СИТУАЦИОННЫЕ СПЕЦИФИКАЦИИ ИМИТАЦИОННЫХ МОДЕЛЕЙ ГИБРИДНЫХ РЕАКТИВНЫХ СИСТЕМ

В. М. Шпаков

Санкт-Петербургский институт информатики и автоматизации РАН
199178, Санкт-Петербург, 14-я линия В.О., д.39
vlad@iias.spb.su

УДК 681.3.06
В. М. Шпаков. **Ситуационные спецификации имитационных моделей гибридных реактивных систем** // Труды СПИИРАН. Вып. 1, т. 2 — СПб: СПИИРАН, 2002.
Аннотация. *Приводится описание формализма спецификации имитационных моделей гибридных реактивных систем, основанного на модификации и объединении известных формализмов машины конечных состояний и гибридного автомата. Формализм ориентирован на создание исполняемых спецификаций систем, пригодных для итерационного процесса их разработки и отладки. Обсуждаются выразительность и эффективность формализма и приводятся результаты его экспериментального исследования.* — Библ. 6 назв.

UDC 681.3.06
V. M. Shpakov. **Situation specifications of hybrid reactive system simulation models** // SPIIRAS Proceedings. Issue 1, v. 2. — SPb: SPIIRAS, 2002.
Abstract. *Description of hybrid reactive system simulation model specification formalism is presented. The formalism is based on modifying and uniting of well known FSM and Hybrid Automata formalisms and oriented on creation of executable system specifications which are suitable for stepwise working out. The formalism expressiveness and effectiveness are discussed and results of its experimental research are presented.* — Bibl. 6 items.

Термин “реактивная система” был введен Д. Харелом для обозначения систем, которые находятся в постоянном обмене информацией со средой и должны реагировать на ее изменения (события) [1]. Среда не может ждать, поэтому реакция должна быть не только логически, но и динамически адекватной изменениям среды. Этим реактивные системы отличаются от интерактивных систем (операционные системы). К реактивным системам относится большинство систем реального времени: от небольших встраиваемых управляющих устройств до больших многоуровневых производственных систем управления. Введение этого термина преследовало цель обратить внимание на отличие реактивных систем от систем алгоритмического типа, которые получают исходную информацию, обрабатывают ее, выдают результат и заканчивают свою работу. Работа же реактивной системы может продолжаться неопределенно долго, причем длительность ее работы зависит не от результата обработки информации, а чаще всего от внешней команды. При имитационном моделировании систем управления модель среды должна быть также представлена в виде реактивной системы, так как ее реакция на управляющие воздействия должна соответствовать реальной среде. В данной работе рассматриваются формальные описания (спецификации) систем из пересечения множества реактивных систем с множеством гибридных систем.

Динамическая система называется гибридной, если ее поведение определяется изменениями, как непрерывных переменных состояния, так и дискретно-событийных. Дискретно-событийные переменные состояния обычно представ-

Труды СПИИРАН. 2002. Вып. 1. Т. 2. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2002. Issue 1. V. 2. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

ляются символьными или лингвистическими переменными, а непрерывные состояния — вещественными переменными. В обширной литературе по гибридным динамическим системам (ГДС) можно встретить следующее образное ее определение:

“Гибридная система = Конечный автомат + Аналоговый компьютер”.

Изучение гибридных динамических систем (ГДС) активно началось в последнем десятилетии и связано с увеличивающимся использованием автоматизированного оборудования и широким внедрением компьютеров на производстве и транспорте. Таким образом, ГДС характеризуются наличием подсистем с непрерывными переменными состояния, динамика которых может быть описана дифференциальными или разностными уравнениями, и дискретно-событийных подсистем, изменения состояния которых происходят асинхронно и определяются наступлением определенных событий. Наиболее популярным формализмом для представления ДСС является формализм машины конечных состояний (Finite State Machine (FSM)) [2].

Гибридные системы образуют обширный класс, отличающийся большим разнообразием, определяемым как количественными соотношениями между дискретной и непрерывной частями, так и различным характером и степенью их взаимодействия друг с другом. На наш взгляд, имеет смысл различать следующие два типа взаимодействий между дискретной и непрерывной частями:

1. Значения символьных переменных определяют режим работы, т.е. параметры непрерывной части (включение контуров, изменение установок); выходы вещественных непрерывных переменных за определенные значения представляют собой события, вызывающие изменения значений символьных переменных.
2. Наряду с непрерывными изменениями вещественных переменных состояния происходят их скачкообразные изменения, после которых продолжаются непрерывные изменения. Эти скачкообразные изменения связаны с наступлением некоторых дискретных событий, вызванных внешними или внутренними управляющими или возмущающими воздействиями, или выходом непрерывного состояния за определенные границы в пространстве состояний.

Для анализа и представления ГДС в последнее время получил широкое распространение формализм гибридного автомата [3,4], который, однако, в первую очередь ориентирован на ГДС с взаимодействием 1-го типа из приведенных выше.

Что касается реактивных гибридных систем, то к ним в полной мере относятся отмеченные в [5] особенности реальных систем, существенные для реализации их моделей. В первую очередь — это наличие и взаимодействие параллельных процессов различной динамики. Далее приводятся описание и результаты экспериментального исследования формализма спецификации ГДС, ориентированного на имитационное моделирование реактивных систем и основанного на совместном применении модифицированного формализма машины конечных состояний и гибридного автомата. Модификация формализма была направлена на обеспечение ему свойств, отвечающих требованиям моделирования реактивных систем. Среди этих требований следует отметить необходимость представления значительного числа параллельных непрерывных и дискретно-событийных процессов, произвольным образом взаимодействующих друг с другом. Далее идут выразительность спецификаций моделей и пригодность их для эффективной реализации. Дело в том, что имитационное моделирование остается пока основным средством разработки, верификации и обес-

Труды СПИИРАН. 2002. Вып. 1. Т. 2. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2002. Issue 1. V. 2. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

печения работоспособности ГДС, несмотря на развитие ряда аналитических и алгоритмических методов. А для разработки наиболее удобны исполняемые спецификации моделей, допускающие итерационный характер разработки (внесение дополнений/изменений — запуск).

Рассмотрение начнем с формализма машины конечных состояний (FSM). Базовый формализм представляет собой следующий кортеж [2]:

$FSM = (Q, \Sigma, \Delta, \sigma, q_0)$,

Q — множество символов, представляющих состояние;

Σ — множество входных символов;

Δ — множество выходных символов;

σ — функция перехода (трансформации состояния (transition)): $Q \times \Sigma \to Q \times \Delta$;

$q_0 \in Q$ — начальное состояние.

FSM может быть представлена в виде ориентированного графа, называемого графом переходов состояния. Вершины графа представляют состояния модели, а дуги — переходы. Каждая дуга маркирована парой "условие/действие", где первое представляет условие перехода, а второе — результат. На Рис. 1 приведен в качестве примера граф модели, у которой $Q = \{\alpha, \beta\}$, $\Sigma = \{a, b\}$, $\Delta = \{\varepsilon, u, v\}$, $q_0 = \alpha$ и $\sigma(\alpha, b) = (\beta, v)$, $\sigma(\beta, a) = (\alpha, u)$. Выходной символ ε обозначает отсутствие действия , он обычно вводится для придания общности и используется для представления автопереходов: $\sigma(\alpha, a) = (\alpha, \varepsilon)$, $\sigma(\beta, b) = (\beta, \varepsilon)$.

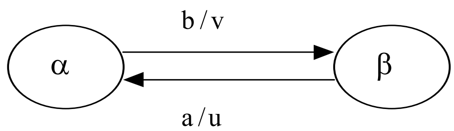

Рис. 1. Пример базовой машины конечных состояний

Входной символ $b \in \Sigma$ переводит модель в состояние $\beta \in Q$ и на выходе появляется символ $v \in \Delta$. Последовательность символов входного алфавита вызывает последовательную смену состояний и производит последовательность символов выходного алфавита.

В [2] показано, что более естественно рассматривать входные и выходные сигналы FSM состоящими из нескольких переменных, каждая из которых может принимать значения из своего множества определения. При этом входной (выходной) алфавит будет представлять собой прямое произведение множеств задания (определения) отдельных переменных. Так для m входных переменных будем иметь $\Sigma = \Sigma_1 \times \cdots \times \Sigma_i \times \cdots \times \Sigma_m$, где Σ_i — множество задания i-ой переменной. В исследуемом нами формализме мы представляем и состояние системы в виде совокупности переменных состояния, принимающих значения из своих множеств определения. Такое составное состояние иногда называют "& состояние" ("and state"). Использование составных состояний обеспечивает более компактное их представление, допускающее, во первых, более эффективную реализацию переходов и, во вторых, более естественную ситуационную

Труды СПИИРАН. 2002. Вып. 1. Т. 2. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2002. Issue 1. V. 2. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

интерпретацию состояний. Действительно, мы получаем более интуитивно понятное описание состояния объекта, если используем совокупность дискретных (символьных, лингвистических) переменных, принимающих значения из различных предметных подмножеств. Например, состояние станка может быть представлено следующей совокупностью значений лингвистических (дискретных) переменных: "исправен", "включен", "операция назначена", "материала нет". Подобные совокупности переменных состояния мы называем описанием ситуации.

Использование ситуаций или слов состояния в алфавите символов состояния приводит к многократному увеличению пространства состояния, а это, в свою очередь, позволяет сократить мощность множества значений каждой переменной до 2 и представлять состояния с помощью логических переменных. Такие состояния называют состояниями типа "исключающее или" ("XOR state"). Они занимают нижний уровень в диаграммах Харела, представляющих иерархию состояний. Дискретную переменную с более, чем двумя значениями всегда можно представить с помощью совокупности логических переменных. Аппаратным средством такого представления являются концевые выключатели. Так, применяя два концевых выключателя положений клапана, мы получаем информацию о 3-х значениях его состояния: (закрыт, не открыт), (не закрыт, открыт) и (не закрыт, не открыт). Избыточное противоречивое значение ситуации (закрыт, открыт) может быть использовано для диагностики ошибок. При этом любое составное состояние может быть представлено элементарной конъюнкцией логических переменных. Применение логических переменных для описания состояний и входных сигналов позволяет упростить вычисление функции переходов σ. В нашем случае она является логической функцией двух аргументов и должна принимать значение "Истина", только когда значение "Истина" принимают конъюнкции состояний и входов, а это значит, что эта функция может быть представлена конъюнкцией соответствующих конъюнкций. Таким образом, область значений функции перехода может быть представлена истинностными значениями локальных ситуаций, каждая из которых является элементарной конъюнкцией переменных состояния и входных переменных:

$S_j = s_{j_1},...s_{j_i},...,s_{j_k}$, где $s_{j_i} = b$ или $s_{j_i} = \neg b$, $b \in Q \bigcup \Sigma$. (1)

Общее количество локальных ситуаций равно мощности множества всех подмножеств объединения множеств состояний и входов, т.е. $2^{|Q \cup \Sigma|}$, однако количество значимых для описания конкретной модели ситуаций определяется содержанием задачи моделирования. Мы посчитали удобным механизм формирования выходных переменных вынести за рамки формализма, то есть рассматривать все переменные состояния как выходные. С учетом этого функцию трансформации состояния σ можно представить с помощью совокупности правил типа "if...then", левая часть (условие) которых является ситуацией, а правая (действие) — значениями совокупности переменных состояния, которые они должны принимать при возникновении данной ситуации (значение соответствующей ей конъюнкции равно "Истина"):

$S_j \to q_{j_1},...q_{j_i},...,q_{j_m}$, $q_j \in Q$. (2)

Для того чтобы приведенные спецификации были исполняемыми, необходимо реализовать соответствующую им исполняющую процедуру. В данном случае она представляет собой бесконечный цикл (останов по внешней команде), в теле которого производится сканирование правил (2), в левые части которых подставляются текущие значения входных переменных и переменных

Труды СПИИРАН. 2002. Вып. 1. Т. 2. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2002. Issue 1. V. 2. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

состояния и, если при этом значение ситуации равно “Истина”, то переменным состояния в правой части присваиваются специфицированные значения (“Истина” или “Ложь”). При этом в результате ошибок спецификации возможно возникновение конфликтов между правилами, когда одной и той же переменной разные правила присваивают разные значения. Чтобы исключить возникновение конфликтов мы пошли на некоторое ограничение выразительных возможностей формализма, допустив присваивание состояниям в правой части правил только значений “Истина”. При этом значение “Ложь” присваивается по умолчанию всем переменным состояния, которым в результате сканирования правил не присвоено значение “Истина”. Экспериментальные исследования формализма показали, что введение этого ограничения повышает надежность спецификаций и несущественно уменьшает их выразительность. Надежности приходится отдавать приоритет перед выразительностью до тех пор, пока не созданы эффективные средства верификации формализма. Очевидно, что быстрота реакции системы в данном случае определяется длительностью цикла сканирования исполняющей процедуры.

Формализм машины трансформации ситуаций может рассматриваться как черный ящик, осуществляющий преобразование последовательности входных ситуаций в последовательность выходных, и в таком качестве встраиваться в другие формализмы. При разработке описываемого формализма спецификации мы прежде всего ориентировались на реальные реактивные системы типа автоматизированных промышленных установок, которые обычно включают в свой состав параллельно функционирующие, взаимодействующие друг с другом дискретно-событийные, непрерывные и гибридные подсистемы. Та часть описываемого формализма, которая ориентирована на представление гибридных подсистем, имеет много общего с наиболее развитым в настоящее время описанием гибридных систем с помощью формализма гибридного автомата (ГА), поэтому описание этой части удобно представить в виде формулировок ее отличий от ГА. Гибридный автомат представляется в виде следующего кортежа:

$H = (V, X, Init, f, Dom, Jump)$,

где V — множество дискретных переменных со значениями в Q, представляющих режимы H.

$X = \{x_1,...,x_n\}$ — множество непрерывных переменных со значениями в R^n.

$Init \subseteq Q \times R^n$ — множество начальных состояний.

$f : Q \times R^n \to R^n$ — векторное поле, определяет функциональные зависимости между непрерывными переменными для каждого режима из Q.

$Dom \subseteq Q \times R^n$ — множество, определяющее область значений (domain) H, т.е. допустимую область изменения X для каждого режима.

$Jump : Q \times R^n \to P(Q \times R^n)$ — функция, определяющая возможность перехода из одного режима в другой и новые значения, которые некоторые непрерывные переменные принимают в случае данного перехода.

Под состоянием H понимают пару (q, z), которая состоит из дискретной переменной режима $q \in Q$ и точки $z \in R^n$, являющейся значением x. Таким образом, ГА отличается от непрерывной системы тем, что в нем происходят мгновенные смены режимов. Функция Jump определяет для каждого режима, при каких значениях непрерывного вектора и в какой режим происходит переход, а также значение непрерывного вектора после перехода. Кроме того, каждый ре-

Труды СПИИРАН. 2002. Вып. 1. Т. 2. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2002. Issue 1. V. 2. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

жим обладает своей динамикой изменения непрерывных переменных, определяемой зависящей от режима векторной функцией f.

Гибридные автоматы удобно представлять в виде ориентированного графа (V,E) с вершинами V, представляющими дискретные режимы, и дугами E, называемыми управляемыми переключениями (control switches), $E=\{(q,q^{'})\in Q\times Q:\exists z,z^{'}\in R^{n},(q^{'},z^{'})\in Jump(q,z)\}$, где $q^{'},z^{'}$, соответственно, дискретное и непрерывное состояния автомата после перехода. С каждой дугой $e(q,q^{'})\in E$ связано $G(e)$ (Guard) — условие перехода в новое дискретное состояние, соответствующее значению непрерывного состояния, определяемому функцией $Jump$: $G(e)=\{x\in R^{n}:\exists x^{'}\in R^{n},(q^{'},x^{'})\in Jump(q,x)\}$, и новое значение непрерывного состояния $J(e)=\{x^{'}\in R^{n}:(q^{'},x^{'})\in Jump(q,x)\}$.

Таким образом, условие перехода G для каждого дискретного режима определяет область значений непрерывного состояния, для которых возможен переход в новое дискретное состояние, а J задает новое значение непрерывного состояния. На Рис. 2 в качестве примера представлена графовая нотация модели термостата. В данном случае гибридный автомат представлен одной дискретной переменной $V=\{v\}$, принимающей два значения (режима) $v\in Q=\{Off,On\}$, соответствующих выключенному и включенному состояниям нагревателя, непрерывной переменной $X=\{x\}$, $x=(x_1,x_2)$, и функцией f, описывающей непрерывные изменения для каждого режима:

$\dot{x}_1 = x_2$; $x_2 = -0{,}1\cdot x_1$, если $v=Off$ и

$\dot{x}_1 = x_2$; $x_2 = -0{,}1\cdot x_1 + 5$, если $v=On$.

Здесь x_1 представляет температуру, а x_2 — скорость ее изменения. При спецификации ГА, естественно, указываются только те координаты вектора непрерывных состояний, которые определяют условие перехода и изменение динамики. В данном случае условия и значения переходов:

$G(Off,On)=x_1<19$, $G(On,Off)=x_1>21$;

$Jump(Off,x_2)=(On,x_2+5)$ $Jump(On,x_2)=(Off,x_2-5)$;

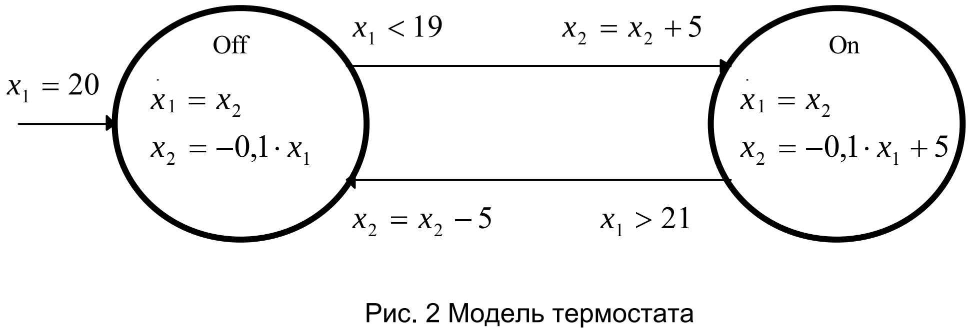

Рис. 2 Модель термостата

начальное состояние $Init=(Off,(20,-2))$ и область значений $Dom=(Off,(x_1>18)),(On,(x_1<22))$.

Таким образом, характерной особенностью поведения ГА являются переходы из одного режима в другой, вызываемые выходом непрерывных состояний за пределы заданных областей, в результате которых изменяется поведе-

Труды СПИИРАН. 2002. Вып. 1. Т. 2. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2002. Issue 1. V. 2. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

ние непрерывной части модели, определяемое функцией f для каждого режима. При объединении ситуационного варианта формализма машины конечных состояний (машины трансформации ситуаций) и гибридного автомата надо в рамках общего формализма специфицировать

- переключение режимов динамики непрерывной части;
- изменение динамики при изменении режима;
- скачкообразное изменение непрерывного состояния при переходе;
- непрерывную динамику.

Поскольку в ГА режимы представляются дискретными значениями, то в объединенном формализме их весьма естественно представлять аналогично представлению состояния дискретных элементов модели, т.е. описаниями ситуаций с помощью логических переменных. Это тем более удобно, что во многих случаях режим непрерывных элементов определяется состоянием дискретных (“Включен”, “Открыт” и т.п.). При этом с помощью правил (1) становится возможной спецификация переходов из одного режима в другой при изменении дискретных состояний. Этим обеспечивается координация непрерывных процессов относительно дискретно — событийных. Для спецификации переходов, вызванных выходом непрерывных состояний из заданных областей необходимо условия перехода включать в описания ситуаций, а для этого условия перехода надо выразить через логические переменные. В общем случае условие перехода задается областью значений непрерывного состояния. Эта область может быть представлена системой неравенств для отдельных составляющих (координат) непрерывного состояния, что обычно и делается при спецификации реальных ГА. Значения неравенств можно представить логическими переменными. Так для термостата можем иметь

$$g_1 = \begin{cases} \text{Истина, если } x_1 < 19, \\ \text{Ложь; иначе} \end{cases} \qquad g_2 = \begin{cases} \text{Истина, если } x_1 > 21, \\ \text{Ложь; иначе} \end{cases}.$$

Логические переменные условий переходов можно рассматривать как логическое описание состояния элемента с непрерывной динамикой (нахождение координат вектора непрерывных состояний в специфицируемой области). Включение этих переменных в описание ситуаций позволяет с помощью правил (2) специфицировать переходы между режимами непрерывных элементов, а также осуществлять координацию дискретно-событийных процессов в соответствии с состоянием непрерывных. В исследовательском прототипе среды моделирования (ENVICON [6]), реализующей рассматриваемый формализм, спецификация логических переменных переходов была реализована в форме правил вида:

$$((x_{j_1} \geq a_k + x_{j_2}) \wedge (x_{j_3} \leq b_k + x_{j_4})) \to g_k, \tag{3}$$

где $x \in X, g \in G \subset Q, a_k$ и b_k — константы, соответствующие условию перехода g_k.Здесь введено обозначение множества логических переменных переходов G. Включение в это правило двух неравенств принципиального значения не имеет. Это сделано лишь для удобства спецификации интервалов. Введение в неравенства помимо констант непрерывных переменных также не принципиально. Оно позволяет расширить выразительные возможности формализма, позволяя специфицировать зависимость условий переходов одних непрерывных переменных от текущих значений других, что часто встречается в реальных гибридных системах.

Теперь, когда ситуация может определять значение режима, для спецификации непрерывной части (векторное поле f) достаточно задать связь между

Труды СПИИРАН. 2002. Вып. 1. Т. 2. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2002. Issue 1. V. 2. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

соответствующими ситуациями и функциональными зависимостями между непрерывными переменными. Однако принципиальное значение имеет форма представления этих связей и зависимостей, так как они должны быть реализованы в рамках одной исполняющей процедуры, рассмотренной выше для событийной части модели. Состояние дискретно-событийного процесса определяется текущей ситуацией, т.е. текущими значениями входов и состояний. Эта процедура реализует пошаговый процесс трансформации ситуаций, при котором на данном шаге вычисляются значения дискретных состояний на следующем шаге в зависимости от текущих значений состояний и входных переменных. Таким образом, текущая ситуация (вместе с правилами трансформации (2)) полностью определяет следующую ситуацию. Ситуационная интерпретация процессов, на наш взгляд, является весьма плодотворной, она применима как к дискретно-событийным, так и к непрерывным процессам. Действительно, состояние непрерывного процесса через бесконечно малый промежуток времени полностью определяется его текущим состоянием (с учетом внешних воздействий) и законами соответствующей динамики. Естественно, погрешность реализации непрерывного процесса дискретным по времени процессом зависит от величины дискрета (длительности цикла исполняющей процедуры Δt).

В описываемом формализме мы использовали два правила для спецификации непрерывных состояний. Первое задает скорость, которую будет иметь данная координата вектора непрерывного состояния на следующем шаге в зависимости от текущих значений непрерывных состояний в случае истинности некоторой ситуации, определяющей режим непрерывной части модели:

$$S_j \to (\dot{x}_k' = c_{jk} + a_{jk} x_m), \; j,k,m \in N, \qquad (4)$$

где c_{jk}, a_{jk} — коэффициенты, соответствующие изменению k-ой переменной в данной ситуации S_j, $\dot{x}_k'$ — скорость специфицируемой данным правилом переменной на следующем шаге. Совокупность правил такого типа эквивалентна системе линейных разностных уравнений и позволяет описывать линейные динамические структуры произвольного вида. Исполняющая процедура должна на каждом шаге производить следующие вычисления переменных заданных правилами этого типа: $x_k' = x_k + \dot{x}_k \Delta t_i$, где x_k' — значение x_k на следующем шаге.

Второе использованное нами правило на каждом шаге вычисляет текущее значение ситуации левой части и в случае его истинности производит вычисление значения специфицируемой непрерывной переменной из своей правой части в соответствии с заданной функциональной зависимостью и текущими значениями аргументов. Вычисленные значения присваиваются этим переменным на следующем шаге.

$$S_j \to (x_k' = f_{jk}(x_m)), \; j,k,m \in N, \qquad (5)$$

где x_k' — значение x_k на следующем шаге, f_{jk} — функция, задающая изменение k-ой непрерывной переменной в ситуации S_j. С помощью правил этого типа могут быть реализованы интегро-дифференциальные и произвольные нелинейные функциональные (в том числе контакторные) зависимости. В правилах (4) и (5) символ = обозначает не операцию отношения, а операцию присваивания.

Естественно, что в одной и той же ситуации изменение одной непрерывной переменной может задаваться лишь одним из правил (4), (5). Для удобства

Труды СПИИРАН. 2002. Вып. 1. Т. 2. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2002. Issue 1. V. 2. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

реализации мы разбили множество непрерывных состояний на два подмножества по типу используемого для спецификации их элементов правил (X_v для правил (4) и X_f для правил (5)). Кроме того, мы включили в формализм множество входных непрерывных воздействий X_i, что является необходимым для модели реактивной системы. В результате для множества непрерывных переменных имеем $X = X_i \bigcup X_v \bigcup X_f$. Множество логических (дискретных) переменных V является объединением входных логических переменных Σ, состояний дискретных элементов Q и режимов непрерывных элементов G, т.е. $V = \Sigma \bigcup Q \bigcup G$. Множество описаний ситуаций S является подмножеством Булеана (множества всех подмножеств) множества логических переменных, т.е. $S \subset P(\Sigma \bigcup Q \bigcup G)$.

С учетом сказанного исследуемый формализм может быть представлен следующим кортежем:

$(\Sigma, Q, G, \sigma, \gamma, X_i, X_v, X_f, \delta_1, \delta_2, q_0, Init)$,

где $\sigma : S \times B \to Q \times B$ — функция трансформации состояний элементов дискретно-событийных процессов; реализована с помощью совокупности правил типа (2);

$\gamma : X \to G \times B$ - функция трансформации режимов элементов непрерывных процессов; реализована с помощью совокупности правил типа (3);

$\delta_1 : S \times B \times X \to X_v$ — функция изменения непрерывных состояний, задаваемых значениями скоростей; реализована с помощью совокупности правил типа (4);

$\delta_2 : S \times B \times X \to X_f$ — функция изменения непрерывных состояний, задаваемых функциями реализована с помощью совокупности правил типа (5);

$B = (\text{Ложь}, \text{Истина})$;

$q_0, Init$ — множества дискретных и непрерывных начальных состояний, соответственно.

Из приведенных описаний нетрудно видеть, что формализм обладает большой выразительной способностью для спецификации моделей гибридных реактивных систем. Он позволяет описывать как независимые параллельные дискретно-событийные, гибридные и непрерывные процессы, так и различные варианты координации их между собой, что достигается за счет включения в описание ситуации одного процесса переменных состояния дискретных элементов и режимов работы непрерывных других процессов.

В большинстве случаев имитационное моделирование реактивных систем проводится с целью отладки программной модели управляющей части системы для последующего использования ее в реальной системе. Поэтому эффективность исполняющей процедуры, реализующей модель, имеет часто решающее значение. Одной из целей настоящего исследования была оценка эффективности возможных реализаций моделей реальных систем, специфицированных с помощью данного формализма. Эффективность реализации определяется запаздыванием реакции системы на входное дискретное воздействие и погрешностью представления непрерывных переменных за счет дискретизации по времени. Как то, так и другое целиком определяется длительностью цикла трансформации состояний (цикла сканирования базы правил (2)–(5)). Высокое быстродействие современных компьютеров делает оправданным использование простых алгоритмов для реализации интегро-дифференциальных и функ-

Труды СПИИРАН. 2002. Вып. 1. Т. 2. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2002. Issue 1. V. 2. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

циональных зависимостей. Так в исследовательском прототипе среды имитационного моделирования (ENVICON) [6], основанной на описываемом формализме, интегральные и дифференциальные зависимости реализованы с помощью правил следующего вида, соответственно: $S_j \to (x_k^{'} = x_k + a_{jk} x_m \Delta t)$, $S_j \to (x_k^{'} = a_{jk}(x_m - x_m^{''}) / \Delta t)$, где a_{jk} коэффициент, соответствующий режиму и переменной, $x_k^{'}$ — значение переменной на следующем шаге, $x_m^{''}$ — значение переменной на предыдущем шаге. В этой среде была разработана имитационная модель автоматизированной промышленной установки по производству жидкого гелия. Установка содержит большое количество автоматических клапанов, вентилей, три компрессора, емкости и теплообменники, имеет 8 контуров автоматического управления и 3 режима работы (ручной, супервизорный и автоматический), большое количество датчиков температур, давлений и уровней. Для описания модели установки было использовано порядка 400 правил различных типов ((2)–(5)). При запуске этой модели на компьютере с процессором Pentium 3 с тактовой частотой 400 МГц длительность цикла обновления состояния не превышала 0,1 миллисекунды. Очевидно, что дискрет такого порядка вполне допустим при реализации большинства производственных систем управления реального времени.

Дальнейшее увеличение выразительных возможностей формализма может быть достигнуто за счет спецификации событий. С каждой ситуацией S_j можно связать два события — одно E_{ja} с ее возникновением (appearance) и второе E_{jd} с ее исчезновением (disappearance). Можно специфицировать составное событие, связанное с исчезновением одной ситуации S_j и возникновением другой S_k, определив его как конъюнкцию соответствующих событий E_{jd}, E_{ka}. Переменные, представляющие события, принципиально не отличаются от переменных, представляющих ситуацию, и могут аналогично последним использоваться в правилах (2), (4) и (5). Единственная их особенность состоит в том, что они могут иметь значение "Истина" только в течение одного шага цикла, непосредственно следующего за циклом, на котором возникла (исчезла) специфицированная ситуация. Использование событий для спецификации систем наряду с расширением возможностей является потенциальным источником ошибок. Поэтому их широкое применение возможно только после разработки и внедрения эффективных алгоритмических процедур верификации баз правил.

Экспериментальные исследования описанного ситуационного формализма спецификации гибридных систем, проведенные в ходе реализации имитационных моделей ряда тестовых систем и двух реальных промышленных установок, позволяют сделать вывод о его высокой эффективности и достаточной для спецификации большинства гибридных систем выразительности. Дальнейшие исследования в данном направлении необходимо, на наш взгляд, сосредоточить на развитии алгоритмических методов верификации описанных совокупностей правил, с целью решения проблем достижимости ситуаций и живучести моделей (отсутствие блокировок и зацикливаний).

Литература

[1] *Dion B., Dissoubray S.* Modeling and implementing critical real-time systems with SyncCharts/Esterel // Real Time Magazine 99–1

[2] *Girault A., Lee B., Lee* A. Hierarchical Finite State Machines with Multiple Concurrency Models // IEEE Transactions on computer-aided design of integrated circuits and systems, vol. 18, No. 6, June 1999. — 742-760 pp.

[3] *Henzinger T. A*. The Theory of Hybrid Automata // Proceedings of the 11th Annual IEEE Symposium on Logic in Computer Science (LICS 96). — 278-292 pp.

[4] *Lygeros J., Johanson K. H., Zhang J., Sastry Sh.* Fundamental Properties of Hybrid Dynamical Systems // Rio, Patras, Greece: Proceedings of Mediterranean Conference on Control and Automation (MED 2000).

[5] *Иванищев В. В*. Некоторые особенности интеллектуальных технологий моделирования // Pskov: International Conference “Intelligent Systems and Information Technologies in Control, June 2000. — 233-255 с.

[6] *Шпаков В. М*. Среда разработки ситуационных исполняемых спецификаций моделей систем управления // Pskov: International Conference “Intelligent Systems and Information Technologies in Control”, June 2000. — 316-319 с.