Algoritmuselmélet 12. előadás

Katona Gyula Y.
Budapesti Műszaki és Gazdaságtudományi Egyetem
Számítástudományi Tsz.
I. B. 137/b
kiskat@cs.bme.hu

2002 Április 9.

Turing-gépek

Az algoritmus fogalmának pontosabb meghatározása.
Számítógép elméleti, egyszerűsített modellje.

Alan Turing 1912-1954
Definíció. *Többszalagos Turing-gép (TG), $k \geq 1$ egész szám*

- k *db szalag cellákra osztva, cellákba jelek, a szalag egyik (mindkét) irányban végtelen*
- *minden szalaghoz tartozik egy fej, ami jobbra és balra is lépegethet a szalagon cellánként*
- *véges vezérlő, ennek véges sok állapota van*
- *Lépés: függ a belső állapottól és a fej alatti jelektől*
  - ⋆ *a gép megáll*
  - ⋆ *átmegy új állapotba, ír valamit minden fej alatti cellába, minden fej lép vagy jobbra, vagy balra egyet vagy helyben marad*

Definíció. *Egy* k-szalagos Turing-gép *egy hetessel jellemezhető:*

$$M = (Q, T, \text{ü}, I, q_0, F, \delta),$$

ahol

Q **:** *egy véges halmaz, az* M *gép* belső állapotainak *halmaza.*

T **:** *egy véges halmaz, a* szalagjelek *halmaza.*

ü **:** *egy kitüntetett szalagjel, az* üresjel*. A bemenetként felírt jeleken és a gép számítása során kiírt jeleken kívül minden szalagcellában ü van.*

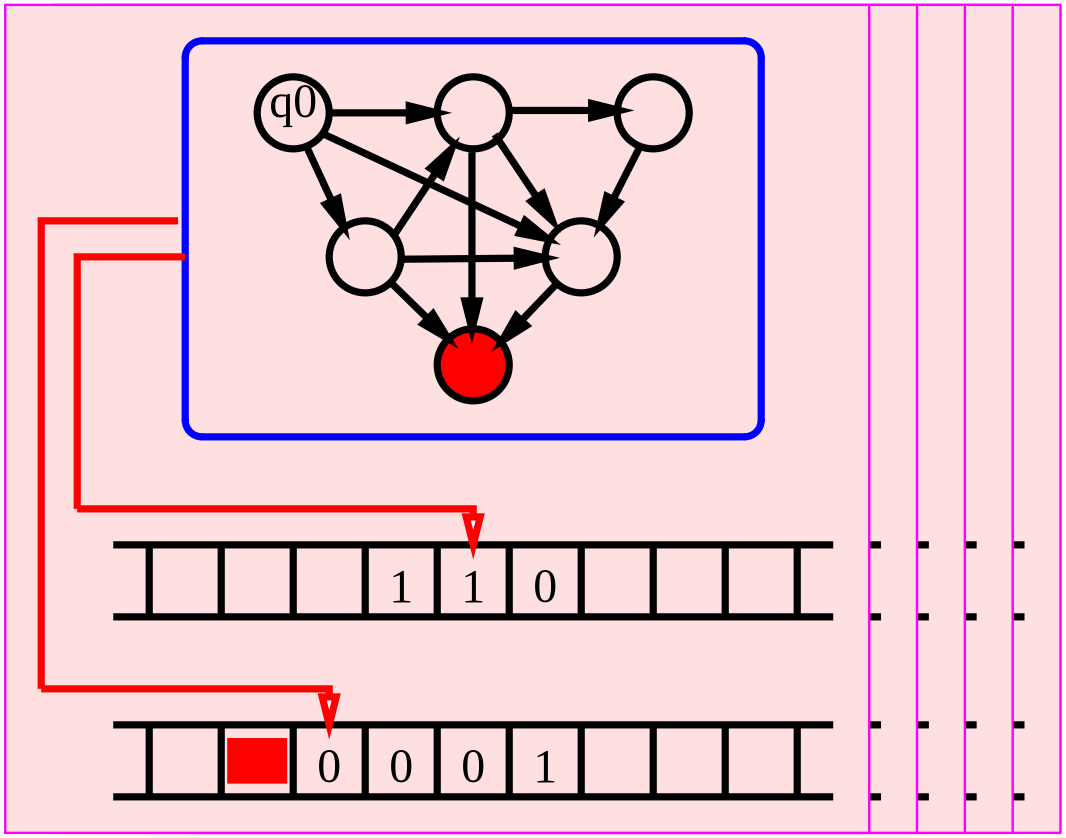
I **:** $I \subseteq T \setminus \{\text{ü}\}$ *az* input jelek *vagy* bemenő jelek *halmaza, más szóval az* *input abc. Az üresjel nem lehet input jel.*

q_0 **:** $q_0 \in Q$ *a* kezdő állapot.

F : $F \subseteq Q$ *az* elfogadó állapotok *halmaza.*
Elfogadó állapotban áll meg $\Longrightarrow$ *IGEN*
Nem elfogadó állapotban áll meg $\Longrightarrow$ *NEM*
$\Longrightarrow$ *a* $Q \setminus F$*-be tartozó állapotokat* elutasító állapotoknak *is nevezik.*

δ : *A* $\delta : Q \times T^k \longrightarrow Q \times (T \times \{\textit{jobb}, \textit{bal}, \textit{helyben}\})^k$ *egy parciálisan értelmezett függvény, a gép* átmenetfüggvénye*.*
Az átmenetfüggvény tekinthető a gép programjának.
Ha a $\delta(q; a_1, a_2, \ldots, a_k)$ *nincs értelmezve,* $\Longrightarrow$ *megállás*

Példa



animáció: Turing gép

Turing-gép működése

- Kezdetben a q_0 állapotban van, az $s \in I^*$ bemenet az első szalag elejére van írva, az összes többi mezőn *ü*
- A δ függvénynek megfelelően lépeget $\Longrightarrow$ új állapot, írás, fej lépése
- Ha δ nincs értelmezve, akkor megáll (akár elfogadó állapotban van, akár nem)

Két felhasználás:

- Függvények kiszámolása
- Kérdések eldöntése ($0 - 1$ értékű függvény)

Definíció. *Az M Turing-gép által* felismert L_M nyelv *azokból az $s \in I^*$ szavakból áll, amelyekkel mint bemenetekkel elindítva az M megáll, mégpedig elfogadó (azaz F-beli) állapotban.*

Ha M az L_M nyelvet ismeri fel, akkor a nyelvbe nem tartozó szavakra vagy megáll elutasító állapotban, vagy végtelen ciklusba kerül.

Definíció. *Legyen* M *egy kitüntetett output szalaggal rendelkező Turing-gép. Az* M *által* kiszámított $f_M : I^* \longrightarrow I^*$ parciális függvényt *így értelmezzük:* $f_M(s) = w$, *ha* M *az* $s \in I^*$ *inputtal indulva megáll, és megállás után az output szalagon a* $w \in I^*$ *szó szerepel az üresjelek óceánja előtt.*

Az f_M függvény tehát csak azokra az $s \in I^*$ szavakra értelmezett, amelyekkel mint bemenetekkel az M gép véges sok lépés megtétele után megáll. Mindegy, hogy a megállás elfogadó vagy elutasító állapotban történt-e.

Példa Turing-gépre

$$Q = \{q_0, q_1, q_2, q_v\},\ T = \{0, 1, \ddot{u}\},\ I = \{0, 1\},\ F = \{q_v\}.$$

$$\begin{aligned} \delta(q_0, 1) &= (q_1, 1, jobb),\ \ \delta(q_0, \ddot{u}) = (q_v, \ddot{u}, helyben), \\ \delta(q_1, 1) &= (q_2, 1, jobb),\ \ \delta(q_2, 1) = (q_0, 1, jobb). \end{aligned}$$

Más párokra δ nincs értelmezve.

Ha 0-t olvas $\Longrightarrow$ elutasít
Ha *ü* üresjelet olvas és nem q_0-ban van $\Longrightarrow$ elutasít
Ha *ü* üresjelet olvas és q_0-ban van $\Longrightarrow$ elfogad
Ha 1-et olvas és q_i-ben van $\Longrightarrow$ jobbra lép, és átmegy a q_{i+1} állapotba;
$\Longrightarrow$ M pontosan azokat a szavakat fogadja el, amelyek csupa egyesekből állnak, és a hosszuk osztható hárommal.
$\Longrightarrow$ Az M által felismert L_M nyelv tehát

$$L_M = \{1^n :\ n \text{ hárommal osztható természetes szám}\}.$$

Példa Turing-gépre

$$Q = \{q_0, q_v\},\ T = \{0, 1, \text{ü}\},\ I = \{0, 1\},\ F = \{q_v\}$$
$$\delta(q_0, 0) = (q_0, 0, helyben),\ \delta(q_0, 1) = (q_0, 1, jobb),$$
$$\delta(q_0, \text{ü}) = (q_v, 1, helyben).$$

Másutt δ nem definiált.

Meghatározzuk az $L_{M'}$ nyelvet és az $f_{M'}$ függvényt is.
Az M' a q_0 belső állapotban maradva lépdel jobbra a szalag mentén, amíg 0 vagy *ü* jelet nem talál.
$0 \implies$ végtelen ciklus
ü $\implies$ még egy 1-est ír az input után, majd elfogad
$\implies L_{M'} = \{1^n; n \geq 0 \text{ egész}\}$
$\implies$ A gép az 1^n bemeneten az 1^{n+1} eredményt adja, vagyis $f_{M'}(1^n) = 1^{n+1}$.

Turing-gép $\iff$ igazi számítógép
Turing-gép többet tud, mint a véges automata.

A kiszámíthatóság alapfogalmai

Algoritmus: ami Turing-géppel kiszámítható

Definíció. *Az* $L \subseteq I^*$ *nyelvet* rekurzíve felsorolhatónak *nevezzük, ha van olyan* M *Turing-gép, melyre* $L = L_M$*, azaz a gép által felismert nyelv éppen* L*.*

Definíció. *Az* $L \subseteq I^*$ *nyelv* rekurzív*, ha van olyan* M *Turing-gép, melyre* $L = L_M$*, és* M *minden* $s \in I^*$ *szóra megáll.*

$$\mathcal{RE} = \{L \subseteq I^* : \quad L \text{ rekurzíve felsorolható}\}.$$

$$\mathcal{R} = \{L \subseteq I^* : \quad L \text{ rekurzív}\}.$$

$\Longrightarrow \mathcal{R} \subseteq \mathcal{RE}$

Definíció. *Az* $f : I^* \longrightarrow I^*$ *parciális függvény* parciálisan rekurzív *függvény, ha létezik olyan* M *Turing-gép, hogy* $f = f_M$*. Ha ezen túl még* f *minden* $s \in I^*$ *inputra értelmezve van, akkor* f *egy* rekurzív *függvény.*

Tétel. *Van olyan $L' \subseteq I^*$ nyelv, amely nem rekurzíve felsorolható.*

Bizonyítás: Egy Turing-gép leírható véges jelsorozattal $\Longrightarrow$ az összes gép számossága megszámlálható $\Longrightarrow$ felsorolható: $M_0, M_1, M_2, \ldots$
$\Longrightarrow$ a rekurzíve felsorolható nyelvek is felsorolhatók: $L_{M_0}, L_{M_1}, L_{M_2}, \ldots$
$\Longrightarrow$ a rekurzíve felsorolható nyelvek halmaza megszámlálható

Belátjuk, hogy az összes nyelv halmaza nem megszámlálható (egyébként kontinuum).

Az I^* elemei, a véges hosszúságú I-beli jelekből képzett szavak is megszámlálhatóak $\Longrightarrow$ felsorolhatóak: $w_0, w_1, w_2, \ldots$
Tegyük fel, hogy az összes nyelvek halmaza megszámlálható, megmutatjuk, hogy van olyan $L' \subseteq I^*$ nyelv, amely nem lehet benne az összes nyelvek $L_{M_0}, L_{M_1}, L_{M_2}, \ldots$ sorozatában.
Az L' nyelvnek a w_i szó pontosan akkor legyen eleme, ha $w_i \notin L_{M_i}$, $i = 1, 2, \ldots$.
$L' \neq L_{M_i}$, hiszen a $w_i \notin L'$ és $w_i \in L_{M_i}$ $\surd$

Cantor-féle átlós módszer

	w_0	w_1	$\ldots$	w_i	w_{i+1}	$\ldots$
L_{M_0}	nem	nem	$\ldots$	nem	nem	$\ldots$
L_{M_1}	igen	nem	$\ldots$	nem	nem	$\ldots$
$\vdots$						
L_{M_i}	nem	igen	$\ldots$	igen	nem	$\ldots$
$L_{M_{i+1}}$	igen	nem	$\ldots$	nem	nem	$\ldots$
$\vdots$						
L'	igen	igen	$\ldots$	nem	igen	$\ldots$

Tétel. *Létezik olyan $f: I^* \longrightarrow I^*$ parciális függvény, amely nem parciálisan rekurzív.*

Church–Turing-tézis

A rekurzív nyelveket és a rekurzív függvényeket fogjuk algoritmussal kezelhető nyelveknek és függvényeknek tekinteni.

Church–Turing-tézis: *Ami algoritmussal – azaz véges eljárással – kiszámítható (eldönthető), az Turing értelmében kiszámítható (eldönthető). Nevezetesen:*

- *Egy* $f : I^* \longrightarrow I^*$ *parciális függvény kiszámítható* $\Leftrightarrow$ f *parciálisan rekurzív.*
- *Egy* $f : I^* \longrightarrow I^*$ *(teljes) függvény kiszámítható* $\Leftrightarrow$ f *rekurzív.*
- *Egy* $L \subseteq I^*$ *nyelvre a nyelvbe tartozás problémája algoritmussal eldönthető* $\Leftrightarrow$ L *rekurzív.*

Idő- és tárigény

Az M Turing-gép *számolási ideje* az s inputon a megállásáig végrehajtott lépések száma *tárigénye* pedig a felhasznált (olvasott) szalagcellák száma.

A tárigénybe nem feltétlenül számítjuk bele az inputot és outputot.
Definíció. *Jelölje $T_M(n)$ az M gép maximális számolási idejét az n jelből álló bemeneteken. Az n hosszú szavakon a maximális tárigényt $S_M(n)$-nel jelöljük.*

Ha van olyan n jelből álló $s \in I^*$ szó, amellyel elindítva M nem áll meg véges sok lépés után $\Longrightarrow T_M(n) = \infty$
Pl. a hárommal oszthatóságot vizsgáló M TG-re: $T_M(n) = n + 1$, $S_M(n) = n + 1$, ha beleszámítjuk az inputot, $S_M(n) = 0$, ha nem.
Ha M és N két Turing-gép, melyekre $T_M(n) < T_N(n)$ teljesül minden elég nagy n-re, akkor az M algoritmust gyorsabbnak mondhatjuk az N algoritmusnál.

Van-e legjobb TG minden L nyelvhez?

Tétel. [gyorsítási tétel] *Van olyan L nyelv, amelyre igazak az alábbiak:*
1. Az L felismerhető egy olyan M Turing-géppel, melyre $T_M(n)$ véges minden n-re.
2. Tetszőleges, az L-et felismerő N Turing-géphez van olyan N' Turing-gép, amelyre $L = L_{N'}$ szintén teljesül, továbbá $T_{N'}(n) = O(\log T_N(n))$.

k-szalagos TG szimulációja 1-szalagossal

Tétel. *Legyen M egy k-szalagos Turing-gép. Van olyan egyszalagos M' Turing-gép, melyre*

$$L_M = L_{M'} \;\; (\textit{vagy } f_M = f_{M'}), \; \textit{továbbá}$$

$$T_{M'}(n) \le 2T_M^2(n),$$
$$S_{M'}(n) \le S_M(n) + n.$$

Bizonyítás: M' építésekor az M-hez képest alaposan felfújjuk a szalag abc-t, és megnöveljük a belső állapotok számát is. Az M' egyetlen szalagján $2k$ csík lesz. Egy cellája egy oszlop.

1. szalag	D	$\cdots$	A	$\cdots$	B	$\cdots$
1. fej	ü	$\cdots$	x	$\cdots$	ü	$\cdots$
.						
.						
.						
k. szalag	D	$\cdots$	D	$\cdots$	B	$\cdots$
k. fej	ü	$\cdots$	ü	$\cdots$	x	$\cdots$

$\Longrightarrow$ szalagjelek száma: $(2t)^k$

1. szalag	D	$\cdots$	A	$\cdots$	B	$\cdots$
1. fej	ü	$\cdots$	x	$\cdots$	ü	$\cdots$
.						
.						
.						
k. szalag	D	$\cdots$	D	$\cdots$	B	$\cdots$
k. fej	ü	$\cdots$	ü	$\cdots$	x	$\cdots$

M' az M gép egy lépését egy legfeljebb $2T_M(n)$ lépésből álló menetben utánozza.
Jobbra elmegy a legmesszebb levő x jelig, és közben leolvassa az x-ek felett található eredeti szalagjeleket. Ezeket a belső állapotaiban tárolja.
Közben egy hellyel jobbra mozdítja az x-eket, kivéve az utolsó oszlopban levő(ke)t.
Most a gép ismeri M állapotát és a fejei alatti jeleket, így meghatározhatja a M következő állapotát és a kiírandó jeleket.

M' visszamegy a szalag elejére, közben kiírja az M fejeinek régi helyére a k darab jelet, amiket M írt volna,
és az M fejmozgásainak megfelelően áthelyezi az x-eket.

Ha n jelből álló bemenettel kezdjük a munkát, akkor egy menetében az M' feje legfeljebb $T_M(n)$ lépést tesz jobbra $\Longrightarrow$ legfeljebb ugyanennyit mehet balra.
Az M' pontosan akkor álljon meg (fogadja el a bemenetet), ha M ezt teszi.
$\Longrightarrow T_{M'}(n) \leq 2T_M(n)T_M(n) = 2T_M^2(n)$

Ha függvényt számítunk, a végén a felesleget letöröljük.

Ha M-nek kitüntetett input szalagja volt, akkor a bemenet hosszát, ami n, nem számítottuk bele $S_M(n)$-be $\Longrightarrow S_{M'}(n) \leq S_M(n) + n.$

Tétel. *Az M k-szalagos Turing-géphez megadható olyan $\mathbf{2}$-szalagos M' Turing-gép, amely az előbbi értelemben szimulálja M-et,*

$$T_{M'}(n) \leq O(T_M(n) \log T_M(n)), \text{ és}$$

$$S_{M'}(n) \leq S_M(n) + n.$$

Tétel. *Tegyük fel, hogy az M Turing-gép az L nyelvet ismeri fel, és $T_M(n) \le cn$ teljesül egy $c > 0$ állandóval. Ekkor tetszőleges $\epsilon > 0$-ra van olyan L-et felismerő M' Turing-gép is, hogy alkalmas $n_0 \in \mathbb{N}$ számmal*

$$T_{M'}(n) \le n(1+\epsilon), \text{ ha } n \ge n_0.$$

Bizonyítás: Az M' gépet úgy tervezzük, hogy az M gép m lépését az új legfeljebb 7 lépésben elvégzi.
Osszuk fel az M szalagjait m egymás utáni mezőből álló blokkokra $\Longrightarrow$ egy új szalagjel $\Longrightarrow$ az új gép t^m betűt használ.
Az M'-nek eggyel több szalagja lesz, mint M-nek. Az első input szalag, ezt először átkódolja egy másik szalagra.

Mi történik a szalagokkal az M gép m egymást követő lépése során?
Ami ezalatt végbemegy, az csak a fejeket tartalmazó blokkoktól és azok közvetlen szomszédaitól függ.
M': szalagonként három szomszédos jel megvizsgálása után a „memóriájában" meglépi M következő m lépését.

$\Longrightarrow$ felülírja a szomszédos mezőhármasokat, helyükre teszi a fejeket.

$\Longrightarrow$ szimuláció elvégezhető egy bal–jobb–jobb–bal–bal lépéssorozatban esetleg további 2 jobbralépés szükséges lehet $\Longrightarrow$ M' feleinek mozgatása $\Longrightarrow$ $\mathbf{7}$ lépés

A bemenet átkódolása, majd a kódolt szalagon a fejnek a szalag elejére mozgatása $\Longrightarrow \leq n + \lceil \frac{n}{m} \rceil$ lépés

$$T_{M'}(n) \leq n + \left\lceil \frac{n}{m} \right\rceil + 7 \left\lceil \frac{T_M(n)}{m} \right\rceil \leq n + \frac{n}{m} + \frac{7T(n)}{m} + 8 \leq$$

$$\leq n + \frac{n}{m} + \frac{7cn}{m} + \frac{8n}{n} \leq n\left(1 + \frac{1}{m} + \frac{7c}{m} + \frac{8}{n}\right) \leq n(1+\epsilon),$$

ha m és n olyan nagyok, hogy $\frac{1}{m} + \frac{7c}{m} + \frac{8}{n} < \epsilon$. ✓

Komolyabb, „hardverrel" könnyebb.

A Turing-gépmodellben a feladatok időigénye függ a jelkészlet méretétől