УДК 004.89

НЕЦЕЛЕОРИЕНТИРОВАННАЯ СТРАТЕГИЯ ВЫВОДА ФОРМУЛ В МОДАЛЬНЫХ ИСЧИСЛЕНИЯХ

В.Б. Новосельцев, Г.Д. Копаница

Томский политехнический университет
E-mail: vbn@osu.cctpu.edu.ru

*Предлагается и обосновывается подход к анализу формул модального исчисления **KT**, опирающийся на обратный метод Маслова. Предлагаемый подход ориентирован на создание систем автоматического доказательства теорем и предназначен для построения когнитивных систем широкого класса. Приводятся модификации исходного исчисления, для которых доказываются теоремы полноты. Предлагается отношение **Φ**-упорядочения, на основе которого формируются стратегии сокращения пространства вывода.*

Введение

Необходимость исследований в области автоматического доказательства теорем определяется постоянно растущим спросом на интеллектуальные системы программирования и невозможностью (или малой эффективностью) использования существующих информационных технологий для слабо-формализованных предметных областей (ПО), а также предметных областей с модальными связями. Модальные теории (в разных модификациях) естественным образом включают понятия *необходимости*, *возможности* и т. д.

Традиционным инструментом автоматического поиска вывода является метод резолюций, применяемый в *классическом* Прологе [1], а также в таких системах для *модальных* логик как *SAT [2] и DLP [3] – традиционно, такие методы называют целеориентированными или прямыми. Не менее мощным, но существенно реже используемым, является обратный (по отношению к целеориентированным) метод [4], предусматривающий при оценке формулы движение «от аксиом». Наиболее эффективная программа автоматического установления выводимости для модальных систем, основанная на обратном методе, реализована лишь недавно [5]. Успешность применения обратного метода, в значительной степени, обусловлена его настройкой на анализируемую формулу и возможностью применения эффективных критериев борьбы с избыточностью пространства вывода, что способствует сокращению ресурсных характеристик поиска (см. Приложение в [1]). Обратный метод ориентирован на формирование пространства вывода (*а не только самого вывода*) в виде леса. – Очевидно, что навигация по частично-упорядоченным структурам, как минимум, не менее эффективна, чем управление в однородном множестве с эквивалентной суммарной мощностью.

В качестве базового исчисления в статье рассматривается модальная логика знания – система ***KT*** [6]. Логика ***KT*** является более богатой системой по сравнению с минимальной модальной логикой K (за счет добавления аксиомы *T*:?A?A) и интересна тем, что аксиома *T* отвечает свойству рефлексивности бинарного отношения *R структуры* модальной логики [6]. Модальные теории с эффективными стратегиями планирования могут быть успешно использованы в автоматизации проектирования (когда не известны строгие правила вывода для анализируемой ПО), в экспертных и других когнитивных системах. Концепции ряда современных систем автоматизированного проектирования предусматривают совместное использование традиционных технологий и модальных компонентов [7-11]. Специфика баз проектных знаний состоит в том, что они включают знания о ПО и знания о ранее полученных решениях. Иногда на содержательном уровне разделяют два класса проектных знаний (по признакам значимости): *глубинные*, основанные на некоторой фундаментальной теории (например, законах механики Ньютона) и *поверхностные* (*эвристические*), которые базируются на индивидуальном опыте конструкторов. Такая категоризация может быть преобразована (простым переименованием) в категоризацию «*известно/возможно*», которая как раз и характерна для системы ***KT***. – Подчеркнем, что степень субъективности разделения проектных знаний не является предметом данного исследования.

Практический интерес к модальным теориям и методам установления выводимости, отличным от метода резолюций, подтверждается обширной библиографией – здесь упоминается только незначительная часть публикаций.

В первой части статьи вводятся основные определения и базовые исчисления. Понятие *мультимножества* и соответствующая нотация взяты из [12]. В последующих разделах рассматривается отношение **Φ**-упорядочения, его свойства, обеспечивающие сокращение пространства вывода, и теоремы полноты.

1. Базовые исчисления логики KT для обратного метода

Для установления выводимости формулы в ***KT*** предлагается некоторая разрешающая процедура, реализации которой осуществляется в четыре этапа:

I. Построение базовых исчислений для ***KT*** – *прямого* секвенциального исчисления $\boldsymbol{KT}_{seq}$ и настроенного на *анализируемую* формулу **Φ** *обратного* исчисления $\boldsymbol{KT}^{\Phi}_{inv}$ (в последнем случае формулами $\boldsymbol{KT}^{\Phi}_{inv}$ являются все подформулы исходной **Φ**).

II. Введение исчислений *путей* $\boldsymbol{KT}^{\Phi}_{\mathrm{path}}$ и $\boldsymbol{KT}^{\Phi}_{\mathrm{IP}}$, кодирующих вывод (связывающих подформулу с примененным к ней в процессе вывода правилом) в базовых исчислениях $\boldsymbol{KT}_{\mathrm{seq}}$ и $\boldsymbol{KT}^{\Phi}_{\mathrm{inv}}$, соответственно.

III. Замена вывода исходной $\mathbf{\Phi}$ формулы в $\boldsymbol{KT}$ на опровержение отрицания $\mathbf{\Phi}$ в $\boldsymbol{KT}^{\Phi}_{\mathrm{IP}}$ (из соображений технического удобства) с применением стратегий сокращения пространства вывода.

IV. Отображение полученного в $\boldsymbol{KT}^{\Phi}_{\mathrm{IP}}$ вывода в исходную систему.

1.1. Исчисления $\boldsymbol{KT}_{\mathrm{seq}}$ и $\boldsymbol{KT}^{\Phi}_{\mathrm{inv}}$. Пусть $\mathbf{\Phi}$ – формула логики $\boldsymbol{KT}$. Для анализа $\mathbf{\Phi}$ удобнее использовать не саму систему $\boldsymbol{KT}$, а эквивалентное ей исчисление секвенций $\boldsymbol{KT}_{\mathrm{seq}}$ [13]. Справедлива теорема (полноты $\boldsymbol{KT}_{\mathrm{seq}}$): $\mathbf{\Phi}$ невыполнима в $\boldsymbol{KT}$ тогда и только тогда, когда существует опровержение $\mathbf{\Phi}$ в $\boldsymbol{KT}_{\mathrm{seq}}$. Доказательство приведено в [13], откуда заимствовано и подходящая версия исчисления секвенций $\boldsymbol{KT}_{\mathrm{seq}}$ (здесь p – пропозициональная переменная):

Аксиомы: $\mathbf{\Gamma}, p, \sim p$.

Правила вывода:

$$\frac{\mathbf{\Gamma}, A, B}{\mathbf{\Gamma}, A\wedge B}(\wedge);\quad \frac{\mathbf{\Gamma}, A\ \ \mathbf{\Gamma}, B}{\mathbf{\Gamma}, A\vee B}(\vee);\quad \frac{\mathbf{\Gamma}, A}{\Box\mathbf{\Gamma}, \Diamond A, \Delta}(\Diamond);\quad \frac{\mathbf{\Gamma}, A}{\mathbf{\Gamma}, \Box A}(\Box)$$

В рамках обратного метода, поиск опровержения переносится в инверсное исчисление $\boldsymbol{KT}^{\Phi}_{\mathrm{inv}}$ (формулами $\boldsymbol{KT}^{\Phi}_{\mathrm{inv}}$ являются все подформулы исходной $\mathbf{\Phi}$, что и определяет настройку на формулу). Исчисление $\boldsymbol{KT}^{\Phi}_{\mathrm{inv}}$ приводится ниже:

Аксиомы: $p, \sim p$.

Правила вывода:

$$\frac{\mathbf{\Gamma}, A, A}{\mathbf{\Gamma}, A}(\mathrm{C});\quad \frac{\mathbf{\Gamma}, A\ \ \vee, B}{\mathbf{\Gamma}, \vee, A\vee B}(\vee);\quad \frac{\mathbf{\Gamma}, A}{\Gamma, A\wedge B}(\wedge_l);\quad \frac{\mathbf{\Gamma}, B}{\mathbf{\Gamma}, A\wedge B}(\wedge_r);$$

$$\frac{\mathbf{\Gamma}, A}{\Box\mathbf{\Gamma}, \Diamond A}(\Diamond);\quad \frac{\mathbf{\Gamma}}{\Box\mathbf{\Gamma}, \Diamond A}(\Diamond);\quad \frac{\mathbf{\Gamma}, A}{\mathbf{\Gamma}, \Box A}(\Box).$$

Заметим, что в общем случае неясно, как в $\boldsymbol{KT}^{\Phi}_{\mathrm{inv}}$ находить опровержение произвольной секвенции. Для доказательства полноты $\boldsymbol{KT}^{\Phi}_{\mathrm{inv}}$ доказывается *лемма подсеквенциальности*, которая позволяет *переносить* найденное в $\boldsymbol{KT}_{\mathrm{seq}}$ опровержение $\mathbf{\Phi}$ в исчисление $\boldsymbol{KT}^{\Phi}_{\mathrm{inv}}$ (индукцией по длине вывода $\mathbf{\Gamma}$ в $\boldsymbol{KT}_{\mathrm{seq}}$).

Лемма (*подсеквенциальности*)**.** Пусть $\mathbf{\Phi}$ – формула $\boldsymbol{KT}$ и Γ – секвенция, состоящая из подформул $\mathbf{\Phi}$ и имеющая опровержение в $\boldsymbol{KT}_{\mathrm{seq}}$, тогда существует секвенция Δ такая, что $\Delta \dot{\subseteq} \mathbf{\Gamma}$ и Δ имеет опровержение в $\boldsymbol{KT}^{\Phi}_{\mathrm{inv}}$.

Теперь может быть доказана

Теорема (*полноты* $\boldsymbol{KT}^{\Phi}_{\mathrm{inv}}$)**.** Формула $\mathbf{\Phi}$ системы $\boldsymbol{KT}$ является невыполнимой тогда и только тогда, когда она имеет опровержение в $\boldsymbol{KT}^{\Phi}_{\mathrm{inv}}$.

1.2. Исчисление путей $\mathbf{KT}^{\Phi}_{\mathrm{path}}$. Для заданной формулы $\mathbf{\Phi}$ строится исчисление путей $\boldsymbol{KT}^{\Phi}_{\mathrm{path}}$ [14,15]. Поиск вывода в этом исчислении технически прост, а дерево вывода пустого пути в нем представляет каркас доказательства $\mathbf{\Phi}$ в $\boldsymbol{KT}_{\mathrm{seq}}$.

Определение. Пусть C – формула системы $\boldsymbol{KT}_{\mathrm{Seq}}$, C_1 и C_2 _ ее подформулы. Путем в $\mathbf{\Phi}$ или $\mathbf{\Phi}$-путем будем называть любую конечную последовательность символов $\wedge_l$, $\wedge_r$, $\vee_l$, $\vee_r$, $\Box$, $\Diamond$, которая удовлетворяет следующим правилам:

- Пустой путь (элемент) ε есть $\mathbf{\Phi}$-путь.
- Пусть π есть $\mathbf{\Phi}$-путь, тогда:
  - если C имеет вид $C_1\wedge C_2$, то $\pi\wedge_l$ *и* $\pi\wedge_r$ есть $\mathbf{\Phi}$-пути ($\wedge$-путь),
  - если C имеет вид $C_1\vee C_2$, тогда $\pi\vee_l$ и $\pi\vee_r$ есть $\mathbf{\Phi}$-пути ($\vee$-путь),
  - если C имеет вид $\Box C_1$, тогда $\pi\Box$ есть $\mathbf{\Phi}$-путь ($\Box$-путь),
  - если C имеет вид $\Diamond C_1$, тогда $\pi\Diamond$ есть $\mathbf{\Phi}$-путь ($\Diamond$-путь).

Подформула для $\mathbf{\Phi}$ на пути π, обозначаемая $\mathbf{\Phi}|_{\pi}$, определяется традиционно [5]. Исчисление путей $\boldsymbol{KT}^{\Phi}_{\mathrm{path}}$ имеет вид:

Аксиомы: $\mathbf{\Gamma}, \pi_1, \pi_2$.

Правила вывода:

$$\frac{\mathbf{\Gamma}, \pi\wedge_l, \pi\wedge_r}{\mathbf{\Gamma}, \mathbf{\Pi}, \pi}(\wedge);\quad \frac{\mathbf{\Gamma}, \pi\vee_l\ \mathbf{\Gamma}, \pi\vee_r}{\mathbf{\Pi}, \pi}(\vee);\quad \frac{\mathbf{\Pi}\Box, \pi\Diamond}{\mathbf{\Gamma}, \pi}(\Diamond);\quad \frac{\mathbf{\Pi}\Box, \pi\Box}{\mathbf{\Gamma}, \pi}(\Box).$$

Все *пути*, входящие в *секвенции* $\mathbf{\Pi}=\pi_1,\ldots,\pi_n$ и $\mathbf{\Pi}\Box=\pi_1\Box,\ldots,\pi_n\Box$ *являются* $\mathbf{\Phi}$-путями.

Определение. *Образом* секвенции пути или дерева вывода в $\boldsymbol{KT}^{\Phi}_{\mathrm{path}}$ называется дерево вывода, полученное из первоначальных формул заменой каждого пути π на $\mathbf{\Phi}|_{\pi}$.

Для доказательства полноты $\boldsymbol{KT}^{\Phi}_{\mathrm{path}}$ используется соответствующая

Лемма (*бимоделирования для* $\boldsymbol{KT}^{\Phi}_{\mathrm{path}}$). (1) Пусть D – дерево вывода в $\boldsymbol{KT}^{\Phi}_{\mathrm{path}}$, тогда образом D является дерево вывода $\mathbf{\Phi}$ в $\boldsymbol{KT}_{\mathrm{seq}}$. (2) Пусть D' – дерево вывода секвенции $A_1,\ldots,A_n$ в $\boldsymbol{KT}_{\mathrm{seq}}$ и $\pi_1,\ldots,\pi_n$ – такие пути, что $\mathbf{\Phi}|\pi_i=A_i\ \forall i=1,\ldots,n$. Тогда существует дерево D для $\pi_1,\ldots,\pi_n$ в $\boldsymbol{KT}^{\Phi}_{\mathrm{path}}$ такое, что D' является образом дерева D. (3) Пункты (1) и (2) справедливы, если везде «дерево вывода» заменить на «опровержение».

Теорема (*полноты для* $\boldsymbol{KT}^{\Phi}_{\mathrm{path}}$). Формула $\mathbf{\Phi}$ логики $\boldsymbol{KT}$ невыполнима тогда и только тогда, когда пустой путь ε имеет опровержение в $\boldsymbol{KT}^{\Phi}_{\mathrm{path}}$.

2. Обратное исчисление путей KT^{Φ}_{IP}

Определим *обратное* исчисление путей $\boldsymbol{KT}^{\Phi}_{\mathrm{IP}}$ по аналогии с $\boldsymbol{KT}^{\Phi}_{\mathrm{inv}}$.

Пусть $\pi_1,\ldots,\pi_n$ – $\mathbf{\Phi}$-пути. $\mathbf{\Pi}=\pi_1,\ldots,\pi_n$, $\mathbf{\Pi}\Box=\pi_1\Box,\ldots,\pi_n\Box$, и $\mathbf{\Gamma}$ – последовательности путей. Тогда аксиомами $\boldsymbol{KT}^{\Phi}_{\mathrm{IP}}$ являются любые формулы вида: π_1, π_2, где $p=\mathbf{\Phi}|_{\pi 1}$, $\sim p=\mathbf{\Phi}|_{\pi 2}$, для некоторой пропозициональной переменной p.

Правила вывода:

$$\frac{\mathbf{\Gamma}, \pi\wedge_l}{\mathbf{\Gamma}, \pi}(\wedge_l);\quad \frac{\mathbf{\Gamma}, \pi\wedge_r}{\mathbf{\Gamma}, \pi}(\wedge_r);\quad \frac{\mathbf{\Gamma}, \pi\vee_l\ \Delta, \pi\vee_r}{\mathbf{\Gamma}, \Delta, \pi}(\vee);\quad \frac{\mathbf{\Gamma}, \pi, \pi}{\mathbf{\Gamma}, \pi}(\mathrm{C});$$

$$\frac{\mathbf{\Gamma}\Box}{\mathbf{\Gamma}, \pi}(\Diamond+);\quad \frac{\mathbf{\Pi}\Box, \pi\Diamond}{\mathbf{\Pi}, \pi}(\Diamond);\quad \frac{\mathbf{\Pi}\Box, \pi\Box}{\mathbf{\Pi}, \pi}(\Box).$$

В [5] приведены свойства исчисления путей, позволяющие избавиться от некоторых избыточных секвенций в дереве вывода. Рассмотрим свойства, ограничивающие поиск опровержения лишь некоторым подмножеством деревьев вывода с помощью упорядочения на множестве всех **Ф**-путей.

Классический метод резолюций упорядочивает литеры в дизъюнктах и требует, чтобы правило резолюций применялось только тогда, когда наибольшие литералы в обоих дизъюнктах разрешимы. Введем подобные ограничения на построение деревьев вывода для логической системы ***KT***. Преобразуем классическое упорядочивание литер на модальный случай. В случае классического исчисления, возможно использовать любое упорядочение на подформулах **Ф**, которое принимает во внимание префиксное отношение. Непосредственный перенос подобного на модальные системы невозможен, поскольку не каждое упорядочение на путях сохраняет полноту. – Рассмотрим вывод:

$$\frac{\wedge_l\Box,\ \wedge_r\wedge_l\Box\vee_l,\ \wedge_r\wedge_r\wedge_l\Diamond \quad \wedge_l\Box,\ \wedge_r\wedge_l\Box\vee_r,\ \wedge_r\wedge_r\wedge_l\Diamond}{\wedge_l\Box,\ \wedge_r\wedge_l\Box,\ \wedge_r\wedge_r\wedge_l\Diamond}(\vee).$$

Существенно то, что любой $(\vee)$-вывод применяется выше правила $(\Diamond)$, потому что правило $(\Diamond)$ не применимо к верхним секвенциям. Тем не менее, если определено упорядочение на путях, где $\wedge_r\wedge_l\Box\vee_l$ является наименьшим в первой предпосылке, то возможность применения $(\vee)$ первым будет исключена, и доказательство не будет найдено. Отсюда следует, что определение **Ф**-упорядочения в модальных логиках является более сложным, чем в классических. – Определим упорядочение строго.

Определение. Пути назовем братьями, если один имеет вид $\pi\wedge_l$, а другой – $\pi\wedge_r$, либо $\pi\vee_r$ и $\pi\vee_l$.

Так, братьями являются пути $\wedge_r\wedge_l\Box\vee_l$ и $\wedge_r\wedge_l\Box\vee_r$ из рассмотренного выше вывода. Таким образом, каждая конъюнкция или дизъюнкция обуславливает пару братьев.

Обозначения. Везде ниже символом $\wedge\!\!\!\vee$ будем обозначать любой из символов $\wedge$ или $\vee$; символом * – любой из символов r или l; символом $\Box\!\!\!\Diamond$ любой из символов $\Box$ или $\Diamond$. Запись $\pi'\sqsubseteq\pi$ обозначает «π' есть префикс π».

Определение. Для заданной формулы **Ф** назовем **Ф**-упорядочением любое отношение полного порядка $\succ$ на множестве всех **Ф**-путей, удовлетворяющее условиям:

1. $\pi_1\succ\pi_2$, всякий раз когда:
   a) модальная длина π_1 строго больше модальной длины π_2, или
   b) π_1 и π_2 имеют одинаковую модальную длину, последний символ π_1 – $\wedge\!\!\!\vee_*$, а последний символ π_2 – $\Box\!\!\!\Diamond$, или
   c) π_1 и π_2 имеют одинаковую модальную длину и $\pi_2(\pi_1$ или
   d) если π_1 и π_2 имеют одинаковую модальную длину и последний символ π_1 – $\wedge\!\!\!\vee_*$, последний символ π_2 – $\wedge\!\!\!\vee_*$, при этом неверны оба утверждения $\pi_2\sqsubseteq\pi_1$ и $\pi_1\sqsubseteq\pi_2$, но π_1 имеет большую обычную длину, чем π_2.
2. Не существует пути между двумя братьями, то есть не существует **Ф-**путей π_1, π_2, π_3 таких, что $\pi_1\succ\pi_2\succ\pi_3$ и π_1, π_3 – братья.

Содержательно, отношение (позволяет управлять порядком применения правил – сначала правила применяются к формулам, большим относительно (. Помимо этого отношение требует, чтобы заключение любого правила было меньше, чем любая его предпосылка в мультимножественном упорядочении. Условие (1a) гарантирует, что заключение меньше посылки при применении правил $(\Diamond)$ или $(\Diamond+)$. Условие (1b) введено для того, чтобы применение $(\Diamond)$ или $(\Diamond+)$ к секвенции, содержащей путь $\pi\wedge\!\!\!\vee_*$ не дало неполное исчисление. Условия (1c-d) и (2) являются не только техническими и служат для облегчения доказательств утверждений этого параграфа, но и однозначно определяют любые два пути по отношению к порядку $\succ$, что важно в плане реализации.

Обозначение. Будем использовать запись $\pi_1\succeq\pi_2$, если $\pi_1\succ\pi_2$ или $\pi_1=\pi_2$.

Для классической логики полнота метода резолюций с упорядочением доказывается чисто семантически. В случае модальной системы ***KT*** необходимо показать, что стратегия выбора наибольшей формулы (или пути) в дизъюнкте не конфликтует с критериями избыточности, рассмотренными ранее. Поэтому доказательство полноты будем проводить в два этапа. На первом этапе докажем свойства деревьев вывода в $\boldsymbol{KT}^{\Phi}_{\text{path}}$, а на втором этапе перенесем их в обратное исчисление $\boldsymbol{KT}^{\Phi}_{\text{IP}}$, используя соответствующий вариант *леммы подсеквенциальности.*

При доказательстве полноты (в отличие от [14]) появляются технические трудности, связанные с тем, что требование упорядочения формулируется в терминах предпосылок вывода, в то время как доказательство полноты для $\boldsymbol{KT}^{\Phi}_{\text{path}}$ отталкивается от следствий. Это приведет к небольшому усложнению определения упорядочения на путях. Покажем, наконец, что **Ф**-упорядочение существует для любой формулы, а затем приведем алгоритм, который упорядочивает **Ф**-пути.

Алгоритм, работает с секвенциями из множества путей, эти секвенции обозначаются $S_n\succ S_{n-1}\succ\ldots\succ S_0$. Содержательно запись означает, что для любого $\pi\in S_i$ и $\pi'\in S_{i-1}$ выполнено $\pi\succ\pi'$. Пути, принадлежащие одинаковым S_i еще не упорядочены, но будут упорядочены позднее. На каждом шаге будем выбирать некоторое множество S_i в секвенции, содержащее один или более членов и заменять S_i двумя или более множествами $S_{i1}\succ\ldots\succ S_{ik}$ такими, что $S_{i1}\cup\ldots\cup S_{ik}=S_i$. Алгоритм заканчивает работу тогда, когда каждое множество содержит только один элемент.

Алгоритм**[?] **упорядочения.

1. *Первоначально S_i есть множество путей в формуле **Ф** модальной длины i.*

2. *Для всех S_i, исключая последнее множество, выполняем*:
   2.1. *выбираем все пути $\pi_1,...,\pi_n$ в S_i заканчивающиеся* ◊;
   2.2. *разбиваем S_i на $S_i\setminus\{\pi_1,...,\pi_n\}\succ\{\pi_1\}\succ...\succ\{\pi_n\}$*;
   2.3. *разбиваем S_0 на $S_0\setminus\{\varepsilon\}\succ\{\varepsilon\}$*;
3. *Пока существуют S_i с более чем одним членом, выполняем*
   3.1. *выбираем $\pi\varkappa_l$* и *$\pi\varkappa_r$ – два брата в S_i такие, что $\pi\notin S_i$*;
   3.3. *выбираем L и R – множества всех префиксов соответственно из $\pi\varkappa_l$ и $\pi\varkappa_r$*;
   3.4. *разбиваем S_i на $S_i\setminus(L\cup R)\succ R\succ L\succ\{\pi\varkappa_r\}\succ\{\pi\varkappa_l\}$*.

Замечание. Некоторые множества, например, L или R, могут быть пустыми, в этом случае они не включаются в секвенцию.

Когда *алгоритм*→ завершится, секвенция состоит из одноэлементных множеств, в этом случае мы допускаем, что $\pi\succeq\pi'$, если секвенция имеет вид $...\{\pi\}\succ...\succ\{\pi'\}...$.

Следующая лемма гарантирует, что *алгоритм*? удовлетворяет определению **Ф**-упорядочения.

Лемма. Любое упорядочение, полученное *алгоритмом*→ на формуле **Ф** является **Ф**-упорядочением.

Поскольку на любом шаге *алгоритма*→ мы получаем **Ф**-упорядочение, имеет место очевидное *следствие*: ***Ф****-упорядочение существует.*

Понятие **Ф**-упорядочения введено для того, чтобы доказать существование опровержения в специальной форме, связанной с этим упорядочением. **Ф**-упорядочение сокращает пространство поиска вывода только таких опровержений. Для дальнейшего понадобится ряд дополнительных определений.

Обозначение. Пусть π – путь, $\mathbf{\Gamma}$ – секвенция путей. Запись $\pi\succ\mathbf{\Gamma}$ является сокращением для утверждения, что $\pi\succ\pi'$ для любого π' из $\mathbf{\Gamma}$.

Определение. $(\vee)$-вывод в $\boldsymbol{KT}^{\Phi}_{\text{path}}$:

$$\frac{\Gamma,\pi\vee_l,\ \pi\vee_r}{\Gamma,\pi}\ (\vee)$$

будем называть относящимся к **Ф**-упорядочению $\succ$, если $\pi\vee_l\succ\mathbf{\Gamma}$ и $\pi\vee_r\succ\mathbf{\Gamma}$.

Аналогично вводится понятие $(\wedge)$-вывода, относящегося к $\succ$.

Определение. $(\wedge)$-вывод *в* $\boldsymbol{KT}^{\Phi}_{\text{path}}$:

$$\frac{\Gamma,\pi\wedge_l,\ \pi\wedge_r}{\Gamma,\pi}\ (\wedge)$$

будем называть относящимся к **Ф**-упорядочению $\succ$, если $\pi\wedge_l\succ\mathbf{\Gamma}$ *и* $\pi\wedge_r\succ\mathbf{\Gamma}$.

Определение. Дерево вывода в $\boldsymbol{KT}^{\Phi}_{\text{path}}$ будем называть относящимся *к* $\succ$, если каждый $(\wedge)$ и каждый $(\vee)$ вывод из этого дерева относится к $\succ$.

Лемма (*о выводе, относящемся к* $\succ$). Пусть **Ф** – невыполнимая формула и $\succ$ – **Ф**-упорядочение. Тогда существует опровержение в $\boldsymbol{KT}^{\Phi}_{\text{path}}$, которое относится к $\succ$.

Непосредственно не очевидно, как доказать это утверждение. Прямое применение индукции по длине пути без учета дополнительных соображений может привести к получению вывода, не относящегося к **Ф**-упорядочению (. Приведем пример подобной ситуации. Пусть, что $\pi_1\succ\pi_2\vee_l$. Рассмотрим секвенцию π_1, π_2. Мы можем применить правило $(\vee)$ из $\boldsymbol{KT}^{\Phi}_{\text{path}}$ для того, чтобы получить эту секвенцию:

$$\frac{\pi_1,\pi_2\vee_l\qquad \pi_1,\pi_2\vee_r}{\pi_1,\pi_2}\ (\vee).$$

Фактически, этот вывод является единственным, и, тем не менее, он не относится к $\succ$. Условия, наложенные на вывод для того, чтобы он относился к $\succ$, сформулированы в терминах посылок вывода, но в таком индуктивном доказательстве мы можем расширить заключение только новым выводом. Для разрешения этой проблемы следует избавиться от секвенций путей, которые приводят к описанной ситуации, ограничиваясь секвенциями специального вида. Это так называемые $\succ$-компактные секвенции путей. Они определяются следующим образом.

Определение. Пусть $\pi_1,...,\pi_n$ _ пути в формуле **Ф** и $\succ$ – **Ф**-упорядочение. Секвенция путей $\mathbf{\Gamma}=\pi_1,...,\pi_n$ называется $\succ$-компактом, если для каждого $i=1,...,n$ выполняются следующие условия: (1) если π_i – $\wedge$-путь, тогда $\pi_i\wedge_l\succ\mathbf{\Gamma}$ и $\pi_i\wedge_r\succ\mathbf{\Gamma}$; (2) если π_i – $\vee$-путь, тогда $\pi_i\vee_l\succ\mathbf{\Gamma}$ и $\pi_i\vee_r\succ\mathbf{\Gamma}$.

Используя обозначения для дизъюнкции, конъюнкции и модальностей, можно переформулировать определение $\succ$-компактности следующим образом.

Определение*. Пусть $\pi_1,...,\pi_n$ – пути в формуле **Ф** и $\succ$ – **Ф**-упорядочение. Секвенция путей $\mathbf{\Gamma}=\pi_1,...,\pi_n$ *н*азывается $\succ$-компактом, если для каждого $i=1,...,n$ и для каждого $\varkappa$-пути π_i в $\mathbf{\Gamma}$ мы имеем $\pi_i\varkappa_*\succ\mathbf{\Gamma}$.

Заметим, что $\pi_i\varkappa_*\succ\pi_i$ гарантируется условиями **Ф**-упорядочения, следовательно требование $\pi_i\varkappa_*\succ\mathbf{\Gamma}$ можно заменить на $\pi_i\varkappa_*\succ\mathbf{\Gamma}\setminus\{\pi_i\}$. Следующая лемма утверждает, что компактная секвенция $\mathbf{\Gamma}$ не может привести к безвыходной ситуации.

Лемма (*о $\succ$-компактах*). Пусть $\succ$ – **Ф**-упорядочение, тогда (1) посылка каждого (◊)-вывода есть $\succ$-компакт; (2) если $\mathbf{\Gamma}$ – $\succ$-компактная секвенция, встречающая в дереве вывода ε, то каждый $(\varkappa)$-вывод, имеющий $\mathbf{\Gamma}$ своим заключением, относится к (; (3) если $\mathbf{\Gamma}$ – $\succ$-компактная секвенция, встречающая в дереве вывода ε и если $\mathbf{\Gamma}$ содержит по крайней мере один $\vee$-путь или $\wedge$-путь, то существует $(\varkappa)$-вывод, все посылки которого – $\succ$-компакты.

Опираясь на доказанную лемму, можно доказать сформулированную ранее *лемму о существовании вывода, относящегося к* $\succ$.

Лемма. Если **Ф** – невыполнимая формула и $\succ$ – **Ф**-упорядочение, то существует опровержение в $\boldsymbol{KT}^{\Phi}_{\text{path}}$, относящееся к $\succ$.

3. Теорема полноты обратного метода без секвенций

Модифицируем исчисление $\boldsymbol{KT}^{\Phi,*}_{\mathrm{IP}}$ с учетом критерия избыточности.

Определение. Обозначим через $\boldsymbol{KT}^{\Phi,\succ}_{\mathrm{IP}}$ исчисление, полученное из $\boldsymbol{KT}^{\Phi}_{\mathrm{IP}}$ удалением всех:

- секвенций путей, содержащих пути различной модальной длины;
- секвенций путей содержащих противоречивую пару путей;
- выводов, не относящихся к $\succ$.

В $\boldsymbol{KT}^{\Phi,\succ}_{\mathrm{IP}}$ справедливы *леммы бимоделирования* и *подсеквенциальности.*

Лемма (*бимоделирования для* $\boldsymbol{KT}^{\Phi,\succ}_{\mathrm{IP}}$). (1) Пусть D – дерево вывода в $\boldsymbol{KT}^{\Phi,\succ}_{\mathrm{IP}}$. Тогда образ D является деревом вывода $\boldsymbol{\Phi}$ в $\boldsymbol{KT}^{\Phi}_{\mathrm{path}}$. (2) Пусть D' – дерево вывода секвенции $A_1,\ldots,A_n$ в $\boldsymbol{KT}^{\Phi}_{\mathrm{path}}$ и $\pi_1,\ldots,\pi_n$ – такие пути, что $\boldsymbol{\Phi}|\pi_i=A_i$, $\forall i=1,\ldots,n$. Тогда существует дерево вывода D для $\pi_1,\ldots,\pi_n$ в $\boldsymbol{KT}^{\Phi,\succ}_{\mathrm{path}}$ такое, что D' является образом формул дерева вывода D. (3) Пункты (1) и (2) справедливы, если везде «дерево вывода» заменить на «опровержение».

Заметим, что исчисление $\boldsymbol{KT}^{\Phi}_{\mathrm{IP}}$ имеет различные правила для обработки конъюнкции. Поэтому мы изменяем определение дерева вывода, относящегося к $\boldsymbol{\Phi}$-упорядочению следующим образом.

Определение. Дерево вывода в $\boldsymbol{KT}^{\Phi}_{\mathrm{IP}}$ называется относящимся к $\boldsymbol{\Phi}$-упорядочению, если на вывод наложены следующие условия*:*

a) для каждого ($\vee$)-вывода этого дерева

$$\frac{\Gamma,\pi\vee_l\ \ \Delta,\pi\vee_r}{\Gamma,\Delta,\pi}\ (\vee)$$

справедливо только, если $\pi\vee_l\succ\boldsymbol{\Gamma}$ и $\pi\vee_r\succ\boldsymbol{\Gamma}$;

b) для каждого ($\wedge_l$)-вывода (соответственно, ($\wedge_r$)-вывода) исходного дерева

$$\frac{\Gamma,\pi\wedge_l}{\Gamma,\pi}\ (\wedge_l)\qquad \frac{\Gamma,\pi\wedge_r}{\Gamma,\pi}\ (\wedge_r)$$

справедливо $\pi\wedge_l\succ\boldsymbol{\Gamma}$ ($\pi\wedge_r\succ\Gamma$).

Лемма (*подсеквенциальности для* $\boldsymbol{KT}^{\Phi,\succ}_{\mathrm{IP}}$). Пусть D – опровержение ε в $\boldsymbol{KT}^{\Phi}_{\mathrm{IP}}$, относящееся к $\succ$ и I – вывод в D в виде:

$$\frac{\Gamma_1\ldots\Gamma_n}{\Gamma}$$

и пусть даны секвенции путей $\Delta_1,\ldots,\Delta_n$ такие, что каждая Δ_i есть подсеквенция Γ_i, тогда существует дерево вывода D' для Δ в $\boldsymbol{KT}^{\Phi,\succ}_{\mathrm{IP}}$ из $\Delta_1,\ldots,\Delta_n$ такое, что Δ – подсеквенция Γ.

Лемма подсеквенциальности *обобщается* для произвольного дерева вывода. Для этого необходимо применить метод математической индукции по длине дерева вывода.

Лемма (*подсеквенциальности для деревьев вывода в* $\boldsymbol{KT}^{\Phi,\succ}_{\mathrm{IP}}$). Пусть D – опровержение ε в $\boldsymbol{KT}^{\Phi,\succ}_{\mathrm{IP}}$, которое относится к $\succ$, D'' – поддерево вывода в D секвенции $\boldsymbol{\Gamma}$ из секвенций $\boldsymbol{\Gamma}_1,\ldots,\boldsymbol{\Gamma}_n$ и $\Delta_1,\ldots,\Delta_n$ – подсеквенции $\boldsymbol{\Gamma}_1,\ldots,\boldsymbol{\Gamma}_n$ соответственно. Тогда существует дерево вывода D' для секвенции Δ в $\boldsymbol{KT}^{\Phi,\succ}_{\mathrm{IP}}$ из $\Delta_1,\ldots,\Delta_n$ такое, что Δ является подсеквенцией $\boldsymbol{\Gamma}$.

Перейдем к доказательству теоремы полноты для $\boldsymbol{KT}^{\Phi,\succ}_{\mathrm{IP}}$.

Теорема (*полноты* $\boldsymbol{KT}^{\Phi,\succ}_{\mathrm{IP}}$) Формула $\boldsymbol{\Phi}$ системы КТ невыполнима тогда и только тогда, когда ε имеет опровержение в исчислении $\boldsymbol{KT}^{\Phi,\succ}_{\mathrm{IP}}$.

Заключение

Предложенный формализм может послужить основой организации машин логического вывода в системах управления знаниями, где предполагается использование «персонифицированности» правил поведения системы (наличие экспертов с различными подходами к формализации конкретной предметной области). Эффективность вывода в предлагаемом подходе, в общем случае, не хуже, чем для классического метода резолюций, в то время как описательные возможности модальных теорий существенно шире классической логики. Следует также отметить, что предложенный подход исключает использование внелогических механизмов (подобных оператору усечения в Прологе) и в большинстве практических ситуаций требует меньших, нежели метод резолюций, ресурсов.

В настоящее время ведется работа по практической реализации машины вывода с целью возможного использования в ряде проектов по менеджменту знаний.

СПИСОК ЛИТЕРАТУРЫ

1. Чень Ч., Ли Р. Математическая логика и автоматическое доказательство теорем. – М.: Наука, 1983. – 386 с.
2. Giuchinglia F. a. o. SAT-based decision procedures for classical modal logics // Journal of automated reasoning. – 1999. – V. 854. – P. 56–78.
3. Horrocks I., Patel-Schneider P. FACT and DLP. Automated Reasoning with Analytic Tableaux and Related methods // Intern. Conf. TABLEAUX'98. – H. de Swart Eds. Lecture Notes in Artificial Intelligence. – 1998. – V. 1397. – P. 27–30.
4. Маслов С.Ю. О поиске вывода в исчисленьях общего типа // Записки ЛОМИ АН СССР. – 1972. – Т. 32. – № 17. – С. 117–121.
5. Voronkov A. Theorem proving in non-standard logics based on the inverse method // 11th Intern. Conf. on Automated Deduction. – 1992. – № 607. – P. 648–662.
6. Фейс Р. Модальная логика. – М.: Наука., 1974. – 481 с.
7. Тарасов В.Б. Интеллектуальные системы в проектировании // Новости искусственного интеллекта. – 1993. – № 4. – С. 17–25.
8. Yoshikawa H. General Design Theory as a Formal Theory of Design // Intelligent CAD I – 1989. – V. 170 – P. 51–61.
9. Treur J.A. A Logical Framework for Design Processes // Intelligent CAD Systems III: Practical Experience and Evaluation. – 1991. – P. 3–19.
10. Кашуба Л.А. Параллельное проектирование средствами CAD, CAM, CAE в жизненном цикле изделий машиностроения // Программные продукты и системы. – 1998. – № 3. – С. 63–89.

11. Смирнов А.В., Юсупов Р.М. Совмещенное проектирование: необходимость, проблемы внедрения. – СПб.: СПИИРАН, 1992. – 439 с.
12. Минц Г.Е. Резолютивные исчисления для неклассических логик // 9-й Советский Кибернетический симпозиум. – 1981. – Т. 2. – № 4. – С. 34–36.
13. Fitting M. Proof methods for modal and intuicionistic logics // Synthesis Library. – 1983. – V. 169. – P. 56–78.
14. Voronkov A. How to optimize proof-search in modal logics: new methods of proving redundancy criteria for sequent calculi // ACM Transactions and Computational logic. – 2001. – V. 1. – P. 1–35.
15. Новосельцев В.Б., Бурлуцкий В.В. Реализация обратного метода для модальной логики ***KT***. – Томск: ТГУ, 2001. – 147 с.

УДК 004.89

СТРАТЕГИЯ УСТАНОВЛЕНИЯ ВЫВОДИМОСТИ ФОРМУЛ В СТРУКТУРНЫХ ФУНКЦИОНАЛЬНЫХ МОДЕЛЯХ

Д.А. Коваленко, В.Б. Новосельцев

Томский политехнический университет
E-mail: msmaklaud@gmail.com

Рассматривается исчисление рекурсивных предложений для теории структурных функциональных моделей. Исследуются вопросы разрешимости и полноты исчисления. Предлагаются стратегия и алгоритм установления выводимости формул исчисления, показывается корректность алгоритма и определяется оценка его эффективности.

Введение

Рассматривается формальное исчисление, используемое для установления выводимости предложений в рамках теории структурных (С-) функциональных моделей [1, 2]. Проведенные исследования традиционно относят к области *искусственного интеллекта*. Прикладной стороной исследования является создание интеллектуальных программных комплексов широкого класса, опирающихся на дедуктивный подход в области менеджмента знаний. Термин «знание» здесь имеет ограниченную трактовку: *знания* представляются специальными строго определенными информационными структурами, допускающими автоматизированную обработку с использованием аппарата формальной логики. Представляемые результаты могут быть использованы при реализации различных информационных комплексов, – от экспертных [3] и когнитивных систем [4] до инструментальных и CASE-оболочек быстрого прототипирования программ [5].

В первом разделе определяются используемые понятия и необходимые соглашения. Во втором вводится понятие интерпретации С-модели и формулируется исчисление функциональных предложений для теории структурных моделей [6], а также исследуются некоторые свойства построенного исчисления. Третий раздел посвящен описанию стратегии и базового алгоритма установления выводимости предложений языка. Для алгоритма доказывается свойство корректности, и устанавливаются сложностные характеристики. Наконец, в последнем разделе кратко рассмотрены особенности введения рекурсии.

1. Базовые определения

Дадим основные определения. Прежде всего, зафиксируем сигнатуру Σ.

Определение 1. Зафиксируем $\Sigma=(\mathrm{A,F,P,D})$, где A, F, P и D – не более чем счетные множества (элементарных) имен объектов, функциональных символов, селекторных символов и имен схем, соответственно. Выделим в D непустое конечное подмножество $\mathrm{E}\subset\mathrm{D}$ имен *примитивных* или *первичных* схем.

Элементы множества A используются для формирования (*имен*) *объектов*. Связь объекта a с некоторой схемой S отражается записью $S(a)$. Если объект a связан со схемой S и $S\in\mathrm{E}$, стандартная запись $S(a)$ заменяется записью a^S либо a, когда ссылка на схему не важна или очевидна из контекста.

Определение 2. Выражение вида

$$f{:}a_1,\dots,a_n\rightarrow a_0, \qquad (1)$$

где $a_i\in\mathrm{A}$, $i=0,\dots,n$ – имена объектов, называется функциональной связью (ФС). В записи (1) $f\in\mathrm{F}$ – это имя ФС, a_i – ее аргументы ($i = 1,\dots,n$), a_0 – результат.

Имена объектов при моделировании прикладной предметной области (ПО) формируются следующим образом.

Определение 3. Пусть $a\in\mathrm{A},\sigma$ – имя объекта, тогда a, $\sigma.a$ и $a.\sigma$ также являются именами. В записи $\sigma.a$ σ называется префиксом. *Длина* имени σ определяется числом вхождений элементов A (с учетом возможных повторений).

Функциональные связи используются при задании (непервичных) схем объектов моделируемой ПО.