

**Н.Г. Топольский, А.В. Крючков, Д.С. Грачёв, К.А. Михайлов, Нгуен Ле Зуй**  
(Россия, Вьетнам)

(Академия ГПС МЧС России; e-mail: mihkir.94@mail.ru)

## МОДЕЛЬ ОЦЕНКИ ВЕРОЯТНОСТИ РЕАЛИЗАЦИИ СПЕЦИАЛЬНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ АВТОМАТИЗИРОВАННЫХ СИСТЕМ ПОЖАРОВЗРЫВОБЕЗОПАСНОСТИ НЕФТЕПРОИЗВОДСТВ

Рассмотрена проблема повышения устойчивости специального программного обеспечения автоматизированных систем пожаровзрывоопасности нефтеперерабатывающих производств. Предложено оценить вероятность реализации специального программного обеспечения на инструментальном средстве. Рассмотрена оценка влияния общего количества единичных программных систем при синтезе специального программного обеспечения традиционными методами.

Ключевые слова: специальное программное обеспечение, автоматизированная система пожаровзрывобезопасности, нефтеперерабатывающие предприятия, единичная программная система, устойчивость специального программного обеспечения.

**Нефтеперерабатывающие предприятия (НПП)** являются объектами повышенной экологической, пожарной и иной опасности. Поэтому при их эксплуатации необходимо соблюдение норм и правил всех видов (но в особенности пожарной) безопасности. Так как НПП – это очень крупные и территориально разнесённые объекты, включающие в себя отдельные элементы инфраструктуры населённых пунктов, в которых они расположены, то для обеспечения их безопасности и мониторинга опасных ситуаций на них необходимо создание отдельной инфраструктурной единицы – системы управления безопасностью. Концепция **автоматизированной системы пожаровзрывобезопасности (АСПВБ)** объекта была предложена в [1]. В настоящее время её развитие в условиях применения большого количества различных **информационных технологий (ИТ)** привело к необходимости анализа применяемого в ней **специального программного обеспечения (СПО)** на предмет значимости влияния на него человеческого фактора в ходе его синтеза.

Общие правила автоматизации разработки и синтеза СПО для **автоматизированных систем управления технологическими процессами (АСУТП)** НПП были разработаны в [2]. Но в настоящее время интерес представляют расчёты "количества влияния", основанные на применении параметров трудоёмкости синтеза СПО. АСПВБ в равной степени может считаться как АСУТП, так и **информационной системой (ИС)**, в связи с тем, что большая часть процессов в ней автоматизирована и работает с хранимыми в ней данными.

При синтезе СПО первой частью процесса считается фаза анализа требований к системе. В ходе этой фазы определяются иерархические требования к СПО АСПВБ, которые математически могут быть представлены в виде графа. На рёбрах этого графа могут быть заданы значения различных параметров.

С их помощью определяются функции на графе. При этом для каждого из операторов АСПВБ можно выделить свою часть СПО, которую условимся называть **единичной программной системой (ЕПС)**.

Наиболее интересной была бы функция оценки влияния человеческого фактора на устойчивость СПО АСПВБ [1, 2]. Однако, в силу того, что различные по смыслу и цели итоговые критерии эффективности в нужной оценке имеют одинаковую математическую интерпретацию, рассмотрим их со следующих позиций. Для построения критерия эффективности реализации СПО АСПВБ на инструментальном средстве посредством формального определения её функциональности построим на графе иерархии требований к СПО функцию оценки трудозатрат. А затем интерпретируем полученные результаты в нужном смысле.

Согласно [4], АСПВБ можно рассматривать как состоящую из технических средств, на которых хранится информация информационного фонда и процедур обработки информации. Процедуры обработки информации есть СПО. Они включают в себя управляющую процедуру, процедуру обновления информационного фонда, информационного поиска и завершающей обработки [4]. Эксплуатация АСПВБ связана с выполнением элементарных процедур обновления и поиска информации, а также её завершающей обработки.

Продолжая обсуждать проблему в тех же терминах, получаем, что для определения математической модели АСПВБ необходимо принять во внимание, что каждая ИС, входящая в её состав, предполагает наличие четырёх языков:

- языка вопросов;
- языка ответов;
- языка формулирования требований к предметной области;
- языка формулирования сообщений о необходимости изменения вида завершающей обработки информации.

Кроме этого, необходимо учитывать множество состояний информационного фонда. Таким образом, окружение АСПВБ может рассматриваться как суперпозиция следующих множеств:

$$\left\{ \begin{array}{l} P = \{p_i\}; \\ R = \{r_i\}; \\ Q = \{q_i\}; \\ S = \{s_i\}; \\ T = \{t_i\} \end{array} \right\}, \quad (1)$$

где  $P, R, Q, S$  – перечисленные выше соответствующие языки;

$p_i$  – вопрос;

$r_i$  – ответ;

$q_i$  – сообщение об обновлении информации;

$s_i$  – сообщение о виде завершающей обработки;

$T$  – множество состояний информационного фонда.

При этом

$$P \cap Q \cap S = \emptyset. \quad (2)$$

Иными словами, такие события как вопрос, ответ и сообщение о необходимости изменения вида завершающей информации не происходят одновременно. Целью применения в АСПВБ процедур СПО является получение необходимой информации. Поэтому необходимо ввести понятия запросов к АСПВБ и получения элемента информации. Они могут быть определены при помощи следующих формул:

$$P \cap Q \cap S = \emptyset, \quad (3)$$

где  $Z$  – это множество запросов к СПО АСПВБ и

$$i_n = f(p_i, t_i), \quad (4)$$

где  $i_n$  – элемент информации АСПВБ, выдаваемый АСПВБ по вопросу  $p_i$ , учитывая состояние информационного фонда  $t_i$ ;

$f$  – поисковая функция.

Формулой (4) определяется процедура информационного поиска. С понятием "ответ" связана процедура завершающей обработки, которая может быть определена как получение значения функции:

$$r_i = \Omega(p_i, i_{ni}), \quad (5)$$

где  $p_i, r_i$  – вопрос и ответ;

$i_{ni}$  – элемент информации;

$\Omega$  – текущий вид завершающей обработки информации.

Процедура обновления информации может быть определена как

$$\varphi(q_i, t_i), \quad (6)$$

где  $q_i$  – сообщение об обновлении информации;

$t_i$  – текущее состояние  $i$ -го элемента информационного фонда.

Процедура изменения вида завершающей обработки информации может быть представлена функцией:

$$\psi(s_i, \varphi), \quad (7)$$

где  $s_i$  – сообщение о виде завершающей обработки информации.

Если обозначить совокупность состояния информационного фонда и четырёх введённых выше функций как  $J$ , или

$$J = (t_i, f, \Omega, \varphi, \psi), \quad (8)$$

то управляющая процедура в СПО АСПВБ может быть определена как

$$W = (Z, J), \quad (9)$$

где  $Z$  – множество запросов к АСПВБ (12).

Таким образом, каждая ИС в АСПВБ может быть определена как кортеж ИС – совокупность состояний информационного фонда и пяти функций:

$$ИС = \langle t, f, \Omega, \varphi, \psi, W \rangle. \quad (10)$$

Как было показано выше, такое определение ИС в АСПВБ относится в равной степени и к ЕПС. Оно составляет основу для построения множества ИС в АСПВБ, на котором будет определена функция, определяющая количественный порядок трудозатрат на синтез СПО АСПВБ. На уровне **автоматизированных рабочих мест (АРМ)** (3-й уровень графа из графа иерархии требований к СПО) количество ЕПС, описываемых формулой (10), значительно более 1000. Соответственно всё СПО АСПВБ задаёт дискретное множество вида

$$\left\{ \begin{array}{l} ИС_1 = (t_1, f_1, \Omega_1, \Phi_1, \Psi_1, W_1); \\ ИС_2 = (t_2, f_2, \Omega_2, \Phi_2, \Psi_2, W_2); \\ \dots \\ ИС_n = (t_n, f_n, \Omega_n, \Phi_n, \Psi_n, W_n) \end{array} \right\} = R_{АСПВБ}, \quad (11)$$

где  $R_{АСПВБ}$  – собственно СПО, при этом в (11) свыше 100 строк.

С другой стороны, любая ИС в АСПВБ, определённая как (10), состоит из более "мелких" программ (или процедур). Поэтому функция оценки трудозатрат на графе иерархии требований, заданная на множестве (11), может иметь вид

$$Z = f_{зам}(R_{ис}), \quad (12)$$

где  $R_{ис}$  – СПО конкретной ИС, заданное как (10);

$f_{зам}$  – многомерная функция трудозатрат;

$Z$  – количественно выраженные общие трудозатраты при создании  $R_{ис}$ .

Функция (12) может быть определена в одном из вариантов как сумма времени на реализацию групп **базовых элементов программ (БЭП)**, входящих в ЕПС для  $R_{ис}$ . Для её точного определения понадобится многомерная матрица, соответствующая графу иерархии требований к СПО в АСПВБ ( $R_{ис}$ ), на последнем уровне которой в двумерной матрице вместо групп функциональности будет использовано время их синтеза. Исходя из (11), параметр функции (12) может быть представлен в виде многомерной матрицы, аналогом (11), в состав элементов которой необходимо будет включить векторы или двумерные матрицы, значениями которых будут трудоёмкости создания БЭП, а сама функция будет её свёрткой. Дальнейшее уточнение значений (12) в терминах **объектно-ориентированного языка программирования (ООЯП)** приведёт к обчёту трудоёмкостей по созданию большого количества объектов для ЕПС, связанных с состоянием информационного фонда и методов их работы.

Известно, что описания объектов в ООЯП и описания конфигураций ЕПС отличаются при синтезе их разными абстрактными программистами и при синтезе их одним абстрактным программистом, а также при применении ими разных версий одинаковых инструментальных средств. Поэтому для (12) в каждой ЕПС (10) необходимо ввести характеристики, требования и допуски, как на производстве.

Для них используем следующие формулы:

$$\left\{ \begin{array}{l} M^{(i)}_1 = (\mu^{(i)}_{11}, \mu^{(i)}_{12}, \dots, \mu^{(i)}_{1n1}); \\ M^{(i)}_2 = (\mu^{(i)}_{21}, \mu^{(i)}_{22}, \dots, \mu^{(i)}_{2n2}); \\ \dots \\ M^{(i)}_m = (\mu^{(i)}_{m1}, \mu^{(i)}_{m2}, \dots, \mu^{(i)}_{mnk}) \end{array} \right\} = T_{ИС}^l, \quad (13)$$

где  $T_{ИС}^l$  – трудоёмкость синтеза СПО в  $l$ -й ИС АСПВБ;

$M^{(i)}_h$  – набор трудоёмкостей реализации наборов характеристик для  $n$  объектов ЕПС (СПО) в  $h$ -м инструментальном средстве;

$\mu^{(i)}_{jk}$  – трудоёмкость реализации  $k$ -й характеристики  $j$ -го объекта.

При этом для каждого набора трудоёмкостей следует ввести допуск, определяемый по формуле

$$\Delta^{(i)} = (\delta^{(i)}_1, \delta^{(i)}_2, \dots, \delta^{(i)}_n), \quad (14)$$

где  $\Delta^{(i)}$  – набор допусков по трудоёмкости реализации  $i$ -й ЕПС в  $h$ -м инструментальном средстве;

$\delta^{(i)}$  – допуск по трудоёмкости при реализации  $i$ -й ЕПС в  $h$ -м инструментальном средстве при синтезе  $k$ -й характеристики.

В (13)  $j$ -м объектом является ЕПС, для которой реализация  $k$ -й характеристики определяется по формуле

$$\left\{ \begin{array}{l} R^{(i)}_1 = (r^{(i)}_{11}, r^{(i)}_{12}, \dots, r^{(i)}_{1n1}); \\ R^{(i)}_2 = (r^{(i)}_{21}, r^{(i)}_{22}, \dots, r^{(i)}_{2n2}); \\ \dots \\ R^{(i)}_m = (r^{(i)}_{m1}, r^{(i)}_{m2}, \dots, r^{(i)}_{mnk}) \end{array} \right\} = \text{ЕПС}, \quad (15)$$

где  $R^{(i)}_j$  – набор трудоёмкостей реализации характеристик объектов в ЕПС;

$r^{(i)}_{jk}$  – трудоёмкость реализации  $k$ -й характеристики  $j$ -го объекта.

В соответствии с (13)-(15), можно говорить об отличиях (невязках) трудоёмкостей при реализации конкретной ЕПС на одном из инструментальных средств. Их можно обобщённо вычислить по формулам:

$$\varepsilon_j^{(i)} = \sqrt{\sum_{k=1}^n (r_{jk}^{(i)} - \mu_{jk}^{(i)})^2}, \quad (16)$$

$$\varepsilon_j^{(sj)} \rightarrow \min \{\varepsilon_j^{(i)}\}, \quad (17)$$

где  $\varepsilon_j^{(i)}$  – невязка трудоёмкостей при реализации  $j$ -го объекта конкретной ЕПС в  $i$ -м инструментальном средстве.

Таким образом, для обобщения всех требований к ЕПС необходимо минимизировать невязку  $\varepsilon_j^{(i)}$  по каждому её объекту с тем, чтобы выполнялось условие:

$$\varepsilon_j^{(i)} \leq |\delta_j^{(i)}|. \quad (18)$$

Исходя из условий (17) и (18), следует ожидать, что многомерная функция (12), определённая на пространстве (11), будет находится рядом с минимальным значением в пределах допусков (14). Значения данной функции могут использоваться в качестве искомого критерия эффективности для оценки. При этом очевидно, что наиболее простым способом снижения её значения может быть снижение размерности множеств или снижение значений некоторых элементов матриц, содержащих время создания БЭП в инструментарии. И то и другое возможно при применении повторно используемых компонент

ЕПС в рамках описанных выше моделей. Применяя аналогию в расчётах, можно говорить, что снижения влияния человеческого фактора на устойчивость СПО в АСПВБ можно добиться этими же методами.

Рассмотрим формулу расчёта условной вероятности для приёма партии продукции отделом технического контроля [5]. По аналогии с ней можно определить для каждого из объектов ЕПС вероятность её корректной реализации на инструментальном средстве при заданных допусках:

$$P_i(\Omega_m, N_i, l, k_m) = \sum_{m=1}^l \sum_{j=1}^{\Omega_m} (C_k^j C_{N-k}^{l-j} / C_N^l); \quad (19)$$

$$(k_m / N_i) < \Omega_m,$$

где  $P_i$  – вероятность корректной реализации  $i$ -й ЕПС на конкретном инструментальном средстве или конкретным абстрактным программистом;

$\Omega_m$  – допуск по  $m$ -й характеристике;

$N_i$  – общее число реализуемых характеристик ЕПС в  $i$ -й ЕПС;

$l$  – число тестируемых характеристик ЕПС в  $i$ -й ЕПС;

$k_m$  – число характеристик ЕПС в  $i$ -й ЕПС, не соответствующих (18).

Формула (19) говорит о том, что вероятность корректной реализации  $i$ -й ЕПС на конкретном инструментальном средстве или конкретным абстрактным программистом меньше 1, если часть характеристик ЕПС не будет реализована или не будет удовлетворять условиям тестирования. Значит, даже очень простые по функциональности ЕПС могут быть в некоторых случаях не реализованы. Следовательно, общая устойчивость СПО АСПВБ будет ниже заданной изначально при окончательном выполнении работ по его созданию.

Теперь рассмотрим оценку того, как общее число ЕПС может повлиять на его устойчивость при его синтезе традиционными методами. Согласно теории вероятностей, вероятность одновременного наступления двух независимых событий равна произведению вероятностей этих событий. Если рассматривать реализации в СПО всех описанных в техническом задании функциональных характеристик для всех предметных областей как независимые, то вероятность их совместной реализации равна произведению вероятностей, получаемых по (19) для конкретных ЕПС. Оценивая граф иерархии требований в 100 вершин на уровне АРМ, можно уверенно оценить вероятность его реализации. Пусть вероятность реализации одного из ЕПС достаточно близка к 1 и равна 0,99. Это означает, что только 1 из 100 требований (а их может быть около 100) не реализовано или работает неправильно. Тогда общая вероятность синтеза 100 ЕПС в АСПВБ равна 100-й степени числа 0,99. Это примерно 0,43. Но если более точно определять такую вероятность, то следует учесть как изменение в инструментальных средствах в процессе синтеза, так и возможную (а в реальности происходящую всегда) смену разработчиков. Тогда эту цифру нужно дважды умножить на саму себя, что даст величину вероятности устойчивой работы СПО для АСПВБ равную 0,0795. Фактически это означает, что применение существующих методов синтеза СПО для АСПВБ при применении для неё нового релиза СПО (например, на новой платформе) сделает её работу не только неустойчивой, а просто невозможной, учитывая те последствия, которые несёт система, работающая с вероятностью меньшей 0,5, на пожаровзрывоопасном производстве.

Вероятностная оценка процесса синтеза СПО может задаваться матрицей, элементами которой являются найденные по (19) вероятности корректной реализации  $i$ -й ЕПС на конкретном инструментальном средстве или абстрактным программистом. Для задания допусков при синтезе конкретной ЕПС на одном из инструментальных средств необходимо учитывать искажения, вносимые в БЭП абстрактным программистом. Увеличение их количества и невязок трудоёмкостей при реализации по (18) снижают вероятность реализации проекта по синтезу СПО в целом, что подтверждается как экспериментальными данными, так и приведённым расчётом.

Таким образом, вероятностный анализ функции оценки, построенной на матрицах оценки трудоёмкости (устойчивости СПО) показывает, что для получения нужного по качеству значения функции оценки необходимо добиться того, чтобы реализация в СПО подавляющего большинства указанных в ТЗ требований осуществлялась с вероятностью равной 1. Это фактически означает необходимость повторного использования хорошо отлаженного кода БЭП. Только для них можно уверенно гарантировать такую высокую вероятность реализации требований.

Задача решения проблемы повышения устойчивости СПО АСПВБ в сформулирована как поиск на графе иерархии требований базового графа функциональности. Этот граф отображается на априори реализованную ЕПС, вероятность реализации которой для устойчивости СПО АСПВБ уже должна быть равна 1. Это с математической точки зрения подтверждает общее направление рассуждений по построению системы методов повышения устойчивости СПО.

Итак, проведённый анализ показал, что для повышения устойчивости СПО АСПВБ и вероятности его реализации, а также снижения влияния на АСПВБ человеческого фактора (в том числе числа обслуживающих СПО специалистов высокой квалификации) необходимо:

- выстроить базовый граф реализации функциональности ЕПС;
- построить на его основе один или несколько компонентов, реализующих эту функциональность на инструментальном средстве.

#### Литература

1. Топольский Н.Г. Основы автоматизированных систем пожаровзрывобезопасности объектов. М.: МИПБ МВД России, 1997. 164 с.
2. Абросимов А.А., Топольский Н.Г., Федоров А.В. Автоматизированные системы пожаровзрывобезопасности нефтеперерабатывающих производств. М.: Академия ГПС МВД России, 2000. 239 с.
3. Ильин В.Д. Система порождения программ: монография. М.: Наука, 1989. 264 с.
4. Криницкий Н.А., Миронов Г.А., Фролов Г.Д. Автоматизированные информационные системы. М.: Наука, Главная редакция физико-математической литературы, 1982. 381 с.
5. Левин А.И. Методические основы управления конфигурацией // Матер. VI междунар. науч.-практ. конференции "Применение ИПИИ-технологий для повышения качества и конкурентоспособности наукоёмкой продукции". М., 2004.
6. Вендров А.М. Проектирование программного обеспечения экономических информационных систем. М.: Финансы и статистика, 2005. 544 с.
7. Избачков Ю.С., Петров В.Н. Информационные системы: учеб. для вузов. СПб.: Питер, 2005. 655 с.
8. Басс Л., Клементс П., Кацман Р. Архитектура программного обеспечения на практике, пер. с англ. СПб.: Питер, 2006. 575 с.
9. Лефингуэлл Д., Уидриг Д. Принципы работы с требованиями к программному обеспечению. Унифицированный подход, пер. с англ. М.: Вильямс, 2002. 448 с.

Статья поступила в редакцию интернет-журнала 11 ноября 2017 г.

**N.G. Topolsky, A.V. Kruchkov, D.S. Grachev, K.A. Mikhaylov, Nguyen Le Zuy  
(Russia, Vietnam)**

## **MODEL OF ESTIMATION OF THE PROBABILITY OF SPECIAL SOFTWARE FOR COMPUTER-AIDED FIRE-EXPLOSION SAFETY SYSTEMS OF PETROLEUM REFINERY INDUSTRIES**

It deals with the problem of increasing the stability and probability of implementing special software for computer-aided fire-explosion safety systems of petroleum refinery industries.

The study uses methods of set theory, probability theory.

Probabilistic analysis of the evaluation function based on the evaluation matrices of labor intensity (special software stability) shows that in order to obtain the desired value of the evaluation function, it is necessary to ensure that the implementation of the requirements in the special software is carried out with a probability equal to 1. This means that you must reuse the well-debugged code of the underlying program elements. Only for them it is possible to guarantee such high probability of requirements implementation.

The obtained results allow estimate probability of implementation of the special software of computer-aided fire-explosion safety systems.

The analysis showed that in order to increase the stability and probability of implementing special software for computer-aided fire-explosion safety systems of petroleum refinery industries it is necessary:

- to build a basic graph of the implementation of the functionality of single software system;
- to build it on the basis of one or more components that implement this functionality of the tool.

Key words: special software, computer-aided fire-explosion safety system, petroleum refinery industries, single software system, special software stability.

### **References**

1. Topolskiy N.G. *Osnovy avtomatizirovannykh sistem pozharovzryvbezopasnosti obyektov* [Bases of the computer-aided fire-explosion safety systems of objects]. Moscow, Moscow Fire Safety Institute of Internal Affairs of Russia Publ., 1997, 164 p.
2. Abrosimov A.A., Topolskiy N.G., Fedorov A.V. *Avtomatizirovannyye sistemy pozharovzryvbezopasnosti neftepererabatyvayushchikh proizvodstv* [Computer-aided fire-explosion safety systems of petroleum refinery industries]. Moscow, Academy of State Fire Service of Emercom of Russia Publ., 2000, 239 p.
3. Ilin V.D. *Sistema porozhdeniya programm* [Programs generation system]. Moscow, Science Publ., 1989, 264 p.
4. Krinitskiy N.A., Mironov G.A., Frolov G.D. *Avtomatizirovannyye informatsionnyye sistemy* [Computer-aided information systems]. Moscow, Science Publ., 1982, 381 c.
5. Levin A.I. *Metodicheskiye osnovy upravleniya konfiguratsiyey* [Methodological basics of configuration management]. *Mater. VI mezhdunar. nauch.-prakt. konferentsii "Primeneniye IPI-tekhnologiy dlya povysheniya kachestva i konkurentosposobnosti naukoemkoy produktsii"* [Proceed. of the materials of the VI international scientific and practical conference "Application of IPI technologies to improve the quality and competitiveness of high-tech products"], Moscow, 2004.
6. Vendrov A.M. *Proyektirovaniye programmnoy obespecheniya ekonomicheskikh informatsionnykh sistem* [The software design of economic information systems]. Moscow, Finansy i statistika Publ., 2005, 544 c.
7. Izbachkov Yu.S., Petrov V.N. *Informatsionnyye sistemy* [Information systems]. Saint Petersburg, Piter Publ., 2005, 655 p.
8. Bass L., Klements P., Katsman R. *Arkhitektura programmnoy obespecheniya na praktike* [Software architecture in practice]. Saint Petersburg, Piter Publ., 2006, 575 c.
9. Lefinguell D., Uidrig D. *Printsipy raboty s trebovaniyami k programmnomu obespecheniyu. Unifitsirovanny podkhod* [Principles of work with software requirements. Unified approach]. Moscow, Viliams Publ., 2002, 448 c.