Распознавание в интеллектуальных системах функционального типа

А.Б. Фролов, Д.А.Фролов, И.Д. Четрафилов

Рассматриваются алгоритмы распознавания с учетом отношений частичного и линейного порядка на множестве объектов. Изучаются принципы построения интеллектуальных систем (ИС) функционального типа с использованием таких алгоритмов и алгоритмов исчисления граней (кубов). Описываются математические модели ИС и алгоритмы исследования непротиворечивости и полноты их исходного описания.

1 Введение

Принятие решений сопровождает этапы деятельности специалиста в различных областях — медицине, управлении движением, геологической разведке, управлении сложными объектами, маркетинге. Оно заключается в выборе определенной альтернативы на основе анализа объекта или ситуации.

Накопление и сохранение опыта принятия решений экспертами позволяет создавать информационные среды, доступные для использования более широким кругом специалистов. Разрабатываемые для этих целей информационные и программные средства называют системами поддержки принятия решений. Они являются разновидностью интеллектуальных систем (ИС).

Изучаемые в настоящей работе ИС функционального типа реализуют информационные среды в виде отображения множества описаний объектов или ситуаций на множество возможных решений.

В силу комбинаторной сложности не представляется возможным иметь в исходном описании отображения указания о решении (классе) для каждого объекта или ситуации. Исходные описания составляются экспертами в обобщенном виде, когда для некоторой достаточно представительной совокупности объектов или ситуаций указывается определенное решение. При этом может оказаться, что один и тот же объект входит в две или более совокупности, которым поставлены в соответствие разные решения. Такое исходное описание является противоречивым и непригодно для использования в ИС. Поэтому одной из задач проектирования ИС функционального типа является тестирование исходного описания на непротиворечивость, или функциональность. Может также оказаться, что некоторые объекты или ситуации не учтены в исходном описании, что также желательно установить до начала проектирования ИС тестированием его на полноту. В настоящей работе рассмотренные неформальные постановки задач исследования ИС функционального типа уточняются с учетом реально существующего или искусственно определяемого упорядочения объектов или ситуаций.

При решении этих задач в работе используются алгоритмы теории распознавания упорядоченных объектов [1, 2], относящейся к комбинаторно-логическому направлению в распознавании образов [3], а также операции и алгоритмы исчисления граней [4, 5], являющегося обобщением исчисления кубических комплексов Рота и Карпа [6]. Использованные в настоящей работе понятия непротиворечивости и полноты соответствуют понятиям семантической непротиворечивости и семантической полноты описанным в работе [7] применительно к таблицам решений, которые также относятся к моделям функционального типа.

2 Формализация задачи

Будем рассматривать множества объектов или ситуаций, описываемых n-мерными векторами $(c_1, c_2, ..., c_n)$, компоненты c_i которых соответствуют тем или иным признакам объекта или ситуации и принимают значения из конечных упорядоченных множеств возможных значений C_i, $i = 1, ..., n$. Будем полагать, что эти множества являются решетками [8], тогда и их декартово произведение обладает свойствами решетки. Некоторым векторам

$$(c_1, c_2, ..., c_n) \in C_1 \times C_2 \times \ldots \times C_n$$

приписываются значения $r_1, r_2, ..., r_k$, соответствующие тем или иным действиям или наборам (спискам) действий ИС. Тем самым определяется некоторая вообще говоря частичная функция $f : C_1 \times C_2 \times \ldots \times C_n \to R$, где $R = \{r_1, r_2, ..., r_k\}$, — решающая функция ИС.

Кроме того может рассматриваться полностью определенная характеристическая функция $\chi : C_1 \times C_2 \times \ldots \times C_n \to \{0, 1\}$, имеющая значение 1 на наборах, где ИС считается определенной, и значение 0 на остальных наборах (в случае $\chi(c_1, c_2, \ldots, c_n) = 0$ ИС считается неопределенной на наборе $(c_1, c_2, \ldots, c_n)$, даже если решающая функция f принимает определенное значение $f(c_1, c_2, \ldots, c_n)$ на нем).

По определению ИС обладает свойством *функциональности* и является *непротиворечивой*, поскольку для всех i, j

$$r_i \neq r_j \to (c_{i_1}, c_{i_2}, \ldots, c_{i_k}, \ldots, c_{i_n}) \neq (c_{j_1}, c_{j_2}, \ldots, c_{j_k}, \ldots, c_{j_n}), \tag{1}$$

где $(c_{i_1}, c_{i_2}, \ldots, c_{i_k}, ..., c_{i_n})$ и $(c_{j_1}, c_{j_2}, \ldots, c_{j_k}, \ldots, c_{j_n})$ — векторы ситуаций, соответствующие решениям $r_i = f(c_{i_1}, c_{i_2}, \ldots, c_{i_k}, \ldots, c_{in})$ и $r_j = f(c_{j_1}, c_{j_2}, \ldots, c_{j_k}, \ldots, c_{j_n})$.

ИС называется *полной*, если

$$(c_1, c_2, \ldots, c_n) \notin Im^{-1}f \to \chi(c_1, c_2, \ldots, c_n) = 0 \tag{1}$$

.

Практическая реализация ИС в виде таблиц, содержащих все пары

$$((c_1, c_2, \ldots, c_n), r) \tag{2}$$

невозможна ввиду того, что такие таблицы должны содержать $\mid C_1 \times C_2 \times \ldots \times C_n \mid$ подобных записей.

В то же время "близким"наборам значений признаков часто соответствуют одинаковые значения. Это позволяет использовать обобщенное представление подмножеств рассматриваемых пар в виде одной пары

$$((\sigma_1, \sigma_2, \ldots, \sigma_n), r), \tag{3}$$

где $\sigma_i \subseteq C_i$ — подмножества множеств $C_i, i = 1, \ldots, n$.

Включаемые в такие описания наборы $(\sigma_1, \sigma_2, \ldots, \sigma_n)$ называются *гранями* декартова произведения $C_1 \times C_2 \times \ldots \times C_n$ — его подмножествами, являющимися декартовыми произведениями:

$$(\sigma_1, \sigma_2, \ldots, \sigma_n) = \sigma_1 \times \sigma_2 \times \ldots \times \sigma_n,$$

где $\sigma_i \subseteq C_i, i = 1, \ldots, n$.

Если использовать только такие грани, где множества σ_i, являются подрешетками решеток C_i, то каждое такое непустое множество как конечная решетка имеет как верхнюю $Sup(\sigma_i)$ так и нижнюю $Inf(\sigma_i)$ универсальные границы, и элементы σ_i векторов σ можно представлять в виде пар $[c_i^1, c_i^2]$ элементов множества C_i, $c_i^1 = Inf(\sigma_i)$, $c_i^2 = Sup(\sigma_i)$, что и дает эффект упрощения описания модели. В этом случае наборы $(c_1^2, \ldots, c_i^2, \ldots, c_n^2)$ и $(c_1^1, \ldots, c_1^1, \ldots, c_n^1)$ являются верхней $Sup(\sigma)$ и нижней $Inf(\sigma)$ универсальными границами грани σ

Всякое непустое подмножество $M \subseteq C_1 \times C_2 \times \ldots \times C_n$ можно представить как объединение некоторых его граней, составляющих в этом случае *покрытие* этого множества.

Описание функции в виде множества пар (3) формируется экспертом при ее первичном задании и возможно преобразуется с помощью алгоритмов минимизации.

Например, если известное действие   должно выполняться в случае конкретных значений $c_1, c_2, \ldots, c_n$ признаков $1, 2, \ldots, n$ независимо от значений остальных признаков, то это правило принятия решений будет описано как пара

$$((\sigma_1, \sigma_2, \ldots, \sigma_s, x, \ldots, x), r),$$

где x обозначает полные множества $C_i, i = k + 1, \ldots, n, r \in R$.

Аналогично формируется обобщенное описание характеристической функции χ как совокупность пар вида

$$((\sigma_1, \sigma_2, \ldots, \sigma_n), 1). \tag{4}$$

Таким образом, математическая модель ИС рассматриваемого вида образуется обобщенными описаниями решающей функции в виде множества

$$F = \{((\sigma_{i_1}, \sigma_{i_2}, \ldots, x, \ldots, x), r_i), i = 0, 1, \ldots, q\} \tag{5}$$

пар вида (3) и характеристической функции в виде множества

$$X = \{(\sigma_{i_1}, \sigma_{i_2}, \ldots \sigma_{i_n}, i = 1, \ldots, t\} \tag{6}$$

граней из пар вида (4), образующих некоторое покрытие области определения ИС (то есть области единичных значений характеристической функции χ).

Пример. На рисунке представлены 4-значные функции $f(x_1, x_2)$ (а), и $\chi(x_1, x_2)$ (б), описывающие модель ИС. Обведены грани, составляющие покрытия областей определенных значений 1,2 и 3 решающей функции (значение 0 соответствует наборам из области неопределенности этой функции) и области определения ИС.

Соответствующая модель функционального типа образуется парой (F, X), где

$$F = (([(0,0),(0,0)],(3)),([(1,0),(3,0)],(2)),([(0,2),(0,3)],(1)),$$

$$([(1,1),(1,2)],(3)),([(1,1),(2,1)],(3)),([(3,3),(3,3)],(1))), \tag{7}$$

$$X = ([(0,0),(2,2)],[(1,1),(3,3)]) \tag{8}$$

и все области представлены кратчайшими покрытиями.

Модель ИС называется *непротиворечивой*, если для всех i, j

$$(r_i \neq r_j) \rightarrow$$

$$\to (\sigma_{i_1}, \sigma_{i_2}, \ldots, \sigma_{i_n}) \bigcap (\sigma_{j_1}, \sigma_{j_2}, \ldots, \sigma_{j_n}) \bigcap (\bigcup_{\sigma \in X} \sigma) = \emptyset,$$

где

$$(\sigma_{i_1}, \sigma_{i_2}, \ldots, \sigma i_n), (\sigma_{j_1}, \sigma_{j_2}, \ldots, \sigma_{j_n})$$

— грани, образованные наборами признаков, которым соответствуют решения r_i и r_j.

Модель ИС называется *полной*, если

$$\bigcup_{(\sigma, r) \in F} \sigma \supseteq \bigcup_{\sigma \in X} \sigma,$$

что согласуется с критерием (1) полноты ИС.

Пусть имеется полная непротиворечивая модель ИС. Тогда поиск решения для данного набора $(c_1, c_2, ..., c_n)$ значений признаков будет состоять в нахождении любой пары

$$((\sigma_{i_1}, \sigma_{i_2}, \ldots, \sigma_{i_n}), r_i) \in F,$$

такой, что

$$(c_1, c_2, \ldots, c_n) \in (\sigma_{i_1}, \sigma_{i_2}, \ldots, \sigma_{i_n}),$$

и грани

$$(\sigma_{j_1}, \sigma_{j_2}, \ldots, \sigma_{j_n}) \in X,$$

такой, что

$$(c_1, c_2, \ldots, c_n) \in (\sigma_{j_1}, \sigma_{j_2}, \ldots, \sigma_{j_n}),$$

(последнее — для подтверждения, что рассматриваемый набор значений признаков принадлежит области определения ИС). При положительном результате поиска и проверки система принятия решений рекомендует набор действий r_j.

ИС, представляемые математическими моделями рассмотренного вида, будем называть системами функционального типа и представлять их парами вида (F, X), т. е. описаниями (5,6).

Таким образом, целью настоящей работы является исследование алгоритмов проверки непротиворечивости и полноты математических моделей ИС функционального типа, а также методов преобразования и аналитического представления таких моделей для последующей программной или схемной реализации решающих функций и указанных алгоритмов.

3 Алгоритмы принятия решений

Механизм принятия решений может быть реализован в виде алгоритмов распознавания упорядоченных объектов [1, 2, 5]. Рассмотрим постановку задачи и общую схему алгоритмов распознавания, использующих упорядочение.

Пусть множество $C = C_1 \times C_2 \ldots \times C_n$ конечно и на нем задано отношение $\leq$ частичного порядка [8]. Если модель ИС полна, то для описания решающей функции f в ряде случаев достаточна информация о ее значениях в некоторых характерных точках декартова произведения C.

Таким образом задачу распознавания можно уточнить как задачу определения значения функции $f : C \to R$ на любом элементе из области ее определения по информации о ее значениях на элементах некоторого подмножества этой области наименьшей мощности.

В работах [1, 2] рассмотрен ряд алгоритмов получения этой минимальной информации о классах $K_0, ..., K_{m-1}$ ядерной эквивалентности функции f (алгоритмы разделения) и распознавания принадлежности определенному классу произвольного элемента $x, x \in C$.

Алгоритмы разделения и распознавания соответствуют следующей общей схеме в предположении, что функция f (возможно частичная) задана списком $L(f)$ пар $(a, f(a))$.

Алгоритм $\mathcal{A}$ разделения включает два этапа:

1) Список $L(f)$ линейно упорядочивается согласованно с отношением частичного порядка на множестве A элементов. При этом пары $(a, v), v = f(a)$ располагаются в соответствии с некоторым линейным упорядочением "$\prec$" элементов множества K таким, что из $a^1 \leq a^2$ всегда следует $a^1 \prec a^2$), и объявляется пустой список $L_{\mathcal{A}} = [\ ]$.

2) для первого и каждого последующего элемента (a, v) списка L определяется значение $\tilde{v} = \psi_{\mathcal{A}}(L_{\mathcal{A}}, a)$, где $\psi_{\mathcal{A}}$ — функция, определенная для конкретного алгоритма разделения $\mathcal{A}$ и принимающая значения из области значений функции f. Если $\tilde{v} = v$, список $L_{\mathcal{A}}$ не изменяется, иначе он пополняется с сохранением линейной упорядоченности парой $(a, \varphi_{\mathcal{A}}(L_{\mathcal{A}}, a, v))$, где $\varphi_{\mathcal{A}}$ некоторая функция, также характерная для данного алгоритма разделения $\mathcal{A}$.

Функции $\psi_{\mathcal{A}}$ и $\varphi_{\mathcal{A}}$ таковы, что после изменения списка $L_{\mathcal{A}}$ выполняется равенство $\psi_{\mathcal{A}}(L_{\mathcal{A}}, a) = v$, которое для данного элемента a сохраняется и при последующих изменениях списка, то есть $\psi_{\mathcal{A}}(L_{\mathcal{A}}, x) = f(x)$, если $x \leq a$. Последнее обеспечивается тем, что при вычислении значения $v' = \psi_{\mathcal{A}}(L_{\mathcal{A}}, a)$ используются только те элементы $(x, v'(x))$ списка $L_{\mathcal{A}}$, в которых $x \leq a$ и, следовательно, $x \prec a$. Это свойство алгоритма разделения позволяет использовать список $L_{\mathcal{A}}$, полученный по окончании обработки исходного списка $L(f)$, для определения значения $f(a)$ функции f для любого элемента a ее области определения.

Поэтому алгоритм распознавания можно описать, используя функцию $\psi_{\mathcal{A}} : f(x) = \psi_{\mathcal{A}}(L_{\mathcal{A}}, x)$.

Пример. Рассмотрим простейший алгоритм S разделения, соответствующий этой общей схеме. Исходное описание графика функции f линейно упорядочивается в виде списка $L(f)$ пар $(a, f(a))$ и объявляется пустой список L_S. Для первой и последующих пар (a, v) исходного списка $L(f)$ в списке L_S выделяется последний элемент $(a', \tilde{v})$, такой, что $a' \leq a$, если такого элемента нет, то принимается $\tilde{v} = 0$. Если $\tilde{v} = v$, то список L_S не изменяется, иначе в конец списка включается текущая пара исходного списка (a, v). В данном случае $\psi_S(L_S, a) = \tilde{v}$, $\varphi_S(L_S, a, v) = v$.

Алгоритм R_S распознавания использует в качестве решающего правила список L_S. Номер класса элемента определяется как значение v' из последней пары (a', v') списка L_S такой, что $a' \leq a$. В данном случае $\psi_S(L_S, a) = \tilde{v}, \varphi_S(L_S, a, v) = v$. Например , пусть упорядоченный список L, соответствующий некоторой функции $f(x_1, x_2) : \{0, 1, 2, 3\}^2 \to \{0, 1, 2, 3\}$ имеет вид

$$L(f) = [((0,0),3),((0,1),0),((1,0),2),((0,2),1),((1,1),3),((2,0),2),((0,3),1),((1,2),3),$$
$$((2,1),3),((3,0),2),((1,3),0),((2,2),0),((3,1),0),((2,3),0),((3,2),0),((3,3),1)]. \quad (9)$$

Для этой функции по списку (9) по алгоритму S получится список

$$L_S = [((0,0),3),((0,1),0),((1,0),2),((0,2),1),((1,1),3),$$
$$((1,3),0),((2,2),0),((3,1),0),((3,3),1)]. \quad (10)$$

Алгоритм R_S распознавания использует в качестве решающего правила список L_S. Номер класса элементаа определяется как значение v' из последней пары (a', v') списка L_S такой, что $a' \leq a$.

Например, по списку (10)

$$L_S = [((0,0),3),((0,1),0),((1,0),2),((0,2)1),((1,1),3),$$
$$((1,3),0),((2,2),0),((3,1),0),((3,3),1)]$$

значение функции $f(x_1, x_2)$, представленной списком (9), на наборе $(3,2) : f(3,2) = f(3,1) = 0$.

В работах [1, 2] показано, что решающие правила алгоритмов рассматриваемого здесь класса могут быть представлены аналитическими выражениями k-значной логики, которые строятся по определенным правилам по спискам $L_{\mathcal{A}}$. Кроме того допускается их реализация в виде схем из функциональных элементов и схем на языке функционального программирования.

Отметим также следующую особенность алгоритмов разделения, являющихся аналогами алгоритмов обучения классической теории распознавания.

Для формирования решающего правила в виде списка $L_{\mathcal{A}}$ требуется лишь однократный просмотр исходного предварительно упорядоченного списка L.

Решающее правило может быть получено и при обработке элементов $(a, f(a))$ обучающей последовательности, поступающих хаотически, но многократно (при таком обучении "с забыванием"осуществляются такие же изменениями списка $L_{\mathcal{A}}$ при обработке текущего элемента обучающей последовательности, но с дополнительным действием предварительного удаления из него пар (a', v), таких, что $a \leq a'$).

При этом число изменений решающего правила можно оценить сверху (имеется аналог теоремы Новикова [9] о существовании верхней границы числа изменений решающего правила при обучении перцептрона), что однако (как и в метрической теории) не приводит к критерию окончания процесса обучения. Таким образом для уверенного завершения процесса обучения необходимо иметь все элементы обучающей выборки. Отметим, что существование критерия окончания процесса обучения, например, по методу минимизации средней квадратичной ошибки в метрической теории распознавания также обусловлено использованием при каждой итерации всех объектов обучающей выборки [10].

Алгоритмы данного класса наряду с метрическими алгоритмами распознавания находят применение при распознавании оптических образов текстов [11, 12].

4 Некоторые операции и отношения исчисления граней

Для реализации алгоритмов распознавания при обобщенном исходном описании функций f и χ (см. раздел 2) будем использовать ряд операций на множестве граней, некоторые из которых имеют аналоги в исчислении бинарных кубических комплексов [6].

Операция *пересечения* граней образуется как набор пересечений элементов описывающих их векторов.

Операция *сопряжения* граней по i-му элементу образуется как набор элементов, в котором i-ый элемент есть объединение i-ых элементов исходных векторов, а остальные элементы являются пересечениями соответствующих элементов исходных векторов.

Эти операции легко описать в терминах универсальных границ граней и компонентов соответствующих векторов (см. раздел 2).

Для каждой грани σ можно определить верхний конус ее нижней универсальной границы $U_c(Inf(\sigma))$ как элемента исходного декартова произведения. Будут использоваться операции построения *дополнения* грани в таком верхнем конусе, которая определяет для грани σ совокупность $C(\sigma)$ граней, покрывающих множество его элементов, не принадлежащих грани. Для построения этой совокупности грани образуется множество $\mathcal{I}(\sigma)$ минимальных элементов множества $Uc(Inf\sigma) \setminus \sigma$.

Покрытие $C(\sigma)$ множества $U_c(Inf\sigma) \setminus \sigma$ можно описать как множество граней

$$C(\sigma) = \{[c, Sup(U_c(Inf(\sigma)))]/c \in \mathcal{I}(\sigma)\}$$

Будем использовать также операции ограничения грани σ_2 гранью σ_1 :

$$O(\sigma^1, \sigma^2) = [sup(Inf(\sigma^2), Sup(\sigma^1)), Sup(\sigma^2)],$$

где sup— операция супремума на исходном декартовом произведении как решетке.

Опишем некоторые отношения на множестве граней. Отношение частичного порядка (*поглощения*) является отношением $\supseteq$ включения множеств. Отношение линейного порядка $\prec$ на множестве граней определим, используя отношения $\prec$ линейного порядка на исходном декартовом произведении, следующим образом: $\sigma^1 \prec \sigma^2$, если $Inf(\sigma^1) \prec Inf(\sigma^2)$ или $Inf(\sigma^1) = Inf(\sigma^1)$, а $Sup(\sigma^2) \prec Sup(\sigma^1)$. Рассмотренные операции и отношения позволяют описать алгоритмы минимизации покрытий подмножеств декартовых произведений гранями и модифицировать алгоритмы распознавания.

Например, аналог алгоритма Блэка-Порецкого для построения сокращенного покрытия по имеющемуся произвольному покрытию можно описать следующим образом:

1. Принять исходное покрытие в качестве текущего покрытия.

2.Для первой и последующих граней текущего покрытия

а) если грань поглощается хотя бы одной из последующих граней списка, исключить ее из списка;

б) пока не будет получена грань, не поглощаемая другими гранями списка, и пока имеется очередная грань, выполнять сопряжение с нею данной грани по всем компонентам. Полученную грань поместить в начало списка текущего покрытия и повторить исполнение п.2 алгоритма

3. Конец.

5 Модифицированные алгоритмы распознавания

При построении решающих функций по базовому алгоритму S, рассмотренному выше в п.2, используется развернутое описание обучающей выборки, в котором представлена каждая пара вида (2) непосредственно.

Рассмотрим модификацию алгоритма, предполагая, что обучающая выборка задается функциональной моделью (F, X) (см. представления (5,6)), то есть покрытиями классов ядерной эквивалентности решающей функции и покрытием области определения ИС функционального типа.

Модифицированный алгоритм S_M описывается следующим образом.

1.Пары вида (σ, v), где $v = f(Inf(\sigma))$ непротиворечивого обобщенного описания решающей функции размещаются в списке L, который линейно ($\prec$) упорядочивается по элементам σ из образующих его пар, и образуется пустой список $L_{\mathcal{A}}$.

Примечание. В список L могут также включаться пары вида (σ, x), где x является символом неизвестности, означающим необходимость уточнения значения $v = f(Inf(\sigma))$.

2.Для первого и каждого последующего элемента (σ, v) списка L по правилам базового алгоритма S вычисляется значение $\tilde{v} = \psi(L_{\mathcal{A}}), Inf(\sigma))$, где $\psi_{\mathcal{A}}$ — определенная для алгоритма A функция, и выполняется одно из следующих действий:

а) если $v = x$, то в список L включаются пары вида $([a, a], x)$, соответствующие тем элементам из множества $\mathcal{I}(\sigma)$, для которых найдется грань σ', представленная в некотором элементе списка L, такая, что $a \in \sigma'$, но $a \neq Inf(\sigma')$ и пары вида $([a, a], 0)$, если для всех $\sigma' \quad a \notin \sigma'$.

б) если $v \neq \tilde{v}$ и $v \neq 0$, то список L изменяется как описано в п. а) и в конец списка $L_{\mathcal{A}}$ включается пара $(Inf(\sigma), \varphi_{\mathcal{A}}(L_{\mathcal{A}}, Inf(\sigma), v)$, здесь $\varphi_{\mathcal{A}}$ — определенная для алгоритма $\mathcal{A}$ функция.

в) если $v \neq \tilde{v}$ и $v = 0$, то список $L_{\mathcal{A}}$ изменяется как описано в п. б) и в список L включаются пары $(O(a, \sigma'), v')$, соответствующие тем элементам a из множества $\mathcal{I}(\sigma)$, для которых найдется грань σ', представленная в некотором элементе (σ', v') исходного списка L, такая, что $a \leq Sup(\sigma')$, но не выполняется условие $(a \leq Inf(\sigma'))\&(a \neq Inf(\sigma'))$.

После этого текущий элемент удаляется из списка L и список упорядочивается, при этом из двух или нескольких одинаковых его элементов оставляется только единственный экземпляр.

Пример. По приведенным в п.2 описаниям (7,8) покрытий областей ненулевых значений решающей функции f на Рис. 1,а и характеристической функции χ на Рис.б можно получить следующие списки

$$L_S(f) = [((0,0),(3)),((0,1),(0)),((1,0),(2)),((0,2),(1)),$$

$$((1,1),(3)),(1,3),(0)),((2,2),(0)),((3,1),(0)),((3,3),(1))]; \quad (11)$$

$$L_S(\chi) = [(((0,0),(1)),((0,3),(0)),((3,0),(0)),((1,3),(1)),((3,1),(1)))]. \quad (12)$$

6 Анализ непротиворечивости и полноты моделей ИС

Пусть дано описание (F,X) модели ИС функционального типа, причем элементы множеств F и X представлены списками L_F и L_X .

Тогда непротиворечивость модели можно проверить по следующему алгоритму сложности, работающему с этими списками:

1. Образуется список L_cur пар $(\sigma^1 \bigcap \sigma^2, \{v_1, v_2\})$, соответствующих парам $(\sigma^1, v_1), (\sigma^2, v_2)$ списка L_F , где $v_1 \neq v_2$.

2. Образуется список L_cntr пар $(\sigma \bigcap \sigma', \{v_1, v_2\})$, соответствующих элементам $(\sigma, \{v_1, v_2\})$ списка L_cur и элементам σ' списка L_X , таким, что $\sigma \bigcap \sigma' \neq \emptyset$.

Примеры.

1.Для модели, заданной списками

$$L_F = (((0,0),(2,3),(1)),((1,2),(3,2),(2)),((2,2),(3,3),(3))),$$

$$L_X = (((2,0),(3,2)),((3,2),(3,3))),$$

с помощью указанного алгоритма получим описание (возможно содержащее повторения) области неоднозначности:

$$L_cntr = (((3,2),(3,3),(3,2)),((2,2),(2,2),(1,2)),((2,2),(2,2),(1,3)),((2,2),(3,2),(2,3))).$$

Здесь четыре списка содержат грани (первые две записи в каждом списке) и два значения, встречающиеся на наборах этих граней (третья запись).

2.Для модели, заданной списками

$$L_F = (((0,0),(0,0),(3)),((1,0),(3,0),(2)),((0,2),(0,3),(1)),$$

$$((1,1),(1,2),(3)),((1,1),(2,1),(3)),((3,3),(3,3),(1))),$$

$$L_X = (((0,0),(3,3)))$$

по этому алгоритму будет получен пустой список, описывающий пустую область неоднозначности

$$L_{cntr} = (),$$

что гарантирует непротиворечивость модели.

Проверка полноты модели ИС функционального типа (F,X) может быть осуществлена по следующему алгоритму:

1. Применяя алгоритм S_M к исходным описаниям в виде списков L_F и L_X построить сжатые представления $L_S(F)$ и $L_S(X)$ решающей и характеристической функций ИС соответственно.

2. Если список $L_S(F)$ не содержит пары $((0,...,0),v)$, то в него в качестве первого элемента включается запись $((0,...,0),x)$.

3. Образовать пустой список L_{unc}, в котором будет формироваться итоговый список.

4. Пока список $L_S(F)$ не пустой для его первого элемента (a,v) выполнить следующие действия:

если $v \neq 0$, исключить элемент (a,v) из списка;

иначе образовать пустой список L_{cur}, в котором будет формироваться очередной список для итогового списка.

вычислить значение $u = \psi_S(L_S(X), a)$, указывающее, принадлежит ли набор a области определения ИС и включить пару (a, v) в список L_{cur};

образовать множество $\mathcal{M}(L_S(F), a)$ минимальных по отношению частичного порядка наборов a'' из пар (a', v) списка $L_S(F)$, таких что $a \leq a'$;

для каждого элемента a'' множества $\mathcal{M}(L_S(F), a)$ включить в список L_{cur} все пары (a''', u) списка $L_S(X)$, такие, что $a \leq a''' \leq a'$ и затем пару (a', v) списка $L_S(F)$;

включить список L_{cur} в список $L_{|bfunc}$;

исключить первый элемент (a, v) из списка $L_S(F)$.

Полученный список содержит списки, каждый из которых описывает один или несколько "отрезков", состоящих из одного или нескольких линейно-упорядоченных наборов, принадлежащих области определения ИС, заданной списком $F_S(X)$, но для которых покрытием в списке $F_S(F)$ не определены решения. Эти "отрезки"можно выделить по значениям u в парах (a''', u) (они "открываются"парой, где $u = 1$ и "закрываются"парой, где $u = 0$, или последней парой (a', v) списка, причем эта "закрывающая"пара к "отрезку"не относится).

Пример. Применением алгоритма к модели (F, X) ИС на Рис. а,б, заданной описаниями (6,7), сначала получаются сокращенные описания (11,12), после чего строится список

$$L_{unc} =$$

$$((((0,1),(1)),(0,2),(1))),(((0,1),(1)),((1,1),(3))),(((1,3),(1)),$$

$$((3,3),(1))),(((2,2),(1)),((3,3),(1))),(((3,1),(1)),((3,3),(1)))).$$

Он описывает область, где модель, заданная двумя функциями на рисунке неопределена: содержащиеся в этой записи пять списков определяют (возможно с повторениями) следующие выявленные области неопределенности:

$$((0,1),(0,1)),((1,0),(1,0)),((1,3),(2,3)),((2,2),(2,3),(3,2)),((3,1),(3,2)).$$

Данный алгоритм позволяет сократить перебор при проверке полноты, исключая анализ наборов, на которых функция f, заданная множеством пар F, заведомо определена или которые не принадлежат области определения, заданной множеством X. Однако его полиномиальность означала бы полиномиальность задачи ВЫПОЛНИМОСТЬ [13], что противоречило бы современным представлениям о сложности последней. В то же время для некоторых частных видов моделей алгоритм строит список $L_S(F)$ за полиномиальное время, например, в случае, когда покрытие области единичных значений булевой функции состоит из граней вида $(a_1, a_2, x, \ldots, x)$, что соответствует задаче 2-ВЫПОЛНИМОСТЬ, имеющей полиномиальный алгоритм решения.

Список литературы

[1] Фролов А.Б., Яко Э. Алгоритмы распознавания частично-упорядонных объектов и их применение. Известия АН СССР, сер. Техническая кибернетика, 1990, N 5.

[2] Фролов А.В., Фролов Д.А., Яко Э. Функциональные схемы для распознавания упорядоченных объектов. Изв. РАН, сер. Теория и системы управления, 1997, N5.

[3] Кудрявцев В.Б., Алешин С.В. Комбинаторо-логический подход к распознаванию образов. Интеллектуальные системы. Том 1, вып. 1—4, 1996.

[4] Фролов А.Б. Модели и методы технической диагностики. М.: Знание, 1990.

[5] Фролов А.Б. Фролов Д.А. Алгоритмы распознавания упорядоченных объектов в системах принятия решений функционального типа. Вестник МЭИ, 1996, N 6.

[6] Roth J.P., Karp R.M. Minimization over Boolean graphs. IBM Journal of Research and Develovment, 1962, v. 6, N 2.

[7] Башлыков А.А., Еремеев А.П. Экспертные системы поддержки принятия решений в энергетике. М.: Издательство МЭИ, 1994.

[8] Биркгоф Г. Теория решеток. М.: Наука, 1984.

[9] Вапник В.Н., Червоненкис А.Я. Теория распознавания образов. М.: Наука, 1974.

[10] Ту Дж. Гонсалес Р. Принципы распознавания образов. М.: Мир, 1978.

[11] Фролов А.Б., Четрафилов И.Д. О некоторых подходах к распознаванию оптических образов текстов. Интеллектуальные системы, 1997, том. 2, вып. 1—4.

[12] Фролов А.Б., Четрафилов И.Д. Сети принятия решений для распознавания оптических образов текстов при наличии искажений. Вестник МЭИ, 1997, N 6. F10

[13] Гэри М., Джонсон Д. Вычислительные машины и трудно-решаемые задачи. М.: Мир,1982.