

УДК 004.891

**В. М. Рувинская**, канд. техн. наук,  
**А. С. Тройнина**,  
**Е. Л. Беркович**, канд. ф.-м. наук,  
**А. Ю. Черненко**

### АВТОМАТИЗАЦИЯ ПРОВЕРКИ ПРАВИЛ ЭКСПЕРТНОЙ СИСТЕМЫ ДЛЯ МОНИТОРИНГА РАБОТЫ КОМПЬЮТЕРНОЙ СЕТИ

*Аннотация.* Посвящена усовершенствованию существующих методов контроля и визуализации правил экспертной системы мониторинга с учетом требований предметной области мониторинга компьютерных сетей, при этом добавлена возможность работы с иерархией правил, проверка их цикличности и избыточности, а также минимизация множества правил. В рамках данной работы был создан усовершенствованный редактор правил и с его помощью выполнена разработка базы знаний для мониторинга компьютерной сети.

*Ключевые слова:* мониторинг, компьютерная сеть, базы правил, непротиворечивость, цикличность, полнота

**V. M. Ruvinskaya**, PhD.,  
**A. S. Troynina**  
**E. L. Berkovich**, PhD.,  
**A. U. Chernenko**

### EXPERT SYSTEM FOR MONITORING OF COMPUTER NETWORK WORK RULES CHECKING AUTOMATION

*Abstract.* This article is devoted to improvement of existing methods of the rules control and visualization for the monitoring expert system to meet the requirements of the subject area “computer networks monitoring”, namely, ability of work with a hierarchy of rules was added, verification of their cyclist and redundancy, as well as minimization of rules sets. As part of this work and improved rules editor was created to use it for developing a knowledge base for network monitoring.

*Keywords:* monitoring, rules base, computer network, consistency, cyclicity, completeness

**В. М. Рувинская**, канд. техн. наук,  
**А. С. Тройнина**,  
**Е. Л. Беркович**, канд. ф.-м. наук,  
**А. Ю. Черненко**

### АВТОМАТИЗАЦІЯ ПЕРЕВІРКИ ПРАВИЛ ЕКСПЕРТНОЇ СИСТЕМИ ДЛЯ МОНИТОРИНГУ РОБОТИ КОМП'ЮТЕРНОЇ МЕРЕЖІ

*Анотація.* Присвячена вдосконаленню існуючих методів контролю та візуалізації правил експертної системи моніторингу з урахуванням вимог предметної області моніторингу комп'ютерних мереж, при цьому, додана можливість роботи з ієрархією правил, перевірка їх циклічності і надмірності, а також мінімізація множини правил. У рамках роботи було створено вдосконалений редактор правил і з його допомогою виконана розробка бази знань для моніторингу комп'ютерної мережі.

*Ключові слова:* моніторинг, комп'ютерна мережа, база правил, несуперечливість, циклічність, повнота

#### Введение

Технологии в области компьютерных систем и сетей постоянно развиваются. Пересматриваются традиционные модели разработки и архитектуры приложений, что приводит к возникновению новых принципов построения программных продуктов, одним из которых являются облачные вычисления [1]. Доступность облака зависит от надежности сетевой инфраструктуры. Однако неза-

висимо от качества настройки и качества аппаратного обеспечения в ее составе нельзя полагаться на возможность самостоятельной стабильной работы сети [2] даже при наличии системного администратора, ответственного за состояние сети.

Помощь в работе сетевого администратора оказывают инструментальные средства для мониторинга и управления сетью. Сравнимая рассмотренные в данной работе системы мониторинга сети [3]: (RRDTool, Cacti, Zabbix, Monit, Nagios, Icinga), можно отметить, что все современные продукты, разработка и исполь-

© Рувинская В.М., Тройнина А.С.,  
Беркович Е.Л., Черненко А. Ю., 2014

зование которых еще продолжают, предоставляют широкие возможности наблюдения за узлами сети по различным протоколам (SNMP, ICMP, FTP, HTTP, SMTP, POP3 и др.) и различными способами (путем отправки простых запросов, с помощью агентов, устанавливаемых на узлы). При этом можно подчеркнуть исчерпывающую информацию о текущих показатели сети и отдельных ее узлов, как объем сетевого трафика, время обработки запросов, загрузка центрального процессора, оперативной памяти, дисковой подсистемы и т.д.). Также большинством систем мониторинга предоставляются возможности визуализации полученных данных, уведомлений системному администратору в определенных ситуациях, построение карты сети и другие. К сожалению, лишь немногие системы мониторинга способны анализировать собранные данные и идентифицировать на основе полученных результатов ряд неполадок, которые могут возникать в сети; при этом возможности анализа ограничены, так как не предоставляют пользователям возможности создания собственных правил для анализа.

Возможность автоматизированного выявления неполадок сети полезна, поскольку позволяет снизить нагрузку на системного администратора, и повысить вероятность раннего обнаружения и исправления. Определение собственных правил при этом позволяет улучшить качество такой автоматизации, а в некоторых случаях оно просто необходимо, особенно если в сети применяется нестандартное аппаратное обеспечение или архитектурные решения. Таким образом, целесообразно применение в системах мониторинга компьютерных сетей базы знаний о возможных неисправностях сети. Это возможно с помощью применения интеллектуальных экспертных систем, которые, опираясь на базу знаний о возможных неполадках, позволяют их идентифицировать [4-5].

Полезность работы таких систем, однако, в значительной степени зависит от качества базы, которая наполняется знаниями экспертов в области администрирования сетей. При этом преобразование знаний экспертов в вид, пригодный для использования экспертной системой, по ряду причин является нетривиальной задачей, поэтому разра-

ботка качественных баз знаний является сложным и трудоемким процессом [6].

В опубликованной ранее работе [7] предлагалось упрощение процесса создания и поддержки базы правил экспертной системы (ЭС) для мониторинга выполнения требований по безопасной эксплуатации электроустановок на основе визуализации правил и автоматизированной проверки их на непротиворечивость и полноту. Под упрощением понималось и сокращение времени при создании базы правил, и понижение уровня сложности, что выражается в уменьшении ошибок при создании правил и их корректировке.

Теперь рассматривается новая предметная область по мониторингу компьютерных сетей, что требует усовершенствования предложенных подходов и их обобщения.

**Целью работы** является снижение сложности разработки, проверки и сопровождения баз знаний для мониторинга (и, в частности, мониторинга работы компьютерных сетей) за счет сокращения времени, необходимого для создания базы знаний, снижения уровня трудоемкости, а также уменьшение количества потенциальных ошибок, допускаемых в процессе изменения знаний, за счет автоматизации проверок правил на противоречивость, цикличность, неполноту, избыточность, а также минимизации множества правил.

В ходе выполнения работы были рассмотрены следующие задачи: рассмотрены типичные неполадки, которые могут возникнуть в компьютерных сетях и на их узлах, а также характерные признаки, позволяющие идентифицировать эти неполадки; в процессе разработки базы знаний для экспертной системы мониторинга сети выполнено усовершенствование существующих инструментов разработки баз знаний.

### **1. Визуализация знаний с помощью И/ИЛИ графа**

На рис. 1 представлена предлагаемая структурно-функциональная схема системы мониторинга с использованием ЭС для анализа. Пунктиром выделены компоненты системы, предложенные авторами, связанные с разработкой правил.

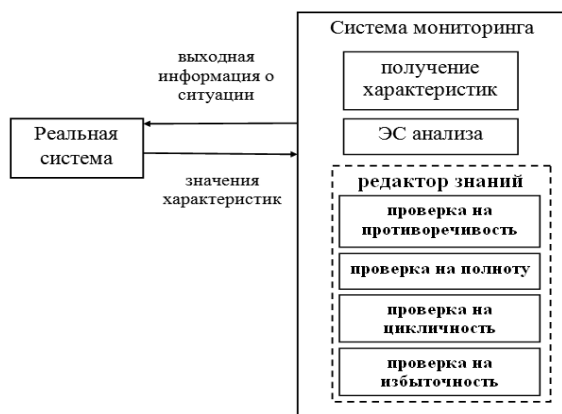


Рис. 1. Структурно–функциональная схема системы мониторинга

В качестве формы представления знаний для использования в системах мониторинга были выбраны факты и правила [8]. Такая форма представления знаний выгодно отличается от других близостью к естественному языку, что означает легкость формулирования знаний. Проверка правил в таком виде, однако, затруднена при большом количестве правил. В частности, крайне трудно отслеживать взаимосвязи между правилами. Решением этой проблемы является альтернативное представление правил в виде И/ИЛИ графа [7], представленное на рис. 2. Вершинами этого графа являются условия и выводы правил, а дуги задают отношения между ними, объединяя их в правила. Веса дуг соответствуют номерам правил. При этом решается проблема недостатка наглядности, так как с помощью графа удобно проследить условия и вывод каждого правила, а также взаимосвязи в иерархии правил. На рис. 2 показаны два правила, связанные между собой одним утверждением, которое является одновременно выводом правила 2 и условием правила 1.

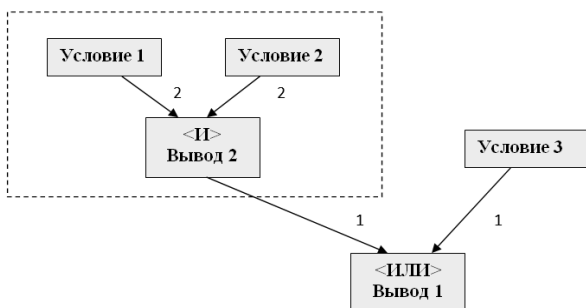


Рис. 2. Схема представления правил с помощью И/ИЛИ графа

Для преобразования И/ИЛИ графа в правила в текстовом виде предложен рекурсивный обход графа, начиная с корня дерева.

Использование графа для представления набора правил обуславливает ряд особенностей. Во-первых, все дуги ориентированы, поскольку выражают связь «условие правила» – «вывод» от условий к выводу, т.е. И/ИЛИ граф является орграфом. Во-вторых, в графе есть как минимум одна вершина, которая является конечной для алгоритма вывода и обозначает вывод из системы правил. Все дуги, инцидентны ей, являются входными. Также И/ИЛИ граф должен иметь набор вершин, которые обозначают исходные условия, которые являются входными для алгоритма вывода. Все инцидентные этим вершинам дуги являются выходными. Другие вершины, которые имеют как входные, так и выходные дуги, являются одновременно и условиями одних правил, и выводами других. При этом следует заметить, что ни одна вершина не может быть одновременно условием и заключением одного правила. Таким образом, при отсутствии циклов И/ИЛИ граф системы правил можно считать деревом (или лесом, в случае наличия нескольких конечных вершин).

В виде И/ИЛИ графа представлен граф для правил, относящихся к базе знаний по неисправностям в сети, который показан на рис. 3.

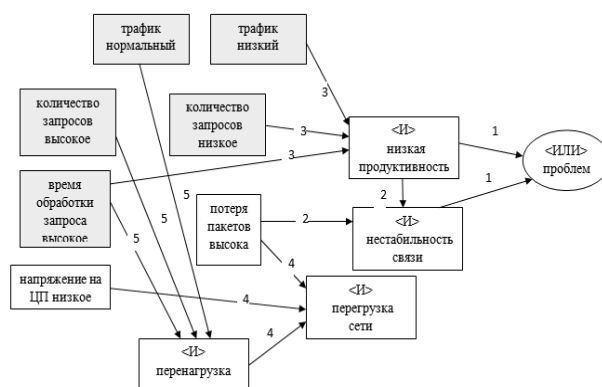


Рис. 3. И/ИЛИ граф для группы правил «Сеть»

## 2. Автоматизация проверки качества баз правил

Существуют различные источники знаний, но люди-эксперты – самый ценный из

них. К сожалению, их знания могут быть субъективными, неполными, противоречивыми либо даже ошибочными, и для устранения этих недостатков приходится проделывать большой объем механической работы, которую можно и нужно автоматизировать – это, в частности, проверки правил на непротиворечивость, отсутствие циклов, полноту и избыточность [9]. Далее в разделе описываются предлагаемые проверки и иллюстрируются на примерах.

### 2.1 Проверка противоречивости базы правил

Один из аспектов, поддающихся автоматизации – проверка противоречивости правил. Противоречие – ситуация, когда истины несколько несовместимых утверждений об одном объекте. В случае правил, в частности, могут быть противоречия между условиями правила, между условием и выводом, а также между правилами.

Такого рода противоречия в [7] было предложено проверять с применением задачи SAT (SATisfiabilityproblem – задача выполнимости булевых формул), так как правила разбивались на группы так, чтобы в каждой группе были правила, имеющие один тот же вывод. Таким образом, можно определить правила, которые никогда не смогут выполняться либо будут выполняться всегда. Оба этих варианта недопустимы и свидетельствуют о противоречиях в системе правил.

На рис. 4 и рис. 5 представлены правила, содержащие противоречия (в текстовом виде и графическом). Рис. 4 состоит из двух частей правил – левого и правого. Каждое правило подаем на вход задачи SAT. Формула для каждого правила означает, что система охлаждения неэффективна. Таким образом, мы определяем, существует ли такой набор значений переменных, при котором эта формула истинна (т.е. система охлаждения неэффективна). В первом случае (левая часть рис. 4), если нет, т.е. задача SAT выдала Ложь, то наша ЭС никогда не скажет, что система неэффективна. А это значит, что такое правило никогда не сработает, и его можно убрать из системы правил.

Справа на рис. 4 изображено отрицание совокупности правил, которое подаем на вход задачи SAT. Инверсная формула означает, что

система охлаждения эффективна. Таким образом мы определяем, есть ли такой набор значений переменных, при котором эта формула истинна (т.е. система охлаждения эффективна.). Если задача SAT выдала Ложь, то нет таких значений переменных, при которых система охлаждения эффективна, т.е. она всегда неэффективна. Это значит, что хоть одно из изначальных правил будут срабатывать всегда, и всегда будет происходить вывод, что система охлаждения неэффективна. Такие правила несовместимы, т.е. противоречивы.

Пример 1	Пример 2
<b>Правило</b>	<b>Правила</b>
Если температура окружающей среды высока и если температура окружающей среды нормальная, то система охлаждения неэффективна.	Если температура окружающей среды высока, то система охлаждения неэффективна.  Если температура окружающей среды нормальная, то система охлаждения неэффективна.
<hr/> а - если температура окружающей среды нормальная <hr/>	<hr/> а - если температура окружающей среды нормальная <hr/>
<b>булева формула:</b> $a \wedge \bar{a} \rightarrow SAT$	<b>«инверсная» булева формула:</b> $a \vee \bar{a} \rightarrow SAT$
<b>всегда = FALSE</b>	<b>всегда = FALSE</b>
Правило никогда не выполняется, убираем из системы	Система правил бессмысленна

Рис. 4. Проверка правил на основе SAT



Рис. 5. Пример наличия противоречивости в правилах

Понятно, что такие противоречия представляются очевидными, однако они могут «затеряться» в правилах, где имеется несколько посылок; а такая проверка осуществляется автоматически.

В [7] таким образом, проверялись базы независимых правил. Такой подход применим и для базы связанных правил, так как база из связанных правил может быть преобразована в базу из одного или нескольких независимых правил, если связанные правила соединить в одно. Например, связанные правила на рис. 2 могут быть приведены к одному правилу. Их изначальное разделение в базе правил могло быть вызвано необходимостью в понятности и простоте описания.

На рис. 5 противоречивое правило 4 представлено в виде И/ИЛИ графа. Такие противоречия с использованием SAT обнаруживаются автоматически.

### 2.2. Поиск циклических зависимостей

Еще одна важная проверка системы правил – поиск циклических зависимостей. Обращая внимание на вершины «перегрев системы», «необходима настройка системы охлаждения» и «необходимо увеличить обороты системы охлаждения» на рис. 6, замечаем, что они находятся в циклической зависимости. Циклические зависимости могут привести к невозможности получения вывода, поэтому они должны быть найдены и устранены.

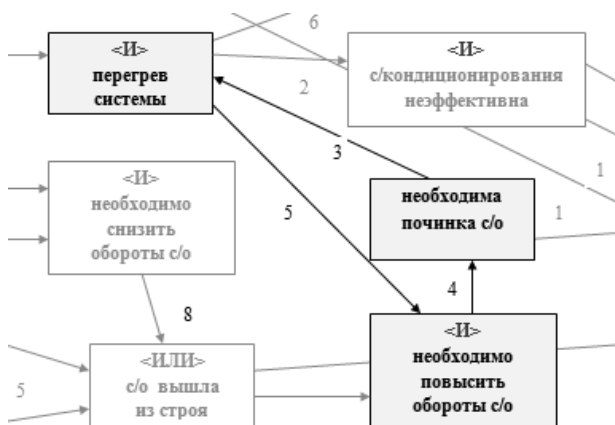


Рис. 6. Пример цикличности в правилах

Подобные проблемы предлагается обнаруживать автоматически. Проверка осуществляется серией поисков в глубину в графе; цикл обнаруживается, если происходит попытка перехода к вершине, которая была посвящена ранее [10]. Асимптотика такого алгоритма  $O(M)$ , где  $M$  – количество ребер.

### 2.3. Проверка полноты

Еще одной важной процедурой контроля качества базы правил является проверка ее полноты. Под полнотой базы правил подразуме-

вается достаточность содержащихся в ней правил для решения задач экспертной системы. Способом проверки полноты, предложенным нами в [7], является автоматическое дополнение набора правил противоположным по смыслу («инверсным») и оценка экспертом правил противоположного набора.

Процедура построения «инверсного» набора правил заключается в следующем:

1) выполняется преобразование И/ИЛИ графа, соответствующего правилам, в булево выражение;

2) берется отрицание этого выражения, которое с помощью тождественных преобразований приводится в ДНФ;

3) по полученной ДНФ выполняется построение И/ИЛИ графа инверсного набора правил.

Рассмотрим проверку на полноту на примере группы правил экспертной системы мониторинга компьютерной сети. В левой части рис. 7 изображен граф исходного набора правил, в правой части – фрагмент графа для инверсного набора. Из рис. 6 правило № 5 гласит, что если трафик не низкий (т.е. высокий) и нагрузка на процессор не низкая (т.е. высокая), то проблемы нет. Очевидно, что это правило некорректно, так как высокие значения трафика и загрузки процессора означают перегрузку узла сети. Наличие такого правила означает, что исходная система правил, возможно, неполная.

В данном случае неполнота заключается в отсутствии правил в исходном наборе для случая, когда нагрузка на процессор высокая.

В [7] был описан предложенный нами способ проверки правил на полноту, здесь приводим формальное доказательство правомочности описанных выше преобразований булевых формул. В приведенных ниже описаниях  $A_i$  обозначают посылки правил, где  $i$  – номер посылки,  $B$  – вывод правил. Длинная горизонтальная черта при формулировке утверждений разделяет то, от чего отталкиваются в утверждении (то, что дано), и то, что нужно доказать.

#### Утверждение 1

$$A_1 \rightarrow B$$

$$A_2 \rightarrow B$$

$$\hline A_1 \vee A_2 \rightarrow B$$

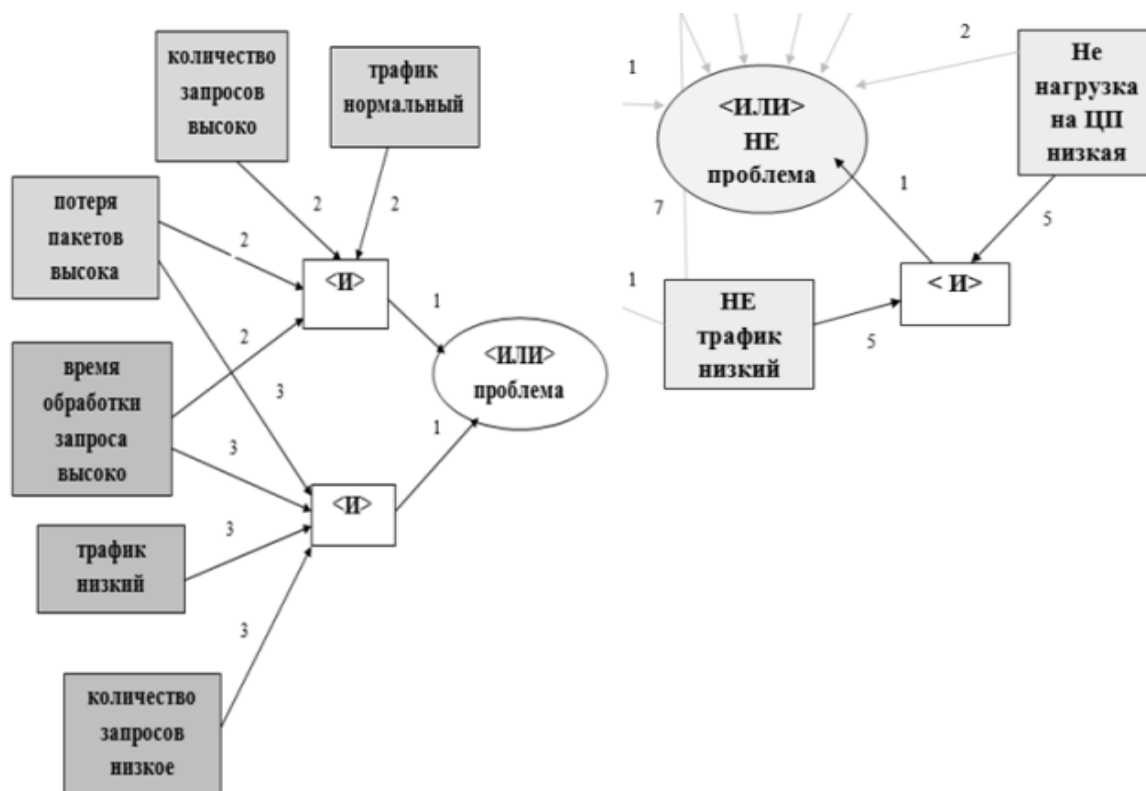


Рис. 7. Пример неполного набора правил

Доказательство:

$$(A_1 \rightarrow B) \wedge (A_2 \rightarrow B) \leftrightarrow (A_1 \vee A_2) \rightarrow B,$$

что и требовалось доказать, т.е. если есть несколько правил в БЗ и вывод у них один и тот же, то из дизъюнкции посылок тоже следует этот же вывод. Таким образом, мы показали, правомочность выполнения п. 1 построения «инверсного» набора правил, т. е. преобразования всего набора правил в булево выражение  $(A_1 \vee A_2 \vee \dots \vee A_i) \rightarrow B$ , где  $A_i$  – логическое выражение, соответствующее посылкам правила  $i$ .

Утверждение 2

$$A_1 \rightarrow B$$

$$A_2 \rightarrow B$$

$$\overline{A_1 \wedge A_2} \rightarrow \overline{B}$$

$$\overline{(A_1 \vee A_2 \leftrightarrow B) \vee (A_1 \vee A_2 \leftrightarrow \overline{B})}$$

Доказательство:

$\overline{A_1 \wedge A_2} = \overline{A_1} \vee \overline{A_2}$  по закону де Моргана, так как по условию  $\overline{A_1 \wedge A_2} \rightarrow \overline{B}$ , то и  $\overline{A_1 \vee A_2} \rightarrow \overline{B}$ , отсюда  $B \rightarrow A_1 \vee A_2$  Из утвер-

ждения 1 следует  $A_1 \vee A_2 \rightarrow B$ . Значит,  $A_1 \vee A_2 \leftrightarrow B$ . Из эквивалентности следует и  $A_1 \vee A_2 \leftrightarrow \overline{B}$ . Что и требовалось доказать.

Мы доказали более сильное утверждение: если есть несколько правил в базе знаний, и вывод у них один тот же, а также, если из того, что все правила неверны, следует, что и общий вывод неверен, то дизъюнкция посылок эквивалентна выводу. Т.е. из дизъюнкции посылок следует этот же вывод, а также из вывода следует дизъюнкция посылок. Это же верно и для отрицаний.

Такое утверждение нужно для того, чтобы дальше можно было ставить отрицание над формулой и получать «инверсные» правила (п. 2 процедуры построения «инверсного» набора правил).

Теперь докажем правомочность следующего вывода.

Утверждение 3

$$\overline{A_1 \vee A_2} \leftrightarrow \overline{B}$$

Приведем левую часть эквивалентности к ДНФ:

$$A_1 \vee A_2 \vee A_3 \leftrightarrow \bar{B}$$

$$A_1 \rightarrow \bar{B}$$

$$A_2 \rightarrow \bar{B}$$

$$A_3 \rightarrow \bar{B}$$

Доказательство:

если  $A_1$  – истина, то  $A_1 \vee A_2 \vee A_3$  – тоже истина. А так как  $A_1 \vee A_2 \vee A_3 \leftrightarrow \bar{B}$ , то  $\bar{B}$  – истина. То есть, из заключения « $A_1$  – истина» следует  $\bar{B}$  – истина.

Аналогично доказывается и для остальных  $A_i$ . Этим мы доказали, что из прямых правил БЗ, у которых один и тот же вывод, правомерно получать так называемые «инверсные» правила, у которых вывод противоположный относительно исходных правил (п. 3 процедуры построения «инверсного» набора правил).

#### 2.4. Проверка избыточности

Еще одна проверка качества базы правил – проверка избыточности. Под избыточностью понимается наличие в базе большего количества правил, чем требуется для решения задачи экспертной системы. Следствием этого может быть снижение производительности при получении вывода либо, если есть правила-дубли, усложнение внесения изменений, а также опасность появления ошибок.

Проверку избыточности предлагается производить путем построения минимального набора правил. Схема процедуры: преобразование И/ИЛИ графа в булево выражение; затем выполняется тождественное преобразование полученного выражения в минимальную дизъюнктивную нормальную форму (МДНФ) с помощью метода Квайна Мак Класки [11]; по МДНФ строится минимизированный набор правил.

Рассмотрим пример проверки на избыточность в системе правил экспертной системы мониторинга компьютерной сети (рис. 8).

В левой части рис. 8 изображен И/ИЛИ граф исходного набора, содержащего пять правил, в правой – минимального, в котором три правила. Здесь видно, что связан-

ные правила объединяются вместе. Таким образом, исходный набор из пяти правил может быть заменен тремя. Эксперт дальше может решать, отдавать ли предпочтение первому набору, где видна детализация проблем, либо в данном случае детализация не важна, и можно сделать только три правила.

#### 3. База правил для мониторинга компьютерной сети

Правила для базы знаний целесообразно разделить на несколько групп, что позволяет облегчить их восприятие, наладку, внесение изменений. Ниже приведены полученные группы.

В группе правил «Климат» собраны правила, касающиеся климатических условий, в которых функционирует сеть и ее узлы, а также средств, обеспечивающих соблюдение необходимых условий. Например, правило из данной группы: если произошел пожар или система охлаждения вышла из строя или система охлаждения неэффективна или система кондиционирования неэффективна или необходимо налаживание системы охлаждения, то имеется проблема, которая требует вмешательства системного администратора.

В группе правил «Узел» собраны правила, касающиеся отдельного узла сети, в частности его состояния и нагрузки на него. Например, правило из данной группы: если система находится в состоянии активного свопинга или произошло зависание системы или система находится в состоянии троттлинга или произошла утечка памяти или произошла перегрузка системы, то имеется проблема, требующая вмешательства системного администратора.

В группе правил «Сеть» собраны правила, которые касаются собственно сети, в частности состояния каналов связи между узлами и нагрузки на них. Например, правило из данной группы: если произошло возгорание или система охлаждения вышла из строя или система охлаждения неэффективна или система кондиционирования неэффективна или необходимо налаживание системы охлаждения, то имеется проблема, требующая вмешательства системного администратора.

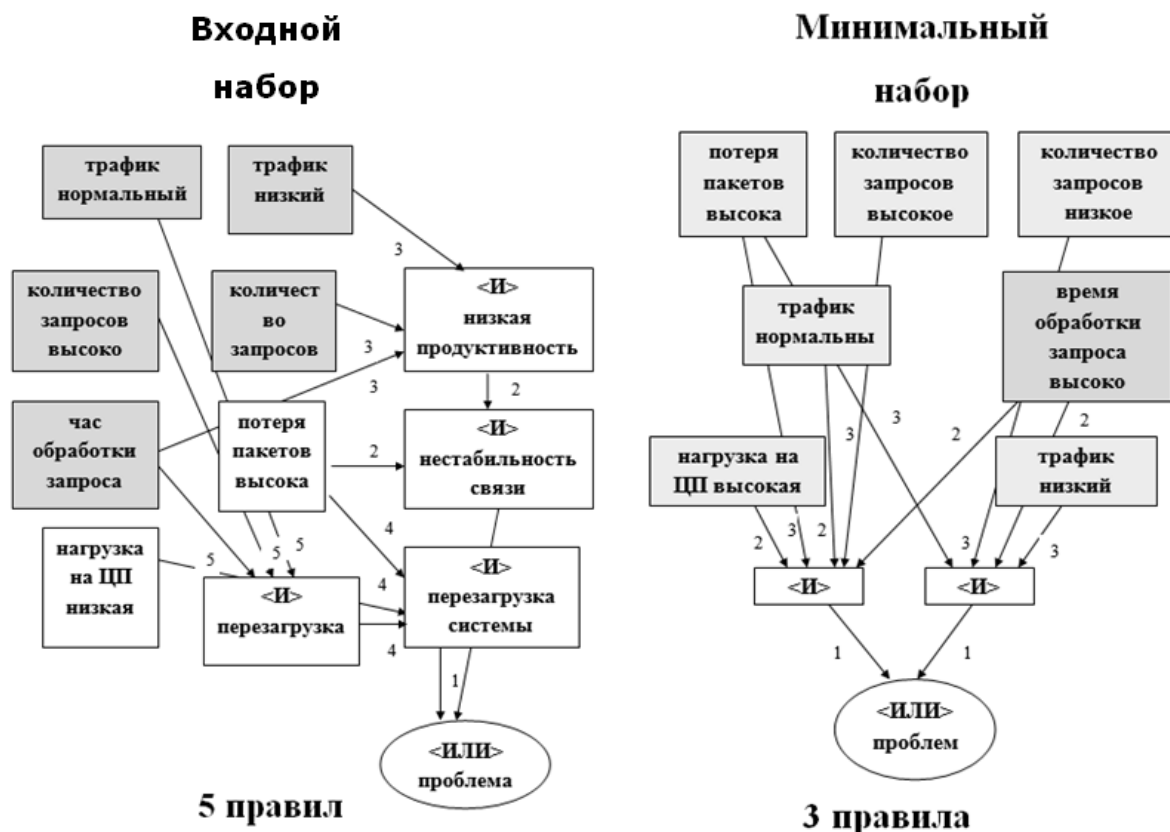


Рис. 8. Пример минимизации набора правил

#### 4. Редактор для создания базы правил

На основе предложенных способов визуализации иерархии правил, а также методов проверки качества базы был создан редактор правил и использован при разработке БЗ для мониторинга компьютерной сети.

Функции системы представлены с помощью диаграммы вариантов использования на рис. 9: редактирование графа, группировка правил, группировка условий, проверка качества знаний, а также сохранение состояния.

В свою очередь в прецеденты, изображенные на рис. 9, входят нижеописанные подгруппы. **Редактирование графа:** создание или редактирование И/ИЛИ графа, открытие существующего И/ИЛИ графа, добавление/редактирование вершин графа, добавление/удаление дуги графа, изменение расположения вершин графа. **Группировка условий:** создание или редактирование группы исходных условий, просмотр группы исходных условий. **Группировка правил:** создание или редактирование группы правил, просмотр группы правил. **Проверка качества знаний:** проверка системы правил на противоречивость, неполноту, избыточ-

ность, цикличность. **Чтение состояния:** чтение состояния существующего графа в текстовом виде, в виде изображения. **Сохранение состояния:** сохранения графа в текстовом виде, сохранения графа в виде изображения.

На рис. 10 изображен скриншот редактора правил, ставшего результатом разработки. В левой части окна находится область редактирования графа, в правой – элементы управления, позволяющие изменять свойства вершины, группы условий, правил, а также поле, в которое выводятся сообщения о событиях (лог). Под областью для редактирования графа находятся кнопки, позволяющие запустить проверки качества базы правил. Основные операции продублированы пунктами в главном меню, а также клавиатурными сокращениями.

Разработка велась на языке Java с использованием библиотеки для работы с графами JUNG 2.0 и библиотеки графического интерфейса пользователя JavaSwing в среде IntelliJ IDEA с использованием Git для контроля версий.





Рис. 9. Функции системы

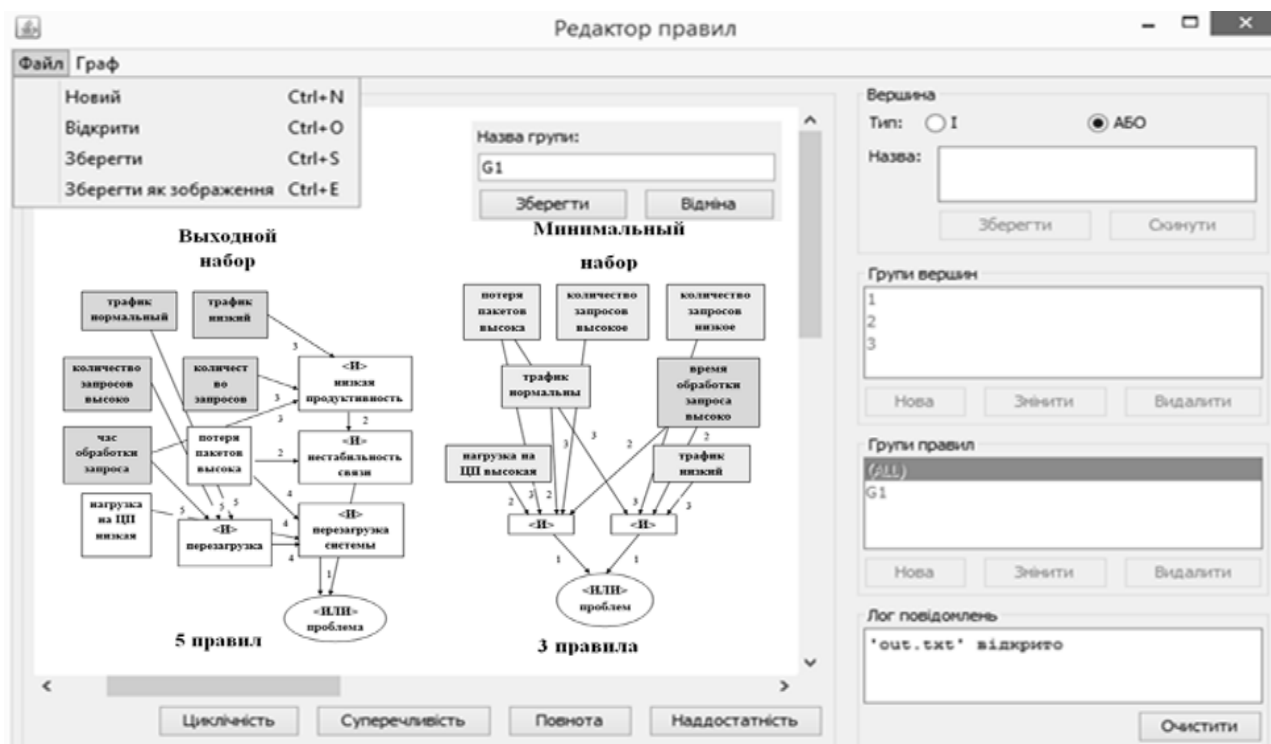


Рис. 10. Редактор правил

Таким образом, предложенные выше способы контроля базы знаний были опробованы при разработке набора правил для мониторинга компьютерной сети. Было показано, что с помощью предложенных методов можно находить ошибки и погрешности, показанные на рисунках 5, 6, 7 и 8.

**Выводы.** В результате выполнения работы были получены следующие результаты.

1) Усовершенствованы существующие формы визуализации правил для мониторинга с использованием многоуровневого И/ИЛИ-графа для визуализации правил ЭС, что позволило представлять связанные правила, характерные для большинства предметных областей.

2) Проведено формальное обоснование предложенных методов контроля качества правил для мониторинга на противоречивость, неполноту, цикличность и избыточность, что доказало их надежность при проверке правил.

3) Создан инструмент для создания, просмотра и редактирования базы правил экспертной системы мониторинга. Таким образом, предложенные выше способы контроля базы знаний, были опробованы при разработке конкретного набора правил и показано, что с помощью предложенных методов возможно и целесообразно находить описанные выше ошибки и неточности.

4) Построена база знаний по неполадкам компьютерной сети, которая может быть использована в экспертной системе мониторинга для анализа состояния сети и выдачи рекомендаций администратору.

Направления дальнейших исследований: анализ эффективности опробованных методов при использовании разработанных с их помощью баз в экспертных системах, интегрированных с существующими системами мониторинга, в частности, мониторинга сетей; усовершенствования и поиск новых способов проверки качества правил.

#### Список использованной литературы

1. Gillam Lee, and Antonopoulos Nick, (2010), *Cloud Computing: Principles, Systems and Applications*, (Computer Communications and Networks), ISBN 9781849962407, London, *Springer*, 379 p.

2. Таненбаум Э. Компьютерные сети / Э. Таненбаум, Д. Уэзеролл; [перевод с англ.]. (Серия: Классика Computer Science; 5-изд.). – Питер : – 2012. – 960 с. (ISBN: 978-5-4461-0068-2).

3. Comparison of Network Monitoring Systems (Электронный ресурс): [Wikipedia] – март 2013 – Режим доступа к статье: [http://en.wikipedia.org/wiki/Comparison\\_of\\_network\\_monitoring\\_systems](http://en.wikipedia.org/wiki/Comparison_of_network_monitoring_systems). – Название с экрана. (10.04.2014).

4. Суботін С. О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень: [навчальний посібник] / С. О. Суботін – Запоріжжя : ЗНТУ, 2008. – 341 с.

5. Wu C.I., Kung H. Y., Chen C. H., Kuo L.C. An Intelligent Slope Disaster Prediction and Monitoring System Based on WSN and ANP, UK.: Elsevier Limited, (2013), *Expert Systems with Applications*, ISBN 09574174, DOI: 10.1016/j.eswa.2013.12.049.

6. Negnevitsky M. *Artificial Intelligence: A Guide to Intelligent Systems*, (2005), New York, *Addison Wesley*, 504 p.

7. Тройніна А. С. Редактор знань для експертних систем моніторингу / А. С. Тройніна, В. М. Рувінська, М. С. Ніколенко // Вестник Херсонского национального технического университета (ХНТУ). – 2013. – № 1(46). – С. 183 – 185.

8. Гаврилова Т. А. Базы знаний интеллектуальных систем / Т. А. Гаврилова, В. Ф. Хорошевский. – СПб. : Питер, 2005. – 384 с.

9. Russell S.J., and Norvig P. *Artificial Intelligence: A Modern Approach*, (2003), *Upper Saddle River, NJ: Prentice Hall*, 1079 p.

10. Майника Э. Алгоритмы оптимизации на сетях и графах / Э. Майника; пер. с англ. – М. : Мир, 1981. – 323с., ил.

11. Савельев А. Я. Основы информатики [Текст] / А. Я. Савельев – (Информатика в техническом университете). – М. : Изд-во МГТУ им. Н. Э. Баумана. – 2001. – С. 232 – 239. – ISBN 5703815150.

Получено 06.04.2014

#### References

1. Gillam Lee, and Antonopoulos Nick, (2010), *Cloud Computing: Principles, Systems and Applications* (Computer Communications and Networks), *Springer*, London, 379 p. (In English).

2. Tanenbaum A., and Wetherall D., (2010), *Computer Networks, 5th edition*, *Prentice Hall*, 962 p. (In English).

3. Wikipedia (2013) Comparison of Network Monitoring Systems, Available at: [http://en.wikipedia.org/wiki/Comparison\\_of\\_network\\_monitoring\\_systems](http://en.wikipedia.org/wiki/Comparison_of_network_monitoring_systems) (accessed 10 April 2014) (In English).

4. Subbotin S.O. Podannja j obrobka znan' u systemah shtuchnogo intelektu ta pidtrymky pryjnattja rishen' [Knowledge Presentation and Processing in Systems of Artificial Intelligence and Decision Support], (2008), *Navchal'nyj Posibnyk*, Zaporozhye, Ukraine, ZNTU, 341 p. (In Ukrainian).

5. Wu C.I., Kung H.Y., Chen C.H., and Kuo L.C., (2013), An Intelligent Slope Disaster Prediction and Monitoring System Based on WSN and ANP (*Expert Systems with Applications*), *Elsevier Limited*, ISBN 09574174, UK, DOI: 10.1016/j.eswa.2013.12.049 (In English)

6. Negnevitsky M. (2005), *Artificial Intelligence: A Guide to Intelligent Systems*, *Addison Wesley*, New York., 504 p. (In English)

7. Troynina A.S., Ruvinska V.M., and Nikolenko M.S. Redaktor znan' dljaekspertnyh system monitoryngu [Knowledge Editor for

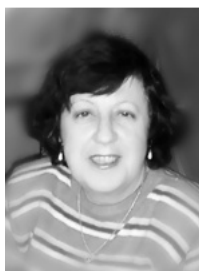
Monitoring Expert Systems], (2013), *VestnikHersonskio Nal'nogotekhnicheskogo Universiteta Publ.*, Kherson, Ukraine, Vol. 1 (46), pp. 183 – 185 (In Ukrainian).

8. Gavrilova T.A., and Horoshevskij V.F. Bazyznaniijintelektual'nyh system [Knowledge Bases for Intelligent Systems], (2005), *Piter-Publ [Piter Russian Publishing House]*, St. Petersburg, Russian Federation, 384 p. (In Russian).

9. Russell S.J., and Norvig P. Artificial Intelligence, (2003). A Modern Approach – Upper Saddle River, *Prentice Hall*, NJ, 1079 p. (In English).

10. Minieka E., (1978), Optimization Algorithms for Networks and Graphs, *Marcel Dekker, Inc.*, New York and Basel, 323 p. (In English).

11. Savel'ev A.Ja. Osnovyinformatiki [Foundations of Computer Science], (2001), *Informatika v Tehnic Heskou Universitete NE Bauman Publ.*, ISBN 5703815150, Moscow, Russian Federation, pp. 232 – 239 (In Russian).



Рувинская Виктория Михайловна; к.т.н., приват-профессор каф. системного программного обеспечения (СПО) Одесского нац. политехнич. ун-та (пр. Шевченко, 1, Одесса, 65044), тел. +380984421842  
E-mail: iolnlen@te.net.ua



Тройнина Анастасия Сергеевна, аспирантка, ст. преподаватель каф. СПО Одесского нац. политехн. ун-та, тел. +380979354470.  
E-mail: anastasiyatroynina@gmail.com



Беркович Евгений Леонидович, к.ф.-м.н., инженер ИКТ «ИТ-инкубатор» Одесского нац. политехн. ун-та.  
E-mail: evg.berkovich@gmail.com



Черненко Андрей Юрьевич, магистр, студент каф. СПО Одесского нац. политехн. ун-та.  
E-mail: andrii.chernenko@gmail.com