Een naald in een hooiberg

Een naald in een hooiberg

Rede

in verkorte vorm uitgesproken bij de aanvaarding van het ambt van
gewoon hoogleraar Kwantitatieve Economie,
in het bijzonder Besliskunde aan de
Rijksuniversiteit Limburg op
vrijdag 17 november 1995

door

Antoon Kolen

Mijnheer de Rector Magnificus,

Dames en heren,

1 Inleiding.

De naam Besliskunde voor mijn vakgebied is bijzonder goed gekozen. Iedereen kan zich bij het woord besliskunde een beeld vormen van wat het vakgebied inhoudt. Dit beeld komt naar alle waarschijnlijkheid dichter in de buurt van de werkelijkheid dan de formele definities, zoals die o.a. gegeven zijn door de 'Operations Research Society of America' [1], en door de 'Operational Research Society of Great Britain' [2]. Ik laat deze definities dan ook graag onuitgesproken. De Nederlandse naam Besliskunde, voor het internationaal gebruikelijke Operations Research, is voor het eerst in 1957 geïntroduceerd door prof.dr.D.van Dantzig in een diësrede, gehouden aan de Universiteit van Amsterdam [3].

Een deelgebied van de besliskunde is de discrete optimalisering. Binnen dit deelgebied werken aan de Rijksuniversiteit Limburg leden van de vakgroep Kwantitatieve Economie van de Faculteit der Economische Wetenschappen en Bedrijfskunde samen met leden van de vakgroep Wiskunde van de Faculteit der Algemene Wetenschappen. Wij wilden onze onderzoekgroep 'Discrete Optimalisering Maastricht' noemen. De binnen onze faculteit gangbare praktijk om groepen met hun afkorting aan te duiden, heeft ons daarvan doen afzien.

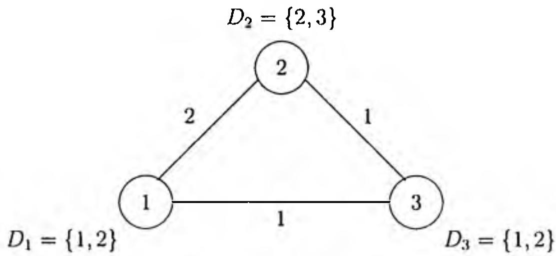
In deze oratie wil ik U een indruk geven van wat een discreet optimaliseringsprobleem is, welke moeilijkheden zich voordoen bij het oplossen er van, en welke ideeën ontwikkeld zijn om deze moeilijkheden het hoofd te bieden.

2 Omschrijving discrete optimalisering.

Mijn voorbeelden van discrete optimaliseringsproblemen komen uit de telecommunicatie en betreffen het toewijzen van frequenties aan basisstations in het mobiele telefoonnetwerk. Aan soortgelijke problemen hebben wij de afgelopen twee jaren in het kader van een Europees onderzoeksproject gewerkt [4].

Een mobiele telefoonnetwerk bestaat uit een aantal basisstations. Elk basisstation is voorzien van een of meer antennes, waarmee een bepaald gebied in de omgeving van het basisstation wordt afgedekt. Wil een gebruiker van het mobiele telefoonnetwerk bellen dan wordt zijn gesprek overgenomen door een van de basisstations binnen het bereik waarvan gebeld wordt. De hoeveelheid gesprekken, welke door een basisstation gelijktijdig kan worden afgewerkt, hangt af van het aantal frequenties dat voor dat basisstation beschikbaar is. De locaties van de basisstations, het soort antenne per basisstation, en het aantal frequenties dat aan een basisstation moet worden toegewezen, wordt bepaald door de ontwerpers van het mobiele netwerk. Onze taak begint na het ontwerp en bestaat uit het bepalen welke specifieke frequenties aan een basisstation moeten worden toegewezen. In mijn voorbeelden ga ik er van uit dat aan ieder basisstation precies één frequentie moet worden toegewezen. Dit is geen wezenlijke beperking ten opzichte van de werkelijkheid, omdat als er meerdere frequenties aan één basisstation moeten worden toegewezen, we dit ook kunnen zien als meerdere basisstations op dezelfde locatie, waarbij aan ieder basisstation slechts één frequentie wordt toegewezen. Twee basisstations kunnen elkaar soms storen. Om storing te voorkomen moeten de frequenties die aan deze basisstations worden toegewezen een bepaalde bandbreedte uit elkaar liggen. De beschikbare frequenties verschillen per basisstation, en hebben o.a. te maken met afspraken met buurlanden. Dit betekent dat er in Limburg minder frequenties beschikbaar zijn dan in de randstad. De verzameling frequenties welke beschikbaar is voor een basisstation noemen we het *domein* van het basisstation.

Een klein voorbeeld, genaamd Voorbeeld 1, ziet U in onderstaande Figuur 1.



Figuur 1. Probleem instantie Voorbeeld 1.

In Figuur 1 zijn er drie basisstations. Het domein van basisstation 1 (aangegeven met D_1) bestaat uit de frequenties 1 en 2; het domein D_2 van basisstation 2 uit de frequenties 2 en 3; en het domein D_3 van basisstation 3 uit de frequenties 1 en 2.

In het voorbeeld ziet U dat er lijnen getekend zijn tussen sommige paren basisstations. In het voorbeeld zijn dit toevallig alle paren. Een lijn tussen twee basisstations betekent dat er storing optreedt tussen deze twee basisstations indien de afstand tussen de toegewezen frequenties minder is dan de aangegeven waarde. Deze waarde is 2 voor de lijn corresponderende met de basisstations 1 en 2, en deze waarde is 1 voor de overige lijnen.

Ik noem twee frequentietoewijzingsproblemen gerelateerd aan dit voorbeeld.

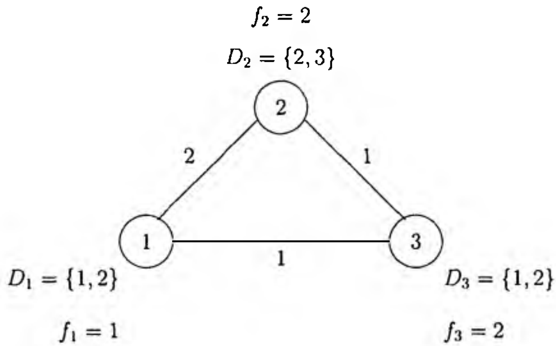
In het eerste probleem is het doel aan ieder basisstation één frequentie uit zijn domein toe te wijzen, zodanig dat er zo weinig mogelijk storing is. Als maat voor de storing van een frequentietoewijzing nemen we het aantal lijnen waarop storing optreedt.

Het tweede probleem krijgen we indien er storingsvrije oplossingen bestaan. In dat geval wil men graag een storingsvrije oplossing, welke zo weinig mogelijk frequenties gebruikt.

Beide problemen zijn discrete optimaliseringsproblemen.
Laten we eens kijken wat deze twee problemen gemeenschappelijk hebben om daaruit de karakteristieken van een discreet optimalisering af te leiden.

Beide problemen hebben een eindig aantal oplossingen. Een *oplossing* is een frequentietoewijzing, en bestaat uit voor ieder basisstation één keuze uit zijn domein.

We geven met f_i de frequentie aan, welke is toegewezen aan basisstation $i, i = 1, 2, 3$. De oplossing $(f_1, f_2, f_3) = (1, 2, 2)$ voor Voorbeeld 1 staat weergegeven in Figuur 2.



Figuur 2. De oplossing (1,2,2).

Voor beide problemen is een deelverzameling van de oplossingen gegeven, welke we *toegelaten oplossingen* noemen. Voor het eerste probleem zijn alle oplossingen toegelaten. Voor het tweede probleem zijn alleen de storingsvrije oplossingen toegelaten.

In beide problemen wordt aan iedere oplossing een *waarde* toegekend. Voor het eerste probleem is dat het aantal lijnen waarop storing optreedt. Voor het tweede probleem is dat het aantal verschillende frequenties.

Voor het eerste probleem heeft de oplossing (1,2,2) waarde 2, omdat storing optreedt op zowel de lijn tussen basisstation 1 en basisstation 2, als op de lijn tussen basisstation 2 en basisstation 3.
Voor het tweede probleem heeft deze oplossing waarde 2, omdat er twee verschillende frequenties voorkomen in de oplossing.

In Figuur 3 staan alle oplossingen met voor iedere oplossing de waarde voor beide problemen.

(f_1, f_2, f_3)	Storing	Frequenties
$(1,2,1)$	3	2
$(1,2,2)$	2	2
$(1,3,1)$	1	2
$(1,3,2)$	0	3
$(2,2,1)$	1	2
$(2,2,2)$	3	1
$(2,3,1)$	1	3
$(2,3,2)$	2	2

Figuur 3. Alle oplossingen.

In beide problemen is de doelstelling het vinden van een toegelaten oplossing met minimale waarde. Een toegelaten oplossing met minimale waarde noemen we een *optimale oplossing*.

We zien dat $(1,3,2)$ de unieke optimale oplossing is voor het probleem waarin de storing wordt geminimaliseerd. Het is tevens de enige toegelaten oplossing, en dus optimale oplossing, voor het probleem waarin het aantal gebruikte frequenties wordt geminimaliseerd.

Samengevat, een probleem is een *discreet optimaliseringsprobleem* als

1. Het aantal oplossingen eindig is.
2. Voor iedere oplossing is aangegeven of het een toegelaten oplossing is of niet.
3. Aan iedere oplossing een waarde is toegekend.
4. De doelstelling is het vinden van een toegelaten oplossing van minimale waarde.

3 Moeilijkheid bij het oplossen.

Wanneer mensen voor het eerst geconfronteerd worden met een discreet optimaliseringsprobleem zien ze vaak niet waar het probleem ligt bij het oplossen van een dergelijk probleem. Het aantal oplossingen is immers per definitie eindig, en daarom kan de optimale oplossing worden gevonden door alle oplossingen één voor één te bekijken, en steeds de beste toegelaten oplossing tot dan toe bij te houden.

Voor het voorbeeld betekent dit dat we het lijstje oplossingen in Figuur 3 moeten doorlopen.

Als deze zogenaamde *expliciete aftellingsmethode* zou werken zou de discrete optimalisering natuurlijk nooit als onderzoeksgebied zijn ontstaan. Een voorbeeld zal duidelijk maken dat deze expliciete aftellingsmethode stuk loopt op het enorme aantal oplossingen.

Het aantal oplossingen voor een instantie van het frequentietoewijzingsprobleem van het beschreven type wordt verkregen door het aantal mogelijke frequenties per basisstation met elkaar te vermenigvuldigen.

Voor Voorbeeld 1 krijgen we dan $2 * 2 * 2 = 8$ oplossingen.
Dit kan als volgt worden ingezien. Voor basisstation 1 zijn er twee mogelijke keuzes voor wat betreft de toe te wijzen frequenties. Bij ieder keuze zijn er twee mogelijke keuzes voor de frequentie, welke toegewezen wordt aan basisstation 2. In totaal geeft dit $2 * 2$ mogelijkheden voor de basisstations 1 en 2. Bij elk van deze mogelijkheden zijn er weer twee mogelijke keuzes voor de frequentie, welke toegewezen wordt aan basisstation 3. In totaal zijn er dus $2 * 2 * 2$ mogelijke frequentietoewijzigingen.

Een van de kleinste frequentietoewijzingsproblemen, waaraan wij in europees verband gewerkt hebben, bestaat uit 200 basisstations met gemiddeld 40 frequenties per domein. Het aantal oplossingen is dan van de orde van grootte van 10 tot de macht 320. Aangenomen dat ik duizend computers, die ieder 1 miljoen oplossingen per seconde kunnen doorrekenen, parallel aan het werk zet, dan zijn na 30 jaren ongeveer 10 tot de macht 18 oplossingen bekeken [5]. Zoals u ziet is dat nog maar een fractie van het totale aantal oplossingen.

De verzameling van alle oplossingen is de hooiberg waar ik in de titel van mijn oratie naar verwijs. De naald die we zoeken is, ofwel een optimale oplossing, ofwel een zo goed mogelijke oplossing indien we optimaliteit onhaalbaar achten.

4 Oplossingsideeën.

Het vinden van de gewenste oplossing kan alleen maar door een bepaalde structuur in de oplossingsruimte aan te brengen. In Sectie 4.1 geef ik een mogelijke structuur voor het geval we een *heuristische oplossing* zoeken, dwz een oplossing die zo goed mogelijk is, maar niet noodzakelijkerwijs optimaal. In Sectie 4.2 geef ik een mogelijke structuur indien de optimale oplossing wordt gezocht.

4.1 Heuristische oplossing.

Ik wil de structuur voor de heuristische benadering inleiden door een vergelijking te maken met een probleem dat U waarschijnlijk allemaal veel liever wilt oplossen dan het frequentietoewijzingsprobleem, namelijk het vinden van het gezelligste cafeetje in Maastricht. Voor de niet Maastrichtenaren onder ons is het goed om te vertellen dat dit aantal dermate groot is dat een expliciete aftellingsmethode gedoemd is te mislukken.

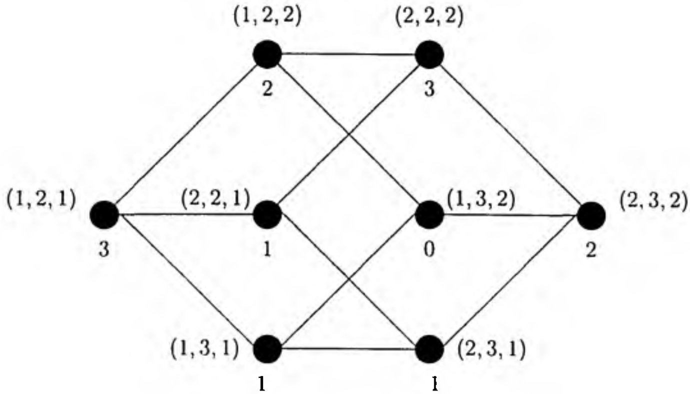
Wat zou nu een redelijke strategie zijn om toch een zo gezellig mogelijk cafeetje te vinden?

Ik stel de volgende strategie voor. U laat U afzetten bij een willekeurig cafeetje ergens in de stad. Vervolgens bezoekt U alle cafeetjes in dezelfde straat, en in alle straten welke verbonden zijn met die straat. Vindt U nergens een beter cafeetje dan is het begincafeetje Uw keuze. Vindt U ergens een beter cafeetje dan herhaalt U bovenstaande procedure vanuit dat cafeetje. Deze procedure wordt herhaald net zo lang totdat U eindigt in een cafeetje, welke geen beter cafeetje in de directe omgeving heeft.

In deze strategie heb ik op de verzameling van alle cafeetjes een structuur opgelegd, welke ik een *buurstructuur* zal noemen. Voor ieder cafeetje is een deelverzameling van alle cafeetjes gedefiniëerd, welke we de buren van dat cafeetje noemen. In ons geval zijn dat alle cafeetjes in dezelfde straat of in aanliggende straten. Het zoeken naar een gezellig cafeetje speelt zich steeds lokaal af, namelijk in de gedefinieerde buurruimte van het tot dan toe gevonden gezelligste cafeetje. Daarom noemen we een dergelijke zoekstrategie een *lokale zoekmethode*. Merk op dat we niet alle cafeetjes van Maastricht hoeven te kennen om deze strategie te kunnen uitvoeren. Het enigste wat we moeten weten zijn alle cafeetjes in de buurruimte van een gegeven cafeetje, en deze kunnen wij eenvoudig achterhalen door de betreffende straten af te lopen.

Merk op dat we de definitie van buren zouden kunnen uitbreiden naar alle cafeetjes in de omringende straatblokken. Het groter maken van de buurruimte heeft tot gevolg dat de kans om een gezelliger cafeetje te vinden toeneemt, maar dit gaat wel gepaard met meer moeite om alle cafeetjes in de gedefinieerde buurt te onderzoeken. De afweging bij het ontwerpen van een buurstructuur is dus enerzijds de kwaliteit van de gevonden oplossing, en anderzijds de moeite die het kost deze oplossing te vinden. Nadeel van deze lokale zoekmethode is dat we in een lokaal optimum terecht komen, namelijk een cafeetje, waar zich geen gezelliger cafeetje in de buurt bevindt. Dit sluit niet uit dat er buiten die buurt een nog gezelliger cafeetje bestaat.

Voor het frequentietoewijzingsprobleem, waarbij de doelstelling is het minimaliseren van de storing, kan men bijvoorbeeld als buren van een gegeven oplossing definieren alle oplossingen die slechts in één frequentie van de gegeven oplossing verschillen. Als we nu een plaatje tekenen waarin we twee oplossingen met elkaar verbinden als het buren zijn, krijgen we onderstaande figuur.



Figuur 4. De buurstructuur.

Starten we in oplossing $(1,2,1)$ met waarde 3 dan vinden we onder de drie buren een betere oplossing, bijvoorbeeld de oplossing $(2,2,1)$ met waarde 1. We zien dat de lokale zoekmethode eindigt in deze oplossing omdat er geen betere buuroplossing is. We zien tevens dat we niet het globale optimum vinden, namelijk de oplossing $(1,3,2)$ met waarde 0, maar een lokaal optimum.

Er zijn vele variaties op deze lokale zoekmethode ontwikkeld met als voornaamste wijziging, ten opzichte van het basisidee, de mogelijkheid uit een lokaal optimum te ontsnappen. Simulated annealing, tabu search, threshold accepting, en genetische algoritmen zijn enkele van deze varianten. Ik ga daar in deze oratie niet nader op in.

4.2 Optimale oplossing.

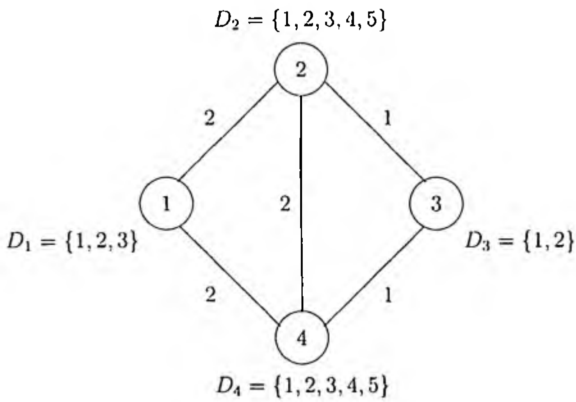
Bij het vinden van een optimale oplossing zien we ons op het eerste gezicht geplaatst voor een onmogelijk taak. Enerzijds kunnen we geen oplossing buiten beschouwing laten, het zou immers wel eens een optimale oplossing kunnen zijn, en anderzijds is het zoals we reeds gezien hebben ondoenlijk alle oplossingen te bekijken.

Bij nader inzien is de taak toch niet zo onmogelijk. We mogen best een oplossing buiten beschouwing laten, mits zich onder de overgebleven oplossingen een even goede dan wel betere oplossing bevindt. Als we dit kunnen garanderen kunnen we ook garanderen dat zich onder de overgebleven oplossingen tenminste één optimale oplossing bevindt.

Een methode voor het vinden van een optimale oplossing voor een discreet optimaliseringsprobleem, welke gebruikt maakt van bovengenoemde observatie, noemen we, in contrast met de expliciete aftellingsmethode, een *impliciete aftellingsmethode*. Naar mate de methode implicieter is, d.w.z., meer oplossingen buiten beschouwing laat, is de methode efficiënter.

Een belangrijke vraag met betrekking tot een impliciete aftellingsmethode is op welke wijze we een garantie kunnen afgeven dat bepaalde oplossingen buiten beschouwing kunnen worden gelaten zonder dat deze oplossingen expliciet worden bekeken.

Ik zal enkele garantiebewijzen illustreren door een impliciet aftellingsmethode te beschreven voor een eenvoudig voorbeeld, namelijk Voorbeeld 2, weergegeven in Figuur 5.



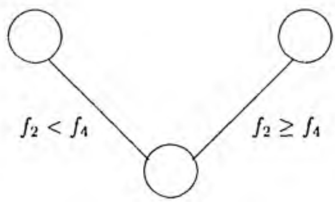
Figuur 5. Probleeminstantie Voorbeeld 2.

Discrete optimalisering is een mengeling van wiskunde en puzzelen. Het is juist deze combinatie, welke mij zo veel plezier in mijn vak bezorgt. Het wiskunde karakter komt in de volgende sectie aan bod. Iets van het puzzel karakter vindt U terug in de nu volgende beschrijving van de impliciete aftellingsmethode. Het volgen van de redenering vereist wel iets meer aandacht dan bij de meeste oraties gebruikelijk is.

Een impliciete aftellingsmethode begint met het aanbrengen van structuur in de oplossingsruimte door de oplossingen te verdelen over twee of meer deelverzamelingen.

Voor Voorbeeld 2 zouden we bijvoorbeeld de oplossingen kunnen verdelen in twee deelverzamelingen, namelijk een deelverzameling waarin zich alle oplossingen bevinden waarin basisstation 2 een kleinere frequentie toegewezen heeft gekregen dan basisstation 4, en een deelverzameling waarin zich alle overige oplossingen bevinden, d.w.z., alle oplossingen waarin basisstation 2 een frequentie toegewezen heeft gekregen, welke groter of gelijk is aan de toegewezen frequentie van basisstation 4. De verdeling staat weergegeven in Figuur 6.

Het probleem van het vinden van de oplossing met kleinste waarde binnen een deelverzameling van alle oplossingen van het oorspronkelijk probleem noemen we een *partieel probleem* van het oorspronkelijk probleem, de deelverzameling is de oplossingsruimte van het partieel probleem, en de beste oplossing binnen de deelverzameling noemen we de optimale oplossing van het partieel probleem. De optimale oplossing van het oorspronkelijk probleem is de beste van alle optimale oplossingen van de partiële problemen. We kunnen dus het oorspronkelijk probleem oplossen door ieder van de partiële problemen op te lossen.



Figuur 6. Partiële problemen Voorbeeld 2.

We hopen uiteraard dat een aantal partiële problemen niet hoeft te worden onderzocht, omdat een garantie kan worden afgegeven dat de corresponderende deelverzamelingen oplossingen buiten beschouwing kunnen worden gelaten.

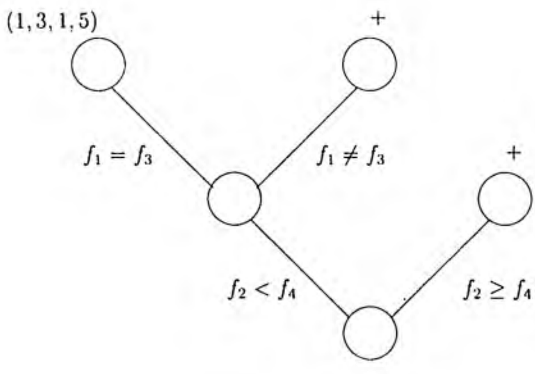
Voor Voorbeeld 2 zullen we zien dat dit het geval is voor het partieel probleem behorende bij de extra restrictie $f_2 \geq f_4$.

De initiële verdeling van de oplossingsruimte is de eerste belangrijke stap in de structurering van de oplossingsruimte. De verdeling moet zodanig gekozen worden dat het de te gebruiken technieken, om oplossingen van beschouwing uit te sluiten, maximaal ondersteunt.

Als alle mogelijkheden om oplossingen buiten beschouwing te laten zijn benut, en het partieel probleem nog steeds niet is opgelost, dan wordt er een fijnere structuur aangebracht in de oplossingsruimte door de oplossingen van het het partieel probleem te verdelen over een aantal deelverzamelingen. Het partieel probleem wordt dan opgelost door de partiële problemen van het partieel probleem zelf op te lossen.

Op deze wijze worden alle oplossingen van het oorspronkelijke discreet optimaliseringsprobleem stapsgewijs verdeeld over steeds kleiner wordende deelverzamelingen. In het slechts denkbare scenario moeten we de verdeling doorzetten totdat iedere deelverzameling uit één oplossing bestaat. In dat geval is de impliciete aftellingsmethode identiek aan de expliciete aftellingsmethode. Gelukkig doet dit slechts denkbare scenario zich in de praktijk zelden voor.

Het totale verdelingsproces voor Voorbeeld 2 staat weergegeven in Figuur 7. Een kruis bij een particel probleem geeft aan dat dit particel probleem buiten beschouwing kon worden gelaten op basis van de nog nader te beschrijven technieken.



Figuur 7. Verdelingsproces Voorbeeld 2.

Laat ik de impliciete aftellingsmethode, geïllustreerd in Figuur 7 nader toelichten. Allereerst een toelichting op het feit dat het particel probleem gedefinieerd door $f_2 \geq f_4$ buiten beschouwing kan worden gelaten.

De storingsrestrictie tussen de basisstations 2 en 4 zorgt er voor dat deze niet dezelfde frequentie toegewezen kunnen krijgen in een toegelaten oplossing. Dit verklaart waarom oplossingen, waarbij de basisstations 2 en 4 dezelfde frequentie hebben, buiten beschouwing kunnen worden gelaten. Afgezien van de onderlinge restrictie, zien we dat de basisstations 2 en 4 volledig identiek zijn, d.w.z ze hebben hetzelfde domein, en dezelfde restricties ten opzichte van de overige basisstations. Dit betekent dat als we een toegelaten oplossing (bijvoorbeeld $(1,5,2,3)$) hebben, we een tweede toegelaten oplossing kunnen vinden door de frequenties van de basisstations 2 en 4 om te wisselen (bijvoorbeeld $(1,5,2,3)$ wordt $(1,3,2,5)$). Bij één van deze oplossingen heeft basisstation 2 een kleinere frequentie dan basisstation 4, bij de andere oplossing heeft basisstation 2 een grotere frequentie dan basisstation 4. Omdat beide oplossingen dezelfde waarde hebben, is het toegestaan een van beide oplossingen buiten beschouwing te laten. Gekozen is om de oplossingen, waarin basisstation 2 een grotere frequentie heeft dan basisstation 4, buiten beschouwing te laten. Merk op dat deze oplossingen buiten beschouwing worden gelaten zonder dat ze expliciet bekeken zijn.

Het gebruikte argument om oplossingen buiten beschouwing te laten is een illustratie van de techniek *dominantie*.

Dominantie heeft betrekking op twee deelverzamelingen oplossingen A en B.

Deelverzameling B *domineert* deelverzameling A indien voor iedere toegelaten oplossing in A er een minstens zo goede toegelaten oplossing is in B.

Als deelverzameling B deelverzameling A domineert, dan kan deelverzameling A buiten beschouwing worden gelaten.

We kunnen ons bij het oplossen dus concentreren op het partieel probleem corresponderend met de extra restrictie $f_2 < f_4$.

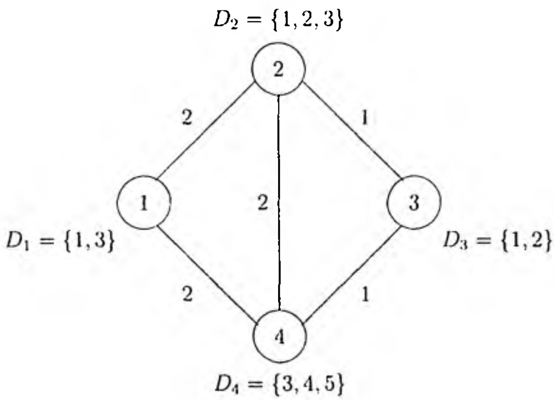
Bekijken we frequentie 4 in het domein van basisstation 2 dan zien we dat er geen enkele frequentie in het domein van basisstation 4 is, welke aan de storingsrestrictie en de restrictie $f_2 < f_4$ voldoet. We concluderen dat er geen enkele toegelaten oplossing is voor het partieel probleem, waarbij basisstation 2 frequentie 4 krijgt toegewezen. We kunnen dus zonder verlies van algemeenheid frequentie 4 uit het domein van basisstation 2 schrappen. Hetzelfde geldt voor frequentie 5 in het domein van basisstation 2. Tevens kunnen we de frequenties 1 en 2 schrappen uit het domein van basisstation 4 omdat er geen enkele frequentie is in het domein van basisstation 2 zodanig dat aan de restrictie $f_2 < f_4$ tussen de basisstations 2 en 4 is voldaan. Het domein van basisstation 2 is dus gereduceerd tot $D_2 = \{1, 2, 3\}$, en het domein van basisstation 4 is gereduceerd tot $D_4 = \{3, 4, 5\}$.
Kijken we nu naar frequentie 2 in het domein van basisstation 1 dan zien we dat er geen frequentie is in het domein van basisstation 2 zodanig dat aan de storingsrestrictie tussen de basisstations 1 en 2 is voldaan. We kunnen dus frequentie 2 uit het domein van basisstation 1 verwijderen. Merk op dat de laatste verwijdering pas mogelijk werd nadat eerst frequenties 4 en 5 uit het domein van basisstation 2 waren verwijderd.

Het gebruikte argument om frequenties uit een domein van een basisstation te verwijderen, omdat deze frequenties in geen enkele toegelaten oplossing kunnen voorkomen, is een illustratie van de techniek *preprocessing*.

Preprocessing is een verzamelnaam voor allerlei technieken, welke toegepast worden om de probleeminstantie te reduceren.

De in het voorbeeld gebruikte vorm van preprocessing staat bekend als arc consistency [6], en is afkomstig uit de informatica.

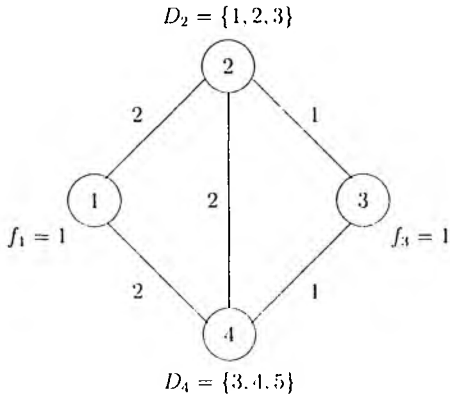
Het partieel probleem na preprocessing wordt gegeven in Figuur 8.



Figuur 8. Partieel probleem Voorbeeld 2 na preprocessing.

Concentreren we ons verder op het partieel probleem in Figuur 8. Omdat het partieel probleem niet verder reduceerbaar is en ook nog niet is opgelost, wordt de oplossingsruimte van het partieel probleem verder gestructureerd. We verdelen de oplossingen in twee deelverzamelingen, namelijk een deelverzameling oplossingen, waarin basisstations 1 en 3 dezelfde frequentie krijgen toegewezen, en een deelverzameling oplossingen, waarin basisstations 1 en 3 verschillende frequenties krijgen toegewezen. Op deze wijze ontstaan twee nieuwe partiële problemen, welke onderzocht moeten worden.

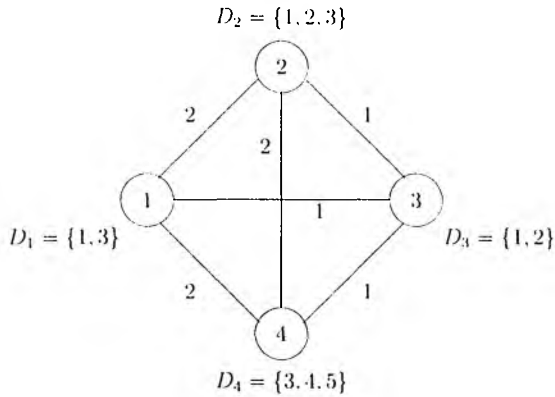
We onderzoeken allereerst het partieel probleem corresponderend met de extra restrictie $f_1 = f_3$. Omdat de domeinen van de basisstations 1 en 3 slechts frequentie 1 gemeenschappelijk hebben, krijgen beide basisstations frequentie 1 toegewezen. Het particel probleem staat weergegeven in Figuur 9.



Figuur 9. Particel probleem Voorbeeld 2 bij $f_1 = f_3 = 1$.

Kijken we naar de storingsrestrictie tussen de basisstations 1 en 2, dan zien we dat alleen frequentie 3 in het domein van basisstation 2 daaraan voldoet. De frequenties 1 en 2 kunnen dus uit het domein verwijderd worden. Als basisstation 2 frequentie 3 krijgt toegewezen kan basisstation 4 alleen frequentie 5 toegewezen krijgen. We vinden dus via preprocessing dat er slechts één toegelaten oplossing is in het particel probleem, en wel de oplossing $(1, 3, 1, 5)$ met waarde 3.

Kijken we nu naar het tweede particel probleem dat correspondeert met de extra restrictie $f_1 \neq f_3$. De extra restrictie komt overeen met een storingsrestrictie met afstand 1. Het particel probleem staat weergegeven in Figuur 10.



Figuur 10. Particel probleem Voorbeeld 2 bij $f_1 \neq f_3$.

Voor ieder tweetal basisstations is er een storingsrestrictie, welke zegt dat de betreffende basisstations een verschillende frequentie moeten krijgen. Iedere toegelaten oplossing heeft dus minimaal 4 verschillende frequenties nodig. We hebben dus een ondergrens van 4 gevonden voor de optimale waarde van het particel probleem zonder dat we de oplossingen van het particel probleem expliciet hebben bekeken. Aangezien deze ondergrens van 4 groter is dan de best bekende oplossingswaarde van 3 kunnen we dit particel probleem buiten beschouwing laten.

Het gebruikte argument illustreert de techniek *begrenzing*.

Voor het toepassen van begrenzing op een deelverzameling oplossingen zijn twee waarden nodig. Enerzijds een ondergrens op de beste waarde van een oplossing binnen de deelverzameling, en anderzijds de beste waarde van alle tot dan toe bekende oplossingen.

Als de ondergrens groter of gelijk is aan de best bekende oplossingswaarde, dan kunnen we concluderen dat de betrokken deelverzameling oplossingen buiten beschouwing kan worden gelaten. De reden waarom we dat kunnen concluderen is weergegeven in Figuur 11. Vanwege de definitie van de ondergrens weten we dat alle oplossingen in de deelverzameling een waarde hebben die groter of gelijk is aan de ondergrens. De ondergrens zelf is weer groter of gelijk aan de beste bekende oplossingswaarde. We concluderen dat geen enkele oplossing in de deelverzameling beter is dan de tot dan toe best bekende oplossing.

Waarde

| → Beste oplossingswaarde deelverzameling
| → Ondergrens deelverzameling
| → Waarde beste bekende oplossing

Figuur 11. Begrenzing.

Indien de ondergrens kleiner is dan de best bekende oplossingswaarde kunnen we geen garantie afgeven dat de betrokken deelverzameling buiten beschouwing kan worden gelaten. Er kunnen zich in dat geval twee situaties voordoen. In de ene situatie is de onbekende beste waarde van een oplossing binnen de deelverzameling kleiner dan de waarde van de best bekende oplossing (zie Figuur 12), en in dat geval geven we terecht geen garantie af.

Waarde

| → Waarde best bekende oplossing
| → Beste oplossingswaarde deelverzameling
| → Ondergrens deelverzameling

Figuur 12. Terecht geen garantie.

In de andere situatie is de onbekende beste oplossingswaarde in de deelverzameling groter of gelijk aan de best bekende oplossing (zie Figuur 13), en in dat geval wordt ten onrechte geen garantie afgegeven.

Waarde

| → Beste oplossingswaarde deelverzameling
| → Waarde best bekende oplossing
| → Ondergrens deelverzameling

Figuur 13. Ten onrechte geen garantie.

De situatie in Figuur 13 toont aan dat het zaak is een ondergrens te berekenen die zo dicht mogelijk bij de beste waarde van de deelverzameling in de buurt komt. Dit om situaties waarin we ten onrechte geen garantie afgeven te vermijden. Over het algemeen geldt echter dat des te beter de ondergrens des te meer tijd het kost om deze te berekenen. Deze extra tijd moet altijd afgewogen worden tegen de tijdswinst die met een betere ondergrens wordt geboekt, namelijk de besparing in het zoeken naar oplossingen binnen de deelverzameling, welke anders buiten beschouwing zou zijn gelaten.

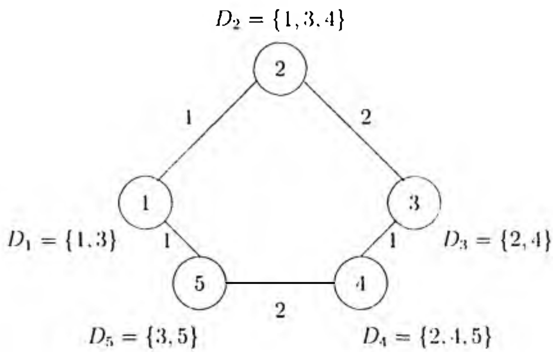
Merk op dat de volgorde, waarin de partiële problemen worden beschouwd, de toepassing van de techniek begrenzing beinvloedt. Hadden we in het voorbeeld het partieel probleem corresponderend met $f_1 \neq f_3$ beschouwd voor het partieel probleem corresponderende met $f_1 = f_3$, dan was het niet mogelijk geweest het partieel probleem corresponderend met $f_1 \neq f_3$ uit te sluiten op basis van de techniek begrenzing, omdat op dat moment nog geen enkele toegelaten oplossing bekend is.

Omdat van de drie besproken technieken om oplossingen te achterhalen, welke buiten beschouwing kunnen worden gelaten, de techniek begrenzing de belangrijkste is, ga ik in de volgende sectie nader in op deze techniek.

5 De techniek begrenzing.

In deze sectie ga ik uit van een wiskundig geschoolde lezer, welke vertrouwd is met wiskundige modellen en met grafentheorie.

Aan de hand van een voorbeeld, namelijk Voorbeeld 3, weergegeven in Figuur 14, bespreek ik een serie toenemende ondergrenzen.



Figuur 14. Probleeminstantie Voorbeeld 3.

We beginnen met een combinatorische ondergrens, welke we ook reeds in de vorige sectie hebben toegepast. Voordat we deze ondergrens formeel kunnen definiëren allereerst enkele definities.

De *restrictie graaf* corresponderende met een instantie van het frequentietoewijzingsprobleem heeft een punt voor ieder basisstation, en een lijn voor ieder tweetal basisstations waartussen een storingsrestrictie bestaat.
Een *kliek* in een graaf is een deelverzameling punten met de eigenschap dat tussen ieder tweetal punten in de kliek een lijn bestaat.
Een *maximum kliek* is een kliek met maximum cardinaliteit.
Een kliek in de restrictie graaf correspondeert met een deelverzameling basisstations met de eigenschap dat tussen ieder tweetal basisstations in de kliek een storingsrestrictie bestaat. In iedere toegelaten oplossingen is het aantal verschillende frequenties toegewezen aan een kliek gelijk aan het aantal basisstations in de kliek. De cardinaliteit van een maximum kliek is dus een ondergrens voor het aantal verschillende frequenties dat in iedere toegelaten oplossing voorkomt. We noemen deze ondergrens de *kliekondergrens*.
Voor Voorbeeld 3 is de kliekondergrens gelijk aan 2.

Elk van de overige ondergrenzen is gebaseerd op een relaxatie van het op te lossen probleem.

Probleem B is een *relaxatie* van probleem A als voldaan is aan de volgende twee eisen.

1) Iedere toegelaten oplossing van A is ook toegelaten in B.

2) Iedere toegelaten oplossing van A heeft in probleem B een waarde die kleiner of gelijk is aan de waarde die deze oplossing heeft in probleem A

Merk op dat de waarde van een oplossing dus kan verschillen in de problemen A en B. De twee eisen gezamenlijk garanderen dat de optimale oplossingswaarde van probleem B kleiner of gelijk is aan de optimale waarde van probleem A. Dit is eenvoudig in te zien als men bedenkt dat de optimale oplossing van probleem A tevens toegelaten is in probleem B (eis 1), en daar een kleiner of gelijke waarde heeft (eis 2). De optimale waarde van de relaxatie B kan dus dienen als ondergrens voor de optimale waarde van A.

Een manier om een relaxatie van een probleem te verkrijgen is om de waarde van een oplossing onveranderd te laten (aan eis 2 is dan automatisch voldaan), en een aantal restricties te versoepelen (aan eis 1 is dan voldaan). Voorbeelden voor het frequentietoewijzingsprobleem zijn het verlagen van de bandbreedte afstand bij een lijn, en het toevoegen van frequenties aan een domein.

Als tweede ondergrens nemen we het kleuringsgetal van de restrictie graaf. Het *kleuringsgetal* van een graaf is het minimaal aantal kleuren dat nodig is om ieder punt van de graaf een kleur te geven zodanig dat twee punten, welke verbonden zijn door een lijn, een verschillende kleur hebben.

Het is eenvoudig in te zien dat het kleuringsprobleem een relaxatie is van het frequentietoewijzingsprobleem. Iedere toegelaten oplossing correspondeert met een toegelaten kleuring door voor iedere gebruikte frequentie een kleur te reserveren. Twee punten, welke verbonden zijn door een lijn, hebben een verschillende kleur, omdat zij in de toegelaten oplossing een verschillende frequentie hebben vanwege de onderlinge storingsrestrictie. Aan eis 1 van een relaxatie is dus voldaan.
Aan eis 2 is voldaan, omdat een toegelaten oplossing in beide problemen dezelfde waarde heeft.
Het is niet moeilijk in te zien dat het kleuringsgetal van de restrictie graaf corresponderende met Voorbeeld 3 gelijk is aan 3.

De derde ondergrens is gebaseerd op de relaxatie van het frequentietoewijzingsprobleem, waarbij als er een storingrestrictie is de bij deze restrictie behorende bandbreedte afstand gelijk wordt genomen aan 1. Het verschil met het kleuringsprobleem is dus dat in deze relaxatie de domeinen gelijk zijn aan die in het oorspronkelijke probleem. De bewering is dat we minimaal vier frequenties nodig hebben. Dit kan als volgt worden ingezien. De basisstations 1, 3, en 4 hebben alle een verschillende frequentie in iedere toegelaten oplossing. Voor basisstation 3 en 4 geldt dit omdat ze verbonden zijn door een storingrestrictie, voor basisstations 1 en 3 geldt dit omdat de domeinen geen gemeenschappelijke frequentie bevatten, en voor basisstations 1 en 4 geldt dit ook omdat de domeinen geen gemeenschappelijke frequenties bevatten. Als er een oplossing is met drie frequenties moet basisstation 5, welke verbonden is met basisstation 1 en basisstation 4, een frequentie hebben, welke verschilt van die van deze beide basisstations. Basisstation 5 moet dus de overgebleven frequentie hebben, en zoals we zagen heeft ook basisstation 3 deze frequentie. Dit is echter onmogelijk omdat de domeinen van de basisstations 3 en 5 geen gemeenschappelijke frequenties hebben. Iedere toegelaten oplossing heeft dus minimaal vier frequenties nodig. Een toegelaten oplossing voor de relaxatie met waarde 4, en dus een optimale oplossing, is (1,3,2,4,3).

De laatste twee ondergrenzen corresponderen met relaxaties gebaseerd op (0,1)-programmeringsformuleringen van het frequentietoewijzingsprobleem. We duiden deze relaxaties aan met respectievelijk de snede methode en kolomgeneratie.

We beschrijven eerst de snede methode.

Een toegelaten oplossing van het frequentietoewijzingsprobleem kunnen we weergeven via een (0,1)-vector, waarbij een coördinaat is gereserveerd voor iedere combinatie van frequentie en basisstation (een 1 geeft aan dat het basisstation de frequentie krijgt toegewezen, een 0 geeft aan dat het basisstation de frequentie niet krijgt toegewezen), en tevens een coördinaat is gereserveerd voor iedere mogelijke frequentie (een 1 geeft aan dat deze frequentie wordt gebruikt, een 0 geeft aan dat deze frequentie niet wordt gebruikt).
We gebruiken de volgende notatie om de coördinaten van de (0,1)-vectoren, welke de toegelaten oplossingen representeren, weer te geven.

$$x(i,f) = \begin{cases} 1 & \text{als frequentie } f \text{ aan basis station } i \text{ wordt toegewezen,} \\ 0 & \text{anders} \end{cases}$$

$$y(f) = \begin{cases} 1 & \text{als frequentie } f \text{ wordt gebruikt,} \\ 0 & \text{anders} \end{cases}$$

Een bekend resultaat uit de polyhedrale theorie (zie stelling van Farkas, Minkowski en Weyl in hoofdstuk 7 van het boek van Schrijver[7]) zegt dat het convex omhulsel van alle (0,1)-vectoren, welke corresponderen met toegelaten oplossingen, kan worden beschreven door een stelsel lineaire ongelijkheden. Gegeven dit stelsel lineaire ongelijkheden kan het frequentietoewijzingsprobleem geschreven worden als een lineair programmeringsprobleem, en opgelost worden met een van de efficiënte oplosmethoden voor lineaire programmeringsproblemen. De optimale oplossing van het lineaire programmeringsprobleem is tevens de optimale oplossing van het frequentietoewijzingsprobleem. Het probleem bij deze aanpak ligt in het vinden van de lineaire ongelijkheden, welke het convex omhulsel beschrijven. In de praktijk zijn we slechts in staat een deelverzameling van deze ongelijkheden te beschrijven. We lossen in de praktijk dan ook een relaxatie op van het probleem door ons te beperken tot een deelverzameling van alle lineaire ongelijkheden. In de gangbare oplossingswijze beginnen we met enkele ongelijkheden, waaraan de toegelaten oplossingen moeten voldoen. Vervolgens wordt het lineaire programmeringsprobleem met deze ongelijkheden opgelost. Voldoet de optimale lineaire programmeringsoplossing ook aan alle overige bekende ongelijkheden dan stopt de procedure. Voldoet de optimale lineaire programmeringsoplossing niet aan alle overige bekende ongelijkheden dan worden de ongelijkheden, welke geschonden zijn alsnog toegevoegd en de procedure herhaalt zich. Deze methode wordt de *snede methode* genoemd omdat de toegevoegde ongelijkheden de oude lineaire programmeringsoplossing afsnijden van het resterende toegelaten gebied. De reden om niet alle bekende ongelijkheden in het begin al toe te voegen ligt in het enorme aantal ongelijkheden, en het feit dat slechts enkele van deze ongelijkheden bindend zijn voor de optimale lineaire programmeringsoplossing, aan de overige ongelijkheden wordt door de optimale lineaire programmeringsoplossing altijd voldaan. Vanwege de efficiëntie van de lineaire programmeringsoplossingstechniek wordt daarom het aantal toegevoegde ongelijkheden zo beperkt mogelijk gehouden.

De snede methode is toegepast op het frequentietoewijzingsprobleem door Van Hoesel en andere [8]. Ik illustreer hun aanpak voor Voorbeeld 3.

De doelstelling is het minimaliseren van het aantal gebruikte verschillende frequenties, en kan nu geformuleerd worden als

$$\text{minimaliseer } \sum_f y(f)$$

In de restricties moeten we vastleggen dat ieder basisstation één frequentie uit zijn domein krijgt toegewezen. Dit wordt afgedwongen door de volgende restricties

(1) $\sum_{f \in D_i} x(i,f) = 1$, voor alle i.

We moeten een relatie leggen tussen het toewijzen van een frequentie aan een basisstation en het gebruik van deze frequentie. Deze relatie wordt vastgelegd in de volgende restricties.

(2) $x(i,f) \leq y(f)$, voor alle i, en voor alle $f \in D_i$.

Als frequentie f wordt gebruikt is er een basisstation i met $f \in D_i$ waarvoor $x(i,f) = 1$. De restrictie dwingt dan af dat $x(i,f) = 1 \leq y(f)$, m.a.w, $y(f) = 1$.

De storingsrestricties worden gemodeleerd door

(3) $x(i,f) + x(j,g) \leq 1$, voor alle $\{i,j\}$, en voor alle $f \in D_i, g \in D_j$ met $|f - g| < d(i,j)$,

waarbij $d(i,j)$ de bandbreedte afstand is behorende bij de storingsrestrictie tussen basisstations i en j.
Als de frequenties f voor basisstation i, en g voor basisstation j niet aan de storingsrestrictie voldoen, dan kan maximaal één van deze frequentietoewijzingskeuzes gemaakt worden.

De eis dat iedere variable een niet-negatieve waarde heeft, d.w.z.,

$$\begin{aligned} x(i,f) &\geq 0, \quad \text{voor alle } i, \quad \text{en voor alle } f \in D_i \\ y(f) &\geq 0, \quad \text{voor alle } f, \end{aligned}$$

complementeert de initiële lineaire programmeringsformulering.

De optimale lineaire programmeringsoplossing wordt gegeven door

$$
\begin{aligned}
&y(1) = y(2) = y(3) = y(4) = y(5) = 0.5,\\
&x(1,1) = x(1,3) = 0.5,\\
&x(2,1) = x(2,3) = 0.5,\\
&x(3,2) = x(3,4) = 0.5,\\
&x(4,2) = x(4,4) = 0.5,\\
&x(5,3) = x(5,5) = 0.5.
\end{aligned}
$$

De relaxatie heeft waarde 2.5.

Een ongelijkheid, welke geschonden wordt door de huidige optimale lineaire programmeringsoplossing is de ongelijkheid

$$x(1,1) + x(2,1) \leq y(1).$$

Deze ongelijkheid zegt dat indien frequentie 1 niet wordt gebruikt, d.w.z., $y(1) = 0$, frequentie 1 niet kan worden toegewezen aan basisstations 1 en 2. De ongelijkheid zegt dat indien frequentie 1 wel wordt gebruikt, d.w.z., $y(1) = 1$, frequentie 1 slechts aan één van beide basisstations 1 en 2 kan worden toegewezen, dit omdat er een storingsrestrictie bestaat tussen deze twee basisstations.
De klasse van ongelijkheden, waartoe deze ongelijkheid behoort, kan beschreven worden door

$$\sum_{i \in C} x(i,f) \leq y(f), \text{ voor alle f, en alle klicks } C \text{ in de graaf } G(f),$$

De graaf $G(f)$ is een graaf met als punten alle basisstations, waarbij f in het domein zit, en als lijnen alle paren, welke niet dezelfde frequentie toegewezen kunnen krijgen. De restrictie zegt dat als frequentie f wordt gebruikt, deze frequentie aan slechts één basisstation in een klick in $G(f)$ kan worden toegewezen.
Bovengenoemde klasse van ongelijkheden bevat in het voorbeeld de volgende 7 ongelijkheden.

$$
\begin{array}{ccccc}
x(1,1) & + & x(2,1) & \leq & y(1)\\
x(1,3) & + & x(2,3) & \leq & y(3)\\
x(2,4) & + & x(3,4) & \leq & y(4)\\
x(3,2) & + & x(4,2) & \leq & y(2)\\
x(3,4) & + & x(4,4) & \leq & y(4)\\
x(4,5) & + & x(5,5) & \leq & y(5)\\
x(1,3) & + & x(5,3) & \leq & y(3)
\end{array}
$$

Vanwege het geringe aantal ongelijkheden, voegen we ze allemaal toe, en niet slechts diegene welke geschonden zijn door de optimale lineaire programmeringsoplossing. De optimale lineaire programmeringsoplossing van het nieuwe model is

$$
\begin{aligned}
&y(1) = y(2) = y(4) = 1, \; y(3) = y(5) = 0.5\\
&x(1,1) = 1\\
&x(2,3) = x(2,4) = 0.5,\\
&x(3,2) = x(3,4) = 0.5,\\
&x(4,2) = x(4,4) = 0.5,\\
&x(5,3) = x(5,5) = 0.5.
\end{aligned}
$$

De relaxatie heeft waarde 4.

Een ongelijkheid, welke geschonden wordt door de huidige optimale lineaire programmeringsoplossing is

$$x(2,3) + x(2,4) + x(3,4) \leq 1.$$

Deze ongelijkheid zegt dat als aan basisstation 3 frequentie 4 wordt toegewezen ($x(3,4) = 1$) men niet frequentie 3 of 4 aan basisstation 2 kan toewijzen, omdat deze frequenties niet afstand minimaal 2 hebben tot frequentie 4, welke aan basisstation 3 is toegewezen. Deze ongelijkheid zegt verder dat als aan basisstation 3 frequentie 4 niet wordt toegewezen ($x(3,4) = 0$) men maximaal één van de frequenties 3 en 4 aan basisstation 2 kan toewijzen. Dit komt overeen met de eis dat slechts één frequentie uit het domein kan worden gekozen.
De klasse van ongelijkheden, waartoe deze ongelijkheid behoort, kan beschreven worden door

$$\sum_{f \in F} x(i,f) + x(j,g) \leq 1,$$

voor alle lijnen $\{i,j\}$, voor alle g, $g \in D_j$, en F, $F \subseteq D_i$, gedefinieerd door $F = \{f \in D_i \mid |f - g| < d(i,j)\}$.
Bovengenoemde klasse ongelijkheden bevat voor het voorbeeld de volgende 4 ongelijkheden

$$
\begin{array}{ccccccc}
x(2,3) & + & x(2,4) & + & x(3,4) & \leq & 1,\\
x(3,2) & + & x(3,4) & + & x(2,3) & \leq & 1,\\
x(4,4) & + & x(4,5) & + & x(5,5) & \leq & 1,\\
x(5,3) & + & x(5,5) & + & x(4,4) & \leq & 1.
\end{array}
$$

Na toevoeging van alle ongelijkheden in deze klasse, vinden we de volgende optimale lineaire programmeringsoplossing

$$
\begin{aligned}
&y(1) = y(2) = y(3) = y(4) = y(5) = 1,\\
&x(1,1) = 1,\\
&x(2,4) = 1,\\
&x(3,2) = 1,\\
&x(4,5) = 1,\\
&x(5,3) = 1.
\end{aligned}
$$

De relaxatie heeft waarde 5.

Aangezien de optimale oplossing van de lineaire programmeringsrelaxatie toegelaten is in het (0,1)-programmeringsprobleem, is deze oplossing optimaal voor het (0,1)-programmeringsprobleem.

De laatste relaxatie is kolomgeneratie.

De kolomgeneratie techniek is gebaseerd op een (0,1)-programmeringsformulering van het probleem. De relaxatie van het model is de standaard lineaire programmeringsrelaxatie, waarbij de eis dat de variabelen de waarde 0 of 1 kunnen aannemen wordt vervangen door de eis dat de waarde van een variabele tussen 0 en 1 moet liggen. De kunst is nu een (0,1)-programmeringsformulering te vinden voor het probleem, waarvan de optimale waarde van de lineaire programmeringrelaxatie dicht ligt bij de optimale waarde van het (0,1)-programmeringsprobleem. Ervaring heeft geleerd dat dit vaak mogelijk is bij (0,1)-programmeringsformuleringen met een zeer groot aantal variabelen. Een lineair programmeringsprobleem met een groot aantal variabelen kan worden opgelost met de zogenaamde *kolomgeneratie techniek*. In deze techniek wordt in het begin een lineair programmeringsprobleem opgelost, waarbij een aantal variabelen uit het probleem wordt weggelaten. Vervolgens is het mogelijk na te gaan of er variabelen zijn, welke ten onrechte zijn weggelaten. Een variabele is ten onrechte weggelaten als het lineaire programmeringsprobleem na toevoeging van deze variabele een betere optimale waarde heeft dan het probleem voor toevoeging. Een lineair programmeringsprobleem wordt vaak in matrixvorm geschreven, waarbij de rijen corresponderen met de restricties en de kolommen met de variabelen. Daarom wordt het vinden van een variabele, welke ten onrechte is weggelaten uit het probleem, het *kolomgeneratie probleem* genoemd. Zijn er variabelen, welke ten onrechte zijn weggelaten, dan worden deze alsnog toegevoegd. Dit herhaalt zich totdat geen enkele variabele ten onrechte is weggelaten, en in dat geval is het lineaire programmeringsprobleem optimaal opgelost. Voor meer informatie over de theoretische en praktische achtergronden van kolomgeneratie wordt verwezen naar Savelsbergh [9].
Voor de lineaire programmeringsrelaxatie van een discreet optimaliseringsprobleem is het kolomgeneratie probleem vaak zelf weer een discreet optimaliseringsprobleem. Omdat het kolomgeneratie probleem een groot aantal keren moet worden opgelost, kan de techniek alleen zinvol worden toegepast indien het kolomgeneratie probleem efficiënter oplosbaar is in de praktijk dan het oorspronkelijke discreet optimaliseringsprobleem.

We illustreren het een en ander aan de hand van Voorbeeld 3.

Als we een toegelaten frequentietoewijzing beschouwen dan vormt de verzameling van alle basisstations, welke dezelfde gegeven frequentie hebben toegewezen gekregen, een onafhankelijke verzameling in de restrictie graaf. Een *onafhankelijke verzameling* in een graaf is een deelverzameling punten met de eigenschap dat geen enkel

twee punten verbonden is door een lijn.
Omdat ieder basisstation precies één frequentie krijgt toegewezen in een toegelaten frequentietoewijzing, correspondeert dus een toegelaten frequentietoewijzing met een partitie van de verzameling basisstations in een aantal onafhankelijke verzamelingen.
In onze (0,1)-programmeringsformulering van het frequentie toewijzingsprobleem associëren we dan ook een variabele met iedere onafhankelijke verzameling in de restrictie graaf met de eigenschap dat aan ieder basisstation in deze onafhankelijke verzameling dezelfde frequentie kan worden toegewezen.
We gebruiken de notatie I_f voor de verzameling van alle onafhankelijke verzamelingen waaraan frequentie f kan worden toegewezen.
We introduceren de volgende (0,1)-variabelen

$$x(f,S) = \begin{cases} 1 & \text{als frequentie } f \text{ wordt toegewezen aan onafhankelijke} \\ & \text{verzameling } S,\ S \in I_f \\ 0 & \text{anders.} \end{cases}$$

De partitie bestaat uit maximaal één onafhankelijke verzameling per frequentie. Dit kan geformuleerd worden als

(1) $\quad \sum_{S \in I_f} x(f,S) \leq 1$, voor alle f.

Iedere basisstation moet in precies één onafhankelijke verzameling voorkomen. Dit kan geformuleerd worden als

(2) $\quad \sum_{f \in D_i} \sum_{S \in I_f : i \in S} x(f,S) = 1$, voor alle i.

Restricties (1) en (2) modelleren het feit dat we een partitie hebben van de basisstations in onafhankelijke verzamelingen. Echter niet iedere partitie van de basisstations in onafhankelijke verzamelingen vormt een toegelaten frequentietoewijzing. We moeten immers de storingsrestricties nog modelleren. De storingsrestrictie tussen basisstations i en j zegt dat uit iedere combinatie van frequenties f, $f \in D_i$ en g, $g \in D_j$ met $| f - g | < d(i,j)$, er maximaal één gekozen kan worden.
Dit kan gedefinieerd worden als

(3) $\quad \sum_{S \in I_f : i \in S} x(f,S) + \sum_{S \in I_g : j \in S} x(g,S) \leq 1,$

voor alle lijnen $\{i,j\}$ in de restrictie graaf, en alle $f \in D_i, g \in D_j$ zodanig dat $| f - g | < d(i,j)$.
De doelstelling is het minimaliseren van het aantal frequenties. Omdat, vanwege restricties (1), maximaal één onafhankelijke verzameling per frequentie gekozen kan worden, correspondeert de doelstelling met het minimaliseren van het aantal onafhankelijke verzamelingen in de partitie.
Dit kan gemodelleerd worden als

Minimaliseer $\sum_f \sum_{S \in I_f} x(f,S)$.

Passen we deze formulering toe op Voorbeeld 3.
We hebben de volgende 14 variabelen $x(1,\{1\})$, $x(1\{2\})$, $x(2,\{3\})$, $x(2,\{4\})$, $x(3,\{1\})$, $x(3,\{2\})$, $x(3,\{5\})$, $x(3,\{2,5\})$, $x(4,\{2\})$, $x(4,\{3\})$, $x(4,\{4\})$, $x(4,\{2,4\})$, $x(5,\{4\})$ en $x(5,\{5\})$.

Restrictie (1) voor frequentie 3 luidt

$$x(3,\{1\}) + x(3,\{2\}) + x(3,\{5\}) + x(3,\{2,5\}) \leq 1.$$

Restrictie (2) voor basisstation 4 luidt

$$x(2,\{4\}) + x(4,\{4\}) + x(4,\{2,4\}) + x(5,\{4\}) = 1.$$

Restrictie (3) voor frequentie 2 bij basisstation 2, en frequentie 4 bij basisstation 3 luidt

$$x(3,\{2\}) + x(3,\{2,5\}) + x(4,\{3\}) \leq 1.$$

De optimale oplossing van de lineaire programmeringsrelaxatie wordt gegeven door

$$\begin{aligned}
&x(1,\{1\}) = 1\\
&x(2,\{3\}) = x(2,\{4\}) = \tfrac{1}{2}\\
&x(3,\{2,5\}) = \tfrac{1}{2}\\
&x(4,\{2,4\}) = x(4,\{3\}) = \tfrac{1}{2}\\
&x(5,\{5\}) = \tfrac{1}{2}
\end{aligned}$$

De waarde van de relaxatie is 4.

Kijken we nader naar restricties (3) dan valt op dat we deze restricties kunnen versterken. Voor een gegeven lijn $\{i,j\}$ in de restrictie graaf, en een gegeven frequentie $f \in D_i$ kunnen we de deelverzameling F_{ij}^f van D_j definiëren als alle frequenties welke niet samen met f aan de restrictie voldoen, m.a.w.

$$F_{ij}^f = \{g \in D_j \mid \mid f - g \mid < d(i,j)\}.$$

Restricties (3) kunnen dan vervangen worden door

$$(4) \qquad \sum_{S \in I_f : i \in S} x(f,S) + \sum_{g \in F_{ij}^f} \sum_{S \in I_g : j \in S} x(g,S) \leq 1.$$

De optimale oplossing van de lineaire programmeringsrelaxatie van deze nieuwe formulering wordt gegeven door

$x(1, \{1\}) = 1$
$x(2, \{3\}) = 1$
$x(3, \{5\}) = 1$
$x(4, \{2\}) = 1$
$x(5, \{4\}) = 1$

De relaxatie heeft waarde 5.

Aangezien de optimale lineaire programeeringsformulering toegelaten is in het (0,1)-programmeringsprobleem is dit de optimale oplossing.
Om de waarde van de relaxatie te berekenen zijn alle variabelen in het model opgenomen. Voor grotere probleeminstanties is het onmogelijk om alle variabelen mee te nemen, omdat het aantal variabelen eenvoudig te groot is. We moeten dan de eerder beschreven kolomgeneratie techniek toepassen. In deze oratie ga ik niet nader in op het kolomgeneratie probleem.

6 Dankbetuiging.

Het einde van een oratie is een mooi moment even terug te blikken op de afgelopen jaren. Ik zie af van een volledig overzicht en volsta met enkele opmerkingen.

Voor het feit dat het Nederlandse onderzoek op het gebied van de discrete optimalisering internationaal zo goed staat aangeschreven is Jan-Karel Lenstra op de allereerste plaats verantwoordelijk. Persoonlijk ben ik hem ook dank verontschuldigd, omdat mijn eerste kennismaking met de discrete optimalisering plaats vond op het Mathematisch Centrum in Amsterdam, waar hij mij als promovendus had aangenomen.

De afgelopen periode was ondenkbaar geweest zonder Mirjam. Al meer dan twintig jaren is zij een stimulerende kracht.

Tenslotte wil ik niet de rol vergeten, welke mijn twee zoontjes Niels en Martijn in mijn leven spelen. Ik hoop dat zij nog lang mijn aandacht blijven opvragen.

Ik heb gezegd.

7 Verwijzingen.

[1] De 'Operations Research Society of America' geeft de volgende definitie van Operations Research. "Operations research is concerned with scientifically deciding how to best design and operate man-machine systems, usually under conditions requiring the allocation of scarce resources".

[2] De 'Operations Research Society of Great Britain' geeft de volgende definitie van Operational Research. "Operational research is the application of the methods of science to complex problems arising in the direction and management of large systems of men, machines, materials and money in industry, business, government, and defence. The distinctive approach is to develop a scientific model of the system, incorporating measurements of factors such as chance and risk, with which to predict and compare the outcomes of alternative decisions, strategies or controls. The purpose is to help management determine its policy and actions scientifically".

[3] De diësrede van prof. dr. D. van Dantzig heeft als titel "Reeckeningh in spelen van geluck" en is nog steed verkrijgbaar bij de Universiteit van Amsterdam.

[4] Aan het CALMA (Combinatorial Algorithms for Military Applications) project werd deelgenomen door onderzoekteams uit de landen Engeland, Frankrijk, en Nederland. Het betreft een studie waarin een aantal oplosmethoden voor discrete optimaliseringsproblemen wordt vergeleken met betrekking tot het frequentietoewijzingsprobleem.

[5] Het aantal oplossingen is ongeveer 40^{200}.

$40^{200} = 2^{400} * 10^{200} = (2^{10})^{40} * 10^{200}$.

$2^{10} = 1024$ is ongeveer 10^3.

Dus $40^{200} \approx (10^3)^{40} * 10^{200} = 10^{320}$.

Het aantal door te rekenen oplossingen is gelijk aan het aantal seconden in 30 jaren vermenigvuldigt met 10^9, zijnde het aantal oplossingen dat duizend computers per seconde kunnen doorrekenen.

Het aantal seconden in 30 jaren is gelijk aan het aantal seconden per dag (86400) vermenigvuldigt met het aantal dagen in 30 jaren (ongeveer 10950), en is ongeveer gelijk aan $10^5 * 10^4 = 10^9$.

Het aantal door te rekenen oplossing is dus ongeveer gelijk aan 10^{18}.

[6] Meer informatie over de geschiedenis van arc-consistency en corresponderende algoritmen kan gevonden worden in het artikel "Arc-consistency and arc consistency again" van C. Bessière, Artificial Intelligence 65 (1994), 179-190.

[7] Zie voor meer informatie over polyhedrale theorie het boek "Theory of linear and integer programming" van A. Schrijver, Wiley (1986).

[8] Deze bijdrage aan het CALMA-project staat beschreven in het artikel "A Branch & Cut algorithm for the frequency assignment problem" van Aardal, K., B. Jansen, A. Hipolito & S. van Hoesel.

[9] Zie het artikel van "Branch-and Price: Column Generation for Solving Huge Integer Programs" van C. Barnhart, E.L. Johnson, G.L. Nemhauser, M.W.P. Savelsbergh & P.H. Vance in het boek "Mathematical Programming: State of the Art 1994" van J.R. Birge & K.G. Morty (The University of Michigan).