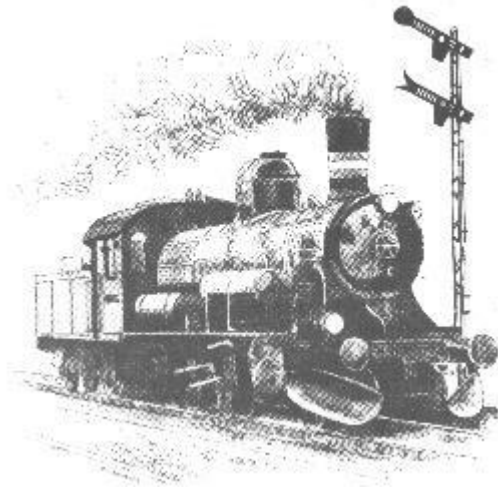


**Optimeringsmodeller
innen materiellturnering for NSB**



**Hovedoppgave ved
Institutt for industriell økonomi og teknologiledelse, NTNU**



Kenneth Aschehoug og Marte Fodstad

Trondheim, juni 2002



Forord

Denne hovedoppgaven inngår som siste del i våre sivilingeniørstudier ved Institutt for industriell økonomi og teknologiledelse ved Norges teknisk-naturvitenskapelige universitet. Arbeidet har blitt utført over 21,5 uker i vårsemesteret 2002.

Arbeidet med oppgaven har foregått med NSB ved Planavdelingen som samarbeidsbedrift. Målsetningen har vært å utvikle og teste en operasjonsanalytisk modell som kan støtte materiellplanleggingen for nærtrafikken omkring Oslo.

Vi ønsker å takke en rekke mennesker som har hjulpet oss underveis i arbeidet. For det første vår faglige veileder Asgeir Tomasgard for gode innspill. Dessuten vår samarbeidsbedrift NSB, med en spesiell takk til Lasse Bjur Olsen, Anniken Refseth og vår kontaktperson Hans Petter Krane som tålmodig har svart på våre spørsmål. Professor Bjørn Nygreen fortjener en stor takk for gode råd angående bruk av programvaren Xpress-MP. Vi har også satt pris på raske og gode tilbakemeldinger fra kundeserviceenheten i Dash Optimization. Ikke minst vil vi takke Institutt for datateknikk og informasjonsvitenskap ved stipendiat Zoran Constantinescu Fülöp for å ha gitt oss tilgang til å gjennomføre testing på deres datamaskiner. Til slutt vil vi rette en stor takk til Martin Thorsen Ranang og Ingunn Grytting for å ha stilt opp med datateknisk bistand.

Trondheim, 21. juni 2002

Kenneth Aschehoug

Marte Fodstad



Sammendrag

Planavdelingen ved NSB Drift har blant annet ansvaret for å utarbeide materiellturneringsplaner for nærtrafikken omkring Oslo. Slik planlegging går i korte trekk ut på å bestemme serier av turer som skal dekkes av de ulike materiellindividene. For å håndtere den høye kompleksiteten i planleggingsarbeidet, utvikles det i den taktiske planleggingsfasen et tidlig utkast til en materiellturnering som gradvis detaljeres frem mot den operative driften. I tillegg gjøres det en utsjekk om man har nok materiell til å dekke ruteplanen.

Målsetningen med denne hovedoppgaven har vært å utvikle en operasjonsanalytisk modell med en tilhørende løsningsalgoritme som kan støtte materiellplanleggingen i den taktiske fasen. Etter en vurdering av alternative løsningsstrategier opp i mot modellformuleringen vår, falt valget på Benders dekomponeringsalgoritme. Foruten en implementasjon av algoritmen i programmeringsspråket Java, ble modellen implementert i en kommersiell programpakke med tilhørende generelle løsningsalgoritmer. Implementasjonen av Benders dekomponering førte ikke frem på grunn av konvergensproblematikk i forbindelse med toleransegrenser. Løsningsresultatene som danner grunnlaget for konklusjonene baserer seg derfor bare på implementasjonen i den kommersielle programpakken.

Mangel på variasjon i testsettene gjør at vi ikke kan være bastante når vi trekker konklusjoner. Resultatene viser at modellformuleringen vår er en god tilnærming til virkeligheten. Et arbeid må imidlertid legges ned for å differensiere tidsparametrene som inngår i modellen for at den skal kunne benyttes i en praktisk planleggingssituasjon. Løsningstidene er svært gode for å gjøre en utsjekk av materiell og akseptable, men ikke gode, for å lage utkast til materiellturneringer hvor man baserer seg på dagens bruksmønster. Løsningstidene er uakseptable for alternative bruksmønstre som krever større modeller.

Anskaffelse av et beslutningsstøtteverktøy er kostbart og endringsprosessen er krevende. Dersom NSB går til anskaffelse av et slikt verktøy må det være fleksibelt nok til kunne anvendes for flere ulike bruksmønstre. Mangelen på denne fleksibiliteten gjør at vi ikke kan anbefale en integrering av modellen med uttestet løsningsalgoritme i dagens eksisterende systemer.



Innholdsfortegnelse

1	INNLEDNING.....	1
2	NSB.....	2
2.1	NÆRTRAFIKKEN OMKRING OSLO	2
2.1.1	<i>Ruteplanen</i>	3
2.1.2	<i>Materiell</i>	3
2.1.3	<i>Vedlikehold</i>	4
2.1.4	<i>Hensetting</i>	4
2.2	PRODUKSJONSPANLEGGING.....	4
2.2.1	<i>Tidsdekomponering</i>	5
2.2.2	<i>Funksjonsdekomponering</i>	5
2.2.3	<i>Planprosessen</i>	6
2.3	MATERIELLPLANLEGGING FOR NÆRTRAFIKKEN	8
2.3.1	<i>Inndata</i>	8
2.3.2	<i>Restriksjoner</i>	8
2.3.3	<i>Retningslinjer</i>	9
2.3.4	<i>Datastøtte</i>	9
3	MODELL.....	11
3.1	MODELLBEGRENSNINGER	11
3.1.1	<i>Vedlikehold</i>	11
3.1.2	<i>Sportilgang</i>	12
3.1.3	<i>Hensettingskapasitet på dagtid</i>	12
3.1.4	<i>Snutid som avhenger av toglengde</i>	12
3.1.5	<i>Kostnadsfunksjon</i>	13
3.2	MODELLBESKRIVELSE.....	13
3.2.1	<i>Schrijver</i>	13
3.2.2	<i>Enkel modellformulering</i>	14
3.2.3	<i>Kostnadsfunksjon</i>	18
3.2.4	<i>Cordeau et al.</i>	18
3.2.5	<i>Utvidelser</i>	19
3.2.6	<i>Endelig modellformulering</i>	22
4	OPERASJONSANAYTISK TEORI	23
4.1	LAGRANGE.....	23
4.2	DANTZIG-WOLFE	26
4.3	BENDER.....	28
4.3.1	<i>Metoden</i>	28



4.3.2	<i>Eksempel</i>	35
4.3.3	<i>Benders dekomponering og heltallskrav</i>	36
5	LØSNINGSMETODER OG DEKOMPONERING	39
5.1	TO LØSNINGSMETODER	39
5.2	DEKOMPONERINGSMULIGHETER	40
5.2.1	<i>Lagrange-relaksering</i>	40
5.2.2	<i>Dantzig-Wolfe dekomponering</i>	41
5.2.3	<i>Benders dekomponering</i>	43
5.2.4	<i>Valg av dekomponeringsstrategi</i>	45
5.3	HELTALLIGHET OG VALG AV SUBPROBLEMER MED BENDER	45
5.3.1	<i>Separabilitet og heltallighet</i>	46
5.3.2	<i>Valg av subproblemer</i>	47
6	IMPLEMENTASJON	49
6.1	MODELL I XPRESS-MOSEL	49
6.2	BENDERS DEKOMPONERING	49
6.2.1	<i>Løsningsalgoritmen</i>	50
6.2.2	<i>Multikutt</i>	53
6.2.3	<i>Kutthåndtering</i>	53
6.2.4	<i>Kuttype</i>	54
6.2.5	<i>Branch and bound</i>	54
6.2.6	<i>Masterproblem til heltallighet – 1. alternativ</i>	55
6.2.7	<i>Branch and bound for subproblemer og Gomory-kutt – 2. alternativ</i>	55
7	TESTING OG RESULTATER	56
7.1	TESTING I FORBINDELSE MED DEN INKREMENTELLE IMPLEMENTASJONEN	56
7.2	BESKRIVELSE OG VALG AV TURER I DATASETTENE	56
7.3	BESKRIVELSE AV PARAMETRE OG VALG AV VERDIER.....	57
7.3.1	<i>Stasjoner</i>	58
7.3.2	<i>Materiell</i>	58
7.3.3	<i>Turer</i>	59
7.3.4	<i>Kompatible typer</i>	61
7.3.5	<i>Snu- og ventekostnader</i>	62
7.4	KVALITATIVE TESTTILFELLER	62
7.4.1	<i>Valg av pendel og parametre</i>	62
7.4.2	<i>Testtilfeller</i>	63
7.4.3	<i>Resultater</i>	64
7.5	KVANTITATIVE TESTTILFELLER.....	72
7.5.1	<i>Dimensjoner for testing</i>	72



7.5.2	<i>Testtilfeller og resultater</i>	75
8	DRØFTING	80
8.1	ET INTEGRERT BESLUTNINGSSTØTTEVERKTØY	80
8.2	MATERIELLPLANLEGGERNES LEVERANSER OG BRUK AV MODELLEN	81
8.2.1	<i>Utsjekk av materiell</i>	81
8.2.2	<i>Utarbeidelse av grunnplan</i>	83
8.2.3	<i>Krav til løsningsstid og modellstørrelse</i>	85
8.3	VURDERING AV KVALITATIVE TESTRESULTATER	85
8.3.1	<i>Tidsparametre</i>	86
8.3.2	<i>Kostnader</i>	88
8.3.3	<i>Kvalitative modellbegrensninger</i>	89
8.4	VURDERING AV KVANTITATIVE TESTRESULTATER	91
8.4.1	<i>Kritikk av datasettene</i>	91
8.4.2	<i>Kostnadsfunksjon</i>	91
8.4.3	<i>Stasjonskapasitet og materielltilgjengelighet</i>	92
8.4.4	<i>Kompatible typer</i>	93
8.4.5	<i>Bruksmønstre og løsningsstider</i>	93
8.4.6	<i>Datamaskinressurser og optimalisering av kode</i>	94
8.5	DETERMINISTISK MODELL I EN STOKASTISK VERDEN	94
8.6	SELVIMPLEMENTERT ALGORITME UTEN RESULTAT	96
8.6.1	<i>Konvergensproblemer</i>	96
8.6.2	<i>Verifikasjon av implementasjonen</i>	97
8.6.3	<i>Tilpasning av presisjon</i>	98
9	KONKLUSJON	100
10	REFERANSER	102
11	BIBLIOGRAFI	105

Vedleggsliste

VEDLEGG A KILDEKODE FOR XPRESS-MOSEL

VEDLEGG B KILDEKODE FOR BENDERS DEKOMPONERING I JAVA



Tabelliste

Tabell 7-1: Tidsdata fra NSB og våre valg av parameterverdier i minutter	59
Tabell 7-2: Stasjoner brukt for de kvalitative testtilfellene	63
Tabell 7-3: Oversikt over kvalitative testtilfeller.....	64
Tabell 7-4: Materiellforbruk i antall Type 69 tre-vognsett.....	64
Tabell 7-5: Datasettene og deres størrelser	72
Tabell 7-6: Stasjonskapasiteter for det stramme settet av restriksjoner.....	73
Tabell 7-7: Kombinasjoner av stramme og slakke restriksjoner	74
Tabell 7-8: Serie av varierende vekting mellom <i>dist</i> og <i>matr</i>	74
Tabell 7-9: Serie av konstant vekting mellom <i>dist</i> og <i>matr</i> per datasett	75
Tabell 7-10: Størrelse og restriksjonskombinasjon A.....	76
Tabell 7-11: Størrelse og restriksjonskombinasjon B.....	76
Tabell 7-12: Størrelse og restriksjonskombinasjon C.....	76
Tabell 7-13: Størrelse og restriksjonskombinasjon D.....	77
Tabell 7-14: Varierende kostnadsfunksjon og restriksjonskombinasjon A	77
Tabell 7-15: Varierende kostnadsfunksjon og restriksjonskombinasjon B	77
Tabell 7-16: Varierende kostnadsfunksjon og restriksjonskombinasjon C	78
Tabell 7-17: Varierende kostnadsfunksjon og restriksjonskombinasjon D	78
Tabell 7-18: Variasjon i tillatte kompatible typer for datasett 2, <i>dist</i>	78
Tabell 7-19: Variasjon i tillatte kompatible typer for datasett 2, <i>matr</i>	79
Tabell 7-20: Variasjon i tillatte kompatible typer for datasett 3, <i>dist</i>	79
Tabell 7-21: Variasjon i tillatte kompatible typer for datasett 3, <i>matr</i>	79



Figurliste

Figur 2-1: Nærtrafikken omkring Oslo [www.nsb.no]	2
Figur 2-3: Planleggingsprosessen med fokus på materiell	7
Figur 3-1: Graf for ruteplan over to dager	15
Figur 4-1: Koeffisientmatrise med blokkstruktur [Birge og Louveaux, 1997].....	29
Figur 4-2:Kjeglene pos W og pol pos W [Kall et al., 1994].....	31
Figur 4-3: Eksempel på løsning med Benders dekomponering [Kall et al., 1994].....	35
Figur 6-1: Overordnet prosessflyt.....	50
Figur 6-2: Prosessflyt for løsningsalgoritmen. Del 1	51
Figur 6-3: Prosessflyt for løsningsalgoritmen, del 2.....	52
Figur 7-1: Turnering for MATR_7/3, del 1	66
Figur 7-2: Turnering for MATR_7/3, del 2	67
Figur 7-3: Turnering for DIST_SV_7/3, del 1	68
Figur 7-4: Turnering for DIST_SV_7/3, del 2.....	69
Figur 7-5: NSB's turnering, del 1	70
Figur 7-6: NSB's turnering, del 2.....	71



1 Innledning

Ved drift av jernbane er det en rekke faktorer som virker inn på hverandre og disse må koordineres for at trafikken skal fungere. Dette skaper en høy kompleksitet i planleggingsarbeidet, noe som er bakgrunnen for at planleggingen i NSB både deles opp i ulike funksjonsområder og ulike planleggingsfaser. Vi har i denne rapporten begrenset oss til å se på materiellplanleggingsarbeidet i en taktisk fase.

I arbeidet med materiellplanene brukes dataverktøy i dag hovedsakelig for registrering av planer. Det finnes ingen beslutningsstøtte som kan bidra med forslag til løsninger. Dette fører til at planleggingsarbeidet er svært arbeidskrevende slik at man ikke har tid til å utvikle flere alternative planer. Resultatet av dette er at man i liten grad kan vurdere kvaliteten og konsekvensen av de valgene som gjøres og at man dermed blir svært avhengig av å kunne videreføre løsninger fra gamle planer. Videre er det svært utfordrende for planleggerne, selv innenfor delområdet materiellplanlegging, å få tilstrekkelig oversikt til å kunne gjøre helhetsvurderinger framfor å ta beslutninger basert på lokale vurderinger.

Målsetningen med vårt arbeid har vært å utvikle en modell med en tilhørende løsningsalgoritme som kan inngå i et beslutningsstøttesystem for materiellplanleggerne. Siden NSB's ulike tilbud er allsidige vil ikke nødvendigvis én og samme modell passe godt til alle typer trafikk. Derfor har vi begrenset oss til å tilpasse modellen til nærtrafikken omkring Oslo.

Innledningsvis kommer en beskrivelse av planleggingsarbeidet med fokus på materiellplanlegging for nærtrafikken. Deretter gir kapittel tre en presentasjon av vårt forslag til modellformulering som bygger på tilsvarende modeller fra faglitteraturen. I kapittel fire beskrives tre ulike løsningsalgoritmer basert på operasjonsanalytisk teori, før vi i kapittel fem argumenterer for våre valg av løsningsstrategier. Kapittel seks omhandler implementasjon og de valgene som er gjort i den forbindelse. De ulike testtilfellene med tilhørende datasett, valg av parameterverdier og testresultater presenteres i kapittel sju. Dernest følger en vurdering av modellformulering og løsningsalgoritme opp mot et tiltenkt bruksmønster før vi konkluderer.

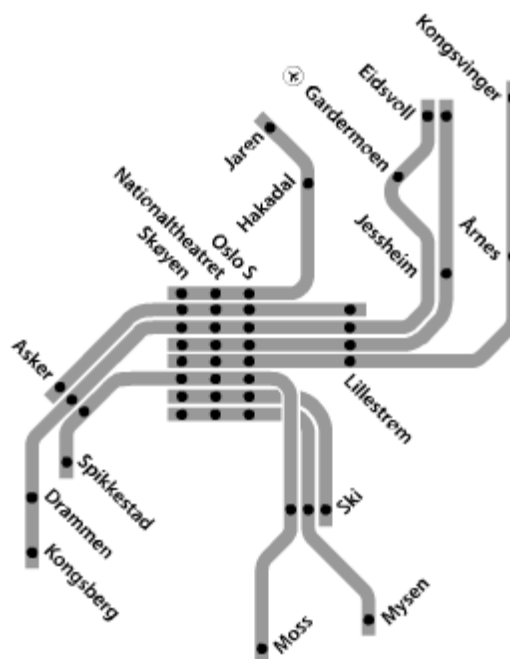


2 NSB

Det har vært jernbanedrift i Norge siden opprettelsen av Hovedbanen mellom Oslo, dengang Christiania, og Eidsvoll i 1854. I dag er organiseringen av jernbane-Norge inndelt i adskilte ansvarsområder. Jernbaneverket (JBV) har ansvaret for utbygging og drift av infrastrukturen mens Statens Jernbanetilsyn ser til at aktørene som bedriver jernbanevirksomhet følger jernbaneloven. NSB BA er største operatør og er heleid av staten ved Samferdselsdepartementet. [www.nsb.no]

2.1 Nærtrafikken omkring Oslo

Nærtrafikken omkring Oslo hører organisatorisk under enheten Persontog Østlandet. Produktet er strukturert som et stjernemønster av åtte pendler med knutepunkt ved Oslo Sentralstasjon, Nasjonalteateret og Skøyen, som vist i Figur 2-1. Hver pendel klassifiseres som *innerstrekning* eller *ytterstrekning* avhengig av distansen mellom endestasjonene på pendelen, og denne klassifiseringen legger føringer for hvilken frekvens avgangene på pendelen har. Samlet består nærtrafikken av omlag 450 turer på en hverdag fordelt på mellom 35 og 100 turer på hver pendel. Til sammen på en uke er trafikken på omlag 2850 turer.



Figur 2-1: Nærtrafikken omkring Oslo [www.nsb.no]



2.1.1 Ruteplanen

Ruteplanen inneholder informasjon om avgangs- og ankomst- tider og -steder for togene som er planlagt å kjøre. Ruteplanen gjentar seg i utgangspunktet ukentlig og varierer i løpet av uka. Variasjonen dreier seg om reduksjon i trafikken i forbindelse med helg i forhold til hverdager, på bakgrunn av lavere etterspørsel. Dessuten er tilbudet noe endret i forbindelse med ferier og helligdager som bryter med de ordinære ukene. Ved endringer eller arbeid på infrastrukturen kan det også bli behov for endringer i ruteplanen. Foruten turene som er beregnet på passasjertrafikk inneholder ruteplanen tomtogturer. Dette er kjøring uten passasjerer som blant annet gjøres for å komme til vedlikehold eller posisjonere materiellet for senere turer.

2.1.2 Materiell

I nærtrafikken benyttes hovedsakelig motorvognsett Type 69. Disse er spesielt tilpasset nærtrafikk med blant annet god akselerasjon, store fjernstyrte dører og høy setekapasitet. Et sett består av to eller tre vogner som ikke splittes fra hverandre, henholdsvis to- og tre-vognsett. Hvert sett har en motorvogn i den ene enden og en styrevogn i den andre. Motorvognsettene kan føres fra både motorvognen og styrevognen slik at man ikke behøver å snu toget ved endestasjoner. Skifte av kjøreretning omtales i resten av rapporten som snuing. Flere motorvognsett kan skjøtes sammen, og opptil ni vogner har vært forsøkt kjørt sammenkoblet. I praksis har man opplevd at mer enn åtte vogner i sammen har ugunstige effekter og brukes derfor ikke i dag. Antall individer som er skjøtt sammen påvirker snutiden, slik at flere individer krever lengre snutid. Totalt har man 70 sett av type 69, og av disse er 59 sett i samtidig drift med dagens ruteplan.

Motorvognsett benyttes så langt det rekker, og tre-vognsett prioriteres framfor to-vognsett. I rushtrafikken, som er førende for hvor mye materiell som trengs, brukes også noen lokomotiv med vogner. Disse brukes kun på de lengre strekningene, altså ytterstrekningene, og forsøkes holdt vekk fra de sterkest rushbelastede områdene, siden utstyret ikke er særlig godt tilpasset denne typen trafikk.

En ny type materiell er planlagt tatt i bruk. Dette er motorvognsett Type 72 som kun finnes som fire-vognsett. Når dette nye materiellet kommer tar man sikte på å fase ut



lokomotivene med vogner fra nærtrafikken, men vil bevare motorvognsettene man allerede har, slik at man slakker litt opp på restriksjonen for tilgjengelig materiell. De to typene motorvognsett, Type 69 og 72, skal ikke skjøtes sammen.

De ulike vognene som inngår i motorvognsettene av type 69 er tilnærmet like lange og har alle en setekapasitet på omlag 100 seter. Vognene i Type 72 er noe kortere og har en setekapasitet på omlag 75 seter.

2.1.3 Vedlikehold

Man har to hovedkategorier av vedlikehold, tungt og lett. Det lette vedlikeholdet er driftspausebasert. Dette betyr at vedlikeholdsjobben gjøres når materiellet har ledige pauser i følge materiellturneringen. Hver type materiell har en maksimumsgrense for hvor langt det kan kjøre mellom hver gang det er inne til slikt vedlikehold. Tungt vedlikehold innebærer som navnet tilsier større og mer tidkrevende operasjoner. Dette er aktiviteter som ikke integreres i materiellturneringen. I stedet tas individene ut av turnering når slikt vedlikehold skal gjennomføres og erstattes med andre.

2.1.4 Hensetting

Materiell som har et lengre opphold i turneringen kan hensettes. Hensetting tillates på en rekke ulike stasjoner foruten på vedlikeholdsbasene Sundland og Filipstad. Hvilke stasjoner som kan brukes avhenger først og fremst av infrastrukturen. Dessuten kan man velge å ikke hensette materiell på stasjoner hvor hensetting ville kreve bruk av ekstra ressurser for å skaffe nødvendig tilsyn.

2.2 Produksjonsplanlegging

Planavdelingen i NSB Drift har ansvar for å utarbeide produksjonsplaner som senere skal settes ut i live under styring av Driftsoperativt senter (DROPS) og toglederne i Jernbaneverket. Nye produksjonsplaner utvikles til det årlige terminskiftet i juni, men endringene er relativt små fra termin til termin slik at man i stor grad tar utgangspunkt i gamle planer. I januar har man en justering som i hovedsak forutsetter de samme aktivitetene som for terminskiftet. Videre har man grunnruteendringer som gir større endringer i produksjonen og skjer omkring hvert 4.-5. år, avhengig av behov. Vi vil ha



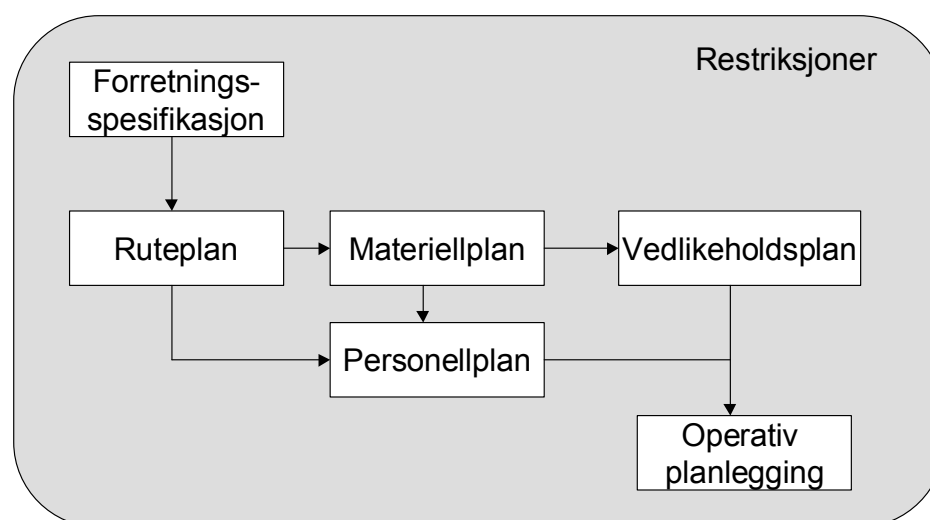
fokus på arbeidet med terminskifter da prosessen i forbindelse med grunnruteendringer er lite formalisert. [Stølan, Sæbø, Sætermo og Tomasgard, 2000]

2.2.1 Tidsdekomponering

Planleggingsarbeidet kan dekomponeres langs en tidsakse i strategisk, taktisk og operativ planlegging, hvor stadiene er organisert med synkende lengde på planleggingshorisonten og økende detaljgrad i planene. Den strategiske planleggingen dreier seg i stor grad om konsekvensvurderinger knyttet til større endringer som materiellinnkjøp og infrastrukturendringer. På taktisk nivå finner man utarbeidelsen av overordnede produksjonsplaner, som deretter detaljeres i det operative nivået før de settes i operativ drift. [Stølan et al., 2000, Sætermo og Tomasgard, 2000]

2.2.2 Funksjonsdekomponering

Ved utarbeidelsen av produksjonsplaner har man en rekke hensyn å ta. Dette dreier seg blant annet om infrastrukturbeskravninger, kjøreegenskaper til materiellet, lovverk og ønsker fra markedet. I Figur 2-2 inngår disse i den grå boksen som omgir de ulike aktivitetene i produksjonsplanleggingen. På grunn av denne høye kompleksiteten dekomponeres produksjonsplanene i ruteplaner, materiellplaner, vedlikeholdsplaner og personellplaner. Mellom disse komponentene ligger det sterke bindinger ved at en komponent er grunnlaget for en annen. [Stølan et al., 2000, Humphreys og Watson, 1999]



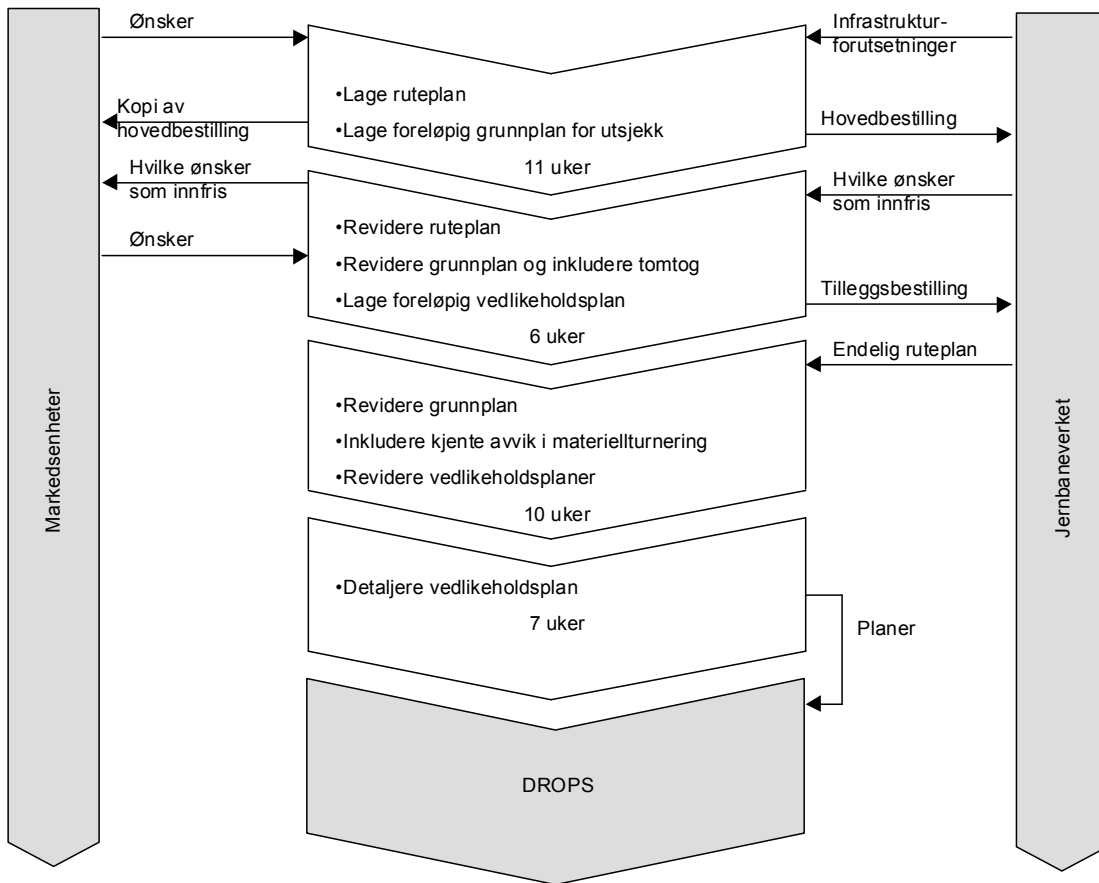
Figur 2-2: Funksjonsdekomponering på taktisk nivå



2.2.3 Planprosessen

I den følgende framstillingen av planprosessen ligger fokus på materiellturnering på taktisk nivå og de aktivitetene som er nært knyttet til denne. Figur 2-3 gir en illustrasjon av prosessen. Vær oppmerksom på at denne figuren har et sterkt fokus på materiellplanlegging og ikke illustrerer alle deler av planleggingsprosessen. De grå boksene er enheter utenfor Planavdelingen.

Forberedelsene til et terminskifte starter omkring ni måneder før ruteendringen ved at Jernbaneverket informerer om infrastrukturforutsetningene for den aktuelle perioden. Videre kommer markedsenhetene med ønsker om rutetider. Med bakgrunn i denne informasjonen utvikler rutekoordinatorne en ruteplan. Denne ruteplanen representerer en generell uke uten å ta hensyn til unntak som ferier eller planlagte sporbrudd. For å sjekke at man har tilstrekkelig materiell til å gjennomføre ruteplanen legges også en foreløpig grunnplan som ikke inkluderer tomtogturer. En *grunnplan* er et sett av materiellturneringer som heller ikke tar hensyn til unntakene nevnt over. Innen en frist gitt av Jernbaneverket sendes så en hovedbestilling som består av en ønsket ruteplan til Jernbaneverket. Fristen for når markedsenhetene må ha kommet med sine ønsker er noe flytende avhengig av om rutekoordinatorne og materiellplanleggerne får tid til å inkludere ønskene i sine planer før hovedbestillingen skal gå.



Figur 2-3: Planleggingsprosessen med fokus på materiell

Når Jernbaneverket har gjennomgått bestillingen gis det tilbakemeldinger til Planavdelingen. Tilbakemeldingene gis videre til markedsenhetene som får mulighet til å komme med ytterligere innspill om ruteønsker. På grunnlag av dette revideres ruteplanen. Tilsvarende revideres grunnplanen og man bestemmer nødvendige tomtogturer og posisjonskjøring. Med utgangspunkt i grunnplanen utvikles foreløpige planer for det driftspausebaserte vedlikeholdet. En tilleggsbestilling som inneholder tomtogturer sendes til Jernbaneverket.

Etter at Jernbaneverket har behandlet tilleggsbestillingen sendes den endelige ruteplanen tilbake til Planavdelingen. På grunnlag av denne revideres grunnplanen etter behov. Deretter starter man med å utvikle materiellturneringer tilpasset hver uke gjennom året. For ordinære uker vil turneringen tilsvare grunnplanen, mens man for enkelte uker må gjøre endringer for å tilpasse turneringene til kjente avvik. Inn i disse turneringene bestemmes det også når materiellet skal til vedlikehold. Videre revideres



vedlikeholdsplanen. Kort tid før operativ drift detaljeres vedlikeholdsplanene ned til dagshorisont. [Stølan et al., 2000]

2.3 Materiellplanlegging for nærtrafikken

Materiellturneringene for nærtrafikk omkring Oslo legges adskilt fra de andre produktene. Dette lar seg gjøre fordi en viss mengde materiell er satt av til denne trafikken før arbeidet med materiellturneringer starter.

2.3.1 Inndata

Når materiellturneringene skal fastlegges tar materiellplanleggeren utgangspunkt i en ruteplan [Figur 2-2]. Denne ruteplanen skal være kjørbar med tanke på sportilgang slik at materiellplanleggeren vet at den lar seg gjennomføre. Ruteplanen inneholder alle de turene som er bestemt at skal kjøres, og som materiellplanleggeren dermed må sørge for å tildele materiell. Dog har materiellplanleggerne mulighet til å komme med innspill overfor rutekoordinatoren om endringer som trengs for å oppnå en kjørbar turnering eller kan gi betydelige forbedringer i turneringene. Hvilken setekapasitet de ulike avgangene skal tildeles baserer seg i stor grad på ønsker fra markedsenheten som kjenner behovet best.

2.3.2 Restriksjoner

Oppgaven til materiellplanleggeren er å sette opp gyldige turneringer. Dette innebærer å bestemme serier av turer som er kompatible med tanke på ankomst- og avgangstider og steder. Det er en rekke restriksjoner som må overholdes for at en turnering skal være gjennomførbar og disse er listet under. Vær oppmerksom på at enkelte av disse restriksjonene, eksempelvis kravet til lett vedlikehold, blir delvis oversett i tidlige faser av planleggingsprosessen, i tråd med beskrivelsene i delkapittel 2.2.3.



- Sørge for lett vedlikehold innenfor maksimumsgrensen for antall kjørte kilometer
- Ikke bruke mer materiell enn tilgjengelig
- Ikke overstige maksimal toglengthe
- Ha nok tid tilgjengelig ved skjøting og splitting
- Ha nok tid tilgjengelig ved sving
- Ikke bryte hensettingskapasiteten på stasjonene
- Sikre sportilgang for tomtog
- Dekke etterspørselen på alle turer

2.3.3 Retningslinjer

Målet for materiellplanleggerne er først og fremst å komme fram til gjennomførbare turneringer. Samtidig er det ønskelig at disse turneringene er minst mulig kostnadsdrivende med tanke på blant annet personellkostnader og vedlikeholdskostnader. Videre baserer man seg på et prinsipp om ”rene pendler” som innebærer at man forsøker å la hvert individ kun kjøre turer tilknyttet én pendel. Dog fravikes dette prinsippet i enkelte situasjoner, eksempelvis når man legger til besøkene ved vedlikeholdsbasen da kun én pendel fører dit. Hensikten med prinsippet er å øke robustheten til materiellplanene. Av hensyn til robustheten ønsker man også å holde syklene i materiellturneringene relativt korte. Med sykler menes det her at hvert individ etter en tid kommer tilbake til utgangspunktet og kan gjenta turneringen. En sykkel består av et helt antall uker. Denne strukturen bidrar dessuten til å forenkle jobben for materiellplanleggerne siden turneringene kan gjenbrukes.

2.3.4 Datastøtte

Systemet ResourcePlan brukes i dag til å støtte arbeidet med materiellturneringer. Dette systemet bidrar blant annet med visualiseringsmuligheter og feildeteksjon, men har ikke beslutningsstøtte i den forstand at programmet foreslår løsninger for brukeren. Dette betyr at det er opp til planleggeren å tenke ut de gode løsningene og registrere disse i systemet. I praksis gjøres mye av planleggingsarbeidet på den



tradisjonelle måten med penn og papir framfor direkte i dataverktøyet. En sentral årsak til dette er at programvaren stiller større krav til at ruteplanen man jobber ut i fra er komplett enn hva som kreves ved manuell planlegging. Dermed kan man komme i gang med materiellplanleggingen tidligere ved å begrense bruken av verktøyet. Dette er en problemstilling som berører planleggingsprosessen som helhet, og som man jobber med å utbedre.



3 Modell

Vi har tatt sikte på å utvikle en modell som kan støtte materiellplanleggingsarbeidet i NSB i en tidlig taktisk fase. Med dette tenker vi oss en støtte til å sjekke om man har tilstrekkelig mengde materiell tilgjengelig for å gjennomføre en ruteplan og utviklingen av en grunnplan i forbindelse med hovedbestillingen [delkapittel 2.2.3]. I dette kapitlet blir grensene trukket for hva som skal inngå i modellen og hva som faller utenfor modellens domene. Videre gis en inngående beskrivelse av modellen.

3.1 Modellbegrensninger

Under arbeidet med formuleringen har vi forsøkt å dekke flest mulig av de restriksjonene en materiellturnering må oppfylle i en tidlig taktisk planleggingsfase. Likevel må det påpekes at en anvendbar modell aldri kan bli en tro kopi av virkeligheten. Det ligger i modellens natur at den skal representere et forenklet utsnitt av virkeligheten og derigjennom bidra til å kunne vise de essensielle aspektene blant alle detaljene som finnes i det virkelige domenet. Dette betyr at modelleringen krever visse avveininger mellom hvilke faktorer som må inngå for å få en meningsfull modell, og hvilke som bør abstraheres bort. Disse avveiningene avhenger av bruksområdet modellen er tiltenkt. I det følgende presenteres vurderinger av enkelte forhold som er nært knyttet til materiellplanleggingen, men som likevel er utelatt fra vår modell.

3.1.1 Vedlikehold

Kravet om at materiellet skal til vedlikehold innenfor en gitt akkumulert kjøredistanse er ikke modellert. Hovedårsaken til dette er den hyppige frekvensen på avvik i tidsrommet mellom den tidlige taktiske planleggingen og den operative driften. Sannsynligheten for at eventuelle beregnede akkumulerte kjøredistanser i den taktiske planleggingen skulle være sammenfallende med de faktiske forholdene ved driften 6 måneder senere er liten. Derfor er det lite gunstig å inkludere restriksjoner for å ta høyde for det driftspausebaserte vedlikeholdet i modellen vår. I det hovedbestillingen går har man i dag heller ikke sørget for at materiellturneringene oppfyller dette kravet.



3.1.2 Sportilgang

For å kunne kontrollere at ikke flere tog benytter samme banestrekning samtidig må man ha en fullstendig oversikt over alle typer trafikk på den aktuelle strekningen med informasjon om avgangstider og –steder, kjørehastigheter og infrastrukturegenskaper. Dette er sentralt for arbeidet til en ruteplanlegger ved utviklingen av en ruteplan. Tomtogkjøring er en konsekvens av materiellturneringene og ikke et krav innebygd i ruteplanen. Derfor har også materiellplanleggeren et ansvar i samarbeid med ruteplanleggeren for å kontrollere sportilgang når man har behov for å tomtogkjøre. Dersom alle tomtogturer som angis i inndataene til modellen er kontrollert med tanke på sportilgang vil løsningen ikke bryte kravet, men modellen kan ikke gjennomføre slike kontroller da dette som nevnt forutsetter inngående kjennskap til ruteplanleggingen. Hvorvidt en materiellturnering generert av en modell innfrir kravet til sportilgang er dermed avhengig av forarbeidet brukeren av modellen legger ned.

3.1.3 Hensettingskapasitet på dagtid

I praksis har det i svært liten grad vist seg nødvendig å tomtogkjøre for å unngå at det på dagtid hensettes flere individer enn hva en stasjon har kapasitet til. For å forsikre seg at denne begrensningen ikke brytes må det legges til et stort antall restriksjoner i modellen. Dette er bakgrunnen for at vi har valgt å ikke modellere denne begrensningen. En utvidelse for å ta hensyn til dette aspektet kan gjøres på samme måte som tilsvarende begrensning for overnattinger modelleres [delkapittel 3.2.2].

3.1.4 Snutid som avhenger av toglengde

Tiden det tar å snu avhenger av hvor mange individer som er koblet sammen. I vår modell vil dette innebære at hvilke beslutningsvariabler som skal genereres avhenger av løsningen, og vi stiller oss tvilende til at det lar seg gjøre å modellere dette på en hensiktsmessig måte. Vi er heller ikke kjent med noe litteratur hvor denne faktoren modelleres. I delkapittel 7.3.3 beskrives det hvordan vi har gjort en tilnærming til denne egenskapen ved snutidene.



3.1.5 *Kostnadsfunksjon*

Vi har ikke foretatt noen grundig analyse av hvilke kostnadsdrivere som påvirkes av materiellturneringen og hvordan de påvirkes. Dette er i seg selv en stor oppgave som ligger utenfor rekkevidden for dette arbeidet. Kostnadsfunksjonene som vil bli brukt ved testingen har dermed delvis konstruerte verdier. En nærmere beskrivelse av disse finnes i delkapittel 3.2.3.

3.2 *Modellbeskrivelse*

Ulike modellformuleringer har blitt brukt for å beskrive materiellplanlegging for tog. Prosjektrapporten ”Materiellplanlegging for NSB med en operasjonsanalytisk vinkling” [Aschehoug og Fodstad, 2001] beskriver et utsnitt av disse. Ved arbeidet med å utvikle en modell for NSB's nærtrafikk har vi tatt utgangspunkt i modellen presentert av Schrijver (1993). Dessuten har vi integrert aspekter fra modellen til Cordeau, Soumis og Desrosiers (2000), Cordeau, Soumis og Desrosiers (2001a) og Cordeau, Desaulniers, Lingaya, Soumis og Desrosiers (2001b). I dette delkapitlet vil vi kort presentere disse modellene i tillegg til vår modellformulering. Vi starter med en utgave som ligger nær opp mot Schrijvers modell for så å beskrive to ulike utvidelser. Til slutt oppsummerer vi ved å vise hvordan modellen blir med begge utvidelsene integrert. Bakgrunnen for denne inkrementelle presentasjonen er knyttet til ønsket om å tydeliggjøre noen aspekter vi har valgt å modellere.

3.2.1 *Schrijver*

Inndata til Schrijvers modell (1993) er en ruteplan som gjentar seg daglig og etterspørselen gitt i setebehov for første og andre klasse tilknyttet hver tur i ruteplanen. Det antas at det i ruteplanen er lagt inn tid til skjøting, splitting og snuing i tilknytning til hver tur. Materiellet er definert ved en setekapasitet for første og andre klasse, og det antas at alle materielltyper er kompatible. Ved bruk av én materielltype kan etterspørselen konverteres fra setebehov til antall motorvognsett, men ved bruk av flere materielltyper finnes det ikke en slik unik konvertering. Dette fører til at modellen ikke er separabel med hensyn på materielltyper.



3.2.2 Enkel modellformulering

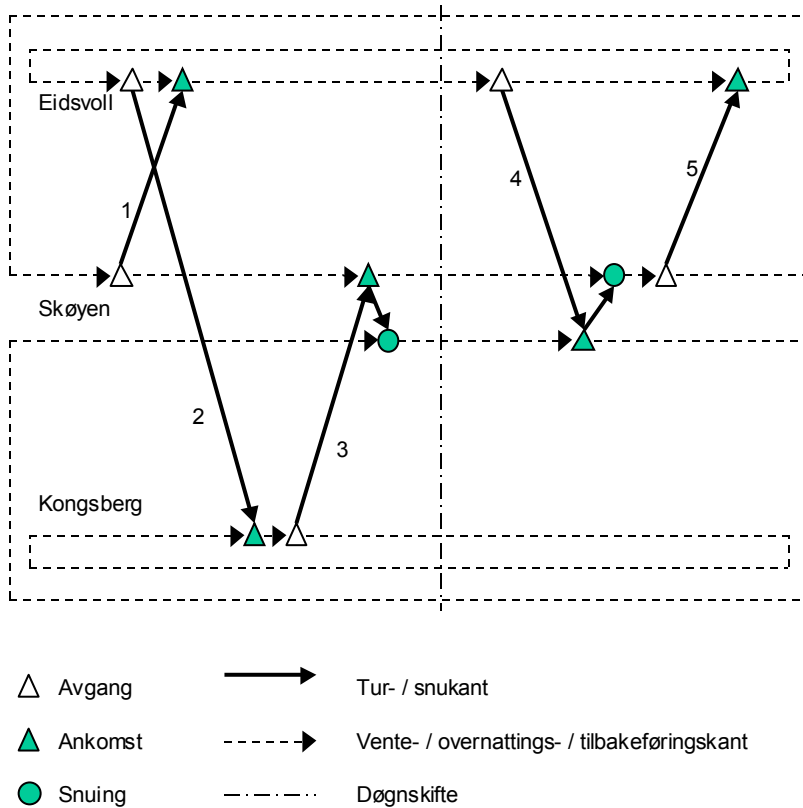
Den enkle utgaven av vår modellformulering er i stor grad lik modellen til Schrijver, dog med visse unntak. I stedet for å forlange at ruteplanen alltid skal gi tid til sning, modellerer vi sning eksplisitt på de stasjonene som ikke alltid fungerer som endestasjoner i nærtrafikken. Dette er gjort for å unngå og tilføre restriksjoner som ikke finnes i det reelle problemet, nemlig ekstra tid for sning når dette ikke er et behov. Konsekvensen av dette valget er at modellen blir noe større i form av flere beslutningsvariabler.

Notasjon

La K være settet av materielltyper og la L være settet av turer. En tur $l \in L$ er definert ved en avgangsstasjon, en ankomststasjon, en etterspørsel i form av antall vogner, et avgangstidspunkt og et ankomsttidspunkt. La V være settet av stasjoner og P være settet av døgnskifter.



Grafbeskrivelse



Figur 3-1: Graf for ruteplan over to dager

Det defineres en rettet tid-sted-graf $G=(N, A)$ der N er settet av noder i grafen og A er settet av kanter i grafen.

Settet av noder N

Det defineres noder for hver hendelse i tid og sted, det vil si at det for hver stasjon $v \in V$ defineres en node for hver ankomst og avgang på stasjonen. Dette gjelder både for ruteplanlagte turer og tomtogturer. For hver stasjon som kan ha avganger eller ankomster i flere kjøreretninger grupperes nodene etter kjøreretning. Videre defineres det for disse stasjonene for hver ankomst en snunode i motsatt kjøreretning med et tidspunkt lik summen av ankomsttidspunktet og snutiden. For hver materielltype $k \in K$ defineres det et subsett av noder $N^k \subseteq N$ som er knyttet til turer eller stasjonsopphold som tillates for materielltype k .



Tidspunktet for ankomstnodene er forskjøvet til et senere tidspunkt enn hva ruteplanen tilsier. Hensikten med dette er å tvinge modellen til å sette av tilstrekkelig tid mellom turene til på- og avstigning, bremseprøve og skjøting eller splitting. For endestasjonene lages det ikke snunoder. Tiden det tar å snu inkluderes i forskyvningen av ankomstnodene, og endestasjonene representeres derfor kun med én kjøreretning.

Settet av kanter A

For hver tur defineres en kant mellom tilhørende avgangs- og ankomstnode. Nedre grense for disse kantene er 0 dersom turen representerer tomtogkjøring. Ellers vil grensen bestemmes av antall vogner som definerer etterspørselen. Kantene nevnt så langt definerer subsettet $A_S \subseteq A$. Videre defineres det en snukant mellom hver ankomstnode og snunode. Disse kantene danner settet $A_{SNU} \subseteq A$. Øvre grense for alle turer og snukanter er gitt av den maksimale tog lengden u . For hver stasjon defineres en ventekant mellom hver node og den påfølgende noden i tid. De stasjonene som har noder gruppert etter kjøreretning får en slik kjede av ventekanter for hver kjøreretning. Ventekanter som går fra siste node en dag til første node påfølgende dag for en stasjon kalles overnattingskanter. Disse overnattingskantene inngår i settet $A^{pv} \subseteq A$, der $p \in P$ er døgnskifte. Øvrige ventekanter danner settet $A_V \subseteq A$. Både snukanter, ventekanter og overnattingskanter har nedre grense lik 0. Det defineres dessuten et subsett $A^k \subseteq A$ for hver materielltype $k \in K$ med de kantene som tillates dekket av k . For hver node $n \in N^k$ danner alle kantene inn i noden settet $I^n \subseteq A^k$ og alle kanter ut av noden danner settet $O^n \subseteq A^k$ slik at $I^n \cap O^n = \emptyset$.

Foruten kantene med en reell tolkning i virkeligheten, nevnt over, lages det kanter fra siste node til første node i planleggingsperioden for hver stasjon. For de stasjonene som har flere kjøreretninger lages det én slik tilbakeføringskant for hver retning.

Matematisk formulering

For hver kant $a \in A$ og hver materielltype $k \in K$ som tillates brukt på a , la x_a^k være en ikke-negativ heltallig variabel som representerer flyten av materielltype k langs kant



a , og la c_a^k representere kostnaden ved å bruke én enhet av materielltypen langs kant a . For hver kant $a \in A_S$, definer d_a som det minste antall vogner som trengs for å dekke etterspørselen. La u være det største antall vogner som kan være sammenkoblet for en tur. For hver materielltype $k \in K$, la m^k være antall vogner materielltypen består av. For hver stasjon $v \in V$, la h^v være det største antall vogner som har lov til å oppholde seg ved stasjonen om natten. For hver materielltype $k \in K$, la w^k være antall individer av materielltypen som er tilgjengelig. Modellen kan formuleres på følgende måte:

$$\underset{x_a^k}{Min} \quad \sum_{k \in K} \sum_{a \in A} c_a^k x_a^k \quad (1)$$

$$\sum_{k \in K} m^k x_a^k \geq d_a \quad , \forall a \in A_S \quad (2)$$

$$\sum_{k \in K} m^k x_a^k \leq u \quad , \forall a \in A_S \quad (3)$$

$$\sum_{a \in I^n} x_a^k = \sum_{a \in O^n} x_a^k \quad , \forall k \in K, n \in N^k \quad (4)$$

$$\sum_{k \in K} \sum_{a \in A^{pv}} m^k x_a^k \leq h^v \quad , \forall p \in P, v \in V \quad (5)$$

$$\sum_{v \in V} \sum_{a \in A^{pv}} x_a^k \leq w^k \quad , p = 1, \forall k \in K \quad (6)$$

$$x_a^k \geq 0, \text{ heltall} \quad , \forall k \in K, a \in A^k \quad (7)$$

(1) er målfunksjonen. (2) sørger for at nødvendig mengde materiell settes inn slik at etterspørselen dekkes, mens (3) sørger for at materiellmengden ikke overstiger den maksimale tog lengden for en tur. Restriksjonene (4) er flytkonserveringsrestriksjoner. I restriksjonene (5) sikres det at ikke hensettingskapasiteten på noen av stasjonene sprenges på nattertid, mens (6) sørger for at det ikke brukes mer materiell enn hva som er tilgjengelig. Senere i rapporten omtales restriksjonene som henholdsvis *etterspørsels-* (2), *tog lengde-* (3), *flytkonserverings-* (4), *stasjonskapasitets-* (5) og *materielltilgjengelighetsrestriksjoner* (6).



3.2.3 Kostnadsfunksjon

Parameteren c_a^k i kostnadsfunksjonen (1) kan dekomponeres i flere faktorer. Hvilke faktorer som inngår avhenger av hva slags kant a representerer. For kantene som representerer turer brukes følgende dekomponering:

$$c_a^k = l_a t_a f^k, \forall a \in A_S$$

der l_a er distansen for tur a , t_a er en vektning av kostnaden ved å kjøre tomtog i forhold til ruteplanlagte turer og f^k er en vektning per materielltype. For tilbakeføringskantene avhenger kostnaden kun av materielltype og har formen

$$c_a^k = f^k, \forall a \in A^{lv}$$

Kantene som representerer venting har ulik kostnad avhengig av hvilken stasjon de tilhører. Dette gir

$$c_a^k = q^v, \forall a \in A_V$$

For snukantene modelleres kun en fast kostnad s slik at

$$c_a^k = s, \forall a \in A_{SNU}$$

Kostnadsfunksjonen som helhet består dermed av fire ledd og kan uttrykkes slik:

$$\alpha \sum_{k \in K} \sum_{a \in A_S} l_a t_a f^k x_a^k + \beta \sum_{k \in K} \sum_{v \in V} \sum_{a \in A^{lv}} f^k x_a^k + \delta \sum_{k \in K} \sum_{a \in A_V} q^v x_a^k + \gamma \sum_{k \in K} \sum_{a \in A_{SNU}} s x_a^k$$

der α , β , δ og γ er parametere som angir vektningen av de ulike leddene i forhold til hverandre. Eksempelvis kan man rendyrke minimering av mengde materiell som brukes ved å sette α , δ og γ lik 0.

3.2.4 Cordeau et al.

Cordeau et al. (2000, 2001a, 2001b) beskriver to tilnærmet like modeller og disse vil her bli presentert som én modell. Denne bygger på de samme prinsippene som modellen til Schrijver, men har noen forskjeller som kort vil bli kommentert her. Cordeau et al. antar at etterspørselen er avhengig av materielltype slik at etterspørselsrestriksjonene er separable med hensyn på materielltype. Videre tillater



modellen til Cordeau et al. bruk av ulike materielltyper, også materielltyper som ikke kan skjøtes sammen med andre. Konseptet kompatible typer (consistent types) brukes til å beskrive hvilke materielltyper som tillates skjøtt sammen. Man unngår kravet om at alle turene i ruteplanen alltid skal ha tid til skjøting og splitting ved å innføre konseptet sekvens (sequences). En sekvens representerer flere etterfølgende turer uten skjøting eller splitting i mellom. En nærmere beskrivelse av disse to konseptene kommer i delkapittel 3.2.5. I tillegg modellerer Cordeau et al. sning eksplisitt slik som beskrevet i starten av delkapittel 3.2.2.

Foruten disse forskjellene fra Schrijvers modell, foreslår Cordeau et al. flere mulige utvidelser. De gir en formulering av et vedlikeholds krav, der dette er gitt i form av krav om besøk i vedlikeholdsdepoter innenfor bestemte tidsintervaller. I utgangspunktet antas det at alt vedlikehold foregår på natten, men det vises også hvordan formuleringen kan utvides slik at vedlikeholdet foregår i driftspauser på dagtid. Videre viser de en måte å straffe skjøting og splitting for å begrense bruken av slike aktiviteter. Det beskrives også hvordan man kan innføre grupper med et bestemt antall av ulike materielltyper som definerer grunnleggende enheter som brukes i trafikken (basic consists).

3.2.5 Utvidelser

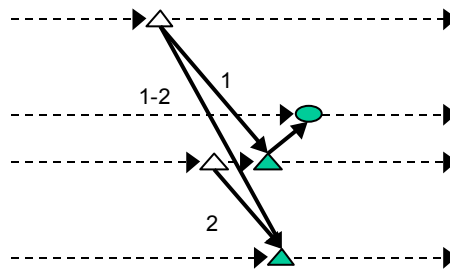
I dette delkapitlet vil vi presentere to mulige utvidelser av den enkle modellen presentert i delkapittel 3.2.2, henholdsvis *sekvenser* og *kompatible typer*. Hensikten med begge utvidelsene er å gjøre modellen mer sammenfallende med restriksjonene i det reelle problemet. Sekvenser brukes for å fjerne en kunstig restriksjon som modellen ellers ville inneholde, mens kompatible øker modellens bruksområde ved at ikke-kompatible materielltyper kan inngå i samme modell. Konsekvensen av begge utvidelsene, sammen eller hver for seg, er først og fremst at det innføres binærvariabler tilknyttet hver kant $a \in A_S$.

Sekvenser

Når grafene genereres legges tiden for skjøting eller splitting til for hver tur, slik at ankomstnoden forskyves til et senere tidspunkt. Dersom man ved denne forskyvningen bytter rekkefølge på ankomstnoden og en avgangsnode betyr det at



turene knyttet til disse nodene ikke regnes som kompatible. Dette kommer av at ankomsttidspunktet etter forskyvningen er etter avgangstidspunktet for den andre turen. I praksis er disse turene kompatible dersom man vet at det ikke skjer noen skjøting eller splitting mellom dem. For å ikke miste muligheten til å bruke disse løsningene innføres konseptet sekvens, som er en sammenkobling av turer. I figuren under er 1-2 en sekvens bestående av tur 1 og tur 2. I en sekvens legges det kun til tid for skjøting eller splitting én gang, ikke en gang for hver tur som inngår.



Figur 3-2: Eksempel på en sekvens

Sekvenser kan også brukes for å forhindre skjøting og splitting av andre grunner enn tidsmangel, eksempelvis hvis infrastrukturen ikke tillater det for enkelte par av turer.

Det defineres en ny kant for hver sekvens som genereres. Disse kantene inngår i settet $A_S \subseteq A$. En slik ny kant tillates kun brukt av en materielltype k dersom alle turene som dekkes av kanten kan kjøres med materielltype k . Nye kanter legges til i ett eller flere subsett A^k . Etterspørselen til kanten, d_a , settes lik den største etterspørselen til turene som inngår. Videre defineres en binærparameter e_a^l som er lik 1 hvis tur l inngår i kant $a \in A_S$, og 0 ellers.

For å integrere sekvenser i modell (1)-(7) må følgende restriksjoner legges til:

$$\sum_{a \in A_S} e_a^l y_a = 1 \quad , \forall l \in L \quad (8)$$

$$y_a \in \{0,1\} \quad , \forall a \in A_S \quad (9)$$

y_a er en variabel som er lik 1 dersom kant a skal brukes for å dekke tur l , og 0 ellers. Restriksjonene (8) sørger for at hver tur dekkes av én og bare én kant.



Videre må etterspørselsrestriksjonene (2) og tog lengderestriksjonene (3) endres til henholdsvis (10) og (11).

$$\sum_{k \in K} m^k x_a \geq d_a y_a \quad , \forall a \in A_S \quad (10)$$

$$\sum_{k \in K} m^k x_a \leq u y_a \quad , \forall a \in A_S \quad (11)$$

Kompatible typer

I modell (1)-(7) kan man ikke kontrollere hvilke materiellindivider som kobles sammen. Dermed kan ikke denne modellen brukes hvis man i samme modell ønsker å inkludere flere materielltyper som ikke kan skjøtes sammen. For å muliggjøre dette kan konseptet compatible typer tas i bruk. En kompatibel type $r \in R$ er et sett av materielltyper $\{k_i, k_{i+1}, \dots\} \subseteq K$ som kan skjøtes sammen. Videre defineres $R^a \subseteq R$ som settet av alle compatible typer som kan dekke kant a og $K^r \subseteq K$ som settet av alle materielltyper som inngår i kompatibel type r . Det defineres en binærparameter e_a^l som er lik 1 hvis tur l inngår i kant $a \in A_S$, og 0 ellers.

For å integrere compatible typer i modell (1)-(7) må følgende restriksjoner legges til:

$$\sum_{a \in A_S} \sum_{r \in R^a} e_a^l y^r = 1 \quad , \forall l \in L \quad (12)$$

$$y^r \in \{0,1\} \quad , \forall r \in R \quad (13)$$

y^r er en variabel som er lik 1 dersom kompatibel type r skal brukes på tur l , og 0 ellers. Restriksjonene (13) sørger for at hver tur dekkes av én og bare én kompatibel type.

Videre må etterspørselsrestriksjonene (2) og tog lengderestriksjonene (3) erstattes med henholdsvis restriksjonene (14) og (15).

$$\sum_{k \in K^r} m^k x_a \geq d_a^r y^r \quad , \forall r \in R \quad (14)$$

$$\sum_{k \in K^r} m^k x_a \leq u^r y^r \quad , \forall r \in R \quad (15)$$



3.2.6 Endelig modellformulering

Dersom både konseptet kompatible typer og sekvenser skal inkluderes i modellen vil y -variabelen både ha indeksen r og a . Videre defineres $R^a \subseteq R$ som settet av alle kompatible typer som tillates å dekke kanten $a \in A_S$. Hvis a er en sekvens vil R^a inneholder alle kompatible typer som alle turene i a tillater. En fullstendig formulering blir som følger:

$$\text{Min}_{x_a^k, y_a^r} \sum_{k \in K} \sum_{a \in A} c_a^k x_a^k \quad (16)$$

$$\sum_{a \in A_S} \sum_{r \in R^a} e_a^l y_a^r = 1, \quad \forall l \in L \quad (17)$$

$$\sum_{k \in K^r} m^k x_a^k \geq d_a^r y_a^r, \quad \forall a \in A_S, r \in R^a \quad (18)$$

$$\sum_{k \in K^r} m^k x_a^k \leq u^r y_a^r, \quad \forall a \in A_S, r \in R^a \quad (19)$$

$$\sum_{a \in I^n} x_a^k = \sum_{a \in O^n} x_a^k, \quad \forall k \in K, n \in N^k \quad (20)$$

$$\sum_{k \in K} \sum_{a \in A^{pv}} m^k x_a^k \leq h^v, \quad \forall p \in P, v \in V \quad (21)$$

$$\sum_{v \in V} \sum_{a \in A^{pv}} x_a^k \leq w^k, \quad p = 1, \forall k \in K \quad (22)$$

$$y_a^r \in \{0, 1\}, \quad \forall a \in A_S, r \in R^a \quad (23)$$

$$x_a^k \geq 0, \text{ heltall}, \quad \forall k \in K, a \in A^k \quad (24)$$

Siden denne modellen som inkluderer de to utvidelsene med sekvenser og kompatible typer beskriver det virkelige domenet for modellen mer presist enn modell (1)-(7) vil hovedfokus for implementasjon og testing knyttes til denne.



4 Operasjonsanalytisk teori

Antall beslutningsvariabler og restriksjoner i en modell har stor innvirkning på tiden det tar å finne en løsning. For store problemer er det ofte ønskelige å foreta operasjoner for å redusere løsningstiden. En mulighet er å dekomponere modellen i flere mindre modeller. For at en slik dekomponering skal ha noen hensikt må problemene man kommer fram til være enklere å løse enn det opprinnelige problemet. Ofte ønsker man derfor ved dekomponering å komme fram til delproblemer som har en kjent form med en tilhørende effektiv løsningsmetode. Her vil de klassiske teknikkene Lagrange-relaksering, Dantzig-Wolfe- og Benders dekomponering bli presentert. Det gis en mer utfyllende beskrivelse av Benders dekomponering enn de andre, da dette er den metoden som ble valgt til implementasjon.

4.1 Lagrange

Lagrange-relaksering er en måte å gjøre et problem lettere å løse ved å flytte restriksjoner som kompliserer problemet inn i målfunksjonen. Metoden brukes særlig innenfor ikke-lineær programmering og heltallsprogrammering. Fokus i denne framstillingen vil ligge på heltallsprogrammering.

Denne teknikken er strengt tatt ikke en dekomponeringsteknikk da man ikke deler opp det aktuelle problemet i flere delproblemer. Likevel kan teknikken brukes med samme hensikt som dekomponeringsteknikkene beskrevet i delkapitlene 4.2 og 4.3 og vil i denne rapporten i stor grad bli sidestilt med disse.

Gitt et heltallsproblem (G^p)

$$\underset{x}{\text{Min}} \quad \mathbf{c}^T \mathbf{x} \quad (1)$$

$$\mathbf{Ax} \geq \mathbf{b} \quad (2)$$

$$\mathbf{Bx} \geq \mathbf{d} \quad (3)$$

$$\mathbf{x} \in Z_+^n \quad (4)$$



der Z_+^n er settet av positive heltall i n dimensjoner. Her har restriksjonene (3) en kjent struktur med en kjent og effektiv løsningsalgoritme, mens restriksjonene (2) gjør løsning av problemet vanskeligere og ønskes derfor relaksert bort. En Lagrange-relaksering av (G^p) , (G_λ) , kan ha formen

$$\underset{x}{\text{Min}} \mathbf{c}^T \mathbf{x} + \boldsymbol{\lambda}^T (\mathbf{b} - \mathbf{A}\mathbf{x})$$

$$\mathbf{B}\mathbf{x} \geq \mathbf{d}$$

$$\mathbf{x} \in Z_+^n$$

$$\boldsymbol{\lambda} \in R_+^{m_1}$$

der $\boldsymbol{\lambda}$ kalles Lagrange-multiplikatorer og $R_+^{m_1}$ representerer den positive delen av det euklidiske rommet i m_1 -dimensjoner. m_1 tilsvarer her antall rader i matrisa A . Dette er en relaksjon av (G^p) siden restriksjonene (2) er fjernet og løsningsrommet til (G^p) dermed er et subsett av løsningsrommet til (G_λ) . Det igjen betyr at (G_λ) danner en nedre grense for (G^p) , noe man også ser av at $(\mathbf{b} - \mathbf{A}\mathbf{x}) \leq 0$ når restriksjonene (2) er oppfylt og $\boldsymbol{\lambda} \geq 0$. [Geoffrion, 1974]

Hvor stram nedre grense relaksjonen (G_λ) gir avhenger av valget av multiplikatorene $\boldsymbol{\lambda}$. Den strammeste grensen får man ved å bruke multiplikatorverdiene man får ved å løse problemet (G^d) gitt ved

$$\underset{\lambda}{\text{Max}} \mathbf{c}^T \mathbf{x} + \boldsymbol{\lambda}^T (\mathbf{b} - \mathbf{A}\mathbf{x})$$

$$\boldsymbol{\lambda} \in R_+^{m_1}$$

Dette tilsvarer Lagrange-dualen for (G^p) med hensyn på restriksjonene (2). Gitt at Lagrange-multiplikatorene settes lik dualverdiene for restriksjonene (2) fra lineærrelaksjonen av (G^p) og kalles $\bar{\boldsymbol{\lambda}}$. Da kan det vises at Lagrange-relaksjonen med $\bar{\boldsymbol{\lambda}}$, $(G_{\bar{\boldsymbol{\lambda}}}^p)$, alltid vil gi en like god eller bedre nedre grense enn lineærrelaksjonen selv. Altså

$$f(G_{\bar{\boldsymbol{\lambda}}}^p) \geq f(\bar{G}^p) \quad (5)$$



der $f(\cdot)$ representerer målfunksjonsverdien for det angitte problemet og \bar{G}^p representerer lineærrelaksjonen av (G^p) . En forutsetning for dette resultatet er at lineærrelaksjonen, (\bar{G}^p) , er løsbart. [Geoffrion, 1974]

Dersom problemet, (G^p) , har heltallsegenskapen som sikrer at den optimale løsningen på lineærrelaksjonen (\bar{G}^p) er lik den optimale løsningen for heltallsproblemet (G^p) kan det vises at $f(G^d) = f(\bar{G}^p)$. Altså at lineærrelaksjonen og Lagrange-relaksjonen gir samme nedre grense. Dette tilsier at man har mer å hente på å bruke Lagrange-relaksjonen framfor lineærrelaksjonen som nedre grense for problemer som ikke har denne heltallsegenskapen enn for problemer som har den. Bakgrunnen er at Lagrange-relaksjonen i de førstnevnte tilfellene gir en strammere nedre grense enn lineærrelaksjonen på grunn av (5). [Geoffrion, 1974]

Foruten å vite at man har en nedre grense, kan man med Lagrange-relaksjon kjenne størrelsen på gapet, ε , mellom denne grensen og den optimale målfunksjonsverdien. For at dette skal være mulig må følgende betingelser for \mathbf{x} være oppfylt:

- a) \mathbf{x} må være optimal i (G_λ)
- b) $\mathbf{Ax} \geq \mathbf{b}$

Under disse betingelsene er gapet $\varepsilon = \lambda^T (\mathbf{b} - \mathbf{Ax})$. Dersom $\varepsilon = 0$ har man altså en optimal løsning på (G^p) . [Geoffrion, 1974]

Egenskapene ved Lagrange-relaksjonen beskrevet her kan utnyttes for å gjøre løsning av problemer mer effektiv. En mulighet er å utnytte relaksjonen til en heuristisk tilnærming til et problem, og samtidig vite hvor mye man har å hente på å løse problemet til optimum. En alternativ bruksmåte for et heltallsproblem er å kombinere Lagrange-relaksjon med branch and bound. Tradisjonelt løses lineærrelaksjonen til et heltallsproblem i hver node i et branch and bound-tre, men man kan bruke Lagrange-relaksjonen i stedet. Denne relaksjonen oppfyller de nødvendige kravene ved at (G^p) er uløselig dersom (G_λ) er uløselig, en optimal løsning for (G_λ) er optimal for (G^p) dersom betingelsene a) og b) er oppfylt og $\varepsilon = 0$ og dersom $f(G_\lambda) \geq \bar{z}$ er $f(G^p) \geq \bar{z}$ når \bar{z} representerer beste kjente løsning i branch and bound-prosessen.



Samtidig tilsier (5) at man har en strammere grense og fortere kan finne en god \bar{z} som gir mulighet til å avskjære greiner før alle noder lenger ned i greina er undersøkt.

4.2 Dantzig-Wolfe

Dantzig-Wolfe dekomponering bygger på prinsipper fra Dantzig og Wolfe (1960) hvor det beskrives hvordan et lineært problem kan dekomponeres i subproblemer og et koordinerende masterproblem.

Et problem

$$\begin{aligned} \underset{\mathbf{x}}{\text{Min}} \quad & \mathbf{c}^T \mathbf{x} \\ & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{k} \leq \mathbf{x} \leq \mathbf{l} \end{aligned}$$

kan dekomponeres til

$$\begin{aligned} \underset{\mathbf{x}}{\text{Min}} \quad & \mathbf{c}^T \mathbf{x} \\ & \mathbf{A}'\mathbf{x} = \mathbf{b}' \\ & \mathbf{A}''\mathbf{x} = \mathbf{b}'' \quad (1) \\ & \mathbf{k} \leq \mathbf{x} \leq \mathbf{l} \end{aligned}$$

der \mathbf{A}' og \mathbf{A}'' er en disjunkt oppdeling av \mathbf{A} og \mathbf{b}' og \mathbf{b}'' en tilsvarende disjunkt oppdeling av \mathbf{b} . En løsning, \mathbf{x} , som tilfredsstiller restriksjonene (1) kan uttrykkes som en lineær kombinasjon av de lovlige basisene, $\mathbf{v}^i \in M$, og de lovlige basiske retningene, $\mathbf{w}^i \in N$, for restriksjonene. I en løsning for det totale problemet skal kun en av de $|M|$ basisene velges. Dermed kan problemet uttrykkes som følger

$$\underset{\mathbf{v}, \mathbf{w}}{\text{Min}} \quad \sum_{i \in M} \mathbf{c}^T \mathbf{v}^i r^i + \sum_{i \in N} \mathbf{c}^T \mathbf{w}^i s^i \quad (2)$$



$$\sum_{i \in M} \mathbf{A}' \mathbf{v}^i r^i + \sum_{i \in N} \mathbf{A}' \mathbf{w}^i s^i = \mathbf{b}' \quad (3)$$

$$\sum_{i \in M} r^i = 1 \quad (4)$$

$$r^i \geq 0 \quad , 1 \leq i \leq |M| \quad (5)$$

$$s^i \geq 0 \quad , 1 \leq i \leq |N| \quad (6)$$

Problemet (2)-(6) er det såkalte masterproblemet i dekomponeringen. Dette problemet har typisk færre rader enn det opprinnelige problemet, men betydelig flere kolonner. Subproblemet er det problemet som må løses for å komme fram til settene M og N som altså gir disse kolonnene. Dersom man har separabilitet i restriksjonene (1) kan man løse flere subproblemer i stedet for ett stort. [Chvátal, 1983]

En gunstig egenskap med denne dekomponeringen er at man ikke behøver å finne alle kolonnene til masterproblemet på forhånd. I stedet kan man iterere mellom å løse subproblemet og et begrenset masterproblem som inneholder et subsett av M og N . Hvert subproblem som løses gir en ny kolonne som legges til masterproblemet. For at løsningsprosessen skal kreve færrest mulig iterasjoner er det ønskelig å finne fram til de kolonnene som har lovende egenskaper med tanke på å forbedre løsningen i masterproblemet. Dette oppnår man ved å utforme målfunksjonen til subproblemet slik at den minimerer den reduserte kostnaden. Dermed får subproblemet følgende form

$$\underset{\mathbf{x}}{\text{Min}} \quad (\mathbf{c} - \boldsymbol{\pi}' \mathbf{A}')^T \mathbf{x}$$

$$\mathbf{A}'' \mathbf{x} = \mathbf{b}''$$

$$\mathbf{k} \leq \mathbf{x} \leq \mathbf{l}$$

der $\boldsymbol{\pi}'$ er dualvariablene fra masterproblemet. Optimal løsning for masterproblemet og dermed også for det opprinnelige problemet er nådd når subproblemets optimale verdi er større enn null. Dette fordi en slik løsning betyr at det ikke finnes flere kolonner som ikke allerede er inkludert i masterproblemet og som kan forbedre masterproblemløsningen. [Chvátal, 1983]



4.3 Bender

Benders dekomponering er en metode som baserer seg på prinsippene presentert i Bender (1962). Målsetningen er å kunne løse et blandet heltallsproblem effektivt ved å utnytte strukturer i problemet.

4.3.1 Metoden

Et blandet heltallsproblem kan formuleres som følger

$$\begin{aligned}
 \underset{y,x}{\text{Min}} \quad & \mathbf{c}^T \mathbf{y} + \mathbf{q}^T \mathbf{x} \\
 & \mathbf{T}\mathbf{y} + \mathbf{W}\mathbf{x} = \mathbf{h} \\
 & \mathbf{A}\mathbf{y} = \mathbf{b} \\
 & \mathbf{y} \in R^{n_1} \\
 & \mathbf{x} \in R^{n_2}
 \end{aligned}$$

der, R^{n_1} og R^{n_2} representerer hele det euklidiske rommet i henholdsvis n_1 og n_2 dimensjoner.

Ved å dele settet av beslutningsvariabler i to sett og legge til en variabel, θ , kan dette problemet omformuleres til et master problem

$$\begin{aligned}
 \underset{y,\theta}{\text{Min}} \quad & \mathbf{c}^T \mathbf{y} + \theta \\
 & \theta \geq G(\mathbf{y}) \\
 & \mathbf{A}\mathbf{y} = \mathbf{b} \\
 & \mathbf{y} \in R^{n_1}
 \end{aligned}$$

og et primalt subproblem



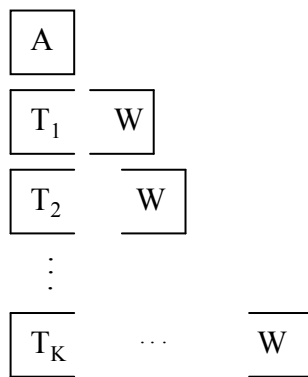
$$G^p(\bar{y}) = \underset{x}{Min} \quad \mathbf{q}^T \mathbf{x}$$

$$\mathbf{T}\bar{y} + \mathbf{W}\mathbf{x} \geq \mathbf{h}$$

$$\mathbf{x} \in R_+^{n_2}$$

der streken over y angir at y er en konstant i subproblemet. $R_+^{n_2}$ er den positive delen av R^{n_2} . Metoden forutsetter at dette problemet er konvekst. [Bender, 1962]

Denne måten å dekomponere på er særlig nyttig dersom strukturen i problemet gir en naturlig oppdeling av beslutningsvariablene. Det kan eksempelvis være en struktur av separable blokker som kun bindes sammen av y -variablene. Dette er illustrert i Figur 4-1 hvor man har K blokker som bindes sammen i leddene T_k . En slik struktur gir ett subproblem for hver blokk. Formen på denne blokkstrukturen er bakgrunnen for at metoden, særlig innenfor stokastisk programmering, ofte kalles "L-shaped" metoden. [Birge og Louveaux, 1997]



Figur 4-1: Koeffisientmatrise med blokkstruktur [Birge og Louveaux, 1997]

Masterproblemet er i praksis en relaxering av det opprinnelige problemet siden enkelte restriksjoner er fjernet. Ved løsning av problemet legges restriksjonene inn i masterproblemet, men dette gjøres gradvis. Hensikten med dette er å legge til kun de restriksjonene som er nødvendige for å finne den optimale løsningen. Man itererer mellom å løse masterproblemet og dualen til subproblemene med y -variablene låst. På grunn av relaxeringene i masterproblemet finnes det ikke nødvendigvis en løsning på subproblemet for en gitt løsningen av masterproblemet. I slike tilfeller legges en av de



relakserte restriksjonene inn i masterproblemet i form av et lovlighetskutt. Dette kuttet fjerner en del av løsningsrommet for masterproblemet som inneholder løsninger som gjør et subproblem uløselig. Kuttet har formen $\boldsymbol{\sigma}_k^T \mathbf{T}_k \mathbf{y} \geq \boldsymbol{\sigma}_k^T \mathbf{h}_k$ der k angir hvilket subproblem dualvariablene $\boldsymbol{\sigma}_k$ kommer fra. For å ikke bare få en gyldig løsning, men en optimal løsning legges det til optimalitetskutt i masterproblemet. Disse har formen $\sum_{k \in K} \boldsymbol{\pi}_k^T \mathbf{T}_k \mathbf{y} + \theta \geq \sum_{k \in K} \boldsymbol{\pi}_k^T \mathbf{h}_k$. [Birge et al., 1997] En nærmere forklaring av kuttene kommer nedenfor.

Benders dekomponering kan både brukes til å finne optimal løsning og til å finne en nedre grense (eventuelt øvre grense for et maksimeringsproblem). En slik grense oppnår man ved å avbryte itereringen mellom masterproblem og subproblem før optimum er nådd, eksempelvis når forbedringen i målfunksjonsverdi for hver iterasjon er liten. [Birge et al., 1997]

Lovlighetskutt

For å forstå gyldigheten til et lovlighetskutt trengs noen teoretiske byggesteiner. Som nevnt i delkapittel 4.2 kan løsningen til et problem $\min \{ \mathbf{c}^T \mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq 0 \}$ uttrykkes ved en lineærkombinasjon av basisene, kalt \mathbf{v} , og de basisk gyldige retningene \mathbf{w} . Basisene spenner det konvekse hylsteret, \mathbf{P} , og de basisk gyldige retningene spenner en konveks mangesidet kjegle, \mathbf{C} . \mathbf{C} omtales gjerne som det positive hylsteret eller pos \mathbf{A} . Til sammen danner \mathbf{P} og \mathbf{C} et mangesidet sett, \mathbf{B} , som utgjør det gyldige utfallsrommet for \mathbf{x} . [Kall og Wallace, 1994]

Videre er Farkas' lemma sentralt for forståelsen av lovlighetskutt:

$$\mathbf{B} = \{ \mathbf{x} \mid \mathbf{W}\mathbf{x} = \mathbf{h} - \mathbf{T}\mathbf{y}, \mathbf{x} \geq 0 \} \neq \emptyset \quad (1)$$

$$\Downarrow$$

$$\mathbf{W}^T \mathbf{u} \leq 0 \Rightarrow (\mathbf{h} - \mathbf{T}\mathbf{y})^T \mathbf{u} \geq 0 \quad (2)$$

Et bevis for dette finnes i [Kall og Wallace, 1994].

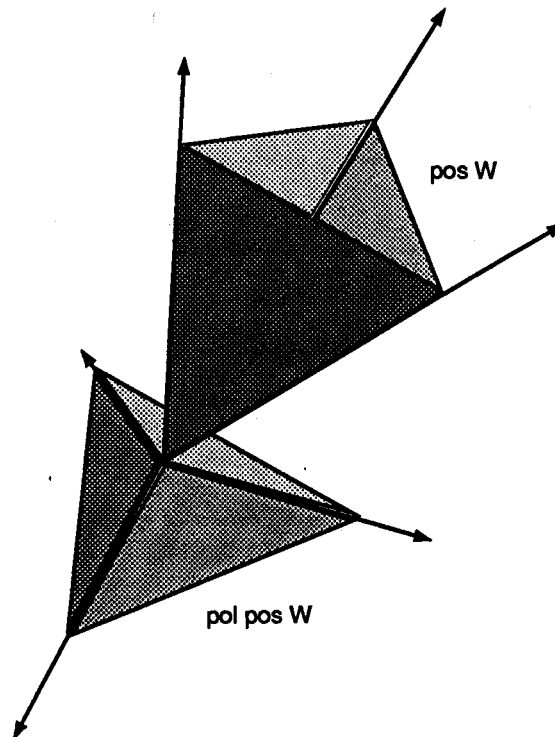
Med utgangspunkt i en løsning av masterproblemet $\bar{\mathbf{y}}$, ønsker man å vite hvorvidt subproblemene har løsninger. Dersom problemene ikke er løsbare ønsker man å finne en restriksjon som kan legges til masterproblemet og som kan tvinge masterproblemet



vekk fra løsningen som gjør et subproblem uløselig. (1) fra Farkas' lemma sier nettopp at et problem skal ha minst en løsning. Dette betyr det samme som at $(\mathbf{h} - \mathbf{T}\bar{\mathbf{y}}) \in \text{pos } \mathbf{W}$. Betingelsen (2) kan alternativt uttrykkes på følgende måte

$$\mathbf{t} \in \text{pol pos } \mathbf{W} \Rightarrow (\mathbf{h} - \mathbf{T}\bar{\mathbf{y}})^T \mathbf{t} \leq 0 \quad (3)$$

der $\text{pol pos } \mathbf{W} = \{\mathbf{u} \mid \mathbf{u}^T (\mathbf{h} - \mathbf{T}\bar{\mathbf{y}}) \leq 0, \forall (\mathbf{h} - \mathbf{T}\bar{\mathbf{y}}) \in \text{pos } \mathbf{W}\}$



Figur 4-2:Kjeglene $\text{pos } \mathbf{W}$ og $\text{pol pos } \mathbf{W}$ [Kall et al., 1994]

$\text{pol pos } \mathbf{W}$ er en konveks mangesidet kjegle som vist i Figur 4-2 og \mathbf{t} representerer strålene i kjeglen. Kravet til en gyldig løsning er altså i følge (3) at produktet mellom enhver slik stråle og $(\mathbf{h} - \mathbf{T}\bar{\mathbf{y}})$ skal være ikke-positiv. Dette kan forstås geometrisk ved at hver stråle som spenner $\text{pol pos } \mathbf{W}$ ligger 90° på en flate i $\text{pos } \mathbf{W}$. Dermed vil enhver stråle \mathbf{t} i $\text{pol pos } \mathbf{W}$ ha en vinkel på minimum 90° i forhold til alle stråler i $\text{pos } \mathbf{W}$. Fra matematikken vet vi at prikk-produktet mellom to vektorer som har en vinkel på 90° - 270° mellom seg er mindre eller lik 0. For at $(\mathbf{h} - \mathbf{T}\bar{\mathbf{y}})$ skal ligge i det gyldige området $\text{pos } \mathbf{W}$ må dermed produktet med alle stråler i $\text{pol pos } \mathbf{W}$, inkludert \mathbf{t} , være mindre eller lik 0.



Siden pol pos \mathbf{W} er konveks er det tilstrekkelig å sjekke at dette gjelder for strålene som spenner kjeglen. Disse strålene er vanligvis ikke kjent, men man kan løse et problem som enten bekrefter at betingelsen er oppfylt eller gir den strålen i pol pos \mathbf{W} som er lengst fra å oppfylle betingelsen. Problemet kan formuleres som følger

$$\max_{\sigma} \sigma^T (\mathbf{h} - \mathbf{T}\bar{\mathbf{y}}) \quad (4)$$

$$\sigma^T \mathbf{W} \leq 0 \quad (5)$$

$$\|\sigma\| \leq 1 \quad (6)$$

(5) tilsvarer definisjonen av pol pos \mathbf{W} og (6) sørger for at lengden på strålen, σ , normeres til 1 ettersom det kun er retningen som er interessant. Målfunksjonen (4) maksimerer uttrykket som skal være ikke-positivt og sørger dermed for at strålen σ som velges fra pol pos \mathbf{W} i størst mulig grad bryter betingelsen (3). Dersom målfunksjonsverdien er positiv ligger ikke $(\mathbf{h} - \mathbf{T}\bar{\mathbf{y}})$ i pos \mathbf{W} og det må lages et lovlighetskutt. På grunn av maksimeringen og at pol pos \mathbf{W} er konveks vil alltid den av strålene som inngår i settet som spenner pol pos \mathbf{W} og som dessuten gir grunnlag for den strammeste restriksjonen velges.

Dualen til problem (4)-(6) kan gis på følgende form

$$Q(\bar{\mathbf{y}}) = \underset{\mathbf{x}, \mathbf{v}^+, \mathbf{v}^-}{\text{Min}} \mathbf{I}\mathbf{v}^+ + \mathbf{I}\mathbf{v}^-$$

$$\mathbf{T}\bar{\mathbf{y}} + \mathbf{W}\mathbf{x} + \mathbf{I}\mathbf{v}^+ - \mathbf{I}\mathbf{v}^- = \mathbf{h}$$

$$\mathbf{x} \in R_+^{n_2}$$

$$\mathbf{v}^+ \in R_+^{m_2}$$

$$\mathbf{v}^- \in R_+^{m_2}$$

der \mathbf{I} er identitetsmatrisa. $Q(\bar{\mathbf{y}})$ omtales gjerne som fase I problemet for et subproblem. Variablene \mathbf{v}^+ og \mathbf{v}^- sørger for at problemet alltid er løsbart. I tråd med dualitetsteoremet har $Q(\bar{\mathbf{y}})$ samme verdi som (4)-(6) i optimum, hvilket innebærer at



$Q(\bar{\mathbf{y}}) = 0$ for en gyldig $\bar{\mathbf{y}}$. For å sjekke gyldigheten til $\bar{\mathbf{y}}$ må et slikt fase I problem løses for hver k og et lovlighetskutt på formen $\boldsymbol{\sigma}_k^T \mathbf{T}_k \mathbf{y} \geq \boldsymbol{\sigma}_k^T \mathbf{h}_k$ legges til i masterproblemet dersom målfunksjonsverdien er større enn 0. [Kall et al., 1994]

Det finnes et endelig antall stråler som spenner pol pos \mathbf{W} og dermed finnes det også et endelig antall lovlighetskutt som kan legges til. [Kall et al., 1994]

Optimalitetskutt

Optimalitetskuttene baserer seg på svak dualitet som innebærer at $G^d(\mathbf{y}) \leq G^p(\mathbf{y})$ og sterk dualitet som innebærer at $G^d(\mathbf{y}) = G^p(\mathbf{y})$ i optimum når $G^p(\mathbf{y})$ er et minimeringsproblem og $G^d(\mathbf{y}) = \{ \max \boldsymbol{\pi}^T (\mathbf{h} - \mathbf{T}\mathbf{y}) \mid \boldsymbol{\pi}^T \mathbf{W} \leq \mathbf{q}^T \}$ dualproblemet til $G^p(\mathbf{y})$ [Rardin, 1998]. Videre utnytter man at subproblemene $G(\mathbf{y})$ er konvekse og stykkevis lineære, hvor hvert lineært stykke representerer en basis for problemet. [Kall et al., 1994]

Det antas her at de primale subproblemene er bundet, noe som i følge dualitetsteorien samtidig innebærer at dualproblemet er løsbart. Dette kan kontrolleres før løsningsprosessen starter. For å lage et optimalitetskutt løses et subproblem $G(\mathbf{y})$ for en gitt løsning av masterproblemet, $\bar{\mathbf{y}}$. Etersom dette skjer etter at $\bar{\mathbf{y}}$ er sjekket for lovlighet vet man at subproblemet er løsbart. Man får dualvariablene $\bar{\boldsymbol{\pi}}$ og dual målfunksjonsverdi $\bar{\boldsymbol{\pi}}^T (\mathbf{h} - \mathbf{T}\bar{\mathbf{y}})$. Etersom $\bar{\mathbf{y}}$ ikke inngår i restriksjonene til dualproblemet er $\bar{\boldsymbol{\pi}}$ gyldig uansett \mathbf{y} . Ut i fra dualitetsteorien har man dermed at $\boldsymbol{\pi}^T (\mathbf{h} - \mathbf{T}\mathbf{y}) \geq \bar{\boldsymbol{\pi}}^T (\mathbf{h} - \mathbf{T}\bar{\mathbf{y}})$, ettersom $\bar{\boldsymbol{\pi}}$ er gyldig for alle \mathbf{y} , men ikke nødvendigvis optimal. Dette betyr at $\bar{\boldsymbol{\pi}}^T (\mathbf{h} - \mathbf{T}\bar{\mathbf{y}})$ er et hyperplan som tangerer $G(\mathbf{y})$ for $\bar{\mathbf{y}}$. Siden θ representerer målfunksjonsverdien for et subproblem i masterproblemet er $\theta \geq \bar{\boldsymbol{\pi}}^T (\mathbf{h} - \mathbf{T}\bar{\mathbf{y}})$ et gyldig kutt. [Kall et al., 1994]

Optimalitetskutt kan legges til masterproblemet i to alternative former. Enten legger man til ett kutt for hvert subproblem. Dermed får man også en kontinuerlig variabel θ_k for hvert subproblem som alle summeres i masterproblemets målfunksjon. Denne varianten kalles multikutt fordi man kan legge til flere kutt mellom hver gang masterproblemet løses. [Birge et al., 1997] Alternativt kan man summere sammen



målfunksjonsverdiene for subproblemene i ett kutt $\theta \geq \sum_{k \in K} \bar{\pi}_k^T (\mathbf{h}_k - \mathbf{T}_k \bar{\mathbf{y}})$. Dette er gyldig siden en summering av konvekse og stykkevis lineære funksjoner vil gi en ny konveks og stykkevis lineær funksjon slik at $\sum_{k \in K} \pi_k^T (\mathbf{h}_k - \mathbf{T}_k \mathbf{y}) \geq \sum_{k \in K} \bar{\pi}_k^T (\mathbf{h}_k - \mathbf{T}_k \mathbf{y})$ er oppfylt. [Kall et al., 1994]

Dersom $\theta \geq G(\bar{\mathbf{y}})$ når subproblemene er løst er optimum nådd. Denne betingelsen gjelder for alle subproblemene for seg eller summen av alle subproblemene, avhengig av om man bruker multikutt eller ikke. Metoden vil konvergere i et endelig antall steg fordi man har et endelig antall basiser i subproblemene. [Kall et al., 1994]

Algoritmen

Gangen i L-shaped metoden er som følger:

```

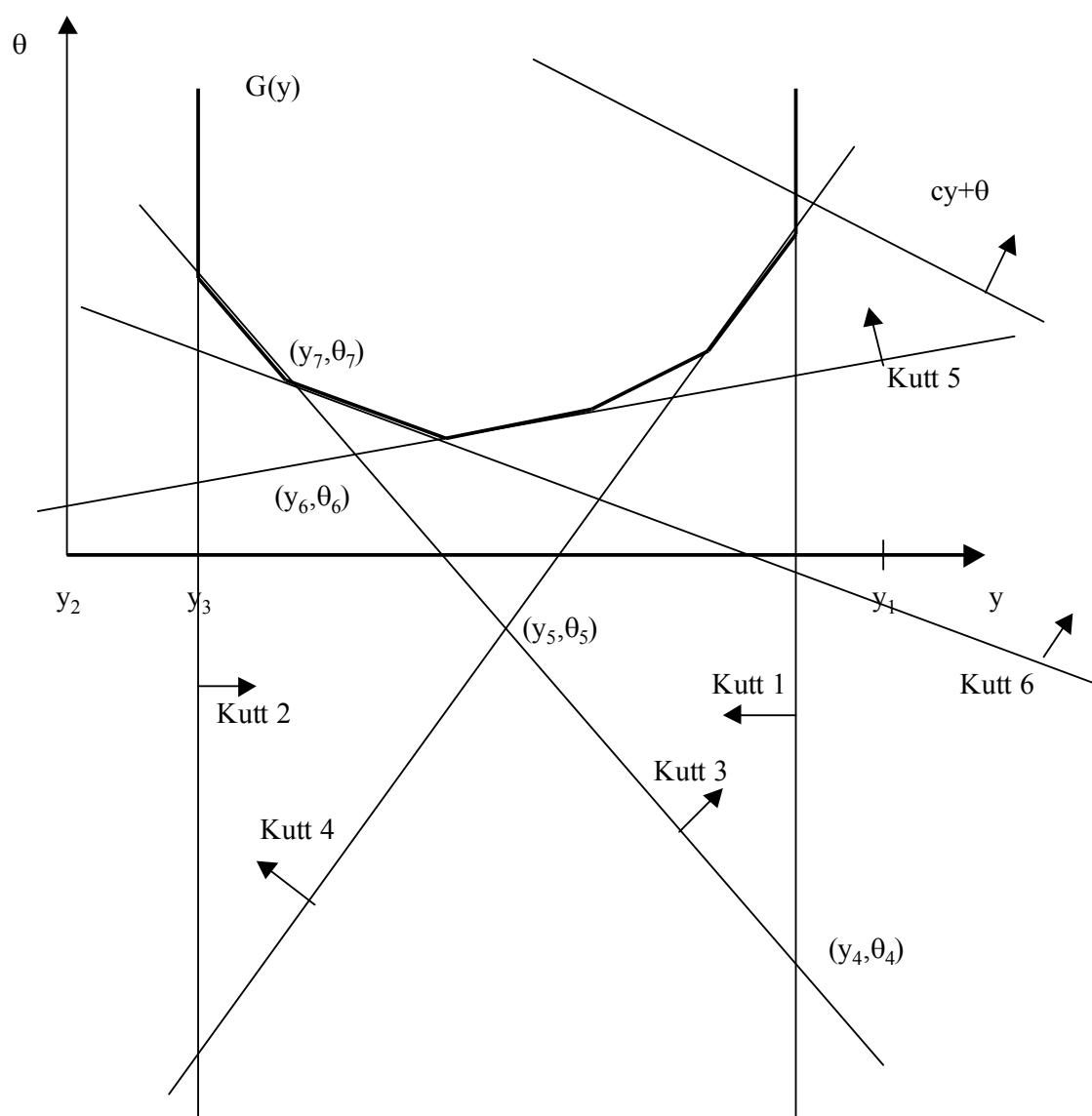
forAll(k)  $\theta_k = -\infty$ 
while(forAll(k)  $\theta_k < G_k^P(\bar{\mathbf{y}})$ )
{
    løs masterproblemet
    forAll(k)
    {
        løs fase I problemet  $Q_k(\bar{\mathbf{y}})$ 
        if( $Q_k(\bar{\mathbf{y}}) > 0$ )
            legg til et lovlighetskutt for  $k$ 
    }
    if(nytt lovlighetskutt)
        continue while
    forAll(k)
    {
        løs det primale subproblemet  $G_k^P(\bar{\mathbf{y}})$ 
        if( $\theta_k < G_k^P(\bar{\mathbf{y}})$ )
            legg til optimalitetskutt
    }
}

```



4.3.2 Eksempel

Et eksempel på bruk av Benders dekomponering til å løse et problem er vist i. Linja merket $cy+\theta$ angir hvilken retning målfunksjonen øker, og ettersom det illustrerte eksempelet er et minimeringsproblem er det ønskelig med en løsning som ligger motsatt vei av hva pila viser. y_l er valgt tilfeldig og $\theta_1 = \theta_2 = \theta_3 = -\infty$. Indeksene angir rekkefølge kuttene legges til. Kutt 1 og 2 er lovlighetskutt, mens de resterende kuttene er optimalitetskutt. Optimal løsning finnes i punktet merket (y_7, θ_7) . [Kall et al., 1994]



Figur 4-3: Eksempel på løsning med Benders dekomponering [Kall et al., 1994]



4.3.3 Benders dekomponering og heltallskrav

Benders dekomponering kan brukes på problemer som har heltallskrav for noen eller alle beslutningsvariabler. I slike tilfeller kan en klassisk branch and bound algoritme brukes til å oppnå en heltallig løsning på problemet. Problemet i hver node i branch and bound-treet løses da ved hjelp av Benders dekomponering. [Laporte og Louveaux, 1993] I Laporte og Louveaux (1993) bevises det også at optimalitetskutt generert for lineærrelaksjonen til et problem er gyldig når heltallskravet inkluderes. Siden løsningsrommet for et heltallsproblem er et subset av løsningsrommet til den tilsvarende lineærrelaksjonen, må også lovlighetskuttene generert fra lineærrelaksjonen være gyldige for heltallsproblemet.

Selv om kuttene generert ut i fra lineærrelaksjonen er gyldige, er de ikke nødvendigvis stramme for et heltallsproblem. Carøe og Tind (1997) presenterer L-shaped metoden for heltallsproblemer basert på generell dualitetsteori, noe som gir grunnlag for strammere kutt. I utgangspunktet forutsetter L-shaped metoden at subproblemet er konvekst for å kunne definere subproblemene inn i masterproblemet gjennom bruk av tangerende hyperplan som bygger på dualvariabler. Når subproblemene inneholder heltallsrestriksjoner er disse ikke konvekse. For å kunne lage lovlighetskuttene og optimalitetskuttene brukes derfor dualfunksjoner og dualproblemet for subproblemet $G(\mathbf{y})$ har da følgende form

$$\underset{F}{\text{Max}} F(\mathbf{d})$$

$$F(\mathbf{W}\mathbf{x}) \leq \mathbf{q}\mathbf{x}, \forall \mathbf{x} \in Z_+^{n_2}$$

$$F \in B$$

der $Z_+^{n_2}$ er settet av positive heltall i n_2 dimensjoner, $\mathbf{d} = \mathbf{h} - \mathbf{T}\mathbf{y}$, $\mathbf{d} \in R^{m_2}$ og B er settet av dualfunksjoner.

Ulike algoritmer for heltallsprogrammering genererer ulike slike funksjoner. I Wolsey (1981) beskrives funksjonen som dannes ved branch and bound. Denne har følgende form



$$F(\mathbf{d}) = \underset{i=1, \dots, P}{\text{Min}} \{ \boldsymbol{\pi}^i \mathbf{d} + \mathbf{b}^i \}$$

$$\boldsymbol{\pi}^i = (\pi_1^i, \dots, \pi_{m_2}^i) \geq 0$$

der P er settet av løvnoder i det aktuelle branch and bound-treet som dannes på følgende måte: Gjennom en branch and bound-prosess legges det til øvre og nedre grenser for \mathbf{y} -variablene slik at subproblemet får formen

$$\underset{\mathbf{x}}{\text{Min}} \mathbf{q}^T \mathbf{x}$$

$$\mathbf{T}\bar{\mathbf{y}} + \mathbf{W}\mathbf{x} \geq \mathbf{h}$$

$$\mathbf{k}^i \leq \mathbf{x} \leq \mathbf{l}^i$$

der \mathbf{k}^i og \mathbf{l}^i er henholdsvis nedre og øvre grenser for løvnode $i \in P$ i branch and bound-treet. Hver løvnode inneholder altså et sett av øvre og nedre grenser som danner grunnlag for ytterligere dualvariabler for subproblemet ut over $\boldsymbol{\pi}$. Disse benevnes henholdsvis $\underline{\boldsymbol{\pi}}$ og $\bar{\boldsymbol{\pi}}$. Dualfunksjonen for en løvnode i defineres som

$f_i(\mathbf{d}) = \boldsymbol{\pi}^i \mathbf{d} + \underline{\boldsymbol{\pi}}^i \mathbf{k}^i + \bar{\boldsymbol{\pi}}^i \mathbf{l}^i = \boldsymbol{\pi}^i \mathbf{d} + \mathbf{b}^i$. Transponeringstegnet er heretter utelatt for å bedre lesbarheten. Et lovlighetskutt for subproblem k dannet på grunnlag av en dualfunksjon ser dermed slik ut

$$\sigma_k^i T_k y \geq \sigma_k^i h_k + \underline{\sigma}_k^i k^i + \bar{\sigma}_k^i l^i$$

Tilsvarende ser et optimalitetskutt ved bruk av multi-kutt slik ut:

$$\pi_k^i T_k y + \theta_k \geq \pi_k^i h_k + \pi_k^i k^i + \bar{\pi}_k^i l^i$$

[Carøe et al. 1997]

I branch and bound-prosessen deles løsningsrommet for subproblemet inn i $|P|$ disjunkte løsningsrom, ett for hver løvnode. Målet er at løsningen skal ligge i ett av disse løsningsrommene, noe som betyr at settet av heltallskutt for en løvnode skal oppfylles. Dette innebærer også at lovlighetskuttet eller optimalitetskuttet knyttet til



kun denne løvnoden må oppfylles, noe som også gjenspeiles i valget av én dualfunksjon i uttrykket for $F(\mathbf{d})$ over. Uttrykket har form som et Gomory-kutt som kan legges til i masterproblemet på følgende måte: Hvert av de $|P|$ kuttene legges til masterproblemet sammen med en ny binærvariabel. Samtidig legges det til en ekstra restriksjon som sørger for at minst et av kuttene oppfylles. Et Gomory-kutt ser for eksempel slik ut:

$$\sigma_k^i T y \geq \sigma_k^i h_k + \sigma_k^i k^i + \bar{\sigma}_k^i l^i - M^i (1 - \delta^i) \quad , \forall i \in P \quad (1)$$

$$\sum_{i \in P} \delta^i \geq 1$$

$$\delta^i \in \{0, 1\} \quad , \forall i \in P$$

der δ^i er nye binærvariabler og M^i er et tilstrekkelig stor tall slik at (1) alltid er oppfylt for $\delta^i=0$. Tilsvarende Gomory-kutt lages også for optimalitetskuttene. [Birge et al., 1997]

Gitt at branch and bound-prosessen terminerer i et endelig antall steg og masterproblemet består av heltallsvariabler, så vil også denne versjonen av metoden konvergere i et endelig antall steg. Antall mulige branch and bound-trær er endelig, men vokser eksponentielt med antall subproblemvariabler, n_2 . [Carøe et al. 1997]



5 Løsningsmetoder og dekomponering

Valg av løsningsmetode bør avhenge av det aktuelle problemet slik at man kan dra nytte av eventuelle gunstige struktur problemet har. Vurderingene som leder frem til et valg av dekomponeringsstrategi er knyttet opp i mot den endelige modellformuleringen i delkapittel 3.2.6, da dette er den modellen som mest presist beskriver virkeligheten. I dette kapitlet viser vi hvordan problemet kan formuleres for hver av de ulike dekomponeringene som ble presentert i forrige kapittel.

5.1 To løsningsmetoder

Det finnes en rekke kommersielle programvarer som løser operasjonsanalytiske problemer. Disse gjør det mulig å beskrive og løse problemer med betydelig mindre arbeidsinnsats enn hva en selvimplementert løsningsmetode krever. Dette muliggjøres ved at programvaren anvender generelle algoritmer som kan brukes til å løse mange ulike problemer. Ulempen er en begrenset mulighet for å tilpasse løsningsmetoden til egenskapene ved hvert enkelt problem. Konsekvensen er at anvendelse av kommersiell programvare reduserer arbeidet fram til en modell kan tas i bruk, men samtidig gir et potensielt tregere verktøy etter ferdigstillelse.

Ved å implementere en løsningsalgoritme selv kan man tilpasse denne spesielt til det aktuelle problemet, eksempelvis gjennom dekomponering. Generelt løses gjerne små problemer raskest uten dekomponering, mens større problemer får redusert løsningstiden når de dekomponeres [Cordeau et al., 2000]. Dette betyr at kommersielle pakker som ikke benytter dekomponering kan gi gode løsningstider på enkelte problemer. Likevel refererer Cordeau et al. (2000) til resultater hvor en dekomponeringsstrategi ga best løsnings tid for alle problemstørrelser for en modell. Deres modell hadde nære paralleller til vår, men vår modell mangler enkelte positive egenskaper med tanke på heltallighet sammenlignet med deres. Dette beskrives nærmere i delkapittel 5.3.1.

Vi har valgt å implementere modellen i to utgaver, en i en kommersiell pakke og en selvimplementert algoritme som baserer seg på en dekomponeringsteknikk. Dette gir oss fordelene av å kunne sammenligne løsnings tider. Videre kan vi sammenligne løsninger som en del av testingen for å verifisere at implementasjonene er rette.



5.2 Dekomponeringsmuligheter

Til dekomponering har de tre teknikkene nevnt i kapittel 4 blitt vurdert for å omforme problemformuleringen gitt i delkapittel 3.2.6 slik at det skal bli enklere å finne en løsning.

5.2.1 Lagrange-relaksering

Med Lagrange-relaksering kan hvilket som helst sett av restriksjoner relakseres, slik at man har en rekke problemformuleringer å velge blant. Ved vurderingen av ulike relakseringsmuligheter kom vi ikke fram til noe problem med en kjent struktur og tilpasset løsningsmetode. En mulig relaksering kan være å følge en strategi Cordeau et al. (2000) brukte ved å relaksere alle restriksjoner bortsett fra flytkonserveringsrestriksjonene og variabeldeklarasjonene. Dette gir følgende problem som har en enkel struktur, men samtidig et meget stort løsningsrom:



$$\begin{aligned}
 \underset{\alpha^l, \beta_a^r, \gamma_a^r, \rho^{pv}, \sigma^{kp}}{\text{Max}} \quad & \underset{y_a^r, x_a^k}{\text{Min}} \quad \left[\sum_{k \in K} \sum_{a \in A} c_a^k x_a^k + \sum_{l \in L} \alpha^l (1 - \sum_{a \in A_S} \sum_{r \in R^a} e_a^l y_a^r) \right. \\
 & + \sum_{a \in A} \sum_{r \in R^a} \beta_a^r (d_a^r y_a^r - \sum_{k \in K} m^k x_a^k) \\
 & + \sum_{a \in A} \sum_{r \in R^a} \gamma_a^r (u_a^r y_a^r - \sum_{k \in K} m^k x_a^k) \\
 & + \sum_{p \in P} \sum_{v \in V} \rho^{pv} (h^v - \sum_{k \in K} \sum_{a \in A^{pv}} m^k x_a^k) \\
 & \left. + \sum_{k \in K} \sigma^{kp} (w^k - \sum_{v \in V} \sum_{a \in A^{pv}} x_a^k) \right] \\
 \\
 \sum_{a \in I^n} x_a^k &= \sum_{a \in O^n} x_a^k \quad , \forall k \in K, n \in N^k \\
 \\
 y_a^r &\in \{0,1\} \quad , \forall a \in A_S, r \in R^a \\
 \\
 x_a^k &\geq 0, \text{ heltall} \quad , \forall k \in K, a \in A^k \\
 \\
 \alpha^l &\geq 0 \quad , \forall l \in L \\
 \\
 \beta_a^r &\geq 0 \quad , \forall a \in A_S, r \in R^a \\
 \\
 \gamma_a^r &\leq 0 \quad , \forall a \in A_S, r \in R^a \\
 \\
 \rho^{pv} &\leq 0 \quad , \forall p \in P, v \in V \\
 \\
 \sigma^{kp} &\leq 0 \quad , p = 1, \forall k \in K
 \end{aligned}$$

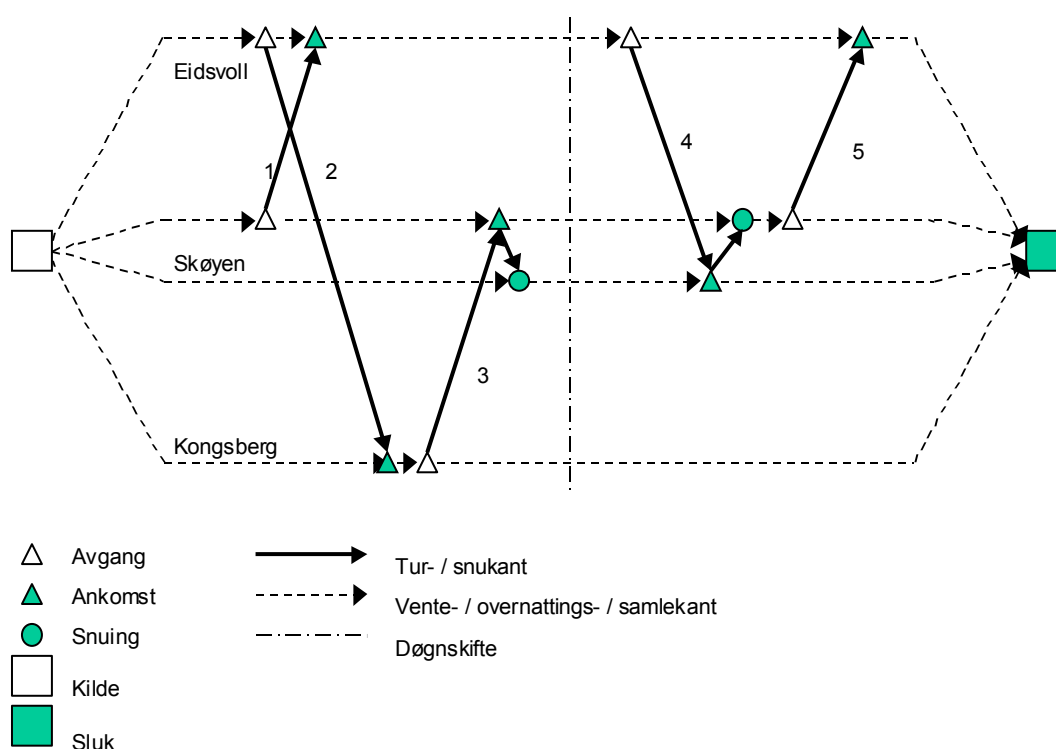
α^l , β_a^r , γ_a^r , ρ^{pv} og σ^{kp} er Lagrange-multiplikatorer.

5.2.2 Dantzig-Wolfe dekomponering

En naturlig dekomponering ved anvendelse av Dantzig-Wolfe på vår modell er å definere det totale problemet med unntak av flytkonserveringsrestriksjonene [likningene (20) i delkapittel 3.2.5] som masterproblem. Subproblemene blir da



asykliske korteste vei problemer, ett for hver materielltype. Dette forutsetter at tilbakeføringskantene erstattes med to ekstra noder, en kilde med tilbud lik 1 og kanter til første node for hver stasjon og et sluk med etterspørsel lik 1 og kanter fra siste node for hver stasjon som vist i Figur 5-1. Subproblemet vil på denne måten gi mulige stier fra kilde til sluk som representerer turneringer, mens masterproblemet bestemmer hvor mange individer som skal kjøre hver turnering.



Figur 5-1: Graf med kilde og sluk



Matematisk har masterproblemet følgende form:

$$\begin{aligned}
 & \underset{y_a^r, \theta_\varpi}{\text{Min}} \sum_{k \in K} \sum_{\varpi \in \Omega^k} c_\varpi \theta_\varpi \\
 & \sum_{a \in A_S} \sum_{r \in R} e_a^l y_a^r = 1 \quad , \forall l \in L \\
 & \sum_{k \in K^r} \sum_{\varpi \in \Omega^k} m^k t_\varpi^a \theta_\varpi \geq d_a^r y_a^r \quad , \forall a \in A_S, r \in R^a \\
 & \sum_{k \in K^r} \sum_{\varpi \in \Omega^k} m^k t_\varpi^a \theta_\varpi \leq u^r y_a^r \quad , \forall a \in A_S, r \in R^a \\
 & \sum_{k \in K} \sum_{\varpi \in \Omega^k} m^k g_\varpi^{pv} \theta_\varpi \leq h^v \quad , \forall p \in P, v \in V \\
 & \sum_{\varpi \in \Omega_k} \theta_\varpi \leq w^k \quad , \forall k \in K \\
 & y_a^r \in \{0,1\} \quad , \forall a \in A_S, r \in R^a \\
 & \theta_\varpi \geq 0, \text{ heltall} \quad , \forall k \in K, \varpi \in \Omega_k
 \end{aligned}$$

der $\varpi \in \Omega_k$ er turneringene gitt av subproblemene. θ_ϖ er en beslutningsvariabel som forteller hvor mange individer som skal kjøre turnering ϖ . t_ϖ^a er en binær parameter som er lik 1 dersom kant a inngår i turnering ϖ , og 0 ellers. g_ϖ^{pv} er en binær parameter som er lik 1 dersom turnering ϖ inneholder kanten over døgnskifte p for stasjon v . Øvrige definisjoner er som beskrevet i kapittel 3.

5.2.3 Benders dekomponering

Ved bruk av Benders dekomponering kan man skille de ulike klassene av variabler i adskilte problemer slik at masterproblemet bestemmer y -variablene mens subproblemene bestemmer x -variablene. Subproblemet blir et flervareflytproblem med tilleggsrestriksjonene (5) og (6) og har følgende form:



$$G_r(\bar{y}) = \underset{x_a^k, y_a^r}{\text{Min}} \quad \sum_{k \in K} \sum_{a \in A} c_a^k x_a^k \quad (1)$$

$$\sum_{k \in K^r} m^k x_a^k \geq d_a^r \bar{y}_a^r, \quad \forall a \in A_S, r \in R^a \quad (2)$$

$$\sum_{k \in K^r} m^k x_a^k \leq u^r \bar{y}_a^r, \quad \forall a \in A_S, r \in R^a \quad (3)$$

$$\sum_{a \in I^n} x_a^k = \sum_{a \in O^n} x_a^k, \quad \forall k \in K, n \in N^k \quad (4)$$

$$\sum_{k \in K} \sum_{a \in A^{pv}} m^k x_a^k \leq h^v, \quad \forall p \in P, v \in V \quad (5)$$

$$\sum_{v \in V} \sum_{a \in A^{pv}} x_a^k \leq w^k, \quad p = 1, \forall k \in K \quad (6)$$

$$x_a^k \geq 0, \text{ heltall}, \quad \forall k \in K, a \in A^k \quad (7)$$

Ser man bort fra lokomotiv og vogn, som skal fases ut av nærtrafikken, har man omkring Oslo to kompatible typer, type 69 og type 72. Disse omfatter henholdsvis to-vognsett av type 69 og tre-vognsett av type 69 og fire-vognsett av type 72. Dette betyr at de kompatible typene representerer disjunkte sett av materielltyper. Dermed er det kun restriksjonene (5) som forhindrer separabilitet mellom de kompatible typene r . Dersom denne restriksjonen relaxeres bort kan man få flere subproblemer, ett for hver kompatibel type. Med bakgrunn i antall materielltyper som inngår i hver kompatibel type gir denne dekomponeringen ett envareflytproblem og ett towareflytproblem, begge med tilleggsrestriksjonene (6).

Masterproblemet får følgende form gitt at man velger varianten med flere subproblemer og multikutt som beskrevet i delkapittel 4.3.1:



$$\text{Min}_{y_a^r, \theta^r} \sum_{r \in R} \theta^r \quad (8)$$

$$\sum_{a \in A_S} \sum_{r \in R^a} e_a^l y_a^r = 1, \forall l \in L \quad (9)$$

$$- \sum_{a \in A_S} (d_a^r \alpha_a^i + u^r \beta_a^i) y_a^r \geq \sum_{k \in K^r} w^k \gamma^{ik} + n \sigma_{-}^i + o \sigma_{+}^i, \forall i \in I \quad (10)$$

$$\theta^r - \sum_{a \in A_S} (d_a^r \alpha_a^j + u^r \beta_a^j) y_a^r \geq \sum_{k \in K^r} w^k \gamma^{jk} + n \sigma_{-}^j + o \sigma_{+}^j, \forall j \in J \quad (11)$$

$$y_a^r \in \{0,1\}, \forall a \in A_S, r \in R^a \quad (12)$$

der I er settet av alle lovlighetskutt og J er settet av alle optimalitetskutt. α_a , β_a og γ_a er dualvariablene til henholdsvis restriksjonene (2), (3) og (6). σ_{-} og σ_{+} er dualvariablene for heltallskuttene, ved henholdsvis nedre grense n og øvre grense o .

5.2.4 Valg av dekomponeringsstrategi

Modellen som er testet i Cordeau et al. (2000) er identisk med vår med unntak av at de ikke har noen begrensning i stasjonskapasitet (restriksjonene (5) i delkapittel 5.2.3) og at de har separabilitet i materielltype framfor kompatibel type. For lineærrelaksjonen til denne modellen refereres det til tester med alle de tre nevnte dekomponeringsstrategiene. De konkluderer med at Lagrange-relaksering og Dantzig-Wolfe-dekomponering er lite egnet for deres modell, da løsningsstiden er betydelig høyere enn for Benders dekomponering. Med bakgrunn i disse resultatene valgte vi Benders som dekomponeringsstrategi.

5.3 Heltallighet og valg av subproblemer med Bender

Vi har i stor grad basert oss på Cordeau et al. (2000) sine erfaringer ved valg av dekomponeringsstrategi. Dette til tross for at våre problemer ikke er helt identiske. Forskjellene ligger først og fremst i mulige subproblemer og derigjennom heltallsegenskapene til den dekomponerte modellen. De ulike subproblemene og argumentasjonen for vårt valg gir her.



5.3.1 Separabilitet og heltallighet

Modellen til Cordeau et al. (2000) er separabel i materielltype, noe som er en betydelig fordel ved bruk av Benders dekomponering. Bakgrunnen er at dette gir envareflyproblemer når man ser bort fra begrensningen i tilgjengelig materiellmengde. Envareflyproblemer har den gunstige egenskapen at de gir heltallige løsninger av lineærrelaksjonen forutsatt at høyresidene i restriksjonene er heltallig og at koeffisientene i hver rad i matrisa går opp i tilhørende høyreside. Dette gjelder ikke for flervareflyproblemer [Williams, 1999]. Denne egenskapen ved envareflyproblemer er ikke lenger til stede når materielltilgjengelighetsrestriksjonene inkluderes i modellen, men problemet inkludert heltallskrav er løsbart dersom lineærrelaksjonene er løsbart [Cordeau et al., 2000]. Ved testing erfarte Cordeau et al. små gap mellom målfunksjonsverdien til lineærrelaksjonene og heltallsproblemene og at optimalitetskuttene fra lineærrelaksjonene ga gode approksimasjoner av kuttene for heltallsproblemene. Dette førte til at kun få iterasjoner i branch and bound prosessen var nødvendig. Videre viste testing at heltallsgapet, og dermed antall iterasjoner i branch and bound-prosessen, ble redusert når materielltilgjengelighetsrestriksjonen ble gjort slakkere.

Som nevnt i delkapittel 5.2.3 kan vår modell dekomponeres til ett subproblem inkludert alle tre materielltyper og både materielltilgjengelighetsrestriksjonene og stasjonskapasitetsrestriksjonene. Alternativt kan begrensningen i stasjonskapasitet relaxeres bort og problemet dekomponeres i to subproblemer som begge har begrensninger i tilgjengelig materiellmengde, ett problem med en materielltype og ett med to materielltyper. Begge disse strategiene mangler de gode egenskapene som gir Cordeau et al. små heltallsgap, og vil derfor sannsynligvis kreve flere iterasjoner i branch and bound-prosessen.

En alternativ tolkning av kompatible typer gir ytterligere en mulig dekomponeringsmulighet. Da defineres en kompatibel type som et bestemt antall individer av hver type. Eksempelvis tolkes ett to-vognsett og to to-vognsett som to forskjellige kompatible typer. Med de tre materielltypene som brukes og en maksimal tog lengde på åtte vogner gir dette 11 kompatible typer. Med en slik definisjon kan man etter å ha låst y -variablene i masterproblemet bestemme etterspørselen for hver



enkelt materielltype for seg. Dermed får man separabilitet i materielltype hvis begrensningen i stasjonskapasitet fortsatt relaxeres bort. Dette gir mulighet til å dekomponere i tre envareflytproblemer med tilleggsrestriksjonene for materielltilgjengelighet. Denne strategien vil gi de positive egenskapene med tanke på heltallighet som Cordeau et al. (2000) refererer til. Ulempen er at antall binærvariabler i masterproblemet vil øke voldsomt.

5.3.2 Valg av subproblemer

Vi har valgt å relaxere bort restriksjonene for stasjonskapasitet (5), men for å redusere feilen som gjøres ved denne relaxsjonen settes grensen for maksimal tog lengde lik 0 for de overnattingskantene som tilhører stasjoner med 0 hensettingskapasitet. Dette valget baserer seg på en avveining mellom forventninger til løsningsstidene og nytten av løsningene modellen gir.

Relakseringen av stasjonskapasitetsrestriksjonene fører til at en løsning fra modellen ikke nødvendigvis er en gyldig løsning på problemet definert i delkapittel 3.2.6, men en nedre grense for målfunksjonsverdien. Etter vår mening bør det være relativt uproblematisk å i etterkant finne et gyldig sett av turneringer ut i fra en slik løsning. Begrensningen i stasjonskapasitet var ment modellert kun for natten, da det i praksis ytterst sjelden ser ut til å være en bindende restriksjon ellers. Dermed finner man en gyldig løsning ved å i ettertid legge til bruk av et passende antall påsett og tomtogturer vekk fra en overbelastet stasjon om kvelden og tilbake om morgenen. Dersom målet er å minimere materiellbehovet vil denne løsningen ha sammenfallende målfunksjonsverdi med den nedre grensen gitt av modellen. Grunnlaget for den påstanden er at tomtogkjøring på natten ikke er materielldrivende når hensikten kun er å forflytte det materiellet som modellen allerede har funnet behov for. Dersom målet er å minimere driftskostnader vil målfunksjonsverdien til den bearbejdede løsningen være en øvre grense. Hvor stort gapet mellom de to grensene blir, er avhengig av inndataene.

Uten en relaxering av restriksjonene (5) er det ingen mulighet til å dele opp i flere subproblemer. Dette gir ett subproblem med mange heltallsvariabler, noe som tradisjonelt gir høy løsningsstid. Videre vil disse restriksjonene kunne bidra til at



variablene som representerer Type 72 oftere får fraksjonsverdier. Dette har sammenheng med egenskapene til envareflytproblemer og flervareflytproblemer beskrevet i delkapittel 5.3.1. Selv om både Type 72 og Type 69 behandles i ett stort problem, gjelder disse egenskapene så lenge ingen restriksjoner forhindrer separabiliteten mellom dem. En restriksjon på hensettingskapasiteten for stasjonene vil fjerne denne separabiliteten, slik at Type 72 ikke lenger representerer et envareflytproblem. Dette vi igjen ha en uheldig effekt på heltallsegenskapene til modellen. I tillegg vil en restriksjon på hensettingskapasiteten akkurat som for begrensning i tilgjengelig materiell [delkapittel 5.3.1] fjerne garantien for heltallsløsning til et envareflytproblem isolert sett.

Valget mellom de to måtene å definere kompatible typer på er i praksis et valg mellom et stort masterproblem sammen med tre relativt enkle subproblemer, eller et mindre masterproblem sammen med to subproblemer, derav ett som er mer komplisert. Forskjellen i masterproblemene baserer seg på antall binærvariabler som er proporsjonalt med antall kompatible typer. Cordeau et al. testet alternativet med flest binærvariabler og erfarte blant annet at antall kompatible typer har stor innvirkning på løsnings tiden. Ut i fra dette valgte vi alternativet med to kompatible typer og det minste masterproblemet.



6 Implementasjon

Som nevnt i delkapittel 5.1 har vi valgt å implementere modellen i to forskjellige programvarepakker. Arbeidet med begge utgavene har foregått med en inkrementell implementasjonsstrategi slik at funksjonaliteten har blitt utvidet gradvis. Her vil de to løsningsmotorene og implementasjonsvalgt tilknyttet dem bli presentert.

6.1 *Modell i Xpress-Mosel*

Xpress-MP er en matematisk modellerings- og optimeringsprogrampakke som inkluderer verktøy for å formulere, løse og analysere operasjonsanalytiske problemer. To sentrale komponenter i Xpress-MP er Xpress-Mosel og Xpress-Optimizer [Dash Optimization, 2001a]. Modellen ble implementert i modelleringspråket Mosel versjon 1.0.0, og ytterligere funksjonalitet ble benyttet ved å inkludere biblioteker. Modellen ble også løst fra Xpress-Mosel ved å inkludere Xpress-Optimizer som en biblioteksmodul. Prosessflyten består av fire sekvensielle faser, lese inn data, generere problemet, løse problemet med branch and bound og til slutt skrive ut løsningen. Kildekoden finnes i vedlegg A.

6.2 *Benders dekomponering*

Benders dekomponering ble implementert ved bruk av Xpress-BCL Builder Component Library versjon 2.0.2 i kombinasjon med Java 1.1. Xpress-BCL gjør det mulig å formulere og endre et problem i egen applikasjon og å bruke Xpress-Optimizer for å løse problemet [Dash Optimization, 2001b]. Biblioteket finnes med full støtte for C, C++ og Java, og på grunn av kjennskap til Java fra tidligere ble dette språket valgt. Javakoden brukes til å implementere formuleringen av problemene, branch and bound prosessen og manipuleringen av problemene i tråd med kuttgenereringsprosessen. For hver iterasjon gjennom kuttgenereringsprosessen brukes Xpress-Optimizer til å løse problemet med simplex-algoritmen.

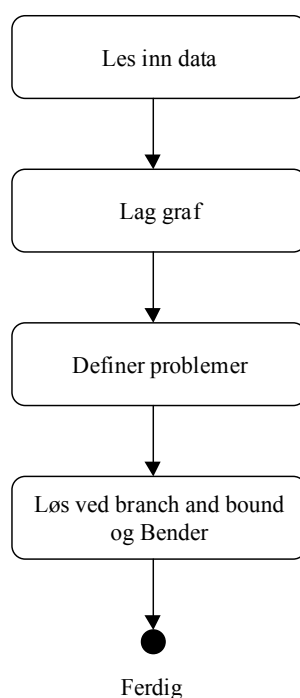
Det er implementert tre utgaver av løsningsalgoritmen. Hver av disse kombinerer branch and bound og Benders dekomponering på litt ulike måter. Her vil først ett av alternativene bli presentert fullstendig. Deretter beskrives det hvordan de to utgavene skiller seg fra den første. Denne oppdelingen er kun et presentasjonsvalg og ikke noen



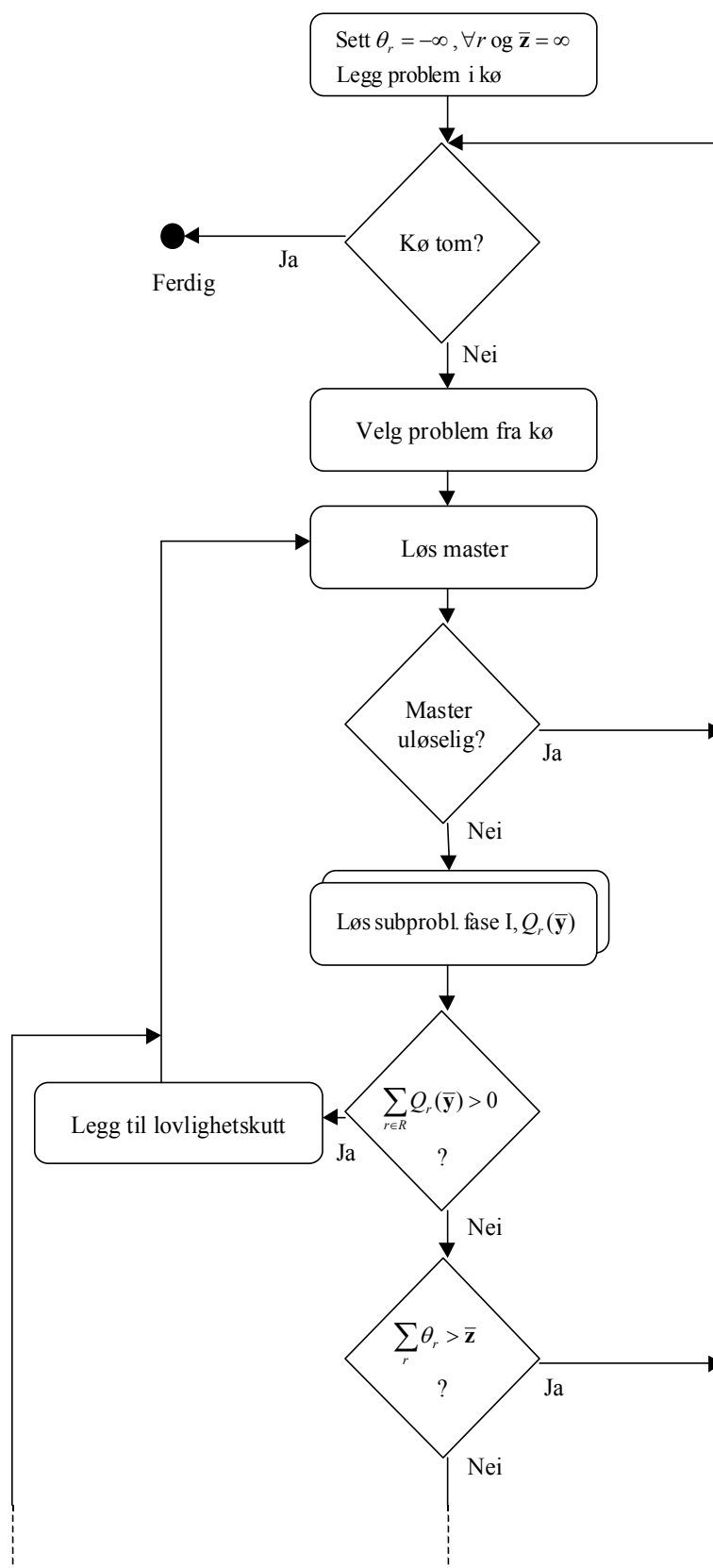
form for prioritering av de ulike utgavene. Kildekoden til det siste alternativet finnes i vedlegg B. Kildekoden for de andre utgavene er utelatt da disse inneholder tilnærmet det samme.

6.2.1 Løsningsalgoritmen

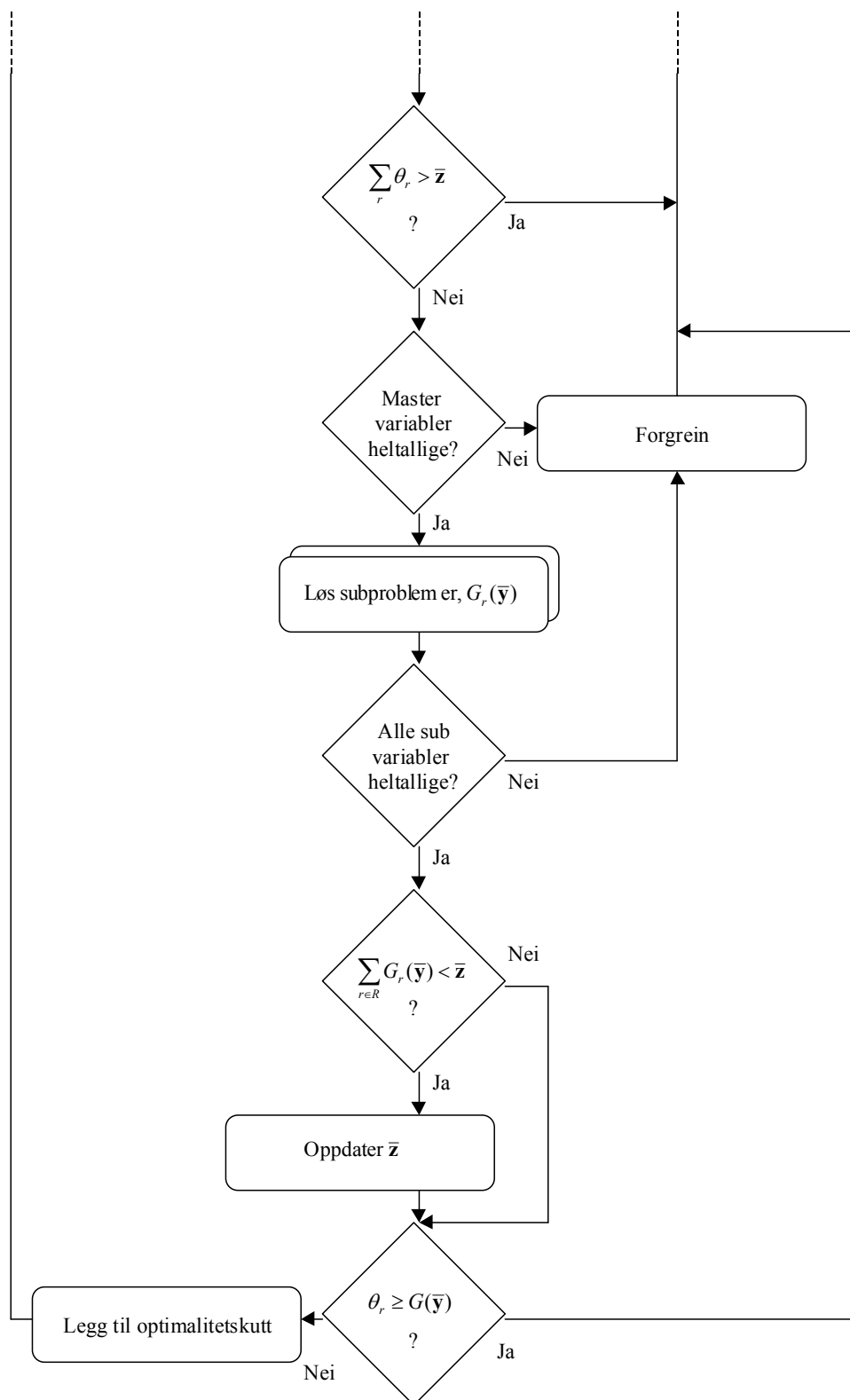
I Figur 6-1 vises en overordnet oversikt over prosessflyten i det implementerte programmet. Data leses inn fra filer og settes sammen til en graf av node- og kantobjekter. Disse objektene brukes som grunnlag for å generere et masterproblem og til sammen fire subproblemer, ett fase I og ett primalt subproblem for hver av de to kompatible typene. Problemene bevares gjennom hele løsningsprosessen og manipuleres i tråd med endringer i masterproblemvariabler og tilføring av nye kutt. I løsningsprosessen bygges det opp ett branch and bound-tre hvor både variabler fra masterproblemet og subproblemene kan påføres heltallskutt. Problemet i hver node løses ved Benders dekomponering. En mer detaljert oversikt over prosessflyten for løsningsprosessen er vist i Figur 6-2 og Figur 6-3.



Figur 6-1: Overordnet prosessflyt



Figur 6-2: Prosesstilt for løsningsalgoritmen, Del 1



Figur 6-3: Prosessflyt for løsningsalgoritmen, del 2.



6.2.2 Multikutt

Cordeau et al. (2000) refererer til en reduksjon i løsningstiden ved bruk av multikutt [delkapittel 4.3.1] framfor felles kutt for alle subproblemer. Derfor har også vi valgt å bruke multikutt.

6.2.3 Kutthåndtering

Et kutt generert i en node i branch and bound-treet er gyldig for alle barn av denne noden. Dette følger av Laporte et al. (1993) sitt bevis av at et kutt generert for en lineærrelaksjon også er gyldig for det tilsvarende heltallsproblemet [delkapittel 4.3.3]. Brukt i en barnenode er ikke et kutt nødvendigvis stramt [Cordeau et al., 2000], men det kan likevel sørge for en raskere konvergens i iterasjonene mellom masterproblem og subproblemer. Hvor mange og hvilke kutt som skal bringes med til en ny node i branch and bound-treet blir dermed en avveining mellom hvor mye informasjon som skal overføres mellom nodene, og hvor mye tid og lagringsressurs som skal brukes på å bevare kuttene [Dye, Tomasgard og Wallace, 2000]. I tråd med anbefalinger i Cordeau et al. (2000) har vi valgt i hver node å bevare alle kutt fra forfedrene til noden.

Ved å løse problemet i rotnoden til lineær optimalitet før første forgreining får man et sett av kutt som approksimerer lineærrelaksjonen til subproblemene. Disse kuttene kan bringes til barnenodene og dermed raskt tilføre masterproblemet informasjon om subproblemene. Denne teknikken, foreslått av McDaniel og Devine (1977), er testet ut i Cordeau et al. (2000) med betydelig reduserte løsningstider som resultat. Vi har implementert denne teknikken i vår løsningsmotor, og ved å sette et flagg kan vi velge om vi vil benytte den eller ikke. En annen fordel med denne utvidelsen er at den gir mulighet til å avbryte løsningsprosessen før optimum er nådd med et kjent gap til den potensielt beste løsningen. Dette kommer av at lineærrelaksjonen løst i rotnoden gir en nedre grense (NG) for målfunksjonen. Under branch and bound-prosessen vil den til en hver tid beste kjente løsningen danne en øvre grense ($ØG$). Dermed kan prosessen avbrytes når $ØG - NG < e$, der $e > 0$ er det gapet som er valgt som stoppkriterium. [Cordeau et al. 2000]



6.2.4 Kutttype

Vi har valgt å implementere et felles branch and bound-tre for variablene fra både masterproblemet og subproblemene. For hver node i dette treet legges det til kutt som baserer seg på dualfunksjoner som beskrevet i delkapittel 4.3.3. Dette betyr at kuttene inkluderer dualvariablene for heltallskuttene på *subproblemvariablene* i de nodene i treet som har slike heltallskutt. Som det ble beskrevet i forrige delkapittel bevares et kutt for alle etterkommernodene til noden hvor kuttet ble generert, men ikke for andre noder i treet.

6.2.5 Branch and bound

Branch and bound-strategien som er implementert bygger på en dybde-først strategi. Dette betyr at det alltid prioriteres å foreta en forgreiningsoverasjon på en ny ikke-heltallig variabel framfor å undersøke de aktive nodene, altså de nodene som ennå ikke er undersøkt. Ved valg av variabel til å forgreine på prioriteres variablene i masterproblemet framfor variablene i subproblemet. Dette gjøres fordi y -variablene i masterproblemet inngår i høyresiden til etterspørsels- og toglengderestriksjonene i subproblemet. Dermed er sjansen for å få heltallig løsning på subproblemene betydelig redusert hvis ikke y -variablene er heltallige. For subproblemene forgreines det kun på de variablene som representerer kantene i settet A_s og snukantene. Førstnevnte prioriteres først. Som et resultat av at disse variablene er heltallige vil øvrige variabler i subproblemene få heltallige løsninger. Dersom flere variabler med samme prioritet har en fraksjonsløsning velges den variabelen som har minst gap fra en heltallig løsning. Siden y -variablene er binærvariabler låses forgreiningvariabelen til henholdsvis 0 og 1 i hver sin node. I den noden hvor variabelen låses til 1 låses samtidig alle andre y -variabler som dekker samme tur med en annen kompatibel type eller sekvens til 0. Etter forgreining undersøkes først den noden hvor forgreiningvariabelen kan plassere seg nærmest den lineære løsningen. Beste kjente gyldige løsning funnet så langt bevares til enhver tid gjennom branch and bound prosessen og brukes som grunnlag for å forkaste noder som gir dårligere målfunksjonsverdi.



6.2.6 *Masterproblem til heltallighet – 1. alternativ*

I denne utgaven av løsningsalgoritmen skilles variablene for masterproblemet og subproblemene fra hverandre i adskilte branch and bound-trær. For hver node for subproblemvariablene løses masterproblemet til heltallighet i et nytt branch and bound-tre for hver iterasjon i kuttgenereringsprosessen. Fordelen med dette er at man sikrer seg heltallige høyresider i alle subproblemene, også for fase I, i motsetning til hva man har i den foregående utgaven. Heltallsalgoritmen i Xpress-Optimizer er brukt for å løse masterproblemene, mens subproblemene håndteres i et selvimplementert branch and bound-tre. Ut over denne forskjellen er denne utgaven identisk med den foregående.

6.2.7 *Branch and bound for subproblemer og Gomory-kutt – 2. alternativ*

I denne implementasjonen av Benders dekomponering forgreines det heller ikke på variabler fra subproblemene og masterproblemet i samme branch and bound-tre. Derimot brukes ett tre for variablene i masterproblemet, og hver node i dette treet kan inneholde null, ett eller flere trær hvor det forgreines på subproblemvariabler. Trærne for subproblemvariablene brukes for å lage Gomory-kutt som beskrevet i delkapittel 4.3.3. Når det legges til Gomory-kutt innføres det også ekstra binærvariabler i masterproblemet. Disse kan danne grunnlag for forgreining i samme tre som de andre masterproblemvariablene, og omtales her som Gomory-variabler. For å gjøre Gomory-kuttene som legges til strammest mulig, og dermed minimere antall kutt og Gomory-variabler, gjennomføres sjekken på heltallighet og eventuell forgreining på Gomory-variablene før fase I subproblemene løses og eventuelle nye kutt legges til. Optimalitet i en node i branch and bound-treet for masterproblemet er nådd når løsningen av masterproblemet ikke har blitt endret etter at siste optimalitetskutt ble lagt til. Optimal løsning for subproblemene er i dette tilfellet beste løsning funnet i branch and bound-treet som dannet grunnlag for det siste optimalitets-Gomory-kuttet, eller alternativt siste løsning for subproblemene dersom siste optimalitetskutt ikke er et Gomory-kutt.



7 Testing og resultater

Hovedhensikten med testingen er å bestemme om modellen vår egner seg til å støtte de oppgavene den er tiltenkt. Dette innebærer både å bestemme om løsningstiden er akseptabel for ulike modellstørrelser og om modellformuleringen vi har valgt beskriver virkeligheten godt nok. Testtilfellene og tilhørende resultater presenteres adskilt i disse to kategoriene. Mindre tester gjennomført i forbindelse med den inkrementelle implementasjonsstrategien er ikke gjengitt i denne rapporten. Det gis en beskrivelse av valgene gjort i forbindelse med testsettene.

Testene ble utført på en AMD Athlon XP 1700+ (1.46 GHz) med 2GB RAM, 1*40GB IDE HD og 1*100MBit Ethernet. Da implementasjonen av Benders dekomponering ikke førte frem finnes ingen resultater for denne algoritmen, men en drøfting gis i delkapittel 8.6.

7.1 Testing i forbindelse med den inkrementelle implementasjonen

Knyttet til den inkrementelle implementasjonen ble det utført en mengde tester på mindre og varierte datasett etter hvert som ny funksjonalitet ble lagt til modellen. Disse testene kan grovt sett deles inn i to grupper. Hensikten med den første av gruppene var å styre programflyten gjennom hele kildekoden slik at man kunne kvalitetssikre denne (white-box). Den andre gruppen av tester hadde som hensikt å verifisere at modellen oppførte seg som forventet når man endret bestemte parametre (black-box). Denne formen for testing førte til oppretting av feil underveis, samtidig som den ga oss en trygghet for at modellen ble riktig implementert og en forståelse for modellen som helhet.

7.2 Beskrivelse og valg av turer i datasettene

Modellen tar blant annet et sett av turer som inndata. Med turer menes i det følgende både ruteplanlagte turer og tomtogturer. Dagens ruteplanlagte turer har en spesiell struktur, noe som skyldes at planleggerne forsøker å følge prinsippene rundt integrerte tidstabeller med faste intervaller [Borgman, 2001] i så utstrakt grad som det lar seg gjøre. Videre har ruteplanen og materiellturneringene gjensidig tilpasset seg hverandre over tid. For å ta høyde for denne strukturen har vi i testsammenheng valgt



turer fra dagens ruteplan til å utgjøre de ulike testsettene. Samtlige turer knyttet til tre tilfeldige pendler for en uke uten helligdager er hentet fra NSB's database TrainPlan. Disse er Kongsberg-Eidsvoll, Skøyen-Kongsvinger og Skøyen-Eidsvoll.

I TrainPlan finnes imidlertid ikke alle tomtogturer. For turer som eksempelvis har avgang fra Skøyen med materiell som står hensatt på Filipstad, vil posisjonskjøringen mellom Filipstad og Skøyen ikke finnes i databasen. Slike tomtogturer har vi manuelt lagt til utvalget av turer fra databasen. Unionen av turer nevnt så langt er identisk med den delen av grunnplanen som tilsvarer de tre utvalgte pendlene.

Ut over dette har vi generert noen ekstra tomtogturer som NSB ikke benytter seg av. Dette skyldes tre ulike faktorer. I en del testtilfeller inkluderes stasjoner som i utgangspunktet ikke er endestasjoner for noen av turene som inngår i testen. Dette er gjort for å tilby hensettingskapasitet i nærheten av stasjoner som ikke selv har kapasitet for hensetting. Siden ingen ruteplanlagte turer går til disse stasjonene, er vi avhengige av å inkludere antatt fornuftige tomtogturer til stasjonene for at de i det hele tatt skal kunne benyttes av modellen. For det andre finnes det eksempler på at materiell fra en av de utvalgte pendlene er påsett på ruteplanlagte turer som tilhører en ikke-valgt pendel. I slike situasjoner vil de ruteplanlagte turene mangle i tursettet vårt. Tomtogturer som tilsvarer de opprinnelige ruteplanlagte turene er under slike omstendigheter lagt til. Den tredje faktoren gjelder for materiell som er påsett på turer som finnes i tursettet, men hvor påsettet ikke skal helt frem til turens endestasjon. Her er det lagt til tomtogturer parallelt med turen for å ta høyde for dette.

De ulike testsettene vi har valgt å benytte, gjengitt senere i dette kapitlet, består av alle eller et utvalg av turene i grunnplanen og disse egendefinerte tomtogturene. For hver kategori av testtilfeller vil det nærmere presiseres hvilke turer som inngår.

7.3 Beskrivelse av parametre og valg av verdier

Inndata til modellen er en rekke parametre. Med parametre menes konstanter som enten inngår i selve modellformuleringen eller som brukes i en pre-prosesseringsfase for å generere de riktige beslutningsvariablene (grafene). I testsammenheng har vi valgt å endre verdien på bestemte parametre fra testtilfelle til testtilfelle for å se hvordan variasjonene påvirker løsningen eller løsningstiden. Andre har fått samme verdi i alle



testsettene. Alle parametrene presenteres i det følgende. I de tilfellene hvor parameterverdien er den samme i alle testsettene gjengis den her. Parametre som varierer får gjengitt sin verdi i forbindelse med hvert enkelt testsett. Modellen krever følgende inndata:

7.3.1 Stasjoner

I modellen defineres en endestasjon som en stasjon som er ytterpunkt i geografien for turene som inngår i den aktuelle testen. Slike stasjoner har kun blitt modellert med én kjøreretning da materiellet likevel må snu her. Alle andre stasjoner er modellert med to kjøreretninger. Det minste settet av stasjoner som må inkluderes i en test er settet av avgangs- og ankomststasjonene for turene som inngår i testen, men som nevnt har også andre stasjoner blitt inkludert i enkelte tester. Hver kjøreretning på hver stasjon har et eget løpenummer. I tillegg må det oppgis hvilke av løpenummerne som er endestasjoner. Hver stasjon har også en hensettingskapasitet.

7.3.2 Materiell

For hver av materielltypene Type 69 to-vognsett, Type 69 tre-vognsett og Type 72 fire-vognsett har vi angitt en kostnadsfaktor som vektet bruken av de ulike typene. Disse er henholdsvis 9, 11 og 10. Dette er på ingen måte tall som gjenspeiler reelle kostnader forbundet med driften, men det ligger likevel en tanke bak valget av verdier. For å se effekten av å benytte alle tre typene i én og samme modell når modellen ikke gis en begrensning på materielltilgjengelighet, må Type 72 for det første ha en kostnadsfaktor lavere enn tre-vognsettene, da begge disse typene har identisk setekapasitet i tillegg til at tre-vognsettene har den gunstige egenskapen at de kan skjøtes sammen med to-vognsettene. For det andre må både kostnadsfaktoren for Type 72 og tre-vognsettene være høyere enn for to-vognsettene, siden to-vognsettene har lavere setekapasitet.

Det er gunstig å modellere hensettingskapasiteten på stasjonene i antall vogner fremfor antall sett, da det er skinnelengden avsatt til hensetting som angir kapasiteten. Derfor har Type 72 blitt modellert med tre vogner til tross for at hvert sett består av fire vogner. Dette lar seg gjøre fordi et fire-vognsett er like langt som et tre-vognsett og har samme setekapasitet. Vognene i to-vognsettene og tre-vognsettene er like



lange. I modellen har altså materielltypene 2, 3 og 3 vogner for henholdsvis to-, tre- og fire-vognsett.

Hver materielltype har videre en øvre grense på tilgjengelighet. Med tilgjengelighet menes hvor mange individer av hver materielltype modellen kan benytte i løsningen. I noen testsett er den øvre grensen satt lik 0 for enkelte materielltyper for å forhindre at disse typene blir brukt.

7.3.3 Turer

Turene representeres som kanter mellom noder i en tid-sted-graf. For å sette av tilstrekkelig tid til på- og avstigning, bremseprøve, snuing og skjøting eller splitting, er ankomstnodene og snunodene forskjøvet i tid fra turenes opprinnelige ankomsttidspunkt [delkapittel 3.2.2]. Gitt sammenhengen mellom operasjonene over og tidsforbruket som kreves ved utføring av de ulike operasjonene, er lengden av forskyvningene avhengig av tre faktorer. Disse er på hvilken stasjon noden befinner seg, når på døgnet noden befinner seg og om noden er en ankomstnode eller en snunode.

Oppgitte data fra NSB Drift over hvor lang tid hver av operasjonene tar, samt våre valg av tidsforbruk, er gjengitt i Tabell 7-1.

Tabell 7-1: Tidsdata fra NSB og våre valg av parameterverdier i minutter

Operasjoner	Tidsforbruk oppgitt av NSB	Våre valg av tidsforbruk
Av- og påstigning	1-2	2
Skjøte- og splitte	2	2
Bremseprøve	3-4	4
Snuing	2-4	2-3

Av- og påstigningstid må naturligvis tas hensyn til for alle ruteplanlagte turer. En bremseprøve må gjennomføres dersom minst én av operasjonene snuing, skjøting eller splitting blir utført. For å forhindre at materiell skjøtes eller splittes hvis det ikke er tid til å gjennomføre operasjonen, har derfor alle snunoder og ankomstnoder, uavhengig av hvilken stasjon nodene befinner seg på, blitt forskjøvet med på- og avstigningstid, tid til bremseprøve og tid til skjøting og splitting. Dette utgjør til



sammen 8 minutter for våre valg av tidsforbruk for disse tre operasjonene. Valgene våre er identisk med den øvre grensen for hvert tidsintervall, noe som skyldes et ønske om å være helt sikre på at løsningen vi får er gyldig.

Intervallet for snutid varierer mellom 2 og 4 minutter. Denne variasjonen kommer av tog lengden. For et enkelt sett bruker togføreren 2 minutter å gå fra den ene enden av toget til den andre. For to sett skjøtet sammen er denne tiden 3 minutter og for tre sett 4 minutter. Tre sett skjøtet sammen forekommer svært sjelden, mens to sett blir benyttet en del i rushtiden inn til Oslo om morgenen og ut av Oslo om ettermiddagen. For å ta høyde for disse rushtidsvariasjonene på en enkel måte, samtidig som vi ikke ønsker å påføre modellen for stramme betingelser, har vi latt snutiden variere mellom 2 og 3 minutter avhengig av tid på døgnet og ukedag. I testsettene blir 2 minutters snutid praktisert i helgene og på hverdager etter kl 1900. På hverdager før kl 1900 benyttes 3 minutter. Dette innebærer at ankomstnodene på endestasjonene og snunodene blir ytterligere forskjøvet med snutiden.

Verdien på tidene over er de samme i alle testtilfellene og disse er manuelt lagt inn i rutetabellen som er inndata til modellen. Dette er gjort for å forenkle implementasjonen. Ulempen er at verdiene ikke kan endres uten en gjennomgang av hele ruteplanen.

Som beskrevet i delkapittel 3.2.5 vil bruk av sekvenser gjeninnføre kjørbare løsninger som ble fjernet fra løsningsrommet ved å forflytte en mengde noder i tid. For endestasjoner har det for ankomstnodene blitt lagt til tid for alle de fire operasjonene, det vil si enten 10 eller 11 minutter. Siden materiellet er tvunget til å snu ved endestasjonene er det potensielt bare tiden til skjøting og splitting som kan redusere det nødvendige stasjonsoppholdet. Altså kan man maksimalt redusere stasjonsoppholdet ved endestasjoner med 2 minutter. For alle andre ankomstnoder kan man maksimalt redusere stasjonsoppholdet med 6 minutter ved å fjerne tillagt tid til skjøting, splitting og bremseprøve. For snunodene gjelder samme resonnement som for ankomstnodene på endestasjonene, og man kan maksimalt redusere stasjonsoppholdet for disse med 2 minutter.

Disse to parametrene med verdi 6 og 2 minutter, heretter kalt parametre for sekvensgenerering, henholdsvis S og L, inngår i datasettene og benyttes for å kunne



generere et gyldig sett av sekvenser. For å oppnå en spesiell effekt har vi i noen testtilfeller valgt å endre parametrene til 7 og 3 minutter. En nærmere beskrivelse av hva vi var ute etter og tolkningen av denne endringen beskrives i drøftingen i delkapittel 8.3.1.

Hver tur har en kostnad forbundet med seg. Vi har valgt å benytte kjøredistansen multiplisert med en faktor for å utgjøre denne kostnaden. Kostnadsfaktoren kan i prinsippet være forskjellig for hver tur, men vi har valgt å kun skille mellom ruteplanlagte turer og tomtogturer. For ruteplanlagte turer er kostnadsfaktoren 1 og for tomtogturer 3. Dette innebærer at det for en gitt strekning er tre ganger dyrere å kjøre en tomtogtur enn å kjøre en ruteplanlagt tur eller henge på som påsett. I delkapittel 8.3.2 diskuteres valget av disse verdiene.

7.3.4 Kompatible typer

Den maksimale tog lengden er i dag åtte vogner. I testsettene finnes det imidlertid én øvre tog lengde for hver av de to kompatible typene. Som vi har vært inne på har Type 72 blitt modellert med 3 vogner per sett. Siden aldri mer enn to sett har lov til å skjøtes sammen for Type 72 er det gunstig å la den øvre grensen gå opp i antall vogner, jamfør egenskapene til et énvareflytproblem [delkapittel 5.3.1]. Øvre tog lengde for Type 72 er dermed 6 vogner. For Type 69 er de to materielltypene kompatible, så den øvre grensen kan ikke justeres på samme vis og er derfor 8 vogner.

Etterspørselen for en tur kan variere mellom 2 og 8 vogner. På samme vis som for maksimal tillatt tog lengde, men nå for hver tur, finnes det én etterspørsel for hver kompatibel type. For Type 69 er etterspørselen identisk med den opprinnelige, men for Type 72 har etterspørselen blitt rundet av oppover til 3 eller 6 for på samme vis som over å utnytte énvareflytproblemets heltallsegenskaper.

Til slutt finnes det for hver tur parametre som angir hvilken kombinasjon av kompatible typer som har lov til å dekke turen. Verdien på disse parametrene kommer vi tilbake til for de ulike testtilfellene.



7.3.5 *Snu- og ventekostnader*

I noen testtilfeller har vi benyttet en kostnad forbundet med snuing og stasjonsopphold. Snuing og stasjonsopphold representeres som henholdsvis snukanter og ventekanter i grafen. Kostnaden for begge disse ble valgt til 1, og en nærmere forklaring gis i drøftingen i delkapittel 8.3.2.

7.4 **Kvalitative testtilfeller**

Denne kvalitative testkategorien har som hovedformål å vurdere om vårt valg av modellformulering beskriver virkeligheten godt nok.

7.4.1 *Valg av pendel og parametre*

For å kunne konkludere om modellen er virkelighetsnær eller ikke, ønsker vi å kunne sammenligne vår løsning med kjøremønsteret og materiellmengden NSB benytter i dag. Dermed må blant annet inndatasettet av turer være et utvalg fra en reell ruteplan. Det er videre ønskelig å ha et lite testsett, siden sammenligningene må gjøres manuelt. Med tanke på materiellmengden og kjøremønstrene er det enklest å velge en pendel hvor materiellet som brukes ikke benyttes på andre pendler. Pendelen vi har valgt er Skøyen-Kongsvinger med tidshorisont én dag, da denne oppfylder kriteriene over.

Stasjoner

Som det fremgår av Tabell 7-2 er seks stasjoner modellert og stasjonene Filipstad og Kongsvinger er endestasjoner for pendelen. Hensettingskapasiteten på Filipstad er ubegrenset, og derfor representert ved 100 i testsettene. Hensettingskapasiteten på de øvrige stasjonene er satt lik kapasiteten man opererer med i dag.



Tabell 7-2: Stasjoner brukt for de kvalitative testtilfellene

Stasjonsnr	Stasjonsnavn	Kapasitet
1	Filipstad	100
2	Skøyen Vest	0
3	Skøyen Øst	
4	Årnes Vest	0
5	Årnes Øst	
6	Kongsvinger	18

Turer

To ruteplanlagte turer fra Skøyen til Kongsvinger har i dagens materiellturnering påsett som settes igjen på Årnes. Som vi allerede har vært inne på, er ikke modellen vår i stand til å ha med påsett på kun deler av en strekning. To tomtogturer som går parallelt med de to ruteplanlagte turene er derfor innført som en tilnærming. Kostnadsfaktoren for disse er den samme som for de andre ruteplanlagte turene, og tolkningen av at disse benyttes er at materiellet er påsett på de parallelle ruteplanlagte turene. Etterspørselen for hver ruteplanlagt tur er satt til 3 vogner, i tråd med den materiellmengden NSB kjører i dag.

Materiell og compatible typer

I dag er fire Type 69 tre-vognsett benyttet for å dekke turene på pendelen. Materielltilgjengeligheten er satt til 100 for denne materielltypen for å indikere at tilgangen er ubegrenset. De to andre typene har 0 tilgjengelighet. Siden bare tre-vognsett skal benyttes, tillates kun Type 69 å dekke turene.

7.4.2 Testtilfeller

Det finnes fem testtilfeller i denne kategorien. Målfunksjonen og parametrene for sekvensgenerering varierer i de ulike settene. En oversikt over testene er gitt i Tabell 7-3.



Tabell 7-3: Oversikt over kvalitative testtilfeller

Tester	Antall turer	Parametre for sekvensgenerering		Parametre i målfunksjonen				Beskrivelse
		S	L	α	β	δ	γ	
MATR_6/2	50	6	2	0	1	0	0	Minimere materiell
MATR_7/3	50	7	3	0	1	0	0	Minimere materiell
DIST_6/2	50	6	2	1	0	0	0	Minimere distanse
DIST_7/3	50	7	3	1	0	0	0	Minimere distanse
DIST_SV_7/3	50	7	3	1	0	1	1	Minimere distanse inkl. snu- og ventekostnader

Som det fremgår av tabellen minimeres materiell når $\beta = 1$ og de andre leddene i målfunksjonen er 0. For å minimere distanse benyttes $\alpha = 1$ og de andre leddene 0. Dette er ikke helt riktig, siden β i dette tilfellet har en svært liten verdi større enn 0, noe som ikke fremgår av tabellen. Dette er gjort for å forhindre at materiell blir stående på stasjoner uten å bli brukt til å dekke noen turer. I de fire første testtilfellene varieres målfunksjonene mellom minimering av distanse og materiell i kombinasjon med variasjon i tidsparametrene for sekvensgenerering. Den siste testen minimerer distanse, men har i tillegg en kostnad på vente- og snukantene.

7.4.3 Resultater

I Tabell 7-4 vises materiellforbruket for testtilfellene og for sammenligningens skyld er også materiellmengden NSB benytter i dag vist.

Tabell 7-4: Materiellforbruk i antall Type 69 tre-vognsett

Tester	Materiellforbruk
NSB sin løsning	4
MATR_6/2	4
MATR_7/3	4
DIST_6/2	4
DIST_7/3	4
DIST_SV_7/3	4



Av de fem testene har vi valgt å presentere kjøremønsteret for MATR_7/3 og DIST_SV_7/3, samt NSB sin løsning [Figur 7-1 - Figur 7-6]. Dette utvalget er gjort da disse tre grafene til sammen illustrerer de momentene vi vil trekke frem i drøftingen i neste kapittel.

Siden løsningen fra modellen kun gir en allokering av materiellet til ulike kanter finnes det flere mulige valg av turneringer. I figurene representerer hver farge en turnering for ett materiellindivid. Grå kanter representerer ventekanter, snukanter, sekvenser eller tomtogturer som ikke brukes. Figurene representerer rettede grafer med flyt mot høyre, på tross av at vi har utelatt pilspissene.



Figur 7-1: Turnering for MATR_7/3, del 1



Figur 7-2: Turnering for MATR_7/3, del 2



Figur 7-3: Turnering for DIST_SV_7/3, del 1



Figur 7-4: Turnering for DIST_SV_7/3, del 2



Figur 7-5: NSB's turnering, del 1



Figur 7-6: NSB's turnering, del 2



7.5 Kvantitative testtilfeller

Hensikten med denne kategorien av tester er først og fremst å se hvordan løsnings tiden for modellen avhenger av størrelsen på datasettene. Videre er det ønskelig å se på hvordan ulike målfunksjoner og restriksjoner virker inn på løsnings tiden. Vi har ikke testet på alle mulige faktorer, men valgt å konsentrere oss om de aspektene som vi med bakgrunn i teorien forventer vil virke inn på løsnings tiden.

7.5.1 Dimensjoner for testing

I testene totalt sett er det fire dimensjoner som varierer. Disse er størrelse, kostnadsfunksjon, tillatte kompatible typer og hvor stramme enkelte restriksjoner er. I hver gruppe av tester har flere av disse dimensjonene blitt variert.

Størrelse

Størrelsen på datasettene er gitt av antall turer som inngår. Hvert sett inneholder alle turene for de tre valgte pendlene i et helt antall dager. Settene spenner fra én dag til en uke som vist i Tabell 7-5. De to siste dagene er lørdag og søndag som til sammen har omlag like mange turer som en hverdag.

Tabell 7-5: Datasettene og deres størrelser

Datasekk	Antall turer
1	176
2	352
3	528
4	704
5	880
6	964
7	1059

Restriksjoner

Med bakgrunn i argumentasjonen rundt heltallighet i delkapittel 5.3.1 har vi testet med to forskjellige sett av verdier for både materielltilgjengelighet og



stasjonskapasitet. Disse verdiene er for hver type restriksjon satt slik at det ene settet gir minst en bindende restriksjon, mens det andre settet gir slakke restriksjoner.

For materielltilgjengelighet har det slakke settet verdiene 100, 100 og 100 for henholdsvis Type 69 to-vognsett, Type 69 tre-vognsett og Type 72 fire-vognsett. Tilsvarende verdier for det stramme settet av verdier er 13, 10 og 6.

Stasjonskapasiteten er alltid satt til 0 for de stasjonene hvor kapasiteten er 0. For øvrige stasjoner er kapasiteten satt til 100 når restriksjonene ikke skal være bindende. Settet med stramme verdier er vist i Tabell 7-6.

Tabell 7-6: Stasjonskapasiteter for det stramme settet av restriksjoner

Stasjonsnr	Stasjonsnavn	Kapasitet
1	Kongsberg	12
2	Hokksund vest	0
3	Hokksund øst	
4	Gulskogen vest	0
5	Gulskogen øst	
6	Asker vest	0
7	Asker øst	
8	Skøyen vest	0
9	Skøyen øst	
10	Oslo S vest	0
11	Oslo S øst	
12	Lillestrøm vest	18
13	Lillestrøm øst	
14	Årnes vest	6
15	Årnes øst	
16	Dal vest	0
17	Dal øst	
18	Kongsvinger	18
19	Eidsvoll	18
20	Filipstad	100
21	Sundland	100



Disse settene av verdier gir grunnlag for fire kombinasjoner av restriksjonene som vist i Tabell 7-7.

Tabell 7-7: Kombinasjoner av stramme og slakke restriksjoner

Navn	Stasjonskapasitet	Materielltilgjengelighet
A	Slakk	Slakk
B	Stram	Slakk
C	Slakk	Stram
D	Stram	Slakk

Kostnadsfunksjon

Kostnadsfunksjonsparametrene δ og γ er begge satt lik 0 i alle tester, hvilket betyr at leddene med ventekostnader og snukostnader ikke har inngått i kostnadsfunksjonen. Verdiene for α og β varieres i de ulike gruppene av testtilfeller. I hver gruppe inngår tester hvor materiellminimering rendyrkes ved å sette β lik 1 og α lik 0. Dessuten brukes det motsatte ytterpunktet, minimering av distanse, med α lik 1 og β lik 0. Som nevnt i delkapittel 7.4.2 har β i praksis alltid en verdi større enn null, noe som også gjelder her. Disse to ytterpunktene omtales heretter som henholdsvis *matr* og *dist*.

To ulike serier av vektinger mellom *matr* og *dist* blir testet. I den ene serien er det tre forskjellige vektinger mellom α og β som vist i Tabell 7-8.

Tabell 7-8: Serie av varierende vekting mellom *dist* og *matr*

Navn	α	β
<i>Dist</i>	1	0
1	0.001	0.999
2	0.0001	0.9999
3	0.00001	0.99999
<i>Matr</i>	0	1

I den andre serien er verdiene på α og β satt slik at kostnadsforholdet mellom minimal materiellbruk og minimal kjøredistanse for de ulike størrelsene av datasett er tilnærmet konstant. Verdiene vises i Tabell 7-9.

Tabell 7-9: Serie av konstant vektning mellom *dist* og *matr* per datasett

Datasett	α	β
1	0.00002	0.99998
2	0.00001	0.99999
3	0.000007	0.999993
4	0.000005	0.999995
5	0.000004	0.999996
6	0.000004	0.999996
7	0.000003	0.999997

Tillatte kompatible typer

I en gruppe av tester har vi variert hvilke kompatible typer som tillates å dekke en tur. Vi har ikke differensiert mellom de ulike turene, men tillatt de samme kompatible typene for alle turer i hvert testtilfelle. De ulike variantene som er sammenlignet er dermed *kun Type 69*, *kun Type 72* eller *begge typer*. For de gruppene av tester hvor det ikke er oppgitt noen kompatibel type har standardverdien *begge typer* blitt benyttet.

7.5.2 Testtilfeller og resultater

Testtilfellene er inndelt i tre grupper av tester som varierer ulike utvalg av de fire dimensjonene nettopp beskrevet. Gruppene og de tilhørende testresultatene beskrives her. Enkelte tester inngår i flere grupper, og resultatene gjengis da i hver gruppe. Alle løsningsstidene er oppgitt i antall minutter og sekunder CPU-tid. Tester som har blitt avbrutt før de terminerte er i tabellene oppgitt i grå felter. Felter med ”-” er tester som ikke ble gjennomført på bakgrunn av høye løsningsstider for tilsvarende tester på mindre datasett. Testene med datasett 6 er merket med ”*” fordi dette settet ikke ga en lovlig løsning. Dette skyldtes mangel på tomtogturer slik at sykliske turneringer ikke kunne oppnås.

Størrelse

I denne testgruppen har alle de sju størrelsene og de fire variantene av materielltilgjengelighets- og stasjonskapasitetsrestriksjoner blitt kombinert. Videre



har disse blitt kombinert med kostnadsfunksjonene *dist*, *matr* og kombinasjonen med tilnærmet konstant vektning av disse. Resultatene fra testingen er vist i de fire tabellene under.

Tabell 7-10: Størrelse og restriksjonskombinasjon A

Datasett	<i>Dist</i>	kombinasjon	<i>matr</i>
1	0m22.520s	0m22.510s	0m49.140s
2	2m28.190s	2m31.480s	697m26.720s
3	7m45.670s	8m4.010s	3454m58.950s
4	17m42.420s	20m35.200s	-
5	33m45.390s	34m6.200s	-
6 *	39m2.850s	39m10.570s	-
7	2662m19.990s	53m6.190s	-

Tabell 7-11: Størrelse og restriksjonskombinasjon B

Datasett	<i>Dist</i>	kombinasjon	<i>matr</i>
1	0m27.870s	0m26.220s	1m27.620s
2	2459m15.300s	43m11.570s	97m18.940s
3	1608m16.730s	3463m47.810s	3215m26.220s
4	-	-	-
5	-	-	-
6 *	-	-	-
7	-	-	-

Tabell 7-12: Størrelse og restriksjonskombinasjon C

Datasett	<i>Dist</i>	kombinasjon	<i>matr</i>
1	0m51.040s	0m23.700s	7m48.410s
2	735m11.410s	2m35.570s	3463m59.230s
3	3183m26.440s	8m13.000s	-
4	-	29m29.570s	-
5	-	174m59.520s	-
6 *	-	39m8.470s	-
7	-	1783m19.100s	-



Tabell 7-13: Størrelse og restriksjonskombinasjon D

Datasett	<i>Dist</i>	kombinasjon	<i>matr</i>
1	0m22.920s	0m23.870s	3m12.630s
2	2m31.850s	2m43.580s	17m44.260s
3	8m6.190s	8m1.440s	3581m5.860s
4	18m1.520s	18m22.720s	-
5	40m0.720s	34m36.160s	-
6 *	39m1.650s	39m9.490s	-
7	3150m0.420s	2472m46.840s	-

Variasjon i kostnadsfunksjon

Denne gruppen er fokusert mot å se hvordan ulike vektninger mellom *dist* og *matr* virker inn på løsnings tiden og materiellbruken. Variasjonen i kostnadsfunksjonen er kombinert med de ulike kombinasjonene av stasjonskapasitet og materielltilgjengelighet. Alle testene er gjennomført med datasett 2. Resultatene fra testingen er gjengitt i de fire tabellene under.

Tabell 7-14: Varierende kostnadsfunksjon og restriksjonskombinasjon A

Kostnadsfunksjon	Løsningstid	Antall vogner
<i>Dist</i>	2m28.190s	61
1	5m34.800s	54
2	3m58.900s	52
3	2m31.480s	52
<i>Matr</i>	697m26.720s	52

Tabell 7-15: Varierende kostnadsfunksjon og restriksjonskombinasjon B

Kostnadsfunksjon	Løsningstid	Antall vogner
<i>Dist</i>	2459m15.300s	57
1	2929m32.270s	-
2	29m42.620s	52
3	43m11.570s	52
<i>Matr</i>	97m18.940s	52



Tabell 7-16: Varierende kostnadsfunksjon og restriksjonskombinasjon C

Kostnadsfunksjon	Løsningstid	Antall vogner
<i>Dist</i>	735m11.410s	59
1	4m26.000s	53
2	8m10.680s	53
3	2m35.570s	53
<i>Matr</i>	3463m59.230s	-

Tabell 7-17: Varierende kostnadsfunksjon og restriksjonskombinasjon D

Kostnadsfunksjon	Løsningstid	Antall vogner
<i>Dist</i>	2m31.850s	53
1	2m34.870s	53
2	2m37.110s	53
3	2m43.580s	53
<i>Matr</i>	17m44.260s	53

Variasjon i tillatt kompatibel type

I denne gruppen av testtilfeller varieres hvilke kompatible typer som tillates. Dette gjøres for alle de fire restriksjonskombinasjonene i tillegg til ytterpunktene *dist* og *matr* for kostnadsfunksjonen. Datasettene 2 og 3 brukes i disse testtilfellene. Resultatene fra testingen er presentert i de fire tabellene under.

Tabell 7-18: Variasjon i tillatte kompatible typer for datasett 2, *dist*

Restriksjoner	<i>begge typer</i>	<i>kun Type 69</i>	<i>kun Type 72</i>
A	2m28.190s	2m26.410s	2m25.270s
B	2459m15.300s	2m26.340s	2m25.330s
C	735m11.410s	2m27.290s	2m25.290s
D	2m31.850s	2m26.190s	2m25.420s

Tabell 7-19: Variasjon i tillatte kompatible typer for datasett 2, *matr*

Restriksjoner	<i>begge typer</i>	<i>kun Type 69</i>	<i>kun Type 72</i>
A	697m26.720s	4m35.500s	2m25.880s
B	97m18.940s	2m41.880s	2m25.780s
C	3463m59.230s	3m5.060s	2m25.500s
D	17m44.260s	4m55.840s	2m25.260s

Tabell 7-20: Variasjon i tillatte kompatible typer for datasett 3, *dist*

Restriksjoner	<i>begge typer</i>	<i>kun Type 69</i>	<i>kun Type 72</i>
A	17m42.420s	7m41.460s	7m38.360s
B	1608m16.730s	7m40.290s	7m38.770s
C	3183m26.440s	7m43.330s	7m38.260s
D	8m6.190s	7m42.610s	7m38.650s

Tabell 7-21: Variasjon i tillatte kompatible typer for datasett 3, *matr*

Restriksjoner	<i>begge typer</i>	<i>kun Type 69</i>	<i>kun Type 72</i>
A	3454m58.950s	28m21.670s	7m40.790s
B	3215m26.220s	12m11.860s	7m40.560s
C	2058m43.060s	1247m16.150s	7m40.070s
D	3581m5.860s	1173m48.010s	7m39.570s



8 Drøfting

Den overordnede målsetningen i diplomoppgaven har vært å utvikle en modell som kan bidra til å støtte materiellplanleggerne i en tidlig taktisk fase. Drøftingene i dette kapitlet skal lede frem til å gi svar på følgende hovedspørsmål:

- egner modellen seg til å gjennomføre en utsjekk av materiell?
- egner modellen seg som utgangspunkt for å utvikle en grunnplan?
- er løsningstiden akseptabel for påtenkt bruksmønster?

For å kunne evaluere om modellen er velegnet, vurderes modellens løsninger opp i mot de kvalitative kriteriene som stilles til materiellturneringene. Like viktig er det å sette modellen inn i en større sammenheng hvor modellens tilpasning til planprosessen tas i betraktning.

8.1 *Et integrert beslutningsstøtteverktøy*

Vår implementasjon av modellen har liten verdi som frittstående applikasjon. For å kunne fungere som en beslutningsstøtte bør modellen inkluderes i et helhetlig verktøy. Dette er nødvendig for å håndtere inndataene på en effektiv måte og for å kunne tolke og nyttegjøre seg av resultatene.

For at verktøyet skal bli effektivt i bruk er det en forutsetning at både tomtogturer og ruteplanlagte turer kan leses inn automatisk. Brukeren skal altså ikke måtte manipulere inndatasettene slik vi har gjort for å kunne gjennomføre testene våre. Et naturlig valg vil derfor være å sørge for kompatibilitet, og aller helst integrering, med TrainPlan slik at ruteplaner kan lastes direkte inn. Dette stiller store krav til ruteplanleggingsarbeidet. Verktøyet vil ha svært liten verdi dersom det ikke foreligger ruteplandata i et kompatibelt format. Dette innebærer at ruteplanene må være registrert elektronisk før verktøyet kan tas i bruk av materiellplanleggerne. Ruteplanene trenger på ingen måte å være endelige planer, men gjerne tidlige utkast og forslag til avgangs- og ankomst- tider og steder. Modellen tar ikke hensyn til om turene er klarerte med tanke på sportilgang, men dersom ruteplanen ikke er fullstendig må nødvendige tomtogturer legges til for å være sikker på at modellen skal gi løsning.



Foruten om effektiv innlesning av data, er grensesnittet mot brukeren av særlig betydning. God visualisering og intuitiv funksjonalitet er nødvendig blant annet for å kunne endre parametre på en enkel måte. De parametrene som sjelden endres bør settes opp med standardverdier slik at brukeren slipper å oppgi disse for hver test.

Visualisering av løsningen i form av tabeller og grafer er viktig for at brukeren tidlig skal kunne se konsekvensene av egne valg. Modellen genererer ikke fullstendige materiellturneringer, men allokterer materiell til ulike kanter i grafen. Det bør derfor utvikles en modul som kommer med minst ett fullstendig turneringsforslag på grunnlag av denne allokeringen. For at arbeidet skal bli rasjonelt bør beslutningsstøtteverktøyet også være integrert med øvrige verktøy som materiellplanleggerne benytter, slik som ResourcePlan.

Det er viktig å påpeke at et beslutningsstøttesystem ikke kan legge planer alene og på denne måten erstatte eller overflødiggjøre materiellplanleggerne. Det er et verktøy som legger til rette for og tar hensyn til de behovene planleggerne har, og det gir på denne måten en mulighet for å teste ut hypoteser og dermed bidra til å legge bedre planer. I beste fall vil en eventuell innføring av et beslutningsstøtteverktøy effektivisere planprosessen og dermed frigjøre tid til å arbeide med alternative strategier og utarbeide konsekvensanalyser. Kort oppsummert er det samspillet mellom menneske og verktøy som danner grunnlaget for suksess og gjør planleggeren til en bedre planlegger.

8.2 Materiellplanleggerens leveranser og bruk av modellen

Dagens planprosess preges av at en stor del av arbeidet gjøres manuelt. Modellen vi har utviklet vil ikke være i stand til å generere endelig planer, men hjelpe planleggerne et stykke på vei i arbeidet med å legge materiellturneringer.

8.2.1 Utsjekk av materiell

Før hovedbestilling må materiellplanleggerne undersøke om man har tilstrekkelig materiell til å gjennomføre ruteplanen. I dag gjøres dette ved å utarbeide en foreløpig grunnplan som ikke inkluderer tomtogturer [delkapittel 2.2.3]. Grunnplanen inngår



altså ikke i selve hovedbestillingen, men i denne fasen er den et middel for å kunne gjøre en utsjekk av materiellmengden ruteplanen krever.

Modellen kan benyttes til å gjøre en slik utsjekk ved å minimere materiellmengden som kreves for å oppfylle ruteplanen eller deler av ruteplanen. For å holde løsnings tiden lav, slik at brukeren har større mulighet til å bruke modellen på en interaktiv måte, kan man utnytte at planleggerne baserer seg på prinsippet om ”rene pendler” [delkapittel 2.3.3] og at tilbudet er redusert i helgene. For det første holder det å operere med en tidshorisont på én hverdag, siden materiellmengden brukt for å dekke disse turene også er tilstrekkelig for å dekke alle andre dager i uka. For det andre kan man dele opp turene i ruteplanen i like mange disjunkte sett som det er pendler og løse hver pendel for seg. Det totale materiellforbruket som må til for å dekke hele ruteplanen vil dermed være summen av materiellforbruket for hver modell som er løst.

Det er en nær sammenheng mellom antall turer modellen tar som inndata og tiden det tar å løse problemet [delkapittel 7.5.2]. Siden løsnings tiden er eksponentiell er det derfor gunstig å løse flere mindre modeller fremfor én stor. Det er også nødvendig å gjøre det på denne måten hvis man ønsker å holde materiellet adskilt på ulike pendler. Modellen vår kan i utgangspunktet ikke benyttes til å følge en *delvis* ”rene pendler” strategi på linje med dagens materiellturneringer [delkapittel 2.3.3]. Den allokere med andre ord materiell til turene i inndatasettet uten å ta hensyn til om turene tilhører én eller flere pendler. Resultatet dersom pendlene løses hver for seg blir fullstendig ”rene pendler”, mens hvis to eller flere pendler inngår som inndata vil det være umulig å på forhånd vite hvordan materiellet blir fordelt på pendlene. For å oppnå en tilnærming til *delvis* ”rene pendler” kan man eksempelvis tvinge ett eller flere av materiellindividene til å ha lengre pauser i løpet av modellens tidshorisont, slik at individene blir tilgjengelige for bruk på andre pendler i disse tidsrommene. Dette er imidlertid ikke en god tilnærming, da planleggeren på forhånd må definere både tidsrommets lengde og plassering på døgnet.

Som nevnt er leveransen til hovedbestillingen fra materiellplanleggerne en bekreftelse eller en avkreftelse på om man har tilstrekkelig materiell til å oppfylle ruteplanen.



Ved bruk av modellen er man ikke, i motsetning til dagens planprosess, avhengig av å utarbeide et utkast til en grunnplan for å kunne gjøre denne utsjekken.

8.2.2 Utarbeidelse av grunnplan

Etter hovedbestillingen og frem mot tilleggsbestillingen revideres og detaljeres grunnplanen gradvis. Tomtogturer og posisjonskjøring blir blant annet lagt til [delkapittel 2.2.3]. Dersom modellen blir benyttet som et ledd i utviklingen av en grunnplan, vil dagens prosess med større og større detaljeringsgrad fremdeles kunne benyttes. I tillegg gir modellen rom for å tilnærme seg en endelig plan på alternative måter.

Tomtogturer og sykliske turneringer

Arbeidet med å utvikle en grunnplan kan ved bruk av modellen starte så snart ruteplandata finnes i et kompatibelt format. Som vi allerede har vært inne på er modellen avhengig av at inndatasettet av turer er av en spesiell struktur. For det første må settet av turer være slik at sykler kan oppnås. Sykler er oppnådd når det på hver stasjon befinner seg like mange individer av hver materielltype i starten og slutten av planhorisonten. For det andre må settet av turer være slik at alle ruteplanlagte turer kan dekkes. Dette betyr for eksempel at en tidlig avgang fra en stasjon uten kapasitet for hensetting er avhengig av at det finnes en tomtogtur til stasjonen i forkant av avgangen.

Planleggerne skal ikke være avhengig av å manuelt legge til tomtogturer for å sikre at sykliske turneringer kan oppnås. Dette kan tas hensyn til ved å inkludere en modul i verktøyet som automatisk sørger for å legge til tomtogturer på starten og slutten av hver dag. En slik modul kan basere seg på en regelstyrt algoritme som utnytter informasjonen som ligger i ruteplanen. Det er viktig å presisere at en slik modul *ikke vil forbedre* løsningen, bare sørge for at modellen finner en gyldig løsning. Modulen vil generere flere tomtogturer enn strengt tatt nødvendig, og først når løsningen foreligger vil brukeren kunne se hvilke av disse tomtogturene som må inkluderes i planene for å oppnå en gyldig turnering.



Modellen som verktøy vil kunne åpne for et bredere spekter av alternative arbeidsformer for planleggerne. Modellen kan benyttes for å støtte dagens inkrementelle utvikling av en grunnplan ved å løse en modell som bare består av ruteplanlagte turer, for deretter å vurdere løsningen visuelt og gradvis legge flere detaljer til turneringene. Antatt fornuftige tomtogturer kan inkluderes i datasettet og modellens løsning vil vise effekten av disse. I slike tilfeller vil det være naturlig at planleggeren bruker modellen aktivt og gjennomfører flere iterasjoner.

Alternativt kan man nærme seg en endelig materiellturneringsplan raskere ved å legge til tomtogturer i en tidligere fase enn hva som gjøres i dag. Det er liten grunn til å tro at strukturen på ruteplanene vil endres oftere enn hva som har vært tilfelle de senere årene. Derfor vil ruteplanen ofte bare være en justering av tidligere års ruteplaner. Under slike omstendigheter kan planleggeren bruke tomtogturer som ble benyttet i foregående års plan som en indikasjon på hvor det kan være fornuftig å legge til tomtogturer. I de tilfellene man skal utarbeide en grunnplan som baserer seg på en helt ny ruteplan, vil ikke lenger planleggeren i like stor grad kunne utnytte erfaringer gjort i forbindelse med tidligere planer. Her vil det derfor være fordelaktig om verktøyet genererer et sett med antatt fornuftige tomtogturer som kan *forbedre* løsningen. Konsekvensen av å tilføre modellen mange slike autogenerated tomtogturer vil være at modellen får mange alternative løsninger å velge mellom. Dette øker sannsynligheten for å få en god løsning, men samtidig vokser problemstørrelsen og dermed også løsningstiden.

Tidshorisonter

Argumentene knyttet til ”rene pendler” ved utsjekk av materiell gjelder også hvis modellen skal benyttes som et ledd i å utarbeide en grunnplan. Et viktig aspekt som må tas hensyn til når man skal utvikle turneringer, og som legger føringer for lengden på tidshorisonten, er overgangen mellom de ulike dagene. Ved å benytte en ukes horisont er problemet beskrevet mest mulig presist, og modellen har flest valgmuligheter til å kunne utnytte materiellet på en best mulig måte. Ved å korte ned tidshorisonten til eksempelvis én dag, kan løsningen man får være påvirket av at man tvinger modellen til å oppnå sykler. Med andre ord har man ved å redusere modellens



tidshorisont i forhold til det opprinnelige problemet, også redusert utfallsrommet ved å påføre modellen restriksjoner som ikke finnes i virkeligheten.

Videre er overgangen mellom dager med ulik ruteplan et aspekt man mister ved å modellere kun én dag. Når ruteplanene for to dager ikke er identiske bør heller ikke plasseringen av materiellet om natten nødvendigvis være identisk. Dermed er det behov for en overgang mellom de ulike dagene, noe som kan kreve ulike tomtogturer for de ulike dagene for å oppnå kjørbare planer. Ved å benytte en horisont på mindre enn en uke kan man altså ikke nødvendigvis bruke dagsverkene som genereres til å legge turneringer for hele uken direkte.

8.2.3 Krav til løsnings tid og modellstørrelse

Kravet til løsnings tid avhenger sterkt av hvilket bruksmønster man ønsker seg for modellen. Med dagens prinsipper hvor man *etterstreber* å ha ”rene pendler”, er man avhengig av å benytte modellen på en interaktive måte for å eventuelt knytte pendlene sammen. Videre er det gunstig at planleggerne tidlig kan se konsekvensene av valgene sine, eksempelvis i forbindelse med tilføring av tomtogturer som kan forbedre løsningen. I en fase hvor man skal legge planer vil et minstekrav derfor være at man må kunne løse flere modeller i løpet av én og samme arbeidsdag.

For å bruke modellen til å utføre konsekvensvurderinger er ikke løsnings tiden like kritisk, men for å kunne ha et fleksibelt verktøy bør modellen takle vesentlig større modeller enn én pendel med en ukes horisont. Den største pendelen har i underkant av 100 avganger for hverdager og omtrent 550 avganger i løpet av en uke.

8.3 Vurdering av kvalitative testresultater

Med modellens kvalitative egenskaper menes hvor godt modellen beskriver virkeligheten og hvor godt den egner seg som et utgangspunkt for å utarbeide en grunnplan. Med bakgrunn i dette gjøres en vurdering av hvor godt den tar høyde for restriksjonene som må overholdes for at en turnering skal være gjennomførbar, samt sammenligne modellens løsninger med kjøremønsteret NSB benytter i dag. Vi evaluerer valgene vi har gjort både i forbindelse med modellformulering og datasett.



8.3.1 Tidsparmetre

I virkeligheten har man ikke absolutte verdier for tiden de ulike operasjonene ved et stasjonsopphold tar, da disse varierer avhengig av ulike faktorer. Ved testing valgte vi å la tidsparmetrene få en verdi tilsvarende den lengste tiden innenfor hvert av de angitte intervallene. Vi går i dette delkapitlet nærmere inn på konsekvensene av dette valget og vurderer konsekvensene ved alternative valg.

Forskyvning av ankomstnoder

For å ta hensyn til på- og avstigningstid, skjøte- og splittetid og tid til bremseprøve ble alle ankomstnoder forskjøvet frem i tid tilsvarende summen av disse tidene. Påstigningstiden for en tur er dermed innregnet i ankomstnoden til foregående tur. Dette viste seg å være et uheldig valg da denne fremgangsmåten også forutsetter på- og avstigningstid for tomtogturer, noe som er unødvendig siden disse turene ikke skal ha med passasjerer. En bedre strategi for å håndtere på- og avstigningstiden ville vært å dele den opp i to, slik at avgangsnoden ble forskjøvet tilbake i tid tilsvarende påstigningstiden og ankomstnoden frem i tid tilsvarende avstigningstiden. For tomtogturer kunne begge disse forskyvningene utelates.

Generering og bruk av sekvenser

Som en konsekvens av valgene av tidsparmeterverdier og måten nodene ble forskjøvet på, opplevde vi at enkelte kombinasjoner av turer ikke ble kompatible til tross for at turene kjøres av NSB i dag og dermed er kompatible i praksis. Konkret innebar dette at enkelte sekvenser ikke ble generert på grunn av en tidsdifferanse som var 1-2 minutter for stor. I testene MATR_6/2 og DIST_6/2 finnes det to slike eksempler. For de kvantitative testtilfellene [delkapittel 8.5.4] er differansen mellom turer som er kompatible i praksis og antallet turer modellen tolker som kompatible betydelig større.

I testene MATR_7/3, DIST_7/3 og DIST_SV_7/3 økte vi verdien på parametrene for sekvensgenerering fra 6 og 2 til 7 og 3. Den praktiske tolkningen av denne endringen kan være at på- og avstigningstiden reduseres med ett minutt, men for oss var hele hensikten å få generert sekvenser og dermed danne et bedre sammenligningsgrunnlag opp i mot NSB sine turneringer.



Konsekvensen av stramme ruteplaner

Med en stram ruteplan menes det at det er så små marginer mellom ankomster og avganger på samme stasjon at man ikke alltid vil ha tid til å gjennomføre snuing, skjøting eller splitting hvis de to turene skal kunne dekkes av samme materiellindivid. Sekvenser og snukanter ble modellert som et middel for å ta hensyn til stramme ruteplaner. Med våre valg av tidsparametre opplevde vi som referert flere tilfeller hvor sekvenser ikke ble generert mellom de turene som NSB anser som kompatible. Dette tilsier at man i dag opererer med svært stramme planer, og det er ingen grunn til å tro at planene vil bli gjort betydelig slakkere. Dermed vil en implementasjon av et beslutningsstøttesystem for materiellplanleggingen som ikke inkluderer snukanter og sekvenser være lite egnet for nærtrafikken.

For at modellen skal kunne beskrive virkeligheten på en bedre måte må det legges ned et arbeid for å forsøke å bestemme hvilke faktorer som påvirker tiden de ulike operasjonene tar. Eksempelvis vet man at snutiden avhenger av både lengden på togsettet og av infrastrukturen på snustedet. Noen av disse faktorene vil avhenge av løsningen som modellen gir (toglengden) mens andre er uavhengige (infrastrukturen). En konsekvens av at en del faktorer avhenger av løsningen er at man aldri kan oppnå en perfekt tilpasning til virkeligheten, men det vil være et steg i riktig retning å forsøke å differensiere tidsparametrene i langt større grad enn hva vi har gjort. Eksempelvis vil man kunne operere med ulike snutider på de forskjellige stasjonene for å ta høyde for infrastrukturforskjellene.

Gruppen av faktorer som avhenger av løsningen vil ikke kunne gi grunnlag for å differensiere tidsparameterne i samme grad. Her vil det være opp til planleggerne som benytter verktøyet å gjøre et valg. Dersom man velger å benytte høye verdier på tidsparametrene slik vi har gjort, vil man være sikker på at løsningen man får er gyldig i en praktisk situasjon. Hvis man derimot velger å redusere parametrene vil man gi modellen flere valgmuligheter, men ikke alle disse vil være en lovlig løsning. Dersom en slik strategi velges, må løsningen undersøkes manuelt i ettertid for å verifisere at løsningen man har fått er gjennomførbar.



8.3.2 Kostnader

Ulike parameterverdier i målfunksjonen vil virke inn på materiellets kjøremønster i løsningen. Det drøftes her hvilke kostnader og kostnadsforhold som er gunstige og hvordan man kan tilrettelegge for preferanser ved å vekte kostnadsverdiene innbyrdes.

Vente- og snukanter

Testene MATR_6/2, DIST_6/2, MATR_7/3 og DIST_7/3 får alle en løsning hvor et motorvognsett forlater depotet 5 timer før det kjører en ruteplanlagt tur [Figur 7-1 og Figur 7-2]. For å unngå slike løsninger kan man legge til en kostnad for å oppholde seg på de stasjonene hvor det ikke er ønskelig å ha materiell stående. Dette kan gjøres ved å legge til en kostnad på ventekantene i grafen. I ett av disse testtilfellene fikk vi også en løsning hvor materiell etter en ankomst snudde frem og tilbake før det forlot stasjonen. Derfor bør det også legges på en kostnad på snukantene for å unngå unødvendig mye snuing fram og tilbake.

Hensikten med å bruke ventekostnader på enkelte stasjoner er å unngå at det kjøres tomtog eller påsett til stasjonen tidligere enn hva turneringen har nytte av. Det er ikke meningen å påvirke andre deler av løsningen. Derfor bør kostnadene for stasjonsopphold være betydelig lavere enn kostnaden for å kjøre turer. Tilsvarende bør også kostnadene på snukanter være lave. Størrelsesforholdet mellom kostnaden for snuing og venting er avgjørende for hvorvidt man får snuing fram og tilbake eller ikke. Uavhengig av om kostnadene på ventekantene avhenger av lengden på tidsrommet de dekker eller ikke, vil lengden på stasjonsoppholdet påvirke hvor stor snukostnaden må være for å sikre at unødvendig snuing ikke foretas. Lengden på stasjonsoppholdet bestemmes av løsningen. Dette betyr at snukostnaden vil avhenge av løsningen, noe som naturligvis ikke er mulig å modellere. Dermed kan man ikke sikre seg mot slike snuoperasjoner uten å ha snukostnader som overstiger ventekostnaden gjennom hele planleggingshorisonten. Dette er ingen god løsning siden man vil ha lave kostnader som ikke påvirker øvrige deler av løsningen. Da løsningen fra modellen i alle fall må bearbeides før den settes i drift er det uproblematisk å sørge for å fjerne eventuelle gjentatte snuoperasjoner fra løsningen i ettertid.



Ut i fra dette resonnementet valgte vi i test DIST_SV_7/3 å bruke snu- og ventekostnader som var små, nærmere bestemt én per kant. Resultatet var som antatt at motorvognsettet forlot depotet senere enn for de andre løsningene, og dermed unngikk det uønskede stasjonsoppholdet. Det ble heller ikke foretatt noen unødvendige snuoperasjoner [Figur 7-3 og Figur 7-4].

Tomtogturer

Hvis målfunksjonen er av en slik form at kun materiell minimeres, vil man ikke kunne fortelle modellen om man foretrekker å kjøre tomtogturer eller benytte påsett. I løsningen av test MATR_7/3 [Figur 7-1 og Figur 7-2] ble de to sekvensene, som inneholdt en tomtogtur hver, benyttet uten at det fikk noen innvirkning på målfunksjonsverdien sammenliknet med MATR_6/2. For alle andre målfunksjoner kan man derimot gi en slik indikasjon på hva man foretrekker ved å variere kostnadsfaktoren for tomtogturer. Påsett krever skjøting og splitting og dette er en mer ressurskrevende operasjon i form av personell enn å kjøre tomtog. Tomtogkjøring på sin side krever at det er ledig sportilgang på strekningen i det aktuelle tidsrommet. Ved å sette kostnadsfaktoren for tomtogturer større enn kostnadsfaktoren for ruteplanlagte turer vil modellen foretrekke å benytte påsett. Dette ser vi eksempler på i DIST_7/3 og DIST_SV_7/3 hvor de to sekvensene ikke ble benyttet fordi de inneholdt en ”dyr” tomtogtur [Figur 7-3 og Figur 7-4]. Dersom kostnadsfaktoren settes lik for alle typer turer, vil modellen tilfeldig velge om det benyttes påsett eller kjøres tomtogturer.

8.3.3 Kvalitative modellbegrensninger

Modellen vi har beskrevet har svakheter ved at den ikke tar hensyn til alle faktorene som finnes i det virkelige domenet. Noen av disse faktorene er grunnleggende egenskaper som bør inngå i de tidlige planene, mens andre faktorer abstraherer også planleggerne bort i denne fasen.

Påsett på deler av en tur

Modellen vår kjenner kun avgangs- og ankomststasjonen for en tur, ikke de mellomliggende stasjonene. Dette fører til at det ikke lar seg gjøre å kjøre påsett på



deler av en tur. Det er ikke akseptabelt å dele turen opp i flere delstrekninger for å ta høyde for dette, fordi man risikerer at turen ikke blir dekket av det samme materiellindividet, noe som ville ha ført til at passasjerene måtte bytte tog opptil flere ganger underveis. I stedet kan brukeren vurdere om en tomtogtur som har blitt benyttet på tilsvarende strekning kan erstattes av et påsett. Dette innebærer at det kan være et behov for å generere én eller flere tomtogturer tett opptil tidene for en ruteplanlagt tur, dersom denne turen passerer minst en stasjon som fungerer som avgangsstasjon.

Hensetting på dagtid

NSB's turnering for en hverdag på pendelen Skøyen-Kongsvinger har to relativt lange perioder på dagen hvor materiellet tomtogkjøres til vedlikeholdsbasen Filipstad for hensetting og deretter tomtogkjøres tilbake i trafikk. Dette kan være hensiktsmessig når alternativet ville vært å oppholde seg på en stasjon med begrenset kapasitet over en lengre periode. Tilsvarende løsninger vil modellen vår generelt ikke gi, forutsatt at ventekostnadene holdes lave i forhold til kostnadene for å kjøre en tomtogtur. Dersom man kun minimerer materiell, og dermed verken har tur-, snu- eller ventekostnader, kan denne egenskapen ved en løsning inntreffe, men da kun som et tilfeldig valg blant alternative jevn gode løsninger.

For å kunne tvinge fram en slik egenskap i modellen vår måtte tomtogkjøring være billig sammenlignet med stasjonsopphold, noe som strider med den grunnleggende idéen med kostnader som knyttes til kjøredistanse. Dessuten vil slike muligheter ofte forutsette tomtogturer som man på forhånd må ha lagt til i et passende tidsrom, noe som kan vise seg vanskelig. Under videre arbeidet med løsningen for å utvikle turneringene vil det imidlertid være relativt enkelt å identifisere de stedene i grafen hvor det kan være aktuelt å kjøre materiellet bort for midlertidig hensetting ved å se etter lange sammenhengende pauser på stasjoner med liten hensettingskapasitet.

I vår modell legges det ingen føringer for å forsøke å samle driftspausene til materiellet i få lange pauser framfor mange korte. Dette førte til at vi i test MATR_7/3 fikk flere pauser på én til to timer fordelt på ulike materiellindivider [Figur 7-1 og Figur 7-2], mens eksempelvis test DIST_7/3_SV gav en løsning hvor en stor andel av



pausene kunne allokeres sammenhengende til samme materiellindivid [Figur 7-3 og Figur 7-4]. Slik oppsamling av pausene legger bedre til rette for tilpasning til driftspausebasert vedlikehold i ettertid, siden dette vedlikeholdet forutsetter minimum fire timers opphold på vedlikeholdsbasen. Videre vil det trolig være gunstigere for personellplanleggingen at man har lange pauser så personellet kan frigis for andre oppgaver. Dette er et aspekt som heller ikke planleggerne tar hensyn til i de taktiske planene.

8.4 Vurdering av kvantitative testresultater

De kvantitative testene har hatt fokus på løsnings tid og hvordan denne avhenger av kostnadsfunksjonen og enkelte restriksjoner. Her vurderes resultatene fra disse testene.

8.4.1 Kritikk av datasettene

Ved utviklingen av datasettene var hovedfokuset å kunne se hvordan løsnings tiden utviklet seg med tanke på størrelse. Derfor bygger alle datasettene på ruteplanen for de samme pendlene. Dette førte til at de fem første settene har samme struktur gjentatt i et økende antall eksemplarer. De to siste settene har en noe annen struktur siden disse inneholder helgedagene.

Ettersom vi i mange tilfeller kun har resultater for de minste settene, inntil tre dager, betyr dette at vi i liten grad vet om tendensene vi ser i resultatene skyldes datasettene, mønsteret i ruteplanene for nærtrafikken eller er egenskaper med modellen relativt uavhengig av datasett.

For å kunne komme med generelle utsagn rundt egenskapene til modellen bør den altså testes på et bredere utvalg av testsett. Siden vi baserer oss på reelle datasett mener vi likevel at trendene i våre resultater er nyttige indikatorer ved vurderinger tilknyttet den tiltenkte bruken.

8.4.2 Kostnadsfunksjon

Testresultatene viser at løsnings tiden er sterkt avhengig av valg av kostnadsfunksjon. Med unntak av et datasett for restriksjonskombinasjon B, tar minimering av



materiellbruk betydelig lengre tid enn minimering av kjøredistanse. Den kombinerte kostnadsfunksjonen gir alltid betydelig lavere løsnings tid enn minimering av materiell. En mulig årsak til dette er at minimering av materiell har langt flere jevngode løsninger enn de andre kostnadsfunksjonene siden kjøremønsteret ikke virker inn. Dermed kan en mindre del av branch and bound-treet kuttes vekk på grunnlag av høye målfunksjonsverdier for lineærrelaksjonen slik at mer arbeid må gjøres i branch and bound-prosessen.

På grunnlag av denne trenden vil det trolig være fornuftig å bruke en kombinasjon av å minimere materiellbruk og å minimere distanse også når minimal materiellbruk er målet, eksempelvis ved utsjekk før hovedbestilling. Ved denne kostnadsfunksjonen må man imidlertid sørge for at vektingen mellom distansedrevne kostnader og materiellbruk virkelig gir minimal materiellbruk. Man må altså sørge for at vektingen av kostnadene per materiellindivid er tilstrekkelig mye større enn distansekostnadene slik at løsningen ikke blir en avveining mellom kjøremønster og materiellbruk, men en minimering av materiellbruk først og av distanse dernest. Denne vektingen må bestemmes før man kjenner kostnadene for hvert enkelt ledd. Med bakgrunn i erfaring fra tidligere materiellplaner mener vi likevel at dette bør være relativt uproblematisk. Det bør bemerkes at løsnings tidene også avhenger av hvordan denne vektingen gjøres (Tabell 7-14 - Tabell 7-17), men våre tester er ikke tilstrekkelig til å avdekke noen klare mønstre i denne variasjonen.

8.4.3 Stasjonskapasitet og materielltilgjengelighet

Resultatene fra minimering av distanse viser store forskjeller i løsnings tid avhengig av hvilke restriksjoner som er stramme. Når hverken materielltilgjengelighets- eller stasjonskapasitetsrestriksjonene er stramme, det vil si kombinasjon A, er løsnings tidene gjennomgående forholdsvis lave, men når en av restriksjonstypene er stramme øker løsnings tiden betraktelig [Tabell 7-10 - Tabell 7-12]. Dette kan ha sammenheng med at disse restriksjonene forverrer heltallsegenskapene som beskrevet i delkapittel 5.3.1 noe som stemmer godt med at antall noder i branch and bound-prosessen øker betydelig. For kombinasjon D er utviklingen i løsnings tid tilnærmet like gode som for A på tross av at både materielltilgjengelighets- og stasjonskapasitetsrestriksjonene er stramme [Tabell 7-13]. En mulig forklaring på



dette er at disse restriksjonene danner kutt som skjærer hverandre i heltallspunkter slik at flere hjørnepunktene i løsningsrommet blir heltallige.

Ved en eventuell framtidig praktisk bruk av modellen vil restriksjonskombinasjon D i de fleste tilfeller være den mest presise beskrivelsen, siden materielltilgjengelighet og stasjonskapasitet er virkelige restriksjoner. Testresultatene tyder på at det vil være gunstig å benytte nettopp denne kombinasjonen da løsningstiden er lav. Dette betyr at man bør unngå å utelate tilgjengelig informasjon ved å slakke på restriksjonene selv i tilfeller hvor disse antas å ikke ha innvirkning på løsningen.

8.4.4 Kompatible typer

Tabell 7-18 - Tabell 7-21 viser at løsningstiden er svært stabil når kun Type 72 tillates. Dette stemmer godt med teorien siden Type 72 representerer énvareflyt. Testene hvor kun Type 69 er tillatt viser konsekvent høyere og langt mer varierende løsningstider enn for Type 72, noe som også er naturlig siden tovarerflytproblemer har dårligere heltallsegenskaper. For bruk av begge de kompatible typene er løsningstidene høyere enn når de holdes adskilt. En sentral del av forklaringen på dette er at løsningsrommet for binærvariablene reduseres betraktelig når kun én kompatibel type er tillatt siden de fleste binærvariabelverdiene da er gitt direkte av inndataene. Denne egenskapen ved modellen kan utnyttes til å holde løsningstiden nede når man før optimering har bestemt hvilken kompatibel type som skal dekke en tur.

8.4.5 Bruksmønstre og løsningstider

For å utføre en utsjekk av materiellbehovet kan man som nevnt nøye seg med å modellere én dag, slik at modellstørrelsen blir omlag 100 turer for en stor pendel alene. Dette gir en mindre modell enn datasett 1 som generelt har gode løsningstider [Tabell 7-10 - Tabell 7-13]. Dersom man fraviker prinsippet om "rene pendler" og inkluderer alle pendler i samme modell får man omlag 450 turer, noe som ligger mellom datasett 2 og 3 i størrelse. Dersom den kombinerte kostnadsfunksjonen benyttes gir dette også gode løsningstider i de fleste tilfellene.



Ved utvikling av turneringer bør planleggingshorisonten være en uke. Dette gir større datasett med inntil 550 turer dersom man modellerer pendlene adskilt. Datasett 3 med 528 turer har løsningsstider på omlag åtte minutter for den kombinerte kostnadsfunksjonen og restriksjonskombinasjon A, C og D. Tilsvarende gjelder for minimering av distanse og restriksjonskombinasjon A og D. For de øvrige restriksjonskombinasjonene er løsningsstiden som nevnt betydelig større.

8.4.6 Datamaskinressurser og optimalisering av kode

Under utviklingen av modellen i Xpress-Mosel opplevde vi at det krevdes svært mye minne- og harddiskkapasitet for å løse problemet. En sentral årsak til dette var at enkelte matriser med variable var svært glisne. Etter anbefaling ble disse omdefinert, noe som reduserte både minnebruk og løsningsstid betydelig. Dog kreves det fortsatt mye minne- og harddiskplass ved løsning av de største problemene. Det kan fortsatt være rom for ytterligere optimalisering av koden for å redusere minnebruken og tiden det tar å generere problemet. Eksempelvis kan man representere problemet som beskrevet ved bruk av Dantzig-Wolfe [delkapittel 5.2.2] der variablene representerer hele turneringer fra kilde til sluk [Nygreen, 2002].

8.5 Deterministisk modell i en stokastisk verden

Modellen vi har utviklet baserer seg på en rekke parametre som leses inn. Verdien til parametrene defineres og håndteres av modellen som ufravikelige fastverdier, til tross for at mange av dem er høyst usikre. Eksempelvis kan snutider og tid til skjøting og splitting variere avhengig av situasjon og hvem det er som utfører operasjonene. Etterspørsel er en annen usikker faktor siden man ikke kjenner tilstrømningen av passasjerer før driftstidspunktet. I begge tilfellene må man ty til erfaringsdata for å anslå parametrene. Dette betyr at verdiene som leses inn baserer seg på antakelser og erfaringer hos planleggerne, uten noen garanti for at valget er riktig ved gjennomføringstidspunktet.

Teoretisk sett er det mulig å lage en stokastisk modell hvor man tar hensyn til usikkerheten i parametrene. Dette kan gjøres ved å knytte stokastiske variabler til de usikre parametrene og definere tilhørende utfallsrom og sannsynlighetsfordelinger over disse variablene. Ulempen med en slik stokastisk modell er at man ofte får



mange stokastiske variabler med store utfallsrom som fører til at problemet blir svært vanskelig og tar lang tid å løse [Birge et al., 1997].

Når vi har valgt å arbeide med en deterministisk modell er det viktig å være klar over hvilke svakheter dette gir. Siden verdiene i modellen ansees som sikre gir modellen ingen insentiver til å velge fleksible og robuste løsninger. I dette ligger det at en løsning som lett lar seg tilpasse ved avvik fra de oppgitte verdiene, ikke vil bli valgt dersom løsningen ikke samtidig er den billigste for disse verdiene. Med andre ord vil det aldri lønne seg å påta seg ekstra kostnader som en forsikring for å begrense kostnadene som må til for å håndtere en avvikssituasjon [Wallace, 1999].

Hva karakteriserer så en fleksibel og robust løsning sammenlignet med en deterministisk? Dette er et umulig spørsmål å besvare på et generelt grunnlag, siden det avhenger av hvilke mulige avvik som kan inntreffe og hvilke mulige tiltak som kan settes inn for å håndtere avvikssituasjonen. Typisk vil en fleksibel og robust løsning i størst mulig grad holde ulike handlingsalternativer åpne til etter usikkerheten er avdekket, slik at man kan korrigere avgjørelsene når man ikke lenger opererer under usikkerhet. Ved bruk av en stokastisk modell vil sannsynligheten for at et avvik inntreffer og kostnaden ved de ulike handlingsalternativene ha avgjørende betydning for hvilken løsning som velges.

Det er vanskelig å oppnå robuste planer for et funksjonsområde uten å se det i sammenheng med de øvrige produksjonsplanene. Som vi tidligere har vært inne på tas det eksempelvis hensyn til materiellturneringene allerede når ruteplanene legges. Dermed er det lite hensiktsmessig å velge høye verdier på tidsparemetrene som gir romslige stasjonsopphold for materiellet, da dette fører til at det ikke blir samsvar mellom rute- og materiellplaner slik at unødvendig mye materiell må brukes. Likevel kan man til en viss grad ha et utbytte av å fokusere på robusthet innenfor hvert enkelt funksjonsområde. Eksempelvis kan materiellplanleggerne påvirke fleksibiliteten til materiellturneringene ved å plassere motorvognsett på strategiske steder i jernbanenettet slik at materiellet kan settes inn ved eventuelle avvik.

Et mangelfullt fokus på fleksibilitet og robusthet i den taktiske planleggingsfasen kan skape uheldige konsekvenser ved den operative driften. British Airways (BA) erfarte eksempelvis at avvikshåndteringen ved operativ kontroll ble svært vanskelig når



planene fra den taktiske planprosessen var *for* gode ut i fra en kostnadsvurdering. Dette kom av at planene var svært stramme og at ulike turneringer var flettet inn i hverandre, noe som førte til at det ble vanskelig å isolere avvik. For å bedre på denne situasjonen utviklet man arbeidsprosesser og beslutningsstøttesystemer for den operative kontrollen, slik at man enklere og raskere kunne finne alternative løsninger når en avvikssituasjon oppstod. SAS har opplevd lignende problemstillinger og valgt å gå tilbake til en tradisjonell strategi hvor flymaskinene kjører faste ruter [Larsen, 2002], noe som har nære paralleller med NSB's "rene pendler".

Disse erfaringene vil trolig til en viss grad kunne overføres til NSB og tilsier at arbeidsprosessene og verktøyene i den operative planleggingen også må vurderes når beslutningsstøttesystemer tas i bruk i den taktiske planleggingen.

8.6 Selvimplementert algoritme uten resultat

Vi oppnådde ingen resultater med vår implementasjon av en egen løsningsalgoritme basert på Benders dekomponering og branch and bound. Her vil bakgrunnen for denne mangelen på resultater bli beskrevet.

8.6.1 Konvergensproblemer

Ved numeriske beregninger i datamaskiner regnes det alltid med en endelig presisjon på tallene. Dette kommer av at datamaskiner representerer tallene som bitstrenger som har en endelig lengde og dermed kun kan representere tall innenfor en viss presisjon. Jo lengre bitstrenger som brukes for å representere tallene, jo mer presise kan tallene være, men samtidig øker behovet for lagringsplass. På grunn av denne endelige presisjonen kan det oppstå avrundingsfeil i beregninger. Dette kan blant annet få konsekvenser når flere beregnede verdier skal sammenlignes, slik at det må tillates en viss toleransegrense i sammenligningen som kan kompensere for eventuelle avrundingsfeil.

Som beskrevet i teorien og om implementasjonen i henholdsvis delkapittel 4.3.1 og 6.2.1 er det særlig to sentrale sammenligninger, eller betingelser, som styrer den implementerte løsningsalgoritmen. Dette er betingelsen som avgjør om en løsning på masterproblemet gir løsbare subproblemer, $Q_k(\bar{y}) \leq 0$, og betingelsen som avgjør om



optimal løsning er nådd, $\theta_k \geq G_k^P(\bar{y})$. Dersom en av disse betingelsene ikke er oppfylt kjøres en ny iterasjon med tilføring av et nytt kutt til masterproblemet. Videre løses problemene og betingelsene sjekkes på nytt. Dette gjentas til betingelsene oppfylles. I følge teorien skal dette skje i et endelig antall iterasjoner, men ved implementasjon kan man på grunn av mulige avrundingsfeil i datamaskiner risikere at programmet ender i en uendelig løkke.

Denne typen problematikk mener vi er bakgrunnen for at vi ikke har noen testresultater fra implementasjonen av Benders dekomponering. Alle tester vi har kjørt på datasett fra NSB har endt opp i uendelige løkker ved en av de to nevnte betingelsene. Dette gir seg utslag i at løsningsprosessen gjennomgår flere iterasjoner uten at masterproblemløsningen endres. Dermed endres heller ikke løsningen til subproblemene, slik at kuttene som legges til i masterproblemet er identiske og altså ikke bindende.

8.6.2 Verifikasjon av implementasjonen

For å kunne utelukke andre forklaringer på konvergensproblemet ble flere former for testing gjennomført. Underveis i implementasjonen ble det gjennomført både white- og black-box testing. Mindre konstruerte tester ble gjennomført med en toleranse på $1,0E-09$ og ga korrekte løsninger, dog med overraskende mange iterasjoner.

Siden selv disse små testtilfellene krevde relativt mange iterasjoner var det ønskelig med en forsikring om at de reelle testtilfellene virkelig endte i uendelige løkker og ikke bare hadde oppsiktsvekkende høy løsningstid. Derfor ble det gjort forsøk hvor løsningsrommet ble redusert. Dette ble gjort ved å legge til ekstra restriksjoner i masterproblemet slik at masterproblemvariablene måtte gis en verdi maksimalt 10% vekk fra den optimale verdien. Den optimale verdien som dannet grunnlaget for disse restriksjonene ble funnet ved å løse det samme problemet i modellen implementert i Xpress-Mosel. Heller ikke dette forsøket ga noe resultat.

Siste ledd i verifikasjonen var en manuell gjennomgang av algoritmen for to små testtilfeller. De fem problemene, master, to fase I problemer og to fase II problemer, ble implementert i Visual-Xpress (studentversjon 3.0.2) som er Xpress-MP i et grafisk grensesnitt. Deretter ble algoritmen stegvis gjennomgått ved å legge til kutt i



masterproblemet og oppdatere høyresidene i subproblemet manuelt i tråd med løsningene som ble oppnådd ved å løse problemene med simplex i Visual-Xpress. Underveis ble de oppdaterte høyresidene i subproblemene, dualverdiene for subproblemet, variabelverdiene for masterproblemene, alle målfunksjonsverdiene og kuttene sammenlignet med tilsvarende verdier fra en kjøring av den implementerte utgaven av algoritmen. Da vi underveis erfarte at Visual-Xpress tidvis ga ulovlige verdier på dualvariablene, ved at produktet av dualvariablene og høyresidene ikke var lik målfunksjonsverdien, ble problemene også implementert i Xpress-Mosel. Den implementerte algoritmen og den manuelle gjennomgangen ga samme resultater underveis med unntak av en forskjell i hvor mange desimaler som ble oppgitt, noe som baserer seg på at BCL oppgir løsninger med større presisjon enn Visual-Xpress og Xpress-Mosel.

8.6.3 Tilpasning av presisjon

Xpress-Optimizer berøres av datamaskiners begrensning beskrevet i delkapittel 8.6.1 slik at løsningene kan inneholde avrundingsfeil [Dash Optimization, 2001c]. Dette kan også føre til at løsningene som returneres er litt ulovlig ("slightly infeasible") [Laundy, 2002]. Løsningene som hentes ut i fra Xpress-BCL, heretter omtalt som BCL, er beregnet i Xpress-Optimizer og endrer ikke presisjon i BCL.

På bakgrunn av dette har vi implementert en toleransegrense som skal sørge for at et begrenset antall siffer brukes som grunnlag for sammenligningene som styrer algoritmen. Dessuten har vi forsøkt å bruke tilsvarende antall siffer ved oppdateringene av høyresidene i subproblemene og ved definering av kuttene som legges til masterproblemet. Ettersom det ikke finnes noen garanti for hvor mange gyldige siffer som returneres fra BCL blir størrelsen på denne toleransegrensen et vurderingsspørsmål. En høy toleranse kan gi den mest presise angivelsen av optimum, men er samtidig mer utsatt for avrundingsfeil. Samtidig kan lav toleranse gi rom for flere iterasjoner med svært små justeringer av løsningsverdiene. Vi forsøkte først $1,0E-09$ og $1,0E-06$ som er henholdsvis matrisetoleransen og toleransen i høyresidene til Xpress-Optimizer. Da dette ikke ga den ønskede effekten forsøkte vi også større verdier opp til $1,0E-03$. For den implementasjonen hvor masterproblemet løses til heltallighet for hver iterasjon (6.2.6) forsøkte vi også å runde



masterproblemvariablene til heltall, da løsningene tidvis hadde små avvik fra heltallighet og altså var litt ulovlige.

Disse forsøkene viste at lave toleransegrenser for de fleste testene ikke klarte å forhindre de uendelige løkkene. Toleransegrenser større enn $1,0E-06$ derimot førte til uløselige subproblemer i fase II. Vi fant ingen grenser som unngikk begge disse effektene.

Forklaringen på de uløselige subproblemene ligger i sammenhengen mellom toleransegrensen for høyresidene i Xpress-Optimizer, kalt lovlighetstoleranse (feasibility tolerance), og den grensen vi brukte. Betingelsen $Q_k(\bar{y}) \leq 0$ er nettopp en sjekk på om subproblemet er løsbart. Dersom denne betingelsen brytes er subproblemet ikke løsbart og masterløsningen må endres ved hjelp av et lovlighetskutt. Dermed kan ikke toleransegrensen for betingelsen settes høyere enn toleransegrensen i Xpress-Optimizer som berører høyresidene i subproblemet.

I forbindelse med arbeidet for å tilpasse toleransen i vår kode har vi vært i kontakt med representanter for leverandøren av Xpress-MP og blitt anbefalt å endre den omtalte lovlighetstoleransen [Laundy, 2002]. Dette har ikke blitt gjort da det viser seg at BCL likevel ikke har full støtte for Java, slik at kontrollparametre i Xpress-Optimizer ikke lar seg endre via BCL i Javakode. Dette lar seg kun gjøre i C-kode [Heipcke, 2002].



9 Konklusjon

Planavdelingen ved NSB Drift har blant annet ansvaret for å utarbeide materiellturneringsplaner for nærtrafikken rundt Oslo. I denne hovedoppgaven har vi utviklet en operasjonsanalytisk modell som kan benyttes for å støtte dette arbeidet i den taktiske fasen av planprosessen.

Mangel på variasjon i testsettene gjør at vi ikke kan trekke konklusjoner på et generelt grunnlag. Deler av konklusjonen forutsetter at trendene vi ser i testresultatene også er representative for et bredere utvalg av tester. Da testsettene baserer seg på reelle ruteplandata, mener vi likevel at trendene i resultatene er nyttige indikatorer.

Resultatene fra den kvalitative testingen viser at modellformuleringen vi har utviklet beskriver virkeligheten på en god måte. Ved å vekte parametre i målfunksjonen kan det gis insentiver til at enkelte kjøremønstre er å foretrekke. Siden ruteplanene som benyttes i nærtrafikken er svært stramme, er det essensielt å modellere tidsaspektet mest mulig presist. En modell uten sekvenser og snukanter vil derfor ha liten verdi. Det må imidlertid legges ned et arbeid i å differensiere tidsparametrene som inngår for at modellen skal kunne gi nytte i en praktisk planleggingssituasjon.

Testingen viser at løsnings tidene er svært sensitive både overfor ulike målfunksjoner og enkelte typer restriksjoner. Det er derfor viktig at brukeren er bevisst sine valg. For å gjøre en utsjekk av materiell er løsnings tidene svært gode fordi det er tilstrekkelig å modellere én dags horisont. Ved utarbeidelse av et utkast til en grunnplan hvor man følger et bruksmønster som baserer seg på prinsippene rundt "rene pendler" er løsnings tidene akseptable, men ikke gode. Dette skyldes at man er avhengig av å modellere en ukes horisont, noe som øker modellstørrelsen betraktelig. Løsnings tidene er uakseptable for alternative bruksmønstre som krever større modellstørrelser.

Planavdelingen ved NSB Drift vil kunne utvikle seg som organisasjon ved å ta i bruk et beslutningsstøttesystem. Mangelen på fleksibilitet i bruksmønstre gjør at vi ikke kan anbefale en integrering av modellen med uttestet løsningsalgoritme i dagens eksisterende systemer. Bakgrunnen er at en slik endringsprosess er krevende og kostbar, så dersom man går til anskaffelse av et verktøy som bidrar med



beslutningsstøtte må verktøyet være fleksibelt nok til å kunne anvendes også for større modeller. For å utføre konsekvensanalyser er dette ofte et behov. Derfor må det legges ressurser i å utarbeide en passende løsningsalgoritme. Modellformuleringen vår i kombinasjon med en alternativ løsningsalgoritme eller heuristikk vil være en mulig vei å gå.



10 Referanser

Aschehoug, K., Fodstad, M., Materiellplanlegging for NSB med en operasjonsanalytisk vinkling. (Prosjektoppgave ved Institutt for industriell økonomi og teknologiledelse, NTNU, 2001).

Bender, J. F., Partition procedures for solving mixed-variables programming problems. *Numerische Mathematik*, Vol. 4, s. 238-252, 1962.

Birge, J. R., Louveaux, F., *Introduction to Stochastic Programming*. New York: Springer-Verlag, 1997.

Borgman, J., Integrerad tidtabell med fasta intervaller – en enkel idé med stor effekt. *Nordisk Järnbane Tidsskrift*, Vol. 4, 2001. (Oversatt fra G. Pellandini)

Carøe, C. C., Tind, J., L-shaped decomposition of two-stage stochastic programs with integer recourse. *Mathematical Programming*, Vol. 83, s. 407-424, 1998.

Chvátal, V., *Linear Programming*. New York: W. H. Freeman and Company, 1983.

Cordeau, J.-F., Soumis, F., Desrosiers, J., A Benders Decomposition Approach for the Locomotive and Car Assignment Problem. *Transportation Research*, Vol. 34, Nr. 2, s. 133-149, mai 2000.

Cordeau, J.-F., Soumis, F., Desrosiers, J., Simultaneous assignment of locomotives and cars to passenger trains. *Operations Research*, Vol. 49, Nr. 4, juli-august s. 531-548, 2001a.

Cordeau, J.-F., Desaulniers, G., Lingaya, N., Soumis, F., Desrosiers, J., Simultaneous locomotive and car assignment at VIA Rail Canada. *Transportation Research*, Part B, Vol. 35, s. 767-787, 2001b.

Dantzig, G. B., Wolfe P., Decomposition principles for linear programs. *Operations Research*, Vol. 8, s. 101-111, 1960.

Dash Optimization, *Xpress-MP Essentials*, 2001b.

Dash Optimization, *Xpress-BCL Reference Manual*, 2001b.



Dash Optimization, *Xpress-Optimizer Reference Manual*, 2001c.

Dye, S., Tomasgard, A., Wallace, S. W., Two-Stage Service Provision by Branch and Bound. Presentert på ORSNZ 2000, Wellington, 1-2 desember 2000.

Geoffrion, A. M., Lagrangean relaxation for integer programming. *Mathematical Programming Study*, Vol. 2, s. 82-114, 1974.

Heipcke, Susanne, personlig kommunikasjon, 29.05.2002.

<http://www.nsb.no>, aksessert 05.04.2002

Kall, P., Wallace, S. W., *Stochastic Programming*. Chichester, UK: John Wiley and Sons, 1994.

Laporte, G., Louveaux, F. V., The integer L-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters*, Vol. 13, s. 133-142, 1993.

Larsen, J., Anvendelse af operasjonsanalyse. Disruption Management at British Airways. Presentert på NSB/SINTEF seminar Gardermoen, 25.04.2002.

Laundy, Richard, personlig kommunikasjon, 28.05.2002.

McDaniel, D., Devine, M., I modified Benders' partitioning algorithm for mixed integer programming. *Management Science*, Vol. 24, s. 312 – 379, 1977.

Nygreen, B., personlig kommunikasjon, 05.06.2002.

Rardin, R. L., *Optimization in Operations Research*. Upper Saddle River, New Jersey: Prentice Hall, Inc., 1998.

Schrijver, A., Minimum Circulation of Railway Stock. *CWI Quarterly*, Vol. 6, Nr. 6, 1993.

Stølan, A., Sæbø, H. J., Sætermo, I.-A. F., Tomasgard, A., Planleggingspraksis i NSB, Sintef Teknologiledelse, Rapport STF38 F00610, 2000.

Sætermo, I.-A. F., Tomasgard, A., NSB Drift og teknikk – forslag til endringer i planprosessen, Sintef Teknologiledelse, Rapport STF38 F01616, 2000.



Wallace, S. W., *Decision making under uncertainty: The art of modeling*. NTNU, 1999.

Williams, H. P., *Model Building in Mathematical Programming*. Chichester: John Wiley & Sons, Ltd., 1999.

Wolsey, L. A., Integer programming duality: price functions and sensitivity analysis. *Mathematical Programming*, Vol. 20, s. 173-195, 1981.



11 Bibliografi

Ahuja, R. K., Optimization in Action: Solving Real-Life Locomotive Scheduling Problems, <http://www.ise.ufl.edu/ahuja/>, aksessert 30.01.2002.

Ben-Khedher, N., Kintanar, J., Queille, C., Stripling, W., Schedule Optimization at SNCF: From Conception to Day of Departure, *Interfaces*, Vol. 28, Nr. 1, s. 6-23, jan-feb., 1998.

Christiansen, M., Inventory and time constrained ship routing : a mathematical programming approach. (Doktoringeniøravhandling, Institutt for industriell økonomi og teknologiledelse, NTNU, 1996.)

Dash Optimization, *Modeling with Xpress-MP*, 2001.

Detlefsen, N. K., Basis properties in multicommodity networks. (Doktoringeniøravhandling, Institutt for industriell økonomi og teknologiledelse, NTNU, 1999)

Gabteni, S., The Fleet Assignment Problem and its resolution, <http://www.cs.chalmers.se/Cs/Grundutb/Kurser/tranopt/slides/samig.ppt>, aksessert 30.01.2002.

Lingaya, N., Cordeau, J.-F., Desaulniers, G., Desrosiers, J., Soumis, F., Operational Car Assignment at VIA Rail Canada,

Löbel, A., Optimal Vehicle Scheduling in Public Transit. (Doktorgradsavhandling, Technischen Universität Berlin, 1998).

McBride, R., Advances in Solving the Multicommodity-Flow Problem, *Interfaces*, Vol. 28, Nr. 2, s. 32-41, mars-april, 1998.



Vedlegg A

Kildekode for Xpress-Mosel



Vedlegg B

Kildekode for Benders dekomponering i Java

(inkluderer Gomory-kutt)