DOI: 10.15514/ISPRAS-2020-32(2)-9

Верифицированная тактика Isabelle/HOL для теории ограниченных целых на основе инстанцирования и SMT

[1] Р.Ф. Садыков, ORCID: 0000-0002-2792-2465 <sadykov@ispras.ru>
[2] М.У. Мандрыкин, ORCID: 0000-0002-9306-7719 <mandrykin@ispras.ru>
[1] Московский государственный университет имени М.В. Ломоносова, 119991, Россия, Москва, Ленинские горы, д. 1
[2] Институт системного программирования им. В.П. Иванникова РАН, 109004, Россия, г. Москва, ул. А. Солженицына, д. 25

Аннотация. При дедуктивной верификации Си-программ как с помощью различных платформ верификации (Why3, Frama-C/WP, F*), так и с помощью систем интерактивного доказательства теорем (Isabelle, HOL4, Coq) достаточно широко используются SMT-решатели. Их разрешающие алгоритмы полны для некоторых комбинаций логических теорий (логик), в частности для логики QF_UFLIA. В то же время при верификации Си-программ часто необходимо оперировать формулами в других разрешимых логиках, поддерживаемых не всеми SMT-решателями. Типичными примерами таких логик могут служить комбинации QF_UFLIA с теориями ограниченных целых как с переполнением (для беззнаковых целых в Си), так и без переполнения (для знаковых целых), а также теорией конечных интерпретируемых множеств (для поддержки рамочных условий) и др. Одним из возможных способов поддержки таких логик является их непосредственная реализация в SMT-решателях, однако этот способ часто является трудоемким, а также недостаточно гибким и универсальным. Другим способом является реализация пользовательских стратегий инстанцирования кванторов для сведения формул в неподдерживаемых логиках к формулам в широко распространенных разрешимых логиках, таких как QF_UFLIA. В данной статье представлена процедура инстанцирования лемм для трансляции формул в теории ограниченных целых без переполнения в логику QF_UFLIA. Для процедуры трансляции даны доказательства корректности и полноты, а также описана формализация этих доказательств в системе Isabelle/HOL. Аналогичный подход можно использовать для формулирования и доказательства полноты процедур трансляции формул в других теориях, таких как теория ограниченных целых с переполнением и теория ограниченной адресной арифметики.

Ключевые слова: статическая верификация; задача выполнимости формул в теориях; автоматизированные процедуры принятия решений

Для цитирования: Садыков Р.Ф., Мандрыкин М.У. Верифицированная тактика Isabelle/HOL для теории ограниченных целых на основе инстанцирования и SMT. Труды ИСП РАН, том 32, вып. 2, 2020 г., стр. 107-124. DOI: 10.15514/ISPRAS–2020–32(2)–9

Благодарности. Исследование поддержано Министерством образования и науки РФ (проект №RFMEFI60719X0295).

Verified Isabelle/HOL tactic for the theory of bounded integers based on quantifier instantiation and SMT

[1] R. Sadykov ORCID: 0000-0002-2792-2465 <sadykov@ispras.ru>
[2] M. Mandrykin ORCID: 0000-0002-9306-7719 <mandrykin@ispras.ru>
[1] Lomonosov Moscow State University, GSP-1, Leninskie Gory, Moscow, 119991, Russia
[2] Ivannikov Institute for System Programming of the Russian Academy of Sciences, 25, Alexander Solzhenitsyn st., Moscow, 109004, Russia

Abstract. SMT solvers are widely applied for deductive verification of C programs using various verification platforms (Why3, Frama-C/WP, F*) and interactive theorem proving systems (Isabelle, HOL4, Coq) as the decision procedures implemented in SMT solvers are complete for some combinations of logical theories (logics), in particular for the QF_UFLIA logic. At the same time, when verifying C programs, it is often necessary to discharge formulas in other logical theories and their combinations, that are also decidable but not supported by all SMT solvers. Theories of bounded integers both with overflow (for unsigned integers in C) and without overflow (for signed integers), and also theory of finite interpreted sets (needed to support frame conditions) are good examples of such theories. One of the possible ways to support such theories is to directly implement them in SMT-solvers, however, this method is often time-consuming, as well as insufficiently flexible and universal. Another way is to implement custom quantifier instantiation strategies to reduce formulas in unsupported theories to formulas in widespread decidable logics such as QF_UFLIA. In this paper, we present an instantiation procedure for translating formulas in the theory of bounded integers without overflow into the QF_UFLIA logic. We formally proved soundness and completeness of our instantiation procedure in Isabelle. The paper presents an informal description of this proof as well as some considerations on the efficiency of the proposed procedure. Our approach is sufficient to obtain efficient decision procedures implemented as Isabelle/HOL proof methods for several decidable logical theories used in C program verification by relying on the existing capabilities of well-known SMT solvers, such as Z3 and proof reconstruction capabilities of the Isabelle/HOL proof assistant.

Keywords: static verification; quantifier instantiation; SMT formulas; SMT solvers; automated decision procedures; software verification

For citation: Sadykov R., Mandrykin M. Verified Isabelle/HOL tactic for the theory of bounded integers based on quantifier instantiation and SMT. Trudy ISP RAN/Proc. ISP RAS, vol. 32, issue 2, 2020. pp. 107-124 (in Russian). DOI: 10.15514/ISPRAS–2020–32(2)–9

Acknowledgement. The research was carried out with funding from the Ministry of Science and Higher Education of the Russian Federation (the project unique identifier is RFMEFI60719X0295).

1. Введение

В процессе разработки программного обеспечения на языке Си часто возникают ошибки, связанные с неправильным использованием арифметических операций, которые могут привести к переполнению, и, как следствие, непредвиденному поведению программ, несмотря на долгое и тщательное тестирование. Но благодаря формальным методам верификации программного кода возможно найти неочевидные, но в то же время многочисленные ошибки, связанные с операциями над целочисленными типами и их значениями. Многие методы и инструменты дедуктивной верификации используют решатели задачи выполнимости формул в теориях, называемые также SMT-решателями. Алгоритмы таких решателей полны для некоторых комбинаций логических теорий, в частности, для логики QF_UFLIA. Несмотря на применимость SMT-решателей к широкому классу задач, часто необходимо применять решающие процедуры к формулам в логических теориях, которые являются разрешимыми, но не всегда полностью поддерживаются SMT-решателями. К таким логическим теориям можно отнести ограниченные целые как с переполнением (для беззнаковых целых в Си), так и без переполнения (для знаковых целых), теории конечных интерпретируемых множеств (для поддержки рамочных условий) и др. В

данной работе с целью автоматизированного получения доказательств в системе интерактивного доказательства теорем Isabelle/HOL [1] мы сформулировали метод проверки выполнимости формул с ограниченными целочисленными значениями переменных, который основан на применении SMT-решателя в логике QF_UFLIA. Наш метод является решающей процедурой над формулами в теории ограниченных целых (BLIA), которая реализована в виде метода Isabelle/HOL и состоит из инстанцирования исходной формулы аксиомами теории ограниченных целых (являющихся леммами теории HOL-Word[2]), интерпретации полученной формулы в теории QF_UFLIA, применения решающей процедуры SMT-решателя и последующего восстановления доказательства средствами Isabelle. В свою очередь, доказательство полноты предложенного метода основывается на преобразовании полученной от SMT-решателя модели в логике QF_UFLIA в модель в теории ограниченных целых.

Несмотря на инстанцирование аксиом исходной теории, которое увеличивает длину анализируемой формулы, нам не требуется вносить существенные изменения в инструменты воспроизведения доказательств системы Isabelle и SMT-решатель, а благодаря оптимизации аксиом теории ограниченных целых мы получаем линейное по количеству вхождений функциональных символов увеличение длины исходной формулы, поэтому сложность метода не существенно увеличивается по сравнению с использованием решающих процедур, реализованных непосредственно внутри SMT-решателей. Основной целью нашей работы являлось доказательство полноты и корректности предложенной процедуры преобразования формул из теории ограниченных целых в логику QF_UFLIA.

Также мы рассмотрели некоторые статьи о модульной арифметике [3], [4], [5], где описаны ошибки с переполнениями переменных типа int и верификация программ, использующих ограниченные целые. В работе [3] сформулированы методы преобразования для формул в нелинейной модульной арифметике с ограниченными целыми, которые предполагают изменение внутренних алгоритмов SMT-решателя. Потенциально получаемые при этом доказательства (соответствующая дедуктивная система подробно не описана в статье), иногда сложно воспроизвести с использованием существующих возможностей систем автоматизированных доказательств. В статье [4] получена оценка сложности решающей процедуры для IDL. IDL – это логика разности над целыми числами. По определению, это булевы комбинации неравенств вида $x - y < b$, где x и y – целочисленные переменные, а b – целочисленная константа. Решающая процедура данной логики является NP-трудной; аналогичная логика QF_UFLIA, которая является расширением логики IDL операцией умножения на константу, – тоже NP-полная задача.

2. Предварительная информация

В стандарте SMT-LIB [8] QF_UFLIA – это логика бесквантворных формул с неинтерпретируемыми символами и равенством в комбинации теорий линейной целочисленной арифметики и неинтерпретируемых функций. Логику QF_UFLIA можно рассматривать как комбинацию логик QF_LIA и QF_UF. QF_LIA обозначают замкнутые бесквантворные формулы с равенством в теории линейной целочисленной арифметики (LIA). В сигнатуру этой теории входят следующие функциональные символы: $\{+, c \times \cdot, \leq\}$, где c – целочисленная константа. QF_UF обозначает замкнутые бесквантворные формулы с равенством в теории неинтерпретируемых функций.

Общеизвестным методом решения задачи доказательства условий корректности с ограниченными целыми числами является представление целых чисел в виде битовых векторов. Но как написано в статье [11], логика QF_BV очень сложная в общем случае, на практике различие во времени разрешения формул в QF_BV и QF_LIA часто выглядит как экспоненциальное. Кроме того, сложность решения в QF_BV зависит от размера ограниченного целого, поэтому для больших ограниченных целых, например, 256 бит,

различие эффективности между QF_BV и QF_LIA может быть очень большим. Для QF_LIA размер никак не влияет на время решения. Для QF_BV нет поддержки воспроизведения доказательств в Isabelle. И доказательства обычно больше по сравнению с LIA, так как основаны на переборе большого числа вариантов и не переиспользуют встроенную процедуру для линейной арифметики в Isabelle. Но есть фрагменты логики QF_BV, которые можно решать существенно более эффективно, но в решателях реализованы не эти алгоритмы, а общий случай. Эффективную решающую процедуру для линейного фрагмента со сдвигами и побитовыми операциями можно тоже сделать на основе процедуры конечного инстанцирования, аналогично как в этой работе.

Теорию линейной арифметики с ограниченными целыми мы будем задавать аксиоматически как расширение теории линейной целочисленной арифметики. В ее сигнатуру помимо символов теории LIA включим следующие функциональные символы: $\{\cdot +_b \cdot, c \times_b \cdot, v(\cdot), (\cdot)_b, \leq_b \cdot\}$, где c – неограниченная целочисленная константа. Термы в этой теории могут быть двух различных типов – ограниченные и неограниченные целые. Ограниченные целые принимают целочисленные значения из некоторого диапазона $[L..U]$. Эти значения являются неограниченными целыми и могут быть получены с помощью применения функции $v(\cdot)$. Будем использовать a, b и d в качестве переменных, относящихся к сорту ограниченных целых. Для сорта неограниченных целых будем использовать переменные x, y и c, при этом c всегда будет обозначать интерпретируемую целочисленную константу. Таким образом, для любого ограниченного целого a будет выполнено соотношение $L \leq v(a) \leq U$. Семантика операций $\cdot +_b \cdot$ и $\cdot \times_b \cdot$ такова, что их результаты совпадают с результатами соответствующих целочисленных операций, примененных к значениям ограниченных аргументов, если эти результаты могут быть представлены в виде ограниченных целых. В противном случае эти результаты не определены, то есть их модель не фиксирована и может быть выбрана произвольно для каждой конкретной формулы. Операция $\cdot_b$ возвращает ограниченное целое с заданным значением, если это значение лежит в диапазоне $[L..U]$ и имеет неопределенный результат в противном случае. Операция $\cdot \leq_b \cdot$ сравнивает значения ограниченных целых и по сути является сокращенным обозначением для выражения вида $v(\cdot) \leq v(\cdot)$. В качестве примеров формул в теории ограниченных целых рассмотрим следующие:

$$v(a) \leq 5 \wedge 5 \leq v(a) \wedge (5)_b \times_b a +_b (1)_b \neq (26)_b \text{ и} \quad (1)$$

$$a \leq_b (5)_b \wedge (5)_b \leq_b a \wedge (5 \times v(a) + 1 - 26)_b \neq (0)_b. \quad (2)$$

Будем считать $L = 0$ и $U = 25$. В этих предположениях формула (1) будет являться выполнимой, например, если взять в качестве значения $v(a)$ ограниченной переменной a число 5 ($v(a) = 5$, $(5)_b \times_b a = (25)_b$, но $(5)_b \times_b a +_b (1)_b$ и $(26)_b$ не определено, так как $25 + 1 = 26 > 25$ и может быть выбрано равным, например, $(0)_b$). Формула (2) же является не выполнимой, так как из $a \leq_b (5)_b \wedge (5)_b \leq_b a$ следует, что $v(a) = 5$, а значит $5 \times v(a) + 1 - 26 = 0$ и $(0)_b = (0)_b$.

Для решения задачи о выполнимости формул в теории ограниченных целых будем использовать процедуру трансляции в логику QF_UFLIA. Процедура преобразования формулы F состоит из последовательного инстанцирования аксиом теории ограниченных целых. Инстанцированием аксиомы для некоторой формулы F будем называть взятие конъюнкции от аксиомы с некоторой подстановкой и формулы F. Операция инстанцирования аксиомы довольна проста, мы применяем инстанцирование, когда можем найти подходящий терм в формуле F согласно правилам процедуры преобразования. Такое преобразование создает некоторую формулу F^*, которую мы интерпретируем в теории QF_UFLIA, и запускаем решающую процедуру SMT-решателя, получая доказательство, которое затем воспроизводится (сертифицируется) средствами Isabelle [6], [7]. Для доказательства полноты процедуры преобразования рассматриваем случай, когда мы получили некоторую модель R преобразованной формулы F^* в логике QF_UFLIA. Далее будем называть модель R реализацией формулы F. В данной статье мы показываем, как из

реализации R может быть восстановлена модель M исходной формулы F в теории ограниченных целых. В следующих разделах рассмотрены основные обозначения и определения, описывающие данную процедуру, пример работы алгоритма, формулировка и доказательство соответствующей теоремы о полноте.

3. Постановка задачи

Пусть $\mathbb{Z}$ – множество целых чисел, $\mathbb{Z}_b$–множество ограниченных целых и $\sum = \{+_b, \times_b, \leq_b\}$ – сигнатура теории T линейной арифметики с ограниченными целыми (BLIA). F – формула в BLIA, в которой переменные принимают значения из множества $\mathbb{Z}_b$.

Множество аксиом теории ограниченных целых состоит из аксиом линейной целочисленной арифметики и 5 формул (см. рис. 1). Обозначим это множество за T_0, где L и U являются ограничениями на целые, символ $+_b$ обозначает операцию суммирования в $\mathbb{Z}_b$, $\times_b$ – умножение в $\mathbb{Z}_b$, функция v отображает из множества $\mathbb{Z}_b$ в $\mathbb{Z}$. В практической реализации нашего инструмента в Isabelle/HOL мы рассматриваем эти 5 формул как леммы теории «HOL-Word» [2]. Так как аксиома $A6$ в теории T_0 зависит от двух переменных a и b, мы вводим расширение теории T_0 – теорию T_1, чтобы избежать квадратичного увеличения размера формулы с инстанцированными аксиомами. Поэтому наш алгоритм реализует процедуру инстанцирования аксиомами из теории T_1.

Теория T_1 отличается от T_0 аксиомой $A6$ (см. аксиому $A6'$ на рис. 1).

Покажем эквивалентность теорий T_0 и T_1.

Теорема 1. *Каждая формула F без свободных переменных в сигнатуре* $\sum$ *выполнима в теории* T_0 *тогда и только тогда, когда она выполнима в теории* T_1.

Доказательство. Покажем, что для любой формулы F, имеющей модель M в теории T_0, существует та же модель M в теории T_1 и наоборот.

Пусть формула F имеет модель M в теории T_0. Тогда все аксиомы, кроме ($A6'$), содержатся в M. Возьмем произвольное ограниченное целое a. Поскольку ($A4$) сохраняется, мы имеем

$$L \leq v^M(a) \leq U.$$

Обозначим $v^M(a)$ как c. Тогда $L \leq c = v^M(a) \leq U$, и поскольку ($A5$) сохраняется, имеем

$$v^M((c)_b^M) = c.$$

Получим, что $v^M((c)_b^M) = c = v^M(a)$ и по аксиоме ($A6$) имеем

$$(c)_b^M = a.$$

Теперь из $v^M(a) = c$ согласно равенству имеем

$$v^M((v^M(a))_b^M) = v^M((c)_b^M),$$

и согласно ($A6$),

$$(v^M(a))_b^M = (c)_b^M.$$

Следовательно, $(v^M(a))_b^M = (c)_b^M = a$ и получаем ($A6'$) для произвольного a.

Теперь предположим, что F имеет модель M в T_1. Тогда все аксиомы, кроме ($A6$) содержатся в M. Следовательно, для любых a и b таких, что $v^M(a) = v^M(b)$, по аксиоме ($A6'$) имеем

$$(v^M(a))_b^M = a \text{ и } (v^M(b))_b^M = b.$$

Из $v^M(a) = v^M(b)$ получим $(v^M(a))_b^M = (v^M(b))_b^M$ согласно приведенному выше равенству. Таким образом, и ($A6$) сохраняется для любых a и b. □

4. Процедура инстанцирования

Зададим процедуру трансляции формулы F в теории BLIA в эквивалентную формулу F^* в теории QF_UFLIA для того, чтобы проверить выполнимость формулы F.

Эта процедура использует аксиомы теории T_1 для составления правил инстанцирования по формуле F следующим образом:

- ($A1$) инстанцирована с a и b для любого терма $a+_b b$ в формуле F.
- ($A2$) инстанцирована с c и a для любого терма $c \times_b a$ в формуле F.
- ($A3$) инстанцирована с a и b для любого терма $a \leq_b b$ в формуле F.
- ($A4$) инстанцирована с a для любого терма типа $\mathbb{Z}_b$ в формуле F.
- ($A5$) инстанцирована с c для любого терма вида $(c)_b$ в формуле F.
- ($A6'$) инстанцирована с a для любого терма типа $\mathbb{Z}_b$ в формуле F.

Обозначим за F^* формулу F после выполнения алгоритма инстанцирования аксиом. Обозначим за F^+ множество формул, полученных инстанцированием аксиомы ($A1$) для любого терма $a+_b b$ в формуле F.

Аналогично определим множества $F^\times$ для ($A2$), $F^\leq$ для ($A3$), $F^\in$ для ($A4$), F^c для ($A5$), F^m для ($A6'$).

После всех трансляций, символы умножения и суммирования воспринимаются как неинтерпретируемые символы. И символом $\wedge\wedge$ обозначим операцию взятия конъюнкции всех формул из множества.

Тогда полученная формула равна следующему

$$F^* = F \wedge (\wedge\wedge F^+) \wedge (\wedge\wedge F^\times) \wedge (\wedge\wedge F^\leq) \wedge (\wedge\wedge F^\in) \wedge (\wedge\wedge F^\mathrm{c}) \wedge (\wedge\wedge F^\mathrm{m}).$$

Формула F^* решается в теории QF_UFLIA с помощью SMT-решателя. В этом подходе мы пользуемся тем, что нам не требуется разрабатывать инструмент для решения нашей задачи, вместо этого мы можем без изменения SMT-решателей, которые предназначены для широкого класса задач, произвести проверку выражений и осуществить решающую процедуру, описывающую некоторые операции и типы языка программирования Си.

$$\sum = \{+_b, \times_b, \leq_b\}, a, b \in \mathbb{Z}_b, v(a) \in \mathbb{Z}, \mathrm{c} \in \mathbb{Z}, \mathrm{c} - \text{константа}$$
$$\forall a, b \in \mathbb{Z}_b.\ L \leq v(a) + v(b) \leq U \Longrightarrow v(a+_b b) = v(a) + v(b), \quad (A1)$$
$$\forall a \in \mathbb{Z}_b.(c \times_b a) = c \times v(a), \quad (A2)$$
$$\forall a, b \in \mathbb{Z}_b\ a \leq_b b \Longrightarrow v(a) \leq v(b), \quad (A3)$$
$$\forall a \in \mathbb{Z}_b.\ L \leq v(a) \leq U, \quad (A4)$$
$$\forall c \in \mathbb{Z}\ .\ L \leq c \leq U \Longrightarrow v((c)_b) = c \quad (A5)$$
$$\forall a, b \in \mathbb{Z}_b.(a) + v(b) \Longrightarrow a = b \quad (A6)$$
$$\forall a \in \mathbb{Z}_b.\ ((v(a))_b = a \quad (A6')$$

Рис. 1. Аксиомы, определяющие теорию ограниченных целых
Fig. 1. Axioms defining the theory of bounded integes

2.1 Пример процедуры трансляции

Пример. Пусть

$$F = (25)_b \leq_b a \wedge -1 \times_b a +_b (25)_b \neq (0)_b.$$

И ограничения равны следующим значениям: $L = -25$, $U = 25$.

Шаги инстанцирования:

1.Используя аксиому ($A1$):

$$\underline{-1 \times_b a} +_b \underline{(25)_b} \in F \Rightarrow$$
$$-25 \leq v(-1 \times_b a) + v((25)_b) \leq 25 \Rightarrow$$

$v(-1 \times_b a +_b (25)_b) = v(-1 \times_b a) + v((25)_b)$,

2.Используя аксиому (A2):

$\underline{-1} \times_b \underline{a} \in F \Rightarrow$

$-25 \leq -1 \times v(a) \leq 25 \Rightarrow v(-1 \times_b a) = -1 \times v(a)$,

3. Используя аксиому (A3):

$\underline{(25)_b} \leq_b \underline{a} \in F \Rightarrow (25)_b \leq_b a \Leftrightarrow v((25)_b)) \leq v(a)$,

4. Используя аксиому (A4):

$a \in F \Rightarrow -25 \leq v(a) \leq 25$,

5. Используя аксиому (A5):

$\underline{(25)_b} \in F \Rightarrow -25 \leq 25 \leq 25 \Rightarrow v((25)_b) = 25$,

6. Используя аксиому (A6'):

$-1 \times_b a +_b (25)_b \in F \Rightarrow (v(-1 \times_b a +_b (25)_b))_b = -1 \times_b a +_b (25)_b$.

Тогда формула F^* равна $F \wedge (-25 \leq v(-1 \times_b a) + v((25)_b) \leq 25 \Rightarrow \cdots) \wedge \ldots$. Стоит заметить, что правила A4 и A6' выглядят одинаково, но в примере были применены для разных термов, данная запись была сделана для наглядности, в программной реализации в Isabelle/HOL такие правила инстанцировния применяются для одинаковых термов. Покажем, что формула F невыполнима в логике BLIA.

Утверждение 1. *Формула F невыполнима в логике BLIA.*

Доказательство. Воспользуемся полученными формулами после применения процедуры инстанцирования. Из инстанцирования (A5) получим $v((25)_b) = 25$. Тогда из инстанцирования аксиомы (A3) получим $25 = v((25)_b)) \leq v(a)$ и из инстанцирования аксиомы (A4) имеем $v(a) \leq 25$. Таким образом, $v(a) = 25$.

Теперь из инстанцирования (A2) получим $v(-1 \times_b a) = -1 \times v(a) = -25$, тогда из (A1) $v(-1 \times_b a +_b (25)_b) = v(-1 \times_b a) + v((25)_b) = -25 + 25 = 0$.

Из аксиомы (A6') получим $-1 \times_b a +_b (25)_b = (v(-1 \times_b a +_b (25)_b))_b = (0)_b$.

Отсюда получаем следующее противоречие со вторым конъюнктом формулы F $-1 \times_b a +_b (25)_b \neq (0)_b$

Следовательно, формула F невыполнима.□

Нам нужно доказать полноту и корректность процедуры инстанцирования для теории ограниченных целых.

Корректность очевидно следует из того, что если формула F выполнима в BLIA, то формула F^* также выполнима в QF_UFLIA, так как состоит из результата инстанцирования аксиом BLIA и формулы F.

3. Доказательство полноты

Здесь мы рассмотрим произвольную формулу без свободных переменных F в теории BLIA, её трансляцию F^*, полученную через процедуру инстанцирования аксиом, и модель R, полученную из трансляции F^* в QF_UFLIA, которую мы называем реализацией.

Термы и предикаты формул F и F^* будем обозначать как $v(u) + v(t)$, а их соответствующие выражения в реализации как $v^R(u^R) +_b^R v^R(t^R)$.

Свободные термы ограниченных целых будем обозначать буквами t и u, свободные термы целых чисел обозначим через k и n, а произвольные выбранные ограниченные целые как a и b, соответствующие неограниченные целые через x, y или c.

Доказательство осуществляется с помощью восстановления модели M исходной формулы F в теории BLIA из реализации R.

Для начала докажем несколько вспомогательных лемм.

Лемма 1. *Терм вида $v(t)$ принадлежит формуле F^* тогда и только тогда, когда ограниченный целый терм t принадлежит F.*

Доказательство. Пусть терм t содержится в F. Тогда согласно правилу инстанцирования аксиомы (A4) терм $v(t)$ содержится в F^*.

Теперь предположим, что $v(t)$ принадлежит множеству термов F^*. Покажем, что этот терм возникает только благодаря процедуре инстанцирования с термом t, который принадлежит множеству термов F.

Доказательство перечислением правил инстанцирования. Рассмотрим правило (A1). Оно содержит 3 случая применения функции v, обозначаемые $v(t +_b u)$, $v(t)$ и $v(u)$.

Аксиома (A1) применяется только тогда, когда термы $t +_b u$ содержатся F. Следовательно, термы t, u и $t +_b u$ уже содержатся в F.

Доказательство относительно других правил аналогично. Случай когда $v(t)$ принадлежит множеству термов F очевиден.□

Лемма 2. *Терм вида $(v(t))_b$ принадлежит формуле F^* тогда и только тогда, когда ограниченный целый терм t принадлежит F.*

Доказательство. Пусть терм t содержится в F. Тогда согласно правилу инстанцирования аксиомы (A6') терм $(v(t))_b$ содержится в F^*.

Теперь предположим, что $(v(t))_b$ принадлежит множеству термов F^*. Аксиома (A6') применяется только тогда, когда терм t содержится в F. Случай $(v(t))_b$ принадлежит множеству термов F очевиден.□

Лемма 3. *Терм вида $(n)_b$ принадлежит формуле F^* тогда и только тогда, когда он уже принадлежит F, либо n равен $v(t)$, где t принадлежит F.*

Доказательство. Пусть терм t содержится в F. Тогда согласно лемме 2 терм $(v(t))_b$ содержится в F^*. Таким образом, $(n)_b$ принадлежит формуле F*, где $n = v(t)$.

Теперь предположим, что терм $(n)_b$ содержится в F^* и не содержится в F. В таком случае нам нужно показать, что либо $(n)_b$ содержится в F, либо n вида $v(t)$, где t содержится в F. Докажем это перечислением правил инстанцирования. Только аксиомы (A5) и (A6') содержат символ $(\cdot)_b$. Согласно правилу (A5) $(n)_b$ содержится в F, в случае (A6') n имеет требуемый вид. □

Далее дадим несколько дополнительных обозначений, которые мы будем применять в доказательствах.

Определим образ подмножества X на множестве A с помощью следующей функции $f: A \to B$ как $f[X]$. Введем следующее обозначение:

$\{f(x) \mid P(x)\} \equiv f[\{x \mid P(x)\}]$, где $\{x \mid P(x)\}$ − множество термов x, удовлетворяющих предикату $P(x)$.

Далее определим следующие множества:

$B^R \equiv \{t^R \mid v(t) \in F^*\}$,

$C^R \equiv \{n^R \mid (n)_b \in F^*\}$, где, как мы определили в начале разд. 3, t^R и n^R обозначают означивание определенных термов t и n в реализации R. Реализация R содержит частичные модели функций $v(\cdot)$ и $(\cdot)_b$ на соответствующих множества, определяемых термами в F^*. Рассмотрим функции $v^R(\cdot)$ и $(\cdot)_b^R$ (которые будем считать реализациями соответствующих функций $v(\cdot)$ и $(\cdot)_b$), определенные на множествах B^R и C^R соответственно.

Далее докажем некоторые свойства функций $v(\cdot)$ и $(\cdot)_b$.

Лемма 4. $v^R[B^R] \subseteq C^R$.

Доказательство.

$$\begin{aligned}
v^R[B^R] &= \{v^R(a) \mid a \in B^R\} \\
&= \{v^R(a) \mid a \in \{t^R \mid v(t) \in F^*\}\} \\
&= \{v^R(t^R) \mid v(t) \in F^*\} \\
&= \{v^R(t^R) \mid t \in F\} \text{ (по лемме 1)} \\
&= \{v^R(t^R) \mid (v(t))_b \in F^*\} \text{ (по лемме 2)} \\
&= \{(v(t))^R \mid (v(t))_b \in F^*\} \\
&= \{n^R \mid (n)_b \in F^* \wedge \exists t.\, n = v(t)\} \\
&\subseteq \{n \mid (n)_b \in F^*\} \\
&= C^R.\square
\end{aligned}$$

Лемма 5. $v^R[B^R] \subseteq [L, U]$.

Доказательство. По определению B^R для любого целого с такого, что $c \in v^R[B^R]$ имеем $c = v^R(t^R)$, где $v(t) \in F^*$. Из леммы 1 следует $t \in F$. Это, в свою очередь, означает, что аксиома (A4) была инстанцирована с термом t, то есть $L \le v(t) \le U$ − подтерм без свободных переменных в F^*. Поскольку R является моделью F^*, $L \le v^R(t^R) \le U$ и, следовательно, $L \le c \le U$ для любого $c \in B^R$.□

Лемма 6. $v^R[B^R] \subseteq C^R \cap [L, U]$.

Доказательство. Следует напрямую из лемм 4 и 5. □

Лемма 7. *v^Rинъективна на множестве определения B^R.*

Доказательство. Возьмем произвольные значения ограниченно целых x и y из множества B^R. Тогда имеем $x = t^R$, $y = u^R$, $v(t) \in F^*$ и $v(u) \in F^*$. Тогда по лемме 1 $t \in F$ и $u \in F$ и, таким образом, аксиома (A6') была инстанцирована с этими термами, сохраняющие предикаты $(v(t))_b = t$ и $(v(u))_b = t$ являются подтермами без свободных переменных в F^*. Так как R модель F^*, мы имеем $(v^R(t^R))_b^R = t^R$и $(v^R(u^R))_b^R = u^R$ и поскольку R является моделью в теории QF_UFLIA, которая включает в себя сравнение неинтерпретированных функций (таких как $(\cdot)_b$) имеем $v^R(t^R) = v^R(u^R) \Rightarrow (v^R(t^R))_b^R = (v^R(u^R))_b^R$ или, что то же самое, что $v^R(t^R) = v^R(u^R) \Rightarrow t^R = u^R$, то есть $v^R(x) = v^R(y) \Rightarrow x = y$ для любых x и y из множества B^R.□

Лемма 8. *v^R сюръективна на множестве значений $C^R \cap [L, U]$.*

Доказательство.

$$\begin{aligned}
C^R &= \{n^R \mid (n)_b \in F^*\} \text{ (по определению } C^R) \\
&= \{n^R \mid (n)_b \in F\} \cup \{(v(t))^R \mid t \in F\} \text{ (по лемме 3)} \\
&= \{n^R \mid (n)_b \in F\} \cup \{(v(t))^R \mid v(t) \in F^*\} \text{ (по лемме 1)} \\
&= \{n^R \mid (n)_b \in F\} \cup \{v^R(t^R) \mid v(t) \in F^*\} \\
&= \{n^R \mid (n)_b \in F\} \cup v[\{t^R \mid v(t) \in F^*\}] \\
&= \{n^R \mid (n)_b \in F\} \cup v[B^R] \text{ (по определениюо} B^R).
\end{aligned}$$

Таким образом, чтобы доказать $C^R \cap [L, U] \subseteq v^R[B^R]$ мы должны показать, что $\{n^R \mid (n)_b \in F\} \cap [L, U] \subseteq v^R[B^R]$. Рассмотрим любой терм n такой, что $(n)_b \in F$.

Поскольку $(n)_b \in F$, (A5) была инстанцирована с n и, следовательно, $L \le n \le U \Rightarrow v((n)_b) = n$ является подтермом без свободных переменных в F^* и, R является моделью для F^*, $L \le n^R \le U \Rightarrow v^R((n^R)_b^R) = n^R$. Так что, если $n^R \in [L, U]$, то $n^R = v^R((n^R)_b^R) \in \{v^R((n^R)_b^R) \mid (n)_b \in F\}$. Следовательно,

$$\begin{aligned}
\{n^R \mid (n)_b \in F\} \cap [L, U] &\subseteq \{v^R((n^R)_b^R) \mid (n)_b \in F\} \\
&= \{v^R(((n)_b)^R) \mid (n)_b \in F\} \\
&= \{v^R(t^R) \mid t \in F \wedge \exists n.\, t = (n)_b\} \subseteq \{v^R(t^R) \mid t \in F\}
\end{aligned}$$

$$\begin{aligned}
&= \{v^R(t^R) \mid v(t) \in F^*\} \text{ (по лемме 1)} \\
&= v^R[\{t^R \mid v(t) \in F^*\}] \\
&= v^R[B^R].\square
\end{aligned}$$

Лемма 9. *v^R биективна на множествах B^Rи $C^R \cap [L, U]$ и функция $(\cdot)_b^R$ является обратной к ней.*

Доказательство. Утверждение о том, что v^R является биекцией между B^R и $C^R \cap [L, U]$ следует непосредственно из лемм 7 и 8. Таким образом, v^R имеет обратную биективную функцию. Эта обратная функция может быть однозначно охарактеризована уравнением $(v^R)^{-1}(v^R(a)) = a$ для любого $a \in B^R$. Но для каждого $a \in B^R$, значение a можно представить как t^R, где $v(t) \in F^*$. По лемме 1 терм $t \in F$ и, следовательно, аксиома (A6') была инстанцирована с термом t. Таким образом, $(v^R(t^R))_b^R = t^R$ и, следовательно, $(v^R(a))_b^R = a$ для любого $a \in B^R$. Таким образом, обратная биективная функция совпадает с функцией $(\cdot)_b^R$.□

Лемма 10. $(\cdot)_b^R[C^R] \subseteq B^R$**.**

Доказательство.

$$\begin{aligned}
(\cdot)_b^R[C^R] &= (\cdot)_b^R[\{n^R \mid (n)_b \in F^*\}] \text{ (по определению } C^R) \\
&= (\cdot)_b^R[\{n^R \mid (n)_b \in F\} \cup \{(v(t))^R \mid t \in F\}] \text{ (по лемме 3)} \\
&= (\cdot)_b^R[\{n^R \mid (n)_b \in F\}] \cup (\cdot)_b^R[\{(v(t))^R \mid t \in F\}] \text{ (по определению)} \\
&= \{(n^R)_b^R \mid (n)_b \in F\} \cup \{((v(t))^R)_b^R \mid t \in F\} \\
&= \{((n)_b)^R \mid (n)_b \in F\} \cup \{(v^R(t^R))_b^R \mid t \in F\}.
\end{aligned}$$

$$\begin{aligned}
\{((n)_b)^R \mid (n)_b \in F\} &= \{t^R \mid t \in F \wedge \exists n.\, t = (n)_b\} \\
&\subseteq \{t^R \mid t \in F\} \\
&= \{t^R \mid v(t) \in F^*\} \text{ (по лемме 1)} \\
&= B^R \text{ (по определению } B^R).
\end{aligned}$$

Для каждого $t \in F$ (A6') был инстанцирован с t, поэтому $(v(t))_b = t$ является подтермом без свободных переменных в формуле F^*, и поскольку R является моделью F^*, $(v^R(t^R))_b^R = t^R$. Таким образом, по определению соответствующего множества,

$$\begin{aligned}
\{(v^R(t^R))_b^R \mid t \in F\} &= \{t^R \mid t \in F\} \\
&= \{t^R \mid v(t) \in F^*\} = B^R \text{ (по лемме 1 и определению } B^R).
\end{aligned}$$

Используя приведенное выше представление $(\cdot)_b^R[C^R]$ мы наконец имеем требуемое выражение $(\cdot)_b^R[C^R] \subseteq B^R$.□

Получим ситуацию, изображенную на рис. 2, где v^R является биекцией между B^R и $C^R \cap [L, U]$. $(\cdot)_b^R$ является её обратной функцией на B^R и определена на множестве C^R.

Без ограничения общности, рассмотрим случай, когда $[L, U] \setminus C^R \neq \emptyset$.

Если множества C^Rи $[L, U]$ совпадают, следующее доказательство по-прежнему правильно (некоторые аргументы не принимают никаких значений и их можно не рассматривать). Соответственно, также предположим $L \le 0 \le U$.

Теперь рассмотрим множество $[L, U] \setminus C^R$, мощность которого не превышает $U - L + 1$. Поэтому мы произвольно выбираем несколько $|[L, U] \setminus C^R|$ различных элементов из любой области, которая не пересекается с B^R, и устанавливаем биекцию $v'(\cdot)$ между этими неявными элементами и множеством $[L, U] \setminus C^R$. Обозначим результирующий набор соответствующих отдельных неявных элементов как $\mathbb{Z}'_b$ и соответствующая обратная биекция $v'(\cdot)$ как $(\cdot)'_b$. Теперь мы готовы ввести определение восстановленной модели M (из реализации R)

исходной формулы F, которая задана на рис. 3. Далее докажем необходимые свойства этого определения.

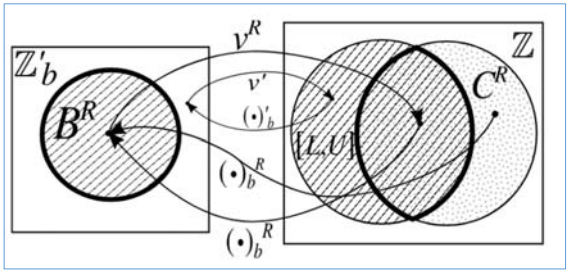

Рис. 2. Расширение биекции между множествами B^R и $C^R \cap [L,U]$ на множества $\mathbb{Z}_b^M = B^R \cup \mathbb{Z}_b{}'$ и $[L,U]$
Fig. 2. Extending the bijection between B^R and $C^R \cap [L,U]$ to the whole sets $\mathbb{Z}_b^M = B^R \cup \mathbb{Z}_b{}'$ and $[L,U]$

$$
\begin{array}{l}
\mathbb{Z}_b^M = B^R \cup \mathbb{Z}'_b, \\
v^M(a) = \begin{cases} v^R(a), & a \in B^R, \\ v'(a), & a \notin B^R, \end{cases} \\
(c)_b^M = \begin{cases} (c)_b^R, & c \in C^R, \\ (c)'_b, & c \in [L,U] \setminus C^R, \\ \epsilon\, \mathbb{Z}_b^M, & c \notin C^R \cup [L,U], \end{cases} \\
a +_b^M b = \begin{cases} a +_b^R b, & (a,b) \in \{(t^R,u^R) \mid t +_b u \in F^*\}, \\ \left(v^M(a) + v^M(b)\right)_b^M, & \\ \qquad (a,b) \notin \{(t^R,u^R) \mid t +_b u \in F^*\}, \\ \qquad L \le v^M(a) + v^M(b) \le U, \\ (0)_b, & \end{cases} \\
c \times_b^M a = \begin{cases} c \times_b^R a, & (c,a) \in \{(n,t^R) \mid n \times_b t \in F^*\}, \\ \left(c \times v^M(a)\right)_b^M, & \\ \qquad (c,a) \notin \{(n,t^R) \mid n \times_b t \in F^*\}, \\ \qquad L \le c \times v^M(a) \le U, \\ (0)_b, & \end{cases} \\
a \le_b^M b = v^M(a) \le v^M(b).
\end{array}
$$

Рис. 3. Определение восстановленной модели M в теории BLIA
Fig. 3. Definition of the reconstructed model M in BLIA

Лемма 11. *Восстановленная модель M на рис. 3 однозначно определена.*

Доказательство. Восстановленная модель M, показанная на рис. 3, включает в себя определение области $\mathbb{Z}_b$ ограниченных целых, а также определения для всех функций из соответствующей сигнатуры теории BLIA $\{+_b, \times_b, \le_b, v(\cdot), (\cdot)_b\}$. Таким образом, мы должны показать, что функции, определенные как на рисунке, действительно отображают любую комбинацию своих аргументов, взятые из соответствующих доменов в элементы из соответствующих диапазонов. Случаи для функции v и предикат $\le_b$ очевидны. Тем не менее,

мы должны убедиться, что выбранное множество $\mathbb{Z}_b^M$ действительно содержит все значения, взятые функциями $(\cdot)_b^M$, $+_b^M$ и $\times_b^M$, иначе наше определение модели противоречиво.

Рассмотрим случаи для функции $(\cdot)_b^M$. В первом случае $(c)_b^R \in B^R$ для любого $c \in C^R$ согласно лемме 10, поэтому $(c)_b^R \in \mathbb{Z}_b^M$. Во втором случае $(c)'_b \in \mathbb{Z}'_b \subseteq \mathbb{Z}_b^M$ согласно построению.

В третьем случае $\epsilon\mathbb{Z}_b^M \in \mathbb{Z}_b^M$ по определению эпсилон-оператора Гильберта ϵ (выбор произвольного элемента непустого множества), так как $|\mathbb{Z}_b^M| \ge |\mathbb{Z}'_b| = |[L,U] \setminus C^R| > 0$ (в общем случае либо $\mathbb{Z}'_b$, либо B^R не пусты).

Теперь рассмотрим функцию $+_b^M$. Второй и третий случаи ($(v^M(a) + v^M(b))_b^M$ и $(0)_b$) явно определены, поскольку $(c)_b^M$ явно определен для любого c, как показано выше.

Во-первых, в случае, если мы поступаем по индукции по правилам инстанцирования и показываем, что терм вида $t+_b u$ может возникать только во время инстанцирования, когда он уже содержится в F (единственная аксиома (A1) инстанцируется всякий раз, когда $t+_b u$ содержится в F). Затем из $t+_b u \in F^*$ следует $t+_b u \in F$ и по лемме 1, $v(t+_b u) \in F^*$. Поскольку $a = t^R$, $b = u^R$, $a+_b b = t^R +_b^R u^R = (t+_b u)^R \in B^R \subseteq \mathbb{Z}_b^M$ по определению B^R.

Доказательство для функции $\times_b^M$ аналогично. □

Лемма 12. *Аксиомы (A4), (A5) и (A6') теории BLIA сохраняются в модели M.*

Доказательство. Рассмотрим аксиому (A4) для произвольного $a \in \mathbb{Z}_b^M = B^R \cup \mathbb{Z}'_b$. В случае $a \in B^R$, $v^M(a) = v^R(a) \in [L,U]$ по лемме 5.

В случае, если $a \in \mathbb{Z}'_b$, $v^M(a) = v'(a) \in [L,U] \setminus C^R \subseteq [L,U]$ согласно построению.

Рассмотрим аксиому (A5) для произвольного целого числа c. Согласно правилу инстанцирования аксиомы, мы должны показать равенство $v^M((c)_b^M) = c$ для любого $c \in [L,U] = (C^R \cap [L,U]) \cup ([L,U] \setminus C^R)$. В случае $c \in C^R \cap [L,U]$, используя определения на рисунке 3 и в лемме 9 мы получаем $(c)_b^M = (c)_b^R \in B^R$, а также $v^M((c)_b^M) = v^R((c)_b^R) = c$. В случае $c \in [L,U] \setminus C^R$ мы получим $(c)_b^M = (c)'_b \in \mathbb{Z}'_b$и $v^M((c)_b^M) = v'((c)'_b) = c$ согласно построению.

Наконец, в случае аксиомы (A6') мы должны показать $(v^M(a))_b^M = a$ для любого $a \in \mathbb{Z}_b^M = B^R \cup \mathbb{Z}'_b$. В случае $a \in B^R$ по лемме 9 имеем $v^M(a) = v^R(a) \in C^R \cap [L,U]$и $(v^M(a))_b^M = (v^R(a))_b^R = a$. В случае, если $a \in \mathbb{Z}'_b$, получаем $v^M(a) = v'(a) \in [L,U] \setminus C^R$ и $(v^M(a))_b^M = (v'(a))'_b = a$ согласно построению. □

Лемма 13. *Аксиомы (A1), (A2) и (A3) теории BLIA сохраняются в модели M.*

Доказательство. Сначала мы проверим аксиому (A1). Рассмотрим случаи в определении операции $+_b^M$.

Рассмотрим случай $(a,b) \in \{(t^R,u^R) \mid t+_b u \in F^*\}$. В доказательстве леммы 11 (последний абзац) показано, что из $t+_b u \in F^*$ следует $t+_b u \in F$. По правилу инстанцирования аксиомы (A1) заключаем, что $L \le v(t) + v(u) \le U \Rightarrow v(t+_b u) = v(t) + v(u)$является основным подтермом F^* и, поскольку R является моделью F^*, мы имеем $L \le v^R(t^R) + v^R(u^R) \le U \Rightarrow v^R(t^R +_b^R u^R) = v^R(t^R) + v^R(u^R)$.

Из факта $(a,b) \in \{(t^R,u^R) \mid t+_b u \in F^*\}$ следует $a = t^R$ и $b = u^R$, поэтому $L \le v^R(a) + v^R(a) \le U \Rightarrow v^R(a+_b^R b) = v^R(a) + v^R(b)$.

Теперь, так как $t+_b u \in F$, мы имеем $t \in F$, $u \in F$, и в силу леммы 1, $v(t+_b u) \in F^*$, $v(t) \in F^*$ и $v(u) \in F^*$. Это означает $a+_b^R b = t^R +_b^R u^R = (t+_b u)^R \in B^R$, $a = t^R \in B^R$ и $b = t^R \in B^R$ по определению B^R. Таким образом, согласно определениям на рисунке 3, $L \le v^M(a) + v^M(b) \le U \Rightarrow v^M(a+_b^M b) = v^M(a) + v^M(b)$.

Теперь рассмотрим случай $L \le v^M(a) + v^M(b) \le U$. Из леммы 12 аксиома (A5) сохраняется в M и, следовательно, $L \le c \le U \Rightarrow v^M((c)_b^M) = c$ для любого c и, в частности, $L \le v^M(a) + v^M(b) \le U \Rightarrow v^M((v^M(a) + v^M(b))_b^M) = v^M(a) + v^M(b)$.

По определению $+_b^M$ на рисунке 3 получаем $L \le v^M(a) + v^M(b) \le U \Rightarrow v^M(a +_b^M b) = v^M(a) + v^M(b)$.

Наконец, в случае $v^M(a) + v^M(b) \notin [L, U]$, предположение $L \le v^M(a) + v^M(b) \le U$ не выполняется и, следовательно, аксиома (A1) очевидно сохраняется.

Доказательство для аксиомы (A2) аналогично и в случае (A3) непосредственно следует из определения $\le_b^M$ на рис. 3□

Лемма 14. *Модель M может быть расширена неинтерпретируемыми константами, которые содержатся в F такие, что для любого подтерма $t \in F$ его интерпретации в модели M и в реализации R совпадают, то есть $t^M = t^R$.*

Доказательство. Без ограничения общности рассмотрим случай, когда t не содержит вхождения неинтерпретированных функций с арностью больше нуля. Расширение этого доказательства на термы с неинтерпретируемыми функциями (а не константами) очевидно. Доказательство проводится индукцией по структуре терма t.

Мы начнем с рассмотрения подтермов нулевой арности, то есть констант.

Интерпретируемые константы (числа) имеют одинаковые фиксированные интерпретации как в R, так и в M. Выберем интерпретацию неинтерпретируемых констант, встречающиеся в F, они будут одинаковыми в обоих моделях M и R. Это согласуется, поскольку если терм t содержится в F, то по лемме 1, $v(t)$ встречается в F^* и поэтому $t^R \in B^R \subseteq \mathbb{Z}_b^M$.

Аналогично, если t является аргументом функции v, который содержится в F и $t^M = t^R$, то $t^R \in B^R$ и по определению на рисунке 3 $v^M(t^M) = v^R(t^R)$, что по определению означивания дает $(v(t))^M = (v(t))^R$. Если $(n)_b$ входит в F, то по лемме 3 он также содержится в F^* и, таким образом, $n^R \in C^R$. Следовательно, если $n^M = n^R$ тогда $((n)_b)^M = (n^M)_b^M = (n^R)_b^M = (n^R)_b^R = ((n)_b)^R$.

Если $n \times_b t$ встречается в F, то по построению инстанцированная формула также содержится в F^*.

Если, кроме того, $t^M = t^R$, то $(n \times_b t)^M = n^M \times_b^M t^M = n^R \times_b^M t^R = n^R \times_b^R t^R = (n \times_b t)^R$. То же относится и к $+_b$.

Наконец, если $t \le u_b$ содержится в F, то согласно правилам инстанцирования, аксиома (A3) была инстанцирована с t и u, следовательно, $t^R \le_b^R u^R \Leftrightarrow v^R(t^R) \le v^R(u^R)$.

По предположению индукции имеем $v^M(t^M) = v^R(t^R)$ и $v^M(u^M) = v^R(u^R)$. Кроме того, по определению на рис. 3 у нас есть $t^M \le_b^M u^M \Leftrightarrow v^M(t^M) \le v^M(u^M)$.

Таким образом, $t^M \le_b^M u^M \Leftrightarrow v^R(t^R) \le v^R(u^R) \Leftrightarrow t^R \le_b^R u^R$.□

Теперь полнота процедуры инстанцирования напрямую следует из вышеуказанных лемм.

Теорема 2. *Каждая формула F без свободных переменных выполнима в теории BLIA тогда и только тогда, когда её трансляция F^* выполнима в QF_UFLIA.*

Доказательство. Если F выполнима в BLIA, то F^* выполнима в QF_UFLIA из-за корректности процедуры инстанцирования.

Предположим, что F^* выполнима в QF_UFLIA с моделью R. Затем по леммам 11, 12 и 13 восстановленная модель M является моделью теории BLIA. Кроме того, по лемме 14, вся формула F как терм имеет одинаковое значение в обоих R и M, расширенная соответствующими неинтерпретированными константами. Так как F является основным подтермом F^* и, следовательно, является истиной в R, она также выполнима в M. □

4. Формализация.

Доказательство полноты представлено в предыдущем разделе и было оформлено в Isabelle/HOL. В этой формализации мы использовали безтиповый синтаксис над формулами (untyped deep embedding), чтобы применять простую индукцию с двумя возможными

конструкциями (функция и переменная) в структуре формул и составить обоснование интерпретаций этих формул в различных моделях объектной логики.

Существуют два наиболее заметных различия, отличающих полностью формализованное доказательство в Isabelle/HOL от доказательств в статье.

Во-первых, использование безтипового синтаксиса над формулами (untyped deep embedding) значительно упрощает структурное представление формул, позволяет представлять бессмысленные искаженные формулы и потенциально интерпретировать их в объектной логике. Например, такие формулы: $((a)_b \le n) + (b)_b \times_b 2$, где каждый символ функции применяется по крайней мере к одному неупорядоченному типу, который может быть представлен в нашем синтаксисе. Чтобы исключить такие формулы, мы сформулировали явное ограничение однозначной определенности (в форме специального предиката) и использовали его в качестве предварительных условий в различных леммах, а также в нашем определении выполнимости.

В нашем подходе интерпретация может только моделировать однозначно определенную формулу.

Вторая проблема не так очевидна. Традиционное определение интерпретации для формулы с кванторами (такой как аксиома) включает в себя дополнительный параметр, обычно называемый означиванием μ, который отображает неявные переменные в соответствующие им интерпретации. Таким образом, с помощью этого определения пришлось переопределять не только формулы и интерпретации (модели), но и соответствующие означивания. В этом стиле рассуждений установление казалось бы, тривиального правила подстановки, т.е. если $\forall a. F^M(a)$ мы имеем $F^M(x^M) = (F(x))^M$ если a не свободная переменная в x, требует определения промежуточного означивания $\mu' = \mu \circ [a \mapsto x^M]$ и оценивая формулы F в модели M с обоими означиваниями μ и μ'. Вместо этого мы используем определение интерпретации, основанное на подстановке, и формализуем следующим образом:

$\forall a \in \mathcal{D}. F^M(a) \Leftrightarrow (\forall t. vars(t) = \emptyset \Rightarrow ([t/a]F(a))^M)$, где любая модель M должна удовлетворять следующему ограничению:

$\forall a \in \mathcal{D}. \exists t. t^M = a$. Здесь $\mathcal{D}$ - множество, выбранное для интерпретации термов (в частности, переменных) соответствующего типа, $[t/a]$ обозначает подстановку переменной a термом t. Поскольку терм t с подстановкой не имеет свободных переменных, то нет необходимости в дополнительных методах по предотвращению замыкания переменных. Более того, в нашей формализации нам никогда не требовались вложенных кванторов, поэтому мы использовали абстрактные схематические переменные вместо кванторов и, таким образом, значительно упростили представление формулы. В самом деле, мы еще больше упростили наш подход на основе замещения путем ограничения формы целочисленных ограниченных целых термов t в подстановке.

Если $\forall a \in \mathbb{Z}_b. \exists c \in \mathbb{Z}. (c)_b = a$, мы можем перейти от кванторов над ограниченными целыми термами t к кванторам над целыми:

$(\forall t. vars(t) = \emptyset \Rightarrow ([t/a]F(a))^M) \Leftrightarrow \forall c \in \mathbb{Z}. (F(c)_b)^M$, таким образом, полностью изменяя использование кванторов в объектной логике (в BLIA/QF_UFLIA) в металогику с использованием кванторов в HOL. Это наиболее общая форма объектной логики с кванторами, которую мы используем в нашем формальном доказательстве [9] (соответствующий документ Isabelle называется TSMT_Bound_Complete).

5. Сравнение с другими тактиками

Мы реализовали решающую процедуру на основе процедуры инстанцирования, описанную в этой статье, внутри системы Isabelle в тактике расширения SMT с помощью инстанцирования кванторов триггерами.

Данная тактика называется TSMT [9] и не описана в этой работе. Здесь мы лишь кратко отметим, что это позволяет выполнить предварительное подготовку текущей цели (будучи формулой с кванторами) со всеми инстанцированиями триггеров в соответствующих подформулах в цели.

Кванторы представлены в виде лемм и триггеры подформул могут быть определены с использованием специальных атрибутов леммы.

Тактика также поддерживает корректное восстановление доказательства для формулы с инстанцированиями (используя существующие возможности тактики Z3 для доказательства в Isabelle), а также извлечение модели (контрпримера) текущей цели и показывая полученную модель для пользователя.

Оценка конкретной тактики для системы автоматизированных доказательств – это непростая задача, так как имеется не так много легкодоступных контрольных показателей для таких инструментов и подавляющее большинство доступных доказательств уже специально адаптированы для использования существующих тактик со всеми их особенностями и ограничениями. Ближайшая доступная тактика в Isabelle/HOL, которая предоставляет аналогичные возможности, это тактика uint_arith из теории HOL-Word. Она также пытается инстанцировать текущую цель дополнительными предположениями об арифметике на ограниченных целых числах и полагается на работу по упрощению цели по стандартной арифметической тактике (стандартная процедура упрощения, presburger и linarith).

Тем не менее, в отличие от нашего подхода, трансляция, реализованная в тактике uint_arith, вообще говоря, не является полной. Тактика uint_arith не очень широко используется в большинстве практических приложений, поскольку ее возможности аналогичны возможностям тактики unat_arith (похожая тактика для натуральных, а не целых чисел), которая часто используется вместо uint_arith.

Поскольку арифметика натуральных чисел не реализуется напрямую в большинстве решателей SMT, мы разрабатывали нашу тактику на арифметике целых чисел. Таким образом, чтобы оценить нашу реализацию, мы взяли несколько наиболее трудоемких целей, решаемых с помощью тактики unat_arith и вручную преобразовали их в соответствующие цели, подходящие как для uint_arith, так и для нашей тактики (преобразование в основном чисто синтаксическое, кроме добавления некоторых недостающих ограничений вида $n \geq 0$). Цели были взяты из реальных примеров с участием верифицированных функций на языке Си, таких как memcpy и quicksort, формализованные в рамках AutoCorres [10]. Пример включает в себя более 130 вызовов uint_arith, из которых мы выбрали 11 наиболее трудоемких. В добавок к существующим примерам мы добавили несколько очень простых лемм в целочисленной арифметике с ограничениями, где тактика uint_arith не может доказать цель.

Результаты оценки показаны в табл. 1. Время решения задач показывается в секундах, если оно превышает 0,1, в противном случае время помечено как < 0.1. Время для целей, не решаемых с помощью тактики, обозначается как «-» вместе с требуемым временем на попытку до возврата к результирующему состоянию доказательства (инстанцированная цель в случае uint_arith). В Isabelle/HOL процесс доказательства основан на интерактивных формальных документах, время решения важно, так как оно напрямую влияет на общее время, требуемое для обновления модели документа на каждое взаимодействие с пользователем (особенно для неструктурированных доказательств).

Табл. 1. Оценка тактики TSMT BLIA и uint_arith на формулах и реальных подзадачах из примеров AUTOCORRES.

Tabl1e 1. Evaluation of TSMT BLIA and uint_arith tactics on sample formulas and real subgoals from AUTOCORRES examples.

Цель	**Время работы, сек.**	
	uint_arith	
memcpy wp' 1	0.366	0.329

memcpy wp' 2	< 0.1	0.450
memcpy word 1	< 0.1	0.320
memcpy word 2	< 0.1	0.350
memcpy wp' 3	0.201	0.320
memcpy wp' 4	0.508	0.315
memcpy wp' 5	0.250	0.509
partition correct 1	0.468	0.581
partition correct 2	0.401	0.441
qsort unat sub sub1	< 0.1	0.526
quicksort correct 1	0.251	0.493
simple example	– (< 0.1)	0.196
simple 2a mn 1 pl b	– (1.202)	0.860
simple div2 mul2	– (0.102)	0.756
simple 2 min mn 1 pl min	– (0.897)	< 0.1
simple div3 mul3	– (0.145)	0.779

Результаты демонстрируют, что наша реализация не значительно медленнее, чем существующая тактика, несмотря на использование исчерпывающей процедуры инстанцирования, вызов внешних решателей (Z3), а также парсинг и восстановление полученных доказательств. Более того, это обеспечивает несколько заметных преимуществ.

- Тактика обладает свойством полноты, поэтому он гарантированно решит любую цель в пределах определенного класса, в то время как тактика uint_arith не полна и ее способность решить конкретную цель не так легко предсказать;
- Благодаря своей полноте, наша тактика способна не только верифицировать правильную цель, но, что еще важнее, осмысленно опровергать неверные цели в определенном классе (где тактика полна). Эта способность важна в контексте итеративной разработки доказательства, где это широко известно, что промежуточные попытки доказать неправильные утверждения более распространены, чем неудачи из-за неполноты тактики/метода доказательства.
- Наша тактика имеет примерно одинаковую производительность в случае успешной попытки доказательства и в случае поиска контрпримера, потому что оно ведет себя одинаково в обоих случаях, в то время как uint_arith последовательно пытается решить инстанцированную цель со всеми зарегистрированными арифметическими тактиками и может потратить дополнительное время на неудачные попытки.
- Наш подход легко расширяемый, так как семантика новых функциональных символов может быть легко дополнена соответствующими леммами с их триггерами. На самом деле, легкость расширяемости нашей тактики сопоставима со стандартным упрощением в Isabelle. В примерах мы расширили нашу тактику поддержкой целочисленного деления, вычитания, максимума и минимума, чтобы соответствовать возможностям стандартных арифметических тактик в Isabelle.
- Расширение нашей тактики не требует программирования на языке ML, тактика реализована поверх общей тактики TSMT (основная часть нашего инструмента инстанцирования), просто как группа лемм с соответствующими инстанцированием триггеров, в то время как uint_arith реализуется непосредственно в Isabelle/ML.

Основное ограничение нашей текущей реализации в сравнении с существующей тактикой uint_arith – невозможность правильно обрабатывать кванторы, вложенные в саму цель и не дополнены соответствующими триггерами. Наша тактика в настоящее время игнорирует любые кванторы, присутствующие в цели. Примеры, упомянутые на рисунке 4, доступны в нашем примере теории [9] (соответствующий документ Isabelle называется TSMT_Bounded_Examples).

6. Дальнейшие исследования.

Основными направлениями будущей работы включают разработку некоторых предсказуемых (хотя и неполных) подходов к обработке кванторов, встречающиеся в цели без соответствующих триггеров (стратегии для обработки кванторов внутри существующих решателей очень эффективны, но редко предсказуемы). Еще одно важное направление это формализация и оценка решения, основанного на реализации процедуры для других разрешимых теорий, таких как эффективно разрешимый фрагмент теории бит-векторов, модульная линейная целочисленная арифметика, разрешимый фрагмент теории, формализующей различные операции над списками, теория адресной арифметики с ограниченными адресами (но с Си-подобным разделением на непересекающиеся блоки памяти), теория интерпретируемых множеств (для которых есть несколько доказательств полноты, но без формализации в системе интерактивных доказательств) и прочие практические актуальные теории и их фрагменты.

7. Заключение.

В работе представлена теория ограниченных целых, позволяющая составить решающую процедуру в Isabelle/HOL на основе решателей SMT. Эта решающая процедура позволяет проверить некоторые конструкции языка программирования Си. Стоит заметить, что разработка решающих процедур по множеству различных теорий имеет важное значение при верификации программного обеспечения. Мы охарактеризовали сложность проблемы выполнимости и обеспечили эффективное сокращение сложности решения в QF_UFLIA. Наш результат показывает, что более эффективно разрабатывать специализированные алгоритмы, чем применять общий алгоритм для арифметики Пресбургера. Кроме того, в нашем случае число инстанцирований является линейным относительно длины входной формулы.

Список литературы / References

[1]. T. Nipkow, M. Wenzel, and L.C. Paulson (editors). Isabelle/HOL: A Proof Assistant for Higher-Order Logic. Lecture Notes in Computer Science book series, vol. 2283, 2002, 218 p.

[2]. J. Dawson. Isabelle theories for machine words. Electronic Notes in Theoretical Computer Science, vol. 250, no. 1, 2009, pp. 55-70.

[3]. D. Babic and M. Musuvathi. Modular arithmetic decision procedure. Technical Report MSR-TR-2005-114, Microsoft Research, 2005.

[4]. N. Bjørner, A. Blass, Y. Gurevich, and M. Musuvathi. Modular difference logic is hard. Technical Report MSR-TR-2008-140, Microsoft Research, 2008.

[5]. B.-Y. Wang. On the satisfiability of modular arithmetic formulae. Lecture Notes in Computer Science book series, vol. 4218, 2006, pp. 186–199.

[6]. S. Böhme. Proof reconstruction for Z3 in Isabelle/HOL. In Proc. of the 7th International Workshop on Satisfiability Modulo Theories (SMT ’09), 2009.

[7]. S. Böhme and T. Weber. Fast LCF-style proof reconstruction for Z3. Lecture Notes in Computer Science book series, vol. 6172, 2010, pp. 179–194.

[8]. C. Barrett, P. Fontaine, and C. Tinelli. The SMT-LIB Standard: Version 2.6. Department of Computer Science, The University of Iowa, Technical Report, 2017.

[9]. R. Sadykov and M. Mandrykin. Completeness of instantiation procedure for bounded linear integer arithmetic. Formalization in Isabelle/HOL. Anailable at: https://forge.ispras.ru/projects/tsmt/repository/, accessed April 2020.

[10]. D. Greenaway, J. Andronick, and G. Klein. Bridging the gap: Automatic verified abstraction of C. Lecture Notes in Computer Science book series, vol. 7406, 2012, pp. 99–115.

[11]. Kovásznai G. How Hard is Bit-Precise Reasoning? In Proc. of the 10th International Conference on Applied Informatics, 2017. pp. 179-190.

Информация об авторах / Information about authors

Рафаэль Фаритович САДЫКОВ – стажер-исследователь в ИСП РАН, аспирант кафедры МаТИС механико-математического факультета МГУ им. М.В. Ломоносова. Сфера научных интересов: верификация программ, теория распределенных вычислений, теория графов, теория автоматов, математическая логика.

Rafael Faritovich SADYKOV – intern researcher of ISP RAS, PhD student of Faculty of Mechanics and Mathematics, Moscow State University. Research interests: program verification, distributed computing theory, graph theory, automata theory, mathematical logic.

Михаил Усамович МАНДРЫКИН – младший научный сотрудник ИСП РАН, кандидат физико-математических наук по специальности «математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей». Сфера научных интересов: формальные методы верификации программ, математическая логика, функциональное программирование, системы типов в языках программирования.

Mikhail Usamovich MANDRYKIN – researcher at ISP RAS, PhD. Research interests: formal methods, program verification, mathematical logic, functional programming, type systems.