

A digitális számítás elmélete

9. előadás április 4.

- Függvények aszimptotikus összehasonlítása
- Algoritmusok (TG-ek) idő és tárigénye
- A P osztály
- Tár-idő tétel

Hogyan értelmezzük algoritmusok hatékonyságát?

- Eldönthető nyelv (feladat) felismerésére van-e praktikusán használható algoritmus?

Milyen fajta feladatok vannak?

Eldöntendő kérdés Pl. prímszám-e az adott szám?

– nyelvbe tartozás

Hozzárendelés Pl. lnko, Hamilton-kör keresés

– fv kiszámítás

- Egy problémára (pl. szám prím voltának eldöntése, utazó ügynök) adott algoritmusokat hogyan értékeljük, hasonlítsuk össze?

Algoritmusok összehasonlítása

1. Időigény (lépésszám) illetve tárigény szerinti összehasonlítás
2. A feladat nagyságának (a megadás hosszának) függvényében, a nagy feladatokra koncentrálnak (asszimptotikus viselkedés)
3. A legrosszabb esetet tekintve (worst case)
4. Azonos fajta TG-pel és be és kimeneti reprezentációval megvalósított algoritmusok (ezt majd még pontosítjuk)

Algoritmusok összehasonlítása

Pl: T. fel hogy a gép 10^{12} művelet/sec sebességű. A és B algoritmust hasonlítjuk össze.

időigény	1 perc alatt	1 óra alatt
A $3000 n^5$	$n = 114$	$n = 260$
B 2^n	$n = 45$	$n = 52$

A 'nagy ordó' jelölés függvények asszimptotikus viselkedésének jellemzésére

Def: $f, g: \mathbb{N} \rightarrow \mathbb{R}^+$ függvények. $f(n) = O(g(n))$, ha található

c és n_0 pozitív egészek, hogy

$$f(n) \leq c g(n) \text{ ha } n \geq n_0.$$

Azt mondjuk, hogy $g(n)$ **asszimptotikus felső korlátja** $f(n)$ -nek.

<u>Pl:</u>	$f(n)$	$g(n)$
	$6n^3 + 3000n^2 + 10^{100}$	n^3
	$n^{100} + 2^n$	2^n
	$\log n + n$	n
	$\log_2 n + \log_3 n$	$\log n$ nem számít az alap

TG időkorlátja

Jelölés: Egy adott M TG esetén:

$T_M(n)$ jelöli M maximális lépésszámát (számolási idejét) az n hosszúságú bemenetek esetén.

$S_M(n)$ jelöli az M által maximálisan elolvasott munkaszalag jelek számát n hosszúságú bemenetek esetén.

Megj: Ha M nem áll meg valamely n hosszú bemenetre, akkor $T_M(n)$ végtelen. Ekkor $S_M(n)$ -t is végtelennek tekintjük.

Definíció: Legyen $t: \mathbb{N} \rightarrow \mathbb{N}$ függvény olyan, hogy $t(n) \geq n$ minden n -re. Az M TG **$t(n)$ időkorlátos**, ha az n hosszú bemenetek esetén legfeljebb $t(n)$ lépést tesz meg, azaz $T_M(n) \leq t(n)$.

Definíció: $\text{TIME}(t(n)) = \{L \mid L \text{ felismerhető egy } O(t(n)) \text{ időkorlátos } M \text{ TG-pel.}\}.$

TG tárkorlátja

Definíció: Legyen $s: \mathbb{N} \rightarrow \mathbb{N}$ függvény olyan, hogy $s(n) \geq \log_2 n$ minden n -re. Az M TG **$s(n)$ tárkorlátos**, ha az n hosszú bemenetek esetén legfeljebb $s(n)$ db cellát használ a munkaszalagon, azaz $S_M(n) \leq s(n)$.

Definíció: $\text{SPACE}(s(n)) = \{L \mid L \text{ felismerhető egy } O(s(n)) \text{ tárkorlátos } M \text{ TG-pel.}\}.$

TG időkorlátja

P1: $L = \{0^k 1^k \mid k \geq 0\}$. Egy L -et eldöntő TG:

1. Keres 0 jelet 1 után, ha van elutasít, ha nincs, visszamegy a szalag elejére.
2. Ha van még 0 és 1 a szalagon, ismételd:
Menj a szalag elejére, és húzd át a legelső 0-t és 1-t.
3. Ha maradt áthúzatlan 0 vagy 1, elutasít, különben **elfogad**.

Időigény: $O(n^2)$

Megj: 2-szalaggal $O(n)$ lépésben eldönthető a nyelv

A **P** osztály

P1: $\text{TIME}(n)$ a lineáris időben felismerhető nyelvek osztálya.

$\text{TIME}(n^2)$ a négyzetes időben felismerhető nyelvek osztálya.

Definíció: $\mathbf{P} = \bigcup_{k \geq 1} \text{TIME}(n^k)$ a polinom időben
egyszalagos TG-pel eldönthető nyelvek osztálya.

1. Megj: Valamely, az alappal ekvivalens speciális TG modellt
használva, ha az polinom időben szimulálható alap TG-pel, akkor
 $\mathbf{P}$ invariáns arra nézve, hogy az alap vagy a speciális TG-et
használjuk számítási modellként.

2. Megj: $\mathbf{P}$ a praktikus számítógéppel megoldható problémáknak
felel meg. Általában találunk $\mathbf{P}$ -beli problémákra gyors
(négyzetes, köbös,...) algoritmust

Példák P-beli problémákra

Ismert polinomiális algoritmus az alábbi problémákra:

PATH = $\{ \langle G, s, t \rangle \mid G \text{ irányított gráf, és van benne irányított út } s\text{-ből } t\text{-be} \}$.

RELRPIME = $\{ \langle x, y \rangle \mid x \text{ és } y \text{ relatív prímek} \}$.

GRAPH-2-COLOR = $\{ G \mid G \text{ gráf csúcsai kiszínezhetők 2 színnel} \}$.

EULER-CYCLE = $\{ G \mid G \text{ gráfban van Euler kör} \}$.

PRIME = $\{ x \mid x \text{ prím} \}$. **2002-es eredmény!**

Példák mindmáig nem **P**-beli problémákra

Nem sikerült eddig polinomiális algoritmus találni pl. az alábbi problémákra:

HAMILTON-CYCLE = $\{G \mid G \text{ gráfban van Hamilton kör}\}$.

GRAPH-3-COLOR = $\{G \mid G \text{ gráf csúcsai kiszínezhetők 3 színnel}\}$.

TG-ek szimulációjának idő és tárigénye

Láttuk, hogy speciális TG-ek (többszalagos, RAM, ... nem determinisztikus) szimulálhatók alap TG-pel. Ehhez több idő illetve tár kell, mennyivel? Mi az egyszerűbb gépeken a hatékonyságromlás mértéke?

Tétel: Az N k -szalagos TG-hez van olyan egyszalagos M TG, hogy $L(M) = L(N)$, és ha N $f(n)$ időigényű, akkor M $O(f(n)^2)$ időigényű.

Biz: nem részletezzük, a korábbi konstrukcióhoz hasonló

TG-ek szimulációjának idő és tárigénye

Tétel: Az N TG-hez van olyan többszalagos M TG, hogy egy alkalmas n_0 számon túl

a) tetszőleges $\varepsilon > 0$ esetén $T_M(n) \leq n(1 + \varepsilon)$, ha $T_N(n) = O(n)$

b) tetszőleges $c > 0$ esetén $T_M(n) \leq c T_N(n)$, ha $\lim T_N(n)/n = \infty$

Biz: nem részletezzük.

Megj: A fenti tétel magyarázza, hogy miért tekinthettünk el a konstans szorzótól az időigény definiálásakor.

Tár-idő tétel

Tétel: Ha $L \in \text{SPACE}(s(n))$, akkor van olyan L -től függő c konstans, hogy $L \in \text{TIME}(2^{s(n)})$, .

Biz: (vázlat) Legyen M egy L -et $O(s(n))$ tárkorláttal eldöntő Turing-gép és legyen w az M egy tetszőleges n hosszú bemenete. Ekkor a kezdőkonfigurációból elérhető konfigurációk száma $cn2^{O(s(n))} = 2^{O(s(n))}$, ahol c egy a szalagábécétől függő konstans.

Mivel M $2^{O(s(n))}$ lépése után konfiguráció ismétlődés áll fenn, egyes konstrukcióval elérhető, hogy az M -et szimuláló N Turing-gép $2^{O(s(n))}$ lépés után észrevegye ha végtelen ciklusba esik. Következik, hogy N az L -et ismeri fel és $2^{O(s(n))}$ időigényű, ami bizonyítja a tétel állítását.

Nyelvosztályok komplementere

Jelölés: L I feletti nyelv esetén $L^C = I^* \setminus L$ az L komplementere.

Def: $\mathcal{L}$ nyelvosztály esetén $\text{co}\mathcal{L} = \{L \mid L^C \in \mathcal{L}\}$ az $\mathcal{L}$ nyelvosztály komplementere.

Tétel:

a) $\text{co}(\text{co}\mathcal{L}) = \mathcal{L}$

b) Ha $\mathcal{A} \subseteq \mathcal{B}$ nyelvosztályok, akkor $\text{co}\mathcal{A} \subseteq \text{co}\mathcal{B}$.

Biz:

a) $(L^C)^C = L$ alapján.

b) $L \in \text{co}\mathcal{A} \Rightarrow L^C \in \mathcal{A} \Rightarrow L^C \in \mathcal{B} \Rightarrow L \in \text{co}\mathcal{B}$

Tár-idő tételek komplement nyelvosztályokra

Tétel: $\text{TIME}(t(n)) = \text{coTIME}(t(n))$, azaz

$L \in \text{TIME}(t(n))$ akkor és csak akkor, ha $L^C \in \text{TIME}(t(n))$.

Biz: Az L -et eldöntő M TG-ben az elfogadó és elutasító állapotokat felcserélve, egy azonos futásidejű, L^C -t eldöntő TG-et kapunk, tehát $L^C \in \text{TIME}(t(n))$, azaz $\text{coTIME}(t(n)) \subseteq \text{TIME}(t(n))$.

Másrészt $\text{TIME}(t(n)) = \text{co}(\text{coTIME}(t(n))) \subseteq \text{coTIME}(t(n))$.

Tehát $\text{TIME}(t(n)) = \text{coTIME}(t(n))$.

Tár-idő tételek komplement nyelvosztályokra

Tétel: $\text{SPACE}(s(n)) = \text{coSPACE}(s(n))$, azaz

$L \in \text{SPACE}(s(n))$ akkor és csak akkor, ha $L^c \in \text{SPACE}(s(n))$.

Biz: Legyen $L \in \text{SPACE}(s(n))$. A tár-idő tételbeli N gép minden inputra megáll, $O(s(n))$ tárkorlátos.

Ebben felcserélve az elfogadó és elutasító állapotokat, adódik

hogy $L^c \in \text{SPACE}(s(n))$, azaz

$\text{coSPACE}(s(n)) \subseteq \text{SPACE}(s(n))$.

A másik irány mint előbb.

Nevezetes nyelvosztályok

Definíció: **EXPTIME** = $\bigcup_{k \geq 1} \text{TIME}(2^{n^k})$ az exponenciális időben felismerhető nyelvek osztálya.

Definíció: **PSPACE** = $\bigcup_{k \geq 1} \text{SPACE}(n^k)$ a polinom tárban felismerhető nyelvek osztálya.

Tétel: **P** $\subseteq$ **PSPACE** $\subseteq$ **EXPTIME**

Biz: Ha M TG $t(n)$ időkorlátos, akkor $t(n)$ tárkorlátos is (lépésenként max. egy jelet ír a munkaszalagra).

Tehát $\text{TIME}(n^k) \subseteq \text{SPACE}(n^k)$, és így

$$\mathbf{P} = \bigcup_{k \geq 1} \text{TIME}(n^k) \subseteq \bigcup_{k \geq 1} \text{SPACE}(n^k) = \mathbf{PSPACE}$$

Nevezetes nyelvosztályok

Ha $L \in \text{PSPACE}$, akkor valamely k -ra $L \in \text{SPACE}(n^k)$

A tár-idő tétel szerint ekkor hogy

$$L \in \text{TIME}(2^{n^k}) \subseteq \text{EXPTIME}.$$

Tétel: $\text{EXPTIME} \subset \mathbf{R}$ (=Rekurzív nyelvek osztálya)

Biz: $\subseteq$ következik EXPTIME időkorlátosságából.

A valódi tartalmazáshoz mutatunk egy L nyelvet, mely rekurzív, de
nem **EXPTIME**-beli,

Nevezetes nyelvosztályok

$L = \{ w \in \{0, 1\}^*, \text{ az } M_w \text{ gép létezik, és legfeljebb } 2^{|w|} \text{ lépésben elutasítja } w\text{-t, ahol } |w| \text{ a szó hossza.} \}$

1. L -nek végtelen sok eleme van, mivel egy M_w -t kibővíthetünk a kiinduló helyzetből soha nem elérhető állapotokkal. Egy ilyen y leírás esetén M_y éppen úgy fog működni, mint M_w , tehát az időkorlátja is azonos.
2. Megmutatjuk, hogy L semmilyen n -re nincs benne $\text{TIME}(2^{2^{n-1}})$ -ban. Ez elég is, mivel $\text{EXPTIME} \subseteq \text{TIME}(2^{2^{n-1}})$.

Nevezetes nyelvostályok

Indirekte feltesszük, hogy L felismerhető $c2^{2^{n-1}}$ időkorlátos M TG-pel.

Legyen n_0 olyan szám, hogy $n \geq n_0$ esetén $c2^{2^{n-1}} < 2^{2^n}$

Legyen w egy n_0 -nál hosszabb szó, melyre M_w létezik és úgy viselkedik mint M . Most megmutatjuk, hogy mind $w \in L$, mind a $w \notin L$ feltételezés ellentmondásra vezet.

Ha $w \in L$, akkor M_w elfogadja w -t $c2^{2^{n-1}} < 2^{2^n}$ lépésben, de ekkor L def. szerint $w \notin L$

Ha $w \notin L$, akkor M_w nem utasítja el w -t $c2^{2^{n-1}} < 2^{2^n}$ lépésben, de megáll, ezért elfogadja w -t, amiből $w \in L$ következne.