Tiende college complexiteit

1 april 2008

Polynoomevaluatie

NP-volledigheid I

Zij $p(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0$ een polynoom van graad $n \geq 1$, met alle a_i reële getallen ($a_i \in \mathbb{R}$).

Probleem:

Gegeven: $a_0, a_1, \ldots, a_{n-1}, a_n$ en x
Gevraagd: $p(x)$

Van links naar rechts de termen $a_i x^i$ berekenen en optellen levert een $O(n^2)$-algoritme.
Als we het polynoom daarentegen van rechts naar links evalueren kunnen we eenvoudig een ordeverbetering bereiken.

Algoritme 1: "gewoon"

```
pol := a_0 + a_1 * x; // n ≥ 1
macht := x;
for i := 2 to n do
    macht := macht * x;
    // berekent x^2, x^3, ...
    pol := pol + a_i * macht;
od
```

Basisoperatie: $*$ en $+, -$

Complexiteit: aantal $* = 2n - 1$

aantal $+, - = n$

Algoritme 2: methode van Horner

```
pol := a_n;
for i := n − 1 downto 0 do
    pol := pol ∗ x + a_i;
od
```

Gebaseerd op:

$$p(x) = ([\ldots([a_n{*}x{+}a_{n-1}]{*}x{+}a_{n-2}){*}x{+}\ldots a_2]{*}x{+}a_1){*}x{+}a_0$$

Complexiteit:

aantal $* = n$
aantal $+, - = n$

Vraag: kan het met minder $*$ en $+, -$?
Antwoord: nee !

Algoritmen gebaseerd op het doen van vergelijkingen konden we beschrijven met **beslissingsbomen**.
Een model om rekenkundige algoritmen (algoritmen die gebaseerd zijn op $+, -, *$ en $/$) mee te beschrijven: schema's.

Een schema

- is een eindige serie stappen van de vorm $s_i := q \circ r$;
- hierin is $\circ$ een rekenkundige operatie: $*, /, +$ of $-$
- q en r zijn constanten (bijvoorbeeld $1, -1, \pi^2, \ldots$) of invoerwaarden (hier a_k's of x) of tussenresultaten van eerdere stappen
- de laatste stap uit het schema berekent het eindresultaat (hier dus $p(x)$)

Stelling. Een schema (dat alleen $+, -$ en $*$ gebruikt) om $a_0 + a_1x + a_2x^2 + \ldots + a_nx^n$ te berekenen moet ten minste n $(+, -)$ stappen doen en n $*$ stappen.
Het bewijs gaat met inductie naar n.

De methode van Horner berekent $p(x)$ met n vermenigvuldigingen en n optellingen $(+/-)$. Er bestaat geen algoritme dat het probleem voor algemene p en x met minder $*$'s en +/-'s kan oplossen. De methode van Horner is derhalve optimaal.

Maar misschien kan het wel beter voor polynomen die een heel speciale vorm hebben.

Polynoomevaluatie met preprocessing: bewerk het polynoom tot een polynoom in een speciale vorm waarop een nieuw evaluatie-algoritme sneller werkt.

Het polynoom

Laat $p(x) = x^n + a_{n-1}x^{n-1} + \ldots + a_1 x + a_0$, met $n = 2^k - 1$. $p(x)$ is dus een monisch polynoom, dat wil zeggen dat $a_n = 1$. We kunnen dit zonder beperking der algemeenheid aannemen.

De speciale vorm

$$p(x) = (x^j + b) * q(x) + r(x),$$

waarin q en r ook weer monisch zijn en in de speciale vorm staan, beide van graad $2^{k-1} - 1$ zijn, en $j = 2^{k-1}$.

Voorbeeld (met $n = 7$)

$$p(x) = x^7 + 6x^6 + 5x^5 + 4x^4 + 3x^3 + 2x^2 + x + 1 =$$

$$(x^4+2)[(x^2+4)(x+6)+(x-20)]+[(x^2-10)(x-10)+(x-107)]$$

Om dit polynoom in x te evalueren gebruikt Horner 7 $*$'s en 7 $+/-$'s, maar het kan met 5 $*$'s en 10 $+/-$'s.

Een gegeven monisch polynoom p van graad $n = 2^k - 1$ is eenvoudig om te zetten naar de speciale vorm.

We willen p schrijven als: $p(x) = (x^j + b) * q(x) + r(x)$ met q en r monisch, beide van graad $2^{k-1} - 1$ en $j = 2^{k-1}$. De waarde van b en de coëfficiënten van q en r zijn hieruit simpel af te lezen.
Immers: als $q(x) = x^{j-1} + q_{j-2}x^{j-2} + \ldots + q_0$ en $r(x) = x^{j-1} + r_{j-2}x^{j-2} + \ldots + r_0$, dan geldt:

$$b + 1 = a_{j-1}, q_l = a_{l+j}, b * q_l + r_l = a_l,$$

voor $l = 0, 1, \ldots, j - 2$ en de a_i de coëfficiënten van p.

Vervolgens kunnen q en r op dezelfde manier in de speciale vorm gebracht worden, etcetera. Algoritme: zie opgave 44.

Als het polynoom p in de juiste vorm staat kan $p(x)$ als volgt geëvalueerd worden:

1. Evalueer $q(x)$ en $r(x)$ recursief

2. Bereken de x^j's: nodig hiervoor zijn $x, x^2, x^4, \ldots, x^{2^{k-1}}$.
   Bereken deze alle van tevoren: $k-1$ $*$'s

3. Vermenigvuldig $(x^j + b)$ met $q(x)$ en tel er $r(x)$ bij op:
   1 $*$ en 2 $+/-$'s

Zij $M(k)$ = het aantal $*$'s dat gedaan wordt om een monisch polynoom (in de speciale vorm) van graad $2^k - 1$ te eva- lueren, zonder de berekening van de x^j mee te tellen.

Dan voldoet $M(k)$ aan de volgende recurrente betrekking:

$$M(k) = \begin{cases} 0 & k = 1 \\ 2M(k-1) + 1 & k > 1 \end{cases}$$

Oplossing: $M(k) = 2^{k-1} - 1 \longrightarrow \frac{n-1}{2}$

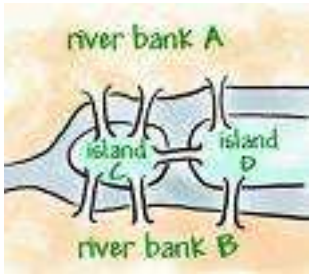
Zij $A(k)$ = het aantal $+/-$'s dat gedaan wordt om een monisch polynoom (in de speciale vorm) van graad $2^k - 1$ te evalueren.

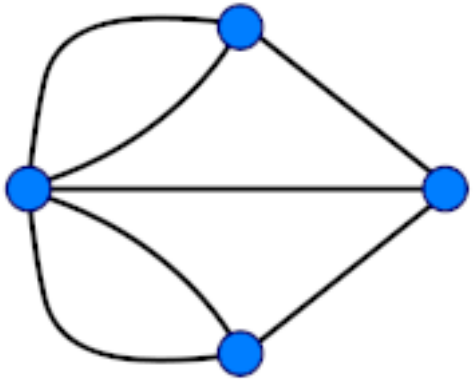
Dan voldoet $A(k)$ aan de volgende recurrente betrekking:

$$A(k) = \begin{cases} 1 & k = 1 \\ 2A(k-1) + 2 & k > 1 \end{cases}$$

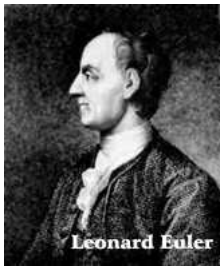
Oplossing: $A(k) = 3 * 2^{k-1} - 2 \longrightarrow \frac{3n-1}{2}$

Koningsberger bruggenprobleem:





Kun je een wandeling door de stad maken waarbij je elke brug precies één keer beloopt en je weer terugkeert in het beginpunt?



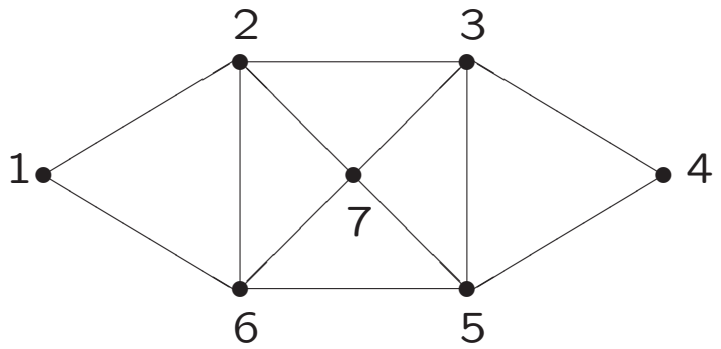
Leonard Euler, 1736

Gegeven een samenhangende, ongerichte graaf $\mathcal{G} = (V, E)$.

Definitie: een kring* in $\mathcal{G}$ die alle takken van $\mathcal{G}$ bevat heet een Eulerkring.

* Een kring bestaat per definitie uit allemaal verschillende takken.

Voorbeeld:



Voor deze graaf is 1 2 3 4 5 3 7 5 6 7 2 6 1 een Eulerkring.

Eulerkringprobleem. Gegeven een samenhangende ongerichte graaf $\mathcal{G} = (V, E)$. Heeft $\mathcal{G}$ een Eulerkring?

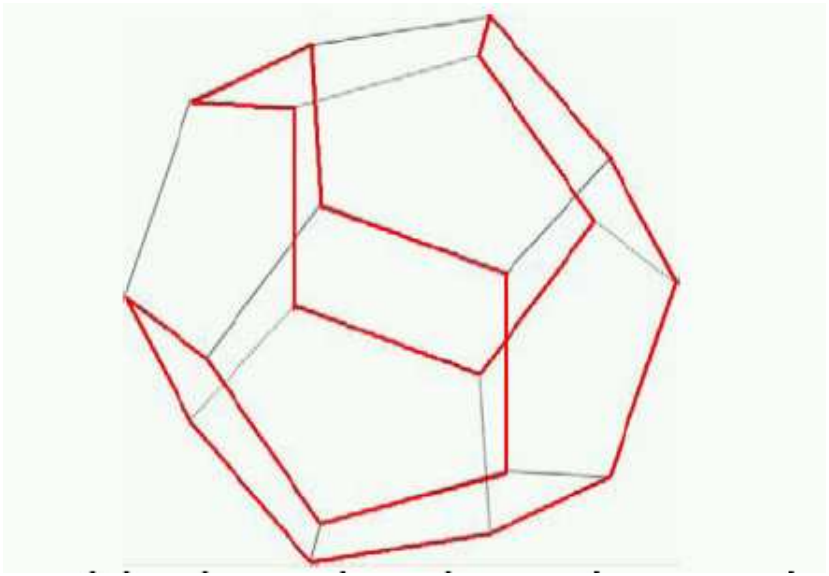
Dit is een voorbeeld van een beslissingsprobleem: het antwoord is ja of nee.

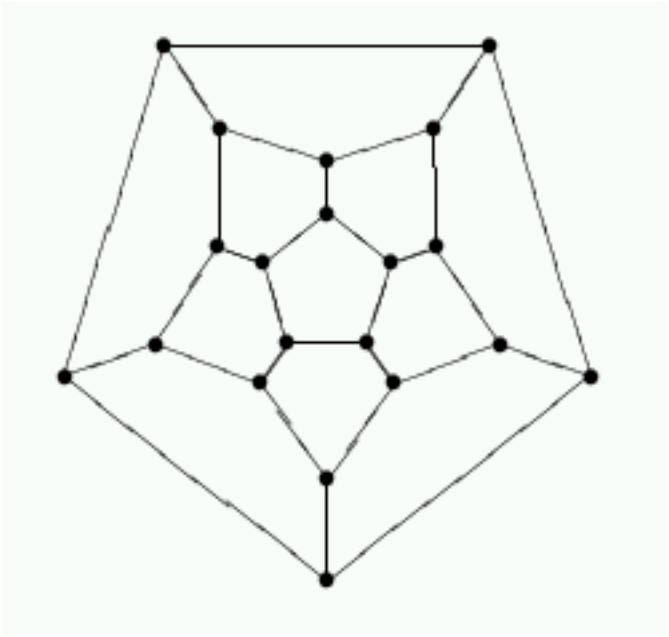
Stelling
Een samenhangende ongerichte graaf heeft een Eulerkring $\Longleftrightarrow$ de graad van iedere knoop is even.

Het is dus heel gemakkelijk (=polynomiaal) na te gaan of een graaf een Eulerkring heeft: $O(|E|)$.

Opmerking
Het bewijs van "$\Longleftarrow$" is constructief: het geeft je meteen een (overigens $O(|E| + |V|)$) algoritme om een Eulerkring te vinden.

Kun je een wandeling door de graaf rechtsonder maken waarbij elke knoop precies één keer bezocht wordt en je weer in het startpunt eindigt?





Sir William Rowan Hamilton, 1859

Gegeven een ongerichte (of gerichte) graaf $\mathcal{G} = (V, E)$.

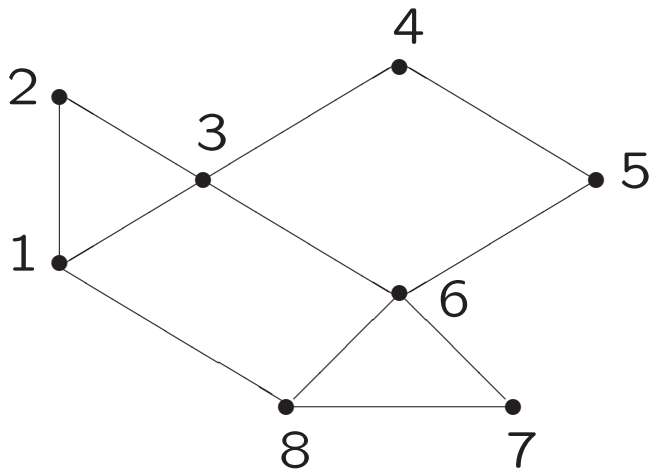
Definitie: een Hamiltonkring in $\mathcal{G}$ is een kring die elke knoop precies één keer bevat.

Bijbehorend beslissingsprobleem: Hamiltonkringprobleem (HC) voor ongerichte grafen. Gegeven een ongerichte graaf $\mathcal{G} = (V, E)$. Heeft $\mathcal{G}$ een Hamiltonkring?

Naamgeving: Als een graaf $\mathcal{G}$ een Hamiltonkring heeft spreken we van een ja-instantie van het probleem. Als zo'n kring niet bestaat is $\mathcal{G}$ een nee-instantie.

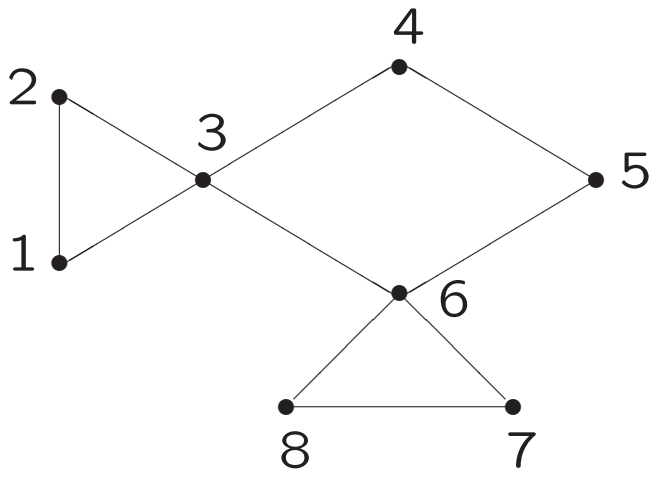
* analoog voor gerichte grafen

Voorbeeld 1:



Deze graaf heeft een Hamiltonkring, namelijk: 1, 2, 3, 4, 5, 6, 7, 8

Voorbeeld 2:



Deze graaf heeft geen Hamiltonkring, maar wel een Hamiltonpad, namelijk: 1, 2, 3, 4, 5, 6, 7, 8

.

1. Een exponentieel algoritme is snel gevonden.
2. Er is geen polynomiaal algoritme voor dit probleem bekend.
3. Er is ook niet bewezen dat een exponentieel algoritme nodig is.
4. Als $\mathcal{G}$ een Hamiltonkring heeft is er een makkelijke (=polynomiaal) manier om dat aan te tonen (certificaat*).
5. De enige manier om te laten zien dat $\mathcal{G}$ geen Hamiltonkring bevat lijkt te zijn: som alle $n!^{\dagger}$ kandidaat Hamiltonkringen op en laat zien dat ze geen Hamiltonkring zijn.

* voor dit probleem: certificaat = Hamiltonkring

† $n = |V|$ = aantal knopen van $\mathcal{G}$

N	10	50	100	300	1000
$\log_2 N$	3	5	6	8	9
$5N$	50	250	500	1500	5000
$N \cdot \log_2 N$	33	282	665	2469	9966
N^2	100	2500	10.000	90.000	7 cijfers
N^3	1000	125.000	7 cijfers	8 cijfers	10 cijfers
2^N	1024	16 cijfers	31 cijfers	91 cijfers	302 cijfers
$N!$	7 cijfers	65 cijfers	161 cijfers	623 cijfers	onvoorstelbaar
N^N	11 cijfers	85 cijfers	201 cijfers	744 cijfers	onvoorstelbaar

Ter vergelijking:

het aantal protonen in het heelal is een getal met 79 cijfers
het aantal microseconden sinds de Oerknal heeft 24 cijfers

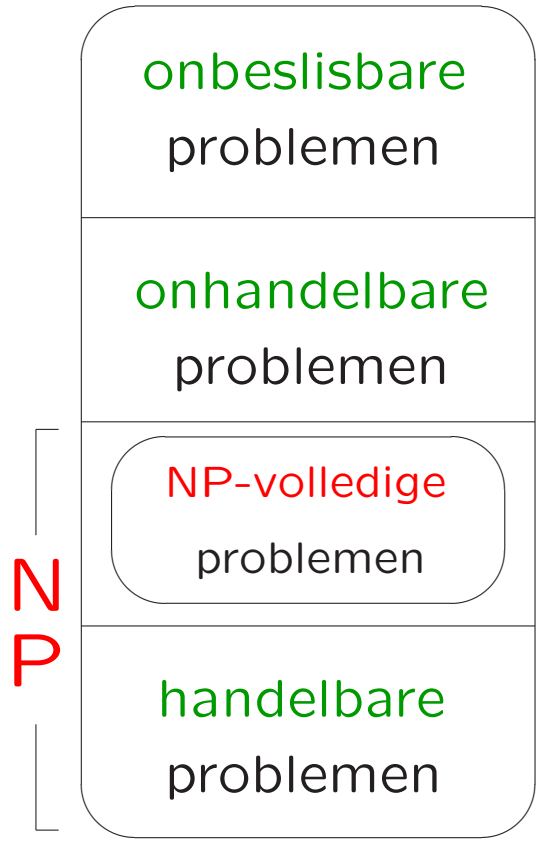
Handelbaar/onhandelbaar -2-

Stel dat de computer 1 instructie doet per microseconde (10^{-6} sec).

N	10	20	50	100	300
N^2	$\frac{1}{10000}$ sec	$\frac{1}{2500}$ sec	$\frac{1}{400}$ sec	$\frac{1}{100}$ sec	$\frac{9}{100}$ sec
N^5	$\frac{1}{10}$ sec	3,2 sec	5,2 min	2,8 uur	28,1 dag
2^N	$\frac{1}{1000}$ sec	1 sec	35,7 jaar	400 biljoen eeuwen	75-cijfers veel eeuwen
N^N	2,8 uur	3,3 biljoen jaar	70-cijfers veel eeuwen	185-cijfers veel eeuwen	728-cijfers veel eeuwen

Ter vergelijking: de oerknal was ongeveer 15 miljard jaar geleden.

Problemen kunnen als volgt in klassen worden ingedeeld:



Problemen waarvoor geen algoritme mogelijk is (stop-probleem)

Problemen waarvoor geen polynomiaal algoritme mogelijk is

????

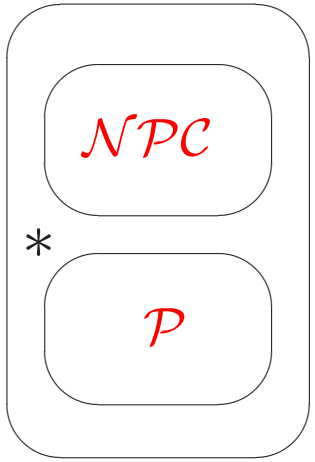
Problemen waarvoor een polynomiaal algoritme bestaat: P

De klasse van NP-volledige problemen $\mathcal{NPC}$ (ook wel: NP-complete problemen), heeft enkele interessante eigenschappen, zoals:

1. Voor geen enkel NP-volledig probleem is tot dusver een polynomiaal algoritme gevonden. Men vermoedt dat ze onhandelbaar zijn, maar dat heeft tot dusver ook nog niemand kunnen bewijzen

2. Als er een polynomiaal algoritme bestaat voor willekeurig welk NP-volledig probleem, dan is meteen elk NP-volledig probleem in polynomiale tijd oplosbaar.
   Omgekeerd: als er van één enkel NP-volledig probleem bewezen wordt dat het onhandelbaar is, dan zijn alle NP-volledige problemen onhandelbaar.

Uit de theorie van NP-volledigheid:



$\mathcal{NP}$

Nondeterministic

Polynomial

$\mathcal{P} = \mathcal{NP}$???

Plaatje

Open vraag

De theorie van NP-volledigheid beperkt zich tot beslissingsproblemen. Bij een beslissingsprobleem zijn slechts twee antwoorden mogelijk: ja of nee.

Probleeminvoer waarop het antwoord *ja* is noemen we ja-instanties, als het antwoord *nee* is spreken we van nee-instanties.

Optimalisatieproblemen worden omgezet naar beslissingsproblemen, en wel zo dat geldt: als het optimalisatieprobleem handelbaar is, dan is het corresponderende beslissingsprobleem dat ook. En dus ook omgekeerd: als het beslissingsprobleem onhandelbaar is, dan is het corresponderende optimalisatieprobleem dat ook.

Handelsreizigersprobleem of Travelling Salesperson Problem

Optimalisatieprobleem

Gegeven een volledige *, ongerichte graaf $\mathcal{G} = (V, E)$ met gewichten op de takken. Geef een Hamiltonkring in $\mathcal{G}$ met minimaal totaalgewicht.

Beslissingsprobleem TSP

Gegeven een volledige *, ongerichte graaf $\mathcal{G} = (V, E)$ met gewichten op de takken, en een geheel getal $k \geq 0$. Bestaat er in $\mathcal{G}$ een Hamiltonkring met totaalgewicht $\leq k$?

* tussen elk tweetal knopen van $\mathcal{G}$ zit een tak.

Voorbeeld.
Een Hamiltonkring in onderstaande graaf is bijvoorbeeld $1, 2, 3, 4$. Deze heeft totaalgewicht 14.
De Hamiltonkring met minimaal gewicht is $2, 4, 3, 1$. Deze heeft totaalgewicht 7.