

FFI RAPPORT

UTVIKLING AV EFFEKTIV ILDKRAFT MODELL

ANDRESEN Nils Ulrik

FFI/RAPPORT-2005/00313

Utvikling av Effektiv Ildkraft Modell (EIM)

Nils Ulrik Andresen

FFI/RAPPORT-2005/00313

FORSVARETS FORSKNINGSINSTITUTT
Norwegian Defence Research Establishment
Postboks 25, 2027 Kjeller, Norge

FORSVARETS FORSKNINGSINSTITUTT (FFI)
Norwegian Defence Research Establishment

UNCLASSIFIED

P O BOX 25
 NO-2027 KJELLER, NORWAY
REPORT DOCUMENTATION PAGE

SECURITY CLASSIFICATION OF THIS PAGE
 (when data entered)

1) PUBL/REPORT NUMBER FFI/RAPPORT-2005/00313	2) SECURITY CLASSIFICATION UNCLASSIFIED	3) NUMBER OF PAGES 40
1a) PROJECT REFERENCE FFI-I/858/911	2a) DECLASSIFICATION/DOWNGRADING SCHEDULE -	
4) TITLE UTVIKLING AV EFFEKTIV ILDKRAFT MODELL DEVELOPMENT OF AN EFFECTIVE ENGAGEMENT MODEL		
5) NAMES OF AUTHOR(S) IN FULL (surname first) ANDRESEN Nils Ulrik		
6) DISTRIBUTION STATEMENT Approved for public release. Distribution unlimited. (Offentlig tilgjengelig)		
7) INDEXING TERMS IN ENGLISH:		
a) <u>Operational Analysis</u>		IN NORWEGIAN:
b) <u>Simulation</u>		a) <u>Operasjonsanalyse</u>
c) <u>Modelling</u>		b) <u>Simulering</u>
d) <u>Mathematical Programming</u>		c) <u>Modellering</u>
e) <u>Effective Engagement</u>		d) <u>Matematisk programmering</u>
		e) _____
THESAURUS REFERENCE:		
8) ABSTRACT A method and model for evaluating effective engagement is developed. The resulting model shows what a set of weapons and platforms can achieve in a scenario when used effectively across a wide range of possible circumstances. The uses of weapons are optimised to achieve the most valuable result. The model is suitable for a computer implementation.		
9) DATE 2007-02-28	AUTHORIZED BY This page only Jan Erik Torp	POSITION Director

ISBN 978-82-464-1220-7

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE
 (when data entered)

INNHOOLD

	Side	
1	SAMMENDRAG	7
2	INNLEDNING	7
2.1	Bakgrunn	7
2.2	Problemstilling	8
3	TEORI	9
3.1	Relevant litteratur	9
3.2	Kill Chain	9
3.3	Operative krav	10
3.4	Verdifunksjon	12
4	VALG AV MODELL OG METODE	14
4.1	Optimering og simulering	14
4.2	Matematisk formulering av EIM	15
5	UTVIKLING AV MODELLEN	17
5.1	Effektiv Ildkraft Modell	18
5.2	Algoritme	19
5.3	Monte Carlo-simulering	21
5.4	Detaljer for å optimere bruk av våpen	25
5.5	Implementering	31
6	KONKLUSJON	31
APPENDIKS		
A	STØRRELSEN PÅ FEIL I OPTIMERING AV VÅPENBRUK PER DAG	32
B	INPUT	35
B.1	Struktur	35
B.2	Scenario	35
B.3	Våpenprioritets matriser (VPM)	37
B.4	Verdifunksjon	38
	Litteratur	40

UTVIKLING AV EFFEKTIV ILDKRAFT MODELL

1 SAMMENDRAG

Effektiv Ildkraft Modell (EIM) er en modell utviklet av prosjekt 858 Delprosjekt 1 *Langtrekkende våpenkapasitet*. Prosjektet studerer mulige systemløsninger for norsk LPV, og modellen er også utviklet for dette.

Effektiv ildkraft er i denne rapporten brukt i betydning å anvende våpensystemer på en optimal måte til å oppnå ønsket effekt til riktig tidspunkt under alle tenkelige forhold. Utrykket "effektiv ildkraft" blir brukt omtent på samme måten som "Effective Engagement" i NATO-terminologi.

For å studere effektiv bruk av våpensystemer (våpen og plattform) er det nødvendig å se på noen teorier om våpenlevering (kill chain), operasjonelle krav og hvordan å evaluere ulike løsninger (verdifunksjon). Med dette utgangspunkt blir det valgt et modellkonsept for EIM som baserer seg på en stokastisk simulering (Monte Carlo simulering) og optimering av våpen. Problemstillingen EIM skal løse, blir formulert matematisk før modellen blir detaljert. For å optimere bruk av våpen under usikkerhet blir det brukt en metode som kombinerer heltallsprogrammering og bruk av regler som styrer våpenvalg. (Regler blir også optimert.) Metoden for heltallsprogrammering som ble valgt, er en approksimativ metode som ofrer en eksakt løsning av optimeringsproblemet for å redusere antallet beregninger.

Resultatet er en detaljert metode og modell for å kunne sammenligne hva ulike strukturer kan utrette i et scenario. Modellen (EIM) vil produsere resultater som tar hensyn til kravene oppdragene stiller og som er optimert med hensyn til hensikten med Forsvarets deltakelse i scenariet. (Hensikt er input til modellen.) EIM er egnet for implementering på PC.

2 INNLEDNING

2.1 Bakgrunn

Denne rapporten utgjør en del av dokumentasjonen fra FFI-prosjekt 858 Delprosjekt 1 *Langtrekkende våpenkapasitet*. Formålet med rapporten er å dokumentere en metode og modell prosjektet har utviklet for å sammenligne LPV-strukturers effektive ildkraft. Metoden er også anvendbar for andre våpentyper enn LPV og er et godt eksempel på hvordan et slikt problem kan løses.¹

¹ I alle eksempler i denne rapporten er det tatt utgangspunkt i LPV. Det å studere andre typer våpen kan bety at det finnes andre begrensninger enn de som er omtalt, men metoden kan fungere like bra uavhengig av hvilke begrensninger som gjelder.

Formålet til prosjektet er å gi vurderinger rundt innføringer og bruk av LPV i Forsvaret i et fellesoperativt perspektiv, innenfor realistiske økonomiske rammer og i en alliert kontekst. Rapportens innhold utgjør en del av grunnlaget for prosessen med å etablere alternative systemløsninger for LPV. Rent konkret er modellen brukt for å identifisere hva som er hensiktsmessige antall våpen av ulike typer i systemløsningene, og også til en hvis grad sammenligne nytte i forhold til kostnad for ulike våpentyper. Hvordan modellen er brukt og resultater er dokumentert i rapporten *Strukturutvikling og analyse av LPV-systemløsninger* (1).

2.2 Problemstilling

Spørsmålet som blir forsøkt besvart med EIM er

- Hvilken evne har denne systemløsningen til å utføre oppdrag i en alliert operasjon,² under forutsetning av at våpensystemene blir brukt på en optimal måte, til å levere effekt til riktig tidspunkt, under alle tenkelige forhold?

For å kunne si noe fornuftig om denne spørsmålsstillingen er det nødvendig å forutsette at det er behov for og ønske om at norsk LPV bidrar med så mye som mulig til operasjonen.

Med forutsetningen nevnt ovenfor er det mulig å konkretisere enklere spørsmål:

- Hvor mange oppdrag av forskjellig type kan systemløsningen påta seg?
- Er systemløsningen i stand til å ta oppdragene til riktig tid?
- Er det begrensninger som gjør at hele kapasiteten ikke blir brukt selv om behovet er der?

Prosjektet har gjort en avgrensning om ikke å se på ledelses- og sensorressurser. Spørsmålene vil derfor dreie seg om begrensninger som kommer av våpen og plattformer.³

Rapporten tar for seg en løsning av denne problemstillingen med teori, metode, modell og utvikling av modellen. Implementasjon av modellen er i liten grad omtalt i rapporten da detaljer på dette nivået fort blir spesifikt for den typen våpen som skal studeres.

Kapittel 3 tar for seg teori og bakgrunnsinformasjon som er brukt for å velge modell. Kapittel 4 presenterer den valgte modellen og en matematisk formulering av problemet. Kapittel 5 presenterer nødvendig input og output og detaljerer hvordan modellen må behandle denne inputen for å få resultatet på hensiktsmessig form. (Kapittel 5 inneholder også et lite avsnitt om implementasjon.) Kapittel 6 er konklusjon.

² En alliert operasjon forstås som en operasjon hvor Forsvaret ikke har ansvaret for å løse hele oppdraget. Om det er krav eller ikke til å løse alle oppdragene gir to fundamentalt forskjellige situasjoner. I forbindelse med modellutviklingen vil det bli tatt valg for å fokusere på operasjoner hvor Forsvaret ikke har ansvaret for å løse alle oppdragene, men det er meget mulig at en implementasjon også kan gi svar på den andre problemstillingen.

³ Antall og type plattformer kan kalles systemløsningens struktur. (En systemløsning er metoder, struktur (personell, materiell og organisasjon), prosedyrer, doktriner, bygg og anlegg.) Siden EIM bare skiller systemløsningene på struktur og er ordet *struktur* benyttet istedenfor *systemløsning* i resten av rapporten.

3 TEORI

For å kunne si noe om en systemløsnings evne til å levere den riktige effekten og levere den til rett tid er det nødvendig å ha en god forståelse av hva som skal til for å levere denne effekten. Dette kapitlet gir et eksempel på hvordan prosessen med å levere LPV blir fremstilt, hvilke krav dette stiller til systemløsningene (strukturene) og gir grunnlaget for hvilke egenskaper Effektiv Ildkraft Modell skal ha. Aller først vil noen referanser til aktuell litteratur om operasjonsanalyse bli gitt.

3.1 Relevant litteratur

Det finnes mengder av litteratur som studerer lineærprogrammering, beslutningsregler og stokastiske modeller, men å finne noen artikler som studerer en kombinasjon av heltallsprogrammering, optimering av beslutningsregler og stokastisk modellering, som brukt i denne rapporten, har vist seg vanskelig. De Farias og Van Roy diskuterer en generell metode for å bruke lineærprogrammering til "omtrentlig" dynamisk programmering i sin artikkel (2). Problemstillingen som diskuteres i artikkelen (2), er ikke så lik at den kunne vært brukt her, men artikkelen er interessant fordi det fokuseres på en metodikk som kan optimere store stokastiske problem på en effektiv måte. Antallet tilstander det er nødvendig å beregne i slike problem, øker ofte eksponentielt med antall variable. Derfor vil større problemer være umulige å løse i praksis selv om det finnes en analytisk løsning. De Farias og Van Roy velger derfor bort en eksakt løsning for å være i stand til å løse store problemer også i praksis.

I denne rapporten har det vært nødvendig å velge en ikke eksakt løsning for i det hele tatt å være i stand til å løse optimeringsproblemet (se kapittel 5.4 og fotnote 13 i det samme kapitlet). For å se eksakte løsninger av heltallsprogrammering og optimering er lærebok av Jaiswal: *Military Operations Research – Quantitative decision making* (3) en god kandidat. Alternativt kan Winston's lærebok *Operations Research - Applications and Algorithms* (4) brukes.

I denne rapporten blir fordeling av sorties (kampfly) foretatt gjennom optimering. En alternativ måte å gjøre dette på er diskutert i artikkelen *A Ruled Based Approach for Aircraft Dispatching to Emerging Targets* av Mishra, Dr. Batta og Dr. Szczerba (5). I artikkelen blir det brukt en løsning som involverer et beslutningstre istedenfor en matematisk optimering.

3.2 Kill Chain

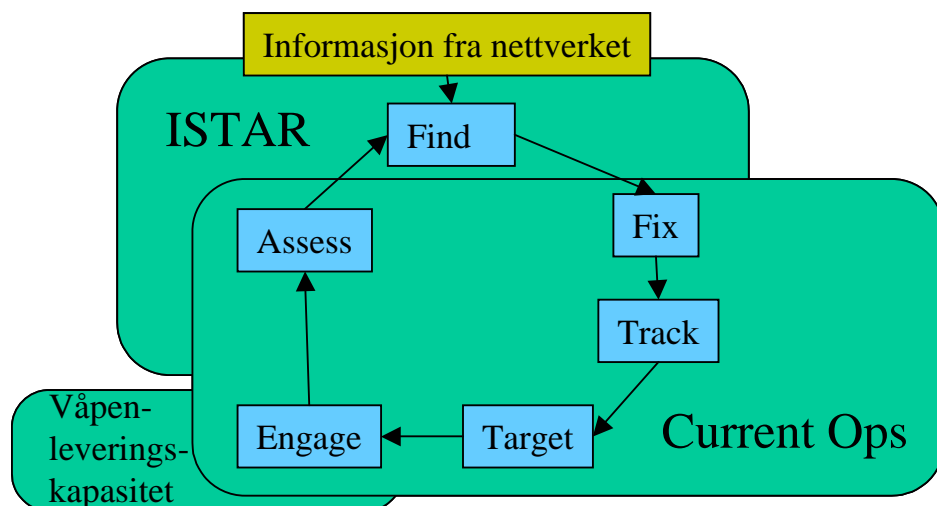
”Kill Chain” er et mye brukt begrep for å omtale prosessen fra å oppdage et nytt mål til å levere effekt på målet. Dale Scott Caffall og James Bret Michael diskuterer *kill chain*-begrepet i sin rapport (6). Nedenfor vises det som Caffall og Michael identifiserer som United States Air Force sin definisjon av *kill chain*. Den består av følgende elementer:

- Finne (find): Oppdage et nytt mål
- Identifisere (fix): Identifisere målet (venn fiende og type), nøyaktig posisjon og dets betydning

- Følge (track): Overvåke målet (følge eventuell forflytning) og identifisere trusler rundt
- Vurdere målet (target): Weaponeering og velge plattform til å utføre oppdraget, ta hensyn til ROE i weaponeering
- Engasjere (engage): Overføre K2 til shooter, ta hensyn til ROE for å levere våpen, og levere våpen
- Evaluere (assess): Evaluere effekten av våpenet

I det nettverksbaserte forsvar kan det å oppdage et nytt mål (finne) være utført av hvem som helst. De andre oppgavene vil være styrt og kontrollert av beslutningsnoder som driver den pågående engasjementledelsen (current ops) (se Figur 3.1). Beslutninger kan være sentralisert til et taktisk hovedkvarter eller delegert til "brukere."

Forskjellige ISTAR-kapasiteter vil være de som utførerer Find, Fix, Track og Assess, mens en plattform med våpenleveringskapasitet vil utføre Engage. Modellen fokuserer på kapasitetene og ikke hvilke plattformer som gjør hva. (Det er selvfølgelig fullt mulig at en plattform med våpenleveringskapasitet også har en ISTAR-kapasitet.)



Figur 3.1 *Prosessen for å angripe et mål som blir identifisert som fiendtlig. Fix, track target, engage og assess vil være styrt av et taktisk hovedkvarter som driver engasjementsledelse. ISTAR-kapasiteter vil utføre find, fix, track og assess. En LPV-kapasitet vil utføre engage.*

3.3 Operative krav

Kill Chain fokuserer på prosessen med å utføre et enkelt oppdrag. Effektiv Ildkraft Modell (EIM) skal derimot gi svar på hvor mye systemløsningen utretter totalt og det skal være mulig å sammenligne strukturer til systemløsningene. For å klare dette på en enkel måte bør mest mulig av detaljene for hvert enkelt oppdrag være håndtert eller i hvert fall tatt hensyn til utenfor modellen, i input. Modellen må isteden ta et litt mer generelt utgangspunkt og se på sammenhengen mellom flere oppdrag.

Å dekke behovet for effektiv ildkraft er en tjeneste systemløsningen utfører for en sjef. Det sjefen trenger, er en kapasitet som kan levere ønsket effekt i målet til riktig tid hver gang det er nødvendig. Dette stiller krav

- til å ha de riktige våpnene (krav til våpen),
- til å ha nok våpen av forskjellige typer (krav til antall våpen)
- og til å være beredt til å løse oppdragene når behovet er der selv om det skjer flere ting på en gang (krav til å kunne løse flere oppdrag samtidig).

Disse punktene blir gjennomgått i de neste avsnittene.

Krav til våpenet er at våpenet skal ha den riktige presisjonen, ha den riktige sprengkraften i målet, ha nok *stand-off* avstand i forhold til trusler og ikke være hemmet av værforhold. Hvor mange våpen det er nødvendig å bruke mot ulike mål under ulike omstendigheter bør være input til modellen. Metoder og nødvendig informasjon for å regne ut antall våpen på et oppdrag er inkludert i rapporten *Systemløsninger for langtrekkende presisjonsstyrte våpen* (7). Med informasjonen i (7) er det mulig å samle informasjonen om forskjellige våpens ytelse og begrensninger på et konkret oppdrag og ut fra dette prioritere hvilke våpen som bør brukes. En prioritert liste over våpnenes ytelse kalles våpenprioritetsmatrise (VPM), og herfra er det antatt at all nødvendig informasjon om våpnene er inkludert i en eller flere VPM.⁴ Eksempler på VPM blir presentert i rapporten *Systemløsninger for langtrekkende presisjonsstyrte våpen* (7).

Krav til antall våpen bør være et krav som stilles på hvert enkelt oppdrag. Det er naturlig å anta at i enhver sammenheng vil Forsvaret være en mindre del av en større styrke - mest sannsynlig en NATO-styrke. Derfor er det på ingen måte meningen at Forsvaret skal klare alle oppdragene i operasjonen. Samtidig er det vanskelig å avgrense hva som er Norges andel av oppgavene, men det er sannsynlig at det er ønske om at Norge skal bidra maksimalt innenfor norsk kompetanse og kapasitet. Har ikke Forsvaret den beste løsningen, skal det ikke mye til før det blir satt restriksjoner på norsk deltakelse. (Operasjon Allied Force (Kosovo) i 1999 er et historisk eksempel.)

Resonnementet i forrige avsnitt betyr at EIM må ha en avgrensning av hvilke oppdrag som skal utføres som er dynamisk i forhold til hvilken struktur som testes. Ved å inkludere alle oppdragene som skal utføres i scenariet (uansett hvor stort scenariet er), og samtidig ha en dynamisk tilpasning av hvilke oppdrag systemløsningen skal utføre, er det mulig å få fornuftige resultater både for ulike nisjeløsninger og for bredere sammensatte systemløsninger. Det blir da mulig å bruke type og antall oppdrag utført som parametre for å sammenligne systemløsningene (i motsetning til at det skal være et krav til å ha et visst antall våpen for å utføre et bestemt antall oppdrag). Totalt antall oppdrag av forskjellig type blir nødvendigvis input, dette kalles scenario, men å velge hvilke oppdrag som skal utføres, må løses inne i modellen (se kapittel 4.1).

Krav til å kunne løse flere oppdrag samtidig er nødvendig for å sikre at en systemløsning kan løse oppdrag når det er behov for det. Hvor mange oppdrag som kan løses på en gang er

⁴ Nødvendig informasjon vil inkludere alt fra antall våpen på oppdraget, hva det koster å løse oppdraget, krav til værforhold og hvor mange JSF sorties som er nødvendig.

avhengig av egenskaper ved hver enkelt systemløsning og delsystemer til løsningen. De viktigste er

- hvor mange plattformer som trengs for å ødelegge eller nøytralisere et mål,
- hvor lang tid det går fra et mål er oppdaget til våpen er levert (reaksjonstid),
- hvor lang tid det tar fra en plattform har gjennomført et oppdrag til det kan påta seg et nytt (avhengig av om det må laste opp nye våpen),⁵
- tilgjengelighet på plattformer, sensorer og C4I,
- mobiliteten til plattformer og sensorer og
- rekkevidden til våpenet.

Som tidligere nevnt skal ikke prosjektet evaluere sensorer og ledelse. Reaksjonstid og hvor mange oppdrag systemløsningen klarer samtidig, er da kun avhengig av hvor mange plattformer som kan levere våpen og hvilke oppdrag systemløsningen skal utføre. Antall leveringsplattformer som finnes, vil i EIM være en karakteristikk av systemløsningen (strukturen). Antall oppdrag strukturen kan utføre på en gang, kan da være en parameter for å sammenligne strukturene.

3.4 Verdifunksjon

Effektiv Ildkraft Modell (EIM) skal vurdere evne til å utføre oppdrag når våpensystemene blir brukt på en optimal måte (se kapittel 2.2). Det å definere hva som er en optimal måte er ingen enkel spørsmålsstilling. Et grunnleggende prinsipp må være at et våpensystem skal brukes til de oppgavene det er best egnet til, men hvis Forsvaret er ute etter en kapasitet som kan gi en bestemt type effekt, er det nødvendig å ta hensyn til dette. Hvis hensikten med LPV kun er å bidra med mest mulig i konflikten må bruken av våpen reflektere dette. Er hensikten med LPV å kunne ta ut flest mulig mål på stor rekkevidde må EIM kunne optimere bruken av våpen på de oppdragene som krever størst rekkevidde og anta at de oppdragene som krever kortere rekkevidde løses av andre systemer. Flere slike eksempler kunne nevnes. Konklusjonen må være at metodene som brukes i EIM må kunne optimere bruken av våpen slik at en får testet strukturene på det eller de kriteriene som blir lagt til grunn for en eventuell anskaffelse.

Forrige avsnitt avslører at for å kunne optimere bruken av våpen er det nødvendig å gjøre en avveining mellom ulike oppdrag. En måte å gjøre dette på er å gi hvert oppdrag en verdi avhengig av karakteristika ved oppdraget. (Karakteristika kan for eksempel være måltype, størrelse, lokalisering, tidspunkt, hvor godt beskyttet målet er og så videre.) Det å tildele en verdi til hvert oppdrag på denne måten kalles en verdifunksjon. En slik verdifunksjon kan være input til modellen slik at det er mulig å forandre hvilke kriterier som blir lagt til grunn for optimeringen.

Inne i modellen vil verdifunksjonen gi en verdi til hvert mål og det oppdraget som har høyest verdi vil i utgangspunktet bli prioritert først. Men modellen skal optimere resultatet, det vil si at det ikke er om å gjøre å utføre det mest verdifulle oppdraget, men å utføre det settet av oppdrag

⁵ Ulempen ved at plattformen må "lade om" vil variere fra system til system. Det er for eksempel stor forskjell på å "lade om" et fly og en MLRS.

som har høyest sum av verdier.⁶ På denne måten er det entydig definert hva som er en optimal bruk av våpnene, og resultater for ulike strukturer kan da sammenlignes ved hjelp av statistikk på hvilke og hvor mange oppdrag strukturene utførte.

Med denne framgangsmåten vil betydningen av resultatene være avhengig av hvilke kriterier verdifunksjonen bruker. Det eneste som helt generelt er mulig å si om resultatene er at EIM optimerer bruken av våpen etter kriterier spesifisert i input. (Å ha verdifunksjonen som input bør gjøre det mulig å unngå skjulte valg inne i modellen, slik at betydningen av resultatene helt og holdent er et produkt av input.)

Verdifunksjonen er en vurdering av oppdragene, men implisitt kan verdifunksjonen også oppfattes som en vurdering av egenskapene til våpensystemene. Det er viktig å være oppmerksom på dette. Hvis kriteriene verdifunksjonen bruker blir for detaljerte, vil det kunne oppstå en situasjon der bare en type våpen kan tilfredsstille kriteriene. Det våpenet som er mest kosteffektivt mot denne måltypen vil da automatisk være det våpenet som er best egnet. Resultatet i dette tilfellet er innlysende og det å bruke EIM på å analysere et slikt tilfelle vil i liten grad bidra med ny informasjon. EIM kan heller ikke bli brukt som alibi for å påstå at et våpensystem er optimalt hvis kriteriene i utgangspunktet er så snevre at kun et våpen kan være svaret. Men med en mer generell verdifunksjon bør det være mulig å bruke modellen til å gi resultater som er interessante å sammenligne.

Et eksempel på en fornuftig bruk av verdifunksjonen kan være å gi oppdragene en verdi som er størst den første dagen og som så reduseres utover i scenariet. Resultatene blir i dette tilfellet en vurdering av samtidighet, hvor resultatet er hvor mange oppdrag strukturen klarte å utføre når det var om å gjøre å utføre flest oppdrag over kortest mulig tid. Hva ulike strukturer klarer å oppnå i et slikt tilfelle kan være en interessant vurdering, som kan hjelpe til med å foreta et valg av struktur.

I det som er beskrevet over, er det antatt at verdifunksjonen bare blir brukt inne i modellen til å optimere valg av oppdrag. Det er antatt at statistikk på hvilke og hvor mange mål er den eneste outputen fra modellen, men det å ha en verdifunksjon som input gir en mulighet for å summere verdien til alle utførte oppdrag og la denne summen være en del av resultatene fra modellen. En slik bruk av verdifunksjonen setter strenge krav til modellen. For at summert verdi for to strukturer skal være sammenlignbare er det nødvendig å sørge for at de to strukturene blir prøvd på identiske scenarier hvor absolutt alle parametre er helt like. Valg av modell og metode i de senere kapitlene tar ikke hensyn til dette kravet. Det er likevel mulig å bruke EIM på denne måten ved å gjøre noen mindre justeringer (se siste avsnitt i appendiks B.2).

⁶ Eksempel: De tre høyest prioriterte oppdragene, en, to og tre, har verdi 5, 4 og 3, men det er ikke nok våpen til å løse alle tre oppdragene. Våpnene kan enten brukes til å løse oppdrag en eller til å løse både oppdrag to og tre. Da vil optimeringen velge å løse oppdrag to og tre fordi $3+4$ er større enn 5.

4 VALG AV MODELL OG METODE

4.1 Optimering og simulering

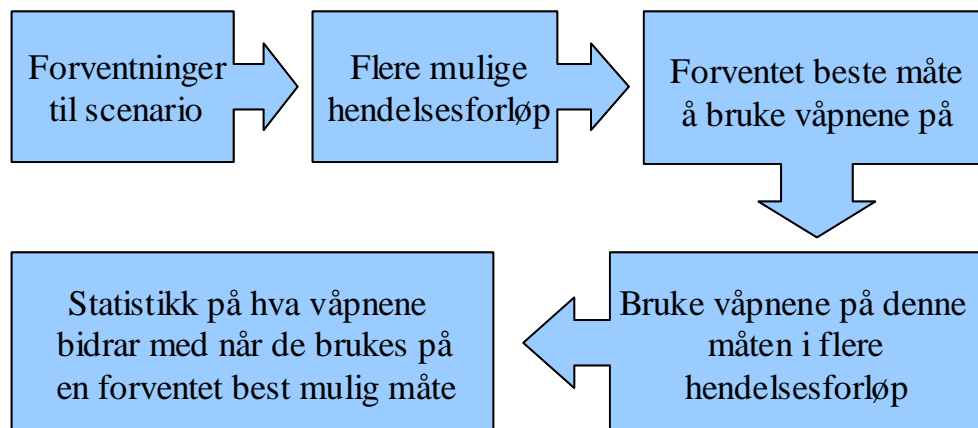
Kapittel 3.3 gir noen føringer for hvordan modellen skal se ut. Det ble funnet at en metode for å velge hvilke oppdrag systemløsningen utfører, må skje inne i modellen. For at resultatene for ulike systemløsninger skal være sammenlignbare med hverandre bør det i størst mulig grad være slik at den valgte måten å bruke våpnene på er den *beste* måten å bruke våpnene på. Med andre ord må valg av oppdrag optimeres.

Å optimere bruk av strukturen (våpen og plattformer) i et enkelt scenario vil gjøre resultatene sterkt avhengige av hendelsesforløpet i det konkrete scenariet. Det er heller ikke realistisk å tro at utviklingen av et enkelt scenario skal kunne gi gode generelle resultater som også gjelder andre situasjoner. En ide er å representere variasjoner ved å simulere flere hendelsesforløp. Hvert hendelsesforløp kan representere en mulig utvikling av det samme scenariet, og bruk av våpen og plattformer optimeres innenfor hvert hendelsesforløp til å gi et best mulig resultat.

Ideen fra forrige avsnitt sørger for at resultatene ikke er bundet av et enkelt hendelsesforløp, men den gir ikke noe godt svar på om, og i tilfelle hvordan, usikkerhet blir representert. En videreutvikling av ideen er å la optimering av oppdrag skje for en dag av gangen og uten å kjenne detaljer om de andre dagene. Med en slik ide er det helt nødvendig å inkludere et sett med kriterier eller regler som kan begrense valg av oppdrag (slik at en ikke bruker opp alle våpen av en type på første dag når det er stort behov for den samme våpentypen senere i scenariet).

En måte å representere usikkerhet kan være å ha det samme settet med kriterier eller regler for alle hendelsesforløpene i et enkelt scenario. Kriteriene eller reglene for valg av oppdrag kan representere den forventet beste måten å bruke strukturen på i det aktuelle scenariet. Hendelsesforløpene kan representere ulike måter et scenario kan utvikle seg på, og å bruke strukturen på den forventet beste måten vil da antakelig lede til en god, men ikke nødvendigvis optimal bruk av våpnene. Ideen representerer en optimering av forventning. Å finne de kriteriene og reglene som gjennomsnittlig gir best resultat representerer den måten å bruke våpnene på som "sjefen" forventer er best, istedenfor å si at all informasjon er kjent og så finne ut hva som ville skjedd hvis "sjefen" alltid gjorde de riktige valgene.

Skal EIM bruke ideene beskrevet ovenfor, blir modellen som beskrevet i Figur 4.1. Først defineres forventninger til hva som kommer til å skje i scenariet og flere hendelsesforløp blir utviklet på bakgrunn av denne informasjonen. En analyse av hendelsesforløpene må så identifisere den antatt beste måten å bruke strukturen på. Den antatt beste måten å bruke strukturen på er et sett med kriterier eller regler som forteller hvilke oppdrag denne strukturen bør utføre. Til slutt gjøres en simulering hvor det blir trukket ulike hendelsesforløp hvor bruk av strukturen blir optimert for hver dag, men optimeringen er styrt av reglene og kriteriene.



Figur 4.1 En ide for å simulere flere hendelsesforløp for å introdusere usikkerhet i scenariet. Modellen optimerer bruk av våpen og plattformer (strukturen) samtidig som det blir tatt hensyn til at i en enkelt situasjon må operativ sjef velge å bruke våpnene på den måten han eller hun forventer er best og det er ikke nødvendigvis den måten som faktisk er best fordi operativ sjef ikke kjenner det nøyaktige hendelsesforløpet i resten av scenariet

Ideen utviklet så langt betyr at modellen skal optimere forventningen til hva det er mulig å oppnå i et scenario og at denne optimeringsoppgaven består av to deloppgaver:

- Å optimere kriterier og regler slik at en finner den forventet beste måten å bruke våpen og plattformer på.
- Å optimere bruk av våpen og plattformer (strukturen) for hver dag (innenfor de begrensningene kriteriene og reglene setter).

Å bruke denne metoden vil da produsere to resultater: Hva som er beste måte å bruke våpnene på, og statistikk på hva våpnene bidrar med når de brukes på en forventet optimal måte.

4.2 Matematisk formulering av EIM

For hver dag i scenariet lages det en liste over oppdrag. Hvert oppdrag inneholder opplysninger om måltype, lokalisering, aktiv beskyttelse av målet og værforhold. Ressursutnyttelse den dagen skal optimeres for å få høyest mulig verdi innenfor scenariets begrensninger og reglene for bruk av våpen. Reglene bestemmer hvilke typer oppdrag systemet påtar seg og er en kvantitativ framstilling av hvordan våpnene skal brukes. Reglene skal optimeres for hvert våpensystem i hvert scenario, og den kombinasjonen som gav best utnyttelse av systemløsningen er en del av svaret og dette er den beste måten å bruke våpensystemene på.

I denne sammenhengen er våpensystem definert til å være en bestemt våpentype eller et sett av flere lignende våpentyper. (Det siste gjør det lettere å sammenligne kategorier av våpen som for eksempel "bomber" og "kryssermissiler" uten å skulle spesifisere hver enkelt våpentype i strukturen (se avsnitt om *våpensystem* i appendiks B.3).)

Det er enklest å først formulere hvordan bruken av våpen optimeres for hver dag (innenfor de

begrensningene kriteriene og reglene setter). Dette problemet kan formuleres som et heltallsprogrammeringsproblem, og en slik beskrivelse av problemet er gitt nedenfor.

Mengder

- P Den totale mengden oppdrag i løpet av en dag. For hver dag har alle oppdrag et nummer $1, 2, \dots, N$
- K Mengden våpensystem i LPV-strukturen
- J Mengden dager i scenario (dag $1, 2, \dots, N$)

Parametere

- V_{kp} Verdi av oppdrag p når utført av våpensystem av type k , hvor $k \in K$ og $p \in P$
- A_{kp} Antall våpen av type k som trengs på oppdrag p , hvor $k \in K$ og $p \in P$
- M_{kj} Antall våpen av type k som er igjen på dag j , hvor $k \in K$ og $j \in \{0, J\}$
(M_{k0} er definert som det totale antall våpen av type k i strukturen før scenariet starter)
- L_{kp} Antall flysorties våpensystem k trenger på oppdrag p , hvor $k \in K$ og $p \in P$
- L_j Antall flysorties tilgjengelig på dag j , hvor $j \in J$
- R_{kp} Om våpensystem k kan utføre oppdrag p (kan være begrenset av regler eller faktiske begrensninger slik som begrensninger på rekkevidde), hvor $k \in K$, $p \in P$ og $R_{kp} = \{0; 1\}$

Variabel

- x_{kpj} Om oppdrag p er utført av våpensystem k på dag j , hvor $k \in K$, $p \in P$, $j \in J$
og $x_{kpj} = \{0; 1\}$

Maksimer uttrykket

$$\forall j: \quad \sum_p \sum_k x_{kpj} V_{kp} \quad (4.1)$$

med begrensninger

$$\forall k, p, j: \quad 0 \leq x_{kpj} \leq R_{kp} \quad (4.2)$$

$$\forall p, j: \quad \sum_k x_{kpj} \leq 1 \quad (4.3)$$

$$\forall k, j: \quad \sum_p A_{kp} x_{kpj} \leq M_{kj} \quad (4.4)$$

$$\forall j: \quad \sum_k \sum_p L_{kp} x_{kpj} \leq L_j \quad (4.5)$$

Alle uttrykkene (4.1) til (4.5) gjelder for hver dag, j , noe som uttrykker at (4.1) optimeres dag for dag uten å ta hensyn til de andre dagene. M_{kj} i (4.4) vil være en konstant som settes for hver dag. Antall våpen som finnes på dag j , M_{kj} , er lik antall våpen som fantes dagen før (dag $j-1$) minus antall våpen brukt dagen før,

$$M_{kj} = M_{k(j-1)} - \sum_p A_{kp} x_{kp(j-1)}.$$

Konstanten M_{kj} må for hver dag settes før optimeringsproblemet blir løst. Dette representerer derfor ingen tilbakekobling mellom dagene i optimeringsproblemet.

Betingelse (4.2) sier at hvert våpensystem kan utføre hvert oppdrag minimum null antall ganger og maksimalt R_{kp} antall ganger. R_{kp} har verdien null eller en. Noe som begrenser x_{kpj} til verdien null eller en.

Betingelse (4.3) sier at maksimalt en type våpen (k) kan utføre et konkret oppdrag (p). (To våpentyper kan ikke bli brukt til å løse det identisk samme oppdraget, en blir alltid valg.)

Betingelse (4.4) begrenser våpenforbruket på dag j til å være mindre enn antall våpen som finnes på dag j .

Betingelse (4.5) krever at antall sorties brukt må være mindre enn antall sorties tilgjengelig.

Problemet beskrevet over er et heltallsprogrammeringsproblem som løses for en dag av gangen uten kobling mellom dagene. En løsning av dette problemet vil gi en optimal utnyttelse av våpen og plattformer innenfor de begrensningene reglene, R_{kp} , setter. Å optimere reglene er et annet problemet som også må løses (se kapittel 4.1). En matematisk formulering av dette problemet følger.

Reglene defineres som variable, r_{kp} (i motsetning til å definere dem som parametre R_{kp}). En enkel måte å optimere reglene på er å betrakte det andre optimeringsproblemet som en funksjon, f , som gir

$$x_{kpj} = f(r_{kp}) \quad (4.6)$$

hvor funksjonen, f , er definert av uttrykkene (4.1) til (4.5). Problemet som må løses er da:

Maksimer uttrykket

$$\sum_j \sum_p \sum_k x_{kpj} V_{kp} \quad (4.7)$$

når x_{kpj} er gitt av (4.6), og r_{kp} er variabelen.

5 UTVIKLING AV MODELLEN

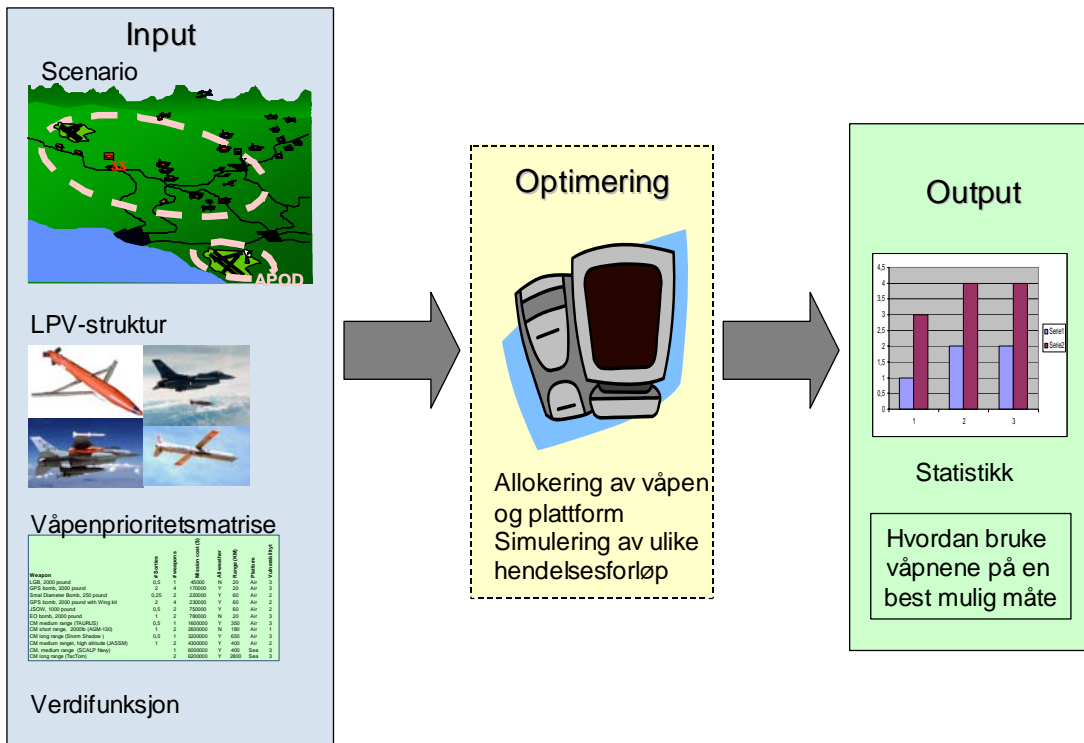
I dette kapittelet blir modellen presentert. Først gis en beskrivelse av hvordan EIM fungerer. I kapittel 5.2 presenteres algoritmen tenkt brukt i modellen, og kapittel 5.3 og 5.4 detaljerer hvordan modellen skal fungere. Kapittel 5.5 nevner kort hvordan modellen har blitt implementert.

5.1 Effektiv Ildkraft Modell

Figur 5.1 gir et enkelt bilde av modellen. Modellen tar følgende input:

- Scenariet må for hver type oppdrag spesifisere følgende informasjon
 - totalt antall oppdrag av denne typen,
 - fordeling som forteller nødvendig rekkevidde eller *stand-off* avstand for land-, luft- og sjøplattformer (lokalisering og trusselbilde) og
 - forventning til når disse oppdragene vil dukke opp i scenariet.
- LPV-struktur må fortelle antall og typer våpen, og antall kampflysorties tilgjengelig per dag
- En våpenprioritetsmatrise for hver type oppdrag. Våpnene i denne matrisen må være prioritert i forhold til kosteffektivitet. Matrisene må i tillegg inneholde følgende informasjon for hver våpentype
 - hvor mange våpen det er nødvendig å bruke på oppdraget,
 - om våpnene leveres fra en land-, luft- eller sjøplattform,
 - hvor lang rekkevidde våpenet har,
 - hvor mange kampflysorties som er nødvendig for å utføre oppdraget,
 - kostnad for å bruke disse våpnene på dette oppdraget og
 - hvor ofte værforholdene er gode nok til at våpnene kan brukes i dette scenariet (prosent).
- En verdifunksjon som kan være avhengig av måltype, kostnad, hvilken rekkevidde som kreves på oppdraget, antall oppdrag av samme type totalt, antall oppdrag av samme type den dagen og hvilken dag det er i scenariet (se appendiks B.4).

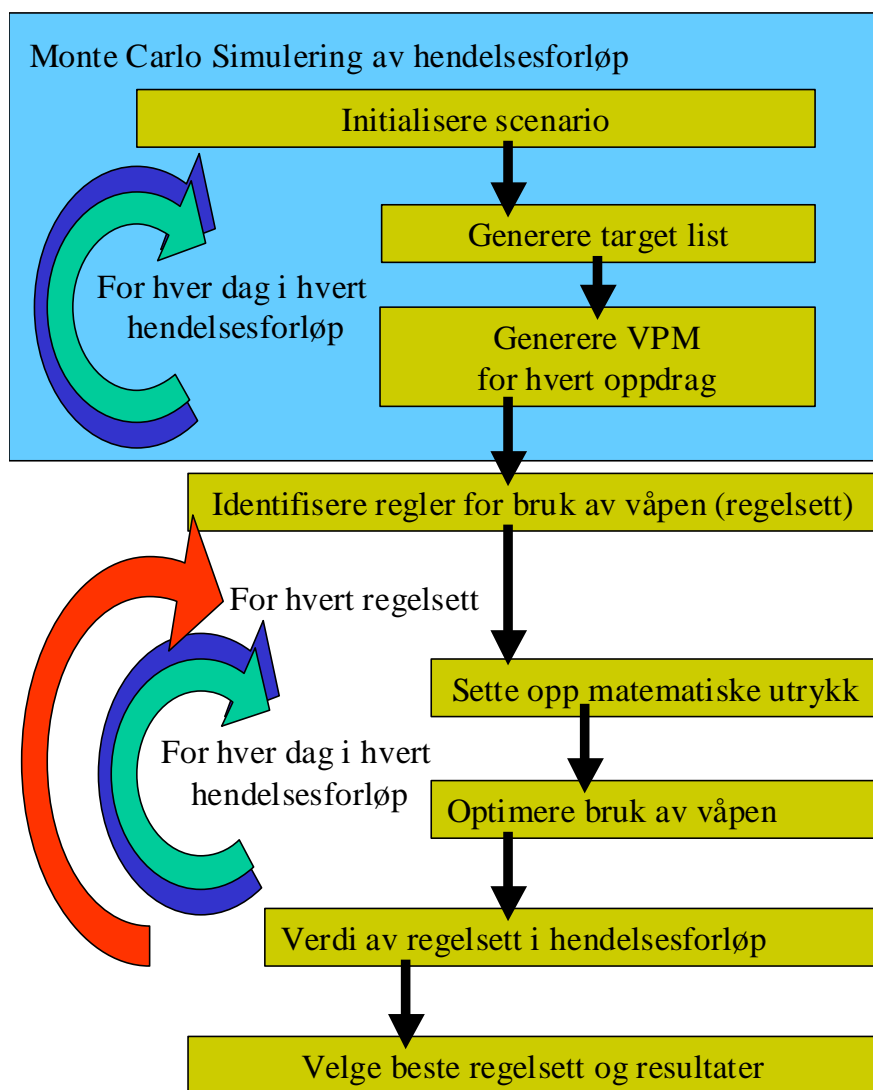
Som vist i Figur 5.1 optimerer modellen bruk av våpen Dette gjøres ved å simulere flere hendelsesforløp i scenariet. Output er hvordan våpnene bør brukes i dette scenariet og statistikk på hva systemløsningene utretter når våpnene blir brukt på denne måten.



Figur 5.1 Effektivildkraft Modell. Oversikt over input og output

5.2 Algoritme

Figur 5.2 viser en oversikt over hvordan modellen fungerer. En mer detaljert beskrivelse følger i kapittel 5.3 og 5.4. Modellen trekker flere hendelsesforløp som alle er basert på ett og samme scenario. Trekningene er en Monte Carlo-simulering hvor hvert hendelsesforløp er en liste med oppdrag for hver dag, hvert oppdrag har en prioritert liste over hvilke våpen som kan og bør brukes på oppdraget og hvert oppdrag har også en verdi. Forhold rundt oppdragene som lokalisering, luftvern (som setter krav til *stand-off* avstand) og værforhold på stedet er forskjellige for hvert oppdrag, slik at alle oppdrag i prinsippet er unike.



Figur 5.2 *Algoritme for Effektiv Ildkraft Modell. Regler for hvordan våpnene kan brukes blir optimert slik at forventningen blir høyest mulig*

Når Monte Carlo-simuleringen er ferdig, identifiseres ulike måter å bruke våpnene på. Kombinasjonen av måter å bruke de forskjellige våpnene på danner regelsett. Alle regelsett blir prøvd på alle hendelsesforløpene for å finne gjennomsnittlig verdi til hvert regelsett. Den forventet beste måten å bruke våpnene på er det regelsettet som gir høyest gjennomsnittlig verdi.⁷ (Regelsettet med høyest verdi *velges*.)

For å finne gjennomsnittlig verdi til et regelsett blir våpenbruk optimert for en dag av gangen innenfor de begrensninger som blir satt av reglene, oppdrag og ressurser som våpen og

⁷ Under forutsetning av at antallet hendelsesforløp er så stort at det er statistisk signifikant kan gjennomsnittet betraktes som et estimat på forventningsverdien til regelsettene. Det regelsettet som har høyest estimert forventningsverdi blir betraktet å være det regelsettet som beskriver den forventet beste måte å bruke våpnene på.

leveringsplattformer.⁸ Verdien til et regelsett er så en funksjon av verdien til disse oppdragene. For å kunne kontrollere resultatene er det nødvendig å lage statistikk på hvilke og hvor mange oppdrag som ble utført (for alle regelsett). (Dette er ikke nødvendig for å finne resultatet, men kun for senere kontroll av resultatet, derfor vises ikke dette som et eget punkt i Figur 5.2.)

Metoden brukt i modellen (Figur 5.2) er grovt beskrevet at den bruker forventninger til scenariet for å lage flere mulige hendelsesforløp, og så identifiserer den forventet beste måten å bruke våpnene på ved å teste alle måter å bruke våpen på i disse hendelsesforløpene. Så langt er dette i henhold til ideen i Figur 4.1 (de tre første boksene i den figuren). Det neste steget i Figur 4.1 er å bruke våpnene på forventet beste måte i flere mulige hendelsesforløp. Med metoden beskrevet over er dette allerede gjort, og det er ikke nødvendig å gjøre nye simuleringer hvis all nødvendig informasjon om de allerede utførte simuleringene er tatt vare på. Det siste steget i modellen er dermed å hente fram de resultatene som gjelder for det beste regelsettet.

5.3 Monte Carlo-simulering

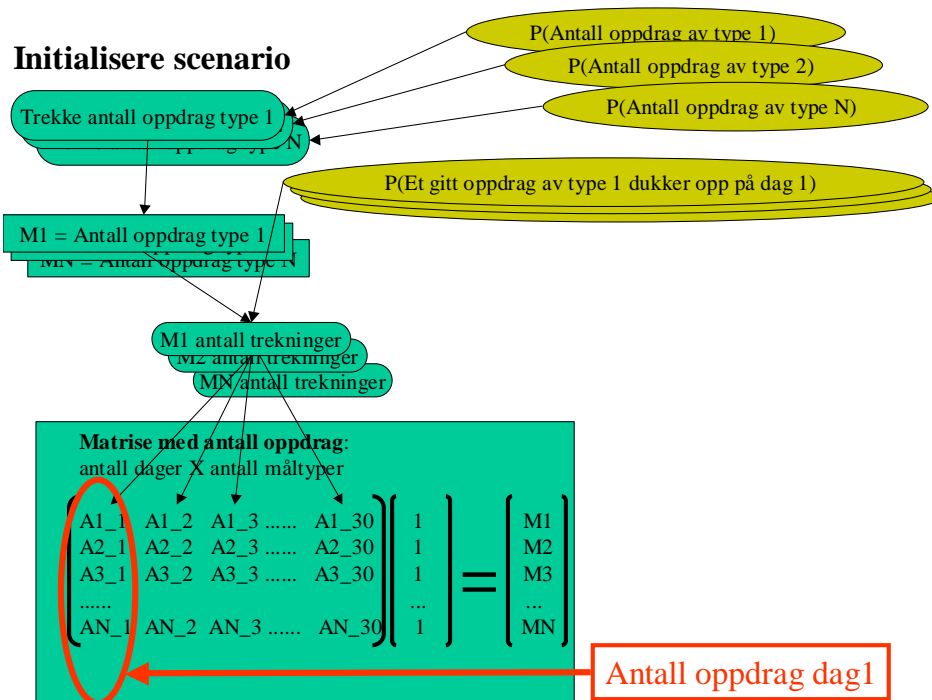
Figur 5.3 viser hvordan grunnlaget for et hendelsesforløp blir laget ved å trekke hvor mange oppdrag som dukker opp hver dag. Trekningene er avhengig av input (gulgrønne ovaler i figuren). Resultatene av trekningen blir lagret i en matrise hvor en kolonne er en dag og en rad en type oppdrag. Summen av en rad er det totale antall oppdrag mot en måltype. Summen av en kolonne er antall oppdrag en på en dag.

Fra matrisen som viser antall oppdrag per dag blir det utviklet en *target list* for hver dag. En *target list* inneholder alle oppdragene og opplysninger som er unike for hvert oppdrag slik som værforhold, lokalisering og trusselbildet rundt målet.

Værforhold blir modellert ved å vurdere hvor dårlig været er i forhold til perfekte forhold (vindstille og skyfritt). For hver dag blir det trukket et tall mellom null og en som sier hvor ofte det er bedre vær enn denne dagen. (Er tallet 0,09 en dag vil det si at det i 9 % av dagene er det bedre vær, og i 91 % av dagene er det dårligere vær.) Været blir trukket fra en uniform (flat) fordeling og representerer de generelle værforholdene den dagen i scenariet, mens værforholdene på et spesifikt oppdrag blir bestemt av dette tallet pluss en justering for lokale værforhold på $\pm 0,2$.⁹

⁸ Kun kampfly er vurdert som en leveringsplattform som har begrensninger i den utgaven av modellen prosjekt 858 Delprosjekt 1 *Langtrekkende våpenkapasitet* har implementert.

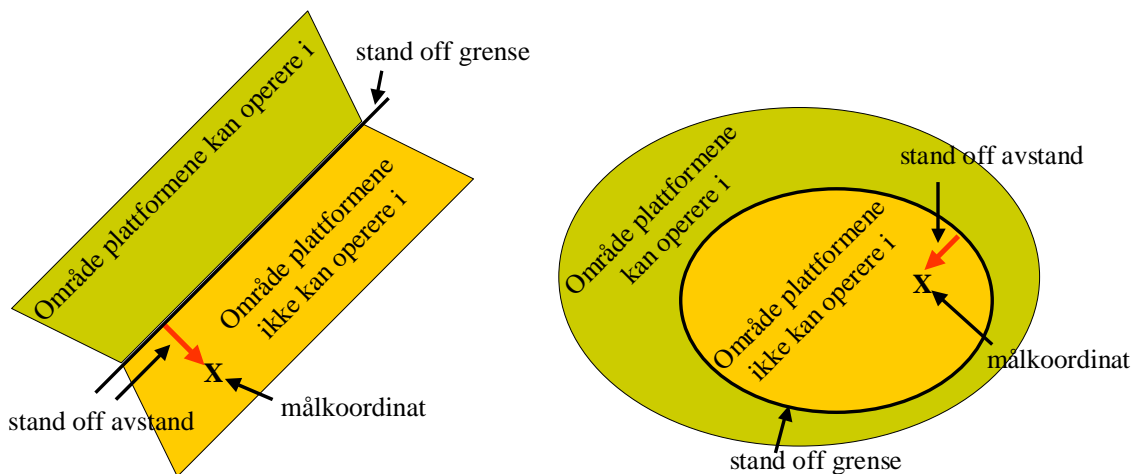
⁹ Dette gjøres på en slik måte at den sammensatte fordelingen fortsatt er uniform mellom null og en.



Figur 5.3 Initialisere scenario. Verdier for totalt antall oppdrag og når oppdragene skjer blir trukket fra fordelinger i scenario input. En matrise med antall oppdrag er grunnlaget for å lage et mulig hendelsesforløp i dette scenariet

Lokalisering og trusselbildet rundt målet er modellert ved å sette en grense på området hvor våpenleveringsplattformene kan operere og vurdere hvor langt fra denne grensen målet er. Denne avstanden er heretter kalt *stand-off* avstand. Denne modellen er illustrert i Figur 5.4. Den røde pila i figuren er *stand-off* avstanden som alltid måles fra grensen. (Dette gjelder også når målet ligger inne i området hvor leveringsplattformene kan operere i, men da defineres *stand-off* avstand til å være negativ. En negativ *stand-off* avstand blir betraktet som om det ikke er krav til *stand-off* avstand.)

Stand-off avstand er avhengig av type plattform som skal utføre oppdraget. Dette tas hensyn til ved å gi muligheten for å legge til en verdi som er avhengig av om plattformen er luft-, land- eller sjøbasert. (Størrelsen på denne verdien kan også være avhengig av type oppdrag.)



Figur 5.4 Modell for lokalisering i scenario. I stedet for å trekke målkoordinat trekkes avstand til nærmeste område hvor våpenleveringsplattformer kan operere

Modellene for værforhold, lokalisering og trusselbildet rundt målet betyr at de opplysningene som må legges til for hvert oppdrag i *target list* er en *stand-off* avstand og et tall som beskriver værforhold (se Figur 5.5).

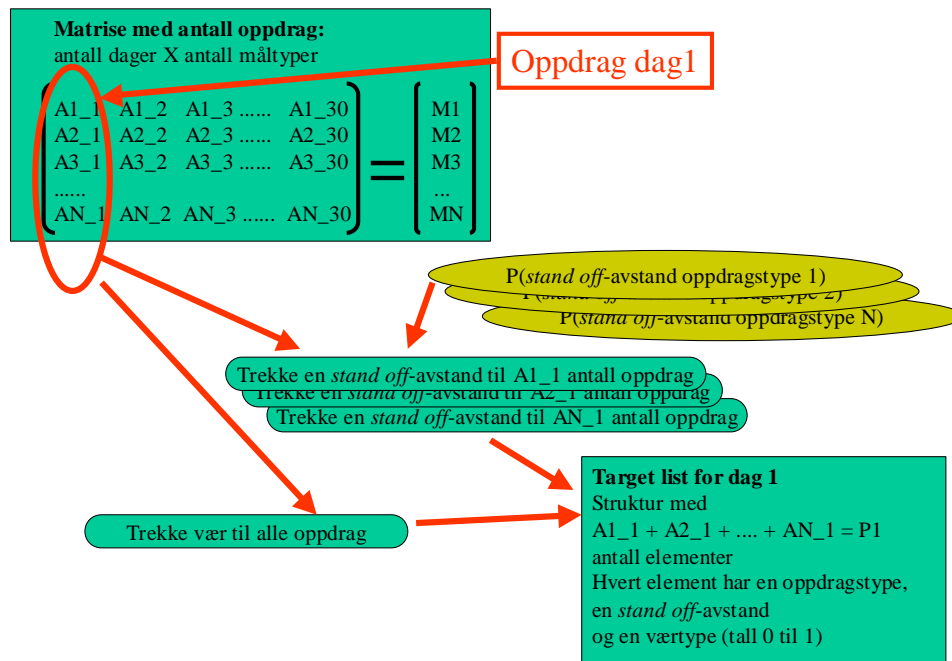
For hvert oppdrag i *target list* blir det laget en rangert liste over hvilke våpensystem som kan løse oppdraget. Figur 5.6 viser prosessen fra *target list* til en våpenprioritetsmatrise (VPM) for hvert oppdrag.

Den viktigste inputen i denne prosessen er en våpenprioritetsmatrise (VPM) hvor hvert våpen tilhører et våpensystem. VPM er en rangert liste over hvilke våpen som egner seg best til å løse oppdrag mot en bestemt måltype. Våpnene i VPM kategoriseres i våpensystem (se appendiks B om input). For hvert enkelt oppdrag blir våpnene i den aktuelle VPM sammenlignet med værforhold og kravene til *stand-off* avstand i oppdraget, og de våpnene som ikke kan brukes blir strøket fra listen i en oppdragsesifikk VPM.

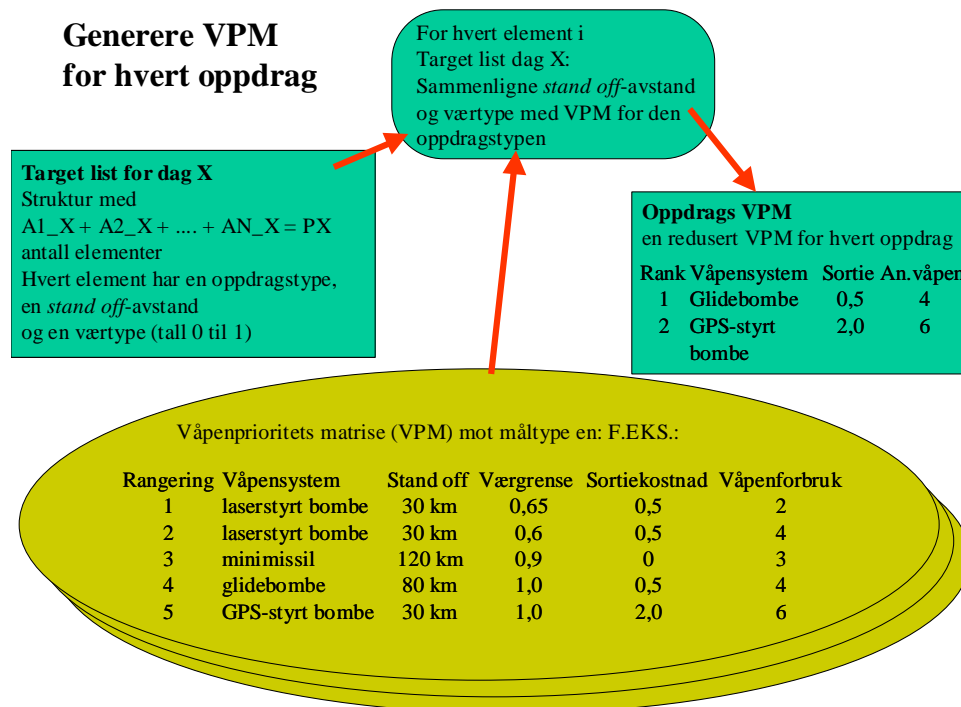
I tillegg til det som er blitt beskrevet til nå blir også verdien til hvert enkelt oppdrag beregnet i Monte Carlo-simuleringen. Denne beregningen er bestemt av verdifunksjonen som er input. Verdien er avhengig av hvilket våpen som utfører oppdraget. Se appendiks B for detaljer.

Resultatene fra Monte Carlo simuleringen er da *target list* for hver eneste dag i alle hendelsesforløpene, en VPM til hvert enkelt oppdrag og en verdi for hvert oppdrag avhengig av hvilket våpen som utfører oppdraget.

Generere target list



Figur 5.5 Generere target list. En liste med oppdrag blir spesifisert for hver dag



Figur 5.6 Generere VPM for hvert oppdrag. Fra en våpenprioritetsmatrise med informasjon om våpensystemene (VPM) lages en VPM for hvert oppdrag

5.4 Detaljer for å optimere bruk av våpen

Valg av modell i kapittel 4 betyr at det å optimere bruk av våpensystemer er et todelt problem. De to problemene som må løses, er å optimere regler (kriterier) slik at en finner den forventet beste måten å bruke våpen og plattformer på og å optimere bruk av våpen for hver enkelt dag.

Hensikten med reglene er å ta hensyn til andre faktorer enn de som allerede er tatt hensyn til i våpenprioritetsmatrisene (VPM). Resultatet fra *Monte Carlo* simuleringen er en VPM for hvert enkelt oppdrag. De våpnene som står i en slik VPM er de våpnene som tilfredsstillende alle kravene som oppdraget setter, og disse kravene er effekt (nok effekt av riktig type, men heller ikke for mye), rekkevidde og krav som værforholdene stiller. Andre faktorer som det også er nødvendig å ta hensyn til i *weaponing*¹⁰, er antall våpen av ulik type, antall sortier (plattformer) og eventuelt andre faktorer som verdifunksjonen er avhengig av. (Verdifunksjon er også input, og kan være avhengig av for eksempel kostnad (kosteffektivitet) eller måltype (se kapittel 5.1).)

Et enkelt eksempel er at en bombetype A er den eneste bomben i strukturen som har allværskapasitet. Strukturen inneholder lite av denne typen bombe slik at det er viktig å spare disse bombene til de dagene det faktisk er dårlig vær.

Definisjonen av en regel er at den bestemmer hvor høyt prioritert en våpentype minst må være i VPM for at det skal være aktuelt å bruke dette våpenet på dette oppdraget. For hver våpentype er det kun en regel som gjelder for alle oppdrag uansett type oppdrag. (Det betyr at det for hver våpentype vil det være et antall regler som er like stort som antall mulige plasseringer i en VPM.)

Den relativt enkle definisjonen av regler i forrige avsnitt tilfredsstillende hensikten. Våpnene i VPM er prioritert etter kostnad forbundet med de ulike våpenløsningene (se appendiks B.3 for definisjon av kostnad). Den strengeste regelen for bruk av våpen er at våpenet må være prioritert først i VPM for å kunne brukes på dette oppdraget. Den strengeste regelen betyr at våpenet bare kan bli brukt når det er det mest kosteffektive våpenet, og dette er implisitt det samme som å si at dette våpenet bare kan bli brukt i de situasjonene hvor det faktisk er det beste våpenet til å utføre oppdraget. (En slik regel vil typisk bli brukt i tilfeller der det er mange oppdrag i forhold til antall våpen som kan brukes på oppdragene.) I motsatt ende av skalaen vil en regel som tillater bruk uavhengig av plassering i VPM bety at det er akseptabelt å bruke våpenet i alle tilfeller hvor våpenet tilfredsstillende kravene oppdraget stiller. (Typisk vil en slik regel bli brukt i tilfeller hvor det er få oppdrag denne våpentypen er god på i forhold til antallet våpen.) Mellom disse to ytterpunktene er det et antall regler som definerer grader av hvor mye våpenbruken begrenses.

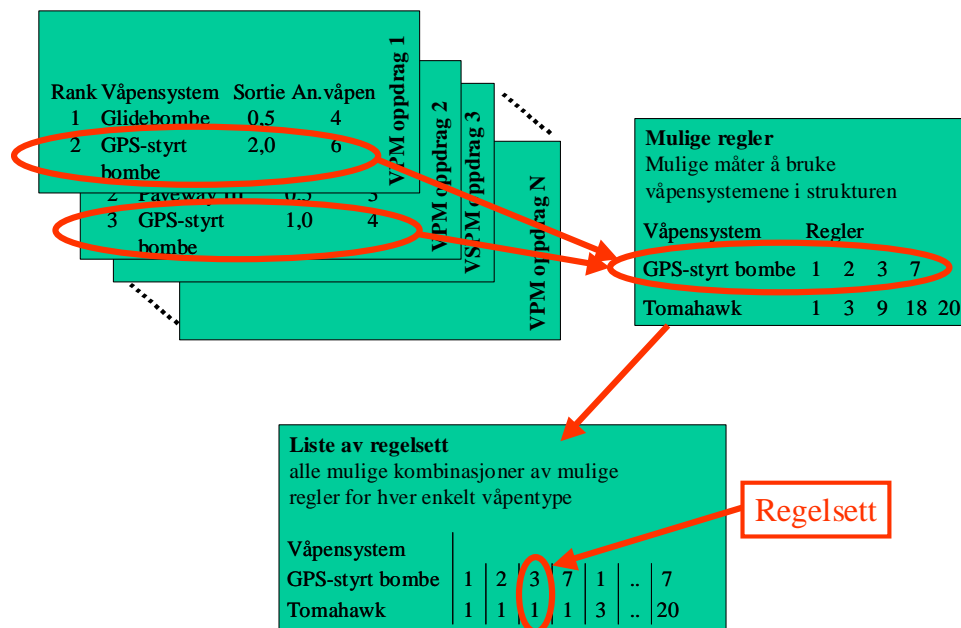
Det er et konkret tilfelle hvor reglene ikke tilfredsstillende hensikten. Det er hvis den våpentypen som det finnes en begrenset mengde av er den mest kosteffektive våpentypen også i tilfeller der

¹⁰ *Weaponing* er å velge hvilket våpen som skal bli brukt på hvilket oppdrag.

det finnes en stor mengde andre nesten like gode våpen i strukturen. For eksempel hvis en bombe med allværskapasitet er det mest kosteffektive våpenet også på dager med fint vær, så vil det ikke være mulig å finne en regel som sørger for at bomben med allværskapasitet blir spart til dager med dårlig vær. Det betyr at feilprioriteringer kan oppstå i slike tilfeller. En måte å løse dette på kan være i tillegg å definere en regel om at en type våpen alltid skal prioriteres brukt foran en annen type våpen (i tilfeller hvor begge står på samme VPM) og teste ut denne regelen på lik linje med de andre reglene. Denne metoden er ikke benyttet i den nåværende implementasjonen. Det betyr at bruk av EIM forutsetter at slike sammenhenger blir identifisert utenfor modellen.¹¹

Hvilke regler som skal brukes, velges ikke, men EIM identifiserer selv det regelsettet som er mest optimalt for strukturen i scenariet.¹² Det første EIM må gjøre, er å identifisere alle regelsett for de våpnene som er med i strukturen. (Det gir ingen ekstra informasjon å identifisere regler for våpentyper som ikke er inkludert i strukturen.) Denne prosessen er vist i Figur 5.7. Resultatet er en liste med regelsett som definerer alle måter å bruke våpnene på. Antall regelsett er lik antall regler for hver våpentype ganget sammen. (Eksempelet vist i figuren har $4 \cdot 5 = 20$ måter å bruke våpnene på.) Det betyr at antallet regelsett kan bli veldig stort hvis strukturen inneholder mange våpentyper (se kapittel 5.5 og appendiks B.1).

Identifisere regler for bruk av våpen (regelsett)



¹¹ Måten reglene er definert på i nåværende implementasjon forutsetter at våpen som er mer fleksible (at de har lengre rekkevidde eller bedre allværskapasitet) er dyrere enn alternativene. Når dette ikke er tilfelle er optimeringsproblemet egentlig enklere. Er det veldig enkelt kan hele problemet løses analytisk. Hvis det allikevel er et komplekst delproblem, kan dette løses i EIM ved å tilpasse struktur og scenario.

¹² Et regelsett er en kombinasjon av en regel for en eller flere våpentyper. Hvilket regelsett som er optimalt er avhengig av både strukturen og scenariet.

Figur 5.7 Identifisere regler for bruk av våpen (regelsett). Liste av regelsett definerer alle mulige måter våpnene som er inkludert i strukturen, kan brukes på

Når en liste av regelsett er definert er all nødvendig informasjon til stede for å sette opp de matematiske uttrykkene som skal optimere løsningen (se kapittel 4.2). Dette gjøres for hvert regelsett for en dag av gangen. Prosessen med å sette opp parametrene er illustrert i Figur 5.8. Utrykkene A_{kp} , L_{kp} , V og R i Figur 5.8 tilsvarer parametrene A_{kp} , L_{kp} , V_{kp} og R_{kp} , definert i kapittel 4.2. De to siste nødvendige parametrene, L_j og M_{kj} , er begrensninger som kommer direkte fra input (se Figur 5.9). L_j er antall sorties per dag og M_{kj} er antall våpen av ulik type i strukturen. Antall sorties tilgjengelig er forhåndbestemt i input og antall våpen i strukturen er også input hvor det kun er nødvendig å trekke fra antall våpen brukt tidligere i hendelsesforløpet.

Optimeringsproblemet i Figur 5.9, å maksimere verdien av oppdragene utført (x_{kpj}), har relativt enkle matematiske sammenhenger og begrensninger beskrevet i kapittel 4.2. Det finnes metoder for heltallsprogrammering (*IP-programming*) som løser dette problemet eksakt (se for eksempel (3) eller (4)). Det viser seg allikevel at kompleksiteten er så stor at disse metodene ikke kan benyttes på så store problemer som studeres her.¹³ I stedet brukes en enklere metode som tar utgangspunkt i at de oppdragene som må utføres for å maksimere den totale verdien, er de oppdragene som har størst verdi i forhold til innsatsen. Alle mulige kombinasjoner av oppdrag og våpentyper blir derfor prioritert i en liste etter

$$\frac{V_{kp}}{C_{kp}} \quad (5.1)$$

hvor

V_{kp} er verdien på oppdrag p når utført med våpen k og

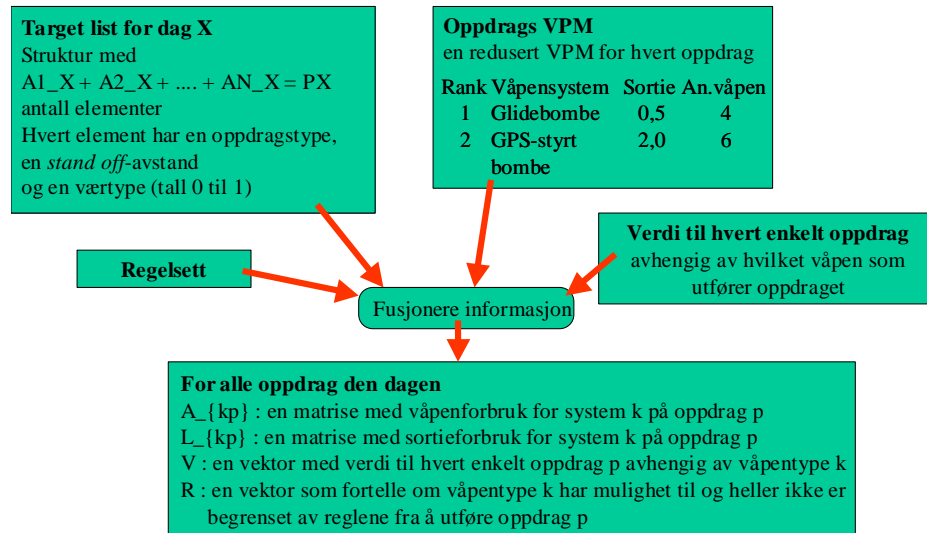
C_{kp} er en kostnad for å utføre oppdrag p med våpentype k som er proporsjonal med hvor stor andel av ressursene (våpen og sorties) som er nødvendig på oppdrag p . (C_{kp} spesifiseres i våpenprioritetsmatrisene, se appendiks B.)

C_{kp} inneholder en komponent som er proporsjonal med antall våpen og en komponent som er proporsjonal med antall sorties. Dette er viktig for at resultatene skal være noenlunde fornuftige. Det å balansere med hvor mye sorties teller i forhold til våpen kan være en utfordring, og den utfordringen må løses i input.

¹³ Optimeringsproblemet i Figur 5.9 skal løses kun for en dag av gangen, men på en dag kan alle de ulike typene våpen bli brukt til å løse alt fra ingen til alle oppdragene. Antall kombinasjoner av våpen og oppdrag er lik antall våpen ganget med antall oppdrag, og dette er da antall variable i problemet. Hver variabel kan ha to verdier, at oppdraget blir utført med dette våpenet, eller at det ikke ble utført med dette våpenet. Antall mulige kombinasjoner av slike binære variable er 2^n hvor n er antall variable. Med 500 oppdrag og 3 våpentyper blir antall variable 1500 og det totale antallet kombinasjonsmuligheter er da 2^{1500} som er ca 10^{451} , dvs 451 nuller bak det første sifferet. I dette problemet er allikevel antall kombinasjonsmuligheter litt mindre, fordi hvert oppdrag kan bli løst kun en gang. Kompleksiteten i optimeringsproblemet i Figur 5.9 er $(\text{antall våpentyper i strukturen} + 1)^m$ hvor m er antall oppdrag. Med 500 oppdrag og 3 våpentyper blir antall mulige kombinasjoner da bare ca 10^{301} . Dette er mer enn dagens datamaskiner klarer å håndtere.

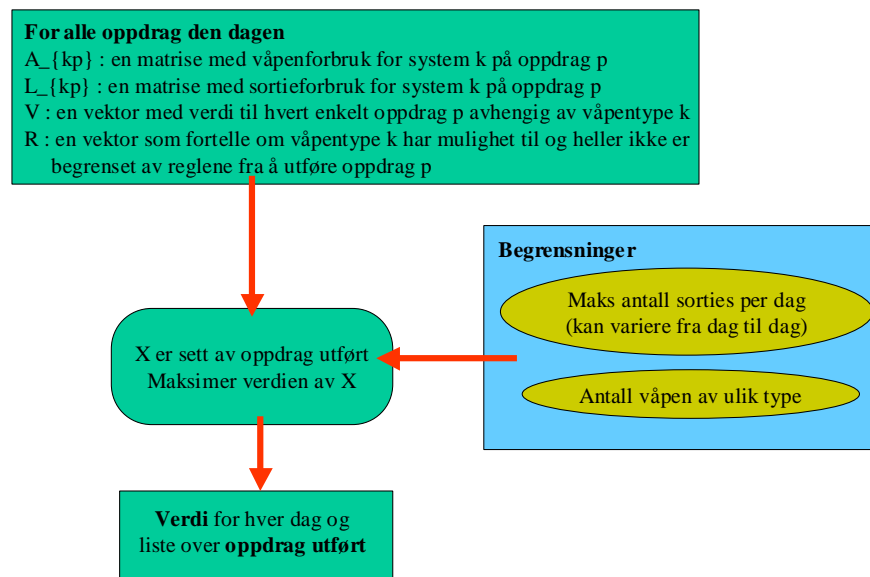
Den kombinasjonen av oppdrag og våpentype som er høyest prioritert i forhold til (5.1) blir valgt først, så fjernes andre muligheter for å løse det samme oppdraget fra listen, før neste kombinasjon av oppdrag og våpentype blir valgt. Hele listen blir gjennomgått på denne måten og hvis det alltid er nok ressurser (våpen og sortier) til å velge den neste uvalgte kombinasjonen av oppdrag og våpentype så er denne løsningen den optimale løsningen.

Sette opp matematiske uttrykk



Figur 5.8 Sette opp matematiske uttrykk for hver dag i et hendelsesforløp. Variablene A , L , V og R er alle parametre nødvendig for å løse optimeringsproblemet

Optimere bruk av våpen



Figur 5.9 Optimere bruk av våpen for hver dag i et hendelsesforløp. To av parametrene i optimeringen er input (merket som begrensninger)

I optimeringen kan det oppstå feil ved at våpnene er brukt på en måte som ikke er den optimale. Svaret som er funnet er en gyldig løsning, så en eventuell feil kan bare bety at strukturen kan utrette enda mer enn det svaret som er funnet. Type og størrelsen på feilen er diskutert i appendiks A. En kort oppsummering kommer i neste avsnitt.

Mulige feil som oppstår i forbindelse med allokering av våpen vil være begrenset til et antall dager som er mindre enn eller lik antall våpentyper. Dette betyr at totalt sett er forventet feil mindre jo flere dager en struktur bruker på å løse oppdrag. Mulige feil i allokering av sorties er for hver dag avhengig av den maksimale verdien et oppdrag kan ha. Er den maksimale feilen et oppdrag kan ha lite i forhold til den totale verdien oppnådd, er feilen liten. Det betyr at forventet feil for en struktur som løser mange oppdrag per dag er mindre enn for en struktur som løser få oppdrag per dag. Feil i forbindelse med allokering av våpen og sorties betyr at med mindre strukturen løser alle oppdragene på veldig få dager vil den løsningen som blir funnet, ligge relativt nært den optimale løsningen og kan i noen tilfeller være lik den optimale løsningen.

Den viktigste årsaken til å være litt nøye med denne optimeringen er at det skal være mulig å skape en rettferdig sammenligning av våpentyper. Det er derfor viktig å ha en metode som klarer å utnytte alle typer våpen optimalt. Det som er viktig å merke seg er derfor at for strukturer med få våpentyper og mange våpen av hver type vil den løsningen som er funnet forventningsmessig ligge nærmere den optimale enn for en struktur som inneholder mange typer våpen, men få av hver type.

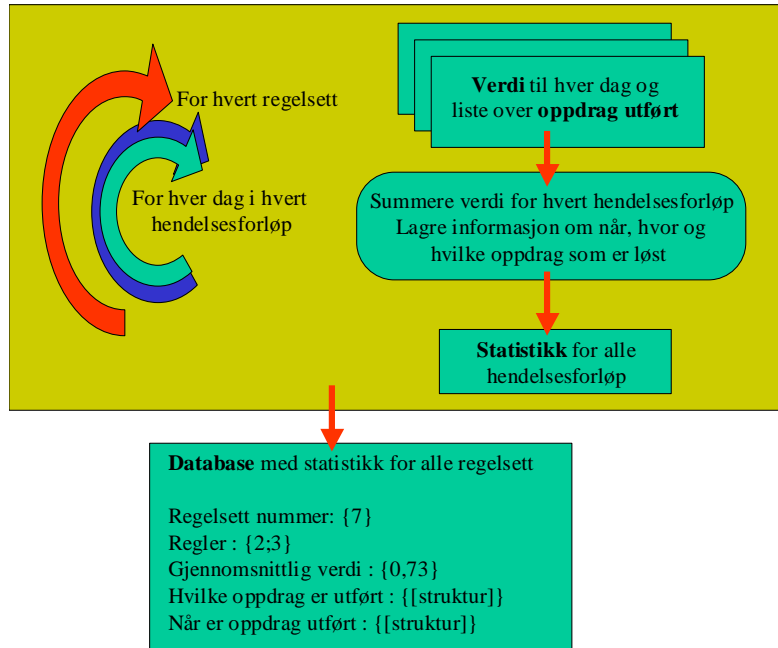
I forhold til realisme er det ikke viktig at løsningen er den mest optimale. At det ikke er mulig å løse problemet eksakt er en utfordring også i den gruppen som skal allokere våpen i et virkelig hovedkvarter. Enkelte hensyn vil være enklere å løse i en weaponeering gruppe, for eksempel at enkelte typer våpen blir brukt opp i løpet av dagen. Stort sett vil det likevel være begrenset hvor godt en sjef får utnyttet sine begrensede ressurser. Prosjektet kjenner ikke til at det finnes en standardisert måte for våpenallokering på dette detaljnivået. Alt i alt er den valgte optimeringsmetoden derfor en akseptabel løsning i forhold til hva metoden skal modellere.

Optimering av bruk av våpen skjer for en dag om gangen og resultater er en liste av oppdrag utført den dagen, hvilke våpen som ble brukt på hvilke oppdrag og en total verdi for disse oppdragene (se Figur 5.9). For et hendelsesforløp blir verdiene til alle dagene summert for å gi en verdi til hvert enkelt hendelsesforløp. Både total verdi og hvor mange oppdrag som ble utført med hver enkelt våpentype for hver eneste dag blir tatt vare på for hvert hendelsesforløp, for til slutt å kunne lage statistikk på hva strukturen utfører med et bestemt regelsett (se Figur 5.10). Det blir funnet statistikk for strukturen for alle regelsett og denne informasjonen blir tatt vare på i en database.

Med statistikk for alle regelsett kan det optimale regelsettet bli funnet ved å spørre etter det regelsettet med størst gjennomsnittlig verdi. Hva strukturen utretter når våpnene blir brukt på denne måten finnes ved å slå opp i databasen. Dette resultatet er output fra modellen (se Figur

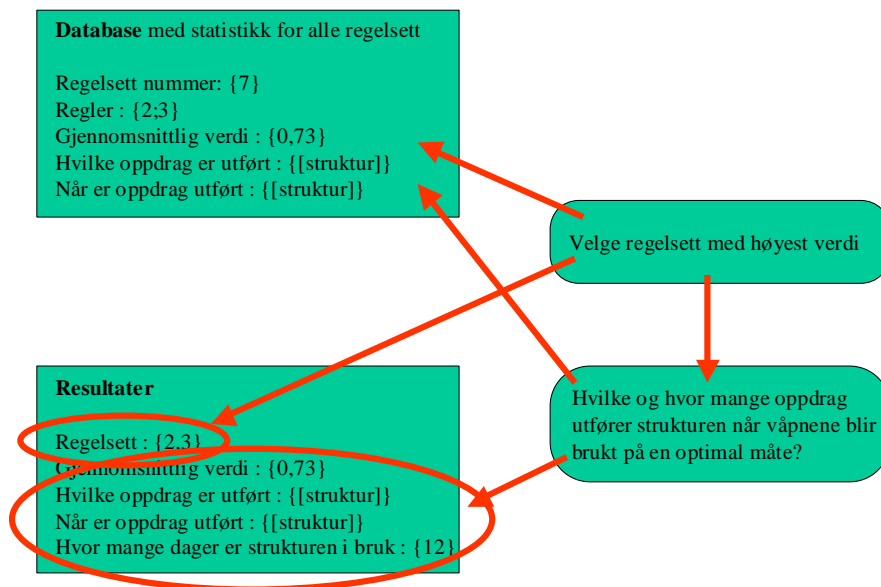
5.11)

Verdi av regelsett i hendelsesforløp



Figur 5.10 Verdi av regelsett i hendelsesforløp. Figuren illustrerer at det blir beregnet en verdi for hver dag i et hendelsesforløp, at denne blir summert for å finne en verdi for hendelsesforløpene og at resultatet er statistikk for alle hendelsesforløpene. Alle mulige regelsett blir prøvd ut og statistikk for hvert regelsett blir tatt vare på i en datastruktur (database)

Velge beste regelsett og resultater



Figur 5.11 Velge beste regelsett og resultater. Beste regelsett blir funnet ved å spørre databasen for hvilket regelsett som gjennomsnittlig har høyest verdi. Statistikk for dette regelsettet blir funnet og dette resultatet er output

5.5 Implementering

Effektiv Ildkraft Modell (EIM) er en blanding av en stokastisk simulering og heltallsprogrammering. Matlab[®] er verktøy som er egnet for begge disse tingene og Matlab[®] ble derfor valgt for implementering av modellen. Nåværende implementering av modellen er på totalt 680 linjer kode (input ikke inkludert).

I den nåværende utgaven av modellen blir resultatene (output) lagret i en tekstfil som er beregnet for import til Microsoft[®] Excel. Også databasen med statistikk for alle regelsett blir lagret for senere referanse. Denne er på et format som en Matlab[®] workspace fil (.mat).

6 KONKLUSJON

I denne rapporten er det utviklet en detaljert metode og modell for å kunne sammenligne hva ulike strukturer kan utrette i et scenario. Dette kalles effektiv ildkraft eller Effective Engagement. Effektiv Ildkraft Modell (EIM) egner seg for implementering på PC.

EIM optimerer bruk av våpen når det er begrensninger på ressurser som våpen og sorties (plattformer) i forhold til antall oppdrag og hvor ulike forhold gjør seg gjeldende med hensyn på vær og lokalisering og i hvor stor grad målet er beskyttet av luftvern. Begrensninger på sensorer og andre ledelsesressurser er ikke tatt hensyn til.

Hva som er hensikten med Forsvarets deltakelse i et scenario kan endre seg, og modellen krever derfor at hensikten defineres i input (i form av en verdifunksjon). Dette gir EIM mulighet for å optimere bruk av våpen tilpasset de prioriteringer som ønskes. Prioriteringer kan ta hensyn til bestemte måltyper, hvordan intensiteten i scenariet utvikler seg, kosteffektivitet, totalt antall mål angrepet, eller totalt antall mål i scenariet.

Resultatene (output) fra EIM er hva våpnene i en systemløsning utretter når brukt på en optimal måte i forhold til hensikten og i forhold til begrensningene i scenariet. Denne metoden gjør EIM til en robust og fleksibel modell som kan brukes til å vurdere en systemløsnings evne til å utføre oppdrag og utnytte kapasiteten optimalt i et scenario hvor systemløsningen bare utgjør en del av en større styrke.

APPENDIKS

A STØRRELSEN PÅ FEIL I OPTIMERING AV VÅPENBRUK PER DAG

I kapittel 5.4 blir det beskrevet en metode for å optimere bruk av våpen som ikke nødvendigvis gir et eksakt riktig svar. Dette appendikset diskuterer hvor stor feil som kan oppstå ved å bruke denne metoden.

Metoden for å optimere bruk av våpen er å prioritere kombinasjoner av oppdrag og våpentype i forhold til det matematiske uttrykket (5.1).¹⁴ De høyest prioriterte kombinasjonene blir valgt først, så fjernes andre muligheter for å løse det samme oppdraget fra listen, før neste kombinasjon av oppdrag og våpentype blir valgt. Hele listen blir gjennomgått på denne måten, og hvis det alltid er nok ressurser (våpen og sorties) til å velge den neste uvalgte kombinasjonen av oppdrag og våpentype, er denne løsningen den optimale løsningen. Da er oppdragslisten tømt uten at det er blitt underskudd på våpen.

Begrensninger i ressurser kan medføre at det er nødvendig å *hoppe over* en kombinasjon av oppdrag og våpentype for å velge en annen våpentype på det samme oppdraget eller for å ta et helt annet oppdrag som krever færre sorties eller våpen. Hvis dette skjer, er det ikke lenger sikkert at den løsningen som blir funnet er den mest optimale, ettersom det kan finnes flere kombinasjoner av oppdrag som bruker opp ressursene.

Eksempel:

I en kombinasjon av oppdrag og våpen er det bedre å utføre oppdrag A enn oppdrag B, B er bedre enn C, og så videre. Når oppdrag A, B og C er valgt så er antallet sorties så lite at det eneste oppdraget som kan velges er oppdrag G selv om dette ikke er det neste i rekken. Resultatet fra metoden er da at A, B, C og G blir utført, men det finnes en annen måte å bruke sortiene på som kan være like god. Ved å la være å utføre oppdrag C kan både oppdrag D og E utføres. Det gir en ny rekke A, B, D og E. Denne løsningen kan være bedre enn den første, men metoden vurderer ikke slike alternative løsninger.

For et konkret tilfelle hvor den optimale løsningen ikke er funnet, er det mulig å beregne en teoretisk maksimal verdi til den beste løsningen. For å kunne gjøre dette er den en forutsening at oppdragene er prioritert etter kostnad og sorties (se kapittel 5.4 og appendiks B.3). For å finne en maksimal verdi må utvelgelsesmetoden av oppdrag tillate prosentvis løsning av oppdrag i de tilfellene det ikke er nok ressurser til å utføre oppdragene.

Fortsettelse av eksempel over:

¹⁴ Oppdragene kan ha forskjellig verdi (med hensyn på kosteffektivitet) avhengig av hvilken våpentype som blir brukt på å løse oppdraget. Derfor prioriteres ikke kun oppdrag, men kombinasjoner av oppdrag og våpentyper.

Oppdrag A, B og C er valgt. Det er kun en sortie igjen, men oppdrag D krever to sorties. Er det tillatt å bidra på oppdraget uten å utføre hele oppdraget kan verdien uttrykkes som

$$V_m = V_{kA} + V_{kB} + V_{kC} + \frac{1}{2}V_{kD}.$$

Verdien til den løsningen som ble funnet er

$$V_1 = V_{kA} + V_{kB} + V_{kC} + V_{kG}$$

mens verdien til den faktisk beste faktiske løsningen er

$$V_2 = V_{kA} + V_{kB} + V_{kD} + V_{kE}.$$

Hvis oppdragene er sortert etter en total kostnad som er avhengig av både sortie- og våpenforbruk slik at

$$\frac{V_{kA}}{\text{ressursforbruk}} \geq \frac{V_{kB}}{\text{ressursforbruk}} \geq \frac{V_{kC}}{\text{ressursforbruk}} \geq \dots.$$

Så må

$$V_m \geq V_2 \geq V_1,$$

fordi verken V_{kG} eller $V_{kD} + V_{kE}$ kan gi en større verdi på bruk av den siste sortien som skal fordeles enn det $(0,5 * V_{kD})$ representerer. V_1 og V_m representerer da en nedre og øvre skranke på den faktiske løsningen V_2 . Den maksimale feilen som kan oppstå i optimeringen er da

$$V_m - V_1 = \frac{1}{2}V_{kD} - V_{kG}$$

En ressurs det er for lite av og som ikke kan erstattes av andre ressurser vil gi en prosentvis løsning av akkurat et oppdrag med mindre mengden ressurser oppdrag er slik at det tilfeldigvis går opp. En ressurs som ikke kan erstattes av andre ressurser vil derfor aldri gi en feil som er større enn verdien til et enkelt oppdrag.

I den nåværende implementasjonen er sorties en ressurs som ikke kan erstattes av andre ressurser. Feilen som kan oppstå på grunn av feilprioritering av sorties er da et

$$\text{Maksimalt avvik pga. sorties} < V_{maks} \tag{A.1}$$

der

V_{maks} er den maksimale verdien et enkelt oppdrag kan ha den dagen.

Kan en ressurs erstattes av andre ressurser, vil uttrykket derimot bli mer avansert. I slike tilfeller kan det være optimalt å prioritere ressursene etter andre kriterier enn hvilken kombinasjon av oppdrag og våpentype som gir størst verdi. Dette gjelder våpen. På de fleste oppdrag vil det være mulig å velge mellom flere ulike våpentyper til å utføre oppdraget.

Eksempel:

Våpentype K er bedre enn våpentype L på alle oppdrag. En dag er det totalt 10 oppdrag, men på et av oppdragene er våpentype K så mye bedre at det kun er type K som kan brukes. Det finnes nok av våpentype K til å utføre kun et oppdrag, men type L finnes det nok av til å utføre alle oppdragene.

I dette tilfellet kan det hende at den største totale verdien oppnås ved å bruke opp den siste resten av våpentype K på det oppdraget som våpentype L ikke kan utføre. Metoden brukt i modellen vil isteden (som alltid) velge den kombinasjonen av oppdrag og våpentype som gir størst verdi i forhold til innsatsen først, og fordi våpentype K er bedre enn våpentype L på alle oppdragene vil det bare være tilfeldig om det oppdraget som bare K kan ta blir prioritert først. Resultatet blir da at det blir løst ni istedenfor ti oppdrag.

Dette eksempelet viser at en optimal bruk i enkelte tilfeller kan være å prioritere de oppdragene som bare kan utføres av en våpentype først.

Når det går tomt for en type våpen slik at det er optimalt å prioritere bruk av våpen etter andre kriterier enn verdi og kostnad, da er det i verste fall slik at alle våpnene av den typen som fordeles den dagen skulle vært fordelt til andre oppdrag. Optimeringsmetoden slik den er implementert velger alltid den kombinasjonen av oppdrag og våpentype som har høyest verdi først. Hvis dette er feil så betyr det at det opprinnelige oppdraget får redusert verdi (en dårligere kombinasjon av oppdrag og våpentype), mens det andre oppdraget (eller oppdragene) får økt verdi (en bedre kombinasjon av oppdrag og våpentype). Verdireduksjonen på det første oppdraget er minimum null, mens verdien på det andre oppdraget ikke kan bli større per våpen enn det første oppdraget. Det betyr at den maksimale feilen er lik verdien av alle de oppdragene den våpentypen det går tomt for, utfører den dagen. Altså

$$\text{Maksimalt avvik pga. våpen} < \sum_{p \in P} X_{kpj} V_{kp} \quad (\text{A.2})$$

der

P er alle oppdrag utført av våpentypen det går tomt for den dagen det går tomt for en våpentype.

Utrykket (A.2) gjelder bare så lenge det er igjen mer enn en våpentype i strukturen. Når det er igjen bare en våpentype, vil det maksimale avviket fra optimal løsning bli redusert til verdien til et enkelt oppdrag slik som for sorties (se uttrykk (A.1)). I det spesielle tilfellet at ingen våpen i strukturen kan utføre de samme oppdragene, vil også feilen begrense seg til uttrykk tilsvarende (A.1).

EIM skiller mellom våpen og sorties på den måten at sorties er en begrensning per dag, mens våpen er en begrensning per scenario. Feilen i uttrykket (A.1) kan derfor oppstå hver dag i scenariet mens feilen i uttrykket (A.2) oppstår kun de dagene det går tomt for en våpentype, og kun så lenge det finnes flere våpentyper som kan utføre det samme oppdraget. (Hvis strukturen inneholder tre våpentyper kan det oppstå en feil av typen (A.2) maksimalt to ganger.)

Optimering av våpenbruk skjer for en dag om gangen. Hvor stor feilen i optimeringen blir er da avhengig av om det går tomt for våpen eller sorties den dagen. Er antall oppdrag eneste begrensningen så er feilen null. Er sorties en begrensning så gjelder uttrykk (A.1), og går det tomt for en eller flere våpentyper gjelder også uttrykk (A.2). For å minimere feilen er det mulig å se på tiltak som reduserer størrelsen på (A.1) og (A.2), eller tiltak som sørger for å redusere

antallet ganger det går tomt for en ressurs. En slik mulighet som er implementert i modellen, er muligheten til å kategorisere våpnene i våpensystem (se appendiks B.3). Utrykket (A.2) vil da skje kun en gang for hvert våpensystem istedenfor for hver våpentype.

EIM er en simuleringsmodell hvor hvert scenario blir repetert som flere ulike hendelsesforløp. Derfor vil faktisk feil være omtrent lik forventet feil. Utrykkene (A.1) og (A.2) gir maksimal feil, ikke forventet feil. Forventet feil vil være avhengig av input til modellen og det er umulig å si noe generelt om forventet feil uten å kjenne verdifunksjon og annen input.

Hensikten med å optimere er å kunne gjøre en rettfærdig sammenligning av våpentypene. Det er verdt å merke seg at den løsningen som blir funnet, alltid er en gyldig løsning, og en mer optimal løsningen må derfor være en løsning som klarer å utrette enda mer. At det er mulig å optimere våpnene enda mer, kan bety at resultatene ikke gir det rette inntrykket. Det er nødvendig å være observant på mulig størrelse på feilen som i spesielle tilfeller kan bli stor. Ved å ta hensyn til feilkildene bør det være mulig å lage resultater hvor optimeringsfeilen ikke påvirker resultatene.

B INPUT

Dette appendikset gir en definisjon av input slik EIM er implementert. Hensikten med appendikset er først og fremst å gi en detaljert oversikt over hvilke parametre EIM tar hensyn til og hvilke definisjoner som må gjøres i input. Er det ønskelig å endre inputfilene for å gjøre egne kjøring er det nødvendig å ha en viss kjennskap til Matlab[®]. Alle inputfilene har en overskriftslinje og et par linjer kode nederst i filen som ikke må endres. For å gjøre dette er det enklest å ta utgangspunkt i allerede eksisterende eksempler på inputfiler og endre disse.

B.1 Struktur

Egen systemløsning spesifiseres av to inputvariable *flysories* og *struktur*. For nåværende implementasjon er dette en egen Matlab[®] tekstfil kalt *input_struktur.m*.

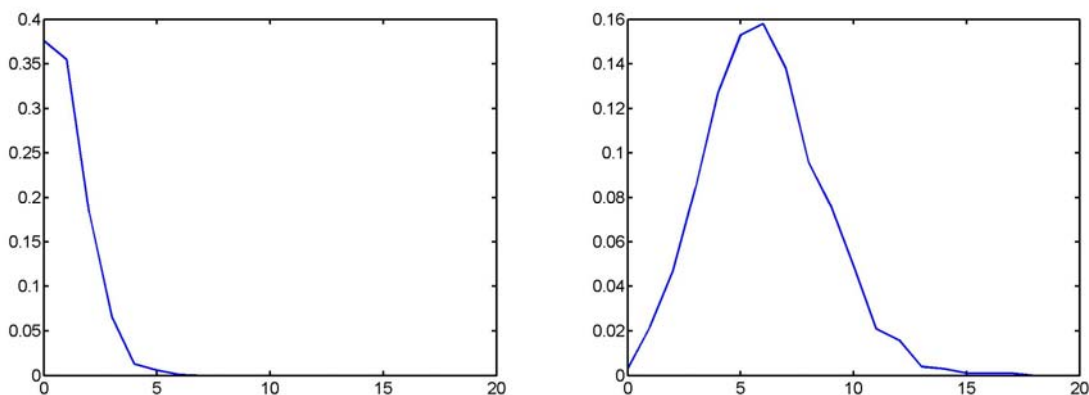
- *Flysories* er en vektor med antall flysorties per dag i scenariet.
- *Struktur* er en matrise med to kolonner hvor den første kolonnen er et tall som spesifiserer type våpen, og den andre kolonnen er antall våpen. Type våpen må være et av våpnene spesifisert i *bibliotek* og tallet som spesifiserer våpentypen er avhengig av rekkefølgen i bibliotek. (Den første våpentypen i bibliotek er nummer en, og så videre.) Bibliotek defineres fra våpenprioritetsmatrisene (se appendiks B.3).

B.2 Scenario

Scenario input (kalt *input_sannsynligheter.m*) definerer en lang rekke variable som lengde på scenariet, hvor mange hendelsesforløp som skal lages og hvilke oppdrag som finnes i scenariet

(et oppdragsbibliotek). For hver type oppdrag må det spesifiseres et forventet antall oppdrag, en forventet fordeling av oppdragene utover i scenariet, en fordeling på hvor stor rekkevidde som er nødvendig for å utføre oppdraget og om denne rekkevidden er avhengig av leveringsplattform. Det følgende vil beskrive hvordan denne inputen blir behandlet i EIM.

Totalt antall oppdrag av en type trekkes fra en poissonfordeling. Forventet antall oppdrag er middelveiden til poissonfordelingen. Eksempler på poissonfordelinger er gitt i Figur B.1. (Poissonfordeling kjennetegnes ved at den er diskret; den gir bare positive tall og null; og variansen er lik gjennomsnittet.)



Figur B.1 Eksempler på poissonfordeling. Til venstre vises et eksempel med tusen trekninger med snitt en. Til høyre vises et eksempel med tusen trekninger med snitt seks

En forventet fordeling av oppdragene utover i scenariet kan enten være spesifisert som en eksempelfordeling, eller det kan være en middelveid på når oppdragene dukker opp. Spesifiseres det en eksempelfordeling, vil det - når hvert enkelt oppdrag dukker opp - trekkes tilfeldig fra den fordelingen eksempelet spesifiserer. (Er det i eksempelet spesifisert 10 oppdrag på dag en og fem, og ingen de andre dagene, så vil alle oppdrag dukke opp på dag en eller fem dagene.) Spesifiseres det en middelveid for når oppdragene dukker opp så vil dagen oppdragene skjer trekkes fra en poissonfordeling med denne middelveiden.

En fordeling på hvor stor rekkevidde som er nødvendig for å utføre oppdraget, kan også defineres ved hjelp av en eksempelfordeling eller en middelveid og varians for oppdragene. Hvis rekkeviddefordeling skal defineres fra et eksempel må det først defineres en vektor hvor kategorier av rekkevidde er definert. Det må så lages et eksempel for hver type oppdrag hvor antall oppdrag i de forskjellige rekkeviddekategoriene er definert. Blir rekkeviddefordeling isteden definert fra en middelveid og varians, blir nødvendig rekkevidde på hvert oppdrag trukket fra en normalfordeling med middelveid og varians som spesifisert.

Om rekkevidden er avhengig av leveringsplattform spesifiseres i en matrise med en verdi for hver type plattform og oppdrag. I nåværende implementasjon er plattformtype kun skilt på land,

sjø og luft.

I kapittel 3.4 sies det at modellen må teste strukturene på de identiske scenarier hvor absolutt alle parametre er helt like. Verdifunksjonen er i stand til å sikre at våpnene blir brukt på en optimal måte i et hendelsesforløp, men det er ingen garanti for at summen av verdifunksjonen i et hendelsesforløp er sammenlignbar med verdien i et annet hendelsesforløp. For å sikre at den totale verdien på oppdragene utført i to ulike hendelsesforløp er matematisk sammenlignbare er det nødvendig at hendelsesforløpene er helt like. Strukturene blir testet ikke bare i ett, men mange hendelsesforløp. For å sikre at totaleverdien for to strukturerer er sammenlignbare er det derfor nødvendig å teste de to strukturene i de identisk samme scenariene. Det er mulig å gjøre dette ved å trekke hendelsesforløpene bare en gang, lagre dem til fil, og så lese hendelsesforløp fra denne filen hver gang en ny struktur skal testes. (Dette er da et avvik fra metoden og algoritmen i kapittel 5.)

B.3 Våpenprioritets matriser (VPM)

En VPM er en prioritert liste av våpen som er ønsket brukt på en bestemt type oppdrag. Type oppdrag bestemmes av måltype, passiv beskyttelse av målet, antall mål og ønsket effekt. Derimot er lokalisering, luftvern og værforhold ikke noe som skiller to typer oppdrag. De siste tre faktorene tas hensyn til i modellen ved å lage variasjoner av oppdragene.

VPM er matriser hvor våpnene er rader og våpenparametrene er kolonner. I den nåværende implementasjonen av modellen leses VPM fra filer av typen Microsoft[®] Excel arbeidsbok. Bokstavkoder i parentes nedenfor refererer til kolonne i filene. Parametre som tas hensyn til i våpenprioritetsmatrisene står nevnt nedenfor.

- **Våpen** er navn på våpenet. Denne blir ikke brukt i EIM (kolonne B).
- **Våpensystem** er en kategorisering av våpnene. Våpensystem kan for eksempel være identisk med navne på våpnene slik at alle våpen blir optimert hver for seg. I mange tilfeller kan det derimot være hensiktsmessige å lage litt grovere grupper av våpen og optimere bruk av våpen mellom disse gruppene. Den viktigste årsaken til dette er at antall regelsett øker kraftig med antall våpensystem (se kapittel 5.4) og antall regelsett er omtrent direkte proporsjonalt med antall regneoperasjoner for å finne resultatet. Et eksempel på dette er at hvis det tar 10 minutter å beregne resultatet for en systemløsning med fire våpentyper, så vil en systemløsning med en ekstra våpentype som har 8 regler, ta 8 ganger så lang tid, dvs. 80 minutter (kolonne A).
- **Antall våpen** er antallet våpen av denne typen som er nødvendig for å utføre oppdraget med den sannsynlighet oppdraget krever (kolonne D).
- **Antall sorties** er antallet sorties for å levere dette antallet våpen fra et kampfly. Hvis andre luftplattformer vurderes kan det isteden representere en annen plattformbegrensning enn kampfly (kolonne C).
- **Plattform** er type plattform som leverer våpenet. Det kan være land, sjø eller luft. Det er også mulig å lage en finere oppdeling hvis ønskelig (kolonne I).

- **Rekkevidde** er våpenets rekkevidde. Rekkevidde blir i modellen brukt til å sammenligne med nødvendig *stand off*-avstand på det aktuelle oppdraget for å avgjøre om våpenet kan brukes (se kapittel 5.3) (kolonne H).
- **Værstatistikk** er en prosentjanse for hvor ofte våpenet kan brukes i forhold til hva som er forventet værforhold i dette scenariet. Sannsynlighet for skydekke, tåke, nedbør og vind kan påvirke dette tallet. Tallet er derfor scenarioavhengig. Værstatistikk for det enkelte våpen blir i modellen sammenlignet med tallet som indikerer vær på det aktuelle oppdraget for å avgjøre om våpenet kan brukes (se kapittel 5.3) (kolonne N).
- **Kostnad** ved å utføre oppdraget med denne våpentypen. *Kostnad* må være avhengig av hvor mange våpen og sortier som blir brukt på oppdraget. Dette kan være et uttrykk for kosteffektivitet i kroner, hvor våpenets kostnad og en kostnad per sortie er lagt sammen. Det kan også være en annen parameter enn oppdragskostnad så lenge denne parameteren er avhengig av mengden ressurser som blir brukt på å løse dette oppdraget og så lenge det er denne parameteren som er brukt for å prioritere våpnene. (Det vil si *kostnad* må være en parameter som har minst verdi for det våpenet som er høyest prioritert på et oppdrag og så større verdier jo lengre nede på VPM det står.) Inne i modellen bestemmer *kostnad* C_{kp} i det matematiske uttrykket (5.1). *Kostnad* kan også bli brukt i verdifunksjonen (se nedenfor).

B.4 Verdifunksjon

Verdifunksjonen spesifiseres i input ved å definere hvordan de tre variable W_{kp} , V_{1kp} og V_{2kp} bestemmes. Variablene bestemmes for hver kombinasjon av oppdrag p og våpen k . De tre variablene er alle på ulike måte med å uttrykke en vektning av oppdragene eller en verdi ved å utføre et oppdrag. Den matematiske sammenhengen mellom variablene inne i modellen er gitt ved uttrykket

$$V_{kp} = \frac{V_{1kp} * V_{2kp}}{W_{kp}} \quad (\text{B.1})$$

som gir V_{kp} i det matematiske uttrykket (5.1), og uttrykket

$$V_{total} = \left(\frac{\sum_{kp \in A} W_{kp} V_{1kp}}{\sum_{kp \in A} W_{kp}} \right) * \left(\frac{\sum_{kp \in A} V_{2kp}}{\sum_{kp \in B} V_{2kp}} \right) \quad (\text{B.2})$$

hvor

A er mengden oppdrag utført og

B er den totale mengden oppdrag i scenariet,

som bestemmer verdien til regelsettet i et hendelsesforløp (V_{total}).

I uttrykket (B.2) kan den første parentes uttrykke en form for effektivitetsmål. En forutsetning for dette er at V_{1kp} har verdien 1 hvis oppdraget er utført på optimal måte og et tall mellom null og en hvis det er utført på en dårligere måte. I dette leddet uttrykker W_{kp} en vektning av oppdragene. Den andre parentes i uttrykk (B.2) er antall oppdrag løst delt på antall oppdrag totalt. I dette uttrykket fungerer verdien V_{2kp} som en vektning av oppdragene.

Tilgjengelig informasjon når W_{kp} , V_{1kp} og V_{2kp} skal bestemmes er måltype, *kostnad* (som definert i appendiks B.3), hvilken rekkevidde som kreves på oppdraget, antall oppdrag av samme type totalt, antall oppdrag av samme type den dagen og hvilken dag det er i scenariet. All denne informasjonen er tilgjengelig for alle mulige kombinasjoner av oppdrag, p, og typer våpen som kan brukes på oppdraget, k. For å bestemme W_{kp} , V_{1kp} og V_{2kp} kan det lages direkte matematiske sammenhenger til den tilgjengelige informasjonen, men det er også mulig å definere at variablene skal ha spesifikke verdier når gitte vilkår er oppfylt.

Eksempel:

Ønsker å gjøre V_{1kp} avhengig av kostnad for å la det første leddet i (B.2) uttrykke kosteffektivitet. Setter V_{1kp} til å være direkte matematisk avhengig av kostnad med uttrykket:

$$V_{1kp} = \frac{C_{kp}}{\min(C_{kp})} \quad (\text{B.3})$$

hvor

C_{kp} er kostnad ved å utføre oppdrag p med våpen k og
 $\min(C_{kp})$ er minste mulige kostnad for å utføre oppdrag p.

Eksempel:

Måltype *panservogner* inneholder to mål mens alle andre måltyper inneholder bare et mål. Både W_{kp} og V_{2kp} brukes som vektorer i (B.2) og ønsker at oppdragene skal vektet etter størrelsen på oppdragene. Lager da en forutsetning om at hvis måltypen er *panservogner* så er

$$V_{2kp} = W_{kp} = 2 \quad (\text{B.4})$$

hvis ikke så er

$$V_{2kp} = W_{kp} = 1 \quad (\text{B.5})$$

I nåværende implementasjon av modellen er verdifunksjonen en Matlab[®] tekstfil kalt *input_verdi.m*.

Litteratur

- (1) ANDRESEN Nils Ulrik, HENNUM Alf Christian, HOFF Erlend (2005): Strukturutvikling og analyse av LPV-systemløsninger, FFI/RAPPORT-2005/00311, Begrenset
- (2) DE FARIAS D. P., VAN ROY B. (2003): The Linear Programming Approach to Approximate Dynamic Programming, *Operations Research* **51**, 6, 850-865.
- (3) JAISWAL N. K. (1997): Military Operations Research – Quantitative decision making (Eds HILLIER Frerick S.), Kluwer Academic Publishers, Norwell, Massachusetts, USA, side 145 til 150.
- (4) WINSTON Wayne L. (1994): Operations Research - Applications and Algorithms Third Edition (Eds HINRICHS Curt), Duxbury Press, Wadsworth Publishing Company, Belmont, California, USA, side 533 til 538.
- (5) MISHRA S., BATTI R. og SZCZERBA R. J. (2004): A Ruled Based Approach for Aircraft Dipatching to Emerging Targets, *Military Operations Research* **9**, 3, 17-30.
- (6) CAFFALL Dale Scott, MICHAEL James Bret (2003): Developing Highly Predictable System Behavior in Real-Time Battle-Management Software, NAVAL POSTGRADUATE SCHOOL, Monterey California USA, NPS-CS-03-006, Approved for public release, distribution unlimited
- (7) MALERUD Stein, HENNUM Alf Christian, KARLSRUD Øivind (2004): Systemløsninger for langtrekkende presisjonsstyrte våpen - Dokumentasjon av systemutviklingsprosessen, FFI/RAPPORT-2004/03951, Begrenset