

RADBOUD UNIVERSITEIT NIJMEGEN



FACULTEIT DER NATUURWETENSCHAPPEN, WISKUNDE EN INFORMATICA

WISKUNDIG VERSLAG
MODELLENPRACTICUM 2014

Namen:	Baukje Debets Elena Fuentes Bongenaar Giselle Loeffen Jeroen Pijnenburg Daan van Rozendaal André Schemaitat
Studie:	Bachelor Wiskunde
Begeleider:	Dr. W. Bosma
Opdrachtgever:	Drs. G. Haan Topicus Zorg

1 juli 2014

Inhoudsopgave

1	Voorwoord	3
2	Definities	4
3	Probleembeschrijving	7
4	Minimum Cost Flow	10
5	Min Cost Max Flow	16
6	Kortste route	22
7	Oplossing	23

1 Voorwoord

In dit verslag zullen we een probleemstelling oplossen, die het bedrijf topicus heeft aangedragen. Dat bedrijf maakt webapplicaties. Het team Plus, onderdeel van Topicus Zorg, waarvan wij deze opdracht hebben gekregen werkt aan het elektronisch patiëntendossier. De testers uit het team simuleren de gebruikers en bekijken de eventuele fouten die in het programma kunnen optreden.

Voor elke applicatie kan een schema worden gemaakt waarin duidelijk wordt op welke manieren door de applicatie heen kan worden gelopen. Wiskundig kan je een schema als een gerichte graaf zien. Aan de hand van zo'n graaf willen de testers zekere paden doorlopen om de applicatie zo efficiënt mogelijk te testen. Onze opdracht is om een tooltje te maken dat zulke testpaden genereert, op basis van bepaalde eisen.

In dit verslag zullen we de opdracht wiskundig beschrijven en algoritmes geven, die we in onze tool gebruiken. Bovendien zullen we bewijzen dat deze algoritmes werken.

2 Definities

2.1. Het probleem, zoals in het voorwoord aangeduid, kan wiskundig door middel van grafentheorie worden beschreven. Voordat we het probleem wiskundig kunnen formuleren, hebben we eerst een aantal definities nodig.

2.2. In dit verslag nemen we aan dat $0 \in \mathbb{N}$. Als $f : A \rightarrow B$ een functie is en $A' \subset A$, dan noteren wij met $f \upharpoonright A'$ de beperking van f tot A' .

2.3 Definitie. Een **gerichte graaf** $G = (V, E)$ bestaat uit een eindige verzameling punten V en kanten $E \subset V \times V$. Bij deze graaf maken we functies

$$\begin{aligned} \text{in} : V &\rightarrow \mathbb{N} : v \mapsto \sum_{(w,v) \in E} 1, \\ \text{out} : V &\rightarrow \mathbb{N} : v \mapsto \sum_{(v,w) \in E} 1. \end{aligned}$$

Deze functies worden de **in-** resp. **uit-**graad genoemd.

Een punt $s \in V$ heet **bron** als $\text{in}(s) = 0$. Een punt $t \in V$ heet **put** als $\text{out}(t) = 0$.

Als $e = (v, w) \in E$, dan noteren wij met $e^{-1} := (w, v)$ de omgekeerde kant. Deze hoeft natuurlijk niet noodzakelijk in E bevat te zijn. Zie 2.13 voor meer details.

2.4 Definitie. Gegeven een gerichte graaf $G = (V, E)$ maken we hierbij de zogenaamde **lijngraaf** $L(G) = (E, E')$. De punten van $L(G)$ zijn dus de kanten van G . De kanten van $L(G)$ maken wij als volgt: Voor kanten $e_1 = (v_1, w_1)$ en $e_2 = (v_2, w_2) \in E$ definiëren wij

$$(e_1, e_2) \in E' \iff w_1 = v_2.$$

Dat betekent dus dat een kant $(e_1, e_2) \in E'$ precies dan als het eindpunt van e_1 gelijk is aan het beginpunt van e_2 .

2.5 Definitie. Zij $G = (V, E)$ een gerichte graaf. Een $v_0 - v_r$ **pad** is een $r + 1$ -tupel

$$P = (v_0, v_1, v_2, \dots, v_r),$$

zodat $(v_i, v_{i+1}) \in E$ voor $0 \leq i < r$. In het bijzonder is een $v_0 - v_r$ pad P een **cykel** als $v_0 = v_r$. Per conventie zullen we alleen over paden en cyclen praten als deze minstens een kant bevatten. Verder noteren wij de **lengte** van het pad met $\ell(P) := r$. De lengte $\ell(P)$ geeft dus het aantal kanten in P aan.

2.6 Definitie. Als $G = (V, E)$ een gerichte graaf is, dan noemen we een collectie $\mathcal{F}$ van paden een **kantenoverdekking** als iedere kant $e \in E$ in minstens een pad $P \in \mathcal{F}$ voorkomt.

2.7 Definitie. Een **netwerk** bestaat uit een gerichte graaf

$$G = (V \cup \{s, t\}, E).$$

De punten s en t zijn de **unieke** bron resp. put. Verder nemen we aan dat $s, t \notin V$. De punten in V heten **intermediaire punten**. Op de graaf maken we functies

$$\begin{aligned} d, l : E &\rightarrow \mathbb{N}, \\ u : E &\rightarrow \mathbb{N} \cup \{\infty\} \end{aligned}$$

Met d bedoelen we de kosten van de kanten. De **capaciteiten** van de kanten hebben ondergrens $l(e)$ en bovengrens $u(e)$. Een netwerk noteren wij als

$$N = (G, d, l, u).$$

Verder eisen we voor ons netwerk dat G **samenhangend** is, oftewel: Voor ieder intermediair punt $v \in V$ bestaat een $s - v$ pad en een $v - t$ pad.

2.8 Definitie. Een **pre-stroom** f op een graaf $G = (V, E)$ is een functie $f : E \rightarrow \mathbb{N}$.

2.9 Definitie. Zij $N = (G, d, l, u)$ een netwerk. Laat $f_0 := l$. Dan noemen we f_0 de **basis pre-stroom** op G . Als we in de tekst f_0 gebruiken, wordt altijd de basis pre-stroom op de onderliggende graaf bedoeld.

2.10 Definitie. Zij $N = (G, d, l, u)$ een netwerk. Voor een pre-stroom f op G en $v \in V \cup \{s, t\}$ definiëren wij

$$\beta(f, v) := \sum_{(v, w) \in E} f(v, w) - \sum_{(w, v) \in E} f(w, v).$$

Het is duidelijk dat β een $\mathbb{R}$ -lineaire functie in de eerste component is.

Merk op: We sommeren altijd over de kantenverzameling E , waarop f gedefinieerd is.

2.11 Definitie. Stel $N = (G, d, l, u)$ is een netwerk. Dan kunnen we een partitie $\{S, B, D\}$ van V maken door

$$\begin{aligned} v \in S &\iff \beta(f_0, v) < 0, \\ v \in B &\iff \beta(f_0, v) = 0, \\ v \in D &\iff \beta(f_0, v) > 0. \end{aligned}$$

2.12 Definitie. Zij $N = (G, d, l, u)$ een netwerk en f een pre-stroom op G . Dan noemen wij f een **stroom**, als

$$\begin{cases} \forall e \in E : l(e) \leq f(e) \leq u(e), \\ \forall v \in V : \beta(f, v) = 0. \end{cases}$$

De **kosten** van een stroom definiëren wij door

$$\text{cost}(f) := \sum_{e \in E} d(e) \cdot f(e).$$

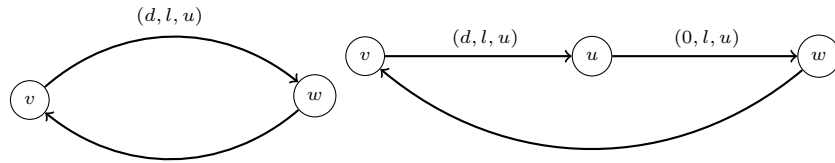
De **waarde** van een stroom geven we aan door

$$\text{val}(f) := \sum_{(s, v) \in E} f(s, v) = \beta(f, s).$$

2.13. We maken de volgende **aannamen** voor een netwerk: Gegeven $N = (G, d, l, u)$ en punten $u, v \in V$ is het **niet** zo, dat $(u, v) \in E$ en $(v, u) \in E$.

Een kant tussen twee punten kan dus of niet bestaan of alleen in een van de twee richtingen. Dit is nodig om een wel gedefiniëerde kostenfunctie in definitie 2.14 te verkrijgen.

Deze aanname kunnen we z.b.d.a. maken: Stel (v, w) en (w, v) zijn kanten in G . Dan voegen we een punt u aan V toe en vervangen de kant (v, w) door (v, u) en (u, w) , zoals hieronder te zien. We laten dan $d(v, u) = d(v, w)$, $d(u, w) = 0$, $l(v, u) = l(v, w) = l(u, w)$ en $u(v, u) = u(v, w) = u(u, w)$.



Figuur 1: Kanten in twee richtingen.

2.14 Definitie. Zij $N = (G, d, l, u)$ een netwerk. Stel f is een stroom op N . Dan kunnen we de **residuële graaf** $G_f = (V \cup \{s, t\}, E_f)$ maken en een **residueel netwerk** $N_f = (G_f, \Delta, l_f, u_f)$. We maken G_f als volgt:

$$(v, w) \in E_f \iff \begin{cases} (v, w) \in E \text{ en } u(v, w) - f(v, w) > 0 \\ (w, v) \in E \text{ en } f(w, v) > 0. \end{cases} \quad \text{of}$$

Voor het eerste geval leggen we vast dat $u_f(v, w) := u(v, w) - f(v, w)$. In het tweede geval laten we $u_f(v, w) := f(w, v)$. Verder laat $l_f = 0$ op E_f . Voor de kosten Δ van kanten in G_f definiëren we

$$\Delta(v, w) := \begin{cases} d(v, w) & \text{als } (v, w) \in E, \\ -d(w, v) & \text{als } (w, v) \in E. \end{cases}$$

Merk op dat Δ van d afhangt. In de tekst is het altijd duidelijk wat de gebruikte kostenfunctie d is, om Δ te berekenen.

2.15 Definitie. Zij N een netwerk en f een stroom op N . Stel dat

$$\sum_{(s, v) \in E} u(s, v) < \infty.$$

Als f een stroom op G is, dan noemen we f **maximaal**, als de waarde $\text{val}(f)$ maximaal is. Voor iedere stroom f' op N geldt dus dat $\text{val}(f') \leq \text{val}(f)$. We merken ook op dat we alleen over maximale stromen spreken, als het netwerk

inderdaad aan de voorwaarde voldoet, dat de som der bovengrensen in s eindig is. Dit is nodig om de existentie van zulke stromen op N te garanderen, waarop we hier niet verder hoeven in te gaan.

2.16 Definitie. Stel f is een stroom op een netwerk N . Laat $\mathcal{F}$ de collectie van alle stromen op N zijn. Dan heet f **extreem** als f

$$\text{cost}(f) = \min\{\text{cost}(f') : f' \in \mathcal{F} \text{ en } \text{val}(f') = \text{val}(f)\}.$$

Dus f heet extreem als hij onder alle stromen met dezelfde waarde minimale kosten heeft.

2.17 Definitie. Zij N een netwerk. Een **Min Cost Max Flow** is een maximale, extreme stroom.

2.18 Definitie. Zij N een netwerk. Een **Min Cost Flow** is een stroom die onder alle stromen op N minimale kosten heeft. Voor iedere stroom f' op N geldt dus $\text{cost}(f) \leq \text{cost}(f')$.

3 Probleembeschrijving

3.1. Nu we alle nodige definities kennen, zullen we ons probleem wiskundig correct beschrijven.

We beginnen met een gerichte en samenhangende graaf $G = (V, E)$, waarbij een graaf **samenhangend** heet als tussen ieder tweetal punten $v, w \in V$ een niet noodzakelijk gericht pad bestaat. Oftewel er zijn kanten $e_1, e_2, \dots, e_n \in E$ zodat

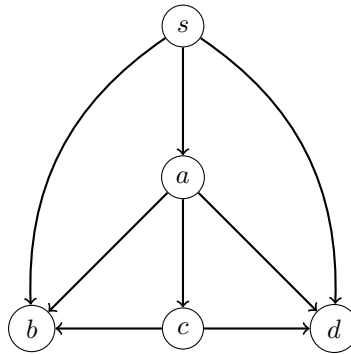
$$(e_1^{\gamma_1}, \dots, e_n^{\gamma_n})$$

een $v - w$ pad is, voor zekere $\gamma_i \in \{-1, 1\}$. We nemen aan dat G één unieke bron s heeft. Verder nemen we aan dat er minstens een put is.

Het probleem is nu een kantenoverdekking, bestaande uit $s - t$ paden, te vinden zodat iedere combinatie van opeenvolgende kanten in G in minstens een pad uit deze collectie voorkomt. We noemen zo een collectie *bruikbaar*, als deze de volgende eigenschappen heeft:

$$(u, v), (v, w) \in E \Rightarrow \begin{cases} \text{Er is een } P = (v_0, \dots, v_r) \in \mathcal{C} \text{ en} \\ \text{een } 0 \leq i \leq r - 2 \text{ met } (u, v, w) = (v_i, v_{i+1}, v_{i+2}), \\ \text{waar } v_0 = s, v_r = t \text{ met } \text{out}(t) = 0. \end{cases}$$

We gaan ons probleem aan de hand van een voorbeeld uitleggen. Bekijk de onderstaande graaf:



Een bruikbare collectie $\mathcal{C}$ is bijvoorbeeld gegeven door de paden

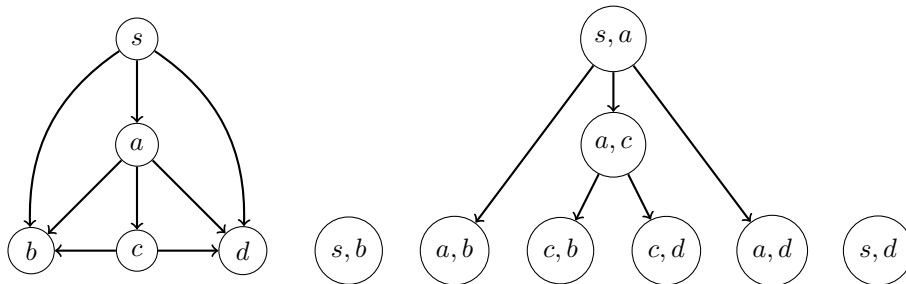
$$\begin{aligned} P_1 &= (s, b), \\ P_2 &= (s, d), \\ P_3 &= (s, a, b), \\ P_4 &= (s, a, d), \\ P_5 &= (s, a, c, b), \\ P_6 &= (s, a, c, d). \end{aligned}$$

Het doel is nu een collectie $\mathcal{C}_0$ te vinden met

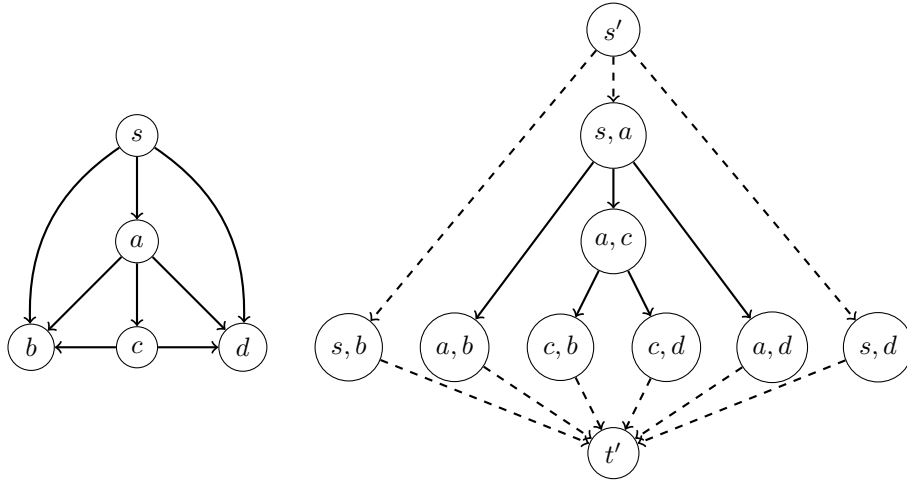
$$\sum_{P \in \mathcal{C}_0} \ell(P) = \min \left\{ \sum_{P \in \mathcal{C}} \ell(P) : \mathcal{C} \text{ is bruikbaar} \right\}.$$

We zoeken dus een bruikbare collectie $\mathcal{C}_0$ zodat het totale aantal kanten minimaal is. We vertalen dit probleem naar de lijngraaf $L(G)$.

Stel we hebben een combinatie van twee opeenvolgende kanten e_1, e_2 . Per definitie van $L(G)$ is dan (e_1, e_2) een kant in $L(G)$. Het onderstaande voorbeeld laat zien hoe je uit de graaf aan de linkerkant zijn lijngraaf kunt maken:



We zien dus dat iedere combinatie van opeenvolgende kanten in G door één kant in $L(G)$ gerepresenteerd wordt. Losse punten corresponderen met kanten van de bron naar een put. Omdat we later met netwerken willen werken, zorgen we ervoor dat we een unieke bron s' en put t' krijgen. Voeg dus s' en t' aan V toe. Verder voegen we de gestippelde lijnen als kanten aan E toe:



Deze aangepaste lijngraaf noteren we door $G' = (V \cup \{s', t'\}, E')$. Als we een kantenoverdekking, waarvan het totale aantal punten minimaal is, van G' door $s' - t'$ paden hebben gevonden zijn we klaar. Uit deze kantenoverdekking kunnen we dan de gezochte bruikbare collectie $\mathcal{C}_0$ maken.

We zullen in stelling 7.2 zien dat zich het vinden van zo een kantenoverdekking tot het vinden van een stroom (op G') met minimale kosten reduceert, waarbij we eisen dat de stroom op iedere kant minstens 1 is. De procedure die we willen volgen laat zich dus als volgt beschrijven:

- 3.2.** 1. Maak $L(G)$. Voeg een unieke bron s' , put t' en bijbehorende kanten, zoals net beschreven, toe. Noteer de resulterende graaf met $G' = (V' \cup \{s', t'\}, E')$.
2. Op G' maken we kosten d . Iedere kant krijgt kosten 1, behalve de kanten met als eindpunt t' . Deze krijgen kosten 0. Verder willen we iedere kant minstens een keer doorlopen, zodat $l(e) := 1$ voor iedere kant. We mogen alle kanten echter wel zo vaak doorlopen als we maar willen, zodat $u(e) := +\infty$ voor iedere kant.
3. Vind een kantenoverdekking $\mathcal{F}_0$ van G' door $s' - t'$ paden zodat

$$\sum_{P \in \mathcal{F}_0} \text{cost}(P)$$

minimaal is, waarbij $\text{cost}(P) = \sum_{i=0}^{r-1} d(v_i, v_{i+1})$ als $P = (v_0, \dots, v_r)$.

4. Herstel uit $\mathcal{F}_0$ een bruikbare collectie $\mathcal{C}_0$ voor G .

3.3. Het wiskundig probleem ligt in punt 3. De vraag is dus hoe we zo een kantenoverdekking $\mathcal{F}_0$ kunnen vinden, zodat de totale kosten van alle paden in $\mathcal{F}_0$ minimaal zijn. In het volgende geven we eerst een aantal algoritmes, die ons uiteindelijk in staat stellen deze collectie te vinden.

4 Minimum Cost Flow

4.1. In het volgende bekijken we een netwerk $N = (G, d, l, u)$. Het doel is om een stroom op N te vinden die minimale kosten heeft. Hiervoor gebruiken een efficiënte methode die het idee van balanceren gebruikt, zie [1].

4.2 Definitie. Zij $N = (G, d, l, u)$ een netwerk. We breiden $G = (V \cup \{s, t\}, E)$ uit. We maken we een nieuw netwerk N^* , door een kant (t, s) aan E toe te voegen, met $l(e) = 0 = d(e)$ en $u(e) = +\infty$. De nieuwe graaf noteren we met G^* .

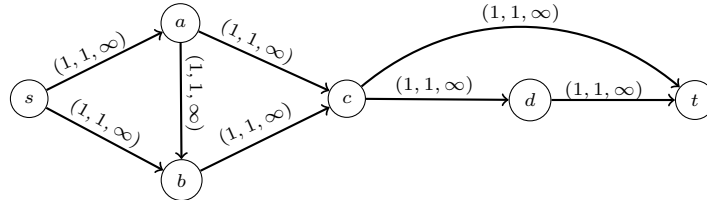
4.3 Definitie. Zij $N = (G, d, l, u)$ een netwerk. Maak N^* zoals in definitie 4.2. Voeg nu punten s', t' aan $V \cup \{s, t\}$ toe. We voegen verder de volgende kanten toe:

$$\begin{aligned} (s', v) &\text{ voor } v \in S \cup \{t\} \text{ met } u(s', v) := -\beta(f_0, v), \\ (v, t') &\text{ voor } v \in D \cup \{s\} \text{ met } u(v, t') := \beta(f_0, v). \end{aligned}$$

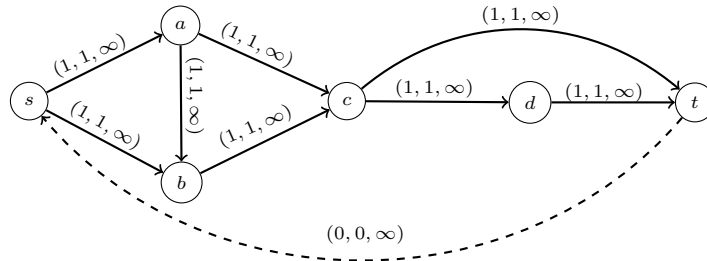
Merk op dat f_0 de basis pre-stroom op G is. Voor iedere toegevoegde kant e laten we $d(e) := 0$. De kosten van kanten in G^* veranderen wij niet. De ondergrens $l(e)$ zetten we gelijk aan 0, voor alle kanten. De resulterende graaf noteren wij met $\overline{G}^*$, en we noemen hem de **balancerende graaf**.

4.4 Voorbeeld. We bekijken even hoe definitie 4.3 werkt.

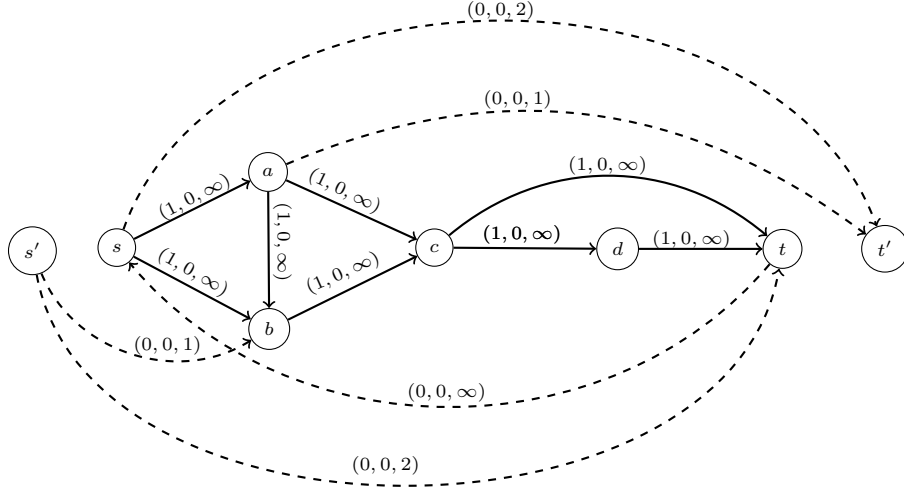
Een tripel (a, b, c) boven een kant e in een gerichte graaf noteert $(d(e), l(e), u(e))$. We beginnen met deze graaf. Voor iedere kant geldt $l(e) = 1 = d(e)$ en $u(e) = \infty$.



Maak nu G^* . Voeg dus een kant (t, s) toe met $l(t, s) = d(t, s) = 0$ en $u(t, s) = \infty$.



Maak de balancerende graaf $\overline{G}^*$:



4.5 Definitie. Zij $N = (G, d, l, u)$ een netwerk, en f_b een pre-stroom op N . Dan noemen we f_b **balancerend**, oftewel een **balancerende pre-stroom**, als

$$\forall v \in V : \beta(f_b, v) = -\beta(f_0, v),$$

waarbij f_0 de basis pre-stroom op G is. Verder moet f_b voldoen aan

$$\forall e \in E : f_b(e) \leq u(e) - l(e).$$

4.6 Lemma. Zij N een netwerk. Dan is f een stroom op G dan en slechts dan als $f = f_0 + f_b$, waar f_0 de basis pre-stroom is en f_b een balancerende pre-stroom.

Bewijs. Stel f_b is een pre-stroom en laat v een intermediair punt zijn. Met $f = f_0 + f_b$ geldt

$$\beta(f, v) = \beta(f_0, v) + \beta(f_b, v) = \beta(f_0, v) - \beta(f_0, v) = 0.$$

Met $0 \leq f_b(e) \leq u(e) - l(e) = u(e) - f_0(e)$ geldt dus $l(e) \leq f(e) \leq u(e)$. Dus is f een stroom.

Andersom laat f een stroom zijn en definieer $f_b(e) := f(e) - f_0(e)$. Dan geldt

$$\beta(f_b, v) = \beta(f, v) - \beta(f_0, v) = -\beta(f_0, v).$$

Bovendien geldt $0 \leq f_b(e) \leq u(e) - l(e)$. Dus is f_b balancerend. ■

4.7 Gevolg. Zij $N = (G, d, l, u)$ een netwerk. Als f_b een balancerende pre-stroom op G is, met minimale kosten, dan is $f = f_0 + f_b$ een Min Cost Flow op G .

4.8 Lemma. *Zij N een netwerk. Dan bestaat er een balancerende pre-stroom f_b^* op G^* dan en slechts dan als er een balancerende pre-stroom f_b op G bestaat.*

In dat geval geldt $\text{cost}(f_b) = \text{cost}(f_b^)$.*

Bewijs. Stel f_b^* is een balancerende pre-stroom op G^* . Laat f_0^* de basis pre-stroom op G^* zijn. Merk op dat $l(t, s) = 0 = f^*(t, s)$. Dus

$$\beta(f_0, t) = \beta(f_0^*, t) \quad \text{en} \quad \beta(f_0, s) = \beta(f_0^*, s). \quad (4.1)$$

Laat $f_b := f_b^* \upharpoonright G$. Met formule (4.1) zien we dat f_b een balancerende pre-stroom op G is.

Stel nu dat er een balancerende pre-stroom f_b op G bestaat. We maken nu f_b^* op G^* als volgt. Voor $e \in E$ laten we $f_b^*(e) := f_b(e)$ en verder

$$f_b^*(t, s) := \beta(f_0, s) + \beta(f_b, s).$$

Merk op dat f_0 de basis pre-stroom op G is. Er geldt $\beta(f_b, v) = \beta(f_b^*, v)$ voor alle intermediaire punten v . In 4.6 hebben we laten zien dat $f := f_0 + f_b$ een stroom op G is. Hieruit volgt $\beta(f, s) = -\beta(f, t)$. Oftewel:

$$\beta(f_0, s) + \beta(f_b, s) = -(\beta(f_0, t) + \beta(f_b, t)).$$

Dat geeft

$$\begin{aligned} \beta(f_b^*, t) &= \beta(f_b, t) + f_b^*(t, s) \\ &= \beta(f_b, t) + \beta(f_0, s) + \beta(f_b, s) \\ &= -\beta(f_0, t). \end{aligned}$$

Vanwege formule (4.1) geldt dus dat $\beta(f_b^*, t) = -\beta(f_0^*, t)$. Net zo geldt

$$\begin{aligned} \beta(f_b^*, s) &= \beta(f_b, s) - f_b^*(t, s) \\ &= \beta(f_b, s) - \beta(f_0, s) - \beta(f_b, s) \\ &= -\beta(f_0, s) = -\beta(f_0^*, s). \end{aligned}$$

Dus is f_b^* een balancerende pre-stroom op G^* .

Omdat $d(t, s) = 0$ zien we dus dat in elk geval geldt $\text{cost}(f_b) = \text{cost}(f_b^*)$. ■

4.9 Gevolg. *Zij N een netwerk. Dan is er een balancerende pre-stroom met minimale kosten op G dan en slechts dan als er een balancerende pre-stroom met minimale kosten op G^* is.*

4.10 Definitie. *Zij N een netwerk en F een pre-stroom op de balancerende graaf $\overline{G}^*$. Definieer F heeft eigenschap (α) als:*

$$\begin{aligned} v \in S \cup \{t\} &\Rightarrow F(s', v) = u(s', v), \\ v \in D \cup \{s\} &\Rightarrow F(v, t') = u(v, t'). \end{aligned}$$

Merk op: Als F eigenschap (α) heeft, geldt automatisch dat F de hele capaciteit in de put s' gebruikt. Als F een stroom is, moet hij in dat geval dus vanzelf maximaal zijn.

4.11 Lemma. *Zij N een netwerk. Zij F een stroom op $\overline{G}^*$. Stel dat F aan eigenschap (α) voldoet. Dan is $f_b^* := F \upharpoonright G^*$ een balancerende pre-stroom op G^* met kosten $\text{cost}(f_b^*) = \text{cost}(F)$.*

Bewijs. (I) Stel $v \in S \cup \{t\}$. Dan geldt $0 = \beta(F, v) = \beta(f_b^*, v) - F(s', v)$ dus $-\beta(f_0^*, v) = -\beta(f_0, v) = u(s', v) = F(s', v) = \beta(f_b^*, v)$.

(II) Stel $v \in D \cup \{s\}$. Dan geldt $0 = \beta(F, v) = \beta(f_b^*, v) + F(v, t')$ dus $-\beta(f_0^*, v) = -\beta(f_0, v) = -u(v, t') = -F(v, t') = \beta(f_b^*, v)$.

(III) Stel $v \in B$. Dan geldt $-\beta(f_0^*, v) = -\beta(f_0, v) = 0 = \beta(F, v) = \beta(f_b^*, v)$.

Dus: f_b^* is een balancerende pre-stroom op G^* . Dat $\text{cost}(F) = \text{cost}(f_b^*)$ volgt omdat de toegevoegde kanten kosten 0 hebben. ■

4.12 Lemma. *Stel dat op G^* een balancerende pre-stroom f_b^* bestaat. Dan kunnen we een stroom F op $\overline{G}^*$ maken die aan eigenschap (α) voldoet. Voor deze stroom geldt $\text{cost}(F) = \text{cost}(f_b^*)$.*

Bewijs. We maken F op $\overline{G}^*$ door $F(e) = f_b^*(e)$ op G^* . Verder laten we $F(e)$ gelijk zijn aan $u(e)$, voor alle overige kanten in $\overline{G}^*$. We maken F dus zodat hij aan eigenschap (α) voldoet. Dat F een stroom is zien we nu makkelijk:

(I) Stel $v \in S \cup \{t\}$. Dan geldt $\beta(F, v) = \beta(f_b^*, v) - F(s', v) = \beta(f_b^*, v) - u(s', v) = \beta(f_b^*, v) - (-\beta(f_0, v)) = \beta(f_b^*, v) + \beta(f_0, v) = 0$.

(II) Stel $v \in D \cup \{s\}$. Dan geldt $\beta(F, v) = \beta(f_b^*, v) + F(v, t') = \beta(f_b^*, v) + \beta(f_0, v) = 0$.

(III) Stel $v \in B$. Dan geldt $\beta(F, v) = \beta(f_b^*, v) = -\beta(f_0^*, v) = \beta(f_0, v) = 0$. ■

4.13 Gevolg. *Zij N een netwerk. Dan bestaat op $\overline{G}^*$ een stroom F die aan eigenschap (α) voldoet dan en slechts dan als op G^* een balancerende pre-stroom f_b^* bestaat. In dit geval geldt $\text{cost}(f_b^*) = \text{cost}(F)$.*

4.14 Stelling. *Zij N een netwerk. Zij F een Min Cost Max Flow op $\overline{G}^*$. Als F aan eigenschap (α) voldoet, is $f_b^* := F \upharpoonright G^*$ een balancerende pre-stroom op G^* met minimale kosten. Als F niet aan eigenschap (α) voldoet, dan bestaat op G geen stroom.*

Bewijs. Stel dat F aan eigenschap (α) voldoet. In gevolg 4.13 hebben we bewezen, dat f_b^* een pre-stroom is met $\text{cost}(F) = \text{cost}(f_b^*)$. Stel f_b^* is niet een balancerende pre-stroom met minimale kosten. Dan bestaat er een balancerende pre-stroom g_b^* op G^* met $\text{cost}(g_b^*) < \text{cost}(f_b^*)$. Met lemma 4.12 maken we een stroom F' op $\overline{G}^*$ met eigenschap (α) zodat $\text{cost}(F') = \text{cost}(g_b^*)$. Nu geldt $\text{cost}(F') = \text{cost}(g_b^*) < \text{cost}(f_b^*) = \text{cost}(F)$. Dat is een tegenspraak omdat nu F' een maximale stroom is die minder kosten dan F heeft.

Dus: f_b^* is een balancerende pre-stroom met minimale kosten.

Stel nu dat F niet aan eigenschap (α) voldoet, maar wél dat er een stroom f op G bestaat. Met $f_b := f - f_0$ is f_b^* (zie lemma 4.8 hoe je f_b^* maakt) een balancerende pre-stroom op G^* . Dus, vanwege lemma 4.12, bestaat er een stroom F' op $\overline{G}^*$ die aan eigenschap (α) voldoet (en dus maximaal is), zodat F niet maximaal is. Tegenspraak. ■

4.15. We vatten samen: Als op G stromen bestaan en F een Min Cost Max Flow op $\overline{G}^*$ is, dan is $f_b^* := F \upharpoonright G^*$ een balancerende pre-stroom met minimale kosten. Met gevolg 4.9 is dan $f_b := f_b^* \upharpoonright G$ een balancerende pre-stroom op G (met minimale kosten), zodat vanwege gevolg 4.7 geldt dat $f := f_0 + f_b$ een stroom op G is, met minimale kosten.

Onze voorbereidingen leiden tot het volgende algoritme:

Algoritme 1 (Minimum Cost Flow)

Input: Netwerk $N = (G, d, l, u)$ waar $G = (V \cup \{s, t\}, E)$.

Output: Een Min Cost Flow op N .

- (1) Gegeven G maak de aangepaste graaf G^* .
- (2) Gegeven G^* maak de balancerende graaf $\overline{G}^*$.
- (3) Zoek een maximale stroom F met minimale kosten op $\overline{G}^*$. Gebruik hiervoor algoritme 3 (zie p. 22).
Als F niet aan eigenschap (α) voldoet, dan stop. Op G bestaat dan geen stroom. Anders ga naar stap (4).
- (4) Laat $f_b^* := F \upharpoonright G^*$ en $f_b := f_b^* \upharpoonright G$.
- (5) Laat $f := f_0 + f_b$, waar f_0 de basis pre-stroom op G is.

4.16. We bekijken nu nog waarom in onze situatie altijd een oplossing bestaat. We moeten dus volgens lemma 4.14 alleen bewijzen dat op G stromen bestaan:

4.17 Definitie. Gegeven een netwerk N en een pad P in G noteren wij

$$\mathbb{1}_P : E \rightarrow \{0, 1\} : e \mapsto \begin{cases} 1 & \text{als } e \text{ in } P \text{ voorkomt,} \\ 0 & \text{anders.} \end{cases}$$

Dit kunnen we analoog voor cykels definiëren.

4.18 Lemma. Zij $N = (G, d, l, u)$ een netwerk. We nemen aan $l(e) = 1$ voor $e \in E$ en $u(e) = +\infty$. Dan bestaat er een stroom op G .

Bewijs. Per definitie van een netwerk bestaat voor elk intermediair punt $v \in V$ een $s - v$ pad P_v en een $v - t$ pad P^v . Nu is

$$F := \sum_{v \in V} \left(\sum_{(w,v) \in E} \mathbb{1}_{P_w - v - P^v} \right)$$

een stroom op G met $F(e) \geq 1$ voor iedere $e \in E$. Hierbij noteren we met $P_w - v - P^v$ het $s - t$ pad dat van s naar w naar v naar t loopt. ■

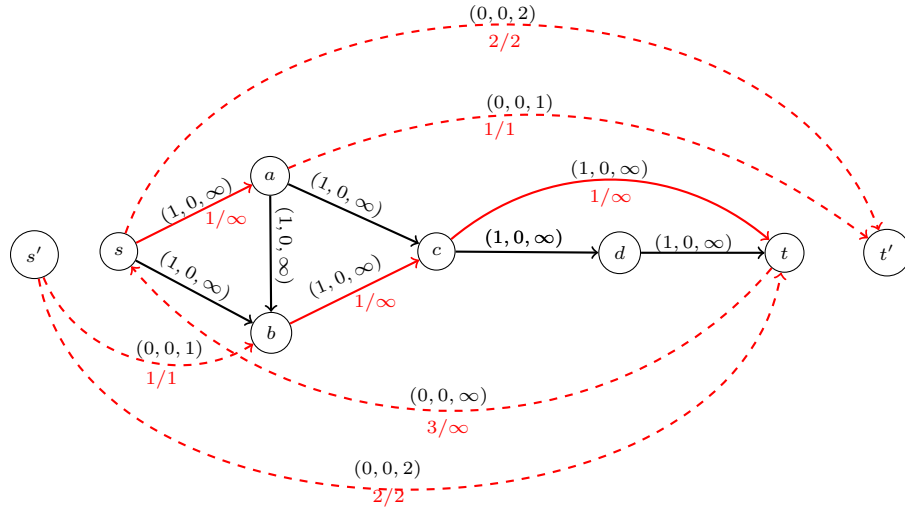
4.19 Gevolg. We kunnen in lemma 4.18 ook aannemen dat $l(e) > 0$, voor iedere $e \in E$.

Bewijs. Zij N zo'n netwerk. Laat $\eta := \max_{e \in E} l(e)$ en zij f een stroom zoals in lemma 4.18, dus $f(e) \geq 1$, voor alle $e \in E$. Dan is nu ηf een stroom op N . Ten eerste geldt voor iedere $v \in V$ dat

$$\beta(\eta f, v) = \eta \beta(f, v) = 0,$$

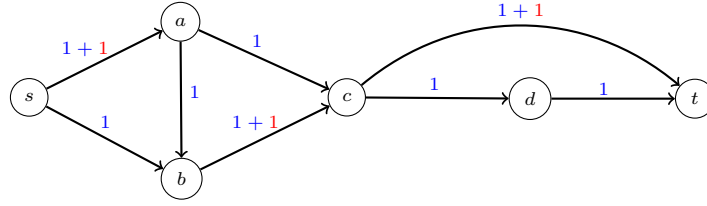
en ten tweede natuurlijk ook dat $\eta f(e) \geq \eta \geq l(e)$, voor iedere $e \in E$. ■

4.20 Voorbeeld. We bekijken het netwerk uit voorbeeld 4.4. Daar hebben we al stappen (1) en (2) van algoritme 1 uitgevoerd. Nu zoeken we nog een maximale stroom F met minimale kosten op $\bar{G}^*$. Deze is door rode letters aangege-



ven.

Als we F tot G beperken, dan krijgen we f_b . Hierbij tellen we de basis pre-stroom f_0 op. De resulterende Min Cost Flow is in de onderstaande graaf te zien:



4.21. In de volgende sectie bekijken we hoe we een Min Cost Max Flow kunnen vinden, zoals in stap 3. van algoritme 1 nodig is.

5 Min Cost Max Flow

5.1. In deze sectie zullen we zien hoe we op een netwerk een maximale stroom met minimale kosten kunnen vinden. Voordat we het algoritme bekijken hebben we eerst een aantal definities en lemma's nodig.

5.2 Definitie. Zij N een netwerk met onderliggende graaf $G = (V \cup \{s, t\}, E)$. Een **circulatie** op N is een functie $f : V \cup \{s, t\} \rightarrow \mathbb{N}$ met

$$\begin{aligned} \forall e \in E : l(e) &\leq f(e) \leq u(e), \\ \forall v \in V \cup \{s, t\} : \beta(f, v) &= 0. \end{aligned}$$

Het verschil met een stroom is, dat voor een circulatie ook in de punten s en t de ingaande stroom gelijk is aan de uitgaande.

5.3 Definitie. We noemen een cykel $C = (v_0, \dots, v_r = v_0)$ een **negatieve cykel** als $\text{cost}(C) = \sum_{i=0}^{r-1} d(v_i, v_{i+1}) < 0$, dus als de totale kosten van alle kanten in C strikt negatief zijn.

5.4 Lemma. Zij N een netwerk en f een circulatie daarop. Dan is f te schrijven als som van stromen op eindig veel cyclen in G . Oftewel: Er zijn $C_1, C_2, \dots, C_r$ cyclen in G en $\delta_1, \delta_2, \dots, \delta_r > 0$ zodat

$$f = \sum_{i=1}^r \delta_i f_i,$$

waar $f_i = \delta_i \mathbb{1}_{C_i}$.

Bewijs. We nemen aan dat f een positieve stroom is. Anders valt er niets te doen. Ons bewijs volgt door middel van een algoritme:

(I) Laat $f_0 := f$.

(II) Gegeven een circulatie f_i voer de volgende subroutine uit:

- (1) Zoek punten $v_0^{(i)}, v_1^{(i)} \in V$ met $(v_0^{(i)}, v_1^{(i)}) \in E$ en $f_i(v_0^{(i)}, v_1^{(i)}) > 0$.
- (2) Gegeven $v_k^{(i)}$ zoek $v_{k+1}^{(i)}$ met $(v_k^{(i)}, v_{k+1}^{(i)}) \in E$ en $f_i(v_k^{(i)}, v_{k+1}^{(i)}) > 0$.

(3) Als $v_{k+1}^{(i)} \in \{v_1^{(i)}, \dots, v_k^{(i)}\}$ dan stop en ga naar **(III)**. Anders ga naar stap (2).

(III) Zij $C_i = (v_1^{(i)}, \dots, v_{n_i}^{(i)})$ de resulterende cykel uit **(II)**, waar $f_i(v_k^{(i)}, v_{k+1}^{(i)}) > 0$ voor iedere $1 \leq k < n_i$. Laat

$$\delta_i := \min\{f(e_k^{(i)}, e_{k+1}^{(i)}) : 1 \leq k < n_i\},$$

waar $e_k^{(i)} = (v_k^{(i)}, v_{k+1}^{(i)})$. Dit geeft een stroom c_i op G , namelijk

$$c_i(e) = \delta_i \mathbb{1}_{C_i}(e) \quad (e \in E).$$

(IV) Laat $f_{i+1} := f_i - c_i$. Als f_{i+1} de nul-stroom is, dan stop. Anders ga naar **(II)**.

Opmerkingen: Het algoritme in stap **(II)** werkt vanwege het behoud van stroom in een circulatie, en het feit dat G eindig is. Verder is f_i in iedere stap een circulatie, omdat het verschil van twee circulaties een circulatie is.

De procedure stopt na maximaal $\#E$ stappen, omdat wij in iedere stap de stroom op minstens een kant gelijk aan 0 maken. Deze kant kan in de volgende stap dus niet meer in een cykel voorkomen. We vinden dus cykel $C_1, C_2, \dots, C_r$ en $\delta_1, \dots, \delta_r$ met $r \leq \#E$ zodat

$$f = \sum_{i=1}^r \delta_i \mathbb{1}_{C_i} = \sum_{i=1}^r c_i,$$

waarbij alle δ_i positief zijn. ■

5.5 Lemma. Zij $N = (G, d, l, u)$ een netwerk. Stel f en f° zijn stromen op N . Als f en f° dezelfde waarde hebben dan is f^* een circulatie op G_{f° . Hierbij is

$$f^*(e) := \begin{cases} \max(0, f(e) - f^\circ(e)) & \text{als } e \in E, \\ \max(0, f^\circ(e^{-1}) - f(e^{-1})) & \text{als } e^{-1} \in E, \\ 0 & \text{anders.} \end{cases}$$

Bewijs. We bewijzen eerst dat f^* een circulatie op $G_{f^\circ} = (V, E_{f^\circ})$ is. Laat u

een willekeurig punt in G_{f° zijn. Dan geldt

$$\begin{aligned}
& \sum_{(v,u) \in E_{f^\circ}} f^*(v,u) - \sum_{(u,v) \in E_{f^\circ}} f^*(u,v) \\
&= \sum_{(v,u) \in E_{f^\circ} \cap E} \max(0, f(v,u) - f^\circ(v,u)) + \sum_{\substack{(v,u) \in E_{f^\circ}, \\ (u,v) \in E}} \max(0, f^\circ(u,v) - f(u,v)) \\
&- \sum_{(u,v) \in E_{f^\circ} \cap E} \max(0, f(u,v) - f^\circ(u,v)) - \sum_{\substack{(u,v) \in E_{f^\circ}, \\ (v,u) \in E}} \max(0, f^\circ(v,u) - f(v,u)) \\
&= \sum_{f(v,u) > f^\circ(v,u)} f(v,u) - f^\circ(v,u) + \sum_{f^\circ(u,v) > f(u,v)} f^\circ(u,v) - f(u,v) \\
&- \sum_{f^\circ(u,v) < f(u,v)} f(u,v) - f^\circ(u,v) - \sum_{f^\circ(v,u) > f(v,u)} f^\circ(v,u) - f(v,u) \\
&= \sum_{(v,u) \in E} f(v,u) - \sum_{(u,v) \in E} f(u,v) + \sum_{(u,v) \in E} f^\circ(u,v) - \sum_{(v,u) \in E} f^\circ(v,u) \\
&= 0 + 0 = 0.
\end{aligned}$$

Nu bewijzen we nog dat f^* ook aan de capaciteitsvoorwaarden op G_{f° voldoet. Noem de capaciteiten op G_{f° nu u_{f° . We moeten alleen de gevallen bekijken waar $f^*(u,v) > 0$. Stel $f^*(u,v) = f(u,v) - f^\circ(u,v)$. Dan $(u,v) \in E$. Met $u_{f^\circ}(u,v) = u(u,v) - f^\circ(u,v)$ vinden we

$$\begin{aligned}
f^*(u,v) &= f(u,v) - f^\circ(u,v) = f(u,v) - (u(u,v) - u_{f^\circ}(u,v)) \\
&= (f(u,v) - u(u,v)) + u_{f^\circ}(u,v) \\
&\leq u_{f^\circ}(u,v).
\end{aligned}$$

De laatste regel volgt omdat $f(u,v) - u(u,v) \leq 0$. Net zo het andere geval: Als $f^*(u,v) = f^\circ(v,u) - f(v,u)$, dan $(v,u) \in E$. Er geldt $u_{f^\circ}(u,v) = f^\circ(v,u)$. Dus

$$f^*(u,v) = f^\circ(v,u) - f(v,u) = u_{f^\circ}(u,v) - f(v,u) \leq u_{f^\circ}(u,v).$$

De laatste regel volgt omdat $f(v,u) \geq 0$. ■

5.6. Merk op: Als $(u,v) \in E$ en $f(u,v) > f^\circ(u,v)$, dan geldt $f^*(u,v) = f(u,v) - f^\circ(u,v)$. Dus $f^\circ(u,v) + f^*(u,v) = f(u,v)$. Net zo: Als $(v,u) \in E$ en $f^\circ(v,u) > f(v,u)$ dan $f^*(u,v) = f^\circ(v,u) - f(v,u)$ zodat $f(v,u) = f^\circ(v,u) - f^*(u,v)$.

5.7 Gevolg. Laat f en f° stromen op een netwerk N zijn, met $\text{val}(f) = \text{val}(f^\circ)$. Dan zijn er cykels $C_1, C_2, \dots, C_N$ in G_{f° en $\lambda_1, \dots, \lambda_N > 0$ zodat het verschil van f en f° te schrijven is als stroom op eindig veel cykels in G_{f° . Bovendien geldt

$$\text{cost}(f) = \text{cost}(f^\circ) + \sum_{i=1}^N \text{cost}(\lambda_i \mathbb{1}_{C_i}).$$

Bewijs. Maak f^* zoals in lemma 5.4. Dan weten we dat f^* een circulatie op G_{f° is. Met lemma 5.4 kunnen we f^* dus schrijven als

$$f^* = \sum_{i=1}^N \lambda_i \mathbb{1}_{C_i},$$

waarbij de C_i cyclen in G_{f° zijn. Dan geldt (zie 5.6) voor $(u, v) \in E_{f^\circ}$ het volgende:

$$\begin{aligned} (u, v) \in E &\Rightarrow f(u, v) = f^\circ(u, v) + \sum \lambda_i \mathbb{1}_{C_i}(u, v) \\ (v, u) \in E &\Rightarrow f(v, u) = f^\circ(v, u) - \sum \lambda_i \mathbb{1}_{C_i}(u, v). \end{aligned}$$

Laat Δ de kosten op G_{f° zijn en d de kosten op G . Als $(u, v) \in E_{f^\circ}$ en $(u, v) \in E$ dan $\Delta(u, v) = d(u, v)$ en als $(v, u) \in E$ dan $\Delta(u, v) = -d(v, u)$. Voor $(u, v) \in E_{f^\circ}$ geldt dus

$$\begin{aligned} \text{cost}(f) &= \sum_{(u,v) \in E \cap E_{f^\circ}} \Delta(u, v) f(u, v) - \sum_{\substack{(u,v) \in E_{f^\circ}, \\ (v,u) \in E}} \Delta(u, v) f(v, u) \\ &= \sum_{(u,v) \in E \cap E_{f^\circ}} \left(d(u, v) f^\circ(u, v) + d(u, v) \sum \lambda_i \mathbb{1}_{C_i}(u, v) \right) \\ &\quad + \sum_{\substack{(u,v) \in E_{f^\circ}, \\ (v,u) \in E}} \left(d(v, u) f^\circ(v, u) + d(v, u) \sum \lambda_i \mathbb{1}_{C_i}(v, u) \right) \\ &= \text{cost}(f^\circ) + \text{cost} \left(\sum \lambda_i \mathbb{1}_{C_i} \right) = \text{cost}(f^\circ) + \sum \text{cost}(\lambda_i \mathbb{1}_{C_i}). \end{aligned}$$

■

5.8 Definitie. Zij N een netwerk, met onderliggende graaf G . Een **potentiaal** op G is een functie

$$\pi : V \cup \{s, t\} \rightarrow \mathbb{R}.$$

5.9 Definitie. Gegeven een potentiaal π en een kostenfunctie Δ op een residuële graaf noteren we de **gereduceerde kosten** met

$$\Delta^\pi(u, v) := \pi(u) - \pi(v) + \Delta(u, v) \quad ((u, v) \in E).$$

5.10 Lemma. Gegeven een netwerk $N = (G, d, l, u)$ en een stroom f erop zijn de volgende uitspraken equivalent:

- (i) f is extreem.
- (ii) G_f bevat geen gerichte negatieve cyclen.
- (iii) Er bestaat een potentiaal π zodat

$$\forall (u, v) \in E_f : \Delta^\pi(u, v) \geq 0.$$

Bewijs. (i) $\Rightarrow$ (ii) Stel G_f bevat een negatieve cykel C . Als $e \in C$ dan verhoog $f(e)$ met 1 en verlaag $f(e)$ met 1 als $e^{-1} \in C$. Noem de resulterende stroom f' en merk op dat f' ook aan de capaciteitsvoorwaarden op G voldoet. Verder geldt

$$\begin{aligned}\text{cost}(f') &= \text{cost}(f) + \sum_{e \in C, e \in G} d(e) - \sum_{e \in C, e^{-1} \in G} d(e) \\ &= \text{cost}(f) + \sum_{e \in C} \Delta(e) \\ &= \text{cost}(f) + \text{cost}(C) < \text{cost}(f).\end{aligned}$$

Merk op dat $\text{cost}(C) < 0$, omdat de kosten van C in G_f worden berekend. Dus is f niet extreem. Tegenspraak.

(ii) $\Rightarrow$ (i) Stel G_f bevat geen negatieve cyclen. Laat $\hat{f}$ een stroom op G zijn met $\text{val}(\hat{f}) = \text{val}(f)$. We nemen verder aan dat $\hat{f}$ extreem is. Dus zeker $\text{cost}(\hat{f}) \leq \text{cost}(f)$. Met gevolg 5.7 zien we dat

$$\hat{f} = f + \sum \lambda_i C_i,$$

waarbij iedere C_i een cykel in G_f is. Per aanname krijgen we dus

$$\text{cost}(\hat{f}) = \text{cost}(f) + \sum \text{cost}(\lambda_i C_i) \geq \text{cost}(f),$$

omdat alle cyclen in G_f niet-negatieve kosten hebben. Dus $\text{cost}(f) = \text{cost}(\hat{f})$, zodat f extreem is.

(ii) $\Rightarrow$ (iii) Noteer met $\pi(u)$ de kosten van het kortste $s-u$ pad in G_f . Omdat G_f geen negatieve cyclen bevat geldt dan

$$\forall (u, v) \in E_f : \Delta^\pi(u, v) = \pi(u) - \pi(v) + \Delta(u, v) \geq 0,$$

omdat $\pi(v) \leq \pi(u) + \Delta(u, v)$ per constructie van π .

(iii) $\Rightarrow$ (ii) Laat π zo een potentiaal zijn met $\Delta^\pi \geq 0$ op E_f . Laat $C = (p_1, p_2, \dots, p_n = p_1)$ een gerichte cykel in G_f zijn. Per aanname geldt

$$\Delta_k := \Delta(p_k, p_{k+1}) \geq \pi(p_{k+1}) - \pi(p_k).$$

Sommeren over k levert

$$\begin{aligned}\text{cost}(C) &= \sum_{k=1}^{n-1} \Delta_k \geq \sum_{k=1}^{n-1} \pi(p_{k+1}) - \pi(p_k) \\ &= \sum_{k=1}^{n-2} \pi(p_{k+1}) + \underbrace{\pi(p_n) - \pi(p_1)}_{=0} - \sum_{k=2}^{n-1} \pi(p_k) \\ &= \sum_{k=2}^{n-1} \pi(p_k) - \sum_{k=2}^{n-1} \pi(p_k) = 0.\end{aligned}$$

Dus $\text{cost}(C) \geq 0$. ■

5.11. We zijn nu in staat een algoritme te formuleren, dat een Min Cost Max Flow oplevert. Na de beschrijving van het algoritme zullen we met behulp van onze voorbereidingen bewijzen, dat het algoritme inderdaad werkt.

Algoritme 2 (Min Cost Max Flow)

Input: Netwerk $N = (G, d, l, u)$ waar $G = (V \cup \{s, t\}, E)$ en $\sum_{(s,u) \in E} u(s, u) < \infty$.

Output: Een Min Cost Max Flow op N .

- (1) Initialiseer $\pi^0 = 0$ en $f^0 = 0$.
- (2) Gegeven f^k en π^k maak G_{f^k} .
- (3) Bereken Δ^{π^k} . Nu bereken de functie σ^k op E_{f^k} waar $\sigma^k(u)$ de kosten van het kortste $s - t$ pad in G_{f^k} aangeeft, t.o.v. de kosten Δ^{π^k} . Gebruik hiervoor algoritme 3 (zie p. 22).
- (4) Als G_{f^k} geen $s - t$ pad meer bevat dan stop. Dat gebeurt als $\sigma^k(t) = +\infty$. Anders ga naar de volgende stap.
- (5) Maak nu f^{k+1} door de stroom langs een kortste pad (volgens σ^k) te verhogen. Deze vinden we ook met algoritme 3.
- (6) We maken een nieuw potentiaal π^{k+1} door

$$\pi^{k+1} = \pi^k + \sigma^k.$$

- (7) Ga naar (2).

5.12 Stelling. *In iedere stap k in algoritme 2 geldt dat f^k extreem is.*

Bewijs. Volgens lemma 5.10 zijn we klaar als voor iedere k geldt dat $\Delta^{\pi^k} \geq 0$ op G_{f^k} . We bewijzen dit met volledige inductie. Voor $k = 0$ valt er niets te bewijzen. Dan geldt namelijk $\Delta^0 = d$, waarbij d de kosten op G zijn. Neem aan het geldt voor $k > 0$, Bekijk:

$$\begin{aligned} \Delta^{\pi^{k+1}}(u, v) &= \pi^{k+1}(u) - \pi^{k+1}(v) + \Delta(u, v) \\ &= \pi^k(u) - \pi^k(v) + \Delta(u, v) + \sigma^k(u) - \sigma^k(v). \end{aligned}$$

Dan is $\Delta^{\pi^{k+1}}(u, v) \geq 0$, precies als $\sigma^k(v) - \sigma^k(u) \leq \Delta^{\pi^k}(u, v)$. Met ons lemma 5.10 weten we dat G_{f^k} geen negatieve cykels bevat zodat σ^k bestaat en per constructie geldt :

$$\sigma^k(v) - \sigma^k(u) \leq \Delta^{\pi^k}(u, v).$$

We zijn dus klaar. ■

5.13. Merk op dat stelling 5.12 ons nu vertelt dat in iedere stap f^k extreem is zodat we iedere keer stap (3) kunnen uitvoeren. Immers: G_{f^k} heeft geen negatieve cycli. Dit stelt ons in staat om σ^k te berekenen. Het gevolg is dus:

5.14 Gevolg. Algoritme 2 stopt na eindig veel stappen. De resulterende stroom f is maximaal en extreem, dus f is een Min Cost Max Flow.

Bewijs. In 5.13 hebben we uitgelegd waarom het algoritme in stap (3) altijd de functie σ^k kan berekenen. Dus kan het algoritme in stap (3) niet vast lopen. Omdat de stroom iedere keer echt groter wordt, moet het algoritme op een gegeven moment stoppen. (Hiervoor zorgt de aanname dat $\sum_{(s,u) \in E} u(s,u) < \infty$). Dat de resulterende stroom maximaal is volgt uit het algoritme van Ford en Fulkerson: Als we iedere keer de stroom langs een $s - t$ pad verhogen, totdat dit niet meer kan, krijgen we een maximale stroom. Het bewijs van dit algoritme is een elementair resultaat in de grafentheorie. ■

5.15. We zullen nu nog bekijken hoe we de functie σ^k in stap (3) kunnen berekenen. Hiervoor gebruiken we het algoritme van Dijkstra.

6 Kortste route

6.1. We zullen in deze sectie naar het algoritme van Dijkstra kijken. Dit algoritme stelt ons in staat een kortste $s - t$ pad in een netwerk N te vinden, zolang N geen negatieve cycli bevat. We merken verder nog op dat we de uitdrukkingen *kortste pad* en *goedkoopste pad* als equivalent beschouwen.

Algoritme 3 (Kortste route)

Input: Netwerk $N = (G, d, l, u)$ waar $G = (V \cup \{s, t\}, E)$.

Output: Een functie $\sigma : V \cup \{s, t\} \rightarrow \mathbb{R} \cup \{\infty\}$ waar

$$\sigma(v) = \min\{\text{cost}(P) : P \text{ is een } s - v \text{ pad}\}.$$

En een functie $\text{pr} : V \cup \{t\} \rightarrow V \cup \{s, t\}$.

(1) Laat $\sigma(s) := 0$ en $\sigma(v) = +\infty$ voor $v \in V \cup \{t\}$.

(2) Laat $S := \{s\}$.

(3) Als $S = \emptyset$, ga dan naar stap (6). Anders zoek een $v^* \in S$ met

$$\sigma(v^*) = \min_{v \in S} \sigma(v).$$

(4) Pas nu σ als volgt aan:

$$\sigma(v) := \min(\sigma(v), \sigma(v^*) + d(v^*, v)) \quad ((v^*, v) \in E).$$

Als deze procedure de waarde van σ in een punt v strikt kleiner maakt, dan voeg v aan S toe en definieer $\text{pr}(v) := v^*$.

- (5) Laat nu $S := S \setminus \{v^*\}$ en ga naar stap (3).
- (6) Als $\sigma(t) = \infty$, dan bestaat er geen $s - t$ pad. Anders maken we nu een kortste pad P : Laat $w_0 := t$.
- (7) Gegeven w_k laat $w_{k+1} := \text{pr}(w_k)$. Herhaal deze stap, totdat $w_{k+1} = s$.
- (8) Laat $P := (w_n = s, w_{n-1}, \dots, w_0 = t)$.

6.2. Het is duidelijk dat het algoritme vast loopt, als G een negatieve cykel bevat. Het algoritme stopt dus alleen als G geen negatieve cyclen bevat. Voor een algemene implementatie moeten we hiermee rekening houden en checken dat zoiets niet kan gebeuren. Voor ons doel hoeft dat niet, omdat we het algoritme alleen op netwerken zullen toepassen, die geen negatieve cyclen bevatten.

6.3 Stelling. *Zij $N = (G, d, l, u)$ een netwerk, zodat G geen negatieve cyclen bevat. Dan stopt algoritme 3 na eindig veel stappen. Verder is het zo gevonden pad P een $s - t$ pad met minimale kosten.*

6.4. Dat het algoritme van Dijkstra werkt is een algemeen bekend resultaat, zodat we dit hier niet hoeven te bewijzen.

7 Oplossing

7.1. We hebben nu de nodige hulpmiddelen om het probleem, zoals in 3.2 beschreven, op te lossen. We formuleren dit als stelling:

7.2 Stelling. *Zij $N = (G, d, l, u)$ gegeven. Laat $l(e) > 0$ en $u(e) = \infty$ voor $e \in E$. Dan bestaan er stromen op G met minimale kosten. Stel f is zo een stroom op G . Door middel van f kunnen we dan een kantenoverdekking $\mathcal{F}_0$, bestaande uit $s - t$ paden, vinden zodat*

$$\sum_{P \in \mathcal{F}} \text{cost}(P)$$

minimaal is. Dit lost punt 3. in onze probleembeschrijving 3.2 op.

7.3. Om stelling 7.2 te bewijzen, hebben we weer een algoritme nodig. Deze laat ons uit een Min Cost Flow zo'n kantenoverdekking $\mathcal{F}_0$ construeren. Hierna moeten we alleen nog stap 4. in 3.2 beschrijven, ofwel hoe we uit $\mathcal{F}_0$ een bruikbare collectie $\mathcal{C}_0$ maken.

Algoritme 4

Input: Netwerk $N = (G, d, l, u)$ met $l(e) > 0$ en $u(e) = \infty$ waar $G = (V \cup \{s, t\}, E)$. Verder is f een stroom op G .

Output: Een kantenoverdekking $\mathcal{F}$ van $s - t$ paden in G .

- (1) Zij $h : E \rightarrow \mathbb{N}$ gegeven door $h := f$.
- (2) Kies een kant (s, v) met $h(s, v) > 0$ en definieer $P := (s, v)$, en $h(s, v) := h(s, v) - 1$. Als zo'n kant niet bestaat, ga naar stap (4).
- (3) Zij v het laatste punt in P . Als $v = t$ dan voeg P aan $\mathcal{F}$ toe en ga naar stap (2). Anders kies een kant (v, w) met $h(v, w) > 0$, en voeg w toe aan P . Definieer $h(v, w) := h(v, w) - 1$. Herhaal deze stap.
- (4) De waarde van de stroom h is gelijk aan 0, dus h is een circulatie. Als $h(e) = 0$, voor alle $e \in E$, dan stop met het algoritme. Zo niet kies een $(v_0, v_1) \in E$ met $h(v_0, v_1) > 0$ en $\exists (u, v_0) \in E : h(u, v_0) = 0$. Definieer dan $h(v_0, v_1) := h(v_0, v_1) - 1$ en $C := (v_0, v_1)$.
- (5) Gegeven $v_k \in C$, met k maximaal, kies een kant (v_k, v_{k+1}) met $h(v_k, v_{k+1}) > 0$, laat $C := (v_0, v_1, \dots, v_{k+1})$ en definieer $h(v_k, v_{k+1}) := h(v_k, v_{k+1}) - 1$. Ga naar stap (6) als $v_{k+1} = v_0$. Herhaal deze stap.
- (6) De kant (u, v_0) uit stap (5) zit in een pad $P = (s, \dots, u, v_0, \dots, t) \in \mathcal{F}$. Vervang dan P door $P = (s, \dots, u, C, \dots, t)$ en ga terug naar stap (4).

Bewijs van stelling 7.2. De existentie van een Min Cost Flow op G wordt in lemma 4.18 bewezen. We bewijzen hier, hoe we uit een Min Cost Flow f op G een kantenoverdekking $\mathcal{F}$ van $s - t$ paden kunnen maken, ofwel dat algoritme 4 werkt.

- In stap (2) is h een stroom, en door behoud van stroom, kunnen stap (3) en (5) altijd uitgevoerd worden.
- Bij stap (6): De kant (u, v_0) zit in een pad $P \in \mathcal{F}$, omdat $h(u, v_0) = 0$ en oorspronkelijk was $h(u, v_0) = f(u, v_0) \geq l(u, v_0) > 0$. Er is dus minstens een pad $P \in \mathcal{F}$ die de kant (u, v_0) bevat.
- In stap (4) kunnen we inderdaad een kant (v_0, v_1) met $h(v_0, v_1) > 0$ vinden, zodat er een kant (u, v_0) is met $h(u, v_0) = 0$. Als dat niet waar zou zijn, dan kunnen we nagaan dat h geen circulatie is: Stel dat iedere kant $(v, w) \in E$

met $h(v, w) > 0$ de volgende eigenschap heeft:

$$(v', v) \in E \quad \Rightarrow \quad h(v', v) > 0.$$

Pak nu een willekeurige $(v, w) \in E$ met $h(v, w) > 0$. Dan is er een $s - v$ pad $P = (s, p_0, p_1, \dots, p_n, v)$. Per aanname geldt $h(p_n, v) > 0$. Dus ook $h(p_{n-1}, p_n) > 0$. Ga zo door. Dan zien we $h(s, p_0) > 0$ zodat h geen circulatie is.

Conclusie: Er moet minstens een kant $(v, w) \in E$ zijn met $h(v, w) > 0$ zodat er een kant $(v, v') \in E$ is met $h(v, v') = 0$.

- Uit een stroom kunnen we dus een collectie paden afleiden, en uit zo'n collectie $s - t$ paden kunnen we een stroom f maken door

$$f(u, v) = \#\{P \in \mathcal{F} : (u, v) \in P\}.$$

In dat geval geldt

$$\begin{aligned} \sum_{P \in \mathcal{F}} \text{cost}(P) &= \sum_{P \in \mathcal{F}} \sum_{(u, v) \in P} d(u, v) \\ &= \sum_{(u, v) \in E} d(u, v) \cdot \#\{P \in \mathcal{F} : (u, v) \in P\} \\ &= \sum_{e \in E} d(e) \cdot f(e) = \text{cost}(f). \end{aligned}$$

Als we dat nu op een Min Cost Flow toepassen, hebben we een collectie $\mathcal{F}$ van $s - t$ paden met minimale kosten gevonden. ■

7.4. We moeten nu nog kijken hoe we punt 4. in de probleembeschrijving 3.2 oplossen. Stel we passen stelling 7.2 op de graaf G' , zoals in punt 1. van 3.2 toe. Dit geeft een kantenoverdekking $\mathcal{F}_0$ van $s' - t'$ paden met minimale totale kosten. Uit $\mathcal{F}_0$ kunnen we $\mathcal{C}_0$ construeren door voor $P = (s', e_1, \dots, e_n, t') \in \mathcal{F}_0$ met $e_i = (v_i, w_i)$ het pad $P = (v_1, v_2, \dots, v_n, w_n)$ aan $\mathcal{C}_0$ toe te voegen. Uit $\mathcal{C}_0$ kunnen we op unieke wijze weer $\mathcal{F}_0$ construeren. Dan geldt $\sum_{P \in \mathcal{F}} \text{cost}(P) = \sum_{P \in \mathcal{C}_0} \ell(P)$, zodat $\mathcal{C}_0$ inderdaad een bruikbare collectie met minimaal totaal aantal kanten is. Hiermee is het probleem dus opgelost. We vatten samen wat we nu hebben gedaan:

We hebben een vertaling van de originele graaf naar de lijngraaf gemaakt. Voor deze vinden we een Min Cost Flow, waaruit we dan zo'n *minimale* kantenoverdekking $\mathcal{F}_0$ vinden. Hieruit kunnen we dan een *minimale* bruikbare collectie $\mathcal{C}_0$ voor de originele graaf maken. Per constructie is $\mathcal{C}_0$ dan een collectie paden, zodat iedere combinatie van opeenvolgende kanten in een zeker pad $P \in \mathcal{C}_0$ voorkomt, en zodanig dat de totale lengte

$$\sum_{P \in \mathcal{C}_0} \ell(P)$$

minimaal is.

Referenties

- [1] A. V. Aho and D. Lee, *Efficient Algorithms for Constructing Testing Sets, Covering Paths, and Minimum-flows*, AT & T Bell Laboratories, 1991
- [2] Jack Edmonds, *Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems*, Journal of the Association for Computing Machinery, Vol. 19, No. 2, April 1972, p. 248-264