Литература

1. Белкин, В.Д. Инновации, инвестиции и институциональные изменения – главные составляющие стратегии долгосрочного развития / В.Д. Белкин, В. П. Стороженко // Экономическая наука современной России. – 2011–. № 2 (53). – С. 60-72
2. Беляков, Г.С. Как оценивать экономическую эффективность инвестиционных проектов / Г.С. Беляков // ЭКО. – 2010. – № 6.– С. 121-130
3. Ржевская, Т.Г. Инвестиционные аспекты модернизации отраслей промышленности / Т.Г. Ржевская // Экономика строительства – 2011. – № 4.– С. 31-36
4. Скляренко, В.К. Экономика предприятия. / В.К.Скляренко, В. М. Прудников – М.: ИНФРА – М, 2008. – 528 с.

АЛГОРИТМ ПРОВЕРКИ КОРРЕКТНОСТИ ТАБЛИЦ РЕШЕНИЙ[1]

В.В. Муромцев,
кандидат технических наук, доцент,
заведующий кафедрой математического и программного обеспечения информационных систем, НИУ «БелГУ»

Постановка задачи

Термин «человеческий потенциал» имеет различные трактовки. Часто под этим термином понимают качества людей принципиально влияющие на результаты их деятельности. К таким качествам относятся запас знаний и умений. В современном динамичном мире знания и умения могут быстро устаревать. Поэтому вопросы совершенствования процесса передачи знаний и повышения скорости обучения являются актуальными. Одним из путей повышения скорости обучения является активное использование информационных технологий. Однако для того, чтобы информационные технологии действительно ускоряли процесс обучения необходимо, чтобы знания были достоверны и представлены в доступной форме на некотором формальном языке. Одним из возможных способов представления знаний являются таблицы решений [1]. Пример представлен в табл.1. Последний столбец определяет правило «иначе».

Таблица 1

Специальный клиент (c_1)	*N*	*N*	*Y*	*
Приоритетный заказ (c_2)	*N*	*Y*	-	*
Международный заказ (c_3)	-	*Y*	-	*
Плата	150	100	70	80
Код предупредительного сигнала	0	0	1	2

При описании сложных условий таблица решений является компактным средством описания знаний и легка для понимания. Использование в проектах таблиц решений позволят упростить организацию взаимодействия между специалистами в различных областях. Таблицы решений могут успешно сочетаться с другими средствами описания проектов, например, с диаграммами UML.

Одной из задач при построении таблиц решений является проверка ее корректности. В данном случае подразумевается проверка таблицы с целью выявления, не приводит ли некоторый вариант выполнения условий к требованию выполнения различных действий или к неопределенному действию. Эту задачу требуется решать многократно при разработке таблицы. Для устранения ошибок проектирования проверка корректности таблицы решений должна осуществляться в автоматическом режиме. В работе предлагается алгоритм проверки корректности таблицы решений. Алгоритм состоит из следующих шагов:

1. Таблица решений представляется системой логических функций.

[1] Исследование выполнено в рамках Государственного задания Министерства образования и науки РФ на выполнение НИР подведомственным вузам в 2013 году. Проект № 8.8600.2013.

2. Полученная на первом шаге система логических функций представляется бинарной диаграммой решений (БДР) [2].

3. Производится анализ результата построения БДР и делается вывод о корректности исходной таблицы решений.

1. Представление таблицы решений системой логических функций

Таблицу решений будем рассматривать как четверку $T=(C,A,S,O)$, где C – множество условий, A – множество действий, $S:W^n \to A_1\times\ldots\times A_k$ - функция, задающая для каждого варианта выполнения условий необходимые действия, $n=|C|$ – число условий, $k=|A|$ – число действий, $W=\{"N","Y","-"\}$, $A_i, i=1,2,\ldots,L$ – множества возможный значений для i-го действия, $O=\varnothing$ или $O\in A_1\times\ldots\times A_k$ - правило «иначе», т.е. действия, которые необходимо выполнить если не подходит ни один из определенных вариантов выполнения условий.

Пусть X и $Y=\{y_1,y_2,\ldots\}$, $|X|\le|Y|$ - область определения и область значений функции S соответственно. Тогда алгоритм представления таблицы решений системой логических функций в дизъюнктивной нормальной форме будет следующий:

1. Для $i=1,\ldots,m$, где $m=|Y|$ выполнить:

- Найти $P=\{x\,|\,S(x)=y_i\}$ - прообраз элемента y_i.
- Для каждого $x=(x_1,\ldots,x_n)\in P$ выписать элементарную конъюнкцию $c_1^{\sigma_1}\cdot\ldots\cdot c_n^{\sigma_n}$, где $c_j\in C$; $\sigma_j=1$, если $x_j="Y"$; $\sigma_j=0$, если $x_j="N"$ и буква $c_j^{\sigma_j}$ отсутствует в конъюнкции, если $x_j="-"$.
- Все выписанные элементарные конъюнкции объединить дизъюнкцией. В результате получим логическую функцию f_i.

2. Все логические функции f_i, $i=1,\ldots,m$ объединить в систему. Для каждой функции f_i запомнить соответствующие действия y_i.

Представим в виде системы логических функций рассмотренную ранее таблицу решений. Обозначим условие «Специальный клиент» через c_1, «Приоритетный заказ» - c_2, «Международный заказ» - c_3. После таких обозначений $C=\{c_1,c_2,c_3\}$. Также для этого примера $A_1=\{150,100,70,80\}$, $A_2=\{0,1,2\}$, $O=(80,2)$, $S=\{(("N","N","-"),(150,0)),(("N","Y","Y"),(100,0)),(("Y","-","-"),(70,1))\}$.

Область определения функции S: $X=\{("N","N","-"),("N","Y","Y"),("Y","-","-")\}$.

Область значений функции S: $Y=\{(150,0),(100,0),(70,1)\}$. В данном примере $|X|=|Y|$, но как было отмечено выше в общем случае $|X|\le|Y|$.

Для $i=1$ прообраз $y_1=(150,0)$ будет следующим $P=\{("N","N","-")\}$. Множество P состоит из одного элемента - вектора $("N","N","-")$. Этому вектору соответствует следующая элементарная конъюнкция – $c_1^0\cdot c_2^0=\bar{c}_1\cdot\bar{c}_2$. В результате получили логическую функцию $f_1=\bar{c}_1\cdot\bar{c}_2$. Выполнив аналогичные действия для $i=2,3$, получим систему из трех логических функций: $f_1=\bar{c}_1\cdot\bar{c}_2$, $f_2=\bar{c}_1\cdot c_2\cdot c_3$, $f_3=c_1$.

2. Построение бинарной диаграммы решений

Пусть исходной таблице решений соответствует система $\Sigma=\{f_1,\ldots,f_m\}$ логических функций от n переменных. БДР, представляющей систему Σ будем называть ациклический граф, в котором:

1. Имеется ровно одна вершина без входящих в нее дуг (начальная вершина).
2. Все вершины по числу выходящих дуг делятся на два типа:

- вершины, помеченные переменной c_j, $j\in\{1,\ldots,n\}$, из которых выходит две дуги, помеченные символами 0 и 1 (условные вершины);

- вершины, помеченные вектором $(\sigma_1,...,\sigma_m)$, $\sigma_i \in \{0,1\}$, $i=1,...,m$, и не имеющие выходящих дуг (заключительные вершины).

В основу алгоритма построения БДР, представляющего систему Σ положено разложение каждой функции системы Σ по некоторой переменной c_j, $j \in \{1,...,n\}$: $f_i(0)=f_i(c_1,...,c_{j-1},0,c_{j+1},...,c_n)$, $f_i(1)=f_i(c_1,...,c_{j-1},1,c_{j+1},...,c_n)$, $i=1,...,m$. Такое разложение отображается на графе (V,E), где $V=\{a,b,c\}$, $E=\{(a,b),(a,c)\}$ следующим образом: вершина a помечается системой Σ и символом c_j, вершины b и c помечаются системами $\Sigma^0=\{f_1(0),...,f_m(0)\}$ и $\Sigma^1=\{f_1(1),...,f_m(1)\}$ соответственно. Кроме того, дуга (a,b) помечается символом «0», а дуга (a,c) - символом «1».

Пусть в результате разложения получена некоторая система $\{\sigma_1,...,\sigma_m\}$, $\sigma_i \in \{0,1\}$, $i=1,...,m$, т.е. получена заключительная вершина БДР, тогда справедливы следующие утверждения:

1. Если $\sigma_p=1$, $p \in \{1,...,m\}$ и $\sigma_i=0$, $i=1,...,m$, $i \neq p$, то получена заключительная вершина, соответствующая действиям y_p.

2. Если $\sigma_i=0$, $i=1,...,m$, то получена заключительная вершина БДР, соответствующая правилу «иначе».

3. Если $\sigma_p=1$, $\sigma_r=1$, $p,r \in \{1,...,m\}$, то исходные данные содержат ошибку (не ясно, какое из действий y_p или y_r следует выполнить).

Учитывая эти утверждения, разработан алгоритм построения БДР, представляющей систему логических функций (рис.1).

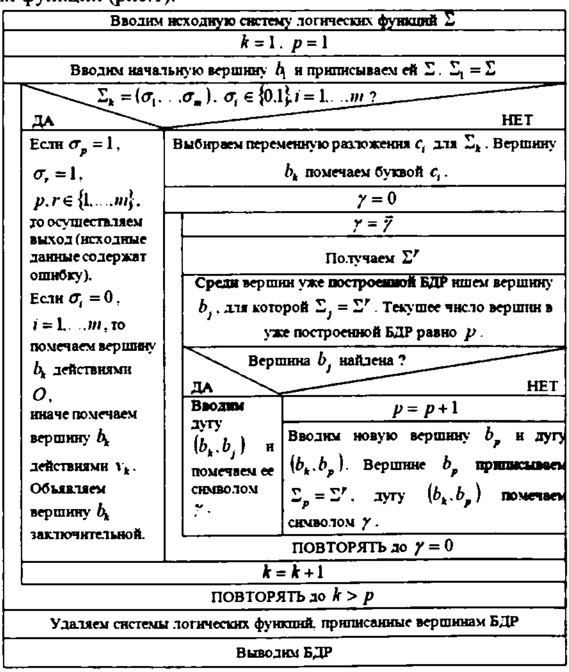

Рис. 1. Базовый алгоритм построения БДР

3. Анализ результата построения БДР

Следует отметить, что число условных вершин в БДР зависит от порядка выбора переменных разложения. Например, для рассматриваемого примера могут получиться БДР представленные на рис.2. На рис.2а показана БДР, состоящая из 3 условных вершин. Данная БДР получилась при следующем порядке выбора переменных разложения: c_1, c_2, c_3. На рис.2б показана неудачная БДР, состоящая из 6 условных вершин. Данная БДР получилась при следующем порядке выбора переменных разложения: c_3, c_2, c_2, c_1, c_1, c_1.

Для проверки корректности исходной таблицы решений не важно, какая из БДР будет получена. В полученных БДР нет условных вершин, помеченных системой $\{\sigma_1,...,\sigma_m\}\ \sigma_i \in \{0,1\}$, в которой $\sigma_p = 1$, $\sigma_r = 1$, $p,r \in \{1,...,m\}$. Т.е. в таблице нет ни одного варианта выполнения условий приводящего к требованию выполнения различных действий. Также следует отметить, что исходная таблица содержала правило "иначе", поэтому в таблице нет ни одного варианта выполнения условий приводящего к неопределенному действию. Если правило "иначе" было бы не определено, то в полученной БДР следовало бы проверить отсутствие условных вершин, помеченных системой $\{\sigma_1,...,\sigma_m\}\ \sigma_i = 0$, $i = 1,...,m$.

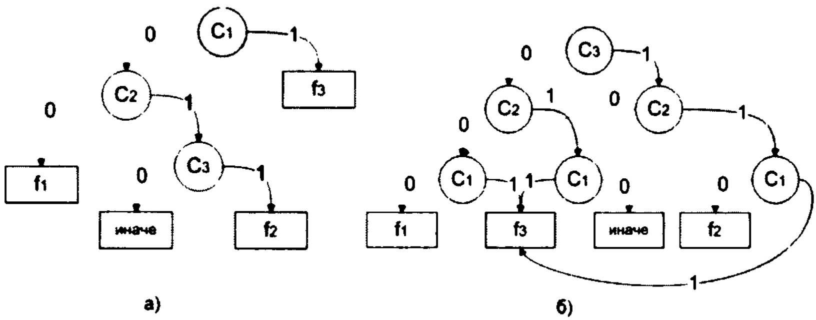

Рис. 4. Бинарные диаграммы решений

Выводы

В работе рассмотрен способ представления знаний с помощью таблиц решений. Отмечено, что использование таких таблиц позволяет повысить скорость обучения и облегчить организацию взаимодействия специалистов в различных областях при работе над некоторым проектом, т.е. организовать командную работу. Также отмечено, что при разработке таблиц решений необходимо контролировать их корректность в автоматическом режиме. Для этих целей предложен алгоритм проверки корректности таблиц решений. Алгоритм основан на представлении таблицы решений с помощью системы логических функций, для которой далее строится БДР и производится ее анализ.

Литература

1. Хамби Э. Программирование таблиц решений. – Пер. с англ. - М.:Мир, 1976.

2. Knuth D. Fun With Binary Decision Diagrams (BDDs). [Электронный ресурс]: видео лекция. – Режим доступа: http://myvideos.stanford.edu/player/slplayer.aspx?coll=ea60314a-53b3-4be2-8552-dcf190ca0c0b&co=18bcd3a8-965a-4a63-a516-a1ad74af1119&o=true, свободный.