

**ИСПОЛЬЗОВАНИЕ ЛИНГВИСТИЧЕСКИХ ПЕРЕМЕННЫХ ПРИ ОПИСАНИИ
ОБРАЗЦОВ ОПЕРАЦИЙ В РДО-МОДЕЛЯХ****И. С. Манжай, А. В. Урусов (Москва)**

В настоящее время наблюдается рост интереса к системам имитационного моделирования при решении задач анализа и синтеза сложных систем. Это связано с постоянным увеличением сложности и стоимости систем, используемых в промышленности и бизнесе. Но в существующих системах имитационного моделирования используется «четкая логика», т. е. они используют два классических значения истинности «ИСТИНА» и «ЛОЖЬ» и требуют четкого определения условий. Это затрудняет создание систем, позволяющих описывать модели на естественном языке пользователя, и усложняет работу с промежуточными значениями, подразумевая ответы «ДА» и «НЕТ».

Этих недостатков лишены нечеткие лингвистические системы, основанные на правилах продукционного типа, где в качестве посылок и заключения в правиле используются лингвистические переменные. Это позволяет избежать ограничений, присущих классическим продукционным правилам. В основу функционирования нечетких систем положен механизм нечеткого вывода.

В данной работе рассматривается вариант использования нечеткой логики в интеллектуальной среде имитационного моделирования РДО, разработанной в Московском государственном техническом университете (МГТУ им. Н. Э. Баумана) на кафедре «Компьютерные системы автоматизации производства».

Основы РДО-метода

В основе системы РДО – «Ресурсы, Действия, Операции» – лежат следующие положения:

- Все элементы сложной дискретной системы (СДС) представлены как ресурсы, описываемые некоторыми параметрами.
- Состояние ресурса определяется вектором значений всех его параметров; состояние СДС – значением всех параметров всех ресурсов.
- Процесс, протекающий в СДС, описывается как последовательность целенаправленных действий и нерегулярных событий, изменяющих определенным образом состояния ресурсов; действия ограничены во времени двумя событиями: событиями начала и конца.
- Нерегулярные события описывают изменение состояния СДС, непредсказуемые в рамках продукционной модели системы (влияние внешних по отношению к СДС факторов либо факторов, внутренних по отношению к ресурсам СДС). Моменты наступления нерегулярных событий случайны.
- Действия описываются операциями, которые представляют собой модифицированные продукционные правила, учитывающие временные связи. Операция описывает предусловия, которым должно удовлетворять состояние участвующих в операции ресурсов, и правила изменения ресурсов в начале и конце соответствующего действия.

Модель, написанная на языке РДО, можно разделить на базу данных и базу знаний. В первую входит описание типов ресурсов и самих ресурсов, которые определяют структуру базы данных и ее начальное состояние соответственно. База знаний – это процедурная часть модели, которая состоит из образцов операций, а также операций, описывающих знания о моделируемой системе или предметной области.

Лингвистические переменные в продукционных правилах

Описание параметров ресурсов лингвистическими переменными было подробно рассмотрено в [5]. В этой работе остановимся более подробно на использовании описанных таким образом параметров в описании образцов операций РДО модели.

Сейчас в РДО существует 4 типа образцов: нерегулярное событие (*irregular_event*), продукционное правило (*rule*), операция (*operation*) и клавиатурная операция (*keyboard*). Остановимся подробнее на образцах типа *rule* и *operation*, представляющих продукционное правило и модифицированное продукционное правило соответственно.

В отличие от обычных продукционных правил, имеющих вид

ЕСЛИ (условие) ТО (действие),

и представляющих собой логическую взаимосвязь действий для описания поведения системы без учета времени, модифицированные продукционные правила имеют длительность выполнения. Такие модифицированные продукционные правила имеют следующий вид:

ЕСЛИ (условие) ТО1 (событие 1) ЖДАТЬ (временной интервал) ТО2 (событие 2)

В РДО описание операции имеет следующий формат:

```
$Pattern <имя_образца> : <тип_образца> [<признак_трассировки> ]
[ $Parameters <описание_параметров_образца> ]
$Relevant_resources <описание_релевантных_ресурсов_образца>
[ <способ_выбора> ]
$Time = <выражение_времени>
$Body <тело_образца>
$End
```

Здесь в теле образца описывается выбор релевантных ресурсов, участвующих в операции, и действия над ними. Для этого указывается имя релевантного ресурса и правило его использования, описываемое для продукционного правила в виде:

```
Choice from <логическое_выражение> [<способ_выбора>]
[Convert_rule <конвертор>]
```

Для операции (модифицированного продукционного правила):

```
Choice from <логическое_выражение> [<способ_выбора>]
[Convert_begin <конвертор>]
[Convert_end <конвертор>]
```

В этой записи в блоке *Choice_from* содержится часть продукционного правила ЕСЛИ (условие), а в конверторах *Convert_rule*, *Convert_begin* и *Convert_end* – события, изменяющие состояние релевантного ресурса. Более детально синтаксис описан в [1].

В данной работе значения посылок, заключений, а также времени в модифицированных продукционных правилах предлагается рассматривать как термы лингвистических переменных, описанных для параметров ресурсов модели.

В синтаксисе РДО для операции это будет выглядеть следующим образом (жирным выделены новые ключевые слова, примерный синтаксис):

```
$Pattern Образец_Обработка_детали: operation
$Relevant_resources
    Деталь1: Деталь Keep Keep
$Time = Деталь1.длительность_обработки
$Body
    Деталь1
    Choice from Деталь1.Качество_обработки = Высокое
    and Деталь1.Сложность_обработки = Низкая
```

```

AND: MIN
ACT: MIN
ACCU: MAX
Convert_begin
    Состояние set Обрабатывается
Convert_End
    Состояние set Обработана
    Качество set Рез_кач (Деталь1.Качество_обработки,
Деталь1.Сложность обработки)
$End

```

В данном примере исходный синтаксис РДО расширен параметрами, необходимыми для работы механизма нечеткого вывода, а именно, для описания нечетких операций, вычисления нечеткой импликации и агрегации среднего значения.

Операции описываются следующим образом:

Определение_операции := операция : алгоритм.

При этом следует использовать парные алгоритмы для операторов AND и OR, т. е. если для оператора AND используется алгоритм MIN, то для OR следует использовать MAX.

Методы вычисления нечеткой импликации и агрегации описываются аналогичным образом. Для импликации используется ключевое слово *ACT*, для агрегации *ACCU*.

ACT : метод_импликации

ACCU : метод_агрегации

В случае, если для всех операций используются одни и те же алгоритмы, то их можно описать в объекте прогона – на закладке smg. Формат записи при этом не изменяется.

Нечеткая логика в алгоритмических функциях

В образцах операций описывается изменение параметров ресурсов при выполнении операции. При этом, помимо прямого присвоения значения, как это было приведено в примере предыдущего раздела, возможно использование определяемых пользователем функций и последовательностей, описываемых на закладке FUN. Один из типов этих функций – алгоритмическая – представляет собой не что иное, как набор продукционных правил ЕСЛИ – ТО. Указывая в этой функции методы импликации, агрегации и алгоритмы для операций AND и OR, возможно использование аппарата нечеткой логики для расчета результирующего значения. Приведем пример такой функции в формате, используемом в РДО (жирным выделены новые ключевые слова, примерный синтаксис):

```

$Function Стоимость_поезда : real
$Type = algorithmic
$Parameters
    Расстояние: real
        Интенсивность_движения: real
$Fuzzy_parameters
And: Min
Or: Max
Act: Min
Accu: Max
$Body
    Calculate_if Расстояние=большое OR

```

```

Интенсивность_движения=пробка Стоимость_проезда=высокая
Calculate_if Расстояние=среднее Стоимость_проезда=средняя
Calculate_if Расстояние=малое OR
Интенсивность_движения=дорога_свободна
Стоимость_проезда=малая
$End

```

При срабатывании образца операции в такую функцию передаются «четкие» значения параметров ресурсов, они обрабатываются в соответствии с определенными в модели нечеткими правилами, а полученное значение приводится обратно к «четкому».

Лингвистические переменные в выражении времени

В предыдущем разделе лингвистические переменные использовались для описания логики модифицированного продукционного правила. Но их можно использовать и для определения длительности выполнения операции (operation). Она задается в РДО следующим образом:

```
$Time = <выражение_времени>
```

Здесь *выражение_времени* представляет собой арифметическое выражение целого или вещественного типа[1]. В приведенном выше примере, выражение времени имело вид:

```
$Time = Деталь1.длительность_обработки
```

Деталь1.длительность_обработки – параметр ресурса Деталь1, который определяет длительность обработки. В РДО это допустимая синтаксическая конструкция, если параметр *длительность_обработки* имеет целый или вещественный тип. Если же параметр будет определен как лингвистическая переменная, то на момент его использования в выражении времени он будет иметь конкретное количественное значение. Т.е. для выражения времени подобное использование лингвистической переменной не будет отличаться от использования параметра вещественного типа и будет иметь вид:

```
$Time = <терм>
```

Пример:

```
$Time = малая
```

Здесь *малая* – один из термов лингвистической переменной *Деталь1.длительность_обработки*. Также для выражения времени можно использовать и алгоритмические функции с нечеткой логикой, описанные в разделе.4. Например:

```
$Time = Выбор_длительности( Деталь1.Качество_обработки,
Деталь1.Сложность_обработки )
```

Функция *Выбор_длительности* имеет структуру, аналогичную функции из примера раздела 4.

Выражение времени используется не только для определения длительности операции (operation), но и определяет интервалы между соседними нерегулярными событиями (irregular_event), которые используются для связи модели с внешней средой. В РДО с помощью нерегулярных событий описываются потоки заявок на обслуживание, отказы оборудования и др. Выражение времени, модифицированное лингвистическими переменными, может быть использовано и в этом случае.

Использование лингвистических переменных позволяет приблизиться к описанию имитационных моделей на естественном языке пользователя, а механизм нечеткого вывода может позволить более адекватно описать процессы обслуживания и принимаемые при этом решения, нежели правила классической четкой логики.

Литература

1. **Емельянов В. В., Ясиновский С. И.** Введение в интеллектуальное имитационное моделирование сложных дискретных систем и процессов. Язык РДО. М.: АНВИК, 1998.
2. **Заде Л.** Понятие лингвистической переменной и его применение к принятию приближенных решений: Пер. с англ. М.: Мир, 1976.
3. **Круглов В. В., Дли М. И.** Интеллектуальные информационные системы: компьютерная поддержка систем нечеткой логики и нечеткого вывода. М.: Физматлит, 2002.
4. **Леоненков А. В.** Нечеткое моделирование в среде MATLAB и fuzzyTECH. СПб.: БХВ-Петербург, 2003.
5. **Манжай И. С., Урусов А. В.** Интеграция лингвистических переменных в модели РДО//Четвертая Международная научно-практическая конференция "Интегрированные модели и мягкие вычисления в искусственном интеллекте". Сб. научных трудов, 2007.
6. **Ярушкина Н. Г.** Основы теории нечетких и гибридных систем. М.: Финансы и статистика, 2004.
7. IEC 1131 – Programmable Controllers Part 7 – Fuzzy Control Programming Committee Draft (Rel. 19 Jan 97).