Математическая логика и логическое программирование

mk.cs.msu.ru → Лекционные курсы
→ Математическая логика и логическое программирование (3-й поток)

Блок 39

Машины Тьюринга

Лектор:
Подымов Владислав Васильевич
E-mail:
valdus@yandex.ru

ВМК МГУ, 2022/2023, осенний семестр

Вступление

Хорновские логические программы — это вычислительная модель, основанная на проверке невыполнимости систем дизъюнктов частного вида (хорновских дизъюнктов) при помощи одного частного способа построения резолютивного вывода (входная резолюция)

При обсуждении полноты операционной семантики ХЛП было показано, что входная резолюция полна для систем хорновских дизъюнктов, но этими дизъюнктами выражаются только знания, «уложенные» в одну из двух частных строгих форм:

1. A;: утверждение, записанное как атом A, верно
2. $A \leftarrow B_1, \dots, B_k$;: если утверждения, записанные как атомы $B_1, \dots, B_k$, верны, то верно и утверждение, записанное как атом A

А не оказываются ли ХЛП слишком невыразительными из-за ограничений на форму представления знаний?

Какие задачи можно решить с помощью ХЛП, а какие нельзя?

Вступление

Есть много разных вычислительных моделей, предназначенных для работы с данными принципиального разного устройства: строками, числами, графами, типами, молекулами ДНК, *запросами к ХЛП*, ...

Естественно возникает потребность **сравнивать** между собой такие модели независимо от природы их данных

Естественный способ сравнения вычислительных моделей основан на том, что их данными кодируются неотрицательные целые числа ($\mathbb{N}_0$), и сравниваются возможности вычисления функций вида $f : \mathbb{N}_0 \to \mathbb{N}_0$

Тезис Чёрча-Тьюринга: все (частичные) функции вида $f : \mathbb{N}_0 \to \mathbb{N}_0$, которые вычисляются алгоритмами в нестрогом понимании, — это в точности все функции, вычислимые на **машинах Тьюринга**

Вступление

Тезис Чёрча-Тьюринга невозможно сформулировать строго и тем более обосновать, но подтверждением можно считать то, что многие модели, придумывавшиеся как формализация понятия алгоритма, оказались в точности такими же выразительными, как и машины Тьюринга:

- Нормальные алгорифмы Маркова
- Частично рекурсивные функции
- Машины Поста
- Машины Минского
- λ-исчисление
- Системы переписывания строк (полу-Туэ)
- RAM-машины
- ... ...

Такие модели, в конечном итоге описывающие возможности вычисления одного и того же класса функций, принято называть алгоритмически полными

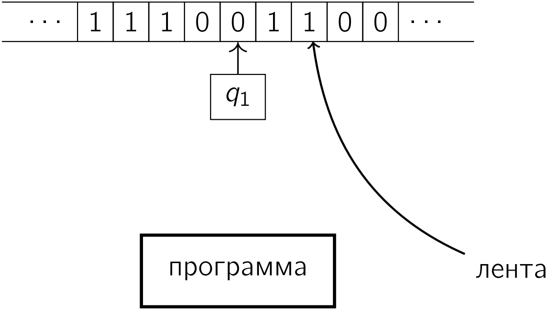
Вступление

А является модель ХЛП алгоритмически полной, или же есть что-то такое, что могут вычислять полные модели, а ХЛП не могут?

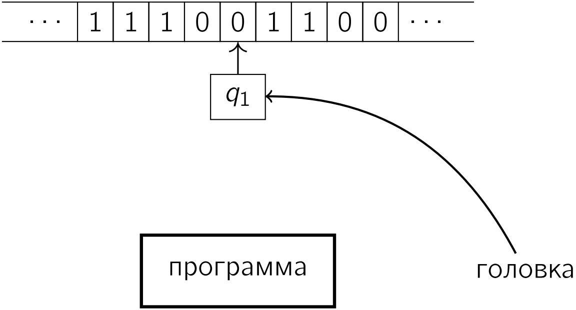
Обсуждение этого вопроса начнём с напоминания о том, что такое машина Тьюринга

Существует немало вариантов этой модели и немало способов её введения, поэтому дальнейшее можно расценивать не только как напоминание, но и как выбор варианта и системы обозначений

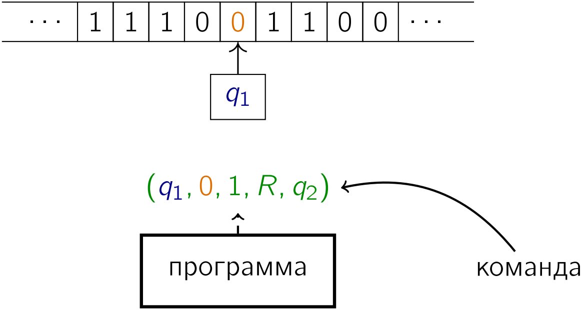
Машины Тьюринга (МТ)



Машины Тьюринга (МТ)



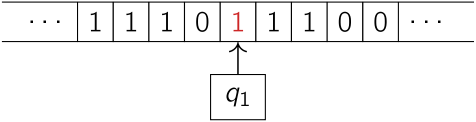
Машины Тьюринга (МТ)



Шаг вычисления выглядит так:

- по текущему состоянию
  и обозреваемому символу выбираем команду

Машины Тьюринга (МТ)



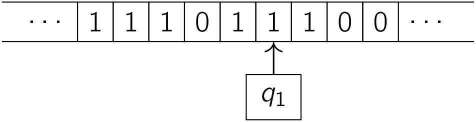
$$(q_1, 0, 1, R, q_2)$$

программа

Шаг вычисления выглядит так:

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

Машины Тьюринга (МТ)



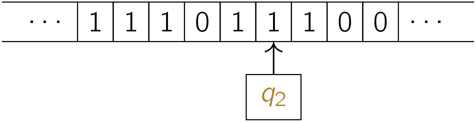
$$(q_1, 0, 1, R, q_2)$$

программа

Шаг вычисления выглядит так:

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

Машины Тьюринга (МТ)



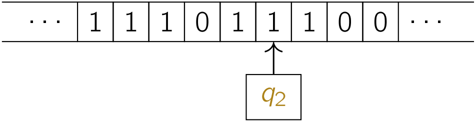
$$(q_1, 0, 1, R, q_2)$$

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)



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

$\boxed{\text{программа}}$

Шаг вычисления выглядит так:

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

Определим всё это по порядку, детально и строго

МТ: синтаксис

Алфавит — это непустое конечное множество символов (букв)

Машина Тьюринга (МТ)[1] — это система $(\mathfrak{A}, \Lambda, \mathcal{Q}, q_0, q_f, \pi)$, где

- $\mathfrak{A}$ — алфавит ленты
- $\Lambda \in \mathfrak{A}$ — пустой символ
- $\mathcal{Q}$ — алфавит состояний
- $q_0, q_f \in \mathcal{Q}$ — соответственно
  начальное состояние и заключительное состояние
- $\pi : (\mathcal{Q} \setminus \{q_f\}) \times \mathfrak{A} \to \mathfrak{A} \times \{L, R\} \times \mathcal{Q}$ — программа

Программа МТ также будет пониматься как множество команд:

$$(q, a, b, S, p) \in \pi \Leftrightarrow \pi(q, a) = (b, S, p)$$

1 Здесь обсуждается только один из вариантов определения машины Тьюринга, удобный для сравнения с ХЛП

МТ: семантика

$$M = (\mathfrak{A}, \Lambda, \mathcal{Q}, q_0, q_f, \pi)$$

Слово в алфавите Σ — это конечная последовательность букв из Σ

В записи слов принято опускать
разделители (запятые) между символами:

$$\cancel{x_1, x_2, \dots, x_n} \qquad x_1 x_2 \dots x_n$$

Σ^* — множество всех слов в алфавите Σ

Σ^+ — множество всех непустых слов в алфавите Σ

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

Конфигурация машины Тьюринга — это набор (α, q, β),
где $\alpha, \beta \in \mathfrak{A}^+$ и $q \in \mathcal{Q}$

Пояснение: конфигурация $(a_1 \dots a_n, q, b_1 \dots b_m)$ означает, что МТ находится в состоянии q, на ленте записано слово $a_1 \dots a_n b_1 \dots b_m$, окружённое пустыми символами, и обозревается символ b_1

$$\begin{array}{c|c|c|c|c|c|c|c|c|c|c|c|c|c}
\hline
\cdots & \Lambda & \cdots & \Lambda & a_1 & \cdots & a_n & b_1 & \cdots & b_m & \Lambda & \cdots & \Lambda & \cdots \\
\hline
\end{array}$$

$$\uparrow$$

$$\boxed{q}$$

МТ: семантика

$$M = (\mathfrak{A}, \Lambda, \mathcal{Q}, q_0, q_f, \pi)$$

Способ преобразования конфигураций командой C можно задать как двуместное отношение $\to_C$ на множестве конфигураций

Если $C = (q, a, b, R, p)$, то $\to_C$ состоит из следующих пар:

$$\begin{array}{lcl} (\alpha, q, ax\beta) & \to_C & (\alpha b, p, x\beta) \\ (\alpha, q, a) & \to_C & (\alpha b, p, \Lambda) \end{array}$$

Если $C = (q, a, b, L, p)$, то $\to_C$ состоит из следующих пар:

$$\begin{array}{rcr} (\alpha xy, q, a\beta) & \to_C & (\alpha x, p, yb\beta) \\ (y, q, a\beta) & \to_C & (\Lambda, p, yb\beta) \end{array}$$

$$(\alpha, \beta \in \mathfrak{A}^*;\ x, y \in \mathfrak{A})$$

Отношение переходов $\to_M$ МТ M — это объединение отношений $\to_C$ по всем командам C МТ M

Трасса МТ M — это последовательность конфигураций, в которой каждая пара соседних конфигураций σ_i, σ_{i+1} входит в отношение переходов: $\sigma_i \to_M \sigma_{i+1}$

$\to_M^*$ — это транзитивное замыкание отношения $\to_M$:

$$\sigma_1 \to_M^* \sigma_2 \quad \Leftrightarrow \quad \text{существует трасса с началом } \sigma_1 \text{ и концом } \sigma_2$$

МТ: семантика

$$M = (\mathfrak{A}, \Lambda, \mathcal{Q}, q_0, q_f, \pi)$$

Конфигурация (α, q, β) — заключительная, если $q = q_f$

Вычисление МТ M на ленточном слове w — это трасса,

- начинающаяся в конфигурации $(\Lambda, q_0, w\Lambda)$ и
- либо бесконечная,
  либо оканчивающаяся в заключительной конфигурации

Утверждение. Для любой МТ M и любого ленточного слова w существует единственное вычисление M на w

Доказательство.
Достаточно заметить, что в соотношении $\sigma_i \to_M \sigma_{i+1}$

- σ_i может быть любой незаключительной конфигурацией
  и не может быть заключительной, и
- σ_{i+1} однозначно определяется конфигурацией σ_i ▼

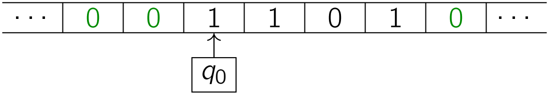
МТ: пример

Пример: $M = (\{0,1\}, 0, \{q_0, q_1, q_f\}, q_0, q_f, \pi)$, где:

$$\pi(q_0, 0) = (0, L, q_1) \qquad \pi(q_1, 0) = (0, R, q_f)$$
$$\pi(q_0, 1) = (1, R, q_0) \qquad \pi(q_1, 1) = (0, L, q_1)$$

Вычисление M на слове 1101:

$$(0, \quad q_0, \quad 11010)$$



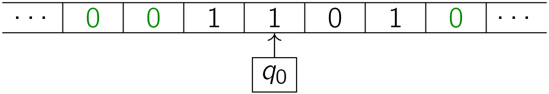
МТ: пример

Пример: $M = (\{0,1\}, 0, \{q_0, q_1, q_f\}, q_0, q_f, \pi)$, где:

$$\begin{array}{ll} \pi(q_0, 0) = (0, L, q_1) & \qquad \pi(q_1, 0) = (0, R, q_f) \\ \pi(q_0, 1) = (1, R, q_0) & \qquad \pi(q_1, 1) = (0, L, q_1) \end{array}$$

Вычисление M на слове 1101:

$$\begin{array}{rccl} (0, & q_0, & 11010) & \quad \to_M \\ (01, & q_0, & 1010) & \end{array}$$



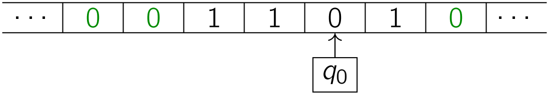
МТ: пример

Пример: $M = (\{0,1\}, 0, \{q_0, q_1, q_f\}, q_0, q_f, \pi)$, где:

$$
\begin{array}{ll}
\pi(q_0, 0) = (0, L, q_1) & \pi(q_1, 0) = (0, R, q_f) \\
\pi(q_0, 1) = (1, R, q_0) & \pi(q_1, 1) = (0, L, q_1)
\end{array}
$$

Вычисление M на слове 1101:

$$
\begin{array}{rccl}
(0, & q_0, & 11010) & \to_M \\
(01, & q_0, & 1010) & \to_M \\
(011, & q_0, & 010) &
\end{array}
$$



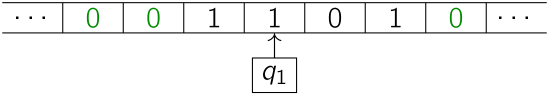
МТ: пример

Пример: $M = (\{0,1\}, 0, \{q_0, q_1, q_f\}, q_0, q_f, \pi)$, где:

$$\begin{array}{ll} \pi(q_0,0) = (0, L, q_1) & \pi(q_1,0) = (0, R, q_f) \\ \pi(q_0,1) = (1, R, q_0) & \pi(q_1,1) = (0, L, q_1) \end{array}$$

Вычисление M на слове 1101:

$$\begin{array}{rlll} (0, & q_0, & 11010) & \to_M \\ (01, & q_0, & 1010) & \to_M \\ (011, & q_0, & 010) & \to_M \\ (01, & q_1, & 1010) & \end{array}$$



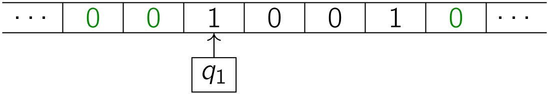
МТ: пример

Пример: $M = (\{0, 1\}, 0, \{q_0, q_1, q_f\}, q_0, q_f, \pi)$, где:

$$\pi(q_0, 0) = (0, L, q_1) \qquad \pi(q_1, 0) = (0, R, q_f)$$
$$\pi(q_0, 1) = (1, R, q_0) \qquad \pi(q_1, 1) = (0, L, q_1)$$

Вычисление M на слове 1101:

$$\begin{array}{rlll}
(0, & q_0, & 11010) & \to_M \\
(01, & q_0, & 1010) & \to_M \\
(011, & q_0, & 010) & \to_M \\
(01, & q_1, & 1010) & \to_M \\
(0, & q_1, & 10010) &
\end{array}$$



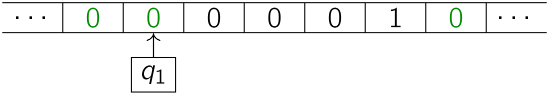
МТ: пример

Пример: $M = (\{0,1\}, 0, \{q_0, q_1, q_f\}, q_0, q_f, \pi)$, где:

$$\begin{array}{ll} \pi(q_0, 0) = (0, L, q_1) & \pi(q_1, 0) = (0, R, q_f) \\ \pi(q_0, 1) = (1, R, q_0) & \pi(q_1, 1) = (0, L, q_1) \end{array}$$

Вычисление M на слове 1101:

$$\begin{array}{rlll} (0, & q_0, & 11010) & \to_M \\ (01, & q_0, & 1010) & \to_M \\ (011, & q_0, & 010) & \to_M \\ (01, & q_1, & 1010) & \to_M \\ (0, & q_1, & 10010) & \to_M \\ (0, & q_1, & 000010) & \end{array}$$



МТ: пример

Пример: $M = (\{0,1\}, 0, \{q_0, q_1, q_f\}, q_0, q_f, \pi)$, где:

$$
\begin{array}{ll}
\pi(q_0, 0) = (0, L, q_1) & \pi(q_1, 0) = (0, R, q_f) \\
\pi(q_0, 1) = (1, R, q_0) & \pi(q_1, 1) = (0, L, q_1)
\end{array}
$$

Вычисление M на слове 1101:

$$
\begin{array}{rlll}
(0, & q_0, & 11010) & \to_M \\
(01, & q_0, & 1010) & \to_M \\
(011, & q_0, & 010) & \to_M \\
(01, & q_1, & 1010) & \to_M \\
(0, & q_1, & 10010) & \to_M \\
(0, & q_1, & 000010) & \to_M \\
(00, & q_f, & 00010) &
\end{array}
$$