Математическая логика и теория алгоритмов

ЛЕКТОР: В.А. Захаров

zakh@cs.msu.su

http://mathcyb.cs.msu.su/courses/logprog.html

Лекция 5.

Полнота табличного вывода. Теорема Левенгейма-Сколема. Теорема компактности Мальцева. Автоматическое доказательство теорем.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Теорема полноты табличного вывода

Если семантическая таблица T_0 невыполнима, то для T_0 существует успешный табличный вывод.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Теорема полноты табличного вывода

Если семантическая таблица T_0 невыполнима, то для T_0 существует успешный табличный вывод.

Доказательство.

Проведем для упрощенного частного случая таблицы T_0, в которой

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Теорема полноты табличного вывода

Если семантическая таблица T_0 невыполнима, то для T_0 существует успешный табличный вывод.

Доказательство.

Проведем для упрощенного частного случая таблицы T_0, в которой

- имеется лишь конечное число формул,

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Теорема полноты табличного вывода

Если семантическая таблица T_0 невыполнима, то для T_0 существует успешный табличный вывод.

Доказательство.

Проведем для упрощенного частного случая таблицы T_0, в которой

- имеется лишь конечное число формул,
- все формулы замкнутые,

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Теорема полноты табличного вывода

Если семантическая таблица T_0 невыполнима, то для T_0 существует успешный табличный вывод.

Доказательство.

Проведем для упрощенного частного случая таблицы T_0, в которой

- имеется лишь конечное число формул,
- все формулы замкнутые,
- в формулах нет функциональных символов.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Теорема полноты табличного вывода

Если семантическая таблица T_0 невыполнима, то для T_0 существует успешный табличный вывод.

Доказательство.

Проведем для упрощенного частного случая таблицы T_0, в которой

- имеется лишь конечное число формул,
- все формулы замкнутые,
- в формулах нет функциональных символов.

Пусть $T_0 = \langle\ \Gamma_0 \mid \Delta_0\ \rangle$ — невыполнимая таблица. Будем строить табличный вывод для T_0, руководствуясь следующей стратегией.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

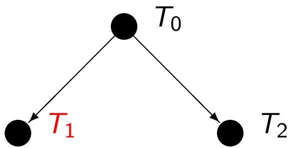
1). Каждая незакрытая таблица в дереве вывода получает порядковый номер, и правила табличного вывода применяются к таблицам в порядке возрастания их номеров.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

1). Каждая незакрытая таблица в дереве вывода получает порядковый номер, и правила табличного вывода применяются к таблицам в порядке возрастания их номеров.

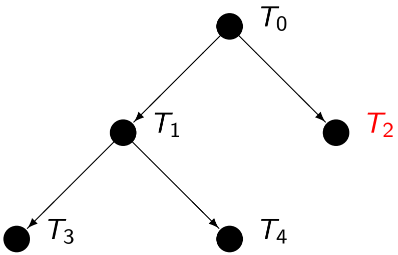
ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

1). Каждая незакрытая таблица в дереве вывода получает порядковый номер, и правила табличного вывода применяются к таблицам в порядке возрастания их номеров.



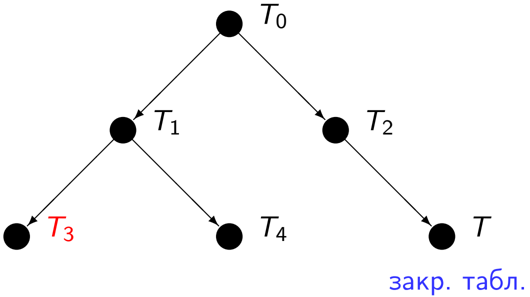
ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

1). Каждая незакрытая таблица в дереве вывода получает порядковый номер, и правила табличного вывода применяются к таблицам в порядке возрастания их номеров.



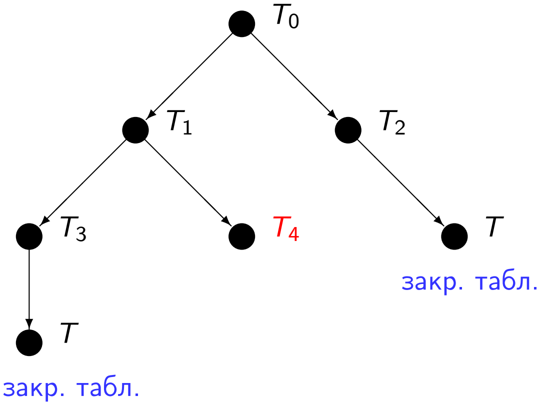
ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

1). Каждая незакрытая таблица в дереве вывода получает порядковый номер, и правила табличного вывода применяются к таблицам в порядке возрастания их номеров.



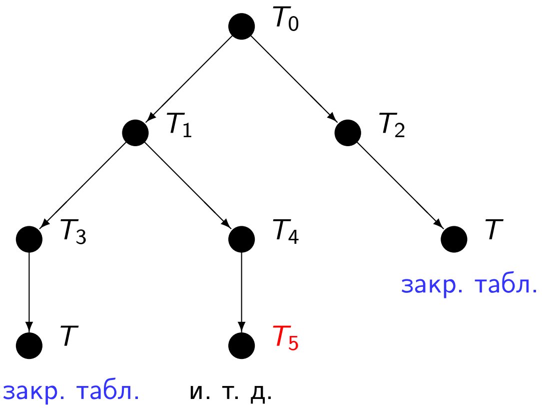
ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

1). Каждая незакрытая таблица в дереве вывода получает порядковый номер, и правила табличного вывода применяются к таблицам в порядке возрастания их номеров.



ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

1). Каждая незакрытая таблица в дереве вывода получает порядковый номер, и правила табличного вывода применяются к таблицам в порядке возрастания их номеров.



ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

2). Таблицы состоят из **упорядоченных** множеств формул (списков). Правила применяются к формулам в порядке их расположения в списках. Формулы, участвующие в применении правил, помещаются в хвост нужного списка. Атомарные формулы просто переходят из головы списка в его хвост.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

2). Таблицы состоят из **упорядоченных** множеств формул (списков). Правила применяются к формулам в порядке их расположения в списках. Формулы, участвующие в применении правил, помещаются в хвост нужного списка. Атомарные формулы просто переходят из головы списка в его хвост.

$$T_0 = \langle\ \forall x\varphi, \psi_1 \to \psi_2, \Gamma \mid \chi_1 \vee \chi_2, \Delta\ \rangle$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

2). Таблицы состоят из **упорядоченных** множеств формул (списков). Правила применяются к формулам в порядке их расположения в списках. Формулы, участвующие в применении правил, помещаются в хвост нужного списка. Атомарные формулы просто переходят из головы списка в его хвост.

$$T_0 = \langle\, \forall x\varphi, \psi_1 \to \psi_2, \Gamma \mid \chi_1 \vee \chi_2, \Delta \,\rangle$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

2). Таблицы состоят из **упорядоченных** множеств формул (списков). Правила применяются к формулам в порядке их расположения в списках. Формулы, участвующие в применении правил, помещаются в хвост нужного списка. Атомарные формулы просто переходят из головы списка в его хвост.

$$\begin{array}{c} T_0 = \langle\ \forall x\varphi, \psi_1 \to \psi_2, \Gamma \mid \chi_1 \vee \chi_2, \Delta\ \rangle \\ \Big\downarrow (L\forall) \\ T_1 = \langle\ \psi_1 \to \psi_2, \Gamma, \forall x\varphi, \varphi' \mid \chi_1 \vee \chi_2, \Delta\ \rangle \end{array}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

2). Таблицы состоят из **упорядоченных** множеств формул (списков). Правила применяются к формулам в порядке их расположения в списках. Формулы, участвующие в применении правил, помещаются в хвост нужного списка. Атомарные формулы просто переходят из головы списка в его хвост.

$$T_0 = \langle\ \forall x\varphi, \psi_1 \to \psi_2, \Gamma \mid \chi_1 \vee \chi_2, \Delta\ \rangle$$

$$\downarrow (L\forall)$$

$$T_1 = \langle\ \psi_1 \to \psi_2, \Gamma, \forall x\varphi, \varphi' \mid \chi_1 \vee \chi_2, \Delta\ \rangle$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

2). Таблицы состоят из **упорядоченных** множеств формул (списков). Правила применяются к формулам в порядке их расположения в списках. Формулы, участвующие в применении правил, помещаются в хвост нужного списка. Атомарные формулы просто переходят из головы списка в его хвост.

$$
\begin{array}{c}
T_0 = \langle\ \forall x\varphi, \psi_1 \to \psi_2, \Gamma \mid \chi_1 \vee \chi_2, \Delta\ \rangle \\
\downarrow (L\forall) \\
T_1 = \langle\ \psi_1 \to \psi_2, \Gamma, \forall x\varphi, \varphi' \mid \chi_1 \vee \chi_2, \Delta\ \rangle \\
\downarrow (R\vee) \\
T_2 = \langle\ \psi_1 \to \psi_2, \Gamma, \forall x\varphi, \varphi' \mid \Delta, \chi_1, \chi_2\ \rangle
\end{array}
$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

2). Таблицы состоят из **упорядоченных** множеств формул (списков). Правила применяются к формулам в порядке их расположения в списках. Формулы, участвующие в применении правил, помещаются в хвост нужного списка. Атомарные формулы просто переходят из головы списка в его хвост.

$$\begin{array}{c}
T_0 = \langle\ \forall x\varphi, \psi_1 \to \psi_2, \Gamma \mid \chi_1 \vee \chi_2, \Delta\ \rangle \\
\Big\downarrow (L\forall) \\
T_1 = \langle\ \psi_1 \to \psi_2, \Gamma, \forall x\varphi, \varphi' \mid \chi_1 \vee \chi_2, \Delta\ \rangle \\
\Big\downarrow (R\vee) \\
T_2 = \langle\ {\color{red}\psi_1 \to \psi_2}, \Gamma, \forall x\varphi, \varphi' \mid \Delta, \chi_1, \chi_2\ \rangle
\end{array}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

2). Таблицы состоят из **упорядоченных** множеств формул (списков). Правила применяются к формулам в порядке их расположения в списках. Формулы, участвующие в применении правил, помещаются в хвост нужного списка. Атомарные формулы просто переходят из головы списка в его хвост.

$$T_0 = \langle\ \forall x\varphi, \psi_1 \to \psi_2, \Gamma \mid \chi_1 \vee \chi_2, \Delta\ \rangle$$

$$\downarrow (L\forall)$$

$$T_1 = \langle\ \psi_1 \to \psi_2, \Gamma, \forall x\varphi, \varphi' \mid \chi_1 \vee \chi_2, \Delta\ \rangle$$

$$\downarrow (R\vee)$$

$$T_2 = \langle\ {\color{red}\psi_1 \to \psi_2}, \Gamma, \forall x\varphi, \varphi' \mid \Delta, \chi_1, \chi_2\ \rangle$$

$$\swarrow \quad {\color{red}(L \to)} \quad \searrow$$

$$T_3 = \langle\ \Gamma, \forall x\varphi, \varphi', {\color{blue}\psi_2} \mid \Delta, \chi_1, \chi_2\ \rangle \qquad T_4 = \langle \Gamma, \forall x\varphi, \varphi' \mid \Delta, \chi_1, \chi_2, {\color{blue}\psi_1}\ \rangle$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

2). Таблицы состоят из <u>**упорядоченных**</u> множеств формул (списков). Правила применяются к формулам в порядке их расположения в списках. Формулы, участвующие в применении правил, помещаются в хвост нужного списка. Атомарные формулы просто переходят из головы списка в его хвост.

$$T_0 = \langle\ \forall x\varphi, \psi_1 \to \psi_2, \Gamma \mid \chi_1 \vee \chi_2, \Delta\ \rangle$$

$$\Big\downarrow\ (L\forall)$$

$$T_1 = \langle\ \psi_1 \to \psi_2, \Gamma, \forall x\varphi, \varphi' \mid \chi_1 \vee \chi_2, \Delta\ \rangle$$

$$\Big\downarrow\ (R\vee)$$

$$T_2 = \langle\ \psi_1 \to \psi_2, \Gamma, \forall x\varphi, \varphi' \mid \Delta, \chi_1, \chi_2\ \rangle$$

$$\swarrow\ (L\to)\ \searrow$$

$$T_3 = \langle\ \Gamma, \forall x\varphi, \varphi', \psi_2 \mid \Delta, \chi_1, \chi_2\ \rangle \qquad T_4 = \langle \Gamma, \forall x\varphi, \varphi' \mid \Delta, \chi_1, \chi_2, \psi_1\ \rangle$$

и. т. д.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

3). С каждой таблицей T ассоциирован **непустой** список использованных констант L_T. Вначале в этот список включаются все константы, содержащиеся в таблице T_0 (если в T_0 констант нет, то полагаем $L_{T_0} = \{c_0\}$). После применения правил $(L\exists)$ и $(R\forall)$ в список порожденной таблицы добавляется «свежая константа». В остальных случаях список наследуется без изменений.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

3). С каждой таблицей T ассоциирован **непустой** список использованных констант L_T. Вначале в этот список включаются все константы, содержащиеся в таблице T_0 (если в T_0 констант нет, то полагаем $L_{T_0} = \{c_0\}$). После применения правил $(L\exists)$ и $(R\forall)$ в список порожденной таблицы добавляется «свежая константа». В остальных случаях список наследуется без изменений.

$$T_0 = \langle\ \exists x\varphi(x), \psi_1 \to \psi_2, \Gamma \mid \forall y\chi(y), \Delta\ \rangle, \qquad L_0 = \{c', c''\}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

3). С каждой таблицей T ассоциирован **непустой** список использованных констант L_T. Вначале в этот список включаются все константы, содержащиеся в таблице T_0 (если в T_0 констант нет, то полагаем $L_{T_0} = \{c_0\}$). После применения правил $(L\exists)$ и $(R\forall)$ в список порожденной таблицы добавляется «свежая константа». В остальных случаях список наследуется без изменений.

$$T_0 = \langle\ \exists x\varphi(x), \psi_1 \to \psi_2, \Gamma \mid \forall y\chi(y), \Delta\ \rangle, \qquad L_0 = \{c', c''\}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

3). С каждой таблицей T ассоциирован **непустой** список использованных констант L_T. Вначале в этот список включаются все константы, содержащиеся в таблице T_0 (если в T_0 констант нет, то полагаем $L_{T_0} = \{c_0\}$). После применения правил $(L\exists)$ и $(R\forall)$ в список порожденной таблицы добавляется «свежая константа». В остальных случаях список наследуется без изменений.

$$T_0 = \langle\ \exists x\varphi(x), \psi_1 \to \psi_2, \Gamma \mid \forall y\chi(y), \Delta\ \rangle, \qquad L_0 = \{c', c''\}$$

$$\Big\downarrow (L\exists)$$

$$T_1 = \langle\ \psi_1 \to \psi_2, \Gamma, \varphi(c_1) \mid \forall y\chi(y), \Delta\ \rangle, \qquad L_1 = \{c', c'', c_1\}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

3). С каждой таблицей T ассоциирован **непустой** список использованных констант L_T. Вначале в этот список включаются все константы, содержащиеся в таблице T_0 (если в T_0 констант нет, то полагаем $L_{T_0} = \{c_0\}$). После применения правил $(L\exists)$ и $(R\forall)$ в список порожденной таблицы добавляется «свежая константа». В остальных случаях список наследуется без изменений.

$$T_0 = \langle\ \exists x\varphi(x), \psi_1 \to \psi_2, \Gamma \mid \forall y\chi(y), \Delta\ \rangle, \qquad L_0 = \{c', c''\}$$

$$\Big\downarrow (L\exists)$$

$$T_1 = \langle\ \psi_1 \to \psi_2, \Gamma, \varphi(c_1) \mid \forall y\chi(y), \Delta\ \rangle, \qquad L_1 = \{c', c'', c_1\}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

3). С каждой таблицей T ассоциирован **непустой** список использованных констант L_T. Вначале в этот список включаются все константы, содержащиеся в таблице T_0 (если в T_0 констант нет, то полагаем $L_{T_0} = \{c_0\}$). После применения правил $(L\exists)$ и $(R\forall)$ в список порожденной таблицы добавляется «свежая константа». В остальных случаях список наследуется без изменений.

$$T_0 = \langle\ \exists x\varphi(x), \psi_1 \to \psi_2, \Gamma \mid \forall y\chi(y), \Delta\ \rangle, \qquad L_0 = \{c', c''\}$$

$$\big\downarrow (L\exists)$$

$$T_1 = \langle\ \psi_1 \to \psi_2, \Gamma, \varphi(c_1) \mid \forall y\chi(y), \Delta\ \rangle, \qquad L_1 = \{c', c'', c_1\}$$

$$\big\downarrow (R\forall)$$

$$T_2 = \langle\ \psi_1 \to \psi_2, \Gamma, \varphi(c_1) \mid \Delta, \chi(c_2)\ \rangle, \qquad L_2 = \{c', c'', c_1, c_2\}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

3). С каждой таблицей T ассоциирован **непустой** список использованных констант L_T. Вначале в этот список включаются все константы, содержащиеся в таблице T_0 (если в T_0 констант нет, то полагаем $L_{T_0} = \{c_0\}$). После применения правил $(L\exists)$ и $(R\forall)$ в список порожденной таблицы добавляется «свежая константа». В остальных случаях список наследуется без изменений.

$$
\begin{array}{ll}
T_0 = \langle\ \exists x\varphi(x), \psi_1 \to \psi_2, \Gamma \mid \forall y\chi(y), \Delta\ \rangle, & L_0 = \{c', c''\} \\
\qquad\downarrow (L\exists) & \\
T_1 = \langle\ \psi_1 \to \psi_2, \Gamma, \varphi(c_1) \mid \forall y\chi(y), \Delta\ \rangle, & L_1 = \{c', c'', c_1\} \\
\qquad\downarrow (R\forall) & \\
T_2 = \langle\ {\color{red}\psi_1 \to \psi_2}, \Gamma, \varphi(c_1) \mid \Delta, \chi(c_2)\ \rangle, & L_2 = \{c', c'', c_1, c_2\}
\end{array}
$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

3). С каждой таблицей T ассоциирован **непустой** список использованных констант L_T. Вначале в этот список включаются все константы, содержащиеся в таблице T_0 (если в T_0 констант нет, то полагаем $L_{T_0} = \{c_0\}$). После применения правил $(L\exists)$ и $(R\forall)$ в список порожденной таблицы добавляется «свежая константа». В остальных случаях список наследуется без изменений.

$$T_0 = \langle\ \exists x\varphi(x), \psi_1 \to \psi_2, \Gamma \mid \forall y\chi(y), \Delta\ \rangle, \qquad L_0 = \{c', c''\}$$

$$\Big\downarrow (L\exists)$$

$$T_1 = \langle\ \psi_1 \to \psi_2, \Gamma, \varphi(c_1) \mid \forall y\chi(y), \Delta\ \rangle, \qquad L_1 = \{c', c'', c_1\}$$

$$\Big\downarrow (R\forall)$$

$$T_2 = \langle\ \psi_1 \to \psi_2, \Gamma, \varphi(c_1) \mid \Delta, \chi(c_2)\ \rangle, \qquad L_2 = \{c', c'', c_1, c_2\}$$

$$\swarrow \quad (L \to) \quad \searrow$$

$$T_3 = \langle\ \Gamma, \varphi(c_1), \psi_2 \mid \Delta, \chi(c_2)\ \rangle \qquad T_4 = \langle \Gamma, \varphi(c_1) \mid \Delta, \chi(c_2), \psi_1\ \rangle$$

$$L_3 = \{c', c'', c_1, c_2\} \qquad L_4 = \{c', c'', c_1, c_2\}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

3). С каждой таблицей T ассоциирован **непустой** список использованных констант L_T. Вначале в этот список включаются все константы, содержащиеся в таблице T_0 (если в T_0 констант нет, то полагаем $L_{T_0} = \{c_0\}$). После применения правил $(L\exists)$ и $(R\forall)$ в список порожденной таблицы добавляется «свежая константа». В остальных случаях список наследуется без изменений.

$$T_0 = \langle\ \exists x\varphi(x), \psi_1 \to \psi_2, \Gamma \mid \forall y\chi(y), \Delta\ \rangle, \qquad L_0 = \{c', c''\}$$

$$\Big\downarrow (L\exists)$$

$$T_1 = \langle\ \psi_1 \to \psi_2, \Gamma, \varphi(c_1) \mid \forall y\chi(y), \Delta\ \rangle, \qquad L_1 = \{c', c'', c_1\}$$

$$\Big\downarrow (R\forall)$$

$$T_2 = \langle\ \psi_1 \to \psi_2, \Gamma, \varphi(c_1) \mid \Delta, \chi(c_2)\ \rangle, \qquad L_2 = \{c', c'', c_1, c_2\}$$

$$\swarrow \quad (L\to) \quad \searrow$$

$$T_3 = \langle\ \Gamma, \varphi(c_1), \psi_2 \mid \Delta, \chi(c_2)\ \rangle \qquad T_4 = \langle \Gamma, \varphi(c_1) \mid \Delta, \chi(c_2), \psi_1\ \rangle$$

$$L_3 = \{c', c'', c_1, c_2\} \qquad L_4 = \{c', c'', c_1, c_2\}$$

и. т. д.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

4). В случае применения к таблице T правил $(L\forall)$ и $(R\exists)$ в качестве подстановочных термов используются все константы из списка L_T.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

4). В случае применения к таблице T правил $(L\forall)$ и $(R\exists)$ в качестве подстановочных термов используются все константы из списка L_T.

$$T_0 = \langle\ \forall x\varphi(x), \Gamma \mid \exists y\chi(y), \Delta\ \rangle, \qquad L_0 = \{c_1, c_2\}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

4). В случае применения к таблице T правил $(L\forall)$ и $(R\exists)$ в качестве подстановочных термов используются все константы из списка L_T.

$$T_0 = \langle\ \forall x\varphi(x), \Gamma \mid \exists y\chi(y), \Delta\ \rangle, \qquad L_0 = \{c_1, c_2\}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

4). В случае применения к таблице T правил $(L\forall)$ и $(R\exists)$ в качестве подстановочных термов используются все константы из списка L_T.

$$T_0 = \langle\ \forall x\varphi(x), \Gamma \mid \exists y\chi(y), \Delta\ \rangle, \qquad L_0 = \{c_1, c_2\}$$

$$\big\downarrow (L\forall)$$

$$T_1 = \langle\ \Gamma, \forall x\varphi(x), \varphi(c_1), \varphi(c_2) \mid \exists y\chi(y), \Delta\ \rangle, \qquad L_1 = \{c_1, c_2\}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

4). В случае применения к таблице T правил $(L\forall)$ и $(R\exists)$ в качестве подстановочных термов используются все константы из списка L_T.

$$T_0 = \langle\, \forall x\varphi(x), \Gamma \mid \exists y\chi(y), \Delta \,\rangle, \qquad L_0 = \{c_1, c_2\}$$

$$\big\downarrow (L\forall)$$

$$T_1 = \langle\, \Gamma, \forall x\varphi(x), \varphi(c_1), \varphi(c_2) \mid \color{red}{\exists y\chi(y)}, \Delta \,\rangle, \qquad L_1 = \{c_1, c_2\}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

4). В случае применения к таблице T правил $(L\forall)$ и $(R\exists)$ в качестве подстановочных термов используются все константы из списка L_T.

$$T_0 = \langle\ \forall x\varphi(x), \Gamma \mid \exists y\chi(y), \Delta\ \rangle, \qquad L_0 = \{c_1, c_2\}$$

$$\Big\downarrow (L\forall)$$

$$T_1 = \langle\ \Gamma, \forall x\varphi(x), \varphi(c_1), \varphi(c_2) \mid \exists y\chi(y), \Delta\ \rangle, \qquad L_1 = \{c_1, c_2\}$$

$$\Big\downarrow (R\exists)$$

$$T_2 = \langle\ \Gamma, \forall x\varphi(x), \varphi(c_1), \varphi(c_2) \mid \Delta, \exists x\chi(x), \chi(c_1), \chi(c_2)\ \rangle,$$

$$L_2 = \{c_1, c_2\}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

4). В случае применения к таблице T правил $(L\forall)$ и $(R\exists)$ в качестве подстановочных термов используются все константы из списка L_T.

$$
\begin{array}{c}
T_0 = \langle\ \forall x\varphi(x), \Gamma \mid \exists y\chi(y), \Delta\ \rangle, \qquad L_0 = \{c_1, c_2\} \\
\big\downarrow (L\forall) \\
T_1 = \langle\ \Gamma, \forall x\varphi(x), \varphi(c_1), \varphi(c_2) \mid \exists y\chi(y), \Delta\ \rangle, \qquad L_1 = \{c_1, c_2\} \\
\big\downarrow (R\exists) \\
T_2 = \langle\ \Gamma, \forall x\varphi(x), \varphi(c_1), \varphi(c_2) \mid \Delta, \exists x\chi(x), \chi(c_1), \chi(c_2)\ \rangle,
\end{array}
$$

$$L_2 = \{c_1, c_2\}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

4). В случае применения к таблице T правил $(L\forall)$ и $(R\exists)$ в качестве подстановочных термов используются все константы из списка L_T.

$$
\begin{array}{cc}
T_0 = \langle\ \forall x\varphi(x), \Gamma \mid \exists y\chi(y), \Delta\ \rangle, & L_0 = \{c_1, c_2\} \\
\big\downarrow (L\forall) & \\
T_1 = \langle\ \Gamma, \forall x\varphi(x), \varphi(c_1), \varphi(c_2) \mid \exists y\chi(y), \Delta\ \rangle, & L_1 = \{c_1, c_2\} \\
\big\downarrow (R\exists) & \\
T_2 = \langle\ \Gamma, \forall x\varphi(x), \varphi(c_1), \varphi(c_2) \mid \Delta, \exists x\chi(x), \chi(c_1), \chi(c_2)\ \rangle, & \\
 & L_2 = \{c_1, c_2\}
\end{array}
$$

и. т. д.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Предположим, что указанная стратегия не приводит к построению успешного вывода для невыполнимой таблицы $T_0 = \langle\ \Gamma_0 \mid \Delta_0\ \rangle$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Предположим, что указанная стратегия не приводит к построению успешного вывода для невыполнимой таблицы $T_0 = \langle\, \Gamma_0 \mid \Delta_0 \,\rangle$. Тогда в дереве вывода либо существует бесконечная ветвь, не содержащая закрытых таблиц

$$T_0 \longrightarrow T_1 \longrightarrow T_2 \longrightarrow \dots \longrightarrow T_n \longrightarrow T_{n+1} \longrightarrow \dots$$

либо существует ветвь, оканчивающаяся незакрытой атомарной таблицей

$$T_0 \longrightarrow T_1 \longrightarrow T_2 \longrightarrow \dots \longrightarrow T_{atom}$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Предположим, что указанная стратегия не приводит к построению успешного вывода для невыполнимой таблицы $T_0 = \langle\, \Gamma_0 \mid \Delta_0 \,\rangle$. Тогда в дереве вывода либо существует бесконечная ветвь, не содержащая закрытых таблиц

$$T_0 \longrightarrow T_1 \longrightarrow T_2 \longrightarrow \dots \longrightarrow T_n \longrightarrow T_{n+1} \longrightarrow \dots$$

либо существует ветвь, оканчивающаяся незакрытой атомарной таблицей

$$T_0 \longrightarrow T_1 \longrightarrow T_2 \longrightarrow \dots \longrightarrow T_{atom} \longrightarrow T_{atom} \longrightarrow T_{atom} \longrightarrow \dots$$

Будем полагать, что в последнем случае последовательность таблиц также бесконечна.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

На основе бесконечной последовательности незакрытых таблиц $T_0, T_1, \ldots, T_n, T_{n+1}, \ldots$, каждая из которых имеет вид $T_n = \langle\ \Gamma_n \mid \Delta_n\ \rangle$ построим три множества:

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

На основе бесконечной последовательности незакрытых таблиц $T_0, T_1, \ldots, T_n, T_{n+1}, \ldots$, каждая из которых имеет вид $T_n = \langle\ \Gamma_n \mid \Delta_n\ \rangle$ построим три множества:

1). $\Gamma_\omega = \bigcup\limits_{i=0}^{\infty} \Gamma_i$ множество всех формул из левых частей таблиц,

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

На основе бесконечной последовательности незакрытых таблиц $T_0, T_1, \ldots, T_n, T_{n+1}, \ldots$, каждая из которых имеет вид $T_n = \langle\ \Gamma_n \mid \Delta_n\ \rangle$ построим три множества:

1). $\Gamma_\omega = \bigcup\limits_{i=0}^{\infty} \Gamma_i$ множество всех формул из левых частей таблиц,

2). $\Delta_\omega = \bigcup\limits_{i=0}^{\infty} \Delta_i$ множество всех формул из правых частей таблиц,

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

На основе бесконечной последовательности незакрытых таблиц $T_0, T_1, \ldots, T_n, T_{n+1}, \ldots$, каждая из которых имеет вид $T_n = \langle\ \Gamma_n \mid \Delta_n\ \rangle$ построим три множества:

1). $\Gamma_\omega = \bigcup_{i=0}^{\infty} \Gamma_i$ — множество всех формул из левых частей таблиц,

2). $\Delta_\omega = \bigcup_{i=0}^{\infty} \Delta_i$ — множество всех формул из правых частей таблиц,

3). $L_\omega = \bigcup_{i=0}^{\infty} L_{T_i}$ — множество всех констант из списков констант, ассоциированных с таблицами из бесконечной ветви.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

На основе бесконечной последовательности незакрытых таблиц $T_0, T_1, \ldots, T_n, T_{n+1}, \ldots,$ каждая из которых имеет вид $T_n = \langle\ \Gamma_n \mid \Delta_n\ \rangle$ построим три множества:

1). $\Gamma_\omega = \bigcup\limits_{i=0}^{\infty} \Gamma_i$ множество всех формул из левых частей таблиц,

2). $\Delta_\omega = \bigcup\limits_{i=0}^{\infty} \Delta_i$ множество всех формул из правых частей таблиц,

3). $L_\omega = \bigcup\limits_{i=0}^{\infty} L_{T_i}$ множество всех констант из списков констант, ассоциированных с таблицами из бесконечной ветви.

Используя множества Γ_ω, Δ_ω, L_ω, построим интерпретацию I_ω, в которой будет выполняться каждая таблица последовательности.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

$$I_\omega = \langle D_I, \overline{Const}, \overline{Pred} \rangle$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

$$I_\omega = \langle D_I, \overline{Const}, \overline{Pred} \rangle$$

$D_I = L_\omega$

предметная область состоит из всех константных символов, входящих в состав формул из Γ_ω и Δ_ω,

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

$$I_\omega = \langle D_I, \overline{Const}, \overline{Pred} \rangle$$

$D_I = L_\omega$

предметная область состоит из всех константных символов, входящих в состав формул из Γ_ω и Δ_ω,

$\overline{Const} \; : \; Const \; \rightarrow \; L_\omega$

значением $\overline{\mathbf{c}}$ каждой константы c (как символа из алфавита) является ее собственное изображение c (как элемент множества L_ω), т. е. $\overline{\mathbf{c}} = c$,

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

$$I_\omega = \langle D_I, \overline{Const}, \overline{Pred} \rangle$$

$D_I = L_\omega$

предметная область состоит из всех константных символов, входящих в состав формул из Γ_ω и Δ_ω,

$\overline{Const} \ : \ Const \ \rightarrow \ L_\omega$

значением $\bar{\mathbf{c}}$ каждой константы c (как символа из алфавита) является ее собственное изображение c (как элемент множества L_ω), т. е. $\bar{\mathbf{c}} = c$,

$\overline{Pred} \ : \ Pred \rightarrow (L_\omega{}^n \ \rightarrow \ \{\mathbf{true}, \mathbf{false}\})$

для каждого предикатного символа $P^{(n)}$ и набора элементов $c_{i_1}, \dots, c_{i_n}$ из L_ω

$$\overline{\mathbf{P}}(c_{i_1}, \dots, c_{i_n}) = \mathbf{true} \iff P(c_{i_1}, \dots, c_{i_n}) \in \Gamma_\omega.$$

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Покажем, что

любая формула φ, $\varphi \in \Gamma_\omega$, выполнима в интерпретации I_ω,

любая формула ψ, $\psi \in \Delta_\omega$, невыполнима в интерпретации I_ω.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Покажем, что

любая формула φ, $\varphi \in \Gamma_\omega$, выполнима в интерпретации I_ω,

любая формула ψ, $\psi \in \Delta_\omega$, невыполнима в интерпретации I_ω.

Доказательство проведем при помощи индукции по числу логических операций (связок и кванторов) в формуле.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Базис индукции

φ, ψ — атомарные формулы.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Базис индукции

φ, ψ — атомарные формулы.

Во всех таблицах T_i содержатся только замкнутые формулы (почему ?). Поэтому формулы φ, ψ имеют вид $P(c_{i_1}, \ldots, c_{i_n})$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Базис индукции

φ, ψ — атомарные формулы.

Во всех таблицах T_i содержатся только замкнутые формулы (почему ?). Поэтому формулы φ, ψ имеют вид $P(c_{i_1}, \ldots, c_{i_n})$.

Если $\varphi \in \Gamma_\omega$, то $\overline{\mathbf{P}}(c_{i_1}, \ldots, c_{i_n}) = \mathbf{true}$, и поэтому $I_\omega \models \varphi$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Базис индукции

φ, ψ — атомарные формулы.

Во всех таблицах T_i содержатся только замкнутые формулы (почему ?). Поэтому формулы φ, ψ имеют вид $P(c_{i_1}, \dots, c_{i_n})$.

Если $\varphi \in \Gamma_\omega$, то $\overline{\mathbf{P}}(c_{i_1}, \dots, c_{i_n}) = \mathbf{true}$, и поэтому $I_\omega \models \varphi$.

Если $\psi \in \Delta_\omega$, то $\psi \notin \Gamma_\omega$ (почему ?).

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Базис индукции

φ, ψ — атомарные формулы.

Во всех таблицах T_i содержатся только замкнутые формулы (почему ?). Поэтому формулы φ, ψ имеют вид $P(c_{i_1}, \ldots, c_{i_n})$.

Если $\varphi \in \Gamma_\omega$, то $\overline{\mathbf{P}}(c_{i_1}, \ldots, c_{i_n}) = \textbf{true}$, и поэтому $I_\omega \models \varphi$.

Если $\psi \in \Delta_\omega$, то $\psi \notin \Gamma_\omega$ (почему ?).

В противном случае существовала бы такая таблица $T_i = \langle \Gamma_i | \Delta_i \rangle$, что $\psi \in \Gamma_i$ и $\psi \in \Delta_i$, вопреки предположению о том, что все таблицы T_i незакрытые

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Базис индукции

φ, ψ — атомарные формулы.

Во всех таблицах T_i содержатся только замкнутые формулы (почему ?). Поэтому формулы φ, ψ имеют вид $P(c_{i_1}, \dots, c_{i_n})$.

Если $\varphi \in \Gamma_\omega$, то $\overline{\mathbf{P}}(c_{i_1}, \dots, c_{i_n}) = \mathbf{true}$, и поэтому $I_\omega \models \varphi$.

Если $\psi \in \Delta_\omega$, то $\psi \notin \Gamma_\omega$ (почему ?).
Значит, $\overline{\mathbf{P}}(c_{i_1}, \dots, c_{i_n}) = \mathbf{false}$, и поэтому $I_\omega \not\models \psi$.

Полнота табличного вывода

Индуктивный переход

Предположим, что утверждение верно для всех формул, имеющих не более *n* связок и кванторов. Рассмотрим формулы φ, ψ, содержащие $n + 1$ связок и кванторов.

Полнота табличного вывода

Индуктивный переход

Предположим, что утверждение верно для всех формул, имеющих не более n связок и кванторов. Рассмотрим формулы φ, ψ, содержащие $n+1$ связок и кванторов.

Если $\varphi = \varphi_1 \to \varphi_2 \in \Gamma_\omega$, то есть такая таблица $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\varphi_1 \to \varphi_2 \in \Gamma_i$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Предположим, что утверждение верно для всех формул, имеющих не более n связок и кванторов. Рассмотрим формулы φ, ψ, содержащие $n+1$ связок и кванторов.

Если $\varphi = \varphi_1 \to \varphi_2 \in \Gamma_\omega$, то есть такая таблица $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\varphi_1 \to \varphi_2 \in \Gamma_i$.
Согласно описанию применяемой стратегии построения табличного вывода, существует такая таблица $T_j = \langle\ \Gamma_j \mid \Delta_j\ \rangle$, $j \geq i$, что $\varphi_1 \to \varphi_2$ — первая формула в списке формул Γ_j.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Предположим, что утверждение верно для всех формул, имеющих не более n связок и кванторов. Рассмотрим формулы φ, ψ, содержащие $n+1$ связок и кванторов.

Если $\varphi = \varphi_1 \to \varphi_2 \in \Gamma_\omega$, то есть такая таблица $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\varphi_1 \to \varphi_2 \in \Gamma_i$.
Согласно описанию применяемой стратегии построения табличного вывода, существует такая таблица $T_j = \langle\ \Gamma_j \mid \Delta_j\ \rangle$, $j \geq i$, что $\varphi_1 \to \varphi_2$ — первая формула в списке формул Γ_j.
Поэтому либо $\varphi_1 \in \Delta_{j+1}$, либо $\varphi_2 \in \Gamma_{j+1}$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Предположим, что утверждение верно для всех формул, имеющих не более n связок и кванторов. Рассмотрим формулы φ, ψ, содержащие $n+1$ связок и кванторов.

Если $\varphi = \varphi_1 \to \varphi_2 \in \Gamma_\omega$, то есть такая таблица $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\varphi_1 \to \varphi_2 \in \Gamma_i$.
Согласно описанию применяемой стратегии построения табличного вывода, существует такая таблица $T_j = \langle\ \Gamma_j \mid \Delta_j\ \rangle$, $j \geq i$, что $\varphi_1 \to \varphi_2$ — первая формула в списке формул Γ_j.
Поэтому либо $\varphi_1 \in \Delta_{j+1}$, либо $\varphi_2 \in \Gamma_{j+1}$.

В первом случае, согласно индуктивному предполжению, $I_\omega \not\models \varphi_1$, и поэтому $I_\omega \models \varphi$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Предположим, что утверждение верно для всех формул, имеющих не более n связок и кванторов. Рассмотрим формулы φ, ψ, содержащие $n+1$ связок и кванторов.

Если $\varphi = \varphi_1 \to \varphi_2 \in \Gamma_\omega$, то есть такая таблица $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\varphi_1 \to \varphi_2 \in \Gamma_i$.
Согласно описанию применяемой стратегии построения табличного вывода, существует такая таблица $T_j = \langle\ \Gamma_j \mid \Delta_j\ \rangle$, $j \geq i$, что $\varphi_1 \to \varphi_2$ — первая формула в списке формул Γ_j.
Поэтому либо $\varphi_1 \in \Delta_{j+1}$, либо $\varphi_2 \in \Gamma_{j+1}$.

В первом случае, согласно индуктивному предполжению, $I_\omega \not\models \varphi_1$, и поэтому $I_\omega \models \varphi$.

Во втором случае, согласно индуктивному предполжению, $I_\omega \models \varphi_2$, и поэтому $I_\omega \models \varphi$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Предположим, что утверждение верно для всех формул, имеющих не более n связок и кванторов. Рассмотрим формулы φ, ψ, содержащие $n+1$ связок и кванторов.

Если $\varphi = \varphi_1 \to \varphi_2 \in \Gamma_\omega$, то есть такая таблица $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\varphi_1 \to \varphi_2 \in \Gamma_i$.
Согласно описанию применяемой стратегии построения табличного вывода, существует такая таблица $T_j = \langle\ \Gamma_j \mid \Delta_j\ \rangle$, $j \geq i$, что $\varphi_1 \to \varphi_2$ — первая формула в списке формул Γ_j.
Поэтому либо $\varphi_1 \in \Delta_{j+1}$, либо $\varphi_2 \in \Gamma_{j+1}$.

В первом случае, согласно индуктивному предполжению, $I_\omega \not\models \varphi_1$, и поэтому $I_\omega \models \varphi$.

Во втором случае, согласно индуктивному предполжению, $I_\omega \models \varphi_2$, и поэтому $I_\omega \models \varphi$.

Итак, в обоих случаях получаем $I_\omega \models \varphi_1 \to \varphi_2$.

Полнота табличного вывода

Индуктивный переход

Аналогичные рассуждения применимы и для других связок.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Если $\varphi = \forall x \varphi_1(x) \in \Gamma_\omega$, то формула $\forall x \varphi_1(x)$ бесконечно часто встречается в левых частях таблиц $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$ (почему ?).

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Если $\varphi = \forall x \varphi_1(x) \in \Gamma_\omega$, то формула $\forall x \varphi_1(x)$ бесконечно часто встречается в левых частях таблиц $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$ (почему ?). Поэтому правило $(L\forall)$ применяется к формуле $\forall x \varphi_1(x)$ бесконечно часто.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Если $\varphi = \forall x \varphi_1(x) \in \Gamma_\omega$, то формула $\forall x \varphi_1(x)$ бесконечно часто встречается в левых частях таблиц $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$ (почему ?).
Поэтому правило $(L\forall)$ применяется к формуле $\forall x \varphi_1(x)$ бесконечно часто.
Тогда, согласно описанию применяемой стратегии построения табличного вывода, для любой константы c, $c \in L_\omega$, существует такая таблица $T_j = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\varphi_1(c) \in \Gamma_j$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Если $\varphi = \forall x \varphi_1(x) \in \Gamma_\omega$, то формула $\forall x \varphi_1(x)$ бесконечно часто встречается в левых частях таблиц $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$ (почему ?). Поэтому правило $(L\forall)$ применяется к формуле $\forall x \varphi_1(x)$ бесконечно часто.
Тогда, согласно описанию применяемой стратегии построения табличного вывода, для любой константы c, $c \in L_\omega$, существует такая таблица $T_j = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\varphi_1(c) \in \Gamma_j$.

Значит, согласно индуктивному предположению, $I_\omega \models \varphi_1(c)$ для любой константы c, $c \in L_\omega$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Если $\varphi = \forall x \varphi_1(x) \in \Gamma_\omega$, то формула $\forall x \varphi_1(x)$ бесконечно часто встречается в левых частях таблиц $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$ (почему ?).
Поэтому правило $(L\forall)$ применяется к формуле $\forall x \varphi_1(x)$ бесконечно часто.
Тогда, согласно описанию применяемой стратегии построения табличного вывода, для любой константы c, $c \in L_\omega$, существует такая таблица $T_j = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\varphi_1(c) \in \Gamma_j$.

Значит, согласно индуктивному предположению, $I_\omega \models \varphi_1(c)$ для любой константы c, $c \in L_\omega$.

Так как $D_I = L_\omega$, приходим к заключению о том, что $I_\omega \models \forall x \varphi_1(x)$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Если $\varphi = \forall x \varphi_1(x) \in \Delta_\omega$, то есть такая таблица $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\forall x \varphi_1(x) \in \Delta_i$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Если $\varphi = \forall x \varphi_1(x) \in \Delta_\omega$, то есть такая таблица $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\forall x \varphi_1(x) \in \Delta_i$.
Согласно описанию применяемой стратегии построения табличного вывода, существует такая таблица $T_j = \langle\ \Gamma_j \mid \Delta_j\ \rangle$, $j \geq i$, что $\forall x \varphi_1(x)$ — первая формула в списке формул Δ_j.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Если $\varphi = \forall x \varphi_1(x) \in \Delta_\omega$, то есть такая таблица $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\forall x \varphi_1(x) \in \Delta_i$.
Согласно описанию применяемой стратегии построения табличного вывода, существует такая таблица $T_j = \langle\ \Gamma_j \mid \Delta_j\ \rangle$, $j \geq i$, что $\forall x \varphi_1(x)$ — первая формула в списке формул Δ_j.
Поэтому существует такая константа c, $c \in L_\omega$, что $\varphi_1(c) \in \Delta_{j+1}$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Если $\varphi = \forall x \varphi_1(x) \in \Delta_\omega$, то есть такая таблица $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\forall x \varphi_1(x) \in \Delta_i$.
Согласно описанию применяемой стратегии построения табличного вывода, существует такая таблица $T_j = \langle\ \Gamma_j \mid \Delta_j\ \rangle$, $j \geq i$, что $\forall x \varphi_1(x)$ — первая формула в списке формул Δ_j.
Поэтому существует такая константа c, $c \in L_\omega$, что $\varphi_1(c) \in \Delta_{j+1}$.

Согласно индуктивному предположению, это означает, что $I_\omega \not\models \varphi_1(c)$, и поэтому $I_\omega \not\models \forall x \varphi_1(x)$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Индуктивный переход

Если $\varphi = \forall x \varphi_1(x) \in \Delta_\omega$, то есть такая таблица $T_i = \langle\ \Gamma_i \mid \Delta_i\ \rangle$, что $\forall x \varphi_1(x) \in \Delta_i$.
Согласно описанию применяемой стратегии построения табличного вывода, существует такая таблица $T_j = \langle\ \Gamma_j \mid \Delta_j\ \rangle$, $j \geq i$, что $\forall x \varphi_1(x)$ — первая формула в списке формул Δ_j.
Поэтому существует такая константа c, $c \in L_\omega$, что $\varphi_1(c) \in \Delta_{j+1}$.

Согласно индуктивному предполжению, это означает, что $I_\omega \not\models \varphi_1(c)$, и поэтому $I_\omega \not\models \forall x \varphi_1(x)$.

Аналогичные рассуждения применимы и для формул с квантором $\exists$.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Таким образом,

любая формула φ, $\varphi \in \Gamma_\omega$, выполнима в интерпретации I_ω,

любая формула ψ, $\psi \in \Delta_\omega$, невыполнима в интерпретации I_ω.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Таким образом,

любая формула φ, $\varphi \in \Gamma_\omega$, выполнима в интерпретации I_ω,

любая формула ψ, $\psi \in \Delta_\omega$, невыполнима в интерпретации I_ω.

Поскольку, $\Gamma_0 \subseteq \Gamma_\omega$ и $\Delta_0 \subseteq \Delta_\omega$, приходим к заключению о том, что

$$I_\omega \models \Gamma_0, \quad I_\omega \not\models \Delta_0.$$

Это означает, что таблица T_0 выполнима в интерпретации I_ω вопреки условию о невыполнимости T_0.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Таким образом,

любая формула φ, $\varphi \in \Gamma_\omega$, выполнима в интерпретации I_ω,

любая формула ψ, $\psi \in \Delta_\omega$, невыполнима в интерпретации I_ω.

Поскольку, $\Gamma_0 \subseteq \Gamma_\omega$ и $\Delta_0 \subseteq \Delta_\omega$, приходим к заключению о том, что

$$I_\omega \models \Gamma_0, \quad I_\omega \not\models \Delta_0.$$

Это означает, что таблица T_0 выполнима в интерпретации I_ω вопреки условию о невыполнимости T_0.

Источник полученного противоречия — предположение о невозможности построить успешный вывод, руководствуясь описанной стратегией. Значит, предложенная стратегия позволяет построить успешный вывод для всякой невыполнимой таблицы.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Это было упрощенное доказательство теоремы полноты. А насколько существенны те упрощения, которые были объявлены в начале доказательства? А именно,

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Это было упрощенное доказательство теоремы полноты. А насколько существенны те упрощения, которые были объявлены в начале доказательства? А именно,

- Какие изменения нужно внести в структуру данных, представляющих таблицы, чтобы доказательство можно было распространить и на таблицы, содержащие бесконечные множества формул?

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Это было упрощенное доказательство теоремы полноты. А насколько существенны те упрощения, которые были объявлены в начале доказательства? А именно,

- Какие изменения нужно внести в структуру данных, представляющих таблицы, чтобы доказательство можно было распространить и на таблицы, содержащие бесконечные множества формул?
- Какие изменения нужно внести в стратегию построения табличного вывода, чтобы построить успешный вывод для невыполнимой таблицы, формулы которой содержат сложные термы?

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Это было упрощенное доказательство теоремы полноты. А насколько существенны те упрощения, которые были объявлены в начале доказательства? А именно,

- Какие изменения нужно внести в структуру данных, представляющих таблицы, чтобы доказательство можно было распространить и на таблицы, содержащие бесконечные множества формул?
- Какие изменения нужно внести в стратегию построения табличного вывода, чтобы построить успешный вывод для невыполнимой таблицы, формулы которой содержат сложные термы?
- Какие изменения нужно внести в определение интерпретации I_ω, чтобы доказательство можно было применить к таблицам, содержащим незамкнутые формулы?

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Теорема Геделя (о полноте)

Если формула φ общезначима, то существует успешный табличный вывод для таблицы $T_\varphi = \langle \emptyset | \varphi \rangle$.

Полнота табличного вывода

Теорема Геделя (о полноте)

Если формула φ общезначима, то существует успешный табличный вывод для таблицы $T_\varphi = \langle \emptyset | \varphi \rangle$.

Доказательство

Следует из теоремы полноты.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Теорема Геделя (о полноте)

Если формула φ общезначима, то существует успешный табличный вывод для таблицы $T_\varphi = \langle \emptyset | \varphi \rangle$.

Доказательство

Следует из теоремы полноты.

Итак, формула φ общезначима

$$\Longleftrightarrow$$

для таблицы T_φ существует успешный вывод.

ПОЛНОТА ТАБЛИЧНОГО ВЫВОДА

Теорема Геделя (о полноте)

Если формула φ общезначима, то существует успешный табличный вывод для таблицы $T_\varphi = \langle \emptyset | \varphi \rangle$.

Доказательство

Следует из теоремы полноты.

Итак, формула φ общезначима

$$\Longleftrightarrow$$

для таблицы T_φ существует успешный вывод.

И в доказательстве теоремы полноты показано, **как** построить этот вывод.

ТЕОРЕМА ЛЕВЕНГЕЙМА-СКОЛЕМА

Теорема

Формула φ выполнима $\iff$ φ имеет модель с конечной или счетно-бесконечной предметной областью.

ТЕОРЕМА ЛЕВЕНГЕЙМА-СКОЛЕМА

Теорема

Формула φ выполнима $\iff$ φ имеет модель с конечной или счетно-бесконечной предметной областью.

Доказательство

Если φ выполнима, то для таблицы $T^*_\varphi = \langle\ \varphi \mid \emptyset\ \rangle$ нельзя построить успешный вывод (см. теорему корректности).

ТЕОРЕМА ЛЕВЕНГЕЙМА-СКОЛЕМА

Теорема

Формула φ выполнима $\iff$ φ имеет модель с конечной или счетно-бесконечной предметной областью.

Доказательство

Если φ выполнима, то для таблицы $T_\varphi^* = \langle\ \varphi \mid \emptyset\ \rangle$ нельзя построить успешный вывод . Применим стратегию из доказательства теоремы полноты. Т. к. T_φ^* выполнима, получим дерево с ветвью, в которой нет закрытых таблиц.

$$T_\varphi^* = T_0 \longrightarrow T_1 \longrightarrow T_2 \longrightarrow \dots \longrightarrow T_n \longrightarrow T_{n+1} \longrightarrow \dots$$

ТЕОРЕМА ЛЕВЕНГЕЙМА-СКОЛЕМА

Теорема

Формула φ выполнима $\iff$ φ имеет модель с конечной или счетно-бесконечной предметной областью.

Доказательство

Если φ выполнима, то для таблицы $T_\varphi^* = \langle\ \varphi \mid \emptyset\ \rangle$ нельзя построить успешный вывод . Применим стратегию из доказательства теоремы полноты. Т. к. T_φ^* выполнима, получим дерево с ветвью, в которой нет закрытых таблиц.

$$T_\varphi^* = T_0 \longrightarrow T_1 \longrightarrow T_2 \longrightarrow \dots \longrightarrow T_n \longrightarrow T_{n+1} \longrightarrow \dots$$

Для этой ветви построим интерпретацию I_ω , в которой

ТЕОРЕМА ЛЕВЕНГЕЙМА-СКОЛЕМА

Теорема

Формула φ выполнима $\iff$ φ имеет модель с конечной или счетно-бесконечной предметной областью.

Доказательство

Если φ выполнима, то для таблицы $T^*_\varphi = \langle\ \varphi \mid \emptyset\ \rangle$ нельзя построить успешный вывод . Применим стратегию из доказательства теоремы полноты. Т. к. T^*_φ выполнима, получим дерево с ветвью, в которой нет закрытых таблиц.

$$T^*_\varphi = T_0 \longrightarrow T_1 \longrightarrow T_2 \longrightarrow \dots \longrightarrow T_n \longrightarrow T_{n+1} \longrightarrow \dots$$

Для этой ветви построим интерпретацию I_ω , в которой

- $D_i = L_\omega = \{c_1, c_2, \dots\}$ — конечное или счетно-бесконечное множество констант из таблиц последовательности;

ТЕОРЕМА ЛЕВЕНГЕЙМА-СКОЛЕМА

Теорема

Формула φ выполнима $\iff$ φ имеет модель с конечной или счетно-бесконечной предметной областью.

Доказательство

Если φ выполнима, то для таблицы $T_\varphi^* = \langle\, \varphi \mid \emptyset \,\rangle$ нельзя построить успешный вывод . Применим стратегию из доказательства теоремы полноты. Т. к. T_φ^* выполнима, получим дерево с ветвью, в которой нет закрытых таблиц.

$$T_\varphi^* = T_0 \longrightarrow T_1 \longrightarrow T_2 \longrightarrow \dots \longrightarrow T_n \longrightarrow T_{n+1} \longrightarrow \dots$$

Для этой ветви построим интерпретацию I_ω , в которой

- $D_i = L_\omega = \{c_1, c_2, \dots\}$ — конечное или счетно-бесконечное множество констант из таблиц последовательности;
- выполнимы все таблицы T_i (в т.ч. и T_φ^*).

ТЕОРЕМА КОМПАКТНОСТИ МАЛЬЦЕВА

Теорема

$\Gamma \models \varphi \iff$ существует такое <u>конечное</u> подмножество Γ', $\Gamma' \subseteq \Gamma$, что $\Gamma' \models \varphi$.

ТЕОРЕМА КОМПАКТНОСТИ МАЛЬЦЕВА

Теорема

$\Gamma \models \varphi \iff$ **существует такое <u>конечное</u> подмножество Γ', $\Gamma' \subseteq \Gamma$, что $\Gamma' \models \varphi$.**

Доказательство

1. $\Gamma \models \varphi \iff$ таблица $T = \langle \Gamma | \varphi \rangle$ невыполнима (почему?)

ТЕОРЕМА КОМПАКТНОСТИ МАЛЬЦЕВА

Теорема

$\Gamma \models \varphi \iff$ **существует такое <u>конечное</u> подмножество Γ', $\Gamma' \subseteq \Gamma$, что $\Gamma' \models \varphi$.**

Доказательство

1. $\Gamma \models \varphi \iff$ таблица $T = \langle \Gamma | \varphi \rangle$ невыполнима (почему?)
2. Таблица $T = \langle \Gamma | \varphi \rangle$ невыполнима $\iff$ существует успешный вывод для T (почему?)

ТЕОРЕМА КОМПАКТНОСТИ МАЛЬЦЕВА

Теорема

$\Gamma \models \varphi \iff$ **существует такое <u>конечное</u> подмножество Γ', $\Gamma' \subseteq \Gamma$, что $\Gamma' \models \varphi$.**

Доказательство

1. $\Gamma \models \varphi \iff$ таблица $T = \langle \Gamma | \varphi \rangle$ невыполнима (почему?)
2. Таблица $T = \langle \Gamma | \varphi \rangle$ невыполнима $\iff$ существует успешный вывод для T (почему?)
3. успешный вывод — это конечное дерево, и поэтому существует лишь конечое множество формул Γ', $\Gamma' \subseteq \Gamma$, к которым применяются правила вывода.

ТЕОРЕМА КОМПАКТНОСТИ МАЛЬЦЕВА

Теорема

$\Gamma \models \varphi \iff$ **существует такое <u>конечное</u> подмножество Γ', $\Gamma' \subseteq \Gamma$, что $\Gamma' \models \varphi$.**

Доказательство

1. $\Gamma \models \varphi \iff$ таблица $T = \langle \Gamma | \varphi \rangle$ невыполнима (почему?)
2. Таблица $T = \langle \Gamma | \varphi \rangle$ невыполнима $\iff$ существует успешный вывод для T (почему?)
3. успешный вывод — это конечное дерево, и поэтому существует лишь конечое множество формул Γ', $\Gamma' \subseteq \Gamma$, к которым применяются правила вывода.
4. Но тогда для таблицы $T = \langle \Gamma' | \varphi \rangle$ существует точно такой же успешный вывод.

ТЕОРЕМА КОМПАКТНОСТИ МАЛЬЦЕВА

Теорема

$\Gamma \models \varphi \iff$ **существует такое <u>конечное</u> подмножество Γ', $\Gamma' \subseteq \Gamma$, что $\Gamma' \models \varphi$.**

Доказательство

1. $\Gamma \models \varphi \iff$ таблица $T = \langle \Gamma | \varphi \rangle$ невыполнима (почему?)
2. Таблица $T = \langle \Gamma | \varphi \rangle$ невыполнима $\iff$ существует успешный вывод для T (почему?)
3. успешный вывод — это конечное дерево, и поэтому существует лишь конечое множество формул Γ', $\Gamma' \subseteq \Gamma$, к которым применяются правила вывода.
4. Но тогда для таблицы $T = \langle \Gamma' | \varphi \rangle$ существует точно такой же успешный вывод.
5. Значит, $\Gamma' \models \varphi$.

Автоматическое доказательство теорем

Стратегия построения табличного вывода, использованная в доказательстве теоремы полноты — это алгоритм построения доказательства истинности утверждений, выраженных формулами логики предикатов.

Автоматическое доказательство теорем

Стратегия построения табличного вывода, использованная в доказательстве теоремы полноты — это алгоритм построения доказательства истинности утверждений, выраженных формулами логики предикатов.
Автоматические системы построения доказательств (логических выводов) называются **пруверами** . От пруверов требуются следующие качества:

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

Стратегия построения табличного вывода, использованная в доказательстве теоремы полноты — это алгоритм построения доказательства истинности утверждений, выраженных формулами логики предикатов.
Автоматические системы построения доказательств (логических выводов) называются **пруверами** . От пруверов требуются следующие качества:

- корректность (абсолютно необходимо)

Автоматическое доказательство теорем

Стратегия построения табличного вывода, использованная в доказательстве теоремы полноты — это алгоритм построения доказательства истинности утверждений, выраженных формулами логики предикатов.
Автоматические системы построения доказательств (логических выводов) называются **пруверами**. От пруверов требуются следующие качества:

- корректность (абсолютно необходимо)
- полнота (очень-очень желательно)

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

Стратегия построения табличного вывода, использованная в доказательстве теоремы полноты — это алгоритм построения доказательства истинности утверждений, выраженных формулами логики предикатов.
Автоматические системы построения доказательств (логических выводов) называются **пруверами**. От пруверов требуются следующие качества:

- корректность (абсолютно необходимо)
- полнота (очень-очень желательно)
- эффективность (желательно)

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

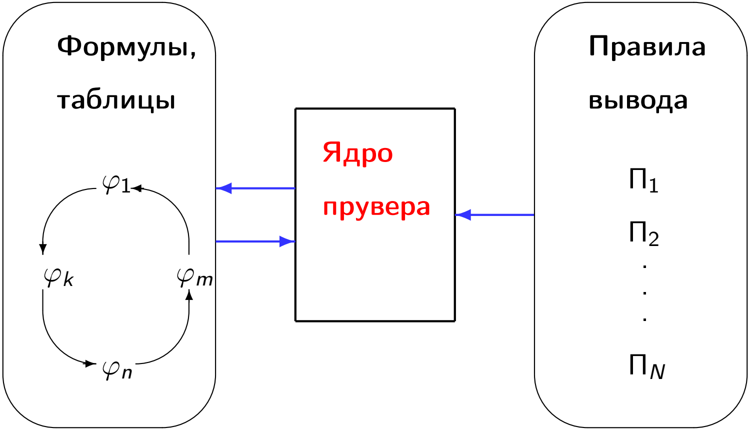
Стратегия построения табличного вывода, использованная в доказательстве теоремы полноты — это алгоритм построения доказательства истинности утверждений, выраженных формулами логики предикатов.
Автоматические системы построения доказательств (логических выводов) называются **пруверами** . От пруверов требуются следующие качества:

- корректность (абсолютно необходимо)
- полнота (очень-очень желательно)
- эффективность (желательно)

Первый прувер (LogicTheoretic крайне неэффективный) был разработан в 1957 в США (Newell (философ, социолог), Simon (математик), Show (программист))

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

Общая схема прувера



Автоматическое доказательство теорем

Наш прувер (табличный вывод) корректен и полон. Но насколько он эффективен?

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

Наш прувер (табличный вывод) корректен и полон. Но насколько он эффективен?

В алгоритме построения дерева табличного вывода применяется двойной переборный поиск:

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

Наш прувер (табличный вывод) корректен и полон. Но насколько он эффективен?

В алгоритме построения дерева табличного вывода применяется двойной переборный поиск:

- выбор формулы , к которой нужно применить правило ввода,

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

Наш прувер (табличный вывод) корректен и полон. Но насколько он эффективен?

В алгоритме построения дерева табличного вывода применяется двойной переборный поиск:

- выбор формулы , к которой нужно применить правило ввода,
- выбор правила , которое нужно применять к формуле.

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

Выбор формулы

Трудно избежать перебора всех формул таблицы.

База знаний	Запрос
В огороде бузина. $\text{Растет}(\text{бузина}, \text{огород})$	В Киеве дядька. $\exists y(\text{Дядька}(y)\ \&\ \text{Живет}(y, \text{Киев}))$

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

Выбор формулы

Трудно избежать перебора всех формул таблицы.

База знаний	Запрос
В огороде бузина. $\text{Растет}(\text{бузина}, \text{огород})$ Все в огороде посадил дядька. $\forall x(\text{Растет}(x, \text{огород}) \rightarrow \exists y(\text{Посадил}(y, x) \& \text{Дядька}(y)))$;	В Киеве дядька. $\exists y(\text{Дядька}(y)\ \&\ \text{Живет}(y, \text{Киев}))$

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

Выбор формулы

Трудно избежать перебора всех формул таблицы.

База знаний	Запрос
В огороде бузина. $\text{Растет}(\text{бузина}, \text{огород})$ Все в огороде посадил дядька. $\forall x(\text{Растет}(x, \text{огород}) \to \exists y(\text{Посадил}(y, x) \& \text{Дядька}(y)))$; Бузину посадил киевлянин. $\forall x(\text{Посадил}(x, \text{бузина}) \to \text{Живет}(x, \text{Киев}))$;	В Киеве дядька. $\exists y(\text{Дядька}(y)\ \&\ \text{Живет}(y, \text{Киев}))$

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

Наиболее критичный этап потроения табличного вывода — выбор нужного правила. Это касается правил ($L\forall$) и ($R\exists$).

$$L\forall \quad \frac{\langle \Gamma, \forall x \varphi(x) | \Delta \rangle}{\langle \Gamma, \forall x \varphi(x), \varphi(x)\{x/t\} | \Delta \rangle}$$

$$R\exists \quad \frac{\langle \Gamma | \Delta, \exists x \varphi(x) \rangle}{\langle \Gamma | \Delta, \exists x \varphi(x), \varphi(x)\{x/t\} \rangle}$$

поскольку выбор терма t правилами не оговаривается. Простой перебор всех термов практически невозможен. (почему ?)

Автоматическое доказательство теорем

Потому что термов очень-очень много. Если есть один двухместный функциональный символ $f^{(2)}$ и две константы c_1, c_2, то с их помощью можно построить более 10^{1000} основных термов высоты 10.

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

Потому что термов очень-очень много. Если есть один двухместный функциональный символ $f^{(2)}$ и две константы c_1, c_2, то с их помощью можно построить более 10^{1000} основных термов высоты 10. Это количество значительно превосходит число наносекунд, прошедших с момента Большого Взрыва.

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

Потому что термов очень-очень много. Если есть один двухместный функциональный символ $f^{(2)}$ и две константы c_1, c_2, то с их помощью можно построить более 10^{1000} основных термов высоты 10. Это количество значительно превосходит число наносекунд, прошедших с момента Большого Взрыва. Значит, нужно придумать способ, позволяющий быстро и точно вычислять тот терм, который нужно подставлять вместо переменных, связанных кванторами.

АВТОМАТИЧЕСКОЕ ДОКАЗАТЕЛЬСТВО ТЕОРЕМ

Потому что термов очень-очень много. Если есть один двухместный функциональный символ $f^{(2)}$ и две константы c_1, c_2, то с их помощью можно построить более 10^{1000} основных термов высоты 10. Это количество значительно превосходит число наносекунд, прошедших с момента Большого Взрыва. Значит, нужно придумать способ, позволяющий быстро и точно вычислять тот терм, который нужно подставлять вместо переменных, связанных кванторами.

Эту задачу в 1964 г. сумели решить Дж. Робинсон (метод резолюций) и С. Маслов (обратный метод).

КОНЕЦ ЛЕКЦИИ 5.