

# Математическая логика и логическое программирование

Лектор:

Подымов Владислав Васильевич

e-mail:

[valdus@yandex.ru](mailto:valdus@yandex.ru)

2016, весенний семестр

# Лекция 5

Полнота табличного вывода

Теорема Лёвенгейма-Сколема

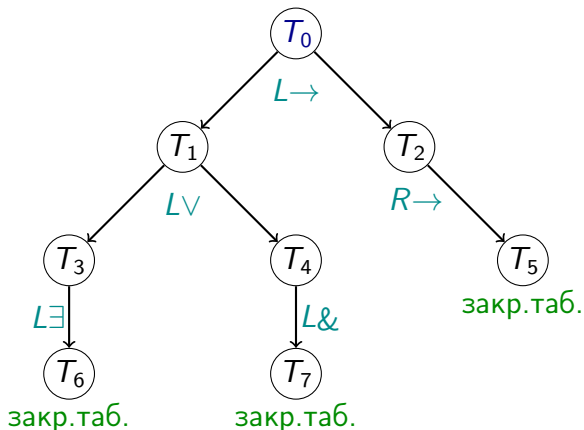
Теорема компактности Мальцева

Автоматическое доказательство теорем

Теорема Чёрча

# Напоминание

Корректность табличного вывода в логике предикатов:



Если для таблицы  $T_0$  построен успешный табличный вывод,  
то  $T_0$  невыполнима

# Полнота табличного вывода

А для любой ли невыполнимой семантической таблицы можно построить успешный табличный вывод?

В частности,

всегда ли мы можем убедиться в общезначимости формулы, используя только метод семантических таблиц?

Это и есть **ПОЛНОТА** табличного вывода

**Теорема полноты табличного вывода**

Если семантическая таблица невыполнима, то для неё существует успешный табличный вывод

# Полнота табличного вывода: доказательство

Пусть  $T_0 = \langle \Gamma_0 \mid \Delta_0 \rangle$  — невыполнимая таблица

Опишем стратегию табличного вывода, позволяющую доказать невыполнимость таблицы  $T_0$

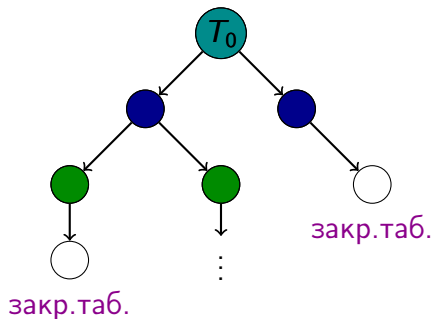
Для простоты считаем, что

- ▶ множества  $\Gamma_0, \Delta_0$  конечны
- ▶ все формулы из  $\Gamma_0, \Delta_0$  замкнуты
- ▶ формулы из  $\Gamma_0, \Delta_0$  не содержат функциональных символов

# Полнота табличного вывода: доказательство

При построении табличного вывода для  $T_0$  будем придерживаться следующих правил:

1. Незакрытые неатомарные таблицы обрабатываются в порядке появления

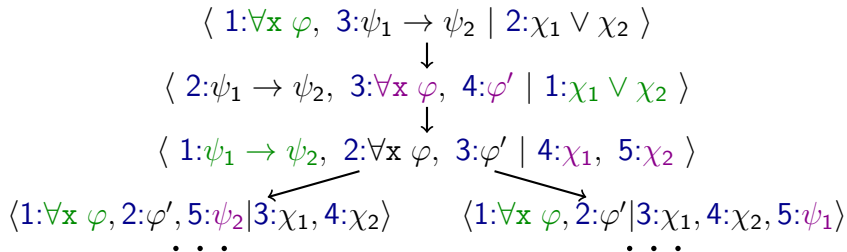


*Результат:* каждая таблица каждой ветви вывода будет построена

# Полнота табличного вывода: доказательство

При построении табличного вывода для  $T_0$  будем придерживаться следующих правил:

2. Все неатомарные формулы таблицы **упорядочены**, правило вывода применяется к **первой формуле**, результат применения записывается **последним**



**Результат:** в каждой бесконечной ветви вывода каждая неатомарная формула рано или поздно будет обработана

# Полнота табличного вывода: доказательство

При построении табличного вывода для  $T_0$  будем придерживаться следующих правил:

3. При применении правил  $L\forall$ ,  $R\exists$  подставляются **все** имеющиеся в таблице константы (**c**, если констант нет)

$$\begin{aligned} & \langle \forall x \varphi(x), \exists x \psi(x) \mid \exists x \chi(x) \rangle \\ & \quad \downarrow L\forall \\ & \langle \forall x \varphi(x), \exists x \psi(x), \varphi(\mathbf{c}) \mid \exists x \chi(x) \rangle \\ & \quad \downarrow L\exists \\ & \langle \forall x \varphi(x), \psi(\mathbf{d}), \varphi(\mathbf{c}) \mid \exists x \chi(x) \rangle \\ & \quad \downarrow R\exists \times 2 \\ & \langle \forall x \varphi(x), \psi(\mathbf{d}), \varphi(\mathbf{c}) \mid \exists x \chi(x), \chi(\mathbf{c}), \chi(\mathbf{d}) \rangle \\ & \quad \dots \end{aligned}$$

*Результат:* в каждой бесконечной ветви вывода каждая константа рано или поздно будет подставлена для каждого квантора



# Полнота табличного вывода: доказательство

Стратегия построения табличного вывода описана

Покажем, что вывод  $\mathcal{D}$ , построенный по этой стратегии для невыполнимой таблицы  $T_0 = \langle \Gamma_0 \mid \Delta_0 \rangle$ , успешен

*Предположим, что это не так:* вывод  $\mathcal{D}$  неуспешен

Заменим в  $\mathcal{D}$  каждую незакрытую атомарную таблицу  $T_{atom}$  на бесконечную ветвь  $T_{atom} \rightarrow T_{atom} \rightarrow T_{atom} \rightarrow \dots$

Тогда в полученном выводе обязательно найдётся бесконечная ветвь, состоящая только из незакрытых таблиц:

$$T_0 \rightarrow T_1 \rightarrow T_2 \rightarrow \dots$$

По имеющейся бесконечной ветви построим интерпретацию  $\mathcal{I}$ :

$$\mathcal{I} \models \Gamma_0, \quad \mathcal{I} \not\models \Delta_0$$

# Полнота табличного вывода: доказательство

Имеем ветвь  $T_0 \rightarrow T_1 \rightarrow T_2 \rightarrow \dots$ ,  $T_i = \langle \Gamma_i \mid \Delta_i \rangle$

$\mathcal{I} = \langle D, \overline{\text{Const}}, -, \overline{\text{Pred}} \rangle$ , где

- ▶ предметная область  $D$  — это все константы во всех  $T_i$ :

$$D = \bigcup_{i \geq 0} \text{Const}_i = \text{Const}_\omega \quad (\text{Const}_i \text{ — все константы в } T_i)$$

- ▶ значение каждой константы — это её изображение:

$$\bar{c} = c$$

- ▶ значение предиката = есть ли он хотя бы в одном  $\Gamma_i$ :

$$\overline{P}(c_1, \dots, c_k) = t \quad \Leftrightarrow \quad P(c_1, \dots, c_k) \in \bigcup_{i \geq 0} \Gamma_i = \Gamma_\omega$$

Осталось показать *индукцией по структуре формулы*, что

- ▶ любая формула из  $\Gamma_\omega$  выполнима в  $\mathcal{I}$
- ▶ любая формула из  $\Delta_\omega = \bigcup_{i \geq 0} \Delta_i$  невыполнима в  $\mathcal{I}$

# Полнота табличного вывода: доказательство

*База индукции:* рассмотрим атом  $\varphi \in \Gamma_\omega \cup \Delta_\omega$

Тогда  $\varphi = P(\mathbf{c}_1, \dots, \mathbf{c}_k)$ , где  $\mathbf{c}_1, \dots, \mathbf{c}_k \in \text{Const}_\omega$  (почему?)

- ▶ Пусть  $P(\mathbf{c}_1, \dots, \mathbf{c}_k) \in \Gamma_\omega$

Тогда  $\bar{P}(\mathbf{c}_1, \dots, \mathbf{c}_k) = \mathbf{t}$ , а значит,  $\mathcal{I} \models \varphi$

- ▶ Пусть  $P(\mathbf{c}_1, \dots, \mathbf{c}_k) \in \Delta_\omega$

Тогда  $P(\mathbf{c}_1, \dots, \mathbf{c}_k) \notin \Gamma_\omega$  (почему?)

Значит,  $\bar{P}(\mathbf{c}_1, \dots, \mathbf{c}_k) = \mathbf{f}$  и  $\mathcal{I} \not\models \varphi$

# Полнота табличного вывода: доказательство

*Индуктивный переход*

*Утверждение:* если  $\chi \in \Gamma_\omega$ , то  $\mathcal{I} \models \chi$ ; если  $\chi \in \Delta_\omega$ , то  $\mathcal{I} \not\models \chi$

*Предположение:* если  $\chi$  содержит менее  $N$  логических операций (связок и кванторов), то утверждение верно

*Рассматриваемый случай:*  $\chi$  содержит  $N$  логических операций

Пусть  $\chi = \varphi \rightarrow \psi \in \Gamma_\omega$

В ветви  $T_0 \rightarrow T_1 \rightarrow \dots$  существует таблица  $T_i$ , такая что правило вывода применяется к  $\chi$  в левой части (почему?)

Тогда верно хотя бы одно из двух:

- ▶  $\psi \in \Gamma_{i+1}$ , и тогда  $\mathcal{I} \models \psi$  и  $\mathcal{I} \models \chi$  (почему?)
- ▶  $\varphi \in \Delta_{i+1}$ , и тогда  $\mathcal{I} \not\models \varphi$  и  $\mathcal{I} \models \chi$

# Полнота табличного вывода: доказательство

*Индуктивный переход*

*Утверждение:* если  $\chi \in \Gamma_\omega$ , то  $\mathcal{I} \models \chi$ ; если  $\chi \in \Delta_\omega$ , то  $\mathcal{I} \not\models \chi$

*Предположение:* если  $\chi$  содержит менее  $N$  логических операций (связок и кванторов), то утверждение верно

*Рассматриваемый случай:*  $\chi$  содержит  $N$  логических операций

Пусть  $\chi = \varphi \rightarrow \psi \in \Gamma_\omega$        $\chi = \varphi \rightarrow \psi \in \Delta_\omega$

Пусть  $\chi = \varphi \& \psi \in \Gamma_\omega$        $\chi = \varphi \& \psi \in \Delta_\omega$

Пусть  $\chi = \varphi \vee \psi \in \Gamma_\omega$        $\chi = \varphi \vee \psi \in \Delta_\omega$

Пусть  $\chi = \neg\varphi \in \Gamma_\omega$        $\chi = \neg\varphi \in \Delta_\omega$

Во всех этих случаях рассуждения аналогичны

# Полнота табличного вывода: доказательство

*Индуктивный переход*

*Утверждение:* если  $\chi \in \Gamma_\omega$ , то  $\mathcal{I} \models \chi$ ; если  $\chi \in \Delta_\omega$ , то  $\mathcal{I} \not\models \chi$

*Предположение:* если  $\chi$  содержит менее  $N$  логических операций (связок и кванторов), то утверждение верно

*Рассматриваемый случай:*  $\chi$  содержит  $N$  логических операций

Пусть  $\chi = \forall x \varphi(x) \in \Gamma_\omega$

Тогда  $\varphi(\mathbf{c}) \in \Gamma_\omega$  для любой константы  $\mathbf{c} \in \text{Const}_\omega$  (почему?)

Значит, для любой константы  $\mathbf{c} \in \text{Const}_\omega$  верно:  $\mathcal{I} \models \varphi(\mathbf{c})$

Но это и означает  $\mathcal{I} \models \forall x \varphi(x)$

Пусть  $\psi = \exists x \varphi(x) \in \Delta_\omega$

Рассуждения в этом случае аналогичны

# Полнота табличного вывода: доказательство

*Индуктивный переход*

*Утверждение:* если  $\chi \in \Gamma_\omega$ , то  $\mathcal{I} \models \chi$ ; если  $\chi \in \Delta_\omega$ , то  $\mathcal{I} \not\models \chi$

*Предположение:* если  $\chi$  содержит менее  $N$  логических операций (связок и кванторов), то утверждение верно

*Рассматриваемый случай:*  $\chi$  содержит  $N$  логических операций

Пусть  $\chi = \forall x \varphi(x) \in \Delta_\omega$

В одной из таблиц  $T_i$  к формуле  $\chi$  в правой части обязательно применится правило  $R\exists$

Значит,  $\varphi(\mathbf{c}) \in \Delta_{i+1}$  для некоторой  $\mathbf{c} \in \text{Const}_{i+1} \subseteq \text{Const}_\omega$

Тогда  $\mathcal{I} \not\models \varphi(\mathbf{c})$ , а значит,  $\mathcal{I} \not\models \forall x \varphi(x)$

Пусть  $\chi = \exists x \varphi(x) \in \Gamma_\omega$

Рассуждения в этом случае аналогичны

# Полнота табличного вывода: доказательство

Итог:  $\mathcal{I} \models \Gamma_\omega$ ,  $\mathcal{I} \not\models \Delta_\omega$

А что требуется найти?

Противоречие с тем, что таблица  $T_0 = \langle \Gamma_0 \mid \Delta_0 \rangle$   
невыполнима

Имеем:  $\Gamma_0 \subseteq \Gamma_\omega$ ,  $\Delta_0 \subseteq \Delta_\omega$

Значит,  $\mathcal{I} \models \Gamma_0$  и  $\mathcal{I} \not\models \Delta_0$

Но это и означает **выполнимость** таблицы  $T_0$

От какого предположения мы отталкивались?

От предположения о том, что построенный для  $T_0$  вывод  $\mathfrak{D}$   
неуспешен

Значит, нами построен **успешный** табличный вывод





# Полнота табличного вывода: доказательство

А что изменится в доказательстве для общего случая?

То есть:

- ▶ какой порядок обработки формул позволит “справедливо” обращаться с бесконечными множествами формул?
- ▶ какие термы подставлять, если в сигнатуре алфавита есть функциональные символы?
- ▶ как определить интерпретацию  $\mathcal{I}$ , если в таблицах встречаются незамкнутые формулы

# Полнота табличного вывода

Теорема Гёделя о полноте

(вариант для *исчисления семантических таблиц*)

$\models \varphi \iff$  для семантической таблицы  $\langle \quad | \varphi \rangle$   
существует успешный табличный вывод

Доказательство.

Следует из **корректности** и **полноты** табличного вывода  
(и утверждения о связи общезначимости формулы и  
выполнимости соответствующей таблицы)

Более того, в доказательстве **теоремы полноты** сказано, как  
**построить** успешный табличный вывод, если он существует

А можно ли извлечь ещё какие-нибудь полезные факты из  
корректности и полноты табличного вывода?

# Теорема Лёвенгейма-Сколема

Формула  $\varphi$  выполнима  $\Leftrightarrow \varphi$  имеет модель с конечной или счётно-бесконечной предметной областью

Доказательство.

*Корректность* и *полнота*:  $\varphi$  выполнима  $\Leftrightarrow$  для таблицы  $T_0 = \langle \varphi \mid \rangle$  не существует успешного табличного вывода

Построив вывод согласно *стратегии из доказательства полноты*, получим

- ▶ бесконечную ветвь  $T_0 \rightarrow T_1 \rightarrow \dots$ , состоящую только из незакрытых таблиц
- ▶ интерпретацию  $\mathcal{I}$  с не более чем счётно-бесконечной предметной областью, в которой выполнимы все таблицы этой ветви — в том числе таблица  $T_0$



# Теорема компактности Мальцева

$\Gamma \models \varphi \Leftrightarrow$  существует конечное подмножество  $\Gamma'$ ,  
 $\Gamma' \subseteq \Gamma$ , такое что  $\Gamma' \models \varphi$

Доказательство.

$\Gamma \models \varphi \Leftrightarrow$  таблица  $T = \langle \Gamma \mid \varphi \rangle$  невыполнима (почему?)  
 $\Leftrightarrow$  существует успешный табличный вывод  $\mathcal{D}$  для  $T$

В выводе  $\mathcal{D}$  обрабатывается правилами вывода и участвует в противоречии лишь конечное множество формул  $\Gamma'$  из  $\Gamma$  (почему?)

Тогда для таблицы  $\langle \Gamma' \mid \varphi \rangle$  также существует успешный табличный вывод (почему?)

Значит,  $\Gamma' \models \varphi$



## Автоматическое доказательство теорем

А что получится, если программно реализовать стратегию построения успешного табличного вывода?

Средство автоматического доказательства теорем:

First-order theorem **prover**

Задача **прувера** — предоставлять **доказательство** общезначимости формулы (т.е. логический вывод)

Требования, предъявляемые к прuverу:

- ▶ **корректность**: обязательно
- ▶ **полнота**: очень желательно
- ▶ **эффективность**: желательно

А что-нибудь полезное с помощью прuverов сделано?

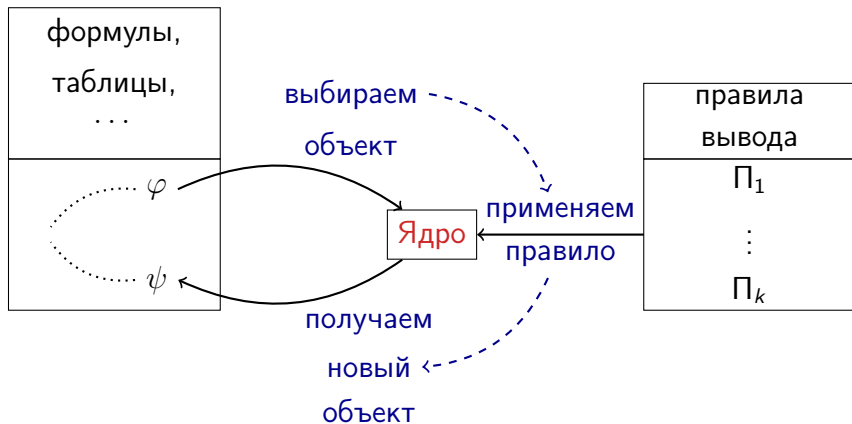
Да, например: найдены сотни ошибок и после их исправления **строго доказана** корректность \*nix-микроядра L4<sup>1</sup>

---

<sup>1</sup> High-order logic theorem prover: Isabelle

# Автоматическое доказательство теорем

Как выглядят проверки:



# Автоматическое доказательство теорем: теоретический взгляд. Теорема Чёрча

Насколько эффективным будет корректный и полный пружер, доказывающий общезначимость формул логики предикатов?

## Теорема Чёрча

Проблема общезначимости формул логики предикатов алгоритмически неразрешима

Иными словами, никакой алгоритм не способен решить проблему “ $\models \varphi$  ?” для произвольной формулы  $\varphi$

Но при этом существует алгоритм, способный подтвердить общезначимость формулы (например, метод семантических таблиц с полной стратегией построения успешного вывода)

Значит, проблема “ $\models \varphi$  ?” **полуразрешима**, или

**частично разрешима**

# Автоматическое доказательство теорем: теоретический взгляд

Как доказать неразрешимость проблемы " $\models \varphi$ "?

Например, взять другую неразрешимую проблему и свести её к " $\models \varphi$ "

Проблема останова машин Тьюринга

*Дано:* машина Тьюринга  $M$ , слово  $w$

*Вопрос*  $\text{HALT}(M, w)$ : остановится ли  $M$  на входном слове  $w$ ?

Что такое "свести"?

Предложить алгоритм, по  $M$  и  $w$  строящий такую формулу  $\varphi$ :

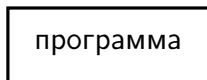
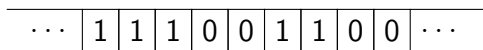
$$\text{HALT}(M, w) \quad \Leftrightarrow \quad \models \varphi$$

Но перед этим надо строго сформулировать все понятия



# Машины Тьюринга

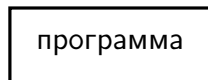
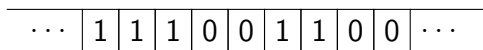
(МТ)



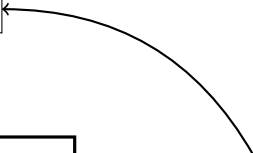
лента

# Машины Тьюринга

(МТ)

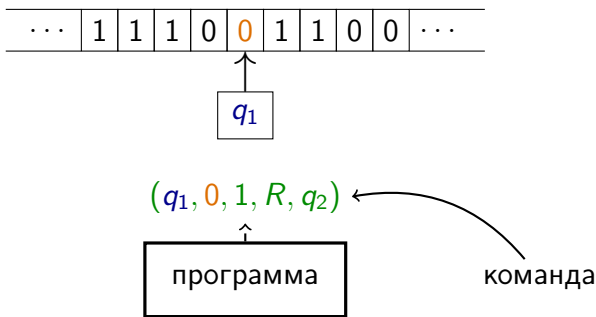


головка



# Машины Тьюринга

(МТ)

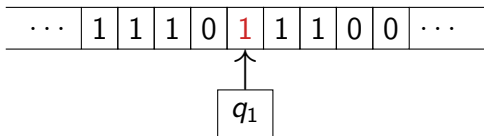


Шаг вычисления выглядит так:

- ▶ по текущему состоянию и обозреваемому символу выбираем команду

# Машины Тьюринга

(МТ)



$(q_1, 0, 1, R, q_2)$

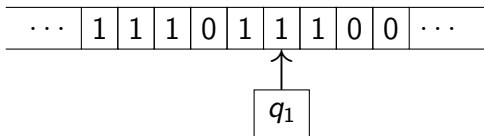
программа

Шаг вычисления выглядит так:

- ▶ по текущему состоянию и обозреваемому символу выбираем команду
- ▶ записываем в ячейку **новый символ**

# Машины Тьюринга

(МТ)



$(q_1, 0, 1, R, q_2)$

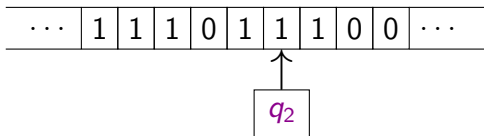
программа

Шаг вычисления выглядит так:

- ▶ по текущему состоянию и обозреваемому символу выбираем команду
- ▶ записываем в ячейку новый символ
- ▶ **сдвигаем** головку

# Машины Тьюринга

(МТ)



$(q_1, 0, 1, R, q_2)$

программа

Шаг вычисления выглядит так:

- ▶ по текущему состоянию и обозреваемому символу выбираем команду
- ▶ записываем в ячейку новый символ
- ▶ сдвигаем головку
- ▶ **меняем состояние**

# Машины Тьюринга

(МТ)

Теперь всё с начала, по порядку и строго

$\mathcal{A}$  — ленточный алфавит,  $\Lambda$  — пустой символ ( $\Lambda \in \mathcal{A}$ )

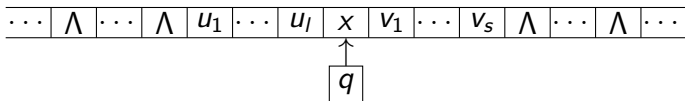
Ленточное слово — это слово в алфавите  $\mathcal{A}$

$\mathcal{A}^*$  — множество всех ленточных слов

$\mathcal{Q}$  — алфавит состояний,  $q_0$  — начальное состояние ( $q_0 \in \mathcal{Q}$ )

Ленточная конфигурация — это слово вида  $u q x v$   
( $q \in \mathcal{Q}, x \in \mathcal{A}, u, v \in \mathcal{A}^*$ )  
( $u = u_1 \dots u_l, v = v_1 \dots v_s$ )

Пояснение:



Команда — это пятёрка вида  $(q, a, b, D, q')$   
( $q, q' \in \mathcal{Q}, a, b \in \mathcal{A}, D \in \{L, R\}$ )

Пояснение: из состояния  $q$ , обозревая символ  $a$ , записать в ячейку  $b$ , сдвинуться влево ( $D = L$ ) или вправо ( $D = R$ ) и перейти в состояние  $q'$

# Машины Тьюринга

(МТ)

Командой  $C$  задаётся бинарное отношение переходов  $\rightarrow_C$  на множестве конфигураций:  $(z \in \mathcal{A})$

$$\triangleright u z q x v \rightarrow_{(q,x,y,L,q')} u q' z y v$$

$$\triangleright q x v \rightarrow_{(q,x,y,L,q')} q' \Lambda y v$$

$$\triangleright u q x z v \rightarrow_{(q,x,y,R,q')} u y q' z v$$

$$\triangleright u q x \rightarrow_{(q,x,y,R,q')} u y q' \Lambda$$



# Машины Тьюринга

(МТ)

Машина Тьюринга — это конечное множество команд

Детерминированная МТ: для любых  $q \in Q$ ,  $a \in \mathcal{A}$  не более чем одна команда имеет вид  $(q, a, b, D, q')$

Будем рассматривать **только** детерминированные МТ

Отношение переходов, задаваемое МТ  $M$  на множестве ленточных конфигураций, определяется так:

$$\rightarrow_M = \bigcup_{C \in M} \rightarrow_C$$

$\alpha$  — **заключительная конфигурация** МТ  $M$ , если не существует конфигурации  $\beta$ , такой что  $\alpha \rightarrow_M \beta$

МТ  $M$  **останавливается** на слове  $w$  (**HALT**( $M, w$ )), если какая-либо заключительная конфигурация  $C_{fin}$  **достижима** из **начальной** конфигурации  $q_0$   $w$ :

$$q_0 w \rightarrow_M \cdots \rightarrow_M C_{fin}$$

# Теорема Чёрча: доказательство

Сведём проблему останова детерминированных МТ к проблеме общезначимости формул логики предикатов

Опишем, как можно по МТ  $M$  и ленточному слову  $w$  построить такую формулу  $\varphi_{M,w}$ :

$$\text{HALT}(M, w) \quad \Leftrightarrow \quad \models \varphi_{M,w}$$

*Что будет записано в формуле:*

Если начальная конфигурация достижима и конфигурация, следующая за достижимой, также достижима, то достижима некоторая заключительная конфигурация

*Start & Step*  $\rightarrow$  *Finish*

# Теорема Чёрча: доказательство

Какие символы понадобятся в формуле:

- ▶ константа **a** для каждого  $a \in \mathcal{A}$
- ▶ константа **q** для каждого  $q \in \mathcal{Q}$
- ▶ операция конкатенации:  $\cdot$ <sup>(2)</sup>  
(ассоциативная вправо в инфиксной записи)
- ▶ отношение достижимости конфигурации из начальной:  
 $\text{Re}$ <sup>(3)</sup>

Что записать в  $\varphi_{M,w}$ :

- ▶ ленточное слово  $a_1 \dots a_p$   
$$\widetilde{a_1 \dots a_p} = \mathbf{a_1 \dots a_p}$$
- ▶ “конфигурация  $u q v$  достижима”  $((a_1 \dots a_p)^- = a_p \dots a_1)$   
$$\text{Re}(\widetilde{u^-}, \mathbf{q}, \widetilde{v})$$

# Теорема Чёрча: доказательство

Что записать в  $\varphi_{M,w}$ :

- ▶ “начальная конфигурация достижима”

$$\text{Start: Re}(\Lambda, \mathbf{q}_0, \tilde{w})$$

- ▶ “если текущая конфигурация достижима, то конфигурация после применения команды  $C$  будет достижима”

- ▶  $C = (q, a, b, R, q')$

$$\varphi_C: \forall x \forall y (\text{Re}(x, \mathbf{q}, \mathbf{a} \cdot y) \rightarrow \text{Re}(\mathbf{b} \cdot x, \mathbf{q}', y))$$

$$\psi_C: \forall x \forall y (\text{Re}(x, \mathbf{q}, \mathbf{a}) \rightarrow \text{Re}(\mathbf{b} \cdot x, \mathbf{q}', \Lambda))$$

- ▶  $C = (q, a, b, L, q')$

$$\varphi_C: \forall x \forall y \forall z (\text{Re}(z \cdot x, \mathbf{q}, \mathbf{a} \cdot y) \rightarrow \text{Re}(x, \mathbf{q}', z \cdot \mathbf{b} \cdot y))$$

$$\psi_C: \forall x \forall y \forall z (\text{Re}(\Lambda, \mathbf{q}, \mathbf{a} \cdot y) \rightarrow \text{Re}(\Lambda, \mathbf{q}', \Lambda \cdot \mathbf{b} \cdot y))$$

- ▶ “конфигурация, следующая за достижимой, также достижима”

$$\text{Step: } \bigwedge_{C \in M} (\varphi_C \ \& \ \psi_C)$$

# Теорема Чёрча: доказательство

Что записать в  $\varphi_{M,w}$ :

- ▶ пусть  $\mathcal{G} \subseteq \mathcal{Q} \times \mathcal{A}$  — множество пар  $(q, a)$ , таких что в  $M$  нет команд, начинающихся с  $q$ ,  $a$ , но есть команды, содержащие  $q$
- ▶ “хотя бы одна заключительная конфигурация достижима”

$$\text{Finish: } \exists x \exists y \left( \bigvee_{(q,a) \in \mathcal{G}} \text{Re}(x, \mathbf{q}, \mathbf{a} \cdot y) \right)$$

Возвращаемся к тому, что нужно доказать:

**HALT**( $M, w$ ): МТ  $M$  останавливается на слове  $w$

$\varphi_{M,w}$ : *Start & Step*  $\rightarrow$  *Finish*

$$\text{HALT}(M, w) \quad \Leftrightarrow \quad \models \varphi_{M,w}$$

# Теорема Чёрча: доказательство

$$\mathbf{HALT}(M, w) \quad \Leftrightarrow \quad \models \varphi_{M,w}$$

( $\Leftarrow$ ): если формула  $\varphi_{M,w}$  истинна в любой интерпретации, то она истинна и в такой интерпретации  $\mathcal{I}$ :

$\overline{\text{Re}}(\widetilde{u}^-, \mathbf{q}, \widetilde{v}) =$  “конфигурация  $u q v$  достижима из  $q_0 w$  для  $M$ ”

( $\Rightarrow$ ): предположим, что  $\mathbf{HALT}(M, w)$ , но  $\not\models \varphi_{M,w}$

Тогда существует интерпретация  $\mathcal{I}$ , такая что  $\mathcal{I} \not\models \varphi_{M,w}$

$$\mathcal{I} \models \mathbf{Start}, \quad \mathcal{I} \models \mathbf{Step}, \quad \mathcal{I} \not\models \mathbf{Finish}$$

Рассмотрим работу  $M$  на  $w$ :

$$q_0 w \rightarrow_M u_1 q_1 v_1 \rightarrow_M u_2 q_2 v_2 \rightarrow_M \dots \rightarrow_M u_k q_k v_k,$$

где  $u_k q_k v_k$  — заключительная конфигурация

При этом из  $\mathbf{Start}$  и  $\mathbf{Step}$  получаем:

$$\begin{aligned} \mathcal{I} \models \text{Re}(\mathbf{\Lambda}, \mathbf{q}_0, \widetilde{w}), \quad \mathcal{I} \models \text{Re}(\widetilde{u}_1^-, \mathbf{q}_1, \widetilde{v}_1), \\ \mathcal{I} \models \text{Re}(\widetilde{u}_2^-, \mathbf{q}_2, \widetilde{v}_2), \dots, \mathcal{I} \models \text{Re}(\widetilde{u}_k^-, \mathbf{q}_k, \widetilde{v}_k) \end{aligned}$$

Значит,  $\mathcal{I} \models \mathbf{Finish}$  (противоречие)



# Автоматическое доказательство теорем: практический взгляд

Как избежать заикливания при исследовании общезначимости формулы  $\varphi$ ?

Если мы абсолютно уверены, что формула  $\varphi$  общезначима, и хотим предоставить доказательство общезначимости, то это можно сделать

Например, с помощью метода семантических таблиц

А в этом случае будет ли всё работать эффективно?

- ▶ К какой формуле применять правило табличного вывода?
- ▶ При применении правил  $L\forall$ ,  $R\exists$  какие термы подставлять?

# Автоматическое доказательство теорем: практический взгляд

## Выбор формулы

Как избежать перебора всех формул таблицы?

База знаний	Запрос
В огороде бузина Растёт(бузина, огород)	В Киеве дядька $\exists u$ (Дядька(u) & Живёт(u, Киев))
Всё в огороде посадил дядька $\forall x$ (Растёт(x, огород) $\rightarrow$ $\exists u$ (Посадил(u, x) & Дядька(u)))	
Бузину сажают только Киевляне $\forall x$ (Посадил(x, бузина) $\rightarrow$ Живёт(x, Киев))	



# Автоматическое доказательство теорем: практический взгляд

## Выбор термина

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

Насколько разумно перебирать все использованные термины?

Из одного функционального символа  $f^{(2)}$  и двух констант  $c_1$ ,  $c_2$  можно построить более  $10^{300}$  термов высоты 10

А это много?

Гугол — это  $10^{100}$ ,

и это больше числа атомов в наблюдаемой вселенной

Можно ли как-то сократить перебор термов?

В некоторой степени — да, можно<sup>1,2</sup>

---

<sup>1</sup> J.A. Robinson: метод резолюций

<sup>2</sup> С.Ю. Маслов: обратный вывод

Конец лекции 5