LP. Leksjon 9: Kapittel 13: Nettverk strøm problemer, forts.2

Vi tar siste runde om (MKS): minimum kost nettverk strøm problemet.

- Skal oppsummere algoritmen.
- Se på noen detaljer.
- Noen kombinatorisk anvendelser

Den primale nettverk simpleks algoritmen

Algoritme oppsummering: Vi starter med et spenntre T der den tilhørende treløsningen x_B er tillatt, dvs. $x_B \geq O$.

- Sjekk optimalitet ved å beregne duale variable y og z og sjekke om $z \geq O$.
- Hvis ikke optimal: gjennomfør en pivotering. Dette skjer ved å velge en kant (u, v) der $z_{uv} < 0$, og finne et nytt spenntre T' ved å legge til e og fjerne en viss annen kant (slik at T' blir et spenntre). Oppdater treløsningen x.

Beregningen av både x og y skjer ved "bladeliminasjon" ved å bruke triangulariteten slik vi har sett. Beregningen av hver z_{uv} skjer direkte fra den tilhørende likningen i det duale problemet:

$$z_{uv} = c_{uv} + y_u - y_v.$$

Effektiv oppdatering av variablene i en pivotering

Oppdatering av x: Vi trenger ikke starte helt på nytt når ny x skal beregnes. Grunnen er at x endrer seg bare for visse kanter. Den nye kanten e (inngående basisvariabel) gir opphav til en syklus C i grafen. Endringen av x blir da

- La inngående variabel få verdien ϵ (som skal bestemmes).
- For hver kant $f \in C$ med samme retning som e i syklusen ("foroverkanter"), så vil x_f økes med ϵ. I de kantene i C som går motsatt vei ("bakoverkanter") vil strømmen bli redusert med ϵ. Ingen andre x_f-er endres.
- ϵ velges som minimum av strømmen i bakoverkanter.

Oppdatering av spenntreet T: Inngående kant e erstatter en bakoverkant f i syklusen som nå har fått verdien $x_f = 0$. Dette gir et nytt spenntree T'.

Kommentarer:

1. Det kan være flere bakoverkanter som får strømverdi 0; da får vi en degenerert løsning etter pivoteringen. Hvis man ikke bruker en bestemt pivoteringsregel, så velges bare en av disse kantene vilkårlig som utgående variabel.
2. Degenerasjon forekommer ofte i (MKS) problemer. Noen hevder at ofte er ca. 70-80% av pivoteringene degenererte! Men heldigvis oppfattes ikke sirkling som noe praktisk problem. Videre finnes en pivoteringsregel (via såkalte sterkt tillatte spenntrær) som unngår sirkling (og som i praksis ofte reduserer antall pivoteringer).

3. Hvis syklusen C ikke har noen bakoverkanter, så er problemet ubegrenset. Altså: vi kan ved å sende passe stor strøm rundt i denne syklusen få totalkostnad til å gå mot $-\infty$. (Dette skjer vanligvis ikke i praktiske problemer.)
4. Som nevnt har man også ofte øvre skranker på x i (MKS) problemer:

$$0 \leq x_{uv} \leq a_{uv} \quad ((u,v) \in E).$$

   Algoritmen kan enkelt tilpasses denne mer generelle situasjonen. Kort fortalt er endringene følgende. Hver x_e der $e \notin T$ (ikkebasisvariable) har enten verdien 0 eller den øvre skranken a_e. Ved bestemmelse av maksimal ϵ må vi også passe på at ingen foroverkanter får strøm over kapasiteten. Ikkebasisvariable som er på sin øvre skranke, kan bare reduseres, så optimalitetstesten må utvides i henhold til dette. Det betyr at en slik variabel kan taes inn i basis dersom den tilhørende z-komponenten er positiv (mens for ikkebasisvariable på sin nedre skranke 0 er de kandidater når z er negativ der).

Oppdatering av y: Anta at vi skal føye til e og fjerne f i vårt spenntre T. $T \setminus \{f\}$ består av to (sub)trær T_r og T' der T_r innholder rotnoden r. Da gjelder følgende for oppdateringen av de duale variablene y:

- Hvis ny kant e går fra T_r til T', så økes alle duale variable i T' med samme størrelse Δ (lik verdien på den duale slakkvariabelen z_e, så $\Delta < 0$). Variablene i T_r forblir uendret.
- Hvis ny kant e går fra T' til T_r, så reduseres alle duale variable i T' med Δ. Variablene i T_r forblir uendret.

Oppdatering av z: Her gjelder:

- Det blir ingen endring på z_{uv} dersom u og v ligger i det samme av de to subtrærne T_r og T' (fordi y_u og y_v er endret med samme størrelse).
- For kanter mellom de to subtrærne og som har samme retning for inngående kant e vil z bli redusert med (den opprinnelige) verdien z_e, mens de som går motsatt vei økes med z_e.

Hvordan finne startløsning?

Vi har hittil bare antatt at vi har en tillatt startløsning, dvs. et spenntre slik at den tilhørende treløsningen er ikkenegativ. Noen ganger er man heldig og ser en slik løsning direkte, men vanligvis kreves mer arbeid.

- Det er flere teknikker for å finne en tillatt startløsning. En mulighet er å bruke den duale simpleks algoritmen, tilpasset strukturen i (MKS).

Duale simpleks algoritme for å finne tillatt startløsning:

- Se på (MKS) problemet med modifisert kostnadsfunksjon nemlig der $c = O$. Da er ethvert spenntre dualt tillatt, så vi velger et vilkårlig spenntre T. Basisen er altså dualt tillatt, og hvis vi er heldig er den også primalt tillatt. I så fall er jobben gjort og vi brukes T som startløsning for vår (MKS) algoritme med den opprinnelig kostnadsfunksjonen c. Hvis derimot x har negative komponenter, går vi til skritt 2.
- Velg en kant $e = (u, v)$ med $x_e < 0$. Vil nå foreta en pivotering der e forlater basis. Grunnen er at vi da får strøm 0 i e (siden variabelen er utenfor basis). Spørsmålet er hvilken variabel som skal inn i basis. Se på de to subtrærne T_r og T' som oppstår hvis e fjernes fra T (T_r inneholder rotnoden r). For å få et nytt spenntre må vi velge en kant f som forbinder de to trærne (har en endenode i hvert tre). Klart at f må gå motsatt vei av e (for ellers ville en økning av x_f gi en lavere x_e og vi vil jo øke x_e). Men det kan være flere slike kanter!

- (forts.) Vi velger da f med z_f minst mulig (blant kanter mellom de to subtrærne og med motsatt retning av e). Grunnen til dette valget er at vi etter pivoteringen vil ha en ny løsning som også er dualt tillatt. Det er jo slik at alle kandidatkantene får redusert sin z-verdi med samme størrelse; dette så vi foran i forbindelse med oppdateringen av z-variabelen i en pivotering. På denne måten er utgående kant e og inngående kant f bestemt og vi gjennomfører pivoteringen og oppdaterer variable som in den vanlige (MKS) algoritmen. Vi gjentar slike pivoteringer inntil x er ikkenegativ. (Dette lar seg gjøre dersom problemet virkelig har en tillatt strøm.)

Vi forlater nå algoritmer og ser på en interessant anvendelse av teorien innenfor kombinatorikk.

Anvendelse: kombinatorikk/heltallighet

Vi kaller en vektor heltallig dersom alle komponenenter er heltall. Her er resultatet.

Teorem *Betrakt (MKS) problemet*

$$\min\{c^T x : Ax = -b, \ \ x \geq O\}$$

der b er heltallig, og anta at en optimal løsning fins. Da fins en optimal løsning x som er heltallig.

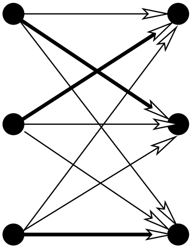
Bevis: I hver iterasjon beregnes treløsningen x ved bladeliminasjon. Den første variabelen som beregnes blir da lik $\pm b_v$ som er heltall. Hver av de neste variablene blir en sum av visse heltall (en viss b_v og $\pm$ andre allerede beregnede strømmer som er heltallige). Dermed er x heltallig. $\square$

Kommentarer:

- Som forklart i beviset er altså alle x underveis i algoritmen heltallige, ikke bare den optimale løsningen.
- Dersom c er heltallig, så fins en optimal dual løsning (y, z) som er heltallig. (Dette ser vi direkte av algoritmene for å beregne y og deretter z.)
- Teoremet over gjelder også når vi har heltallige kapasiteter på x, så $0 \leq x_e \leq a_e$ $(e \in E)$ der a er heltallig. Så også da fins en heltallig optimal løsning x.

Dette teoremet har en lang rekke anvendelser i kombinatorikk (diskret matematikk). La oss se på en slik.

Betrakt grafen



La $b_v = 1$ for hver av de tre nodene til venstre og $b_v = -1$ for de til høyre. Dessuten har vi visse kostnader for kantene, men disse behøver vi ikke bry oss om nå.

- Siden b er heltallig, finnes en optimal strøm x som også er heltallig (og vi vet hvordan vi finner den).
- Men ut fra verdiene i b ser vi at x må ha nøyaktig tre komponenter som er lik 1, mens resten er 0.
- Videre er de tre kantene der $x_e = 1$ "disjunkte", dvs. de har ingen endenoder felles. En slik kantmengde (riktignok i en urettet graf) kalles en perfekt matching, se kantene med tykke linjer i figuren.
- I dette eksemplet er det $3! = 6$ ulike perfekte matchinger og alle svarer til tillatte løsninger i (MKS) problemet.
- Men da følger at løsningen vi fant faktisk er en optimal perfekt matching, dvs. en perfekt matching med minimum total kostnad.

- Dette problem kalles, naturlig nok, minimum kost (vekt) perfekt matching problemet i en bipartitt graf (som er den grafen vi får hvis vi ignorerer retningen på kantene; den er bipartitt (todelt) fordi hver kant har en endenode i hver av de to node-delmengdene (venstre og høyre)).
- Generelt har vi n noder til venstre og n noder til høyre. Antallet perfekte matchinger er da $n!$, som selv for moderat n er et gigantisk tall. Så metoden finner optimal perfekt matching i et problem der enummerering av alle muligheter er umulig. Dette er viktig!
- En praktisk anvendelse av perfekt matching er ved tilordning av jobber til personer, datamaskiner eller noe annet. Anta at hver person skal utføre nøyaktig én jobb og hver jobb skal gjøres av én person. Problemet er da å finne en slik tilordning av jobber med lavest mulig kostnad: kalles tilordningsproblemet. Det finnes også andre varianter av simpleksalgoritmen (også teoretisk effektive) for dette problemet.

Andre spesialtilfeller av (MKS) som er viktige i mange sammenhenger er

- korteste vei problemet.
- maksimum strøm problemet.
- transportproblemet. Dette generaliserer bipartitt perfekt matching ved generell tilførsel/behov b som er negativ "til venstre" og positiv "til høyre"; gi dette problemet en transport-tolkning! Og dette er et grunnleggende problem i feltet transport optimering.

Igjen finnes effektive kombinatoriske algoritmer; mer om disse tingene i høstkurset INF-MAT 5360 Matematisk optimering. Der lærer du også mer om matematikken som danner basis for analyse/forståelse/løsning av disse problemene.

Du inviteres dit!