

FFI RAPPORT

Metoder for effektberegning av smart artilleriammunisjon

HALSØR Marius, KVIFTE Lars

FFI/RAPPORT-2003/00084

FFIBM/798/139

Godkjent
Kjeller 23. april 2003

Stein Grinaker
Forskningsjef

**Metoder for effektberegning av smart
artilleriammunisjon**

HALSØR Marius, KVIFTE Lars

FFI/RAPPORT-2003/00084

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

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-2003/00084 1a) PROJECT REFERENCE FFIBM/798/139	2) SECURITY CLASSIFICATION UNCLASSIFIED 2a) DECLASSIFICATION/DOWNGRADING SCHEDULE -	3) NUMBER OF PAGES 54										
4) TITLE Metoder for effektberegning av smart artilleriammunisjon Methods for determining the effect of smart artillery ammunition												
5) NAMES OF AUTHOR(S) IN FULL (surname first) HALSØR Marius, KVIFTE Lars												
6) DISTRIBUTION STATEMENT Approved for public release. Distribution unlimited. (Offentlig tilgjengelig)												
7) INDEXING TERMS IN ENGLISH: <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; border: none;"> a) <u>Smart ammunition</u></td> <td style="width: 50%; border: none;"> IN NORWEGIAN: a) <u>Smart ammunisjon</u></td> </tr> <tr> <td style="border: none;"> b) <u>Artillery</u></td> <td style="border: none;"> b) <u>Artilleri</u></td> </tr> <tr> <td style="border: none;"> c) <u>Simulation</u></td> <td style="border: none;"> c) <u>Simulering</u></td> </tr> <tr> <td style="border: none;"> d) <u>Optimisation</u></td> <td style="border: none;"> d) <u>Optimalisering</u></td> </tr> <tr> <td style="border: none;"> e) <u>Random numbers</u></td> <td style="border: none;"> e) <u>Tilfeldige tall</u></td> </tr> </table>			a) <u>Smart ammunition</u>	IN NORWEGIAN: a) <u>Smart ammunisjon</u>	b) <u>Artillery</u>	b) <u>Artilleri</u>	c) <u>Simulation</u>	c) <u>Simulering</u>	d) <u>Optimisation</u>	d) <u>Optimalisering</u>	e) <u>Random numbers</u>	e) <u>Tilfeldige tall</u>
a) <u>Smart ammunition</u>	IN NORWEGIAN: a) <u>Smart ammunisjon</u>											
b) <u>Artillery</u>	b) <u>Artilleri</u>											
c) <u>Simulation</u>	c) <u>Simulering</u>											
d) <u>Optimisation</u>	d) <u>Optimalisering</u>											
e) <u>Random numbers</u>	e) <u>Tilfeldige tall</u>											
THESAURUS REFERENCE: 8) ABSTRACT In order to evaluate the effect of smart artillery ammunition, three separate methods have been developed. They are all described in this report. The methods have different areas of applications. For realistic, complex situations, the method involving stochastic simulation is best suited. When performing parameter studies, the method involving exact mathematical formulas is desired. The deterministic model was meant to be a fast alternative to the stochastic simulation, but proved to be slower. However, it is described here since some of the principles examined in the work with this model might prove useful later In addition to the development of these methods, attempts have been made to optimise the firing pattern in given scenarios. Standard methods for optimisation have been applied, and two simple routines for optimisation have been developed and tested. These methods, as well as the results from the tests, are documented in this report.												
9) DATE 23. April 2003	AUTHORIZED BY This page only Stein Grinaker	POSITION Director of Research										

ISBN-82-464-0732-5

UNCLASSIFIED

**SECURITY CLASSIFICATION OF THIS PAGE
(when data entered)**

INNHOOLD

	Side	
1	INNLEDNING	7
1.1	Bakgrunn	7
1.1.1	Strix	7
1.1.2	SMArt	7
1.1.3	Bonus	8
1.2	Formål	8
1.3	Oversikt over innhold	8
2	INNLEDENDE ANTAGELSER OG DEFINISJONER	9
2.1	Terminologi	9
2.2	Definisjon av symboler	11
2.3	Forenklinger	12
3	SMARTE ARTILLERIGRANATERS EFFEKT MOT PANSREDE MÅL - ANALYTISKE BEREGNINGER	13
3.1	Effekt av én granat	13
3.1.1	Skannende søker	13
3.1.2	Stirrende søker	15
3.1.3	Sammenlikning av skannende og stirrende søkere	17
3.2	Effekt ved bruk av flere granater	17
3.2.1	Betydning av overlappende søkeområder	17
3.2.2	Beregning for granater med sammenfallende søkeområder	20
3.3	Oppsummering	22
4	SMARTE ARTILLERIGRANATERS EFFEKT MOT PANSREDE MÅL - SIMULERINGER	24
4.1	Deterministisk modell	24
4.1.1	Deterministisk nedslagsfelt	25
4.1.2	Deterministisk effektberegning	27
4.2	Stokastisk modell	28
4.2.1	Målbeskrivelse	28
4.2.2	Granatbeskrivelse	29
4.2.3	Beregning av granatenes effekt	30
4.2.4	Resultat	30
5	OPTIMALISERING AV SMARTE ARTILLERIGRANATERS SIKTEPUNKT	32
5.1	MatLab-rutiner for optimalisering	32
5.2	Optimalisering ved "brute force"	33
5.3	Optimalisering gitt symmetri for siktemønster	34

5.4	Eksempel på optimalisering	35
5.5	Konklusjon	36
6	KONKLUSJON	37
6.1	Beregningsmetoder	37
6.2	Optimalisering	37
6.3	Veien videre	37
APPENDIKS		
A	BEVIS AV ANALYTISKE UTTRYKK	39
A.1	Effekt av én granat	39
A.2	Sammenfallende søkeområder	42
B	TILFELDIGE OG KVASITILFELDIGE TALL	47
B.1	Tilfeldige og "pseudotilfeldige" tall	47
B.2	Kvasitilfeldige tall	47
B.3	Metodene fra kapittel 4.1	49
B.4	Sammenlikning av metoder	50
	Litteratur	53
	Fordelingsliste	54

Metoder for effektberegning av smart artilleriammunisjon

1 INNLEDNING

1.1 Bakgrunn

FFI-prosjekt 798, panserbekjempelse 2000+, har som oppgave å se på hvordan vi bør drive panserstrid i fremtiden. Dette prosjektet består av fem delprosjekter. Delprosjekt 3 ser på såkalt ”smart” ammunisjon for indirekteskytende plattformer, og studerer hvordan slik ammunisjon kan benyttes mest mulig effektivt. Med ”smart” ammunisjon mener vi ammunisjon som selv søker etter mål i et bestemt område, og angriper eventuelle mål som oppdages i dette området. Strix, SMARt og Bonus er eksempler på ”smart” ammunisjon.

For å finne ut hvordan smart ammunisjon kan benyttes mest mulig effektivt, trenger vi først en metode for å bestemme effekten dersom den brukes på en bestemt måte. Vi har derfor utviklet noen metoder for beregning av effekten av smart ammunisjon. I første omgang har vi konsentrert oss om scenarier der det kun er vi som beskytter fienden, mens han ikke skyter tilbake. Spørsmålet om hvordan ammunisjonen skal benyttes reduseres da til et spørsmål om hvordan siktepunktene skal plasseres. Senere i prosjektet vil vi også studere situasjoner der egne styrker blir kontrabeskutt.

I tillegg til å utvikle metoder for beregning av effekt, har vi også sett på hvordan vi kan optimalisere effekten – med andre ord, hvordan vi kan finne det operasjonsmønsteret (herunder siktemønster) som gir høyest effekt. Det er dette spørsmålet delprosjektet søker å besvare.

1.1.1 Strix

Strix er en granat for 120 mm bombekaster. Den er produsert av Bofors/Saab. Rekkevidden er ca. 7.5 km. Den benytter en IR-søker, av den typen som i dette dokumentet omtales som stirrende søker. Søkeområdet er en ellipse med halvaksler på 65 og 70 meter. Stridshodet er en hulladning, med estimert penetrasjon i RHA på 630 mm. Granaten er i produksjon.

1.1.2 SMARt

SMARt er en granat for 155 mm rørtilleri. Granaten produseres av Diehl. Granaten inneholder to substridsdeler som kastes ut fra granaten. Disse starter å søke etter mål i en høyde på 150 meter. Søkeområdet til hver substridsdel er en sirkel med radius ca. 100 meter, og søkeområdet til de to substridsdelene er delvis overlappende. Det benyttes både IR og mm-bølge radar. stridshodet er en EFP (Explosively Formed Projectile) av tantal, som har vesentlig mindre penetrasjonsevne enn en hulladning. Penetrasjonsevnen antas å være omlag 100 -150 mm RHA. Dersom den først perforerer panseret, har den større effekt mot målet enn en hulladning. Granaten er i produksjon.

1.1.3 Bonus

Bonus er også en granat for 155 rørartilleri som inneholder to substridsdeler. Granaten produseres av Bofors og Giat. Substridsdelene starter søket i en høyde på 175 meter, og har hver et søkeområde med en radius på ca. 90 meter. Søkeområdet til substridsdelene er delvis overlappende. Substridsdelen inneholder kun en IR-søker. Stridshodet er en EFP av tantal, med antatt penetrasjonsevne som for SMARt (ca. 100 - 150 mm RHA). Granaten er i produksjon.

1.2 Formål

Formålet med dette dokumentet er å gi en god beskrivelse av de metodene som er utviklet for å beregne effekten av smart ammunisjon. Vi har eksperimentert en del i utviklingsprosessen, og det ville blitt meget omfattende om alle betraktninger og mellomregninger skulle dokumenteres her. Vi prøver derfor primært å gi en beskrivelse av de metodene som finnes, og gir kun korte begrunnelser for hvorfor ting er gjort som de er. Appendiks A og B beskriver bakgrunnen for metodene i mer detalj.

1.3 Oversikt over innhold

Det er så langt utviklet tre metoder for beregning av effekt. Den ene metoden baserer seg på matematiske formler, og søker å beregne sannsynligheten for en bestemt effekt eksakt. Dette fører til omfattende regning, og metoden er bare i stand til å beskrive en situasjon der man kjenner målenes eksakte posisjon og vet nøyaktig hvor hver enkelt granat lander.

De to andre metodene baserer seg på å finne mulige treffpunkt for granatene, basert på spredning og bomavstand, og så integrere opp effekten over disse treffpunktene. Den ene modellen genererer et strukturert, deterministisk grid med treffpunkter, og beregner effekten eksakt for alle treffpunktene. Denne modellen krever at man kjenner målenes posisjon eksakt. Den andre modellen trekker tilfeldige tall for å bestemme granatenes treffpunkt, og videre trekker den tilfeldige tall for å bestemme effekten gitt et sett med treffpunkt for granatene.

Når det gjelder optimalisering, er det i tillegg til eksisterende MatLab-rutiner benyttet to enkle, egenutviklede metoder.

2 INNLEDENDE ANTAGELSER OG DEFINISJONER

I dette dokumentet blir det brukt en rekke symboler, forkortelser, fagord og spesielle modellantagelser som trenger en forklaring. For å lette lesingen av dokumentet, er de fleste av disse forklaringene samlet i de neste tre delkapitlene.

2.1 Terminologi

En rekke begreper benyttes når vi skal gjøre beregninger for artillerigranaters effekt. Hva vi i dette dokumentet legger i disse er forklart under:

Substridsdel: En granat/rakett kan inneholde flere substridsdeler. En substridsdel er et autonomt våpen som inneholder søker og et stridshode. Én substridsdel kan slå ut ett mål. I dette dokumentet antar vi at granatene bare har én substridsdel. Ordet ”granat” blir derfor i dokumentet ensbetydende med ”substridsdel”. I praksis betyr dette at hver substridsdel har et eget siktepunkt, hvilket gjør det mulig å oppnå en høyere effekt enn dersom flere substridsdeler ligger i samme granat. Neste versjon av programmet vil takle granater med flere substridsdeler.

Smart granat er en granat som autonomt kan gjenkjenne, velge og treffe en måltype. Egentlig er det en granat som inneholder én eller flere smarte substridsdeler, men ettersom vi her antar at hver granat inneholder kun én substridsdel, blir dette det samme.

Søkeområdet er betegnelsen på det totale området substridsdelen ”ser” i løpet av hele søkefasen, dvs hele det området der den kan oppdage mål.

Søker: En eller flere detektorer som bygger opp et ”bilde” over et valgt område, kalt søkeområdet. Søkeren inneholder i tillegg til detektorene også programvare som gjør den i stand til å gjenkjenne eventuelle mål som befinner seg i søkeområdet, samt bestemme ønsket treffpunkt for stridshodet .

Stirrende søker: Søkeren har til en hver tid et komplett bilde av hele søkeområdet. Hele bildet analyseres og søkeren velger det målet som med størst sannsynlighet er et godt mål. Bildet av søkeområdet oppdateres jevnlig. Begrepet ”stirrende søker” brukes i andre sammenhenger i en noe annen betydning. Det som er beskrevet her er betydningen av begrepet i dette dokumentet.

Skannende søker: Søkeren ser bare en liten del av søkeområdet ad gangen. På et systematisk vis leter den gjennom søkeområdet og angriper som regel det første målet som den kjenner igjen.

OP: Forkortelsen står for ObservasjonsPost. Dette er en soldat eller gruppe som befinner seg på fiendtlig område og rapporterer tilbake hvor det befinner seg mål som kan angripes. I tillegg til målenes posisjon angir den eventuell hastighet og retning.

Mål: Et mål er noe som kan slås ut av en granat, for eksempel en stridsvogn eller stormpanservogn. Dokumentet gjør få antagelser om hvilke måltyper granatene kan angripe.

Dersom sannsynligheten for å oppdage målet og sannsynligheten for å slå ut målet er oppgitt, så kan metodene i dokumentet benyttes.

Målområdet er et område som inneholder alle målene som betraktes. I en av simuleringsmodellene angir OP'en ikke hvert enkelt mål, men et rektangel som målene befinner seg innenfor. Dersom målene beveger seg, antas det at alle har samme hastighet, slik at dette kan sees på som at hele målområdet flytter på seg.

Utslått mål er et mål som er truffet av en granat og som er satt ut av spill. I virkeligheten kan selvsagt mål bli "delvis utslått", men vi antar i våre modeller at målene enten blir helt utslått eller ikke utslått i det hele tatt.

Reelt mål er et mål som ikke er utslått.

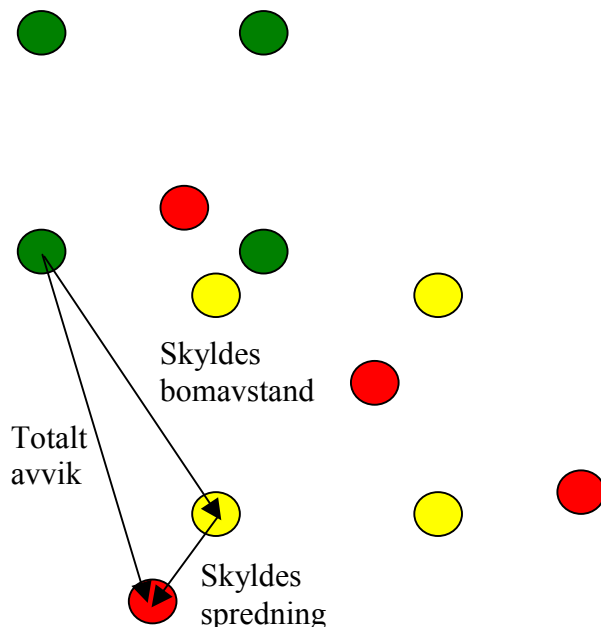
Detektert mål er et mål innenfor søkeområdet som er klassifisert som mål av søkeren.

Valgt mål: Detektert mål som granaten beslutter å angripe. En granat kan detektere flere mål innenfor søkeområdet. Kun ett av disse blir valgt for angrep.

Effekt: Forventet antall utslåtte mål. Hvor mange mål som slås ut, vil være en stokastisk størrelse. Når vi i dokumentet optimaliserer effekten, mener vi at vi gjør det forventede antall utslåtte mål størst mulig.

Avvik: Avstanden en granat lander unna sitt siktepunkt. Avviket til en granat består av to deler; én del som er felles for alle granatene, og én del som er individuell for hver granat.

Bomavstand: Granatenes felles avvik trekkes fra en normalfordeling. Standardavviket i denne fordelingen kalles bomavstand.



Figur 2-1: De grønne sirklene er siktepunktene. På grunn av bomavstanden får alle siktepunktene et like stort bidrag til avviket, og forskyves til de gule sirklene. På grunn av spredningen får hvert siktepunkt et individuelt bidrag til avviket, slik at granatene faktisk lander på de røde sirklene.

Spredning: Granatenes individuelle avvik trekkes også fra en normalfordeling. Standardavviket i denne fordelingen kalles spredning. Det gjøres én trekning pr. granat.

Analytiske beregninger innebærer at resultatene baserer seg på matematiske formler.

Monte-Carlo simuleringer innebærer at man kjører et scenario gjentatte ganger med samme inngangsverdier. Utfallet for hver ”kjøring” avgjøres ved at man trekker tilfeldige tall fra en bestemt sannsynlighetsfordeling. Man får da oversikt over hva som er det typiske utfallet (gjennomsnittsverdien) og hvor stor variasjon det er mellom de forskjellige utfallene (varians).

2.2 Definisjon av symboler

I våre beregninger er følgende symboler benyttet:

P_d : Sannsynligheten for at en granat detekterer et reelt mål som befinner seg i søkeområdet, klassifiserer dette som et mål og velger det ut for angrep. Brukes kun for skannende søkere.

P_o : Sannsynligheten for at en granat detekterer et allerede utslått mål som befinner seg i søkeområdet, klassifiserer dette som mål og velger det ut for angrep. Brukes kun for skannende søkere. Denne størrelsen er knyttet til ammunisjonens såkalte ”overkill-mekanisme”. Dersom $P_o = P_d$, har ammunisjonen ingen overkill-mekanisme. Dersom $P_o = 0$, vil ammunisjonen alltid se om et mål er utslått eller ikke, og utslåtte mål vil aldri bli angrepet.

P_k : Sannsynligheten for at det målet som ammunisjonen angriper faktisk blir slått ut. Vi opererer ikke i dette dokumentet med delvis utslåtte mål; Enten er det utslått eller ikke utslått.

P_D : Sannsynligheten for at søkeren (stirrende eller skannende) detekterer et reelt mål (sier ikke noe om hvorvidt den velger det ut for angrep).

P_u : Sannsynligheten for at søkeren (stirrende eller skannende) detekterer et allerede utslått mål (sier ikke noe om hvorvidt den velger det ut for angrep).

P_e : Sannsynligheten for at et utslått mål som er detektert klassifiseres som utslått, og dermed ikke velges som mål.

E_i : Effekten av en granat som har M mål i sitt søkeområde, hvorav i allerede er utslått.

$P_G(N)$: Dersom det skytes G granater med sammenfallende søkeområde mot et målområde som inneholder M reelle og ingen utslåtte mål, er dette sannsynligheten for at nøyaktig N av disse målene blir slått ut.

W_u : ”Vekt” som beskriver et måls sannsynlighet for å bli valgt ut som mål dersom flere mål i søkeområdet detekteres. Sannsynligheten for at et bestemt mål blir valgt, er målets vekt dividert på summen av vektene til alle mål som er oppdaget. Vi har antatt at vekten til et reelt mål er 1, og til et utslått mål W_u . Benyttes kun for stirrende søkere.

Enkelte av symbolene over kan knyttes sammen. Hvis vi vurderer en skannende søker som alltid velger ut reelle mål som detekteres, og aldri allerede utslåtte mål som gjenkjennes som utslått, gjelder:

$$P_d = P_D \quad (2.1)$$

og

$$P_o = P_u(1 - P_e) \quad (2.2)$$

2.3 Forenklinger

Vi studerer en målformasjon som skal beskyttes med smart artilleriammunisjon. Ammunisjonen skytes frem til målområdet, enten styrt (banekorrigert) eller ustyrt. Når den kommer frem til målområdet, søker den selv etter mål innenfor et søkeområde. I det følgende beskrives hvilke forenklinger som er gjort i simuleringsmodellene, samt hvilke egenskaper som er av spesiell betydning for simuleringsresultatene.

- Hver substridsdel kan kun angripe ett mål, selv om det har flere mål i sitt søkeområde.
- Granatene har en siktepunktsfordeling. Dette beskriver siktepunktet til hver enkelt granat. Det er fullt mulig å sikte på samme punkt med flere granater.
- Hvert enkelt mål er gitt en posisjon. Normalt er målposisjonen kjent. Det finnes imidlertid opsjoner for håndtering av usikker målposisjon og mål i bevegelse i den stokastiske modellen.
- Granatene har én bomavstand i x- og én i y-retning.
- Granatene har én spredning i x- og én i y-retning.
- I simuleringsprogrammene har granatene elliptiske søkeområder.
- Enkelte ammunisjonstyper er mer eller mindre i stand til å skille ut allerede utslåtte mål. Dette kalles for en "overkill-mekanisme". Det tas ikke hensyn til dette i det deterministiske programmet, noe som blir forklart nærmere senere. Det finnes en opsjon for å inkludere en "overkill-mekanisme" i den stokastiske modellen.
- Det er mulig for to granater å angripe det samme målet samtidig. "Samtidig" betyr her at dersom den første granaten slår ut målet, vil ikke den andre granaten rekke å registrere dette. En slik situasjon er ikke modellert, siden de stokastiske modellene benytter en bestemt rekkefølge på granatene. Siden situasjonen vil opptre meget sjelden og kun har betydning for granater med overkillmekanisme, har vi ikke lagt arbeid i å inkludere dette.
- I virkeligheten kan et mål bli "delvis ødelagt", mens det i modellene kun enten blir ødelagt eller ikke ødelagt.
- Det er ønskelig å oppnå et høyest mulig forventet antall utslåtte mål. Det eneste som ikke ligger fast når beskytningen starter, er siktepunktsfordelingen. Denne søkes derfor optimalisert med tanke på forventet antall utslåtte mål. Man kan i enkelte situasjoner ønske å optimalisere sannsynligheten for å oppnå en bestemt effekt heller enn forventet effekt. Dette er det foreløpig ikke mulighet for, men det kreves kun mindre programendringer for å få det til.

3 SMARTE ARTILLERIGRANATERS EFFEKT MOT PANSREDE MÅL - ANALYTISKE BEREGNINGER

I dette kapitlet tar vi for oss hvordan man kan beregne effekten av smarte granater eksakt, og utleder formler for denne effekten. Dette er et teoretisk tungt kapittel, og kan hoppes over for lesere som kun er ute etter hvordan man kan finne effekten av smart ammunisjon. Dette beskrives i kapittel 4, som omhandler simuleringsmodellene.

Analytiske beregninger har visse ulemper. Det blir raskt meget komplisert regning, og de mest kompliserte situasjonene vil stort sett alltid være umulige å løse eksakt. Det blir derfor gjerne til at man kun ser på enkle problemstillinger, og løser de mer komplekse ved hjelp av stokastiske simuleringer. Analytiske uttrykk kan gi svaret på en problemstilling relativt raskt, men den største fordelene er at det blir enklere å få en oversikt over hvilken innflytelse de forskjellige parametrene har på resultatet. Med analytiske beregninger kan man også få et eksakt svar – dersom alle inputparametere er korrekte. Dessuten er det gøy!

For at ikke situasjonen skal bli fullstendig u håndterlig, vil vi i dette kapitlet ikke ta for oss granatens spredning og bomavstand, men konsentrere oss om hva som skjer gitt et bestemt mønster av granater og mål i nedslagsfeltet.

3.1 Effekt av én granat

I dette avsnittet ser vi på situasjonen der det skytes med én granat mot et målområde som inneholder M mål, hvorav N allerede er utslått. Spørsmålet blir da hvilken effekt denne granaten har, det vil si sannsynligheten for å slå ut et mål som ikke allerede er utslått. Denne sannsynligheten er en funksjon av både M og N , men av bekvemmelighetshensyn betegner vi den E_N i dette dokumentet, og lar M -avhengigheten være implisitt. Situasjonen er altså som følger:

- I granatens søkeområde befinner det seg M mål. N av disse målene er utslått
- Søkeren har en viss evne til å skille mellom utslåtte og reelle mål

Det finnes to forskjellige søkertyper, som her behandles separat. De to typene vil her bli betegnet som henholdsvis skannende og stirrende søkere. De har en noe forskjellig virkemåte, som gjør at likningene som gjelder for den ene, ikke nødvendigvis gjelder for den andre. I avsnitt 3.1.3 ser vi på hvilke forutsetninger som skal gjelde for at vi skal kunne benytte samme formelverk for begge søkertypene.

3.1.1 Skannende søker

I første omgang betrakter vi ”skannende” søkere, som søker gjennom målområdet og angriper det første målet som gjenkjennes som et mål og ikke forkastes fordi det er utslått. Det er denne typen søkere som finnes i artillerigranatene ”SMARt” og ”Bonus”. Rekkefølgen som

målene skannes i antas å være tilfeldig (i virkeligheten skannes selvsagt området i en bestemt rekkefølge, ofte utenfra og inn mot midten av søkeområdet). Se forøvrig kapittel 2 for definisjoner av de involverte størrelsene. Det antas at alle mål som ikke er utslått har samme sannsynlighet for å bli detektert.

Noen måler ”overkillmekanisme” ved granatens sannsynlighet for å skille et utslått mål fra et reelt mål. Denne størrelsen har vi kalt P_e . Sannsynligheten for å detektere (men ikke velge ut) et reelt og utslått mål er henholdsvis P_D og P_u . Et reelt mål vil alltid bli valgt ut (dette er selvsagt ikke tilfelle, men effekten av falske alarmer kan tas hensyn til i modellene ved å senke P_d og P_D). Sannsynligheten for å detektere og velge et ikke utslått mål er da $P_D \cdot I = P_d$. Sannsynligheten for å detektere og velge et utslått mål blir $P_u \cdot (I - P_e) = P_o$. Dette er altså sammenhengen mellom en granats ”overkillmekanisme-effekt”, P_e , og den praktisk anvendelige størrelsen P_o . Dersom $P_u = P_D$, vil $P_o \leq P_d$. Dette vil imidlertid ikke alltid være tilfelle, og P_o kan derfor i teorien være større enn P_d . De følgende betraktningene gjelder for øvrig for generelle P_o og P_d .

Dersom det ikke er noen utslåtte mål i søkeområdet, er det M reelle mål. Sannsynligheten for å slå ut et reelt mål, altså effekten til granaten, blir da

$$E_0 = \left(1 - (1 - P_d)^M\right) \cdot P_k \quad (3.1)$$

Med ett utslått og $M-1$ reelle mål spiller rekkefølgen på skanningen en rolle. Vi antar tilfeldige skanne-rekkefølger, det vil si at sannsynligheten for at det utslåtte målet er det første, andre, tredje, fjerde osv. målet som skannes over, er like store, nemlig en M -del. Dersom det utslåtte målet er mål nr. i , blir sannsynligheten for å slå ut et reelt mål lik

$$E_{1,i} = \left[1 - (1 - P_d)^{(i-1)}\right] \cdot P_k + (1 - P_d)^{(i-1)} \cdot (1 - P_o) \cdot \left[1 - (1 - P_d)^{(M-i)}\right] \cdot P_k \quad (3.2)$$

Totalt blir derfor sannsynligheten for å slå ut et reelt mål

$$E_1 = \frac{1}{M} \cdot \sum_{i=1}^M \left[\left(1 - (1 - P_d)^{(i-1)}\right) + (1 - P_d)^{(i-1)} \cdot (1 - P_o) \cdot \left(1 - (1 - P_d)^{(M-i)}\right) \right] \cdot P_k \quad (3.3)$$

Dette er en geometrisk rekke, som kan regnes ut eksakt. Man får da

$$E_1 = P_k \cdot \left[1 - (1 - P_o) \cdot (1 - P_d)^{(M-1)} - \frac{P_o}{M \cdot P_d} \cdot \left(1 - (1 - P_d)^M\right) \right] \quad (3.4)$$

Det er enkelt å sjekke at vi får riktig resultat for grensetilfellene $P_o=0$ og $P_o=P_d$. For $P_o=0$ får vi

$$E_1 = \left[1 - (1 - P_d)^{(M-1)}\right] \cdot P_k \quad (3.5)$$

som er sannsynligheten for å slå ut et reelt mål hvis det er $M-1$ reelle mål og ingen utslåtte mål i området. For $P_o=P_d$ får vi

$$E_1 = \left[1 - (1 - P_d)^M\right] \cdot P_k \cdot \frac{M-1}{M} \quad (3.6)$$

som er sannsynligheten for å slå ut et mål hvis det er M mål i området, ganget med sannsynligheten $(M-1)/M$ for at dette var et av de reelle målene.

Med to utslåtte mål i søkeområdet blir det mer komplisert. Denne utregningen er derfor utelatt her, og det henvises til appendiks A. Imidlertid viser det seg at det er mulig å utlede et generelt resultat for effekten av en granat som har M mål i sitt søkeområde, hvorav N er utslått. Beviset for denne sammenhengen finnes også i appendiks A, og uttrykket ser slik ut:

$$E_N = P_k \cdot \sum_{i=0}^N \left\{ \left(-\frac{P_o}{P_d} \right)^{N-i} \cdot \frac{\binom{M}{i}}{\binom{M}{N}} \cdot \left[1 - (1-P_o)^i \cdot (1-P_d)^{M-i} \right] \right\} \quad (3.7)$$

Her er $\binom{M}{i}$ binomialkoeffisienten, som er $\frac{M!}{(M-i)!i!}$. Bruksområdet for ligning 3.7 er først og fremst følsomhetsanalyse av de forskjellige parametrene som inngår.

3.1.2 Stirrende søker

I forrige avsnitt så vi på en skannende søker. En stirrende søker fungerer litt annerledes. Den ser hele søkeområdet på en gang. Så prøver den å finne ut hvor i søkeområdet det finnes mål. Deretter rangerer den målene og angriper det målet som antas å være mest "høyverdig". Vi antar at det kun finnes én type mål i søkeområdet, men at det finnes både utslåtte og reelle mål av denne typen. Deteksjonssannsynligheten kan være forskjellig for reelle og utslåtte mål. Vi har kalt disse størrelsene henholdsvis P_D og P_u . Legg merke til at disse størrelsene ikke sier noe om sannsynligheten for å velge ut et detektert mål for angrep. Det kan godt hende at P_u er større enn P_D . Søkerens overkillmekanisme består da i å rangere utslåtte mål som mindre høyverdige enn reelle mål. La oss gi et reelt mål en vekt på 1, og et utslått mål en vekt $W_u \leq 1$. Med R reelle mål og U utslåtte mål som er detektert, defineres betydningen av disse vektene slik at sannsynligheten for å velge et reelt mål er

$$P_R = \frac{R}{R + U \cdot W_u} \quad (3.8)$$

mens sannsynligheten for å velge et utslått mål er

$$P_U = \frac{U \cdot W_u}{R + U \cdot W_u} \quad (3.9)$$

Legg merke til at det her antas at dersom granaten detekterer minst ett mål, vil den alltid angripe et av disse. Vi har også antatt at alle reelle mål har samme sannsynlighet for å bli oppdaget, og at de har samme vekt (det samme gjelder for utslåtte mål).

Det målet som angripes, har som tidligere sannsynligheten P_k for å bli slått ut. Overkillmekanisme i form av W_u har ikke samme betydning her som P_o hadde for en skannende søker. P_D og P_u har heller ikke samme betydning som P_d og P_o har for skannende søkere, ettersom de for skannende søkere betydde sannsynligheten for å detektere *og velge som mål*. Vi kommer tilbake til dette senere.

Som tidligere antar vi at det finnes M mål i søkeområdet. Sannsynligheten for å slå ut et mål når det ikke er noen utslåtte mål i søkeområdet, altså effekten til en granat med 0 utslåtte mål i søkeområdet, er

$$E_0 = \left(1 - (1 - P_D)^M\right) \cdot P_k \quad (3.10)$$

Det neste spørsmålet blir hva effekten er dersom det er 1 utslått mål i søkeområdet. Dersom granaten detekterer minst ett reelt mål, og ikke detekterer det utslåtte målet, er sannsynligheten for å slå ut et mål lik P_k . Hvis den detekterer i reelle mål og det utslåtte målet, er sannsynligheten for å velge et reelt mål lik

$$P_{R,i} = \frac{i}{i + W_u} \quad (3.11)$$

Sannsynligheten for å detektere det utslåtte målet er P_u . Sannsynligheten for å detektere nøyaktig i reelle mål, er

$$P(i) = P_D^i \cdot (1 - P_D)^{(M-i)} \cdot \binom{M-1}{i} \quad (3.12)$$

Den totale effekten dersom det er ett utslått mål i søkeområdet, kan da skrives som

$$E_1 = \left(1 - (1 - P_D)^{(M-1)}\right) \cdot (1 - P_u) \cdot P_k + P_u \cdot P_k \cdot \sum_{i=1}^{M-1} \left[\left(\frac{i}{i + W_u} \right) \cdot P_D^i \cdot (1 - P_D)^{(M-i)} \cdot \binom{M-1}{i} \right] \quad (3.13)$$

Den siste summen kan bare regnes ut eksakt for $W_u=0$ og $W_u=1$, dvs for perfekt overkillmekanisme og for ingen overkillmekanisme. Dermed kommer vi ikke så langt som vi gjorde for skannende søkere. For $W_u=0$ blir summen

$$\sum[\dots] = 1 - (1 - P_D)^{(M-1)} \quad (3.14)$$

som gir

$$E_{1,W_u=0} = \left(1 - (1 - P_D)^{(M-1)}\right) \cdot P_k \quad (3.15)$$

akkurat som forventet. For $W_u=1$ blir utregningen mer griset, men til slutt får vi

$$E_{1,W_u=1} = \frac{M-1}{M} \cdot \left(1 - (1 - P_D)^{M-1} \cdot (1 - P_u)\right) \cdot P_k \quad (3.16)$$

som naturlig nok er det samme som vi fikk for en skannende søker.

Dersom det er N utslåtte mål, kan ligning 3.13 generaliseres (se appendiks A). Vi får da en effekt som kan skrives som

$$E_N = \frac{1}{\binom{M}{N}} \cdot \sum_{i=0}^N \left\{ \binom{N}{i} \cdot P_u^i \cdot (1 - P_u)^{N-i} \cdot \sum_{j=0}^{M-N} \left[\left(\frac{j}{j + i \cdot W_u} \right) \cdot P_D^j \cdot (1 - P_D)^{M-N-j} \cdot \binom{M-N}{j} \right] \right\} \quad (3.17)$$

3.1.3 Sammenlikning av skannende og stirrende søkere

For en skannende søker benyttet vi en størrelse P_o , som er sannsynligheten for å detektere og velge som mål et allerede utslått mål. For stirrende søkere benyttet vi en annen størrelse, W_u , som var den relative sannsynligheten for å velge et utslått mål i forhold til et ikke utslått mål. Denne størrelsen er ikke direkte sammenlignbar med P_o for skannende søkere. En analogi til P_o for stirrende søkere blir å si at alle mål som detekteres og klassifiseres som reelle mål, har like stor sjans for å bli valgt ut, og at sjansen for å detektere og klassifisere et allerede utslått mål som et reelt mål er P_o . Analogt blir da sjansen for å detektere og klassifisere et reelt mål som et reelt mål lik P_d . Med en slik antakelse vil effekten av en granat dersom det er ett utslått mål i søkeområdet, være

$$E_1 = \left(1 - (1 - P_d)^{(M-1)}\right) \cdot (1 - P_o) \cdot P_k + P_o \cdot P_k \cdot \sum_{i=1}^{M-1} \left[\frac{i}{i+1} \cdot P_d^i \cdot (1 - P_d)^{(M-i-1)} \cdot \binom{M-1}{i} \right] \quad (3.18)$$

Denne summen er det mulig å regne ut eksakt, og svaret vi får ut, blir

$$E_1 = P_k \cdot \left[1 - (1 - P_o) \cdot (1 - P_d)^{(M-1)} - \frac{P_o}{M \cdot P_d} \cdot \left(1 - (1 - P_d)^M\right) \right] \quad (3.19)$$

som er akkurat det samme svaret som vi hadde for en skannende søker. Det interessante spørsmålet vi må stille oss, er derfor hva overkillmekanismene for henholdsvis skannende og stirrende søkere egentlig gjør. Skal vi benytte P_o eller W_u ?

Et annet viktig poeng er at for skannende søkere ble det gjort en antakelse som ikke er helt riktig, nemlig at søkerrekkefølgen er tilfeldig. De fleste søkere søker i en spiral utenfra og inn mot senter av søkeområdet. Dersom alle granatene lander i samme punkt med samme "retning", vil dermed alle granatene skanne gjennom målene i den samme rekkefølgen. Men granatene vil i virkeligheten **ikke** lande i samme punkt, og definitivt ikke med samme "retning", og derfor kan vi si at antakelsen vår er en akseptabel tilnærming til virkeligheten.

Dersom overkillmekanismen til en granat med stirrende søker fungerer som beskrevet i dette avsnittet, kan vi si at likning 3.7 gjelder både for skannende og stirrende søkere.

3.2 Effekt ved bruk av flere granater

I dette kapitlet studeres effekten av helt eller delvis overlappende søkeområder. Målet er å forstå betydningen av at søkeområdene overlapper, samt å finne et analytisk uttrykk for effekten av å skyte mot et målområde med flere granater. Et slik uttrykk blir utledet under forutsetning av at alle søkeområdene er sammenfallende.

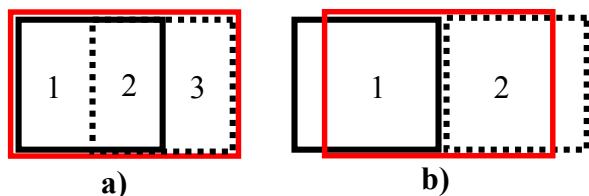
3.2.1 Betydning av overlappende søkeområder

Et interessant spørsmål er: Vil det alltid lønne seg å ha overlappende søkeområder som alle er innenfor målområdet, eller kan det noen ganger være bedre å legge en del av søkeområdet utenfor målområdet, for å unngå overlappende søkeområder og dermed redusere sannsynligheten for overkill?

Anta at alle målene befinner seg innenfor et klart definert, rektangulært målområde. Men hvert enkelt mål kan befinne seg hvor som helst innenfor målområdet, med samme

sannsynlighet for alle posisjoner. La oss si vi har N mål i målområdet (rødt rektangel), og skyter med G granater som alle har et kvadratisk søkeområde med sidekant 100 meter (sort heltrukket og sort stiplede kvadrat). Målområdet er rektangulært med sidekanter 100 meter og 150 meter.

I første omgang antar vi at granatene ikke er i stand til å skille mellom reelle mål og allerede utslåtte mål.



Figur 3-1: Overlapp vs ikke overlapp

La oss først se på et enkelt eksempel, der vi har to mål og to granater, se Figur 3-1. Vi kan satse på å legge begge søkeområdene innenfor målområdet, delvis overlappende (a), eller vi kan legge det ene søkeområdet delvis utenfor, og dermed unngå overlapp (b). La oss først se på tilfellet med overlappende søkeområder. Etersom målene har like stor sannsynlighet for å befinne seg i et hvert punkt i målområdet, og områdene 1, 2 og 3 i Figur 3-1 a) er like store, har hvert av målene sannsynlighet $1/3$ for å befinne seg i hvert av de tre målområdene. Med to mål har vi ni mulige kombinasjoner, og hver av disse kombinasjonene er like sannsynlige. La oss i første omgang anta at granatene har deteksjonssannsynlighet og killsannsynlighet lik 1. Hver granat får selvsagt kun lov til å slå ut ett mål, slik at dersom det finnes to mål i søkeområdet, har hvert mål 50% sjanse for å bli valgt ut. Forventet antall utslåtte mål blir da 1.5 (det er bare å sette opp alle mulighetene og legge sammen effektene).

Uten overlappende søkeområder trenger vi bare å skille målområdet i to deler, som vist i Figur 3-1 b). Med to mål gir dette fire mulige kombinasjoner som alle er like sannsynlige. Forventet antall utslåtte mål blir også i dette tilfellet 1.5, det samme som vi hadde for overlappende søkeområder.

Legg merke til at effekten for overlappende søkeområder vil øke dersom man har en overkillmekanisme i granatene (til 1.67 for perfekt overkillmekanisme), mens den forblir den samme for ikke overlappende søkeområder.

La oss så gjøre en enkel endring: I stedet for to, antar vi nå at det er tre mål i målområdet. For de overlappende søkeområdene, med tre interessante områder, gir dette 27 mulige kombinasjoner med lik sannsynlighet. Som tidligere er det ingen overkillmekanisme, slik at dersom det er flere mål i et målområde, har alle målene lik sannsynlighet for å bli valgt, uavhengig av om de er slått ut eller ikke. Forventet antall utslåtte mål blir da ca. 1.71 for tilfellet med overlappende søkeområder. For tilfellet uten overlappende søkeområder blir forventet antall utslåtte mål 1.75, altså marginalt bedre enn for overlappende søkeområder.

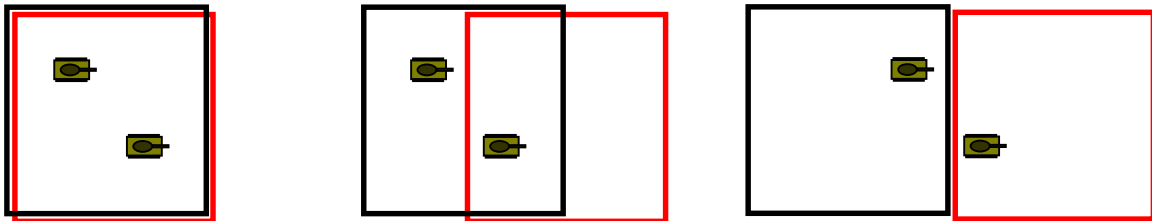
Vi går nå tilbake til tilfellet med to mål i søkeområdet, og ønsker å studere hva som skjer dersom deteksjonssannsynligheten er mindre enn 1, for eksempel 0.5 (killsannsynligheten settes fortsatt til 1). Da får vi 0.997 forventede utslåtte mål med overlappende søkeområder, og 0.875 forventede utslåtte mål uten overlappende søkeområder. Som forventet lønner det seg altså i større grad med overlappende footprint når deteksjonssannsynligheten er liten enn når den er stor. Med unntak av helt ekstreme situasjoner (dvs. når P_d og P_k er veldig nær 1, ingen

overkillmekanisme og det er minimum 3 mål), vil det ikke lønne seg å spre søkeområdene slik at de ligger delvis utenfor målområdet dersom målene er homogent fordelt innenfor et definert målområde.

La oss nå se på en situasjon der vi kjenner målenes posisjon eksakt. Vi har to mål og skal skyte med to granater. Vi ser fortsatt bort fra spredning og bomavstand, og ser på ladningenes nedslagsfelt. Vi har tre interessante muligheter:

- Begge søkeområdene kan inneholde målene innenfor sitt søkeområde (fullstendig overlapp)
- Det ene søkeområdet inneholder begge målene, mens det andre søkeområdet bare inneholder ett mål (delvis overlapp)
- Søkeområdene inneholder hvert sitt mål

I tillegg kan selvsagt begge søkeområdene inneholde ett felles mål, men det er opplagt ugunstig. Situasjonene er illustrert i Figur 3-2.



Figur 3-2: Målenes eksakte posisjon er kjent

La oss finne forventet antall utslåtte mål (E) for alle tre strategiene. Vi betegner disse E_F , E_D og E_I , for Fullstendig, Delvis og Ingen overlapp.

$$\begin{aligned}
 E_F &= \left[1 - (1 - P_d)^2\right] P_k \cdot \left(1 - \frac{1}{2} P_d P_k\right) \\
 &+ \left\{1 - \left[1 - (1 - P_d)^2\right] P_k\right\} \cdot \left[1 - (1 - P_d)^2\right] P_k \\
 &+ 2 \cdot \left[1 - (1 - P_d)^2\right] P_k \cdot \frac{1}{2} P_d P_k
 \end{aligned} \tag{3.20}$$

Dette uttrykket består av tre ledd. Det første leddet er sannsynligheten for at den første granaten slår ut et mål, mens den andre granaten ikke slår ut et (nytt) mål. Det andre leddet er sannsynligheten for at den første granaten ikke slår ut et mål, mens den andre granaten gjør det. Det siste leddet er sannsynligheten for at begge granatene slår ut hvert sitt mål, multiplisert med 2, ettersom dette gir effekten 2 utslåtte mål. Vi trekker sammen dette uttrykket og får

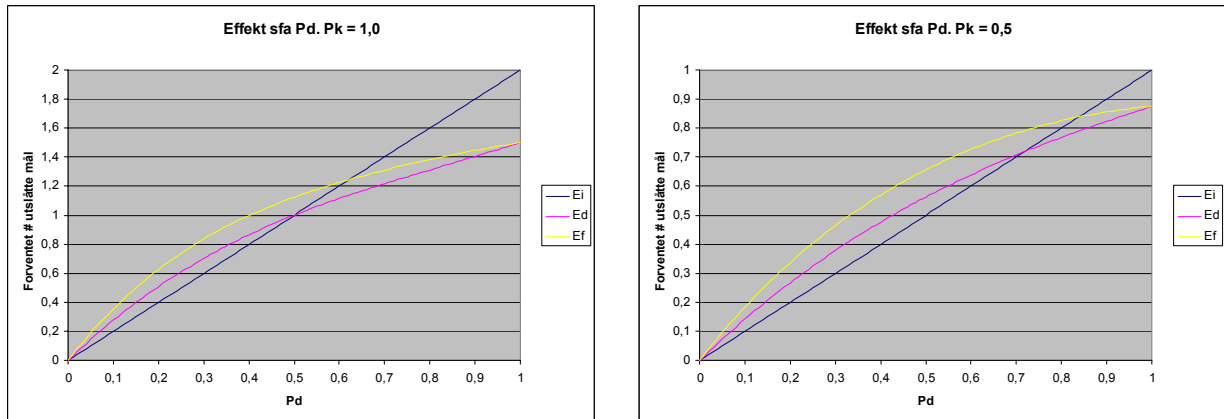
$$E_F = \left[1 - (1 - P_d)^2\right] P_k \cdot \left[2 - P_d P_k \left(\frac{3}{2} - P_d\right)\right] \tag{3.21}$$

Tilsvarende kan man regne ut E_D og E_I . Her skrives resultatene direkte opp:

$$E_D = P_d P_k \cdot \left[3 - P_d - P_d P_k \left(\frac{3}{2} - P_d\right)\right] \tag{3.22}$$

$$E_I = 2 \cdot P_d P_k \quad (3.23)$$

I Figur 3-3 er disse tre effektene sammenliknet. Her er effekten plottet som funksjon av P_d , for en konstant P_k som er henholdsvis 1 og 0,5.



Figur 3-3: Effekt som funksjon av P_d for $P_k=1$ og $P_k=0.5$

I denne figuren er det to ting som er verdt å legge merke til. Det ene er at det alltid er bedre med fullstendig overlapp enn delvis overlapp. Dette kunne vi for så vidt sett direkte på følgende måte: Dersom det første søkeområdet inneholder begge målene, blir spørsmålet hvorvidt det lønner seg å ha ett eller begge målene innenfor det andre søkeområdet. Dersom den første granaten ikke slo ut et mål, er det opplagt best at begge målene er innenfor det andre søkeområdet. Dersom den slo ut et mål, vil det være det samme om det andre søkeområdet inneholder ett eller begge målene, da det uansett er 50% sjanse for at det målet som er innenfor / velges ut er utslått.

Det andre som er verdt å legge merke til er at fordelene med overlappende søkeområder øker dersom P_d eller P_k minker. Med høy P_d er det ikke noe særlig å hente på å ha to mål innenfor søkeområdet, fordi granaten høyst sannsynlig vil klare å detektere et mål selv om det bare er ett mål i søkeområdet. Når P_k er høy, vil det være sannsynlig at den første granaten slår ut et mål, og dette øker sannsynligheten for at den andre granaten angriper et utslått mål.

Dersom granatene har en eller annen form for overkillmekanisme, spiller rekkefølgen på granatene en rolle, og tilfellet at første søkeområde inneholder begge mål mens andre inneholder ett mål blir ikke lenger det samme som at første søkeområde inneholder ett mål mens det andre inneholder begge. Uten overkillmekanismer blir disse tilfellene identiske, ettersom den første granaten ikke påvirker hvilket mål den andre granaten angriper.

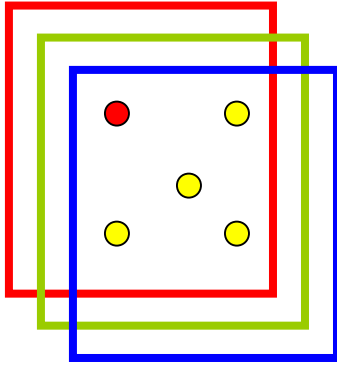
3.2.2 Beregning for granater med sammenfallende søkeområder

Hovedresultatene fra kapittel 3.1 var likningene som beskriver effekten av en granat som ser et søkeområde med M mål, hvorav N er utslått. I dette avsnittet beskrives sannsynligheten for å slå ut et gitt antall mål dersom man skyter med totalt G granater mot et målområde med M mål, dersom alle målene er innenfor søkeområdet til alle granatene. Sannsynligheten for at det er slått ut eksakt N mål etter at det er skutt med G granater, betegnes $P_G(N)$. Nok en gang antas det at rekkefølgen til en skannende søker er tilfeldig, og at både reelle og utslåtte mål betraktes som innbyrdes likeverdige av en stirrende søker. Effekten til en granat som ser N utslåtte mål og $M-N$ reelle mål, ble i kapittel 3.1 betegnet E_N .

Når man starter skytingen, er det ingen utslåtte mål i området. Sannsynligheten for å slå ut et mål med den første granaten er derfor E_0 . Sannsynligheten for å *ikke* slå ut et mål, er $1-E_0$. Vi får altså at

$$P_1(0) = 1 - E_0 \quad (3.24)$$

$$P_1(1) = E_0 \quad (3.25)$$



Figur 3-4: Etter at én granat har landet, er det enten 0 eller 1 utslått mål i området

For granat nummer to blir det noe mer komplisert, se Figur 3-4. Målet som er markert som rødt kan enten være slått ut eller ikke, slik at det enten er M eller $M-1$ reelle mål i søkeområdet (I figuren er $M=5$). Sannsynligheten for at det er M reelle mål i søkeområdet er $P_1(0)$, mens sannsynligheten for at det er $M-1$ reelle og ett utslått mål er $P_1(1)$. Effekten til granat nr. 2 er E_0 dersom det er null utslåtte mål i søkeområdet, og E_1 dersom det er ett utslått mål i søkeområdet. Etter granat 2 ser situasjonen slik ut:

$$P_2(0) = (1 - E_0)^2 \quad (3.26)$$

$$P_2(1) = (1 - E_0) \cdot E_0 + E_0 \cdot (1 - E_1) \quad (3.27)$$

$$P_2(2) = E_0 \cdot E_1$$

Generelt kan man se at når det er skutt G granater, blir sannsynlighetene

$$P_G(0) = (1 - E_0)^G \quad (3.28)$$

$$P_G(1) = E_0 \cdot \sum_{i=0}^{G-1} \left\{ (1 - E_0)^i \cdot (1 - E_1)^{(G-1-i)} \right\} \quad (3.29)$$

$$P_G(2) = E_0 \cdot E_1 \cdot \sum_{i=0}^{G-2} \left\{ (1 - E_0)^i \cdot \sum_{j=0}^{G-2-i} \left\{ (1 - E_1)^j \cdot (1 - E_2)^{(G-2-i-j)} \right\} \right\} \quad (3.30)$$

$$P_G(3) = E_0 \cdot E_1 \cdot E_2 \cdot \sum_{i=0}^{G-3} \left\{ (1 - E_0)^i \cdot \sum_{j=0}^{G-3-i} \left\{ (1 - E_1)^j \cdot \sum_{k=0}^{G-3-i-j} \left\{ (1 - E_2)^k \cdot (1 - E_3)^{(G-3-i-j-k)} \right\} \right\} \right\} \quad (3.31)$$

og så videre. Dette er geometriske rekker, som det er mulig å regne ut eksakt. Dette gir

$$P_G(0) = (1 - E_0)^G \quad (3.32)$$

$$P_G(1) = E_0 \cdot \left\{ \frac{(1 - E_0)^G}{(E_1 - E_0)} + \frac{(1 - E_1)^G}{(E_0 - E_1)} \right\} \quad (3.33)$$

$$P_G(2) = E_0 \cdot E_1 \cdot \left\{ \frac{(1 - E_0)^G}{(E_1 - E_0) \cdot (E_2 - E_0)} + \frac{(1 - E_1)^G}{(E_0 - E_1) \cdot (E_2 - E_1)} + \frac{(1 - E_2)^G}{(E_0 - E_2) \cdot (E_1 - E_2)} \right\} \quad (3.34)$$

og så videre. Dette gir grunnlag for å gjette på den generelle formelen

$$P_G(N) = \left(\prod_{i=0}^{N-1} E_i \right) \cdot \sum_{i=0}^N \left\{ \frac{(1 - E_i)^G}{\prod_{j=0, j \neq i}^N (E_j - E_i)} \right\} \quad (3.35)$$

Det er mulig å bevise at denne formelen faktisk er riktig, men det innebærer en del regning. Dette er derfor utelatt her, og den interesserte leser henvises til appendiks A, der dette beviset er utført.

Denne formelen har den fordelen at den i prinsippet er veldig rask å bruke; det er bare å programmere den og sette inn tallene, og man får svaret meget raskt. Ulempen er at den ”bryter sammen” for store verdier av M og G . Den er selvsagt fortsatt riktig, men den oppfører seg slik at hvert av leddene i summasjonen blir meget store tall. Summen blir imidlertid ikke så stor. Dette betyr at man må ta vare på svært mange siffer i mellomregningene for å få korrekte svar. Ved bruk i MatLab får vi problemer dersom G og M begge er over 40-50. Det er imidlertid sjelden interessant å regne på verken så mange mål eller så mange granater, da større ansamlinger av mål vil tilhøre ulike målgrupper, og således kan sees på som separate scenarier.

En annen ting er selvsagt at virkeligheten ikke er slik at alle målene vil være innenfor alle søkeområdene. SMArt sine substridsdeler har et søkeområde som er tilnærmet en sirkel med radius 130 m. Dette er vesentlig mindre enn utstrekningen til for eksempel en eskadron, som kan være et typisk mål. I tillegg kommer selvsagt granatenes bomavstand og spredning, som gjør bildet vesentlig mer komplisert.

3.3 Oppsummering

Vi har funnet frem til analytiske ligninger som beskriver forskjellige situasjoner. I kapittel 3.1 fant vi effekten av å skyte med én enkelt granat, og i kapittel 3.2 fant vi et uttrykk for effekten av å skyte mot et målområde med G granater. I tillegg så vi at for lave verdier av P_d vil det lønne seg å ha alle målene innenfor alle søkeområdene fremfor å ha ett mål innenfor hvert søkeområde.

Ligning 3.35 forutsetter at alle søkeområdene inneholder alle målene som beskytes. Dette er en grov tilnærming, og vil sjelden være riktig. Verdien av dette uttrykket er derfor begrenset, og er best egnet til å si noe om verdien av overkillmekanismer. Det viser seg at effekten av slike mekanismer normalt ikke er veldig store, med mindre man har ambisjoner om å slå ut

over halvparten av en fiendtlig formasjon. Det er derfor forståelig at produsentene ikke er villig til å lage avanserte overkillmekanismer, da dette antakelig vil ha den bieffekten at det gjør sensorene lettere å narre. Ligningene 3.7 og 3.13 inneholder ikke slike antagelser, men trenger som input hvor mange mål et bestemt søkeområde inneholder, samt hvor mange av disse som allerede er utslått. Disse uttrykkene er det derfor mulig å bruke videre og eventuelt inkludere i mer omfattende simuleringsprogrammer.

4 SMARTE ARTILLERIGRANATERS EFFEKT MOT PANSREDE MÅL - SIMULERINGER

Når vi skal se på granater som skytes mot målområdet med en bestemt spredning og bomavstand, blir situasjonen såpass komplisert at det ikke lenger er mulig å benytte analytiske uttrykk. Disse uttrykkene ville i så fall inneholde integrasjoner over mange variable, og kan være meget tidkrevende å benytte i en datamaskin. Derfor har vi i stedet laget to simuleringmodeller som skal håndtere denne situasjonen. Dette har fordeler og ulemper:

- Det er mulig å programmere simuleringmodeller som i stor grad ligner en virkelig stridssituasjon. Modellene kan bli svært realistiske, siden det er små begrensninger i hvor mange parametre som kan inkluderes
- Dersom konstantene som beskriver våpensystemene/scenariene er gode, fås nøyaktige simuleringresultater som bør ligge nært opp til det man vil få eksperimentelt
- Mange parametre gjør det imidlertid tidkrevende å finne ut hvordan hver enkelt påvirker resultatet
- Det er vanskelig å finne nøyaktige verdier som karakteriserer våpen og scenario
- Dersom en modell blir svært komplisert, øker faren for at:
 1. Bruker gir feil input
 2. Modellen benyttes utover sitt gyldighetsområde
 3. Programmeringsfeil påvirker resultatene

Spesielt er det en risiko for at modellene benyttes utover sitt gyldighetsområde. Dette kan gi resultater som er veldig gale. Det må derfor utvises forsiktighet når man trekker konklusjoner fra simuleringene.

Under prosjekt 798, delprosjekt 3, er det utviklet to versjoner av et simuleringsprogram. Disse blir skissert i de neste avsnittene.

4.1 Deterministisk modell

I denne modellen er målformasjonen kjent - det vil si at hvert enkelt måls eksakte posisjon er kjent. All usikkerhet knytter seg til granatens bane og effekt. De sentrale parametrene i modellen er:

- Granatens bomavstand
- Granatens spredning
- Siktemønster
- Størrelse på søkeområdet til granatene
- Målformasjonen
- P_k
- P_d
- P_o (denne er satt lik P_d)

I dette programmet genereres mange mulige verdier for bomavstand på en deterministisk måte, fremfor å trekke punktene vilkårlig. For hvert bomavstandspunkt beregnes så hver enkelt granats sannsynlighet for å ha de forskjellige mulige kombinasjonene av mål innenfor sitt søkeområde. En ”kombinasjon av mål” eller ”målkombinasjon” beskriver hvilke mål som ligger innenfor søkeområdet. Disse sannsynlighetene finnes ved å integrere over spredningen til de forskjellige granatene. For hvert enkelt bomavstandspunkt er disse sannsynlighetene uavhengige. Basert på disse uavhengige sannsynlighetene beregnes så forventet antall utslåtte mål. Dette gjøres for alle de utvalgte bomavstandspunktene, og en gjennomsnittsverdi for forventet antall utslåtte mål beregnes.

Algoritmen kan beskrives som følger:

Generer et sett med bompunkter

For (hvert bompunkt)

For (hver granat)

Basert på granatens siktepunkt og gjeldende bompunkt, integrer opp spredningen for å finne sannsynligheten for at granaten har de forskjellige målkombinasjonene i sitt søkeområde.

Multipliser sannsynligheten for å ha en gitt målkombinasjon i søkeområdet med effekten av dette. Effekten er i denne sammenheng sannsynligheten for å slå ut de forskjellige målene gitt målkombinasjon i søkeområdet.

End (For)

Nå har man en tabell med sannsynlighetene for at hver enkelt granat skal slå ut hvert enkelt mål, la oss kalle den $P_{g,m}$.

Sannsynligheten for at mål m er utslått, P_m , er da $P_m = 1 - \prod_{g=1}^G (1 - P_{g,m})$ der G er totalt antall granater.

Forventet antall utslåtte mål blir $E = \sum_{m=1}^M P_m$.

End (For)

Forventet antall utslåtte mål settes lik middelveiden til forventet antall mål for hvert bompunkt.

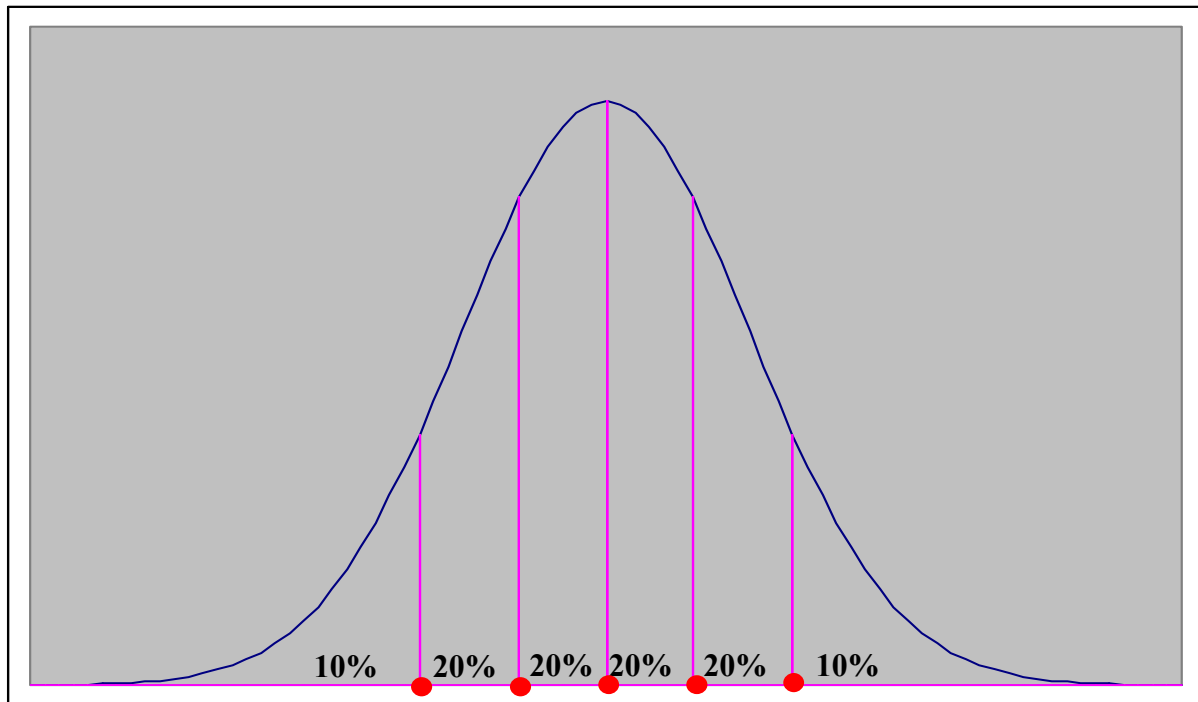
En mer detaljert beskrivelse følger.

4.1.1 Deterministisk nedslagsfelt

Det første som skjer i simuleringsprogrammet, er at det ”plukkes ut” bomavstander fra en normalfordeling. For å få mest mulig representative punkter, gjøres dette på følgende måte:

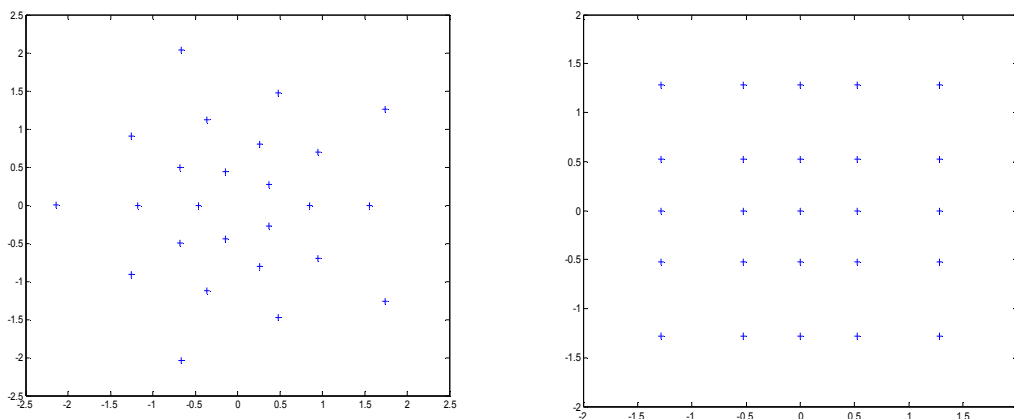
Dersom det er valgt n punkter i en dimensjon, regnes det ut hvilket tall det er $\frac{1}{2n}$ sjanse for å trekke et lavere tall enn. Så beregnes det n punkter som er slik at det er $\frac{1}{n}$ sjanse for å havne mellom to nabopunkter. Denne prosedyren er forsøkt beskrevet i Figur 4-1. Det antas at man ønsker like mange punkter i x- og y-retning, eventuelt like mange punkter i radiell som tangentiell retning, dersom man bruker polarkoordinater.

Dette settet med tilpassede tall genereres med utgangspunkt i fordelingsfunksjonen som beskriver bomavstanden i de enkelte dimensjonene. Resultatet av en slik prosess, der det er brukt 5 punkter per dimensjon, er vist for polarkoordinater og kartesiske koordinater i Figur 4-2.



Figur 4-1: Metode for utvelging av punkter.

I polarkoordinater får man et sett med ”sirkler”, og for å dekke rommet på en riktigst mulig måte er annenhver sirkel forskjøvet noe, slik at vi ikke får et sett med stråler. Avveiningene mellom polarkoordinater og kartesiske koordinater er kort diskutert i appendiks B. I programmet benyttes polarkoordinater.



Figur 4-2: Punkter generert fra normalfordeling i hhv polare og kartesiske koordinater

4.1.2 Deterministisk effektberegning

Når man vet hvilke mål hver enkelt granat har innenfor sitt søkeområde, er det mulig å regne ut den totale effekten eksakt. Men det kan by på problemer. Anta at det er 10 mål som beskyttes med 10 granater. Hvert av målene vil enten være eller ikke være innenfor granatens søkeområde, slik at hver granat har $2^{10}=1024$ forskjellige mulige målkombinasjoner. Dette er ikke et uoverkommelig høyt tall, og absolutt mulig å regne med. Men effekten av en granat avhenger av effekten av de tidligere granatene, som igjen avhenger av deres målkombinasjoner. Antall måter 10 granater kan lande på når hver granat har 1024 forskjellige muligheter, er 1024^{10} . Dette tallet er så stort at det blir praktisk umulig å gjennomføre beregningene.

Hvis man ser bort fra overkillmekanismene til granatene, vil ikke effekten til granatene lenger være avhengig av hverandre når man med ”effekt” i denne sammenheng mener sannsynligheten for at et bestemt mål slås ut. For eksempelet over er vi da tilbake til 1024 muligheter per granat. Det må riktignok gjøres beregninger for hver granat, men siden alle granatene har de samme mulige målkombinasjonene (riktignok med forskjellig sannsynlighet for at de forskjellige kombinasjonene inntreffer), kan dette beregnes én gang. Sannsynligheten for å slå ut et mål dersom det er M mål i søkeområdet, er da gitt som

$$P(M) = P_k \cdot \left(1 - (1 - P_d)^M\right) \quad (4.1)$$

Sannsynligheten for at et bestemt mål blir slått ut er 0 dersom målet ikke var i søkeområdet, og $P(M)/M$ dersom det var i søkeområdet. Ut fra dette kan man beregne sannsynligheten for at hvert enkelt mål blir slått ut av en granat med en bestemt målkombinasjon i sitt søkeområde. Hvis dette multipliseres med den enkelte granats sannsynlighet for å ha de forskjellige målkombinasjonene, får vi hver granats sannsynlighet for å slå ut hvert enkelt mål.

Dersom det er N granater totalt, og granat i har sannsynligheten P_i for å slå ut et bestemt mål, blir den totale sannsynligheten for at dette målet blir slått ut

$$P = 1 - \prod_{i=1}^N (1 - P_i) \quad (4.2)$$

Ved å legge sammen sannsynligheten for at hvert enkelt mål blir slått ut, får vi forventet antall utslåtte mål. Dette gjøres for hvert bomavstandspunkt, og et midlere forventet antall utslåtte mål beregnes.

Det er bare forventet antall utslåtte mål som kan beregnes ved denne modellen, den totale fordelingsfunksjonen for utslåtte mål kan ikke bestemmes på denne måten. En annen ulempe med metoden er at den tar lang tid å kjøre hvis man skal oppnå troverdige resultater. Resultatene vil imidlertid ikke variere fra simulering til simulering, hvilket gjør det lett å reproducere resultater. Dette er også en fordel dersom man ønsker å optimalisere resultatet (hvis resultatet avhenger av tilfeldigheter, risikerer man at det beste resultatet man fikk delvis skyldes tilfeldigheter). Det viser seg imidlertid at metoden er så treg at den ikke kan konkurrere med den stokastiske metoden på noe nivå. Det er derfor den stokastiske metoden som vil bli benyttet videre i prosjektet. Tankene rundt generering av punkter på en annen måte enn ved trekning av tilfeldige tall vil vi imidlertid ta med oss til senere. Se imidlertid appendiks B for en mer detaljert betraktning rundt dette.

4.2 Stokastisk modell

I denne modellen er også posisjonen til de forskjellige målene usikker. Modellen baserer seg på at en OP anslår både størrelse og posisjon til målformasjon. Modellen kan også ta hensyn til målbevegelse, og OP må da anslå formasjonens hastighet i x- og y-retning. De sentrale parametrene i modellen er:

- Granatenes bomavstand
- Granatenes spredning
- Siktemønster
- Størrelse på søkeområdet til granatene
- P_k
- P_d
- P_o
- OP's angivelse av målformasjonen
- Usikkerheten i OP's målangivelse
- OP's hastighetsestimatet med usikkerhet

Beregningene gjøres ved hjelp av Monte Carlo-simuleringer. Programmet er relativt hurtig. Når en målformasjon på 12 stridsvogner blir beskyttet med 12 artillerigranater, utføres beregningene til 100 000 simuleringer på 30 sekunder. Dette resultatet har da et standardavvik på ca 1 promille. Dette gjør at man kan få en vesentlig bedre nøyaktighet ved denne metoden enn ved den deterministiske metoden med samme tidsforbruk.

4.2.1 Målbeskrivelse

Programmet er konstruert for å behandle målenes posisjon som usikker. Usikkerheten kan imidlertid settes til null, slik at målenes posisjoner behandles som kjent.

Dersom målenes posisjon er kjent, er det tilstrekkelig å oppgi individuell posisjon til hvert av målene i målformasjonen. Dersom målenes posisjon er usikker, blir imidlertid situasjonen mer komplisert. Programmet antar da at OP beskriver målformasjonen ved følgende parametre:

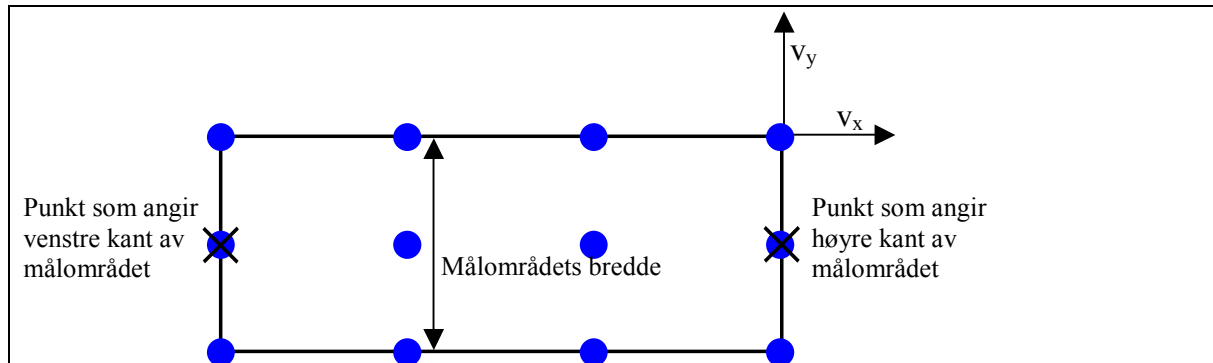
1. Et punkt som angir målformasjonens vestre kant
2. Et punkt som angir målformasjonens østre kant
3. Bredden på målformasjonen
4. Målformasjonens hastighet i x-retning
5. Målformasjonens hastighet i y-retning
6. Mønster for målformasjonen, OP angir antall mål

Dette er i samsvar med de data en OP faktisk vil formidle, med unntak av hastigheten, da man så langt ikke har drevet beskytning av bevegelige mål. Dette er en av fordelene med denne modellen.

Informasjonen som må oppgis er skissert i Figur 4-3. Dataene gis på denne formen fordi dette er standard måte for en OP å angi måldata på.

I programmet oppgis det også en usikkerhet i alle størrelser som OP oppgir. Siden de to punktene som oppgir høyre og venstre kant av målområdet består av både x- og y-koordinat, betyr dette syv størrelser med usikkerhet.

Det er også usikkerhet i mønsteret til målformasjonen. Målene vil blant annet ikke fordele seg i et eksakt grid slik som skissert i Figur 4-3. Vi gir derfor hvert mål et lite normalfordelt bidrag, og denne spredningen oppgis av programmets bruker.



Figur 4-3: Skisse av målformasjonen. Det antas at målformasjonen kan omslutes av et rektangel og at OP karakteriserer dette med to punkter, bredde, samt hastighet i x- og y-retning. De blå punktene angir målenes posisjon

Programmet trekker de usikre størrelsene fra passende normalfordelinger og beregner deretter målenes simulerte posisjon.

Det er ikke et krav fra programmets side at dataene angis slik vi forventer å få dem fra en OP. Brukeren kan alternativt oppgi posisjonen til hvert enkelt mål direkte. Målene kan fremdeles spres etter en normalfordeling definert av brukeren.

4.2.2 Granatbeskrivelse

Programmet tar hensyn til granatens bom og spredning. Det er dessuten mulig å gjøre granatens deteksjonssannsynlighet tidsavhengig. Dette er nyttig, dersom man skal modellere at målene kjører i skjul etter første salve, bruker APS (aktive beskyttelsessystemer) med ”softkill”, eller på annen måte reduserer sannsynligheten for deteksjon.

Programmet krever at brukeren gir følgende informasjon:

1. Siktepunkt til hver granat
2. Størrelsen på søkeområdet til hver granat
3. Tidspunktet da granaten vil nå frem til målet (For bevegelige mål)
4. Bomavstand og spredning i x- og y-retning
5. P_d for alle målene i formasjonen

6. P_o for alle målene i formasjonen
7. P_k for alle målene i formasjonen

Når granatens simulerte treffpunkt skal bestemmes, gjøres trekninger fra normalfordelingen. Først trekkes en bomavstand som er felles for alle granatene, deretter trekkes spredning for én og én granat. Når bom og spredning er trukket, kan granatens treffpunkt beregnes.

4.2.3 Beregning av granatens effekt

Nå er både målenes posisjon og granatens treffpunkt bestemt. Ved hjelp av verdiene for granatens søkeområde og for granatens deteksjons- og kill-sannsynlighet, kan nå effekten beregnes. I disse beregningene antas det at sannsynlighetene er uavhengige.

Beregningen for n granater går som følger:

1. For granat nr 1: Beregn hvilke mål som befinner seg innenfor granatens søkeområde
2. Bestem deteksjonssannsynligheten for de forskjellige målene. Deteksjons-sannsynligheten er P_d dersom målet er reelt, P_o dersom målet er utslått
3. Trekk hvilke av målene som detekteres, og velg tilfeldig ett av disse dersom minst ett detekteres
4. Trekk om det valgte målet slås ut
5. Oppdater matrise over hvor mange ganger de forskjellige målene er slått ut
6. Gjenta punkt 1 - 5 for granat 2 til og med granat n

4.2.4 Resultat

Vi har i avsnittene 4.2.1 - 4.2.3 utført trekninger slik at vi har funnet én målformasjon, én fordeling av treffpunkt og én beregning av denne salvens effekt. Vi har altså oppnådd ett mulig svar. Dersom simuleringen gjentas, vil imidlertid nye trekninger foretas. Trekningene vil ikke bli identiske med forrige simulering, og sluttsvaret vil som regel endre seg noe.

For å finne hvilket svar som er rimelig å oppgi, gjentas simuleringene en mengde ganger. Effekten E oppgis da som middelveien til alle disse kjøringene. For å indikere hvor stor spredning vi får i resultatene, oppgis også standardavviket s_E . Vi ønsker en pålitelig verdi for den midlere effekten og ønsker å vite hvor mange simuleringer som da må kjøres. Med andre ord ønsker vi at standardavviket til middelveien skal bli lite. Standardavviket til middelveien er gitt av:

$$s_{\bar{E}} = \frac{s_E}{\sqrt{n}},$$

der n angir antall simuleringer. For den relative feilen får vi:

$$\frac{s_{\bar{E}}}{\bar{E}} = \frac{s_E}{E \cdot \sqrt{n}} \quad (4.3)$$

Dersom vi ser på oppdrag der minst halve målformasjonen skal slås ut, slik at $\bar{E} > 0.5M$, og dersom vi benytter det konservative anslaget om at $s_E < 0.5M$, får vi:

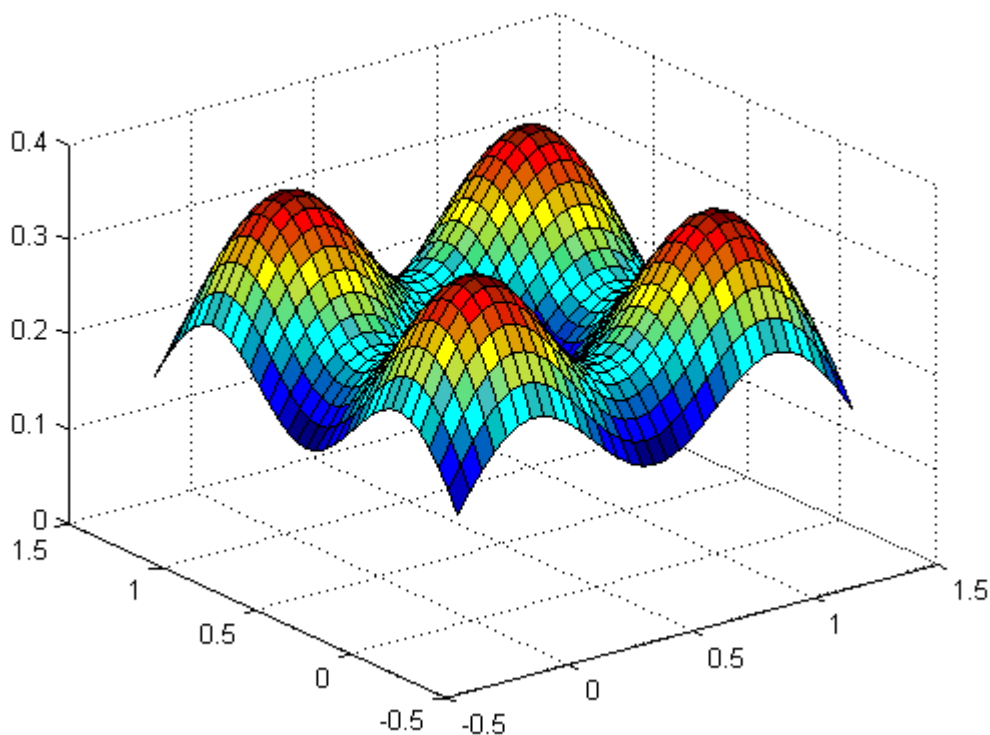
$$\frac{s_{\bar{E}}}{\bar{E}} = \frac{s_E}{E \cdot \sqrt{n}} < \frac{0.5 \cdot M}{0.5 \cdot M \cdot \sqrt{n}} = \frac{1}{\sqrt{n}} \quad (4.4)$$

Dersom standardavviket skal bli mindre enn 1% fås da $n > 10\,000$. Vi har dermed sett at vi for et typisk scenario får en god nøyaktighet dersom programmet gjentas 10 000 ganger. Dette betyr en kjøretid på få sekunder.

En nøyaktighet på 1% kan virke unødvendig nøyaktig med tanke på all den usikkerheten som ligger i input-dataene. Dette er for så vidt riktig, så med tanke på å finne hvilken effekt man oppnår, er ikke en slik nøyaktighet nødvendig. Situasjonen blir imidlertid en annen når vi ønsker å finne den optimale siktepunktsfordelingen. Dette kommer vi tilbake til i kapittel 5.

5 OPTIMALISERING AV SMARTE ARTILLERIGRANATERS SIKTEPUNKT

Vi ønsker å optimalisere siktepunktene til granatene, slik at vi får størst mulig effekt. Den beste og enkleste måten å optimalisere en funksjon på, er å derivere den og sette den deriverte med hensyn på alle variable lik null. Dette krever imidlertid at man har det matematiske uttrykket for funksjonen man ønsker å optimalisere, noe vi ikke har. Det er derfor nødvendig å benytte en annen, numerisk fremgangsmåte. Dette innebærer at man må kalle funksjonen mange ganger. Ettersom det tar tid å kalle både den deterministiske og den stokastiske funksjonen for effekt, er vi begrenset til noen hundre funksjonskall dersom vi skal få et resultat innen rimelig tid (noen minutter). Dette legger føringer på hvordan vi kan utføre optimaliseringen.



Figur 5-1: Med fire mål og én granat, kan effekt som funksjon av siktepunkt for eksempel se slik ut. Her er det altså best å sikte midt på ett av målene. Med større spredning eller bomavstand (med kun én granat har bomavstand og spredning samme virkning) vil det lønne seg å sikte midt mellom alle målene. For mer enn én granat blir bildet vesentlig mer komplisert.

5.1 MatLab-rutiner for optimalisering

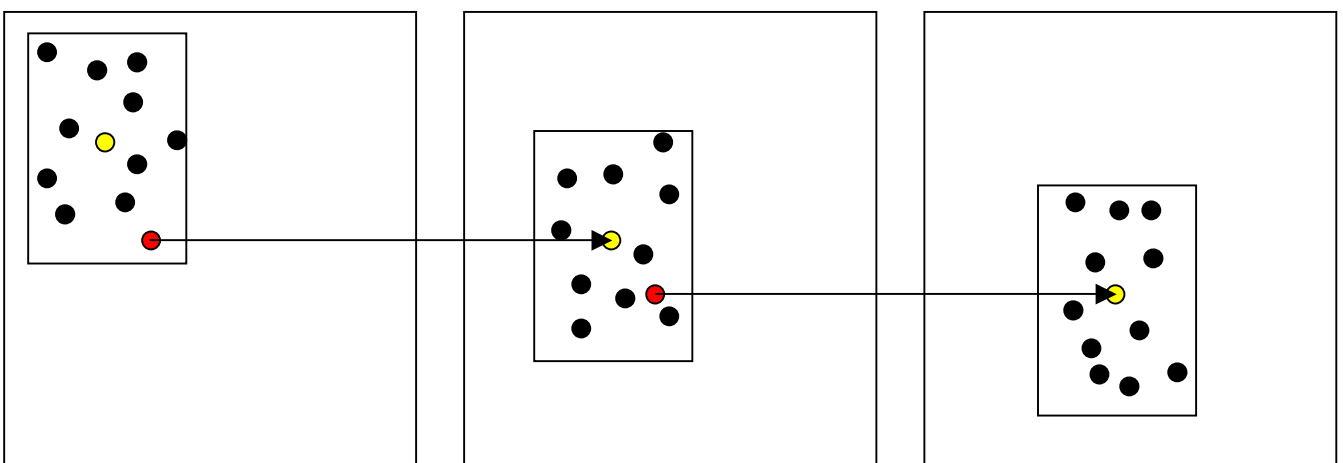
MatLab inneholder et bibliotek med rutiner for optimalisering av forskjellige funksjoner. I disse rutinene er det flere parametre som kan justeres for å gjøre optimaliseringen mest mulig effektiv. De rutinene vi har sett på, synes å basere seg på gradienter som enten oppgis av bruker eller som beregnes av programmet. Dette medfører gjerne raske og gode resultater for

funksjoner som er glatte og oppfører seg ”pent”, men våre funksjoner gjør ikke det.

Det deterministiske programmet vil det for så vidt være mulig å optimalisere med MatLabs rutiner i sin nåværende form, selv om det krever at man er flink til å sette de rette parametrene i optimeringsrutinene. Programmet hadde imidlertid en noe annen utforming i en tidligere fase, og det ble derfor laget et eget, enkelt optimaliseringsprogram for ”stygge” mangedimensjonale funksjoner, som benytter ”brute force”-metoden. Fortsatt er dette nyttig, da MatLab-rutinen som vi har brukt har en tendens til å bare finne lokale optima, ikke det globale optimum. Dette programmet er dessuten nødvendig for optimalisering av det stokastiske programmet, som også i sin nåværende form er vanskelig å optimalisere ved hjelp av MatLabs egne rutiner.

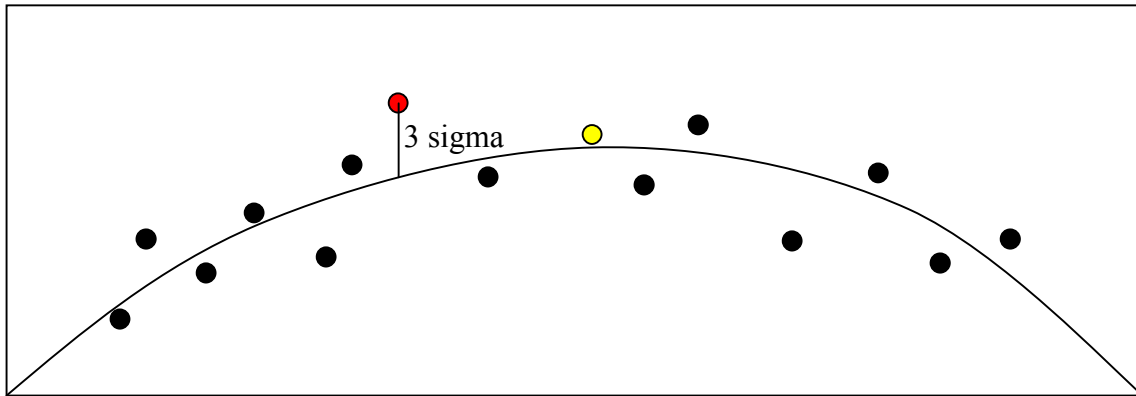
5.2 Optimalisering ved ”brute force”

Denne formen for optimering er veldig enkel, og lite effektiv. Den har imidlertid den fordelen at den vil fungere for alle funksjoner, uansett hvor stygge de er. Man starter med å definere et startpunkt og en mangedimensjonal ”boks” som det skal letes innenfor. I prinsippet kan denne ”letingen” utføres ved å generere et grid med et bestemt antall punkter i hver dimensjon, men når antall dimensjoner blir stort, tar dette meget lang tid, og det må finnes en annen måte å ”lete” på. I vårt tilfelle har vi to dimensjoner for hver granat (x- og y-koordinat til siktepunktet). Vi velger derfor å trekke et bestemt antall tilfeldige punkter innenfor denne boksen, og velger ut det av disse punktene som gir høyest verdi. Dette settes så til det nye startpunktet, og det søkes i en ny boks rundt dette punktet. Når den beste oppnådde verdien i en boks ikke er bedre enn startpunktet i denne boksen, avsluttes programmet. Det er også mulig å søke på nytt med dette punktet som startpunkt og en mindre boks, dersom man ønsker en bedre oppløsning på svaret. Metoden er illustrert i Figur 5-2 for et todimensjonalt problem.



Figur 5-2: De *gule* punktene angir startpunkter, mens de *røde* angir maksimumspunktene blant de undersøkte punktene. Maksimumspunktet i én boks blir startpunktet i den neste. I den siste ruta er det ingen verdier som er høyere enn startpunktet, slik at dette punktet også er maksimumspunktet. Når dette inntreffer, avsluttes søket.

Denne metoden vil også fungere for en stokastisk funksjon, selv om det alltid er visse problemer med å optimalisere stokastiske funksjoner. Anta at funksjonen er temmelig flat, slik at man ikke får store forskjeller i verdi selv om man endrer inputparametrene en del. Den stokastiske funksjonen vil ha et standardavvik forbundet med resultatet, som vil være avhengig av antallet kjøring; jo flere kjøring, desto lavere standardavvik, men desto større tidsforbruk. Ettersom en optimalisering vil kalle funksjonen mange ganger, kan man ikke bruke alt for mange kjøring, da dette vil ta lang tid.



Figur 5-3: Med flere hundre punkter vil det største avviket være omlag 3 sigma. Det betyr at den største oppnådde verdien (rød prikk) kan ligge et stykke unna den faktiske største verdien (gul prikk) dersom funksjonen er tilstrekkelig "flat".

I forbindelse med en optimaliseringsrutine som beskrevet over, vil funksjonen måtte kalles typisk noen hundre ganger. Det betyr at det resultatet som ligger lengst over den "korrekte" verdien, vil ligge i størrelsesorden tre standardavvik over den "korrekte" verdien. Med en relativt flat funksjon, kan dette lett utgjøre mer enn forskjellen mellom verdien til det reelle optimum og verdien i punktet med størst feil, slik at programmet tolker punktet med størst avvik fra reell verdi som optimum. Dette er forsøkt illustrert i Figur 5-3 for et endimensjonalt problem.

På den andre siden betyr jo en "flat" kurve at det ikke er så nøye hvor på kurven man befinner seg, da man får nesten samme verdi uansett. Så selv om vi risikerer å få et relativt stort avvik fra optimal siktepunktfordeling, vil ikke avviket fra optimal effekt være særlig stor.

5.3 Optimalisering gitt symmetri for siktemønster

En annen måte å optimalisere vårt spesielle problem på, er å angi et mønster for siktepunktsfordelingen, definert ved én eller flere parametre, som vi tror er optimal. Så optimaliserer vi disse parametrene i stedet for å optimalisere hele siktepunktsfordelingen. For eksempel kan vi si at vi tror det beste er å skyte seks granater på en linje, med lik avstand mellom granatene. Da kan vi optimalisere senterpunktet for granatene, avstanden mellom granatene og retningen på linja (vinkelen). Antakeligvis er vi i stand til å gjette på vinkel og senterpunkt, slik at vi står igjen med avstanden mellom granatene. Dette vil være en langt raskere måte å optimalisere på, da vi går fra to parametre per granat til én eller to parametre totalt som skal optimaliseres. Metoden forutsetter selvsagt at man er i stand til å foreta en intelligent

antagelse om formen på siktemønsteret.

5.4 Eksempel på optimalisering

Her gir vi et eksempel på et scenario som er forsøkt optimalisert, både ved bruk av den deterministiske metoden og den stokastiske metoden. Verdien i de optimale punktene beregnes så med stor nøyaktighet i det stokastiske programmet.

Målposisjon

x: -300 -300 300 300

y: -300 300 -300 300

Hastighet: I ro

Usikkerhet i hastighet: 0 m/s

Antall granater: 2

Bomavstand

x: 200 m

y: 200 m

Spredning

x: 100 m

y: 100 m

Tid fra observasjon til effekt: 0 s

Radius på søkeområde: 130 m

P_d : 0,8

P_o : 0,8

P_k : 0,7

Vi velger startposisjonene x: 0, 0 og y: 0, 0 for siktepunktene til granatene.

	Effekt	Faktisk effekt	Tidsforbruk [s]	Siktemønster (x1,y1) (x2,y2)
Det, ML-rutine	0,178	0,172	196	(-169,-169) (-169,-169)
Det, random-rutine	0,196	0,187	181	(-279,-304) (-295,295)
Det, system-rutine	0,196	0,189	74	(-267,-267) (267,267)
Sto, random-rutine	0,187	0,185	139	(267,245) (-261,336)
Sto, system-rutine	0,197	0,185	69	(-333,-267) (333,267)

Tabell 5-1: Sammenligning av forskjellige metoder for optimalisering

Det er verdt merke seg at for mer komplekse scenarier øker tiden for det deterministiske programmet vesentlig raskere enn for det stokastiske programmet. For et realistisk scenario kan det ta flere dager. Det er også viktig å få med seg at selv om ”system-rutinen” synes å være den raskeste metoden, og like god som ”random-rutinen”, så krever denne rutinen at man på forhånd har en ganske god idé om hvordan siktemønsteret skal se ut. Har man riktig struktur, er den god, men den hjelper ikke til med å finne selve strukturen på siktemønsteret.

5.5 Konklusjon

Etter å ha testet forskjellige skytemønstre mot diverse målkombinasjoner, og også gjort forsøk på å optimalisere skytemønsteret mot de samme målkombinasjonene, har vi sett at funksjonene er temmelig ”flate” i de fleste tilfeller, det vil si at man får noenlunde samme effekt for et stort spekter av siktemønstre. Det kan riktignok være vanskelig og tidkrevende å finne det absolutt beste siktemønsteret, men det er enkelt å finne et siktemønster som gir 90% eller mer av den maksimalt oppnåelige effekten. Dette forteller oss at det er bortkastet tid å prøve å finne det absolutt beste siktemønsteret, og at vi heller bør konsentrere oss om å finne en metode for å angi et siktemønster som er ”godt nok”. Vi vil derfor videre i prosjektet konsentrere oss om å utvikle en metode for å finne et ”godt nok” siktemønster heller enn å satse på en stringent optimalisering.

6 KONKLUSJON

6.1 Beregningsmetoder

Vi har utviklet tre metoder for beregning av effekt for smarte granater. Den ene metoden baserer seg på et matematisk formelverk, mens de to andre genererer treffpunkt ut fra en sannsynlighetsfordeling, henholdsvis ved bruk av et deterministisk og et stokastisk generert sett med treffpunkter for granatene.

De matematiske formlene gir raske og presise svar, men er kun i stand til å beregne effekten for enkle scenarier, der ikke bare målenes eksakte posisjon er kjent, men også hver enkelt granats treffpunkt. Disse formlene er derfor ikke særlig egnet for å se på realistiske scenarier, men er derimot anvendelige når man skal vurdere effekten av å øke eller senke diverse parametre.

Når det gjelder de to andre metodene, er den stokastiske metoden vesentlig raskere enn den deterministiske, og kan i tillegg behandle mer realistiske scenarier. Den er riktignok noe mindre egnet til optimalisering, men dette blir mer enn oppveid av den mye større hastigheten. Vi fører derfor ikke arbeidet med den deterministiske modellen videre, men konsentrerer oss om å bruke og forbedre den stokastiske modellen.

6.2 Optimalisering

Ved å gjette på startpunkter og så benytte diverse rutiner for optimalisering, er vi i stand til å finne siktepunkter som er nær optimale. Dette tar imidlertid lang tid, og krever god kjennskap til rutinene for optimalisering. Det har vist seg at vi er i stand til å gjette på et startpunkt som gir en relativt høy effekt, som regel over 90% av effekten av den optimale siktepunktsfordelingen.

Det virker da ikke rimelig å bruke meget lang tid på å finne en siktepunktsfordeling som er marginalt bedre enn den vi startet med. Vi vil derfor i det videre arbeidet i prosjekt 798, delprosjekt 3, gå bort fra tanken om optimalisering, og heller prøve å utarbeide rutiner som gir siktepunkts-fordelinger som er ”gode nok”.

6.3 Veien videre

Metodene som er beskrevet i dette dokumentet kan brukes til å studere enkle scenarier. Neste skritt blir å sette opp aktuelle scenarier sammen med offiserer. Noen av disse scenariene vil bli mer kompliserte enn de eksisterende metodene er i stand til å håndtere, slik at en videreutvikling av metodene vil bli nødvendig.

En av de viktigste effektene som det ikke tas hensyn til i de nåværende modellene, er kontrabeskytning. Det antas hele tiden at det kun er den ene siden som beskytter den andre. Dette må implementeres for å få frem effektene av ”skyt og flytt”-taktikk, og også for å vurdere viktigheten av instantan ild.

Forskjellen fra den eksisterende stokastiske modellen blir at skuddene må koples til målene på et vis, slik at en enhet som blir slått ut, ikke blir i stand til å skyte. Videre blir det antakelig behov for at enhetenes hastighet kan endres med tiden, slik at de kan stå stille og skyte, flytte seg og så stå stille og skyte igjen. Tidsfaktoren blir dermed mer sentral enn den er i den nåværende modellen.

APPENDIKS

A BEVIS AV ANALYTISKE UTTRYKK

A.1 Effekt av én granat

Dersom det ikke er noen utslåtte mål i søkeområdet, er det M reelle mål. Sannsynligheten for å slå ut et reelt mål, altså effekten til granaten, blir da

$$E_0 = \left(1 - (1 - P_d)^M\right) \cdot P_k \quad (\text{A.1})$$

Med ett utslått mål og $M-1$ reelle spiller rekkefølgen på skanningen en rolle. Vi har antatt tilfeldige skanne-rekkefølger, det vil si at sannsynligheten for at det utslåtte målet er det første, andre, tredje, fjerde osv. målet som det skannes over, er like store, nemlig en M -del. Dersom det utslåtte målet er mål nr. i , blir sannsynligheten for å slå ut et reelt mål lik

$$E_{1,i} = \left[1 - (1 - P_d)^{(i-1)}\right] \cdot P_k + (1 - P_d)^{(i-1)} \cdot (1 - P_o) \cdot \left[1 - (1 - P_d)^{(M-i)}\right] \cdot P_k \quad (\text{A.2})$$

Totalt blir derfor sannsynligheten for å slå ut et reelt mål

$$E_1 = \frac{1}{M} \cdot \sum_{i=1}^M \left[\left(1 - (1 - P_d)^{(i-1)}\right) + (1 - P_d)^{(i-1)} \cdot (1 - P_o) \cdot \left(1 - (1 - P_d)^{(M-i)}\right) \right] \cdot P_k \quad (\text{A.3})$$

Dette er en geometrisk rekke, som kan regnes ut eksakt. Man får da

$$E_1 = P_k \cdot \left[1 - (1 - P_o) \cdot (1 - P_d)^{(M-1)} - \frac{P_o}{M \cdot P_d} \cdot \left(1 - (1 - P_d)^M\right) \right] \quad (\text{A.4})$$

Det er enkelt å sjekke at vi får riktig resultat for grensetilfellene $P_o=0$ og $P_o=P_d$. For $P_o=0$ får vi

$$E_1 = \left[1 - (1 - P_d)^{(M-1)}\right] \cdot P_k \quad (\text{A.5})$$

som er sannsynligheten for å slå ut et reelt mål hvis det er $M-1$ reelle mål og ingen utslåtte mål i området. For $P_o=P_d$ får vi

$$E_1 = \left[1 - (1 - P_d)^M\right] \cdot P_k \cdot \frac{M-1}{M} \quad (\text{A.6})$$

som er sannsynligheten for å slå ut et mål hvis det er M mål i området, ganget med sannsynligheten $(M-1)/M$ for at dette var et av de reelle målene.

Med to utslåtte mål i søkeområdet blir det mer komplisert. Som tidligere antar vi en skannende søker, med vilkårlig skannerekkefølge. Anta at mål nr. i og mål nr. j er slått ut, der $j > i$. Effekten blir da

$$E_{2,i,j} = \left(1 - (1 - P_d)^{(i-1)}\right) \cdot P_k + (1 - P_d)^{(i-1)} \cdot (1 - P_o) \cdot \left(1 - (1 - P_d)^{(j-i-1)}\right) \cdot P_k \\ + (1 - P_d)^{(j-2)} \cdot (1 - P_o)^2 \cdot \left(1 - (1 - P_d)^{(M-j)}\right) \cdot P_k \quad (\text{A.7})$$

Her er $0 < i < j \leq M$. Ettersom det er totalt $\binom{M}{2}$ mulige måter å ha 2 utslåtte mål i søkerområdet på, blir den totale effekten

$$E_2 = \frac{1}{\binom{M}{2}} \cdot \sum_{j=2}^M \left[\sum_{i=1}^{j-1} E_{2,i,j} \right] \quad (\text{A.8})$$

Nok en gang er dette geometriske rekker som det er mulig å regne ut eksakt. Det blir mye griseete regning, og det endelige svaret er

$$E_2 = \left[1 - (1 - P_d)^{(M-2)} \cdot (1 - P_o)^2\right] \cdot P_k - \frac{2 \cdot P_o}{(M-1) \cdot P_d} \cdot \left[1 - (1 - P_d)^{(M-1)} \cdot (1 - P_o)\right] \cdot P_k \\ + \frac{2 \cdot P_o^2}{M \cdot (M-1) \cdot P_d^2} \cdot \left[1 - (1 - P_d)^M\right] \cdot P_k \quad (\text{A.9})$$

La oss se om det er mulig å utlede et generelt resultat for en søker som har M mål innenfor søkeområdet, hvorav N er utslått. Dette tilsvarer å finne E_N . Ut fra resultatene for E_0 , E_1 og E_2 , samt for E_3 , som ikke er regnet ut her, foreslås følgende resultat:

$$E_N = P_k \cdot \sum_{i=0}^N \left\{ \left(-\frac{P_o}{P_d}\right)^{N-i} \cdot \frac{\binom{M}{i}}{\binom{M}{N}} \cdot \left[1 - (1 - P_o)^i \cdot (1 - P_d)^{M-i}\right] \right\} \quad (\text{A.10})$$

Det er enkelt å se at dette resultatet stemmer for $N = 0, 1$ og 2 . Vi forsøker et induksjonsbevis, og antar at E_N er korrekt for N . E_{N+1} blir da

$$E_{N+1} = P_k \cdot \left\{ \left[1 - (1 - P_o)^{N+1} \cdot (1 - P_d)^{M-N-1}\right] - \left(\frac{P_o}{P_k \cdot P_d}\right) \left(\frac{N+1}{M-N}\right) \cdot E_N \right\} \quad (\text{A.11})$$

Det første leddet i den ytterste parentesen kjenner vi igjen som sannsynligheten for at noe detekteres og velges ut for angrep, enten et reelt eller et utslått mål, når det er $N+1$ utslåtte mål i målområdet. For at E_{N+1} skal være korrekt, må da det siste leddet være sannsynligheten for å oppdage og velge som mål et utslått mål når det er $N+1$ utslåtte mål i målområdet. Kan vi bevise denne sammenhengen, er uttrykket for E_N bevist. La oss betegne sannsynligheten for å detektere og velge ut et reelt mål for D_R , og sannsynligheten for å detektere og velge ut et allerede utslått mål for D_U . Ettersom $D = E/P_k$, må vi da bevise sammenhengen

$$D_{U,N+1} = \frac{P_o}{P_d} \cdot \frac{N+1}{M-N} \cdot D_{R,N} \quad (\text{A.12})$$

La oss se på en alternativ måte å beregne E_N på. Søkeren skanner gjennom alle målene i en bestemt rekkefølge. I denne rekkefølgen kan et bestemt, ikke utslått mål ha 0 til N utslåtte mål og 0 til $(M-N-1)$ reelle mål foran seg. La oss si at det er i utslåtte mål og j reelle mål foran dette ene bestemte målet i skannerekkefølgen. Sannsynligheten for at dette målet skal bli detektert og valgt som mål, er da

$$D_{R,N,i,j} = P_d \cdot (1 - P_o)^i \cdot (1 - P_d)^j \quad (\text{A.13})$$

Antall kombinasjoner som gir i utslåtte mål og j reelle mål før vårt bestemte mål, er

$$A_{i,j} = \binom{i+j}{i} \quad (\text{A.14})$$

mens det totale antall måter å kombinere $M - N$ reelle og N utslåtte mål på, er

$$A_{\text{tot}} = \binom{M}{N} \quad (\text{A.15})$$

Totalt finner man da at sannsynligheten for å detektere og velge ut et reelt mål når det er M mål totalt, hvorav N er utslått, er

$$D_{R,N} = \frac{P_d}{\binom{M}{N}} \cdot \sum_{i=0}^N \left\{ \sum_{j=0}^{M-N-1} \left[\binom{i+j}{i} \cdot (1 - P_o)^i \cdot (1 - P_d)^j \right] \right\} \quad (\text{A.16})$$

Vi har her valgt en litt annen angrepsmåte enn tidligere, men ligning A.16 og ligningene som førte frem til at vi gjettet på ligning A.10, uttrykker den samme sannsynligheten. Grunnen til at det er valgt to forskjellige fremgangsmåter, er at den første fremgangsmåten førte til resultater som ga oss grunnlag for å gjette på ligning A.10, mens ligning A.16 trengs for å bevise at ligning A.10 er korrekt.

La oss endre situasjonen litt, og anta at det er $N+1$ utslåtte mål, og dermed $M-N-1$ ikke utslåtte mål i søkeområdet. Vi ser så på et bestemt, allerede utslått mål, og dets nummer i skannerekkefølgen. Det kan ha 0 til N utslåtte og 0 til $(M-N-1)$ reelle mål foran seg. Dersom det er i utslåtte og j reelle mål tidligere i skannerekkefølgen, er sannsynligheten for at dette målet blir valgt ut gitt ved

$$D_{U,N+1,i,j} = P_o \cdot (1 - P_o)^i \cdot (1 - P_d)^j \quad (\text{A.17})$$

Antall kombinasjoner som gir i utslåtte og j reelle mål før vårt bestemte mål, er

$$A_{i,j} = \binom{i+j}{i} \quad (\text{A.18})$$

mens det totale antall måter å kombinere $M-N-1$ reelle og $N+1$ utslåtte mål på, er

$$A_{tot} = \binom{M}{N+1} = \frac{M!}{(M-N-1)!(N+1)!} = \frac{M-N}{N+1} \cdot \binom{M}{N} \quad (\text{A.19})$$

Totalt finner vi da at sannsynligheten for å detektere og velge ut et allerede utslått mål når det er M mål totalt, hvorav $N+1$ er utslått, er

$$D_{U,N+1} = \frac{P_o}{\frac{M-N}{N+1} \cdot \binom{M}{N}} \cdot \sum_{i=0}^N \left\{ \sum_{j=0}^{M-N-1} \left[\binom{i+j}{i} \cdot (1-P_o)^i \cdot (1-P_d)^j \right] \right\} \quad (\text{A.20})$$

Som vi ser, er

$$D_{U,N+1} = \frac{P_o}{P_d} \cdot \frac{N+1}{M-N} \cdot D_{R,N}, \quad (\text{A.21})$$

som var det vi skulle bevise.

Vi har her vist at sannsynligheten for å slå ut et reelt mål når det finnes N utslåtte og $M-N$ reelle mål i søkeområdet til en granat, under de gitte forutsetningene, er gitt ved

$$E_N = P_k \cdot \sum_{i=0}^N \left\{ \left(-\frac{P_o}{P_d} \right)^{N-i} \cdot \frac{\binom{M}{i}}{\binom{M}{N}} \cdot \left[1 - (1-P_o)^i \cdot (1-P_d)^{M-i} \right] \right\} \quad (\text{A.22})$$

For $P_o=0$ blir dette uttrykket et 0^0 -uttrykk. Man må derfor se på grenseverdien når P_o går mot null. Da blir $\left(-\frac{P_o}{P_d} \right)^0 = 1$, og vi får det korrekte svaret.

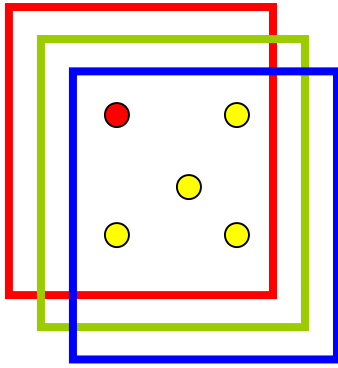
A.2 Sammenfallende søkeområder

I forrige avsnitt fant vi likningene som beskriver effekten av én granat som har et søkeområde med M mål, hvorav N er utslått. I dette kapitlet beskrives sannsynligheten for å slå ut et gitt antall mål dersom man skyter totalt G granater mot et målområde med M mål, dersom alle målene er innenfor søkeområdet til alle granatene. Sannsynligheten for at det er slått ut eksakt N mål etter at det er skutt med G granater, betegnes $P_G(N)$. Nok en gang antas det at rekkefølgen til en skannende søker er tilfeldig (for stirrende søkere er ikke dette noe problem), slik at effektene funnet i kapittel A.1 gjelder. Effekten til en granat som ser N utslåtte mål og $M-N$ reelle mål, ble i kapittel A.1 betegnet E_N .

Når man starter skytingen, er det ingen utslåtte mål i området. Sannsynligheten for å slå ut et mål med den første granaten er derfor E_0 . Sannsynligheten for å *ikke* slå ut et mål, er $1-E_0$. Vi får altså at

$$P_1(0) = 1 - E_0 \quad (\text{A.23})$$

$$P_1(1) = E_0 \quad (\text{A.24})$$



Figur A-1: Ett av de fem målene er slått ut

For granat nummer to blir det noe mer komplisert, se Figur A-1. Målet som er markert som rødt, kan enten være slått ut eller ikke, slik at det enten er M eller $M-1$ reelle mål i søkeområdet (I figuren er $M=5$). Sannsynligheten for at det er M reelle mål i søkeområdet er $P_1(0)$, mens sannsynligheten for at det er $M-1$ reelle mål og ett utslått mål er $P_1(1)$. Effekten til granat nr. 2 er E_0 dersom det er null utslåtte mål i søkeområdet, og E_1 dersom det er ett utslått mål i søkeområdet. Etter granat 2 ser situasjonen slik ut:

$$P_2(0) = (1 - E_0)^2 \quad (\text{A.25})$$

$$P_2(1) = (1 - E_0) \cdot E_0 + E_0 \cdot (1 - E_1) \quad (\text{A.26})$$

$$P_2(2) = E_0 \cdot E_1$$

Generelt kan man se at når det er skutt G granater, blir sannsynlighetene

$$P_G(0) = (1 - E_0)^G \quad (\text{A.27})$$

$$P_G(1) = E_0 \cdot \sum_{i=0}^{G-1} \left\{ (1 - E_0)^i \cdot (1 - E_1)^{(G-1-i)} \right\} \quad (\text{A.28})$$

$$P_G(2) = E_0 \cdot E_1 \cdot \sum_{i=0}^{G-2} \left\{ (1 - E_0)^i \cdot \sum_{j=0}^{G-2-i} \left\{ (1 - E_1)^j \cdot (1 - E_2)^{(G-2-i-j)} \right\} \right\} \quad (\text{A.29})$$

$$P_G(3) = E_0 \cdot E_1 \cdot E_2 \cdot \sum_{i=0}^{G-3} \left\{ (1 - E_0)^i \cdot \sum_{j=0}^{G-3-i} \left\{ (1 - E_1)^j \cdot \sum_{k=0}^{G-3-i-j} \left\{ (1 - E_2)^k \cdot (1 - E_3)^{(G-3-i-j-k)} \right\} \right\} \right\} \quad (\text{A.30})$$

og så videre. Dette er geometriske rekker, som det er mulig å regne ut eksakt. Dette gir

$$P_G(0) = (1 - E_0)^G \quad (\text{A.31})$$

$$P_G(1) = E_0 \cdot \left\{ \frac{(1 - E_0)^G}{(E_1 - E_0)} + \frac{(1 - E_1)^G}{(E_0 - E_1)} \right\} \quad (\text{A.32})$$

$$P_G(2) = E_0 \cdot E_1 \cdot \left\{ \frac{(1-E_0)^G}{(E_1-E_0) \cdot (E_2-E_0)} + \frac{(1-E_1)^G}{(E_0-E_1) \cdot (E_2-E_1)} + \frac{(1-E_2)^G}{(E_0-E_2) \cdot (E_1-E_2)} \right\} \quad (\text{A.33})$$

og så videre. Dette gir grunnlag for å gjette på den generelle formelen

$$P_G(N) = \left(\prod_{i=0}^{N-1} E_i \right) \cdot \sum_{i=0}^N \left\{ \frac{(1-E_i)^G}{\prod_{j=0, j \neq i}^N (E_j - E_i)} \right\} \quad (\text{A.34})$$

La oss anta at denne formelen er riktig for $P_G(N)$ og $P_G(N+1)$. Hva blir da uttrykket for $P_{G+1}(N+1)$? Det er to måter $G+1$ granater kan slå ut $N+1$ mål på: Enten kan de G første granatene slå ut N mål og den siste granaten slå ut ett mål, eller de G første granatene kan slå ut $N+1$ mål og den siste granaten ikke slå ut et mål. Dette kan skrives som

$$P_{G+1}(N+1) = E_N \cdot P_G(N) + (1-E_{N+1}) \cdot P_G(N+1)$$

La oss sette inn $G+1$ og $N+1$ i likning 3.35:

$$\begin{aligned} P_{G+1}(N+1) &= \left(\prod_{i=0}^N E_i \right) \cdot \sum_{i=0}^{N+1} \left\{ \frac{(1-E_i)^{G+1}}{\prod_{j=0, j \neq i}^{N+1} (E_j - E_i)} \right\} \\ &= \left(\prod_{i=0}^N E_i \right) \cdot \left\{ \frac{(1-E_{N+1})^{G+1}}{\prod_{j=0}^N (E_j - E_{N+1})} + \sum_{i=0}^N \left\{ \frac{(1-E_i) \cdot (1-E_i)^G}{(E_{N+1} - E_i) \cdot \prod_{j=0, j \neq i}^N (E_j - E_i)} \right\} \right\} \\ &= \left(\prod_{i=0}^N E_i \right) \cdot \left\{ \frac{(1-E_{N+1}) \cdot (1-E_{N+1})^G}{\prod_{j=0}^N (E_j - E_{N+1})} + \sum_{i=0}^N \left\{ \left(1 + \frac{(1-E_{N+1})}{(E_{N+1} - E_i)} \right) \cdot \frac{(1-E_i)^G}{\prod_{j=0, j \neq i}^N (E_j - E_i)} \right\} \right\} \quad (\text{A.35}) \\ &= E_N \cdot \left(\prod_{i=0}^{N-1} E_i \right) \cdot \sum_{i=0}^N \left\{ \frac{(1-E_i)^G}{\prod_{j=0, j \neq i}^N (E_j - E_i)} \right\} + (1-E_{N+1}) \cdot \left(\prod_{i=0}^N E_i \right) \cdot \sum_{i=0}^{N+1} \left\{ \frac{(1-E_i)^G}{\prod_{j=0, j \neq i}^{N+1} (E_j - E_i)} \right\} \\ &= E_N \cdot P_G(N) + (1-E_{N+1}) \cdot P_G(N+1) \end{aligned}$$

som vi skal. Vi har allerede funnet ut at formelen er riktig for alle G , $N=1$ og alle G , $N=2$ ($N=0$ er et spesialtilfelle). Vi må nå se på formelen for $G=0$, alle $N>0$). Vi vet at dette skal gi null; det er umulig å slå ut noen mål med null granater. Vi må altså vise at

$$P_0(N) = \left(\prod_{i=0}^{N-1} E_i \right) \cdot \sum_{i=0}^N \left\{ \frac{1}{\prod_{j=0, j \neq i}^N (E_j - E_i)} \right\} = 0 \quad (\text{A.36})$$

for $N > 0$. Det første leddet er opplagt ikke generelt lik null, altså må det andre leddet være det. Vi innfører

$$S_i = \frac{1}{\prod_{j=0, j \neq i}^N (E_j - E_i)} \quad (\text{A.37})$$

og skal da vise at

$$\sum_{i=0}^N S_i = 0 \quad (\text{A.38})$$

La oss se litt nærmere på S_0 . Vi forsøker med delbrøksoppspalting:

$$S_0 = \frac{1}{\prod_{l=1}^N (E_l - E_0)} = \sum_{j=1}^N \frac{A_j}{(E_j - E_0)} \quad (\text{A.39})$$

der A_j er konstanter som må bestemmes. Ved å multiplisere med $(E_k - E_0)$ får vi

$$\frac{1}{\prod_{l=1, l \neq k}^N (E_l - E_0)} = A_k + (E_k - E_0) \cdot \sum_{j=1, j \neq k}^N \frac{A_j}{(E_j - E_0)} \quad (\text{A.40})$$

Denne likningen skal være oppfylt for alle E_0 . Spesielt skal den være oppfylt for

$$\lim_{E_0 \rightarrow E_k} (E_k - E_0) \quad (\text{A.41})$$

som gir

$$A_k = \frac{1}{\prod_{l=1, l \neq k}^N (E_l - E_k)} \quad (\text{A.42})$$

Vi benytter sammenhengen

$$\frac{1}{(E_j - E_0)} = -\frac{1}{(E_0 - E_j)} \quad (\text{A.43})$$

og får

$$S_0 = \sum_{j=1}^N \left\{ -\frac{1}{\prod_{l=0, l \neq j}^N (E_l - E_j)} \right\} \quad (\text{A.44})$$

Vi ser at

$$S_0 = -\sum_{i=1}^N S_i \quad (\text{A.45})$$

og derfor

$$\sum_{i=0}^N S_i = 0 \quad (\text{A.46})$$

slik vi skulle vise.

Vi har vist at (A.34) er riktig for alle G , $N=1$, og alle G , $N=2$, samt for $G=0$, alle $N>0$. Vi har også vist at dersom ligning (A.34) er riktig for $P_G(N)$ og $P_G(N+1)$, er den riktig for $P_{G+1}(N+1)$. Dobbel induksjon gir da at ligning (A.34) er riktig for alle G og alle $N>0$.

$N=0$ er et spesialtilfelle, og hvis man prøver å benytte ligning (A.34) på denne situasjonen, får man et uttrykk med produktet av null ledd. Dersom man definerer produktet av null ledd til å være lik 1, blir ligning (A.34) riktig også for dette tilfellet.

B TILFELDIGE OG KVASITILFELDIGE TALL

I forbindelse med den deterministiske metoden ble det sett på forskjellige måter å generere treffpunkter på. I den stokastiske modellen ble disse punktene trukket tilfeldig fra en bestemt fordeling, mens de i den deterministiske metoden ble valgt på en strukturert måte. Tall som er generert på en slik strukturert måte med tanke på å oppføre seg som tilfeldige tall, men med ”bedre” egenskaper, kalles *kvasitilfeldige tall*. Ettersom vi gikk bort fra den deterministiske metoden, kan man kanskje lure på hva poenget er med dette appendikset. Poenget er at arbeidet med denne modellen, sammen med noe arbeid utført i ettertid, gir verdifull innsikt i oppførselen til tilfeldige og kvasitilfeldige tall, som brukere og programmerere av simuleringprogrammer kan finne meget interessant. Vi vil i dette appendikset anta at dersom vi har flere dimensjoner, er verdien i de forskjellige dimensjonene uavhengige.

B.1 Tilfeldige og ”pseudotilfeldige” tall

”Ekte” tilfeldige tall stammer fra fysiske prosesser som radioaktivitet, der for eksempel tiden mellom to fisjonshendelser er tilfeldig. Slike tall følger visse sannsynlighetsfordelinger, der de mest interessante for vårt formål er normalfordelingen og den homogene fordelingen. Har man tall fra en homogen fordeling, kan man i prinsippet generere tall som tilsvarer en hvilken som helst fordelingsfunksjon.

”Pseudotilfeldige” tall er en betegnelse på tilfeldige tall slik de genereres av for eksempel en datamaskin. Gitt alle forutsetninger forut for genereringen av slike tall, kan man i prinsippet forutsi hvilke tall som vil genereres. Disse er derfor ikke ”ekte” tilfeldige tall. Genererer man mange nok tall i en sekvens, kan man også risikere at sekvensen vil begynne å gjenta seg selv. Imidlertid er algoritmen for generering av tilfeldige tall i MatLab så god at dette for våre formål ikke er begrensende, og den er derfor like god som ”ekte” tilfeldige tall. Dette er ikke nødvendigvis tilfelle for alle andre programmeringsspråk!

B.2 Kvasitilfeldige tall

Kvasitilfeldige tall er, i motsetning til ”ekte” tilfeldige tall og pseudotilfeldige tall, ikke tilfeldige i det hele tatt. De følger derimot bestemte systemer. I en éndimensjonal, homogen fordeling vil gode kvasitilfeldige tall ganske enkelt søke å ligge med jevnest mulig mellomrom på tallinja. For flere dimensjoner blir det mer komplisert, men én av egenskapene til en ”god” kvasitilfeldig sekvens for en flerdimensjonal homogen fordeling, er at alle flerdimensjonale volum som er like store, skal inneholde om lag like mange punkter.

Det er kjent fra sentralgrenseteoremet at standardavviket til middelveien i en simulering basert på ekte tilfeldige eller pseudotilfeldige tall (heretter kun kalt tilfeldige tall) vil falle som $\frac{1}{\sqrt{N}}$. Poenget med å benytte kvasitilfeldige tall i stedet, er å klare å generere sekvenser der

standardavviket i svaret faller raskere enn dette. Med en ”hard kant” er den teoretiske grensen noe som faller som $\frac{1}{N}$; man får ikke bedre oppløsning enn man har antall punkter. En ”hard kant” innebærer at man bare er interessert i hvor mange punkter som befinner seg i et område, slik at alle punktene har lik vekt.

De kvasitilfeldige sekvensene som ble forsøkt benyttet i den deterministiske modellen var på noen måter dårlige sekvenser. I dette appendikset blir de sammenliknet med tilfeldige tall og en bedre kvasitilfeldig sekvens.

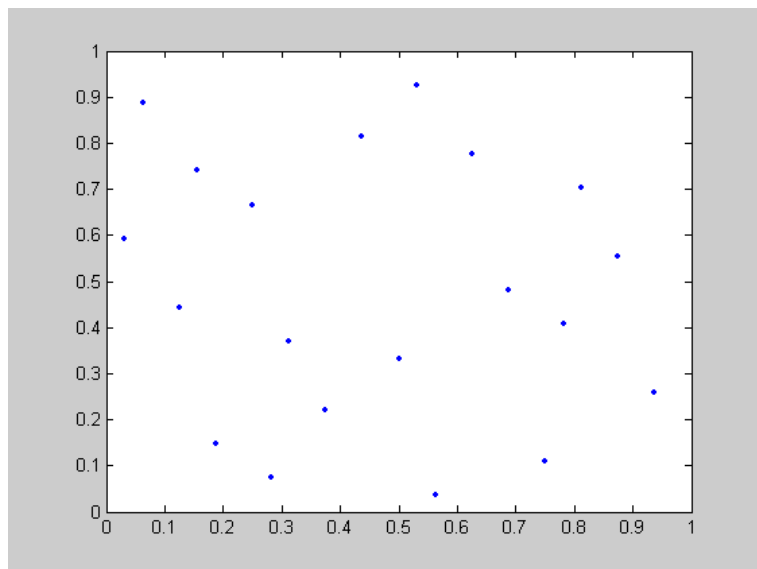
Et eksempel på en kvasitilfeldig sekvens er *Haltons sekvens*. Den fungerer på følgende måte: Anta at vi skal generere T punkter i n dimensjoner. Da trenger vi n forskjellige primtall P_i , typisk velger man de n første primtallene. Så skriver man tallene 1 til og med T i base P_i . Deretter snus sifrene, slik at det første sifferet kommer sist og det siste kommer først. Foran dette setter man så null og komma. Så regnes tallene tilbake fra base P_i til desimaltall. Vi har nå T n -dimensjonale tall. Her følger et eksempel for 20 tall i 2 dimensjoner.

De to første primtallene er 2 og 3, så vi må først skrive tallene 1-20 i base 2 og 3:

Base 10	Dimensjon 1 (Base 2)			Dimensjon 2 (Base 3)		
	Base 2	Snudd	Base 10	Base 3	Snudd	Base 10
1	1	0,1	0,5	1	0,1	0,3333
2	10	0,01	0,25	2	0,2	0,6667
3	11	0,11	0,75	10	0,01	0,1111
4	100	0,001	0,125	11	0,11	0,4444
5	101	0,101	0,625	12	0,21	0,7778
6	110	0,011	0,375	20	0,02	0,2222
7	111	0,111	0,875	21	0,12	0,5556
8	1000	0,0001	0,0625	22	0,22	0,8889
9	1001	0,1001	0,5625	100	0,001	0,0370
10	1010	0,0101	0,3125	101	0,101	0,3704
11	1011	0,1101	0,8125	102	0,201	0,7037
12	1100	0,0011	0,1875	110	0,011	0,1481
13	1101	0,1011	0,6875	111	0,111	0,4815
14	1110	0,0111	0,4375	112	0,211	0,8148
15	1111	0,1111	0,9375	120	0,021	0,2593
16	10000	0,00001	0,03125	121	0,121	0,5926
17	10001	0,10001	0,53125	122	0,221	0,9259
18	10010	0,01001	0,28125	200	0,002	0,0741
19	10011	0,11001	0,78125	201	0,102	0,4074
20	10100	0,00101	0,09375	202	0,202	0,7407

Tabell 6-1: Kvasitilfeldige tall etter Haltons sekvens

Resultatet er vist i Figur B-2. Det er enkelt å forstå logikken bak metoden. Etter hvert som man får høyere tall, øker antall siffer. Når man så ”snur” tallet, tilsvarer dette økt oppløsning.

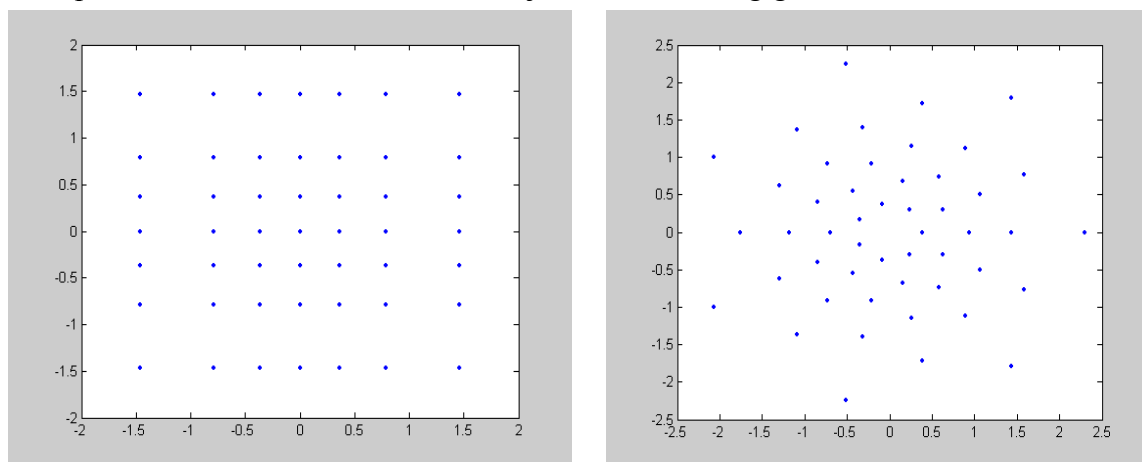


Figur B-1: Tilfeldige punkter generert vha Haltons sekvens

B.3 Metodene fra kapittel 4.1

Tallene i forrige avsnitt representerer en homogen fordeling, men det er ingen problemer med å gjøre dem om til tall som representerer en normalfordeling. I metoden beskrevet i kapittel 4.1 ble det gjort forsøk på å generere tall som skulle representere en normalfordeling. Den ene metoden genererte tall ut fra en kartesisk tankegang, den andre fra en polar tankegang. Den kartesiske måten fungerte omtrent slik:

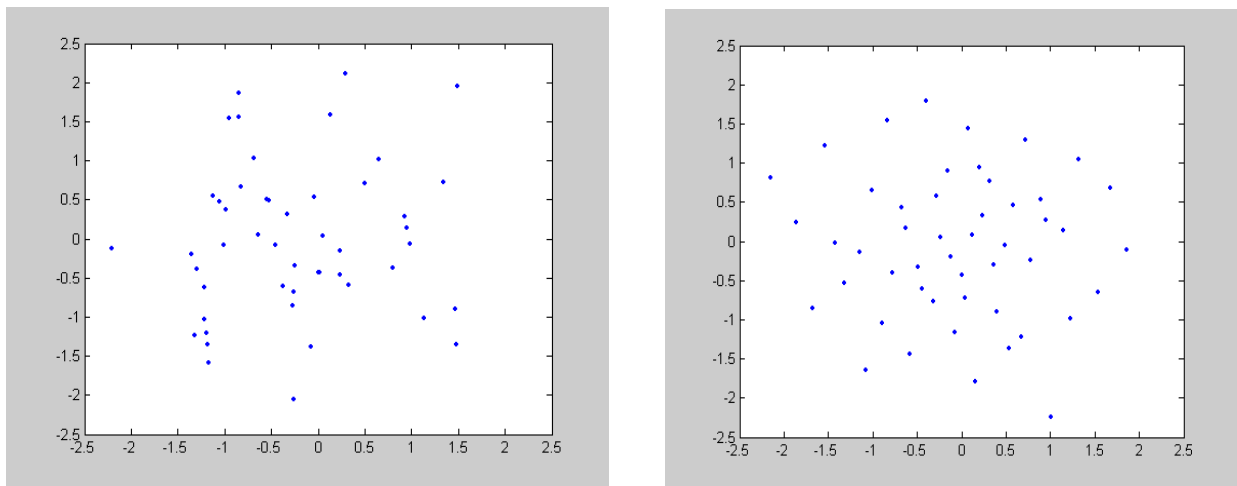
Generer p punkter langs én dimensjon som er slik at sannsynligheten for å havne mellom hvert punkt er nøyaktig like stort. Disse pares med tilsvarende tall langs neste dimensjon, slik at man får p^n punkter i n dimensjoner. I to dimensjoner blir det altså seende ut som Figur B-2 for 49 punkter. Dette forutsetter at dimensjonene er uavhengige.



Figur B-2: Normalfordelte punkter vha henholdsvis kartesisk og polar metode

Den polare metoden antar at det er et partall antall dimensjoner. Så genererer den punkter i to og to dimensjoner. I to dimensjoner genererer den p^2 punkter. Først lages det p verdier radielt, som er slik at sannsynligheten for å havne i en sirkelflate mellom to og to radier er like stor. Så genereres det tangentielle punkter med lik avstand. Annenhver sirkel forskyves så, slik at ikke alle punktene ligger på ”stråler”. I to dimensjoner ser det ut som Figur B-2 for 49 punkter.

For sammenlikning vises 49 punkter trukket tilfeldig fra en normalfordeling, og 49 punkter generert ut fra Haltons sekvens og transformert til en normalfordeling, i Figur B-3.



Figur B-3: Normalfordelte punkter vha henholdsvis stokastisk metode og Haltons sekvens

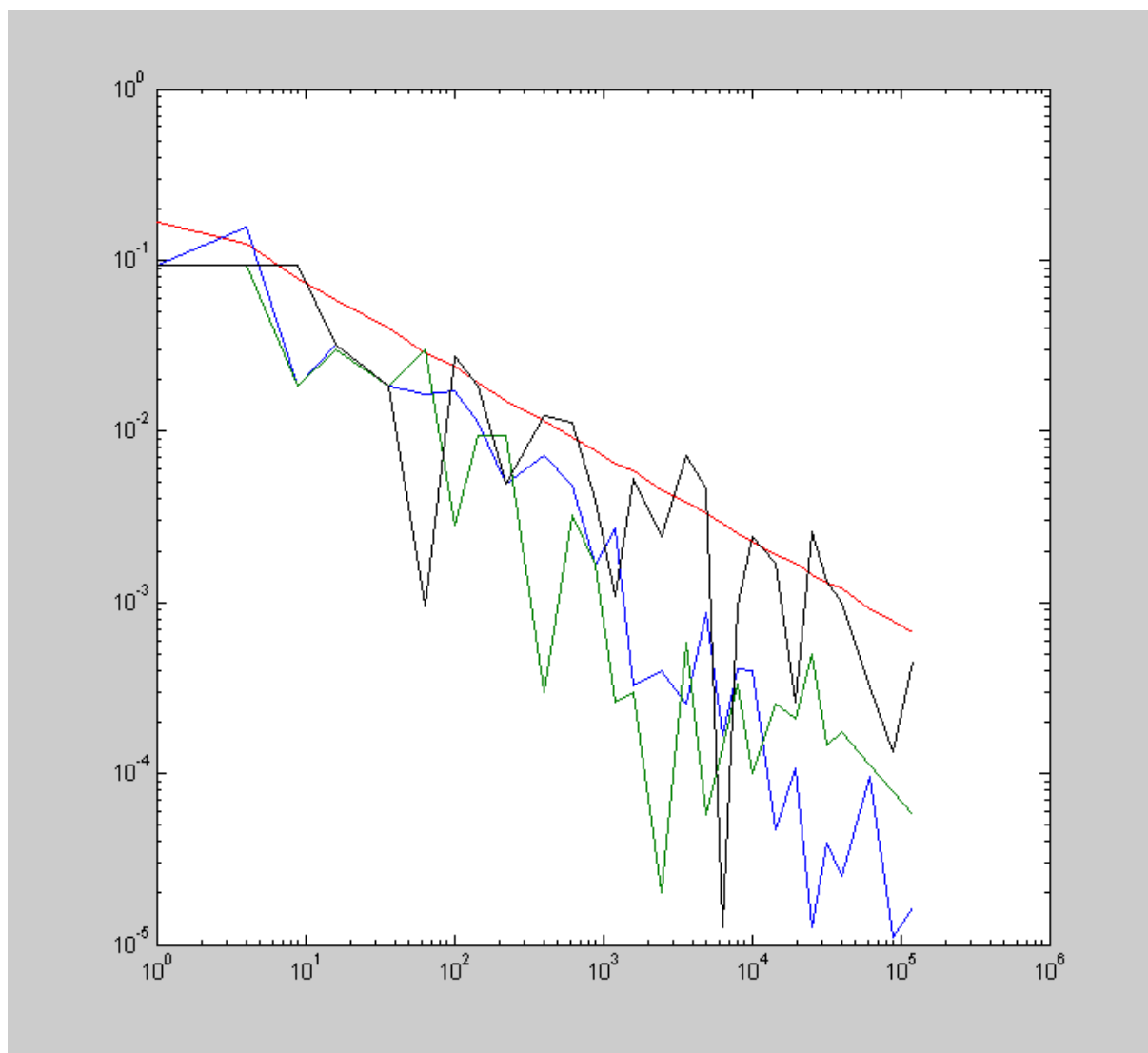
Det er viktig å huske på at metoden beskrevet i avsnitt 4.1 egentlig trenger punkter i fire dimensjoner – to dimensjoner for bomavstand, og to dimensjoner for spredning (tar man hensyn til overkillmekanismer, trenger man faktisk $2+2G$ dimensjoner, der G er antall granater. Overkillmekanismer ble ikke tatt hensyn til i metoden fra kapittel 4.1). Det er til en viss grad dette som forårsaker problemene med disse metodene – de er akseptable i to dimensjoner, i alle fall når det integreres over sirkulære områder som ikke er sentrert i origo.

B.4 Sammenlikning av metoder

La oss sammenlikne integrasjon ved hjelp av tilfeldige tall, Haltons sekvens, polar og kartesisk metode. Vi ser på en fordeling som er uavhengig i x- og y-retning, og normalfordelt med middelværdi lik 0 og standardavvik lik 1 i både x- og y-retning. Først ser vi på et tilfelle der den kartesiske metoden er spesielt svak: Vi integrerer over et rektangel. La oss se på området $-2 < x < -1$ samtidig som $-1 < y < +1$. Den virkelige sannsynligheten for å havne innenfor dette intervallet er gitt ved:

$$P_{real} = \left(\int_{-2}^{-1} N_x(0,1) dx \right) \cdot \left(\int_{-1}^{1} N_y(0,1) dy \right) = 0,1359 \cdot 0,6827 = 0,0928$$

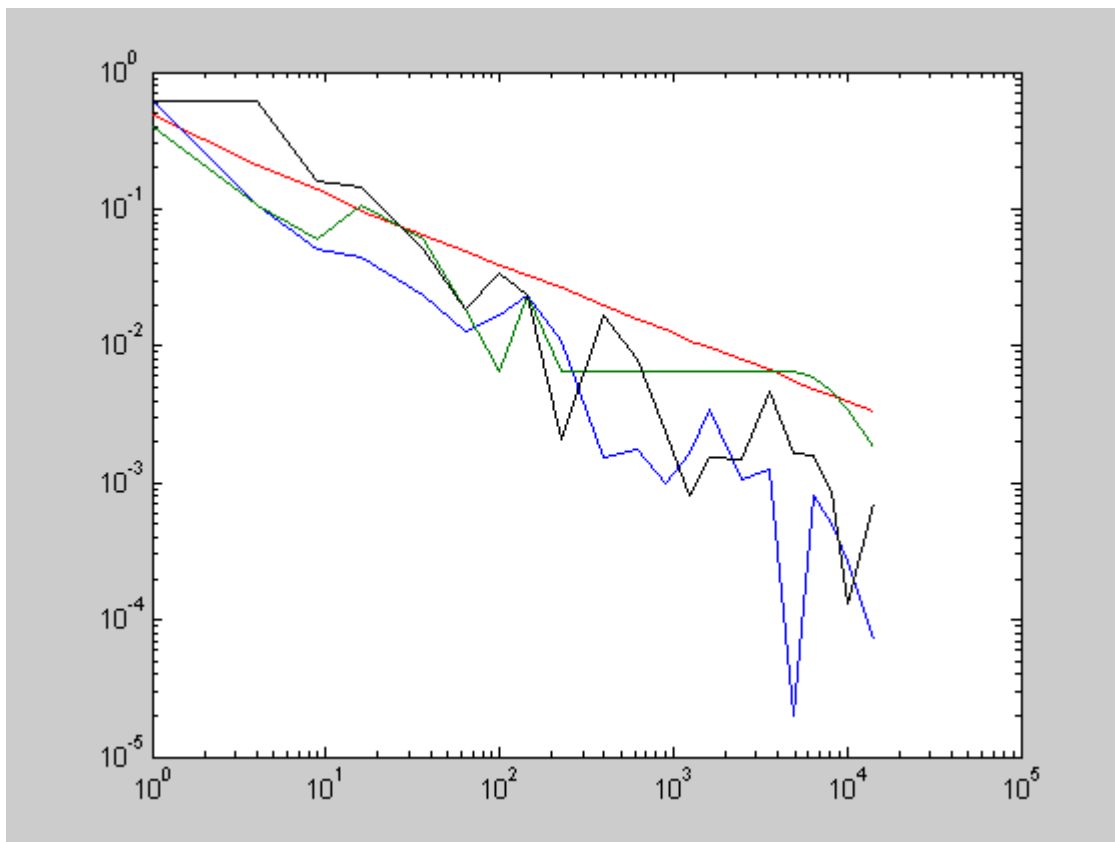
Der N betyr normalfordelingen. Vi genererer p punkter med hver av metodene, og ser hvordan avviket faller med p for de forskjellige metodene. Vi bruker bare kvadrattall, da begge metodene fra kapittel 4.1 krever dette. For den stokastiske metoden må vi gjøre testen flere ganger, da avviket vil variere fra gang til gang. Vi utfører derfor 1000 forsøk og beregner standardavviket for hver p . Som vi ser i Figur B-4, er den polare metoden og Haltons metode bedre enn den stokastiske og den kartesiske metoden. "Bedre" betyr i denne sammenheng selvsagt lavere avvik fra den reelle verdien.



Figur B-4: Sammenlikning av metoder. Figuren viser avvik fra korrekt verdi som funksjon av antall punkter. For stokastisk metode er det tatt gjennomsnitt av 1000 simuleringer. *Rød*: Stokastisk. *Blå*: Haltons sekvens. *Grønn*: Polar metode. *Sort*: Kartesisk metode. Legg merke til at det er brukt log-log skala.

Vi ser nå på et eksempel der den polare metoden får problemer, nemlig en sirkel med sentrum i origo. Vi gjør den samme øvelsen, og ser i Figur B-5 at den stokastiske og den polare metoden peker seg ut som de dårligste. Selv om dette er et eksempel der den polare metoden

har problemer, gjør den det ikke dårligere enn den stokastiske. For begge eksemplene er det viktig å huske på at vi her kun har sett på et todimensjonalt problem. I metoden fra kapittel 4.1 hadde vi et firedimensjonalt problem. For økende antall dimensjoner blir både den polare og den kartesiske metoden dårligere, og da spesielt den kartesiske. Haltons sekvens vil fortsatt gi vesentlig bedre resultater enn en stokastisk metode, så sant vi ikke har **for** mange dimensjoner (Haltons sekvens blir dårligere for mange dimensjoner). Vi ser fra våre eksempler at både den polare og den kartesiske metoden for to dimensjoner gjør det like bra som den stokastiske metoden selv for dårlige tilfeller, de faller som $\frac{1}{\sqrt{N}}$. For d dimensjoner vil de for tilsvarende dårlige tilfeller falle som $N^{-1/d}$. Dette, sammen med enkelte andre faktorer som ikke er berørt her, gjorde at metoden fra kapittel 4.1 ble dårligere enn den stokastiske metoden.



*Figur B-5: Sammenlikning av metoder. Figuren viser avvik fra korrekt verdi som funksjon av antall punkter. For stokastisk metode er det tatt gjennomsnitt av 1000 simuleringer. **Rød:** Stokastisk. **Blå:** Haltons sekvens. **Grønn:** Polar metode. **Sort:** Kartesisk metode. Legg merke til at det er brukt log-log skala.*

Litteratur

- (1) PRESS W H, TEUKOLSKY S A, VETTERUNG W T, FLANNERY B P (1992):
Numerical recipes in Fortran, Cambridge University Press, Australia, 266 - 319.

FORDELINGSLISTE

FFIBM **Dato:** 23. april 2003

RAPPORTTYPE (KRYSS AV) <input checked="" type="checkbox"/> RAPP <input type="checkbox"/> NOTAT <input type="checkbox"/> RR		RAPPORT NR. 2003/00084	REFERANSE FFIBM/798/139	RAPPORTENS DATO 23. april 2003
RAPPORTENS BESKYTTELSESGRAD UGRADERT		ANTALL EKS UTSTEDT 22	ANTALL SIDER 54	
RAPPORTENS TITTEL Metoder for effektberegning av smart artilleriammunisjon		FORFATTER(E) HALSØR Marius, KVIFTE Lars		
FORDELING GODKJENT AV FORSKNINGSSJEF Stein Grinaker		FORDELING GODKJENT AV AVDELINGSSJEF: Jan Ivar Botnan		

EKSTERN FORDELING

INTERN FORDELING

ANTALL	EKS NR	TIL	ANTALL	EKS NR	TIL
1		KAMPUKS / SSFA	2		FFI-Bibl
1		v/ Oblt N A Skaret	1		Adm direktør/stabssjef
1		v/ Maj J Moen	1		FFIE
1		FLO / LAND	1		FFISYS
1		v/ Oblt S Tinholt	1		FFIBM
1		v/ Maj S Skjørholm	1		FFIN
1		KAMPUKS / SVI	3		Forfattereksemplarer
1		v/ Kapt K Nygård			ELEKTRONISK FORDELING:
					S Grinaker, BM (SGr)
					E Østevold, BM (EiO)
					A Skjold, BM (ArS)
					O Dullum, BM (OSD)
					L Kvifte, BM (LKv)
					M Halsør, BM (MHa)
					H Ajer, E (HAj)
					N A Sæthermoen, E (NAS)
					E Willassen, N (EiW)
					F Dahl, SYS (FAD)
					N U Andresen, SYS (NUA)
					D Cappelen, BM (Dca)
					E Haarseth, BM (EgH)
					N G Johansson, BM (NGJ)

FFI-K1 Retningslinjer for fordeling og forsendelse er gitt i Oraklet, Bind I, Bestemmelser om publikasjoner for Forsvarets forskningsinstitutt, pkt 2 og 5. Benytt ny side om nødvendig.