

ФОРМАЛИЗАЦИЯ ПРОЦЕССОВ ИСПЫТАНИЙ БОРТОВОЙ АППАРАТУРЫ КОСМИЧЕСКОГО АППАРАТА

Родион Вячеславович Вогоровский, аспирант

Телефон: 8-913-560-48-43, E-mail: vogorovskiy@icm.krasn.ru

Людмила Федоровна Ноженкова, д.т.н., проф., зав. отделом прикладной информатики

Телефон: 391 2907954 E-mail: expert@icm.krasn.ru

Институт вычислительного моделирования СО РАН,

Сибирский федеральный университет, г. Красноярск

icm.krasn.ru

В работе рассматривается задача формализации процесса испытаний бортовой аппаратуры космического аппарата. Описаны принципы представления процесса проведения испытаний в виде формальных сценариев. Представлены варианты использования сценария на различных этапах испытаний.

Ключевые слова: космический аппарат, командно-измерительная система, поддержка проведения испытаний, сценарий испытаний.

Введение

Бортовая аппаратура космического аппарата – это сложная техническая система с большим количеством характеристик функционирования и высокими требованиями к надежности. Надежность и правильность функционирования таких систем на протяжении всего срока эксплуатации обеспечивается за счет проведения испытаний на всех стадиях производства. Процесс испытаний подразумевает выполнение определенной последовательности этапов, включая подготовку испытаний, выполнение и протоколирование измерений, обработку и оценку результатов. Учитывая сложность бортовой аппаратуры, при проведении испытаний применяются аппаратно-программные системы автоматизации всех этапов испытаний. Такие системы должны обладать возможностями настройки алгоритмов



Л.Ф. Ноженкова



Р.В. Вогоровский

и режимов выполнения испытаний, непрерывного контроля и протоколирования измерений, графической визуализации результатов, построения отчетов. Актуальна задача формализации процесса проведения испытаний с целью описания различных вариантов выполнения измерений и организации управления измерительной аппаратурой в процессе испытаний [1; 2].

В данной работе предлагается формальная модель процесса проведения испытаний в виде сценария, который представляет испытание как совокупность отдельных операций измерения и управления. На основе предложенной формальной модели разработан графический метод построения сценариев испытаний, в котором реализована возможность расширения функций управления и измерения параметров объекта контроля. Реализован модуль исполнения сценария испытаний на контрольно-проверочной аппаратуре, с возможностью контроля параметров в реальном времени и протоколирования результатов. На основе сценариев испытаний выполняется послесеансная обработка и просмотр результатов, реализована функция автоматического построения отчетов в соответствии со структурой сценария.

Задачи организации процесса проведения испытаний

Рассмотрим процесс проведения испытаний бортовой аппаратуры командно-измерительной системы космического аппарата.

Бортовая аппаратура командно-измерительной системы предназначена для осуществления информационного обмена между наземным и бортовым комплексами управления космическим аппаратом. С наземного комплекса управления выполняется прием телекоманд, их первичная обработка и передача для отработки в бортовой комплекс управления. В обратном направлении командно-измерительная система осуществляет передачу пакетов телеметрии с информацией о состоянии бортовых систем космического аппарата и результатами отработки телекоманд. Командно-измерительная система состоит из трех связанных подсистем: передатчика, приемника и интерфейсного модуля командно-измерительной системы. Приемник принимает аналоговый радиосигнал с наземного комплекса управления, преобразует его в цифровой вид и передает для дальнейшей обработки в интерфейсный модуль. Передатчик выполняет обратный порядок действий – принимает пакет данных в цифровом виде, преобразует его в аналоговый сигнал и передает на наземный комплекс управления. В интерфейсном модуле реализована логика обработки пакетов данных и взаимодействия с бортовым и наземным комплексами управления космическим аппаратом, включая выдачу ответов на различные запросы, а также самостоятельную выдачу запросов при необходимости.

Исходя из представленного описания функционала и состава командно-измерительной системы, необходимо проводить испытания как отдельных ее подсистем, так и всей системы в целом. При этом определены два вида проверок, которые должны выполняться совместно и раздельно:

- электрические испытания, направленные на непосредственное измерение параметров передаваемых аналоговых сигналов и различных электрических характеристик подсистем командно-измерительной системы;
- проверки правильности функционирования логики обработки цифровых пакетов данных как в составе борта, то есть с подключением реальных подсистем бортового комплекса управления, так и в автономном режиме с имитацией логики функционирования бортового и наземного комплексов управления.

Проведение таких испытаний и проверок осуществляется средствами контрольно-проверочной аппаратуры, включающей различное измерительное оборудование. Испытания командно-измерительной системы должны охватывать все возможные режимы ее функционирования. Для планирования и выполнения различных испытаний на контрольно-проверочной аппаратуре необходимо разработать формальную модель процесса проведения испытаний. Модель должна описывать различные варианты испытаний с последовательно-параллельным выполнением измерительных операций, а также с использованием циклов и условных переходов. С помощью модели должна решаться задача контроля измеряемых характеристик. Важным требованием является возможность расширения используемых в испытаниях операций с целью проведения испытаний объектов контроля с функционалом и требованиями отличными от тех что предъявляются к командно-измерительной системе.

Представление сценариев испытаний

Процесс проведения испытаний можно представить в виде ряда отдельных операций управления и измерения. Операции управления подразумевают выдачу какого-либо управляющего воздействия или данных на измерительное оборудование или объект контроля с целью изменения его состояния. Измерения направлены на определение и контроль качественных или количественных характеристик объекта контроля. Формальное представление процесса испытаний как ряда операций управления и измерения назовем сценарием испытаний, а формальное представление операций – действиями.

Пусть $A^0 = \{A_1^0, \dots, A_R^0\}$ – исходное множество действий, элементы которого используются для построения сценария. Каждое действие характеризуется множеством параметров настроек и мониторинга. Настройки предназначены для конфигурирования

измерительного оборудования или управляющего воздействия на объект контроля. Параметры мониторинга задают измеряемые характеристики и устанавливают границы допустимых значений для решения задачи контроля. Таким образом, действие из множества A^0 можно представить как:

$$A_r^0 = \langle I_r, M_r \rangle, r \in \overline{1 \dots R}, \text{ где}$$

I_r – множество настроек действия,

M_r – множество параметров мониторинга.

Множество действий A^0 может изменяться в зависимости от решаемых задач и объекта испытаний. Дополнение множества новыми действиями осуществляется при помощи процедуры регистрации, при которой для каждого нового действия указывается наименование, список параметров настроек и мониторинга [3].

Множество действий для проведения испытаний бортовой аппаратуры командно-измерительной системы космического включает: генерацию импульсов, выдачу директивы, измерение номинала частоты, управление мультиметром, измерение мощности СВЧ, измерение параметров сигналов релейных команд, измерение спектра без маски, измерение спектра с использованием маски, измерение мощности спектра, управление тест-транслятором, измерение коэффициента стоячей волны, передачу телекоманды, приём телеметрии, измерение мощности и частоты сигнала, управление конвертером, измерение параметров частотной модуляции, управление коммутацией, управление аттенюатором, измерение параметров фазовой модуляции, измерение амплитудно-частотной характеристики и др.

Совокупность действий, направленная на выполнение определенной задачи в сценарии испытаний, называется заданием. Весь процесс проведения испытаний представляет собой совокупность заданий, состоящих из действий. Для определения порядка выполнения заданий в сценарии, рассмотрим задание как интервал, во время которого оно активно. Временная продолжительность выполнения заданий заранее не известна, так как она напрямую зависит от того, как долго будут выполняться действия в рамках задания. Поэтому введем понятие этапов – относительной меры продолжительности выполнения задания. При этом сценарий разбивается на этапы выполнения различных заданий, и его можно представить как:

$$S = \langle T, K \rangle, \text{ где}$$

T – упорядоченная совокупность заданий в рамках сценария,

K – количество этапов в сценарии.

Интервалы выполнения заданий задаются номерами начального и конечного этапов. Номера начальных этапов заданий определяют очередность их запуска в сценарии. Задания, этапы выполнения которых пересекаются, выполняются параллельно на участках пересечения. На рис. 1 представлен пример сценария, состоящего из трех заданий: задания 1 и 2 выполняются последовательно, и параллельно им обоим выполняется задание 3.



Рис. 1 – Пример сценария испытаний

Каждое задание в сценарии выполняется в цикле, где каждая итерация цикла — это выполнение совокупности действий, определенной внутри задания. Предусмотрено два вида циклов: цикл со счетчиком и цикл по условию, пока значение одного или нескольких параметров действий задания изменяется в определенном интервале с заданным шагом

(например, цикл по устанавливаемым значениям напряжения питания от 1 В до 10 В с шагом 0,5 В). Задание в сценарии можно представить в следующем виде:

$$T_i = \langle A_i, C_i, V_i, k_i, l_i \rangle, T_i \in T, i \in \overline{1 \dots |T|},$$

где

A_i – упорядоченная совокупность действий в рамках задания,

C_i – описание цикла выполнения задания,

V_i – количество этапов в задании.

$k_i \in \overline{1 \dots K}$ – начальный этап выполнения задания,

$l_i \in \overline{1 \dots K}, l_i \geq k_i$ – конечный этап выполнения задания.

Задания строятся из действий исходного множества A^0 , при этом к таким действиям добавляются дополнительные характеристики: условия запуска и остановки выполнения действия, номера начального и конечного этапов выполнения. Эти характеристики задают порядок и правила выполнения действий в задании. Условия запуска выполнения действия позволяют задавать условные переходы между действиями во время выполнения сценария. Предусматривается безусловный запуск действий. Условия остановки определяют продолжительность выполнения действия либо явно – в виде задания времени выполнения, либо в виде условий, накладываемых на параметры мониторинга действия. Номера начального и конечного этапов определяют порядок выполнения действий аналогично этапам для заданий. Таким образом, действие A_{ij} в рамках задания T_i можно представить как:

$$A_{ij} = \langle I_{ij}, M_{ij}, L_{ij}, F_{ij}, q_{ij}, p_{ij} \rangle, A_{ij} \in A_i, j \in \overline{1 \dots |A_i|},$$

где

I_{ij} – множество настроек действия,

M_{ij} – множество параметров мониторинга,

L_{ij} – условия запуска выполнения действия,

F_{ij} – условия остановки выполнения действия,

$q_{ij} \in \overline{1 \dots V_i}$ – начальный этап выполнения действия,

$p_{ij} \in \overline{1 \dots V_i}, p_{ij} \geq q_{ij}$ – конечный этап выполнения действия.

Сценарий испытаний и правила его построения формально можно представить в виде контекстно-свободной грамматики G , для описания будем использовать расширенную форму Бэкуса-Наура [4]:

$$G = (N, T, P, St),$$

где

N – множество нетерминальных символов,

T – множество терминальных символов,

P – конечное множество продукций,

St – начальный символ.

Множество нетерминалов содержит символы, обозначающие основные понятия и конструкции сценария испытаний: $N = \{ \langle \text{сценарий} \rangle, \langle \text{этап_сценария} \rangle, \langle \text{задание} \rangle, \langle \text{этап_задания} \rangle, \langle \text{продолжительность_этапов_сценария} \rangle, \langle \text{действие_в_задании} \rangle, \langle \text{действие} \rangle, \langle \text{продолжительность_этапов_задания} \rangle, \langle \text{условия_запуска} \rangle, \langle \text{условия_остановки} \rangle, \langle \text{цикл} \rangle, \langle \text{наименование} \rangle, \langle \text{идентификатор_действия} \rangle, \langle \text{идентификатор_настройки} \rangle, \langle \text{идентификатор_параметра_мониторинга} \rangle, \langle \text{интервал_с_шагом} \rangle, \langle \text{числовое_значение} \rangle, \langle \text{мин_допуск} \rangle, \langle \text{макс_допуск} \rangle, \langle \text{мин_предупреждение} \rangle, \langle \text{макс_предупреждение} \rangle, \langle \text{операция_сравнения} \rangle, \langle \text{единица_измерения} \rangle, \langle \text{пакет_данных} \rangle, \langle \text{время} \rangle, \langle \text{натуральное_число} \rangle, \langle \text{целое_число} \rangle, \langle \text{рациональное_число} \rangle, \langle \text{цифра} \rangle, \langle \text{буква} \rangle \}$.

Множество терминалов включает элементы исходного множества действий A^0 , цифры, буквы русского и английского алфавитов, знаки операций сравнения и другие символы: $T = \{ A^0, 0, 1, \dots, 9, a, \dots, я, A, \dots, Я, a, \dots, z, A, \dots, Z, <, >, \leq, \geq, =, \neq, (,), ., - \}$.

Начальным символом грамматики является $\langle \text{сценарий} \rangle$. Определим продукции, которые порождают сценарий как последовательность этапов выполнения заданий. Каждый этап представляется множеством заданий, которые должны начинать свое выполнение на этом этапе. Этап не может быть пустым. Порядок следования этапов в сценарии определяет порядок выполнения заданий.

$$\langle \text{сценарий} \rangle = \langle \text{наименование} \rangle, (\langle \text{этап_сценария} \rangle, \{ \langle \text{этап_сценария} \rangle \});$$

$$\langle \text{этап_сценария} \rangle = \text{“Этап сценария:”}, (\langle \text{задание} \rangle, \{ \langle \text{задание} \rangle \});$$

Аналогично задание представляется как последовательность этапов выполнения действий.

$$\langle \text{задание} \rangle = \langle \text{наименование} \rangle, \langle \text{продолжительность_этапов_сценария} \rangle, \langle \text{цикл} \rangle,$$

$$(\langle \text{этап_задания} \rangle, \{ \langle \text{этап_задания} \rangle \});$$

$$\langle \text{этап_задания} \rangle = \text{“Этап задания:”}, (\langle \text{действие_в_задании} \rangle, \{ \langle \text{действие_в_задании} \rangle \});$$

Нетерминальный символ $\langle \text{продолжительность_этапов_сценария} \rangle$ позволяет установить количество этапов выполнения задания. Цикл выполнения задания может быть задан в виде натурального числа, определяющего количество повторений, или в виде ряда изменяемых в цикле параметров настроек действий:

$$\langle \text{цикл} \rangle = \langle \text{натуральное_число} \rangle \mid \{ \langle \text{идентификатор_действия} \rangle, \langle \text{идентификатор_настройки} \rangle, \langle \text{интервал_с_шагом} \rangle \};$$

$$\langle \text{интервал_с_шагом} \rangle = \text{“от”} \langle \text{числовое_значение} \rangle, \text{“до”}, \langle \text{числовое_значение} \rangle,$$

$$\text{“шаг”} \langle \text{числовое_значение} \rangle.$$

Действие в задании представляется как элемент из исходного множества действий ($\langle \text{действие} \rangle$) и характеристик, определяющих правила его выполнения в задании:

$$\langle \text{действие_в_задании} \rangle = \langle \text{действие} \rangle, \langle \text{продолжительность_этапов_задания} \rangle,$$

$$\langle \text{условия_запуска} \rangle, \langle \text{условия_остановки} \rangle.$$

Нетерминал $\langle \text{действие} \rangle$ раскрывается в грамматике с помощью ряда продукций, соответствующих элементам исходного множества действий A^0 . Одна продукция соответствует одному действию из A^0 . Действие представляется в правой части продукции как совокупность идентификатора действия, наименования, параметров настроек и мониторинга. Каждая настройка имеет свой идентификатор, наименование и значение. Предполагается использование двух типов значения: числового и пакета данных. Пакеты данных – это массивы данных, которые задаются в виде последовательности нулей и единиц. Для параметров мониторинга задаются идентификатор, наименование и границы интервалов допустимых значений для числовых параметров. В таблице представлены примеры правых частей продукций вида:

$$\langle \text{действие} \rangle = \dots,$$

соответствующие некоторым элементам множества действий по испытанию бортовой аппаратуры командно-измерительной системы космического аппарата.

Таблица

Действия по испытанию бортовой аппаратуры командно-измерительной системы

ИД	Наименование действия	Настройки			Мониторинг		
		ИД	Наименование	Значение	ИД	Наименование	Допуск
1	Измерение мощности и частоты сигнала	1	Полоса поиска пика	$\langle \text{числовое_значение} \rangle$	1	Частота пика	$\langle \text{допуск} \rangle$
		2	Ожидаемая частота	$\langle \text{числовое_значение} \rangle$			
		3	Ожидаемый уровень сигнала	$\langle \text{числовое_значение} \rangle$	2	Уровень пика	$\langle \text{допуск} \rangle$
		4	Разрешение по частоте	$\langle \text{числовое_}$			

				значение>			
2	Измерение мощности СВЧ	5	Частота	<числовое_значение>	3	Измеренное значение мощности	<допуск>
3	Передача телекоманды	6	Команда	<пакет_данных>	4	Флаг передачи	
			Включение кодера	<числовое_значение>			
4	Прием телеметрии				5	Данные телеметрии	<пакет_данных>

Для представления значений параметров настроек и интервалов допустимых значений параметров мониторинга действий используются следующие продукции:

<числовое_значение> = <целое_число> | <рациональное_число>;
 <пакет_данных> = {"0" | "1"};
 <допуск> = <мин_допуск>, <макс_допуск>, <мин_предупреждение>, <макс_предупреждение>;
 <мин_допуск> = [<числовое_значение>];
 <макс_допуск> = [<числовое_значение>];
 <мин_предупреждение> = [<числовое_значение>];
 <макс_предупреждение> = [<числовое_значение>];

Условия запуска действия задаются как левая часть правила вида «Если ... то запуск». В левой части правил сравниваются значения параметров мониторинга предыдущих действий в задании с определенными значениями.

<условия_запуска> = {"Если ", <идентификатор_действия>, <идентификатор_параметра_мониторинга>, <операция_сравнения>, <числовое_значение>, {" и ", <идентификатор_действия>, <идентификатор_параметра_мониторинга>, <операция_сравнения>, <числовое_значение>}};

Условия остановки задаются либо в левой части правила вида «Если ... то остановка», либо в виде явного указания времени выполнения действия.

<условия_остановки> = ({ "Если ", <идентификатор_параметра_мониторинга>, <операция_сравнения>, <числовое_значение>, {" и ", <идентификатор_параметра_мониторинга>, <операция_сравнения>, <числовое_значение>}} | <время>);

Представленная формальная модель процесса испытаний позволяет описать в виде различных сценариев все возможные варианты испытаний бортовой аппаратуры командно-измерительной системы космического аппарата.

Применение модели сценария для подготовки и проведения испытаний

Сценарий испытаний является основой технологии комплексной поддержки проведения испытаний бортовой аппаратуры командно-измерительной системы космического аппарата. Технология заключается в применении алгоритмического и программного обеспечения для автоматизации испытаний на этапах подготовки, проведения испытаний, обработки и анализа результатов. Сценарий испытаний, как формальное представление процесса проведения испытаний объекта контроля, используется на каждом из этапов технологического процесса испытаний (рис. 2).

На этапе подготовки испытаний выполняется построение сценариев испытаний с помощью графического редактора. Редактор позволяет в графическом виде создавать задания и сценарии испытаний, оперируя элементами исходного множества действий [5]. На этапе проведения испытаний осуществляется исполнение подготовленного сценария. Программный модуль проведения испытаний управляет работой контрольно-проверочной аппаратуры, поэтапно выполняя каждое задание и действие в соответствии со сценарием испытаний. Параллельно осуществляется контроль параметров мо-

ниторинга и архивирование результатов испытаний. Результаты записываются в хранилище данных вместе со структурой сценария и временем получения значений параметров мониторинга. Во время просмотра результатов сценариев в совокупности с результатами испытаний используется для воспроизведения процесса испытаний с возможностью навигации по времени. На этапе построения отчетов сценарий определяет структуру отчета, где задания являются разделами отчета.

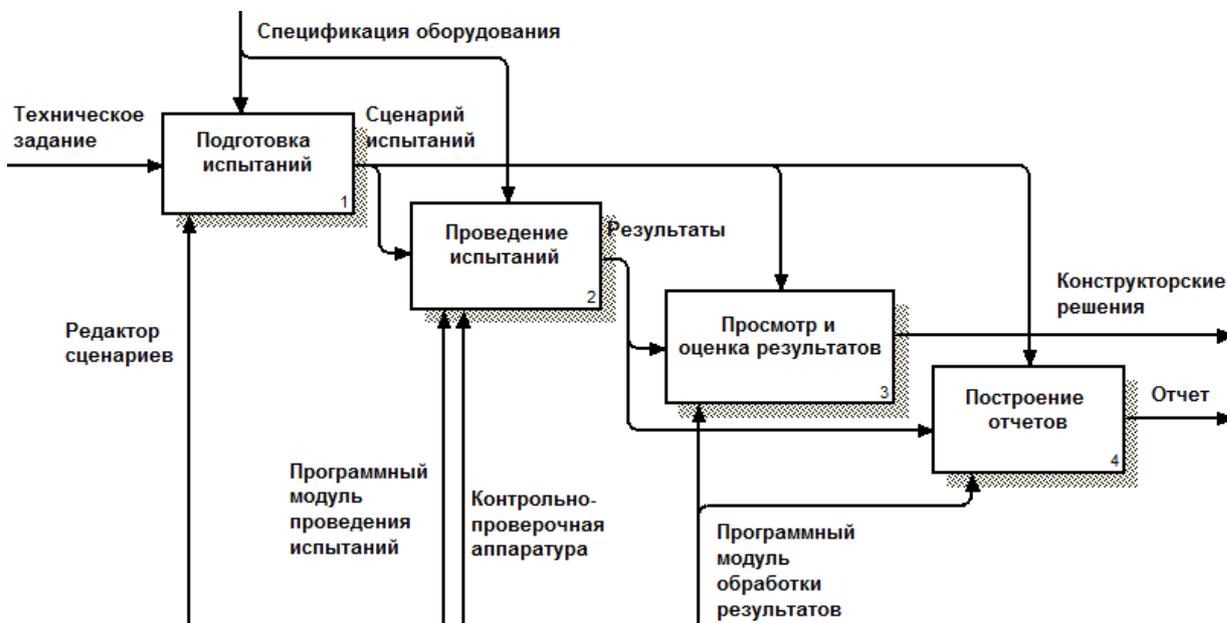


Рис. 2 – Процесс испытаний

Заключение

Разработана формальная модель процесса проведения испытаний бортовой аппаратуры космического аппарата в виде сценария испытаний. Сценарий представляет собой упорядоченную совокупность заданий и действий. Путем составления различных сценариев конструктор может представить все необходимые варианты испытаний логики функционирования и проверки электрических характеристик бортовой аппаратуры командно-измерительной системы космического аппарата.

Формальная модель сценария испытаний является основой технологии комплексной поддержки испытаний, выполняющей автоматизацию задач на этапах подготовки и проведения испытаний, просмотра и оценки результатов, построения отчетов.

Литература:

1. Цапенко М.П. Измерительные информационные системы: структура и алгоритмы, схемотехническое проектирование: Учеб. Пособие для вузов. – М.: Энергоатомиздат, 1985. 438 с.
2. Технология сборки и испытаний космических аппаратов: Учебник для высших технических учебных заведений / И.Т. Беляков, И.А. Зернов, Е.Г. Антонов и др. – М.: Машиностроение, 1990. 352 с.
3. Вогоровский Р.В. Организация взаимодействия с измерительным оборудованием при проведении испытаний бортовой аппаратуры КА // Молодой ученый. 2015. № 11. С. 22-27.
4. ISO/IEC 14977:1996. Information technology – Syntactic metalanguage – Extended BNF. – ISO/IEC, 1996. 12 p.
5. Вогоровский Р.В. Графическая среда построения сценариев испытаний бортовой аппаратуры космического аппарата // Решетневские чтения: материалы XIX Междунар. науч.-практ. конф., посвящ. 55-летию Сиб. гос. аэрокосмич. ун-та им. акад. М.Ф. Решетнева (10-14 ноября 2015, г. Красноярск): в 2 ч. / под общ. ред. Ю.Ю. Логинова; Сиб. гос. аэрокосмич. ун-т. – Красноярск, 2015. Ч. 2. С. 211-213.

Formalization of test processes of the spacecraft onboard equipment

Rodion Vyacheslavovich Vogorovskiy, engineer

Lyudmila Fedorovna Nozhenkova, Ph. D., Professor, the head. Department of applied Informatics

Institute of computational modeling SB RAS, Siberian Federal University, Krasnoyarsk

In this paper, the problem of formalization of test process of the spacecraft onboard equipment is observed. The representation principles of test processes as formal scripts are described. The cases of using scripts at all stages of test process are presented.

Keywords: spacecraft, command and measuring system, test execution support, scripts of testing.

УДК 004.43 (042.4)

**О КЛАССИФИКАЦИЯХ ПАРАДИГМ ПРОГРАММИРОВАНИЯ
И ПАРАЛЛЕЛЬНОМ ПРОГРАММИРОВАНИИ**

*Лидия Васильевна Городняя, к.ф.-м.н., доцент, старший научный сотрудник
Тел.: 8 383 330 6470, e-mail: lidvas@gmail.com*

*Новосибирский национальный исследовательский государственный университет
www.nsu.ru*

*Институт систем информатики им. А.П. Ершова СО РАН
www.iis.nsk.su*

Работа нацелена на привлечение внимания к проблеме изучения парадигм программирования, актуальность которой связана с расширением множества языков программирования и массовым распространением распределённых информационных систем, использующих взаимодействующие параллельные процессы.

Ключевые слова: языки и системы программирования, парадигмы программирования, методы определения компьютерных языков, параллельные алгоритмы, учебное программирование.

Актуальность изучения и систематизации парадигм программирования связана с практическими задачами обоснования при выборе подходов к обеспечению производи-



Л.В. Городняя

тельности, надежности и эффективности сложных информационных систем, конструируемых с использованием разносортных информационных технологий и сервисов, применяемых в разных условиях и требующих учёта многообразных явлений, связанных с поддержкой параллельных процессов на базе многопроцессорных конфигураций [1].

Парадигмами программирования в форме языков и систем программирования представлено знание о потенциале надёжных и безопасных средств и методов создания информационных систем. Теоретически различие парадигм программирования может быть выражено на уровне операционной семантики, представляющей детали структур памяти, механизмы выполнения укрупнённых действий, принципы компиляции и верификации программ [2; 3; 4].

Суть авторской работы заключается в анализе парадигматической классификации систем программирования, допускающей лаконичную характеристику их практически значимых особенностей, определяющих трудоёмкость программирования и живучесть разрабатываемых программ с акцентом на проблемы параллельного программирования [5; 6; 7]. Для выработки практических рекомендаций такого рода требуется чёткая