DOI: 10.18721/JCSTCS.10104
УДК 004.832.32

ЭФФЕКТИВНАЯ ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ОБРАТНОГО МЕТОДА МАСЛОВА ДЛЯ ИНТУИЦИОНИСТСКОЙ ЛОГИКИ

В.А. Павлов

Санкт-Петербургский политехнический университет Петра Великого
Санкт-Петербург, Российская Федерация

Изложены результаты исследования, посвящённого применению обратного метода Маслова для автоматизации логического вывода в интуиционистских логических исчислениях. Подробно рассмотрены адаптированные стратегии поиска вывода для интуиционистских исчислений обратного метода, а также новые стратегии, позволяющие ограничить возникающие переборы и уменьшить размер пространства поиска вывода. Также описаны детали реализации разработанной автором компьютерной программы WhaleProver, использующей обратный метод для логического вывода в интуиционистской логике первого порядка. Приведены результаты экспериментального сравнения предложенных стратегий друг с другом и новые результаты сравнения программы WhaleProver с существующими аналогами. Программа не уступает в эффективности лучшим из существующих аналогов (ileanCoP, Imogen) и даже позволяет решить новые задачи из библиотеки ILTP. Таким образом, на практике подтверждается, что программная реализация обратного метода может быть не менее эффективной, чем реализации других методов автоматического логического вывода (в частности, табличных методов).

Ключевые слова: автоматическое доказательство теорем; логический вывод; обратный метод; метод Маслова; интуиционистская логика.

EFFICIENT IMPLEMENTATION OF THE INVERSE METHOD FOR FIRST-ORDER INTUITIONISTIC LOGIC

V.A. Pavlov

Peter the Great St. Petersburg Polytechnic University
St. Petersburg, Russian Federation

This paper contains the results of the research related to Maslov's inverse method and its application to first-order intuitionistic logic. Several proof search strategies for the inverse-method intuitionistic calculus, which allow reducing the proof search space and avoiding redundant inferences, are proposed and explained in detail. Some strategies are new, others are adapted variants of the existing strategies for classical logic. This article includes a detailed description of the automated theorem-proving system WhaleProver for first-order intuitionistic logic, which is based on the inverse method. This article also describes the experimental comparison of the proposed proof search strategies and comparison of WhaleProver with other first-order intuitionistic logic provers. For problems from the ILTP benchmark library, WhaleProver shows comparable results with state-of-the-art intuitionistic provers (ileanCoP, Imogen). Moreover, WhaleProver solves new problems from ILTP, in comparison with all other provers. The results of the study show that inverse method implementation can be at least as efficient as state-of-the-art implementations of other theorem-proving methods (e.g., tableaux methods).

Keywords: automated theorem proving; sequent calculus; inverse method; intuitionistic logic; ILTP.

Сегодня системы *автоматического доказательства теорем* (*АДТ*) успешно применяются при решении ряда научных и технических задач, которые традиционно считаются интеллектуальными: доказательство математических теорем, верификация аппаратного, программного обеспечения и сетевых протоколов, синтез ПО и т. д.

В интерактивных системах АДТ («proof assistants»), таких как Coq, Agda и Nuprl, важную роль играют конструктивные теории [14], позволяющие из доказательства существования некоторого объекта извлечь его построение. Одной из таких теорий является *интуиционистская логика* первого порядка, в которой стандартные пропозициональные связки и кванторы интерпретируются отличным от классической логики образом. Так, в интуиционистском случае из доказательства $\exists x\, P(x)$ можно извлечь конкретный пример $x = a$ и доказательство соответствующего утверждения $P(a)$. В интуиционистской логике неприемлемы классические законы исключённого третьего $A \vee \neg A$ и снятия двойного отрицания $(\neg\neg A) \supset A$.

С помощью программ Coq и Nuprl можно решать сложные задачи за счёт механизма тактик, включающих автоматизированные тактики для интуиционистской логики первого порядка. Например, автоматизированная система АДТ JProver [26] встроена в Coq, Nuprl и MetaPRL. Однако существующие интуиционистские тактики трудно назвать очень эффективными (см. результаты программы JProver на задачах из ILTP [29]).

Целью работы, результаты которой излагаются в настоящей статье, является разработка автоматизированной системы АДТ, позволяющей решать новые задачи по сравнению с JProver и с другими известными системами АДТ для интуиционистской логики первого порядка. Для апробации и оценки эффективности программы используется библиотека ILTP [25, 29], которая содержит обширную коллекцию задач (более 2500) для тестирования и сравнения систем АДТ для интуиционистской логики.

Для решения поставленной задачи был выбран обратный метод Маслова [4] (см. также [15, 22] и дополнение к [13]). Основные особенности этого метода логического вывода — поиск вывода в направлении «сверху вниз», т. е. от аксиом к конечной гипотезе, и использование *свойства подформульности* логических исчислений для ограничения числа аксиом.

Для реализации эффективного алгоритма поиска вывода на основе обратного метода необходимо использовать стратегии поиска вывода. Различные стратегии для обратного метода рассмотрены в работах [5—7, 11, 15, 20, 21, 28, 30]. Более подробный обзор литературы можно найти в статьях [12, 15, 18].

Далее рассматривается новое исчисление обратного метода $\mathbf{IM}^{\xi}_{\mathrm{inv}}$, описывается, как для этого исчисления и исчисления $\mathbf{I}^{\xi}_{\mathrm{inv}}$ из [15] можно адаптировать стратегии, предложенные разными авторами для других исчислений обратного метода, предлагаются новые стратегии. Рассматривается пример доказательства теоремы обратным методом, описываются результаты экспериментов с разработанной программной реализацией обратного метода для интуиционистской логики.

Предполагается знакомство читателя с математической логикой и основами теории доказательств [3, 9, 13]. С результатами предыдущих этапов данной работы можно ознакомиться в статьях [12, 24].

Основные определения и используемые обозначения

В статье используется стандартный язык логики первого порядка, с символами логических операций $\neg$, $\vee$, $\wedge$, $\supset$; кванторами

$\forall$ и $\exists$; предикатными символами P, Q, R; переменными x, y, z и т. д.; символами для произвольных термов r, s, t; символами для произвольных формул A, B, C и т. д.

В данном разделе стандартные определения опускаются, более подробное изложение терминологии см. в [12].

Мультимножество — обобщение понятия множества, допускающее включение нескольких экземпляров одного и того же элемента. Определяется однозначно с точностью до порядка следования формул.

Произвольные мультимножества (возможно, пустые) обозначаются буквами Γ и Δ (возможно, с подстрочными индексами или со штрихами). Запись вида Γ, Δ (или Γ, A) обозначает мультимножество, получающееся объединением элементов Γ с элементами Δ (соответственно с формулой A).

Под *выражением* в данной работе понимается терм, формула, подстановка, мультимножество формул, секвенция и т. д.

Запись $var(E)$ обозначает множество всех переменных выражения E, а $free(E)$ — множество всех его свободных переменных.

Для подстановки $\theta = \{x_1/t_1, ..., x_n/t_n\}$ запись $dom(\theta)$ обозначает множество переменных $\{x_1, ..., x_n\}$, а $vran(\theta)$ — множество всех переменных, входящих в термы $t_1, ..., t_n$. Символ ε обозначает пустую подстановку.

Сужение подстановки $\theta = \{x_1/t_1, ..., x_n/t_n\}$ на множество переменных Ω — это подстановка, обозначаемая как $\theta|_{\Omega}$ и содержащая те и только те пары x_i/t_i из θ, для которых $x_i \in \Omega$. *Запись* θ_{-x} обозначает подстановку $\theta|_{dom(\theta)\setminus\{x\}}$.

Если E — выражение и θ – подстановка, допустимая для E, то запись $E{\cdot}\theta$ является сокращением для $E(\theta|_{free(E)})$.

Переименование — это подстановка, представляющая собой отображение один к одному из своей области определения в неё же.

Наиболее общий унификатор подстановок σ_1 и σ_2 — это наиболее общий унификатор упорядоченных наборов $(x_1\sigma_1, ..., x_n\sigma_1)$ и $(x_1\sigma_2, ..., x_n\sigma_2)$, где $\{x_1, ..., x_n\} = dom(\sigma_1) \cup dom(\sigma_2)$.

Вхождения подформул[1] в формулу F и *полярности* этих вхождений определяются стандартным образом (см., например [4, с. 31]).

Типы вхождений связок и кванторов в формулу определяются так же, как и в статье [10]. Пусть символ $\odot$ обозначает любой из символов $\neg$, $\wedge$, $\vee$, $\supset$, $\forall$ и $\exists$. Вхождение символа $\odot$ в формулу F будем называть вхождением типа $\odot^+$ (типа $\odot^-$), если оно является положительным (соответственно отрицательным) вхождением в F.

Очищенная формула — формула F, в которой все кванторы связывают разные переменные и ни одна связанная переменная не входит свободно в F.

Соглашение 1. В дальнейшем изложении ξ обозначает замкнутую очищенную формулу логики первого порядка, которая содержит только связки $\neg$, $\wedge$, $\vee$, $\supset$ и кванторы $\forall$ и $\exists$[2].

Определяемые ниже понятия зависят от конкретной формулы ξ, поэтому в названиях терминов явно указан символ ξ[3].

ξ-атом — это вхождение атомарной подформулы в ξ.

ξ-секвенция — это секвенция специального вида: $A_1\sigma_1, ..., A_n\sigma_n \vdash B_1\delta_1, ..., B_m\delta_m$, где $A_i, i \in \{1, ..., n\}$ — отрицательные вхождения подформул в ξ; $B_j, j \in \{1, ..., m\}$ — положительные вхождения подформул в ξ; $\sigma_i, i \in \{1, ..., n\}$ и $\delta_j, j \in \{1, ..., m\}$ — подстановки произвольных термов вместо переменных. В статье термин «секвенция» может использоваться вместо термина «ξ-секвенция», если тип секвенции понятен из контекста.

---

[1] В предыдущих статьях [12] и [24] вместо понятия вхождения подформулы использовалось понятие свободной подформулы, как в [15]. Однако впоследствии выяснилось, что изложение в терминах вхождений подформул является более корректным, а также позволяет избежать ряда типичных ошибок.

[2] Произвольную замкнутую формулу нетрудно привести к указанному виду.

[3] Таким образом, используется стиль наименования терминов, характерный для работ С.Ю. Маслова.

Сами формулы вида $\psi \cdot \sigma$, где ψ – вхождение подформулы в ξ, а σ – подстановка, будем называть ξ-*формулами*. Таким образом, антецедент и сукцедент ξ-секвенции представляют собой мультимножества ξ-формул.

Пусть $S = A_1\sigma_1, ..., A_n\sigma_n \vdash B_1\delta_1, ..., B_m\delta_m$ – ξ-секвенция. Тогда запись вида $S\theta$, где θ – подстановка, обозначает ξ-секвенцию $A_1(\sigma_1\theta), ..., A_n(\sigma_n\theta) \vdash B_1(\delta_1\theta), ..., B_m(\delta_m\theta)$, а запись $free(S)$ обозначает *множество свободных переменных* ξ-*секвенции* S: $free(A_1\sigma_1) \cup ... \cup free(A_n\sigma_n) \cup free(B_1\delta_1) \cup ...$
$... \cup free(B_m\delta_m)$.

Запись $norm(S)$ обозначает *нормальную форму* ξ-секвенции S, которая получается из S сужением областей определения всех входящих в неё подстановок: каждая ξ-формула $\psi \cdot \sigma$ заменяется ξ-формулой $\psi\,\sigma|_{free(\psi)}$.

Пусть S и S' – ξ-секвенции, $norm(S) = (\Gamma \vdash \Delta)$ и $norm(S') = (\Gamma' \vdash \Delta')$. Тогда запись $S \dot\subseteq S'$ означает, что $\Gamma \subseteq \Gamma'$ и $\Delta \subseteq \Delta'$. В этом случае будем говорить, что S является *подсеквенцией* S'.

ξ-формула $\psi \cdot \sigma$ называется *правильной*, если $dom(\sigma) = free(\psi)$ и $vran(\sigma) \cap var(\xi) = \varnothing$. ξ-секвенция S называется *правильной*, если каждая входящая в неё ξ-формула является правильной.

Таблица 1

Исчисление $\mathbf{IM}^{\xi}_{\mathrm{inv}}$

$P_1 \cdot \rho_1\tau \vdash P_2 \cdot \rho_2\tau$	(Px)		
$\dfrac{\Gamma, A_1 \cdot \sigma \vdash \Delta}{\Gamma, A_1 \wedge A_2 \cdot \sigma \vdash \Delta}$	$(L\wedge_1)$	$\dfrac{\Gamma, A_2 \cdot \sigma \vdash \Delta}{\Gamma, A_1 \wedge A_2 \cdot \sigma \vdash \Delta}$	$(L\wedge_2)$
$\dfrac{\Gamma_1 \vdash \Delta_1, A_1 \cdot \sigma_1 \quad \Gamma_2 \vdash \Delta_2, A_2 \cdot \sigma_2}{\Gamma_1\theta, \Gamma_2\theta \vdash \Delta_1\theta, \Delta_2\theta, A_1 \wedge A_2 \cdot \sigma_1\theta}$	$(R\wedge)$	$\dfrac{\Gamma_1, A_1 \cdot \sigma_1 \vdash \Delta_1 \quad \Gamma_2, A_2 \cdot \sigma_2 \vdash \Delta_2}{\Gamma_1\theta, \Gamma_2\theta, A_1 \vee A_2 \cdot \sigma_1\theta \vdash \Delta_1\theta, \Delta_2\theta}$	$(L\vee)$
$\dfrac{\Gamma \vdash \Delta, A_1 \cdot \sigma}{\Gamma \vdash \Delta, A_1 \vee A_2 \cdot \sigma}$	$(R\vee_1)$	$\dfrac{\Gamma \vdash \Delta, A_2 \cdot \sigma}{\Gamma \vdash \Delta, A_1 \vee A_2 \cdot \sigma}$	$(R\vee_2)$
$\dfrac{\Gamma_1 \vdash \Delta_1, A_1 \cdot \sigma_1 \quad \Gamma_2, A_2 \cdot \sigma_2 \vdash \Delta_2}{\Gamma_1\theta, \Gamma_2\theta, A_1 \supset A_2 \cdot \sigma_1\theta \vdash \Delta_1\theta, \Delta_2\theta}$	$(L\supset)$	$\dfrac{\Gamma, A_1 \cdot \sigma \vdash}{\Gamma \vdash A_1 \supset A_2 \cdot \sigma}$	$(R\supset_1)$
$\dfrac{\Gamma \vdash \Delta, A_2 \cdot \sigma}{\Gamma \vdash \Delta, A_1 \supset A_2 \cdot \sigma}$	$(R\supset_2)$	$\dfrac{\Gamma, A_1 \cdot \sigma_1 \vdash A_1 \supset A_2 \cdot \sigma_2}{\Gamma\theta \vdash A_1 \supset A_2 \cdot \sigma_2\theta}$	$(R\supset_3)$
$\dfrac{\Gamma \vdash \Delta, A \cdot \sigma}{\Gamma, \neg A \cdot \sigma \vdash \Delta}$	$(L\neg)$	$\dfrac{\Gamma, A \cdot \sigma \vdash}{\Gamma \vdash \neg A \cdot \sigma}$	$(R\neg)$
$\dfrac{\Gamma, A \cdot \sigma \vdash \Delta}{\Gamma, \forall x A \cdot \sigma_{-x} \vdash \Delta}$	$(L\forall)$	$\dfrac{\Gamma \vdash A \cdot \sigma}{\Gamma \vdash \forall x A \cdot \sigma_{-x}}$	$(R\forall)$
$\dfrac{\Gamma, A \cdot \sigma \vdash \Delta}{\Gamma, \exists x A \cdot \sigma_{-x} \vdash \Delta}$	$(L\exists)$	$\dfrac{\Gamma \vdash \Delta, A \cdot \sigma}{\Gamma \vdash \Delta, \exists x A \cdot \sigma_{-x}}$	$(R\exists)$
$\dfrac{\Gamma, A \cdot \sigma_1, A \cdot \sigma_2 \vdash \Delta}{\Gamma\theta, A \cdot \sigma_1\theta \vdash \Delta\theta}$	(LC)	$\dfrac{\Gamma \vdash \Delta, A \cdot \sigma_1, A \cdot \sigma_2}{\Gamma\theta \vdash \Delta\theta, A \cdot \sigma_1\theta}$	(RC)
$\dfrac{S}{S\rho}$	(Rn)	$\dfrac{S}{norm(S)}$	(Nrm)

Исчисление обратного метода для интуиционистской логики

Секвенциальное исчисление $\mathbf{IM}^{\xi}_{\text{inv}}$ строится индивидуально для каждой доказываемой формулы ξ. Аксиомы и правила вывода исчисления представлены в табл. 1. Во всех правилах, кроме (Rn) и (Nrm), посылки являются правильными ξ-секвенциями и не имеют общих свободных переменных друг с другом. В схеме аксиом (Px): P_1, P_2 — ξ-атомы; ρ_1, ρ_2 — такие переименования, что $free(P_1\rho_1) \cap free(P_2\rho_2) = \varnothing$; τ — наиболее общий унификатор формул $P_1\rho_1$ и $P_2\rho_2$. Подстановка θ — наиболее общий унификатор подстановок σ_1 и σ_2. В правиле (Rn) S — это ξ-секвенция, а ρ — переименование. В правиле (Nrm) S — ξ-секвенция. В правилах $(L\exists)$ и $(R\forall)$ должно выполняться *ограничение на собственную переменную*: $x\sigma$ – это переменная, которая не входит *свободно* в заключение.

Структурные правила (LC) и (RC) называются *правилами сокращения*, или просто *сокращением*. Структурные правила (Rn) и (Nrm) называются *правилами переименования* и *нормализации* соответственно. Само исчисление получено с помощью стандартной методики из [8, 15].

Будем говорить, что формула ξ выводима в исчислении $\mathbf{IM}^{\xi}_{\text{inv}}$ или $\mathbf{I}^{\xi}_{\text{inv}}$, если в этом исчислении выводима секвенция $(\vdash \xi \cdot \varepsilon)$.

Из любой ξ-секвенции в исчислении $\mathbf{IM}^{\xi}_{\text{inv}}$ можно получить правильную ξ-секвенцию применением правил (Rn) и (Nrm), которые мы будем называть *дополнительными правилами* исчисления. Все остальные правила вывода и схему аксиом (Px) будем называть *основными правилами*.

Пусть S — правильная ξ-секвенция, а Π — вывод S в исчислении $\mathbf{IM}^{\xi}_{\text{inv}}$. Π будем называть *правильным выводом*, если выполняются условия:

- применение правила (Rn) входит в Π только непосредственно после применения основных правил, применение правила (Nrm) — только непосредственно после применения правила (Rn), а применение основных правил (кроме аксиом) — только непосредственно после применения правила (Nrm);
- если $(S_1/S_1\rho_1)$ и $(S_2/S_2\rho_2)$ — два различных применения правила (Rn) в Π, то множества $free(S_1\rho_1)$ и $free(S_2\rho_2)$ не имеют общих переменных как друг с другом, так и с множеством $var(\xi)$.

Использование правильных выводов обусловлено следующими двумя соображениями. Во-первых, это позволяет гарантировать соблюдение условий на посылки применения основных правил (см. выше). Во-вторых, исключается возможность повторного применения правил (Rn) и (Nrm).

Лемма 1. Любой вывод произвольной ξ-секвенции S в исчислении $\mathbf{IM}^{\xi}_{\text{inv}}$ можно перестроить, сохраняя порядок применения основных правил, в правильный вывод (правильной) ξ-секвенции S', совпадающей с S с точностью до переименования свободных переменных.

$\triangleright$ Указанное перестроение можно выполнить, добавляя, где нужно, применение правил (Rn) и (Nrm), при необходимости заменяя такое применение этих правил, которое не удовлетворяет условиям из определения правильного вывода, а также удаляя «лишнее» применение. $\triangleleft$

Следствие. Вывод формулы ξ в $\mathbf{IM}^{\xi}_{\text{inv}}$ можно перестроить в правильный вывод этой формулы, сохраняя порядок применения основных правил.

Лемма 2. Правильный вывод в исчислении $\mathbf{IM}^{\xi}_{\text{inv}}$ обладает свойством чистоты переменных[4].

$\triangleright$ Следует из соглашения 1 и определения правильного вывода. $\triangleleft$

Следующая теорема утверждает равнообъёмность исчисления $\mathbf{IM}^{\xi}_{\text{inv}}$ и исчисления GHPC из статьи А.Г. Драгалина [1].

Теорема 1. Пусть ξ замкнутая формула, соответствующая соглашению 1, а ξ' получается из ξ заменой каждого вхождения подформулы вида $\neg A$ формулой $A \supset \perp$. ξ-секвенция $(\vdash \xi \cdot \varepsilon)$ выводима в исчислении $\mathbf{IM}^{\xi}_{\text{inv}}$ тогда и только тогда, когда секвенция $(\vdash \xi')$ выводима в GHPC.

$\triangleright$ Доказательство строится аналогично приведённому в [15] доказательству полно-

[4] Определение доказательства с чистыми переменными см. в [2] или в [3, § 55].

ты и непротиворечивости исчисления $\mathbf{I}^{\xi}_{\mathrm{inv}}$. ⊲

Стратегии поиска вывода

Дополним исчисление $\mathbf{IM}^{\xi}_{\mathrm{inv}}$ стратегиями поиска вывода, которые позволят избавиться от ряда избыточных секвенций и применения правил вывода. Одновременно будем рассматривать возможность применения стратегий к исчислению $\mathbf{I}^{\xi}_{\mathrm{inv}}$ из [15], в котором секвенции содержат не более одной формулы в сукцеденте.

Стратегию X будем называть *полной* для секвенциального исчисления Y, если применение стратегии X не приводит к уменьшению множества выводимых в этом исчислении секвенций.

Пусть S и S' – ξ-секвенции. Будем говорить, что S' *поглощает* S ($S \prec S'$), если существует такая подстановка σ, что $S'\sigma \dot{\subseteq} S$.

Стратегия поглощения для $\mathbf{IM}^{\xi}_{\mathrm{inv}}$ ($\mathbf{I}^{\xi}_{\mathrm{inv}}$) определяется стандартным образом: если в процессе поиска вывода получены две такие секвенции S_1 и S_2, что S_1 поглощает S_2, то секвенцию S_2 разрешается удалить.

Т е о р е м а 2. Стратегия поглощения является полной для $\mathbf{IM}^{\xi}_{\mathrm{inv}}$ и $\mathbf{I}^{\xi}_{\mathrm{inv}}$.

⊳ Индукцией по построению вывода можно показать[5], что если $S_1 \prec S_2$ и существует вывод формулы ξ из секвенций $S_1, S_2, ..., S_k$, то также существует вывод ξ из $S_2, ..., S_k$. ⊲

Теперь рассмотрим некоторые возможности перестановки применения правил в исчислении $\mathbf{IM}^{\xi}_{\mathrm{inv}}$. Будем использовать терминологию из статьи С. Клини [2]. Будем говорить, что применение правила R_1 *непосредственно предшествует* применению правила R_2, если между ними расположено разве лишь применение структурных правил. При этом такое применение правил R_1 и R_2 будем называть *смежным*.

Пусть список $\mathfrak{R}^{\xi}$ состоит из следующих правил исчисления $\mathbf{IM}^{\xi}_{\mathrm{inv}}$: $(L\wedge_1)$, $(L\wedge_2)$, $(R\vee_1)$, $(R\vee_2)$, $(L\neg)$, $(R\supset_2)$, $(L\forall)$, $(R\exists)$ и $(L\exists)$, а также $(R\forall)$ в случае, если ξ не содержит отрицательных вхождений символа $\vee$.

Л е м м а 3. Рассмотрим вывод Π в исчислении $\mathbf{IM}^{\xi}_{\mathrm{inv}}$, обладающий свойством чистоты переменных, и в нём два смежных применения правил R_1 и R_2 (R_1 расположено над R_2), принадлежащие различным вхождениям формул в заключение R_2. Тогда если правило R_2 входит в список $\mathfrak{R}^{\xi}$, то эти два применения можно переставить (с сохранением свойства чистоты переменных), за исключением случаев: $(L\forall)/(L\exists)$, $(R\exists)/(L\exists)$ и $(L\forall)/(R\forall)$.

В указанных трёх случаях перестановку всё же можно выполнить при условии, что собственная переменная $x\sigma$ применения правила R_2 не входит в терм $x\sigma$ применения правила R_1.

⊳ Перестановочность правил удобно сначала доказать для промежуточного исчисления без унификации $\mathbf{IM}_{\mathrm{inv}}$ (это исчисление в статье не приводится, но его правила нетрудно восстановить по правилам исчисления $\mathbf{IM}^{\xi}_{\mathrm{inv}}$, см. для примера [15]). Затем этот результат переносится на исчисление $\mathbf{IM}^{\xi}_{\mathrm{inv}}$, за счёт эквивалентности выводов в исчислениях $\mathbf{IM}_{\mathrm{inv}}$ и $\mathbf{IM}^{\xi}_{\mathrm{inv}}$ с точностью до применения структурных правил.

Для $\mathbf{IM}_{\mathrm{inv}}$ лемма доказывается аналогично лемме 7 из статьи [2]. Основное отличие возникает при рассмотрении правила $(R\supset_2)$, которое можно поднимать над применением правил $(L\neg)$ и $(L\supset)$, т. к. в этом правиле не накладывается ограничение на число формул в сукцеденте. ⊲

Т е о р е м а 3. Произвольный вывод Π формулы ξ в исчислении $\mathbf{IM}^{\xi}_{\mathrm{inv}}$ можно перестроить (без нарушения свойства чистоты переменных, если оно имелось), поднимая наверх любое применение правила R из списка $\mathfrak{R}^{\xi}$ так, чтобы оно входило в новый вывод только в следующих случаях:

1) применению R непосредственно предшествует применение аксиомы;

2) боковая формула применения правила R вводится непосредственно предшествующим ему применением неструктурного правила вывода;

3) пара R'/R, где применение правила R' непосредственно предшествует применению R, совпадает с $(L\forall)/(L\exists)$, $(R\exists)/(L\exists)$ или $(L\forall)/(R\forall)$, или причём терм $x\sigma$ при-

[5] См. доказательства теорем 2 и 3 из [30].

менения правила R' содержит собственную переменную применения правила R.

⊳ Не умаляя общности, можно считать, что Π обладает свойством чистоты переменных (иначе переименуем входящие в вывод переменные, см. [3, § 55]). Тогда можно применять лемму 3. Доказательство несущественно отличается от доказательства теоремы 2 из статьи [2]. ⊲

Лемма 4. В условиях теоремы 3 вывод Π можно перестроить так, чтобы каждое применение правила R, совпадающего с $(L\exists)$ или $(R\forall)$ из списка $\mathfrak{R}^{\xi}$, входило в новый вывод только таким образом, чтобы нигде выше это правило было неприменимо (за исключением применения правила $(R\forall)$, боковой формулой которого является формула вида $(A \supset B)\cdot\sigma$). Свойство чистоты переменных можно сохранить, если оно имелось.

Лемма 4 открывает дополнительные возможности подъёма применения правил $(L\exists)$ и $(R\forall)$, однако это уже не перестановка по Клини, т. к. может потребоваться удвоение применения правила $(L\exists)$, если оно поднимается над применением двухпосылочных правил.

Из теоремы 3 и леммы 4 следует полнота *стратегии упрощения*, которая заключается в следующем. Если в процессе поиска вывода получена секвенция S, к которой применимо какое-либо правило R из списка $\mathfrak{R}^{\xi}$, то S заменяется секвенцией, стоящей в заключении правила R (будем говорить, что S упрощается), кроме случая, когда боковой формулой применения R является формула вида $(A \supset B)\cdot\sigma$ и R является одним из правил $(R\vee_1)$, $(R\vee_2)$, $(L\neg)$, $(R\supset_2)$, $(R\exists)$ или $(R\forall)$. После каждой такой замены к новой секвенции применяются правила переименования и нормализации таким образом, как указано в определении правильного вывода. Так секвенцию S можно упрощать до тех пор, пока не получим секвенцию S', которую далее не упростить. Такую секвенцию S' будем называть *упрощением* секвенции S.

Для исчисления $\mathbf{I}^{\xi}_{\mathrm{inv}}$ из [15] применима аналогичная стратегия, которая в статье Т. Таммета [28] названа «reduction strategy». Для классической логики вариант такой же стратегии сформулировал А. Воронков [30].

Теорема 4. Стратегия упрощения является полной для $\mathbf{IM}^{\xi}_{\mathrm{inv}}$ и $\mathbf{I}^{\xi}_{\mathrm{inv}}$.

⊳ Для исчисления $\mathbf{IM}^{\xi}_{\mathrm{inv}}$ следует из теоремы 3 и леммы 4. Для $\mathbf{I}^{\xi}_{\mathrm{inv}}$ можно доказать аналогичную теорему о перестановочности применения правил. ⊲

Рассмотрим следующую *стратегию удаления секвенций с недопустимыми подстановками* (*УСНП*) для исчислений $\mathbf{IM}^{\xi}_{\mathrm{inv}}$ и $\mathbf{I}^{\xi}_{\mathrm{inv}}$. Эта стратегия включает три множества секвенций $\boldsymbol{U\_subs}_1$, $\boldsymbol{U\_subs}_2$ и $\boldsymbol{U\_subs}_3$, соответствующие частным случаям недопустимых систем зависимостей [4], [11], которые можно достаточно быстро идентифицировать.

Будем называть переменную $x \in var(\xi)$ *параметром* ξ, если x связана квантором типа $\forall^+$ или $\exists^-$.

$\boldsymbol{U\_subs}_1$ состоит из секвенций, содержащих такие ξ-формулы $\psi_1\cdot\sigma_1$ и $\psi_2\cdot\sigma_2$ (не обязательно различные), что $\exists x_1 \in free(\psi_1), x_2 \in free(\psi_2), x_1 \neq x_2$: $x_1\sigma_1 = x_2\sigma_2$ и x_i, $i \in \{1,2\}$ − параметр ξ.

Множество $\boldsymbol{U\_subs}_2$ состоит из секвенций, содержащих такую формулу $\psi\cdot\sigma$, что $\exists x \in free(\psi)$: x − параметр ξ и $x\sigma$ не является переменной, т. е. является предметной константой или составным термом вида $f(t_1, ..., t_n)$.

Пусть x − параметр ξ и $x' \in var(\xi)$. Переменная x *зависима* от x', если $x \neq x'$ и квантор, связывающий x, находится в зоне действия квантора, связывающего x'. Для секвенции S и переменных y, $y' \in free(S)$ запись $y \prec y'$ означает, что существует ξ-формула $\psi\cdot\sigma$, входящая в S, и такие переменные x, $x' \in free(\psi)$, что x − параметр ξ, x зависима от x', $x\sigma = y$ и $x'\sigma = y'$.

Множество $\boldsymbol{U\_subs}_3$ состоит из секвенций S, для которых существует такая цепочка переменных $y_1, y_2, ..., y_k \in free(S), k \geq 1$, что $y_1 \prec y_2 \prec ... \prec y_k \prec y_1$.

Вхождение подформулы ψ в формулу ξ будем называть *строго положительным*[6], если оно является положительным вхождением в ξ и не принадлежит никакому отри-

[6] В статье [10] используется термин «существенно положительное вхождение».

цательному вхождению подформулы в ξ.

Пусть ψ и ψ' – вхождения подформул в ξ. Тогда запись $\psi \in \psi'$ означает, что ψ принадлежит ψ'. Будем говорить, что ψ *критическим образом принадлежит* ψ' и писать $\psi \mathrel{\bar{\in}} \psi'$, если $\psi \in \psi'$ и существует такое положительное вхождение в ξ подформулы ψ'' вида $A \supset B$, $\neg A$ или $\forall x\, A$, что $\psi'' \in \psi'$ и $\psi \in A$.

Теперь рассмотрим новую *стратегию удаления секвенций с недопустимыми формулами* (стратегия *УСНФ*) для исчислений $\mathbf{IM}^{\xi}_{\text{inv}}$ и $\mathbf{I}^{\xi}_{\text{inv}}$. Эта стратегия включает два множества секвенций $\boldsymbol{U\_form}_1$ и $\boldsymbol{U\_form}_2$.

Множество $\boldsymbol{U\_form}_1$ для $\mathbf{IM}^{\xi}_{\text{inv}}$ и $\mathbf{I}^{\xi}_{\text{inv}}$ определяется одинаково. Пусть S – ξ-секвенция. $S \in \boldsymbol{U\_form}_1$, если найдётся такая пара ξ-формул $\psi_1 \cdot \sigma_1$ и $\psi_2 \cdot \sigma_2$ из S, что $\psi_2 \notin \psi_1$ (в частности, ψ_2 не совпадает с ψ_1), вхождение в ξ ближайшей общей надформулы ψ_1 и ψ_2 (обозначим его ψ^*) является строго положительным, и выполняется хотя бы одно из следующих условий:

1. ψ^* имеет вид $A \supset B$ и совпадает с ψ_2, при этом $\psi_1 \mathrel{\bar{\in}} A$ или $\psi_1 \mathrel{\bar{\in}} B$ [7];
2. ψ^* имеет вид $A \supset B$, $\psi_1 \mathrel{\bar{\in}} A$, $\psi_2 \in B$ и ψ_2 – строго положительное вхождение в ξ;
3. ψ^* не имеет вид $A \supset B$, , при этом $\psi_1 \mathrel{\bar{\in}} \psi^*$ или $\psi_2 \mathrel{\bar{\in}} \psi^*$.

Рассмотрим множество $\boldsymbol{U\_form}_2$ для $\mathbf{IM}^{\xi}_{\text{inv}}$. $S \in \boldsymbol{U\_form}_2$, если найдётся такая пара ξ-формул $\psi_1 \cdot \sigma_1$ и $\psi_2 \cdot \sigma_2$ из сукцедента S, что $\psi_i \mathrel{\bar{\in}} \chi_i$, $i \in \{1, 2\}$, где χ_i – ближайшее отрицательное вхождение в ξ надформулы ψ_i, такое что $\chi_i \in \psi^*$ (если такого вхождения не существует, то χ_i совпадает с ψ^*). Как и выше, ψ^* обозначает вхождение в ξ ближайшей общей надформулы ψ_1 и ψ_2.

Множество $\boldsymbol{U\_form}_2$ для $\mathbf{I}^{\xi}_{\text{inv}}$ пусто.

Можно показать, что секвенции из множеств $\boldsymbol{U\_form}_1$ и $\boldsymbol{U\_form}_2$ не могут участвовать в выводе формулы ξ в исчислениях $\mathbf{IM}^{\xi}_{\text{inv}}$ и $\mathbf{I}^{\xi}_{\text{inv}}$.

По аналогии с тактикой перехода к тривиальным спецификациям [7] можно предложить следующую *стратегию тривиального сокращения*: если в процессе поиска вывода получена секвенция S, для которой существует такое сокращение S', что $S \prec S'$, то S можно заменить на S'.

Заметим, что любой вывод в исчислении $\mathbf{I}^{\xi}_{\text{inv}}$ можно перестроить в вывод в исчислении $\mathbf{IM}^{\xi}_{\text{inv}}$. Поэтому в качестве самостоятельной стратегии для исчисления $\mathbf{IM}^{\xi}_{\text{inv}}$ можно рассматривать построение вывода формулы ξ в исчислении $\mathbf{I}^{\xi}_{\text{inv}}$ (со стратегиями для $\mathbf{I}^{\xi}_{\text{inv}}$) с последующим получением из него вывода в $\mathbf{IM}^{\xi}_{\text{inv}}$. Такую стратегию назовём *сингулярной*[8] *стратегией*.

Теорема 5. Любая комбинация следующих стратегий является полной для исчислений $\mathbf{IM}^{\xi}_{\text{inv}}$ и $\mathbf{I}^{\xi}_{\text{inv}}$: стратегия поглощения, стратегия упрощения, стратегии УСНП и УСНФ, стратегия тривиального сокращения.

▹ Стратегии УСНП и УСНФ совместимы с другими стратегиями, т. к. при их использовании удаляются только такие секвенции, которые не могут участвовать в выводе формулы ξ. Тривиальное сокращение представляет собой частный случай применения поглощения. Стратегии поглощения и упрощения также совместимы, поскольку отношение поглощения инвариантно относительно упрощения секвенций. ◃

Следствие. Сингулярная стратегия совместно с любой комбинацией стратегий для $\mathbf{I}^{\xi}_{\text{inv}}$, перечисленных в теореме 5, является полной для $\mathbf{IM}^{\xi}_{\text{inv}}$.

Пример решения задачи из ILTP обратным методом Маслова

Поясним принцип доказательства теорем обратным методом и применение некоторых стратегий на примере задачи SYN408+1 из библиотеки ILTP версии 1.1.2. Необходимо вывести формулу:

$$\xi = (\neg \exists x\; P(x)) \supset (\forall y\; (P(y) \supset Q(y))).$$

[7] Напомним, что запись вида $\psi \mathrel{\bar{\in}} \psi'$ означает, что ψ критическим образом принадлежит ψ'.

[8] Название стратегии связано с термином «сингулярная секвенция» – это секвенция, в сукцеденте которой находится не более одной формулы. Использование сингулярных секвенций может помочь сузить пространство поиска вывода.

Будем строить вывод в исчислении $\mathbf{IM}^{\xi}_{\text{inv}}$ с применением стратегий, описанных выше. Начнём доказательство с применения схемы аксиом (*Px*). Заметим, что формула ξ содержит только два вхождения атомарных подформул $P(x)$ и $P(y)$, первое из которых является положительным вхождением, а второе – отрицательным. Поэтому по правилу (*Px*) получаем единственную (с точностью до переименования переменных) аксиому:

$$P(y)\cdot\{y/v_1\}\vdash P(x)\cdot\{x/v_1\}$$
$$[(Px)]$$

В секвенции 1 v_1 – это новая переменная. К данной секвенции применима стратегия упрощения с применением правила $(R\exists)$. Поэтому заменим секвенцию 1 заключением применения этого правила:

$$P(y)\cdot\{y/v_1\}\vdash \exists x\, P(x)\cdot\varepsilon$$
$$[(Px)+(R\exists)].$$

Эту секвенцию можно дальше упростить с применением правила $(L\neg)$:

$$\neg\exists x\, P(x)\cdot\varepsilon, P(y)\cdot\{y/v_1\}\vdash$$
$$[(Px)+(R\exists)+(L\neg)].$$

Дальнейшие упрощения секвенции 1 невозможны. Применим к этой секвенции правило $(R\supset_1)$:

$$\neg\exists x\, P(x)\cdot\varepsilon\vdash (P(y)\supset Q(y))\cdot\{y/v_1\}$$
$$[(R\supset_1):1].$$

Поскольку формула ξ не содержит отрицательных вхождений $\vee$, то правило $(R\forall)$ входит в список $\Re^{\xi}$. Но так как в сукцеденте секвенции 2 стоит формула с главным символом $\supset$, то эту секвенцию мы не можем упростить с применением $(R\forall)$[9]. Поэтому получаем новую секвенцию по правилу $(R\forall)$. Ограничение на собственную переменную в данном случае выполняется: переменная v_1 не входит свободно в заключение:

$$\neg\exists x\, P(x)\cdot\varepsilon\vdash \forall y\ (P(y)\supset Q(y))\cdot\varepsilon$$
$$[(R\forall):2].$$

Полученную секвенцию уже можно упростить с применением $(R\supset_2)$:

$$\neg\exists x\, P(x)\cdot\varepsilon\vdash (\neg\exists x\ P(x))\supset$$
$$\supset(\forall y\ (P(y)\supset Q(y)))\cdot\varepsilon$$
$$[(R\forall)+(R\supset_2):2].$$

Завершим доказательство, применив $(R\supset_3)$ к секвенции 3 (с $\theta=\varepsilon$):

$$\vdash(\neg\exists x\ P(x))\supset(\forall y\ (P(y)\supset Q(y)))\cdot\varepsilon$$
$$[(R\supset_3):3].$$

Запишем полученный вывод формулы в виде дерева вывода (в данном примере «дерево» состоит из одной ветви), содержащего «укрупнённые» применения правил, в соответствии со стратегией упрощения:

$$\cfrac{\cfrac{\cfrac{\cfrac{}{\neg\exists x\, P(x)\cdot\varepsilon, P(y)\cdot\{y/v_1\}\vdash}\;(Px)+(R\exists)+(L\neg)}{\neg\exists x\, P(x)\cdot\varepsilon\vdash (P(y)\supset Q(y))\cdot\{y/v_1\}}\;(R\supset_1)}{\neg\exists x\, P(x)\cdot\varepsilon\vdash(\neg\exists x\ P(x))\supset(\forall y\ (P(y)\supset Q(y)))\cdot\varepsilon}\;(R\forall)+(R\supset_2)}{\vdash(\neg\exists x\ P(x))\supset(\forall y\ (P(y)\supset Q(y)))\cdot\varepsilon}\;(R\supset_3)$$

Заметим, что кроме секвенций 1–4 мы могли бы вывести и некоторые другие секвенции. Например, применив к (упрощённой) секвенции 1 правило $(R\supset_1)$ с другой боковой формулой, можно получить следующую секвенцию:

$$P(y)\cdot\{y/v_1\}\vdash(\neg\exists x\ P(x))\supset$$
$$\supset(\forall y\ (P(y)\supset Q(y)))\cdot\varepsilon.$$

Однако из этой секвенции нельзя вывести формулу ξ. Действительно, можно заметить, что эта секвенция принадлежит множеству $\boldsymbol{U\_form}_1$, и по стратегии УСНФ её можно удалить.

[9] Вообще говоря, в данном случае такое упрощение допустимо, т. к. ни одна секвенция, выводимая из секвенции 2, не может содержать вхождение $P(y)$ в антецеденте. Нетрудно видоизменить соответствующее условие в стратегии упрощения, чтобы учесть такие случаи.

Программная реализация и результаты экспериментов

Автором статьи разработана программа АДТ WhaleProver для вывода формул в интуиционистских исчислениях $\mathbf{IM}^{\xi}_{\mathrm{inv}}$ и $\mathbf{I}^{\xi}_{\mathrm{inv}}$, в ряде их модификаций, а также в классическом исчислении $\mathbf{C}^{\xi}_{\mathrm{inv}}$ из статьи [15]. Программа написана на C++, в ней используется вариант алгоритма «given clause algorithm» [19], адаптированный для поиска выводов в исчислениях обратного метода (подробнее описание алгоритма см. в [12]).

В программе используются стратегии *прямого* и *обратного поглощения*: при прямом поглощении удаляется вновь порождённая секвенция, если она поглощается какой-либо из выведенных ранее, а при обратном поглощении удаляется одна из порождённых ранее секвенций, если она поглощается новой секвенцией.

Программа WhaleProver использовалась в ряде экспериментов на задачах из библиотеки ILTP [25] версии 1.1.2. Все эксперименты проводились на персональном компьютере с процессором Intel Core i5 3.40 GHz, ОС Windows 7 и 16 Gb ОЗУ (программа запускалась на одном ядре в 32-разрядном адресном пространстве, т. е. фактический объём используемой оперативной памяти не превышал 2 Гб).

В частности, на выборке из 362 задач различной сложности проведено сравнение стратегий для исчисления $\mathbf{IM}^{\xi}_{\mathrm{inv}}$. Основные результаты представлены в табл. 2. Первый столбец содержит номера экспериментов, в следующих шести столбцах знаком «+» отмечены включённые в экспериментах стратегии, в последующих столбцах указаны результаты экспериментов (сложность вывода – это число различных секвенций в дереве вывода, а длина вывода – это число всех секвенций). В таблице стратегиям назначены сокращённые наименования: СИ – сингулярная стратегия, ТС – тривиальное сокращение, УП – стратегия упрощения, УСНП и УСНФ – как раньше (см. выше), ОП – обратное поглощение. В каждом эксперименте использовалось прямое поглощение, чтобы при любой комбинации других стратегий программа могла решить все 362 задачи.

Для расчёта значений суммарного времени работы программы каждый эксперимент повторялся 50 раз, при этом 95 % довери-

Таблица 2

Сравнение стратегий поиска вывода

Эксперимент	СИ	ТС	УП	УСНП	УСНФ	ОП	Средняя высота вывода	Средняя сложность вывода	Средняя длина вывода	Средний размер пространства поиска вывода	Суммарное время, с
1	+						19,4	34,5	85,8	1248,0	136,5
2							19,9	34,9	85,8	1242,9	170,4
3	+	+					17,3	31,5	37,0	1042,5	106,5
4	+		+				6,9	14,3	38,4	189,6	5,6
5	+			+			19,5	34,6	86,0	998,6	73,9
6	+				+		19,5	34,5	85,8	1004,8	93,5
7	+	+	+	+	+		5,2	11,3	14,0	104,4	2,3
8		+	+	+	+		5,2	11,3	13,3	106,2	2,7
9	+	+	+	+	+	+	5,3	11,4	14,0	103,2	2,8
10		+	+	+	+	+	5,2	11,4	13,3	105,2	3,1

тельный интервал не превышал ±3 % от каждого из приведённых в таблице значений.

Эксперименты 3—6 показывают, что стратегия упрощения существенно опережает другие стратегии по основным оцениваемым критериям (время и размер пространства поиска вывода). Новая стратегия УСНФ оказалась по тем же показателям эффективнее стратегии тривиального сокращения.

Сравнение экспериментов 1 и 2, 7 и 8, 9 и 10 показывает, что сингулярная стратегия позволяет уменьшить время решения задач. Эффект от использования этой стратегии максимален, когда другие стратегии отключены, и уменьшается при включении всех остальных стратегий.

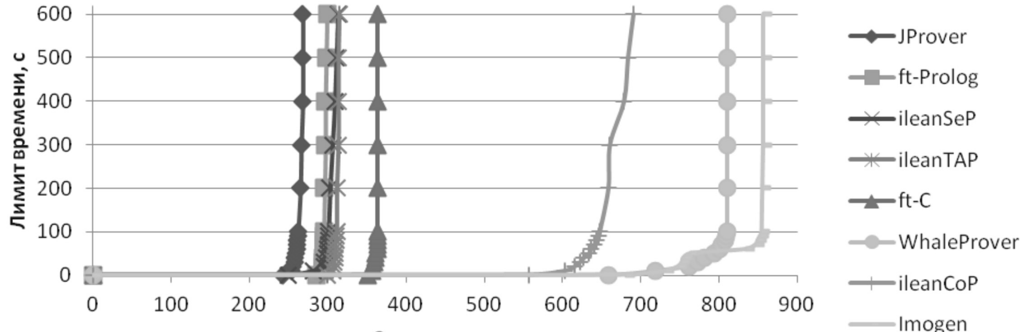
Максимальная эффективность программы достигается при включении всех стратегий (эксперимент 9) или всех, кроме обратного поглощения (эксперимент 7). При этом отключение обратного поглощения позволяет даже уменьшить время решения задач (проверка поглощения является достаточно затратной операцией).

Конфигурация из эксперимента 7 с небольшими дополнительными оптимизациями использовалась при сравнении программы WhaleProver с существующими аналогами на задачах из ILTP. Программа решила 811 задач с лимитом времени 100 с на каждую, что на 183 задачи больше, чем могла решить предыдущая версия программы [12]. Для сравнения, результаты тестирования других программ АДТ для интуиционистской логики первого порядка (с лимитом времени 600 с), опубликованные на сайте ILTP [29], варьируются от 268 (JProver) до 690 (ileanCoP) решённых задач. Существуют и более свежие независимые результаты тестирования других программ на библиотеке ILTP: программа Imogen решила 857 задач [20], программа nanoCoP-i доказала 700 теорем [23], а новая версия программы ileanCoP 1.2 доказала 717 теорем [23]. Таким образом, результаты программы WhaleProver сопоставимы с результатами лучших аналогов.

Отметим, что сравнение с опубликованными на сайте [29] результатами других программ соответствует методике, рекомендуемой признанными экспертами в области АДТ [27]. Суть этой методики наглядно демонстрирует рисунок (зависимость количества решённых задач от лимита времени на каждую задачу). Начиная с некоторой точки, прирост числа решённых задач практически прекращается, поэтому результаты программ существенно не изменятся при линейном увеличении частоты процессора или объёма ОЗУ.

Таким образом, сравнение является корректным, несмотря на то, что результаты разных программ получены на разных вычислительных машинах.

Часть задач из ILTP имеет «интуиционистский статус»: теорема или ложное утверждение. Из 811 задач, решённых программой WhaleProver (621 утверждение доказано и 190 опровергнуто), 632 имеют «интуиционистский статус» (или его можно определить по «классическому статусу»), при этом результат программы соответствует этому статусу (т. е. корректен). В число этих 632 задач входят 70 задач, которые не были решены ни одной из шести программ, официальные результаты которых



Количество решённых задач при увеличении лимита времени представлены на сайте ILTP [29]: JProver, ft-C, ft-Prolog, ileanSeP, ileanTAP, ileanCoP (1.0). Если также учесть неопубликованные данные регрессионного тестирования программы Imogen [16], то программа WhaleProver по сравнению со всеми семью программами решила 16 новых задач с известным статусом. Например, решены следующие задачи, которые содержат ложные (как классически, так и интуиционистски) утверждения: задача из области представления знаний KRS173+1, задачи из области обработки естественного языка NLP198+1 (обработка фрагмента текста про «old dirty white Chevy») и NLP223+1 (обработка фразы «Vincent believes that every man smokes ...»). Кроме того, программа WhaleProver позволяет решить ряд задач значительно быстрее других программ.

В программе WhaleProver также реализован интерактивный режим: перед тем, как воспользоваться тактикой автоматического поиска вывода, пользователь может вручную упростить формулу с помощью простых тактик (подобно тому, как используются тактики в программе Coq) или ввести дополнительные леммы. Такой режим позволяет решить некоторые задачи, которые не удаётся решить в полностью автоматическом режиме.

Заключение

В настоящей статье рассмотрено интуиционистское исчисление обратного метода $\mathbf{IM}^{\xi}_{\mathrm{inv}}$ для вывода формул логики первого порядка, которое отличается от существующих исчислений обратного метода использованием мультиплярных секвенций (т. е. таких, которые могут содержать более одной формулы в сукцеденте). Для исчисления $\mathbf{IM}^{\xi}_{\mathrm{inv}}$ и исчисления $\mathbf{I}^{\xi}_{\mathrm{inv}}$ из [15] адаптированы существующие и разработаны новые стратегии поиска вывода.

На основе разработанного алгоритма поиска вывода, позволяющего комбинировать эти стратегии, удалось создать программу АДТ, решающую новые задачи из библиотеки ILTP по сравнению с существующими аналогами. Выполнено сравнение эффективности исчислений и стратегий обратного метода. В частности, эксперименты показали, что использование мультиплярных секвенций не приводит к сильному снижению эффективности программы: в исчислении $\mathbf{I}^{\xi}_{\mathrm{inv}}$, по сравнению с исчислением $\mathbf{IM}^{\xi}_{\mathrm{inv}}$, время решения тестовых задач в среднем меньше лишь на 15–20 %.

В дальнейшем программу можно использовать для апробации стратегий поиска вывода, а также в качестве наглядного пособия в учебном курсе «Математическая логика». Ещё одна перспективная задача – интеграция программы в существующие интерактивные программы АДТ (Coq, Nuprl).

СПИСОК ЛИТЕРАТУРЫ

1. **Драгалин А.Г.** Математический интуиционизм. Введение в теорию доказательств. М.: Наука, 1979.

2. **Клини С.К.** Перестановочность применений правил в генценовских исчислениях LK и LJ // Математическая теория логического вывода. М.: Наука, 1967. С. 208–236.

3. **Клини С.** Математическая логика. М.: Мир, 1973.

4. **Маслов С.Ю.** Обратный метод установления выводимости для логических исчислений // Труды МИАН СССР. 1968. Т. 98. С. 26–87.

5. **Маслов С.Ю.** Тактики поиска вывода, основанные на унификации порядка членов в благоприятном наборе // Зап. науч. сем. ЛОМИ. 1969. Т. 16. С. 126–136.

6. **Маслов С.Ю.** Связь между тактиками обратного метода и метода резолюций // Зап. науч. сем. ЛОМИ. 1969. Т. 16. С. 137–146.

7. **Маслов С.Ю.** Обратный метод и тактики установления выводимости для исчисления с функциональными знаками // Тр. МИАН СССР. 1972. Т. 121. С. 14–56.

8. **Маслов С.Ю**. О поиске вывода в исчислениях общего типа // Зап. науч. сем. ЛОМИ. 1972. Т. 32. С. 59–65.

9. **Мендельсон Э.** Введение в математическую логику. М.: Наука, 1971. 320 с.

10. **Оревков В.П.** О гливенковских классах секвенций // Логические и логико-математические исчисления. I, Тр. МИАН СССР. 1968. Т. 98. С. 131–154.

11. **Оревков В.П.** Обратный метод поиска вывода для скулемовских предваренных формул исчисления предикатов // Логическое программирование и Visual Prolog. СПб.: БХВ-Петербург, 2003. Приложение 4. С. 952–965.

12. **Павлов В., Пак В.** Экспериментальная

программа для доказательства теорем интуиционистской логики обратным методом Маслова // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2015. № 6 (234). С. 70–80. DOI: 10.5862/JCSTCS.234.7.

13. **Чень Ч., Ли Р.** Математическая логика и автоматическое доказательство теорем. Пер. с англ. М.: Наука, 1983. 360 с.

14. **Constable R.L.** On Building Constructive Formal Theories of Computation. Noting the Roles of Turing, Church, and Brouwer // Proc. of the 27th Annual IEEE/ACM Symp. on Logic in Computer Science. 2012. Pp. 2–8.

15. **Degtyarev A., Voronkov A.** The inverse method // Handbook of Automated Reasoning. Amsterdam: Elsevier, 2001. Vol. 1. Pp. 179–272.

16. Imogen GitHub page. URL: https://github.com/seanmcl/imogen. (Дата обращения: 08.02.2017).

17. **Kunze F.** Towards the Integration of an Intuitionistic First-Order Prover into Coq // Proc. of the 1st International Workshop Hammers for Type Theories. 2016. arXiv:1606.05948.

18. **Lifschitz V.** What is the inverse method? // Journal of Automated Reasoning. 1989. No. 5(1). Pp. 1–23.

19. **McCune W.** Prover9 and Mace4. URL: https://www.cs.unm.edu/~mccune/mace4/. 2005–2010. (Дата обращения: 12.09.2016).

20. **McLaughlin S., Pfenning F.** Efficient Intuitionistic Theorem Proving with the Polarized Inverse Method // CADE-22. LNCS. Springer, Heidelberg, 2009. Vol. 5663. Pp. 230–244.

21. **Mints G.** Resolution strategies for the Intuitionistic Logic // Constraint Programming. NATO ASI Series. Springer, Heidelberg, 1994. Vol. 131. Pp. 289–311.

22. **Mints G.** Decidability of the Class E by Maslov's Inverse Method // Essays Dedicated to Yuri Gurevich on the Occasion of His 70th Birthday. LNCS. Springer, Heidelberg, 2010. Vol. 6300. Pp. 529–537.

23. **Otten J.** Non-clausal Connection-based Theorem Proving in Intuitionistic First-Order Logic // Proc. of the 2nd International Workshop on Automated Reasoning in Quantified Non-Classical Logics. 2016. CEUR Workshop Proceedings 1770. Pp. 9–20.

24. **Pavlov V., Pak V.** The Inverse Method and First-Order Logic Theorem Proving // Nonlinear Dynamics and Applications. 2014. Vol. 20. Pp. 127–135.

25. **Raths T., Otten J., Kreitz C.** The ILTP Library: Benchmarking Theorem Provers for Intuitionistic Logic // Automated Reasoning with Analytic Tableaux and Related Methods, TABLEAUX 2005, LNAI. Springer Verlag, 2005. Vol. 3702. Pp. 333–337.

26. **Schmitt S. Lorigo L., Kreitz C., Nogin A.** JProver: Integrating Connection-Based Theorem Proving into Interactive Proof Assistants // 1st International Joint Conf., IJCAR 2001. LNCS. Springer Verlag, 2001. Vol. 2083. Pp. 421–426.

27. **Sutcliffe G., Suttner C.** Evaluating general purpose automated theorem proving systems // Artificial Intelligence. 2001. Vol. 131. Iss. 1–2. Pp. 39–54

28. **Tammet T.** A resolution theorem prover for intuitionistic logic // CADE-13. LNCS. Springer, Heidelberg, 1996. Vol. 1104. Pp. 2–16.

29. The ILTP Library. Provers and Results. URL: http://www.cs.uni-potsdam.de/ti/iltp/results.html (Дата обращения: 12.09.2016).

30. **Voronkov A.** Theorem Proving in Non-standard Logics Based on the Inverse Method // CADE-11. LNCS. Springer, Heidelberg, 1992. Vol. 607. Pp. 648–662.

REFERENCES

1. **Dragalin A.G.** *Matematicheskiy intuitsionizm. Vvedeniye v teoriyu dokazatelstv* [*Mathematical intuitionism. Introduction to the theory of evidence*]. Moscow: Nauka Publ., 1979. (rus)

2. **Kleene S.C.** Permutability of Inferences in Gentzen's Calculi LK and LJ. Mem. *Amer. Math. Soc*, 1952, No. 10, Pp. 1–26 [Russian transl. in *Matematicheskaya teoriya logicheskogo vyvoda*, Moscow, Nauka Publ., 1967, Pp. 208–236].

3. **Kleene S.** *Matematicheskaya logika* [*Mathematical logic*]. Moscow: Mir Publ., 1973. (rus)

4. **Maslov S.Yu.** Obratnyy metod ustanovleniya vyvodimosti dlya logicheskikh ischisleniy [The inverse method for establishing deducibility for logical calculus]. *Tr. MIAN SSSR*, 1968, Vol. 98, Pp. 26–87. (rus)

5. **Maslov S.Yu.** Taktiki poiska vyvoda, osnovannyye na unifikatsii poryadka chlenov v blagopriyatnom nabore [Deduction search tactics based on the unification of the order of members in favorable sets]. *Zap. nauchn. sem. LOMI*, 1969, Vol. 16, Pp. 126–136. (rus)

6. **Maslov S.Yu.** Svyaz mezhdu taktikami obratnogo metoda i metoda rezolyutsiy [A connection between tactics of the inverse method and the resolution method]. *Zap. nauchn. sem. LOMI*, 1969, Vol. 16, Pp. 137–146. (rus)

7. **Maslov S.Yu.** Obratnyy metod i taktiki ustanovleniya vyvodimosti dlya ischisleniya s funktsionalnymi znakami [The inverse method, and tactics for establishing deducibility for a calculus with functional signs]. *Tr. MIAN SSSR*, 1972, Vol. 121, Pp. 14–56. (rus)

8. **Maslov S.Yu.** O poiske vyvoda v ischisleniyakh obshchego tipa [The search for a deduction in the general type]. *Zap. nauchn. sem. LOMI*, 1972, Vol. 32, Pp. 59–65. (rus)

9. **Mendelson E.** *Vvedeniye v matematicheskuyu*

logiku [*Introduction to Mathematical Logic*]. Moscow: Nauka Publ., 1971, 320 p. (rus)

10. **Orevkov V.P.** On Glivenko sequent classes (Russian). *Trudy Matem. Inst. AN SSSR,* 1968, Vol. 98, Pp. 131—154 (rus) [English transl. in Proc. Steklov Inst. of Mathematics 98, Pp. 147—173, AMS, Providence,1971].

11. **Orevkov V.P.** Obratnyy metod poiska vyvoda dlya skulemovskikh predvarennykh formul ischisleniya predikatov [Reverse output method of searching for skulemovskih prenex predicate calculus formulas]. *Logicheskoye programmirovaniye i Visual Prolog* [*Logic Programming and Visual Prolog*]. St. Petersburg: BKhV-Petersburg Publ., 2003, Pp. 952—965. (rus)

12. **Pavlov V., Pak V.** Eksperimentalnaya programma dlya dokazatelstva teorem intuitsionistskoy logiki obratnym metodom Maslova. [An Experimental Computer Program for Automated Reasoning in Intuitionistic Logic Using the Inverse Method]. *Nauchno-tekhnicheskiye vedomosti SPbGPU. Informatika. Telekommunikatsii. Upravleniye*, [*St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunication and Control Systems*], 2015, Vol. 6 (234), Pp. 70—80. DOI: 10.5862/JCSTCS.234.7 (rus)

13. **Chang Ch., Lee R.** *Matematicheskaya logika i avtomaticheskoye dokazatelstvo teorem* [*Mathematical logic and automated theorem proving*]. Moscow: Nauka Publ., 1983, 360 p. (rus)

14. **Constable R.L.** On Building Constructive Formal Theories of Computation. Noting the Roles of Turing, Church, and Brouwer. *Proceedings of the 27th Annual IEEE/ACM Symposium on Logic in Computer Science*, 2012, Pp. 2—8.

15. **Degtyarev A., Voronkov A.** The inverse method. *Handbook of Automated Reasoning.* Amsterdam: Elsevier, 2001, Vol. 1, Pp. 179—272.

16. *Imogen GitHub page.* Available: https://github.com/seanmcl/imogen (Accessed: 08.02.2017).

17. **Kunze F.** Towards the Integration of an Intuitionistic First-Order Prover into Coq. *Proceedings of the 1st International Workshop Hammers for Type Theories*, 2016, arXiv:1606.05948.

18. **Lifschitz V.** What is the inverse method? *Journal of Automated Reasoning*, 1989, No. 5(1), Pp. 1—23.

19. **McCune W.** *Prover9 and Mace4.* Available: https://www.cs.unm.edu/~mccune/mace4/. 2005—2010 (Accessed: 12.09.2016).

20. **McLaughlin S., Pfenning F.** Efficient Intuitionistic Theorem Proving with the Polarized Inverse Method. *CADE-22. LNCS.* Springer, Heidelberg, 2009, Vol. 5663, Pp. 230—244.

21. **Mints G.** Resolution strategies for the Intuitionistic Logic. *Constraint Programming. NATO ASI Series.* Springer, Heidelberg, 1994, Vol. 131, Pp. 289—311.

22. **Mints G.** Decidability of the Class E by Maslov's Inverse Method. *Essays Dedicated to Yuri Gurevich on the Occasion of His 70th Birthday. LNCS.* Springer, Heidelberg, 2010, Vol. 6300, Pp. 529—537.

23. **Otten J.** Non-clausal Connection-based Theorem Proving in Intuitionistic First-Order Logic. *Proceedings of the 2nd International Workshop on Automated Reasoning in Quantified Non-Classical Logics*, 2016. CEUR Workshop Proceedings 1770, Pp. 9—20.

24. **Pavlov V., Pak V.** The Inverse Method and First-Order Logic Theorem Proving. *Nonlinear Dynamics and Applications*, 2014, Vol. 20, Pp. 127—135.

25. **Raths T., Otten J., Kreitz C.** The ILTP Library: Benchmarking Theorem Provers for Intuitionistic Logic. *Automated Reasoning with Analytic Tableaux and Related Methods, TABLEAUX 2005, LNAI.* Springer Verlag, 2005, Vol. 3702, Pp. 333—337.

26. **Schmitt S. Lorigo L., Kreitz C., Nogin A.** JProver: Integrating Connection-Based Theorem Proving into Interactive Proof Assistants. *1st International Joint Conference, IJCAR 2001, LNCS*, Springer Verlag, 2001, Vol. 2083, Pp. 421—426.

27. **Sutcliffe G., Suttner C.** Evaluating general purpose automated theorem proving systems. *Artificial Intelligence*, 2001, Vol. 131, Issues 1—2, Pp. 39—54.

28. **Tammet T.** A resolution theorem prover for intuitionistic logic. *CADE-13. LNCS.* Springer, Heidelberg, 1996, Vol. 1104, Pp. 2—16.

29. *The ILTP Library. Provers and Results.* Available: http://www.cs.uni-potsdam.de/ti/iltp/results.html (Accessed: 12.09.2016).

30. **Voronkov A.** Theorem proving in non-standard logics based on the inverse method. *CADE-11. LNCS.* Springer, Heidelberg, 1992, Vol. 607, Pp. 648—662.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ПАВЛОВ Владимир Александрович
PAVLOV Vladimir A.
E-mail: vlapav239@gmail.com

Статья поступила в редакцию 17.09.2016