STICHTING
MATHEMATISCH CENTRUM
2e BOERHAAVESTRAAT 49
AMSTERDAM

AFDELING MATHEMATISCHE STATISTIEK

Instituut voor Actuariaat en Econometrie

S 383 (DSB 2)

Een toewijzingsprobleem

door

J.H. van Frankenhuysen

April 1967

Samenvatting

De opdracht was om het beslissingsprobleem, geschetst tijdens een konsult van Lips N.V. te Drunen op te lossen met integer programmering.

I. Een formulering als integer lineair programmeringsprobleem werd gevonden, welke volgens de methode van Land en Doig opgelost kan worden, met een computerprogramma, ontworpen door J.M. Anthonisse.

II. Het probleem kan ook beschreven worden als gegeneraliseerd transportprobleem (zie b.v. [4]). Hiervoor bestaat een specifiek algorithme, maar dit leidt niet-noodzakelijk tot een integer oplossing, zodat ook hier Land en Doig toegepast werd.

III. Ook is nog een direkte, niet optimale aanpak geprobeerd, die in het algemeen tot bevredigende resultaten leidt.

De eerste drie paragrafen behandelen deze drie methoden theoretisch. In de vierde paragraaf zijn de toepassing op vier problemen en de resultaten daarvan vermeld.

Inleiding

Lips N.V. te Drunen is een metaalverwerkende industrie, welke o.a. messing- en aluminiumband levert. Voor de fabricage van het messingband heeft de betreffende afdeling, de beschikking over een aantal walsen, gloeiovens, bijtsmachines, scharen e.d.
Het produktieproces van deze messingbandwalserij kan in het kort als volgt worden beschreven:
Het benodigde materiaal wordt gesmolten en daarna centrifugaal tot een ring gegoten. Na het stollen wordt de ring (middellijn 150 cm, breedte 40 cm) aan de binnen- en aan de buitenzijde schoon gedraaid, de zijkanten worden gefreesd.
Hierna wordt de ring op één plaats opengezaagd en tot een rechte plaat gestrekt. Deze plaat gaat naar de eigenlijke walserij, waar hij door walsen, gloeien, bijtsen en snijden, op de gewenste kwaliteit, dikte en breedte wordt gebracht. Bij het walsen verandert de breedte van de plaat niet, hij wordt slechts langer en dunner. Wanneer de dikte 4 mm of minder bedraagt wordt de plaat opgerold tot een wikkel. De uitgangsdikte in de eigenlijke walserij is ongeveer 30 mm, terwijl de dikte van het eindprodukt varieert van 0,1 tot 13 mm. Zodoende is een aantal walsingen nodig om tot de gewenste einddikte te komen. Bovendien moet de plaat of wikkel tussentijds gegloeid en gebijtst worden, wanneer het materiaal door het walsen te hard geworden is en om een eindprodukt van de gewenste hardheid te verkrijgen.
Wanneer dus aan het einde van het walsprogramma de opdrachten uit de volle breedte van ± 38 cm gesneden worden, is het zaak om deze breedte zo volledig mogelijk te benutten. De afvalstroken boordstroken genaamd, verminderen de zgn. "uitwinning", waaronder de verhouding verstaan wordt van het gewicht aan verkoopbaar band, ten opzichte van het gewicht van de platen, waaruit dit band is gefabriceerd.
Het doel van het konsult was een methode te vinden om, bij een gegeven orderpakket, deze uitwinning te maximaliseren. Om dit te bereiken moeten de orders, die uit dezelfde wikkels gesneden kunnen worden zó gekombineerd worden op de volle wikkelbreedte, dat de hoeveelheid boordstrook die afvalt en daardoor teruggaat naar de oven zo klein mogelijk is.

Daar de materiaalsamenstelling in dit probleem geen rol speelt wordt een opdracht gekenmerkt door vier getallen: breedte, dikte, kwaliteit en het gevraagde aantal wikkels. Na sortering van de orders naar dikte en kwaliteit blijven er orderpakketten over, die elk dus gekenmerkt worden door twee getallen: dikte en kwaliteit. In één pakket onderscheiden de opdrachten (karweien) zich nog door twee getallen, nl. de gewenste breedte en het gevraagde aantal wikkels (frequentie) van die breedte.
Bij het zoeken naar geschikte kombinaties van karweien worden deze pakketten afzonderlijk behandeld. Wij zullen van nu af aan dan ook slechts spreken over één orderpakket. Het beschikbare materiaal bestaat uit een voorraad van reeds bij vorige opdrachten aangesneden strips (boordstroken) en een onbeperkt aantal nieuwe strips (wikkels) van standaardbreedte.
Een oplossing van het probleem is een indeling van de karweien op boordstroken + wikkels, zo dat alle karweien uitgevoerd worden.

§1. Formulering als integer L.P. probleem

Als kriterium voor optimalisatie van de indeling gebruiken we het aantal aangesneden wikkels: indelingen die een minimaal aantal wikkels nodig hebben zijn optimaal. Het aantal karweien is n; karwei j heeft breedte b_j en frequentie f_j.

We starten met het genereren van alle kombinaties van gevraagde strips, die uit één wikkel gemaakt kunnen worden. Dan is een kombinatie gekenmerkt door n gehele getallen, en we schrijven hem als een n-dimensionale vektor. De j-de komponent geeft aan hoeveel strips van karwei j in de kombinatie vertegenwoordigd zijn. Laat N het totale aantal kombinaties zijn, dat uit een volle wikkel gemaakt kan worden. Dan geldt voor iedere kombinatie $\vec{k}_i = (k_{i1}, \ldots, k_{in})$, $1 \leqq i \leqq N$:

$$k_{ij} \leqq f_j \quad \text{en} \quad \sum_{j=1}^{n} k_{ij} b_j \leqq s,$$

waarin s = breedte van een wikkel.

Er is echter ook een voorraad boordstroken aanwezig. Laten er m verschillende breedtes zijn en laat de m-dimensionale vektor $\vec{p} = (p_1, \ldots, p_m)$ de frequenties van deze breedtes aangeven. Een gedeelte van de kombinaties kan ook uit de boordstroken gemaakt worden. Voor de j-de boordstrookbreedte voegen we de N_j kombinaties toe die uit een boordstrook van die breedte gemaakt kunnen worden, en nummeren deze:

$$\vec{K}_{N+\sum_{i=1}^{j-1} N_i+1}, \ldots, \vec{K}_{N+\sum_{i=1}^{j} N_i}, \quad N_0 = 0, \quad \sum_{i=1}^{m} N_i = M.$$

In totaal hebben we nu N+M kombinaties, waarvan de eerste N bij wikkels horen, en van de laatste M kombinaties hoort ieder bij één bepaalde boordstrookbreedte.

We voeren ook N+M integer variabelen in: x_i, i = 1 ... N+M. De waarde die x_i aanneemt geeft aan hoe vaak kombinatie k_i gemaakt wordt. Zodoende moet voldaan zijn aan de m ongelijkheden:

$$\sum_{i=N+N_{j-1}+1}^{N+N_j} x_i \leq p_j \;, \quad 1 \leq j \leq m.$$

Bovendien moet gelden dat precies alle karweien uitgevoerd worden, hetgeen neerkomt op de n vergelijkingen:

$$\sum_{i=1}^{N+M} k_{ij} x_i = f_j \;, \quad 1 \leq j \leq n.$$

Samenvattend vinden we het L.P. probleem (1^o formulering):

minimaliseer $\sum_{i=1}^{N} x_i$ onder de bijvoorwaarden:

(1) $$\sum_{i=N+N_{j-1}+1}^{N+N_j} x_i \leq p_j \;, \quad 1 \leq j \leq m$$

$$\sum_{i=1}^{N+M} k_{ij} x_i = f_j \;, \quad 1 \leq j \leq n$$

$$x_i \geq 0, \text{ integer } \quad 1 \leq i \leq N+M.$$

Merk op dat de laatste M variabelen niet in de kriteriumfunktie opgenomen zijn.
Na invoeren in (1) van verschilvariabelen, die ook ≥ 0 en integer moeten zijn, kan het probleem in verkorte notatie opgeschreven worden als:

minimaliseer $\sum_{i=1}^{N} x_i$

onder $A\vec{x} = \vec{c}$

$\vec{x} \geq 0$, integer.

Hierin is A een $(n+m) \times (N+M+m)$ matrix; $\vec{x}$ een N+M+m-dimensionale vektor en $\vec{c}$ een n+m-dimensionale vektor.

Voor problemen van enige reële omvang is de matrix A, die $(n+m) \times (N+M+m)$ getallen bevat soms te groot om in het geheugen van de X8 bevat te worden. Zelfs als $m = M = 0$ is, dus wanneer er geen voorraad boordstroken aanwezig is, kan N nog zo groot zijn, dat de X8 de $n \times N$ getallen niet kan bevatten.
De omvang van het geheugen bepaalt dus de maximale afmetingen die het probleem mag aannemen. Tabel I (blz.18) geeft deze voor het Land en Doig programma van J.M. Anthonisse [1].

Het is mogelijk om de eerste formulering te verbeteren, door het aantal variabelen te beperken. Dit wordt bereikt door slechts kombinaties toe te staan die "vol" zijn, d.w.z. dat van alle opdrachten, minus degene gebruikt in de kombinatie, geen enkele strip aan de kombinatie toegevoegd kan worden zonder de breedte van de wikkel of boordstrook te overschrijden. Om er zeker van te zijn dat er een oplossing voor het probleem bestaat moeten de vergelijkingen $= f_j$ overgaan in ongelijkheden $\geqq f_j$.
Het probleem wordt nu (2^o formulering):

$$\text{minimaliseer} \sum_{i=1}^{N'} x_i,$$

$$\text{onder} \sum_{i=N'+N'_{j-1}+1}^{N'+N'_j} x_i \leqq p_j \ , \quad 1 \leqq j \leqq m$$

$$\sum_{i=1}^{N'+M'} k_{ij} x_i \geqq f_j \ , \quad 1 \leqq j \leqq n$$

$$x_i \geqq 0, \text{ integer} \ , \quad 1 \leqq i \leqq N'$$

Men ziet eenvoudig in dat het probleem in 1^o en 2^o formulering dezelfde minimumwaarde van de kriteriumfunktie oplevert. Uit iedere oplossing van de tweede formulering wordt een oplossing van de eerste formulering verkregen, door de ongelijkheden

$$\sum_{i=1}^{N'+M'} k_{ij} x_i \geqq f_j \ , \quad 1 \leqq j \leqq n$$

tot gelijkheden te maken: laat zoveel strips uit kombinaties met $x_j > 0$ weg, dat juist de frequenties f_j bereikt worden. Dus min (1° form.) $\leq$ $\leq$ min (2° form.).

Anderzijds wordt uit elke oplossing van de eerste formulering een oplossing van de tweede formulering verkregen door de gemaakte kombinaties vol te maken; dan gaan de gelijkheden $= f_j$ over in ongelijkheden: $\geq f_j$ en er geldt: min (2° form.) $\leq$ min (1° form.).

Evenals in de eerste formulering het geval is, zijn in de tweede formulering de laatste M' kombinaties, die bij boordstroken horen, niet in de kostenfunktie opgenomen. Daaraan ligt de wens ten grondslag om de voorraad zoveel mogelijk te beperken. Meer in het algemeen kan men zich afvragen waar in dit specifieke geval van het indelen van orders op wikkels de kosten liggen, en binnen welke grenzen het proces zich moet bewegen in een lange reeks van opvolgende indelingen.

In feite zijn dit juist de faktoren die de vorm van de te minimaliseren kostenfunktie bepalen.

Eén daarvan is de bovengenoemde wens om de voorraad te beperken. Een ander is b.v. het feit dat boordstroken beneden een bepaalde breedte teruggaan naar de smeltoven, zodat deze zo klein mogelijk moeten zijn, maar dat men de in de voorraad op te nemen boordstroken liefst zo breed mogelijk neemt, omdat daarin de geïnvesteerde arbeid behouden blijft en er bij volgende indelingen meer kombinatiemogelijkheid is.

Samenvattend zijn de drie belangrijkste wensen dus:

a. De voorraad beperken.

b. De hoeveelheid boordstrook die omgesmolten wordt beperken.

c. De hoeveelheid boordstrook die in de voorraad bewaard wordt zo weinig mogelijk versnipperen.

Men kan zich de ideale indeling ontstaan denken uit de eindoplossing van het probleem in 1° formulering, door op de aangesneden wikkels + voorraad met de te maken strips te gaan schuiven, totdat het optimum bereikt is. Dat de oplossing van eerste en tweede formulering in werkelijkheid niet optimaal is moge blijken uit volgend triviaal voorbeeld:

$s = 100$, $n = 2$, $M = 1$, boordstrookbreedte = 40;
$(f_1, b_1) = (2,30)$; $(f_2, b_2) = (2,5)$, boordstroken met breedte ≤ 10 worden weggegooid.
Het minimum van de kriteriumfunktie is 1: één volle wikkel wordt aangesneden. Er zijn echter verschillende eindoplossingen. De beste is degene die 1 × breedte 30 en 2 × breedte 5 op de voorraad indeelt.

De verbeterde oplossing wordt direkt verkregen als eindoplossing van het probleem in eerste formulering, door in plaats van coëfficiënten +1 of 0 in de kostenfunktie nieuwe coëfficiënten $c_1 \ldots c_{N+M}$ te nemen.
c_j hangt af van j en van de breedte van de boordstrook die overblijft bij kombinatie j en wel als volgt:
Zij $f(x)$ de funktie die de kosten als funktie van de breedte x van de boordstrook aangeeft. Deze is samengesteld uit 2 funkties: $g(x)$ en $h(x)$. Hiervan geeft $g(x)$ de kosten aan die gemaakt worden bij het weer omsmelten van een boordstrook van breedte x en $h(x)$ geeft de kosten aan bij het in de voorraad stoppen van een boordstrook.
Voor $f(x)$ geldt: $f(x) = g(x)$ als $x \leq B$ en $f(x) = h(x)$ als $x > B$.
B kan gekozen of berekend worden. Wanneer $g(x)$ en $h(x)$ bekend zijn kunnen voor alle kombinaties de kosten berekend worden.
Het oude optimalisatiekriterium om het aantal aangesneden wikkels te minimaliseren blijft echter het belangrijkste, hierdoor wordt immers ook de voorraad beperkt.
Dit kriterium is vervuld wanneer voor $j \leq N$ bij $f(x)$ een relatief groot getal c opgeteld wordt.
Het probleem kan nu slechts in de eerste formulering gegoten worden.
Laat b_j de boordstrookbreedte van de j-de kombinatie zijn, dan is de formulering (3^o formulering):

$$\text{minimaliseer: } \sum_{j=1}^{N} (f(b_j) + c)x_j + \sum_{j=N+1}^{N+M} f(b_j)x_j$$

$$\text{onder: } \sum_{i=N+N_{j-1}+1}^{N+N_j} x_i \leq p_j \;, \quad 1 \leq j \leq m$$

$$\sum_{i=1}^{N+M} k_{ij} x_i = f_j \ , \quad 1 \leqq j \leqq n$$

$$x_i \geqq 0, \text{ integer} \ , \quad 1 \leqq i \leqq N+M$$

Een nadeel van deze formulering t.o.v. de tweede formulering is dus, dat met een groter aantal variabelen gerekend moet worden, hetgeen de toepasbaarheid beperkt.

Een voordeel is een betere optimalisering.

§2. Formulering als Gegeneraliseerd Transportprobleem

Een transportprobleem is een L.P.-probleem, waarvan de bijvoorwaarden een bepaalde vorm hebben. Wij zullen het in korte trekken schetsen. Op m plaatsen is een produkt beschikbaar in bekende hoeveelheden a_i ($i = 1 \ldots m$) en op n bestemmingen worden hoeveelheden b_j ($j = 1 \ldots n$) gevraagd. De transportkosten van een eenheid van plaats i naar bestemming j bedragen c_{ij}. De beschikbare hoeveelheid is even groot als de gevraagde hoeveelheid:

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j.$$

Zij x_{ij} de hoeveelheid die van plaats i naar bestemming j getransporteerd wordt. Bij het minimaliseren van de totale transportkosten ontstaat volgend L.P.-probleem met mn variabelen en m+n voorwaarden:

$$\text{minimaliseer} \sum_{i,j} c_{ij}x_{ij}$$

$$\text{onder:} \sum_{j=1}^{n} x_{ij} = a_i \,, \quad i = 1 \ldots m$$

$$\sum_{i=1}^{m} x_{ij} = b_j \,, \quad j = 1 \ldots n$$

$$x_{ij} \geqq 0, \text{ integer.}$$

Een probleem van deze vorm heet een transportprobleem. Natuurlijk kan de simplex methode toegepast worden, maar door de speciale vorm van de m+n vergelijkingen (elke variabele komt in precies 2 vergelijkingen voor, met als coëfficiënt +1), is er een betere methode gevonden door A. Charnes en W.W. Cooper [3]. Dit zgn. stepping-stone algorithme werkt met een m×n tabel, waarin, evenals in de gewone simplex methode, steeds een variabele de basis verlaat, ten gunste van een andere, maar het aantal bewerkingen is zeer veel kleiner. Bovendien zijn optellen en aftrekken de enige operaties die uitgevoerd worden, waardoor alle variabelen automatisch integer blijven. Tenslotte komen veel grotere problemen voor oplossing op een komputer in aanmerking. Voor een uitvoerige beschrijving van het transportprobleem zie b.v. [4].

Het gegeneraliseerd Transportprobleem is een probleem dat als volgt geformuleerd kan worden:

$$\text{minimaliseer } \sum_{i,j} c_{ij} x_{ij}$$

$$\text{onder } \sum_{j=1}^{n} p_{ij} x_{ij} = a_i \ , \quad i = 1 \ldots m$$

$$\sum_{i=1}^{m} q_{ij} x_{ij} = b_j \ , \quad j = 1 \ldots n$$

Dit probleem verschilt dus van het transportprobleem, doordat in plaats van coëfficiënten +1 willekeurige getallen mogen optreden. Deze ontstaan door beperkingen in de voorradige hoeveelheid (eerste m vergelijkingen) en beperkingen in de vraag (laatste n vergelijkingen). Bij de meeste transportproblemen bestaan deze hierin, dat van plaats tot plaats verschillende veelvouden van de eenheid aangeboden en gevraagd worden.
Het stepping-stone algorithme kan gebruikt worden, in iets gewijzigde vorm (gegeneraliseerd stepping-stone algorithme genaamd), maar nu zijn in de eindoplossing niet noodzakelijk alle variabelen integer [4].

Het indeelprobleem van n karweien met frequentie f_i en breedte b_i kan geformuleerd worden als gegeneraliseerd transportprobleem. Dit ziet men het eenvoudigst door het probleem als het transport van opdrachten naar volle wikkels voor te stellen. Neem een aantal wikkels m, zodat $\sum_{i=1}^{n} f_i b_i \leq ms$ en m zó groot dat een indeling bestaat. Voer een extra karwei n+1 in, met breedte 1 en frequentie ms - $\sum_{i=1}^{n} f_i b_i$. Dit (n+1)ste karwei bevat dus het totaal aan boordstrook dat ontstaat bij indeling van de n karweien op m wikkels. Dan ontstaat het volgend transportprobleem:
Op n+1 plaatsen is een hoeveelheid $f_i b_i$ (i = 1 ... n+1) aanwezig, die vervoerd moet worden naar m bestemmingen, waar overal een hoeveelheid s gevraagd wordt. De totaal beschikbare hoeveelheid is gelijk aan de totaal gevraagde hoeveelheid.

Zij x_{ij} de hoeveelheid die van plaats i naar bestemming j vervoerd wordt. Het is duidelijk dat voor alle j x_{ij} een veelvoud van b_i moet zijn. Dit zijn de eerder genoemde beperkingen in de voorradige hoeveelheid, die van een gewoon transportprobleem een gegeneraliseerd transportprobleem maken. We herdefiniëren x_{ij} aldus: zij x_{ij} het aantal veelvouden van b_i dat van plaats i naar bestemming j gebracht wordt; nu zijn alle x_{ij} dus gewone integervariabelen.
De formulering van het probleem is: (4^o formulering):

$$\text{minimaliseer} \sum_{i,j} c_{ij} x_{ij}$$

$$\text{onder} \sum_{i=1}^{n+1} b_i x_{ij} = s \;, \quad j = 1 \ldots m$$

$$\sum_{j=1}^{m} x_{ij} = f_i \;, \quad i = 1 \ldots n+1$$

$$x_{i,j} \geqq 0, \text{ integer} \quad \forall\, i,j$$

De coëfficiënten c_{ij} in de kostenfunktie zijn nog niet nader gespecificeerd. We kiezen hiervoor

$$c_{ij} = c_j = j \text{ als } 1 \leqq i \leqq n \text{ en } c_{n+1,j} = 0.$$

Dit heeft tot resultaat dat in de eindoplossing de karweien ondergebracht worden op de laagstgenummerde wikkels, zodat het minimale aantal wikkels aangesneden wordt. Het optimalisatiekriterium van §1 (1^o en 2^o formulering) is dus vervuld.
Bovendien wordt een zekere optimalisering bereikt in de versnippering van de boordstroken, omdat de totale hoeveelheid boordstrook zoveel mogelijk naar de hoogstgenummerde wikkels verschoven wordt.

Evenals in §1 kan men hier ook rekenen met een voorraad boordstroken. Laten er in totaal a boordstroken zijn met breedtes $s_1 \ldots s_a$. Dan komen er a vergelijkingen bij en bovendien $a \cdot (n+1)$ variabelen x_{ij}. We stellen $c_{ij} = 0$ wanneer j bij een boordstrook hoort. Voor de duidelijkheid hernummeren we boordstroken + wikkels: van 1 tot a zijn boordstroken en van a+1 tot a+m zijn wikkels. Dan is de formulering:

$$\text{minimaliseer} \sum_{i=1}^{n+1} \sum_{j=1}^{a+m} c_{ij} x_{ij}$$

$$\text{onder:} \sum_{i=1}^{n+1} b_i x_{ij} = s_j \ , \quad j = 1 \ldots a{+}m \ (s_{a+1} = \ldots = s_{a+m} = s)$$

$$\sum_{j=1}^{a+m} x_{ij} = f_i \ , \quad i = 1 \ldots n{+}1$$

$$x_{i,j} \geqq 0, \text{ integer} \qquad \forall\, i,j$$

Opgemerkt dient te worden, dat m slechts zó groot gekozen hoeft te worden, dat men zeker is van een indeling. De waarde van m kan men dus verminderen, wanneer een voorraad boordstroken in de berekeningen betrokken wordt, zodat, in tegenstelling tot §1, het probleem niet veel groter wordt.

Ook in dit geval kan, door van gelijkheden op ongelijkheden over te gaan, de grootte van het probleem beperkt worden. Dit wordt bereikt door het (n+1)ste karwei geheel weg te laten, waardoor één vergelijking en a+m variabelen afvallen. De reduktie is dus minder groot als in §1. De formulering wordt nu (5° formulering):

$$\text{minimaliseer} \sum_{i=1}^{n} \sum_{j=1}^{a+m} c_{ij} x_{ij}$$

$$\text{onder:} \sum_{i=1}^{n} b_i x_{ij} \leqq s_j \ , \quad j = 1 \ldots a{+}m \ (s_{a+1} \ldots = a_{a+m} = s)$$

$$\sum_{j=1}^{a+m} x_{ij} = f_i \ , \quad i = 1 \ldots n{+}1$$

$$x_{ij} \geqq 0, \text{ integer} \qquad \forall\, i,j$$

§3. Ad-hoc Methodes

A. Kombinaties.

Gegeven zijn n paren getallen (f_i, b_i) waarvan de eerste komponent de frequentie en de tweede de breedte van een karwei aangeeft.
Een n-dimensionale vektor $\vec{a} = (a_1 \ldots a_n)$, waarvoor geldt:

$$(1) \qquad \sum_{i=1}^{n} a_i b_i \leqq s, \quad 0 \leqq a_i \leqq f_i, \quad a_i \text{ geheel } (i = 1 \ldots n)$$

heet een kombinatie. De i-de komponent geeft aan hoeveel exemplaren van karwei i in de kombinatie vertegenwoordigd zijn. De kombinaties vormen een volledig geordende verzameling wanneer ze lexicografisch geordend worden. Dit betekent het volgende: Laten $\vec{a} = (a_1 \ldots a_n)$ en $\vec{b} = (b_1 \ldots b_n)$ twee kombinaties zijn. Zij j de eerste komponent waarin $\vec{a}$ en $\vec{b}$ verschillen. Dan is $\vec{a} < \vec{b}$ als $a_j < b_j$ en $\vec{a} > \vec{b}$ als $a_j > b_j$.
Als $\vec{a}$ en $\vec{b}$ niet identiek zijn geldt dus altijd: òf $\vec{a} < \vec{b}$ òf $\vec{a} > \vec{b}$.
De kleinste kombinatie is de nulvektor, de grootste is de kombinatie die zoveel mogelijk van het eerste karwei bevat, daarna zoveel mogelijk van het tweede karwei enzovoorts. Wanneer een kombinatie $\vec{a} = (a_1 \ldots a_n) \neq \vec{0}$ gegeven is, kan de eerstvolgende die kleiner is eenvoudig gevonden worden. Zij a_j de laatste komponent $\neq 0$. Als $j = n$ is $(a_1 \ldots a_{n-1}, a_n - 1)$ de gezochte en als $j < n$ is het de grootste uit de deelverzameling:

$$\bigcup_{x_{j+1} \ldots x_n} (a_1, \ldots, a_{j-1}, a_j - 1, x_{j+1}, \ldots, x_n).$$

Deze wordt als volgt gevonden: De eerste j komponenten zijn uiteraard $a_1 \ldots a_{j-1}, a_j - 1$; maak daarna de (j+1)ste komponent zo groot mogelijk, daarna de volgende, enz.
Wanneer men, gegeven (f_i, b_i) $i = 1 \ldots n$, het totale aantal kombinaties wil bepalen, kan men ze allemaal aflopen, b.v. beginnende bij de grootste. Dit is de enige manier om dit aantal te weten te komen, omdat er geen kortere formule bestaat.

B - Eerste methode (MA)

De eerste ad-hoc methode is een zeer eenvoudige. De volgende kombinatie wordt gekonstrueerd: men neemt van het eerste karwei zoveel mogelijk, vult de resterende boordstrook zoveel mogelijk met het tweede karwei aan, enzovoorts. De zo verkregen kombinatie is niets anders als de grootste volgens de lexicografische ordening. Deze kombinatie wordt uitgevoerd, d.w.z. men brengt hem in mindering van het totale orderpakket. Daarna wordt steeds weer van het overblijvende orderpakket de grootste kombinatie uitgevoerd, totdat alle karweien uitgevoerd zijn. De methode MA is met name effectief, wanneer de karweien volgens breedte gerangschikt zijn, dus $b_1 > b_2 > \dots > b_n$. Dan worden de kleinere breedtes gebruikt als opvulling bij de grotere breedtes. De methode is eenvoudig met de hand uit te voeren en leidt altijd tot een oplossing, maar de zo verkregen indeling hoeft niet de beste te zijn; dit zal vooral het geval zijn, wanneer het aantal kleine breedtes in het orderpakket klein is. In tabel X staan de oplossingen voor problemen I t/m IV volgens methode MA.

C - Tweede Methode (MB)

Deze is een verbetering van de vorige. De grootste kombinatie wordt opgezocht (de karweien zijn volgens breedte gerangschikt), doch deze wordt nu slechts gemaakt, wanneer de benutte breedte B een minimumwaarde M haalt. Als $B < M$ is, wordt de op één na grootste kombinatie opgezocht, en weer wordt getest of de benutte breedte $\geq M$ is enzovoorts.
Dit proces houdt dus in: de grootste kombinatie, waarvan de benutte breedte groter of gelijk is aan een bepaalde waarde M, wordt gemaakt en in mindering gebracht van het orderpakket. Daarna doet men hetzelfde met het gereduceerde orderpakket.
Aan methode MB zijn twee bezwaren verbonden:
a) er blijft in de regel iets over van het orderpakket
b) het is moeilijk om een goede keuze voor de grenswaarde te doen.
Het eerste bezwaar kan opgeheven worden, door het resterende orderpakket volgens de methode MA in te delen, of bij het orderpakket voor de volgende indeling te voegen.

Het tweede bezwaar kan bij kleine orderpakketten tot moeilijkheden leiden: Zo wordt b.v. bij $s = 380$, $n = 2$, $(f_1, b_1) = (3,130)$, $(f_2, b_2) = (9,80)$, $M = 375$ helemaal niets ingedeeld volgens MB, zodat verdergaand met MA 4 wikkels aangesneden worden, met een nuttige rest van 300 en 60 (boordstroken met breedte ≤ 50 worden weer omgesmolten). Bij de keuze $M = 370$ worden echter slechts 3 wikkels versneden, zonder resterende boordstroken.

§4. Resultaten

Vier problemen zijn met de verschillende methodes opgelost, d.w.z. volgens §1, 2^o formulering, volgens §2, 5^o formulering en §3 MA en MB. De problemen konden niet volgens de derde formulering opgelost worden, omdat onvoldoende gegevens aanwezig waren om de funktie f(x) redelijk te schatten.
De 5^o formulering, die ogenschijnlijk niet oneconomischer werkt als de 2^o formulering, kwam niet binnen redelijke tijd tot een oplossing, waar de 2^o formulering dit wel deed. De reden daarvan is dat in de 2^o formulering de som van de variabelen in de eindoplossing gelijk is aan het aantal aangesneden wikkels m, maar in de 5^o formulering is deze som gelijk aan het aantal te maken strips $\sum_i f_i$, wat meestal veel groter als m is, zodat hier veel meer vertakkingsmogelijkheid aanwezig is voor de Land en Doig methode. Bovendien is de grootte van het probleem (= aantal vergelijkingen × aantal variabelen) in de 5^o formulering niet per sé kleiner als in de 2^o formulering (zie Tabel XII).
Probleem II kon niet volgens formulering 2 opgelost worden, omdat het te groot is. Methode MB geeft echter een bevredigend resultaat, evenals bij de drie andere problemen.

De problemen staan getabelleerd in tabel II, IV, VI en VIII. Het programma JVF 010467 bepaalt het totale aantal kombinaties en het aantal volle kombinaties, bij verschillende breedtes. De resultaten daarvan zijn vermeld in de tabellen III, V, VII en IX. Daaruit blijkt de enorme reduktie in het aantal variabelen, wanneer, in plaats van met de 1^o (of 3^o) formulering, met de tweede gewerkt wordt.
Het programma JVF 200267 maakt de getallenband met volle kombinaties, die nodig is als invoer voor het Land en Doig programma.
In tabel X zijn de aantallen wikkels vermeld, die bij de verschillende oplossingsmethoden aangesneden worden. Steeds werd zonder voorraad gerekend.
In tabel XI is de bij de diverse oplossingsmethoden gebruikte machinetijd vermeld.
Tabel XII tenslotte geeft de grootte van het probleem en de som van de variabelen in de eindoplossing, bij de 2^o en 5^o formulering.

Tabel I: Maximale omvang voor het Land en Doig programma van Anthonisse (overgenomen uit (1))

bijvoorwaarden	variabelen
10	830
20	480
30	300
40	180
50	120
60	60

Tabel II: Probleem I

karwei	frequentie	breedte
1	2	75
2	5	30
3	4	20
4	8	15
5	10	9
6	15	5

Tabel III: Aantal variabelen van probleem I bij verschillende boordstrookbreedtes

Breedte	1^{o} form.	2^{o} form.
100	1054	179
90	723	135
80	478	95
70	304	65
60	186	45
50	107	28
40	57	17
30	28	10

Opmerking: het aantal ongelijkheden bij het oplossen volgens de tweede formulering is gelijk aan het aantal karweien, vermeerderd met het aantal verschillende breedtes in de voorraad.

Tabel IV: Probleem II

karwei	frequentie	breedte
1	1	315
2	4	250
3	1	210
4	3	203
5	13	200
6	1	190
7	4	163
8	5	135
9	8	90
10	18	83
11	3	80
12	8	79
13	5	72
14	10	70
15	8	60
16	52	40
17	16	38
18	12	27
19	8	26
20	16	19

Tabel V: Aantal variabelen van probleem II bij verschillende breedtes (zie opmerking onder tabel III)

Breedte	1° form.	2° form.
380	98.548	30.335
340	44.864	14.529
300	19.391	6.641
260	7.878	2.268
220	2.974	1.158
180	1.016	428
140	309	142
100	79	40
60	15	9

Tabel VI: Probleem III

karwei	frequentie	breedte
1	1	300
2	1	200
3	2	130
4	10	60
5	6	31
6	6	26
7	8	24

Tabel VII: Aantal variabelen van probleem III bij verschillende boordstrookbreedtes (zie opmerking onder tabel III)

Breedte	1^o form.	2^o form.
380	1634	326
340	1119	243
300	735	179
260	461	126
220	266	81
180	142	49
140	66	27
100	26	13
60	9	6

Tabel VIII: Probleem IV

karwei	frequentie	breedte
1	15	160
2	13	110
3	25	32
4	40	18

Tabel IX: Aantal variabelen van probleem IV bij verschillende boordstrookbreedtes (zie opmerking onder tabel III)

Breedte	1^o form.	2^o form.
380	329	43
340	238	34
300	166	25
260	115	20
220	74	14
180	46	10
140	25	6
100	13	4
60	5	2

Tabel X: Oplossingen van de vier problemen volgens 2^o formulering, MA en MB

probleem	1	2	3	4[1)]	5
I	665	6.65	7	7	7
II	14401	37.9		38*	40
III	1894	4.98	5	5	6
IV	5350	14.08	15	15	15

kolom 1: totale breedte van de karweien
2: totale breedte/wikkelbreedte
3: aantal gebruikte wikkels bij 2^o formulering
4: aantal gebruikte wikkels met MB
5: aantal gebruikte wikkels met MA

* Hier werd met toegelaten rest 2 gewerkt; bij toegelaten rest 5 blijft juist één strip van breedte 19 over, zodat dan 39 wikkels aangesneden moeten worden.

1) Probleem I: toegelaten rest 2; de anderen: toegelaten rest 5.

Tabel XI: Gebruikte machinetijd

	1	2	3
I	3 min.	1 uur*	4 sec.
II			7 sec.
III	$7\frac{1}{2}$ min.	30 min.*	4 sec.
IV	1 min.		4 sec.

kolom 1: 2^{o} formulering
2: 5^{o} formulering
3: MB

*: niet tot een oplossing gekomen in de aangegeven tijd.

Tabel XII: Grootte van de vier problemen in de 2^{o} en 5^{o} formulering, benevens de som van de niet-nul zijnde variabelen.
De Grootte = aantal vergelijkingen × aantal variabelen

probleem	I		II		III		IV	
2^{o} form.	1074	7	30.335	38	2282	5	172	15
5^{o} form.	630	44	46.400	196	630	44	1140	93

Literatuur

(1) J.M. Anthonisse - Voorlopige invoerbeschrijving van Algolprogramma's voor gemengde of zuivere, continue geheeltallige en nul-een programmering.
Rapport M.C. - maart 1967.

(2) A. Land en A. Doig - An automatic method of solving discrete programming problems.
Econometrica 28 (1960) 497-520.

(3) A. Charnes en W.W. Cooper - The stepping-stone method of explaining linear programming calculations in transportation problems.
Management Science 1 (1954).

(4) G. Hadley - Linear Programming, Hoofdstuk 9: Transportation Problems.
Addison Wesley.