8 ОБОБЩЕННАЯ СХЕМА ВЫБОРА ЗАДАВАЕМЫХ ПАРАМЕТРОВ ДИАГНОСТИРУЕМОСТИ

В табл. 1 приводится выбор номенклатуры показателей безотказности, ремонтопригодности или комплексных для объектов вида I.

В табл. 2 приводится выбор номенклатуры показателей безотказности, ремонтопригодности или комплексных для объектов вида II.

ВЫВОДЫ

Рассмотренные показатели надежности и диагностируемости КС целесообразно использовать на этапе автоматизированного проектирования диагностического обеспечения КС. Это позволит более эффективно осуществлять поиск возможных возникающих дефектов на этапе эксплуатации КС.

ПЕРЕЧЕНЬ ССЫЛОК

1. *Харченко В. С.* Гарантоспособность и гарантоспособные системы: элементы методологии // Научно-технический журнал «Радиоэлектронные и компьютерные системы». – 2006. – № 5 (17). – С. 7–20.
2. *Кривуля Г. Ф., Липчанский А. И., Механна Сами, Зидат Хабис.* Диагностика компьютерных сетей с использованием экспертных систем // Вестник ХГТУ. – 2004. – № 1 (19). – С. 11–16.
3. *Кривуля Г. Ф., Бабич А. В., Липчанский А. И., Шкиль А. С.* Применение экспертных систем реального времени для диагностики компьютерных систем // Науково-технічний журнал «Інфомаційно-керуючі системи на залізничному транспорті». – Харків. – 2004. – № 1. – С. 67–73.

Надійшла 2.03.07

У статті розглядається класифікація основних видів показників для оцінки надійності та здатності до діагностування комп'ютерних систем. Розглянуті показники безвідмовної роботи, ремонтоздатності, довговічності, експлуатаційної готовності, комплексні показники здатності до діагностування. Наведена класифікація відмов.

In given article is considered the classification of the basic kinds of parameters for a rating of reliability and diagnozability of computer systems. Such parameters were considered: of non-failure operation, of maintainability, of durability, of operational readiness, complex parameters diagnozability. Was given the classification of failures.

УДК 621.3

М. А. Новотарский

АЛГЕБРА ПРОЦЕССОВ ДЛЯ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ СЛОЖНЫХ ДИСКРЕТНЫХ СИСТЕМ

В работе представлена алгебра процессов, как инструмент для формального описания сложных дискретных систем. Рассмотрена структура множеств, на которых определены переменные и выражения алгебры процессов. Описаны синтаксические конструкции и семантические правила выполнения операций, позволяющие задавать формальное описание имитационных моделей с реальной рабочей нагрузкой. Для определения адекватности модели предложено использовать принципы эквивалентности, базирующиеся на конгруэнтности, строгом и слабом взаимном подобии.

ВВЕДЕНИЕ

Ежедневно мы сталкиваемся с огромным количеством сложных дискретных систем, которые, большей частью, являются продуктом человеческой деятельности. Это и телекоммуникационные сети, и персональные компьютеры, и многое другое. Создавать или исследовать такие системы практически невозможно без предварительной проверки принимаемых технических решений, поскольку компоненты систем, как правило, дорогостоящи и требуют существенных затрат на их производство. Одним из эффективных инструментов для проверки и предварительного анализа сложных систем является компьютерное моделирование. Построение моделей особенно актуально, когда речь идет о разработке мультипроцессорных систем, ориентированных на реализацию параллельных асинхронных алгоритмов. При этом не всегда удается эффективно использовать традиционные подходы к формализации исходной задачи, поскольку в данном случае модели должны отображать особенности параллелизма и асинхронного взаимодействия компонент. Например, ориентация на известные языки моделирования заведомо ставит исследователя в зависимость от способов функционирования и взаимодействия компонент, принятых в том или ином языке, а попытка эмулировать в данном языке свои способы взаимодействия существенно снижает адекватность модели. В связи с этим значительно возрос интерес к поиску новых подходов к формализации, которые основываются на понятиях, способных более адекватно отображать различные аспекты функционирования параллельных компонент и их взаимодействие. Все они лежат в русле дальнейшего совершенствования уже известных технологий программирования компьютерных систем.

Первые программисты напрямую использовали машинные коды. Поэтому им приходилось иметь дело с сумматорами, регистрами, ячейками памяти и другими элементами, отображающими архитектуру системы. Существенным шагом к повышению сложности формализации стали языки программирования высокого уровня. Они позволили заменить понятия, связанные с архитектурой, на более близкие к тем, которыми пользуются при описании проблемы. Вместо байтов и регистров стало возможным использовать числа и строки. Языки моделирования добавили к традиционному перечню еще и специализированные структуры данных, облегчающие описание сложных компонент и их взаимодействие. Очевидно, что упомянутые сущности подняли формализацию на более высокий уровень. Однако существенный рост сложности объектов моделирования и нетривиальный характер их взаимодействия привели к ситуации, когда при использовании языков высокого уровня стало весьма затруднительно строить корректные модели. Возникла объективная необходимость введения новых понятий – процессов, событий, активностей – позволяющих представлять модель на более высоком уровне абстракции. Теории, использующие эти понятия в качестве базовых, получили название алгебр процессов. Главной отличительной чертой алгебры процессов является возможность представления в виде математически строгих выражений систем, состоящих из параллельно развивающихся и взаимодействующих процессов. Основы алгебры процессов, называемой «исчисление коммуникационных систем», впервые изложены в [1] и послужили толчком к бурному развитию этого направления. В результате были созданы алгебры процессов, позволяющие исследовать различные аспекты поведения сложных систем. Общим для этих алгебр является то, что все они созданы для описания сложных дискретных систем, базируются на ряде аксиом, задающих синтаксис выражений и семантические правила, и содержащих признаки поведенческой эквивалентности, позволяющие сравнивать системы, поведение которых неразличимо для внешнего наблюдателя.

Эта работа посвящена проблемам дальнейшего развития теории алгебр процессов, ориентированных на описание параллельных процессов, характеризующихся широким спектром взаимодействий, включая асинхронный характер взаимодействия. Еще одна существенная черта предлагаемой алгебры процессов состоит в ее ориентации на создание имитационных моделей с реальной рабочей нагрузкой. Такой подход позволяет значительно повысить поведенческую эквивалентность модели объекту моделирования. Простые и прозрачные способы представления реальной нагрузки осуществлены путем расширения спектра синтаксических конструкций, а новые семантические правила отображают не только традиционные для алгебр процессов коммуникационные аспекты функционирования модели, но и допускают использование прикладных алгоритмов обработки входных данных. Адекватность моделей, построенных с использованием данной алгебры процессов, строго очерчена путем доказательства конгруэнтности выражений, а также обоснования правил строгого и слабого взаимного подобия.

1 БАЗОВЫЕ ПОНЯТИЯ

Алгебра процессов для моделирования с дискретными событиями использует некоторые свойства классических базовых алгебр процессов [1], а также новые тенденции развития, возникшие в результате применения алгебр процессов к решению проблем имитационного моделирования с дискретными событиями [2, 3]. Парадигмы алгебры процессов предполагают наличие в системе активных компонент и взаимодействие между ними. В [1] активные компоненты называются процессами, которые совершают активности, представляющие дискретные действия системы. Любая активность в данной алгебре процессов может принадлежать множеству взаимодействия Λ или множеству внутренних активностей Ψ. Множество взаимодействия Λ содержит подмножество активностей ввода Δ и подмножество активностей вывода $\bar{\Delta}$:$\Lambda = \Delta \cup \bar{\Delta}$.

Собственно акт взаимодействия имеет место только при наличии источника и приемника информации, что обуславливает его бинарный характер. Поэтому каждая активность $a(x) \in \Delta$ всегда имеет сопряженную активность $\bar{a}(x) \in \bar{\Delta}$, и это именно та сопряженная активность, с которой готова синхронизироваться активность $\bar{a}(x)$. Скобки (x) содержат значения переменной, подлежащей вводу или выводу, и могут упускаться в случаях, если важным есть только факт взаимодействия.

Множество внутренних активностей Ψ состоит из подмножеств обработки Φ и подмножества задержек T:$\Psi = \Phi \cup T$.

Подмножество обработки включает активности обработки $\varphi \in \Phi$. Каждая активность такого типа представляет определенные действия процесса по преобразованию информации. Исходя из этого определения, очевидно, что активность обработки может также представлять процессы, и тем самым обеспечивает неявную агрегативность структур описания процессов. В алгебре процессов можно исключить внутреннее поведение процесса и рассматривать его как «черный ящик». В этом случае возникает возможность построить сложную имитационную модель, исключив из описания более простые части, поведение которых не представляет интереса. Такое невидимое поведение моделируется с помощью активностей $\tau \in T$, заменяющих внутренние действия задержкой, выраженной в единицах модельного времени. Активности типа τ также используют в случае явного выделения модельного времени для выполнения активностей обработки.

Таким образом, полный алфавит активностей $Act = \Lambda \cup \Psi$ объединяет множество активностей взаимодействия Λ и множество внутренних активностей Ψ.

Процесс в алгебре процессов задается выражением, содержащим активности как независимые или зависимые переменные. Простейший константный процесс, освобождающий все ресурсы, задается активностью nil, где $nil \in \Phi$. Если процесс содержит более одной активности, то последовательность их свершения регулируется префиксным представлением $\varphi.P$. Смысл префиксного представления состоит в том, что сначала свершается активность φ, а затем процесс ведет себя как процесс P. Примером элементарного процесса P_1 служит выражение: $P_1 := begin.end.nil$, где $\{begin, end, nil\} \subset \Phi$.

Символ := задает присвоение значения выражения в правой части процессу P_1. Действия процесса, согласно данному выражению, состоят из прямой последовательности трех активностей. Активность $begin$ выполняет действия, связанные с началом функционирования процесса P_1, end выполняет действия, связанные с завершением процесса P_1, nil освобождает ресурсы и удаляет процесс P_1 из модели.

2 *СИНТАКСИС*

Синтаксис алгебры процессов для имитационного моделирования систем с реальной рабочей нагрузкой базируется на использовании типов переменных для параметров Var и типов переменных для процессов Pvar.

Множество переменных типа Var обеспечивает временное хранение параметров и включает простые переменные $x, y, z, \ldots$; индексированные переменные $x\{i\}$, $(y\{i\}, z\{i\}\ldots)$, где i – индекс; ссылочные переменные $\hat{x}, \hat{y}, \hat{z}, \ldots$. Множество переменных типа Pvar временно хранит экземпляры процессов и включает простые переменные $X, Y, Z, \ldots$; индексированные переменные $X\{i\}, Y\{i\}, Z\{i\}, \ldots$, где i – индекс; ссылочные переменные $\hat{X}, \hat{Y}, \hat{Z}, \ldots$.

Выражения для представления функциональных зависимостей и времени составляют множество выражений Exp, заданное на множестве элементарных математических операций.

Выражения для процессов объединены во множество Pexp, включающее следующие операции:

– префиксация – $\varphi.P$, $\langle\tau\rangle.P$, $\langle\tau \leftarrow t\rangle(.P)$, $a(x).P$, $\bar{a}(x).P$;

– выбор – $\sum_{i \in I} P_i$, $\sum_{i \in I} v_i!P_i$, $\sum_{i \in I} w_i\#P_i$;

– параллельная композиция и взаимодействие – $P|Q|\ldots|C$, $P \underset{L}{\leftrightarrow} Q|\cdots|C$, $P \underset{L}{\rhd} Q|\cdots|C$, $P \underset{L}{\lhd\rhd} Q|\cdots|C$;

– переименование и рестрикция – $P[K]$, $P \setminus A$;

– клонирование, вставка и сокрытие – $k \times P$, $C\{P\}$, $\varphi_P(C)$;

– рекурсия и цикл – $\Re(X = P)$, $cycle[x](X = P)$.

Рассмотрим кратко смысл упомянутых операций. Префиксная задержка $\langle\tau\rangle.P$ задает процесс с начальной задержкой на τ единиц модельного времени, а параметрическая префиксная задержка $\langle\tau \leftarrow t\rangle.P$ описывает действия процесса, для которого сначала определяется значение параметра τ путем присвоения ему результата вычисления выражения t. После задержки τ в обоих случаях процессы ведут себя как процесс P.

Префиксный ввод $a(x).P$ задает действия процесса по вводу некоторого значения k с последующим присвоением его переменной x. Префиксный вывод $\bar{a}(x).P$ задает действия процесса по выводу того значения переменной x, которое она приобрела на момент свершения активности вывода $\bar{a}(x)$.

Операции выбора предназначены для выбора одного процесса из множества процессов с целью его дальнейшего развития. Недетерминированный выбор $\sum_{i \in I} P_i$ обеспечивает определение одного из процессов $P_1 + \ldots + P_i + \ldots + P_I$ в момент времени, который называется моментом выбора. Управляемый выбор $\sum_{i \in I} v_i!P_i$ реализует выбор процесса P_i, снабженного логической переменной v_i, принявшей истинное значение. Вероятностный выбор $\sum_{i \in I} w_i\#P_i$ обеспечивает выбор процесса P_i с вероятностью w_i при условии $w_1 + \ldots + w_i + \ldots + w_I = 1$.

Операции параллельной композиции и взаимодействия определяют взаимное поведение процессов, которые развиваются параллельно. Операцию $P|Q|\ldots|C$ применяют в случаях, если нет необходимости акцентировать внимание на характере взаимодействия между параллельными процессами или если такого взаимодействия не существует. Операция может применяться для определения параллелизма группы процессов без детализации характера связей между ними. Операция взаимодействия определяет поведение совокупного процесса, который состоит из двух процессов, развивающихся параллельно.

Синхронным взаимодействием $P \underset{L}{\leftrightarrow} Q|\ldots|C$ процесса P с параллельной композицией процессов $Q|\ldots|C$ будем называть систему, реализующую передачу информации от передатчика P к приемникам $Q|\ldots|C$ в случае одновременного свершения комплементарных событий из множества $L \subseteq \Lambda$.

В отличие от операции выбора, при взаимодействии рассматривают процессы, активности которых пользуются собственными скрытыми ресурсами. Это разрешает им развиваться независимо и использовать свои ресурсы в случае, если события, которые их вызывают, не входят во множество событий взаимодействия. При возникновении события, которое входит во множество событий взаимодействия, происходит синхронизация развития параллельных процессов с целью выполнения общей активности.

Управляемой передачей $P \underset{L}{\triangleright} Q|\cdots|C$ от процесса P к параллельной композиции $Q|\ldots|C$ называют систему, состоящую из процесса-передатчика P и множества процессов-приемников $Q|\ldots|C$, которые параллельно развиваются и взаимодействуют в момент свершения в процессе P активности из множества взаимодействия $L \in \Lambda$. Главное отличие этой операции от операции синхронного взаимодействия состоит в том, что передача данных происходит непосредственно в момент свершения активности вывода $\bar{a}(x)$ в процессе P, а композиция процессов $Q|\ldots|C$ всегда готова к приему информации.

Асинхронным взаимодействием $P \underset{L}{\triangleleft\!\triangleright} Q|\cdots|C$ процесса P и параллельной композиции $Q|\ldots|C$ будем называть систему, которая состоит из процесса P и множества параллельно развивающихся процессов $Q|\ldots|C$, способных передавать или принимать информацию от процесса P в соответствии с множеством событий взаимодействия $L \in \Lambda$ независимо друг от друга. При асинхронном взаимодействии каждый из процессов, входящих в систему, может инициировать активность ввода или вывода независимо от готовности сопряженной активности. Поэтому задержки для синхронизации при такой организации взаимодействия процессов не возникает.

Операции переименования и сжатия относят к операциям группового редактирования, поскольку они обеспечивают модификацию группы активностей процесса одновременно. Операция переименования $P[K]$ производит замену всех активностей процесса P с помощью функции K, а операция сжатия $P \setminus \Lambda$ блокирует в процессе P все активности, входящие во множество A.

Операции клонирования, вставки и сокрытия предназначены для динамической модификации множества компонент системы в ходе моделирования. Операция клонирования $k \times P$ представлена системой, которая состоит из k идентичных процессов P, развивающихся параллельно. Операция вставки $C\{P\}$ задает процесс C, включающий процесс P как субпроцесс. Операция сокрытия $\varphi_P(C)$ описывает систему, состоящую из процесса C, в котором все активности, входящие в субпроцесс P, заменяются внутренней активностью φ.

Операции рекурсии и цикла предназначены для многократного повторения некоторой последовательности активностей, входящих в процесс. Операция рекурсии $\Re(X = P)$ образует систему, состоящую из рекурсивно повторяемой последовательности активностей процесса P, присвоенных переменной типа Pvar. Операция цикла $cycle[x](X = P)$, подобно рекурсии, состоит из циклически повторяемой последовательности процесса P, но количество таких повторений ограничено значением переменной x.

3 СЕМАНТИКА

Операционная семантика процессов моделирования, представленных данной алгеброй, сводится к определению переходов, представляющих изменения состояний имитационной модели. Определим динамику выполнения операций, используя нотацию Плоткина [4], представляющую операцию в виде дроби, знаменатель которой содержит общую запись операции, а числитель – возможные варианты ее выполнения.

Операции префиксации $\overline{\varphi.P \xrightarrow{\varphi} P}$, $\overline{\langle\tau\rangle.P \xrightarrow{\langle\tau\rangle} P}$, $\overline{\langle\tau \leftarrow t\rangle.P \xrightarrow{\langle t\rangle} P[t/\tau]}$, $\overline{a(x).P \xrightarrow{a(k)} P[k/x]}$ и $\overline{\bar{a}(x).P \xrightarrow{\bar{a}(m)} P}$ не имеют числителя, поскольку существует единственный вариант их выполнения, заключающийся в продолжении текущего процесса после выполнения активности. Отличия между данными операциями обусловлены различными состояниями переменных процесса и способами их замещения.

Количество вариантов выполнения недетерминированной операции выбора равно количеству процессов, принимающих участие в данной операции:

$$\frac{P_1 \xrightarrow{\alpha} P_1'}{\sum\limits_{i \in I} P_i \xrightarrow{\alpha} P_1'},\ \ldots,\ \frac{P_k \xrightarrow{\beta} P_k'}{\sum\limits_{i \in I} P_i \xrightarrow{\beta} P_k'},\ \ldots,$$
$$\frac{P_I \xrightarrow{\gamma} P_I'}{\sum\limits_{i \in I} P_i \xrightarrow{\gamma} P_I'}.$$

Операция управляемого выбора в нотации алгебры процессов в знаменателе содержит общую запись возможного выбора. Числитель указывает на то, что осуществляется положительный выбор того процесса, для которого соответствующая логическая переменная v_i принимает истинное значение:

$$\frac{\{P_i|v_i = true\}\xrightarrow{\alpha} P'}{\sum_{i\in I} v_i!\,P_i \xrightarrow{\alpha} P'}.$$

Вероятностный выбор структурно напоминает операцию управляемого выбора, за исключением того, что в данном случае задаются вероятности для всех процессов, принимающих участие в выборе. Акт выбора всегда осуществляется как результат выполнения вероятности:

$$\frac{\{P_i|w_i\}\xrightarrow{\alpha} P'}{\sum_{i\in I} w_i\#\,P_i \xrightarrow{\alpha} P'}.$$

Нотация параллельной композиции задает общую формулу взаимодействия между двумя процессами, входящими в группу параллельных процессов:

$$\frac{P\xrightarrow{a(x)} P' \quad Q\xrightarrow{\bar a(x)} Q'}{P|\,Q\xrightarrow{\tau} P'|\,Q'}.$$

Конкретизация допустимых видов взаимодействия реализована с помощью операций взаимодействия. Динамика взаимодействия в алгебре процессов включает операции управляемой передачи, синхронного взаимодействия и асинхронного взаимодействия. Отсутствие взаимодействия рассматривается как параллельное выполнение процессов. Нотации этих операций имеют вид:

$$\frac{P\xrightarrow{\bar a(x)} P' \quad Q\xrightarrow{a(x)} Q'\cdots C\xrightarrow{a(x)} C'}{P \underset{L}{\triangleright} Q|\cdots|C\xrightarrow{\tau} P' \underset{L}{\triangleright} Q'|\cdots|C'}$$ – управляемая передача,

$$\frac{P\xrightarrow{\bar a(x)} P' \quad Q\xrightarrow{a(x)} Q'\cdots C\xrightarrow{a(x)} C'}{P \underset{L}{\leftrightarrow} Q|\cdots|C\xrightarrow{\tau} P' \underset{L}{\leftrightarrow} Q'|\cdots|C'}$$ – синхронное взаимодействие,

$$\frac{P\xrightarrow{\bar a(x)} P' \quad Q\xrightarrow{a(x)} Q'\cdots C\xrightarrow{a(x)} C'}{P \underset{L}{\triangleleft\!\triangleright} Q|\cdots|C\xrightarrow{\tau} P' \underset{L}{\triangleleft\!\triangleright} Q'|\cdots|C'}$$ – асинхронное взаимодействие.

При управляемой передаче инициатором действия является префиксный процесс $\bar a(x).P$. Процессы параллельной композиции $Q|\ldots|C$ постоянно готовы к приему информации благодаря префиксации $a(x).Q|\ldots|a(x).C$. Передача происходит путем одновременного свершения активности $\bar a(x)$ в процессе-передатчике и активностей $a(x)$ в процессах-приемниках. Иначе композиция процессов $Q|\ldots|C$ будет находиться в состоянии ожидания приема данных.

В случае синхронного взаимодействия оба процесса могут инициировать взаимодействие. Первый из процессов, прибывших в точку взаимодействия, отсылает процессу-партнеру активность и останавливается до момента свершения сопряженной активности.

Операция асинхронного взаимодействия предполагает наличие внутренних активностей, обеспечивающих возможность продолжения развития процесса-инициатора взаимодействия и выдачу информации процессу-приемнику в момент активации в нем активности $a(x)$.

Операции переименования и сжатия представлены в алгебре процессов нотациями:

$$\frac{P\xrightarrow{\alpha} P'}{P[K]\xrightarrow{K(\alpha)} P'[K]},$$

$$\frac{P\xrightarrow{\alpha} P'}{P\setminus A\xrightarrow{\alpha} P'\setminus A}\left(\{\alpha,\bar\alpha\}\not\subset A\right).$$

Эти операции, в отличие от операций выбора и взаимодействия, работают с одним процессом, но предназначены для организации групповых модификаций его активностей. Источником модификаций для операции переименования есть функция K, а для операции сжатия – множество активностей A.

Динамика операции клонирования процессов представлена выражением:

$$\frac{P\xrightarrow{\alpha} P_1' \quad P\xrightarrow{\alpha} P_2'\ldots P\xrightarrow{\alpha} P_k'}{P\xrightarrow{\alpha} P_1'|P_2'|\ldots|P_k'}.$$

Суть операции $k\times P$ сводится к генерации группы из k процессов на основе шаблона P.

Операции рекурсии и цикла представлены нотациями, которые отображают динамику их выполнения:

$$\frac{P\{\Re(X=P)/X\}\xrightarrow{\alpha} P'}{\Re(X=P)\xrightarrow{\alpha} P'}$$ – рекурсия,

$$\frac{P\{cycle[x-1](X=P)/X\}\xrightarrow{\alpha} P'}{cycle[x](X=P)\xrightarrow{\alpha} P'}(x\ge 0)$$ – цикл.

Операции состоят из трех основных этапов: инициализации элемента цикла путем присвоения значений процесса переменной X, выполнения созданного экземпляра процесса и проверки условий завершения. Проверка условий завершения рекурсивной операции всегда зависит от внутреннего состояния процесса P, а операция цикла использует значение счетчика x для завершения своей работы.

Операция вставки $C\{P\}$ обеспечивает возможность иерархического построения описания системы путем объединения процессов, входящих в разные уровни стратификации модели. Динамика выполнения данной операции соответствует нотации:

$$\frac{C\xrightarrow{a(X)} C'}{C\{X\}\xrightarrow{a(P)} C'\{P\}},\ \text{где } X\in \mathrm{PVar}.$$

Выполнение активности, инициирующей операцию вставки, сводится к присвоению переменной процесса X значения некоторого процесса P, ссылку на который содержит упомянутая активность. В ходе развития процесс C использует процесс P как субпроцесс.

Операция сокрытия тоже используется при построении и исследовании стратифицированных моделей. С помощью этой операции становится возможным скрывать детализацию функционирования некоторых компонент системы с целью упрощения процесса моделирования. Операция сокрытия представлена нотацией:

$$\frac{C \xrightarrow{\alpha} C'}{\varphi_P(C) \xrightarrow{\varphi} \varphi_P(C')}, \text{ где } a \in P.$$

Выполнение этой операции сводится к замене некоторой последовательности активностей, составляющих процесс P, одной внутренней активностью φ.

4 ЭКВИВАЛЕНТНОЕ ПОВЕДЕНИЕ

Синтаксис и семантика алгебры процессов задают способ представления объектов модели. Но для того, чтобы модель была адекватной, необходимо придерживаться соответствующих принципов эквивалентности поведения, базирующихся на эквивалентном отношении. Пусть P – процесс, тогда эквивалентным отношением R называется множество $R \subseteq P \times P$ с элементами $(p, q) \in R$ или pRq, удовлетворяющее условиям рефлексивности, симметричности и транзитивности. Рефлексивность обуславливает эквивалентность тождественного отношения, то есть pRq при $p \in P$; симметричность выражает тот факт, что из pRq следует qRp при $p, q \in P$; а транзитивностью называют свойство, для которого из pRq и qRz вытекает, что pRz при $p, q, z \in P$.

Для определения эквивалентности поведения применяют также конгруэнтность, то есть сохранение эквивалентного отношения на разных уровнях стратифицированных моделей. Пусть P и Q − процессы, связанные эквивалентным отношением PRQ. Такие процессы будем называть конгруэнтными, если эквивалентное отношение сохраняется при $C\{P\}RC\{Q\}$, где $C\{\ \}$ − процесс высшего уровня, использующий в качестве субпроцесса процесс P или Q. Благодаря конгруэнтности эквивалентность поведения может определяться как видимое поведение процессов, а не через их структуру, выраженную количеством допустимых состояний и переходов между ними. При таком подходе эквивалентные процессы распознаются наблюдателем как одинаково взаимодействующие с окружающей средой, а их взаимодействия приводят к одинаковому результату.

Строгое обоснование эквивалентности поведения базируется на использовании понятия взаимного подобия. Преимущество применения такого подхода состоит в возможности ссылки на определение подобия, исходя из общей теории систем и построения доказательств на основе этого определения. В зависимости от того, принимаются ли во внимание скрытые активности, взаимное подобие разделяют на строгое и слабое.

Поскольку алгебра процессов может отображать бесконечное количество систем, отвечающих ее синтаксическим конструкциям, то для конкретного определения взаимного подобия рассмотрим маркированную систему с переходами, заданную кортежем $\left(\Pi, A, \left\{\xrightarrow{a} \middle| a \in A\right\}\right)$, где Π − множество процессов; A − множество активностей; $\xrightarrow{a}$ − множество переходов: $\xrightarrow{a} \subseteq \Pi \times A \times \Pi$. Пусть в данной маркированной системе с переходами существует отношение $R \subseteq \Pi \times \Pi$ и процессы $P, Q \subseteq \Pi$ с допустимыми состояниями $p, p' \in P$, $q, q' \in Q$ при $(p, q) \in R$. Тогда если для произвольной активности a для каждого перехода $p \xrightarrow{a} p'$ существует состояние q', такое, что $q \xrightarrow{a} q'$ при $(p', q') \in R$, и для каждого перехода $q \xrightarrow{a} q'$ существует состояние p', такое, что $p \xrightarrow{a} p'$ при $(p', q') \in R$, то отношение R будем называть строгим взаимным подобием, если $a \in Act$. Состояния p и q строго взаимно подобны $p \sim q$, если существует отношение R, содержащее (p, q) при условии $a \in Act$.

Рассмотренное строгое взаимное подобие − это основной вид эквивалентности, позволяющий разрабатывать сценарии сравнения сложных дискретных систем. Оно объединяет все допустимые отношения взаимного подобия и потому является наибольшим взаимным подобием. Все операторы рассматриваемой алгебры процессов имеют те или иные свойства строгого взаимного подобия. Но очевидно, что применение данного вида эквивалентности не является исчерпывающим для исследования сложных систем. Отношение строгого взаимного подобия строится с учетом всех без исключения активностей, входящих в алфавит системы. Поэтому затруднительно выполнять обобщающие оценки, учитывающие только некоторое заданное подмножество свойств. Известный механизм обобщения базируется на применении методик вертикальной и горизонтальной стратификации. В рамках рассмотренных подходов упомянутые методики используют операции вставки и сокрытия для стратификации на уровне процессов. С целью абстрагирования от внутреннего поведения системы на уровне отдельных активностей, например, в классической алгебре процессов CCS [1], используют специальный тип активности − внутренняя активность. Однако такой подход нуждается в усовершенствовании принципов эквивалентности с целью распространения их на процессы, в которых существует скрытое внутреннее поведение. Проблема состоит в том, что простое исключение внутренних активностей некорректно, поскольку они могут инициировать видимые активности. При этом создается тупиковая ситуация, нарушающая информационную целостность системы. Для решения этой проблемы вводят новый тип перехода $\overset{a}{\Rightarrow} \subseteq \Pi \times \Pi$ для произвольных активностей $a \in A$ в соответствии с правилом:

$$\overset{a}{\Rightarrow} := \begin{cases} \xrightarrow{\varphi} \circ \xrightarrow{a} \circ \xrightarrow{\varphi} & \text{при} \quad a \neq \varphi, \\ \xrightarrow{\varphi} & \text{при} \quad a = \varphi, \end{cases}$$

где φ – внутренняя активность, $\circ$ – символ, указывающий на допустимость вставки в этом месте композиции бинарных отношений.

Пусть в маркированной системе с переходами $\left(\Pi, A, \left\{ \overset{a}{\Rightarrow} \middle| a \in A \right\}\right)$ существует отношение $R \subseteq \Pi \times \Pi$ и процессы $P, Q \subseteq \Pi$ с допустимыми состояниями $p, p' \in P$, $q, q' \in Q$ при $(p, q) \in R$. Тогда если для произвольной активности a для каждого перехода $p \xrightarrow{a} p'$ существует состояние q' такое, что $q \overset{a}{\Rightarrow} q'$ и $(p', q') \in R$, и для каждого перехода $q \xrightarrow{a} q'$ существует состояние p' такое, что $p \overset{a}{\Rightarrow} p'$ и $(p', q') \in R$, то отношение R будем называть слабым взаимным подобием или наблюдаемой эквивалентностью. Состояния p и q слабо взаимно подобны $p \approx q$, если существует наблюдаемая эквивалентность R, содержащая (p, q).

Строгое и слабое взаимные подобия состояний лежат в основе понимания взаимного подобия процессов. Два процесса P и Q называют строго взаимно подобными $P \sim Q$ или слабо взаимно подобными $P \approx Q$ тогда и только тогда, когда для произвольной активности a существует отношение соответственно строгого или слабого взаимного подобия R такое, что PRQ.

Пусть P, Q, G – процессы. Если $P \sim Q$, то:

– $a.P \sim a.Q$ для произвольной активности a;

– $\langle\tau\rangle.P \sim \langle\tau\rangle.Q$ для произвольной префиксной задержки $\langle\tau\rangle$;

– $P + G \sim Q + G$ для произвольного процесса G;

– $P|G \sim Q|G$ для произвольного процесса G;

– $P[K] \sim Q[K]$ для произвольной функции переименования K;

– $P \setminus \mathrm{A} \sim Q \setminus \mathrm{A}$ для произвольного множества активностей A;

– $k \times P \sim k \times Q$ для произвольного коэффициента клонирования k;

– $\Re(X = P) \sim \Re(X = Q)$, где X переменная процесса;

– $cycle[x](X = P) \sim cycle[x](X = Q)$, где x – индекс цикла.

Для P, Q и активности a справедливы такие наблюдаемые эквивалентности при $P \approx Q$:

– $\langle\tau\rangle.\varphi.P \approx \langle\tau\rangle.P$;

– $a.\varphi.P \approx a.P$;

– $P + \varphi.P \approx \varphi.P$;

– $a.(P + \varphi.Q) \approx a.(P + \varphi.Q) + a.Q$.

ВЫВОДЫ

Большой интерес к использованию алгебр процессов для формального описания сложных систем вызван, прежде всего, возможностью представления правил функционирования и взаимодействия компонент системы в виде набора математических выражений. Такой подход позволяет оценить адекватность модели на различных этапах ее жизненного цикла. Большинство известных алгебр процессов [1], [3] ориентировано на описание и исследование коммуникаций между структурными элементами систем. Однако для представления моделей с реальной рабочей нагрузкой необходимо иметь возможность задавать алгоритмы функционирования компонент, обрабатывающие внешние потоки данных. С этой целью в предлагаемой алгебре процессов введено множество внутренних активностей Ψ, каждая активность которого представляет определенные действия процесса по преобразованию информации. Расширение операций выбора и взаимодействия, а также введение операций клонирования и сокрытия продиктовано стремлением расширить сферу использования данной алгебры и гибкость представления с ее помощью сложных систем. Строгое обоснование адекватности базируется на использовании понятия эквивалентного отношения в модели, заданной маркированной системой с переходами. Свойство конгруэнтности введено с целью обеспечения возможности вертикальной и горизонтальной стратификации модели. Определены строгое и слабое взаимные подобия, а также правила их сохранения при выполнении базовых операций.

ПЕРЕЧЕНЬ ССЫЛОК

1. *Milner R.* Calculus of Communicating Systems // Lecture Notes in Computer Sciences. – 1980. – Vol. 92. – P. 260.
2. *Milner R.* Communication and Concurrency. – Prentice Hall. – 1995. – P. 272.
3. *Bergstra J. A., Middelburg C. A.* Process algebra for hybrid systems // Theoretical Computer Science. – 2005. – Vol. 335, No. 2/3. – PP. 215–208.
4. *Plotkin G. D.* A Structured Approach to Operational Semantics. Technical Report DAIMI FM-19, Computer Science Department, Aarhus University. – 1981. – P. 133.

Надійшла 31.01.07

In the paper the process algebra as the tool for the formal description of complex discrete systems is submitted. The structure of sets on which variables and expressions of process algebra are determined is considered. Syntactic structures and semantic rules of performance of the operations allowing to set the formal description for simulations with real working load are described. For definition of adequacy of model it is offered to use the principles of equivalence basing on congruence, strong and weak bisimilarity.

У роботі представлена алгебра процесів як інструмент для формального опису складних дискретних систем. Розглянуто структуру множин, на яких визначені змінні і вирази алгебри процесів. Описано синтаксичні конструкції і семантичні правила виконання операцій, що дозволяють задавати формальний опис імітаційних моделей з реальним робочим навантаженням. Для визначення адекватності моделі запропоновано використовувати принципи еквівалентності, що базуються на конгруентності, строгій і слабкій взаємній подібності.

УДК 62-52.087

А. Л. Становский, Т. В. Лысенко, Т. И. Носенко, Е. В. Лысенко

СИНХРОНИЗИРУЮЩЕЕ УПРАВЛЕНИЕ ЭКСТРЕМУМАМИ ОТДАЛЕННЫХ СОСТОЯНИЙ

Определено понятие «событие» в теории синхронизирующего управления как поименованное состояние. Выполнена классификация событий. Предложен метод нейросеточной экстраполяции дискретных числовых временных трендов, основанный на замене значений тренда в узловых точках на значения производной интерполяционного полинома в этих же точках.

ВВЕДЕНИЕ

Теория синхронизирующего управления предполагает поиск и реализацию такого внешнего воздействия на объект, которое обеспечивает в будущем (иногда, весьма отдаленном) совпадение некоторых событий, ожидаемых в его подсистемах [1].

ПОСТАНОВКА ЗАДАЧИ

Понятие «событие» определено в теории вероятностей как «всякий результат или исход испытания» [2] и является аксиоматической основой для построения булевой алгебры событий.

Пусть некоторая система Ω определена в n_Ω-мерном пространстве состояний $\mathbf{\Omega}(\tau)$, где τ – время. Разобьем $\mathbf{\Omega}(\tau)$ на k множеств:

$$\mathbf{Y}_1(\tau) = \{y_{11}(\tau),\ y_{12}(\tau),\ \ldots,\ y_{1n_1}(\tau)\},$$
$$\mathbf{Y}_2(\tau) = \{y_{21}(\tau),\ y_{22}(\tau),\ \ldots,\ y_{2n_2}(\tau)\},$$
$$\ldots$$
$$\mathbf{Y}_k(\tau) = \{y_{k1}(\tau),\ y_{k2}(\tau),\ \ldots,\ y_{kn_k}(\tau)\}$$

с размерностями n_1, n_2, … n_k, соответственно, которые могут частично пересекаться между собой. Назовем части общей системы Ω, определенные на $\mathbf{Y}_1(\tau)$, $\mathbf{Y}_2(\tau)$, …, $\mathbf{Y}_k(\tau)$, соответственно подсистемами Y_1, Y_2, …, Y_k.

Выделим в подсистемах Y_1, Y_2, …, Y_k *поименованные состояния*:

$$\mathbf{Y}_1^{S_{11}}(\tau) = \left\{y_{11}^{S_{11}}(\tau), y_{12}^{S_{11}}(\tau), \ldots, y_{1n_1}^{S_{11}}(\tau)\right\},$$
$$\mathbf{Y}_1^{S_{12}}(\tau) = \left\{y_{11}^{S_{12}}(\tau), y_{12}^{S_{12}}(\tau), \ldots, y_{1n_1}^{S_{12}}(\tau)\right\},$$
$$\ldots$$
$$\mathbf{Y}_1^{S_{1q_1}}(\tau) = \left\{y_{11}^{S_{1q_1}}(\tau), y_{12}^{S_{1q_1}}(\tau), \ldots, y_{1n_1}^{S_{1q_1}}(\tau)\right\},$$
$$\mathbf{Y}_2^{S_{21}}(\tau) = \left\{y_{21}^{S_{21}}(\tau), y_{22}^{S_{21}}(\tau), \ldots, y_{2n_2}^{S_{21}}(\tau)\right\},$$
$$\mathbf{Y}_2^{S_{22}}(\tau) = \left\{y_{21}^{S_{22}}(\tau), y_{22}^{S_{22}}(\tau), \ldots, y_{2n_2}^{S_{22}}(\tau)\right\},$$
$$\ldots$$
$$\mathbf{Y}_2^{S_{2q_2}}(\tau) = \left\{y_{21}^{S_{2q_2}}(\tau), y_{22}^{S_{2q_2}}(\tau), \ldots, y_{2n_2}^{S_{2q_2}}(\tau)\right\},$$
$$\ldots$$
$$\mathbf{Y}_k^{S_{k1}}(\tau) = \left\{y_{k1}^{S_{k1}}(\tau), y_{k2}^{S_{k1}}(\tau), \ldots, y_{kn_k}^{S_{k1}}(\tau)\right\},$$
$$\mathbf{Y}_k^{S_{k2}}(\tau) = \left\{y_{k1}^{S_{k2}}(\tau), y_{k2}^{S_{k2}}(\tau), \ldots, y_{kn_k}^{S_{k2}}(\tau)\right\},$$
$$\ldots$$
$$\mathbf{Y}_k^{S_{kq_k}}(\tau) = \left\{y_{k1}^{S_{kq_k}}(\tau), y_{k2}^{S_{kq_k}}(\tau), \ldots, y_{kn_k}^{S_{kq_k}}(\tau)\right\},$$

которые могут (или не могут) быть достигнуты на интервалах времени $\tau_{\min 1} \le \tau \le \tau_{\max 1}$; $\tau_{\min 2} \le \tau \le \tau_{\max 2}$; …; $\tau_{\min k} \le \tau \le \tau_{\max k}$ соответственно.