0.1 Gulzige algoritmen en matroids

De algoritmen van Prim en Kruskal zijn gulzig: je sorteert één keer de bogen - de mogelijke keuzes om een MOB op te bouwen - en dan kies je telkens de kleinste om toe te voegen of je verwerpt die voor altijd. Die keuze is onderworpen aan een beperking: in het ene geval is dat *verbindt de huidige MOB met een knoop erbuiten*, in het andere geval *veroorzaakt geen kring indien toegevoegd*. Het bewijs dat dit werkt is ad hoc, in de zin dat het niet past in een meer algemeen kader. Dat kader bieden we hier nu aan, en we laten ook zien hoe Kruskal past in dit kader. Eerst wat definities en voorbeelden die eraan voldoen.

Definitie 0.1.1. *Deelverzamelingsysteem*
Voor een verzameling S en een verzameling D van deelverzamelingen van S (dus $D \subseteq \mathcal{P}(S)$), is (S, D) een deelverzamelingsysteem alss

$$\forall X, Y : (X \in D \;\&\; Y \subset X) \rightarrow Y \in D$$

We zeggen "D is gesloten onder inclusie".
Als D leeg is, dan noemen we het deelverzamelingsysteem triviaal.
Men noemt de elementen van D dikwijls onafhankelijk.

Hier zijn wat voorbeelden van deelverzamelingsystemen: S is steeds een willekeurige verzameling

Alles: $(S, \mathcal{P}(S))$

Zonder: $(S, Zonder(s, S))$ waarbij $s \in S$ en $Zonder(s, S) = \{X \subset S | s \notin X\}$

Onafh: $(V, Onafh(V))$ waarbij V een vectorruimte is en $Onafh(V)$ de verzameling van alle verzamelingen van onafhankelijke vectoren in V

GeenKring: $(E, GeenKring(G))$ waarbij E de boogverzameling is van een graaf $G(V, E)$ en $GeenKring(G)$ de verzameling van deelverzamelingen van E die geen kring bevatten

GeenBoog: $(V, GeenBoog(G))$ waarbij V de knoopverzameling is van een graaf $G(V, E)$ en $GeenBoog(G)$ de verzameling van deelverzamelingen van V waartussen geen boog bestaat in G

Nu nog wat voorbeelden van structuren die geen deelverzamelingsysteem zijn:

Met: $(S, Met(s, S))$ waarbij $s \in S$ en $Met(s, S) = \{X \subset S | s \in X\}$

(merk op: als (S, D) een niet-triviaal deelverzamelingsysteem is, dan $\phi \in D$)

Basis: $(V, Basis(V))$ waarbij V een vectorruimte is en $Basis(V)$ de verzameling van alle basissen van V

Boom: $E, Boom(G))$ waarbij E de boogverzameling is van een graaf $G(V, E)$ en $Boom(G)$ de verzameling van deelverzamelingen van E die een boom zijn

Als voor een deelverzamelingsysteem (S, D) is aan de elementen van S een gewicht wordt gegeven, dan krijgen we in deze context een natuurlijk optimalisatieprobleem

zoek een element O van D met het grootste gewicht[1]

Zulk een O noemen we een *maximum* element van D.

Dat optimalisatieprobleem kan je proberen op te lossen door een generisch gulzig algoritme:

- orden de elementen van S van groot naar klein, tot een rij R die geïndexeerd is van 1 tot $|S|$
- $O := \phi$;
- $\underline{for}\ i = 1..|S|$

  $\underline{if}\ O \cup \{R[i]\} \in D$

  $\underline{then}\ O := O \cup \{R[i]\}$

Dit algoritme geeft zeker een *maximaal* element in D (je kan geen element van S meer toevoegen, of je bekomt iets dat buiten D ligt - pas op, om dat te bewijzen heb je de definitie van deelverzamelingsysteem echt wel nodig !), maar misschien geeft het geen *maximum* element, en dan lost het het optimalisatieprobleem niet op.

Zelf doen

ga na of voor de 5 voorbeelden van deelverzamelingsystemen hiervoor, bovenstaand algoritme ook een maximum element berekent - je moet zelf de gewichten definiëren

[1] het gewicht van een verzameling is de som van de gewichten van zijn elementen

Eén bijkomende eigenschap van een deelverzamelingsysteem garandeert dat het gulzige algoritme hierboven, ook een maximum element berekent, gelijk wat de gewichten zijn.

Definitie 0.1.2. *Matroid*
Een matroid is een deelverzamelingsysteem (S, D) met de bijkomende eigenschap dat $\forall X, Y : (X \in D, Y \in D, |X| < |Y|) \rightarrow \exists y \in Y \setminus X : X \cup \{y\} \in D$

In woorden: als twee elementen van D van verschillende grootte zijn, dan zit er in de grootste een element dat je kan toevoegen aan de kleinste en dan bekom je nog altijd een element van D. Pas op, *grootte* betekent *aantal elementen*, niet *gewicht*: gewichten spelen (nog) geen rol.

Welke van de deelverzamelingsystemen hierboven is een matroid ?

Lemma 0.1.3. *In matroid (S, D) zijn alle maximale elementen (van D) even groot*

Proof Stel dat A en B maximale elementen zijn van (S, D), en dat $|A| < |B|$, dan $\exists b \in B$ zodat $A \cup \{b\} \in D$ wat de maximaliteit van A tegenspreekt. Dus $|A| = |B|$. ■

Er is ook een soort omgekeerde van vorig lemma: voor een (S, D) geldt

als $\forall A \subset D$ (S, D) alle maximale subsets van A even groot zijn, dan is (S, D) een matroid.

Het lemma bewees het omgekeerde enkel voor $A = D$.

Stelling 0.1.4. *Voor een deelverzamelingsysteem (S, D) geldt: het generisch gulzig algoritme lost het optimalisatieprobleem op voor (S, D) alss (S, D) een matroid is.*

Proof TODO ■

Deze stelling verbindt een eigenschap van deelverzamelingsystemen met een eigenschap van een algoritme: de eigenschap van deelverzamelingsystemen refereert niet naar het algoritme, en zelfs niet naar de gewichten, die toch essentieel zijn voor het algoritme en het optimalisatieprobleem. Het algoritme maakt dan weer geen expliciet gebruik van de specifieke eigenschap van de matroid. Sterk, nietwaar !

todo

- kruskal
- andere ?