УДК 004.94:681.5.015.23

© 2021 г. **О.В. Абрамов**, д-р техн. наук,

Д.А. Назаров, канд. техн. наук

(Институт автоматизации и процессов управления ДВО РАН, Владивосток)

ОБЛАЧНЫЕ ТЕХНОЛОГИИ ДЛЯ РЕШЕНИЯ ЗАДАЧ ПОСТРОЕНИЯ И АНАЛИЗА ОБЛАСТЕЙ РАБОТОСПОСОБНОСТИ

Рассматриваются вопросы использования облачных технологий при решении задач построения и анализа областей работоспособности технических систем. В работе утверждается, что наиболее целесообразной архитектурой для консолидации высокопроизводительных вычислительных ресурсов, программных средств моделирования и хранилищ данных при многопользовательской работе является облачная платформа. Обсуждаются основные компоненты облачной среды для построения и анализа ОР в рамках моделей облачных сервисов SaaS и DaaS.

Ключевые слова: САПР, облачные вычисления, надежность, область работоспособности.

DOI: 10.22250/isu.2021.70.53-66

Введение

Проблема построения и исследования областей работоспособности (ОР) обычно возникает на этапе параметрического синтеза при проектировании технических систем. Эта проблема связана с решением ряда задач проектирования, таких как определение допусков параметров и выбор их номинальных значений, оценка чувствительности системы к параметрическим возмущениям и параметрической надежности. ОР представляет собой ограниченное подмножество точек пространства управляемых параметров элементов исследуемой системы, в каждой точке которого эта система находится в работоспособном состоянии. Получение характеристик ОР или ее аппроксимация другими фигурами может существенно облегчить решение проектных задач, связанных с необходимостью выполнения многократных расчетов

модели проектируемой системы, и позволит применить геометрические методы ее исследования для определения диапазонов допустимой вариации параметров.

Понятие ОР обычно относится к функционально-параметрическому направлению теории надежности [1, 2], когда исследуется модель системы, связывающая ее выходные параметры $y = (y_1, y_2, \dots, y_m)$ с параметрами ее элементов (внутренних параметров) $x = (x_1, x_2, \dots, x_n)$ в виде зависимостей:

$$y_i = y_i(\mathbf{x}), i = 1, 2, \dots, m. \quad (1)$$

Выходные параметры представляют характеристики системы, интересующие потребителя, которые должны удовлетворять требованиям, указанным в техническом задании, и обычно выражающиеся как ограничения:

$$y_{i \min} \leq y_i(\mathbf{x}) \leq y_{i \max}, \forall i = 1, 2, \dots, m. \quad (2)$$

Ограничения (2) определяют работоспособное состояние системы, поэтому называются условиями работоспособности (УР), а их нарушение считается отказом системы. К нарушению УР может привести неудачное проектное решение либо дрейф параметров элементов, возникающий под влиянием факторов внешней среды и внутренних процессов износа и старения. Дрейф внутренних параметров обычно рассматривается внутри заданных ограничений, – например, бруса допусков на их физическую реализуемость:

$$x_{i \min} \leq x_i \leq x_{i \max}, \forall i = 1, 2, \dots, n. \quad (3)$$

Одним из способов исследования влияния отклонений внутренних параметров на работоспособность системы является определение характеристик области в пространстве этих параметров, ограниченном брусом (3), для которых будут выполняться УР (2):

$$D_x = \{\mathbf{x} \in \mathbf{R}^n \mid y_{i \min} \leq y_i(\mathbf{x}) \leq y_{i \max}, \forall i = 1, 2, \dots, m\}. \quad (4)$$

Область D_x , определяемая выражением (4), называется областью работоспособности в пространстве параметров элементов системы, заданной моделью (1) при заданных УР (2).

Процедура построения ОР связана с высокими вычислительными затратами, вызванными многократным моделированием системы при многомерном зондировании, вычислительной сложностью модели и высокой размерностью пространства. Такой подход требует не только высокопроизводительного оборудования и технологий параллельных вычислений, но и ресурсов для хранения больших объемов данных, с обеспечением коллективного доступа к ним. Таким образом, целесообразным решением является ориентация системы построения и анализа ОР [3] на архитектуру облачного сервиса [4]. Помимо задач, облачная архитектура позволяет решить следующие про-

блемы, возникающие при консолидации вычислительных ресурсов и хранилища данных, при развертывании и поддержке настольных систем:

внедрение и квалифицированная поддержка специализированного программного обеспечения, которое может потребоваться в связи с необходимостью моделирования конкретных технических систем при построении ОР (например, SPICE для моделирования электрических цепей [5]);

обеспечение совместного доступа к данным ОР при решении задач проектирования для различных специалистов, с учетом их ролей и полномочий. Как будет описано далее, хранение данных ОР, полученных методом многомерного зондирования, является наиболее критичным аспектом использования ОР, требующим колоссальных ресурсов. Одной из задач облачной среды является предоставление возможности использования информации об ОР для решения задач проектирования в самой среде без необходимости копирования больших объемов данных.

Из указанного выше следует, что рассматриваемая облачная среда реализует две модели облачных сервисов – «Software as a Service» (SaaS) и «Data as a Service» (DaaS) [4]. На рис. 1 проиллюстрирована структурная схема облачного сервиса построения и анализа ОР (ОСПАОР).

Рис. 1. Общая структура ОСПАОР.

Пользователь осуществляет взаимодействие с ресурсами среды через веб-портал, обладающий функциями аутентификации пользователя и его авторизации для решения определенного круга задач. В зависимости от имеющихся у пользователя прав, ему доступны в различной мере программные сервисы построения и анализа ОР, а также непосредственно фонды построенных ранее ОР для загрузки на рабочую станцию. Важным элементом ОСПАОР как облачной среды является блок администрирования пользовате-

лей и пользовательских групп, развернутый на веб-портале. Перед описанием ролей пользователей и их полномочий рассмотрим основные принципы построения и анализа ОР, используемые в системе, и основанные на них модули программной среды.

Модули облачного сервиса построения и анализа ОР

Архитектура облачной среды построения и анализа ОР во многом схожа с модульной структурой настольной системы нахождения и использования ОР (СНИОР) [3], однако реализация ее в виде облачного сервиса добавляет в архитектуру дополнительные функции, главным образом касающиеся разделения ролей пользователей и прав доступа при совместном использовании ресурсов платформы. Каждый модуль системы выполняет определенные функции в рамках программного комплекса и имеет различную степень автономии. Некоторые модули выделяются только функционально, являясь отдельными звеньями единого процесса, работающими в определенной последовательности. На рис. 2 изображены основные модули СНИОР, входящие в ОСПАОР, и потоки данных, протекающие между ними.

Рис. 2. Модульная структура СНИОР.

Подсистема ввода модели

Модуль используется при построении ОР и предназначен для ввода пользователем модели системы (1). Этот программный компонент имеет расширяемый функционал для различных способов ввода описания модели. В настольной версии СНИОР преимущественным способом ввода модели было подключение динамически присоединяемых библиотек (DLL – для операционных систем семейства Microsoft Windows, SO – для операционных систем семейства Linux) с описывающими ее функциями [3]. Эти библиотеки содержат исполняемый программный код и несут потенциальную угрозу безопасности рабочей станции при получении их из непроверенных или неизвестных источников. Такого рода уязвимость особенно критична для систем

коллективного пользования, поэтому добавление в общие фонды моделей систем, заданных в динамически присоединяемых библиотеках, должна быть строго ограничено и выполняться пользователями с административными привилегиями после тщательной проверки источников или исходного текста программы.

Наиболее безопасным для программных систем коллективного пользования, в частности облачной инфраструктуры, является механизм ввода символического описания модели с ограниченным синтаксисом, состоящим из математических функций и базовых управляющих операторов. Для разбора такого описания модели необходим синтаксический анализатор математических выражений, – например, «muParser» [6] или «QScriptEngne» [7].

Не менее важной функциональной особенностью модуля является также использование сторонних программных средств моделирования, – например, имитация работы электрических цепей с помощью программных средств семейства SPICE [8]. Для этого в модуле должны учитываться особенности взаимодействия со сторонними программными моделирующими средствами. С точки зрения программного комплекса, модуль, взаимодействуя с блоком моделирования, предоставляет ему унифицированный интерфейс к модели системы (1), задаваемой с помощью различных средств. Этот интерфейс определяет базовый перечень необходимых функций, которые должны входить в описание модели:

- вычисления значений всех выходных параметров (1);
- получения количества n внутренних параметров;
- получения количества m выходных характеристик;
- перечисления заданных ограничений (3) внутренних параметров;
- перечисления УР (2).

Блок моделирования

Модуль предназначен для выполнения однократного расчета модели с целью вычисления бинарного признака выполнения УР для заданного набора внутренних параметров $\mathbf{x} = (x_1, x_2, \dots, x_n)$. Результат работы модуля можно представить в виде характеристической функции:

$$F_y(\mathbf{x}) = \begin{cases} 1, & y_{imin} \leq y_i(\mathbf{x}) \leq y_{imax}, \forall i = 1, 2, \dots, m, \\ 0, & \text{иначе} . \end{cases} \quad (5)$$

На вход этому модулю подается вектор внутренних параметров $\mathbf{x} = (x_1, x_2, \dots, x_n)$, а на выходе считывается бинарный результат проверки всех выходных характеристик для указанных внутренних параметров в соответствии с выражением (2).

Блок построения ОР

Модуль выполняет построение ОР-системы, заданной моделью (1), при заданных ограничениях на выходные характеристики (2) и внутренние параметры (3). Построение ОР заключается в определении характеристик непосредственно области (4) или нахождении многомерной фигуры, аппроксимирующей ее. Обзор существующих методов построения ОР широко представлен в работе [9].

В работе используется представление ОР дискретным множеством элементарных гиперпараллелепипедов, заданных узлами регулярной сетки на основе результатов многомерного зондирования. Далее такое представление ОР будем называть дискретным представлением ОР (ДПОР). Его основой является метод матричных испытаний, представляющий собой подход к исследованию надежности систем на основе физического моделирования радиоэлектронной аппаратуры с последовательным изменением (обычно дискретным, с определенным шагом) значений внутренних параметров или включением в схему элементов с различными номиналами [10].

В соответствии с методом матричных испытаний диапазон (3) допустимых значений каждого i -го параметра разбивается эквидистантно на q_i интервалов-квантов. Каждый квант i -го параметра пронумеруем индексом $k_i = 1, 2, \dots, q_i$, $i = 1, 2, \dots, n$. Границы квантов образуют сетку с шагом по каждой координатной оси h_i , равным:

$$h_i = (x_{i\max} - x_{i\min}) / q_i. \quad (6)$$

Используя шаг сетки (6), обозначим левую и правую границы k_i -го кванта соответственно:

$$\begin{aligned} a_i^{k_i} &= x_{i\min} + h_i \cdot (k_i - 1), \\ b_i^{k_i} &= a_i^{k_i} + h_i = x_{i\min} + h_i \cdot k_i. \end{aligned} \quad (7)$$

В центре каждого кванта, заданного границами (7), выбирается его точка-представитель с координатами:

$$c_i^{k_i} = (a_i^{k_i} + b_i^{k_i}) / 2 = a_i^{k_i} + h_i / 2. \quad (8)$$

Таким образом, набор индексов квантов $\mathbf{K} = (k_1, k_2, \dots, k_n)$ однозначно определяет элементарный гиперпараллелепипед (ЭГ) e_K с координатами его границ (7) и его точки-представителя $\mathbf{x}_c$ с координатами (8):

$$\mathbf{x}_c(k_1, k_2, \dots, k_n) = (c_1^{k_1}, c_2^{k_2}, \dots, c_n^{k_n}). \quad (9)$$

Согласно методу матричных испытаний значения выходных характеристик (1) во всех остальных точках внутри квантов, заданных индексами $(k_1, k_2, \dots, k_n)$, принимаются равными выходным параметрам в точке-

Следовательно, характеристическая функция (5), вычисленная в точке-представителе каждого ЭГ, определяет его принадлежность к фигуре, аппроксимирующей ОР. Таким образом, характеристическая функция (5) задает разбиение множества $\mathbf{E}$ всех ЭГ:

$$E = E_1 \cup E_0, E_1 \cap E_0 = \emptyset,$$

где $E = \{e_k, \forall k_i = 1, 2, \dots, q_i, \forall i = 1, 2, \dots, n\}$ – множество всех ЭГ; E_1 и E_2 – его подмножества такие, что:

$$E_1 = \{e_{k_1 k_2 \dots k_n} \in E : F_y(\mathbf{x}_c(k_1, k_2, \dots, k_n)) = 1\}, \quad (10)$$

$$E_0 = \{e_{k_1 k_2 \dots k_n} \in E : F_y(\mathbf{x}_c(k_1, k_2, \dots, k_n)) = 0\}.$$

Подмножество E_1 содержит ЭГ, составляющие многомерную фигуру, которая аппроксимирует ОР, т.е. это множество представляет ДПОР. Для хранения информации о принадлежности каждого ЭГ множеству E_1 или E_0 введем множество индикаторов (МИ), однозначно связанных с ЭГ и имеющих значения характеристической функции (5) в соответствии с выражениями (10):

$$S = (s_1, s_2, \dots, s_R), s_i \in \{0, 1\}, \forall i = 1, 2, \dots, R, \quad (11)$$

где $R = q_1 \cdot q_2 \cdot \dots \cdot q_n$ – количество ЭГ. Между индексом p произвольного элемента МИ и набором индексов $(k_1, k_2, \dots, k_n)$ соответствующего ему ЭГ существует взаимно однозначное соответствие [11]. Значение индекса p индикатора принадлежности ЭГ, заданного индексами $(k_1, k_2, \dots, k_n)$, вычисляется как

$$p = k_1 + q_1(k_2 - 1) + q_1q_2(k_3 - 1) + \dots + q_1q_2\dots q_{n-1}(k_n - 1). \quad (12)$$

Таким образом, принадлежность каждого ЭГ множеству E_0 или E_1 определяется его индикатором в соответствии с выражением:

$$s_p = F_y(\mathbf{x}_c(k_1, k_2, \dots, k_n)). \quad (13)$$

Описанный выше метод построения ДПОР наглядно проиллюстрирован на примере двумерного пространства параметров на рис. 3.

Рис. 3. Дискретное представление ОР в двумерном случае.

Резюмируя указанное выше: структура данных, описывающая ДПОР, задается моделью:

$$G_R = (n, B, Q, S), \quad (14)$$

где n – размерность пространства внутренних параметров; $B = \{(x_{i \min}, x_{i \max}), i = 1, 2, \dots, n\}$ – ограничивающий область построения брус; $Q = (q_1, q_2, \dots, q_n)$ – набор показателей количества квантов для каждого параметра; $S = (s_1, s_2, \dots, s_R)$ – множество индикаторов (11).

Алгоритм построения ДПОР состоит в вычислении характеристической функции (5) для каждого ЭГ и записью ее значения в МИ в соответствии с (13). Процедура вычисления каждого индикатора принадлежности не зависит от вычисления других, что позволяет использовать модель параллельных вычислений типа SPMD (Single Program Multiple Data). Использование технологии параллельных вычислений при построении ДПОР обусловливается необходимостью многократного моделирования системы в узлах сетки, что может потребовать большого количества времени. Разделение данных для обработки их на разных вычислительных узлах в параллельном режиме существенно сокращает время построения ДПОР. Разбиение МИ на части для параллельной обработки может выполняться с учетом неравномерной загрузки вычислительных узлов в случае использования несимметричной гетерогенной параллельной вычислительной системы [3, 12].

Модуль сжатия данных ДПОР

Модуль предназначен для уменьшения занимаемого объема данных ДПОР. Важно отметить, что этот модуль используется не только для сжатия данных при построении ОР, но и для работы со сжатыми данными при решении задач анализа ОР. Как видно из описания модели (14), основной объем данных занимает множество индикаторов S . Оно состоит из $R = q_1 \cdot q_2 \cdot \dots \cdot q_n$ элементов, принимающих значения «0» или «1». Эти данные представляют последовательно записанные одномерные сечения ДПОР, что позволяет предположить наличие длинных подпоследовательностей индикаторов с одинаковыми значениями [13]. Такую структуру данных можно оптимизировать путем кодирования длин серий, что дает возможность уменьшить занимаемый массивом индикаторов объем. Например, исходное МИ на рис. 3, имеющее вид: $S = (1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1)$, будет записано в виде: $S_{RLE} = ((2, 1), (3, 0), (2, 1), (3, 0), (3, 1), (2, 0), (5, 1), (1, 0), (4, 1))$. В результате вместо 25 цифр исходного МИ S будет получен массив S_{RLE} , фактически состоящий из 18 цифр, представляющих 9 пар чисел, кодирующих серию повторяющихся элементов массива S . Такой способ сжатия данных без потери

информации называется алгоритмом кодирования длин серий Run-Length Encoding (RLE) [14]. Обозначим элемент сжатого массива S_{RLE} как s_c^i :

$$s_c^i = (l^i, v^i), i = 1, 2, \dots, R_{RLE}, \quad (15)$$

где l_i – длина серии повторяющихся значений v_i ; R_{RLE} – количество кодирующих элементов массива S_{RLE} .

Существенным требованием к алгоритму сжатия данных ДПОР является сохранение высокой скорости доступа к ним без полной их распаковки. Так, при использовании несжатого МИ доступ к произвольному i -му элементу s_i осуществляется по смещению адреса (индекса элемента) относительно начала массива, а алгоритмическая сложность этой операции оценивается как $O(1)$. Для доступа к этому же элементу в сжатом массиве S_{RLE} требуется определить индекс k кодирующего его элемента s_c^k (15). Поиск номера этого элемента состоит в последовательном переборе элементов S_{RLE} , с вычислением суммы длин серий: $L_1 = l_1$, $L_2 = l_1 + l_2$, $L_3 = l_1 + l_2 + l_3 = L_2 + l_3$, ..., и далее:

$$L_k = \sum_{j=1}^k l_j, k = 1, 2, \dots, R_{RLE} \text{ или } L_k = L_{k-1} + l_k, k = 2, 3, \dots, R_{RLE}, \quad (16)$$

пока не будет выполнено условие:

$$L_k > i.$$

Таким образом, алгоритмическая сложность доступа к произвольному индикатору принадлежности в сжатом массиве составляет $O(N)$. Время доступа к каждому из них возрастает с увеличением значений индексов квантов ($k_1, k_2, \dots, k_n$) кратно произведению величин q_i в соответствии с выражением (12). Одним из решений проблемы замедления доступа к «дальним» элементам МИ может быть введение дополнительных меток индексов, условно разбивающих сжатый массив S_{RLE} на небольшое число T интервалов. Это позволяет ускорить доступ к произвольному элементу МИ приблизительно (с поправкой на время поиска нужного интервала) в T раз, избегая суммирования длин серий путем перебора (16).

Подсистема анализа ОР

Главной целью построения ОР является решение задачи оптимального параметрического синтеза, а также ряда других задач проектирования, связанных с анализом области допустимых значений внутренних параметров. Таким образом, компонент ОСПАОР для анализа ОР представляет многомодульную расширяемую подсистему, состоящую из встраиваемых подпрограмм, реализующих отдельные задачи анализа ОР. Ввиду многопользовательской архитектуры, фонды доступных программных модулей анализа ОР

могут различаться в зависимости от специфики решаемых отдельными разработчиками задач. Каждая исследовательская группа или отдельные инженеры по согласованию с администратором системы могут включать специфические модули анализа ОР. Главным условием добавления подпрограммы анализа в фонд, помимо вопросов ее безопасности для всей среды, является корректная работа с моделью данных ДПОР (14) и модулем сжатия. Рассмотрим основные задачи анализа ОР.

Задача оптимизации параметров (внутренних) состоит в выборе оптимальных в определенном смысле значений их номиналов, при которых система в случае параметрических возмущений остается в работоспособном состоянии в течение заданного времени. Критерии оптимальности могут быть:

стохастическими, основанными на известных законах распределения случайных параметрических отклонений от их расчетных значений;

детерминированными, основанными на анализе конфигурации ОР.

Для случая применения вероятностного критерия оптимальности ДПОР используется с целью проверки принадлежности случайной реализации вектора внутренних параметров ОР, что позволяет избежать трудоемкой процедуры многократного расчета модели (1) и проверки УР с целью получения статистических оценок:

$$\mathbf{x}_{nom} = \arg \max P(\mathbf{X}(\mathbf{x}_{nom}, t) \in D_x, \forall t \in [0, T]),$$

где $\mathbf{X}(\mathbf{x}_0, t)$ – случайный процесс изменения параметров в течение заданного времени эксплуатации T от их начального значения $\mathbf{x}_0$ [1].

Использование детерминированных критериев оптимизации подразумевает основанный на данных о конфигурации ОР выбор начальных значений параметров, обеспечивающих наибольший запас дрейфа с точки зрения выбранной его тенденции. В качестве тенденции могут быть использованы модели дрейфа, специфичные для каждого отдельного параметра, однако такой подход чаще применяется в случае отсутствия знаний о закономерностях дрейфа, когда в качестве наиболее рационального решения принимается критерий максимального удаления от границы ОР – наихудшей тенденции дрейфа к границе ОР по кратчайшему расстоянию [15].

Кратчайшее расстояние от некоторой внутренней точки $\mathbf{x} \in D_x$ до границы области обеспечивает вписанный в ОР шар: среди всех возможных отрезков от точки до точки на границе наименьшим из них будет радиус r вписанного шара с наименьшим объемом $V(\mathbf{x}, r)$:

$$v_r(\mathbf{x}) = \min_r V(\mathbf{x}, r),$$

тогда оптимальным по критерию запаса работоспособности решением задачи

будет поиск точки $\mathbf{x}_0 \in D_x$, являющейся центром максимального вписанного шара (рис. 4а):

$$\mathbf{x}_0 = \arg \max_{\mathbf{x} \in D_x} v_r(\mathbf{x}) = \arg \max_{\mathbf{x} \in D_x} \min_r V(\mathbf{x}, r). \quad (17)$$

С точки зрения модели ДПОР (14), критерий (17) запаса работоспособности состоит в построении симметричной фигуры максимального объема, составленной из ЭГ, входящих в множество E_1 (10), с центральным элементом e_K^c . В качестве таких фигур могут выступать r -окрестность (многомерный аналог окрестности фон Неймана) и r -куб (многомерный аналог окрестности Мура), конфигурация которых описывается соотношениями для индексов ЭГ относительно заданного центрального элемента e_K^c и максимального удаления от него («радиуса») r (рис. 4б) [15, 16].

Рис. 4. Выбор оптимальных параметров по критерию максимального запаса работоспособности: а) построение вписанного в ОР шара максимального объема; б) построение вписанного в ДПОР r -куба максимального объема.

Задача анализа чувствительности системы к параметрическим возмущениям состоит в выполнении геометрического анализа ОР, позволяющего сделать заключения о величине допустимых отклонений параметров для заданных номиналов, выявить «узкие места» ОР, в которых случайные параметрические возмущения могут привести к отказу. В работе предлагается подход к получению такой информации на основе построения вписанных в ОР гиперпараллелепипедов, длины ребер которых по каждому измерению будут показывать относительные запасы для дрейфа соответствующих параметров.

Для оценки запасов по каждому отдельному параметру и получения их соотношений требуется алгоритм построения несимметричных фигур, вписанных в ОР. В качестве таких фигур рассмотрим гиперпараллелепипеды с гранями, параллельными осям координат. Как и симметричные фигуры, эти гиперпараллелепипеды по сути являются множествами ЭГ, принадлежащими

дискретному представлению ОР с индексами, удовлетворяющими определенными требованиям для формирования фигуры [11].

Визуализация сечений ОР используется для анализа конфигурации ОР визуально по отдельным ее сечениям, ручной оценки качества выбора номинальных параметров, их корректировки, что может быть полезно для принятия проектных решений.

Среди средств анализа ОР на основе модели ДПОР в состав ОСПАОР также входят такие модули анализа конфигурации области как проверка ее связности, выпуклости, модуль настройки параметров [3]. Подсистема позволяет отдельным группам и пользователям системы добавлять собственные подпрограммы анализа ДПОР в соответствии с имеющимися полномочиями и принципами безопасности облачной среды [17].

Пользователи и их роли

Учитывая описанную специфику задачи и модули системы, можно выделить следующие группы пользователей и их роли.

Корневой администратор системы. Обладает самыми полными административными полномочиями. Он имеет права на создание, редактирование и удаление рабочей группы и отдельных пользователей, а также назначает права пользователям, в том числе и руководителя группы. Корневой администратор имеет право редактировать фонды данных ОР (удаление давно неиспользуемых или имеющих ошибки, возникшие при построении) и фонды модулей анализа ОР.

Руководитель группы администрирует отдельную группу пользователей, занимающихся определенной задачей, в его компетенции входит редактирование прав пользователей, входящих в его группу, добавление и удаление учетных записей пользователей. Руководитель группы взаимодействует с корневым администратором системы по вопросам безопасности включения необходимых программных модулей в фонды и другим административным вопросам.

Пользователь-инженер может работать как в составе рабочей группы, так и самостоятельно. У него нет прав добавления новых пользователей. Самостоятельный пользователь-инженер может стать руководителем группы только после решения корневого администратора системы. Пользователь-инженер имеет доступ к общим фондам, в рамках данных руководителем группы полномочий имеет право запускать задачи, дополнять групповые фонды данных ОР и модулей для ее анализа.

Пользователь-гость имеет ознакомительный доступ к просмотру ос-

новых функций системы или фондов рабочей группы в случае получения персонального приглашения от руководителя группы.

Заключение

Рассмотрена облачная среда для программного комплекса построения и анализа ОР аналоговых технических систем – задач, решаемых в процессе проектирования сложных технических систем, с учетом параметрических возмущений и требований надежности. Решение этих задач сопряжено с использованием мощных вычислительных ресурсов, систем имитационного моделирования и емких ресурсов хранения данных, допускающих многопользовательскую работу над данными большого объема. Перечисленные факторы обуславливают использование облачных технологий при реализации программного комплекса. Следует также отметить, что описанная программная среда может использовать и другие методы представления ОР, отличные от ДПОР. При этом методы анализа ОР и решение задачи выбора оптимальных параметров должны быть также ориентированы на это представление.

Результаты получены с использованием оборудования ЦКП "Дальневосточный вычислительный ресурс" ИАПУ ДВО РАН (<https://cc.dvo.ru>).

ЛИТЕРАТУРА

1. *Абрамов О.В.* Параметрический синтез стохастических систем с учетом требований надежности. – М.: Наука, 1992.
2. *Абрамов О.В.* Возможности и перспективы функционально-параметрического направления теории надежности // Информатика и системы управления. – 2014. – №4(42). – С. 64-77.
3. *Абрамов О.В., Назаров Д.А.* Программно-алгоритмический комплекс построения, анализа и использования областей работоспособности // Информационные технологии и вычислительные системы. – 2015. – №2. – С. 3-13.
4. *Sun, Le, Dong, H., Hussain, F., Hussain O. and Chang E.* Cloud service selection: State-of-the-art and future research directions // Journal of Network and Computer Applications. – 2014. – Vol. 45. – P. 134-150.
5. *Kundert K.S., Zinke O.* The Designer's Guide to SPICE & SPECTRE. – Springer, 2004.
6. Синтаксический анализатор математических выражений «muParser». URL: <https://beltoforion.de/en/muparser/> (Дата обращения: 18.06.2021)
7. QScriptEngine Class. URL: <https://doc.qt.io/qt-5/qscriptengine.html> (Дата обращения: 18.06.2021)
8. *Назаров Д.А.* Модуль взаимодействия с системой моделирования семейства SPICE для построения областей работоспособности // Инновационные, информационные и коммуникационные технологии: сборник трудов XIV Международной научно-практической конференции. – М.: Ассоциация выпускников и сотрудников ВВИА

имени проф. Жуковского, 2017. – С. 269-272.

9. *Саушев А.В.* Области работоспособности электротехнических систем. – СПб.: Политехника, 2013.
10. *Васильев Б.В., Козлов Б.А., Ткаченко Л.Г.* Надежность и эффективность радиоэлектронных устройств. – М.: Советское радио, 1964.
11. *Назаров Д.А.* Подход к анализу параметрической чувствительности систем на основе дискретного представления областей работоспособности // Информатика и системы управления. – 2017. – № 4(54). – С. 94-104.
12. *Abramov O.V., Nazarov D.A.* Regions of Acceptability in Reliability Design // Reliability: Theory & Applications. – 2012. – Vol.7, No.3(26). – P. 43-49.
13. *Назаров Д.А.* Применение алгоритма кодирования длин серий для уменьшения объема данных дискретного представления областей работоспособности // Информатика и системы управления. – 2020. – №4(66). – С. 85-94. DOI: 10.22250/isu.2020.66.85-94.
14. *Salomon D.* Data Compression: the complete reference. Volume 10. – Springer, 2007.
15. *Назаров Д.А.* Использование областей работоспособности для оптимального выбора номиналов параметров // Информатика и системы управления. – 2011. – №2(28). – С. 59-69.
16. *Катуева Я.В., Назаров Д.А.* Методы параметрического синтеза на основе сеточного представления области работоспособности // Информационные технологии. – 2015. – №9. – С. 651-656.
17. *Ермакова А.Ю., Лось А.Б.* О защите информации в облачных средах // Научный журнал «Апробация». – 2014. – №12(27). – С. 16-20.

E-mail:

Абрамов Олег Васильевич – abramov@iacp.dvo.ru;

Назаров Дмитрий Анатольевич – nazardim@iacp.dvo.ru.