

Over versnijden, volgorde's en veilingen: combinatorisch optimaliseren in de praktijk

door F.C.R. SPIEKSMa



Frits C.R. Spieksma
KULeuven, Departement
Economische Wetenschappen,
Leuven.

ABSTRACT

Combinatorische optimaliseringsproblemen zijn problemen waarbij uit een eindig (doch groot) aantal mogelijke oplossingen een beste oplossing gevonden dient te worden. Het snel vinden van een zo goed mogelijke oplossing is een uitdagende onderzoeksvraag. In dit artikel worden een aantal voorbeelden van dergelijke problemen uit de praktijk behandeld, en wordt aangegeven hoe oplossingsmethoden voor combinatorische problemen kunnen helpen bij het bereiken van een efficiënte bedrijfsvoering.

* Met dank aan prof W. Gochet voor het becommentariëren van een eerdere versie van dit werk.

I. INLEIDING

Optimaliseringsproblemen zijn alomtegenwoordig in de huidige maatschappij. Budgetallocatie problemen in grote organisaties, productieplanning problemen in bedrijven, crew-scheduling problemen in de transportsector, en roosterproblemen in onderwijsinstellingen zijn daar voorbeelden van. Het voortdurende streven naar efficiency, de toenemende concurrentiedruk, en het meer en meer wegvallen van nationale grenzen hebben geleid tot de noodzaak zulke optimaliseringsproblemen snel en goed op te lossen. Beslissingen die volgen uit een oplossing van een dergelijk probleem hebben vaak een dichotoom, niet-continu, karakter: wordt er ja dan nee een nieuwe vestiging op die locatie geopend?, wordt deze klas al dan niet aan dit lokaal toegewezen?, wordt deze klant uit dit of uit dat depot bevoorradad?, enzovoort. Problemen die dergelijke beslissingen behelzen zijn combinatorisch van aard. Dat wil zeggen dat het aantal mogelijke oplossingen eindig is.

Dit artikel gaat over zulke combinatorische optimaliseringsproblemen. Combinatorische optimaliseringsproblemen onderscheiden zich van andere optimaliseringsproblemen doordat “klassieke” optimaliseringstechnieken die gebruik maken van een gradiënt niet zomaar toepasbaar zijn. Het niet-continue karakter van de variabelen vereist het gebruik van specifieke oplossingsmethoden. We bespreken zeer kort verschillende soorten methoden om combinatorische optimaliseringsproblemen op te lossen. Verder geeft dit artikel een schets van een aantal situaties uit de (bedrijfseconomische) praktijk, met een combinatorisch optimaliseringsprobleem in de hoofdrol. Doel van deze schets is om de lezer een beeld te geven van de diversiteit van de toepassingen van combinatorische optimaliseringsproblemen. Ook wordt hiermee getracht de lezer behulpzaam te zijn bij het herkennen van een combinatorisch optimaliseringsprobleem in de praktijk van alledag.

Zoals geschetst: combinatorische optimaliseringsproblemen zijn optimaliseringsproblemen waarbij het aantal toelaatbare oplossingen eindig is. Hoewel men in het verre verleden problemen heeft bestudeerd (zie Krarup en Vajda (1997), en referenties in Schrijver (1986)), die we nu als combinatorisch optimaliseren zouden bestempelen, is het vakgebied tot wasdom gekomen met de ontwikkeling van de computer. Immers, tijd (dat wil zeggen de door een computer benodigde rekentijd) is een essentieel ingrediënt van een combinatorisch optimaliseringsprobleem: wanneer rekentijd geen schaarse factor zou zijn,

is het vinden van een beste oplossing een triviale exercitie geworden. Men kan dan immers eenvoudigweg alle mogelijke oplossingen van het combinatorische optimaliseringsprobleem enumereren.

Echter, in de meeste toepassingen, is tijd een schaarse factor. Dat betekent dat er tenminste twee criteria zijn waarmee men een oplossingsmethode (of algoritme) voor een combinatorisch probleem kan beoordelen: kwaliteit van de gevonden oplossing en benodigde rekentijd. Natuurlijk hangt de benodigde rekentijd voor het oplossen van een instantie af van de toevallige computer waarmee gerekend wordt. Vandaar dat men het aantal bewerkingen (een bewerking is bijvoorbeeld het optellen, vermenigvuldigen, of vergelijken van twee getallen) als maatstaf voor de rekentijd van een algoritme neemt. We komen hiermee op het gebied van de complexiteitstheorie; we verwijzen naar Garey and Johnson (1979), of Ausiello et al. (1999) voor een uitgebreide en formele beschrijving van dit onderwerp.

Er zijn verschillende soorten oplossingsmethoden voor combinatorische optimaliseringsproblemen ontwikkeld in de afgelopen decennia. Een belangrijke eigenschap van een methode is of deze een optimale oplossing vindt voor elke instantie. Is dat het geval dan wordt de methode exact genoemd. Methoden die niet noodzakelijkerwijs een beste oplossing vinden worden heuristisch genoemd. Een ander belangrijk kenmerk van een methode wordt gevormd door het aantal rekenstappen dat een methode nodig heeft (als functie van de probleemgrootte) om een instantie op te lossen. Dit aantal rekenstappen kan polynomiaal begrensd zijn in de probleemgrootte, dan heet de methode polynomiaal; of dit kan een exponentiële functie van de probleemgrootte zijn. Een exponentiële methode heeft het risico dat een uitkomst zeer lang op zich kan laten wachten.

Grof gezegd kan men stellen dat combinatorische optimaliseringsproblemen in twee soorten uiteenvallen: die problemen waarvoor een polynomiaal (= snel) exact algoritme bekend is, en die waarvoor een dergelijk algoritme (nog) niet bekend is. Problemen van de eerste soort bevinden zich in de klasse \mathcal{P} , de moeilijkste problemen van de tweede soort worden \mathcal{NP} -moeilijk genoemd.

In vele gevallen is het mogelijk om een combinatorisch optimaliseringsprobleem te formuleren als een zogenaamd geheeltallig lineair programmeringsprobleem. In een dergelijke formulering dient er een lineaire doelstellingsfunctie geoptimaliseerd te worden onder lineaire beperkingen; bovendien moet een gegeven gedeelte van de variabelen geheeltallige waarden krijgen. Technieken voor het oplossen

van dergelijke formuleringen zijn dus van groot belang om combinatorische problemen op te kunnen lossen, zie bijvoorbeeld Degraeve (1994). In de volgende secties wordt een aantal van dergelijke formuleringen, en daarop gebaseerde oplossings technieken beschreven. In de praktijk gaat het erom, voordat men aan het oplossen van probleeminstaties toekomt, het probleem te identificeren, en de wezenlijke kenmerken van de praktijksituatie in een model onder te brengen. In het vervolg van dit artikel wordt aan de hand van real-life cases een aantal van dergelijke toepassingstrajecten geschetst.

II. OVER VERSNIJDEN

Een klassiek probleem uit het vakgebied van het operationeel onderzoek is het zogenaamde versnijdingsprobleem (ook wel cutting stock probleem genoemd).

Dit probleem komt met name voor in bedrijfstakken waar materiaal versneden moet worden, bijvoorbeeld in de papierindustrie of in de textielindustrie. Een beschrijving van het klassieke versnijdingsprobleem is als volgt. Gegeven zijn een aantal orders. Een order wordt gekenmerkt door twee getallen, namelijk zijn breedte b_i (het formaat), en het aantal benodigde items van dat formaat d_i (de vraag), $i = 1, \dots, n$. Verder is er een aantal identieke rollen met een gegeven breedte W beschikbaar waar de orders uit gesneden dienen te worden. De doelstelling is om dat zodanig te doen dat het snijverlies minimaal is. In Tabel 1 wordt een voorbeeld met 5 orders beschreven; verder geldt $W = 100$, (zie Chvátal (1980)).

Een mogelijke manier om een rol te versnijden is om 1 item van order 1, 1 item van order 4, en 4 items van order 5 te nemen. Dit leidt tot een snijverlies van $100 - (1 \times 45) - (1 \times 14) - (4 \times 10) = 1$. In het

TABEL 1
Een instantie

order	1	2	3	4	5
breedte b	45	36	31	14	10
vraag d	97	610	395	211	88

vervolg zullen we een mogelijke manier om zo'n rol te versnijden een patroon noemen. Het versnijdingsprobleem is een combinatorisch probleem: immers, er is een eindig aantal patronen mogelijk (dat echter zeer groot kan zijn). In dit voorbeeld zijn er 139 verschillende toelaatbare patronen mogelijk. Bovendien dient men, om een oplossing te specificeren, een multipliciteit voor elk patroon vast te stellen. Merk op dat, in de beschrijving van deze standaardvariant van het versnijdingsprobleem, het minimaliseren van het snijverlies equivalent is met het minimaliseren van het aantal gebruikte rollen.

Dit probleem wordt al genoemd in 1939 door Kantorovich (1960). In de zestiger jaren ontwikkelen Gilmore en Gomory ((1961), (1963)) hun beroemde kolomgeneratieprocedure voor dit probleem. De huidige state-of-the-art wordt beschreven in het proefschrift van Peeters (2002). Er zijn natuurlijk allerlei varianten op dit klassieke probleem mogelijk; een veel voorkomende variant is het probleem waarbij men voor elke order i tenminste d_i items moet produceren ($1 \leq i \leq n$). Ook zijn generalisaties naar twee en meer dimensies uitgebreid in de literatuur onderzocht. Een classificatie van versnijdingsproblemen wordt gegeven in Dyckhoff en Finke (1992).

A. *De case*

Een mogelijke reden die verklaart waarom het gebruik van optimaliseringstechnieken in deze sector niet zo wijdverbreid is als wellicht mogelijk zou kunnen zijn, is de exclusieve gerichtheid van deze technieken op het minimaliseren van het snijverlies. Oplossingen met een minimaal snijverlies kennen vaak een relatief groot aantal verschillende patronen. Vaak is het zo dat het achtereen produceren van een tweetal verschillende patronen een setup vergt. In de praktijk is men dan bereid meer snijverlies te accepteren als dat leidt tot een vermindering van het aantal verschillende patronen en dus minder setups. Dit fenomeen wordt geïllustreerd aan de hand van de volgende case.

Gelva BV is een middelgrote onderneming gevestigd in Zuid-Limburg (NL), die schuurpapier produceert. Gelva levert (wereldwijd) ongeveer 500 basisartikelen, te weten 50 typen schuurpapier in 10 dikten; in totaal wordt er per jaar ongeveer 3,5 miljoen vierkante meter schuurpapier geproduceerd. Het schuurpapier wordt geproduceerd in de vorm van zogenaamde jumborollen die vervolgens versneden worden tot orders. Eén van de machines die daartoe gebruikt wordt is de

snijmachine. In zo'n snijmachine hangt een jumborol die met een bepaalde snelheid afgewikkeld wordt. De machine heeft een aantal messen (35) tot zijn beschikking die op het af te wikkelen schuurpapier gezet kunnen worden; niet elk mes hoeft ingezet te worden. De positie van de messen wordt ingesteld door een bediende. Gelva BV onderscheidt twee soorten orders: standaard orders en niet-standaard orders. Voor een niet-standaard order i dient exact de gevraagde hoeveelheid van b_i items geproduceerd te worden; voor een standaard order i dient minstens de gevraagde hoeveelheid van b_i items geproduceerd te worden produceren ($1 \leq i \leq n$). De gedachte hiërarchie is dat standaard orders een relatief grote omloopsnelheid hebben, en dat eventuele overschotten door toekomstige orders snel weggenomen zullen worden. Een voorbeeld van een kleine instantie met louter standaard orders is te zien in Tabel 2; voor deze instantie geldt $W = 1380$.

Er waren verschillende motieven voor Gelva BV om tot een herziening van het huidige planningsproces over te gaan. De planning van de snijmachine gebeurde handmatig, en was een relatief tijdrovend karwei. Men was dus afhankelijk van de ervaring van een planner. Bovendien werd er een oplossing gevonden met onbekende kwaliteit.

Beschouw nu in Tabel 3 de oplossing gebruikt door Gelva BV voor de instantie beschreven in Tabel 2. Hierin ziet u dat 3 jumborollen versneden worden volgens patroon 1 en dat 2 jumborollen versneden worden volgens patroon 2. Ter illustratie, patroon 1 houdt in dat er 28 items van order 1, 3 items van order 2 en 2 items van order 3 uit een jumborol versneden worden; het corresponderende snijverlies bedraagt dus $1380 - (28 \times 30) - (3 \times 100) - (2 \times 115) = 10$. Het totale snijverlies van de oplossing bedraagt 70. Wat bepaalt nu de kwaliteit van deze oplossing?

TABEL 2
Een instantie van Gelva BV

order	1	2	3
breedte b	30	100	115
vraag d	146	13	10

TABEL 3
Oplossing van Gelva BV

	aantal items order 1	aantal items order 2	aantal items order 3	snijverlies	frequentie
patroon 1	28	3	2	10	3
patroon 2	31	2	2	20	2
totaal	146	13	10	70	

In een aantal gesprekken met het management van Gelva kwam naar voren dat zowel het snijverlies als het aantal setups bepalend was voor de kwaliteit van een oplossing. Wel bleek het bijzonder lastig te zijn om een eenduidige doelstellingsfunctie te specificeren. Oorzaak was dat in sommige situaties, wanneer het niet druk was, men bereid was meer tijd aan setups te besteden om zo het snijverlies tot een absoluut minimum te beperken, terwijl men, in een situatie met meer tijdsdruk, een andere oplossing met minder setups prefereerde. Met andere woorden, voor dezelfde instantie werd soms de ene oplossing, dan weer de andere geprefereerd. Uiteindelijk werd besloten om dit probleem als een twee-criteria probleem op te lossen, en alle zogenaamde Pareto-optimale oplossingen te genereren. (Een oplossing van een twee-criteria probleem heet Pareto-optimaal wanneer er geen andere oplossing bestaat die op beide criteria minstens even goed scoort en op één criterium beter dan de gegeven oplossing.) Dit leidde tot het oplossen van een klassiek versnijdingsprobleem met daaraan toegevoegd de volgende kenmerken:

- er is een bovengrens op het aantal items dat men uit een jumbo-rol kan snijden. Deze bovengrens volgt uit het aantal messen op de snijmachine. Overigens is dit een bekende beperking in versnijdingsproblemen, zie Gilmore and Gomory (1963);
- er zijn twee typen orders. Vaak ziet men in de praktijk dat afwijkingen van de gevraagde hoeveelheid zijn toegestaan; in dit geval gaat het dus om zeer specifiek bepaalde afwijkingen die zijn toegestaan;
- er dient een twee-criteria model opgelost te worden.

Merk op dat, in contrast met het klassieke versnijdingsprobleem, het nu mogelijk is dat een oplossing die gebruik maakt van meer dan

het minimaal aantal benodigde rollen minder snijverlies heeft dan een oplossing die gebruik maakt van het minimaal aantal benodigde rollen. Om een model voor bovenbeschreven situatie te formuleren introduceren we de volgende parameters:

- n is het aantal items, en definieer $N \equiv \{1, 2, \dots, n\}$,
- $R \subseteq N$ is de verzameling van niet-standaard orders,
- $S \subseteq N$ is de verzameling van standaard orders (met $R \cup S = N$, $R \cap S = \emptyset$),
- W is de breedte van een jumborol,
- C is het maximale aantal items dat uit een jumborol versneden kan worden,
- K is het aantal beschikbare jumborollen, en

voor $i = 1, 2, \dots, n$:

- b_i is het formaat van een item van order i ,
- d_i is het aantal benodigde items van order i .

We gebruiken de volgende beslissingsvariabelen, waarbij de index j loopt over de verschillende patronen volgens welke men een jumborol kan versnijden:

- x_{ij} = het aantal items van order i in jumborol j ($i \in N, j = 1, \dots, K$),
- $y_j = \begin{cases} 1 & \text{wanneer patroon } j \text{ gebruikt wordt,} \\ 0 & \text{anders } (j = 1, \dots, K), \end{cases}$
- f_j is de frequentie van patroon j , dat wil zeggen het aantal jumborollen dat volgens patroon j versneden wordt ($j = 1, \dots, K$).

Een (niet-lineair) model ziet er vervolgens zo uit:

$$\text{minimaliseer} \quad Z_1 = \sum_{j=1}^K f_j \left(W - \sum_{i=1}^n b_i x_{ij} \right) \quad (1)$$

$$\text{minimaliseer} \quad Z_2 = \sum_{j=1}^K y_j \quad (2)$$

onder de beperkingen

$$\sum_{j=1}^K f_j x_{ij} = d_i \quad \text{voor } i \in R; \quad (3)$$

$$\sum_{j=1}^K f_j x_{ij} \geq d_i \quad \text{voor } i \in S; \quad (4)$$

$$f_j \leq Ky_j \quad \text{voor } j = 1, \dots, K; \quad (5)$$

$$\sum_{j=1}^K f_j \leq K \quad (6)$$

$$\sum_{i=1}^n b_i x_{ij} \leq Wy_j \quad \text{voor } j = 1, \dots, K; \quad (7)$$

$$\sum_{i=1}^n x_{ij} \leq Cy_j \quad \text{voor } j = 1, \dots, K; \quad (8)$$

$$x_{ij} \text{ integer} \quad \text{voor } i \in N, j = 1, \dots, K; \quad (9)$$

$$f_j \text{ integer} \quad \text{voor } j = 1, \dots, K; \quad (10)$$

$$y_j \in \{0, 1\} \quad \text{voor } j = 1, \dots, K. \quad (11)$$

Beide doelstellingen zijn geformuleerd middels (1) (minimaliseer het snijverlies) en (2) (minimaliseer het aantal patronen). Beperkingen (3) eisen dat voor niet-standaard orders precies de gewenste hoeveelheid geproduceerd wordt; beperkingen (4) eisen dat voor standaard orders tenminste de gewenste hoeveelheid geproduceerd wordt. De frequentie van een patroon kan alleen positief zijn wanneer dat patroon gebruikt wordt in (5). Beperking (6) zegt dat er niet meer dan K jumborollen gebruikt mogen worden. Beperkingen (7) geven aan dat een patroon niet de breedte van een jumborol mag overschrijden, en beperkingen (8) impliceren dat er niet meer dan C items uit een jumborol gesneden kunnen worden. Tenslotte worden in (9), (10) en (11) de geheeltalligheidseisen vermeld.

Voor dit probleem is een enumeratief algoritme ontwikkeld dat alle Pareto-optimale oplossingen genereert. We verwijzen naar Kolen en Spieksma (2000) voor een precieze beschrijving van dat algoritme. Dit algoritme is geïmplementeerd, geïnstalleerd en geïncorporeerd in het planningsproces bij Gelva BV. In Tabellen 4 en 5 ziet u de door dat algoritme gevonden oplossingen.

De oplossing in Tabel 4 heeft een minimaal snijverlies uitgaande van twee patronen (en 5 jumborollen). Deze oplossing domineert de oplossing beschreven in Tabel 3. De oplossing in Tabel 5 – die gebruik maakt van 6 jumborollen – heeft een minimaal snijverlies. Beide oplossingen tezamen vertegenwoordigen de verzameling van Pareto-optimale oplossingen. Overigens is er recentelijk meer aandacht gekomen in de literatuur voor het minimaliseren van het aantal setups in versnijdingsproblemen (zie Vanderbeck (2000) en McDiarmid (1999)).

TABEL 4
Een oplossing

	aantal items order 1	aantal items order 2	aantal items order 3	snijverlies	frequentie
patroon 1	19	1	6	20	1
patroon 2	32	3	1	5	4
totaal	147	13	10	40	

TABEL 5
Nog een oplossing

	aantal items order 1	aantal items order 2	aantal items order 3	snijverlies	frequentie
patroon 1	23	0	6	0	3
patroon 2	26	6	0	0	3
totaal	147	18	18	0	

III. OVER VOLGORDE'S

Het prototype combinatorisch optimaliseringsprobleem is het zogenaamde *handelsreizigersprobleem* (ook wel *traveling salesman problem* of TSP genoemd): gegeven n steden en hun onderlinge afstanden, vind een volgorde (of tour) door deze steden met minimale totale afstand, zie Lawler et al. (1985) voor een overzicht. Een generalisatie van dit probleem die veel toepassingen kent in de transportsector is het zogenaamde *vehicle routing probleem*, waarbij er een vloot vrachtwagens, gehuisvest in een depot, en een verzameling te bezoeken klanten is gegeven. In dit probleem dient er vervolgens een route voor elk der vrachtwagens geconstrueerd te worden, zodanig dat elke klant bezocht wordt. Van dit *vehicle routing probleem* zijn ook weer vele varianten onderzocht. In de volgende case komt een variant aan bod.

A. De case

Marcon BV is een bedrijf dat kranen verhuurt. Deze kranen worden voornamelijk gebruikt op bouwplaatsen om zware voorwerpen te

verplaatsen. De modus operandi van het bedrijf is als volgt: een kraan vertrekt elke ochtend vanaf het depot naar de locatie van zijn eerste taak, voert die taak uit, reist naar de locatie van de tweede taak, voert die taak uit, enzovoort, totdat de kraan zijn laatste taak van die dag heeft uitgevoerd, waarna de kraan terugkeert naar het depot. Een belangrijke eigenschap van een kraan is zijn liftvermogen; ook is er een kostenparameter voor elke kraan bekend. In Tabel 6 geven we een overzicht van de kranen die Marcon BV tot zijn beschikking heeft: er zijn 13 typen kranen, m_t stelt het aantal kranen van type t voor, cap_t is het liftvermogen van een kraan van type t (in tonnen), en r_t is de kost parameter van een kraan van type t (in EURO's).

Het toewijzen van taken aan kranen (het planningsproces) verloopt als volgt. Klanten bellen met een order. Een order bestaat uit het specificeren van

- de dag waarop de kraan benodigd is (elke dag vanaf morgen is toegestaan),
- de locatie en het tijdstip waarop de kraan benodigd is,
- de periode gedurende welke de kraan benodigd is, en
- het minimaal benodigde hefvermogen.

Merk op dat voor een taak met minimaal benodigd hefvermogen c , het aan *elke* kraan met hefvermogen groter of gelijk aan c toegestaan is om de taak uit te voeren. Tenslotte: een kraan kan slechts één taak tegelijkertijd uitvoeren en een taak mag niet onderbroken worden. De planners van Marcon staan vervolgens voor de taak om

TABEL 6
De kranen

type t	1	2	3	4	5	6	7
m_t	1	2	3	11	1	4	3
cap_t	7	20	25	30	35	40	45
r_t	585	675	680	715	715	735	735
type t		8	9	10	11	12	13
m_t		2	2	2	1	1	1
cap_t		60	70	90	120	150	300
r_t		855	980	990	1555	1610	1670

een toelaatbare toewijzing van taken aan kranen te vinden. De kwaliteit van zo'n toewijzing hangt mede af van het bedrag dat een klant aan Marcon betaalt. Dat bedrag hangt af van een tweetal zaken:

1. de kosten van een kraan met het minimaal benodigd hefvermogen om de taak uit te voeren (de r_i parameters in Tabel 5), en
2. de lengte van de periode dat de kraan benodigd is.

Het product van deze twee zaken is het bedrag dat de klant aan Marcon betaalt (dus wanneer men een kraan nodig heeft om een 48-tons opdracht uit te voeren, gedurende 4 uur betaalt men $4 \times 855 = 3420$ EURO). Hieruit volgt dat de totale opbrengst op een dag niet afhangt van de toewijzing van taken aan kranen (zolang elke taak maar toegewezen wordt). Dat betekent dat het minimaliseren der kosten centraal komt te staan in deze applicatie.

Na overleg met het management van Marcon werd besloten tot een volgende kostenstructuur: laat de *werkperiode* van een kraan gelijk zijn aan de periode dat de kraan werkt aan taken plus de tijd dat de kraan reist tussen locaties waar de taken uitgevoerd worden. Een werkperiode bevat dus niet eventuele wachttijden op locaties. We definiëren nu de kosten van een kraan als het product van z'n werkperiode en de corresponderende kost parameter r_i . De totale kosten zijn dan gelijk aan de de kosten gesommeerd over de kranen. Het probleem is nu om een toewijzing van taken aan kranen te vinden zodanig dat de totale kosten geminimaliseerd worden. Het zal duidelijk zijn dat de keuze van deze doelstellingsfunctie met zich mee brengt dat lange reisafstanden zoveel mogelijk vermeden zullen worden.

Men kan dit probleem zien als een generalisatie van het zogenaamde crew scheduling probleem: het crew scheduling probleem bestaat eruit dat crews dienen te worden toegewezen aan taken die elk een bekende start- en eindtijd hebben. Een crew kan ten hoogste 1 taak tegelijkertijd uitvoeren, en overlappende taken kunnen niet aan een zelfde crew toegewezen worden. Wanneer elke taak door elke kraan uitgevoerd zou kunnen worden verkrijgt men dit probleem (zie Mingozzi et al. (1999) en Beasley en Cao (1998)), (met kranen in de rol van crews). Merk op dat in zo'n crew scheduling probleem aspecten van vehicle routing aanwezig zijn: wanneer men de aanvangstijden van de taken zelf zou mogen vaststellen verkrijgt men een vehicle routing probleem.

We kunnen dit probleem formuleren als een geheeltallig programmeringsprobleem. Daartoe construeren we een gerichte graaf G , waarbij er een knoop in G is voor elke taak, en er is een source s en

een sink f die beide het depot voorstellen. Verder zijn twee knopen (taken) i en j verbonden wanneer het mogelijk is voor een kraan om, na taak i uitgevoerd te hebben, te reizen naar de locatie van taak j , en daar op tijd aan te komen, dat wil zeggen vóór de door de klant gespecificeerde begintijd van taak j .

We gebruiken de volgende parameters:

- m : het aantal kraantypen,
- P : de verzameling van paden in G van s naar f ,
- voor $i = 1, \dots, n, p \in P$:

$$\delta_{ip} = \begin{cases} 1 & \text{als taak } i \text{ ligt op pad } p, \text{ en} \\ 0 & \text{anders} \end{cases}$$

- voor $t = 1, \dots, m, p \in P$: c_{tp} = kosten wanneer een kraan van type t pad p neemt, (deze kosten zijn bekend; merk op dat als een pad p een taak bevat die niet uitgevoerd kan worden door een kraan van type t , dan wordt c_{tp} gelijk gezet aan een groot getal).

Ter illustratie, beschouw het volgende voorbeeld:

Voorbeeld:

Gegeven is een instantie met 2 typen kranen en 3 taken:

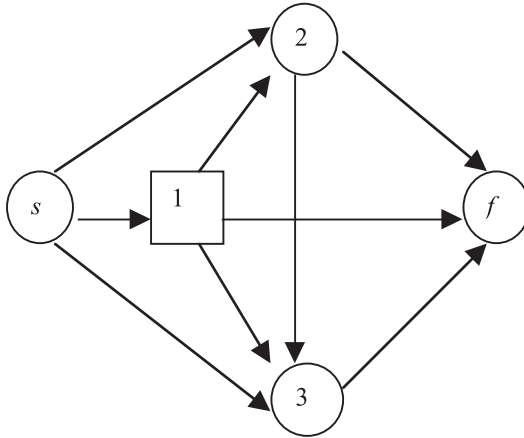
$m = 2, m_1 = m_2 = 1, cap_1 = 10, cap_2 = 20, r_1 = 1, r_2 = 3$, taak 1 vereist een kraan van type 2, terwijl taak 2 en taak 3 door elke kraan uitgevoerd kunnen worden. Bij deze instantie kunnen we de volgende graaf construeren:

De kosten zijn weergegeven in Tabel 7.

TABEL 7
De kosten

type p	c_{1p}	c_{2p}
$s - 1 - f$	–	96
$s - 2 - f$	20	60
$s - 3 - f$	18	54
$s - 1 - 2 - f$	–	114
$s - 1 - 3 - f$	–	111
$s - 2 - 3 - f$	35	105
$s - 1 - 2 - 3 - f$	–	159

FIGUUR 1
De graaf G



Een optimale oplossing is om de kraan van type 1 het pad $s - 2 - 3 - f$ te laten volgen, en de kraan van type 2 het pad $s - 1 - f$; de totale kosten bedragen dan 131.

Ter formulering van het probleem als een geheeltallig lineair programmeringsprobleem gebruiken we de volgende beslissingsvariabelen: voor $t = 1, \dots, m, p \in P$,

$$y_{tp} = \begin{cases} 1 & \text{wanneer een kraan van type } t \text{ pad } p \text{ neemt,} \\ 0 & \text{anders.} \end{cases}$$

Het model ziet er vervolgens zo uit:

$$\text{minimaliseer } \sum_{t=1}^m \sum_{p \in P} c_{tp} y_{tp} \quad (12)$$

onder de beperkingen

$$\sum_{t=1}^m \sum_{p \in P} \delta_{ip} y_{tp} = 1 \quad \text{voor } i = 1, \dots, n; \quad (13)$$

$$\sum_{p \in P} y_{tp} \leq m_t \quad \text{voor } t = 1, \dots, m; \quad (14)$$

$$y_{tp} \in \{0,1\} \quad \text{voor } t = 1, \dots, m, p \in P. \quad (15)$$

De doelstellingsfunctie (12) minimaliseert de kosten zoals beschreven hierboven, beperkingen (13) impliceren dat elke taak vervuld

dient te worden, en beperkingen (14) eisen dat de cardinaliteitseisen met betrekking tot het aantal beschikbare kranen van type t niet overschreden worden. Tenslotte zijn (15) de geheeltalligheidseisen. Het model dat ontstaat wanneer we (15) vervangen door beperkingen $y_{tp} \geq 0$ noemt men de LP-relaxatie van (12)-(15).

Er zijn een tweetal obstakels die overwonnen moeten worden om model (12)-(15) snel te kunnen oplossen. Ten eerste kent het model een potentieel enorm groot aantal variabelen. Het aantal mogelijke paden in de graaf G kan zeer snel oplopen; vandaar dat het geen goed idee zou zijn om alle variabelen y_{tp} expliciet te gebruiken. Een techniek die het mogelijk maakt om tenminste de LP-relaxatie van het model op te lossen zonder alle variabelen te enumereren is *kolomgeneratie*; deze techniek gaat terug tot Dantzig en Wolfe (1960). Hierbij wordt in eerste instantie een model opgelost dat slechts een (klein) gedeelte van de variabelen bevat (het zogenaamde *restricted master* probleem). Vervolgens dient natuurlijk bepaald te worden of de gevonden oplossing optimaal is of dat er wellicht nog variabelen zijn die in de basis gebracht moeten worden. Het beantwoorden van deze vraag staat in de literatuur bekend als het pricing probleem (zie bijvoorbeeld Barnhart et al. (1998)). Het blijkt dat we voor model (12)-(15) dit pricing probleem als een kortste pad probleem kunnen oplossen (zie Faneyte et al. (2001)). Dit maakt het mogelijk om de corresponderende LP-relaxatie snel te kunnen oplossen. Het tweede obstakel wordt gevormd door de vereiste geheeltalligheid van de variabelen. Het toepassen van kolomgeneratie, in combinatie met het efficiënt oplossen van het pricing probleem, levert ons de oplossing van de LP-relaxatie van het model op. Hoe kunnen we die gebruiken om een geheeltallige oplossing te vinden? Laten we terugkeren naar ons eerder beschreven voorbeeld weergegeven in Figuur 1. Een optimale LP-oplossing is:

$$y_{1, s-2-3-f} = y_{2, s-1-2-f} = y_{2, s-1-3-f} = \frac{1}{2}, \text{ en alle andere}$$

variabelen zijn 0. De waarde van deze LP-oplossing is 130.

Voor een optimale geheeltallige oplossing geldt het volgende: of knopen 1 en 2 worden direct na elkaar bezocht, of niet. Laat P_{12} de deelverzameling van P zijn die alle paden van P bevat waarin knoop 1 en knoop 2 elkaar direct volgen. Uitgedrukt in een formule weten we dus dat voor elk type t :

$$\sum_{p \in P_{12}} y_{tp} \in \{0, 1\}$$

De gevonden LP-oplossing voldoet niet aan deze eis, voor type 2 geldt:

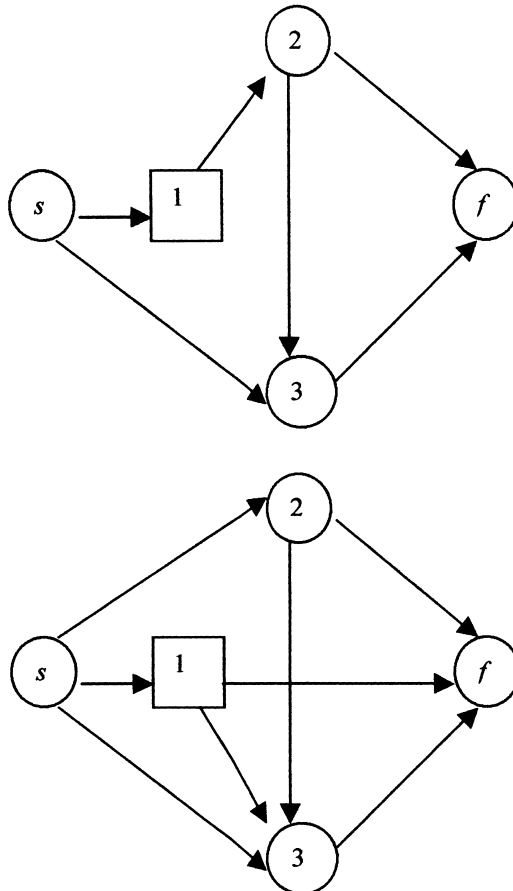
$$\sum_{p \in P_2} y_p = \frac{1}{2}$$

We kunnen nu twee grafen opstellen, weergegeven in Figuur 2, zodanig dat de optimale oplossing in de linkertak of in de rechtertak zit, en zodanig dat de huidige LP-oplossing wordt weggesnoeid.

Merk op dat in G_1 de verbindingen $\{s, 2\}$, $\{1, 3\}$ en $\{1, f\}$ zijn verwijderd. Wanneer we voor deze graaf de LP-relaxatie oplossen vinden

FIGUUR 2

De grafen G_1 (boven) en G_2 (onder)



we een geheeltallige oplossing met waarde 132. G_2 is geconstrueerd door verbinding $\{1, 2\}$ weg te halen. In deze tak vinden we als LP-relaxatie ook een geheeltallige oplossing met waarde 131. Deze oplossing is dus optimaal. De hier beschreven methode werkt niet alleen voor dit voorbeeld, maar kan in het algemeen worden toegepast. Dit is gebaseerd op het feit dat als de LP-oplossing fractioneel is, er altijd een tweetal knopen blijkt te bestaan zodanig dat de som van de fracties corresponderend met paden waarbij die twee knopen elkaar direct opvolgen, fractioneel is. In de literatuur wordt deze aanpak een zogenaamde branch-and-price methode genoemd. Deze benadering is toegepast bij Marcon BV, en heeft geleid tot inzichten die leidden tot een verbetering in de bedrijfsvoering. Meer specifiek, een kostenreductie van minstens 10% werd behaald voor de als representatief beschouwde probleem instanties.

IV. OVER VEILINGEN

Een combinatorische veiling is een veiling waarin meerdere goederen tegelijkertijd aangeboden worden aan meerdere bidders. In een combinatorische veiling is het een bidder toegestaan een bod uit te brengen op een willekeurige deelverzameling van de aangeboden goederen. De veilingmeester besluit – na een aantal ronden, of na een zekere tijdsspanne, dat hangt af van de veilingopzet – om een aantal uitgebrachte biedingen te accepteren, en de betreffende goederen aan de corresponderende bidders toe te wijzen. We verwijzen naar Rothkopf en Park (1998) voor een recente inleiding over veilingen.

A. *Ontwikkelingen*

De populariteit van combinatorische veilingen (en het onderzoek ernaar) is de laatste jaren sterk toegenomen. Daar zijn een tweetal belangrijke redenen voor:

- erkenning van het synergie-effect (of super-additiviteit). Het is goed mogelijk dat een tweetal goederen, zeg goed A en goed B, samen meer waarde voor een bidder vertegenwoordigen dan de som van de waarden van goed A en B individueel. Wanneer men dus de veilingopbrengsten wenst te maximaliseren, is het van belang om dit synergie-effect niet te verwaarlozen. Een voorbeeld

- van een veiling waarbij dit effect een rol speelt betreft de verkoop van vliegtuigtickets. Stel u wilt van A naar B vliegen, en de dag daarop door naar C. In een dergelijke situatie heeft u niets aan louter een ticket A-B, of louter een ticket B-C. In feite wilt u alleen een bod uitbrengen als u beide tickets samen kan verkrijgen; combinatorische veilingen bieden precies die mogelijkheid;
- technologische ontwikkelingen. Meer en meer wordt het internet een plaats waar economische handelingen worden verricht, of, meer concreet, waar goederen worden gekocht en verkocht. Aangezien men, middels het ontwerpen van een elektronische marktplaats, dit proces (mede) vorm kan geven, is het van belang om de consequenties van verschillende mogelijkheden van zo'n ontwerp te onderzoeken. In verschillende e-business toepassingen wordt gebruik gemaakt van veilingen als manier vraag en aanbod op elkaar af te stemmen. De technologische ontwikkelingen in het afgelopen decennium maken het nu mogelijk om een complexe operatie als een combinatorische veiling elektronisch (bijvoorbeeld via Internet) te organiseren.

Het zal duidelijk zijn dat – vergeleken met een traditionele veiling waarin er slechts één goed tegelijkertijd onder de hamer komt – combinatorische veilingen een aantal extra beslissingsproblemen met zich mee brengen. Daarbij gaat het niet alleen om het ontwerp van de veiling (hoeveel ronden, welke informatie gaat naar de bidders, welke beperkingen zijn er op de biedingen), maar zeker ook om, gegeven de biedingen, de betreffende goederen te alloceren onder de bidders. We verwijzen naar De vries en Vohra (2002) voor een goed overzicht van de historie en de huidige stand van zaken met betrekking tot combinatorische veilingen. Om uit te vinden welke biedingen de winnende biedingen zijn, formuleren we een geheeltallig lineair programmeringsprobleem, waarbij we ervan uitgaan dat de veilingmeester tracht de opbrengsten te maximaliseren. We zullen dit probleem het veilingmeesterprobleem noemen. Een formulering ziet er als volgt uit: Laat M de verzameling te veilen goederen zijn, en laat $b(S)$ het hoogste bod op deelverzameling $S \subseteq M$ voorstellen. Wanneer we 0-1 beslissingsvariabelen $x(S)$ gebruiken met $x(S) = 1$ als het hoogste bod op S wordt geaccepteerd en $x(S) = 0$ anders, is het veilingmeester-probleem als volgt te formuleren.

$$\text{Maximaliseer} \quad \sum_{S \subseteq M} b(S)x(S) \quad (16)$$

onder de beperkingen

$$\sum_{S: i \in S} x(S) \leq 1 \text{ voor alle } i; \quad (17)$$

$$x(S) \in \{0,1\} \text{ voor alle } S \subseteq M. \quad (18)$$

De doelstellingsfunctie (16) maximaliseert de veilingopbrengsten, beperkingen (17) impliceren dat elk goed ten hoogste eenmaal wordt verkocht, en (18) zijn de geheeltalligheidsbeperkingen. Dit model is een zogenaamd set-packing probleem, en is in het algemeen tijdrovend om optimaal op te lossen (set-packing problemen zijn \mathcal{NP} -moeilijk). Oplossingsmethoden in de context van veilingen staan beschreven in bijvoorbeeld Andersson et al. (2000). De branch-and-price methodiek zoals beschreven in de vorige sectie is ook hier toepasbaar zie De vries en Vohra [8]. Wanneer de veilingopzet een aantal (snel opeenvolgende) biedingronden kent, kan het noodzakelijk zijn om snel een oplossing van model (16)-(18) te genereren. Dan kunnen heuristische methoden gebruikt worden; dergelijke methoden vinden snel een toelaatbare oplossing die echter niet optimaal hoeft te zijn. Overigens zijn goede oplossingen ook van belang in een branch-and-price aanpak zoals boven geschetst.

Er zijn verschillende varianten van combinatorische veilingen denkbaar (zie Sandholm et al. (2000)). Een mogelijke generalisatie van de bovenbeschreven situatie is dat er sprake is van multipliciteiten van goederen. Dus in plaats van één eenheid van goed A zijn er q eenheden van goed A beschikbaar, en is een bieder in staat om een gewenste hoeveelheid van elk goed te specificeren.

Een ander aspect dat bestudeerd wordt is de mogelijke structuur die – afhankelijk van het specifieke domein van de veiling – in het veilingmeester probleem aanwezig is. Beschouw bijvoorbeeld de veiling beschreven in Rothkopf et al. (1998), waarin stukken land aangeboden werden. Een eis die aan de biedingen werd opgelegd was dat elk bod gedaan moest worden op een aantal aanliggende stukken land, met andere woorden, het was niet toegestaan om een bod uit te brengen op een tweetal stukken land die niet met elkaar verbonden waren. Een dergelijke beperking kan grote gevolgen hebben voor de oplosbaarheid van het veilingmeester probleem; in dit geval, zoals ook beschreven in van Hoesel en Müller (2001), is de consequentie van de beschreven eis inderdaad dat er algoritmen bestaan die snel en optimaal een oplossing genereren. Er zijn meerdere voorbeelden van dit soort eisen mogelijk, zie Erlebach en Spieksma (2000).

Een populaire variant van een combinatorische veiling is de zogenaamde “omgekeerde” veiling. Hier gaat het om een veiling waarbij de bidders ondernemingen zijn en de goederen door de onderneming te verrichten diensten.

Een bod correspondeert vervolgens met de prijs waarvoor de betreffende onderneming die verzameling diensten offreert om uit te voeren. De doelstelling van de veilingmeester is hier om alle diensten te laten verrichten tegen minimale kosten. Voorbeelden van dergelijke inkoop-veilingen zijn een casus van Sears (zie Olsson et al. (2000)), en een casus bij Volvo (zie <http://www.tradeextensions.com/press/volvoPackCase.html>).

Een ander belangrijk aspect betreft de informatie die eenieder krijgt gedurende de veiling. Het is bijvoorbeeld zeer interessant om te weten hoe hoog uw bod op een specifieke deelverzameling zou moeten zijn om, gegeven de huidige biedingen, geaccepteerd te worden (het minimale biedprijs probleem). In het algemeen is het vinden van het vinden van de minimale biedprijs net zo lastig als het vinden van een optimale oplossing van het veiling-meester probleem. Echter, gegeven een instantie, is het zeker wel mogelijk dat het vinden van een minimale biedprijs minder tijdrovend is dan het bepalen van de optimale oplossing van het veiling-meester probleem. Bovendien kan men hier op zoek gaan naar ondergrenzen voor de minimale biedprijs. Hier speelt de duaal van de LP-relaxatie van model (16)-(18) een rol: deze duaal kent zogenaamde schaduwrijzen voor elk goed. Een schaduw-prijs is een bovengrens voor de hoeveelheid die dat goed bijdraagt aan de totale opbrengst.

Samengevat: het ontwerp en de implementatie van combinatorische veilingen vormen een boeiend onderzoeksgebied waaraan combinatorisch optimaliseren zowel modelmatig als in de algoritmieken een bijdrage kan leveren.

V. BESLUIT

Combinatorische optimaliseringsproblemen zijn optimaliseringsproblemen met een eindig aantal toelaatbare oplossingen. Dergelijke problemen hebben toepassingen in allerlei economische situaties; we hebben een versnijdingsprobleem, een volgordeprobleem en een veilingprobleem besproken. Het snel vinden van een (optimale) oplossing van instanties van combinatorische problemen is een uitdagend

onderzoeksveld waar operationeel onderzoek, wiskunde en informatica samenkomen.

REFERENTIES

- Andersson, A., M.Tenhunen en F. Ygge, 2000, Integer Programming for Combinatorial Auction Winner Determination, in Proceedings of the Fourth International Conference on Multi-Agent Systems (ICMAS00), 39-46.
- Ausiello, G., P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela en M. Protasi, 1999, Complexity and Approximation. Combinatorial Optimization Problems and their Approximability Properties, (Springer Verlag, Berlin).
- Barnhart, C., E.L. Johnson, G.L. Nemhauser, M.W.P. Savelsbergh en P.H. Vance, 1998, Branch and Price: Column Generation for Solving Huge Integer Programs, *Operations Research* 46, 316-329.
- Beasley, J.E. en B. Cao, 1998, A Dynamic Programming Based Algorithm for the Crew Scheduling Problem, *Computers and Operations Research* 25, 567-582.
- Chvátal, V., 1980, Linear Programming, (Freeman, New York).
- Dantzig, G.B. en P. Wolfe, 1960, Decomposition Principle for Linear Programs, *Operations Research* 8, 101-111.
- Degraeve, Z., 1994, Recent Developments in Integer Linear Programming to Solve Managerial Decision Problems, *Tijdschrift voor Economie en Management* 29, 353-380.
- De vries, S. en R. Vohra, 2002, Combinatorial Auctions: a Survey, (te verschijnen in), *INFORMS Journal on Computing*.
- Dyckhoff, H. en U. Finke, 1992, Cutting and Packing in Production and Distribution, (Physica-Verlag, Heidelberg).
- Erlebach, T. en F.C.R. Spijksma, 2000, Simple Algorithms for a Weighted Interval Selection Problem, in: Proceedings of the 11th Annual International Symposium on Algorithms and Computation (ISAAC '00), *Lecture Notes in Computer Science* 1969, 228-240.
- Faneyte, D.B.C., 1997, Kranen op (de kortste) weg, Thesis (Universiteit Maastricht).
- Faneyte, D.B.C., F.C.R. Spijksma en G.J. Woeginger, 2002, A Branch-and-Price Algorithm for a Hierarchical Crew Scheduling Problem, *Naval Research Logistics*, 49, 743-759.
- Garey, M.R. en D.S. Johnson, 1979, Computers and Intractability: a Guide to the Theory of NP-Completeness (Freeman, San Francisco).
- Gilmore, P. en R.E. Gomory, 1961, A Linear Programming Approach to the Cutting-Stock Problem, part I, *Operations Research* 9, 849-859.
- Gilmore, P. en R.E. Gomory, 1963, A Linear Programming Approach to the Cutting-Stock Problem, part II, *Operations Research* 11, 863-888.
- Kantorovich, L.V., 1960, Mathematical Methods of Organizing and Planning Production, *Management Science* 6, 366-422.
- Kolen, A.W.J. en F.C.R. Spijksma, 2000, Solving a Bi-Criterion Cutting Stock Problem with Open-Ended Demand: a Case Study, *Journal of the Operational Research Society* 51, 1238-1247.
- Krapu, J. en S. Vajda, 1997, On Torricelli's Geometrical Solution to a Problem of Fermat, *IMA Journal of Mathematics Applied in Business and Industry* 8, 215-224.
- Lawler, E.L., J.K. Lenstra, A.H.G. Rinnooy Kan en D.B. Shmoys, eds, 1985, The Traveling Salesman Problem. A Guided Tour of Combinatorial Optimization (Wiley, Chichester).
- McDiarmid, C., 1999, Pattern Minimisation in Cutting Stock Problems, *Discrete Applied Mathematics* 98, 121-130.
- Mingozzi, A., M.A. Boschetti, S. Ricciardelli en L. Bianco, 1999, A Set Partitioning Approach to the Crew Scheduling Problem, *Operations Research* 47, 873-888.

- Olsson, M., D. Porter, J. Ledyard, J. Swanson, en D. Torma, 2000, The First Use of a Combined Value Action for Transportation Services, Social Science Working Paper No. 1093 (California Institute of Technology).
- Peeters, M., 2002, One Dimensional Cutting and Packing: New Problems and Algorithms, Proefschrift No. 157 (FETEW, Katholieke Universiteit Leuven).
- Rothkopf, M.H. en S. Park, 2001, An Elementary Introduction to Auctions, *Interfaces* 31, 83-97.
- Rothkopf, M.H., A. Pekec en R.M. Harstad, 1998, Computationally Manageable Combinational Auctions, *Management Science* 44, 1131-1147.
- Sandholm, T., S. Suri, A. Gilpin en D. Levine, 2000, Winner Determination in Combinatorial Auction Generalizations, International Conference on Autonomous Agents, Workshop on Agent-Based Approaches to B2B, pp. 35-41 (Montreal, Canada), May 28th.
- Schrijver, A., 1986, Theory of Linear and Integer Programming (Wiley, Chichester).
- Vanderbeck, F., 2000, Exact Algorithm for Minimising the Number of Setups in the One-Dimensional Cutting Stock Problem, *Operations Research* 48, 915-926.
- Van hoesel, S., en R. Müller, 2001, Optimization in Electronic Markets: Examples in Combinatorial Auctions, *Netnomics* 3, 23-33.