



3. Количество разработчиков, не вовлеченных в сообщества по анализу уязвимостей велико, однако оно не оказывает решительного влияния на количество обнаруженных угроз.

4. Влияние открытых сообществ велико и количество обнаруженных и заявленных угроз постоянно растет.

5. Корпоративный сектор (Microsoft, IBM и др.) активно развивает направление в области информационной безопасности и аудита компьютерных сетей и программных комплексов.

### Литература

1. W3Schools. OS Platform Statistics [Электронный ресурс]. – 2016. – Режим доступа: [http://www.w3schools.com/browsers/browsers\\_os.asp](http://www.w3schools.com/browsers/browsers_os.asp)

2. HP. The collateral damage of cybercrime [Электронный ресурс]. – 2016. – Режим доступа: <http://www8.hp.com/us/en/software-solutions/cyber-risk-report-security-vulnerability/>

3. Trinidad Mark. Top 10 Database Vulnerabilities and Vulnerabilities and Misconfigurations [Электронный ресурс]. – 2012. – Режим доступа: [http://www.sifma.org/uploadedfiles/societies/sifma\\_internal\\_auditors\\_society/top10-database-vulnerabilities-and-misconfigurations.pdf](http://www.sifma.org/uploadedfiles/societies/sifma_internal_auditors_society/top10-database-vulnerabilities-and-misconfigurations.pdf)

4. *J. Applebaum* A Model of Outbound Client Traffic on The Tor Anonymity Network. // Wesleyan University. - 2013. - P. 54-58.

5. *Иванов М. А.* Криптографические методы защиты информации в компьютерных системах и сетях. // КУДИЦ-ОБРАЗ. - 2001. - С. 368-376.

6. *Олифер В. Г., Олифер Н. А.* Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов // СПб.: Питер. - 2001. - 672 с..

7. *Столлингс В.* Основы защиты сетей. Приложения и стандарты = Network Security Essentials. Applications and Standards // «Вильямс». - 2002. - С. 429-440.

8. Softonic. Top downloads [Электронный ресурс]. – 2016. – Режим доступа: <http://en.softonic.com/windows/top-downloads>

О.Н. Долинина, Н.К.Сучкова

### ФОРМАЛЬНЫЕ МОДЕЛИ ОШИБОК КЛАССА «ИЗБЫТОЧНОСТЬ» В БАЗАХ ЗНАНИЙ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

(Саратовский государственный технический университет  
имени Гагарина Ю.А.)

#### Введение

Интеллектуальные системы (ИС) принятия решений находят применение во множестве областей: промышленности, медицине, научно-исследовательской деятельности, образовании, что влечет за собой повышение требований к их надёжности. Особую роль при этом играет обеспечение качества баз знаний интеллектуальных систем, являющихся центральным звеном



ИС, в то же время способы их отладки на сегодняшний день всё ещё не формализованы. В большинстве случаев отладкой занимаются эксперты, что влияет на увеличение стоимости разработки и не гарантирует отсутствия ошибок в базах знаний.

В литературе описаны различные подходы для отладки баз знаний проверки корректности и полноты баз знаний [1-7]. Некоторые ошибки в знаниях могут быть выявлены только в процессе тестирования. К таким относятся, ошибки, связанные с противоречивостью самой предметной области, например, ошибки вида «забывание об исключении» [1]. В то же время ряд ошибок, связанных со структурой знаний т.н. называемые структурные ошибки могут быть обнаружены и исправлены на стадии статической отладки базы знаний. Отсутствие структурных ошибок не гарантирует отсутствия ошибок в знаниях, однако повышает эффективность принятия решений за счет приведения базы знаний в состояние статической корректности [2] и соответствующего уменьшения времени на принятие решения. Поэтому первым этапом отладки базы знаний должна быть статическая отладка, реализуемая автоматизированной верификацией. В работах [3-7] приведены некоторые структурные ошибки в базах знаний и описаны алгоритмы их обнаружения, однако отсутствует их формализация. В данной работе обобщается информация о наиболее распространенном классе структурных ошибок – избыточности и предлагаются формальные модели данного класса ошибок, которые могут быть использованы для автоматической верификации структур баз знаний.

#### **Формализация класса ошибок «избыточность» в продукционных базах знаний**

Наиболее распространённый способ представления баз знаний интеллектуальных систем - продукции. Продукционную базу знаний можно представить в следующем виде:

$$P = (F, R, G, C, I), \quad (1)$$

где  $F$  – множество фактов о решаемой проблеме;  $R$  – множество правил вида

$$r_m: \text{если } f_i \text{ и } f_j \dots \text{ и } f_n \text{ то } f_k; \quad (2)$$

$G$  – множество целей;  $C$  – множество разрешенных комбинаций фактов;  $I$  – интерпретатор правил, реализующий вывод.

Пусть  $S$  – множество входных фактов, т.е. фактов, устанавливаемых пользователем в интеллектуальной системе.  $S \subset F$ .

Воспользуемся представлением базы знаний в виде И/ИЛИ графа. Например, пусть база знаний содержит следующие правила:

- $r_1$ : если  $s_1$  и  $s_2$ , то  $f_1$ ;
- $r_2$ : если  $s_2$  и  $s_3$ , то  $f_2$ ;
- $r_3$ : если  $s_3$  и  $s_4$  и  $s_5$ , то  $f_3$ ;
- $r_4$ : если  $f_1$ , то  $g_1$ ;
- $r_5$ : если  $f_2$  и  $f_3$ , то  $g_2$ .

Тогда приведённым правилам соответствует И/ИЛИ граф на рисунке 1.

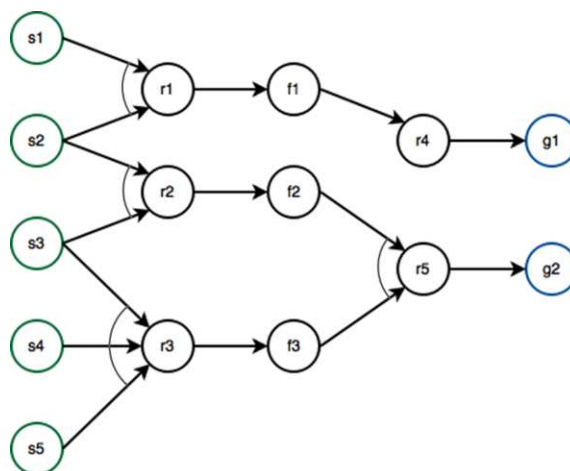


Рисунок 1. И/ИЛИ граф

Правило  $r_i$  вида

$$\text{если } f_{r_{i1}} \text{ и } f_{r_{i2}} \dots f_{r_{in}}, \text{ то } f_{r_{im}} \quad (3)$$

можно представить в виде пары:  $r_i = (D_{r_i}; Q_{r_i})$ , где  $D_{r_i} = \{f_{r_{i1}}, f_{r_{i2}}, \dots, f_{r_{in}}\}$  и  $Q_{r_i} = \{f_{r_{im}}\}$ . Как видно из представления,  $Q_{r_i}$  состоит из единственного элемента, далее обозначаемого как  $q_{r_i}$ .

Пусть  $L$  – множество цепочек вывода.

**Определение 1.** Цепочка вывода  $l_i$  – последовательность правил  $(r_{l_{i1}}, r_{l_{i2}}, \dots, r_{l_{in}})$  такая, что  $\forall r_{l_{ik}}, r_{l_{i(k+1)}}, q_{r_{l_{ik}}} \in D_{r_{l_{i(k+1)}}$  при  $k = 2, \dots, (n-1)$ .

Тогда для графа, приведённого на рис. 1,  $L = \{l_1, l_2, l_3, l_4, l_5, l_6, l_7, l_8\}$ , где  $l_1 = (r_1); l_2 = (r_2); l_3 = (r_3); l_4 = (r_4); l_5 = (r_5); l_6 = (r_1, r_4); l_7 = (r_2, r_5); l_8 = (r_3, r_5)$ ;

**Определение 2.** Начало цепочки вывода  $l_i$  вида  $(r_{l_{i1}}, r_{l_{i2}}, \dots, r_{l_{in}})$  – множество фактов в условии выполнения первого правила,  $D_{l_i} = D_{r_{l_{i1}}}$ .

**Определение 3.** Конец цепочки вывода  $l_i$  вида  $(r_{l_{i1}}, r_{l_{i2}}, \dots, r_{l_{in}})$  – следствие последнего правила,  $Q_{l_i} = Q_{r_{l_{in}}}$ .

**Определение 4.** Структурной ошибкой в продукционной базе знаний называется ошибка, обнаруживаемая в ходе анализа И/ИЛИ графа. Базы знаний, в которых отсутствуют структурные ошибки, являются статически корректными.

Класс ошибок «Избыточность» включает наличие дубликатов, циклов, избыточных, незначущих, некорректных цепочек вывода в базах знаний. Рассмотрим подробнее подклассы избыточности.

**Определение 5.** Правила  $r_i$  и  $r_j$  называются дубликатами, если  $D_{r_i} \cap D_{r_j} \neq \emptyset$  и  $q_{r_i} = q_{r_j}$ . Выделяют 3 вида дубликатов:

- включающие дубликаты;
- полные дубликаты;
- неполные дубликаты.

**Определение 6.** Правила  $r_i$  и  $r_j$  называются включающими дубликатами, если  $D_{r_i} \subset D_{r_j}$ , при этом  $D_{r_i} \neq D_{r_j}$  и  $q_{r_i} = q_{r_j}$ . В этом случае правило  $r_i$  называется включаемым.

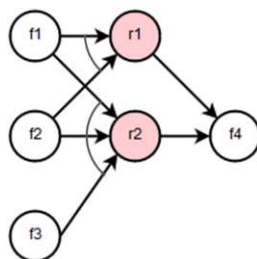


Рисунок 2. Включающие дубликаты

Правила  $r_1$  и  $r_2$  на рисунке 2 являются включающими дубликатами.

**Определение 7.** Правила  $r_i$  и  $r_j$  называются полными дубликатами, если

$$D_{r_i} = D_{r_j} \text{ и } q_{r_i} = q_{r_j}.$$

Правила  $r_1$  и  $r_2$  являются полными дубликатами:

- $r_1$ : если  $f_1$  и  $f_2$ , то  $f_3$ ;
- $r_2$ : если  $f_1$  и  $f_2$ , то  $f_3$

**Определение 8.** Правила  $r_i$  и  $r_j$  называются неполными дубликатами, если

$$D_{r_i} \cap D_{r_j} \neq \emptyset \text{ и } q_{r_i} = q_{r_j}, \text{ при этом } D_{r_i} \setminus D_{r_j} \neq \emptyset \text{ и } D_{r_j} \setminus D_{r_i} \neq \emptyset.$$

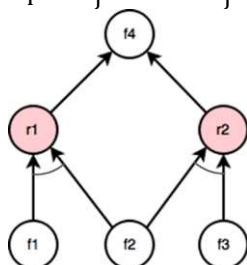


Рисунок 3. Неполные дубликаты

На рисунке 3 приведен пример неполных дубликатов:

$r_1$ : если  $f_1$  и  $f_2$ , то  $f_4$ ;

$r_2$ : если  $f_2$  и  $f_3$ , то  $f_4$ ;

$$D_{r_1} \cap D_{r_2} = f_2$$

Для исправления ошибки вида неполные дубликаты нет универсального способа, в каждом конкретном случае необходимые изменения определяются экспертом.

**Определение 9.** Цепочка вывода  $l_i$  называется избыточной, если  $q_{l_i} \notin G$  и  $\neg \exists l_j$  такой, что  $q_{l_i} \in D_{l_j}$  и  $q_{l_j} \in G$ .

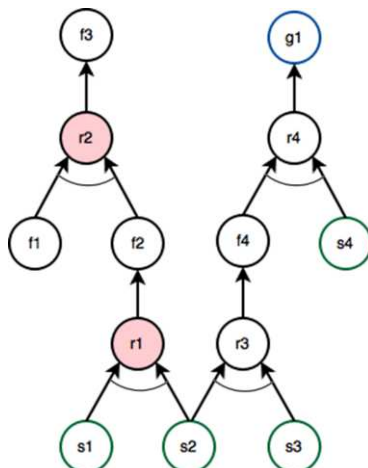


Рисунок 4. Избыточная цепочка вывода



На рисунке 4 приведен пример избыточной цепочки вывода  $l_1 = (r_1, r_2)$ , где

- $r_1$ : если  $s_1$  и  $s_2$ , то  $f_2$ ;
- $r_2$ : если  $f_1$  и  $f_2$ , то  $f_3$ ;
- $q_{l_1} = f_3$ ;
- $f_3 \notin G$ .

**Определение 10.**  $l_i$  вида  $(r_{i1}, r_{i2}, \dots, r_{in})$  называется незначащей цепочкой вывода, если  $\forall r_{ij}, |D_{r_{ij}}| = 1$  при  $j = 1, \dots, n$ .

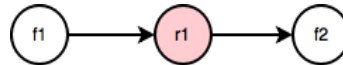


Рисунок 5. Незначащая цепочка вывода

$l_1 = (r_1)$  на рисунке 5 является незначащей цепочкой вывода:

- $D_{r_1} = \{f_1\}$ ;
- $|D_{r_1}| = 1$ ;

Выделяют 2 вида незначащих цепочек вывода:

- явные;
- неявные.

**Определение 11.** Незначащая цепочка вывода  $l_i$  называется явной, если  $\exists r_j$  такое, что  $r_j = (D_{l_i}; Q_{l_i})$ .

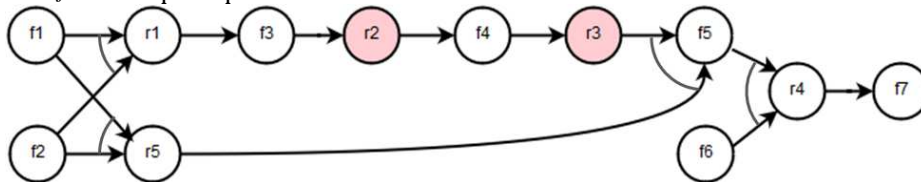


Рисунок 6. Явная незначащая цепочка вывода

Пример явной незначащей цепочки вывода приведён на рисунке 6, где

- $l_1 = (r_2, r_3)$ ;
- $r_2$ : если  $f_3$ , то  $f_4$ ;
- $r_3$ : если  $f_4$ , то  $f_5$ ;
- $D_{l_1} = \{f_3\}$ ;
- $q_{l_1} = \{f_5\}$ ;
- существует правило  $r_5 = \{D_{l_1}; q_{l_1}\}$ .

**Определение 12.** Незначащая цепочка вывода  $l_i$  называется неявной, если  $\neg \exists r_j$  такое, что  $r_j = (D_{l_i}; Q_{l_i})$ .

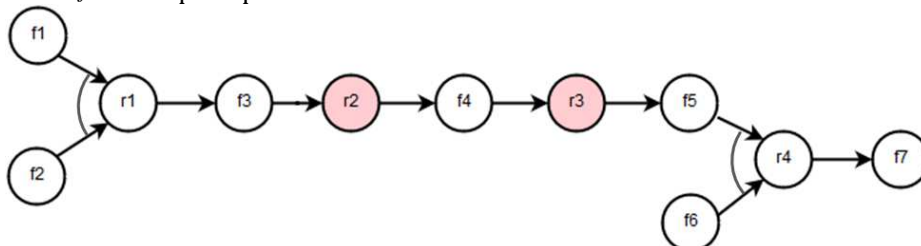


Рисунок 7. Неявная незначащая цепочка вывода



На рисунке 7 приведен пример неявной незначащей цепочки  $l_1 = (r_2, r_3)$ :

- $r_2$ : если  $f_3$ , то  $f_4$ ;
- $r_3$ : если  $f_4$ , то  $f_5$ ;
- $D_{l_1} = \{f_3\}$ ;
- $q_{l_1} = \{f_5\}$ ;
- $r_5 = \{D_{l_1}; q_{l_1}\}$ ;
- не существует правила  $r_k = \{D_{l_1}; q_{l_1}\}$ .

Неявная незначащая цепочка вывода не является критичной ошибкой, однако допускает оптимизацию за счет сведения к предшествующему или последующему правилу.

Различают 2 типа некорректных цепочек вывода:

- цепочки, избыточные по входу;
- цепочки, избыточные по выходу.

**Определение 15.** Цепочка вывода  $l_i$  называется избыточной по входу, если  $D_{l_i} \cap G \neq \emptyset$ .

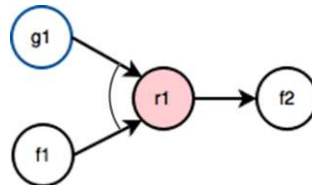


Рисунок 8. Цепочка вывода, избыточная по входу

На рисунке 8 приведен пример цепочки вывода, избыточной по входу,  $l_1 = (r_1)$ :

- $r_1$ : если  $g_1$  и  $f_1$ , то  $f_2$ ;
- $g_1 \in G$ .

**Определение 16.** Цепочка вывода  $l_i$  называется избыточной по выходу, если  $q_{l_i} \in S$ .

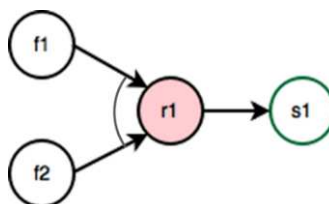


Рисунок 2. Цепочка вывода, избыточная по выходу

Пример цепочки вывода, избыточной по выходу, приведен на рисунке 13:

- $l_1 = (r_1)$ ;
- $r_1$ : если  $f_1$  и  $f_2$ , то  $s_1$ ;
- $s_1 \in S$ .

**Определение 17.** Цепочка вывода  $l_i$  называется циклом, если  $q_{l_i} \in D_{l_i}$ .

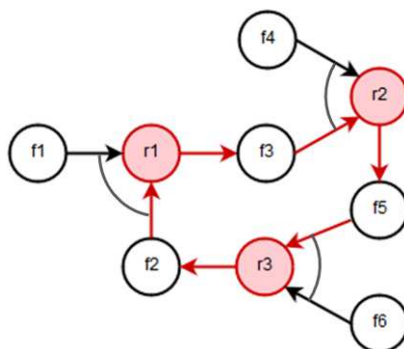


Рисунок 10. Цикл

На рисунке 10 приведен пример цикла  $l_1 = (r_1, r_2, r_3)$ , где

- $r_1$ : если  $f_1$  и  $f_2$ , то  $f_3$ ;
- $r_2$ : если  $f_3$  и  $f_4$ , то  $f_5$ ;
- $r_3$ : если  $f_5$  и  $f_6$ , то  $f_2$ ;
- $D_{l_1} = \{f_1, f_2\}$ ;
- $q_{l_1} = \{f_2\}$ ;
- $q_{l_1} \in D_{l_1}$ .

Частным случаем цикла является простой цикл.

**Определение 18.** Правило  $r_i$  называется простым циклом, если  $q_{r_i} \in D_{r_i}$ .

Например,  $r_1$ : если  $f_1$  и  $f_2$ , то  $f_1$ .

### Заключение

В работе приведена формализация класса структурных ошибок типа «избыточность» в продукционных базах знаний интеллектуальных систем, характеризующегося наличием дубликатов, избыточных, незначащих некорректных цепочек вывода и циклов.

Формальные модели структурных ошибок могут быть использована в качестве основы для проведения автоматической верификации структуры продукционных баз знаний.

### Литература

1. Долинина О.Н. Разработка метода тестирования продукционных баз знаний экспертных систем с учетом ошибок типа «забывания об исключении»: дис...канд. техн. наук / О.Н. Долинина. Саратов, 1999. 171 с.
2. Dolinina, O. Method of the Debugging of the Knowledge Bases of Intellectual Decision Making Systems /Advances in Intelligent Systems and Computing /Automation Control Theory Perspectives in Intelligent Systems // Proc. Of the 5th Computer Science On-line Conf. 2016 (CSOC2016), vol.3, 2016, p.301-315.
3. Checking Expert System Knowledge bases for consistency and completeness / T. Nguen, W. Perkins, T. Laffey, W. Pecors // Proc. of the 9th Int. Joint Conf. on AI, Los.-Ang., August, 1985. P. 375-378.
4. Cragun B.J. A decision-table-based processor for checking completeness and consistency in rule-based expert systems / B.J. Cragun, H.J Stendel // Int. J. Man-Mach. Stud. 1987. Vol. 26. N. 5. P. 633-648.



5. Nguen T.A. Verifying consistency of Production Systems / T.A. Nguen // Proc. of Conf.on AI Applications (CAIA), 3rd, Kissimmee, Fl, 1987. P. 4-8.
6. Suwa H., Scott A.C. , Shortliffe. An Approach to Verifying Consistency and Completeness in a Rule-Based Expert System // Rule-Based Expert Systems. – London: Addison – Wesley, 1984, P. 159-170.
7. Nguen T., Perkins W., Laffey T., Pecora W. Checking Expert System Knowledge Bases for consistency and completeness // Proc. Of the 9th Int. Joint Conf.on AI, Los.Ang. August 1985. –P. 375-378.

В.Е. Гвоздев, Д.В. Блинова, Г.Ф. Байtimiрова

## ЭЛЕМЕНТЫ ТЕРРИТОРИАЛЬНО-ОРИЕНТИРОВАННОЙ СТРАТЕГИИ УПРАВЛЕНИЯ ПРОМЫШЛЕННЫМИ ОТХОДАМИ НА ТЕРРИТОРИИ РБ

(Уфимский государственный авиационный технический университет)

### 1. Введение

Проблема мусора и несанкционированных свалок становится все актуальнее в наш 21 век, век современной цивилизации, развития науки и техники.

Средняя норма образования твердых бытовых отходов (ТБО) на одного человека в год достигает 300-350 кг, и эта цифра продолжает неуклонно расти. Ежегодно образуется порядка 35-40 млн тонн твердых бытовых отходов [2,3]. В Республике Башкортостан ежегодный объем образования ТБО за последние 5 лет увеличился в два раза и в 2012 году составил 1,77 млн.т [4]. Республика Башкортостан входит в десятку регионов Российской Федерации с наибольшим объемом образования ТБО за год.

На территории Республики Башкортостан расположено 14 городских поселений, 819 сельских поселений, 54 муниципальных района и 8 городских округов. Количество населенных пунктов, охваченных системами сбора и удаления отходов, по отношению к общему количеству населенных пунктов Республики Башкортостан составляет около 35% от общего количества.

Образование и поступление ТКО по Республике Башкортостан за 2015 год составило 1 232 477,29 тонн. Заметим, что это данные по форме 2-ТП, а учет образования ТКО от малых и средних предприятий ведется не в полном объеме ввиду несдачи предприятиями вышеуказанной формы.

Согласно ФЗ-458 «О внесении изменений в Федеральный закон «Об отходах производства и потребления» предполагалось, что к 1 января 2017 года в каждом субъекте Федерации должна быть разработана, утверждена и готова к запуску территориальная схема обращения с твердыми бытовыми отходами и определены региональные операторы. Но большинство регионов не успевает вовремя выполнить эту работу и введение этих схем было отложено.

### 2. Концептуальная основа оригинальности подхода

Отходами нужно управлять, а для этого нужно иметь эффективную систему информационной поддержки. Эта мысль не нова, но ее особенность в дан-