В. П. Мухачев, А. С. Полушин

Индексный метод

В современной логике существует несколько типов исчислений: аксиоматические, натуральные, секвенциальные, натурально-секвенциальные, табличные исчисления, а также нормальные формы. Каждый из перечисленных типов исчислений имеет как определенные преимущества, так и недостатки. Например, натуральные исчисления позволяют вполне адекватно моделировать ход естественных рассуждений, но не позволяют для произвольно взятой формулы определять за конечное число шагов, выводима она или нет, то есть *натуральные исчисления не могут использоваться в качестве разрешающих процедур*.

Секвенциальные исчисления типа генценовского LK проигрывают натуральным в естественности вывода, но обладают рядом важных в плане поиска вывода свойств, в частности, *свойством подформульности* (оно заключается в следующем: если некоторая формула появляется в выводе, не содержащем сечения, то она далее уже не исчезает, а записывается на каждом последующем шаге и входит в конечном счете в состав заключительной секвенции) и *свойством обратимости правил вывода*, позволяющим строить вывод снизу вверх, от заключительной секвенции к самым верхним. Если сравнивать исчисления с точки зрения поиска вывода, то наиболее эффективными являются именно секвенциальные исчисления типа LK (наряду с аналитическими таблицами), которые можно использовать в качестве разрешающих процедур для логики одноместных предикатов первого порядка. Однако использованию их в этом качестве препятствует значительный рост вывода для формул со сложной структурой (дерево вывода с увеличением числа ветвей стремительно расширяется) и многократном дублировании подформул заключительной секвенции.

Мы предлагаем индексный метод – метод определения выводимости формул, основанный не на традиционном для секвенциальных исчислений конструировании вывода, а на некоторой его перекодировке, которая:

а) не уступает прямому конструированию в эффективности извлечения из заключительной секвенции информации, необходимой для ответа на вопрос о выводимости данной секвенции;

б) позволяет представлять эту информацию не в виде дерева – содержащего в явном виде все промежуточные шаги и параметрические формулы, – а в сравнительно компактной форме столбца строк, каждая из которых отображает только самую верхнюю секвенцию соответствующей ветви; в процессе индексации секвенциальное дерево вывода как бы сворачивается и предстает в значительно сокращенном виде.

Индексный метод является, на наш взгляд, одной из наиболее эффективных процедур поиска вывода (в том числе и автоматического) для исчисления одноместных предикатов первого порядка. При его применении происходит не обычное для секвенциальных исчислений построение вывода, а проводимая специальным образом перекодировка, направленная на анализ структуры формулы. Как правило, первый этап индексации позволяет делать оценку выводимости (невыводимости) формул, не прибегая к полному построению вывода (идет поиск симптомов невыводимости), что очень важно при работе с большими базами данных.

Идея индексного метода впервые была реализована для логики высказываний И.Н. Бродским[1]; расширение этого метода на логику одноместных предикатов первого порядка было предложено в статье В.П. Мухачева[2]. Мы рассмотрим некоторую модификацию индексного метода с указанным расширением; также будет рассмотрен пример применения индексного метода к логике двуместных предикатов; кроме того, мы разберем возможность ряда общих предварительных оценок (большая часть которых была предложена в упомянутой статье В.П. Мухачева), могущих на ранних стадиях процедуры индексации показать бесперспективность проведения ее в полном объеме.

Следует особо подчеркнуть тот факт, что предлагаемый индексный метод благодаря особенностям секвенциальных исчислений, на

---

[1] *Бродский И.Н.* Восстановление энтимемы // Вестник Ленинградского университета. 1988. Сер. 6. Вып. 1. С. 40–44.

[2] *Мухачев В.П.* Об одном усовершенствовании индексного метода Бродского И.Н. // Вестник СПбУ. 2000. Сер. 6. Вып. 2. С. 47–54.

которых он базируется, является процедурой *автоматического* вывода формул логики одноместных предикатов первого порядка.

Заметим, что индексный метод отличается от секвенциальных исчислений лишь внешним, синтаксическим образом. Безусловно, его можно разрабатывать и интерпретировать как самостоятельное исчисление, вне связи с исчислениями секвенциальными, но, поскольку для них уже давно получены соответствующие результаты, в частности, теоремы подформульности, полноты и разрешимости, удобно рассматривать индексный метод как эквивалент определенного варианта секвенциального исчисления. Этот вариант должен удовлетворять следующим условиям:

- для него верна Основная теорема Генцена[3];
- он не содержит структурных фигур заключения;
- он является исчислением одноместных предикатов;
- фигуры заключения допускают возможность обратного применения;
- во всех фигурах заключения, кроме кванторных, число вхождений логических знаков в каждой из верхних секвенций меньше числа вхождений логических знаков в нижней секвенции;
- во всех однопосылочных фигурах заключения, кроме кванторных, число вхождений переменных в верхнюю секвенцию равно числу вхождений переменных в нижнюю секвенцию;
- во всех двухпосылочных фигурах заключения число вхождений переменных в каждую из верхних секвенций на единицу меньше числа вхождений переменных в нижнюю секвенцию;
- основная секвенция и фигуры заключения не имеют ограничений на число входящих в них формул.

Заметим, что последнее условие не налагает ограничений на порядок расположения формул в основной секвенции и фигурах заключения. Это означает, что при построении вывода мы можем не придерживаться строгого порядка действий. Однако, если мы хотим совершенно автоматизировать процедуру вывода, уточнения, касающиеся порядка расположения формул, могут быть полезны.

Несколько видоизменив исчисление G4 Клини,[4] мы получаем нужный вариант – исчисление G4 :

---

[3] *Генцен Г.* Исследования логических выводов // Математическая теория логического вывода. М., 1967.

[4] *Клини С.К.* Математическая логика. М., 1969.

Основная секвенция

$\Gamma_1, D, \Gamma_2 \to \Delta_1, D, \Delta_2$

Пару формул ...*D*...→...*D*... мы будем называть контрадикторной парой.

Схемы фигур заключения

$$\frac{\Gamma_1, A, B, \Gamma_2 \to \Delta}{\Gamma_1, A \& B, \Gamma_2 \to \Delta}\ \& \to \qquad \frac{\Gamma \to \Delta_1, A, \Delta_2 \qquad \Gamma \to \Delta_1, B, \Delta_2}{\Gamma \to \Delta_1, A \& B, \Delta_2}\ \to \&$$

$$\frac{\Gamma_1, A, \Gamma_2 \to \Delta \qquad \Gamma_1, B, \Gamma_2 \to \Delta}{\Gamma_1, A \vee B, \Gamma_2 \to \Delta}\ \vee \to \qquad \frac{\Gamma \to \Delta_1, A, B, \Delta_2}{\Gamma \to \Delta_1, A \vee B, \Delta_2}\ \to \vee$$

$$\frac{\Gamma_1, \Gamma_2 \to \Delta_1, A, \Delta_2 \qquad \Gamma_1, B, \Gamma_2 \to \Delta_1, \Delta_2}{\Gamma_1, A \supset B, \Gamma_2 \to \Delta_1, \Delta_2}\ \supset \to \qquad \frac{\Gamma_1, A, \Gamma_2 \to \Delta_1, B, \Delta_2}{\Gamma_1, \Gamma_2 \to \Delta_1, A \supset B, \Delta_2}\ \to \supset$$

$$\frac{\Gamma_1, \Gamma_2 \to \Delta_1, A, \Delta_2}{\Gamma_1, \neg A, \Gamma_2 \to \Delta_1, \Delta_2}\ \neg \to \qquad \frac{\Gamma_1, A, \Gamma_2 \to \Delta_1, \Delta_2}{\Gamma_1, \Gamma_2 \to \Delta_1, \neg A, \Delta_2}\ \to \neg$$

$$\frac{\Gamma_1, Ft, \forall x Fx, \Gamma_2 \to \Delta}{\Gamma_1, \forall x Fx, \Gamma_2 \to \Delta}\ \forall \to \qquad \frac{\Gamma \to \Delta_1, Fa, \Delta_2}{\Gamma \to \Delta_1, \forall x Fx, \Delta_2}\ \to \forall$$

$$\frac{\Gamma_1, Fa, \Gamma_2 \to \Delta}{\Gamma_1, \exists x Fx, \Gamma_2 \to \Delta}\ \exists \to \qquad \frac{\Gamma \to \Delta_1, \exists x Fx, Ft, \Delta_2}{\Gamma \to \Delta_1, \exists x Fx, \Delta_2}\ \to \exists$$

В фигурах заключения →∀ и ∃→ действуют стандартные ограничения на собственную переменную.

Процедура построения вывода некоторой формулы *F* в таком исчислении сводится к преобразованию заключительной секвенции →*F* в основные секвенции путем обратных применений фигур заключения (построение дерева вывода снизу вверх). Заметим, что схемы фигур заключения построены так, что при обратном их применении в произвольном выводе число логических связок в секвенциях любой ветви неуклонно уменьшается от заключительной секвенции к основной. В фигурах заключения ∀→ и →∃ приходится все же повторять в их верхних секвенциях главные формулы ($\forall xFx$ и $\exists xFx$) для обеспечения возможности неоднократного использования этих формул при построении вывода (поскольку в нашем исчислении G4 отсутствуют структурные фигуры заключения, в частности, фигура «сокращение»), и поэтому вывод несколько усложняется. В индексном методе этот недостаток устраняется применением специального символа произвольной объектной переменной «γ».

Для выводимости заключительной секвенции необходимо и достаточно, чтобы все ветви вывода оканчивались основными секвенциями. Рассмотрим вывод в исчислении G4* формулы Пирса:

Пример 1.

$$\dfrac{\dfrac{\dfrac{p \to p, q}{\to p, p \supset q}\,{\to}{\supset} \qquad p \to p}{(p \supset q) \supset p \to p}\,{\supset}{\to}}{\to ((p \supset q) \supset p) \supset p}\,{\to}{\supset}$$

Обе ветви оканчиваются основными секвенциями, следовательно, формула выводима.

Индексный метод позволяет, не прибегая к конструированию вывода, указать, какие переменные (или предикаты) на каких местах (справа или слева от знака секвенции) входят в самые верхние секвенции всех ветвей, что, очевидно, дает ответ на вопрос о выводимости заключительной секвенции.

Процедура индексации делится на два этапа:

1) Определение индексов «справа» и «слева» для переменных/предикатов и логических знаков; становятся возможными различные оценки выводимости данной формулы;

2) Определение принадлежности переменных/предикатов данной формулы ветвям предполагаемого вывода. Этот полный и окончательный анализ проводится в том случае, если все предварительные оценки ничего не дают.

Первый этап процедуры проводится следующим образом:

1.1. Главный знак доказываемой формулы получает индекс «R».

1.2. Пусть главный знак некоторой подформулы имеет индекс «R» или «L», тогда приписывание индексов главным знакам (или единичным переменным/предикатам) ее подформул производится в соответствии с таблицей.

	A & B	A ∨ B	A ⊃ B	¬ A	∃x A	∀x A
R	+ R R	- R R	- L R	L	* R	R
L	- L L	+ L L	+ R L	R	L	* L

Верхняя строка указывает вид рассматриваемой подформулы, правый столбец – индекс ее главного знака. В пересечении указывается дополнительный индекс главного знака («+», «-» или «*»), а также индексы главных знаков (или единичных переменных/предикатов) соответствующих подформул.

«R» читается как «справа», «L» – как «слева»; эти символы указывают места формул относительно знаков секвенций в предполагаемом дереве вывода. Если отвлекаться от секвенциальной интерпретации, их можно понимать как «истину» и «ложь». Индекс «+» говорит о том, что при конструировании вывода к соответствующей формуле была бы применена двухпосылочная фигура заключения, а индекс «-» – о том, что была бы применена однопосылочная фигура заключения. Сразу поясним, что индексирование отрицаний и кванторов как «однопосылочных» было бы не только излишним, но и создало бы помеху использованию некоторых предварительных оценок. Наконец, символ «*» отмечает кванторы, фигуры заключения которых не имеют ограничения на собственную переменную.

1.3. Пусть в доказываемой формуле существует хотя бы одно безындексное вхождение переменной или предиката, тогда процедура возвращается к пункту 1.2.; в противном случае первый этап процедуры завершается.

Пример 1 (продолжение). Пусть дана формула Пирса ((p⊃q)⊃p)⊃p. Процедура индексации проходит следующим образом:

Записываем секвенцию →((p⊃q)⊃p)⊃p.

Главным знаком формулы является импликация, поэтому ей мы приписываем индекс «R»:

$$\to((p \supset q) \supset p) \underset{R}{\supset} p$$

В соответствии с таблицей эта же импликация получает дополнительный индекс «-», а главные знаки ее подформул – индексы «L» и «R»:

$$\begin{array}{ccccccccccc}
\rightarrow & ((& p & \supset & q &) & \supset & p &) & \supset & p \\
 & & & & & & & & & - & \\
 & & & & & & L & & & R & R
\end{array}$$

Далее рассматриваем уже подформулу (p⊃q)⊃p, поскольку индексация подформулы p уже завершена:

$$\begin{array}{ccccccccccc}
\rightarrow & ((& p & \supset & q &) & \supset & p &) & \supset & p \\
 & & & & & & + & & & - & \\
 & & & R & & & L & L & & R & R
\end{array}$$

И завершаем процедуру следующим образом:

$$\begin{array}{ccccccccccc}
\rightarrow & ((& p & \supset & q &) & \supset & p &) & \supset & p \\
 & & & - & & & + & & & - & \\
 & & L & R & R & & L & L & & R & R
\end{array}$$

Все вхождения переменных получили свои индексы, первый этап завершен. Информация, необходимая для второго этапа индексации, получена, но возникает вопрос: нельзя ли сделать какое-либо заключение относительно выводимости или невыводимости предложенной формулы уже на данной стадии проведения разрешающей процедуры? Переходя к ответу на этот вопрос, сделаем несколько наблюдений, касающихся свойств формул и их G4 -выводов.

Всякая формула содержит переменные или предикаты по крайней мере одного из следующих видов:

а) *участвующие в образовании контрадикторных пар;*

б) *не участвующие в образовании контрадикторных пар.*

G4 -вывод всякой формулы может содержать фигуры заключений по крайней мере одного из следующих видов:

в) *однопосылочные*, т.е. не образующие новые ветви;

г) *двухпосылочные*, т.е. образующие новые ветви.

Исходя из этих элементарных свойств, заметим, что:

1. Пусть некоторая формула содержит вхождения только различных переменных/предикатов, тогда эта формула не является

$G4^*$-выводимой; такая негативная оценка позволяет для соответствующего класса формул вообще не начинать процедуру индексации;

2. Пусть существуют переменные/предикаты, каждая из которых входит в некую формулу более одного раза, но только с одной стороны от знаков секвенций (то есть все ее вхождения отмечены либо только индексом «R», либо только индексом «L»), тогда эта формула не является $G4^*$-выводимой; эта и последующие оценки становятся возможными по завершении первого этапа процедуры индексации;

3. Пусть некоторая формула содержит такие логические знаки, которые вводятся применением только однопосылочных фигур заключения, тогда данная формула является $G4^*$-выводимой, если она содержит хотя бы одну контрадикторную пару; очевидно, это частный случай оценки 4;

4.1. Если число контрадикторных пар P (pair) меньше числа ветвей B (branch), т.е. $P < B$, то данная формула не является $G4^*$-выводимой;

4.2. Если вхождений переменных/предикатов в самые верхние секвенции вывода количественно не хватает для образования числа контрадикторных пар, хотя бы равного числу ветвей (т.е. число «потенциальных» контрадикторных пар P_p меньше числа ветвей), то данная формула не является $G4^*$-выводимой;

4.3. Пусть число «гарантированных» (заведомо находящихся в различных ветвях) контрадикторных пар P_g не меньше числа ветвей, тогда формула является $G4^*$-выводимой;

5. Для сравнительно несложных формул наблюдается такая закономерность: если число однопосылочных фигур заключения (S, single) не больше числа двухпосылочных фигур заключения (T, two), то формула невыводима. Это правило, будь оно применимо к любым формулам, было бы очень мощным критерием отбора, но, как мы покажем ниже, о такой применимости не может быть и речи;

6. В некоторых формулах существуют так называемые «чисто сукцедентные» или «чисто антецедентные» вхождения переменных/предикатов, т.е. такие вхождения, которые указывают на присутствие переменной/предиката в сукцедентах или антецедентах самых верхних секвенций всех ветвей; в примере 1 «чисто сукцедентным» вхождением переменной p является самое правое ее вхождение; способ выявления таких вхождений станет ясен при детальном разборе оценки 4;

7. Если в предполагаемую контрадикторную пару входит предикат, связанный в доказываемой формуле квантором, фигура заключения для которого имеет ограничения, тогда тот же предикат должен быть связан в доказываемой формуле также и квантором, фигура заключения для которого ограничений не имеет – то есть должно быть так называемое «равновесное» сочетание фигур с ограничениями и без них; такая оценка является особым случаем оценки 4.

Как мы видим, оценки 1 и 2 тривиальны, прочие же, кроме оценки 5, суть частные случаи оценки 4.

Рассмотрим оценку 5. Существует класс формул, опровергающих данную оценку. Скажем, в приводимом чуть далее примере 3 она явно не работает; более того, каким бы образом мы не формулировали подобный критерий – S>2T, T≤(3S+1) и т.п. – можно привести формулу, в него не укладывающуюся. Поясним это на примере:

Пример 2.

$$\begin{array}{c}
p\to p \quad p\to p \\ \hline
(p\vee p)\to p \quad p\to p \\ \hline
((p\vee p)\vee p)\to p \quad p\to p \\ \hline
(((p\vee p)\vee p)\vee p)\to p \\ \hline
\to(((p\vee p)\vee p)\vee p)\supset p \\
+ \quad + \quad + \quad - \\
L\;L\;L\;L\;L\;L\;\;R\;R
\end{array}$$

Очевидно, подобным способом легко сконструировать формулу с любым соотношением «+/-» (используя кратные дизъюнкции или конъюнкции). Можно, однако, описать ситуацию, когда критерий S>T (однопосылочных фигур заключения в выводимой формуле должно быть больше, чем двухпосылочных) выполняется. Назовем «потенциалом секвенции» (P) минимальное число применений однопосылочных фигур заключения, требуемое для приведения ее к виду заключительной секвенции. Потенциал любой заключительной секвенции равен нулю, потенциал любой основной секвенции – больше нуля, поэтому для справедливости критерия S>T достаточно, чтобы всякое применение в выводе двухпосылочной фигуры заключения порождало секвенцию с потенциалом большим, чем потенциал любой из двух верхних секвенций данной фигуры заключения. Легко видеть, что во всех двухпосылочных фигурах G4* потенциалы верхних и нижних секвенций равны, а потому нам пришлось бы видоизменить их, например, таким образом:

$$\frac{\Gamma_1 \to A, \Delta_1 \quad \Gamma_2 \to B, \Delta_2}{\Gamma_1, \Gamma_2 \to A \& B, \Delta_1, \Delta_2} \to\& \qquad \frac{A, \Gamma_1 \to \Delta_1 \quad B, \Gamma_2 \to \Delta_2}{A \vee B, \Gamma_1, \Gamma_2 \to \Delta_1, \Delta_2} \vee\to$$

$$\frac{\Gamma_1 \to A, \Delta_1 \quad B, \Gamma_2 \to \Delta_2}{A \supset B, \Gamma_1, \Gamma_2 \to \Delta_1, \Delta_2} \supset\to$$

Однако данные фигуры заключения не допускают возможности обратного применения; кроме того, поскольку мы не можем использовать структурные фигуры заключения (они никак не отображаются в составе заключительной секвенции), определенный класс формул – в частности, формулу $p \supset (p\&p)$ – уже нельзя вывести. Поэтому такой вариант исчисления для нас неприемлем, и сомнительным представляется существование его аналога (удовлетворяющего соотношению S>T), пригодного для построения индексного метода.

Вообще говоря, в случае нашего исчисления G4* потенциалы всех самых верхних секвенций одинаковы (в силу обратимости фигур заключения) и численно равны S.

Будь мы всегда в состоянии оценить, хватает ли «гарантированных» контрадикторных пар на все ветви (критерий 4., $P_g \geq B$), второй этап процедуры индексации был бы ненужен. Сделать это мы можем лишь иногда, но проверить, достаточно ли у нас «потенциальных» контрадикторных пар (критерий 4.2., $P_p > B$), можно для любой формулы.

Число ветвей B подсчитывается следующим образом: в доказываемой формуле все вхождения переменных/предикатов заменяются единицами, логические знаки с индексом «+» – арифметическими плюсами, с индексом «-» – арифметическими умножениями, кванторы и отрицания элиминируются. Число, полученное после проведения всех арифметических операций над результирующим выражением, и является искомым B.

Пример 3. После первого этапа индексации мы имеем

$$\begin{array}{c}\to((p\supset((p\&q)\&r))\vee(q\supset((p\&q)\&r)))\vee(r\supset((p\&q)\&r))\\ -\quad +\quad +\quad -\quad -\quad +\quad +\quad -\quad -\quad +\quad +\\ LR\ \ RRRRR\ \ RLR\ \ RRRRR\ \ RLR\ \ RRRRR\end{array}$$

В = ((1 · ((1+1)+1)) · (1 · ((1+1)+1))) · (1 · ((1+1)+1)) = 27, то есть дерево вывода для данной формулы содержит 27 ветвей.

Обоснованием такого подсчета служит следующее рассуждение: ясно, что дерево вывода формулы, состоящей из единственной переменной или предиката, состоит из одной ветви; далее, допустим, что для некоторой формулы известно, какое количество ветвей образует каждая из двух ее главных подформул (подформул, связанных главным знаком формулы), тогда

а) если главный знак имеет индекс «+», главные подформулы разводятся по разным ветвям, и ветви, ими порождаемые, суммируются;

б) если главный знак имеет индекс «-», главные подформулы остаются в одной ветви, но при этом они разделены запятой или знаком секвенции, следовательно, все ветви, образуемые одной из главных подформул, параметрически наследуют другую (а значит, и все ее ветви) – и наоборот, и естественно, что для подсчета используется умножение.

Вычисление числа P_p, основанное на той же идее, будет продемонстрировано на примере 4. С подсчетом же числа P и, тем более, P_g дело обстоит сложнее; мы пока не имеем метода их нахождения, который выигрывал бы в простоте у проводимой до конца процедуры индексации.

Пример 4. Рассмотрим формулу (p∨q) ⊃ p. Первый этап процедуры индексации дает нам расстановку символов «+», «-», «L» и «R» для логических знаков и переменных:

$$\begin{array}{c}\to(p\vee q)\supset p\\ +\quad -\\ LLL\ \ RR\end{array}$$

Вычислим количество вхождений переменной p слева от знаков секвенций в самых верхних секвенциях вывода (pL). Берем первое левое вхождение p в данную формулу и принимаем pL за 1. Поскольку формула p связана с формулой q «+»-знаком, на этом структурном уровне (в подформуле $(p \vee q)$) pL остается прежним. Далее, так как формула $(p \vee q)$ связана с формулой p «-»-знаком, текущее значение pL домножается на число B формулы p (равное единице). На структурном уровне подформулы $(p \vee q) \supset p$, таким образом, pL=1, но подформула $(p \vee q) \supset p$ эквивалентна исходной формуле, поэтому подсчет pL прекращается. Если бы существовало несколько левых вхождений переменной p в исходную формулу, значения pL для каждого из них в сумме и составили бы итоговое число pL.

Итак, pL=1. Подсчет pR производим аналогичным образом – pR=2. Какое же количество контрадикторных пар потенциально могли бы образовать одно левое и два правых вхождения переменной p в самые верхние секвенции рассматриваемой формулы? Очевидно, они могли бы образовать лишь одну пару. Число pP_p – число «потенциальных» контрадикторных пар, порожденных переменной p – равно меньшему из чисел (pL, pR), в данном случае – pL. Формально (поскольку мы априори не знаем соотношения этих чисел) pP_p является результатом выражения:

$$pP_p = \frac{pL + pR - |pL - pR|}{2},$$

а общее количество P_p по всем переменным/предикатам находится по формуле $P_p = \sum_{i=1}^{n} \frac{p_iL + p_iR - |p_iL - p_iR|}{2}$, где $p_1, \ldots, p_i, \ldots, p_n$ – список переменных либо предикатов.

Что же касается нашего случая и переменной q, то qL=1, qR=0 (так как правых вхождений q нет совсем). Число qP_p равно нулю, следовательно, $P_p = 1$; а поскольку B = 2, мы получаем отрицательный ответ на вопрос о выводимости формулы $(p \vee q) \supset p$.

Существуют пути усиления такой оценки. Действуя описанным выше образом, мы не учитываем то обстоятельство, что в какой-либо из самых верхних секвенций некоторая переменная или предикат по какую-нибудь сторону от знака секвенции может входить многократно, нежелательно увеличивая число «потенциальных» контрадикторных пар. Данный недостаток устраняется следующим изменением процедуры подсчета:

1) число xL_g (xR_g) для каждого конкретного левого (правого) вхождения некоторой переменной/предиката x в заключительную секвенцию принимаем равным единице;

2) пусть для некоторых подформул 1 и 2 данной формулы, связанных логическим знаком, и некоторой переменной/предиката x известны числа B_1, B_2, xL_{g1} (xR_{g1}) и xL_{g2} (xR_{g2}), т.е. количества ветвей, образуемых каждой из подформул, и количества самых верхних секвенций выводов этих подформул, в которые входит данная переменная/предикат с указанной стороны от знаков секвенций; тогда

а) если подформулы связывает «+»-знак, то результирующие $B_3 = B_1 + B_2$, $xL_{g3} = xL_{g1} + xL_{g2}$ ($xR_{g3} = xR_{g1} + xR_{g2}$);

б) если подформулы связывает «-»-знак, то результирующие $B_3 = B_1 \cdot B_2$, $xL_{g3} = xL_{g1} \cdot B_2 + xL_{g2} \cdot B_1 - xL_{g1} \cdot xL_{g2}$ ($xR_{g3} = xR_{g1} \cdot B_2 + xR_{g2} \cdot B_1 - xR_{g1} \cdot xR_{g2}$).

Полученные для всей заключительной секвенции в целом числа xL_g и xR_g «гарантированных» левых и правых вхождений (т.е. вхождений, находящихся в заведомо различных самых верхних секвенциях) переменной/предиката x подставляются в формулу для подсчета P_p вместо соответствующих xL и xR.

Такое усиление оценки числа P_p весьма существенно. Скажем, для формулы из примера 3 pR – 27, а pR_g – 19, число «потенциальных» контрадикторных пар P_p во втором случае сразу сокращается на 8 (хотя для данной формулы это не играет никакой роли – P_p все равно значительно больше числа ветвей B).

На этом мы оставляем тему предварительных оценок и переходим к описанию второго этапа процедуры индексации. Она проходит так:

2.1. Главный знак доказываемой формулы записывается в строке номер 1.

2.2. Пусть процедура уже породила j строк, и пусть главный знак некоторой подформулы в i-той строке не заключен в ломаные скобки, тогда он заключается в ломаные скобки, при этом

а) пусть главным знаком данной подформулы является квантор, тогда, если он имеет индекс «*», все связанные им предикаты записываются в этой же строке с символом произвольной объектной переменной «γ», если не имеет – с символом объектной переменной, не встречающейся в данной строке;

б) пусть главный знак данной подформулы не имеет индекса «+», тогда главные знаки (или единичные переменные/предикаты) ее подформул записываются в той же строке;

в) пусть главный знак данной подформулы имеет индекс «+», тогда главный знак (или единичная переменная/предикат) одной из ее подформул записывается в той же строке, а главный знак (или единичная переменная/предикат) другой ее подформулы – в (j+1)-й строке; в этом случае все символы, записанные в i-той строке и не заключенные в ломаные скобки, переписываются в (j+1)-ю строку;

г) если при разборе главного знака данной подформулы главным знаком ее подформулы оказался предикат, этот предикат записывается с квадратными скобками или (если он уже был записан на соответствующей позиции) заключается в квадратные скобки.

2.3. Справа от каждой строки выписываем (если это еще не сделано) ту переменную или предикат (значимыми являются лишь предикаты, заключенные в квадратные скобки), который образует в данной строке контрадикторную пару (в принципе, можно было бы проверять и более сложные формулы на предмет образования ими контрадикторных пар, но мы пока не усматриваем приемлемого способа такой проверки). Предикат с символом произвольной объектной переменной «γ» образует контрадикторную пару с любым одноименным предикатом – вне зависимости от его объектной переменной.

2.4. Если на некотором шаге вывода для всех существующих строк найдена контрадикторная пара, процедура прекращается – доказываемая формула выводима. Если существует строка, в которой уже нет логических знаков без скобок, но нет и контрадикторной пары, процедура прекращается – доказываемая формула невыводима. Если процедура не завершена, она продолжается с шага 2.2.

Следует уточнить, что запись всех символов осуществляется на одной вертикали с соответствующими вхождениями этих символов в разбираемую секвенцию, поэтому последующий анализ получаемых строк на предмет наличия в них контрадикторных пар подразумевает поиск одинаковых переменных (или предикатов), на одной вертикали с одной из которых находится индекс «L», а на одной вертикали с другой – индекс «R». Отметим, что последовательность индексации – как и порядок построения дерева вывода – не оказывает влияния на итоговый результат, но для программной реализации нашего метода требуется, естественно, формулирование некоторого алгоритма – включающего в себя, например, требования рассматривать верхние строки прежде нижних, левые индексы – прежде правых и постоянно возвращать «глаз» анализатора в начало строки. В плане удобства такой реализации было бы актуально и некоторое изменение процедуры – вместо заключения в ломаные скобки (которые мы используем ввиду первоначальной ориентации метода на рукописную демонстрацию вывода) *символы логических связок можно попросту стирать.*

Нужно пояснить смысл употребления с предикатами квадратных скобок, так как, казалось бы, разница между пропозициональной переменной и предикатом в рамках нашей процедуры невелика. Дело в том, что при рассмотрении некоторого квантора сразу во многих вхождениях предиката в формулу может произойти подстановка переменной, и этот факт надо отразить (записать все вхождения) немедленно, поскольку любой квантор и логическая связка рассматривается в ходе процедуры лишь единожды. Однако принадлежность вхождений предикатов строкам может определиться много позже и только тогда должна быть отмечена особым образом (скобками).

По поводу переписывания бесскобочных предикатов из одних строк в другие нужно сказать следующее: хотя зачастую такая операция непродуктивна – это бывает, когда в строке, в которую переписываются предикаты, нет бесскобочных логических знаков, при разборе которых эти предикаты были бы заключены в квадратные скобки – разумнее все же проводить ее, поскольку проверка существования упомянутых логических знаков излишне усложнила бы всю процедуру.

Пример 5. Рассмотрим секвенцию → (p∨q) ⊃ (p&q). Дерево ее вывода в рамках исчисления G4' выглядит так:

$$\cfrac{\cfrac{\cfrac{p \to p \quad p \to q}{p \to p \& q}\,{\to}\& \qquad \cfrac{q \to p \quad q \to q}{q \to p \& q}\,{\to}\&}{p \vee q \to p \& q}\,{\vee}{\to}}{\to (p \vee q) \supset (p \& q)}\,{\to}{\supset}$$

Вторая и третья верхние секвенции не являются основными, следовательно, формула (p∨q) ⊃ (p&q) невыводима. А теперь применим к данной секвенции полную процедуру индексации. Первый этап завершается следующим выражением:

$$\begin{array}{cccccccccccc}
\rightarrow & (& p & \lor & q &) & \supset & (& p & \& & q &) \\
 & & & + & & & - & & & + & & \\
 & & L & L & L & & R & & R & R & R &
\end{array}$$

Второй этап начинается с записи главного знака, то есть импликации, в первой строке, затем в соответствии с пунктом 2.2. эта импликация заключается в ломаные скобки, и главные знаки подформул записываются в той же строке:

$$\begin{array}{ccccccccccccc}
 & \rightarrow & (& p & \lor & q &) & \supset & (& p & \& & q &) \\
 & & & & + & & & - & & & + & & \\
 & & & L & L & L & & R & & R & R & R & \\
1 & & & & \lor & & & \langle\supset\rangle & & & \& & &
\end{array}$$

Пункт 2.3. не вносит изменений, а пункт 2.4. возвращает процедуру к пункту 2.2.. Дизъюнкция здесь имеет индекс «+», и по пункту процедуры 2.2.в записываем переменную p в той же строке, а переменную q – в новой строке, при этом конъюнкция (как не заключенная в ломаные скобки) также записывается в новой строке:

$$\begin{array}{ccccccccccccc}
 & \rightarrow & (& p & \lor & q &) & \supset & (& p & \& & q &) \\
 & & & & + & & & - & & & + & & \\
 & & & L & L & L & & R & & R & R & R & \\
1 & & & p & \langle\lor\rangle & & & \langle\supset\rangle & & & \& & & \\
2 & & & & & q & & & & & \& & &
\end{array}$$

Разбирая конъюнкцию первой строки, пишем переменную p в той же строке, а переменую q – в новой, при этом в новую строку записывается и (как бесскобочная) переменная p. По пункту 2.3. выписываем справа от первой строки переменную p, образующую контрадикторную пару:

$$\begin{array}{cccccccccccccc}
 & \rightarrow & (& p & \lor & q &) & \supset & (& p & \& & q &) & \\
 & & & & + & & & - & & & + & & & \\
 & & & L & L & L & & R & & R & R & R & & \\
1 & & & p & \langle\lor\rangle & & & \langle\supset\rangle & & p & \langle\&\rangle & & & p \\
2 & & & & & q & & & & & \& & & & \\
3 & & & p & & & & & & & & q & &
\end{array}$$

Наконец рассматриваем конъюнкцию второй строки и получаем итоговый результат:

$\rightarrow$	$(p$	$\vee$	$q)$	$\supset$	$(p$	$\&$	$q)$	
		$+$		$-$		$+$		
	L	L	L	R	R	R	R	
1	p	$\langle\vee\rangle$		$\langle\supset\rangle$	p	$\langle\&\rangle$		p
2			q		p	$\langle\&\rangle$		
3	p						q	
4			q				q	q

Во второй и третьей строке нет контрадикторных пар, следовательно, формула (p∨q) ⊃ (p&q) невыводима. Для демонстрации соответствия индексных строк самым верхним секвенциям традиционного вывода мы на одном из шагов процедуры пренебрегли пунктом 2.4., по которому индексация должна была бы прекратиться после записи третьей строки (в ней нет логических знаков без скобок и нет контрадикторной пары).

Пример 1 (продолжение). Вывод формулы Пирса при помощи индексного метода выглядит следующим образом:

$\rightarrow$	$((p$	$\supset$	$q)$	$\supset$	$p)$	$\supset$	p	
		$-$		$+$		$-$		
	L	R	R	L	L	R	R	
1	p	$\langle\supset\rangle$	q	$\langle\supset\rangle$		$\langle\supset\rangle$	p	p
2					p		p	p

Формула выводима, так как в обеих строках есть контрадикторные пары. Если же разобранные связки просто удалять из строк, то вывод выглядит так:

$\rightarrow$	$((p$	$\supset$	$q)$	$\supset$	$p)$	$\supset$	p	
		$-$		$+$		$-$		
	L	R	R	L	L	R	R	
1	p		q				p	p
2					p		p	p

Заметим, что существует альтернативный способ записи индексов, когда номера строк не пишутся, а соответствующие номерам строк цифры записываются в строках вместо символов переменных и связок. Такая запись использовалась в статье Мухачева В.П. «Об одном усовершенствовании индексного метода Бродского И.Н.» и в компьютерной программе «LogicMethods» (созданной под руководством Мухачева В.П. и реализующей – среди прочих исчислений – индексный метод для логики высказываний).

Пример 1 (продолжение). Вывод формулы Пирса в альтернативной записи индексного метода:

Первый этап завершается следующим выражением:

$$\begin{array}{cccccccccc}
\to((& p & \supset & q &) & \supset & p &) & \supset & p \\
 & & - & & & + & & & - & \\
 & L & R & R & & L & L & & R & R
\end{array}$$

Второй этап:

$$\begin{array}{cccccccccc}
\to((& p & \supset & q &) & \supset & p &) & \supset & p \\
 & & - & & & + & & & - & \\
 & L & R & R & & L & L & & R & R \\
 & & & & & 1 & & & \langle 1\rangle & 1
\end{array}$$

$$\begin{array}{cccccccccc}
\to((& p & \supset & q &) & \supset & p &) & \supset & p \\
 & & - & & & + & & & - & \\
 & L & R & R & & L & L & & R & R \\
 & & 1 & & & \langle 1\rangle & & & \langle 1\rangle & 1 \\
 & & & & & & 2 & & & 2
\end{array}$$

И завершается процедура так:

$$\begin{array}{ccccccccccc}
\to((& p & \supset & q &) & \supset & p &) & \supset & p & \\
 & & - & & & + & & & - & & \\
 & L & R & R & & L & L & & R & R & \\
 & 1 & \langle 1\rangle & 1 & & \langle 1\rangle & & & \langle 1\rangle & 1 & p \\
 & & & & & & 2 & & & 2 & p
\end{array}$$

Такой вид записи представляется не очень удобным (в особенности, для логики предикатов), так как здесь скрыты логические связки, и не видны явным образом переменные, образующие контрадикторные пары.

Пример 6. Пусть дана формула $\exists x(Fx \supset (\forall yFy \& Fx))$

Первый этап:

$$
\begin{array}{ccccccccccc}
\rightarrow \exists x & (& Fx & \supset & (& \forall y & Fy & \& & Fx &) &) \\
* & & & - & & & & + & & & \\
R & & L & R & & R & R & R & R & &
\end{array}
$$

Второй этап. Главным знаком является квантор $\exists$, не имеющий ограничения на собственную переменную, поэтому в оба его вхождения подставляется произвольная предметная переменная γ:

$$
\begin{array}{cccccccccccc}
 & \rightarrow \exists x & (& Fx & \supset & (& \forall y & Fy & \& & Fx &) &) \\
 & * & & & - & & & & + & & & \\
 & R & & L & R & & R & R & R & R & & \\
1 & \langle\exists\rangle & & F_\gamma & \supset & & & & & F_\gamma & &
\end{array}
$$

При разборе импликации одно из вхождений предиката F_γ заключается в квадратные скобки, что означает его наличие в этой строке (другое вхождение предиката F_γ присутствует в строке лишь виртуально и необходимо для того, чтобы помнить, какая переменная была в него подставлена; реальным это вхождение станет только во второй строке, когда будет заключено в квадратные скобки):

$$
\begin{array}{cccccccccccc}
 & \rightarrow \exists x & (& Fx & \supset & (& \forall y & Fy & \& & Fx &) &) \\
 & * & & & - & & & & + & & & \\
 & R & & L & R & & R & R & R & R & & \\
1 & \langle\exists\rangle & & [F_\gamma] & \langle\supset\rangle & & & & \& & F_\gamma & &
\end{array}
$$

Разбор конъюнкции:

$$
\begin{array}{lllllllll}
 & \rightarrow \exists x\ (& Fx & \supset & (\ \forall y & Fy & \& & Fx\ )\ ) & \\
 & * & & - & & & + & & \\
 & R & L & R & R & R & R & R & \\
1 & \langle\exists\rangle & [F_\gamma] & \langle\supset\rangle & \forall & & \langle\&\rangle & F_\nu & \\
2 & & [F_\gamma] & & & & & [F_\gamma] & Fa
\end{array}
$$

Разбор квантора $\forall$:

$$
\begin{array}{lllllllll}
 & \rightarrow \exists x\ (& Fx & \supset & (\ \forall y & Fy & \& & Fx\ )\ ) & \\
 & * & & - & & & + & & \\
 & R & L & R & R & R & R & R & \\
1 & \langle\exists\rangle & [F_\gamma] & \langle\supset\rangle & \langle\forall\rangle & [F_a] & \langle\&\rangle & F_\gamma & Fa \\
2 & & [F_\gamma] & & & & & [F_\gamma] & Fa
\end{array}
$$

Итак, формула выводима, поскольку в обеих строках образовались контрадикторные пары.

Применение символа произвольной объектной переменной «γ» необходимо, чтобы не оговаривать в процедуре индексации многократное рассмотрение подформул, главными знаками которых являются кванторы без ограничений (то есть с индексом «*»). Поясним нашу мысль на примере

Пример 7. В рамках исчисления G4* вывод формулы $\exists x(Fx \supset \forall yFy)$ выглядит так:

$$
\begin{array}{cl}
Fb, Fa \rightarrow \exists x(Fx \supset \forall yFy), \forall yFy, Fb & \\
\hline
Fa \rightarrow \exists x(Fx \supset \forall yFy), Fb \supset \forall yFy, Fb & \rightarrow\supset \\
\hline
Fa \rightarrow \exists x(Fx \supset \forall yFy), Fb & \rightarrow\exists \\
\hline
Fa \rightarrow \exists x(Fx \supset \forall yFy), \forall yFy & \rightarrow\forall \\
\hline
\rightarrow \exists x(Fx \supset \forall yFy), Fa \supset \forall yFy & \rightarrow\supset \\
\hline
\rightarrow \exists x(Fx \supset \forall yFy) & \rightarrow\exists
\end{array}
$$

Как мы видим, фигура заключения $\rightarrow\exists$ используется два раза, поскольку подставленная в первый раз вместо x переменная a накладывает ограничения на применение фигуры заключения $\rightarrow\forall$, и только повторная подстановка (уже переменной b вместо x) позволяет получить контрадикторную пару.

А вот как лаконично будет выглядеть результат полной процедуры индексации для данной формулы:

$$\begin{array}{cccccccc}
 & \to \exists x\ (& Fx & \supset & \forall y & Fy\ ) & & \\
 & * & & _ & & & & \\
 & R & L & R & R & R & & \\
1 & \langle\exists\rangle & [F_\gamma] & \langle\supset\rangle & \langle\forall\rangle & [F_a] & & F_a
\end{array}$$

Напоследок нам хотелось бы рассмотреть пример формулы с двуместными предикатами и пробный ее вывод при помощи исчисления G4* и, соответственно, индексного метода.

Пример 8. Совместим на одной схеме секвенциальный и индексный вывод формулы $\exists x \forall y Fxy \supset \forall x \exists y Fxy$:

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\forall yFav, Fab \to Fba, \exists yFby}{\forall yFay, Fab \to \exists yFby}\ {\to}\exists}{\forall yFay \to \exists yFby}\ \forall{\to}}{\forall yFay \to \forall x\exists yFxy}\ {\to}\forall}{\exists x\forall yFxy \to \forall x\exists vFxy}\ \exists{\to}}{\to \exists x\ \forall y\ Fxy\ \supset\ \forall x\ \exists y\ Fxy}\ {\to}{\supset}$$

$$\begin{array}{cccccccc}
 & & * & & _ & & * & \\
 & L & L & L & R & R & R & R \\
1 & \langle\exists\rangle & \langle\forall\rangle & [Fa\gamma] & \langle\supset\rangle & \langle\forall\rangle & \langle\exists\rangle & [Fb\gamma]
\end{array}$$

Контрадикторная пара не образуется, следовательно, формула необщезначима. Однако она выполнима, в частности, она истинна для всех интерпретаций предиката Fxy, при которых он обладает свойством симметричности, например, для предиката «=» (тогда первую γ следует понимать как b, а вторую – как a). Формула становится ложной, если интерпретировать Fxy как предикат, не обладающий свойством симметричности, например, «>». Здесь помимо теоретических сложностей (поскольку исчисление двуместных предикатов, вообще говоря, неразрешимо) возникают и сложности чисто практические – в нашем примере мы подставили в предикаты и записали сразу обе переменные, тогда как при правильной последовательности операций надо было бы подставлять переменные по одной и записывать предикаты в таком виде, и места для записи предикатов с обеими подставленными переменными уже не осталось бы.

Возникает вопрос: насколько все же компактнее вывод, построенный средствами индексного метода, по сравнению с выводом в секвенциальном исчислении G4*? Попробуем произвести примерный подсчет.

Пусть нам дана формула F с произвольным числом бинарных связок Bin (для простоты мы не будем учитывать кванторы и отрицания; кроме того, мы предполагаем, что вывод продолжается до тех пор, пока не будут разобраны все бинарные связки, и не останавливается даже в случае очевидной выводимости или невыводимости формулы F). Тогда число переменных в ней будет равно Bin+1, число скобок – 2(Bin-1), а общая длина в символах будет равна 4Bin (если учитывать знак секвенции).

Минимальное возможное число ветвей вывода равно 1, максимальное – $3^{(Bin+1):3}$ (предполагая, что максимальное ветвление достигается, когда формула разбита на блоки по 3 переменные так, что внутри блоков переменные соединены связками, требующими применения двухпосылочных фигур заключения, а сами блоки соединены связками, требующими применения однопосылочных фигур заключения). Высота вывода, то есть максимальное число секвенций, находящихся в одной нити (с учетом заключительной секвенции), равна Bin+1.

Рассмотрим два крайних случая:

1. Вывод состоит из одной ветви. Длина самой верхней его секвенции равна 2Bin+2. Общее число символов вывода равно

$$(Bin+1)\cdot(4Bin+2Bin+2):2-4Bin = 3Bin^2+1$$

(по формуле для площади трапеции за вычетом символов заключительной секвенции);

2. Вывод состоит из $3^{(Bin+1):3}$ ветвей. Длина самых верхних секвенций равна (2Bin+2):3-1. Суммарная длина самых верхних секвенции равна $((2Bin+2):3-1)\cdot 3^{(Bin+1):3}$. Общее число символов вывода равно

$$(Bin+1)\cdot(4Bin+((2Bin+2):3-1)\cdot 3^{(Bin+1):3}):2-4Bin$$

(по формуле для площади трапеции за вычетом символов заключительной секвенции).

Усредняя эти два крайних случая, получаем приблизительное число символов секвенциального вывода произвольной формулы F с числом бинарных связок Bin:

$$(3Bin^2+1+(Bin+1)\cdot(4Bin+((2Bin+2):3-1)\cdot3^{(Bin+1):3}):2-4Bin):2$$

Если рассматривать вариант индексного метода *со стиранием разобранных логических связок* и если учитывать, что в индексных строках записываются лишь самые верхние секвенции подразумеваемого секвенциального вывода, то в аналогичных крайних случаях числа символов вывода равны:

1. Bin+1;
2. $(Bin+1):3\cdot3^{(Bin+1):3}$

и среднее арифметическое равно $(Bin+1+(Bin+1):3\cdot3^{(Bin+1):3}):2$.

В пределе значимыми будут лишь показательные функции (как растущие быстрее всего) и коэффициенты при них, и тогда число символов вывода, построенного средствами индексного метода, будет соотноситься с числом символов секвенциального вывода как $(Bin+1)\cdot((2Bin+2):3-1)\cdot3^{(Bin+1):3}$ и $(Bin+1):3\cdot3^{(Bin+1):3}$, то есть первый из этих выводов будет компактнее второго в $3\cdot((2Bin+2):3-1)$ раз – грубо говоря, в 2Bin раз. Тогда, скажем, для формулы, включающей 100 бинарных связок, индексный вывод будет компактнее в 200 раз.

Рассмотренная нами процедура индексации эквивалентна построению дерева вывода в G4, но имеет выгодное в известном смысле отличие. Благодаря перекодировке вывода устраняются указанные нами недостатки секвенциальных исчислений: вывод имеет фиксированную ширину, линейно зависящую от размера доказываемой формулы – что существенно в плане его демонстрации; кроме того, строки индексов фактически отображают лишь самые верхние секвенции соответствующих ветвей, а все промежуточные секвенции из него удалены, что, в силу экономиии ресурсов, существенно уже и для машинных реализаций вывода.