Детерміновані класи складності за часом. Клас складності $\mathcal{P}$

Андрій Фесенко

Клас складності *DTIME*

Означення (*DTIME*)

Для довільної конструктивної за часом функції $t\colon \mathbb{N} \to \mathbb{N}$ **класом складності** $DTIME(t)$ називають множину таких мов $L_1 \subseteq \{0, 1\}^*$, для яких існує детермінована машина Тюрінга M, що вирішує мову L_1 за час $\mathcal{O}(t(n))$.

Клас складності *DTIME*

Означення (*DTIME*)

Для довільної конструктивної за часом функції $t\colon \mathbb{N} \to \mathbb{N}$ **класом складності** $DTIME(t)$ називають множину таких мов $L_1 \subseteq \{0,1\}^*$, для яких існує детермінована машина Тюрінга M, що вирішує мову L_1 за час $\mathcal{O}(t(n))$.

Зауваження

- також позначають як $DTIME(t(n))$ ($TIME(t(n))$, $TIME(t)$)

Клас складності *DTIME*

Означення (*DTIME*)

Для довільної конструктивної за часом функції $t\colon \mathbb{N} \to \mathbb{N}$ **класом складності** $DTIME(t)$ називають множину таких мов $L_1 \subseteq \{0, 1\}^*$, для яких існує детермінована машина Тюрінга M, що вирішує мову L_1 за час $\mathcal{O}(t(n))$.

Зауваження

- також позначають як $DTIME(t(n))$ ($TIME(t(n))$, $TIME(t)$)
- можна використовувати не конструктивні за часом функції

Клас складності *DTIME*

Означення (*DTIME*)

Для довільної конструктивної за часом функції $t\colon \mathbb{N} \to \mathbb{N}$ **класом складності** $DTIME(t)$ називають множину таких мов $L_1 \subseteq \{0, 1\}^*$, для яких існує детермінована машина Тюрінга M, що вирішує мову L_1 за час $\mathcal{O}(t(n))$.

Зауваження

- також позначають як $DTIME(t(n))$ ($TIME(t(n))$, $TIME(t)$)
- можна використовувати не конструктивні за часом функції
- теорема про лінійне прискорення призводить до $\mathcal{O}$ нотації

Клас складності *DTIME*

Означення (*DTIME*)

Для довільної конструктивної за часом функції $t\colon \mathbb{N} \to \mathbb{N}$ **класом складності** $DTIME(t)$ називають множину таких мов $L_1 \subseteq \{0, 1\}^*$, для яких існує детермінована машина Тюрінга M, що вирішує мову L_1 за час $\mathcal{O}(t(n))$.

Зауваження

- також позначають як $DTIME(t(n))$ ($TIME(t(n))$, $TIME(t)$)
- можна використовувати не конструктивні за часом функції
- теорема про лінійне прискорення призводить до $\mathcal{O}$ нотації
- $DTIME^{strict}(t)$ — вирішує мову за час $t(n)$

Клас складності *DTIME*

Означення (*DTIME*)

Для довільної конструктивної за часом функції $t\colon \mathbb{N} \to \mathbb{N}$ **класом складності** $DTIME(t)$ називають множину таких мов $L_1 \subseteq \{0,1\}^*$, для яких існує детермінована машина Тюрінга M, що вирішує мову L_1 за час $\mathcal{O}(t(n))$.

Зауваження

- також позначають як $DTIME(t(n))$ ($TIME(t(n))$, $TIME(t)$)
- можна використовувати не конструктивні за часом функції
- теорема про лінійне прискорення призводить до $\mathcal{O}$ нотації
- $DTIME^{strict}(t)$ — вирішує мову за час $t(n)$
- **може** залежати від обраної моделі обчислень $DTIME_1(t)$, $DTIME_k(t)$, $DTIME_*(t)$

Детерміновані класи складності за часом

$REALTIME = DTIME^{strict}(id)$
$LIN = DTIME(n)$ (лінійний)
$NLT = DTIME(n(\log n)^{\mathcal{O}(1)})$ (майже лінійний)
$QL = \bigcup_{k\in\mathbb{N}} DTIME(n(\log n)^k + k)$ (квазі-лінійний)
$P = \bigcup_{k\in\mathbb{N}} DTIME(n^k) = DTIME(n^{\mathcal{O}(1)})$ або $PTIME$
$QPLIN = DTIME\left(n^{\mathcal{O}(\log n)}\right)$ (лінійний квазі-поліноміальний)
$QP = \bigcup_{k\in\mathbb{N}} DTIME(2^{\log^k n})$ (квазі-поліноміальний)
$SUBEXP = \bigcap_{\varepsilon\in\mathbb{R}^+} DTIME(2^{n^\varepsilon})$ (субекспоненційний)
$E = \bigcup_{k\in\mathbb{N}} DTIME(k^n) = DTIME(2^{\mathcal{O}(n)})$ або $ETIME$
$EE = \bigcup_{k\in\mathbb{N}} DTIME(k^{k^n}) = DTIME\left(2^{2^{\mathcal{O}(n)}}\right)$
$EEE = \bigcup_{k\in\mathbb{N}} DTIME(k^{k^{k^n}}) = DTIME\left(2^{2^{2^{\mathcal{O}(n)}}}\right)$
$EXP = \bigcup_{k\in\mathbb{N}} DTIME(2^{n^k})$ або $EXPTIME$
$2\text{-}EXP = \bigcup_{k\in\mathbb{N}} DTIME(2^{2^{n^k}})$ або $2\text{-}EXPTIME$, $EEXP$
$m\text{-}EXP = \bigcup_{k\in\mathbb{N}} DTIME(2 \uparrow\uparrow m_m^{n^k})$ або $m\text{-}EXPTIME$
$ELEMENTARY = \bigcup_{m\in\mathbb{N}} m\text{-}EXP$
PR — клас примітивно рекурсивних мов

Сучасні можливості комп'ютерів

Лідер списку TOP500 (`top500.org`)

Найкращий суперкомп'ютер на даний момент (11.2021)

Supercomputer Fugaku, A64FX 48C 2.2GHz має теоретичну швидкодію 0.5 exaFLOPS $\approx 5 \cdot 10^{17}$ FLOPS

Сучасні можливості комп'ютерів

Лідер списку TOP500 (`top500.org`)

Найкращий суперкомп'ютер на даний момент (11.2021) **Supercomputer Fugaku, A64FX 48C 2.2GHz** має теоретичну швидкодію 0.5 exaFLOPS $\approx 5 \cdot 10^{17}$ FLOPS

t	$n = 10$	$n = 100$	$n = 1000$
n	$2 \cdot 10^{-17}$ с	$2 \cdot 10^{-16}$ с	$2 \cdot 10^{-15}$ с
n^2	$2 \cdot 10^{-16}$ с	$2 \cdot 10^{-14}$ с	$2 \cdot 10^{-12}$ с
n^3	$2 \cdot 10^{-15}$ с	$2 \cdot 10^{-12}$ с	$2 \cdot 10^{-9}$ с
n^6	$2 \cdot 10^{-12}$ с	$2 \cdot 10^{-6}$ с	2 с
n^{12}	$2 \cdot 10^{-6}$ с	24 дні	$7 \cdot 10^{10}$ років
2^n	$2 \cdot 10^{-15}$ с	70000 років	$7 \cdot 10^{275}$ років
3^n	$1.2 \cdot 10^{-13}$ с	$3 \cdot 10^{22}$ років	10^{452} років
2^{2n}	$2 \cdot 10^{-12}$ с	10^{35} років	$7 \cdot 10^{576}$ років

Сучасні можливості комп'ютерів

t	$n = 10$	$n = 100$	$n = 1000$
n	$2 \cdot 10^{-17}$ с	$2 \cdot 10^{-16}$ с	$2 \cdot 10^{-15}$ с
n^2	$2 \cdot 10^{-16}$ с	$2 \cdot 10^{-14}$ с	$2 \cdot 10^{-12}$ с
n^3	$2 \cdot 10^{-15}$ с	$2 \cdot 10^{-12}$ с	$2 \cdot 10^{-9}$ с
n^6	$2 \cdot 10^{-12}$ с	$2 \cdot 10^{-6}$ с	2 с
n^{12}	$2 \cdot 10^{-6}$ с	24 дні	$7 \cdot 10^{10}$ років
2^n	$2 \cdot 10^{-15}$ с	70000 років	$7 \cdot 10^{275}$ років
3^n	$1.2 \cdot 10^{-13}$ с	$3 \cdot 10^{22}$ років	10^{452} років
2^{2n}	$2 \cdot 10^{-12}$ с	10^{35} років	$7 \cdot 10^{576}$ років

Вік Всесвіту становить ≈ 13.8 мільярдів років або $\approx 13.8 \cdot 10^9$ років
$7 \cdot 10^{275}$ років — це $7 \cdot 10^{266}$ мільярдів років

Вплив збільшення швидкодії

Закон Гордона Мура

Кількість транзисторів на кристалі подвоюється кожні 2 роки (1975)
Швидкодія: подвоюється кожні 2 роки (*Давид Хаус*)
або збільшується в $\sqrt{2}$ кожні 2 роки (*Фред Поллак*)

Вплив збільшення швидкодії

Закон Гордона Мура

Кількість транзисторів на кристалі подвоюється кожні 2 роки (1975)
Швидкодія: подвоюється кожні 2 роки (*Давид Хаус*)
або збільшується в $\sqrt{2}$ кожні 2 роки (*Фред Поллак*)

t	сьогодні	в 100 разів (≈ 26 років)	в 1000 разів (≈ 40 років)	в 10000 разів (≈ 53 роки)
n	N_1	$100N_1$	$1000N_1$	$10000N_1$
n^2	N_2	$10N_2$	$31.6N_2$	$100N_2$
n^3	N_3	$4.6N_3$	$10N_3$	$21.5N_3$
n^6	N_4	$2.1N_4$	$3.2N_4$	$4.6N_4$
n^{12}	N_5	$1.5N_5$	$1.8N_5$	$2.2N_5$
2^n	N_6	$N_6 + 6.64$	$N_6 + 9.97$	$N_6 + 13.29$
3^n	N_7	$N_7 + 4.19$	$N_7 + 6.29$	$N_7 + 8.38$
2^{2n}	N_8	$N_8 + 3.32$	$N_8 + 4.98$	$N_8 + 6.64$

Задачі, які розв'язуються ефективно

Теза Кобхема(Кобхема-Едмондса), 1965 р.

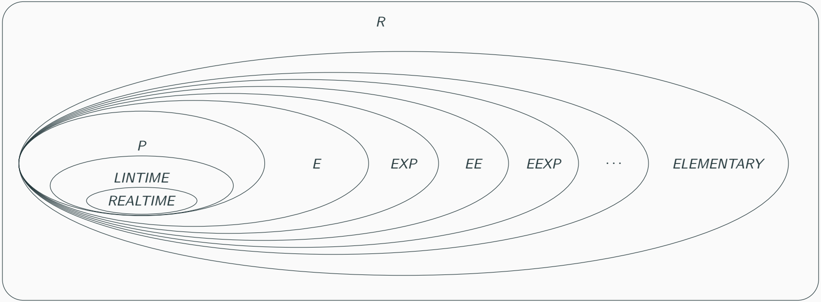
Ефективним розв'язком є тільки розв'язок з поліноміальним обмеженням на використовувані ресурси.

P — клас мов (задач), які розв'язуються **ефективно** (tractable problems)
EXP — клас мов (задач), які **не розв'язуються ефективно**

Властивості класу складності P

Pros	Cons
Всі обчислювальні моделі є поліноміально еквівалентними	Можлива складність $10^{100} n^{2000}$
Всі "розумні" схеми кодування є поліноміально еквівалентними	Можлива складність $10^{-100} 2^n$
P є найменшим незмінним класом, який містить $DTIME(n)$	Відсутність конструктивного алгоритму
На практиці $\mathcal{O}(n^2)$ або $\mathcal{O}(n^3)$	
Замкненість відносно операцій та композиції	

Діаграма класів складності за часом



Поліноміальне зведення

Клас складності P є замкненим відносно перетину, об'єднання, конкатенації, замикання Кліні, обернення та доповнення мов.

Поліноміальне зведення

Клас складності P є замкненим відносно перетину, об'єднання, конкатенації, замикання Кліні, обернення та доповнення мов.

Для довільних нетривіальних мов $L_1, L_2 \in P$ є правильним твердження, що $L_1 \leq_m L_2$.

Поліноміальне зведення

Клас складності P є замкненим відносно перетину, об'єднання, конкатенації, замикання Кліні, обернення та доповнення мов.

Для довільних нетривіальних мов $L_1, L_2 \in P$ є правильним твердження, що $L_1 \leq_m L_2$.

Означення

Мова $L_1 \subseteq \{0, 1\}^*$ **поліноміально** (або **за Карпом**) зводиться до мови $L_2 \subseteq \{0, 1\}^*$, якщо існує m-зведення мови L_1 до мови L_2, в якому для функції зведення існує поліноміальна детермінована машина Тюрінга, яка обчислює значення функції зведення. Позначають це як $L_1 \leq_p L_2$ (іноді позначають як $L_1 \propto L_2$ або $L_1 \leq_m^P L_2$).

Зведення за Куком

Означення

Мова $L_1 \subseteq \{0, 1\}^*$ **за Куком** (або **поліноміально за Тюрінгом**) зводиться до мови $L_2 \subseteq \{0, 1\}^*$, якщо мова L_1 зводиться за Тюрінгом до мови L_2 за час $T(n)$ і при цьому існує поліном $p \colon \mathbb{N} \to \mathbb{N}$ такий, що $T(n) \leq p(n)$ для всіх чисел $n \in \mathbb{N}$. Позначають це як $L_1 \leq_C L_2$ (іноді позначають як $L_1 \leq_T^P L_2$).

Зведення за Куком

Означення

Мова $L_1 \subseteq \{0,1\}^*$ **за Куком** (або **поліноміально за Тюрінгом**) зводиться до мови $L_2 \subseteq \{0,1\}^*$, якщо мова L_1 зводиться за Тюрінгом до мови L_2 за час $T(n)$ і при цьому існує поліном $p\colon \mathbb{N} \to \mathbb{N}$ такий, що $T(n) \leq p(n)$ для всіх чисел $n \in \mathbb{N}$. Позначають це як $L_1 \leq_C L_2$ (іноді позначають як $L_1 \leq_T^P L_2$).

Наслідок

Зведення за Карпом та зведення за Куком є рефлексивними та транзитивними відношеннями на множині всіх мов.

Зведення за Куком

Означення

Мова $L_1 \subseteq \{0,1\}^*$ **за Куком** (або **поліноміально за Тюрінгом**) зводиться до мови $L_2 \subseteq \{0,1\}^*$, якщо мова L_1 зводиться за Тюрінгом до мови L_2 за час $T(n)$ і при цьому існує поліном $p\colon \mathbb{N} \to \mathbb{N}$ такий, що $T(n) \le p(n)$ для всіх чисел $n \in \mathbb{N}$. Позначають це як $L_1 \le_C L_2$ (іноді позначають як $L_1 \le_T^P L_2$).

Наслідок

Зведення за Карпом та зведення за Куком є рефлексивними та транзитивними відношеннями на множині всіх мов.

Твердження

Поліноміальне зведення є сильнішим за m-зведення.
Зведення за Куком є сильнішим за зведення за Тюрінгом.
Поліноміальне зведення є сильнішим за зведення за Куком.
$L_1 \le_p L_2 \Rightarrow L_1 \le_m L_2$, $L_1 \le_p L_2 \Rightarrow L_1 \le_C L_2 \Rightarrow L_1 \le_T L_2$

Властивості зведень

Клас складності P є замкненим відносно поліноміального зведення та зведення за Куком.

Властивості зведень

Клас складності P є замкненим відносно поліноміального зведення та зведення за Куком.

- клас складності P є незамкнутим відносно зведення за Куком, якщо розглянути множину функціональних задач — 2^{2^k} через піднесення до квадрату;
- але зведення за Куком є корисним при зведенні інших типів задач до задач розпізнавання

Властивості зведень

Клас складності P є замкненим відносно поліноміального зведення та зведення за Куком.

- клас складності P є незамкнутим відносно зведення за Куком, якщо розглянути множину функціональних задач — 2^{2^k} через піднесення до квадрату;
- але зведення за Куком є корисним при зведенні інших типів задач до задач розпізнавання

Довільна мова $L_1 \in P$ поліноміально зводиться до довільної нетривіальної мови $L_2 \subset \{0, 1\}^*$.