Complexiteit 2023 — college 10

18 april 2023

Reducties en $\mathcal{NPC}$

Vorige keer

- de klasse $\mathcal{NP}$: beslissingsproblemen waarvoor een niet-deterministisch polynomiaal algoritme bestaat
  - niet-deterministisch: je mag een oplossing gokken
  - polynomiaal: oplossing kan in polynomiale tijd worden geverifieerd (gecontroleerd)
  - ja-instantie: alle invoeren waarvoor *een* executie bestaat die "ja" uitvoert
- niet-deterministisch polynomiaal algoritme:
  1. gokfase: genereer string s
  2. verificatiefase: controleer of s een correcte oplossing is in $O(|s|^p \times |x|^q)$
  3. uitvoerfase: antwoord "ja" als de controle slaagt, anders geen uitvoer
  4. als "ja": $|s| \in O(|x|^k)$ en fase 2 dus in $O(|x|^r)$

Vandaag

- Reducties: definitie en voorbeelden
- De klasse $\mathcal{NPC}$ en NP-volledigheidsbewijzen

TSP en $\mathcal{NP}$

Korte herhaling: $\mathcal{NP}$

Als voorbeeld bekijken we het handelsreizigersprobleem **TSP**. Hiervoor geldt:

$$\text{TSP} \in \mathcal{NP}$$

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 $\leq k$?
De invoer van het probleem is dus $x = \langle \mathcal{G}, k \rangle$.

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

Invoer

Terzijde: het maakt niet uit voor de polynomialiteit van het algoritme welke representatie is gekozen voor de invoer.

- Neem aan dat $V = \{1, 2, \ldots, n\}$. We nemen nu voor $\mathcal{G}$ de adjacency-list-representatie, en voor k de binaire of decimale representatie (bijvoorbeeld). Dan is zoeken van een tak met bijbehorend gewicht $O(|V|) \subseteq O(|x|)$.
- Hetzelfde als hierboven, maar nu nemen we voor $\mathcal{G}$ de adjacency-matrix-representatie. Dan is zoeken van een tak met bijbehorend gewicht $O(1)$.
- (denkend aan een Turingmachine) Representeer $\langle \mathcal{G}, k \rangle$ als een rij van knopen, gevolgd door de takken met gewichten, gevolgd door k, alles binair of decimaal. Dan is zoeken van een tak met bijbehorend gewicht $O(|x|)$.

We gaan hier uit van de eerste optie.

TSP $\in \mathcal{NP}$

Een polynomiaal begrensd niet-deterministisch algoritme A voor TSP:

1. Fase 1 (gokfase)
   Er wordt een string s gegenereerd, hierna te interpreteren als een rij gehele getallen.
2. Fase 2 (verificatiefase)
   Er wordt gecontroleerd of s een Hamiltonkring voorstelt met gewicht $\leq k$:
   (1) controleer of er precies $|V|$ integers staan: $O(|s|)$
   (2) controleer of elke integer tussen 1 en $|V|$ zit: $O(|s|)$
   (3) controleer of alle knopen uit s verschillen: $O(|s|^2)$
   (4) controleer of het totale gewicht van de Hamiltonkring (dat stelt s voor als is voldaan aan (1)–(3)) $\leq k$ is: $O(|s| \times |x|)$ (want . . .)

TSP $\in \mathcal{NP}$

Als de vier tests positief zijn wordt True geretourneerd. Zodra een test negatief uitvalt wordt False teruggegeven (of wordt er een oneindige loop begonnen of ...).

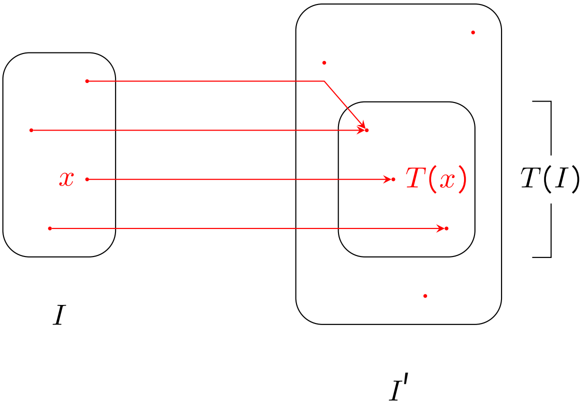
3. Fase 3 (uitvoerfase)
   Als fase 2 True oplevert wordt "ja" uitgevoerd, anders niets.

Voor het hierboven beschreven algoritme geldt:

- Het antwoord van A op invoer $x = \langle \mathcal{G}, k \rangle$ is "ja" $\Leftrightarrow$ er bestaat een string s waarvoor fase 2 True geeft $\Leftrightarrow$ er bestaat een string s die een Hamiltonkring voorstelt met gewicht $\leq k$ $\Leftrightarrow$ $\mathcal{G}$ heeft een Hamiltonkring met gewicht $\leq k$ $\Leftrightarrow$ x is een ja-instantie voor TSP.
- Het algoritme is polynomiaal, want voor een ja-instantie stelt s een (goede) Hamiltonkring voor, dus dan is $|s| \in O(|V|) \subseteq O(|x|)$ en derhalve is fase 2 dan $O(|x|^2)$.

Reductions

Zij T een functie van de invoerverzameling I van een beslissingsprobleem P naar de invoerverzameling I' van een beslissingsprobleem Q.
T beeldt dus elke $x \in I$ af op een $T(x) \in I'$.



Reducties

Definitie

T heet een polynomiale reductie (of *polynomiale transformatie*) van P naar Q als geldt:

1. T kan worden berekend in polynomiaal begrensde tijd (als functie van $|x|$). D.w.z.: de constructie van $T(x)$ uit x kan in $O(|x|^k)$ stappen in de worst case ($k \geq 0$).
2. voor elke x uit I geldt: als x een ja-instantie is voor P dan is $T(x)$ een ja-instantie voor Q.
3. voor elke x uit I geldt: als x een nee-instantie is voor P dan is $T(x)$ een nee-instantie voor Q.

3'. voor elke x uit I geldt: als $T(x)$ een ja-instantie is voor Q dan is x een ja-instantie voor P. (Dit is equivalent met 3.)

Reduceerbaar

Definitie
Een probleem P is polynomiaal reduceerbaar (of polynomiaal transformeerbaar) naar Q als er een polynomiale reductie bestaat van P naar Q.

Notatie: $P \leq_{\mathsf{P}} Q$.

Stelling
Als $P \leq_{\mathsf{P}} Q$ en $Q \in \mathcal{P}$, dan ook $P \in \mathcal{P}$.

Bewijs: zie college 9

Samengevat: reducties

$P \leq_\mathsf{P} Q$ betekent dat er een polynomiale reductie T bestaat van P naar Q:

1. T beeldt elke invoer x van beslissingsprobleem P af op een invoer $T(x)$ van beslissingsprobleem Q.
2. de constructie van $T(x)$ uit x is polynomiaal $(O(|x|^k))$.
3. reductie-eigenschap: voor elke x uit I (de invoerverzameling van P) geldt: x is een ja-instantie voor $P \Leftrightarrow T(x)$ is een ja-instantie voor Q.

HC1 $\leq_P$ HC2

HC1: **gegeven** een *gerichte* graaf $\mathcal{G} = (V, E)$.
Vraag: heeft $\mathcal{G}$ een Hamiltonkring?

HC2: **gegeven** een *ongerichte* graaf $\mathcal{G} = (V, E)$.
Vraag: heeft $\mathcal{G}$ een Hamiltonkring?

Bewering: HC1 $\leq_P$ HC2: zie volgende sheet.

Opmerking: er geldt ook: HC2 $\leq_P$ HC1. Bedenk zelf een eenvoudige reductie.

Een reductie van HC1 naar HC2

Transformatie T die een gerichte graaf $\mathcal{G} = (V, E)$ afbeeldt op een ongerichte graaf $T(\mathcal{G}) = \mathcal{G}' = (V', E')$:

- $V' = \{v_1, v_2, v_3 \mid v \in V\}$: elke knoop $v \in V$ wordt afgebeeld op een drietal knopen v_1, v_2, v_3.
- $E' = \{(v_1, v_2), (v_2, v_3) \mid v \in V\} \cup \{(v_3, w_1) \mid (v, w) \in E\}$: binnen elk drietal knopen corresponderend met v loopt een tak tussen v_1 en v_2 en tussen v_2 en v_3, en voor elke tak (pijl) van v naar w in $\mathcal{G}$ komt een tak in $\mathcal{G}'$ tussen v_3 en w_1.

Dan geldt

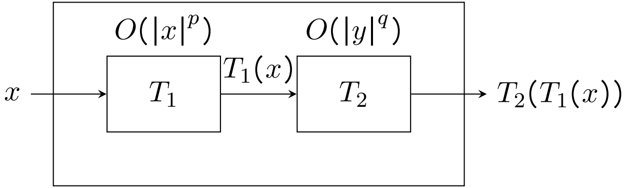
1. T kan in polynomiaal begrensde tijd worden berekend (constructie van $T(\mathcal{G})$ kan zeker in $O(|\mathcal{G}|^q)$, bijv. met $q = 2$)
2. $\mathcal{G}$ is een ja-instantie voor HC1 $\Leftrightarrow$ $T(\mathcal{G})$ is een ja-instantie voor HC2, ofwel: $\mathcal{G}$ heeft een gerichte Hamiltonkring $\Leftrightarrow$ $\mathcal{G}'$ heeft een ongerichte Hamiltonkring

Transitief

Lemma

$\leq_P$ is transitief, d.w.z.: als $P_1 \leq_P P_2$ en $P_2 \leq_P P_3$ dan ook $P_1 \leq_P P_3$.

Het bewijs:



- De samenstelling van T_1 en T_2 is polynomiaal begrensd omdat T_1 en T_2 dat zijn: $O(|x|^{pq})$ (let op: **niet** $O(|x|^{p+q})$!)
- $T_2 \circ T_1(x) = T_2(T_1(x))$ is een ja-instantie van $P_3 \Leftrightarrow T_1(x)$ is een ja-instantie van $P_2 \Leftrightarrow x$ is een ja-instantie van P_1 (volgt uit de reductie-eigenschap van T_2 en T_1)

NP-hard en NP-volledig

Definitie
Een probleem Q is NP-hard (ook wel: NP-moeilijk) als elk probleem P in $\mathcal{NP}$ polynomiaal reduceerbaar is tot Q: dus $P \leq_{\mathsf{P}} Q$ voor alle $P \in \mathcal{NP}$.

Definitie
Een probleem Q is NP-volledig als

1. $Q \in \mathcal{NP}$
2. Q is NP-hard

Notatie De klasse van NP-volledige problemen geven we aan met $\mathcal{NPC}$ (NP-complete).

NP-volledigheid bewijzen

Stelling

Stel Q is een probleem waarvoor geldt dat $P \leq_{\mathsf{P}} Q$ voor een of andere $P \in \mathcal{NPC}$. Dan is Q NP-hard.
Als bovendien $Q \in \mathcal{NP}$, dan geldt dat $Q \in \mathcal{NPC}$.

Bewijs volgt uit de transitiviteit van $\leq_{\mathsf{P}}$ (twee slides terug) en de definitie van NP-hard (vorige slide).

M.a.w.: door een bekend NP-volledig probleem te reduceren tot Q reduceren we impliciet alle problemen uit $\mathcal{NP}$ tot Q. Dit geeft ons derhalve een methode om aan te tonen dat een probleem Q NP-volledig is.

Reductiemethode: $Q \in \mathcal{NPC}$

1. Bewijs dat $Q \in \mathcal{NP}$
2. Kies een bekend NP-volledig probleem P
3. Toon aan dat $P \leq_{\mathsf{P}} Q$

   Stap 3 valt uiteen in:

   3a. Geef een functie T van I (de invoerverzameling van P) naar I' (de invoerverzameling van Q) die elke $x \in I$ afbeeldt op een element $T(x)$ van I'

   3b. Laat zien dat $T(x)$ uit x kan worden geconstrueerd in polynomiaal begrensde tijd ($O(|x|^k)$ voor zekere $k \geq 0$)

   3c. Toon aan dat T voldoet aan: $x \in I$ is een ja-instantie voor P $\Leftrightarrow T(x) \in I'$ is een ja-instantie voor Q

Maar...

Deze reductiemethode is **inductief**:

$$\frac{P \in \mathcal{NPC} \quad P \leq_{\mathsf{P}} Q \quad Q \in \mathcal{NP}}{Q \in \mathcal{NPC}}$$

Wat ontbreekt hier nog?

Maar...

Deze reductiemethode is **inductief**:

$$\frac{P \in \mathcal{NPC} \quad P \leq_{\mathsf{P}} Q \quad Q \in \mathcal{NP}}{Q \in \mathcal{NPC}}$$

Wat ontbreekt hier nog?

Een **basisgeval**:

$$\frac{}{P_0 \in \mathcal{NPC}}$$

Stephen Cook

In 1971 bewees Stephen Cook op een directe manier (dus door een reductie te geven van alle problemen uit $\mathcal{NP}$) dat SAT NP-volledig is. (Details in volgend college.)

Stelling

Gegeven een *arbitrair* probleem $P \in \mathcal{NP}$. Dan is P reduceerbaar tot SAT: $P \leq_{\mathrm{P}} \mathrm{SAT}$.

Sindsdien is met behulp van de reductiemethode van veel bekende problemen aangetoond dat ze NP-volledig zijn. Bijvoorbeeld:

$$\begin{aligned} \mathrm{SAT} \leq_{\mathrm{P}} \mathrm{3SAT} &\leq_{\mathrm{P}} \mathrm{Kliek} \leq_{\mathrm{P}} \mathrm{VC} \\ \mathrm{3SAT} &\leq_{\mathrm{P}} \mathrm{HC2} \leq_{\mathrm{P}} \mathrm{TSP} \\ \mathrm{SAT} \leq_{\mathrm{P}} \mathrm{3Kleur} &\leq_{\mathrm{P}} \mathrm{4Kleur} \end{aligned}$$

3SAT en Kliek

3SAT

Gegeven een logische formule ϕ in 3-CNF. Bestaat er een waardering die ϕ waar maakt?

Definitie Een logische formule ϕ staat in 3-CNF als ϕ een conjunctie is van clausules, waarbij elke clausule een disjunctie is van drie verschillende* literals. variabele: $x_5, x_7, \ldots$; literal: $x_3, \neg x_6, \ldots$; clausule: $x_5 \vee \neg x_6 \vee x_8, \ldots$

Kliek

Gegeven een ongerichte graaf $\mathcal{G} = (V, E)$ en een geheel getal k $(0 \leq k \leq |V|)$. Is er in $\mathcal{G}$ een kliek ter grootte k?

Definitie

Een kliek in een ongerichte graaf $\mathcal{G} = (V, E)$ is een deelverzameling $V' \subseteq V$ zodanig dat voor elk tweetal knopen $u, v \in V'$ $(u \neq v)$ geldt dat $(u, v) \in E$. (Oftewel: V' is volledig.)

*deze eis wordt ook wel weggelaten bij de definitie van 3SAT

3SAT $\leq_P$ Kliek

Er geldt: 3SAT $\leq_P$ Kliek. Om dit aan te tonen moeten we een logische formule in 3-CNF afbeelden op een invoer voor Kliek, dus op een ongerichte graaf en een geheel getal.

Zij ϕ een logische formule in 3-CNF, met zeg m clausules: $\phi = C_1 \wedge C_2 \wedge \ldots \wedge C_m$. Hierin is $C_r = \ell_1^r \vee \ell_2^r \vee \ell_3^r$ $(r = 1, \ldots, m)$ en ℓ_1^r, ℓ_2^r en ℓ_3^r steeds verschillend bij vaste r.

Construeer nu een ongerichte graaf $\mathcal{G}_\phi = (V, E)$ als volgt.

Voor elke clausule C_r uit ϕ doen we drie knopen v_1^r, v_2^r en v_3^r in V (deze corresponderen met ℓ_1^r, ℓ_2^r en ℓ_3^r). $\mathcal{G}_\phi$ heeft dus $3m$ knopen.

3SAT $\leq_P$ Kliek

Er komt een tak tussen twee knopen v_i^r en v_j^s als:

- v_i^r en v_j^s in verschillende drietallen zitten (dus $r \neq s$), en
- de bijbehorende ℓ_i^r en ℓ_j^s zó zijn dat $\ell_i^r \neq \neg \ell_j^s$, met andere woorden: ℓ_i^r en ℓ_j^s zijn niet elkaars negatie.

Definieer nu de transformatie T als: $T(\phi) = \langle \mathcal{G}_\phi, m \rangle$. Dan geldt:

- De constructie van $T(\phi)$ uit ϕ kan in $O(|\phi|^q)$ stappen voor zekere $q \geq 0$ (dus polynomiaal).
- Er is een waardering die ϕ waarmaakt $\Longleftrightarrow$ $\mathcal{G}_\phi$ heeft een kliek ter grootte m.

3SAT $\leq_P$ Kliek: voorbeeld

Laat $\phi = C_1 \wedge C_2 \wedge C_3$, met $C_1 = x_1 \vee \neg x_2 \vee \neg x_3$, $C_2 = \neg x_1 \vee x_2 \vee x_3$ en $C_3 = x_1 \vee x_2 \vee x_3$. Hier is dus $m = 3$.

Dan $v_1^1 = x_1$, $v_2^1 = \neg x_2$ en $v_3^1 = \neg x_3$; alle uit clausule C_1, enz.

- een waardering w die ϕ waarmaakt is bijvoorbeeld: $w(x_1) = w(x_2) = \text{False}$ en $w(x_3) = \text{True}$. Een bijbehorende kliek in $\mathcal{G}_\phi$ ter grootte 3 is dan $\{v_2^1, v_3^2, v_3^3\}$*.
- een kliek ter grootte 3 in $\mathcal{G}_\phi$ is bijvoorbeeld $\{v_1^1, v_2^2, v_2^3\}$. Een bijbehorende waardering is $w(x_1) = w(x_2) = w(x_3) = \text{True}$. Deze maakt ϕ waar.

*De bovenindex komt overeen met de clausule. Uit elk drietal (clausule) één knoop (literal).

SAT en 3SAT

SAT

Gegeven een logische formule ϕ in CNF. Bestaat er een waardering die ϕ waar maakt?

Definitie

Een logische formule ϕ staat in CNF als ϕ een conjunctie is van clausules, waarin elke clausule een disjunctie is van literals.

3SAT

Gegeven een logische formule ϕ in 3-CNF. Bestaat er een waardering die ϕ waar maakt?

Definitie

Een logische formule ϕ staat in 3-CNF als ϕ in CNF staat en elke clausule bestaat uit precies drie (doorgaans verschillende) literals.

SAT $\leq_P$ 3SAT

Er geldt: SAT $\leq_P$ 3SAT. Om dit aan te tonen moeten we een logische formule ϕ in CNF afbeelden op een logische formule ϕ' in 3-CNF. We gaan er voor het gemak van uit dat de $\ell_1, \ell_2, \ldots, \ell_k$ per clausule al verschillend zijn (kan anders worden bewerkstelligd in $O(|\phi|^2)$). Op clausuleniveau werkt de transformatie T als volgt:

$$\ell_1 \mapsto (\ell_1 \vee \widetilde{\ell_2} \vee \widetilde{\ell_3}) \wedge (\ell_1 \vee \widetilde{\ell_2} \vee \neg\widetilde{\ell_3}) \wedge (\ell_1 \vee \neg\widetilde{\ell_2} \vee \widetilde{\ell_3}) \wedge (\ell_1 \vee \neg\widetilde{\ell_2} \vee \neg\widetilde{\ell_3})$$
$$\ell_1 \vee \ell_2 \mapsto (\ell_1 \vee \ell_2 \vee \widetilde{\ell_3}) \wedge (\ell_1 \vee \ell_2 \vee \neg\widetilde{\ell_3})$$
$$\ell_1 \vee \ell_2 \vee \ell_3 \mapsto \ell_1 \vee \ell_2 \vee \ell_3$$
$$\ell_1 \vee \ell_2 \vee \ell_3 \vee \ell_4 \mapsto (\ell_1 \vee \ell_2 \vee \widetilde{\ell_5}) \wedge (\ell_3 \vee \ell_4 \vee \neg\widetilde{\ell_5})$$
$$\ell_1 \vee \ell_2 \vee \ell_3 \vee \ell_4 \vee \ell_5 \mapsto (\ell_1 \vee \ell_2 \vee \widetilde{\ell_6}) \wedge (\ell_3 \vee \neg\widetilde{\ell_6} \vee \widetilde{\ell_7}) \wedge (\ell_4 \vee \ell_5 \vee \neg\widetilde{\ell_7})$$

SAT $\leq_P$ 3SAT

En in het algemeen voor $k \geq 4$:

$$\ell_1 \vee \ell_2 \vee \ldots \vee \ell_{k-1} \vee \ell_k \mapsto (\ell_1 \vee \ell_2 \vee \widetilde{\ell_{k+1}}) \wedge (\ell_3 \vee \neg\widetilde{\ell_{k+1}} \vee \widetilde{\ell_{k+2}}) \wedge (\ell_4 \vee \neg\widetilde{\ell_{k+2}} \vee \widetilde{\ell_{k+3}}) \wedge \ldots \wedge (\ell_{k-2} \vee \neg\widetilde{\ell_{2k-4}} \vee \widetilde{\ell_{2k-3}}) \wedge (\ell_{k-1} \vee \ell_k \vee \neg\widetilde{\ell_{2k-3}})$$

Hierin zijn $\widetilde{\ell_{k+1}}, \widetilde{\ell_{k+2}}, \ldots, \widetilde{\ell_{2k-3}}$ steeds nieuwe, frisse logische variabelen.

Een clausule met k (verschillende) literals wordt zo getransformeerd in een conjunctie van $k-2$ clausules met elk 3 verschillende literals. Het beeld van een conjunctie van clausules definiëren we als een conjunctie van de beelden van de samenstellende clausules:

$$\phi = C_1 \wedge C_2 \wedge \ldots \wedge C_m \mapsto T(C_1) \wedge T(C_2) \wedge \ldots \wedge T(C_m) = T(\phi)$$

SAT $\leq_P$ 3SAT

Voor deze transformatie T geldt:

- de constructie van $T(\phi)$ uit ϕ kan met een polynomiaal begrensd ($O(|\phi|^q)$) algoritme.
- ϕ is een ja-instantie van SAT $\Leftrightarrow$ $T(\phi)$ is een ja-instantie van 3SAT.
- ofwel: er is een waardering die ϕ waarmaakt $\Leftrightarrow$ er is een waardering die $T(\phi)$ waarmaakt.
- conclusie uit de vorige punten: SAT $\leq_P$ 3SAT.

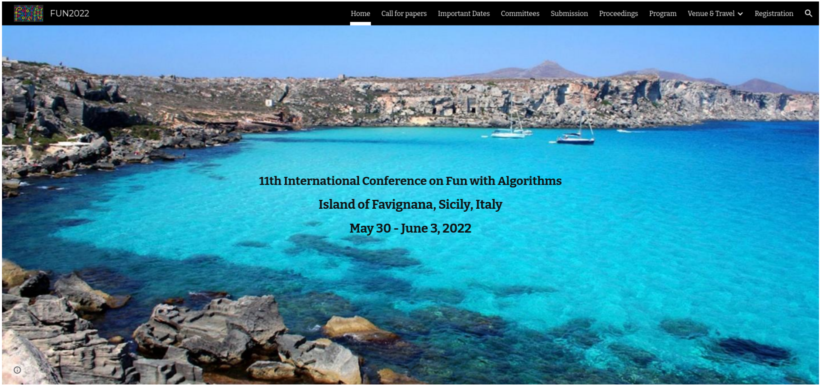
Vragen

We hebben nu: SAT $\leq_P$ 3SAT (*)

Verder is 1SAT: gegeven een logische formule ϕ in 1-CNF. Bestaat er een waardering die ϕ waar maakt? Een logische formule ϕ in 1-CNF heeft de volgende vorm: $\phi = \ell_1 \wedge \ell_2 \wedge \ldots \wedge \ell_n$.

1. Stel dat we weten dat 3SAT $\in \mathcal{NPC}$ en SAT $\in \mathcal{NP}$. Volgt dan uit (*) dat SAT $\in \mathcal{NPC}$?
2. Stel dat we weten dat SAT $\in \mathcal{NPC}$ en 3SAT $\in \mathcal{NP}$. Volgt dan uit (*) dat 3SAT $\in \mathcal{NPC}$?
3. Stel dat we weten dat 3SAT $\in \mathcal{NPC}$. Is 1SAT $\leq_P$ 3SAT?
4. Stel dat we weten dat 3SAT $\in \mathcal{NPC}$. Is 3SAT $\leq_P$ 1SAT?

FUN



FUN 2022: `https://sites.google.com/view/fun2022/home`

FUN 2016: Two-Dots

FUN 2016: https://www2.idsia.ch/cms/fun16/

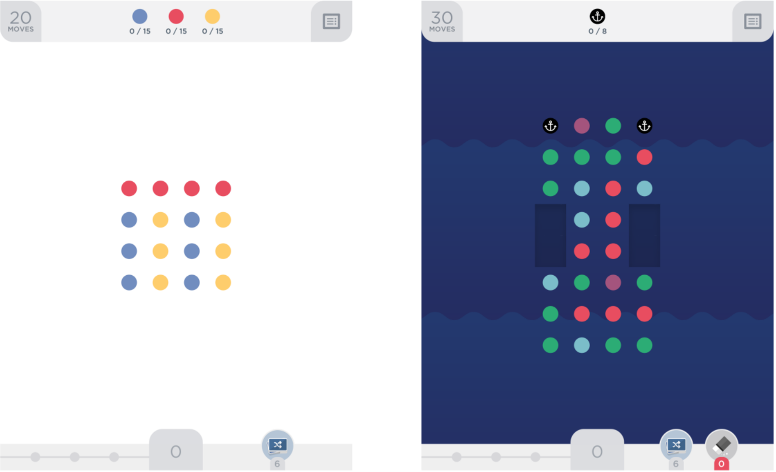
Proceedings: https://drops.dagstuhl.de/opus/portals/lipics/index.php?semnr=16004

Two-Dots is NP-Complete: https://drops.dagstuhl.de/opus/frontdoor.php?source_opus=5883

Two-Dots: https://www.dots.co/twodots/

Online alternatief: https://plays.org/game/two-dots/

Two-Dots



(a) Level One of Two Dots. **(b)** A more advanced level of Two Dots.

Doel: gegeven k zetten, verzamel x

Two-Dots

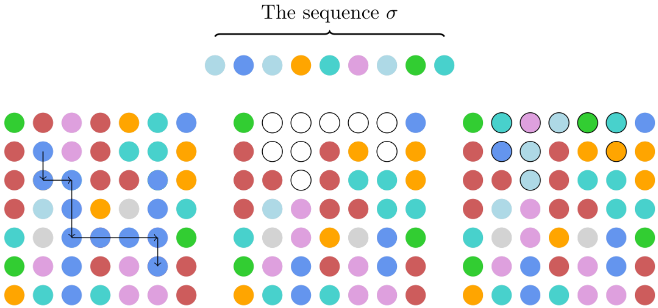


Figure 2 A depiction of the regular move. The first panel shows the set of locations that are committed to the move, the second panel shows the voids created, and the third panel shows how the voids are filled in accordance with σ.

Reguliere zet: orthogonaal pad, verzamelt pad

Two-Dots

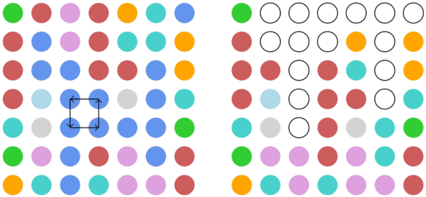


Figure 3 A depiction of the square move, which has the effect of eliminating all the blue dots from the board. The example is rather similar to the above, but note the difference in the number of voids created. The process for filling up the voids is identical, and is therefore omitted.

Speciale zet: 2×2 vierkant, verzamelt hele kleur

Two-Dots en Kliek

Kliek: gegeven een graaf $\mathcal{G} = (V, E)$ en een getal k. Bevat $\mathcal{G}$ een kliek (volledige deelgraaf) van grootte k?

(a) The Vertex Gadgets.

(b) A Tester Gadget for the edges (u, v) and (w, v).

Figure 4 The vertex and tester gadgets used in the reduction from Clique. The colors α and β are depicted, respectively, by blue and red.

Two-Dots: $2k + \binom{k}{2}$ zetten, doel is $2k$ blauw en $4\binom{k}{2}$ rood

Two-Dots en Exact Cover: zonder de vierkante zet

Exact Cover: gegeven een verzameling $U = \{u_1, \ldots, u_n\}$ en een verzameling van verzamelingen $F = \{S_1, \ldots, S_m\}$ (waar $|S_i| = 3$ voor alle i). Bestaat er een deelverzameling $G \subseteq F$ zodat elk element van U precies één keer voorkomt in de elementen van G?

(Dit is NP-volledig.)

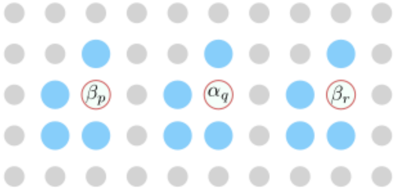
Figure 5 The gadget corresponding to a set $S = \{u_p, u_q, u_r\}$, from an instance of X3C.

Two-Dots: $n + \frac{n}{3}$ zetten, doel is 2 van elke c_i en $\frac{5n}{3}$ ★

Two-Dots en 3-SAT: doel van constante grootte

3-SAT: gegeven een formule ϕ in 3-CNF met clausules $C_1, \ldots, C_m$ en variabelen $x_1, \ldots, x_n$, bestaat er een waardering die ϕ waar maakt?

(a) The variable gadget, where the two colors shown correspond to the two possible assignments.



(b) A Clause gadget corresponding to the clause $(\overline{x_p}, x_q, \overline{x_r})$. The color corresponding to the clause is depicted in Blue.

Figure 6 The variable and clause gadgets used in the reduction from 3-SAT.

Two-Dots: $2n + m + 1$ zetten, doel is $2 \star$
(buiten beeld: twee $\star$ gescheiden door een stapel overeenkomend met de kleuren van de clausules)

Om te onthouden van vandaag

- polynomiale reductie T van P naar Q: $P \leq_{\mathrm{P}} Q$
  - T beeldt elke invoer x van P af op een invoer $T(x)$ van Q
  - constructie van $T(x)$ uit x is polynomiaal (in $|x|$)
  - reductie-eigenschap: x is ja-instantie voor $P \Leftrightarrow T(x)$ is ja-instantie voor Q
  - hoop voorbeelden
- Q is NP-hard (NP-moeilijk) $\Leftrightarrow P \leq_{\mathrm{P}} Q$ voor alle $P \in \mathcal{NP}$
- Q is NP-volledig ($\mathcal{NPC}$) als Q NP-hard is en $Q \in \mathcal{NP}$
- NP-volledigheidsbewijs voor Q: bewijs dat $Q \in \mathcal{NP}$ en dat $P \leq_{\mathrm{P}} Q$ voor $P \in \mathcal{NPC}$
- SAT is het oer-$\mathcal{NPC}$-probleem

(Werk)college

Volgende college: dinsdag 25 april, 9u00–10u45, zaal 407-409 (Snellius)

Werkcollege: zodadelijk van 11u00 tot 12u45, computerzalen 302–304 en 303 (Snellius, onderaan de trap)
Opgaven uit het dictaat: 58, 63, 64, 65

Huiswerk

Huiswerk 3

Verschijnt vanmiddag op de website.

Inleveren via Brightspace, uiterlijk dinsdag 9 mei, 23u59.

Huiswerk 4

Vervalt.

Universitaire verkiezingen

Universiteitsraad: `https://www.organisatiegids.universiteitleiden.nl/medezeggenschap/universiteitsraad`

Faculteitsraad: `https://www.organisatiegids.universiteitleiden.nl/faculteiten-en-instituten/wiskunde-en-natuurwetenschappen/medezeggenschap/faculteitsraad`

Verkiezingen: `https://www.organisatiegids.universiteitleiden.nl/medezeggenschap/universitaire-verkiezingen`

Partijen en standpunten: `https://www.student.universiteitleiden.nl/organisatie/medezeggenschap/universitaire-verkiezingen/partijen/wiskunde-en-natuurwetenschappen/informatica-bsc`

Stemmen kan tot **donderdag 20 april 16u00**