

Aggregatie versus decompositie bij een
stochastisch één-machine-meer-produkten probleem

R.P.H.G. Bemelmans en J. Wijngaard

Technische Hogeschool Eindhoven

Afdeling Bedrijfskunde

Samenvatting

Na eerst de vorm van het door ons beschouwde één-machine-meer-produkten probleem nader omschreven te hebben, zullen we ingaan op aggregatie- en decompositiestrategieën. Met behulp van simulatie worden vervolgens de aldus te verkrijgen strategieën met elkaar en met een ondergrens voor de kosten bij de optimale strategie vergeleken.

Belangrijke begrippen

Aggregatie, decompositie, één-machine-meer-produkten probleem.

1. Inleiding

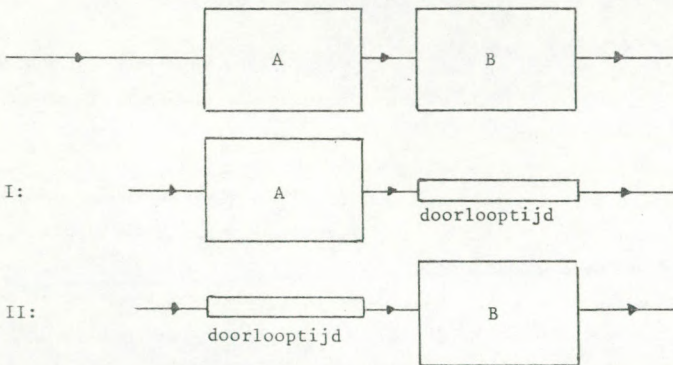
Bij bestudering van zeer complexe problemen zal men al snel met onoplosbaarheid, onzekerheid en ondoorzichtigheid van het beschouwde model te maken krijgen. Men zal daarom geneigd zijn vereenvoudigingen aan te brengen in het model. De twee belangrijkste methodes die ons daartoe ter beschikking staan zijn aggregatie en decompositie. Wismer schrijft in het voorwoord van "Optimization methods for large scale systems ... with applications" [11]:

"We have confined our attention to those developments which are applicable to large-scale systems. As a rule such developments are characterized by (1) decomposition of the large system into many smaller subsystems which are later composed or coordinated to reconstruct the original system or by (2) aggregation of the variables in the large system thereby reducing its dimensionality. The ultimate gain from these methods is not only to obtain a computer solution in a reasonable time (or at all) but also to aid in the conceptualization and understanding of large-scale interactions."

Voor aggregatie voegen Hax en Golovin in het artikel "Hierarchical Production Planning Systems" [3] een derde belangrijk voordeel toe, namelijk de grotere betrouwbaarheid van uitspraken bij voorspellingen. Zie voor verdere algemene beschouwingen over aggregatie onder andere Liesegang [7].

In dit rapport gaat het over aggregatie en decompositie in een productieplanningsprobleem. Bij productieplanningsproblemen gaat het uiteindelijk om het afstemmen van de productie en de afzet op elkaar. De afzet betreft afzet van individuele produkten, de productie moet gerealiseerd worden door de inzet van allerhande resources, die slechts beperkt beschikbaar zijn. Resources kunnen zijn materialen, mankracht, machines, gereedschappen, etc. Aan deze afstemming zijn verschillende handelingen te onderscheiden: Bepaling van de afzet (verkooppolitiek), Verwerving van de resources en Inzet van de resources. Bij productieplanning gaat het vooral om de inzet van de resources ten einde in een gegeven vraag te voorzien. Van de andere twee aspecten wordt afgezien. Het bestaan van een productieplanningsprobleem is dus in feite het resultaat van een soort van decompositie. Maar ook het productieplanningsprobleem op zich vormt in de regel nog een zeer complex probleem. Deze complexiteit wordt veroorzaakt door het feit dat we te maken hebben met meer-

dere resources van verschillende soorten. Een aspect dat in veel productieplanningssituaties voorkomt is dat meerdere produkten met elkaar concurreren om toewijzing van één bepaalde, in beperkte mate aanwezige resource. Het is dit aspect dat we in dit rapport hebben geïsoleerd in de vorm van een één-machine-meer-produkten probleem. Dit is een probleem dat op zich al interessant is aangezien het ook in tamelijk pure vorm in de praktijk voorkomt. Denk bijvoorbeeld aan een glasfabriek die vaak voor een groot deel bestaat uit één oven. In dergelijke gevallen valt het productieplan samen met het plan voor deze ene machine, terwijl de rest zich aanpast. In andere gevallen met meer gelijkwaardige capaciteiten geldt dat natuurlijk niet, maar ook daar kan binnen het proces van het bepalen van een goede strategie het aanpakken van het één-machine-meer-produkten probleem zinvol zijn. Dit is dan het resultaat van decompositie. Denk bijvoorbeeld aan het geval waarin we een productiefase A hebben die vooraf gaat aan een gelijkwaardige productiefase B. We zouden nu machine A als een één-machine-meer-produkten probleem kunnen beschouwen, waarbij productiefase B als een doorlooptijd wordt weergegeven en andersom.



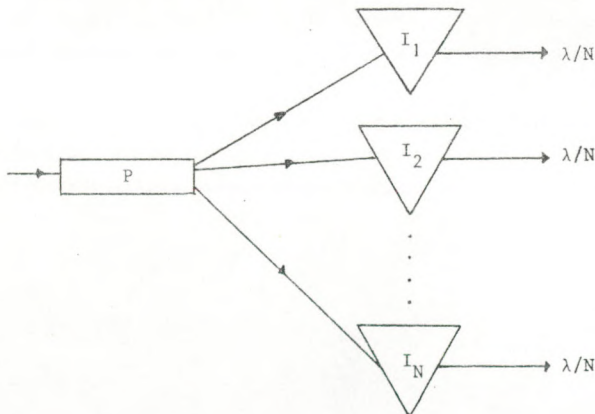
Achteraf kunnen we de met behulp van deze twee problemen gevonden productie-schema's met elkaar vergelijken en, waar nodig, op elkaar afstemmen.

Het één-machine-meer-produktenprobleem is ook uit theoretisch oogpunt interessant. Het kent een zeer lange geschiedenis. Voor een overzicht zie S.E. Elmaghraby [2]. Nieuwerwerk is dat van P. van Beek [1], P. van Schuylenburg [8], F. Lansdorp en J.W.M. Bertrand [6], Th.H.B. Hendriks en J. Wessels [4], M. Starremans en J. Wessels [9] en J. Wessels en N. Thijssen [10].

Door P. van Beek en P van Schuylenburg wordt een decompositie aanpak beschreven. Wessels/Thijssen, Starremans/Wessels en Hendriks/Wessels gaan uit van een indeling in groepen. Eén van de weinige analyses van het stochastische probleem wordt gegeven in Bertrand/Lansdorp. Ook in dit rapport zullen we een specifiek stochastisch model beschouwen. We gaan uit van identieke produkten waarvan de vraag Poisson verdeeld is. Voorraadkosten en stock-out kosten worden aangenomen lineair te zijn. Produktie vindt plaats in batches ter grootte q . Het is daarom niet nodig omstelkosten mee te nemen. Een beperkte capaciteit realiseren we door een factor p in te voeren die aangeeft met welke kans een produktierun gestart kan worden. Wij zullen kijken naar de kwaliteit van besturingsmethodes die gebaseerd zijn op aggregatie over en decompositie naar de produkten. Daartoe behandelen we in sectie 2 eerst het model uitvoeriger. Vervolgens gaan we in de secties 3, 4 en 5 nader op aggregatie- en decompositiemethodes in. Om enig inzicht te krijgen in de kwaliteit van aggregatie- en decompositiemethodes leiden we in sectie 6 een ondergrens af voor de kosten van de optimale strategie. In sectie 7 bepalen we voor een special geval numerieke resultaten alvorens we in sectie 8 ingaan op resultaten die we met behulp van simulatie gevonden hebben. Tenslotte trachten we in sectie 9 nog enige aanwijzingen voor mogelijk toekomstig onderzoek op dit gebied te formuleren.

2. Modelformulering

Beschouw het één-machine-meer-produkten systeem



Zeg we hebben N verschillende produkten. Klanten komen binnen volgens een Poissonproces met parameter λ en bestellen bij binnenkomst met kans $\frac{1}{N}$ een eenheid van produkt j (voor $j=1,2,\dots,N$). Deze veronderstelling is zeer redelijk indien we veel klanten hebben die onafhankelijk van elkaar bestellen.

We geven de voorraad van produkt 1 tot en met produkt N aan met respectievelijk $I_1, I_2, \dots, I_N$. We staan ook nalevering toe, dat wil zeggen de voorraad van een produkt kan negatief zijn.

In ons model zullen we geen produktiekosten meenemen, omdat we ervan uitgaan dat de gevraagde produkten toch ooit geproduceerd worden, maar wel voorraadkosten in de vorm van

$$\sum_{j=1}^N (aI_j^+ + bI_j^-) \quad (2.1)$$

waarbij geldt $I^+ = \max(0, I)$, $I^- = \max(0, -I)$.

Als er produktie plaatsvindt dan zullen we een hoeveelheid q produceren, terwijl een doorlooptijd d in acht wordt genomen. We zullen te maken hebben met een beperkte capaciteit die we als volgt introduceren: Indien we beslissen dat we willen produceren dan zal die produktie met kans p inderdaad aanvangen en met kans $1-p$ niet.

Er zullen nu bepaalde tijdstippen zijn waarop we de beslissing willen nemen om tot produktie over te gaan. Het zal duidelijk zijn dat indien op een bepaald tijdstip geen klant binnenkomt, dat wil zeggen geen der voorraden wordt verlaagd, men niet geneigd zal zijn een produktierun te starten (dit is een gevolg van de aanname dat het aankomstproces van de klanten een Poissonproces is). We formuleren daarom de volgende afhankelijkheid: de tijdstippen waarop we mogen beslissen te produceren vallen samen met binnenkomsttijdstippen van klanten.

Merk tenslotte nog op dat het geval $p=1$ overeenkomt met een onbeperkte capaciteit. Immers telkens wanneer een klant binnenkomt kunnen we het door hem gevraagde produkt maken, zodat we nooit hoeven te kiezen tussen verschillende produkten.

3. Aggregatie versus decompositie

We zoeken een strategie die aangeeft wanneer we welk produkt moeten produceren zodanig dat de verwachte kosten per tijdseenheid minimaal zijn. We kunnen dit formuleren als een stochastisch dynamisch programmeringsprobleem. Indien we dit probleem willen oplossen dan zullen, voor niet te kleine N , de numerieke problemen zeer groot worden, terwijl de gevonden strategie zeer ondoorzichtig wordt. We zullen daarom geneigd zijn benaderende methodes te gebruiken. Er bestaan daartoe twee duidelijk verschillende mogelijkheden:

- Aggregatie aanpak. Hierbij tellen we eerst alle voorraden bij elkaar op, beslissen op grond van deze totale voorraad of produktie gewenst is en, pas als dit het geval is wijzen we het produkt aan dat we willen produceren.
- Decompositie aanpak. Hierbij beslissen we op grond van het individuele produkt of hiervoor produktie moet plaatsvinden. Als mocht blijken dat we meerdere produkten willen maken dan kiezen we één van deze.

We zullen nu eerst nader op deze beide methodes ingaan, waarbij we opmerken dat indien we produceren altijd dat produkt gemaakt wordt waarvan de economische voorraad het laagste is, omdat we met identieke produkten te maken hebben. Indien meerdere produkten de laagste economische voorraad bezitten dan loten we tussen deze.

Opmerking: onder de economische voorraad verstaan we de voorraad + de hoeveelheid produkten die nog geleverd moeten worden.

4. Aggregatie

Beschouw alleen de totale voorraad I . Zij I_{ec} de economische voorraad. We zullen ons bij de aggregatie bezighouden met het zoeken naar een optimale strategie niet binnen de klasse van alle mogelijke strategieën (S) maar binnen de klasse S_{tot} van strategieën met de gedaante:

$$\text{produceer als } I_{ec} \leq I_0 \quad (\text{voor zekere } I_0 \in \mathbb{Z})$$

Dat wil zeggen we produceren als de totale voorraad een bepaalde grens onderschrijdt.

We beschouwen dus in plaats van:

$$\min_{S \in \mathcal{S}} E_S \limsup_{T \rightarrow \infty} \frac{1}{T} \int_0^T \sum_{j=1}^N (aI_j^+(t) + bI_j^-(t)) dt \quad (4.1)$$

het volgende:

$$\begin{aligned} \min_{S \in \mathcal{S}_{tot}} E_S \limsup_{T \rightarrow \infty} \frac{1}{T} \int_0^T \sum_{j=1}^N (aI_j^+(t) + bI_j^-(t)) dt &= \\ = \min_{S \in \mathcal{S}_{tot}} \limsup_{T \rightarrow \infty} \frac{1}{T} \int_0^T E_S \left[\sum_{j=1}^N (aI_j^+(t) + bI_j^-(t)) \right] dt & \end{aligned} \quad (4.2)$$

Laat nu $P_S(I)$ de kans zijn dat de voorraad gelijk is aan I in de stationaire situatie indien we strategie S uit $\mathcal{S}_{tot}$ volgen. Op de existentie en vorm van $P_S(I)$ zullen we later terugkeren. We kunnen dan (4.2) verder uitwerken. Als we met I_j de voorraad van produkt j aangeven in de stationaire situatie, dan geldt

$$\begin{aligned} \min_{S \in \mathcal{S}_{tot}} \limsup_{T \rightarrow \infty} \frac{1}{T} \int_0^T E_S \left[\sum_{j=1}^N (aI_j^+(t) + bI_j^-(t)) \right] dt &= \\ = \min_{S \in \mathcal{S}_{tot}} \limsup_{T \rightarrow \infty} \frac{1}{T} \int_0^T E_S \left[\sum_{j=1}^N (aI_j^+ + bI_j^-) \right] dt &= \\ = \min_{S \in \mathcal{S}_{tot}} E_S \left[\sum_{j=1}^N (aI_j^+ + bI_j^-) \right] &= \\ = \min_{S \in \mathcal{S}_{tot}} \int_{-\infty}^{\infty} E_S \left[\sum_{j=1}^N (aI_j^+ + bI_j^-) \mid I \right] dP_S(I) & \end{aligned}$$

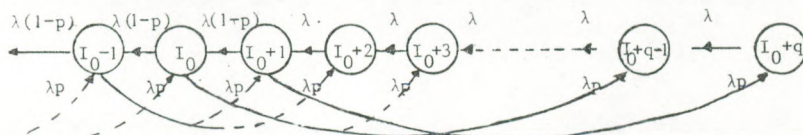
Als nu $g_S(I) := E_S \left[\sum_{j=1}^N (aI_j^+ + bI_j^-) \mid I \right]$ dan vinden we uiteindelijk dat

(4.2) overeenkomt met:

$$\min_{S \in \mathcal{S}_{tot}} \int_{-\infty}^{\infty} g_S(I) dP_S(I) \quad (4.3)$$

Wij hebben in het bovenstaand gesproken over stationaire kansen. Nu zullen we aantonen dat deze inderdaad bestaan terwijl we tevens nader op de vorm ervan ingaan.

Zij de strategie S weergegeven door: produceer als $I_{ec} \leq I_0$. Voor $d = 0$ zal dan gelden:



$$I \geq I_0 + q + 1 : P_S(I) = 0$$

$$I_0 + q \geq I \geq I_0 + 1 : P_S(I) = P_S(I+1) + pP_S(I-q+1)$$

$$I_0 \geq I : P_S(I) = (1-p) P_S(I+1) + pP_S(I-q+1)$$

Indien we overgaan op $x = I_0 + q - I$ dan krijgen we een wachttijdprobleem waarbij behandeling in batches plaatsvindt. Als we bovendien $P(I)$ in plaats van $P_S(I)$ schrijven terwille van de notatie dan krijgen we de volgende vergelijkingen:

$$x < 0 : P(x) = 0$$

$$0 \leq x \leq q - 1 : P(x) = P(x-1) + pP(x+q-1)$$

$$q \leq x : P(x) = (1-p)P(x-1) + pP(x+q-1)$$

De oplossing voor dit probleem luidt:

$$1) p = 1 : P(0) = P(1) = \dots = P(q-1) = \frac{1}{q}, P(x) = 0 \text{ anders} \quad (4.4)$$

$$2) 0 < p < 1 : 0 \leq x \leq q - 1 \Rightarrow P(x) = \frac{cp(z_0^{-q} - pz_0^{-2q+1})}{1-p} + pc \sum_{k=0}^{x-1} z_0^{-q-k}$$

$$q \leq x \Rightarrow P(x) = c z_0^{-x}$$

waarbij $c > 0$ een normeringsconstante is

en $z_0 > 1$ voldoet aan $f_p(z_0) = 0$ als geldt

$$f_p(z) = (1-p)z^q - z^{q-1} + p. \quad (4.5)$$

Het is duidelijk dat indien een dergelijke z_0 bestaat het aldus gedefinieerde stelsel $\{P(x)\}_{x=0}^{\infty}$ voldoet aan de genoemde voorwaarden, terwijl

$$P(x) \geq 0 \text{ en } \sum_{x=0}^{\infty} P(x) = 1 \text{ mits}$$

$$c = \left\{ \frac{qp(z_0^{-q} - pz_0^{-2q+1})}{1-p} + p \cdot \frac{(q-1)z_0^{-q+2} - qz_0^{-q+1} + z_0^{-2q+1}}{(z_0-1)^2} + \frac{z_0^{-q+1}}{z_0-1} \right\}^{-1} \quad (4.6)$$

Rest ons aan te tonen dat een dergelijke z_0 bestaat. Dit volgt echter uit de continuïteit van f_p (NB $p > 0$ volgt uit $p \cdot q > 1$) gecombineerd met:
- voor z voldoende groot geldt $f_p(z) > 0$

$$- \left(\frac{\partial}{\partial z} f_p(z) \right)_{z=1} = ((q-p)z^{q-1} - (q-1)z^{q-2})_{z=1} = 1 - qp < 0 \text{ immers}$$

$p \cdot q > 1$ zodat voor $\epsilon > 0$ voldoende klein: $f_p(1+\epsilon) < 0$ (want $f_p(1) = 0$)

Indien $d > 0$ dan kunnen we dit resultaat als volgt gebruiken om de vorm van de stationaire verdeling af te leiden:

Gegeven I_0 heeft d geen invloed op het gedrag van I_{ec} . Dat betekent dat we de stationaire verdeling van I_{ec} met bovenstaande kunnen bepalen, waarna geldt:

$$\begin{aligned} P(I(t) = I) &= \sum_{k=0}^{\infty} P(I_{ec}(t-d) = k+I) \cdot \frac{(\lambda d)^k}{k!} e^{-\lambda d} = \sum_{k=0}^{I_0+q-I_0} P(I_{ec}(t-d) = k+I) \cdot \frac{(\lambda d)^k}{k!} e^{-\lambda d} \\ &= (k+I) \cdot \frac{(\lambda d)^k}{k!} e^{-\lambda d} \end{aligned} \quad (4.7)$$

Dit betekent dat indien we $g_S(I)$ voor iedere strategie S uit S_{tot} kennen we de oplossing van (4.3) gevonden hebben. Het bepalen van de $g_S(I)$ is zeer moeilijk. Simulatie staat ons hiervoor ter beschikking, maar dit

is een tijdrovende oplossing. We hebben liever een methode om de gezochte strategie numeriek te bepalen. Daartoe benaderen we

$$\begin{aligned} g_S(I) &= F_S \left[\sum_{j=1}^N (aI_j^+ + bI_j^-) \mid I \right] \\ &= E_S \left[a \sum_{j=1}^N (I_j^+) + b \sum_{j=1}^N (I_j^-) \mid I \right] \end{aligned}$$

door

$$\begin{aligned} &E_S \left[a \left(\sum_{j=1}^N I_j \right)^+ + b \left(\sum_{j=1}^N I_j \right)^- \mid I \right] \\ &= aI^+ + bI^- \end{aligned} \quad (4.8)$$

Indien we (4.8) als benadering voor $g_S(I)$ nemen dan vinden we een benaderende aggregaatstrategie langs numerieke weg.

In sectie 8 gaan we zowel in op de kwaliteit van de optimale als de benaderende aggregaatstrategie.

5. Decompositie

In de decompositie aanpak zullen we voor produkt j beslissen dat we willen produceren indien de economische voorraad van dit produkt kleiner of gelijk is aan een zekere I_0 . Vanwege het identieke gedrag van de produkten zal deze I_0 niet afhankelijk zijn van j . Dit betekent dat we tot produktie van enig produkt zullen overgaan als de kleinste voorraad een bepaalde grens overschrijdt. We zoeken nu dus naar een optimale strategie binnen de klasse $\mathcal{C}_{min}$ van strategieën met de gedaante:

$$\text{produceer als } \min_j \{I_{ec,j}\} \leq I_0$$

Ook hier kunnen we met behulp van simulatie voor iedere willekeurige strategie de kosten benaderen. In sectie 8 wordt dat ook gedaan. Om dezelfde reden als in het geval van aggregatie zoeken we nu een benaderende decompositiestrategie die we numeriek kunnen bepalen. De benadering die we kiezen ligt daarin dat we ervan uit gaan dat indien we voor een bepaald produkt besliss-

6. Ondergrens kosten

Zoals we reeds vermeld hebben levert het oplossen van het oorspronkelijke stochastisch dynamisch programmeringsprobleem zeer grote praktische problemen. Toch willen we graag enig inzicht in de kwaliteit van aggregatie en decompositie trachten te krijgen. Daarom zullen we een ondergrens van de kosten afleiden voor de optimale strategie, zodat we de kosten bij aggregatie en decompositie hiermee kunnen vergelijken:

- Stel we volgen de optimale strategie, echter de kosten als de voorraad van produkt j gelijk is aan I_j . $\forall j = 1, 2, \dots, N$ zijn nu niet $\sum_{j=1}^N (aI_j^+ + bI_j^-)$ maar $a(\sum_{j=1}^N I_j)^+ + b(\sum_{j=1}^N I_j)^-$. Dan zijn de kosten in iedere toestand kleiner of gelijk aan de oorspronkelijke kosten. Dat betekent dat indien we over gaan op deze nieuwe kostenfunctie de optimale strategie een verwachting van de kosten op zal leveren die een ondergrens is voor het oorspronkelijke probleem. Dit is echter precies de kostenfunctie die wij bij het bepalen van de benaderende aggregaat strategie hebben gebruikt, zodat de hierbij berekende kosten een ondergrens vormen.
- Bij het bepalen van de benaderende decompositiestrategie namen we aan dat we nooit hoefden te kiezen tussen een aantal produkten, die we willen maken. In werkelijkheid is dat wel het geval. Dat betekent dat ook

$N \cdot \int_{-\infty}^{\infty} (aI^+ + bI^-) dP(I)_{j,d}$ een ondergrens voor de kosten vormt als $P_{j,d}$ de bij decompositie bepaalde stationaire kansen zijn in het geval dat de doorlooptijd gelijk is aan d .

7. Bijzonder geval $p = 1$

In het geval $p = 1$ hebben we te maken met een onbeperkte capaciteit. Het aantal bereikbare toestanden (in de stationaire situatie) zal nu eindig zijn. Dit geeft ons de mogelijkheid om in dit geval numerieke resultaten te bepalen, die bij grote overcapaciteit als benadering kunnen dienen en bij de simulatie een referentiepunt vormen.

Beschouw alleen $d = 0$ ($d > 0$ kan op de reeds vermelde wijze hieruit worden afgeleid).

- Aggregatie.

Beschouw toestanden waarvoor geldt $\sum_{j=1}^N I_j - I_0 = 1, 2, \dots, Q - 1$ of Q .

In het stationaire geval zal de kans op toestanden buiten deze klasse gelijk zijn aan nul.

Als $1 \leq \sum_{j=1}^N I_j - I_0 < Q$ dan

$$P(I_1, I_2, \dots, I_N) = \frac{1}{N} \sum_{j=1}^N P(I_1, I_2, \dots, I_j+1, \dots, I_N) \quad (7.1)$$

immers we kunnen niet in toestand $(I_1, I_2, \dots, I_N)$ komen doordat q eenheden binnenkomen. Dit betekent dat we alleen de kansen op toestanden in het hypervlak $\sum_{j=1}^N I_j - I_0 = Q$ hoeven te bepalen.

Beschouw een toestand uit het genoemde hypervlak. Deze is bereikbaar vanuit een aantal toestanden uit het hypervlak $\sum_{j=1}^N I_j - I_0 = 1$. De kans op deze laatste toestanden kunnen we met behulp van (7.1) uitdrukken in kansen op toestanden uit het hypervlak $\sum_{j=1}^N I_j - I_0 = Q$. Maar dat betekent dat we voor iedere toestand uit het hypervlak $\sum_{j=1}^N I_j - I_0 = Q$ de kans op deze toestand kunnen uitdrukken in kansen op andere toestanden uit hetzelfde hypervlak. We vinden zo een stelsel lineaire vergelijkingen met evenveel onbekenden als vergelijkingen (Opm. dit aantal kan gereduceerd worden door te bedenken dat $P(I_1, I_2, \dots, I_N)$ niet afhangt van de volgorde van $I_1, I_2, \dots, I_N$).

voorbeeld

$N = 3, Q = 3, I_0 = 0$

De enige toestanden waarvoor geldt dat de kans op ieder ongelijk is aan nul, waarbij we toestand (I_1, I_2, I_3) zo noteren dat $I_1 \geq I_2 \geq I_3$, wordt gegeven door:

$(3, 1, -1)$	$(3, 1, -2)$	$(3, 1, -3)$
$(3, 0, 0)$	$(3, 0, -1)$	$(3, 0, 2)$
$(2, 2, -1)$	$(2, 2, -2)$	$(3, -1, -1)$
$(2, 1, 0)$	$(2, 1, -1)$	$(2, 2, -3)$
$(1, 1, 1)$	$(2, 0, 0)$	$(2, 1, -2)$
	$(1, 1, 0)$	$(2, 0, -1)$
		$(1, 1, -1)$
		$(1, 0, 0)$
$\sum_{j=1}^3 I_j = 3$	$\sum_{j=1}^3 I_j = 2$	$\sum_{j=1}^3 I_j = 1$

immers indien men in één van deze toestanden zit dan zal men de genoemde klasse nooit meer verlaten. Voor de vijf toestanden met $\sum_{j=1}^3 I_j = 3$ willen

we nu een vergelijking opstellen. Als voorbeeld zullen we de vergelijking bij toestand (2,2,-1) beschouwen:

$$\begin{aligned}
 27P(2,2,-1) &= 18P(2,0,-1) + 9P(3,-1,-1) + 9P(2,2,-3) = \\
 &= 12P(3,0,-1) + 6P(2,1,-1) + 6P(2,0,0) + 3P(2,2,-2) = \\
 &= 6P(3,1,-1) + 6P(3,0,0) + 3P(2,2,-1) + 6P(2,1,0).
 \end{aligned}$$

Een dergelijke vergelijking kan men voor ieder van de vijf toestanden afleiden. Men vindt uiteindelijk als oplossing:

$$P(3,1,-1) = \frac{27}{5400}; \quad P(3,0,0) = \frac{198}{5400}; \quad P(2,2,-1) = \frac{198}{5400};$$

$$P(2,1,0) = \frac{567}{5400}; \quad P(1,1,1) = \frac{594}{5400}.$$

- Decompositie

Voor decompositie vinden we de volgende overgangsvergelijkingen:

$$\left\{ \begin{aligned}
 P(I_1, I_2, \dots, I_N) &= \frac{1}{N} \sum_{j=1}^N P(I_1, I_2, \dots, I_j + 1, \dots, I_N) \\
 &\text{voor } I_0 + 1 \leq I_1, I_2, \dots, I_N < I_0 + q \\
 P(I_1, I_2, \dots, I_N) &= \frac{1}{N} \sum_{j=1}^N P(I_1, I_2, \dots, I_j + 1, \dots, I_N) + \\
 &\quad j \neq \ell_1, \ell_2, \dots, \ell_r \\
 &+ \frac{1}{N} \sum_{j=\ell_1, \ell_2, \dots, \ell_r} P(I_1, \dots, I_{j-1}, I_0 + 1, I_{j+1}, \dots, I_N) \\
 &\text{voor } I_0 + 1 \leq I_j < I_0 + q \quad \forall j \neq \ell_1, \ell_2, \dots, \ell_r \\
 &\text{en } I_{\ell_1} = I_{\ell_2} = \dots = I_{\ell_r} = I_0 + q
 \end{aligned} \right.$$

$$\text{Hieruit volgt } P(I_1, I_2, \dots, I_N) = \begin{cases} \frac{1}{Q^N} & \text{voor } I_0 + 1 \leq I_1, I_2, \dots, I_N \leq I_0 + q \\ 0 & \text{anders} \end{cases}$$

Opmerking

In dit geval levert decompositie natuurlijk de optimale strategie.

8. Overzicht resultaten en conclusies

Praktische uitvoering aggregatie:

1. Bepaal z_0 .
2. Bepaal met gegeven p, q, z_0 en d eerst c , op grond hiervan een benadering van $P_{d=0}(I)$ en vervolgens $P_d(I)$.
3. Bepaal numeriek een strategie $\{I_0\}$ zdd. $\int_{-\infty}^{\infty} (aI^+ + bI^-) dP_{d,I_0}(I)$ minimaal is.
4. Bepaal met behulp van simulatie het minimum van $\int_{-\infty}^{\infty} g_{I_0}^*(I) dP_{I_0}^*(I)$.

De in stap 3 gevonden strategie levert de benaderende aggregatiestrategie. De bij deze stap berekende kosten vormen een ondergrens voor de werkelijke kosten. De werkelijke kosten bij de in stap 3 gevonden strategie worden eveneens met simulatie bepaald.

Praktische uitvoering decompositie:

1. Bepaal z .
2. Bepaal met gegeven p, q, z en d eerst c , op grond hiervan een benadering van $P_{j,d=0}(I)$ en vervolgens $P_{j,d}(I)$.
3. Bepaal numeriek een strategie $\{I_0\}$ zdd $N \cdot \int_{-\infty}^{\infty} (aI^+ + bI^-) dP_{j,d,I_0}(I)$ minimaal is.
4. Bepaal met behulp van simulatie de I_0^* waarvoor de verwachte kosten minimaal zijn.

De in stap 3 gevonden strategie levert de benaderende decompositie strategie. De bij deze stap berekende kosten vormen een ondergrens voor de werkelijke kosten. De werkelijke kosten bij de in stap 3 gevonden strategie worden eveneens met simulatie bepaald.

Tenslotte zij nog opgemerkt dat we bij de simulatie twee verschillende realiseringen van het aankomstproces beschouwd hebben om enig inzicht in de betrouwbaarheid te hebben. In de onderstaande tabellen wordt dit als volgt weergegeven: (resultaten 1e realisering)/(resultaten 2e realisering).

q	d	N	a	b	p	decompositie		aggregatie		ondergrens kosten
						kosten	I_0^*	kosten	I_0^*	
2	0	2	1	3	0.6	3.91/3.85	0/0	3.90/3.81	2/2	3.38
2	0	2	1	3	1	1.00/1.00	-1/-1	1.49/1.50	-1/-1	1.00
2	0	2	1	3	0.55	6.96/7.38	2/2	7.31/7.48	5/5	6.69
20	0	2	1	3	0.6	15.00/15.02	-5/-5	17.10/17.05	1/0	15.04
2	0	5	1	3	0.6	5.83/6.05	-1/-1	5.73/5.65	3/3	3.38
2	0	20	1	3	0.6	13.21/13.10	-1/-1	14.89/14.79	8/9	10.71
2	0	2	1	1	0.6	2.06/1.99	-1/-1	2.10/2.08	0/0	1.69
2	0	2	1	10	0.6	6.57/6.70	2/1	6.43/6.24	4/4	5.69
2	5	2	1	3	0.6	5.72/5.71	3/3	5.70/5.66	8/8	5.03
2	10	2	1	3	0.6	7.05/7.00	6/6	7.08/6.99	14/14	6.49

Op grond van deze resultaten kunnen we zeggen dat als N groot is of de bezettingsgraad zeer klein is (dat wil zeggen $p \cdot q \gg 1$) decompositie beter zal werken dan aggregatie. In de overige gevallen echter geven de beide methodes vrijwel gelijke resultaten.

Deze resultaten zijn theoretisch goed te verklaren, immers:

- Bij toenemende N zal het verschil tussen $a \sum_{j=1}^N (I_j^+)$ + $b \sum_{j=1}^N (I_j^-)$

$a(\sum_{j=1}^N I_j)^+ + b(\sum_{j=1}^N I_j)^-$ groter worden. Hierdoor zal aggregatie slechter gaan werken.

- Bij toename van $p \cdot q$ neemt de bezettingsgraad af zodat de produkten elkaar minder zullen storen. Als gevolg hiervan zal de decompositie aanpak goede resultaten geven.

In de onderstaande tabel kunnen we de kwaliteit zien van de langs numerieke weg gevonden, benaderende strategieën. Met een * hebben we in deze tabel de plaatsen aangegeven waar de kosten van de benaderende strategie meer dan 5% afwijken van de kosten van respectievelijk de optimale aggregaat strategie en de optimale decompositie strategie.

q	d	N	a	b	p	benaderend decompositie		benaderend aggregatie		ondergrens kosten gevonden bij benaderend
						kosten	I ₀	kosten	I ₀	
2	0	2	1	3	0.6	* 5.33/5.05	-1	3.95/3.89	1	aggregatie
2	0	2	1	3	1	1.00/1.00	-1	1.49/1.50	-1	decompositie
2	0	2	1	3	0.55	* 11.80/12.10	-1	7.31/7.48	5	aggregatie
20	0	2	1	3	0.6	15.00/15.02	-5	* 18.39/18.48	-5	decompositie
2	0	5	1	3	0.6	5.83/6.05	-1	* 6.28/6.13	1	aggregatie
2	0	20	1	3	0.6	13.21/13.10	-1	* 20.69/20.61	1	decompositie
2	0	2	1	1	0.6	2.06/1.99	-1	2.10/2.08	0	aggregatie
2	0	2	1	10	0.6	* 9.38/8.96	0	6.43/6.24	4	aggregatie
2	5	2	1	3	0.6	5.72/5.71	3	5.82/5.74	7	decompositie
2	10	2	1	3	0.6	* 7.75/7.66	5	7.09/6.99	13	decompositie

We kunnen hier dezelfde opmerking over aggregatie en decompositie weer maken. Alleen zien we nu de effecten in versterkte mate optreden. Ook zien we dat de benaderende decompositie strategie slecht wordt ten opzichte van de werkelijke decompositie strategie als $p.q \approx 1$, terwijl de benaderende aggregatie strategie slecht wordt ten opzichte van de werkelijke aggregatie strategie als N groot is.

Als we nu in plaats van steeds de decompositie strategie of steeds de aggregatie strategie te nemen, telkens die strategie nemen waarvoor we de ondergrens van de kosten gevonden hebben (zie laatste kolom uit de vorige tabel) dan zullen we vrijwel altijd de beste benaderende strategie kiezen. Verbetering van de benaderende decompositiemethode en van de benaderende aggregatiemethode zal dan ook geen grote kostenbesparingen tot gevolg hebben, immers juist waar de benadering slecht wordt kiezen we toch de andere strategie. Aangezien een van de redenen om op een aggregatie/decompositie aanpak over te gaan was dat we een eenvoudige strategie zouden verkrijgen zouden we de volgende vuistregel kunnen formuleren:

"Gebruik voor niet te grote N en hoge bezettingsgraad de benaderende aggregatie strategie en in andere gevallen de benaderende decompositie strategie."

In geval van twijfel zouden we de aggregaat strategie moeten kiezen omdat deze minder gevoelig is voor de gekozen waarde van I_0 en daardoor minder kans geeft op zeer grote fouten.

9. Aanbevelingen toekomstig onderzoek

Allereerst zou toekomstig onderzoek de gevoeligheid van aggregatie en decompositie voor het onderliggende stochastische proces moeten nagaan. Ook moeten in de toekomst niet-identieke produkten aan de orde komen, die niet gelijk behandeld worden.

Het zou ook wenselijk zijn om een model te beschouwen, waarin we geen uitspraak doen omtrent het onderliggende stochastische proces maar alleen werken met bepaalde voorspellingen van de toekomst.

Misschien zou na verder onderzoek over het gedrag van aggregatie en decompositie in het één-machine-meer-produkten model een uitspraak over hun gedrag bij meer machines gedaan kunnen worden.

10. Referenties

- [1] P. van Beek, An application of decomposition and dynamic programming to the N -product, 1-machine problem, Methods of Operations Research 41 (1981) pp. 63-67.
- [2] S.E. Elmaghraby, The economic lot scheduling problem (ELSP) Review and extensions, Management Science 24 (1978) pp. 587-598.
- [3] A.C. Hax (ed.), Studies in operations management, Amsterdam-New York- Oxford, 1978.
- [4] Th.H.B. Hendriks en J. Wessels, Repetitive schemes for the single-machine multi-product lot-size scheduling problem, Operations Research Verfahren XXVI (1978) pp. 571-581.
- [5] L. Kleinrock, Queueing systems, volume I, Wiley, 1975.
- [6] F. Lansdorp en J.W.M. Bertrand, Control of stochastic single-machine multi-product process, intern rapport TH Eindhoven, 1978.

- [7] G. Liesegang, Aggregation bei Linearen Optimierungsmodellen, Ph.D. Thesis, Köln, 1981.
- [8] P. van Schuylenburg, Een onderzoek betreffende een dynamisch programmeringsalgorithme voor het N-producten één machine probleem, Kwantitatieve Methoden 2 (1981) pp. 7-18.
- [9] M. Starremans en J. Wessels, Clustering of products in the single-machine lot-size scheduling problem, Memorandum COSOR-81-07.
- [10] J. Wessels en N. Thijssen, Repetitive schemes with performance bounds for the economic-lot scheduling problem with critical set-up, Mitteilungen der mathematische Gesellschaft der Deutschen Demokratischen Republik, Heft 1 (1978) pp. 49-54.
- [11] D.A. Wismer (ed.), Optimization methods for large scale systems ... with applications, New York, 1971.