Коллоквиум по Дискретной математике, 2 курс

Залялов Александр, @bcategorytheory,
Солодовников Никита, @applied_memes
Шморгунов Александр, @Owlus

10 Определение машин Тьюринга и вычислимых на машинах Тьюринга функций. Тезис Чёрча-Тьюринга. Неразрешимость проблемы остановки машины Тьюринга.

Машина Тьюринга задаётся[1]

- непустым конечным алфавитом Σ, среди которого выделен пробельный символ $\textvisiblespace$ и не содержащее пробела подмножество Γ — входной алфавит;
- непустым конечным множеством состояний Q, среди которых выделено начальное состояние s_0 и множество терминальных состояний F;
- функцией переходов $\delta : (Q \setminus F) \times \Sigma \to Q \times \Sigma \times \{-1, 0, +1\}$.

Машина Тьюринга состоит из бесконечной ленты, разбитой на ячейки, головки, в любой момент времени указывающей на одну ячейку и одной ячейки памяти, в которой хранится текущее состояние. В начальный момент времени на ленте записано некоторое слово, составленное из букв входного алфавита, головка смотрит на первый символ этого слова, во всех остальных ячейках пробелы. Затем в каждый момент времени вычисляется $\delta(q, c) = (q', c', \Delta)$, где q — текущее состояние, c — символ записанный в ячейке, на которую сейчас смотрит головка. Состояние меняется на q', символ в текущей ячейке на c', головка остаётся на месте или передвигается на один влево или вправо в соответствии со значением Δ. Если q' оказалось терминальным, на этом работа машины заканчивается, иначе этот процесс продолжается.

Машины Тьюринга естественным образом отождествляются с частичными функциями $f : \Gamma^* \to \Gamma^*$ — аргументом функции является входное слово, а возвращает функция слово, записанное на ленте после завершения работы машины(то есть всё, что написано на ленте, кроме бесконечного числа пробелов слева и справа). Функции будут частичными, поскольку машина Тьюринга может продолжать работать бесконечно или в данной конструкции может оказаться, что на выходе есть символ, не содержащийся в Γ. Функции, которые можно таким образом получить по некоторой машине Тьюринга, называются вычислимыми на машине Тьюринга.

Тезис Чёрча-Тьюринга. *Любая вычислимая функция вычислима на машине Тьюринга.*

Здесь понятие "вычислимая функция"используется в неформальном смысле, под ним понимается функция, вычислимая в любой разумной модели, которая может прийти вам в голову. Тезис не является формальным утверждением, он никак не доказывается и принимается нами на веру.

Теорема. *Не существует вычислимой функции, определяющей по машине Тьюринга и входному слову, остановится ли эта машина.*

Теперь, когда мы отождествили вычислимые и вычислимые на машине Тьюринга функции, эта теорема непосредственно следует из доказательства теоремы о существовании полного перечислимого множества из 7 билета.

[1] Здесь машина Тьюринга определяется в соответсвии с лекцией. Следует понимать, что это определение не является общепринятым. Вариаций масса: кто-то запрещает головке оставаться на месте, кто-то выделяет выходной алфавит, отличный от входного и т. д.

11 Неразрешимость проблемы достижимости в односторонних ассоциативных исчислениях. Полугруппы, заданные порождающими и соотношениями. Теорема Маркова–Поста: неразрешимость проблемы равенства слов в некоторой конечно определенной полугруппе (без доказательства).

Определение. *Односторонним ассоциативным исчислением* называется множество из всех слов над некоторым конечным алфавитом и конечный набор подстановок. Каждая подстановка представляет собой пару слов (s, t) и позволяет в любом слове содержащем s как подстроку заменить её на t (но не наоборот).

Теорема. *Существует одностороннее ассоциативное исчисление, в котором не разрешима задача проверить по паре слов, можно ли некоторой последовательностью подстановок перейти от первого ко второму.*

Доказательство. Возьмём некоторую машину Тьюринга M, для которой неразрешима проблема остановки, при чём если такую, что если она останавливается, то на ленте записано пустое слово. Построим по ней одностороннее ассоциативное исчисление, в котором из $[X]$ можно получить Y, если и только если M преобразует X в Y. В качестве алфавита для исчисления возьмём объединение алфавита M и её множества состояний (а также квадратные скобки и символы $\triangleleft, \triangleright$). Будем сопоставлять конфигурациям машины слова исчисления. Если машина находится в состоянии s, на ленте записано слово PQ(конкатенация слов P и Q) и головка указывает на первый символ слова Q, сопоставим такой конфигурации слово $[PsQ]$ в нашем исчислении. Тут важно, что мы считаем, что у машины не пересекаются алфавит и множество состояний. Построим по переходам машины подстановки для исчисления.

Переход МТ	Подстановка одностороннего ассоциативного исчисления
$(s, c) \mapsto (s', c', 0)$	$sc \to s'c'$
$(s, c) \mapsto (s', c', +1)$	$sc \to c's'$
$(s, c) \mapsto (s', c', -1)$	$xsc \to s'xc'$ — для каждого символа x из алфавита машины, а также $[sc \to [s'\_c'$
$(s, \_) \mapsto (s', c', 0)$	$s] \to s'c']$
$(s, \_) \mapsto (s', c', +1)$	$s] \to c's']$
$(s, \_) \mapsto (s', c', -1)$	$xs] \to s'xc']$

Дополнительно к этому введём подстановки, позволяющие получить пустое слово, если машина остановится.

- $f \to \triangleleft$, f — терминальное состояние;
- $c\triangleleft \to \triangleleft, c \neq [$;
- $[\triangleleft \to \triangleright$;
- $\triangleright c \to \triangleright, c \neq]$;
- $\triangleright] \to \varepsilon$(пустое слово).

Это можно было бы реализовать проще без двух дополнительных символов, но так мы получаем, что всегда существует ровно одна последовательностей подстановок, моделирующих работу машины Тьюринга. Осталась одна деталь — мы пообщеали, что мы начнём с $[X]$, а не с $[s_0X]$. Она решается просто — добавлением подстановки $[x \to [s_0x$ для всех символом x из алфавита машины.

Итак, мы свели задачу остановки машины Тьюринга (про которую было известно, что она неразрешима) к задаче достижимости в одностороннем ассоциативном исчислении и показали этим, что эта задача тоже неразрешима. □

Оказывается, если потребовать, чтобы все подстановки были двухсторонними, то задача останется неразрешимой, но доказывать этот факт от нас не требуют. При чём такую задачу можно сформулировать на языке алгебры:

Пусть про некоторую полугруппу известно, что она содержит элементы $a_1, \ldots, a_n$ и в ней выполняются некоторые (конечное количество) равенства вида $a_{i_1}a_{i_2}\ldots a_{i_k} = a_{j_1}a_{j_2}\ldots a_{j_m}$. Обязательно ли в ней выполняется заданное равенство такого же вида?

12 Исчисление высказываний (аксиомы и правила вывода), понятие вывода. Теорема корректности исчисления высказываний

Высказываниями мы называем утверждения, которые либо истинны, либо ложны. При этом если A, B являются высказываниями, то $\neg A, A \vee B, A \wedge B, A \to B$ — тоже высказывания. Из такого определения никак не следует, что высказывания вообще существуют, так что в любом применении исчисления высказываний также описывают некоторые атомарные высказывания. Но нам для доказательства общих фактов это никак не потребуется. Исчисление высказываний задаётся аксиомами и правилами вывода. У нас имеется 11 аксиом:

1. $(A \to (B \to C)) \to ((A \to B) \to (A \to C))$
2. $A \to (B \to A)$
3. $A \wedge B \to A$
4. $A \wedge B \to B$
5. $A \to (B \to A \wedge B)$
6. $A \to A \vee B$
7. $B \to A \vee B$
8. $(A \to C) \to ((B \to C) \to ((A \vee B) \to C))$
9. $\neg A \to (A \to B)$
10. $(A \to B) \to ((A \to \neg B) \to \neg A)$
11. $A \vee \neg A$

и 1 правило вывода(modus ponens)

$$\frac{A \to B,\ A}{B}$$

Вывод в исчислении высказываний — это последовательность из операций двух видов

- Подстановка в некоторую аксиому любых высказываний вместо A, B, C.
- Применение правила вывода. Если уже выведены $A \to B$ и A, можно вывести B

Теорема. *(Корректность исчисления резолюций) Любая формула, которую можно вывести в исчислении высказываний, истинна(тавтологична).*

Здесь мы называем формулу истинной(тавтологичной), если она как булева формула верна при всех значениях входящих в неё переменных. Отметим, что только в этом контексте мы понимаем $\neg, \vee, \wedge$ как привычные логические операции. С точки зрения исчисления высказываний, это просто какие-то символы, всё что мы про них знаем — это аксиомы и правило.

Доказательство. Достаточно убедиться, что все действия, который мы можем производить в ходе вывода не позволяют получить ложное выражение. Во-первых, все аксиомы истинны при любых значениях входящих в них переменных. Во-вторых, если $A \to B$ истинно и A истинно, то B истинно. □

13 Вывод из гипотез. Лемма о дедукции. Полезные производные правила.

Пусть Γ — некоторое множество высказываний(гипотез). Тогда говорят, что формула A выводится из Γ, если её можно вывести, разрешая пользоваться не только аксиомами и правилом вывода, но и высказываниями из Γ. Можно сказать, что у нас появилась третья операция: бесплатно получить формулу из Γ. Обозначение: $\Gamma \vdash A$. В таких терминах можно сказать, что формула, которую можно вывести в исчислении высказываний, выводится из пустого множества гипотез, обозначение: $\vdash A$.

Лемма. $\vdash A \to A$

Доказательство. 1. $A \to (A \to A)$(2 аксиома)

2. $\neg A \to (A \to A)$ (9 аксиома)
3. $A \vee \neg A$ (11 аксиома)
4. $(A \to (A \to A)) \to ((\neg A \to (A \to A)) \to ((A \vee \neg A) \to (A \to A)))$ (8 аксиома, подставлены $A, \neg A, A \to A$)
5. $(\neg A \to (A \to A)) \to (A \vee \neg A) \to (A \to A)$ (modus ponens)
6. $(A \vee \neg A) \to (A \to A)$ (modus ponens)
7. $A \to A$ (modus ponens)

□

Теорема. *(Лемма о дедукции)* $\Gamma \cup \{A\} \vdash B \implies \Gamma \vdash (A \to B)$

Доказательство. Пусть с набором гипотез $\Gamma \cup \{A\}$ мы могли вывести формулу B, последовательно выводя формулы $B_1, B_2, \ldots, B_n, B_n = B$. По индукции докажем, что с набором гипотез Γ можно доказать последовательность $A \to B_1, \ldots, A \to B_n$. Разберём для этого все способы, которыми мы умеем выводить

1. B_i получено как гипотеза. Если $B_i = A$, то по лемме мы сможем вывести $A \to A$. Иначе нам доступен такой вывод: $B_i, B_i \to (A \to B_i), A \to B_i$.
2. B_i получено подставлением формул в аксиому. Работает последовательность из предыдущего пункта.
3. B_i получено по modus ponens из B_j и $B_k (j < i, k < i)$. Тогда без потери общности считаем, что $B_j = B_k \to B_i$. По предположению индукции мы уже вывели $A \to B_k$ и $A \to (B_k \to B_i)$. По первой аксиоме выведем $A \to (B_k \to B_i) \to ((A \to B_k) \to (A \to B_i))$. Дважды применив к этому modus ponens, получим $A \to B_i$.

□

Некоторые производные правила — следствия из леммы о дедукции:

- Из A и B можно вывести $A \wedge B$.
- Если из $\Gamma \cup \{A\}$ можно вывести B и $\neg B$, то из Γ можно вывести $\neg A$ (производное правило доказательства от противного).
- Если из $\Gamma \cup \{A\}$ можно вывести B и из $\Gamma \cup \{\neg A\}$ можно вывести B, то из Γ можно вывести B (производное правило разбора случаев).
- Из $A, \neg A$ можно вывести что угодно.

14 Теорема полноты исчисления высказываний.

Лемма. *Пусть формула A зависит от переменных $p_1, \ldots, p_n$. При этом при $(p_1, \ldots, p_n) = (\varepsilon_1, \ldots, \varepsilon_n)$ формула выдаёт значение ε. Тогда $\{p_1^{\varepsilon_1}, \ldots, p_n^{\varepsilon_n}\} \vdash A^\varepsilon$, где*

$$P^\varepsilon = \begin{cases} \neg P, & \varepsilon = 0 \\ P & \varepsilon = 1 \end{cases}$$

Доказательство. Доказательство индукцией по построению формулы A. Разбираем все способы, которыми она была построена, а для них все значения её составных частей.

1. $A = p$. Очевидно, $p \vdash p$ и $\neg p \vdash \neg p$.
2. $A = B \wedge C$.
   (a) Пусть B и C истинны. Тогда по предположению индукции мы можем вывести B и C и требуется показать, что мы можем вывести $B \wedge C$. Мы умеем это делать по производному правилу.
   (b) Пусть B истинно, а C ложно. Тогда по предположению индукции мы можем вывести B и $\neg C$, и хотим вывести $\neg(B \wedge C)$. Воспользуемся правилом доказательства от противного и добавим себе в гипотезы $B \wedge C$. Из $B \wedge C$ нетрудно вывести C, и мы умеем выводить $\not C$, противоречие достигнуто.

Оставшиеся два случая аналогичны второму.

3. $A = B \vee C$. Если хотя бы одно из B или C истинно, то ясно, что можно вывести $B \vee C$. Пусть теперь мы умеем выводить $\neg B, \neg C$ и нужно вывести $\neg(B \vee C)$. Снова будем выводить от противного и предположим $B \vee C$. Заметим, что из $\neg B, \neg C, B$ можно вывести всё что угодно, и из $\neg B, \neg C, C$ можно вывести всё что угодно. Тогда всё что угодно можно вывести и из $\neg B, \neg C, B \vee C$, в том числе и противоречие.

4. $A = B \to C$. Для случаев с истинным B или ложным C вывод простой — достаточно воспользоваться 2 или 9 аксиомой (и modus ponens). Пусть мы умеем выводить $B, \neg C$ и нужно вывести $\neg(B \to C)$. Опять докажем от противного и по modus ponens из $B, B \to C$ выведем C. Это даст противоречие, поскольку у нас есть $\neg C$.

5. $A = \neg B$. Ясно, что $\neg B \vdash \neg B$. Нужно доказать, что $B \vdash \neg\neg B$. Для этого нужно в очередной раз доказать от противного и вывести из $B, \neg B$ какое-нибудь противоречие. Но $B, \neg B$ уже противоречие.

□

Теорема. *(Полнота исчисления высказываний) Любую тавтологию можно вывести в исчислении высказываний.*

Доказательство. Пусть тавтология A зависит от переменных $p_1, \ldots, p_n$. Тогда по лемме $p_1, \ldots, p_n \vdash A$. И $p_1, \ldots, \neg p_n \vdash A$. И вообще как угодно можно расставить отрицания, потому что A — тавтология. Из двух приведённых фактов по производному правилу $p_1, \ldots, p_{n-1}, p_n \vee \neg p_n \vdash A$. Но $p_n \vee \neg p_n$ можно получить из аксиомы, значит это можно выкинуть из списка гипотез и получить $p_1, \ldots, p_{n-1} \vdash A$. Аналогично начав с $p_1, \ldots, \neg p_{n-1}, p_n \vdash A$ и $p_1, \ldots, \neg p_{n-1}, \neg p_n \vdash A$, мы получим $p_1, \ldots, \neg p_{n-1} \vdash A$. Из этих двух результатов мы сможем избавиться от p_{n-1} и получить $p_1, \ldots, p_{n-2} \vdash A$. Долго повторяя этот процесс, мы избавимся от всех переменных и получим $\vdash A$, а это то, что требовалось. □

15 Исчисление резолюций для опровержения пропозициональных формул в конъюнктивной нормальной форме (КНФ): дизъюнкты, правило резолюции, опровержение КНФ в исчислении резолюций. Теорема корректности исчисления резолюций (для пропозициональных формул в КНФ)

Если исчисление высказываний работало с произвольными формулами, построенными с помощью отрицания, конъюнкции, дизъюнкции и импликации, исчисление резолюций работает только с дизъюнктами.

Определение. *Литерал* — переменная или отрицание переменной

Определение. *Дизъюнкт* — это дизъюнкция по некоторому конечному множеству литералов

Обратите внимание, что в этом определении речь про множество. Хотя мы записываем дизъюнкты как формулы $\lambda_1 \vee \lambda_2 \vee \ldots \vee \lambda_n$, мы считаем, что, к примеру, $\lambda_1 \vee \lambda_2, \lambda_2 \vee \lambda_1, \lambda_1 \vee \lambda_2 \vee \lambda_1$ — это всё один и тот же дизъюнкт.

У исчисления резолюций нет аксиом и есть одно правило — правило резолюции

$$\frac{A \vee p,\ B \vee \neg p}{A \vee B}$$

Отметим, что при применении правила к p и $\neg p$ результатом будет пустой дизъюнкт, который обозначается как $\bot$(или как $\square$).

На записанные в КНФ пропозиональные формулы можно смотреть как на множества дизъюнктов в исчислении резолюций. Будем говорить, что множество дизъюнктов совместно, если есть набор значений переменных, при котором каждый дизъюнкт возвращает истину. Утверждается, что из множества дизъюнктов можно вывести в исчислени пустой дизъюнкт, если и только если множество несовместно.

Теорема. *(Корректность исчисления резолюций) Если множества дизъюнктов можно вывести пустой дизъюнкт, то оно несовместно.*

Доказательство. Можно убедиться, что из истинных (при каких-то значениях переменных) формул можно вывести только истинные (при тех же значениях). Но пустой дизъюнкт всегда ложен.

Если вам по каким-либо причинам не нравятся слова о ложности пустого дизъюнкта, можно сказать, что пустой дизъюнкт можно вывести только из $p, \neg p$, а они не могут быть истинны одновременно. □

16 Теорема полноты исчисления резолюций (для пропозициональных формул в КНФ). Доказательство только для конечных и счетных множеств формул.

Теорема. *(Полнота исчисления резолюций) Если множество дизъюнктов S несовместно, то из него можно вывести пустой дизъюнкт.*

Докажем для случая, когда S не более чем счётно.

Доказательство. Применим контрапозицию и докажем, что если из S нельзя вывести пустой дизъюнкт, то оно совместно. Обозначим за S' множество всех формул, которые можно вывести из S. Поскольку множество дизъюнктов не более чем счётно, а сами дизъюнкты конечны, множество используемых переменных тоже будет не более чем счётно. Занумеруем их $x_1, x_2, \ldots$. Докажем, что можно так выбрать значения переменным, что для любого n все дизъюнкты из S', содержащие только переменные с номерами не больше n, истинны. Ясно, что это и означает совместность. Доказывать будем индукцией по n.

База индукции. Это могло бы быть неверно для $n = 1$, только если бы в S' содержались x_1 и $\neg x_1$. Но такого быть не может, ведь тогда мы могли бы вывести пустой дизъюнкт.

Шаг индукции. По предположению индукции мы уже как-то умеем выбирать значения для переменных $x_1, \ldots, x_n$. Предположим, выбрать значение для x_{n+1} нельзя.

- $x_{n+1} = 0$ не подходит $\implies A \vee x_{n+1} \in S'$, где A содержит только $x_1, \ldots, x_n$ и ложно при выбранных для них значениях.
- $x_{n+1} = 1$ не подходит $\implies B \vee \neg x_{n+1} \in S'$, где B содержит только $x_1, \ldots, x_n$ и ложно при выбранных для них значениях.

Но тогда можно вывести $A \vee B$. Поскольку $A \vee B$ содержит только $x_1, \ldots, x_n$, по предположению индукции оно верно при выбранных значениях. А значит не может быть, что и A, и B ложны, противоречие. □

На самом деле аналогичным образом можно было бы доказать корректность и для несчётных множеств формул (для корректности индукции пришлось бы прибегнуть к теореме Цермело), но несчётное число переменных — это крайне нетипичная ситуация и этим мы тут не занимаемся.

17 Полиномиальный алгоритм сведения задачи распознавания совместности конечных множеств произвольных формул к задаче распознавания совместности конечных множеств дизъюнктов.

Перейти от формулы к конечному множеству дизъюнктов — то же самое, что привести её к КНФ. Ясно, что формулу длины l, зависящую от m переменных, можно за $O(2^m \cdot l)$ — построить таблицу истинности и взять дизъюнкты, соответствующие строкам, в которых формула ложна, но этот метод не полиномиальный.

Пусть наша формула имеет вид $f(A, B)$ — где A, B — некоторые формулы, а f — операция, выполняющаяся в нашей формуле последней. Тогда введём новые переменные x', x'' и заменим нашу формулу на $f(x', x'') \wedge (x' \equiv A) \wedge (x'' \equiv B)$. Длины формулы $f(x', x'')$ —константа, следовательно экспоненциальный метод сведёт её к КНФ за $O(1)$. Повторим процедуру для формул $x' \equiv A$ и $x'' \equiv B$, если A и B – это не просто переменные. За каждый запуск наивного алгоритма мы избавляемся от одной операции в исходной формуле, поэтому время работы можно оценить как $O(l)$.

Мы дали описание для бинарных операций, но ясно, что тот же самый подход замены выражений на переменные применим и для отрицания, и для каких-то экзотических операций.