Что можно делать с вещественными числами и нельзя делать с целыми числами

Ю. В. Матиясевич

Санкт-Петербургское отделение
Математического института им. В. А. Стеклова РАН

http://logic.pdmi.ras.ru/~yumat

Что можно делать с вещественными числами и нельзя делать с целыми числами

Ю. В. Матиясевич

Санкт-Петербургское отделение
Математического института им. В. А. Стеклова РАН

http://logic.pdmi.ras.ru/~yumat

Что можно делать с вещественными числами и нельзя делать с целыми числами

Часть 2. Десятая проблема Гильберта

Ю. В. Матиясевич

Санкт-Петербургское отделение
Математического института им. В. А. Стеклова РАН

http://logic.pdmi.ras.ru/~yumat

Что можно делать с вещественными числами и нельзя делать с целыми числами

Часть 2. Десятая проблема Гильберта

Шестая лекция

Ю. В. Матиясевич

Санкт-Петербургское отделение
Математического института им. В. А. Стеклова РАН

http://logic.pdmi.ras.ru/~yumat

Формализмы для описания параллельных/распределенных вычислений

Формализмы для описания параллельных/распределенных вычислений

Сети Петри (Petri net, place/transition net, P/T net)

Формализмы для описания параллельных/распределенных вычислений

Сети Петри (Petri net, place/transition net, P/T net)

Wikipedia:

Формализмы для описания параллельных/распределенных вычислений

Сети Петри (Petri net, place/transition net, P/T net)

Wikipedia: Petri nets were invented by Carl Adam Petri

Формализмы для описания параллельных/распределенных вычислений

Сети Петри (Petri net, place/transition net, P/T net)

Wikipedia: Petri nets were invented by Carl Adam Petri in August 1939

Формализмы для описания параллельных/распределенных вычислений

Сети Петри (Petri net, place/transition net, P/T net)

Wikipedia: Petri nets were invented by Carl Adam Petri in August 1939 – at the age of 13

Формализмы для описания параллельных/распределенных вычислений

Сети Петри (Petri net, place/transition net, P/T net)

Wikipedia: Petri nets were invented by Carl Adam Petri in August 1939 – at the age of 13 – for the purpose of describing chemical processes

Формализмы для описания параллельных/распределенных вычислений

Сети Петри (Petri net, place/transition net, P/T net)

Wikipedia: Petri nets were invented by Carl Adam Petri in August 1939 – at the age of 13 – for the purpose of describing chemical processes

Системы векторного сложения (systems of vector addition)

Системы векторного сложения

$$\Delta_1 = \langle \delta_{1,1}, \ldots, \delta_{1,n} \rangle$$
$$\vdots$$
$$\Delta_k = \langle \delta_{k,1}, \ldots, \delta_{k,n} \rangle$$

Системы векторного сложения

$$\begin{aligned}\Delta_1 &= \langle \delta_{1,1}, \ldots, \delta_{1,n} \rangle \\ &\vdots \\ \Delta_k &= \langle \delta_{k,1}, \ldots, \delta_{k,n} \rangle\end{aligned} \qquad \delta_{j,k} \in \mathbb{Z}$$

Системы векторного сложения

$$\begin{aligned} \Delta_1 &= \langle \delta_{1,1}, \ldots, \delta_{1,n} \rangle \\ &\vdots \\ \Delta_k &= \langle \delta_{k,1}, \ldots, \delta_{k,n} \rangle \end{aligned} \qquad \delta_{j,k} \in \mathbb{Z}$$

$$\alpha_1 \in \mathbb{N}, \ldots, \alpha_n \in \mathbb{N}$$

Системы векторного сложения

$$\Delta_1 = \langle \delta_{1,1}, \ldots, \delta_{1,n} \rangle$$
$$\vdots \qquad\qquad \delta_{j,k} \in \mathbb{Z}$$
$$\Delta_k = \langle \delta_{k,1}, \ldots, \delta_{k,n} \rangle$$

$$\alpha_1 \in \mathbb{N}, \ldots, \alpha_n \in \mathbb{N}$$

$A = \langle \alpha_1, \ldots, \alpha_n \rangle$

Системы векторного сложения

$$\begin{aligned} \Delta_1 &= \langle \delta_{1,1}, \ldots, \delta_{1,n} \rangle \\ &\vdots \\ \Delta_k &= \langle \delta_{k,1}, \ldots, \delta_{k,n} \rangle \end{aligned} \qquad \delta_{j,k} \in \mathbb{Z}$$

$$\alpha_1 \in \mathbb{N}, \ldots, \alpha_n \in \mathbb{N}$$

$$A = \langle \alpha_1, \ldots, \alpha_n \rangle \to \langle \alpha_1 + \delta_{j,1}, \ldots, \alpha_n + \delta_{j,n} \rangle = A + \Delta_j$$

Системы векторного сложения

$$\begin{aligned} \Delta_1 &= \langle \delta_{1,1}, \ldots, \delta_{1,n} \rangle \\ &\vdots \\ \Delta_k &= \langle \delta_{k,1}, \ldots, \delta_{k,n} \rangle \end{aligned} \qquad \delta_{j,k} \in \mathbb{Z}$$

$$\alpha_1 \in \mathbb{N}, \ldots, \alpha_n \in \mathbb{N}$$

$$A = \langle \alpha_1, \ldots, \alpha_n \rangle \to \langle \alpha_1 + \delta_{j,1}, \ldots, \alpha_n + \delta_{j,n} \rangle = A + \Delta_j$$

$$\alpha_1 + \delta_{j,1} \in \mathbb{N}, \ldots, \alpha_n + \delta_{j,n} \in \mathbb{N}$$

Системы векторного сложения

$$\Delta_1 = \langle \delta_{1,1}, \ldots, \delta_{1,n} \rangle$$
$$\vdots \qquad\qquad \delta_{j,k} \in \mathbb{Z}$$
$$\Delta_k = \langle \delta_{k,1}, \ldots, \delta_{k,n} \rangle$$

$$\alpha_1 \in \mathbb{N}, \ldots, \alpha_n \in \mathbb{N}$$

$$A = \langle \alpha_1, \ldots, \alpha_n \rangle \to \langle \alpha_1 + \delta_{j,1}, \ldots, \alpha_n + \delta_{j,n} \rangle = A + \Delta_j$$

$$\alpha_1 + \delta_{j,1} \in \mathbb{N}, \ldots, \alpha_n + \delta_{j,n} \in \mathbb{N}$$

$A \to A + \Delta_{j_1}$

Системы векторного сложения

$$\Delta_1 = \langle \delta_{1,1}, \ldots, \delta_{1,n} \rangle$$

$$\vdots \qquad\qquad \delta_{j,k} \in \mathbb{Z}$$

$$\Delta_k = \langle \delta_{k,1}, \ldots, \delta_{k,n} \rangle$$

$$\alpha_1 \in \mathbb{N}, \ldots, \alpha_n \in \mathbb{N}$$

$$A = \langle \alpha_1, \ldots, \alpha_n \rangle \to \langle \alpha_1 + \delta_{j,1}, \ldots, \alpha_n + \delta_{j,n} \rangle = A + \Delta_j$$

$$\alpha_1 + \delta_{j,1} \in \mathbb{N}, \ldots, \alpha_n + \delta_{j,n} \in \mathbb{N}$$

$$A \to A + \Delta_{j_1} \to A + \Delta_{j_1} + \Delta_{j_2}$$

Системы векторного сложения

$$\begin{aligned} \Delta_1 &= \langle \delta_{1,1}, \ldots, \delta_{1,n} \rangle \\ &\vdots \\ \Delta_k &= \langle \delta_{k,1}, \ldots, \delta_{k,n} \rangle \end{aligned} \qquad \delta_{j,k} \in \mathbb{Z}$$

$$\alpha_1 \in \mathbb{N}, \ldots, \alpha_n \in \mathbb{N}$$

$$A = \langle \alpha_1, \ldots, \alpha_n \rangle \to \langle \alpha_1 + \delta_{j,1}, \ldots, \alpha_n + \delta_{j,n} \rangle = A + \Delta_j$$

$$\alpha_1 + \delta_{j,1} \in \mathbb{N}, \ldots, \alpha_n + \delta_{j,n} \in \mathbb{N}$$

$$A \to A + \Delta_{j_1} \to A + \Delta_{j_1} + \Delta_{j_2} \to A + \Delta_{j_1} + \Delta_{j_2} + \Delta_{j_3}$$

Системы векторного сложения

$$\Delta_1 = \langle \delta_{1,1}, \ldots, \delta_{1,n} \rangle$$
$$\vdots$$
$$\Delta_k = \langle \delta_{k,1}, \ldots, \delta_{k,n} \rangle$$

$$\delta_{j,k} \in \mathbb{Z}$$

$$\alpha_1 \in \mathbb{N}, \ldots, \alpha_n \in \mathbb{N}$$

$$A = \langle \alpha_1, \ldots, \alpha_n \rangle \to \langle \alpha_1 + \delta_{j,1}, \ldots, \alpha_n + \delta_{j,n} \rangle = A + \Delta_j$$

$$\alpha_1 + \delta_{j,1} \in \mathbb{N}, \ldots, \alpha_n + \delta_{j,n} \in \mathbb{N}$$

$$A \to A + \Delta_{j_1} \to A + \Delta_{j_1} + \Delta_{j_2} \to A + \Delta_{j_1} + \Delta_{j_2} + \Delta_{j_3} \to \ldots$$

Проблема достижимости

Проблема достижимости

ВХОД: Система векторного сложения $\{\Delta_1, \ldots, \Delta_k\}$ и два вектора A и B

Проблема достижимости

ВХОД: Система векторного сложения $\{\Delta_1, \ldots, \Delta_k\}$ и два вектора A и B

ВОПРОС: Верно ли, что вектор B достижим из вектора A в этой системе?

Проблема включения

Проблема включения

ВХОД: Две системы векторного сложения $\{\Gamma_1, \ldots, \Gamma_k\}$ и $\{\Delta_1, \ldots, \Delta_k\}$ и вектор A

Проблема включения

ВХОД: Две системы векторного сложения $\{\Gamma_1, \ldots, \Gamma_k\}$ и $\{\Delta_1, \ldots, \Delta_k\}$ и вектор A

ВОПРОС: Верно ли, что каждый вектор, достижимый из вектора A во второй системе, достижим из вектора A также и в первой системе?

Проблема включения

ВХОД: Две системы векторного сложения $\{\Gamma_1, \ldots, \Gamma_k\}$ и $\{\Delta_1, \ldots, \Delta_k\}$ и вектор A

ВОПРОС: Верно ли, что каждый вектор, достижимый из вектора A во второй системе, достижим из вектора A также и в первой системе?

Теорема (Michael Rabin, не опубликовано). *Проблема включения для систем векторного сложения неразрешима.*

Проблема включения

ВХОД: Две системы векторного сложения $\{\Gamma_1, \ldots, \Gamma_k\}$ и $\{\Delta_1, \ldots, \Delta_k\}$ и вектор A

ВОПРОС: Верно ли, что каждый вектор, достижимый из вектора A во второй системе, достижим из вектора A также и в первой системе?

Теорема (Michael Rabin, не опубликовано). *Проблема включения для систем векторного сложения неразрешима.*

Проблема эквивалентности

ВХОД: Две системы векторного сложения $\{\Gamma_1, \ldots, \Gamma_k\}$ и $\{\Delta_1, \ldots, \Delta_k\}$ и вектор A

ВОПРОС: Верно ли, что каждый вектор, достижимый из вектора A в одной из этих систем, достижим из вектора A также и в другой системе?

Проблема эквивалентности

ВХОД: Две системы векторного сложения $\{\Gamma_1, \ldots, \Gamma_k\}$ и $\{\Delta_1, \ldots, \Delta_k\}$ и вектор A

ВОПРОС: Верно ли, что каждый вектор, достижимый из вектора A в одной из этих систем, достижим из вектора A также и в другой системе?

Теорема (M. Hack; T. Araki и T. Kasami). *Проблема эквивалентности для систем векторного сложения неразрешима.*

Обобщенные кони на многомерной шахматной доске

$$\Delta_1 = \langle \delta_{1,1}, \dots, \delta_{1,n} \rangle$$

$$\vdots$$

$$\Delta_k = \langle \delta_{k,1}, \dots, \delta_{k,n} \rangle$$

$$\delta_{j,k} \in \mathbb{Z}$$

Обобщенные кони на многомерной шахматной доске

$$\Delta_1 = \langle \delta_{1,1}, \dots, \delta_{1,n} \rangle$$
$$\vdots \qquad\qquad \delta_{j,k} \in \mathbb{Z}$$
$$\Delta_k = \langle \delta_{k,1}, \dots, \delta_{k,n} \rangle$$

Обозначение. $\mathfrak{K}(\alpha_1, \dots, \alpha_n)$ – это множество всех полей, на которых может побывать конь, начав свой путь с поля $\langle \alpha_1, \dots, \alpha_n \rangle$

Обобщенные кони на многомерной шахматной доске

$$\begin{aligned} \Delta_1 &= \langle \delta_{1,1}, \ldots, \delta_{1,n} \rangle \\ &\vdots \\ \Delta_k &= \langle \delta_{k,1}, \ldots, \delta_{k,n} \rangle \end{aligned} \qquad \delta_{j,k} \in \mathbb{Z}$$

Обозначение. $\mathfrak{K}(\alpha_1, \ldots, \alpha_n)$ – это множество всех полей, на которых может побывать конь, начав свой путь с поля $\langle \alpha_1, \ldots, \alpha_n \rangle$

Проблема включения

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, $\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A})$

Обобщенные кони на многомерной шахматной доске

$$\begin{aligned}\Delta_1 &= \langle \delta_{1,1}, \ldots, \delta_{1,n} \rangle \\ &\vdots \\ \Delta_k &= \langle \delta_{k,1}, \ldots, \delta_{k,n} \rangle\end{aligned} \qquad \delta_{j,k} \in \mathbb{Z}$$

Обозначение. $\mathfrak{K}(\alpha_1, \ldots, \alpha_n)$ – это множество всех полей, на которых может побывать конь, начав свой путь с поля $\langle \alpha_1, \ldots, \alpha_n \rangle$

Проблема включения

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, $\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A})$

Проблема эквивалентности

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, $\mathfrak{K}'(\mathcal{A}) = \mathfrak{K}''(\mathcal{A})$

Сравнение проблем

Обозначение. $\mathfrak{K}(\alpha_1, \ldots, \alpha_n)$ – это множество всех полей, на которых может побывать конь, начав свой путь с поля $\langle \alpha_1, \ldots, \alpha_n \rangle$

Сравнение проблем

Обозначение. $\mathfrak{K}(\alpha_1, \ldots, \alpha_n)$ – это множество всех полей, на которых может побывать конь, начав свой путь с поля $\langle \alpha_1, \ldots, \alpha_n \rangle$

Проблема включения

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, что $\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A})$

Сравнение проблем

Обозначение. $\mathfrak{K}(\alpha_1, \ldots, \alpha_n)$ – это множество всех полей, на которых может побывать конь, начав свой путь с поля $\langle \alpha_1, \ldots, \alpha_n \rangle$

Проблема включения

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, что $\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A})$

Проблема эквивалентности

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, что $\mathfrak{K}'(\mathcal{A}) = \mathfrak{K}''(\mathcal{A})$

Сравнение проблем

Обозначение. $\mathfrak{K}(\alpha_1, \ldots, \alpha_n)$ – это множество всех полей, на которых может побывать конь, начав свой путь с поля $\langle \alpha_1, \ldots, \alpha_n \rangle$

Проблема включения

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, что $\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A})$

Проблема эквивалентности

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, что $\mathfrak{K}'(\mathcal{A}) = \mathfrak{K}''(\mathcal{A})$

$$\mathfrak{K}'(\mathcal{A}) = \mathfrak{K}''(\mathcal{A}) \iff \mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A}) \& \mathfrak{K}''(\mathcal{A}) \supseteq \mathfrak{K}'(\mathcal{A})$$

Специальные кони

$$\langle \delta_1, \ldots, \delta_n \rangle \qquad\qquad \delta_m \in \{-1, 0, 1\}$$

Специальные кони

$$\langle \delta_1, \ldots, \delta_n \rangle \qquad \delta_m \in \{-1, 0, 1\}$$

$$\delta_{i_1} = \cdots = \delta_{i_a} = -1$$

Специальные кони

$$\langle \delta_1, \ldots, \delta_n \rangle \qquad \delta_m \in \{-1, 0, 1\}$$

$$\delta_{i_1} = \cdots = \delta_{i_a} = -1$$
$$\delta_{j_1} = \cdots = \delta_{j_b} = 1$$

Специальные кони

$$\langle \delta_1, \dots, \delta_n \rangle \qquad\qquad \delta_m \in \{-1, 0, 1\}$$

$$\delta_{i_1} = \cdots = \delta_{i_a} = -1$$
$$\delta_{j_1} = \cdots = \delta_{j_b} = 1$$
$$\delta_m = 0, \text{ если } \ m \notin \{i_1, \dots, i_a, j_1, \dots, j_b\}$$

Специальные кони

$$\langle \delta_1, \ldots, \delta_n \rangle \qquad\qquad \delta_m \in \{-1, 0, 1\}$$

$$\begin{aligned}
&\delta_{i_1} = \cdots = \delta_{i_a} = -1 \\
&\delta_{j_1} = \cdots = \delta_{j_b} = 1 \\
&\delta_m = 0, \text{ если } \; m \notin \{i_1, \ldots, i_a, j_1, \ldots, j_b\}
\end{aligned}$$

$$\langle i_1, \ldots, i_a \rangle \rightarrowtail \langle j_1, \ldots, j_b \rangle$$

Специальные кони

$$\langle \delta_1, \ldots, \delta_n \rangle \qquad \delta_m \in \{-1, 0, 1\}$$

$$\begin{aligned}
&\delta_{i_1} = \cdots = \delta_{i_a} = -1 \\
&\delta_{j_1} = \cdots = \delta_{j_b} = 1 \\
&\delta_m = 0, \text{ если } \ m \notin \{i_1, \ldots, i_a, j_1, \ldots, j_b\}
\end{aligned}$$

$$\langle i_1, \ldots, i_a \rangle \rightarrowtail \langle j_1, \ldots, j_b \rangle$$

$$R_{i_1} \ldots R_{i_a} \rightarrowtail R_{j_1} \ldots R_{j_b}$$

Шахматная машина

Шахматная машина

Шахматная машина имеет конечное количество *регистров* $R_1, \ldots, R_n$, каждый из которых может содержать произвольно большое натуральное число.

Шахматная машина

Шахматная машина имеет конечное количество *регистров* $R_1, \ldots, R_n$, каждый из которых может содержать произвольно большое натуральное число. Машина может находиться в одном из конечного числа состояний $S_1, \ldots, S_m$;

Шахматная машина

Шахматная машина имеет конечное количество *регистров* $R_1, \ldots, R_n$, каждый из которых может содержать произвольно большое натуральное число. Машина может находиться в одном из конечного числа состояний $S_1, \ldots, S_m$; *инструкции* машины имеют вид

$$S_{i_0} R_{i_1} \ldots R_{i_a} \rightarrowtail S_{j_0} R_{j_1} \ldots R_{j_b}$$

где все числа $i_0, i_1, \ldots, i_a, j_0, j_1, \ldots, j_b$ попарно различны.

Шахматная машина

Шахматная машина имеет конечное количество *регистров* $R_1, \ldots, R_n$, каждый из которых может содержать произвольно большое натуральное число. Машина может находиться в одном из конечного числа состояний $S_1, \ldots, S_m$; *инструкции* машины имеют вид

$$S_{i_0} R_{i_1} \ldots R_{i_a} \rightarrowtail S_{j_0} R_{j_1} \ldots R_{j_b}$$

где все числа $i_0, i_1, \ldots, i_a, j_0, j_1, \ldots, j_b$ попарно различны.

Шахматная машина является недетерминированной!

Вычисления на шахматной машине

Определение. Пусть в некоторой шахматной машине выделено начальное состояние S_b

Вычисления на шахматной машине

Определение. Пусть в некоторой шахматной машине выделено начальное состояние S_b , заключительное состояние S_e

Вычисления на шахматной машине

Определение. Пусть в некоторой шахматной машине выделено начальное состояние S_b , заключительное состояние S_e , входные регистры $I_{i_1}, \ldots, I_{i_k}$ и выходной регистр O_j.

Вычисления на шахматной машине

Определение. Пусть в некоторой шахматной машине выделено начальное состояние S_b , заключительное состояние S_e , входные регистры $I_{i_1}, \ldots, I_{i_k}$ и выходной регистр O_j. Поместим коня на поле

$$\langle r_1, \ldots, r_n \rangle \qquad (*)$$

где $r_b = 1$, $r_{i_1} = x_1, \ldots$, $r_{i_k} = x_k$, а все остальные регистры пусты.

Вычисления на шахматной машине

Определение. Пусть в некоторой шахматной машине выделено начальное состояние S_b , заключительное состояние S_e , входные регистры $I_{i_1}, \ldots, I_{i_k}$ и выходной регистр O_j. Поместим коня на поле

$$\langle r_1, \ldots, r_n \rangle \qquad (*)$$

где $r_b = 1$, $r_{i_1} = x_1$,..., $r_{i_k} = x_k$, а все остальные регистры пусты. Мы говорим, что шахматная машина вычисляет функцию $F(x_1, \ldots, x_k)$, если выполены следующие три условия.

Вычисления на шахматной машине

Определение. Пусть в некоторой шахматной машине выделено начальное состояние S_b , заключительное состояние S_e , входные регистры $I_{i_1}, \ldots, I_{i_k}$ и выходной регистр O_j. Поместим коня на поле

$$\langle r_1, \ldots, r_n \rangle \qquad (*)$$

где $r_b = 1$, $r_{i_1} = x_1, \ldots, r_{i_k} = x_k$, а все остальные регистры пусты. Мы говорим, что шахматная машина вычисляет функцию $F(x_1, \ldots, x_k)$, если выполены следующие три условия.

1. Если поле $\langle r'_1, \ldots, r'_n \rangle$ достижимо с поля $(*)$, то $r'_j \leq F(x_1, \ldots, x_k)$

Вычисления на шахматной машине

Определение. Пусть в некоторой шахматной машине выделено начальное состояние S_b , заключительное состояние S_e , входные регистры $I_{i_1}, \ldots, I_{i_k}$ и выходной регистр O_j. Поместим коня на поле

$$\langle r_1, \ldots, r_n \rangle \tag{$*$}$$

где $r_b = 1$, $r_{i_1} = x_1, \ldots$, $r_{i_k} = x_k$, а все остальные регистры пусты. Мы говорим, что шахматная машина вычисляет функцию $F(x_1, \ldots, x_k)$, если выполены следующие три условия.

1. Если поле $\langle r'_1, \ldots, r'_n \rangle$ достижимо с поля $(*)$, то $r'_j \leq F(x_1, \ldots, x_k)$
2. Для любого y такого, что $y \leq F(x_1, \ldots, x_k)$, существует поле $\langle r'_1, \ldots, r'_n \rangle$, достижимое с поля $(*)$ и такое, что $r_j = y$, $r_e = 1$.

Вычисления на шахматной машине

Определение. Пусть в некоторой шахматной машине выделено начальное состояние S_b , заключительное состояние S_e , входные регистры $I_{i_1}, \ldots, I_{i_k}$ и выходной регистр O_j. Поместим коня на поле

$$\langle r_1, \ldots, r_n \rangle \tag{$*$}$$

где $r_b = 1$, $r_{i_1} = x_1, \ldots,$ $r_{i_k} = x_k$, а все остальные регистры пусты. Мы говорим, что шахматная машина вычисляет функцию $F(x_1, \ldots, x_k)$, если выполены следующие три условия.

1. Если поле $\langle r_1', \ldots, r_n' \rangle$ достижимо с поля $(*)$, то $r_j' \leq F(x_1, \ldots, x_k)$
2. Для любого y такого, что $y \leq F(x_1, \ldots, x_k)$, существует поле $\langle r_1', \ldots, r_n' \rangle$, достижимое с поля $(*)$ и такое, что $r_j = y$, $r_e = 1$.
3. Состояние S_e не встречается в левых частях инструкций машины.

Сложение чисел

$$
\begin{aligned}
S_1 I_4 &\rightarrowtail S_2 O_6 \\
S_1 I_5 &\rightarrowtail S_2 O_6 \\
S_1 &\rightarrowtail S_3 \\
S_2 &\rightarrowtail S_1
\end{aligned}
$$

Умножение чисел

$$
\begin{aligned}
S_1 I_6 &\rightarrowtail S_2 \\
S_2 I_7 &\rightarrowtail S_3 R_8 O_9 \\
S_3 &\rightarrowtail S_2 \\
S_3 &\rightarrowtail S_4 \\
S_4 R_8 &\rightarrowtail S_1 I_7 \\
S_1 &\rightarrowtail S_4 \\
S_1 &\rightarrowtail S_5
\end{aligned}
$$

Сложение функций

Лемма. Имея две шахматные машины $\mathfrak{K}_1$ и $\mathfrak{K}_2$, вычисляющие функции $F_1(x_1, \ldots, x_{m_1})$ и $F_2(y_1, \ldots, y_{m_2})$ соответственно, мы можем построить машину $\mathfrak{K}$, вычисляющую функцию

$$F(z_1, \ldots, z_{m_1+m_2}) = F_1(z_1, \ldots, z_{m_1}) + F_2(z_{m_1+1}, \ldots, z_{m_1+m_2}).$$

Сложение функций

Лемма. Имея две шахматные машины $\mathfrak{K}_1$ и $\mathfrak{K}_2$, вычисляющие функции $F_1(x_1, \ldots, x_{m_1})$ и $F_2(y_1, \ldots, y_{m_2})$ соответственно, мы можем построить машину $\mathfrak{K}$, вычисляющую функцию

$$F(z_1, \ldots, z_{m_1+m_2}) = F_1(z_1, \ldots, z_{m_1}) + F_2(z_{m_1+1}, \ldots, z_{m_1+m_2}).$$

$$\begin{aligned} S_1 I_4 &\rightarrowtail S_2 O_6 \\ S_1 I_5 &\rightarrowtail S_2 O_6 \\ S_1 &\rightarrowtail S_3 \\ S_2 &\rightarrowtail S_1 \end{aligned}$$

Умножение функций

Лемма. Имея две шахматные машины $\mathfrak{K}_1$ и $\mathfrak{K}_2$, вычисляющие функции $F_1(x_1, \ldots, x_{m_1})$ и $F_2(y_1, \ldots, y_{m_2})$ соответственно, мы можем построить машину $\mathfrak{K}$, вычисляющую функцию

$$F(z_1, \ldots, z_{m_1+m_2}) = F_1(z_1, \ldots, z_{m_1}) \times F_2(z_{m_1+1}, \ldots, z_{m_1+m_2}).$$

Умножение функций

Лемма. Имея две шахматные машины $\mathfrak{K}_1$ и $\mathfrak{K}_2$, вычисляющие функции $F_1(x_1, \ldots, x_{m_1})$ и $F_2(y_1, \ldots, y_{m_2})$ соответственно, мы можем построить машину $\mathfrak{K}$, вычисляющую функцию

$$F(z_1, \ldots, z_{m_1+m_2}) = F_1(z_1, \ldots, z_{m_1}) \times F_2(z_{m_1+1}, \ldots, z_{m_1+m_2}).$$

$$\begin{aligned}
S_1 I_6 &\rightarrowtail S_2 \\
S_2 I_7 &\rightarrowtail S_3 R_8 O_9 \\
S_3 &\rightarrowtail S_2 \\
S_3 &\rightarrowtail S_4 \\
S_4 R_8 &\rightarrowtail S_1 I_7 \\
S_1 &\rightarrowtail S_4 \\
S_1 &\rightarrowtail S_5
\end{aligned}$$

Альтернативы

- либо
$$\exists x_1 \dots x_m \{D(x_1, \dots, x_m) = 0\}$$
- либо
$$\forall x_1 \dots x_m \{D(x_1, \dots, x_m) \neq 0\}$$

Альтернативы

- либо
$$\exists x_1 \ldots x_m \{D(x_1, \ldots, x_m) = 0\}$$
- либо
$$\forall x_1 \ldots x_m \{D(x_1, \ldots, x_m) \neq 0\}$$

$$D^2(x_1, \ldots, x_m) = A(x_1, \ldots, x_m) - B(x_1, \ldots, x_m)$$

Альтернативы

- либо
$$\exists x_1 \ldots x_m \{D(x_1, \ldots, x_m) = 0\}$$
- либо
$$\forall x_1 \ldots x_m \{D(x_1, \ldots, x_m) \neq 0\}$$

$$D^2(x_1, \ldots, x_m) = A(x_1, \ldots, x_m) - B(x_1, \ldots, x_m)$$

$$A(x_1, \ldots, x_m) \geq B(x_1, \ldots, x_m)$$

Альтернативы

- либо
$$\exists x_1 \dots x_m \{D(x_1, \dots, x_m) = 0\}$$
- либо
$$\forall x_1 \dots x_m \{D(x_1, \dots, x_m) \neq 0\}$$

$$D^2(x_1, \dots, x_m) = A(x_1, \dots, x_m) - B(x_1, \dots, x_m)$$

$$A(x_1, \dots, x_m) \geq B(x_1, \dots, x_m)$$

- либо
$$\exists x_1 \dots x_m \{A(x_1, \dots, x_m) = B(x_1, \dots, x_m)\}$$
- либо
$$\forall x_1 \dots x_m \{A(x_1, \dots, x_m) \geq B(x_1, \dots, x_m) + 1\}$$

Альтернативы

- либо
$$\exists x_1 \ldots x_m \{D(x_1, \ldots, x_m) = 0\}$$
- либо
$$\forall x_1 \ldots x_m \{D(x_1, \ldots, x_m) \neq 0\}$$

$$D^2(x_1, \ldots, x_m) = A(x_1, \ldots, x_m) - B(x_1, \ldots, x_m)$$

$$A(x_1, \ldots, x_m) \geq B(x_1, \ldots, x_m)$$

- либо
$$\exists x_1 \ldots x_m \{A(x_1, \ldots, x_m) = B(x_1, \ldots, x_m)\}$$
- либо
$$\forall x_1 \ldots x_m \{A(x_1, \ldots, x_m) \geq B(x_1, \ldots, x_m) + 1\}$$

Машина $\mathfrak{K}'_A$

$$A(x_1, \ldots, x_m)$$

Машина $\mathfrak{K}'_A$

$$A(x_1, \ldots, x_m) \quad = A'(\underbrace{x_1, \ldots, x_1}_{w \text{ times}}, \ldots, \underbrace{x_m, \ldots, x_m}_{w \text{ times}}),$$

Машина $\mathfrak{K}'_A$

$$A(x_1, \ldots, x_m) \quad = A'(\underbrace{x_1, \ldots, x_1}_{w \text{ times}}, \ldots, \underbrace{x_m, \ldots, x_m}_{w \text{ times}}),$$

Построим шахматную машину, вычисляющую A'.

Машина $\mathfrak{K}'_A$

$$A(x_1, \ldots, x_m) \quad = A'(\underbrace{x_1, \ldots, x_1}_{w \text{ times}}, \ldots, \underbrace{x_m, \ldots, x_m}_{w \text{ times}}),$$

Построим шахматную машину, вычисляющую A'. Пусть $R_{i_{1,1}}$, …, $R_{i_{m,w}}$ – её входные регистры

Машина $\mathfrak{K}'_A$

$$A(x_1, \ldots, x_m) \quad = A'(\underbrace{x_1, \ldots, x_1}_{w \text{ times}}, \ldots, \underbrace{x_m, \ldots, x_m}_{w \text{ times}}),$$

Построим шахматную машину, вычисляющую A'. Пусть $R_{i_{1,1}}$, ..., $R_{i_{m,w}}$ – её входные регистры, B_{m+1} – выходной регистр

Машина $\mathfrak{K}'_A$

$$A(x_1, \ldots, x_m) \quad = A'(\underbrace{x_1, \ldots, x_1}_{w \text{ times}}, \ldots, \underbrace{x_m, \ldots, x_m}_{w \text{ times}}),$$

Построим шахматную машину, вычисляющую A'. Пусть $R_{i_{1,1}}$, ..., $R_{i_{m,w}}$ – её входные регистры, B_{m+1} – выходной регистр, S_b – начальное состояние

Машина $\mathfrak{K}'_A$

$$A(x_1,\dots,x_m) \quad = A'(\underbrace{x_1,\dots,x_1}_{w\ \text{times}},\dots,\underbrace{x_m,\dots,x_m}_{w\ \text{times}}),$$

Построим шахматную машину, вычисляющую A'. Пусть $R_{i_{1,1}}$, ..., $R_{i_{m,w}}$ – её входные регистры, B_{m+1} – выходной регистр, S_b – начальное состояние, а номера всех остальных регистров и состояний превосходят $m+9$.

Машина $\mathfrak{K}'_A$

$$A(x_1, \dots, x_m) \quad = A'(\underbrace{x_1, \dots, x_1}_{w \text{ times}}, \dots, \underbrace{x_m, \dots, x_m}_{w \text{ times}}),$$

Построим шахматную машину, вычисляющую A'. Пусть $R_{i_{1,1}}, \dots, R_{i_{m,w}}$ – её входные регистры, B_{m+1} – выходной регистр, S_b – начальное состояние, а номера всех остальных регистров и состояний превосходят $m+9$. Добавим следующие инструкции:

$$\begin{aligned}
S_{m+2} &\rightarrowtail S_{m+3} R_{i_{1,1}} \dots R_{i_{1,w}} X_1 \\
\dots &\rightarrowtail \dots \\
S_{m+2} &\rightarrowtail S_{m+3} R_{i_{m,1}} \dots R_{i_{m,w}} X_m \\
S_{m+3} &\rightarrowtail S_{m+2} \\
S_{m+3} &\rightarrowtail S_b,
\end{aligned}$$

Обозначим полученную машину через $\mathfrak{K}'_A$

Машина $\mathfrak{K}'_A$

$$A(x_1, \dots, x_m) \quad = A'(\underbrace{x_1, \dots, x_1}_{w \text{ times}}, \dots, \underbrace{x_m, \dots, x_m}_{w \text{ times}}),$$

Построим шахматную машину, вычисляющую A'. Пусть $R_{i_{1,1}}$, ..., $R_{i_{m,w}}$ – её входные регистры, B_{m+1} – выходной регистр, S_b – начальное состояние, а номера всех остальных регистров и состояний превосходят $m+9$. Добавим следующие инструкции:

$$\begin{aligned}
S_{m+2} &\rightarrowtail S_{m+3} R_{i_{1,1}} \dots R_{i_{1,w}} X_1 \\
\dots &\rightarrowtail \dots \\
S_{m+2} &\rightarrowtail S_{m+3} R_{i_{m,1}} \dots R_{i_{m,w}} X_m \\
S_{m+3} &\rightarrowtail S_{m+2} \\
S_{m+3} &\rightarrowtail S_b,
\end{aligned}$$

Обозначим полученную машину через $\mathfrak{K}'_A$, её начальным состоянием объявим S_{m+2}.

Машина $\mathfrak{K}''_B$

$$B(x_1,\dots,x_m)+1 = B''(\underbrace{x_1,\dots,x_1}_{w \text{ times}},\dots,\underbrace{x_m,\dots,x_m}_{w \text{ times}}).$$

Построим шахматную машину, вычисляющую B'', Пусть $R_{i_{1,1}}, \dots, R_{i_{m,w}}$ – её входные регистры, B_{m+1} – выходной регистр, S_b – начальное состояние, а номера всех остальных регистров и состояний превосходят $m+9$. Добавим следующие инструкции:

$$\begin{aligned}
S_{m+2} &\rightarrowtail S_{m+3}R_{i_{1,1}}\dots R_{i_{1,w}}X_1\\
\dots &\rightarrowtail \dots\\
S_{m+2} &\rightarrowtail S_{m+3}R_{i_{m,1}}\dots R_{i_{m,w}}X_m\\
S_{m+3} &\rightarrowtail S_{m+2}\\
S_{m+3} &\rightarrowtail S_b,
\end{aligned}$$

Обозначим полученную машину через $\mathfrak{K}''_B$, её начальным состоянием объявим S_{m+2}.

Машина $\mathfrak{K}'$

Добавим к машине $\mathfrak{K}'_A$ следующие инструкции

Машина $\mathfrak{K}'$

Добавим к машине ${\mathfrak{K}'}_A$ следующие инструкции

1. для каждого регистра R_i машины $\mathfrak{K}'_A$ кроме $X_1, \ldots, X_{m+1}$:

$$S_s R_i \rightarrowtail S_t$$

Машина $\mathfrak{K}'$

Добавим к машине $\mathfrak{K}'{}_A$ следующие инструкции

1. для каждого регистра R_i машины $\mathfrak{K}'_A$ кроме $X_1, \ldots, X_{m+1}$:

$$S_s R_i \rightarrowtail S_t$$

2. для каждого регистра R_j машины $\mathfrak{K}_B$ кроме $X_1, \ldots, X_{m+1}$:

$$S_s \rightarrowtail S_t R_j$$

Машина $\mathfrak{K}'$

Добавим к машине $\mathfrak{K}'{}_A$ следующие инструкции

1. для каждого регистра R_i машины $\mathfrak{K}'_A$ кроме $X_1, \ldots, X_{m+1}$:

$$S_s R_i \rightarrowtail S_t$$

2. для каждого регистра R_j машины $\mathfrak{K}_B$ кроме $X_1, \ldots, X_{m+1}$:

$$S_s \rightarrowtail S_t R_j$$

3. для каждого состояния S_k машины $\mathfrak{K}_B$:

$$S_s \rightarrowtail S_k$$

Машина $\mathfrak{K}'$

Добавим к машине $\mathfrak{K}'_A$ следующие инструкции

1. для каждого регистра R_i машины $\mathfrak{K}'_A$ кроме $X_1, \ldots, X_{m+1}$:
$$S_s R_i \rightarrowtail S_t$$
2. для каждого регистра R_j машины $\mathfrak{K}_B$ кроме $X_1, \ldots, X_{m+1}$:
$$S_s \rightarrowtail S_t R_j$$
3. для каждого состояния S_k машины $\mathfrak{K}_B$:
$$S_s \rightarrowtail S_k$$
4. $$S_t \rightarrowtail S_s$$

Машина $\mathfrak{K}''$

Машина $\mathfrak{K}''$ имеет те же инструкции, что и машина $\mathfrak{K}''_B$, но к ней добавлены все регистры машины $\mathfrak{K}'_A$.

Машина $\mathfrak{K}''$

Машина $\mathfrak{K}''$ имеет те же инструкции, что и машина $\mathfrak{K}''_B$, но к ней добавлены все регистры машины $\mathfrak{K}'_A$.

$$\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A}) \quad \Longleftrightarrow \quad \neg\exists x_1 \dots x_m \{D(x_1, \dots, x_m) = 0\}$$

Машина $\mathfrak{K}''$

Машина $\mathfrak{K}''$ имеет те же инструкции, что и машина $\mathfrak{K}''_B$, но к ней добавлены все регистры машины $\mathfrak{K}'_A$.

$$\begin{aligned}
\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A}) \quad &\Longleftrightarrow \quad \neg\exists x_1 \dots x_m \{D(x_1, \dots, x_m) = 0\} \\
&\Longleftrightarrow \quad \forall x_1 \dots x_m \{D(x_1, \dots, x_m) \neq 0\}
\end{aligned}$$

Машина $\mathfrak{K}''$

Машина $\mathfrak{K}''$ имеет те же инструкции, что и машина $\mathfrak{K}''_B$, но к ней добавлены все регистры машины $\mathfrak{K}'_A$.

$$\begin{aligned}
\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A}) \quad &\Longleftrightarrow \quad \neg\exists x_1 \dots x_m \{D(x_1, \dots, x_m) = 0\} \\
&\Longleftrightarrow \quad \forall x_1 \dots x_m \{D(x_1, \dots, x_m) \neq 0\}
\end{aligned}$$

$$\mathfrak{K}'(\mathcal{A}) \not\supseteq \mathfrak{K}''(\mathcal{A}) \quad \Longleftrightarrow \quad \exists x_1 \dots x_m \{D(x_1, \dots, x_m) = 0\}$$

Машина $\mathfrak{K}''$

Машина $\mathfrak{K}''$ имеет те же инструкции, что и машина $\mathfrak{K}''_B$, но к ней добавлены все регистры машины $\mathfrak{K}'_A$.

$$\begin{aligned}
\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A}) \quad &\Longleftrightarrow \quad \neg\exists x_1 \dots x_m \{D(x_1, \dots, x_m) = 0\} \\
&\Longleftrightarrow \quad \forall x_1 \dots x_m \{D(x_1, \dots, x_m) \neq 0\}
\end{aligned}$$

$$\begin{aligned}
\mathfrak{K}'(\mathcal{A}) \not\supseteq \mathfrak{K}''(\mathcal{A}) \quad &\Longleftrightarrow \quad \exists x_1 \dots x_m \{D(x_1, \dots, x_m) = 0\} \\
&\Longleftrightarrow \quad \neg\forall x_1 \dots x_m \{D(x_1, \dots, x_m) \neq 0\}
\end{aligned}$$

Машина $\mathfrak{K}''$

Машина $\mathfrak{K}''$ имеет те же инструкции, что и машина $\mathfrak{K}''_B$, но к ней добавлены все регистры машины $\mathfrak{K}'_A$.

$$\begin{aligned}
\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A}) \quad &\Longleftrightarrow \quad \neg\exists x_1 \ldots x_m \{D(x_1, \ldots, x_m) = 0\} \\
&\Longleftrightarrow \quad \forall x_1 \ldots x_m \{D(x_1, \ldots, x_m) \neq 0\}
\end{aligned}$$

$$\begin{aligned}
\mathfrak{K}'(\mathcal{A}) \not\supseteq \mathfrak{K}''(\mathcal{A}) \quad &\Longleftrightarrow \quad \exists x_1 \ldots x_m \{D(x_1, \ldots, x_m) = 0\} \\
&\Longleftrightarrow \quad \neg\forall x_1 \ldots x_m \{D(x_1, \ldots, x_m) \neq 0\}
\end{aligned}$$

$$\mathfrak{K}'(\mathcal{A}) \not\supseteq \mathfrak{K}''(\mathcal{A}) \quad \Leftarrow \quad \exists x_1 \ldots x_m \{D(x_1, \ldots, x_m) = 0\}$$

Машина $\mathfrak{K}''$

Машина $\mathfrak{K}''$ имеет те же инструкции, что и машина $\mathfrak{K}''_B$, но к ней добавлены все регистры машины $\mathfrak{K}'_A$.

$$\begin{aligned}\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A}) \quad &\Longleftrightarrow \quad \neg\exists x_1 \ldots x_m\{D(x_1, \ldots, x_m) = 0\} \\ &\Longleftrightarrow \quad \forall x_1 \ldots x_m\{D(x_1, \ldots, x_m) \neq 0\}\end{aligned}$$

$$\begin{aligned}\mathfrak{K}'(\mathcal{A}) \not\supseteq \mathfrak{K}''(\mathcal{A}) \quad &\Longleftrightarrow \quad \exists x_1 \ldots x_m\{D(x_1, \ldots, x_m) = 0\} \\ &\Longleftrightarrow \quad \neg\forall x_1 \ldots x_m\{D(x_1, \ldots, x_m) \neq 0\}\end{aligned}$$

$$\mathfrak{K}'(\mathcal{A}) \not\supseteq \mathfrak{K}''(\mathcal{A}) \quad \Leftarrow \quad \exists x_1 \ldots x_m\{D(x_1, \ldots, x_m) = 0\}$$

$$\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A}) \quad \Leftarrow \quad \forall x_1 \ldots x_m\{D(x_1, \ldots, x_m) \neq 0\}$$

Включение и эквивалентность

Проблема включения

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, что $\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A})$

Проблема эквивалентности

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, что $\mathfrak{K}'(\mathcal{A}) = \mathfrak{K}''(\mathcal{A})$

Включение и эквивалентность

Проблема включения

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, что $\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A})$

Проблема эквивалентности

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, что $\mathfrak{K}'(\mathcal{A}) = \mathfrak{K}''(\mathcal{A})$

$$\mathfrak{K}'(\mathcal{A}) = \mathfrak{K}''(\mathcal{A}) \iff \mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A}) \ \wedge\ \mathfrak{K}''(\mathcal{A}) \supseteq \mathfrak{K}'(\mathcal{A})$$

Включение и эквивалентность

Проблема включения

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, что $\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A})$

Проблема эквивалентности

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, что $\mathfrak{K}'(\mathcal{A}) = \mathfrak{K}''(\mathcal{A})$

$$\mathfrak{K}'(\mathcal{A}) = \mathfrak{K}''(\mathcal{A}) \iff \mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A}) \ \wedge \ \mathfrak{K}''(\mathcal{A}) \supseteq \mathfrak{K}'(\mathcal{A})$$

$$\mathfrak{K}'(r_1, \ldots, r_n) \supseteq \mathfrak{K}''(r_1, \ldots, r_n) \iff$$
$$\iff \mathfrak{K}'(r_1, \ldots, r_n) = \mathfrak{K}'(r_1, \ldots, r_n) \cup \mathfrak{K}''(r_1, \ldots, r_n)$$

Включение и эквивалентность

Проблема включения

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, что $\mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A})$

Проблема эквивалентности

ВХОД: Два обобщеных коня $\mathfrak{K}'$ и $\mathfrak{K}''$ и поле $\mathcal{A}$

ВОПРОС: Верно ли, что $\mathfrak{K}'(\mathcal{A}) = \mathfrak{K}''(\mathcal{A})$

$$\mathfrak{K}'(\mathcal{A}) = \mathfrak{K}''(\mathcal{A}) \iff \mathfrak{K}'(\mathcal{A}) \supseteq \mathfrak{K}''(\mathcal{A}) \ \wedge\ \mathfrak{K}''(\mathcal{A}) \supseteq \mathfrak{K}'(\mathcal{A})$$

$$\mathfrak{K}'(r_1, \ldots, r_n) \supseteq \mathfrak{K}''(r_1, \ldots, r_n) \iff$$
$$\iff \mathfrak{K}'(r_1, \ldots, r_n) = \mathfrak{K}'(r_1, \ldots, r_n) \cup \mathfrak{K}''(r_1, \ldots, r_n)$$

$$\mathfrak{K}(r_1, \ldots, r_n) = \mathfrak{K}'(r_1, \ldots, r_n) \cup \mathfrak{K}''(r_1, \ldots, r_n)$$

Включение и эквивалентность

$\mathfrak{K}'$

$$R_{i_1} \dots R_{i_a} \rightarrowtail R_{j_1} \dots R_{j_b}$$

$\mathfrak{K}''$

$$R_{i_1} \dots R_{i_a} \rightarrowtail R_{j_1} \dots R_{j_b}$$

$\mathfrak{K}'''$

$$T_{m+4} R_{i_1} \dots R_{i_a} \rightarrowtail T_{m+5} R_{j_1} \dots R_{j_b}$$
$$T_{m+5} R_{i_1} \dots R_{i_a} \rightarrowtail T_{m+4} R_{j_1} \dots R_{j_b}$$
$$S_{m+8} \rightarrowtail T_{m+4} S_{m+2}$$
$$S_{m+8} \rightarrowtail T_{m+5} S_{m+2}$$
$$T_{m+4} \rightarrowtail$$
$$T_{m+5} \rightarrowtail$$

$$T_{m+6} R_{i_1} \dots R_{i_a} \rightarrowtail T_{m+7} R_{j_1} \dots R_{j_b}$$
$$T_{m+7} R_{i_1} \dots R_{i_a} \rightarrowtail T_{m+6} R_{j_1} \dots R_{j_b}$$
$$S_{m+8} \rightarrowtail T_{m+6} S_{m+2}$$
$$S_{m+8} \rightarrowtail T_{m+7} S_{m+2}$$

$\mathfrak{K}''''$

все инструкции $\mathfrak{K}'''$

$$T_{m+6} \rightarrowtail$$
$$T_{m+7} \rightarrowtail$$

Включение и эквивалентность

$\mathfrak{K}'$	$\mathfrak{K}''$
$R_{i_1}\dots R_{i_a} \rightarrowtail R_{j_1}\dots R_{j_b}$	$R_{i_1}\dots R_{i_a} \rightarrowtail R_{j_1}\dots R_{j_b}$

$\mathfrak{K}'''$

$T_{m+4}R_{i_1}\dots R_{i_a} \rightarrowtail T_{m+5}R_{j_1}\dots R_{j_b}$	$T_{m+6}R_{i_1}\dots R_{i_a} \rightarrowtail T_{m+7}R_{j_1}\dots R_{j_b}$
$T_{m+5}R_{i_1}\dots R_{i_a} \rightarrowtail T_{m+4}R_{j_1}\dots R_{j_b}$	$T_{m+7}R_{i_1}\dots R_{i_a} \rightarrowtail T_{m+6}R_{j_1}\dots R_{j_b}$
$S_{m+8} \rightarrowtail T_{m+4}S_{m+2}$	$S_{m+8} \rightarrowtail T_{m+6}S_{m+2}$
$S_{m+8} \rightarrowtail T_{m+5}S_{m+2}$	$S_{m+8} \rightarrowtail T_{m+7}S_{m+2}$
$T_{m+4} \rightarrowtail$	
$T_{m+5} \rightarrowtail$	

$\mathfrak{K}''''$

все инструкции $\mathfrak{K}'''$

$T_{m+6} \rightarrowtail$

$T_{m+7} \rightarrowtail$

$$\mathfrak{K}' \supseteq \mathfrak{K}'' \iff \mathfrak{K}''' = \mathfrak{K}''''$$

Archiving Diophantine Sets

$$a \in \mathfrak{M} \Longleftrightarrow \exists x_1 \ldots x_m [P(a, x_1, \ldots, x_m) = 0]$$

Archiving Diophantine Sets

$$a \in \mathfrak{M} \iff \exists x_1 \dots x_m [P(a, x_1, \dots, x_m) = 0]$$

$$B = b_1 \dots b_k \dots \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

Archiving Diophantine Sets

$$a \in \mathfrak{M} \Longleftrightarrow \exists x_1 \dots x_m [P(a, x_1, \dots, x_m) = 0]$$

$$B = b_1 \dots b_k \dots \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$B_n = b_1 \dots b_k \dots b_n$$

Archiving Diophantine Sets

$$a \in \mathfrak{M} \Longleftrightarrow \exists x_1 \dots x_m [P(a, x_1, \dots, x_m) = 0]$$

$$B = b_1 \dots b_k \dots \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$B_n = b_1 \dots b_k \dots b_n$$

Archiving Diophantine Sets

$$a \in \mathfrak{M} \Longleftrightarrow \exists x_1 \dots x_m [P(a, x_1, \dots, x_m) = 0]$$

$$B = b_1 \dots b_k \dots \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$B_n = b_1 \dots b_k \dots b_n$$

Alice

Bob

Archiving Diophantine Sets

$$a \in \mathfrak{M} \Longleftrightarrow \exists x_1 \ldots x_m [P(a, x_1, \ldots, x_m) = 0]$$

$$B = b_1 \ldots b_k \ldots \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$B_n = b_1 \ldots b_k \ldots b_n$$

Alice	Bob
B_n

Archiving Diophantine Sets

$$a \in \mathfrak{M} \Longleftrightarrow \exists x_1 \dots x_m [P(a, x_1, \dots, x_m) = 0]$$

$$B = b_1 \dots b_k \dots \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$B_n = b_1 \dots b_k \dots b_n$$

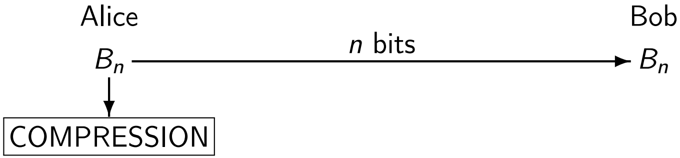
$$\begin{array}{ccc} \text{Alice} & & \text{Bob} \\ B_n & \xrightarrow{\quad n \text{ bits} \quad} & B_n \end{array}$$

Archiving Diophantine Sets

$$a \in \mathfrak{M} \iff \exists x_1 \dots x_m [P(a, x_1, \dots, x_m) = 0]$$

$$B = b_1 \dots b_k \dots \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$B_n = b_1 \dots b_k \dots b_n$$



Archiving Diophantine Sets

$$a \in \mathfrak{M} \iff \exists x_1 \dots x_m [P(a, x_1, \dots, x_m) = 0]$$

$$B = b_1 \dots b_k \dots \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$B_n = b_1 \dots b_k \dots b_n$$

$$\begin{tikzcd}
\text{Alice} & & \text{Bob} \\
B_n \arrow[d] \arrow[rr, "n \text{ bits}"] & & B_n \\
\boxed{\text{COMPRESSION}} \arrow[d] & & \\
C_n & &
\end{tikzcd}$$

Archiving Diophantine Sets

$$a \in \mathfrak{M} \Longleftrightarrow \exists x_1 \dots x_m [P(a, x_1, \dots, x_m) = 0]$$

$$B = b_1 \dots b_k \dots \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$B_n = b_1 \dots b_k \dots b_n$$

$$\begin{tikzcd}
\text{Alice} & & \text{Bob} \\
B_n \arrow[rr, "n \text{ bits}"] \arrow[d] & & B_n \\
\boxed{\text{COMPRESSION}} \arrow[d] & & \\
C_n \arrow[rr, "\text{Length}(C_n) \text{ bits}"] & & C_n
\end{tikzcd}$$

Archiving Diophantine Sets

$$a \in \mathfrak{M} \iff \exists x_1 \dots x_m [P(a, x_1, \dots, x_m) = 0]$$

$$B = b_1 \dots b_k \dots \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$B_n = b_1 \dots b_k \dots b_n$$

Alice | Bob

$$\begin{tikzcd}[column sep=huge]
B_n \arrow[rr, "n \text{ bits}"] \arrow[d] & & B_n \\
\boxed{\text{COMPRESSION}} \arrow[d] & & \boxed{\text{DECOMPRESSION}} \\
C_n \arrow[rr, "\text{Length}(C_n) \text{ bits}"] & & C_n \arrow[u]
\end{tikzcd}$$

Archiving Diophantine Sets

$$a \in \mathfrak{M} \Longleftrightarrow \exists x_1 \dots x_m [P(a, x_1, \dots, x_m) = 0]$$

$$B = b_1 \dots b_k \dots \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$B_n = b_1 \dots b_k \dots b_n$$

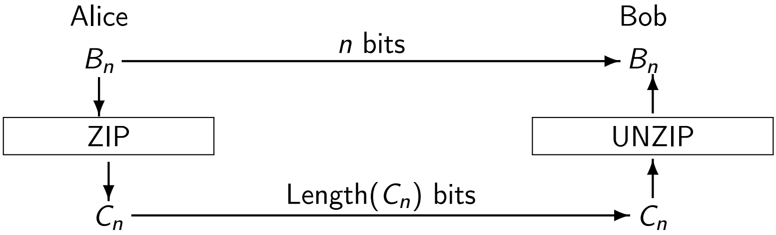
$$\begin{tikzcd}
\text{Alice} & \text{Bob} \\
B_n \arrow[r, "n \text{ bits}"] \arrow[d] & B_n \\
\boxed{\text{COMPRESSION}} \arrow[d] & \boxed{\text{DECOMPRESSION}} \arrow[u] \\
C_n \arrow[r, "\text{Length}(C_n) \text{ bits}"] & C_n \arrow[u]
\end{tikzcd}$$

Archiving Diophantine Sets

$$a \in \mathfrak{M} \Longleftrightarrow \exists x_1 \ldots x_m [P(a, x_1, \ldots, x_m) = 0]$$

$$B = b_1 \ldots b_k \ldots \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$B_n = b_1 \ldots b_k \ldots b_n$$



Archiving Diophantine Sets

$$a \in \mathfrak{M} \iff \exists x_1 \ldots x_m [P(a, x_1, \ldots, x_m) = 0]$$

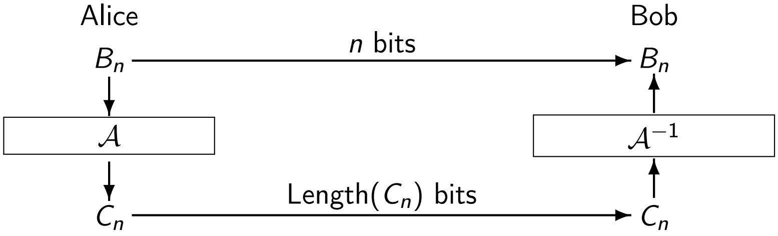
$$B = b_1 \ldots b_k \ldots \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$B_n = b_1 \ldots b_k \ldots b_n$$

$$\begin{tikzcd}
\text{Alice} & & \text{Bob} \\
B_n \arrow[rr, "n \text{ bits}"] \arrow[d] & & B_n \\
\boxed{\mathcal{A}} \arrow[d] & & \boxed{\mathcal{A}^{-1}} \arrow[u] \\
C_n \arrow[rr, "\mathrm{Length}(C_n) \text{ bits}"] & & C_n \arrow[u]
\end{tikzcd}$$

Saving Constant Number of Bits

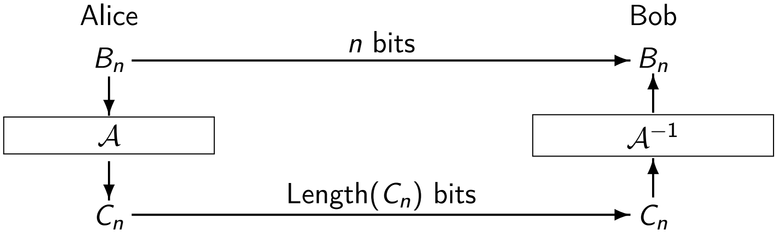
$$B = b_1 \dots b_k \dots$$



Saving Constant Number of Bits

$$B = b_1 \dots b_k \dots$$

WE CAN always save any constant number k of bits (for $n > k$):



Saving Constant Number of Bits

$$B = b_1 \ldots b_k \ldots$$

WE CAN always save any constant number k of bits (for $n > k$):

$$C_n = \mathcal{A}(B_n)$$

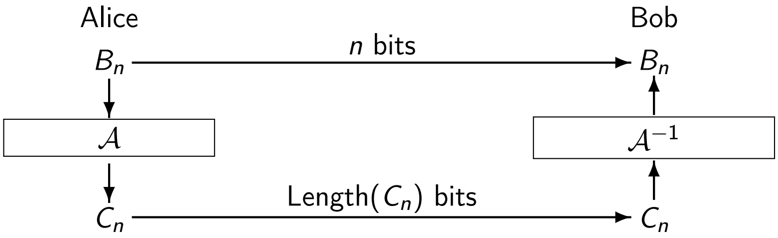
$$\begin{tikzcd}
\text{Alice} & & \text{Bob} \\
B_n \arrow[rr, "n \text{ bits}"] \arrow[d] & & B_n \\
\boxed{\mathcal{A}} \arrow[d] & & \boxed{\mathcal{A}^{-1}} \arrow[u] \\
C_n \arrow[rr, "\text{Length}(C_n) \text{ bits}"] & & C_n \arrow[u]
\end{tikzcd}$$

Saving Constant Number of Bits

$$B = b_1 \dots b_k \dots$$

WE CAN always save any constant number k of bits (for $n > k$):

$$C_n = \mathcal{A}(B_n) \quad = \quad \begin{cases} & \text{if } n \leq k+1 \\ & \text{otherwise} \end{cases}$$

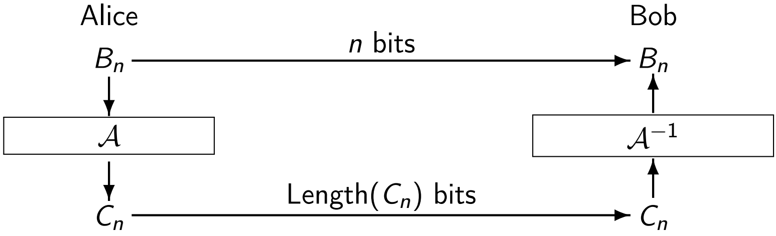


Saving Constant Number of Bits

$$B = b_1 \dots b_k \dots$$

WE CAN always save any constant number k of bits (for $n > k$):

$$C_n = \mathcal{A}(B_n) \quad = \quad \begin{cases} 0B_n & \text{if } n \leq k+1 \\ & \text{otherwise} \end{cases}$$

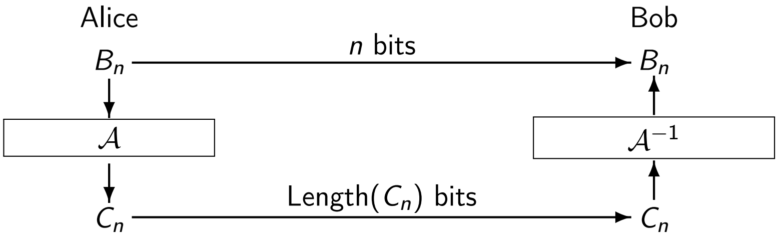


Saving Constant Number of Bits

$$B = b_1 \dots b_k \dots$$

WE CAN always save any constant number k of bits (for $n > k$):

$$C_n = \mathcal{A}(B_n) \quad = \quad \begin{cases} 0B_n & \text{if } n \le k+1 \\ 1b_{k+2}b_{k+3}\dots b_n & \text{otherwise} \end{cases}$$



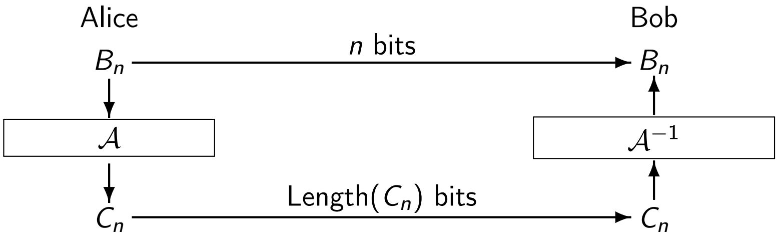
Saving Constant Number of Bits

$$B = b_1 \ldots b_k \ldots$$

WE CAN always save any constant number k of bits (for $n > k$):

$$C_n = \mathcal{A}(B_n) = \begin{cases} 0B_n & \text{if } n \leq k+1 \\ 1b_{k+2}b_{k+3}\ldots b_n & \text{otherwise} \end{cases}$$

$$\mathcal{A}^{-1}(0C) = C$$



Saving Constant Number of Bits

$$B = b_1 \dots b_k \dots$$

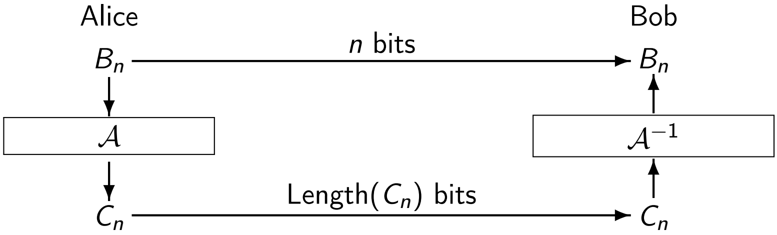
WE CAN always save any constant number k of bits (for $n > k$):

$$\begin{aligned}
C_n = \mathcal{A}(B_n) &= \begin{cases} 0B_n & \text{if } n \leq k+1 \\ 1b_{k+2}b_{k+3}\dots b_n & \text{otherwise} \end{cases} \\
\mathcal{A}^{-1}(0C) &= C \\
\mathcal{A}^{-1}(1C) &= b_1 \dots b_{k+1} C
\end{aligned}$$

$$\begin{tikzcd}
\text{Alice} & & \text{Bob} \\
B_n \arrow[rr, "n \text{ bits}"] \arrow[d] & & B_n \\
\boxed{\mathcal{A}} \arrow[d] & & \boxed{\mathcal{A}^{-1}} \arrow[u] \\
C_n \arrow[rr, "\text{Length}(C_n) \text{ bits}"] & & C_n \arrow[u]
\end{tikzcd}$$

Case of Decidable $\mathfrak{M}$

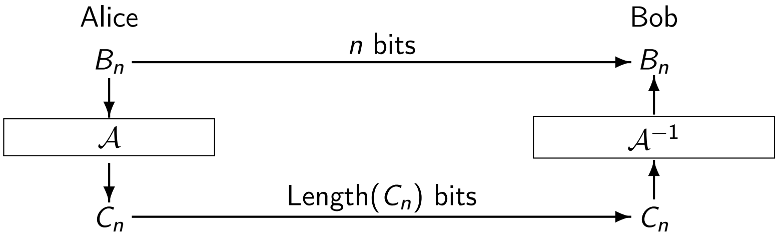
$$B_n = b_1 b_2 \ldots b_k \ldots b_n \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$



Case of Decidable $\mathfrak{M}$

$$B_n = b_1 b_2 \ldots b_k \ldots b_n \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

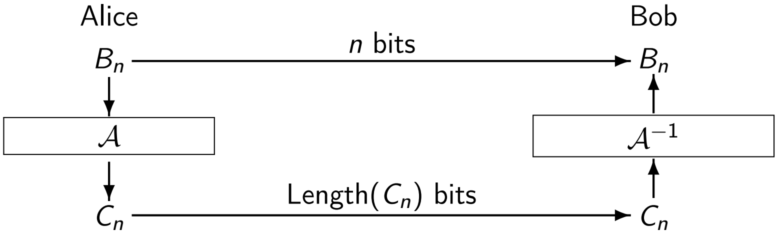
$$C_n = \mathcal{A}(B_n)$$



Case of Decidable $\mathfrak{M}$

$$B_n = b_1 b_2 \dots b_k \dots b_n \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

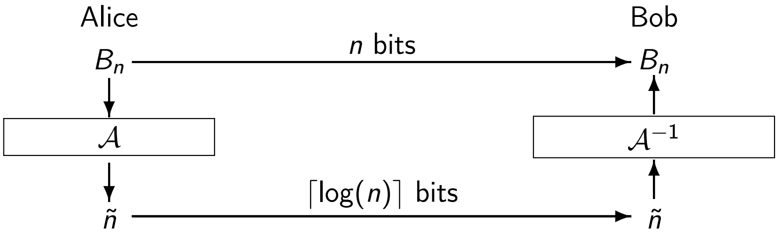
$$C_n = \mathcal{A}(B_n) \quad = \tilde{n} = \text{binary notation of } n$$



Case of Decidable $\mathfrak{M}$

$$B_n = b_1 b_2 \dots b_k \dots b_n \qquad b_k = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

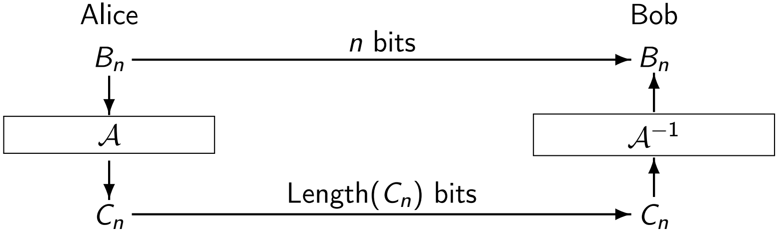
$$C_n = \mathcal{A}(B_n) \quad = \tilde{n} = \text{binary notation of } n$$



Case of Diophantine $\mathfrak{M}$

$$a \in \mathfrak{M} \iff \exists x_1 \ldots x_m [P(a, x_1, \ldots, x_m) = 0]$$

$$B_n = b_1 \ldots b_k \ldots b_n \qquad b_a = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$



Case of Diophantine $\mathfrak{M}$

$$a \in \mathfrak{M} \iff \exists x_1 \ldots x_m [P(a, x_1, \ldots, x_m) = 0]$$

$$B_n = b_1 \ldots b_k \ldots b_n \qquad b_a = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$C_n = \mathcal{A}(B_n)$$

$$\begin{tikzcd}
\text{Alice} & & \text{Bob} \\
B_n \arrow[rr, "n \text{ bits}"] \arrow[d] & & B_n \\
\boxed{\mathcal{A}} \arrow[d] & & \boxed{\mathcal{A}^{-1}} \arrow[u] \\
C_n \arrow[rr, "\text{Length}(C_n) \text{ bits}"] & & C_n \arrow[u]
\end{tikzcd}$$

Case of Diophantine $\mathfrak{M}$

$$a \in \mathfrak{M} \iff \exists x_1 \dots x_m [P(a, x_1, \dots, x_m) = 0]$$

$$B_n = b_1 \dots b_k \dots b_n \qquad b_a = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$C_n = \mathcal{A}(B_n) \quad = \tilde{n}\widetilde{q_n}$$

where q_n is the number of "1" in B_n and $\widetilde{q_n}$ is the binary notation of q_n padded by leading zeros to the length of n.

$$\begin{tikzcd} \text{Alice} & & \text{Bob} \\ B_n \arrow[rr, "n \text{ bits}"] \arrow[d] & & B_n \\ \boxed{\mathcal{A}} \arrow[d] & & \boxed{\mathcal{A}^{-1}} \arrow[u] \\ C_n \arrow[rr, "\text{Length}(C_n) \text{ bits}"] & & C_n \arrow[u] \end{tikzcd}$$

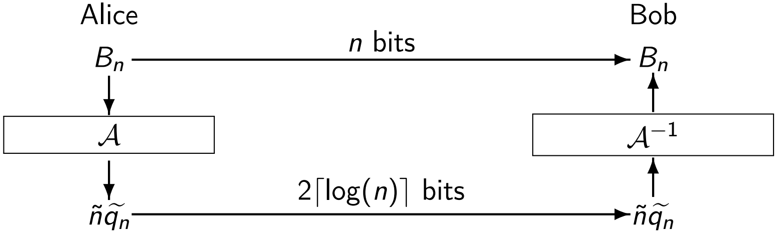
Case of Diophantine $\mathfrak{M}$

$$a \in \mathfrak{M} \Longleftrightarrow \exists x_1 \dots x_m [P(a, x_1, \dots, x_m) = 0]$$

$$B_n = b_1 \dots b_k \dots b_n \qquad b_a = \begin{cases} 1, & \text{if } k \in \mathfrak{M} \\ 0 & \text{otherwise} \end{cases}$$

$$C_n = \mathcal{A}(B_n) \quad = \tilde{n}\widetilde{q}_n$$

where q_n is the number of "1" in B_n and $\widetilde{q}_n$ is the binary notation of q_n padded by leading zeros to the length of n.



Computational Chaos in Number Theory

Gregory Chaitin [1987] constructed a particular one-parameter exponential Diophantine equation and considered the set of all values of the parameter for which the equation has infinitely many solutions:

$$a \in \mathfrak{M} \iff \exists^{\infty} x_1 \ldots x_m [E(a, x_1, x_2, \ldots, x_m) = 0]$$

Computational Chaos in Number Theory

Gregory Chaitin [1987] constructed a particular one-parameter exponential Diophantine equation and considered the set of all values of the parameter for which the equation has infinitely many solutions:

$$a \in \mathfrak{M} \Longleftrightarrow \exists^{\infty} x_1 \ldots x_m [E(a, x_1, x_2, \ldots, x_m) = 0]$$

He proved that so called prefix-free Kolmogorov complexity (defined up to an additive constant) of this set is equal to n.

Computational Chaos in Number Theory

Gregory Chaitin [1987] constructed a particular one-parameter exponential Diophantine equation and considered the set of all values of the parameter for which the equation has infinitely many solutions:

$$a \in \mathfrak{M} \iff \exists^{\infty} x_1 \dots x_m [E(a, x_1, x_2, \dots, x_m) = 0]$$

He proved that so called prefix-free Kolmogorov complexity (defined up to an additive constant) of this set is equal to n.

Informally, one can say that the set $\mathfrak{M}$ is completely chaotic.

More Computational Chaos in Number Theory

Toby Ord and Tien D. Kieu [2003] constructed another particular one-parameter exponential Diophantine equation which for every value of the parameter has only finitely many solutions and considered the set of all values of the parameter for which the equation has even number of solutions:

$$a \in \mathfrak{M} \Longleftrightarrow \exists^{\mathrm{even}} x_1 \ldots x_m [E(a, x_1, x_2, \ldots, x_m) = 0]$$

More Computational Chaos in Number Theory

Toby Ord and Tien D. Kieu [2003] constructed another particular one-parameter exponential Diophantine equation which for every value of the parameter has only finitely many solutions and considered the set of all values of the parameter for which the equation has even number of solutions:

$$a \in \mathfrak{M} \Longleftrightarrow \exists^{\text{even}} x_1 \ldots x_m [E(a, x_1, x_2, \ldots, x_m) = 0]$$

They proved that the prefix-free Kolmogorov complexity of this set is also equal to n (up to an additive constant).

Even More Computational Chaos in Number Theory

Theorem (Matiyasevich [2006]). *Let $\mathfrak{U}$ be a decidable infinite set with infinite complement. WE CAN construct an exponential Diophantine equation which for every value of the parameter has only finitely many solutions and such that the prefix-free Kolmogorov complexity of the set*

$$a \in \mathfrak{M} \Longleftrightarrow \exists^{\mathfrak{U}} x_1 \ldots x_m [E(a, x_1, x_2, \ldots, x_m) = 0]$$

is equal to n (up to an additive constant).

Унификация (невсеобщее равенство)

$$E_1(x_1, \ldots, x_m) = E_2(x_1, \ldots, x_m)$$

Унификация (невсеобщее равенство)

$$E_1(x_1, \ldots, x_m) = E_2(x_1, \ldots, x_m)$$

Проблема унификации для чистого исчисления предикатов первого порядка разрешима.

Унификация (невсеобщее равенство)

$$E_1(x_1, \ldots, x_m) = E_2(x_1, \ldots, x_m)$$

Проблема унификации для чистого исчисления предикатов первого порядка разрешима.

Проблема унификации для исчисления предикатов третьего порядка неразрешима.

Унификация (невсеобщее равенство)

$$E_1(x_1, \ldots, x_m) = E_2(x_1, \ldots, x_m)$$

Проблема унификации для чистого исчисления предикатов первого порядка разрешима.

Проблема унификации для исчисления предикатов третьего порядка неразрешима. L. D. Baxter [1978] дал новое доказательство этого факта с использованием неразрешимости диофантовых уравнений.

Унификация (невсеобщее равенство)

$$E_1(x_1, \dots, x_m) = E_2(x_1, \dots, x_m)$$

Проблема унификации для чистого исчисления предикатов первого порядка разрешима.

Проблема унификации для исчисления предикатов третьего порядка неразрешима. L. D. Baxter [1978] дал новое доказательство этого факта с использованием неразрешимости диофантовых уравнений.

W. D. Golfarb [1981] установил неразрешимость проблема унификации для исчисления предикатов второго порядка, исходя из неразрешимости 10-й проблемы Гильберта.

Одна история

Одна история

Разрешима ли проблема так называемой *одновременной жесткой E-унификации (simultaneous rigid E-unification)*?

Одна история

Разрешима ли проблема так называемой *одновременной жесткой* E*-унификации (simultaneous rigid* E*-unification)*?

А. Воронков и А. Дегтярев установили неразрешимость одновременной жесткой E-унификации

Одна история

Разрешима ли проблема так называемой *одновременной жесткой* *E-унификации (simultaneous rigid E-unification)*?

А. Воронков и А. Дегтярев установили неразрешимость одновременной жесткой E-унификации путем сведения к ней так называемой *проблемы монадической полуунификации (monadic semi-unification problem)*.

Одна история

Разрешима ли проблема так называемой *одновременной жесткой E-унификации (simultaneous rigid E-unification)*?

А. Воронков и А. Дегтярев установили неразрешимость одновременной жесткой E-унификации путем сведения к ней так называемой *проблемы монадической полуунификации (monadic semi-unification problem)*.

Неразрешимость проблемы монадической полуунификации установил ранее M. Baaz [1993] путем сведения к ней проблемы унификации для исчисления предикатов второго порядка.

Одна история

Разрешима ли проблема так называемой *одновременной жесткой E-унификации (simultaneous rigid E-unification)*?

А. Воронков и А. Дегтярев установили неразрешимость одновременной жесткой E-унификации путем сведения к ней так называемой *проблемы монадической полуунификации (monadic semi-unification problem)*.

Неразрешимость проблемы монадической полуунификации установил ранее M. Baaz [1993] путем сведения к ней проблемы унификации для исчисления предикатов второго порядка.

W. D. Golfarb [1981] установил неразрешимость проблемы унификации для исчисления предикатов второго порядка, исходя из неразрешимости 10-й проблемы Гильберта.

Одна история

Разрешима ли проблема так называемой *одновременной жесткой E-унификации (simultaneous rigid E-unification)*?

А. Воронков и А. Дегтярев [1996] дали прямое доказательство неразрешимость одновременной жесткой E-унификации на основе неразрешимости 10-й проблемы Гильберта.