

МОДЕЛИ ОБРАБОТКИ ИНФОРМАЦИИ В ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМАХ, ОСНОВАННЫХ НА СЕМАНТИЧЕСКИХ ТЕХНОЛОГИЯХ

В. П. Ивашенко, А. С. Бельчиков, А. П. Еремеев

Кафедра интеллектуальных информационных технологий, Факультет информационных технологий и управления, Белорусский государственный университет информатики и радиоэлектроники
Кафедра прикладной математики, Филиал федерального государственного бюджетного образовательного учреждения высшего образования «Национальный исследовательский университет «МЭИ»
Минск, Республика Беларусь; Москва, Российская Федерация
E-mail: ivashenko@bsuir.by

В статье рассматриваются модели обработки информации, операции и команды, обеспечивающие программирование решения задач в системах, управляемых знаниями

ВВЕДЕНИЕ

Развитие технологий создания интеллектуальных систем в направлении систем, управляемых знаниями (СУЗ) [1], требует спецификации различных уровней СУЗ [1], что необходимо для решения задач управления СУЗ. При первом рассмотрении выделяются три уровня управления (УУ) в СУЗ: устройствами, данными и знаниями [2]. На каждом из них присутствуют структуры данных, которые необходимо обрабатывать в процессе управления. Этим структурам соответствуют формальные языки (ФЯ), которые рассматриваются в рамках соответствующих моделей [3]. Спецификация процессов обработки структур данных на разных УУ, описывающих состояние объекта управления, строится на основе этих языков и на соответствующих формальных моделях обработки информации (МОИ) [4,5].

МОДЕЛИ ОБРАБОТКИ ИНФОРМАЦИИ

Формальная МОИ задаётся четвёркой $\langle A, S, B, W \rangle$. ФЯ L формальной МОИ задан на алфавите A синтаксисом S ($L \subseteq A^*$). Множество состояний, носитель Q формальной МОИ является множеством текстов языка L ($Q \subseteq L$). Множество начальных состояний B формальной МОИ является подмножеством Q ($B \subseteq Q$). Операции из множества операций W формальной МОИ определены на множестве Q ($W \subseteq 2^{Q \times Q}$). Детерминированные операции формальной МОИ являются элементами множества Q_+^Q , из них отображения являются элементами множества Q^Q . Число недетерминированных операций, являющихся отображениями, не превышает мощности множества $[2^Q \setminus \{\emptyset\}]^Q$. Операции формальной МОИ являются унарными и задают переходы из одного состояния модели в другое, однако распространённые команды в языках программирования имеют, как правило, несколько аргументов того же или аналогичного типа, что и результат этих команд, что соответствует бинарным операциям. Переход от бинарных операций к унарным может быть осуществлён че-

рез «карирование» операций, при этом команды одной бинарной операции соответствуют семейству команд унарных операций. Логические МОИ рассматриваются как базовые, в простейших из них состояния формальной модели состоят из набора битов, которые обрабатываются операциями модели независимо друг от друга и параллельно, и в качестве множества состояний рассматриваются носители, имеющие мощность 2^d . Операции в этих МОИ могут быть заданы соответствующими бинарными командами с кодами операций строгой дизъюнкции, конъюнкции и импликации: xor, and, imp. Структура каждой команды имеет вид: $\langle \text{код операции} \rangle V, A$, где вычисления осуществляются по схеме $A = A \langle \text{операция} \rangle V$. Любая функция одной логической переменной (ЛП) может быть запрограммирована одной из команд, а их любой набор – парой команд (xor, and) или (xor, imp). Для d таких функций длина программы может составлять не более $4+2d$ бит, тогда как минимально возможная – $2d$. Более сложные логические модели позволяют задавать логические функции многих переменных и ставить в зависимость множество значений ЛП от исходного множества значений ЛП. Для их реализации добавляются ещё две команды lor ($A \text{ lor } B = (A < 2^d - 1)?B:2^d - 1$) и land ($A \text{ land } B = (A > 0)?B:0$). Эти команды позволяют вычислить конъюнкцию и дизъюнкцию за четыре шага, используя дополнительно $3d$ бит. Длина программы для любой функции может составлять не более $(15 + 10d) * 2^d - 3 - 2d$ бит, тогда как минимально возможная – $2^d * d$ бит. Арифметические МОИ строятся на основе логических МОИ с операциями, соответствующими командам арифметики вычетов add, sub, mul, div, max и min. При количестве элементов носителя p^d (p – простое ($p=2$)) перечисленные операции позволяют запрограммировать операции полей Галуа порядка $q \leq p^{d-1}$. Строковые МОИ строятся на основе логических и арифметических МОИ, и как правило вместе с увеличением мощности носителя позволяют вводить ассоциации в виде ссылок (указателей) на структуры данных формальной МОИ. Операции строковых

МОИ [6] включают добавление, удаление и поиск элементов строки, конкатенацию и разъединение строк [7]. Проверка работоспособности операций рассматриваемых МОИ в соответствии со спецификацией обеспечивается виртуальной машиной (<https://bitbucket.org/version/openjsvnm/>), реализованной средствами javascript, так тестирование арифметических операций проведено на задаче моделирования одномерного обратимого клеточного автомата. Теоретико-множественные МОИ строятся на основе строковых МОИ и задают операции пересечения, объединения, дополнения (разности) [2]. Онтологические МОИ строятся на основе строковых МОИ и ассоциативных языков [3] для представления понятий (концептов), среди которых различаются абсолютные и относительные. Каждому концепту сопоставляется некоторое имя (термин, знак) и его означаемое (десигнат) и значение (денотат). Термины рассматриваются как «внешние» знаки, которые поступают извне к понимающему субъекту, а сами понятия (концепты) полагаются выполняющими функцию знака внутри мышления субъекта («внутренние знаки»). В четвёрках имя (знак) n будем относить к именам нарицательным, если и только если все четвёрки во множестве $\{n\} \times \{c\} \times \{s\} \times \{d_1, d_2\}$, состоят из (внешнего) знака, его концепта, десигната и денотата, иначе n будем относить к именам собственным. Понятие c будем относить к абсолютным понятиям, если любая четвёрка, состоящая из знака (имени нарицательного), его концепта c , десигната и денотата, состоит во множестве вида $N \times \{c\} \times \{D\} \times D$, в ином случае c будем относить к относительным понятиям. В онтологических МОИ одной из задач обработки онтологий является задача формирования структуры онтологии, построения концептуальной решётки. Для этого применимы методы анализа формальных понятий [8], применяющиеся для понятий, которые рассматриваются как формальные понятия (ФП), имеющие объём и содержание. При формировании решётки относительных понятий в случае бинарных отношений и отношений, арность которых больше двух, ФП выражаются в n -мерном формальном контексте как прямые произведения $A_1 \times A_2 \times A_3 \dots \times A_n$, где выбор доменов $A_1, A_2, \dots, A_{i-1}, A_{i+1}, \dots, A_n$ является полупроизвольным, а A_i рассматривается как содержание (множество атрибутов). В этом случае требуется, чтобы сохранялось $n^*(n-1)$ условий для оператора замыкания, для каждой пары доменов. Построение ФП большей арности возможно на основе уже построенных ФП. Важной является возможность разложения онтологии в базис (БО), минимальный набор понятий, через которые с помощью соответствующих операций можно выразить её любое понятие, т.е. найти такое «сжатое» состояние онтологической модели, из которого можно достичь состояний с частично или полностью представленной онтологией. При

построении БО одновременно для понятий различной арности не следует упускать из виду зависимости (проекции) между ними. Построение БО можно свести к итеративной процедуре. Если в онтологии есть (непустое) понятие (множество), то оно выбирается в качестве БО. Для расширения имеющегося БО применяется одно из не менее чем десяти правил, которые реализуются операцией (функцией) на носителе $B \times M \times E \times X$, где B – множество разбиений, задающих компоненты БО, в соответствии с их взаимными пересечениями, M – новый, формируемый компонент БО, в виде разбиения на пересечения с компонентами сформированного БО, E – множество просмотренных элементов разбиений из B , X – множество множеств подлежащих рассмотрению и используемых в качестве основы формирования БО. Процедура повторяется, пока вся онтология, все понятия, не окажутся рассмотренными, в новом БО. Также возможно применение эвристических правил, балансирующих или перебалансирующих мощность формируемых элементов БО, учитывая мощность исходных множеств.

ЗАКЛЮЧЕНИЕ

Ключевым принципом построения МОИ для СУЗ является наличие единой модели соответствующих ФЯ и базовых операций, обеспечивающих достижимость за требуемое количество шагов любого целевого состояния из базового множества состояний.

1. Ивашенко, В. П. Алгоритмы параллельной реализации анализа формальных понятий для приближённых множеств в однородных семантических сетях / В. П. Ивашенко, С. В. Синцов // BIG DATA and Advanced Analytics.: сб. материалов II Междунар. науч.-практ. конф. – Минск : БГУИР, 2016. С. 110–118.
2. Ивашенко, В. П. Принципы платформенной независимости и платформенной реализации OSTIS / В. П. Ивашенко, М. М. Татур // OSTIS. 2016. С. 145–150.
3. Ивашенко, В. П. От теоретико-множественных моделей к симплицимальным моделям языков / В. П. Ивашенко // Карповские научные чтения: сб. науч. ст. – Минск, 2016. – Вып. 10, ч. 1. – С. 248–253.
4. Голенков, В. В. Представление и обработка знаний в графодинамических ассоциативных машинах / В. В. Голенков [и др.] – Минск : БГУИР, 2001.
5. Кузьмицкий, В. М. Принципы построения графодинамического параллельного ассоциативного компьютера, ориентированного на переработку сложноструктурированных знаний / В. М. Кузьмицкий // Интеллектуальные системы : сб. науч. тр. / НАН Беларуси, Ин-т техн. кибернетики ; науч. ред. А. М. Крот. – Минск, 1998. – Вып. 1. – С. 156–166.
6. Смит, Б. Методы и алгоритмы вычислений на строках. : Пер. с англ. – М.: ООО «И.Д. Вильямс», 2006. – 496 с.
7. Bates, R. Sequence-Trees: Slicing and Concatenation of Sequences in Logarithmic Time / R. Bates // 14th Midwest Conference on Combinatorics, Cryptography, and Comp. Wichita: WSU, – 2002. – Vol.42. – P. 33–60.
8. Poelmans, J. Fuzzy and rough formal concept analysis: a survey J. Poelmans, D. Ignatov, S. Kuznetsov, et al. // INT J GEN SYST, 2014. – Vol. 43, Issue 2, P. 105–134.