ikt.2016.14.3.08

Mostovoi Jakob Anatolevich, Povolzhskiy State University of Telecommunications and Informatics, 77 Moscovskoe shosse, Samara, Russian Federation; Professor of the Department of Software and Control in Technical Systems, Doctor of Technical Sciences, Professor. Tel.: +79171147885. E-mail: jakob.mostovoi@yandex.ru

Burzueva Natalia Novruzovna, Povolzhskiy State University of Telecommunications and Informatics, 77 Moscovskoe shosse, Samara, Russian Federation; PhD-student of the Department of Software and Control in Technical Systems. Tel.: +79179572910. E-mail: nburzueva@gmail.com

References

1. Mayers G. *Nadozhnost' programmnoy obespecheniya* [The reliability of the software]. Moscow. Mir Publ., 1980. 360 p.
2. Blagodatskikh V.A. *Standartizatsiya razrabotki programmnykh sredstv*. Uchebnoye posobiye [Standardization of software development. Tutorial]. Moscow. Finansy i statistika Publ., 2005. 288 p.
3. Romanyuk S.N. Otsenka nadozhnosti PO [Reliability Evaluation Software]. *Otkrytyye sistemy – Open Systems*, 1994, no. 4, pp. 95-104.
4. Vasilenko I.V., Makarov V.A. Modeli otsenki nadozhnosti programmnoy obespecheniya [Models of evaluating the reliability of software]. *Vestnik Novgorodskogo gosudarstvennogo universiteta*, 2004, no. 28, pp. 126-132

5. Kozlov D.I. *Upravleniye kosmicheskimi apparatami zondirovaniya Zemli* [Earth sensing space-craft Management]. Moscow. Mashinostroyeniye Publ., 1998. 368 p.
6. Mostovoi J.A *Leksii po tekhnologii razrabotki programmnogo obespecheniya. Uchebnoye posobiye* [Lectures on software development technology. Tutorial]. Samara, PSUTI Publ., 2014. 178 p.
7. Venttsel E.S. *Issledovaniye operatsiy. Zadachi, printsipy, metodologiya: Uchebnoye posobiye* [Operations research. Objectives, principles, methodology: Textbook. Manual for schools.] Moscow, Drofa Publ., 2004. 208 p.
8. Sandler J. *Tekhnika nadozhnosti system* [Reliability Engineering systems.] Moscow, Nauka Publ., 1966. 300 p.
9. Dech Gustav. *Rukovodstvo k prakticheskomu primeneniyu preobrazovaniya Laplasa* [Guide to the practical application of the Laplace transform]. Moscow, Nauka Publ., 1965. 288 p.
10. Directum. Available at: <https://ru.wikipedia.org/wiki/Directum> (accessed 07.03.2016)
11. O sisteme Directum [About Directum system]. Available at: <http://www.directum.ru/system> (accessed 10.05.2016)
12. Directum (SED/ECM-sistema). Available at: [http://www.tadviser.ru/index.php/Продукт:Directum_\(СЭД/ECM-система\)](http://www.tadviser.ru/index.php/Продукт:Directum_(СЭД/ECM-система)) (accessed 10.05.2016)

Received 20.05.2016

УДК 681.5

НЕЙРОСЕТЕВОЙ АНАЛИЗ ДИНАМИЧЕСКИ НЕРАВНОВЕСНЫХ СИТУАЦИЙ ВЗАИМОДЕЙСТВИЯ СЕКТОРОВ ЭКОНОМИКИ С ИСПОЛЬЗОВАНИЕМ ИМИТАЦИОННОЙ МОДЕЛИ

Димов Э.М.¹, Ильясов Б.Г.², Макарова Е.А.², Ефтонова Т.А.²

¹Поволжский государственный университет телекоммуникаций и информатики, Самара, РФ

²Уфимский государственный авиационный технический университет, Уфа, РФ

E-mail: ea-makarova@mail.ru

Предложена структура трехуровневой системы управления взаимодействием секторов в составе многосекторной макроэкономической системы с использованием имитационной модели. Разработана процедура формирования интеллектуальных алгоритмов ситуационного управления на основе нейронных сетей Кохонена. Построены самоорганизующиеся карты для кластеров динамически неравновесных ситуаций.

Ключевые слова: ситуационное управление, имитационная модель, нейронные сети, самоорганизующиеся карты, кластер, динамически неравновесная ситуация

Введение

Современные условия развития национальной экономики характеризуются высокой степенью неопределенности и динамизма, что связано с усилением борьбы за ограниченные ресурсы, возрастанием конкуренции, продолжением негативных тенденций поляризации государств по уровню экономического развития. Это, в свою очередь, обуславливает возрастание сложности и необходимости решения проблем прогнозирования и стратегического планирования национальной экономики. Для решения этой проблемы разрабатываются модельные комплексы и программные инструменты, предназначенные для анализа экономических ситуаций на макро-, мезо- и микроуровнях. Одним из известных подходов к разработке таких инструментариев явля-

ется подход, основанный на методе Димова-Маслова (МДМ), согласно которому моделирующий комплекс должен включать имитационную модель функционирования социально-экономической системы, встроенную в контур управления реальной СЭС [1]. Имитационная модель в этом случае является, с одной стороны, средством проведения сценарных исследований с целью формирования управляющих решений, а с другой стороны, средством уточнения сгенерированной по определенному сценарию цепочки управляющих воздействий как по их значениям, так и по времени их реализации [2-4].

В статье рассматриваются вопросы, связанные с решением проблемы разработки системы имитационного моделирования и интеллектуального управления процессом взаимодействия