Теоремы Гёделя о неполноте
версия 2020 г.

Л.Д. Беклемишев

1 Теорема Гёделя о неполноте

1.1 Арифметика Пеано и арифметика Робинсона

Мы рассматриваем сигнатуру формальной арифметики, содержащую символы 0, S, $+$, $\cdot$, $=$.

Определение 1.1 *Арифметика Пеано* PA задаётся следующими нелогическими аксиомами:

1. аксиомы равенства для сигнатуры 0, S, $+$, $\cdot$, $=$;
2. $\neg S(a) = 0, \quad S(a) = S(b) \to a = b$,
3. $a + 0 = a, \quad a + S(b) = S(a + b)$,
4. $a \cdot 0 = 0, \quad a \cdot S(b) = a \cdot b + a$,
5. (Схема аксиом индукции)
   $A[a/0] \wedge \forall x\,(A[a/x] \to A[a/S(x)]) \to \forall x\, A[a/x]$,
   для любой формулы A.

Стандартной моделью арифметики Пеано называем модель

$$(\mathbb{N}; 0, S, +, \cdot, =).$$

Следующие лемма и следствие очевидны.

Лемма 1.2 $\mathbb{N} \vDash \mathsf{PA}$.

Следствие 1.3 PA *непротиворечива.*

Определение 1.4 *Арифметика Робинсона* Q получается из PA заменой схемы аксиом индукции на $a = 0 \vee \exists x\, a = Sx$ (аксиома предшественника).

Упражнение 1.5 *Показать, что* $\mathsf{PA} \vdash \mathsf{Q}$.

Таким образом, теория Q представляет собой конечную подтеорию арифметики PA.

Замечание 1.6 В теории Q не возможны доказательства по индукции, поэтому она не позволяет вывести сколько-нибудь содержательные свойства арифметических операций (см. упражнение ниже). Другими словами, Q является очень слабой подтеорией арифметики PA. Она играет роль минимально достаточной теории, для которой справедливы теоремы Гёделя о неполноте. Выбор такой теории, в отличие от PA, в значительной степени произволен.

Упражнение 1.7 *Докажите, что в теории* Q *не выводима коммутативность сложения:* $a + b = b + a$.

Упражнение 1.8 *Докажите, что в теории* Q *не выводимы следующие формулы:* $\neg\, a = S(a)$, $a \cdot b = b \cdot a$.

1.2 Отношение порядка

Отношение порядка определимо в стандартной модели $\mathbb{N}$ как

$$a \leq b \leftrightarrow \exists x\, (x + a = b). \qquad (*)$$

Это определение позволяет ввести новый предикатный символ в язык формальной арифметики и рассматривать $(*)$ как еще одну аксиому. Именно эта формула (с данным порядком слагаемых $x + a$) дает удобное определение порядка не только в стандартной модели, но и в слабой теории Q. Поэтому на практике Q отождествляют с ее дефинициальным расширением $(*)$.

Поучительно рассмотреть более слабую теорию отношения порядка, для которой также можно получить основные результаты, которые известны для Q. В этой теории, которую мы обозначаем MA, отношение $\leq$ задается следующими тремя аксиомами:

1. $0 \leq a$;
2. $a \leq 0 \to a = 0$;

3. $S(a) \leq S(b) \leftrightarrow a \leq b$.

Лемма 1.9 *В теории* Q *выводимы аксиомы* MA.

Доказательство. Рассуждаем неформально в теории Q.

1. $0 \leq a$ эквивалентно $\exists x\ x + 0 = a$. Возьмем $x = a$, тогда $x + 0 = a + 0 = a$.

2. $a \leq 0$ эквивалентно $\exists x\ \ x + a = 0$. Если $a \neq 0$, то по аксиоме предшественника $\exists y\ \ a = Sy$. Тогда $0 = x + a = x + Sy = S(x + y)$, что противоречит первой аксиоме для функции следования.

3. $Sa \leq Sb$ равносильно $\exists x\ \ x + Sa = Sb$, что равносильно $\exists x\ \ S(x + a) = Sb$. Поскольку $Su = Sv$ равносильно $u = v$, мы получаем, что $\exists x\ \ S(x + a) = Sb$ равносильно $\exists x\ \ x + a = b$, то есть $a \leq b$. $\dashv$

Замечание 1.10 Аксиомы порядка теории MA определяют отношение $\leq$ рекурсией по паре аргументов a и b. Значит, индукция позволяет доказать единственность отношения, удовлетворяющего этим аксиомам. Из этих соображений получается следующий факт.

Лемма 1.11 *Из аксиом* MA *вместе со схемой индукции (в языке с* $\leq$*) выводимо определение порядка теории* Q.

1.3 Формулировки теорем Гёделя о неполноте

Теперь мы можем дать формулировки теорем Гёделя о неполноте.

Определение 1.12 Теория T в языке, содержащем арифметический, называется Σ_1-корректной, если любое арифметическое Σ_1-предложение, доказуемое в T, истинно в $\mathbb{N}$.

Определение 1.13 Теория T называется эффективно аксиоматизируемой, если сигнатура теории разрешима, а множество её нелогических аксиом перечислимо.

Очевидно, что множество всех теорем эффективно аксиоматизируемой теории также является перечислимым.

Замечание 1.14 В силу известной леммы Крейга, всякая эффективно аксиоматизируемая теория дедуктивно эквивалентна некоторой теории с разрешимым множеством аксиом.

Теорема 1.15 (первая теорема Гёделя о неполноте) *Если теория* T

- *в арифметическом языке,*
- *эффективно аксиоматизируема,*
- Σ_1*-корректна,*

то T неполна, то есть существует арифметическое Π_1-предложение A такое, что $T \nvdash A$ и $T \nvdash \neg A$.

Следствие 1.16 PA *неполна.*

Замечание 1.17 Сам Гёдель установил свою теорему при более слабом, чем Σ_1-корректность, предположении ω-непротиворечивости теории T. В дальнейшем Дж.Б. Россер усилил теорему Гёделя, доказав неполноту теории T всего лишь при условии её обычной непротиворечивости. Однако при этом возникает дополнительное (но не очень ограничительное) требование $T \vdash \mathsf{Q}$.

Теорема 1.18 (теорема Гёделя–Россера) *Если*

- *теория T содержит* Q*,*
- *T эффективно аксиоматизируема,*
- *T непротиворечива,*

то T неполна.

Замечание 1.19 Теорема Гёделя–Россера также применима к теориям в произвольном языке, в которых интерпретируема Q. К таким теориям относится, в частности, теория множеств ZFC.

Следствие 1.20 *ZFC неполна (при условии своей непротиворечивости).*

Теорема 1.21 (вторая теорема Гёделя о неполноте) *Если*

- PA *интерпретируема в T,*
- *T эффективно аксиоматизируема,*
- *T непротиворечива,*

то $T \nvdash \mathrm{Con}(T)$, где $\mathrm{Con}(T)$ – арифметическая формула, выражающая непротиворечивость T.

Замечание 1.22 Условие интерпретируемости PA в T во второй теореме Гёделя о неполноте было в дальнейшем ослаблено до знакомого нам условия интерпретируемости теории Q в T. Такое ослабление, однако, требует привлечения существенных новых идей в первоначальное доказательство Гёделя. В окончательном виде этот результат был получен чешским математиком П. Пудлаком (в 1985 году).

Замечание 1.23 Одним из следствий второй теоремы Гёделя о неполноте является то, что непротиворечивость PA нельзя доказать средствами самой теории PA. Подчеркнём, что речь в этой теореме не идёт о том, что непротиворечивость PA может вызывать сомнения, а лишь о том, что обоснование (очевидным образом) верного факта непротиворечивости PA требует допущений, выходящих за рамки этой теории.

Ситуация менее очевидная с теорией ZFC: мы также верим в непротиворечивость ZFC, но предположения, на основании которых мы могли бы обосновать этот факт, не могут быть формализованы внутри самой ZFC, то есть должны выходить за рамки «обычной», общепринятой математики! Поэтому, в частности, в формулировке следствия 1.20 мы сделали оговорку относительно условия о непротиворечивости ZFC.[1]

2 Вычислимость и определимость

Первая теорема Гёделя о неполноте и теорема Гёделя–Россера будут выведены нами из одного результата, указывающего на фундаментальную связь между понятиями вычислимости и определимости в арифметике. Мы называем этот результат теоремой о Σ_1-определимости. Мы введём три класса арифметических формул: ограниченные формулы, Σ_1-формулы и Π_1-формулы.

Определение 2.1 *Ограниченными* называются формулы, все вхождения кванторов в которые имеют вид

- $\forall x\,(x \le t \to A(x))$ (сокращённо $\forall x{\le}t\,A(x)$), или
- $\exists x\,(x \le t \wedge A(x))$ (сокращённо $\exists x{\le}t\,A(x)$),

где t — произвольный терм арифметического языка.

Множество всех ограниченных формул обозначаем Δ_0.

[1] В теории множеств рассматриваются дополнительные аксиомы, так называемые аксиомы больших кардиналов, из которых следует непротиворечивость ZFC. Однако эти аксиомы все-таки нельзя считать общепринятыми.

Определение 2.2 Σ_1-*формулами* называются формулы вида $\exists \vec{x}\, A(\vec{x}, \vec{a})$, где $A \in \Delta_0$. Множество всех Σ_1-формул обозначаем Σ_1. Π_1-*формулами* называются формулы вида $\forall \vec{x}\, A(\vec{x}, \vec{a})$, где $A \in \Delta_0$. Множество всех Π_1-формул обозначаем Π_1.

Нетрудно видеть, что всякая Δ_0-формула определяет в стандартной модели $\mathbb{N}$ некоторый разрешимый предикат, а Σ_1-формула — перечислимый предикат, то есть имеет место следующая лемма.

Лемма 2.3 *Пусть список* $\vec{a} = (a_1, \ldots, a_k)$ *содержит все свободные переменные формулы* $A(\vec{a})$. *Тогда*

(i) *если* $A(\vec{a}) \in \Delta_0$, *то множество* $\{\vec{n} \in \mathbb{N}^k : \mathbb{N} \vDash A[\vec{n}]\}$ *разрешимо;*

(ii) *если* $A(\vec{a}) \in \Sigma_1$, *то множество* $\{\vec{n} \in \mathbb{N}^k : \mathbb{N} \vDash A[\vec{n}]\}$ *перечислимо.*

Доказательство. Утверждение (ii) следует из (i) по теореме о проекции разрешимого множества. Утверждение (i) доказывается индукцией по построению A. Атомарные формулы арифметического языка очевидным образом определяют разрешимые множества, и разрешимые множества замкнуты относительно булевых операций.

Рассмотрим формулу $A(\vec{a}) = \forall x{\leq}t(\vec{a})\, B(x, \vec{a})$, где мы считаем, что список переменных $\vec{a}$ содержит все свободные переменные формулы B и терма t. Тогда истинностное значение формулы $A[\vec{n}]$ можно узнать, вычислив значение $m = t(\vec{n})$ и проверив полным перебором, что $\mathbb{N} \vDash B[i, \vec{n}]$ для каждого $i \leq m$. Аналогично рассматривается ограниченный квантор существования. $\dashv$

2.1 Теорема о Σ_1-определимости и вывод из неё первой теоремы Гёделя

Определение 2.4 Множество $P \subseteq \mathbb{N}^k$ Σ_1-*определимо в* $\mathbb{N}$, если существует $A(a_1, \ldots, a_k) \in \Sigma_1$ такая, что для всех $n_1, \ldots, n_k \in \mathbb{N}$

$$\langle n_1, \ldots, n_k \rangle \in P \iff \mathbb{N} \vDash A[n_1, \ldots, n_k].$$

Теорема 2.5 (о Σ_1-определимости) $P \subseteq \mathbb{N}^k$ *перечислимо* $\iff$ P Σ_1-*определимо в* $\mathbb{N}$.

Из этой теоремы мы получаем ряд важных следствий, включающих первую теорему Гёделя о неполноте.

Теорема 2.6 *Множество $Th_{\Pi_1}(\mathbb{N})$ всех предложений $A \in \Pi_1$ таких, что $\mathbb{N} \vDash A$, неперечислимо.*

Доказательство. Пусть $K \subseteq \mathbb{N}$ перечислимо и неразрешимо. По теореме о Σ_1-определимости найдётся Σ_1-формула $K(a)$ такая, что

$$n \in K \iff \mathbb{N} \vDash K[n] \iff \mathbb{N} \vDash K(\overline{n}).$$

Отсюда получаем

$$n \notin K \iff \mathbb{N} \nvDash K(\overline{n}) \iff \mathbb{N} \vDash \neg K(\overline{n}).$$

Если $Th_{\Pi_1}(\mathbb{N})$ перечислимо, то таково и $\{n \in \mathbb{N} : \mathbb{N} \vDash \neg K(\overline{n})\}$, так как по n эффективно восстанавливается Π_1-формула, эквивалентная $\neg K(\overline{n})$ (подстановка нумерала в фиксированную формулу является вычислимой операцией). Таким образом, будет перечислимым также и дополнение множества K, что противоречит теореме Чёрча–Поста. $\dashv$

Теорема 2.7 *Если T эффективно аксиоматизируема и Σ_1-корректна, то найдётся предложение $A \in \Pi_1$ такое, что $T \nvdash A$ и $T \nvdash \neg A$.*

Доказательство. Сначала заметим, что Σ_1-корректность теории T влечет ее Δ_0-корректность, а Δ_0-корректность эквивалентна формально более сильному условию Π_1-корректности: если $T \vdash \forall x\, \varphi(x)$ и $\varphi \in \Delta_0$, то для любого $n \in \mathbb{N}$ $T \vdash \varphi(\overline{n})$, откуда в силу Δ_0-корректности $\mathbb{N} \vDash \varphi(\overline{n})$. Таким образом $\mathbb{N} \vDash \forall x\, \varphi(x)$.

Из Π_1-корректности теории T следует, что $\{\pi \in \Pi_1 : T \vdash \pi\} \subseteq Th_{\Pi_1}(\mathbb{N})$. Поскольку теория T эффективно аксиоматизируема множество всех теорем T перечислимо, значит по теореме 2.6 найдётся $A \in Th_{\Pi_1}(\mathbb{N})$ такое, что $T \nvdash A$. Теперь заметим, что формула $\neg A$ эквивалентна Σ_1-предложению. Так как $\mathbb{N} \nvDash \neg A$, а теория T Σ_1-корректна, имеем $T \nvdash \neg A$. $\dashv$

2.2 Доказательство теоремы о Σ_1-определимости

Идея доказательства состоит в том, чтобы для каждой машины Тьюринга M выписать Σ_1-формулу $T_M(\vec{x})$, выражающую тот факт, что на входе, кодирующем $\vec{x}$, машина M завершает работу. Это достигается путём кодирования машин Тьюринга и описания их вычислений на арифметическом языке.

2.2.1 Обогащение модели с помощью Δ_0-определений

Искомую формулу удобно строить, обогащая сигнатуру арифметики новыми предикатными и функциональными символами с помощью Δ_0-определений.

Пусть Σ — сигнатура, содержащая арифметическую, и $\mathbb{N}_\Sigma$ — обогащение стандартной модели арифметики до некоторой модели сигнатуры Σ. Говорим, что модель $\mathbb{N}_\Sigma$ обладает *свойством ограниченности*, если для любого терма $t(\vec{a})$ сигнатуры Σ найдётся арифметический терм $t'(\vec{a})$ такой, что $\mathbb{N}_\Sigma \vDash \forall \vec{x}\,(t(\vec{x}) \leq t'(\vec{x}))$. *Ограниченными формулами* сигнатуры Σ называем формулы сигнатуры Σ, все вхождения кванторов в которые ограничены термами Σ. Множество всех таких формул обозначаем $\Delta_0(\Sigma)$.

Мы рассматриваем два типа определений:

- Определение предиката P формулой $A \in \Delta_0(\Sigma)$, обозначаемое

  $$P(\vec{a}) :\leftrightarrow A(\vec{a}).$$

  Сигнатура Σ расширяется новым предикатным символом P. В стандартной модели $\mathbb{N}$ символу P соответствует предикат

  $$P_\mathbb{N} \rightleftharpoons \{\vec{n} \in \mathbb{N}^k : \mathbb{N}_\Sigma \vDash A[\vec{n}]\}.$$

- Определение функции f формулой $F \in \Delta_0(\Sigma)$, обозначаемое

  $$f(\vec{a}) = b :\leftrightarrow F(\vec{a}, b).$$

  Сигнатура Σ расширяется новым функциональным символом f. В стандартной модели $\mathbb{N}$ символу f соответствует функция $f_\mathbb{N}$ с графиком

  $$F_\mathbb{N} \rightleftharpoons \{\langle \vec{n}, m\rangle : \mathbb{N}_\Sigma \vDash F[\vec{n}, m]\}.$$

  Такое определение считается корректным, если

  – $F_\mathbb{N}$ действительно задаёт график функции, то есть

  $$\mathbb{N}_\Sigma \vDash \forall \vec{x} \exists! y\, F(\vec{x}, y);$$

  – функция $f_\mathbb{N}$ ограничена некоторым термом $t(\vec{a})$ сигнатуры Σ, то есть

  $$\mathbb{N}_\Sigma \vDash \forall \vec{x}, y\,(F(\vec{x}, y) \to y \leq t(\vec{x})).$$

Следующие простейшие примеры показывают, как строить одни Δ_0-определения на основе других.

$$\begin{aligned}
x \neq y &\;:\leftrightarrow\; \neg\, x = y \\
x < y &\;:\leftrightarrow\; x \leq y \wedge x \neq y \\
x \dot{-} y = z &\;:\leftrightarrow\; (y \leq x \wedge x = z + y) \vee (\neg\, y \leq x \wedge z = 0) \\
x \mid y &\;:\leftrightarrow\; \exists z \leq y\ z \cdot x = y \\
p \text{ просто} &\;:\leftrightarrow\; p \neq 0 \wedge p \neq 1 \wedge \forall m \leq p\, (m \mid p \to m = 1 \vee m = p) \\
x \text{ степень простого } p &\;:\leftrightarrow\; x \neq 0 \wedge p \text{ просто} \wedge \forall m \leq x\, (m \mid x \to p \mid m \vee m = 1)
\end{aligned}$$

Перевод $f \mapsto F$, $P \mapsto A$ задает интерпретацию модели $(\mathbb{N}_\Sigma; P_\mathbb{N}, f_\mathbb{N})$ в $\mathbb{N}_\Sigma$. Такие интерпретации I называем *ограниченными*. Как обычно, всякой формуле A в расширенной сигнатуре соответствует её перевод A^I в сигнатуру Σ.

Лемма 2.8 *Пусть $\mathbb{N}_\Sigma$ обладает свойством ограниченности. Тогда*

(i) *$(\mathbb{N}_\Sigma; P_\mathbb{N}, f_\mathbb{N})$ обладает тем же свойством;*

(ii) *если A — ограниченная формула расширенного языка, то перевод A^I эквивалентен $\Delta_0(\Sigma)$-формуле в модели $\mathbb{N}_\Sigma$.*

Доказательство. Утверждение (i) получается простой индукцией по построению терма t, с учётом монотонности всех функций сигнатуры арифметики.

Утверждение (ii) очевидно для случая определения предиката P, поскольку формула A^I получается заменой в A всех вхождений вида $P(t_1, \ldots, t_k)$ на $A(t_1, \ldots, t_k)$.

Для случая определения функции f рассуждаем индукцией по построению формулы A.

Сначала докажем утверждение для атомарных формул A. Такие формулы имеют вид $Q(t_1, \ldots, t_k)$, для некоторого предикатного символа Q сигнатуры Σ и некоторых термов $t_1, \ldots, t_k$ расширенной сигнатуры. Применяем индукцию по общему количеству вхождений символа f в термы $t_1, \ldots, t_k$.

Допустим, например, что f входит в t_1. Рассмотрим самое внутреннее такое вхождение; тогда t_1 имеет вид $t_1'(f(s_1, \ldots, s_n))$, где термы s_i

не содержат символа f, и t_1' имеет на одно вхождение f меньше, чем t_1. Поскольку функция f ограничена некоторым Σ-термом t, перевод $Q(t_1,\ldots,t_k)^I$ равносилен в $\mathbb{N}_\Sigma$ формуле

$$\exists x \le t(s_1,\ldots,s_n)\,[(Q(t_1'(x),t_2,\ldots,t_k))^I \wedge F(s_1,\ldots,s_n,x)],$$

где $(Q(t_1'(x),t_2,\ldots,t_k))^I$ эквивалентна ограниченной формуле по предположению индукции.

Если формула A имеет вид $(A_1 \wedge A_2)$, $(A_1 \vee A_2)$, $\neg A_1$ или $(A_1 \to A_2)$, утверждение легко следует из предположения индукции.

Пусть A имеет вид $\forall x{\le}s\, B(x)$. Воспользуемся частью (i) и рассмотрим арифметический терм s' такой, что

$$(\mathbb{N}_\Sigma; P_\mathbb{N}, f_\mathbb{N}) \vDash s \le s'.$$

Тогда перевод A^I равносилен формуле

$$\forall x{\le}s'\,((x \le s)^I \to B(x)^I).$$

Заметим, что формула $(x \le s)^I$ ограничена как перевод атомарной формулы, а ограниченность $B(x)^I$ следует из предположения индукции. Случай ограниченного квантора существования рассматривается аналогично. $\dashv$

Следствие 2.9 *Композиция ограниченных интерпретаций ограничена.*

Теперь мы применим технику Δ_0-определений к формализации в арифметике вычислений машин Тьюринга.

2.2.2 Кодирование p-ичных слов

Пусть p — простое. Любое $x > 0$ однозначно представляется в виде

$$x = a_n \cdot p^n + a_{n-1} \cdot p^{n-1} + \cdots + a_1 \cdot p + a_0,$$

где $a_0,\ldots,a_n < p$ и $a_n \neq 0$. Мы обозначаем p-ичную запись x

$$x = (a_n a_{n-1} \ldots a_0)_p.$$

Слово $a_{n-1}\ldots a_0$ в алфавите $\{0,\ldots,p-1\}$ кодируем числом x, представимым как $1a_{n-1}\ldots a_0$ в p-ичной записи. Таким образом, пустое слово Λ кодируется числом 1, а 0 не является кодом никакого p-ичного слова. Однобуквенная последовательность $\langle a\rangle$ кодируется числом $(1a)_p = p+a$.

Наша первая задача — определить Δ_0-формулой функцию конкатенации p-ичных слов. Затем мы сможем определить и функцию длины слова.

Пусть числа x и y в p-ичной системе счисления записываются как

$$x = (1a_{n-1} \dots a_0)_p, \quad y = (1b_{m-1} \dots b_0)_p.$$

Тогда конкатенация $x *_p y = (1a_{n-1} \dots a_0 b_{m-1} \dots b_0)_p$ представляется как

$$x *_p y = x \cdot \nu(y, p) + (y \dot{-} \nu(y, p)),$$

где $\nu(y, p) = (1\underbrace{0 \dots 0}_{m \text{ раз}})_p$. Значение функции ν Δ_0-определимо как наибольшая степень p, не превосходящая y:

$$\nu(y, p) = u \leftrightarrow u \text{ степень } p \wedge u \le y \wedge p \cdot u > y.$$

Тем самым $x *_p y$ также Δ_0-определима.

Число x кодирует некоторое p-ичное слово, если выполнено условие

$$\mathrm{Word}_p(x) :\leftrightarrow x < 2\nu(x, p).$$

Через конкатенацию легко определить

$$\begin{array}{lcll}
x \preccurlyeq_p y & :\leftrightarrow & \exists z \le y\, (\mathrm{Word}_p(z) \wedge x *_p z = y) & \text{«}x \text{ начало } y\text{»} \\
x \subseteq_p y & :\leftrightarrow & \exists u, v \le y\, (\mathrm{Word}_p(u) \wedge \mathrm{Word}_p(v) \wedge u *_p x *_p v = y) & \text{«}x \text{ подслово } y\text{»} \\
a \in_p y & :\leftrightarrow & a < p \wedge (p + a) \subseteq_p y & \text{«цифра } a \text{ входит в } y\text{»} \\
\langle abc \rangle_p & :\leftrightarrow & (p + a) *_p (p + b) *_p (p + c) &
\end{array}$$

2.2.3 Длина слова

Теперь мы дадим Σ_1-определение функции длины p-ичного слова x, обозначаемой $|x|_p$. Предикат $|x|_p = n$ выполняется, если и только если существует взаимно однозначное и сохраняющее порядок отображение из $\{0, \dots, n\}$ в множество всех начальных отрезков слова x, упорядоченное по $\preccurlyeq_p$. Такое отображение можно закодировать как q-слово вида

$$s = \#0x_0\#1x_1\#2x_2\# \dots \#nx_n, \tag{1}$$

где $x_i \preccurlyeq_p x$. Простое число q при этом можно выбрать заведомо большим, чем $\max(x, n) + 1$; таким образом числа $0, \dots, n$, символ $\#$ и каждый из

начальных отрезков x_i представляются одной цифрой q-ичного слова s. Через $\ulcorner\#\urcorner$ обозначаем число, соответствующее символу $\#$. Определим отношение, имитирующее значение функции s на аргументе i:

$$s[i] \approx y \quad :\leftrightarrow \quad \langle \ulcorner\#\urcorner i y \rangle_q \subseteq_q s.$$

Условия, которым удовлетворяет слово s, можно описать Δ_0-формулами (параметр p — фиксированный нумерал) следующим образом:

1. $\mathrm{Word}_q(s) \wedge \forall i \leq n\, \exists y < q\, (s[i] \approx y \wedge y \preccurlyeq_p x)$
2. $\forall i \leq n\, \forall u, v < q\ \ (s[i] \approx u \wedge s[i] \approx v \to u = v)$
3. $\forall i < n\, \forall u, v < q\ \ (s[i] \approx u \wedge s[i+1] \approx v \to \exists a < p\ \ v = u *_p (p + a))$
4. $s[0] \approx 1 \wedge s[n] \approx x$

Отметим, что условия не гарантируют, что s должно иметь в точности вид (1), слово s может содержать какие-то лишние части, не имеющие отношения к кодируемой им функции.

Пусть $F(s, q, x, n)$ — конъюнкция этих условий. Тогда $|x|_p = n$ определимо как

$$\exists s, q\, (q > p \wedge q > x \wedge q \text{ просто} \wedge F(s, q, x, n)).$$

2.2.4 Последовательности слов в алфавите Σ

Пусть Σ — конечный алфавит, $|\Sigma| = n > 1$. Выберем константу p большей, чем n. Последовательность $\langle w_1, \ldots, w_s \rangle$ Σ-слов кодируем числом, соответствующим слову $w_1; w_2; \ldots; w_s$ при p-ичном кодировании. Здесь «;» — разделительный символ, $; \notin \Sigma$ и можно считать $\ulcorner;\urcorner = n$. Код пустой последовательности $\langle\ \rangle$ положим равным 0.

Заметим, что для любого слова $w \in \Sigma^*$, $\ulcorner\langle w\rangle\urcorner = \ulcorner w\urcorner$, в частности, $\ulcorner\langle\Lambda\rangle\urcorner = 1$.

Если константа p фиксирована, то мы опускаем индекс p для предикатов и функций $*_p$, Word_p, $|\cdot|_p$, $\in_p$, определенных ранее.

Определяем следующие предикаты и функции: $\Sigma(x)$ «x есть буква алфавита Σ», $\mathrm{Word}_\Sigma(x)$ «x есть слово в алфавите Σ», $\mathrm{Seq}_\Sigma(x)$ «x есть последовательность Σ-слов», $x; y$ «конкатенация последовательностей x и y», $x \subseteq_s y$ «x есть подпоследовательность y», $x \in_s y$ «x есть элемент

последовательности y».

$$\begin{aligned}
\Sigma(x) &:\leftrightarrow x < \underline{n} \\
\mathrm{Word}_\Sigma(x) &:\leftrightarrow \mathrm{Word}(x) \wedge \forall y \le x\, (y \in x \to \Sigma(y)) \\
\mathrm{Seq}_\Sigma(x) &:\leftrightarrow \mathrm{Word}(x) \wedge \forall y \in x\, (\Sigma(y) \vee y = \ulcorner ; \urcorner) \vee x = 0 \\
x; y = z &:\leftrightarrow (x = 0 \wedge z = y) \vee (y = 0 \wedge z = x) \vee \\
&\quad\ (x \neq 0 \wedge y \neq 0 \wedge z = x * \ulcorner ; \urcorner * y) \\
x \in_s y &:\leftrightarrow \mathrm{Word}_\Sigma(x) \wedge (x = y \vee \ulcorner ; \urcorner * x * \ulcorner ; \urcorner \subseteq y)
\end{aligned}$$

2.2.5 Кодирование Машин Тьюринга

Зафиксируем произвольную машину Тьюринга M с рабочим алфавитом Σ, содержащим символ пробела $\#$, и алфавитом состояний Q с выделенными состояниями q_1 (начальное) и q_0 (завершающее). Мы будем кодировать слова и последовательности слов в алфавите $\Sigma \cup Q \cup \{L, N, R\}$ и соответствующим образом фиксируем константу p (см. выше). Конечные множества символов и команд для данной машины легко определить формулами, перечисляющими их поэлементно. Пусть формула $\Sigma(x)$ определяет рабочий алфавит и $Q(x)$ — алфавит состояний. Формула $\Gamma(x) \rightleftharpoons Q(x) \vee \Sigma(x)$ задаёт их объединение.

Команда $q_i S_j \to q_k S_l \nu$, где $\nu \in \{L, N, R\}$, кодируется как

$$\ulcorner q_i \urcorner * \ulcorner S_j \urcorner * \ulcorner q_k \urcorner * \ulcorner S_l \urcorner * \ulcorner \nu \urcorner.$$

Формула $P(x)$ определяет множество команд машины M.

2.2.6 Конфигурации

Конфигурация машины M кодируется словом вида uqv, где u, v — слова в рабочем алфавите, слово v непусто, головка находится в состоянии $q \in Q$ и обозревает первый символ слова v. Таким образом, множество конфигураций определяется как

$$\begin{aligned}
\mathrm{Config}(z) &:\leftrightarrow \exists u, v, q \le z\, (\mathrm{Word}_\Sigma(u) \wedge \mathrm{Word}_\Sigma(v) \wedge Q(q) \wedge \\
&\quad\ v \neq 1 \wedge z = u * q * v)
\end{aligned}$$

2.2.7 Переходы

Следующая формула $\mathrm{Step}_M(x, y)$ определяет отношение «машина M переходит за один шаг из конфигурации x в конфигурацию y». Тем самым

эта формула описывает применение одной команды из программы P заданной машины Тьюринга.

Пусть некоторая команда имеет вид $ra \to qb\nu$. В зависимости от направления движения головки разбираются один или два случая: если $\nu = N$, то конфигурация $urav$ переходит в $uqbv$. Если $\nu = L$ и слово слева от головки непусто (имеет вид uc), то конфигурация $urav$ переходит в $uqcbv$, иначе конфигурация имеет вид rav и переходит в $q\#bv$ (слева лента заполнена пробелами). Аналогично описывается движение головки направо, то есть случай $\nu = R$.

$$
\begin{array}{l}
\mathrm{Step}_M(x,y) :\leftrightarrow \\
\mathrm{Config}(x) \wedge \mathrm{Config}(y) \wedge \exists u,v,r,q,a,b,c \subseteq_w x * y \\
\quad [\mathrm{Word}_\Sigma(u) \wedge \mathrm{Word}_\Sigma(v) \wedge Q(r) \wedge Q(q) \wedge \Sigma(a) \wedge \Sigma(b) \wedge \Sigma(c) \wedge \\
\qquad [(x = u * r * a * v \wedge y = u * q * b * v \wedge P(r * a * q * b * \ulcorner N \urcorner)) \\
\qquad \vee (x = u * c * r * a * v \wedge y = u * q * c * b * v \wedge P(r * a * q * b * \ulcorner L \urcorner)) \\
\qquad \vee (x = r * a * v \wedge y = q * \ulcorner \# \urcorner * b * v \wedge P(r * a * q * b * \ulcorner L \urcorner)) \\
\qquad \vee (x = u * r * a * v \wedge v \neq 1 \wedge y = u * b * q * v \wedge P(r * a * q * b * \ulcorner R \urcorner)) \\
\qquad \vee (x = u * r * a \wedge y = u * b * q * \ulcorner \# \urcorner \wedge P(r * a * q * b * \ulcorner R \urcorner)) \\
\qquad] \\
\quad]
\end{array}
$$

Теперь мы можем определить понятие (протокола) вычисления машины M.

2.2.8 Вычисления

Определим отношения $\mathrm{Init}_M(x,z)$ «z есть начальная конфигурация с входом x», $\mathrm{Stop}_M(z)$ «z есть заключительная конфигурация», и $\mathrm{Comp}_M(x,z)$ «z есть протокол завершающегося вычисления машины M на входе x». Определения, приводимые ниже, говорят сами за себя.

$$
\begin{array}{rcl}
\mathrm{Init}_M(x,z) & :\leftrightarrow & \mathrm{Config}(z) \wedge z = \ulcorner q_1 \urcorner * \ulcorner \# \urcorner * x \\
\mathrm{Stop}_M(z) & :\leftrightarrow & \mathrm{Config}(z) \wedge \ulcorner q_0 \urcorner \in z \\
\mathrm{Comp}_M(x,z) & :\leftrightarrow & \mathrm{Seq}_\Gamma(z) \wedge \exists v \in_s z\ \mathrm{Stop}_M(v) \wedge \forall u,v,w \leq z \\
 & & (z = u;v;w \wedge \mathrm{Word}_\Gamma(v) \to \\
 & & \quad (\mathrm{Init}_M(x,v) \vee \exists y \in_s u\ \mathrm{Step}_M(y,v)))
\end{array}
$$

2.2.9 Кодирование входа и предикат остановки

Наконец, мы должны вспомнить, что для машины Тьюринга, вычисляющей функцию натуральных аргументов, вместо последовательности чисел $\langle n_1, \ldots, n_k \rangle$ мы подаём на вход слово $1^{n_1}\$ \ldots \$1^{n_k}$ в алфавите $\{1, \$\}$.

Пусть Σ содержит $1, \$$. Положим для любого $n \in \mathbb{N}$

$$\mathrm{code}(n) \rightleftharpoons 1^n \rightleftharpoons 1 \ldots 1 \quad (n \text{ раз}).$$

Функция code определяется как

$$\mathrm{code}(x) = y \quad :\leftrightarrow \quad \mathrm{Word}(y) \wedge |y|_p = x \wedge \forall y \in_p x \; y = \ulcorner 1 \urcorner$$

Теперь мы можем выразить тот факт, что машина M на входе, кодирующем $\langle x_1, \ldots, x_k \rangle$, завершает работу:

$$T_M(x_1, \ldots, x_k) :\leftrightarrow \exists z \, \mathrm{Comp}_M(\mathrm{code}(x_1) * \ulcorner \$ \urcorner * \cdots * \ulcorner \$ \urcorner * \mathrm{code}(x_k), z)$$

Имеем:

$$\mathbb{N} \vDash T_M[n_1, \ldots, n_k] \iff !M(n_1, \ldots, n_k).$$

Тем самым доказательство теоремы о Σ_1-определимости завершено. Заметим, что построенная нами формула содержит один единственный неограниченный квантор существоваения по модулю определения функции code. Определение code также содержит неограниченный квантор существования из-за использования функции длины, однако эти кванторы можно вынести вперед и тем самым формула логически эквивалентна Σ_1-формуле. $\dashv$

3 Теорема Гёделя–Россера

Теорема Гёделя–Россера базируется на одном принципиальном факте, касающемся теории MA и содержащих её теорий. Как было отмечено выше, теория MA очень слаба для доказательства утверждений с неограниченными кванторами всеобщности. С другой стороны, следующая теорема показывает, что MA достаточно сильна для доказательства всех истинных Σ_1-утверждений.

3.1 Σ_1-полнота

Определение 3.1 Теория T в арифметическом языке называется Σ_1-*полной*, если для любого предложения $A \in \Sigma_1$

$$\mathbb{N} \vDash A \;\Rightarrow\; T \vdash A.$$

Теорема 3.2 *Теория* MA Σ_1-*полна.*

Доказательство. Идея доказательства Σ_1-полноты проста: истинность любого Σ_1-предложения A может быть эффективно установлена с помощью процедуры, описанной в лемме 2.3. Это вычисление, по существу, представляет собой доказательство A в MA.

Более аккуратное доказательство получается из последовательности простых лемм, приводимой ниже.

Лемма 3.3 *Для любых* $m, n \in \mathbb{N}$, *в* MA *доказуемо*

(i) $\overline{m} + \overline{n} = \overline{m+n}$

(ii) $\overline{m} \cdot \overline{n} = \overline{m \cdot n}$

Доказательство. Каждое из утверждений доказывается «внешней» индукцией по n. То есть мы используем индукцию для обоснования выводимости в MA, а не в рамках самой теории MA (где индукция не постулируется в качестве аксиомы). Напомним, что $\overline{0}$ есть 0 и $\overline{n+1}$ есть $S(\overline{n})$.

(i) Базис: $\overline{m} + 0 = \overline{m}$, по аксиоме 3.

Шаг индукции. Допустим, что в MA доказуемо $\overline{m} + \overline{n} = \overline{m+n}$. Достроим этот вывод до вывода формулы $\overline{m} + S(\overline{n}) = S(\overline{m+n})$:

1. $\overline{m} + \overline{n} = \overline{m+n}$ (гипотеза)
2. $S(\overline{m} + \overline{n}) = S(\overline{m+n})$ (по аксиоме равенства)
3. $\overline{m} + S(\overline{n}) = S(\overline{m} + \overline{n})$ (по аксиоме 3)
4. $\overline{m} + S(\overline{n}) = S(\overline{m+n})$ (из 2, 3)

Доказательство (ii) аналогично. $\dashv$

Лемма 3.4 *Для любого арифметического терма* $t(b_1, \ldots, b_m)$ *и любых* $k_1, \ldots, k_m, l \in \mathbb{N}$,

$$\mathbb{N} \vDash t(k_1, \ldots, k_m) = l \quad \Longrightarrow \quad \text{MA} \vdash t(\overline{k}_1, \ldots, \overline{k}_m) = \overline{l}.$$

Доказательство. Внешняя индукция по построению t. Если t — переменная или константа 0, утверждение очевидно. Для составных термов утверждение получается из леммы 3.3 по предположению индукции. Например, если t имеет вид $t_1 + t_2$, то для некоторых $l_1, l_2 \in \mathbb{N}$ формулы $t_1(\overline{k}_1, \ldots, \overline{k}_m) = \overline{l}_1$ и $t_2(\overline{k}_1, \ldots, \overline{k}_m) = \overline{l}_2$ доказуемы. Мы достраиваем эти выводы следующей последовательностью формул:

1. $t_1(\overline{k}_1, \ldots, \overline{k}_m) = \overline{l}_1$ (гипотеза)
2. $t_2(\overline{k}_1, \ldots, \overline{k}_m) = \overline{l}_2$ (гипотеза)
3. $t_1(\overline{k}_1, \ldots, \overline{k}_m) + t_2(\overline{k}_1, \ldots, \overline{k}_m) = \overline{l}_1 + \overline{l}_2$ (по аксиоме равенства)
4. $\overline{l}_1 + \overline{l}_2 = \overline{l_1 + l_2}$ (лемма 3.3)
5. $t_1(\overline{k}_1, \ldots, \overline{k}_m) + t_2(\overline{k}_1, \ldots, \overline{k}_m) = \overline{l_1 + l_2}$ (по аксиоме равенства)

Функции последователя и умножения рассматриваются аналогично. $\dashv$

Лемма 3.5 *Для любых $m, n \in \mathbb{N}$,*

(i) *если $m \neq n$, то $\mathsf{MA} \vdash \neg\, \overline{m} = \overline{n}$;*

(ii) *если $m \leq n$, то $\mathsf{MA} \vdash \overline{m} \leq \overline{n}$;*

(iii) *если $m < n$, то $\mathsf{MA} \vdash \neg\, \overline{n} \leq \overline{m}$.*

Доказательство. (i) Считаем без ограничения общности, что $m < n$. Рассуждаем индукцией по m. Если $m = 0$, то $\overline{n}$ совпадает с $S(\overline{n-1})$, и результат следует из аксиом следования. Если же $m > 0$, то по предположению индукции найдётся вывод в MA формулы $\neg\, \overline{m-1} = \overline{n-1}$. Продолжим этот вывод следующим образом:

1. $\neg\, \overline{m-1} = \overline{n-1}$ (гипотеза)
2. $\overline{m} = \overline{n} \to \overline{m-1} = \overline{n-1}$ (аксиома 2)
3. $\neg\, \overline{m} = \overline{n}$ (из 1, 2)

и получаем вывод в MA формулы $\neg\, \overline{m} = \overline{n}$.

(ii) Внешняя индукция по m.

Базис. Для $m = 0$ утверждение сводится к аксиоме $0 \leq \overline{n}$.

Шаг индукции. Пусть $m+1 \leq n$, тогда $m \leq n-1$. По предположению индукции $\mathsf{MA} \vdash \overline{m} \leq \overline{n-1}$. Отсюда $\mathsf{MA} \vdash S(\overline{m}) \leq S(\overline{n-1})$. При этом $S(\overline{m})$ совпадает с $\overline{m+1}$, а $S(\overline{n-1})$ — с $\overline{n}$. Значит, $\mathsf{MA} \vdash \overline{m+1} \leq \overline{n}$.

(iii) Внешняя индукция по m.

Базис. Допустим $0 = m < n$. Тогда $\overline{n} \leq 0$ влечёт $\overline{n} = 0$, откуда следует противоречие по утверждению (i). Значит, $\mathsf{MA} \vdash \neg\, \overline{n} \leq 0$.

Шаг индукции. Допустим $m + 1 < n$. Тогда $m < n - 1$ и по предположению индукции $\mathsf{MA} \vdash \neg\overline{n-1} \leq \overline{m}$. С другой стороны, в теории MA выводимо, что $\overline{n} \leq S(\overline{m})$ влечет $\overline{n-1} \leq \overline{m}$, отсюда $\mathsf{MA} \vdash \neg\, \overline{n} \leq S(\overline{m})$. $\dashv$

Лемма 3.6 *Для любого $m \in \mathbb{N}$, в MA доказуемо*

$$a \leq \overline{m} \leftrightarrow (a = 0 \vee \cdots \vee a = \overline{m}).$$

Доказательство. Импликация ($\leftarrow$) следует из Леммы 3.5(ii). Импликацию ($\rightarrow$) докажем внешней индукцией по m.

Базис. $a \leq 0 \rightarrow a = 0$ есть аксиома.

Шаг индукции. Рассуждая в MA, допустим $a \leq S\overline{m}$. Тогда $a = 0 \vee \exists x\, a = Sx$. Если $a = 0$, то все доказано. Если $a = Sx$, то $Sx \leq S\overline{m}$, откуда $x \leq \overline{m}$. По предположению индукции отсюда вытекает $x = 0 \vee x = S0 \vee \ldots x = \overline{m}$. Тогда $a = S0 \vee a = SS0 \vee \cdots \vee a = S\overline{m}$, что и требовалось доказать. $\dashv$

Еще одно свойство MA не нужно для доказательства теоремы о Σ_1-полноте, однако понадобится в дальнейшем.

Лемма 3.7 *Для любого $m \in \mathbb{N}$, в MA доказуемо*

$$\forall x\, (x \leq \overline{m} \vee \overline{m} \leq x).$$

Доказательство. Внешняя индукция по m. Для $m = 0$ утверждение следует из аксиомы $0 \leq \overline{m}$.

Шаг индукции. Допустим, что в MA доказуемо $\forall y\, (y \leq \overline{m} \vee \overline{m} \leq y)$. Рассуждаем в MA.

Рассмотрим произвольное x. По аксиоме предшественника $x = 0 \vee \exists y\, x = Sy$. В первом случае очевидно $x \leq S\overline{m}$. Во втором случае рассмотрим y такой, что $x = Sy$. Имеем $y \leq \overline{m} \vee \overline{m} \leq y$ по предположению. Если $y \leq \overline{m}$, то $Sy \leq S\overline{m}$, откуда $x \leq S\overline{m}$. Если же $\overline{m} \leq y$, то $S\overline{m} \leq Sy$, откуда $S\overline{m} \leq x$. В любом случае получаем $x \leq S\overline{m} \vee S\overline{m} \leq x$, что и требуется. $\dashv$

Лемма 3.8 *Для любой ограниченной формулы $A(b_1, \ldots, b_m)$ и любых $k_1, \ldots, k_m \in \mathbb{N}$,*

(i) $\mathbb{N} \vDash A(k_1, \ldots, k_m) \;\Rightarrow\; \mathsf{MA} \vdash A(\overline{k_1}, \ldots, \overline{k_m})$;

(ii) $\mathbb{N} \nvDash A(k_1, \ldots, k_m) \;\Rightarrow\; \mathsf{MA} \vdash \neg A(\overline{k_1}, \ldots, \overline{k_m})$.

Доказательство. Утверждения (i) и (ii) доказываем одновременно индукцией по построению формулы A. Рассмотрим следующие случаи.

1. A — атомарная формула вида $t_1(b_1, \ldots, b_m) = t_2(b_1, \ldots, b_m)$.

Если $\mathbb{N} \vDash A(k_1, \ldots, k_m)$, то для некоторого $l \in \mathbb{N}$, по лемме 3.4 мы имеем выводы формул $t_1(\overline{k_1}, \ldots, \overline{k_m}) = \overline{l}$ и $t_2(\overline{k_1}, \ldots, \overline{k_m}) = \overline{l}$ в MA. Отсюда получаем вывод $t_1(\overline{k_1}, \ldots, \overline{k_m}) = t_2(\overline{k_1}, \ldots, \overline{k_m})$, пользуясь аксиомами равенства.

Если $\mathbb{N} \nvDash A(k_1, \ldots, k_m)$, то для некоторых $l_1 \neq l_2$ имеем выводы формул $t_1(\overline{k_1}, \ldots, \overline{k_m}) = \overline{l_1}$ и $t_2(\overline{k_1}, \ldots, \overline{k_m}) = \overline{l_2}$ в MA по лемме 3.4. Лемма 3.5(i) даёт вывод $\neg \overline{l_1} = \overline{l_2}$, откуда мы получаем вывод

$$\neg\, t_1(\overline{k_1}, \ldots, \overline{k_m}) = t_2(\overline{k_1}, \ldots, \overline{k_m}),$$

пользуясь аксиомами равенства.

2. A — атомарная формула вида $t_1(b_1, \ldots, b_m) \leq t_2(b_1, \ldots, b_m)$.

Этот случай рассматривается аналогично, на основе леммы 3.5 (ii) и (iii).

3. A имеет вид $B \to C$ или $\neg B$.

В этом случае утверждение получается непосредственно из предположения индукции для формул B и C.

4. A имеет вид $\forall v \leq t\; B(v, b_1, \ldots, b_m)$. Можно считать, что терм t зависит от тех же переменных $b_1, \ldots, b_m$.

(i) Допустим $\mathbb{N} \vDash A(k_1, \ldots, k_m)$. По лемме 3.4 найдётся $l \in \mathbb{N}$ такое, что в MA доказуемо $t(\overline{k_1}, \ldots, \overline{k_m}) = \overline{l}$. Значит, для всех $k \leq l$ имеем $\mathbb{N} \vDash B(k, k_1, \ldots, k_m)$, и по предположению индукции получаем выводы формул $B(\overline{k}, \overline{k_1}, \ldots, \overline{k_m})$ для каждого $k \leq l$. Достроим их до вывода формулы $A(\overline{k_1}, \ldots, \overline{k_m})$ следующим образом:

1. $(a = 0 \vee \cdots \vee a = \overline{l}) \to B(a, \overline{k_1}, \ldots, \overline{k_m})$ (предп. индукции)
2. $a \leq \overline{l} \to B(a, \overline{k_1}, \ldots, \overline{k_m})$ (1, лемма 3.6)
3. $a \leq t(\overline{k_1}, \ldots, \overline{k_m}) \to B(a, \overline{k_1}, \ldots, \overline{k_m})$ (2, равенство)
4. $\forall v\, (v \leq t(\overline{k_1}, \ldots, \overline{k_m}) \to B(v, \overline{k_1}, \ldots, \overline{k_m}))$ (3)

(ii) Допустим $\mathbb{N} \nvDash A(k_1, \ldots, k_m)$. Тогда для некоторого

$$k \leq l = t(k_1, \ldots, k_m)$$

имеем $\mathbb{N} \not\models B(k, k_1, \ldots, k_m)$, а значит

$$\mathsf{MA} \vdash \neg B(\overline{k}, \overline{k_1}, \ldots, \overline{k_m})$$

по предположению индукции. Достраиваем этот вывод до вывода формулы $\neg A(\overline{k_1}, \ldots, \overline{k_m})$:

1. $\neg B(\overline{k}, \overline{k_1}, \ldots, \overline{k_m})$ (гипотеза)
2. $\overline{k} \leq \overline{l} \wedge \neg B(\overline{k}, \overline{k_1}, \ldots, \overline{k_m})$ (1, лемма 3.5(ii))
3. $\overline{k} \leq t(\overline{k_1}, \ldots, \overline{k_m}) \wedge \neg B(\overline{k}, \overline{k_1}, \ldots, \overline{k_m})$ (2, лемма 3.4)
4. $\exists v\, (v \leq t(\overline{k_1}, \ldots, \overline{k_m}) \wedge \neg B(v, \overline{k_1}, \ldots, \overline{k_m}))$ (3)

Лемма 3.8 доказана. $\dashv$

Завершим доказательство теоремы 3.2. Рассмотрим Σ_1-формулу $A(b_1, \ldots, b_m)$. Можно считать, что A имеет вид $\exists v A_0(v, b_1, \ldots, b_m)$, где $A_0 \in \Delta_0$. Если $\mathbb{N} \vDash A(k_1, \ldots, k_m)$, то для некоторого k имеем $\mathbb{N} \vDash A_0(k, k_1, \ldots, k_m)$. По лемме 3.8 (i) в MA доказуемо $A_0(\overline{k}, \overline{k_1}, \ldots, \overline{k_m})$. Отсюда логически следует $\exists v A_0(v, \overline{k_1}, \ldots, \overline{k_m})$. $\dashv$

Следствие 3.9

1. *Любая арифметическая теория T, содержащая* MA, *Σ_1-полна.*
2. *Арифметика* PA *Σ_1-полна.*

3.2 Доказательство теоремы Гёделя–Россера

Доказательству этой теоремы предпошлём следующую лемму.

Лемма 3.10 *Пусть $A, B \subseteq \mathbb{N}$ перечислимы и $A \cap B = \varnothing$. Тогда найдётся Σ_1-формула $\varphi(a)$ такая, что для любого $n \in \mathbb{N}$*

(i) $n \in A \;\Rightarrow\; \mathsf{MA} \vdash \varphi(\overline{n})$,

(ii) $n \in B \;\Rightarrow\; \mathsf{MA} \vdash \neg\varphi(\overline{n})$.

Доказательство. По теореме о Σ_1-определимости найдутся Δ_0-формулы A_0 и B_0 такие, что

$$\begin{aligned} n \in A &\iff \mathbb{N} \vDash \exists x\, A_0(\overline{n}, x), \\ n \in B &\iff \mathbb{N} \vDash \exists y\, B_0(\overline{n}, y). \end{aligned}$$

Для любой формулы C и терма t обозначим

$$\forall x < t\, C(x) \stackrel{\text{def}}{\iff} \forall x \le t\, (x = t \vee C(x)).$$

Положим теперь

$$\varphi(a) \stackrel{\text{def}}{\iff} \exists x\, (A_0(a,x) \wedge \forall y < x\, \neg B_0(a,y)).$$

Неформально $\varphi(a)$ утверждает, что работа алгоритма, принимающего множество A, на входе a заканчивается раньше работы алгоритма, принимающего B («Россеровское сравнение свидетелей»).

Если $n \in A$, то для некоторого m истинна формула

$$A_0(\overline{n}, \overline{m}) \wedge \forall y < \overline{m}\, \neg B_0(\overline{n}, y).$$

По теореме о Σ_1-полноте арифметики MA получаем, что эта формула доказуема в MA, откуда $\mathsf{MA} \vdash \varphi(\overline{n})$.

Если $n \in B$, то для некоторого m истинна формула

$$B_0(\overline{n}, \overline{m}) \wedge \forall y \le \overline{m}\, \neg A_0(\overline{n}, y). \qquad (*)$$

По теореме о Σ_1-полноте арифметики MA получаем, что эта формула доказуема в MA. Отсюда следует, что $\mathsf{MA} \vdash \neg\varphi(\overline{n})$. Поясним это следующим рассуждением, которое легко преобразовать в формальный вывод противоречия из гипотезы $\varphi(\overline{n})$ в MA:

Допустим $\varphi(\overline{n})$. Тогда для некоторого x

$$A_0(\overline{n}, x) \wedge \forall y < x\, \neg B_0(\overline{n}, y).$$

Если $x \le \overline{m}$, то имеем $\neg A_0(\overline{n}, x)$ в силу $(*)$, что противоречит $A_0(\overline{n}, x)$. Если же $\overline{m} \le x$, то можно считать $x \ne \overline{m}$ (иначе применим предыдущий случай). В силу $\forall y < x\, \neg B_0(\overline{n}, y)$ получаем $\neg B_0(\overline{n}, \overline{m})$, что противоречит $B_0(\overline{n}, \overline{m})$ из $(*)$. Осталось заметить, что по лемме 3.7 в MA выводимо

$$\forall x\, (x \le \overline{m} \vee \overline{m} \le x),$$

откуда следует требуемое противоречие. $\dashv$

Доказательство теоремы Гёделя–Россера. Пусть A, B — неотделимая пара перечислимых подмножеств $\mathbb{N}$. Воспользуемся леммой и рассмотрим соответствующую формулу φ. Для данной теории T рассмотрим множества

$$\begin{aligned} A' &\rightleftharpoons \{n \in \mathbb{N} : T \vdash \varphi(\overline{n})\}, \\ B' &\rightleftharpoons \{n \in \mathbb{N} : T \vdash \neg\varphi(\overline{n})\}. \end{aligned}$$

Поскольку T эффективно аксиоматизируема, оба эти множества перечислимы. Так как T непротиворечива, $A' \cap B' = \varnothing$. По лемме мы также имеем $A \subset A'$ и $B \subset B'$. Докажем, что найдётся $n \notin A' \cup B'$. Действительно, в противном случае A' и B' разбивают $\mathbb{N}$ (взаимно дополнительны) и по теореме Чёрча–Поста должны быть разрешимыми. Но это невозможно, так как в этом случае они отделяли бы A от B.

Если $n \notin A' \cup B'$, то очевидно $T \nvdash \varphi(\overline{n})$ и $T \nvdash \neg\varphi(\overline{n})$, то есть T неполна. Заметим, что построенное нами независимое утверждение принадлежит классу Σ_1 (а его отрицание — классу Π_1). $\dashv$

3.3 Неразрешимость арифметических теорий и исчисления предикатов

Теорема 3.11 *Пусть теория T удовлетворяет условиям теоремы Гёделя–Россера. Тогда множество доказуемых и множество опровержимых в T предложений неотделимы.*

Доказательство. Обозначим

$$\begin{aligned} P_T &\rightleftharpoons \{\varphi : T \vdash \varphi\}, \\ R_T &\rightleftharpoons \{\varphi : T \vdash \neg\varphi\}. \end{aligned}$$

В силу непротиворечивости T эти множества не пересекаются. Допустим, что некоторое разрешимое множество C отделяет P_T от R_T, то есть $P_T \subseteq C$ и $C \cap R_T = \varnothing$.

Как и в теореме Гёделя–Россера, рассмотрим неотделимую пару перечислимых множеств A, B, воспользуемся леммой и рассмотрим соответствующую формулу φ. Если $n \in A$, то $T \vdash \varphi(\overline{n})$, то есть $\varphi(\overline{n}) \in P_T$ и $\varphi(\overline{n}) \in C$. Если же $n \in B$, то $T \vdash \neg\varphi(\overline{n})$, то есть $\varphi(\overline{n}) \in R_T$ и $\varphi(\overline{n}) \notin C$. Значит, множество $\{n \in \mathbb{N} : \varphi(\overline{n}) \in C\}$ отделяет A от B. Это множество разрешимо, поскольку по n эффективно восстанавливается формула $\varphi(\overline{n})$ (для фиксированной φ). $\dashv$

Следствие 3.12 *Всякая теория T, удовлетворяющая условиям теоремы Гёделя–Россера, неразрешима.*

Следствие 3.13 *Неразрешимы следующие теории:* Q, MA, PA, *ZFC*.

Замечание 3.14 Заметим, что из неразрешимости теории T, удовлетворяющей условиям теоремы Гёделя–Россера, следует её неполнота (поскольку полные эффективно аксиоматизированные теории разрешимы).

Следствие 3.15 *Исчисление предикатов в арифметическом языке неразрешимо.*

Доказательство. Пусть $\tilde{\mathsf{MA}}$ означает конъюнкцию всех нелогических аксиом теории MA (включая аксиомы равенства). Для любого арифметического предложения A, по теореме о дедукции, $\mathsf{MA} \vdash A \iff \vdash \tilde{\mathsf{MA}} \to A$. Таким образом, для проверки выводимости A в MA было бы достаточно проверить выводимость формулы $\tilde{\mathsf{MA}} \to A$ в чистом исчислении предикатов, но первое невозможно. $\dashv$

Замечание 3.16 Последнее следствие, полученное американским логиком А. Чёрчем, показывает неразрешимость проблемы, которую Д. Гильберт считал одной из центральных проблем в математической логике (так называемая «Entscheidungsproblem»): не существует алгоритма, проверяющего данную формулу логики первого порядка на общезначимость.

Упражнение 3.17 *Докажите, что существует конечная сигнатура без функциональных символов и констант, для которой исчисление предикатов неразрешимо.*