
Конструктивная математика: обзор достижений, недостатков и уроков. Часть III

Н. Н. НЕПЕЙВОДА

ABSTRACT. Constructivist concepts not assuming that we know programs for each effective method (intuitionism et al) are analyzed here.

Keywords: intuitionism, constructivism, realizability, computability, application of constructivism

1 Введение

В данной работе рассматриваются конструктивные направления в математике, как предполагающие, что имеется полная информация об исходном тексте алгоритма (советский конструктивизм), так и не предполагающие, что алгоритм каждого эффективного преобразования полностью известен (интуиционизм и др.). Обращается внимание на то, как теоретические абстрактные положения и выбор идеальных объектов связаны с практикой, какие результаты могут быть полезны как ориентирующие, и какие из напрашивающихся прямых применений ведут в тупик.

Конструктивизм рассматривается здесь не как единственно истинное мировоззрение, а как один из вариантов точного рационального мышления, максимально чистой и глубокой реализацией которого является математика, и прежде всего как материал, на котором можно показать одновременно переменность понятия рациональности и глубочайшие практические и мировоззренческие последствия принятия любого из вариантов,

делающие невозможной как безразличную «терпимость» современного вульгарного мировоззрения, так и самоуверенность тех, кто отождествляет свой вариант логики с истиной, свой рассудок с разумом.

Работа является третьей частью единого исследования, первые две части которого опубликованы в [6, 7]. Понятия и определения из этих работ используются в данной.

2 Недостатки советского конструктивизма

В предыдущих статьях из данной серии было показано, какие преимущества дает предположение, что известен алгоритм для любого эффективного метода.

Покажем отрицательные эффекты этого знания. Парадоксальным результатом, влекущим многие неприятные следствия в конструктивном математическом анализе, является теорема о сингулярных покрытиях, установленная Заславским и Цейтиным [1] и независимо Крайзелем и Лакомбом [20].

Используя прекрасно разработанную в информатике технику кодирования, можно закодировать программу каждого алгоритма натуральным числом и представить в виде списков натуральных чисел рациональные числа. Поэтому в технические подробности такого кодирования мы вдаваться ни здесь, ни в дальнейшем не будем. Фиксируем какую-то функцию кодирования программ p и кодирования рациональных чисел q . Рациональный интервал — кортеж из двух рациональных чисел, второе из которых больше. Проверить, является ли объект рациональным интервалом и вычислить его длину можно алгоритмически. Предикат проверки обозначим $RATINTERV$, алгоритм вычисления длины RLH .

Фиксируем некоторое натуральное число n_0 . Построим следующее перечислимое множество.

$$X = \left\{ p \mid \begin{array}{l} \neg(RATINTERV(\Upsilon p(p))) \\ \&(RLH(\Upsilon p(p)) < \frac{1}{2^{n_0+3+(p)}}) \end{array} \right\}$$

Если сдвинуть любую быстро стягивающуюся последовательность рациональных сегментов на $n_0 + 3$, то получится элемент

этого множества. Значит, оно бесконечно. Поскольку это множество программ, оно счетно-перечислимо. Можно построить алгоритмическую последовательность φ , перечисляющую его без повторений. Поскольку для каждого действительного числа оно, сдвинутое на n_0 , попадет в первые члены этой последовательности по принципу конструктивного подбора можно найти для него покрывающий сегмент, перечисляемый этой последовательностью. А сумма длин всех таких сегментов (если взять ее в классическом смысле, поскольку в конструктивном это будет шпекерова последовательность) будет меньше $\frac{1}{2^n}$.

Немного улучшая это построение, покрывая действительную ось сегментами, сумма длин которых (в классическом смысле) меньше любого заданного $\varepsilon > 0$, пересекающимися лишь своими концами, и применяя конструкцию, позволяющую соединить две конструктивные функции, определенные на сегментах, пересекающихся в одной точке, легко построить всюду определенную и непрерывную на замкнутом отрезке $[0, 1]$, но неограниченную, конструктивную функцию. Грубо говоря, она имеет точками разрыва неалгоритмические числа.

Рассмотрение алгоритмической непрерывности выявило различие двух понятий непрерывности в классической математике: на языке последовательностей и на языке окрестностей. Непрерывность на языке последовательностей означает, что

$$\lim_{n \rightarrow \infty} (f x_n) = (f \lim_{n \rightarrow \infty} x_n).$$

Эту теорему легко доказал А. А. Марков, но в отрицательной формулировке: что значение конструктивной функции в точке предела не может отличаться от предела ее значений. А вот непрерывность на языке окрестностей доказывается тяжело. Но заниматься этим нет смысла, поскольку неограниченная на $[0, 1]$ функция лишь один из безобиднейших монстров, рождающихся в алгоритмической теории действительных чисел в версии советского конструктивизма. Однако выявившаяся разница двух понятий непрерывности важна для понимания многих тонких явлений в практических вычислительных алгоритмах.

Укажем еще несколько мест, где конструктивная алгоритмическая математика показала свою высокую аналитическую силу.

В классической математике по традиции игнорируется проблема эквивалентных представлений данных. Фактор-множество состоит из классов эквивалентности, и «вопрос снят». В алгоритмическом конструктивизме классы эквивалентности не могут считаться элементарным понятием, поскольку они не являются конструктивными объектами. Поэтому пришлось детально разбираться со строением пространств данных, у которых имеется много различных представлений в виде конструктивных объектов. Выяснилось, что это исключительно сильно влияет на алгоритмы обработки данных. Не обязательно даже иметь алгоритм проверки эквивалентности, важно иметь систематические способы строить алгоритмы, гарантированно сохраняющие эквивалентность (алгоритмические морфизмы структур данных). Результаты этого анализа до сих пор не в полной мере использованы ни информатикой, ни другими направлениями конструктивной математики.

ПРИМЕР 1. Как было замечено в части I (примеры 14, 15), не всегда конструктивные пространства описываются классической топологией. В частности, пример 15 показывает, что распространенная и безоглядно применяемая в современной машинной графике топологическая операция склейки пространств на самом деле содержит нетривиальные подводные камни. Пространства $[0, 1] \cup [1, 2]$ и $[0, 2]$ конструктивно не изоморфны. В частности, нет ни одного всюду определенного конструктивного отображения $[0, 2] \rightarrow [0, 1] \cup [1, 2]$. Интересно то, что с классической точки зрения алгоритмические отображения $[0, 1] \cup [1, 2] \rightarrow [0, 1] \cup [1, 2]$ такие же, как $[0, 2] \rightarrow [0, 2]$. Но алгоритмически мы получаем целый класс новых алгоритмов, вычисляющих те же функции, порою с меньшей затратой ресурсов. Пространства, получающиеся склейкой, удалось классически интерпретировать (но не до конца адекватно) лишь как топосы.

ПРИМЕР 2. Пример 14 из [6] (принадлежащий Шанину) заодно показывает роль явной и неявной информации. В советской конструктивной математике по привычке, перешедшей в традицию, действительное число задавалось парой алгоритмов: сходящая-

ся последовательность и ее «регулятор сходимости», выдающий по каждому n такое N , что

$$\forall m_1, m_2 : \text{Nat} \left(m_1 > N \& m_2 > N \Rightarrow |(a\ m_1) - (a\ m_2)| < \frac{1}{2^n} \right).$$

Если этот регулятор не задать явно, вычисления и преобразования становились более трудными, а то и невозможными. Это еще раз показало необходимость снабдить алгоритм всей необходимой исходной информацией, даже если она в принципе может быть получена в ходе вычислений.

Далее, явная информация о программах, точное понятие алгоритма позволила многие отрицательные результаты интуиционизма сделать явными контрпримерами, не зависящими от наших знаний и незнаний. Такова, например, приведенная ранее последовательность Шпекера. Более того, выявилась разная сила отрицательных результатов. Существование предела монотонной последовательности и существование максимального значения равномерно непрерывной функции на отрезке опровергаются на примере: явно строится соответствующая последовательность (функция). А вот построить функцию, имеющую разные знаки на концах отрезка, всюду определенную на нем и не имеющую нуля, не удастся. Легко (методом, описанными в [6]) доказать несуществование алгоритма преобразования каждой такой функции в приближение к ее корню. Но можно доказать также, что предположение о несуществовании корня противоречиво. Сделаем это.

ЛЕММА 1. Пусть f — функция действительных чисел, всюду определенная на $[0, 1]$ и такая, что $(f\ 0) < 0$, $(f\ 1) > 0$. Тогда $\neg\neg\exists x : \text{Real}(f\ x) = 0$.

ДОКАЗАТЕЛЬСТВО. Прежде всего покажем, что из принципа конструктивного подбора следует возможность снятия двойного отрицания с отношения $x > y$. Это отношение означает, что есть некоторое приближение к x : $[a_n, b_n]$, такое, что для соответствующего приближения к y : $[c_n, d_n]$ выполнено $a_n > d_n$.

Отношение порядка на рациональных числах разрешимо. Значит, по принципу конструктивного подбора, $\neg\neg x > y \Rightarrow x > y$. Точно так же показывается возможность снятия двойного отрицания с отношения $x \# y$ « x и y различны», уточняемого как «для них существуют непересекающиеся приближения». Отсюда, при выполнении принципа конструктивного подбора, следует $\forall x, y : \text{Real}(\neg x = y \Rightarrow x \# y)$. Легко видеть, что в интуиционистской логике $\forall x, y : \text{Real}(x \# y \Rightarrow x < y \vee x > y)$.

Теперь предположим, что

$$\neg\exists x : \text{Real}(f x) = 0.$$

Тогда

$$\forall x : \text{Real}\neg(f x) = 0.$$

Значит,

$$\forall x : \text{Real}(f x) \# 0.$$

Отсюда

$$\forall x : \text{Real}(f x) > 0 \vee (f x) < 0.$$

Следовательно, методом деления пополам строим x_0 , такое, что на каждом отрезке от $x_0 - 2^{-n}$ до $x_0 + 2^{-n}$ функция меняет знак. Тогда $(f x_0) = 0$ по непрерывности f . Значит,

$$\neg\exists x : \text{Real}(f x) = 0 \Rightarrow \exists x : \text{Real}(f x) = 0,$$

что эквивалентно заключению теоремы. \square

Таким образом, неконструктивные теоремы делятся на ряды.

1. Слегка неконструктивные, когда алгоритм «почти что» есть, и случаи, когда он не работает, часто списывают на ошибки реализации (например, теорема о существовании нуля конструктивной функции, меняющей знак). Формальный признак: в конструктивной математике доказывается отрицание теоремы и одновременно несуществование контрпримера.

2. Сильно неконструктивные, когда приводится явный пример, опровергающий алгоритмичность. Например, теорема о существовании предела монотонной ограниченной сверху последовательности.
3. Совершенно неконструктивные, когда из предположения о существовании метода решения такой задачи следует закон исключенного третьего. Например, это теорема о верхней грани.
4. Грязные, когда метода построения нет даже при использовании абстракции актуальной бесконечности. Например, таковы теорема о существовании неизмеримого множества действительных чисел и теорема о существовании модели классической теории в несчетной сигнатуре.

Еще одним положительным фактором было то, что советский конструктивизм впервые начал систематически рассматривать не всюду определенные функции в качестве основной модели. Дальнейшее развитие теоретической информатики (в частности, модели Скотта) показало, что предположение о возможности ошибки является краеугольным камнем хорошей теории в данной области. Слегка неконструктивные теоремы легко модифицируются в алгоритмы, работающие «почти всюду», и в советском конструктивизме было замечено, что многие неразрешимые задачи решаются, если допустить ошибку на «множестве меры нуль». Так, например, легко строится функция

$$(\operatorname{sgn} x) = \begin{cases} 1, & \text{при } x > 0; \\ -1, & \text{при } x < 0. \end{cases}$$

В нуле эта функция не определена.

Это наблюдение показало свою эффективность на практике при анализе высокопроизводительных и высокооптимизированных систем. Как правило, в них используются для некоторых критических задач «супероптимальные» алгоритмы, почти всегда работающие лучше теоретической оценки, но порою значительно хуже, а то и вылетающие. Здесь, в случае действительно хорошо сделанной системы, есть простой практический совет:

чуть-чуть изменить исходные данные или их представление (в случае баз данных). Опять же почти всегда после этого система вновь начинает прекрасно работать. Беда сильных программистов часто в том, что, относясь с недоверием к «теоретическим спекуляциям», они начинают «вылизывать» такую программу, делая заплатки в выявившихся частных случаях. Но каждое такое «улучшение» приводит к нескольким ухудшениям, и хорошая система начинает разваливаться. . .

Еще одним важнейшим достижением советского конструктивизма стало четкое разделение абстракций, лежащих в основе математики, и, соответственно, точное установление факта, что классическая логика полностью работает при обосновании уже найденного решения. Тем самым области действия классической и конструктивной логики оказались разделены.

Осознав базовые принципы, легко построить систему, последовательно их реализующую. Конструктивный подбор оказался в некотором смысле максимально сильным логическим решением, совместимым с абстракцией потенциальной осуществимости и концепцией конструктивизма. Разберем это положение подробнее.

Карнап предложил для «формализации» фиксированных и хорошо определенных структур данных, неформализуемых ввиду теоремы Гёделя о неполноте, правило вывода с бесконечным числом посылок. Приведем его для случая натуральных чисел.

$$\frac{A(0) \dots A(\mathbf{n}) \dots}{\forall x : \text{Nat } A(x)}$$

Если рассматривать это правило «классически», не интересуясь способом задания доказательств всех посылок, то оно очевидно формализует определение истинности в стандартной модели арифметики и приводит к полной системе. С. Феферман доказал, что в случае, если доказательства посылок правила порождаются алгоритмом, оно также полно. Правило Карнапа с алгоритмически порождаемыми доказательствами посылок называется ω -правилом.

А. Г. Драгалин (1978) показал, что в случае советского конструктивизма (интуиционистской арифметики с тезисом Чёрча

и принципом конструктивного подбора) ω -правило тоже дает полную систему. Таким образом, доказана в некотором смысле максимальность и полнота используемых советским конструктивизмом принципов, если ограничиваться конструктивными объектами, явно строящимися при помощи конечного числа операций из конечного списка из конечного набора исходных сущностей и такими, что их равенство просто и механически распознаваемо. Все такие объекты могут быть однозначно закодированы натуральными числами.

Тем самым доказано, что советский конструктивизм довел до логического конца концепцию потенциальной осуществимости, не останавливаясь перед тем, что тогда с неизбежностью появляются чистые теоремы существования (способ построения есть, но он абсолютно нереалистичен по ресурсам). Такие теоремы — неизбежная плата за полноту метода.

Оговорка насчет распознаваемости равенства весьма существенна. Конструктивные действительные числа тоже изображаются финитными объектами (программами для вычисления сходящейся последовательности сегментов). Но отношение равенства на них нераспознаваемо, и из-за этого удастся доказать конструктивную несчетность конструктивных действительных чисел. Проведем это доказательство.

Пусть дана последовательность действительных чисел α . Тогда для каждого i и n $((\alpha i) n)$ — сегмент длины не больше $\frac{1}{2^n}$. Построим число x , не принадлежащее этой последовательности. Пусть

$$\text{ord} = \lambda a (\text{entier} (\log_2 (\text{length} (a n))).$$

Для вычисления $(x n + 1)$ возьмем $((\alpha n) (\text{ord} (x n) + 3))$. Длина этого отрезка в 8 раз меньше длины $(x n)$, и можно найти в $(x n)$ подсегмент в 4 раза меньшей длины, чем $(x n)$, не пересекающийся с $((\alpha n) (\text{ord} (x n) + 3))$.

Этот результат заодно развеивает ходячий предрассудок популяристики о том, что мощность бесконечного множества — число элементов в нем. Это сложность описания данного множества.

Далее, конструктивная парадигма позволяет тоньше анализировать понятия. Есть метод, который в полной общности был осознан В. Крейновичем [2]. Некоторое положение можно прове-

речь на степень его потенциальной конструктивности, посмотрев, что из него следует в прикладной теории. Например, из разрешимости равенства пустому множеству следует закон исключенного третьего, и поэтому равенство для множеств — принципиально неконструктивное понятие, что делает весьма сомнительными «конструктивные» варианты теории множеств, популярные в среде математических логиков, полностью потерявших связь с задачами логики как науки и занявшихся увлекательным спортом: доказательством трудных теорем, формально выражаемых в «логических терминах». Методы нестандартного анализа не могут быть перенесены на конструктивизм, поскольку из принципов переноса следует закон исключенного третьего [4]. Из снятия двойного отрицания с утверждения $x > 0$ следует принцип конструктивного подбора. Таким образом, считать отношение порядка для действительных чисел классическим допустимо лишь в некоторых вариантах конструктивизма.

Рассмотрим лемму о конечном покрытии (лемму Гейне–Бореля): «Из каждого покрытия сегмента открытыми интервалами можно извлечь конечное подпокрытие». Для построенных у нас сингулярных покрытий она, очевидно, не выполняется. Эти покрытия построены при помощи принципа конструктивного подбора. Но лемма о конечном покрытии не противоречит тезису Чёрча и из нее закон исключенного третьего не следует. Поэтому в принципе возможно построение конструктивизма при помощи леммы Гейне–Бореля, что и было успешно проделано, в частности, Э. Бишопом и П. Мартин-Лёфом.

* * *

Теперь перейдем к недостаткам.

Ярче всего конструктивная математика показала вред избыточных предположений о знаниях и даже избыточной информации. Сингулярные покрытия и связанные с ними контрпримеры используют знание кода программы, а не просто применение программы как модуля. Таким образом, избыточное знание повредило.

Рассмотрим основное предположение алгоритмического конструктивизма с точки зрения прикладной информатики. На са-

мом деле он в некотором смысле математика свободного программного обеспечения. Каждый алгоритм должен быть снабжен открытым кодом. Перерабатывая алгоритм, необходимо предоставить открытый код для полученного результата. Преобразования программ могут использовать всю информацию о программах. А раз так, то в принципе возможны грязные хакерские трюки¹.

Тем самым недостатки конструктивизма обосновывают, что основная аксиома СПО сформулирована принципиально неверно. Необходимо не открытый код, а открытая, свободная, полная и корректная программная документация. В этом отношении автор полностью согласен с выводами профессора А. А. Шалыто из СПбГУИТМО. Тем более что эта документация отвечает духу патентного законодательства (требуется описание патентуемой системы, такое, чтобы специалист мог ее воссоздать). СПО появилось в результате принципиальной ошибки: перенести на программное обеспечение права собственности, аналогичные правам на материальные предметы. А нужно было, оказывается, перенести в область СПО нормы патентного права, переработав их применительно к информационным объектам. Тогда для обеспечения юридической защиты своего продукта та же фирма Microsoft вынуждена была бы представить его полную и корректную спецификацию. А сейчас отсутствует зачастую даже спецификация того, что должно быть открыто: интерфейсов и структур данных. Предоставляемая же документация пестрит ошибками, устаревшими и прямо ложными утверждениями.

Теперь рассмотрим одно из основных достижений алгоритмического конструктивизма: алгоритм конструктивной расшивки и четкое разделение формул на конструктивную задачу и ее классическое обоснование. Использование конструктивной расшивки в работах советских конструктивистов привело

¹ Анекдотическим подтверждением этого является появление в СПО понятия «версии программы, совместимой с предыдущей вплоть до багов». Если никто не потребовал исправить ошибку настолько категорично, что у СПО поднялись руки сделать это, то нужно позаботиться прежде всего о своих: о хакерах, которые могли эксплуатировать эту ошибку.

к утяжелению конструкций. Все построения необходимо было полностью и заранее провести, а потом обосновать.

С практической точки зрения это означает следующее. Стандартная форма не всегда самая лучшая для конкретных задач. Явное построение часто хуже неявного, автоматически извлекаемого из краткого и понятного конструктивного описания. И опять-таки мы приходим к принципиальной методологической ошибке СПО.

А с точки зрения математической техники можно извлечь следующий урок. Делить утверждение на конструктивную и дескриптивную части очень хорошо. Но стремиться полностью расшифровать конструктивную часть и локализовать ее в виде точного законченного предложения в начале вывода — плохо. Нужно соблюдать баланс.

Привязка к конкретным представлениям данных, являющаяся одним из распространенных грехов в математике, была в связи с «требованием» явных построений доведена до крайности. Все выражалось как слова в конечном алфавите. Тем самым заодно была предопределена привязка к конкретной предметной области, причем уже морально устаревшей. И тем самым все попытки прямого применения алгоритмического советского конструктивизма наталкивались на непреодолимые технические трудности.

Подытоживая, можно сказать следующее. Опора лишь на чистое знание привела к неестественным конструкциям, уступающим с эстетической точки зрения интуиционизму. А эстетика является одним из важнейших неформальных критериев, позволяющих оценить применимость (сейчас или в будущем) абстрактной теории.

3 Интуиционизм как альтернатива алгоритмическому конструктивизму

Брауэр [16, 17] после появления теории алгоритмов, реализуемости по Клини и советского конструктивизма почувствовал привлекательность этой альтернативы для тех, кто хотел бы большей конструктивности, но не был готов к глубокой перестройке своего мышления. Поэтому он явно выделил неалгоритмические

аспекты эффективности. При этом было сделано принципиальное методологическое открытие: Брауэр показал, что можно позитивно использовать принцип *ignotamus et ignorabimus*, и что знание о незнании — одна из важнейших и сильнейших форм знания.

Рассмотрим концепцию беззаконных последовательностей Брауэра.

Беззаконная последовательность α — метод, применяя который, мы можем получать все новые и новые данные, но все наше знание о методе исчерпывается тем, которое можно получить из уже имеющейся конечной совокупности его результатов. Тип беззаконных последовательностей является подтипом типа всех последовательностей. Формально это можно выразить следующим принципом:

$$(1) \quad \forall \alpha : \text{Lawless} (A(\alpha) \Rightarrow \\ \exists N : \text{Nat} \forall \beta : \text{Sequence} \\ (\forall n : \text{Nat} (n < N \Rightarrow (\alpha n) = (\beta n)) \Rightarrow A(\beta))).$$

Заметим, что уже из формулировки этого принципа следует, что никакие новые свойства, специфические именно для беззаконных последовательностей, не вводятся. Беззаконная последовательность — во всех остальных отношениях обычная эффективная последовательность.

Прежде всего, заметим, что $\forall \alpha : \text{Lawless} \neg \forall n : \text{Nat} (\alpha n) = 0$. В самом деле, возьмем произвольную беззаконную последовательность. Пусть для нее выполнено $\forall n : \text{Nat} (\alpha n) = 0$. Тогда это же выполнено для любой последовательности, совпадающей с ней на некотором начальном отрезке, и по начальному отрезку любой последовательности можно было бы установить, равна ли она тождественно нулю.

Далее, тем самым у нас для беззаконных последовательностей опровергается принцип конструктивного подбора. Ввиду того, что с равенства натуральных чисел двойное отрицание снимается, как и с любого разрешимого свойства, $\neg \forall n : \text{Nat} (\alpha n) = 0 \Leftrightarrow \neg \neg \exists n : \text{Nat} \neg (\alpha n) = 0$. Но, если бы мы могли найти такое n для любой беззаконной последовательности, то мы могли бы

найти его по заранее известному числу членов. Таким образом,

$$\neg\forall\alpha : \text{Lawless}(\neg\neg\exists n : \text{Nat} \neg(\alpha n) = 0 \Leftrightarrow \exists n : \text{Nat} \neg(\alpha n) = 0).$$

А поскольку беззаконные последовательности являются подтипом произвольных последовательностей, то принцип конструктивного подбора опровергается также на типе всех последовательностей.

Можно рассматривать типы беззаконных двоичных последовательностей и так далее. Они моделируются при помощи общих беззаконных последовательностей. Например, применив следующее преобразование:

$$\beta = \lambda n : \text{Nat} \text{ if } (\alpha n) = 0 \text{ then } 0 \text{ else } 1 \text{ fi}$$

получаем двоичные беззаконные последовательности. Возражения, что вероятность появления 0 и 1 разная, отменяются аргументом, что понятие вероятности не применимо к беззаконным последовательностям, поскольку оно означало бы глобальное знание о ней, невыводимое из конечного числа элементов.

Далее, тип данных «Lawless» нельзя заменить предикатом. Тогда, в частности, по конечному числу членов беззаконной последовательности можно было бы установить, беззаконна ли она. А это приводит к очевидному логическому противоречию.

Тем самым Брауэр на 30 лет раньше, чем это было осознано в информатике, показал, что типы данных не являются множествами.

Беззаконные последовательности имеют практическую интерпретацию. Они являются идеализацией физических датчиков.

Еще один вид эффективных, но не алгоритмических, последовательностей Брауэр предложил, рассматривая проблему знания. В конструктивизме мы знаем только то, что уже успели реализовать и обосновать. Поэтому Брауэр выдвинул следующую модель знания.

Имеется последовательность, члены которой вычисляются один раз в день. Если за день n проблема A полностью решена, либо она была решена еще раньше, то $(\alpha_A n) = 1$. До тех пор,

пока реализации и ее обоснования нет, $(\alpha_A n) = 1$. Тогда выполнен следующий принцип, формальная запись которого принадлежит С. Крипке и Г. Крайзелю:

$$\exists n : \text{Nat} (\alpha_A n) = 1 \iff A.$$

Такие последовательности называются *творческими*.

Творческие последовательности можно представить как результат взаимодействия компьютера с человеком (в данном случае с ученым). Ученый пытается решить некоторую проблему. А компьютер выжидает, сможет ли он в отведенное для вычислений время получить от ученого подтверждение, что проблема решена, и в дальнейшем использует этот факт. Так что тем самым Брауэр показал, что концепция человеко-машинного взаимодействия в рамки обычной теории алгоритмов не укладывается.

Конечно же, наличие беззаконных либо творческих последовательностей противоречит тезису Чёрча. Но парадоксально оказалось то, что можно построить модель интуиционистской математики, в которой такие понятия не приводят к противоречию. Тем самым Брауэр вновь продемонстрировал мощь своей интуиции.

Модель интуиционистской математики была построена при помощи топологических конструкций (гораздо более тяжелых, чем использованные нами в первой части для пояснения концепций эффективной вычислимости). Для практиков этот результат прежде всего идеальный: можно пользоваться интуиционистскими абстракциями ничуть не хуже, чем советским конструктивизмом. Сама по себе модель представляет собой идеальную конструкцию и интерпретирована в вычислениях быть не может. Но некоторые следствия из такой модели имеют прямое отношение и к теории, и к практике, и к педагогике информатики.

Стоит отметить блестящий результат Трулстра (1974), имеющий не только математическое, но и педагогическое и прикладное значение. Адекватной моделью интуиционистских вычислений являются преобразователи с тремя входами: конструктивные данные, результаты измерений (беззаконная последователь-

ность), решения человека (творческая последовательность). А сами преобразователи могут считаться алгоритмическими (хотя их алгоритма мы не знаем).

Для того чтобы обоснованно оценить результаты по анализу конструктивных понятий на базе интуиционизма, необходимо подробнее рассмотреть соответствующие понятия из теории моделей. Поэтому мы еще раз вернемся к ним позже.

4 Промежуточные варианты конструктивизма

Во второй половине 60-х и в 70-х годах XX века наступил ренессанс конструктивизма в мире одновременно с началом упадка его в СССР. Алгоритмический конструктивизм и применение бурно развивавшейся нечисленной математики дали возможность строить глубокие точные модели интуиционистских конструкций. Возродился интуиционизм, появились альтернативные конструктивные концепции. Общая черта альтернативных концепций: мы пользуемся лишь алгоритмами, но не хотим явно знать и явно утверждать это.

Пожалуй, принципиальным прорывом здесь были наблюдения Э. Бишопа и М. Бизона [11, 12], что алгоритмический подход совместим с принципом бар-индукции, который считался одним из важнейших отличий интуиционизма от алгоритмического конструктивизма. Прекрасный обзор истории этого принципа дан в [14]. Рассмотрим подробнее принцип, переведем его в абстрактную форму, а также его обоснование в интуиционизме и его опровержение в алгоритмическом конструктивизме. Пусть $(\text{Cortege } \alpha \ n)$ — алгоритмический функционал, выдающий кортеж из первых n членов последовательности натуральных чисел α . Пусть $(\text{Add } l \ x)$ — алгоритм, преобразующий список объектов и объект в список, образуемый добавлением в конец еще одного члена.

Бар-индукция Брауэра

$$\begin{aligned}
 (2) \quad & \forall x : \text{List}(\text{Object}) (A(x) \vee \neg A(x)) \& \\
 & \forall \alpha : (\text{Nat} \rightarrow \text{Object}) \exists n : \text{Nat} A[x | (\text{Cortege } \alpha \ n)] \& \\
 & \forall a : \text{List}(\text{Object}) (\forall y : \text{Object} A[x | (\text{Add } a \ y)] \Rightarrow A[x | a]) \\
 & \Rightarrow A[x | \text{NIL}]
 \end{aligned}$$

Базисом индукции служит возможность найти на каждом эффективно вычисляемом пути в бесконечном дереве список, обладающий требуемым свойством. Шагом — возможность перехода от всевозможных продолжений списка на один элемент к самому списку. Заключение — возможность спуститься вплоть до корня дерева списков. С классической точки зрения это одна из вариаций принципа трансфинитной индукции. Если вполне упорядочить множество объектов, то можно вполне упорядочить списки так, что это упорядочение будет согласовано с отношением «список a является началом списка b ». А если множество объектов достаточно просто (счетно), то и аксиомы выбора не требуется.

В чисто алгоритмическом конструктивизме принцип бар-индукции опровергается на примере.

В работе [21] было показано, что принцип бар-индукции совместим с негативным вариантом тезиса Чёрча (НСТ): для каждой функции в принципе есть алгоритм, который ее вычисляет, но он не обязательно нам известен:

$$(3) \quad \begin{aligned} & \forall x (\neg A(x) \Rightarrow \neg \neg \exists y B(x, y)) \Rightarrow \\ & \exists z \forall x (\neg A(x) \Rightarrow \exists u (\Upsilon [z, x]) \rightarrow u) \ \& \\ & \forall u (\Upsilon [z, x]) \rightarrow u \Rightarrow B(x, u) \end{aligned}$$

Это наблюдение обосновывает американский конструктивизм Э. Бишоп [12], далее развитый Д. Бизоном [11], и либеральный конструктивизм П. Мартин-Лефа [22], соединявшие алгоритмичность с отказом от всезнания: все методы — алгоритмы, но мы запрещаем явно использовать это знание.

5 Модели конструктивных теорий

Две статьи Гливенко² [18, 19] дали первую полную формализацию интуиционистской логики и изоморфизм между классической логикой и подсистемой интуиционистской. Он же (1932) показал, что классическая математика моделируется внутри кон-

²В. И. Гливенко (1897-1940) — российский математик. Большинство его работ посвящено теории вероятностей, математической статистике и функциональному анализу. В логику, которая тогда для российских математиков была чем-то маргинальным, он сделал всего пару вылазок, зато каких!

структивной, тем самым неявно сделав еще одно важнейшее методологическое открытие.

УТВЕРЖДЕНИЕ 1. (Теорема Гливенко) Классическая логика изоморфно погружается в интуиционистскую.

ДОКАЗАТЕЛЬСТВО. Построим следующий погружающий алгоритм:

$$\begin{aligned} G(A) &= \neg\neg A \quad (A \text{ элементарна}) \\ G(A \& B) &= G(A) \& G(B) \\ G(A \vee B) &= \neg(\neg G(A) \& \neg G(B)) \\ G(A \Rightarrow B) &= G(A) \Rightarrow G(B) \\ G(\neg A) &= \neg G(A) \\ G(\forall x A) &= \forall x G(A) \\ G(\exists x A) &= \neg \forall x \neg G(A) \end{aligned}$$

Итак, систематически устраняются дизъюнкция и существование. Остальные связки остаются без изменения. \square

Этот результат не произвел шокирующего впечатления лишь потому, что он был совершенно неадекватно объяснен. Автор комментировал его: «Мы доказали непротиворечивость классической логики относительно интуиционистской». Но ведь непротиворечивость классической логики очевидна всем, так что кого такой результат интересует, кроме заскоружлых формальных логиков! Ныне изоморфизм Гливенко удивляет традиционно ориентированных математиков (например, сильных студентов МФТИ³). Ослабив одну из логических аксиом, мы получили на самом деле более сильную систему, включающую классическую логику как подсистему. Но, надеюсь, Вы уже привыкли к тому, что самоограничение и отказ от лишнего слишком часто на самом деле означает получение дополнительных возможностей.

³Хотя они могли бы, применив свое великолепное знание физики, получить точные аналогии: в механической системе иногда убрать лишнюю связь означает существенно расширить возможности системы (правда, захотев полной свободы, мы получим примерно тот же эффект, как разобрав часовой механизм: все традиционные «мешающие» связи устранены, свободы сколько угодно («Запрещается запрещать»), но дела никакого нет!)

Чисто синтаксический анализ интуиционистской логики быстро уступил место гораздо более сильным и эффективным косвенным методам анализа, основанным на теории моделей и на ее комбинациях с понятием реализуемости. Само по себе понятие реализуемости, хотя является главным обоснованием и главным преимуществом конструктивной функциональной логики с точки зрения информатики, с точки зрения чистой логики неадекватно. О причине мы уже говорили: формально в интерпретации реализуемости реализуема каждая формула $A \vee \neg A$, а то, что мы ее реализации не знаем, для чистого математика — китайские церемонии. Этого в формулировке теоремы не учесть.

Первой систематической моделью интуиционистской логики была такая, которая производила впечатление некоторой искусственности. Понятие булевой алгебры было ослаблено до понятия псевдобулевой алгебры, где дополнение стало интерпретацией интуиционистского отрицания. Но уже следующая оказалась естественной для математиков и используется до настоящего времени.

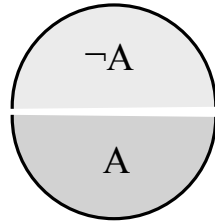
ОПРЕДЕЛЕНИЕ 1. Топологическая интерпретация конструктивной интуиционистской логики.

Значение каждой формулы — открытое множество.

1. $\|A \& B\| = \|A\| \cap \|B\|$.
2. $\|A \vee B\| = \|A\| \cup \|B\|$.
3. $\|A \Rightarrow B\| = (\text{Int } \overline{\|A\|}) \cup \|B\|$.
4. $\|\neg A\| = (\text{Int } \overline{\|A\|})$.
5. $\|\forall x A\| = (\text{Int } \bigcap_{a \in U} \|A[x|a]\|)$.
6. $\|\exists x A\| = \bigcup_{a \in U} \|A[x|a]\|$.

Заметим, что здесь изменяются значения следования, отрицания и всеобщности, а не дизъюнкции и существования, как можно было ожидать, исходя из изоморфизма Гливенко.

На этой модели легко показывается сущность (с точки зрения значений истинности) отличия $\neg\neg A$ от A . Возьмем в качестве

Рис. 1. Топологическая интерпретация $A \vee \neg A$

пространства круг. На рисунке 1 показано, что значением $\neg A$ будет верхний открытый полукруг, вместе $A \vee \neg A$ дает круг без линии. $\neg(A \vee \neg A)$ станет пустым множеством, так как у дополнения $A \vee \neg A$ нет внутренней, а $\neg\neg(A \vee \neg A)$, соответственно, займет весь круг и будет истинным.

Но принципиальным прорывом явилась идея Э. Бета, повторно реализованная затем С. Крипке на 10 лет позже и носящая его имя⁴. Он взял за основу концепцию развивающегося неполного знания Брауэра и рассмотрел логическую модель как множество миров, связанных отношением частичного порядка (конкретнее, достаточно рассматривать деревья миров). В позднейших мирах могут появиться новые факты, но старые исчезнуть уже не могут. На этой концепции возможных миров Бет построил систему моделей интуиционистской логики, дал метод построения таких моделей (семантические таблицы), доказал теорему полноты, а позже обобщил свой метод на классическую логику (именно так!).

ОПРЕДЕЛЕНИЕ 2. Модели возможных миров (вариант Крайзеля–Коэна–Крипке).

Задается сигнатура σ и универс U . Модель является чумом миров V с отношением порядка \preceq . В каждом мире определяется множество «освоенных» в нем объектов $(\text{Uni } v) \subseteq U$ и множество замкнутых элементарных формул $(T v)$: фак-

⁴Названия по именам в науке — один из примеров постоянных ошибок, нелепостей и несправедливостей. Сам Лопиталь протестовал против того, чтобы правило, открытое Бернулли, называли его именем, но название так и осталось...

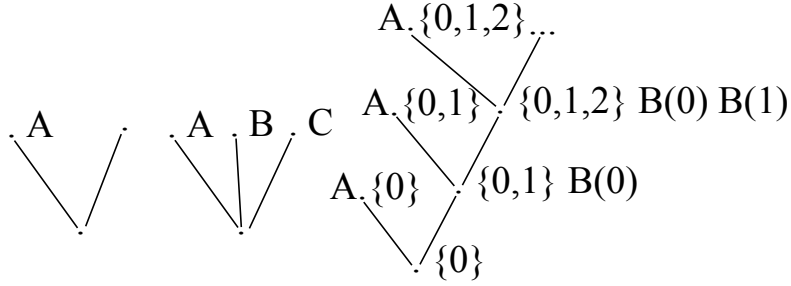


Рис. 2. Опровергающие модели Крипке

тов, установленных в этом мире. Все факты содержат лишь объекты из $(\text{Uni } v)$. Оба эти отображения неубывающие по \subseteq . Одновременной индукцией определяются два отношения между формулой и миром: формула подтверждается в мире $v \models A$ и формула не подтверждена в мире $v \not\models A$, которое по умолчанию определяется $v \not\models A \equiv v \models A$, и поэтому зададим его явно лишь для элементарных формул.

1. $u \models A \equiv A \in (T v)$; $u \not\models A \equiv A \notin (T v)$ для элементарных формул.
2. $(u \models A \& B) \equiv u \models A \wedge u \models B$.
3. $(u \models A \vee B) \equiv u \models A \text{ or } u \models B$.
4. $(u \models A \Rightarrow B) \equiv \forall v : \text{World } (v \models A \wedge v \not\approx u \supset v \models B)$.
5. $(u \models \forall x A) \equiv \forall v : \text{World } \forall a : \text{Obj } (v \not\approx u \wedge a \in (\text{Uni } v) \supset v \models A[x|a])$.
6. $(u \models \exists x A) \equiv \exists a : \text{Obj } (a \in (\text{Uni } u) \wedge u \models A[x|a])$.

ПРИМЕР 3. Опровергающие модели для $A \vee \neg A$, $(\neg A \Rightarrow B \vee C) \Rightarrow (\neg A \Rightarrow B) \vee (\neg A \Rightarrow C)$, $\forall x (A \vee B(x)) \Rightarrow A \vee \forall x B(x)$ приведены на рисунке 2. Последняя из них бесконечна. А на модели 3 истинна противоречащая классической логике формула $\neg \forall x (\neg \neg A(x) \Rightarrow A(x))$.

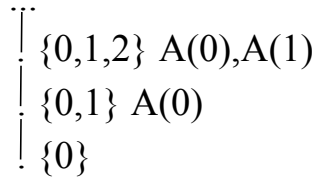


Рис. 3. Сильно опровергающая модель Крипке

На самом деле этот вид моделей интуиционистской логики является частным случаем топологических моделей. Ведь любой чум может рассматриваться как топологическое пространство, базисом окрестностей в котором являются верхние конусы $\{y \mid y \succ x\}$. Тогда область истинности любой формулы становится открытым множеством, и все пункты определения моделей Крипке превращаются в пункты, касающиеся топологических пространств.

Теперь дадим идею доказательства теоремы полноты для интуиционистской логики относительно моделей Крипке (и одновременно, как уже было замечено, относительно топологических пространств).

Как известно, одним из способов доказательства полноты классической логики является пополнение теории в некотором смысле до предела: она должна стать

- 1) слабо конструктивной, то есть $\vdash \exists x A$ должно влечь $\vdash A[x|t]$ для некоторого терма t ,
- 2) корректной, то есть $\vdash A \vee B$ должно влечь доказуемость одного из членов дизъюнкции.
- 3) полной, то есть для каждой A должно быть доказуемо A либо $\neg A$.

Полнота и слабая конструктивность влекут корректность. Такая теория сама дает свою модель (универс — термы, истинность совпадает с доказуемостью). Ослабим условия на теорию: приемлемая теория — слабо конструктивная и корректная. Приемлемая теория может не быть максимальной, если она не полна.

Далее, рассмотрим вдобавок к позитивным аксиомам аксиомы незнания $\neg A$ и потребуем следующие условия непротиворечивости:

1. Если $\neg B$ и $\vdash A \Rightarrow B$, то $\neg A$.
2. Не может быть одновременно $\vdash A$ и $\neg A$.
3. Если $\vdash \neg A$, то $\neg A$.

Условие максимальности для приемлемых теорий.

Для любой формулы A либо $\vdash A$, либо $\neg A$.

Рассмотрим множество всех максимальных приемлемых расширений данной интуиционистской теории. Упорядочим его отношением вложенности множества доказуемых формул. Тогда каждая из теорий дает мир (универс — термы, истинные утверждения — элементарные доказуемые формулы), а вся эта совокупность — модель для теории. Более того, если тривиально пополнить теорию, сделав все недоказуемые формулы отвергаемыми, и рассматривать лишь расширения, где эти отвергаемые формулы остаются отвергаемыми, то любая приемлемая теория имеет точную модель, где истинны доказуемые формулы и только они.

Рассмотрение моделей Крипке ставит еще один вопрос, ясно осознанный в советском конструктивизме, а после независимо переоткрытый в более абстрактной, но гораздо менее продвинутой с практической точки зрения форме, Д. Скоттом. Не все объекты в модели Крипке доступны с самого начала. Таким образом, не все объекты могут быть определены в рассматриваемом сейчас мире. Поэтому в модели Крипке естественно рассматривать квазипредикат $!t$, который считается подтвержденным в тех мирах, где определено t . Согласно определениям всеобщности и существования, выполнены следующие эквивалентности:

$$\forall x A(x) \Leftrightarrow \forall x (!x \Rightarrow A(x)); \quad \exists x A(x) \Leftrightarrow \exists x (!x \& A(x)).$$

Последний из рассматриваемых нами вариантов моделей интуиционистской логики — оригинальный вариант моделей Бета, который мы несколько обобщим.

Модели Бета отличаются от моделей Крипке в трех отношениях. Во-первых, универс един для всех миров. Во-вторых, дополнительно обобщаются определения выполнимости для дизъюнкции и существования. Пусть Way — тип путей в упорядоченном множестве миров.

1. $u \models A \vee B \equiv \forall \alpha : \text{Way} (u \in \alpha \supset \exists v : \text{World} (v \in \alpha \wedge (v \models A \text{ or } v \models B)))$.
2. $u \models \exists x A \equiv \forall \alpha : \text{Way} (u \in \alpha \supset \exists v : \text{World} \exists a : \text{Obj} (v \in \alpha \wedge v \models A[x|a]))$.

Заметим, что все эти виды моделей не имеют отношения к конструктивной семантике интуиционистской логики. Их можно считать исследованием ее дескриптивной силы. Однако это все не исключает их применения для анализа некоотрых конструктивных аспектов. В качестве примера рассмотрим блестящее доказательство Смориного [23] свойства корректности интуиционистской арифметики. То, что выводимость существования дает конкретный пример, безусловно, конструктивное свойство. Докажем его окольным путем, через модели.

Поскольку, согласно изоморфизму Гливенко, любая формула классической арифметики после ее перевода доказуема в интуиционистской, любой мир в модели Крипке интуиционистской арифметики является моделью классической. Но среди таких моделей есть наименьшая: стандартный натуральный ряд. Значит, к любой модели Крипке можно присоединить снизу стандартный натуральный ряд. Пусть для некоторой формулы доказуемо $\exists x A(x)$, но ни для какого натурального числа не доказуемо $A[x|n]$. Тогда для каждого n имеется модель Крипке интуиционистской арифметики M_n , в которой опровергается $A[x|n]$. Объединив все эти модели и поместив в качестве нижней точки дерева стандартный натуральный ряд, опровергаем $\exists x A(x)$, которое доказуемо по предположению.

Это доказательство неконструктивно, зато исключительно коротко и прозрачно по структуре. А имея его, можно спокойно строить алгоритм получения числа по доказательству. В случае принципа конструктивного подбора он получается автоматически: перебирать все n , пока не найдется нужное.

6 Различие вместо равенства

Пожалуй, Д. ван Дален первым заметил, что выделенное Брауэром понятие различия может в некоторых конструктивных теориях рассматриваться как более фундаментальное, чем равенство. Опишем основные свойства различия.

1. $\forall x \neg(x \# x)$.
2. $\forall x \forall y (x \# y \Rightarrow y \# x)$.
3. $\forall x \forall y (x \# y \Rightarrow \forall z (x \# z \vee y \# z))$ (глобальная отличимость).
4. $\forall x \forall y (\neg x \# y \& A(x) \Rightarrow A(y))$.

Из этих свойств следует, что $\forall x \forall y \forall z (\neg z \# x \& \neg z \# y \Rightarrow \neg x \# y)$ (контрапозиция свойства глобальной отличимости). Следовательно, можно определить равенство как $\neg x \# y$.

Определить различие через равенство можно лишь в том случае, если понятие равенства разрешимо, то есть $\forall x \forall y (x = y \vee \neg x = y)$. Таков, например, тип данных «натуральные числа».

В топологически определенных пространствах различимость естественно вводится как наличие дизъюнктивных приближений. Например, различимость действительных чисел, определяемых как быстро стягивающиеся последовательности интервалов, может быть задана следующим образом:

$$x \# y \iff \exists n : \text{Nat} (((x.\text{low } n) > (y.\text{up } n)) \vee ((y.\text{low } n) > (x.\text{up } n))).$$

Это условие в случае его выполнимости эффективно проверяемо, в отличие от равенства. Тем самым различие в неразрешимых пространствах не сводимо к равенству.

В советском конструктивизме заметили, что дизъюнкция выражается через существование. То же подметил для арифметики и ван Дален, но не обобщил на случай произвольного типа данных с двумя различимыми элементами. В самом деле, если $a \# b$,

$$A \vee B \iff \exists x : \text{Obj} (\neg(x \# A \& x \# B) \& (x = a \Rightarrow A) \& (x = b \Rightarrow B)).$$

7 Анализ логических принципов

Прием Крейнвича оказалось возможным использовать также при анализе логических систем. Многие профессиональные логики называют всевозможные исчисления, не содержащие конкретных объектов и сформулированные в терминах логического языка, «логиками». Но не все такие исчисления заслуживают столь почетного звания. Еще с 30-х годов XX века известен пример.

Йогансон, недовольный трактовкой отрицания в интуиционистской логике Брауэра, предложил убрать из нее принцип «Из лжи следует все, что угодно». Полученную логику назвали минимальной. Но она не прижилась в качестве конструктивной, поскольку в прикладных теориях, основанных на минимальной логике, брауэровское отрицание легко моделируется. Например, в арифметике достаточно определить $\neg A$ как $A \Rightarrow 0 = 1$, поскольку из $0 = 1$ выводится любая формула, не содержащая отрицания.

В работе [5] показано, что внутрилогических критериев недостаточно для отбора исчислений, пригодных для построения на их основе прикладных теорий. Рассмотрим два примера.

ПРИМЕР 4. Рассмотрим логический принцип (который открыли Крайзел и Путнам; мы его называем *принцип тотальной определенности* (КР)):

$$(4) \quad \forall x (\neg A(x) \Rightarrow \exists y B(x, y)) \Rightarrow \forall x \exists y (\neg A(x) \Rightarrow B(x, y)).$$

С формальной точки зрения пропозициональная логика после добавления этого принципа к интуиционистской логике не изменяется, поскольку, если отбросить кванторы, получается тавтология. Но с фактической точки зрения в любой теории, где есть хотя бы два различимых элемента $a \# b$

$$\forall x ((x \# a \vee x \# b),$$

из принципа тотальной определенности следует принцип Медведева

$$(\neg A \Rightarrow B \vee C) \Rightarrow (\neg A \Rightarrow B) \vee (\neg A \Rightarrow C).$$

В самом деле, посылка принципа Медведева эквивалентна утверждению

$$\forall x (\neg A \Rightarrow \exists y ((y = a \vee y = b) \& (y = a \Rightarrow B) \& (y = b \Rightarrow C))).$$

Преобразуя его, получаем

$$\forall x \exists y (\neg A \Rightarrow (y = a \vee y = b) \& (y = a \Rightarrow B) \& (y = b \Rightarrow C)).$$

Отбрасывая фиктивный квантор всеобщности и применяя пропозициональную эквивалентность, получаем:

$$\exists y ((\neg A \Rightarrow (y = a \vee y = b)) \& ((\neg A \Rightarrow (y = a \Rightarrow B)) \& (\neg A \Rightarrow (y = b \Rightarrow C)))).$$

Отсюда по разрешимости равенства a, b получается

$$\exists y ((\neg \neg A \vee y = a \vee y = b) \& ((\neg A \Rightarrow (y = a \Rightarrow B)) \& (\neg A \Rightarrow (y = b \Rightarrow C)))).$$

А теперь по дистрибутивности и по закону «из лжи следует все, что угодно»

$$\neg \neg A \vee \exists y ((y = a \vee y = b) \& (\neg A \Rightarrow (y = a \Rightarrow B)) \& (\neg A \Rightarrow (y = b \Rightarrow C))).$$

Второй член дизъюнкции следует из первого, и первый можно отбросить.

$$\exists y ((y = a \vee y = b) \& (\neg A \Rightarrow (y = a \Rightarrow B)) \& (\neg A \Rightarrow (y = b \Rightarrow C))).$$

А теперь, преобразуя обратно к дизъюнкции, получаем

$$(\neg A \Rightarrow B) \vee (\neg A \Rightarrow C).$$

Следующий пример не укладывается в рамки одного доказательства.

Рассмотрим интуиционистскую арифметику. Сама по себе формальная система интуиционистской арифметики, как показали результаты Сморолинского [23], полна относительно интуиционистской пропозициональной логики в том смысле, что в

ней можно построить невыводимую формулу, являющуюся подстановочным примером любой пропозициональной формулы, не выводимой в интуиционистской логике. Но приблизимся еще на один шаг к содержательному смыслу арифметических формул. Рассмотрим интуиционистскую арифметику с конструктивным ω -правилом (правилом Карнапа)

$$(5) \frac{A[x|0], \dots, A[x|\mathbf{n}], \dots}{\forall x : \text{Nat } A(x)}$$

В конструктивном правиле Карнапа вывод является алгоритмически разрешимым деревом с конечными путями. В классическом правиле Карнапа снимается требование разрешимости. Вначале упомянем один результат, принадлежащий фольклору.

ЛЕММА 2. Интуиционистская арифметика с классическим правилом Карнапа совпадает с классической.

ДОКАЗАТЕЛЬСТВО. Всякая доказуемая формула истинна в стандартной модели.

Наоборот, индукцией по определению истинности тривиально строится вывод любой истинной формулы. Критические шаги здесь следующие. Если $A \vee B$ классически истинно, то один из членов дизъюнкции интуиционистки выводим, по предположению индукции. Но тогда можно построить вывод и самого $A \vee B$. Таким способом можно построить вывод $A \vee \neg A$. А вывод $\forall x : \text{Nat } (A(x) \vee \neg A(x))$ строится по правилу Карнапа, поскольку каждая из формул $A(x|\mathbf{n}) \vee \neg A(x|\mathbf{n})$ доказуема, а знать конкретные доказательства не обязательно. \square

Гораздо более тонкий результат принадлежит Феферману.

ЛЕММА 3. (Феферман [8]) В классической арифметике с конструктивным ω -правилом выводима любая истинная формула.

Данный результат легко обобщается на класс интуиционистских формул, являющихся переводами по Колмогорову–Гливленко классических (их будем называть просто классическими формулами), поскольку для таких формул классический вы-

вод алгоритмически перестраивается в интуиционистский. Таким образом, все арифметические системы с конструктивным ω -правилом и интуиционистской логикой полны на множестве классических формул и совпадают на нем.

Теперь отметим простейший факт из теории моделей Крипке.

ЛЕММА 4. Если в модели все пути конечны, то все классические формулы истинны тогда и только тогда, когда они истинны в листьях данной модели.

В самом деле, классическая формула A истинна тогда и только тогда, когда $\neg\neg A$, а последняя формула истинна для моделей с конечными путями тогда и только тогда, когда она истинна в последней точке каждого пути, т. е. в листьях.

ЛЕММА 5. В модели Крипке суперинтуиционистской арифметики с ω -правилом все миры элементарно эквивалентны стандартной арифметике.

ДОКАЗАТЕЛЬСТВО. Согласно теореме Сморолинского [23], каждый мир в модели интуиционистской арифметики является моделью классической арифметики. Но все истинные классические формулы выводимы в рассматриваемой теории. Значит, все они истинны в любой модели. Далее, согласно результатам той же работы, к любой модели интуиционистской арифметики можно добавить без нарушения истинности формул стандартный натуральный ряд в качестве корня дерева модели. Классические формулы истинны в этой нижней точке и во всех конечных, значит, они истинны и во всех промежуточных. \square

Теперь рассмотрим логики, обладающие следующим свойством. Класс моделей, в которых все пути конечны, является полным. Все суперинтуиционистские логики, обладающие свойством Крейга, за исключением логики линейных цепей, обладают данным свойством.

Расширенной формулой назовем формулу, в которую могут быть подставлены в качестве констант элементы универса данного мира. В конечных вершинах истинны все классически истинные формулы. Докажем бар-индукцией и индукцией по по-

строению формул, что тогда они будут истинны и во всех остальных точках модели.

Пусть для всех миров, находящихся непосредственно выше данного, доказано, что в них истинность совпадает с классической. Пусть это доказано и для подформул данной формулы. Рассмотрим различные случаи построения формулы. Нетривиальны из них дизъюнкция, импликация, всеобщность и существование.

Дизъюнкция. Поскольку она классически истинна, классически истинен один из ее членов. Значит, по предположению индукции, он будет истинен и в данной точке.

Импликация. Если она классически истинна, то либо ее посылка ложна, либо ее заключение истинно, и сводим случай к предыдущему.

Существование. Если для каждой из последующих моделей существует n , то оно может быть выбрано стандартным, а тогда $A[x|n]$ истинно в данной точке.

Всеобщность. Универс данной точки является подуниверсом следующих за ней, и применяем предположение индукции для всех $A[x|n]$ со стандартными и нестандартными n .

Таким образом, мы доказали следующий результат:

ТЕОРЕМА 1. *Если логика обладает свойством обрыва цепей (каждый путь в дереве миров конечен), то арифметика, основанная на ней, становится классической после добавления конструктивного правила Карнапа.*

Остается рассмотреть случай логики линейных цепей. Она задается схемой аксиом $(A \Rightarrow B) \vee (B \Rightarrow A)$. Но частным случаем данной схемы является $(\neg A \Rightarrow A) \vee (A \Rightarrow \neg A)$. Но это эквивалентно $\neg\neg A \vee \neg A$. А логика слабого закона исключенного третьего уже обладает свойством конечных моделей: ее достаточно рассматривать на моделях с двумя последовательными мирами. Таким образом, все пропозициональные суперинтуиционистские логики, обладающие свойством Крейга, превращают арифметику в классическую.

Теперь рассмотрим два принципа, несовместимость которых при условии тезиса Чёрча показана ранее: принцип конструк-

тивного подбора и принцип **КР**. В чистой логике они вообще не изменяют множества пропозициональных теорем ни по отдельности, ни вместе: вычеркнув кванторы и переменные, мы получаем интуиционистски доказуемые формулы. Как уже было показано, в любой теории, где есть два различных элемента, **КР** влечет принцип Медведева. Но в арифметике с ω -правилом ситуация еще интереснее.

ТЕОРЕМА 2 (BIRTHDAY THEOREM).⁵ *В интуиционистской арифметике с ω -правилом, **МР** и **КР** доказуем закон исключенного третьего.*

ДОКАЗАТЕЛЬСТВО. Будем называть класс формул Δ *замкнутым относительно подстановки*, если наряду с формулой $A[x|t]$ он содержит и $A[x|u]$ для любого замкнутого термина u . Будем говорить, что на классе Δ выполнена пропозициональная формула \mathfrak{A} , если есть алгоритм, строящий доказательство любого примера \mathfrak{A} , получаемого замещением пропозициональных букв на формулы класса Δ . Докажем несколько лемм.

ЛЕММА 6. *Если на классе арифметических формул Δ в арифметике с конструктивным ω -правилом выполнена $\mathfrak{A} \vee \neg\mathfrak{A}$, то для любой формулы $A(x)$, подстановки в которую лежат в классе Δ , доказуемо $\forall x (A(x) \vee \neg A(x))$.*

Очевидно. Заметим, что обобщение этой леммы на другие теории не всегда выполнено.

Рассмотрим теперь иерархию Клини–Мостовского формул классической арифметики. Формулы нулевого класса K_0 — примитивно-рекурсивно разрешимые. Формулы класса Σ_{n+1} — формулы вида $\exists x A(x)$, где формула $A[x|k]$ принадлежит классу n . Формулы класса Π_{n+1} — формулы вида $\forall x A(x)$, где формула $A[x|k]$ принадлежит классу n . $\Delta_n = \Sigma_n \cap \Pi_n$. $K_n = \Sigma_n \cup \Pi_n$. Любая арифметическая формула эквивалентна формуле одного (и всех более высоких) классов. Δ_1 — алгоритмически разрешимые формулы. Σ_1 — алгоритмически перечислимые. Известно, что имеются формулы класса Π_n , не принадлежащие Σ_n . Далее,

⁵Теорема доказана в день рождения, по дороге на логическую конференцию.

известно, что теорема об универсальном интерпретаторе может быть обобщена следующим образом: есть формула $U_n(x, y, z)$ класса Δ_n , такая, что для любой формулы $A(x)$ класса Σ_n можно алгоритмически построить такое a , что при каждом n

$$A[x|m] \iff \exists z U_n[x|a, y|m].$$

ЛЕММА 7. *Если формула A принадлежит классу $\Sigma_n \setminus \Pi_n$, то функция*

$$\lambda m. \mu z U_n[x|a, y|m]$$

непродолжима до тотальной функции, определенной формулой класса Δ_n .

ДОКАЗАТЕЛЬСТВО. В противном случае мы могли бы проверить $A(m)$, просто вычислив значение этой функции и применив затем к нему U .

Теперь по индукции докажем, что в интуиционистской арифметике с ω -правилом, **МР** и **КР** все формулы, получающиеся переводом Гливенко формул классов K_n разрешимы, то есть для них докажем закон исключенного третьего.

Пусть разрешимы все формулы класса Δ_n . Докажем, что тогда разрешимы все формулы из K_n .

Возьмем произвольную формулу $A(x)$ класса Σ_n . Тогда для нее доказуемо

$$\forall x (A(x) \iff \exists m U_n(a, x, m)).$$

Но, поскольку все формулы класса Δ_n разрешимы, доказуемо

$$\forall x \forall z (U_n(a, x, z) \vee \neg U_n(a, x, z)).$$

Тогда по принципу конструктивного подбора

$$\forall x (\neg \neg \exists z (U_n(a, x, z)) \Rightarrow \exists z (U_n(a, x, z))).$$

Но тогда по принципу **КР**

$$\forall x \exists z (\neg \neg \exists z (U_n(a, x, z)) \Rightarrow (U_n(a, x, z))).$$

Теперь при каждом m можно доказать

$$(\neg\neg\exists z (U_n(a, m, z) \Rightarrow (U_n(a, m, c_m))),$$

где c_m — новая вспомогательная константа. Воспользовавшись разрешимостью U_n , можно разобрать случаи истинности и ложности. Если $U_n(a, m, c_m)$ оказывается истинно, то $A(m)$. Если нет, то $\neg A(m)$. Поскольку доказательства $A(m) \vee \neg A(m)$ строятся алгоритмически,

$$\forall x (A(x) \vee \neg A(x)).$$

Теперь осталось заметить, что формулы из Π_n эквивалентны отрицаниям формул из Σ_n .

Лемма доказана. \square

А теперь покажем, что из разрешимости K_n следует разрешимость Δ_{n+1} . Здесь достаточно одного принципа конструктивного подбора вместе с ω -правилом. Представим и саму формулу, и ее отрицание как Σ_{n+1} -формулы $\exists y A1(x, y)$, $\exists y A2(x, y)$. Тогда $\forall x \neg\neg\exists y (A1(x, y) \vee A2(x, y))$. По принципу конструктивного подбора, $\forall x \exists y (A1(x, y) \vee A2(x, y))$. И разбором случаев $A1(m, c_m) \vee A2(m, c_m)$ доказываем искомый закон исключенного третьего.

Базис. Разрешимость Δ_1 является известным фактом.

Теорема доказана. \square

8 Неудачная попытка приложения

Приведем важнейший результат.

ТЕОРЕМА 3 (TROELSTRA). *Имеется модель интуиционистской математики, в которой каждая последовательность представляется как результат алгоритмического оператора, примененного к беззаконным и творческим последовательностям.*

Эта теорема дает общую форму вычислимости, которая нужна для практиков: алгоритм, перерабатывающий физические измерения и человеческие решения.

Мы не приводим этого доказательства, поскольку пока что не можем изложить его менее чем на 30 страницах, оно требует целого раздела лекционного курса. Более того, анализ доказательства результата Трулстра показывает: чтобы его аккуратно рассказать, необходимо полностью перестроить ВЕСЬ курс математики на нечисленной основе: алгебра, топология, теория категорий. Далее, необходим глубокий курс ЛОГИКИ как математизированной, но самостоятельной науки. Поразительно, что требования к перестройке преподавания математики, которые предъявляет нынешняя высшая практика (в частности, информатика), те же самые!

Все это показало еще одну грань математических объектов. Фундаментальные математические концепции, такие, как действительные числа, на самом деле не являются фиксированными. Они имеют важный неявный параметр: степень конструктивности. Мы можем либо уменьшить ее до нуля и получить более эффективное орудие для доказательства дескриптивных теорем (нестандартный анализ; результат Драгалина о сокращении выводов по сравнению с классическим анализом и результат автора о выводимости *tertium non datur* в интуиционистском нестандартном анализе), либо повысить конструктивность, причем разными способами. В еще большей мере такая вариабельность свойственна понятиям высших уровней: даже зафиксировав понятие алгоритма, можно варьировать понятия алгоритмического функционала.

Достигнутый прорыв позволил попытаться применить конструктивизм в информатике. Но, как слишком часто бывало в истории науки, попытка лобового применения глубокой и трудной концепции привела к неудаче (наиболее знаменитым аспектом которой был провал японского проекта «ЭВМ пятого поколения»), которая надолго дискредитировала все направление. Автор сошлется здесь лишь на свою работу [3], в которой он также поддался телячьему оптимизму, еще не понимая, что технические трудности слишком часто являются следствием принципиальных недоработок, и на рекламируемый в последние годы как великое достижение язык Agda [13], в котором на основе грубой силы (гигабайты оперативной памяти и гигафлопсы ско-

рости) реализован еще более безнадежный с практической точки зрения подход Мартин-Лёфа конструктивной теории типов. В Агде даже попытка сложить 1000 и 1000 вызывает переполнение гигабайт памяти. А равнодушные и ленивые, зато «объективные» «peer reviewers» уже требуют во многих случаях, не желая проверять доказательства сами, записать их на Агде.

В целом как раз в тот момент, когда наметился принципиальный прорыв, силы наступающих выдохлись, и конструктивизм начал длительное отступление. Чтобы возобновить развитие, необходимо поколение исследователей, глубоко знающих нечисленную математику и информатику и не боящихся трудностей.

Вся история конструктивизма XX века показывает, каких глубоких результатов может достичь небольшая группа высококвалифицированных исследователей (все время число конструктивистов было порядка 0,001 от числа всех математиков), воодушевленная высокой идеей и готовая преодолевать тяжелые препятствия, но прежде всего не грубой силой, а искусством. Стоит отметить еще один урок конструктивизма: не стоит перестраивать существующую математику под другую разновидность рациональности. Эта стратегическая ошибка была неизбежна, типичное направление главного удара не помешало достичь выдающихся результатов и, самое главное, осознать особенности и специфику конструктивизма.

И, наконец, нынешняя выродившаяся и гибнущая цивилизация совершенно разучилась слышать неприятные вещи. Язык Agda [13], полностью проигнорировавший известные уже 30 лет теоретические предупреждения, служит здесь ярким примером. Впрочем, для получения гранта нужно как можно больше эффективно и «правдоподобно» наобещать, а техника, как выбить и как отчитаться, стала подменять технику научного поиска и полностью убила научную честь и этику.

Литература

- [1] Заславский И. Д., Цейтин Г. С. О сингулярных покрытиях и связанных с ними свойствах конструктивных функций // Труды МИАН СССР. 1962. Т. 67. С. 458–502.

- [2] Крейнвич В. Я. Из чего вытекает закон исключенного третьего? // Записки научных семинаров ЛОМИ. 1974. Т. 40. С. 30–37.
- [3] Непейвода Н. Н. О построении правильных программ // Языки моделирования и программное обеспечение гибридных вычислительных систем. М.: НС по комплексной проблеме «Кибернетика» АН СССР, 1978. С. 88–122.
- [4] Непейвода Н. Н. Замечания о конструктивном нестандартном анализе // Теория множеств и топология. Вып. 2. Ижевск, 1980. С. 45–50.
- [5] Непейвода Н. Н. О прикладных теориях с суперинтуиционистскими логиками // Логические исследования. Вып. 7. М.: Наука, 2000. С. 61–71.
- [6] Непейвода Н. Н. Конструктивная математика: обзор достижений, недостатков и уроков. Часть I // Логические исследования. Вып. 17. М.–СПб.: ЦГИ, 2011. С. 191–239.
- [7] Непейвода Н. Н. Конструктивная математика: обзор достижений, недостатков и уроков. Часть II // Логические исследования. Вып. 18. М.–СПб.: ЦГИ, 2012. С. 157–181.
- [8] Feferman S. Transfinite recursive progressions of axiomatic theories // J. Symbolic Logic. 1962. Vol. 27. P. 259–316 (русский перевод: Феферман С. Трансфинитные рекурсивные последовательности теорий // Математика. 1971. № 3. С. 102–223).
- [9] Шанин Н. А. О конструктивном понимании математических суждений // Тр. МИАН СССР. 1958. Вып. 52. С. 226–311.
- [10] Шурьгин В. А. Основы конструктивного математического анализа. М.: URSS, 2009. 326 с.
- [11] Beeson M. Foundations of Constructive Mathematics. Heidelberg: Springer-Verlag, 1985.
- [12] Bishop E. Foundations of Constructive Analysis. New York: McGraw-Hill, 1967.
- [13] Bove A., Dybjer P., Norell U. A Brief Overview of Agda — A Functional Language with Dependent Types // Proceedings of TPHOLs 2009. Springer, 2009.
- [14] Bridges D. A Reverse Look at Brouwer’s Fan Theorem // One Hundred Years of Intuitionism (1907–2007) / M. van Atten, P. Boldini, M. Bourdeau, G. Heinzmann (eds.). Basel: Birkhauser Verlag, 2008.
- [15] Brouwer L.E.J. Besitzt jede reele Zahl eine dezimalbruchentwicklung? // Proc. Acad. Amsterdam. Vol. 23. P. 949–954.

- [16] *Brouwer L.E.J.* Richtlijnen der intuitionistische wiskunde // Proc. Acad. Amsterdam. Vol. 50. P. 339.
- [17] *Brouwer L.E.J.* Consciousness, philosophy and mathematics // Proc. X Intern. Congr. Philosophy. Amsterdam, 1948. P. 1235–1249.
- [18] *Glivenko V.* Sur la logique de M.Brouwer // Academie Royale de Belgique.
- [19] *Glivenko V.* Sur quelques points de la logique de M.Brouwer // Academie Royale de Belgique. Bulletins de la classe des sciences. Ser. 5. 1929. Vol. 15. P.183–188.
- [20] *Kreisel G., Lacombe D.* Ensembles récursivement mesurables et ensembles récursivement ouverts // Compt. rend Acad. si. Paris. 1957. Vol. 245. № 14. P. 1106–1109.
- [21] *Kreisel G., Troelstra A.S.* Formal systems for some branches of intuitionistic analysis // Ann. Math. Log. 1970. Vol. 1. P. 229–387.
- [22] *Martin-Löf P.* Notes on constructive mathematics. Almqvist & Wiskell, Stockholm, 1970.
- [23] *Smorinsky C.A.* Applications of Kripke Models // Metamathematical Investigations of Intuitionistic Arithmetic and Analysis. Ed. A. S. Troelstra. Springer. 1973. P. 324–391.
- [24] *Troelstra A.S.* History of Constructivism in the Twentieth Century. University of Amsterdam, ITLI Prepublication Series ML-91-05, 1991.
- [25] *Troelstra A.S.* From Constructivism to Computer Science // Theoretical Computer Science. 1999. Vol. 211. P. 233–252.

References (transliteration)

- [1] *Zaslavskij I. D., Cejtin G. S.* O singularnyh pokrytijah i svjazannyh s nimi svojstvah konstruktivnyh funkcij // Tr. MIAN SSSR. 1962. T. 67. S. 458–502.
- [2] *Krejnovich V. Ja.* Iz chego vytekaet zakon iskljuchennogo tret'ego? // Zap. nauchn. sem. LOMI. 1974. T. 40. S. 30–37.
- [3] *Nepejvoda N.N.* O postroenii pravil'nyh programm // Voprosy kibernetiki. 1978. № 46. C. 88–122.
- [4] *Nepejvoda N.N.* Zamechanija o konstruktivnom nestandarstnom analize // Teorija mnozhestv i topologija. Vyp. 2. Izhevsk, 1980. S. 45–50.
- [5] *Nepejvoda N.N.* O prikladnyh teorijah s superintuicionistskimi logikami // Logicheskie issledovanija. Vyp. 7. M.: Nauka, 2000. S. 61–71.

- [6] *Nepejvoda N.N.* Konstruktivnaja matematika: obzor dostizhenij, nedostatkov i urokov. Chast' I // Logicheskie issledovanija. Vyp. 17. M.–SPb.: CGI, 2011. S. 191–239.
- [7] *Nepejvoda N.N.* Konstruktivnaja matematika: obzor dostizhenij, nedostatkov i urokov. Chast' II // Logicheskie issledovanija. Vyp. 18. M.–SPb.: CGI, 2012. S. 157–181.
- [8] *Feferman S.* Transfinite recursive progressions of axiomatic theories // J. Symbolic Logic. 1962. Vol. 27. P. 259–316.
- [9] *Shanin N.A.* O konstruktivnom ponimanii matematicheskikh suzhdenij // Tr. MIAN SSSR. 1958. Vyp. 52. S. 226–311.
- [10] *Shurygin V.A.* Osnovy konstruktivnogo matematicheskogo analiza. M.: URSS, 2009. 326 s.
- [11] *Beeson M.* Foundations of Constructive Mathematics. Heidelberg: Springer-Verlag, 1985.
- [12] *Bishop E.* Foundations of Constructive Analysis. New York: McGraw-Hill, 1967.
- [13] *Bove A., Dybjer P., Norell U.* A Brief Overview of Agda — A Functional Language with Dependent Types // Proceedings of TPHOLs 2009. Springer, 2009.
- [14] *Bridges D.* A Reverse Look at Brouwer's Fan Theorem // One Hundred Years of Intuitionism (1907–2007) / M. van Atten, P. Boldini, M. Bourdeau, G. Heinzmann (eds.). Basel: Birkhauser Verlag, 2008.
- [15] *Brouwer L.E.J.* Besitzt jede reele Zahl eine dezimalbruchentwicklung? // Proc. Acad. Amsterdam. Vol. 23. P. 949–954.
- [16] *Brouwer L. E. J.* Richtlijnen der intuitionistische wiskunde // Proc. Acad. Amsterdam. Vol. 50. P. 339.
- [17] *Brouwer L. E. J.* Consciousness, philosophy and mathematics // Proc. X Intern. Congr. Philosophy. Amsterdam, 1948. P. 1235–1249.
- [18] *Glivenko V.* Sur la logique de M.Brouwer // Academie Royale de Belgique.
- [19] *Glivenko V.* Sur quelques points de la logique de M.Brouwer // Academie Royale de Belgique. Bulletins de la classe des sciences. Ser. 5. 1929. Vol. 15. P.183–188.
- [20] *Kreisel G., Lacombe D.* Ensembles récursivement mesurables et ensembles récursivement ouverts // Compt. rend Acad. si. Paris. 1957. Vol. 245. № 14. P. 1106–1109.
- [21] *Kreisel, G., Troelstra A. S.* Formal systems for some branches of intuitionistic analysis // Ann. Math. Log. 1970. Vol. 1. P. 229–387.

- [22] *Martin-Löf P.* Notes on constructive mathematics. Almqvist & Wiskell, Stockholm, 1970.
- [23] *Smorinsky C.A.* Applications of Kripke Models // *Metamatematical Investigations of Intuitionistic Arithmetic and Analysis*. Ed. A. S. Troelstra. Springer. 1973. P. 324–391.
- [24] *Troelstra A. S.* History of Constructivism in the Twentieth Century. University of Amsterdam, ITLI Prepublication Series ML-91-05, 1991.
- [25] *Troelstra A. S.* From Constructivism to Computer Science // *Theoretical Computer Science*. 1999. Vol. 211. P. 233–252.