

UNIVERSITETET I OSLO

Institutt for informatikk

# Optimering av tynnede arrayer

Anvendt og Industriell  
Matematikk (AIM)

Jon-Fredrik  
Hopperstad

Mai 1998





# Forord

Denne oppgaven er den skriftlige delen av Candidatus Scientiarum graden i Anvendt og Industriell Matematikk ved Institutt for Informatikk, Universitetet i Oslo. Arbeidet begynte i januar 1997 og avsluttes nå mai 1998.

Før dette arbeidet hadde jeg liten kjennskap til array prosessering og romlig signalbehandling generelt. Takket være min veileder, professor Sverre Holm, har jeg fått en sterk interesse for fagfeltet og ønsker å jobbe videre innen relaterte problemstillinger. Det har vært svært inspirerende å arbeide sammen med Sverre, og jeg ønsker å takke ham for den tålmodighet og tillit han har vist meg.

Takk også til min medveileder, stipendiat Andreas Austeng, som har vært til stor hjelp med praktiske problemstillinger. Andreas har hele tiden vist interesse for mitt arbeid og har alltid tid til en prat, enten det er faglig eller sosialt.

Jeg ønsker også å takke Kamran Iranpour, Skjalg Lepsøy og Hermann Lia som har kommet med viktige innspill.

Til sist vil jeg takke min samboer og nærmeste venn, Siri, som har hatt uendelig tålmodighet i denne perioden. Hun har vist interesse for hva jeg holder på med, men samtidig minnet meg på at det finnes et liv utenfor lesesalen. Spesielt takk for at hun ofret påskeferien og tilbrakte den på lesesalen sammen med meg.

Oslo, Mai 1998

---

Jon-Fredrik Hopperstad



# Innhold

<b>1 Innledning</b>	<b>1</b>
1.1 Bølger og arrayer . . . . .	1
1.2 Oppgavens målsetning . . . . .	1
<b>2 Signaler i rom og tid</b>	<b>3</b>
2.1 Koordinatsystemer . . . . .	3
2.2 Ultralydbølger . . . . .	3
2.3 Bølgeligningen . . . . .	5
2.3.1 Planbølger . . . . .	5
2.3.2 Sfæriske bølger . . . . .	5
2.3.3 Egenskaper ved bølgeligningen . . . . .	6
<b>3 Arrayer og strålingsdiagram</b>	<b>7</b>
3.1 Aperture glattingsfunksjonen . . . . .	7
3.2 Romlig Sampling . . . . .	9
3.3 Strålingsdiagrammet . . . . .	10
3.3.1 Hovedloben . . . . .	11
3.3.2 Sidelobene . . . . .	12
3.3.3 Gitterlober . . . . .	13
<b>4 Tynnde arrayer</b>	<b>14</b>
4.1 Egenskaper til tynnde arrayer . . . . .	14
4.2 Stokastiske arrayer . . . . .	18
4.3 Arrayer med betingelser . . . . .	23
4.3.1 Binnede arrayer . . . . .	23
4.3.2 Symmetriske arrayer . . . . .	26
4.3.3 Periodiske arrayer . . . . .	28
<b>5 Coarrayet</b>	<b>30</b>
5.1 Definisjoner . . . . .	30
5.2 Minimum redundans arrayer . . . . .	32

5.3	Minimum hull arrayer . . . . .	33
5.4	Arrayer med hull og redundans . . . . .	36
5.5	Coarrayet til optimale arrayer . . . . .	38
5.5.1	Flathet . . . . .	38
5.5.2	Hullenes plassering . . . . .	39
5.5.3	Redundansenes plassering . . . . .	41
5.6	Coarrayet til arrayer med betingelser . . . . .	42
5.6.1	Periodositeter i elementfordelingen . . . . .	42
5.6.2	Symmetri i elementfordelingen . . . . .	44
5.6.3	Coarrayet til binnede arrayer . . . . .	44
5.7	Arrayer med coarray betingelser . . . . .	46
<b>6</b>	<b>Simulated Annealing</b>	<b>48</b>
6.1	Anvendelser . . . . .	48
6.2	Metoden . . . . .	49
6.2.1	Analogien . . . . .	49
6.2.2	Algoritmen . . . . .	50
6.3	Parametre . . . . .	53
6.3.1	Temperaturen . . . . .	53
6.3.2	Nabokonfigurasjonene . . . . .	57
<b>7</b>	<b>Optimering av 1D arrayer med Simulated Annealing</b>	<b>61</b>
7.1	Implementering . . . . .	61
7.1.1	Beskrive konfigurasjonene . . . . .	61
7.1.2	Stokastiske endringer . . . . .	62
7.1.3	Kostfunksjonen . . . . .	62
7.2	Resultater . . . . .	64
7.2.1	Et lite testproblem . . . . .	64
7.2.2	Et referanseproblem . . . . .	67
7.3	Konklusjon . . . . .	75
<b>8</b>	<b>Optimering av 2D arrayer med Simulated Annealing</b>	<b>76</b>
8.1	Implementering . . . . .	76
8.1.1	Beskrive konfigurasjonene . . . . .	76
8.1.2	Kostfunksjonen . . . . .	78
8.2	Et lite array . . . . .	80
8.2.1	Tidlige resultater . . . . .	80
8.2.2	Nye resultater . . . . .	81
8.3	Et stort array . . . . .	85

<b>9 Oppsummering</b>	<b>88</b>
9.1 Konklusjon . . . . .	88
9.2 Forbedring av algoritme . . . . .	89
9.2.1 Komposisjon av optimale arrayer . . . . .	89
9.2.2 Optimale coarrayer . . . . .	90
<b>A NORSIG artikkell</b>	<b>91</b>
<b>B Simula-kode for å generere alle mulige konfigurasjoner</b>	<b>97</b>
<b>C Matlab-kode</b>	<b>99</b>
C.1 Finne max sidelobenivå og hovedlobebredde . . . . .	99
C.2 Optimering av 1D arrayer med Simulated Annealing og FFT . . . . .	101
C.3 Optimering av 1D arrayer med Simulated Annealing og bidrags-metoden . . . . .	103
C.4 Optimering av 1D og 2D arrayer med Simulated Annealing og bidragsmetoden . . . . .	106



# Kapittel 1

## Innledning

### 1.1 Bølger og arrayer

Bølger er en av de viktigste informasjonsbærerne i hverdagen vår. Ørene oppfatter variasjoner i akustiske bølger og øynene registrerer et bredt spekter av elektromagnetiske bølger. Mennesket har etterhvert utviklet nye sanse-instrumenter for å registrere bølger i andre frekvensintervaller. Radar, sonar, seismikk og ultralyd er områder hvor man benytter et nettverk av sensorer for å registrere de ulike bølgenefeltene. Dette nettverket av sensorer kalles et array. 1D arrayer har sensorene posisjonert i én dimensjon i rommet. Ved fasestyring kan disse arrayene beskrive bølgenefeltet i to romlige dimensjoner. Hvis et slikt array skal være i ro i forhold til mediet det avbilder krever 3D prosessering et 2D fasestyrt array. For å oppnå de samme oppløsningssegenskapene som et 1D array må et 2D array ha kvadratet av antall sensorer til 1D arrayet, typisk 2000–10000 sensorer i medisinsk ultralyd. Fysiske begrensninger gjør at man ikke klarer å produsere arrayer med så mange sensorer. Det er derfor nødvendig å fjerne en god del av sensorene i 2D arrayet. Denne prosessen kalles tynning og er en kjent metode med anvendelser også i radar, sonar og seismikk, men denne oppgaven vil i hovedsak koncentrere seg om arrayer benyttet i ultralyd.

### 1.2 Oppgavens målsetning

Denne oppgaven skal se nærmere på egenskapene til tynnede arrayer. Disse arrayene gir store støybidrag (sidelober) fra enkelte retninger. Derfor vil denne oppgaven koncentrere seg om optimalisering av fjernfeltsresponsen. Optimering av tynnede arrayer er et kombinatorisk problem, slik at det er svært vanskelig å finne globalt optimale løsninger. I denne oppgaven vil man studere hvilke effekter betingelser på elementfordelingen har for fjernfeltsresponsen i forhold til egen-

skapene optimale tynnede arrayer har. I tillegg til dette benyttes Simulated Annealing som er en heuristisk optimeringsmetode til å optimere 1D og 2D arrayer uten noen betingelser på elementfordelingen. Ut fra egenskapene til de globalt optimale arrayene og egenskapene til de optimerte arrayene konkluderer oppgaven med forslag til forbedrede søkerutiner. Men aller først presenteres den mest nødvendige teorien for å forstå oppgavens problemstilling.

# Kapittel 2

## Signaler i rom og tid

I dette kapittelet presenteres de fundamentale begrepene og den underliggende teorien som oppgaven bygger på.

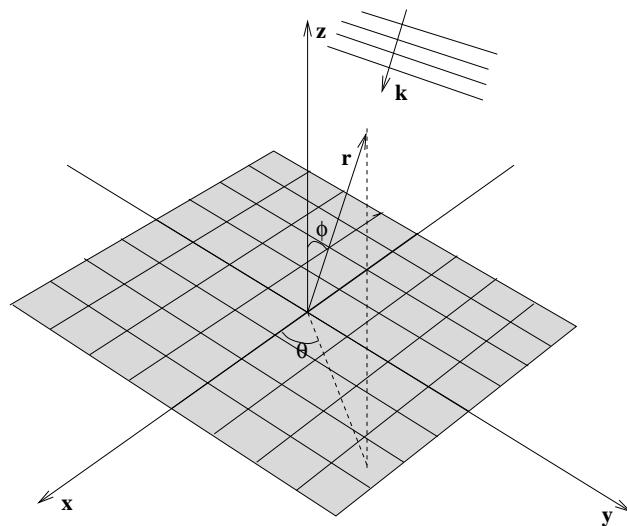
### 2.1 Koordinatsystemer

I frekvens-tid signalbehandling kan signalet representeres med én variabel; tid  $t$  eller frekvens  $\omega$ . I romlig signalbehandling skal man i tillegg bestemme posisjonen  $\vec{x}$  til signalets kilde. Følgelig er signalet  $s(\vec{x}, t)$  en funksjon av rom og tid. Pga. symmetri er det ofte hensiktsmessig å benytte sfæriske koordinater slik som vist i figur 2.1.

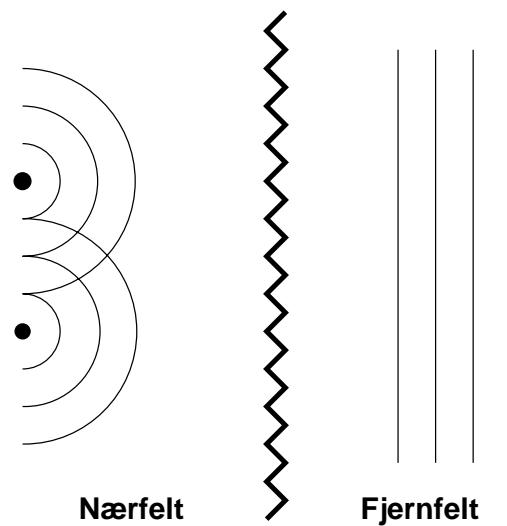
### 2.2 Ultralydbølger

Ultralydbølger er longitudinale trykkbølger med frekvens høyere enn hørbar lyd. I medisinsk ultralyd brukes frekvenser i området 2–10 MHz. Vev har lydhastighet på 1540 m/s, slik at bølgelengden  $\lambda$  blir 0.15–0.75 mm [Holm, 1995]. Bølgene formes ved interferens mellom sfæriske bølger fra transducer elementene. Bølgefronten forplanter seg gjennom mediet som er i kontakt med transduceren ved at molekylene i mediet oscillerer parallelt med bølgefrontens forplantningsretning.

Bestemmelse av posisjonen til refleksjonskilden avhenger av om kilden befinner seg i nærfeltet eller fjernfeltet. Denne grensen defineres som avstanden til bølgefronten når denne avstanden er tilnærmet lik for alle elementene. Dermed vil azimuth vinkel være lik for alle elementene. Hvis man antar et homogent medium kan man bestemme avstanden og retningen til kilden i fjernfeltet, men ikke kildens utstrekning. For å kunne bestemme kildens utstrekning må man derfor variere utbredelsesretningen. I fjernfeltet kan man anta plane bølger, mens i nærfeltet antar man superposisjon av sfæriske bølger, slik figur 2.2 viser.



Figur 2.1: Koordinat-systemet til signalet  $s(r, \phi, \theta, t)$ .  $\phi$ ; azimuth vinkel,  $\theta$ ; elevsjonsvinkel.



Figur 2.2: Bølgefronten i nær- og fjernfeltet.

## 2.3 Bølgeligningen

Alle typer bølger er løsninger av bølgeligningen:

$$\frac{\partial^2 s(\vec{x}, t)}{\partial x^2} + \frac{\partial^2 s(\vec{x}, t)}{\partial y^2} + \frac{\partial^2 s(\vec{x}, t)}{\partial z^2} = \frac{1}{c^2} \frac{\partial^2 s(\vec{x}, t)}{\partial t^2} \quad (2.1)$$

hvor  $s(\vec{x}, t)$  er et skalart felt og  $c$  er bølgehastigheten. Bølgeligningen er utledet fra Maxwells ligninger som beskriver elektromagnetiske bølger.

### 2.3.1 Planbølger

En av de enkleste, men likevel mest anvendte, løsningene av bølgeligningen er monokromatiske (én frekvens) plane bølger. Anta at bølgefeltet  $s(\vec{x}, t)$  er separabelt og kan uttrykkes på kompleks eksponential form:

$$s(x, y, z, t) = A \exp\{j(\omega t - k_x x - k_y y - k_z z)\} \quad (2.2)$$

hvor  $A$  er en kompleks konstant og  $k_x, k_y, k_z$  og  $\omega$  er reelle konstanter med  $\omega \geq 0$ . Innsetting i lign. (2.1) gir betingelsen:

$$k_x^2 + k_y^2 + k_z^2 = \frac{\omega^2}{c^2} \quad (2.3)$$

Man kan så vise (se f.eks. [Johnson and Dudgeon, 1993]) at  $|\vec{k}|$  er bølgetallet og  $\omega$  frekvensen. Bølgelengden  $\lambda = 2\pi/|\vec{k}|$  uttrykker forflyttningen i perioden  $T = 2\pi/\omega$ . Figur 2.3 viser forplantningen av bølgefrontene. Disse ligger i planet  $\vec{k} \cdot \vec{x} = C$  (konstant fase) og står normalt på forplantningsretningen.

### 2.3.2 Sfæriske bølger

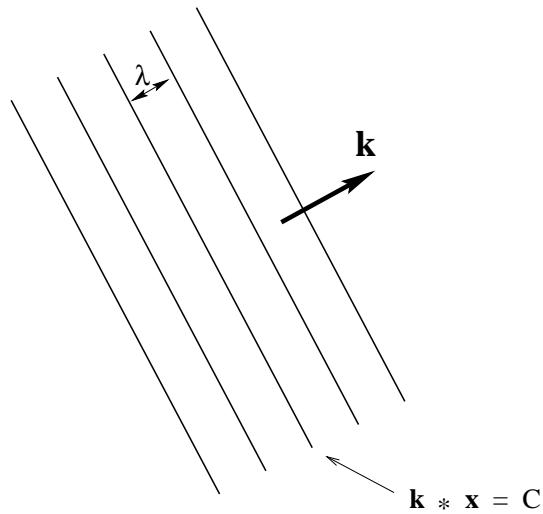
For bølger med sfærisk symmetri, altså  $s(r, \phi, \theta, t) = s(r, t)$ , kan bølgeligningen (2.1) skrives på formen:

$$\frac{1}{r^2} \frac{\partial}{\partial r} \left( r^2 \frac{\partial s}{\partial r} \right) = \frac{1}{c^2} \frac{\partial^2 s}{\partial t^2} \quad (2.4)$$

En av de enkleste løsningene til denne er en monokromatisk sfærisk bølge som forplanter seg fra origo:

$$s(r, t) = \frac{A}{r} \exp\{j(\omega t - kr)\} \quad (2.5)$$

Bølgeformen til denne løsningen går som  $1/r$  i motsetning til planbølger (2.2) hvor den bevares. For store avstander  $r$ , relativt til aperturens utstrekning  $D$ , så er krumningen til bølgefronten så liten at  $s(r, t)$  kan betraktes som en planbølge.



Figur 2.3: Forplantningen til en planbølge.

### 2.3.3 Egenskaper ved bølgeligningen

Enhver lineær kombinasjon av løsninger av bølgeligningen danner en ny løsning.  
Altså vil

$$s(\vec{x}, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} S(\omega) \exp \left\{ j(\omega t - \vec{k} \cdot \vec{x}) \right\} \quad (2.6)$$

hvor  $s(\vec{x}, t)$  er en vilkårlig funksjon. Ligning (2.6) er den inverse Fourier transformen av frekvensfunksjonen  $S(\omega)$ . Siden en superposisjon av komplekse eksponentialer er løsning av bølgeligningen, så gir (2.6) at ethvert signal, uansett bølgeform, tilfredstiller bølgeligningen.

# Kapittel 3

## Arrayer og strålingsdiagram

Et array består av flere sensorer som omdanner bølgeformen til elektriske signaler. Geometrien til sensorene bestemmer hvilke retninger sensorene opptar energi fra. Sensorer med endelig utstrekning kalles aperturer, og kan sammenlignes med vinduer mot bølgefeltet.

### 3.1 Aperture glattingsfunksjonen

Signalet  $z(\vec{x}, t)$  som aperturen produserer avhenger av aperturens geometri  $w(\vec{x})$  og bølgefeltet  $f(\vec{x}, t)$ :

$$z(\vec{x}, t) = w(\vec{x})f(\vec{x}, t) \quad (3.1)$$

Fouriertransformen til signalet  $z(\vec{x}, t)$  er gitt ved konvolusjonen

$$Z(\vec{k}, \omega) = F(\vec{k}, \omega) * W(\vec{k}) \quad (3.2)$$

hvor  $F(\vec{k}, \omega)$  er Fouriertransformen til bølgefeltet;

$$F(\vec{k}, \omega) = \frac{1}{(2\pi)^4} \iint f(\vec{x}, t) \exp\{j(\omega t - \vec{k} \cdot \vec{x})\} d\vec{x} dt \quad (3.3)$$

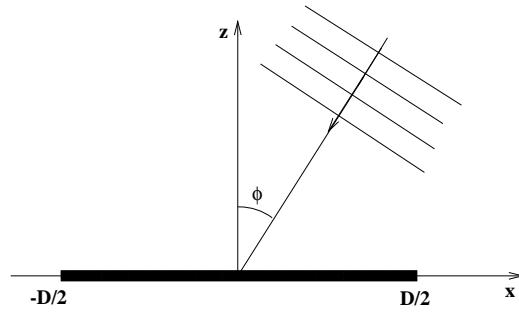
og  $W(\vec{k})$  er *aperture glattingsfunksjonen*:

$$W(\vec{k}) = \int_{-\infty}^{\infty} w(\vec{x}) \exp\{j\vec{k} \cdot \vec{x}\} d\vec{x} \quad (3.4)$$

$W(\vec{k})$  kalles en glattingsfunksjon siden den jevner ut (glatter) responsen til bølgefeltet.

En av de enkleste aperturene er en lineær aperture hvor

$$w(x) = \begin{cases} 1 & , |x| \leq D/2 \\ 0 & , \text{ellers} \end{cases} \quad (3.5)$$

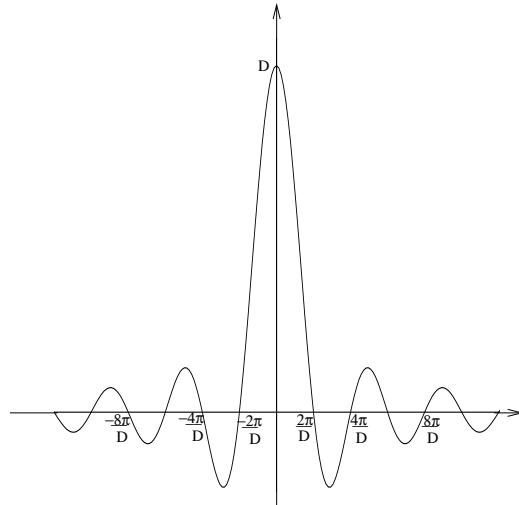


Figur 3.1: Lineær aperture.

slik som figur 3.1 viser. I frekvens-tid signalbehandling så svarer ligning (3.5) til et rektangulært vindu. Aperture glattingsfunksjonen  $W(k)$  er gitt som Fouriertransformen til  $w(x)$ ;

$$\begin{aligned} W(k_x) &= \int_{-D/2}^{D/2} w(x) \exp\{jk_x x\} dx \\ &= \frac{\sin k_x \frac{D}{2}}{k_x / 2} \end{aligned} \quad (3.6)$$

og er vist i figur 3.2.



Figur 3.2: Aperture glattingsfunksjonen  $W(k_x)$  som funksjon av bølgetallet  $k_x$ .

## 3.2 Romlig Sampling

For å utnytte fordelene med digital signalbehandling må aperturen samples til et endelig antall diskrete punkter. I tidsdomenet er dette gitt ved Nyquists samplings-teorem: *Ethvert båndbegrenset signal med frekvenser mindre eller lik  $\omega_0 = 2\pi/T_0$  kan rekonstrueres hvis samplingsfrekvensen  $\omega_s$  tilfredstiller:*

$$\omega_s \geq 2\omega_0 \quad (3.7)$$

Tilsvarende kan man rekonstruere den kontinuerlige apertures glattingsfunksjon hvis

$$\frac{2\pi}{d_s} = k_s \geq 2k_0 \quad (3.8)$$

Relasjonen  $k = 2\pi/\lambda$  gir

$$d_s \leq \frac{\lambda_0}{2} \quad (3.9)$$

Altså *samplings-avstanden  $d_s$  må være under halvparten av den korteste bølgelengden  $\lambda_0$ .*

Den diskrete aperture funksjonen  $W(k)$  er gitt ved

$$W(\vec{k}) = \sum_n w_n e^{j\vec{k} \cdot \vec{x}_n} \quad (3.10)$$

Hvis samplingsavstanden  $d_s$  er konstant (periodisk sampling) forenkles lign. (3.10) til

$$W(k_x, k_y) = \sum_n w_n e^{j(k_x n_x + k_y n_y)d} \quad (3.11)$$

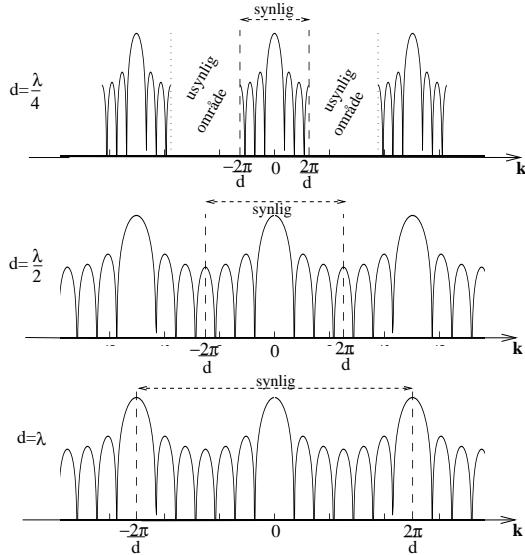
$W(k)$  glatter det samplede bølgefeltet på samme måte som det kontinuerlige tilfellet, men den diskrete aperture funksjonen er periodisk med periode  $k = 2\pi/d$  i motsetning til den kontinuerlige funksjonen. Den diskrete aperture funksjonen til aperturen gitt i lign. (3.5) samplet over  $M$  punkter er

$$W(k) = \frac{\sin \frac{k M d}{2}}{\sin \frac{k d}{2}} \quad (3.12)$$

hvor  $d$  er samplingsavstanden. Avstanden  $d$  kontrollerer hvilke deler av bølgefeltet som beregnes i responsen. Figur 3.3 viser det synlige området til den lineære aperture funksjonen gitt i (3.12).

Bølgetallskomponenten til aperturen er gitt ved

$$k_x = -\frac{2\pi}{\lambda} \sin \phi \quad (3.13)$$



Figur 3.3: Det synlige området av  $W(k_x)$  for samplingsavstand  $d = \lambda/4, \lambda/2, \lambda$  for lineær aperture.

slik at

$$|k_x| \leq \frac{2\pi}{\lambda} \quad (3.14)$$

Ved undersampling, altså  $d > \lambda/2$ , vil det synlige området inkludere mer enn én periode av  $W(k)$ . Dette kan medføre at bølgens forplantningsretning ikke er entydig definert.

Hvis  $d < \lambda/2$  så er aperturen oversamplet. Nøyaktig én periode av  $W(k)$  er synlig, men  $k > 2\pi/\lambda$  definerer et usynlig område hvor det ikke eksisterer bidrag fra bølgefeltet, men hvor støy kan forekomme. I arrayprosessering kan dette utnyttes til å ekskludere støybidrag.

### 3.3 Strålingsdiagrammet

Strålingsdiagrammet er et arrays respons til en monokromatisk plan bølge hvis utgangssignalet  $z(\vec{x}, t)$  er generert lineært og tidsinvariant fra array elementene, slik som for forsinkelse-sum stråleformeren (se f.eks. [Johnson and Dudgeon, 1993]). Hvis dette er oppfylt vil strålingsdiagrammet til et array av  $M$  punktelementer være

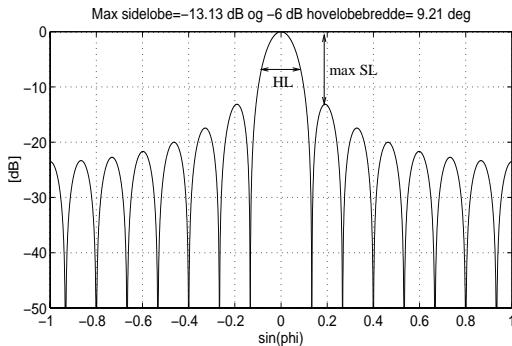
$$W(\vec{k}) = \sum_{n=0}^{M-1} w_n e^{j\vec{k} \cdot \vec{x}_n} \quad (3.15)$$

Hvis arrayet er lineært så kan man benytte at  $k_x = \frac{2\pi}{\lambda}(\sin \phi - \sin \phi_0) = \frac{2\pi}{\lambda}u$ , hvor  $\phi_0$  er styringsvinkelen til hovedloben, slik at

$$W(u) = \sum_{n=0}^{M-1} w_n \exp \left( j \frac{2\pi}{\lambda} u n d \right) \quad (3.16)$$

Ligning (3.15) er identisk med den diskrete aperture funksjonen (3.10). Altså kan man benytte aperture funksjonen til å studere bølger i fjernfeltet til arrayet. For bølger i nærfeltet vil strålingsdiagrammet avhenge av arrayets fokus og posisjonen til bølgekilden. I resten av denne oppgaven antas det at refleksjonskilden befinner seg i fjernfeltet. I tillegg antas det at sender- og mottager-elementene er like, slik at for kontinuerlige bølger så vil 2-veis responsen til arrayet være kvadratet av strålingsdiagrammet. Siden sender- og mottager-elementene er like må den utsendte bølgen pulses, men i [Smith *et al.*, 1991] viser man at 2-veis responsen kan tilnærmes som kvadratet av strålingsdiagrammet også for det pulsedes tilfellet.

Figur 3.4 viser strålingsdiagrammet for fjernfeltet til et  $M = 15$  elements lineært array med elementavstand  $d = \lambda/2$ . Strålingsdiagrammet karakteriseres ved én høy hovedlobe og flere lavere sidelober, i tillegg til eventuelle gitterlober.



Figur 3.4: Strålingsdiagrammet til et  $M = 15$  elements array med  $d = \lambda/2$  og dets hovedlobe(HL)-bredde og max sidelobe(SL)-nivå.

### 3.3.1 Hovedloben

Hovedloben (HL) i strålingsdiagrammet angir forplantningsretningen til bølgefronten. Bredden på hoveloben angir hvor godt arrayet kan skille mellom to retninger, slik at en smal hovedlobe gir bedre oppløsning enn en bred hovedlobe. Et vanlig mål er den fulle bredden 6 dB under toppen. Dette tilsvarer den fulle bredden ved halvparten av maksimal amplitud. Et annet mål er vinkelen hvor 1.

nullpunkt til strålingsdiagrammet inntreffer. For et fullt array så er 1. nullpunkt gitt ved

$$k_x^0 = \frac{2\pi}{D} \quad (3.17)$$

hvor D er arrayets aperture<sup>1</sup>. For lineære arrayer med konstant element-avstand  $d = \lambda/2$  så kan (3.17) skrives på formen

$$\begin{aligned} \frac{2\pi}{\lambda} u^0 &= \frac{2\pi}{Md} \\ u^0 &= \frac{2}{M} \end{aligned} \quad (3.18)$$

### 3.3.2 Sidelobene

Sidelobene (SL) i strålingsdiagrammet angir energien til bølgefeltet som arrayet opptar fra de andre retningene. Disse bør være så lave som mulig for å minime påvirkningen fra uønskede signaler, men som en konsekvens av Heisenbergs uskarphetsrelasjon kan ikke arrayet ha minimum sidelobenivå og minimum hovedlobebredde samtidig. Sidelobenivået karakteriseres ofte ved den maksimale amplituden til sidelobene relativt til amplituden til hovedloben. Maksimalt sidelobenivå er en “i værste fall” karakteristikk av arrayets evne til å undertrykke støy.

Energien i sidelobene er gitt ved

$$E_s = 2 \int_{\theta_0}^{\theta_{\max}} \int_{\phi_0}^{\phi_{\max}} |W(\phi, \theta)|^2 d\phi d\theta \quad (3.19)$$

hvor  $\theta_0$  og  $\phi_0$  er vinklene for 1. nullpunkt til  $W(\phi, \theta)$ . Tilsvarende er energien i hovedloben gitt som

$$E_h = 2 \int_0^{\theta_0} \int_0^{\phi_0} |W(\phi, \theta)|^2 d\phi d\theta \quad (3.20)$$

Forholdet mellom energien i sidelobene og energien i hovedloben påvirker kontrastoppløsningen i ultralydbilder. For store  $E_s$  vil det lekke energi fra lyse felter inn i mørke felter av bildet [Adams, 1991]. Integrert sidelobee rate (ISLR) uttrykker energifordelingen mellom sidelobene og hovedloben:

$$ISLR = 10 \log_{10} \frac{E_s}{E_h} \quad (3.21)$$

I denne oppgaven er avstanden mellom de 1. nullpunktene  $\phi_0$  tilnærmet med  $2\phi_{-6dB} \approx 3\phi_{-3dB}$  som benyttes av ESA<sup>2</sup>.

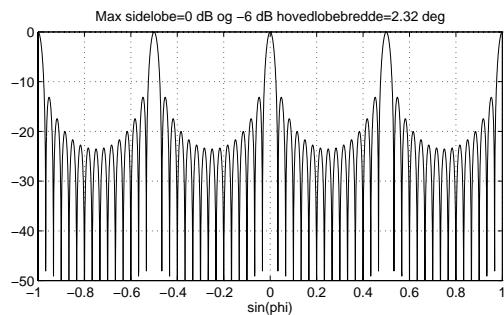
---

<sup>1</sup>Hittil har aperturen betegnet det kontinuerlige arrayets utstrekning, men her betegner aperturen utstrekningen til det diskrete arrayet.

<sup>2</sup>European Space Agency

### 3.3.3 Gitterlober

Gitterlober er periodiske gjentagelser av hovedloben og er et resultat av romlig undersampling. Hvis elementavstanden  $d$  er konstant og  $\lambda/2 < d < \lambda$  så vil man få gitterlober hvis hovedloben styres til vinkler større enn null slik figur 3.3 antyder. Derimot for  $d > \lambda$  vil man alltid få gitterlobe slik figur 3.5 viser.



Figur 3.5: Strålingsdiagram for lineær aperture,  $M = 15$  elementer og  $d = 2\lambda$ .

# Kapittel 4

## Tynnede arrayer

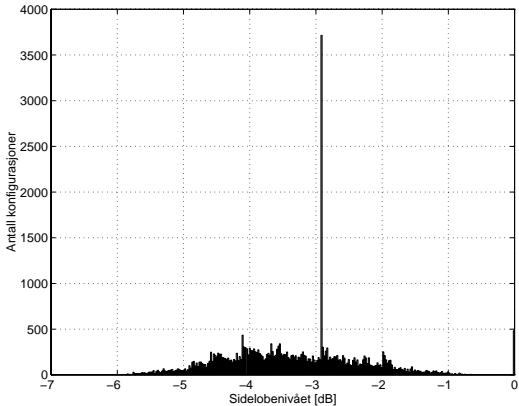
Periodiske arrayer uten elementvekting og element avstand  $d = \lambda/2$  har fallende sidelobenivå, altså er max sidelobenivå gitt av amplituden til 1. sidelobe. Lineære og rektangulære arrayer har 1. sidelobe med nivå  $-13.5$  dB, mens sirkulære arrayer har  $-17.7$  dB. I ultralyd avbildning trenger man arrayer med maksimalt sidelobenivå under  $-30$  dB. For å endre sidelobefordelingen kan alle elementene vektes med vinduer (se f.eks. [Harris, 1978, Oppenheim and Schafer, 1989]) eller man kan tynne arrayet ved å fjerne noen av elementene. 1D arrayer i radar og medisinsk ultralyd består av 48–192 elementer. For 2D arrayer trenger man tilnærmet et kvadratisk antall elementer, slik at 2D arrayer i ultralyd og sonar avbildninger trenger 2000–10000 elementer. Disse elementene har høy impedans og liten sensitivitet, slik at elementvekting ikke er praktisk anvendbart [Holm, 1997]. Likevel benyttes ofte elementvekting sammen med tynning for å få bedre forståelse for den ukjente optimale elementfordelingen [Holm *et al.*, 1997, Trucco, 1997].

### 4.1 Egenskaper til tynnede arrayer

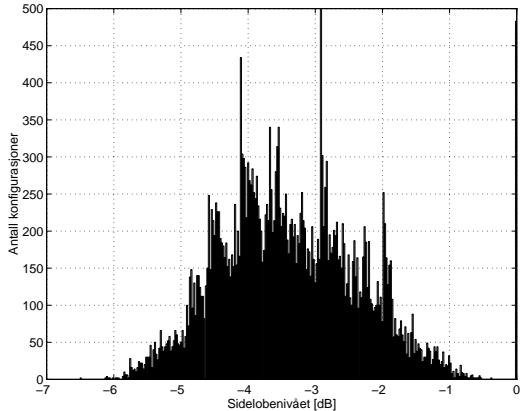
Elementene i et tynnet array ligger på et regulært grid, slik at elementavstanden  $d$  er et multiplum av  $\lambda/2$ . Nyquist betingelsen er altså oppfylt, men man kan få periodositeter i elementfordelingen og dermed mulige gitterlober. Disse kan unngås ved å benytte et irregulært grid slik at arrayet blir aperiodisk, men irregulære grid har uendelig mange kombinasjoner. Derfor er det gunstig å benytte et regulært grid og unngå periodositeter i elementfordelingen.

Tynning av et array med regulært grid uten elementvekting er et binært problem. Et  $M$  elements array tynnet til  $N$  aktive elementer uten noen begrensninger har

$$\binom{M}{N} = \frac{M!}{(M-N)!N!} \quad (4.1)$$



Figur 4.1: Histogram over sidelobenivået til alle arrayene med  $M=25$  elementer i aperturen og tynnet til  $N=7$  aktive elementer.



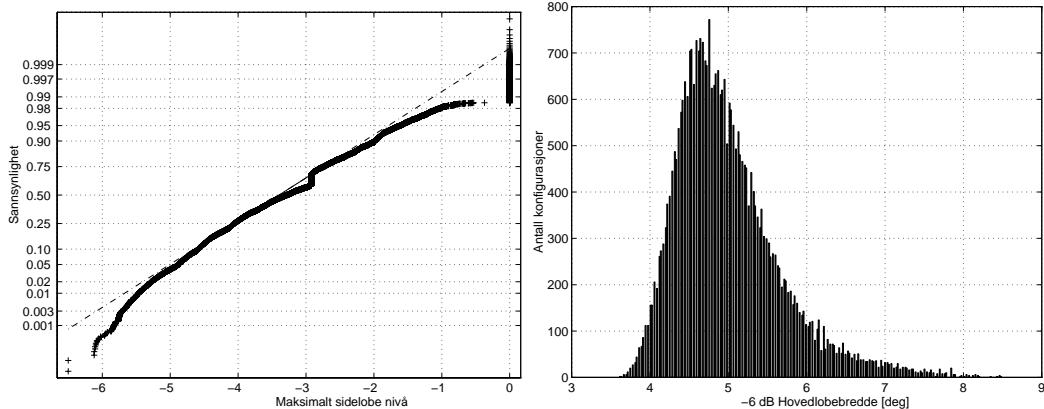
Figur 4.2: Utsnitt av histogrammet av sidelobenivåene i figur 4.1.

mulige kombinasjoner av aktive og passive elementer. For et  $50 \times 50$  array tynnet til 500 aktive elementer (80% tynning) så finnes det  $4 \cdot 10^{542}$  mulige kombinasjoner. I det lineære tilfellet krever man at ende-elementene er aktive, slik at aperturens størrelse er konstant. Et  $M$  elements array tynnet til  $N$  aktive elementer har altså

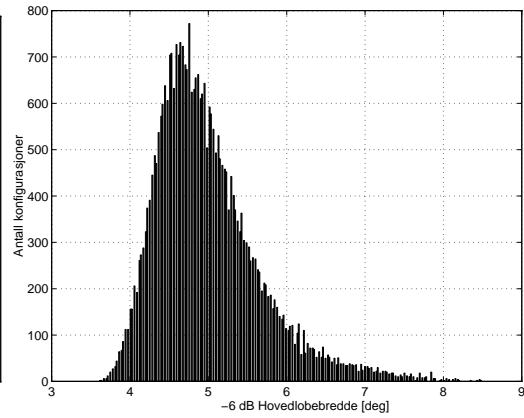
$$\binom{M-2}{N-2} = \frac{(M-2)!}{(M-N)!(N-2)!} \quad (4.2)$$

mulige kombinasjoner. Altså er det en uoverkommelig oppgave å generere alle mulige elementkombinasjoner, selv for middels store arrayer. Det å generere alle elementkombinasjonene er likevel en lett oppgave sammenlignet med å generere strålingsdiagrammet til alle arrayene, til tross for at strålingsdiagrammet til arrayer med regulære grid kan beregnes med en FFT av elementvektene. Et  $M = 25$  elements lineært array tynnet til  $N = 7$  aktive elementer har  $\binom{25-2}{7-2} = 33649$  mulige kombinasjoner. Det tar kun et par sekunder å generere permuteringene ved en rekursiv prosedyre (tillegg B), mens det tar ca 1t å generere en 4096-punkts FFT og finne maksimal sidelobe amplitude og hovedlobebredde for hver array på en SGI Power Challenge med 1 av 4 CPUer aktive (Matlab-kode i C.1). Et lignende studie er gjort i [Lo and Lee, 1966], hvor man studerte elementkonfigurasjonene til et symmetrisk array med  $D = 19\lambda$  aperture og 9 aktive elementer på et grid med  $d = \lambda/4$  elementavstand. Dette gir totalt 7770 konfigurasjoner som ble evaluert med en diskretisering 1/19 på intervallet  $u \in [0, 2]$ .

Figur 4.1 viser et histogram over maksimalt sidelobenivå til konfigurasjonene. Vi ser at det er en veldig høy søyle ved  $\max SL = -2.92\text{dB}$ . Dette skyldes at arrayet har  $24d = 2^3 \cdot 3d$  aperture, slik at det finnes veldig mange konfigurasjoner



Figur 4.3: Normalt sannsynlighets-  
lott av maksimalt sidelobenivå til  
alle arrayer med  $M = 25$  og  $N = 7$ .



Figur 4.4: Histogram over  $-6$  dB ho-  
vedlobebredden til alle arrayer med  
 $M = 25$  og  $N = 7$ .

med elementavstand lik et multiplum av *både*  $2d$  og  $3d$ , men dette skal vi komme tilbake til i kapittel 5. I tillegg finnes det 483 arrayer med gitterlober (max  $SL = 0$  dB). Elementene til disse arrayene ligger på et grid med elementavstand  $2d$  eller  $3d$ . Antall arrayer med gitterlober er gitt ved:

$$\begin{array}{rcl} \binom{24/3-1=7}{5} & = & 21 \\ + \binom{24/2-1=11}{5} & = & 462 \\ \hline & = & 483 \end{array}$$

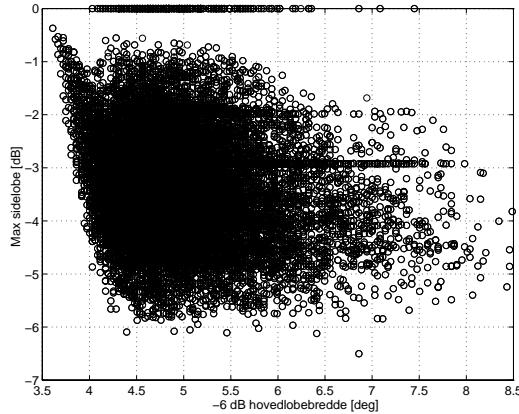
Et utsnitt av figur 4.1 er gitt i figur 4.2. Vi kan se at det er svært få konfigurasjoner med relativt lavt sidelobenivå. Det finnes f.eks. kun 12 konfigurasjoner (0.035%) med maksimalt sidelobenivå under  $-6$  dB. I [Lo and Lee, 1966] fant man at ensemble fordelingen av maksimalt sidelobe nivå er nær gaussisk. Figur 4.3 viser et normalitetslott av sidelobenivået til alle arrayene, hvor gaussisk fordelte data skal ligge omkring den rette linjen. Man kan se at de fleste sidelobenivåene svarer til en gaussisk fordeling, men de største og minste sidelobe verdiene har mindre sannsynlighet enn verdier fra en gaussisk fordeling.

Figur 4.4 viser et histogram av hovedlobebreddene. Histogrammet har en gamma-lignende fordeling med middelverdi  $5.00^\circ$ . Til sammenligning er hovedlobebreden til det fulle arrayet  $5.45^\circ$ . For dette tynningsproblemet har hele 80% av arrayene smalere hovedlobe enn det fulle arrayet, men pga. få elementer har de et mye høyere sidelobenivå.

Hovedlobebredder og sidelobenivåer gir ikke mye mening hver for seg. I figur 4.5 kan man se verdiene for samtlige arrayer som punkter i sidelobe-hovedlobe planet. De optimale sidelobe verdiene (minimum sidelobe for en gitt hovedlobebredde) minker for økende hovedlobebredde, men fra ca  $4.4^\circ$  ligger de optimale

sidelobe verdiene på et platå. Hvis man i en optimeringsrutine søker et sidelobenivå nær det globale minimum kan man i dette tilfellet få hovedlobebredde  $1^\circ$  smalere enn bredden til det fulle arrayet.

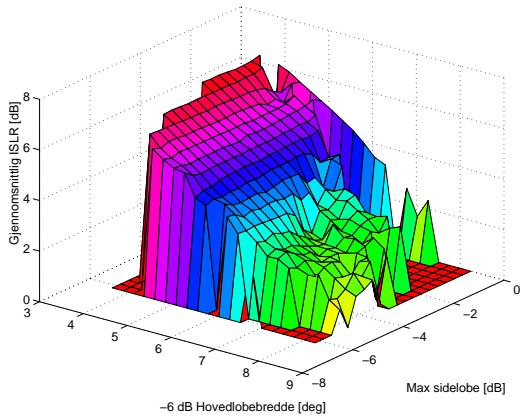
Integrert sidelobe rate med bibetingelser på sidelobenivået benyttes som optimeringskriterium i [Austeng *et al.*, 1997, Boni *et al.*, 1994, Holm, 1996]<sup>1</sup>. Derfor er det interessant å se hvordan den integrerte sidelobe raten varierer med max sidelobe og hovedlobebredden. I figur 4.6 er plottet i figur 4.5 delt inn i et  $20 \times 20$  grid og figuren viser midlet integrert sidelobe rate for dette gridet som funksjon av max sidelobe og hovedlobebredde. Man kan se at midlet ISLR er tilnærmet lineært avhengig av hovedlobebredden og at midlet ISLR varierer betydelig mindre med max sidelobenivået. Figur 4.7 viser integrert sidelobe rate som funksjon av maksimalt sidelobenivå for det samme optimeringsproblemet. Arrayene med hovedlobebredde gitt ved  $4.4^\circ < HL < 4.5^\circ$  og  $5.9^\circ < HL < 6.0^\circ$  er plottet i hhv. rødt og blått. Arrayene med tilnærmet konstant hovedlobebredde har svært liten variasjon i ISLR for lave max sidelobe verdier. Disse ISLR verdiene er blant de største verdiene for denne hovedlobebredden. I en optimeringsrutine for å finne minimum sidelobenivå for en gitt hovedlobebredde virker det som det er lite å tjene på å samtidig minimere integrert sidelobe rate. Derfor vil ikke minimum ISLR benyttes som optimerings-kriterium i denne oppgaven.



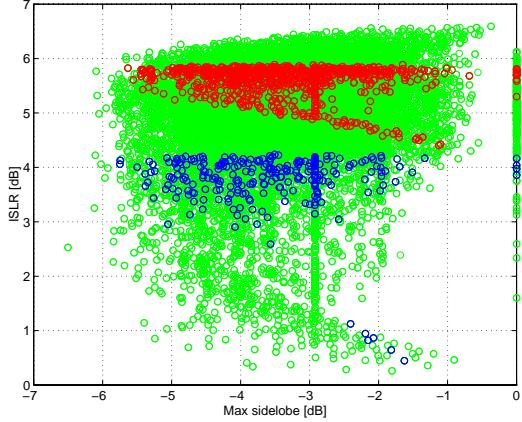
Figur 4.5: Sidelobenivå og hovedlobebredde for alle arrayer med  $M = 25$  elementers aperture tynnet til  $N = 7$  aktive elementer.

---

<sup>1</sup> [Holm, 1996] sammenligner max sidelobe og ISLR



Figur 4.6: Gjennomsnittlig integrert sidelobe rate som funksjon av maksimal sidelobe amplitude og hovedlobebredde.



Figur 4.7: Integrert sidelobe rate som funksjon av maksimalt sidelobenivå. Røde 'o' har  $-6 \text{ dB}$  hovedlobebredde ( $HL$ ) ved  $4.4^\circ < HL < 4.5^\circ$  og blå 'o' har  $5.9^\circ < HL < 6.0^\circ$ .

## 4.2 Stokastiske arrayer

Stokastisk tynede arrayer er interessante siden selve tynningsprosessen er enkel og det eksisterer en omfattende teori [Steinberg, 1976] om deres egenskaper, hvor de viktigste poengene blir gjennomgått her. Det normerte strålingsdiagrammet til et  $N$  elements array med stokastiske elementposisjoner  $X_n$  er gitt som

$$f(u) = \frac{1}{N} \sum_{n=1}^N \exp \left[ j \frac{2\pi}{\lambda} u X_n \right] \quad (4.3)$$

hvor

$$\frac{2\pi}{\lambda} u = k_x - k_{x0} = \frac{2\pi}{\lambda} (\sin \phi - \sin \phi_0)$$

Forventningen til det normerte strålingsdiagrammet er

$$\begin{aligned} \overline{f(u)} &= \frac{1}{N} \sum_{n=1}^N \mathcal{E} \left\{ \exp \left[ j \frac{2\pi}{\lambda} u X_n \right] \right\} \\ &= \mathcal{E} \left\{ \exp \left[ j \frac{2\pi}{\lambda} u X \right] \right\} \\ &= \sum_x \hat{g}_X(x) \exp \left[ j \frac{2\pi}{\lambda} ux \right] \end{aligned} \quad (4.4)$$

De statistiske egenskapene til et array med elementposisjoner gitt ved en kontinuerlig sannsynlighetsfordeling  $g_X(x)$  og et array tynnet ved en diskret sannsynlighetsfordeling  $\hat{g}_X(x)$  er de samme [Steinberg, 1976]. Hvis man antar en uniform fordeling på intervallet  $(-D/2, D/2)$  hvor  $D = Md$ , så kan (4.4) utrykkes ved:

$$\begin{aligned}\overline{f_{\text{U}}(u)} &= \int_{-D/2}^{D/2} \frac{1}{D} \exp \left[ j \frac{2\pi}{\lambda} ux \right] \\ &= \frac{\sin(D\pi u/\lambda)}{D\pi u/\lambda} \\ &= \frac{\sin(Md\pi u/\lambda)}{Md\pi u/\lambda}\end{aligned}\quad (4.5)$$

Tilsvarende kan man finne forventningen til  $f(u)$  for andre sannsynlighetsfordelinger. Det 2. ordens momentet til  $f(u)$  er forventningen til effekt-tethetsspekteret:

$$\begin{aligned}\overline{|f(u)|^2} &= \mathcal{E}\{f(u)f^*(u)\} \\ &= \frac{1}{N^2} \sum_{n=1}^N \sum_{m=1}^N \mathcal{E} \left\{ \exp \left[ j \frac{2\pi}{\lambda} u(X_n - X_m) \right] \right\} \\ &= \frac{1}{N^2} \left[ \sum_{n=m} e^0 + \sum_{n \neq m} \mathcal{E} \left\{ \exp \left[ j \frac{2\pi}{\lambda} u X_n \right] \cdot \exp \left[ -j \frac{2\pi}{\lambda} u X_m \right] \right\} \right] \\ &= \frac{1}{N^2} \left[ N + (N^2 - N) \mathcal{E} \left\{ \exp \left[ j \frac{2\pi}{\lambda} ux \right] \right\} \mathcal{E} \left\{ \exp \left[ -j \frac{2\pi}{\lambda} ux \right] \right\} \right] \\ &= \frac{1}{N^2} \left[ N + (N^2 - N) \overline{f(u)f^*(u)} \right] \\ &= \frac{1}{N} + (1 - \frac{1}{N}) |\overline{f(u)}|^2\end{aligned}\quad (4.6)$$

For hovedloben så er

$$\begin{aligned}\overline{|f(u)|^2} &= \frac{1}{N} + (1 - \frac{1}{N}) \\ &= 1\end{aligned}$$

For sidelobene ( $u \neq 0$ ) så vil  $\frac{1}{N} \gg |\overline{f(u)}|^2$  slik at gjennomsnittlig sidelobenivå for strålingsdiagrammet er

$$\overline{SL} = 10 \log_{10} \frac{1}{N} \quad (4.7)$$

For et stokastisk tynnet array så er altså gjennomsnittlig sidelobenivå invers proporsjonal med antall gjenværende elementer. Dette målet gir en indikasjon på hvor mange elementer som må være i et algoritmisk optimert tynnet array.

Variansen til strålingsdiagrammet er gitt ved

$$\begin{aligned}
\sigma^2(u) &= \mathcal{E}\{f(u)f^*(u)\} - \mathcal{E}\{f(u)\}\mathcal{E}\{f^*(u)\} \\
&= \frac{1}{N} + (1 - \frac{1}{N})|\overline{f(u)}|^2 - |\overline{f(u)}|^2 \\
&= \frac{1}{N}(1 - |\overline{f(u)}|^2)
\end{aligned} \tag{4.8}$$

Variansen går mot null når  $u \rightarrow 0$  ( $|\overline{f(u)}|^2 \rightarrow 1$ ). Dette betyr at fasongen til hovedloben og de første sidelobene påvirkes lite av at elementposisjonene er stokastiske. For store verdier av  $u$  så vil variansen gå som  $\frac{1}{N}$ . Altså for sidelobene vekk fra hovedloben så vil variansen til strålingsdiagrammet være like stor som middelverdien. Strukturen til disse sidelobene er mao. primært avhengig av den stokastiske elementfordelingen.

Sannsynlighets-tetthetsfunksjonen til amplituden til sidelobene kan utledes tilsvarende. Det normerte strålingsdiagrammet (4.3) kan skrives på formen:

$$\begin{aligned}
f(u) &= \operatorname{Re}\{f(u)\} + j\operatorname{Im}\{f(u)\} \\
&= a(u) + j \cdot b(u) \\
&= \frac{1}{N} \sum_{n=1}^N \cos \frac{2\pi}{\lambda} u \chi_n + j \frac{1}{N} \sum_{n=1}^N \sin \frac{2\pi}{\lambda} u \chi_n
\end{aligned}$$

$a(u)$  og  $b(u)$  er uavhengige stokastiske funksjoner. I henhold til sentralgrense teoremet så vil sannsynlighets-tetthetsfunksjonen til  $a(u)$  og  $b(u)$  konvergere mot en gaussisk fordeling så lenge ingen av leddene i summene dominerer. Sannsynlighets-tetthetsfunksjonen til strålingsdiagrammet er

$$\begin{aligned}
w_{a,b}(u) &= w_a(u) \cdot w_b(u) \\
&= \frac{1}{2\pi\sigma_a\sigma_b} \exp \left[ -\frac{(a - \bar{a})^2}{2\sigma_a^2} - \frac{(b - \bar{b})^2}{2\sigma_b^2} \right]
\end{aligned} \tag{4.9}$$

hvor  $\sigma^2 = \sigma_a^2 + \sigma_b^2$  og  $\overline{f(u)} = \overline{a(u)} + j\overline{b(u)}$ . For sidelobene tilstrekkelig unna hovedloben så er  $\sigma^2 = \frac{1}{N}$  slik at  $\sigma_a^2 = \sigma_b^2 = \frac{1}{2N}$ . Dette gir at

$$w_{a,b}(u) = \frac{1}{\pi\sigma^2} \exp \left[ -\frac{(a - \bar{a})^2 + (b - \bar{b})^2}{\sigma^2} \right] \tag{4.10}$$

Polar transformasjon ved  $a = |f| \cos \phi$  og  $b = |f| \sin \phi$  gir

$$w_{a,b}(|f|, \phi) = \frac{1}{\pi\sigma^2} \exp \left[ -\frac{(|f| \cos \phi - |\bar{f}| \cos \bar{\phi})^2 + (|f| \sin \phi - |\bar{f}| \sin \bar{\phi})^2}{\sigma^2} \right] \tag{4.11}$$

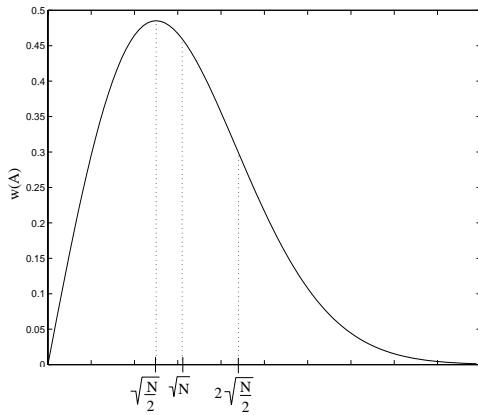
Integrasjon over  $\phi$  vil gi en sannsynlighets-tetthetsfunksjon av amplituden  $|f|$  til strålingsdiagrammet.

$$\begin{aligned} w(|f|) &= \int_{\phi} w_{a,b}(|f|, \phi) d\phi \\ &= \frac{2|f|}{\sigma^2} \exp \left[ -\frac{|f|^2 + |\bar{f}|^2}{\sigma^2} \right] \mathfrak{J}_0 \left( \frac{2|\bar{f}||f|}{\sigma^2} \right) \end{aligned}$$

hvor  $\mathfrak{J}_0$  er 0. ordens første type modifisert Besselfunksjon. For tilstrekkelige store vinkler slik at  $|\bar{f}|^2 \ll \sigma^2 = \frac{1}{N}$  reduseres dette til *Rayleigh fordelingen*:

$$\begin{aligned} w(|f|) &= \frac{2|f|}{\sigma^2} \exp \left( -\frac{|f|^2}{\sigma^2} \right) \\ &\Downarrow \\ w(A) &= \frac{2A}{N} \exp \left( -\frac{A^2}{N} \right) \quad (4.12) \end{aligned}$$

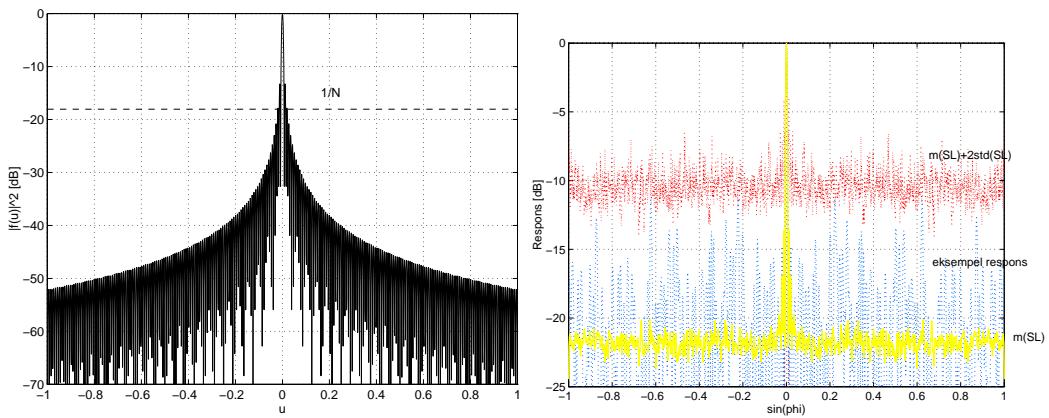
hvor den unormaliserte amplituden  $A = N|f|$ . Ligning (4.12) er sannsynlighets-tetthetsfunksjonen til amplituden til en tilfeldig sidelobe til et array med stokastiske elementposisjoner. Sidelobe amplituden er Rayleigh fordelt uavhengig av elementfordelingen. Figur 4.8 viser sannsynlighets-tetthetsfunksjonen til sidelobe



Figur 4.8: Fordelingsfunksjonen  $w(A)$  som funksjon av den unormaliserte sidelobe amplituden  $A$ .

amplituden  $A$ . Forventningen til fordelingen er  $\sqrt{N}$  og variansen er  $\frac{N}{2}$ .

Sidelobene omkring hovedloben er ikke Rayleigh fordelt, siden  $|\bar{f}|^2 \ll \frac{1}{N}$  ikke er oppfylt her. Figur 4.9 viser  $|\bar{f}_{\text{U}}(u)|^2$  sammenlignet med  $\frac{1}{N}$  for et array med  $M = 256$  elementer tynnet til  $N = 64$ . For store arrayer så vil vi dette



Figur 4.9: Forventningen kvadrert  $|f(u)|^2$  til strålingsdiagrammet til 64 uniformt fordelte elementer over en 256 elementers aperture.

Figur 4.10: **Gul kurve:** Midlet strålingsdiagram for 100 arrayer med uniformt fordelte elementer. Midlingen er gjort i lineær skala.

**Rød kurve:** Midlet strålingsdiagram +2 standardavvik av strålingsdiagrammet.

**Blå kurve:** Strålingsdiagrammet til et tilfeldig array med uniformt fordelte elementer.

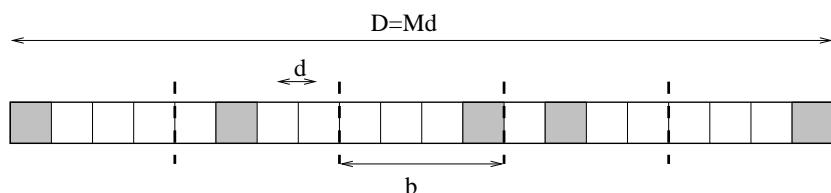
gjelde for tilnærmet hele sidelobe regionen. I praksis observeres ofte maksimalt sidelobenivå 10 dB høyere enn midlet sidelobenivå. Et eksempel på at maksimal sidelobe amplitude er mye høyere enn middelverdien er gitt i figur 4.10, som viser midlet strålingsdiagram for 100 arrayer med 64 uniformt fordelt elementer i et array med 256 element posisjoner. Teoretisk middelverdi for sidelobene tilstrekkelig unna hovedloben er  $10 \log_{10} \frac{1}{64} = -18.06$  dB. For verdier av  $u > 0.05$  så stabiliserer de midlede sidelobe verdiene omkring -21 dB. Av figur 4.9 kan man se at dette tilsvarer at  $|f_u(0.05)|^2 = -27$  dB. Maksimalt sidelobenivå til et tilfeldig array i ensemblet er -10.3dB, altså 8 dB høyere enn teoretisk middelverdi og 11 dB høyere enn ensemble middelverdien. Sidelobe verdier omkring -10dB tilsvarer 2 standardavvik over middelverdien slik figur 4.10 viser.

## 4.3 Arrayer med betingelser

Antall totalt mulige arrayer vokser ekstremt fort med antall elementer. Derfor er det interessant å se på egenskapene til arrayer med ulike begrensninger.

### 4.3.1 Binnede arrayer

En variant av stokastiske arrayer er binnede arrayer, hvor arrayets aperture  $D = Md$  er delt i  $N$  binne av lengde  $b = M/N$ . I hvert binne finnes det kun ett element slik figur 4.11 viser. Binnede arrayer har spesielle statistiske egenskaper som er gitt i [Hendricks, 1991] og hovedtrekkene vil bli gjengitt her. I tillegg gir binnede arrayer en drastisk reduksjon av totalt antall arrayer, siden antall binnede arrayer med betingelsen om aktive ende-elementer er  $b^{N-2}$  og uten tilleggsbetingelse  $b^N$ .



Figur 4.11: Binnet array med  $N = 5$  binne og  $M = 20$  element posisjoner.

Det normerte strålingsdiagrammet til et binnet array med  $N$  elementer og binnelengde  $b = M/N$  hvor  $M$  er antall elementer i fullt array er gitt ved

$$f_B(u) = \frac{1}{N} \sum_{n=1}^N \exp \left[ j \frac{2\pi}{\lambda} u X_n \right] \quad (4.13)$$

hvor den stokastiske elementposisjonen  $X_n$  er gitt ved

$$X_n = -\frac{D}{2} + (n-1)b + Y_n \quad (4.14)$$

$Y_n$  er en kontinuerlig stokastisk variabel som i dette tilfellet er uniformt fordelt over intervallet  $(0, b)$ . Forventningen til det binnede strålingsdiagrammet er:

$$\begin{aligned} \overline{f_{\mathfrak{B}}(u)} &= \frac{1}{N} \sum_{n=1}^N \mathcal{E} \left\{ \exp \left[ j \frac{2\pi}{\lambda} u \left( -\frac{D}{2} + (n-1)b + Y_n \right) \right] \right\} \\ &= \frac{1}{N} \sum_{n=1}^N \frac{1}{b} \int_0^b \exp \left[ j \frac{2\pi}{\lambda} u \left( -\frac{D}{2} + (n-1)b + y \right) \right] dy \end{aligned}$$

Ved uniform sannsynlighets-tetthetsfunksjon på hvert binne, så vil summen av integrasjonen over hver binne være lik integrasjonen over hele arrayet:

$$\begin{aligned} \overline{f_{\mathfrak{B}}(u)} &= \frac{1}{Nb} \int_{-D/2}^{D/2} \exp \left[ j \frac{2\pi}{\lambda} uz \right] dz \\ &= \overline{f_{\mathfrak{U}}(u)} \end{aligned} \quad (4.15)$$

Hvor man substituerer  $z = -\frac{D}{2} + (n-1)b + y$ . Forventingen til det binnede arrayet med uniform sannsynlighet er lik forventingen til det stokastiske arrayet med uniformt fordelte elementer og er gitt i ligning (4.5). Dette resultatet gjelder kun ved uniform sannsynlighet på hvert binne, siden for andre fordelinger vil ikke summen av integrasjonen over hvert binne være identisk med integrasjonen over hele arrayet.

Variansen til det stokastiske arrayet har vi fra ligning (4.8), og ved uniform fordeling får man:

$$\begin{aligned} \text{var}\{f_{\mathfrak{U}}(u)\} &= \frac{1}{N} (1 - |\overline{f(u)}|^2) \\ &= \frac{1}{N} \left[ 1 - \left( \frac{\sin(Md\pi u/\lambda)}{Md\pi u/\lambda} \right)^2 \right] \end{aligned} \quad (4.16)$$

Variansen til det binnede arrayet er gitt ved:

$$\begin{aligned} \text{var}\{f_{\mathfrak{B}}(u)\} &= \frac{1}{N^2} \sum_{n=1}^N \text{var} \left\{ \exp \left[ j \frac{2\pi}{\lambda} u \left( -\frac{D}{2} + (n-1)b + Y_n \right) \right] \right\} \\ &= \frac{1}{N^2} \sum_{n=1}^N \left| \exp \left[ j \frac{2\pi}{\lambda} u \left( -\frac{D}{2} + (n-1)b \right) \right] \right|^2 \cdot \text{var} \left\{ \exp \left[ j \frac{2\pi}{\lambda} u Y_n \right] \right\} \end{aligned}$$

$$\begin{aligned}
&= \frac{1}{N} \operatorname{var} \left\{ \exp \left[ j \frac{2\pi}{\lambda} u Y_n \right] \right\} \\
&= \frac{1}{N} \operatorname{var} \left\{ \exp \left[ j \frac{2\pi}{\lambda} \frac{uN}{N} N Y_n \right] \right\}
\end{aligned}$$

Hvis  $Y_n$  er uniform på  $(0, b)$  så vil  $N Y_n$  være uniform på  $(0, D)$  slik at

$$\operatorname{var}\{f_{\mathfrak{B}}(u)\} = \operatorname{var}\{f_{\mathfrak{U}}(u/N)\} \quad (4.17)$$

I kapittel 4.2 fant man at  $\operatorname{var}\{f_{\mathfrak{U}}(u)\} \rightarrow 0$  når  $u \rightarrow 0$  og at  $\operatorname{var}\{f_{\mathfrak{U}}(u)\} \rightarrow 1/N$  for store  $u$ . For binnede arrayer øker variansen med  $u/N$  slik at de første sidelobene til et binnet array vil ha mindre varians enn et stokastisk array. Variansen til det stokastiske arrayet når sitt 1. maksimum for

$$\frac{Md\pi u}{\lambda} = \pi \Rightarrow u = \frac{\lambda}{Md}$$

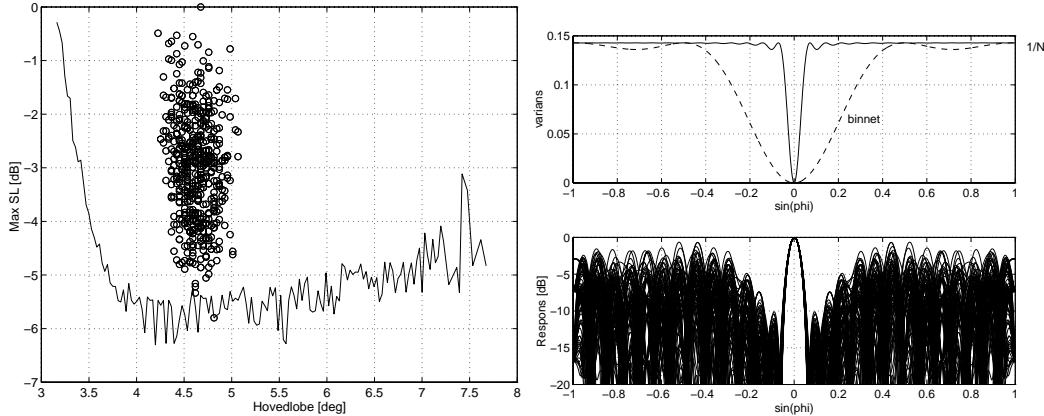
For  $d = \lambda/2$  så får man  $u = 2/M$  som er 1. nullpunkt til det fulle arrayet slik man så i (3.18). Tilsvarende vil variansen til det binnede arrayet nå sitt maksimum for

$$\frac{Md\pi u}{\lambda N} = \pi \Rightarrow u = \frac{N\lambda}{Md}$$

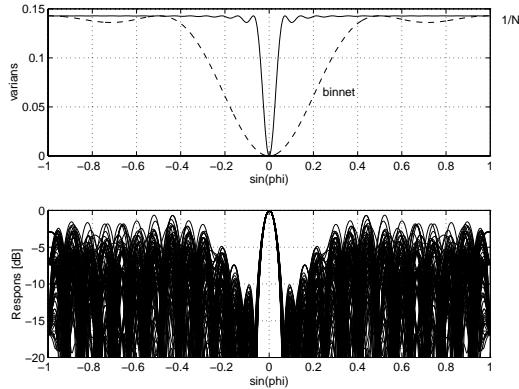
Ved element-avstand  $d = \lambda/2$  så vil sidelobene til det binnede arrayet påvirkes lite av at elementene er stokastiske i intervallet  $u \in [0, \frac{2N}{M}]$ .  $u = \frac{2N}{M}$  tilsvarer 1. nullpunkt til en full binne. Mindre binner vil altså gi kontroll over sidelobene over et større intervall, men det betyr også en økning i antall aktive elementer.

Et fullt array med  $M = 28$  elementer tynnet til  $N = 7$  aktive elementer vil ha binnede løsninger hvor binnelengden  $b = 4$ . Hvis man krever at ende-elementene skal være aktive så finnes det totalt  $\binom{28-2}{7-2}$  arrayer hvorav  $4^{7-2} = 1024$  er binnede arrayer. Maksimalt sidelobenivå og hovedlobebredde ble funnet ved en 4096-punkts FFT. Figur 4.12 viser minimum max sidelobe for hver hovedlobebredde og de binnede arrayenes plassering i forhold til den optimale kurven. Alle de binnede arrayene har  $-6\text{dB}$  hovedlobebredde mellom  $4.3^\circ$  og  $5.1^\circ$ . Til sammenligning så er hovedlobebredden til det fulle arrayet  $4.95^\circ$ . Variansen til det stokastiske arrayet har sitt 1. maksimum for  $\phi = 4.1^\circ$ , mens det binnede arrayet har sitt 1. maksimum for  $\phi = 30^\circ$  slik figur 4.13 viser. Dermed vil hovedlobebredden til det binnede arrayet påvirkes mye mindre av at elementfordelingen er stokastisk enn hva tilfellet er for det stokastiske arrayet uten begrensninger.

De maksimale sidelobe verdiene for binnede arrayer strekker seg fra 0 dB (gitterlober) ned til de optimale verdiene. Hvis man ikke hadde benyttet kravet om aktive ende-elementer så ville man ha fått periodiske løsninger. Arrayet med max sidelobe  $\approx 0$  dB i figur 4.12 er nær periodisk da elementavstanden er et multiplum av  $3d$ . Det nederste plottet i figur 4.13 viser strålingsdiagrammene til 100



Figur 4.12: Maksimalt sidelobenivå og hovedlobebredde for de binnede arrayene i forhold til de optimale verdiene for  $M = 28$  og  $N = 7$ .



Figur 4.13: Det øverste plottet viser variansen til det binnede arrayet i forhold til det stokastiske, og det nederste viser strålingsdiagrammene til 100 tilfeldig valge binnede arrayer.

tilfeldig valgte binnede arrayer. Hovedlobebredden varierer lite som tidligere observert, og i tillegg kan man se at sidelobene for  $\phi < 30^\circ$  har lavere verdier enn resten av responsen. Binnede arrayer har altså den fremragende egenskapen at de har redusert sidelobenivå omkring hovedloben, men dette gjør at de ikke vil få en Dolph-Chebyshev lignende respons, som har det laveste maksimale sidelobe nivået. I en optimeringsroutine for å finne minimum sidelobenivå for binnede arrayer, så vil hovedlobens bredde ble bestemt av binnelengden  $b$  og dermed variansen omkring hovedloben.

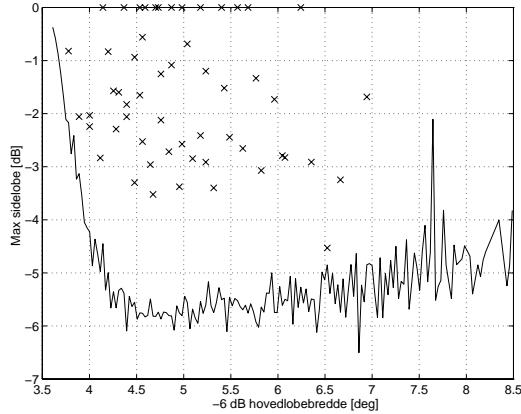
### 4.3.2 Symmetriske arrayer

Symmetriske arrayer har de aktive elementene symmetrisk plassert om arrayets senter. Totalt antall symmetriske arrayer tynnet fra  $M$  elementer (fullt array) til  $N$  elementer er gitt ved:

$$\# \text{ sym. arrayer} = \begin{cases} \binom{M/2-1}{N/2-1} & M, N \text{ like} \\ 0 & M \text{ like}, N \text{ odd} \\ \binom{\frac{M-1}{2}-1}{\frac{N}{2}-1} & M \text{ odd}, N \text{ like} \\ \binom{\frac{M-1}{2}-1}{\frac{N-3}{2}} & M, N \text{ odd} \end{cases} \quad (4.18)$$

For tynningsproblemet gitt i figur 4.14 så finnes det kun 55 symmetriske arrayer av totalt 33649, siden  $M$  og  $N$  odd gir en tilleggsbetingelse om at senterelementet

må være aktivt. Figur 4.14 viser hovedlobebredde og max sidelobe til disse relativt til kurven til de optimale verdiene. Sidelobenivået til de symmetriske arrayene er langt fra de optimale verdiene, men dette er ikke nødvendigvis representativt for større arrayer som vil ha større frihetsgrad.



Figur 4.14: Max sidelobe og hovedlobebredde til de 55 symmetriske arrayene relativt til de optimale verdiene for de 33649 mulige arrayene for  $M = 25$  og  $N = 7$ .

Strålingsdiagrammet til symmetriske arrayer med odde antall elementer er gitt ved:

$$\begin{aligned}
 W(u) &= \sum_{n=-(N-1)/2}^{(N-1)/2} \exp\left(j\frac{2\pi}{\lambda}ux_n\right) \\
 &= \exp\left(j\frac{2\pi}{\lambda}ux_0\right) + \sum_{n=1}^{(N-1)/2} \left[ \exp\left(j\frac{2\pi}{\lambda}ux_n\right) + \exp\left(-j\frac{2\pi}{\lambda}ux_n\right) \right] \\
 &= 1 + 2 \sum_{n=1}^{(N-1)/2} \cos\left(\frac{2\pi}{\lambda}ux_n\right)
 \end{aligned} \tag{4.19}$$

Symmetriske arrayer (også plane) gir et reellt strålingsdiagram. Denne betingelsen gjør at man kan optimere arrayets strålingsdiagram ved heltalls lineær programmering [Holm *et al.*, 1997]. Optimering med lineær programmering konvergerer mot det globale optimum, men optimeringstiden blir lang for store problemer, slik at det største arrayet til [Holm *et al.*, 1997] er et sirkulært array med 112 elementer tynnet til 64 aktive elementer.

### 4.3.3 Periodiske arrayer

Hittil har periodiske arrayer blitt betraktet som et uønsket tynningsmønster siden gitterlober oppstår. Den harmoniske elementfordelingen gjør at man lett kan beregne ved hvilke vinkler gitterlobene oppstår, og mellom hver gitterlobe så er sidelobenivået fallende. Derfor er responsen til et periodisk array hvor gitterlobene kanselleres ønskelig. Ved å benytte forskjellige arrayer for sender-transduceren og mottaker-transduceren så er dette mulig. Denne fremgangsmåten er benyttet i [Lockwood *et al.*, 1996, Lockwood and Foster, 1996, Austeng *et al.*, 1997] og følgende eksempel er hentet fra [Lockwood *et al.*, 1996]:

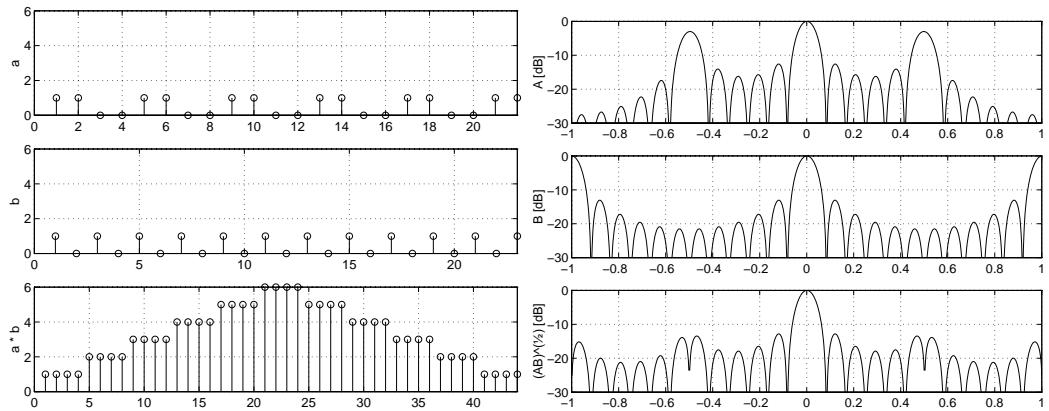
Anta 2 tynede arrayer  $a, b$  med 12 elementer og hhv. 22 og 23 mulige elementposisjoner slik at

$$\begin{aligned} w_a &= [1100110011001100110011] \\ w_b &= [1010101010101010101010101] \end{aligned}$$

Array  $a$  har strålingsdiagram  $W_a$  og tilsvarende har array  $b$  strålingsdiagram  $W_b$ . 2-veis strålingsdiagrammet til  $a$  og  $b$  er gitt ved

$$\begin{aligned} W_{a,b} &= W_a \cdot W_b \\ &= W_{a*b} \end{aligned}$$

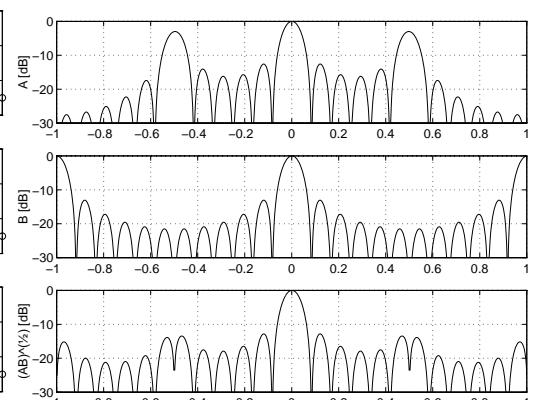
2-veis strålingsdiagrammet er altså lik produktet av hvert av strålingsdiagrammene eller strålingsdiagrammet til konvolusjonen av  $w_a$  og  $w_b$ . Figur 4.15 viser elementvektene til array  $a, b$  og den tilhørende konvolusjonen. De to øverste plottene i figur 4.16 viser hhv. strålingsdiagram  $W_a$  og  $W_b$ . Man ser at gitterlobene til array  $a$  inntreffer for samme vinkel som array  $b$  sine minima og vice versa. Det nederste plottet i figur 4.16 viser  $\sqrt{W_a \cdot W_b}$  hvor man kan se at gitterlobene og minimumspunktene har kansellert hverandre.



Figur 4.15: Øverst: De binære vektene til array  $a$  hvor  $N = 12$  og  $M = 22$ .

Midten: De binære vektene til array  $b$  hvor  $N = 11$  og  $M = 22$ .

Nederst: Konvolusjonen av vektene til array  $a$  og  $b$ .



Figur 4.16: Øverst: Strålingsdiagrammet til array  $a$ .

Midten: Strålingsdiagrammet til array  $b$ .

Nederst: 2-veis responsen til array  $a$  og  $b$  som er identisk med produktet av figurene over.

# Kapittel 5

## Coarrayet

Hittil har kun det tynnede arrayets egenskaper i Fourierdomenet blitt diskutert. Disse egenskapene er også de viktigste i henhold til de spesifikke krav man stiller til arrayet, men man får ikke tilstrekkelig forståelse for sammenhengen mellom Fourieregenskapene og elementfordelingen uten å studere arrayets romlige korrelasjon. Arrayer med bestemte korrelasjonssegenskaper viser seg å ha optimale Fourieregenskaper. Deler av dette arbeidet er sammenfattet i [Hopperstad and Holm, 1998] og er gitt i tillegg A.

### 5.1 Definisjoner

Et array over differansene til elementposisjonene kalles et coarray og er definert ved

$$c(\vec{\chi}) = \sum_{m_1, m_2 \in \vartheta} w_{m_1} w_{m_2} \quad (5.1)$$

hvor  $\vartheta(\vec{\chi}) = \{(m_1, m_2) | \vec{x}_{m_2} - \vec{x}_{m_1} = \vec{\chi}\}$  og  $w_m \in \{0, 1\}$ . For lineære arrayer på regulære grid med  $M$  posisjoner er coarrayet gitt som autokorrelasjonen til elementvektene.

$$c(l) = \sum_{n=0}^{M-|l|-1} w_n w_{n+|l|} \quad (5.2)$$

En differanse mellom to elementer kalles en baselinje. Hvis alle baselinjene har en unik representasjon så er arrayet perfekt. Antall elementer i aperturen ( $M$ ) til et perfekt array tynnet til  $N$  elementer er gitt ved

$$M_{perfekt} = \sum_{n=1}^{N-1} n + 1 = \frac{(N-1)N}{2} + 1 \quad (5.3)$$

Det eksisterer ingen perfekte arrayer for  $N > 4$ .

Hvis alle baselinjene er representert, men ikke nødvendigvis unikt, så er arrayet redundant. Antall redundanser  $R$  er gitt ved

$$R = \sum_{\forall l | c(l) > 1} [c(l) - 1] \quad (5.4)$$

Antall elementer i aperturen til et redundant array er gitt ved

$$M_{red} = \frac{(N - 1)N}{2} - R + 1 \quad (5.5)$$

I motsatt fall dvs. en unik representasjon, men med hull i coarrayet, så er arrayet ikke-redundant. Antall hull  $H$  defineres som antall baselinjer med coarrayverdi lik null, altså:

$$H = \sum_{\forall l | c(l) = 0} 1 \quad (5.6)$$

Antall elementer i aperturen til et ikke-redundant array er gitt ved

$$M_{ikke-red} = \frac{(N - 1)N}{2} + 1 + H \quad (5.7)$$

For  $N > 4$  så er

$$M_{red} < M_{perfekt} < M_{ikke-red} \quad (5.8)$$

Et array med  $N (> 4)$  av  $M$  mulige elementer har altså både redundanser og hull, og antall elementer i aperturen er gitt ved

$$M = \frac{(N - 1)N}{2} + 1 + H - R \quad (5.9)$$

Coarrayet er relatert til strålingsdiagrammet ved

$$|W(\vec{k})|^2 = \sum_{\vec{\chi}} c(\vec{\chi}) \exp(j\vec{k}\vec{\chi}) \quad (5.10)$$

For et  $N$  elements 1D-array med  $M$  posisjoner aperturen og regulær elementavstand  $d$ , så får man:

$$\begin{aligned} |W(k)|^2 &= \sum_{l=-(M-1)}^{M-1} c(l) \exp(jkld) \\ &= c(0) + \sum_{l=1}^{M-1} c(l) [\exp(jkld) + \exp(-jkld)] \\ &= N + \sum_{l=1}^{M-1} 2c(l) \cos(kld) \end{aligned} \quad (5.11)$$

Strålingsdiagrammet kvadrert (effekt-spekteret) er altså en superposisjon av cosinus til elementavstanden. I [De Graaf and Johnson, 1984] så er (5.11) gitt som:

$$\begin{aligned} |W(k)|^2 &= N - 1 + \frac{\sin(\frac{2N-1}{2}kd)}{\sin(kd/2)} \\ &\quad + 2 \sum_{l|c(l)>1} [c(l) - 1] \cos(kld) - 2 \sum_{l|c(l)=0} \cos(kld) \end{aligned} \quad (5.12)$$

Ligning (5.12) uttrykker at strålingsdiagrammet kvadrert er lik strålingsdiagrammet til et fullt array med aperture lik coarrayet pluss bidrag fra redundansene minus bidrag fra hullene.

## 5.2 Minimum redundans arrayer

Minimum redundans (MR) arrayer er de konfigurasjonene som for et gitt antall elementer  $N$  tilfredsstiller  $\min(R|H = 0)$ . Det finnes et ekvivalent tallteoretisk problem kalt *minimum differanse basis* [Leech, 1956]:

$$\{N_1, N_2, \dots, N_k\} \in \mathbb{N}$$

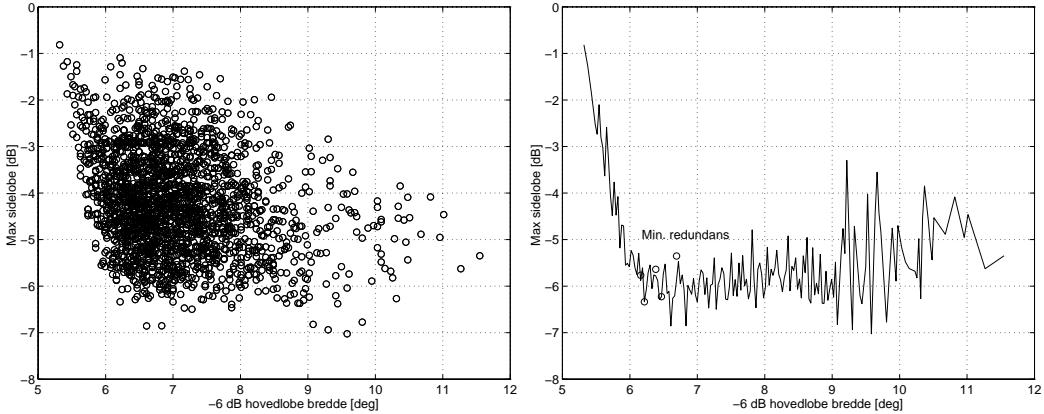
er en differanse basis mhp.  $L$  hvis

$$N = N_i - N_j \quad \forall N \in \{1, 2, \dots, L\} \quad (5.13)$$

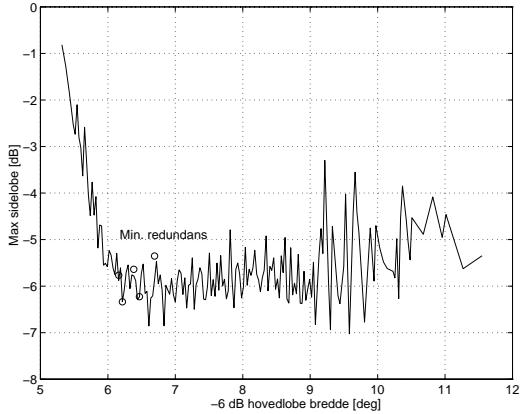
Minimum redundans arrayene finnes ved å minimere  $k$  eller maksimere  $L$ . Dette er ekvivalent med å minimere antall redundanser  $R$  eller maksimere aperturen  $M$ .

I signalbehandlingsteorien har man i lengre tid studert minimum redundans arrayer, motivert ut fra den gode oppløsningen disse har ([Ishiguro, 1980, Moffet, 1968, Ruf, 1993]). Derimot har man ikke tilstrekkelig studert deres egenskaper i Fourierdomenet. I [Bedrosian, 1986] hevder man at MR-arrayer er et alternativ til Dolph-Chebyshev arrayer og deres maksimalt flate sidelobenivå. I [De Graaf and Johnson, 1984] blir det antydet at hull og redundanser har samme potensielle for økning av sidelobenivået. I [Rossouw *et al.*, 1997] derimot benyttet man elementvektning for å oppnå et flatt sidelobenivå for ikke-optimale minimum redundans arrayer. For å se nærmere på karakteristikken til disse arrayene ser vi derfor på et tynningsproblem med beregningsmessig overkommelig utfallsrom og kjente MR-løsninger.

For 7 elementer så finnes det 5 (+5 speilinger) MR-konfigurasjoner [Linebarger *et al.*, 1993]. Disse har 18 elementer i aperturen, altså  $\binom{16}{5} = 4368$  mulige løsninger. For alle disse er hovedlobebredden og max sidelobenivå beregnet. I figur 5.1 kan man se at det er relativt stor spredning på løsningene. Den



Figur 5.1: Max sidelobenivå og hovedlobebredde for alle arrayene med  $N = 7$  aktive elementer på  $M = 18$  mulige elementposisjoner i aperturen.

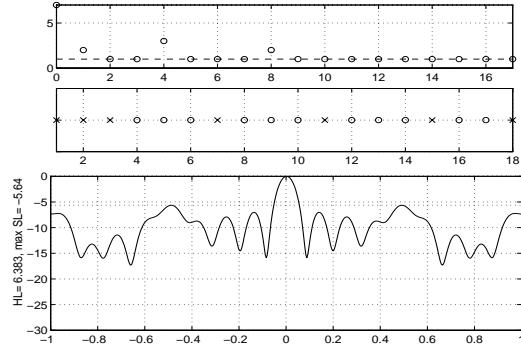


Figur 5.2: Optimal kurve og de 5 minimum redundans arrayene for  $N = 7$  og  $M = 18$ .

nedre randen av figur 5.1 defineres som den optimale kurven for dette problemet. Figur 5.2 viser den optimale kurven, og MR-løsningene i forhold til denne. Tre av løsningene ligger på kurven, mens to ligger rett over. Disse to har  $\max c(l) = 3$ , mens de tre andre har  $\max c(l) = 2$ . Alle disse fem arrayene har  $L_1$ -norm av redundansene lik 4, mens de tre optimale arrayene har lavere høyre ordens norm enn de to andre. F.eks. har de tre optimale arrayene  $L_2(R) = 2$ , mens de to andre har  $L_2(R) = (1 + 2^2 + 1)^{1/2} \approx 2.4$ . Dette kan være en indikasjon på at  $R$  ikke kan minimeres i  $L_1$ -norm for å finne det laveste sidelobenivået. Minimum redundans arrayene karakteriseres ved relativt smal hovedlobebredde og er blant de med lavest max sidelobenivå. Figur 5.3 viser høyre side av coarrayet, elementfordelingen og strålingsdiagrammet for det ikke-optimale MR-arrayet med lavest max sidelobe. Sideloben ved  $u = 0.5$  skyldes hovedsakelig periodisiteter i redundansene.

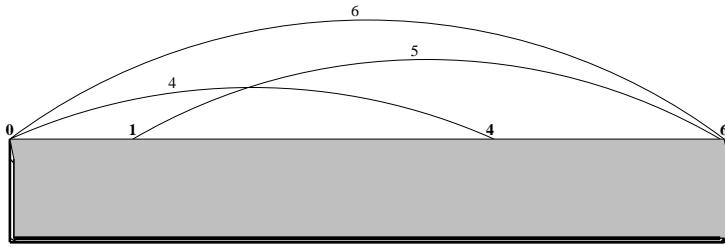
### 5.3 Minimum hull arrayer

Minimum hull (MH) arrayer er de beste ikke-redundante arrayene for et gitt antall elementer, altså konfigurasjoner gitt ved  $\min(H|R = 0)$  ([Dellas *et al.*, 1995, Ver-tatschitsch and Haykin, 1986]). Dette kan også uttrykkes som et differanse basis problem, men minimum hull arrayet er mest kjent som en “Golomb-ruler” ([Gardner, 1972, Bloom and Golomb, 1977]). Disse løsningene kan tolkes som en linjal med lengde  $L$  og som kun har  $k$  posisjonsmerker. Golomb-linjalen er den linjal-



Figur 5.3: Coarray (høyre side), elementfordeling (aktive elementer markert ved  $X$ ) og strålingsdiagram som funksjon av  $u$  for et MR-array for  $N = 7$  og  $M = 18$ .

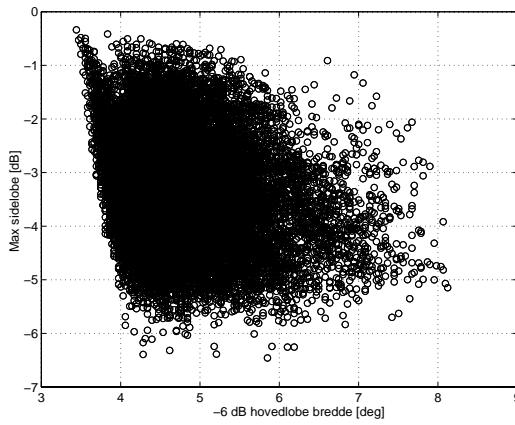
en med  $k$  merker som kan måle flest lengder av  $\{1, 2, \dots, L\}$ . Figur 5.4 viser en perfekt “Golomb-ruler” for  $N = 4$ .



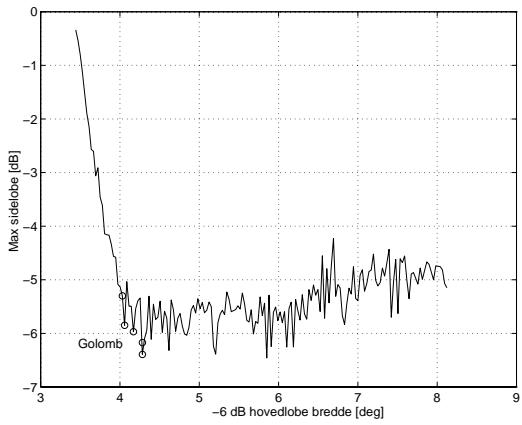
Figur 5.4: Perfekt "Golomb-ruler" for  $N = 4$ .

Minimum hull problemet har større utfallsrom enn MR-problemet for et gitt antall elementer  $N$ . Dette er en av grunnene til at egenskapene til minimum hull arrayet ikke har fått den samme oppmerksomheten som MR-arrayet.

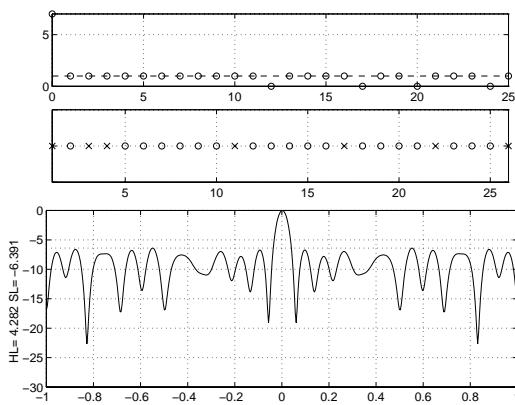
For 7 elementer så finnes det 5 MH-løsninger [Dellas *et al.*, 1995]. Aperturen er her lik 26 elementer. Dette gir  $\binom{24}{5} = 42504$  mulige konfigurasjoner. Figur 5.5 viser fordelingen av maksimal sidelobe og hovedlobebredde. Vi ser at konfigurasjonene med lavest max sidelobenivå ligger spredt i forhold til resten av fordelingen. Figur 5.6 viser den optimale kurven og MH-arrayenes plassering i forhold til denne. Vi ser at MH-arrayene har smal hovedlobe og det laveste max sidelobenivået. Minimum hull arrayene har altså de beste Fourieregenskapene for dette problemet. Figur 5.7 viser coarray, elementfordeling og strålingsdiagram for minimum hull arrayet med lavest max sidelobe. Man kan se at alle sidelobene har tilnærmet lik amplitude. Denne tendensen kommer enda tydeligere frem i figur 5.8 som viser strålingsdiagrammet til det største kjente minimum hull arrayet [Dellas *et al.*, 1995].



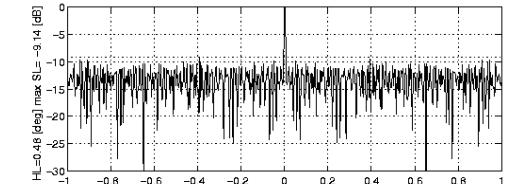
Figur 5.5: Max sidelobenivå og hovedlobebredde for alle arrayene med  $N = 7$  og  $M = 26$ .



Figur 5.6: Optimal max SL–HL kurve og de 5 minimum hull arrayenes plassering i forhold til denne.



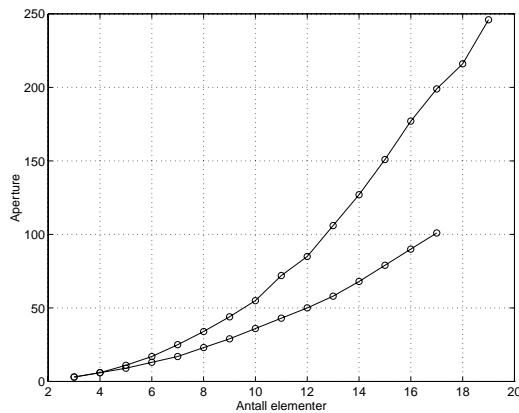
Figur 5.7: Coarray, elementfordeling (aktive elementer markert med  $X$ ) og strålingsdiagram som funksjon av  $u$  for et av minimum hull arrayene for  $N = 7$  og  $M = 26$ .



Figur 5.8: Det største kjente Golomb-arrayet, hvor  $N=19$  og  $M=247$  og de aktive elementposisjonene er 1, 5, 14, 16, 43, 57, 60, 78, 94, 117, 127, 139, 147, 175, 215, 222, 241, 246 og 247.

## 5.4 Arrayer med hull og redundans

Både minimum hull arrayer og minimum redundans arrayer er tilnærminger til et perfekt array. Motivert ut fra opplosningsegenskapene viser [Linebarger *et al.*, 1993] et eksempel på at man kan få en bedre tilnærming i  $L_1$ -norm ved å tillate både redundanser og hull. Altså arrayer som tilfredsstiller  $\min(R + H)$ . Dermed bør man optimere arrayer mellom de to kurvene i figur 5.9, som er basert på tall fra [Linebarger *et al.*, 1993] og [Dellas *et al.*, 1995].



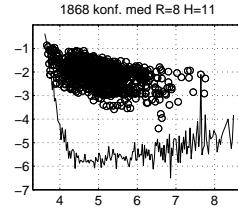
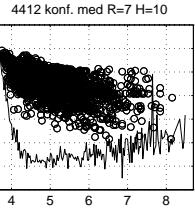
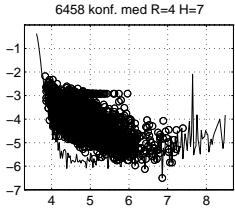
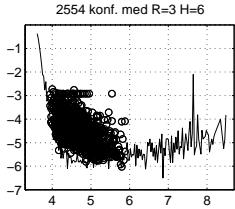
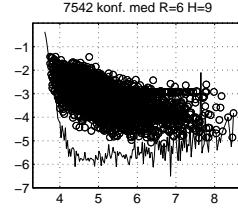
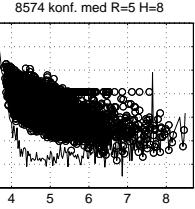
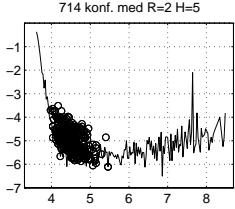
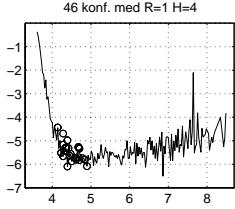
Figur 5.9: Aperturen til MH-arrayene (øvre kurve) og aperturen til MR-arrayene (nedre kurve) som funksjon av antall elementer  $N$  i aperturen.

For  $N = 7$  elementer så har minimum redundans arrayet aperture  $M_{MR} = 18$  og minimum hull arrayet har aperture  $M_{MH} = 26$ . Vi skal nå se på Fourierdomenet til alle arrayene mellom  $M_{MR}$  og  $M_{MH}$ , altså arrayer med både redundanser og hull.

Arrayene i tabell 5.1 har det laveste sidelobenivået for en relativt smal hovedlobe og/eller har den laveste  $L_1$ -normen av  $R$  og  $H$  for hver aperture  $M = 18, 19, \dots, 26$ . Hovedlobebredden er normalisert til  $M \sin \phi$ , siden 1. nullpunkt til et fullt array er  $\sin \phi = 4\pi/Md$  hvor  $d$  er elementavstanden. Teoretisk middelverdi for stokastisk tynne arrayer er  $-10 \log_{10} N = -8.45$  dB. Alle verdiene i tabell 5.1 er lavere enn dette. Minimum redundans arrayet ( $M = 18$ ) og minimum hull arrayet ( $M = 26$ ) har en liten normalisert hovedlobebredde og et lavt max sidelobenivå. Ingen arrayer har lavere max sidelobenivå for denne hovedlobebredden. Minimum hull arrayet har lavere max sidelobe enn minimum redundans arrayet, men et array med samme aperture som sistnevnte og  $R = 5, H = 1$  har det laveste max sidelobenivået av alle arrayene. For  $M = 18$  kan man altså oppnå et bedre max sidelobenivå ved å tillate et hull i coarrayet. Arrayet med  $M = 21$  og

Tabell 5.1: Egenskapene til arrayene med minimum antall redundanser ( $R$ ) og minimum antall hull ( $H$ ) for  $M=18,19,\dots,26$ . Arrayene med lavest max  $SL$  er valgt i hvert tilfelle. Middel  $SL$  og middel(max  $SL$ ) er beregnet fra de lineære verdiene.

$M$	$R$	$H$	# arrayer	$M \sin(HL)$	max( $SL$ ) [dB]	middel( $SL$ ) [dB]	max $SL$ – middel (max $SL$ ) [dB]	Kommentar
18	4	0	5	1.94	-6.34	-9.75	1.88	Min. redundans arrayet
	5	1	112	2.06	-6.86	-9.99	0.79	Lavest max SL, men ikke et MR-array.
19	4	1	40	2.18	-6.51	-10.00	1.33	
20	3	1	10	2.02	-6.76	-9.69	1.60	
21	2	1	1	1.90	-5.72	-9.53	1.92	Beste tilnærming til et perfekt array, men har høyest max SL og det største max SL avviket
	3	2	68	2.18	-6.77	-9.92	0.94	Større verdier av $R + H$ , men bedre max SL
22	2	2	13	2.09	-6.70	-9.66	1.01	
23	2	3	49	2.10	-6.28	-9.67	1.37	
24	1	3	10	1.93	-6.19	-9.57	1.14	
25	1	4	46	1.91	-6.09	-9.62	1.06	
26	0	4	5	1.94	-6.39	-9.40	0.62	Min. hull arrayet har det laveste max SL nivået for denne hovedlobebredden, i tillegg til det minste max SL avviket



Figur 5.10: Hovedlobebredde [deg] og max sidelobe [dB] som funksjon av  $H$  og  $R$  for  $M = 25$  og  $N = 7$ .

Figur 5.11: Hovedlobebredde [deg] og max sidelobe [dB] som funksjon av  $H$  og  $R$  for  $M = 25$  og  $N = 7$ .

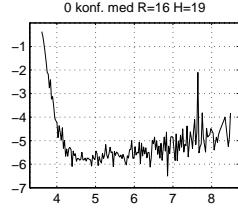
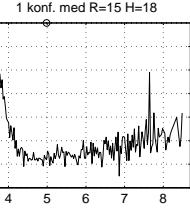
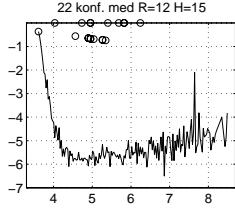
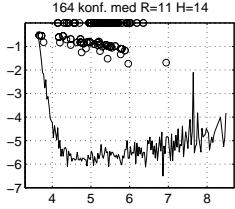
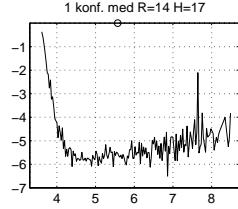
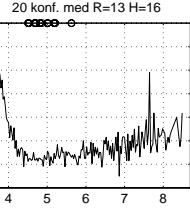
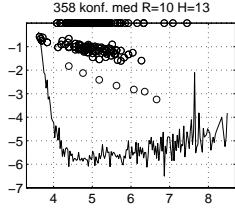
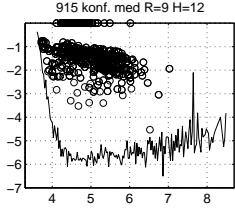
$R = 2, H = 1$  er den beste tilnærmingen til et perfekt array i  $L_1$  forstand. Dette arrayet har det største max sidelobenivået av alle arrayene i tabell 5.1. Sett ut fra Fourieregenskapene er det altså ikke hensiktsmessig å la summen av  $R$  og  $H$  bli for liten. Et perfekt array er heller ikke ønskelig siden det gir et strålingsdiagram med fallende sidelober hvor 1. sidelobe er  $-6.7$  dB. Differansen mellom max sidelobenivå og middelet av maksimumsverdiene til sidelobenivået uttrykker avviket fra en Dolph-Chebyshev sidelobefordeling. Minimum hull arrayet har det minste avviket fra middelverdien og minimum redundans arrayet har et av de største avvikene. Denne tendensen forsterkes for større arrayer hvor sidelobefordelingen til minimum hull arrayet er tilnærmet uniform.

## 5.5 Coarrayet til optimale arrayer

Vi skal se nærmere på hva som skiller coarrayet til arrayene med lavest sidelobenivå fra resten. Det er hovedsaklig 3 karakteristiske egenskaper:

### 5.5.1 Flathet

Minimum hull og minimum redundans arrayer er gode tilnærninger til et perfekt array. Vi har sett at disse har lavt sidelobenivå og smal hovedlobebredde. Fra ligning (5.9) så har man at  $H - R = \text{konst.}$  for gitt  $N$  og  $M$ . Vi skal se nærmere på en av aperturene i tabell 5.1. For  $M = 25$  så finnes det  $\binom{23}{5} = 33649$  mulige konfigurasjoner. Dette er det samme tynningsproblemet som i kap. 4.1.



Figur 5.12: Hovedlobebredde [deg] og max sidelobe [dB] som funksjon av  $H$  og  $R$  for  $M = 25$  og  $N = 7$ .

Figur 5.13: Hovedlobebredde [deg] og max sidelobe [dB] som funksjon av  $H$  og  $R$  for  $M = 25$  og  $N = 7$ .

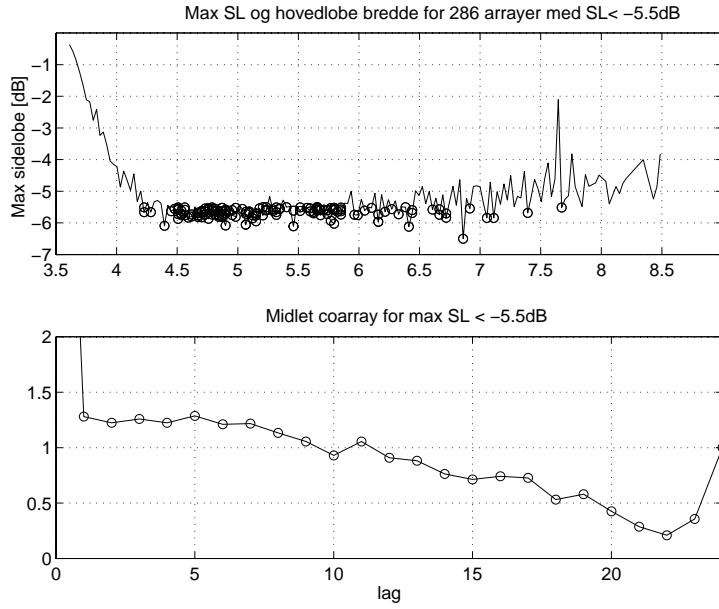
Figurene 5.10 – 5.13 viser hovedlobebredde og max sidelobe for alle eksisterende kombinasjoner av  $H - R$ . Man kan se at det er 46 konfigurasjoner med  $H = 4$  og  $R = 1$ . Disse har smal hovedlobebredde og lavt max sidelobenivå. For større verdier av  $H$  og  $R$  så blir løsningene liggende i en sky som strekker seg fra smal hovedlobebredde med relativ høy max sidelobe til bred hovedlobe med lav max sidelobe. Man kan se at andelen med gode løsninger minker med økende verdi av  $H$  og  $R$ . I tabell 5.1 så man at arrayer med små verdier av  $H$  og  $R$  ikke alltid inneholder arrayet med lavest max sidelobe for en relativt smal hovedlobe. Dette gjelder arrayer uten hull (MR) og arrayer hvor summen av  $H$  og  $R$  er liten.

### 5.5.2 Hullenes plassering

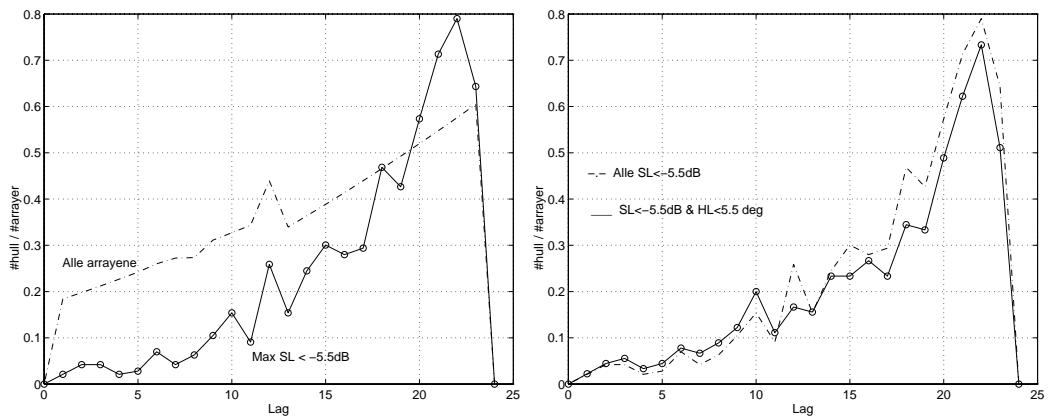
Vi ser på det samme problemet som i forrige avsnitt. Kun 0.85% av alle konfigurasjonene har maksimalt sidelobenivå under  $-5.5$  dB. Utvalgets plassering i forhold til den optimale kurven og utvalgets gjennomsnittlige coarray er gitt i figur 5.14. I dette avsnittet skal man se nærmere på plasseringen av hull for dette utvalget sammenlignet med alle konfigurasjonene.

Figur 5.15 viser gjennomsnittlig antall hull som funksjon av baselinjene. Man kan se at utvalget med lav max sidelobe har færre hull for små baselinjer enn hva tilfellet er for alle arrayene.

I figur 5.16 så er det samme utvalget i tillegg sortert på hovedlobebredde. Utvalget med smal hovedlobe har litt færre hull for store baselinjer og noen flere for små baselinjer enn hva tilfellet er for hele utvalget. Likevel så skiller det seg ikke vesentlig fra utvalget uavhengig av hovedlobebredde.

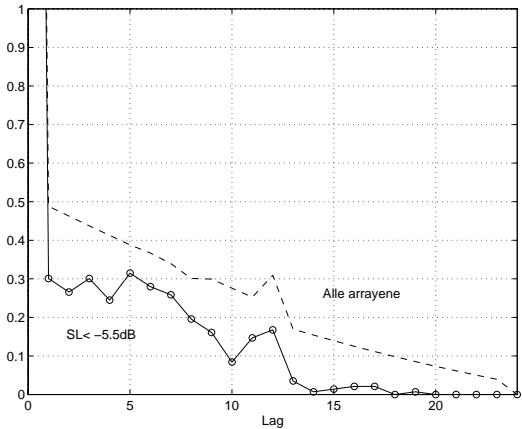


Figur 5.14: Øverst: Arrayene med max  $SL < -5.5$  dB relativt til kurven over de optimale verdiene. Nederst: Midlet coarray for utvalget.

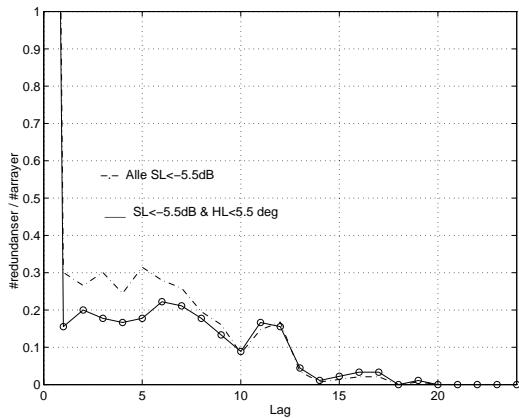


Figur 5.15: Gjennomsnittlig antall hull som funksjon av baselinjene for arrayene med max  $SL < -5.5$  dB sammenlignet med alle arrayene.

Figur 5.16: Gjennomsnittlig antall hull som funksjon av baselinjene for arrayene med max  $SL < -5.5$  dB og hovedlobebredde  $HL < 5.5^\circ$  sammenlignet med alle arrayene.



Figur 5.17: Gjennomsnittlig antall redundanser som funksjon av baselinjene for arrayene med  $\max SL < -5.5 \text{ dB}$  sammenlignet med alle arrayene.



Figur 5.18: Gjennomsnittlig antall redundanser som funksjon av baselinjene for arrayene med  $\max SL < -5.5 \text{ dB}$  og hovedlobebredde  $HL < 5.5^\circ$  sammenlignet med alle arrayene.

Hull i coarrayet for store baselinjer er altså gunstig for et lavt sidelobenivå. Fra ligning (5.11) ser man at dette svarer til manglende hurtig oscillante cosinusledd i strålingsdiagrammet. Man kan forklare dette ved at disse cosinusleddene vil forsterke deler av sidelobene i hele spekteret til cosinusleddene som oscillatorer langsommere, slik at man vil få høye smale sideløb. Dette forklarer også hvorfor minimum redundans arrayene ikke har det laveste sidelobenivået. I tillegg har vi sett at hovedlobebredden er relativt uavhengig av hullenes plassering.

### 5.5.3 Redundansenes plassering

Tilsvarende kan man se på plasseringen av redundansen for utvalget med max sidelobe under  $-5.5 \text{ dB}$ . Figur 5.17 viser gjennomsnittlig antall redundanser som funksjon av baselinjene. Utvalget med max sidelobe under  $-5.5 \text{ dB}$  har nesten ingen redundanser for store baselinjer. Man kan også se at utvalget har gjennomsnittlig færre redundanser for samtlige baselinjer enn for alle arrayene.

I figur 5.18 så er utvalget i tillegg sortert på hovedlobebredde slik som i forrige avsnitt. Utvalget med smal hovedlobe har færre redundanser enn hele utvalget. For store baselinjer er det fortsatt nesten ingen redundanser.

Man har altså sett at coarrayet bør ha få redundanser for at arrayet skal ha et lavt sidelobenivå. Gjennomsnittlig antall redundanser minker med økende baselinjeverdi og for store baselinjer er middelverdien nær null. Redundanser for små

baselinjer medfører noe bredere hovedlobe. Man ser av ligning (5.11) at bidraget fra en baselinje  $l_1$  med coarrayverdi  $c(l_1) = 2$  (altså  $R = 1$ ) er 2 ganger større enn om  $c(l_1) = 1$  (ikke-redundant). Langsomt oscillerende cosinusledd vil ut fra betrakningen til forrige avsnitt forsterke hele sideloben til hurtig oscillerende cosinusledd, men bare for deler av spekteret. For å vise et eksempel på dette er arrayet med lavest max sidelobe for  $M = 21$  og  $N = 7$  gitt i figur 5.19. Coarrayet har redundanser for små baselinjer og hull for store baselinjer slik man har observert gunstig for lavt max sidelobenivå. Figur 5.20 sammenligner strålingsdiagrammet med de ulike bidragene gitt av ligning (5.12):  $W = N - 1 + W_P + W_R - W_H$  hvor  $W_P$  er bidraget fra det perfekte arrayet,  $W_R$  bidraget fra redundansene og  $W_H$  bidraget fra hullene. Det øverste plottet viser bidraget fra det perfekte arrayet i forhold til strålingsdiagrammet  $W$ . I det midterste plottet er bidraget fra redundansene sammenlignet med strålingsdiagrammet. Redundansene forsterker alle de første sidelobene, men reduserer den relative sidelobe amplituden for større vinkler slik som antatt. Det nederste plottet viser hullenes bidrag i forhold til strålingsdiagrammet. Siden hullene opptrer for store baselinjer, så oscillerer hullenes bidrag mer enn hva tilfellet er for redundansene. Dermed forsterkes bare deler av potensielle max sidelober.

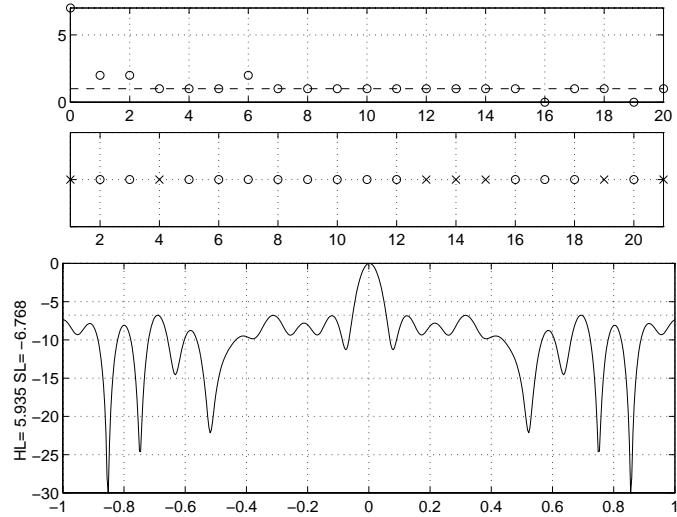
## 5.6 Coarrayet til arrayer med betingelser

I kapittel 4.3 ble arrayer klassifisert etter egenskaper i elementfordelingen og Fouriergenskapene deres ble sammenlignet. I dette kapittelet har man sett at minimum hull arrayer og minimum redundans arrayer har hhv. optimale og nærmest optimale korrelasjons- og Fouriergenskaper. For å gjøre sammenligningen komplett skal man nå se på coarrayet til arrayene med betingelser i elementfordelingen.

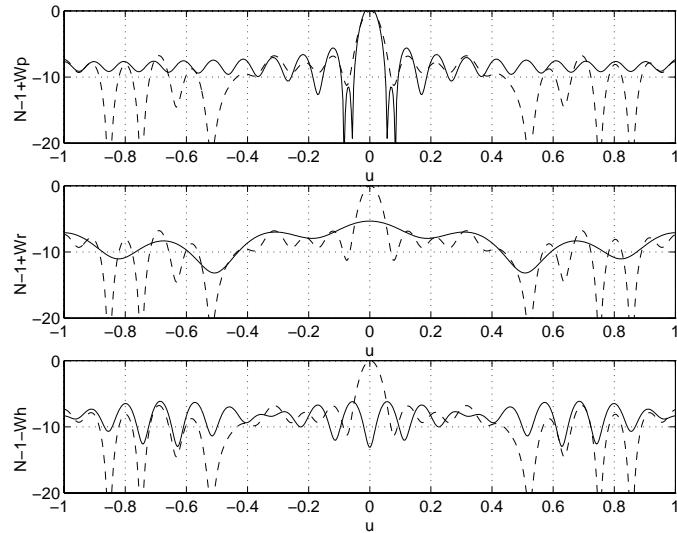
### 5.6.1 Periodisiteter i elementfordelingen

Periodiske arrayer gir gitterlober i strålingsdiagrammet. Coarrayet til et periodisk array med  $M = 25$  elementer i aperturen og  $N = 7$  aktive elementer er gitt i figur 5.21. Elementavstanden  $4d$  i arrayet gjør at kun baselinjer som er et multiplum av 4 bidrar i strålingsdiagrammet.

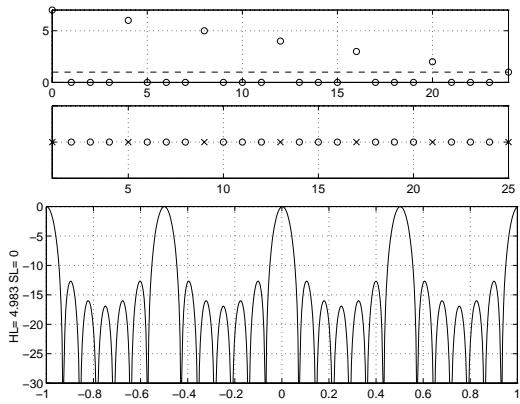
Ikke-periodiske arrayer kan likevel ha periodisiteter i coarrayet. Et eksempel på dette er vist i figur 5.22 hvor bla. baselinje 4, 8 og 12 er redundante. Bidraget fra disse baselinjene til strålingsdiagrammet er gitt i det nederste plottet. Formen og tildels amplituden til sidelobefordelingen domineres totalt av disse baselinjene.



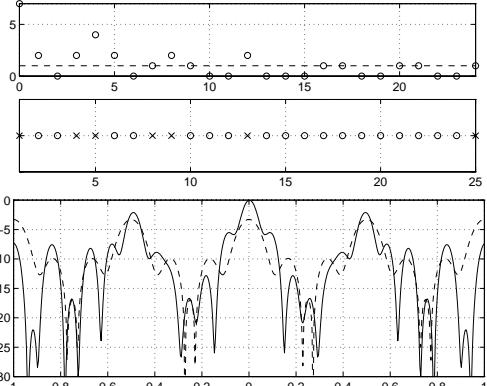
Figur 5.19: Coarray, elementfordeling (aktive elementer markert med X) og strålingsdiagram for arrayet med lavest max sidelobe for  $M = 21$  og  $N = 7$ .



Figur 5.20: Strålingsdiagrammet  $W = N-1 + W_P + W_R - W_H$  sammenlignes med de ulike bidragene for arrayet i figur 5.19. Øverst: Bidraget fra det perfekte arrayet i forhold til strålingsdiagrammet (stiplet). Midten: Bidraget fra redundansene sammenlignet med strålingsdiagrammet (stiplet). Nederst: Bidraget fra hullene i forhold til strålingsdiagrammet (stiplet).



Figur 5.21: Coarray, elementfordeling (aktive elementer markert med x) og strålingsdiagram for et periodisk arrayet med  $M = 25$  elementer i aperturen og  $N = 7$  aktive elementer.



Figur 5.22: Coarray, elementfordeling (aktive elementer markert med x) og strålingsdiagram for et arrayet med  $M = 25$  elementer i aperturen og  $N = 7$  aktive elementer. Bidraget fra baselinjene  $l = 0, 4, 8, 12$  til strålingsdiagrammet er stiplet i nederste plott.

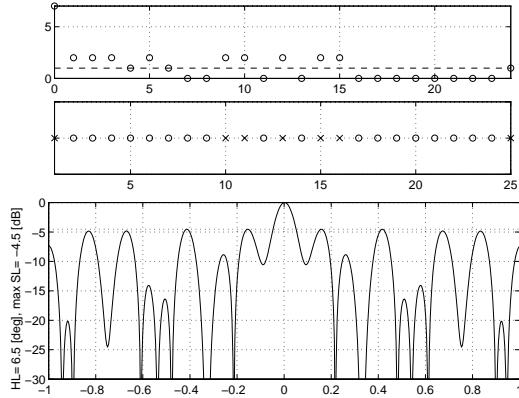
## 5.6.2 Symmetri i elementfordelingen

I kapittel 4.3.2 så man at symmetriske arrayer generelt hadde høy max sidelobe verdi i forhold til de asymmetriske arrayene. Symmetri i elementfordelingen gjør at man får veldig mange redundanser (og hull), noe som ødelegger for flatheten i coarrayet. Figur 5.23 viser strålingsdiagrammet til det symmetriske arrayet med lavest max sidelobe for  $N = 7$  og  $M = 25$ . Dette arrayet skiller seg ut fra de andre symmetriske arrayene ved at  $\max c(l) = 2$ , og har dermed en viss flathet i coarrayet. Likevel er det stor dynamikk i sidelobefordelingen med mange høye sidelober, som også skyldes periodositeter i coarrayet. Større symmetriske arrayer har større frihetsgrad og kan til en viss grad unngå periodositeter i coarrayet, men symmetriske arrayer vil generelt ha mange hull og redundanser. Ut fra observasjonene i avsnitt 5.5.1 vil derfor symmetriske arrayer ikke ha både smal hovedlobe og lavt sidelobenivå.

## 5.6.3 Coarrayet til binnede arrayer

Binnede arrayer har betingelser på den stokastiske elementfordelingen gitt ved (4.14), slik at elementavstanden kan uttrykkes ved

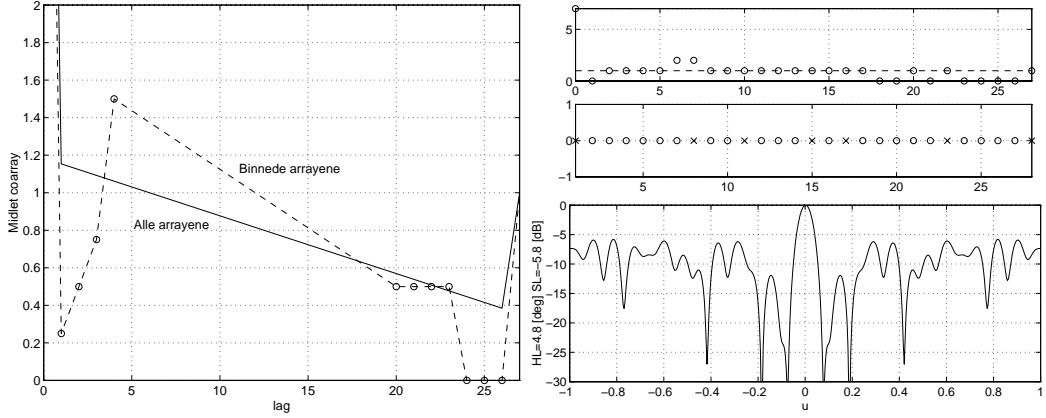
$$X_m - X_n = (m - n)b + (Y_m - Y_n) \quad (5.14)$$



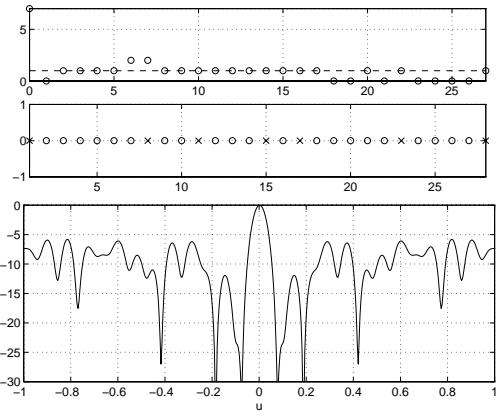
Figur 5.23: Coarray, elementfordeling (aktive elementer markert med x) og strålingsdiagram som funksjon av  $u$  til det symmetriske arrayet med lavest max sidelobe for  $N = 7$  og  $M = 25$ .

hvor  $1 \leq n < m \leq N$ . Ved samme sannsynlighets-tethetsfunksjon på hver binne så vil  $\mathcal{E}[\mathbf{Y}_m] = \mathcal{E}[\mathbf{Y}_n]$  og forventningen til coarrayet vil være et multiplum av  $b$ . Figur 5.24 viser midlet coarray for de binnede arrayene, sammenlignet med det midlede coarrayet for alle arrayene, som funksjon av baselinjene for det samme tynningsproblemet som i figur 4.12. Man kan se at det midlede coarrayet for binnede arrayer har et maksimum ved baselinjeverdi 4, som er identisk med binnelengden  $b$ . For de 3 minste og de 3 største baselinjene (ser bort fra  $l = 27$  som skyldes aktive ende-elementer) så er det midlede coarrayet betydelig mindre for det binnede arrayet. Altså er bidraget fra langsomt oscillerende og hurtig oscillerende cosinuser fraværende i strålingsdiagrammet. I kapittel 5.5.2 observerte man at arrayer med lave max sidelobe verdier hadde sjeldent hull ved små baselinjer. Således burde man anta at binnede arrayer ikke har lave max sidelobe verdier.

Figur 5.25 viser coarrayet, elementfordelingen og strålingsdiagrammet til det binnede arrayet med lavest max sidelobe. Dette arrayet har kun ett hull ved små baselinjer og redundansene er også lokalisert ved små baselinjer. Denne trenden kan man også observere for de andre binnede arrayene med lave max sidelober. For arrayer med samme antall hull og redundanser så forventes andelen av arrayer med lavt sidelobenivå å minke med økende  $H$  og  $R$  ut fra observasjonene i kapittel 5.5.1. Denne trenden kan også observeres for binnede arrayer. For tynningsproblemet hvor  $M = 28$  og  $N = 7$  så er de globale verdiene  $H_{\min} = 6$ ,  $R_{\min} = 0$ , mens for de binnede arrayene så er  $H_{\min} = 8$ ,  $R_{\min} = 2$ . Dette forklarer også hvorfor binnede arrayer ikke har globale minimum max sidelobe verdier.



Figur 5.24: Midlet coarray for alle binnede arrayer sammenlignet med alle arrayene med  $N = 7$  aktive elementer og  $M = 28$  elementer i aperturen.

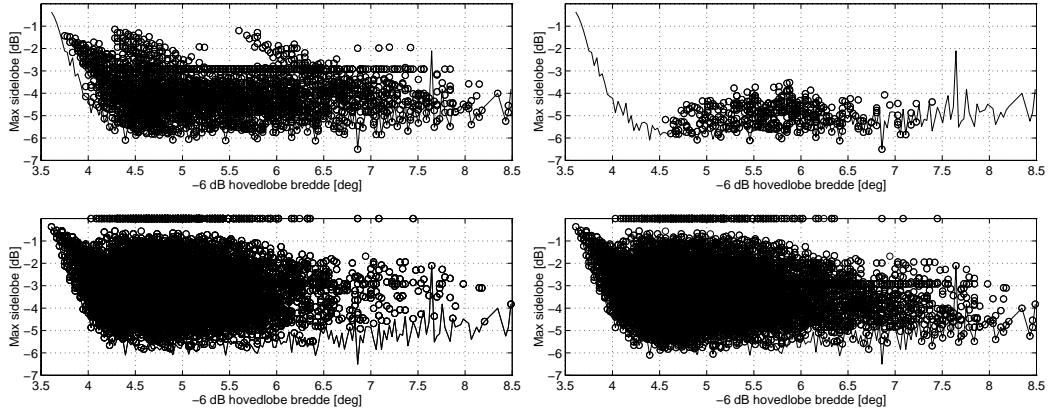


Figur 5.25: Coarray, elementfordeling (aktive elementer markert med x) og strålingsdiagram for det binnede arrayet med lavest max sidelobe for  $M = 28$  og  $N = 7$ .

## 5.7 Arrayer med coarray betingelser

Ved å sette begrensninger på coarrayet istedenfor elementfordelingen vil man oppnå optimale Fourieregenskaper. Betingelser av type minimum hull og minimum redundans gir smale hovedlober og optimalt lave max sidelober. Disse arrayene er nesten like vanskelige å finne i korrelasjonsdomenet som i Fourierdomenet. Desto strengere betingelser man setter på coarrayet desto vanskeligere er det å relatere disse betingelsene til elementfordelingen. Gjennom flere eksempler fra tynningsproblemer med aperturer  $M = 18, 19, \dots, 28$  og  $N = 7$  aktive elementer har man observert trender i coarrayet i forhold til Fourieregenskapene. For noen av disse egenskapene trenger man ikke å beregne hele coarrayet. Ett eksempel er hentet fra utfallsrommet til arrayene med  $N = 7$  og  $M = 25$ :

I kapittel 5.5.2 observerte man at arrayer med lavt max sidelobe nivå hadde færre hull i coarrayet for små baselinjer enn andre. En mulig begrensning er derfor å kreve at  $c(l) > 0 \forall l \leq L$ , altså ingen hull for baselinjer mindre enn eller lik  $L$ . Altså trenger man kun å beregne de  $L$  første baselinjene i korrelasjonen. I figur 5.26 og 5.27 er utvalgene og avskjæringene sammenlignet med de optimale hovedlobebreddene og max sidelobe verdiene for  $L = 6$  og  $L = 11$ . I dette tilfellet så ligger utvalget nærmere den optimale kurven for økende  $L$  for t.o.m  $L = 18$  (finnes ingen arrayer for  $L = 19$ ). Tilsvarende resultat kan oppnås for  $N = 7$  og verdier av  $M$  ned mot  $M = 19$ . For  $M < M_{MR}$  så kan man oppnå tilsvarende resultater ved å legge begrensninger på redundansene for store baselinjer. Binnede arrayer har betingelser på elementfordelingen og gir strålingsdiagram med hoved-



Figur 5.26: Max sidelobe og hovedlobebredde for  $M = 25$  og  $N = 7$ . Øverst: 6433 arrayer hvor  $c(l) > 0 \forall l \leq 6$ . Nederst: 27216 arrayer som ikke oppfyller betingelsen.

Figur 5.27: Max sidelobe og hovedlobebredde for  $M = 25$  og  $N = 7$ . Øverst: 838 arrayer hvor  $c(l) > 0 \forall l \leq 11$ . Nederst: 32811 arrayer som ikke oppfyller betingelsen.

lobebredder omkring det fulle arrayet og alle mulige max sidelobenivåer, altså et vertikalt snitt i max sidelobe-hovedlobebredde planet. Betingelser på coarrayet er vanskeligere å relatere til elementfordelingen, men gir et horisontalt snitt i max sidelobe-hovedlobebredde planet. Med andre ord lave max sidelobe verdier for nesten alle hovedlobebreddene.

# Kapittel 6

## Simulated Annealing

Det finnes ingen enkel sammenheng mellom minimum max sidelobenivå og elementplasseringen for tynnede arrayer. Dette er et kombinatorisk minimerings problem og er således svært vanskelig å løse. Simulated Annealing<sup>1</sup> er en heuristisk metode som har gitt gode løsninger for en rekke kombinatoriske problemer.

### 6.1 Anvendelser

Kombinatoriske problemer har svært mange løsninger. For å løse kombinatoriske problemer finner man løsninger med den beste kostverdiens, som er et kvalitativt mål for hver konfigurasjon. Svært mange kombinatoriske problemer med mange lokale minima kan ikke løses optimalt i polynomisk tid [Aarts and Lenstra, 1997]. For slike problemer er det derfor interessant å finne nær optimale løsninger. *Lokale søk* algoritmer er en type algoritmer som leter etter den optimale konfigurasjonen blant nabokonfigurasjonene. Simulated Annealing, Tabu søk, Genetisk algoritme og Nevrale Nettverk er metoder som baserer seg på lokale søk [Aarts and Lenstra, 1997]. I denne oppgaven benyttes Simulated Annealing siden denne algoritmen har gitt gode resultater for flere tusen ulike problemer, hvor noen av disse er diskutert i [Aarts and Lenstra, 1997, van Laarhoven and Aarst, 1987, Ingber, 1993, Davis, 1989]. I tillegg er algoritmen relativ enkel å implementere.

For å kunne benytte Simulated Annealing til å optimere en prosess må man:

1. Beskrive alle mulige konfigurasjoner til systemet.
2. Generere stokastiske endringer på konfigurasjonen.
3. Definere en kostfunksjon, som relaterer hver konfigurasjon til en kostverdi, hvor man ønsker å finne minimumsverdien.

---

<sup>1</sup>Simulated Annealing kan oversettes med simulert størkning, men i denne oppgaven benyttes Simulated Annealing siden metoden betegnes med dette i litteraturen.

Disse betingelsene imøtekommes av de fleste optimeringsproblemer, og den fleksible kostfunksjonen gjør at man kan kombinere flere kvalitetsmål for samme prosess.

I [Aarts and Lenstra, 1997] sammenligner man resultatene til ulike lokale søk metoder for “*Travelling Salesman*” problemet. Kort fortalt går dette problemet ut på å finne den korteste reiseruten mellom et gitt antall byer. Dette er et kombinatorisk problem som det er vanlig å teste ut nye metoder på. I [Aarts and Lenstra, 1997] fant man at Simulated Annealing og Genetisk algoritme gav noen av de beste løsningene når man tillater tilstrekkelig lang simuleringstid. For 1000 byer klarte Simulated Annealing å få et midlet avvik fra optimum på 1.18% i løpet av 3t på en SGI Challenge.

Simulated Annealing har gitt gode resultater for optimering av arrayer for sonar [Hayward, 1994] og akustisk radar [Miyashita, 1997]. [Ruf, 1993] benyttet Simulated Annealing til å finne arrayer med liten redundans for radio astronomi. Simulated Annealing har også vært benyttet for å minimere max sidelobenivå for tynnede arrayer. I [Murino *et al.*, 1996] minimerte man max sidelobenivå for et lineært array med aperture  $D = 50\lambda$  med  $N = 25$  aktive elementer på et grid med  $d = \lambda/2$  elementavstand. Både tilfellet med og uten vekting ble undersøkt. Resultatene vil bli diskutert i kapittel 7. I [Trucco and Repetto, 1996] benyttet man fortsatt Simulert Annealing, men kostfunksjonen ble endret til å finne den *vektede* elementkonfigurasjonen for et lineært array med minst mulig aperture og færrest mulig aktive elementer. I tillegg skulle strålingsdiagrammet oppfylle gitte betingelser. Dette gav bedre resultater enn referanseproblem som var optimert med lineær programmering og hadde betingelser om symmetri. Simulated Annealing har blitt benyttet for å optimere 2D-arrayer [Trucco, 1997]. Denne kostfunksjonen minimerte antall vektede elementer hvor strålingsdiagrammet skulle oppfylle gitte betingelser. Resultatene vil bli diskutert i kapittel 8.

## 6.2 Metoden

Simulated Annealing er basert på analogien til størknings-prosessen til en perfekt krystall. En krystall har en minimum energi tilstand. Denne kan nås ved at størkningen har foregått gjennom systemets likevektstilstander. [Kirkpatrick *et al.*, 1983] benyttet dette prinsippet sammen med Metropolis-algoritmen til å finne nærmeste løsninger på kombinatoriske problemer.

### 6.2.1 Analogien

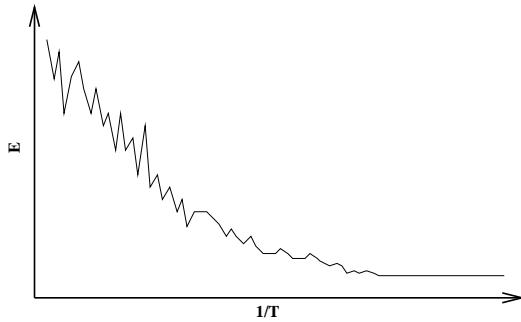
Anta en samling atomer i termisk likevekt; de ulike konfigurasjonene av atomposisjonene er delt i klasser med ulik energi  $E$ , og Boltzmannfordelingen forteller

sannsynligheten for å observere en konfigurasjon ved et bestemt energinivå:

$$P(E) = \frac{1}{Z(T)} \exp\left(-\frac{E}{k_B T}\right) \quad (6.1)$$

hvor  $Z(T)$  er partisjonsfunksjonen,  $T$  temperaturen og  $k_B$  Boltzmanns konstant. Partisjonsfunksjonen<sup>2</sup> er en temperaturavhengig normaliseringsfaktor. Ved høye temperaturer eksisterer det mange konfigurasjoner slik at  $Z(T)$  er stor, mens ved lave temperaturer er det få mulige konfigurasjoner slik at  $Z(T)$  er liten. Metropolis-algoritmen [Kirkpatrick *et al.*, 1983] simulerer den termiske bevegelsen til atomer i kontakt med et varmereservoir med temperatur  $T$ . Den undersøker forandringen i energi  $\Delta E$  ved en liten tilfeldig forandring av systemet. Hvis den nye konfigurasjonen har et lavere energinivå ( $\Delta E < 0$ ), så benyttes den nye konfigurasjonen som utgangspunkt for neste forandring. Derimot, hvis  $\Delta E > 0$ , så benyttes den nye konfigurasjonen med sannsynlighet  $P(\Delta E) = \exp(-\Delta E/k_B T)$ . Ved repetering vil systemet utvikle seg til en Boltzmannfordeling.

Simulated Annealing består i å utføre Metropolis-algoritmen ved de ulike temperaturene  $T$ . For å finne minimum energi tilstanden til et system må man “smelte” systemet for å sikre tilstrekkelig tilstands frihet. Deretter må man senke temperaturen gradvis og tilbringe lang tid omkring størkningspunktet, for å sikre at systemet fryser i minimum energi tilstanden. Dette gir en adaptiv “divide-and-conquer” strategi; grove trekk avdekkes ved høye temperaturer, mens finere trekk kommer frem ved størkningspunktet slik figur 6.1 illustrerer.



Figur 6.1: Energiforløpet til en simulering ved ulike tempetaturer.

### 6.2.2 Algoritmen

For hver konfigurasjon  $\mathbf{X}_i$  finnes det en kostverdi  $C(\mathbf{X}_i)$ . Kostfunksjonen  $C(\mathbf{X})$  definerer et kvalitetsmål for konfigurasjonene  $\mathbf{X}$  som man ønsker å minimere.

---

<sup>2</sup>Se enhver bok om termodynamikk f.eks. [Kittel and Kroemer, 1980].

Man kan f.eks. definere  $C(\mathbf{X}_i)$  som den maksimale sidelobe verdien til array  $\mathbf{X}_i$ . Sannsynligheten for å observere en konfigurasjon med kostverdi  $C(\mathbf{X}_i)$  er gitt ved:

$$P(\mathbf{X} = \mathbf{X}_i) = \frac{1}{Q(c)} \exp\left(-\frac{C(\mathbf{X}_i)}{c}\right) \quad (6.2)$$

hvor  $c$  er en kontrollparameter og  $Q(c)$  er en normaliseringsfaktor tilsvarende partisjonsfunksjonen.  $Q(c)$  sikrer at  $P(\mathbf{X})$  er en sannsynlighets-tetthetsfunksjon i statistisk forstand, altså at

$$\sum_{\mathbf{X}} P(\mathbf{X}) = 1 \quad (6.3)$$

Dette krever at man kjenner kostverdiene til alle konfigurasjonene, noe som er praktiske umulig for dette formålet. Man definerer derfor

$$P(\mathbf{X} = \mathbf{X}_i) = \exp\left(-\frac{C(\mathbf{X}_i)}{c}\right) \quad (6.4)$$

I Simulated Annealing er det mer interessant å undersøke forandringen i kostverdi  $\Delta C_{ij} = C(\mathbf{X}_j) - C(\mathbf{X}_i)$  for en perturbering  $\mathbf{X}_j$ . Derfor defineres

$$P(\Delta C_{ij}) = \exp\left(-\frac{\Delta C_{ij}}{c}\right) \quad (6.5)$$

For å beholde analogien til termodynamikken benyttes notasjonen

$$P(\Delta E_{ij}) = \begin{cases} \exp\left(\frac{-\Delta E_{ij}}{T}\right) & , \Delta E_{ij} \geq 0 \\ 1 & , \Delta E_{ij} < 0 \end{cases} \quad (6.6)$$

Simulated Annealing består av iterative forbedringer med en stokastisk komponent. Iterative metoder starter med en kjent konfigurasjon og gjør en standard omorganisering (perturbering) av konfigurasjonen. Alle konfigurasjonene man kan nå med én slik omorganisering definerer nabokonfigurasjonene. Etter den siste perturberingen ved temperatur  $T_i$  antas systemet å være i likevekt, dvs. at  $\Delta E$  er Boltzmannfordelt. Etter at temperaturen er redusert til  $T_{i+1}$  vil ikke systemet lengre være i likevekt, og må derfor perturberes tilbake til likevektstilstanden. Ved høye temperaturer vil nesten alle økninger av kostfunksjonen bli godtatt. Dette hindrer algoritmen fra å bli sittende fast i lokale minima. Derimot vil slike kostøkninger avta i sannsynlighet for minkende temperatur, og ved systemets kritiske temperatur vil ingen kostøkninger bli akseptert. Systemet antas da å være i det globale minimum.

Figur 6.2 viser strukturen i en Simulated Annealing algoritme. Man starter med en initiell konfigurasjon  $\mathbf{X}_0$  og bestemmer et temperatur forløp  $T_0 > T_1 > \dots > T_{num\_iter}$  hvor  $T_0$  svarer til temperaturen hvor systemet er flytende og

```

Procedure Simulated Annealing
begin
    initialiser( $X_0, T_0$ )
    for  $i=1$  to  $num\_iter$ 
         $T_i = temp(T_{i-1})$ 
        for  $j=1$  to  $num\_pert$ 
             $X_p = perturber(X_{j-1})$ 
             $\Delta E = E(X_p) - E(X_{j-1})$ 
            if  $\Delta E < 0$  or  $\exp\left(-\frac{\Delta E}{T_i}\right) < \text{rnd}[0,1]$ 
                 $X_j = X_p$ 
            else
                 $X_j = X_{j-1}$ 
            endif
        endfor
    endfor
end

```

Figur 6.2: Simulated Annealing strukturen.  $\text{rnd}[0,1]$  returnerer en reell verdi uniformt fordelt på intervallet  $[0, 1]$ .

$T_{num\_iter}$  temperaturen hvor systemet er frosset. For hver temperatur  $T_i$  gjør man  $num\_pert$  perturberinger for å bringe systemet tilbake til likevekt (Boltzmannfordelte data). Perturberingen  $X_p$  aksepteres med sannsynlighet  $P(\Delta E)$  og er gitt i lign. (6.6).

## 6.3 Parametre

Konvergensegenskapene til Simulated Annealing algoritmen avhenger av Parametrene. Verdiene til disse vil variere fra problem til problem, slik at prøving og feiling må til for å bestemme de beste verdiene. Sannsynligheten for å nå det globale optimum avhenger av:

- Temperaturforløpet
  1. Initialverdien  $T_1$
  2. Sluttverdien  $T_{num\_iter}$
  3. Dekrementeringen  $\Delta T_i$
- Nabokonfigurasjonene
  1. Naborommet
  2. Antall perturberinger pr iterasjon

### 6.3.1 Temperaturen

Temperaturen er selve kontrollparameteren i algoritmen. Den styrer sannsynligheten for hvor store kostøkninger algoritmen aksepterer.

#### Temperaturens initialverdi

$T_1$  er temperaturen ved første iterasjon. Verdien til  $T_1$  bør være slik at flest mulig perturberinger aksepteres i henhold til analogien om at systemet skal være flytende. Det er hensiktsmessig å definere en akseptrate  $\chi_1$ , som definerer andelen av aksepterte kostøkninger ved første iterasjon.

$$\begin{aligned}\chi_1 &= P(\Delta E^+) = \exp\left(\frac{-\overline{\Delta E}^+}{T_1}\right) \\ \Rightarrow T_1 &= \frac{-\overline{\Delta E}^+}{\ln \chi_1}\end{aligned}\tag{6.7}$$

$\overline{\Delta E}^+$  er midlet økning av kostfunksjonen. Denne metoden krever mye eksperimentering da  $\chi_1$  bestemmes uavhengig av data.  $\overline{\Delta E}^+$  kan bestemmes for noen få iterasjoner av algoritmen.

I [Aarts and Lenstra, 1997] argumenterer man for at

$$T_1 = \Delta E_{\max} \quad (6.8)$$

Den initiale temperaturen bør være lik de maksimale kostdifferansene mellom to nabokonfigurasjoner, altså konfigurasjoner som er perturberinger av nåværende konfigurasjon. Dette medfører at  $P(\Delta E_{\max}) = e^{-1} \approx 0.37$  for første iterasjon.

En tredje metode antar at konfigurasjons-tetthetsfunksjonen  $\omega(E)$  er gaussisk og ved analogi til termodynamiske relasjoner krever man at [van Laarhoven and Aarst, 1987]:

$$T_1 \geq \sigma_\infty \quad (6.9)$$

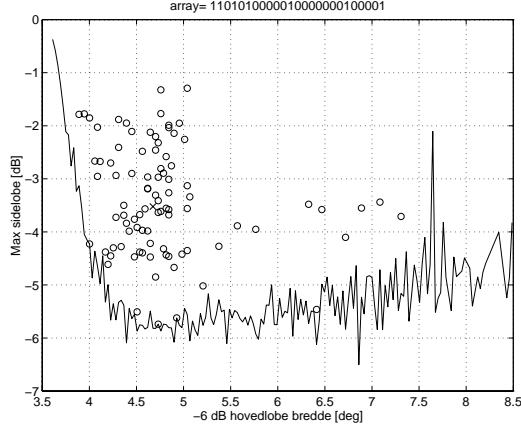
hvor  $\sigma_\infty$  er standardavviket til  $\omega(E)$  når temperaturen  $T$  er stor. (6.8) og (6.9) krever at man undersøker hele eller deler av nabokonfigurasjonsrommet. For et tynnet array kan antall nabokonfigurasjoner  $R$  defineres som antall mulige arrayer ved å endre ett vilkårlig element:

$$R = (M - N)N_{frie} \quad (6.10)$$

$M$  er antall elementer i fullt array,  $N$  er antall aktive elementer og  $N_{frie}$  er antall elementer uten begrensninger. For eksempelet man har studert tidligere med et lineært array hvor  $M = 25$  og  $N = 7$ , slik at  $N_{frie} = 5$  når man antar aktive ende-elementer, så finnes det  $R = 90$  nabokonfigurasjoner. For et stokastisk tynnet array med max sidelobe lik  $-3.5$  dB varierer max sidelobe verdien til nabokonfigurasjonene fra  $-5.8$  dB til  $-1.2$  dB slik man kan se i figur 6.3. Man legger merke til at max sidelobe verdiene for nabokonfigurasjonene strekker seg helt ned mot de optimale verdien. Dette skyldes at det er få elementer i arrayet, slik at perturbering av ett element endrer  $1/7$  av strålingsdiagrammet. Hvis kostfunksjonen er definert som max sidelobe verdi til strålingsdiagrammet, så gir (6.8)  $T_1 = 0.36$  og (6.9)  $T \geq 0.08$ . Til sammenligning gir en initiell akseptrate  $\chi_1 = 0.70$  initiell temperatur  $T_1 = 0.23$  fra ligning (6.7), hvor den midlede kostøkningen  $\overline{\Delta E}^+$  er beregnet ut fra 100 tilfeldig perturberinger. For dette problemet gir (6.9) den minste verdien for  $T_1$  mens (6.8) gir en verdi som garanterer at systemet er flytende.

## Temperaturens sluttverdi

Algoritmen skal terminere når temperaturen er så lav at ingen flere perturberinger aksepteres, dvs. at  $P(\Delta E)$  blir svært liten. Tilsvarende som for den initielle



Figur 6.3: Max sidelobenivå og -6 dB hovedlobebredde for nabokonfigurasjone-ne (markert med  $O$ ) til et stokastisk tynnet array (markert med  $X$ ) med  $N = 7$  elementer og aperture  $M = 25$  i forhold til de optimale verdiene.

temperaturen kan man kreve at den midlede kostøkningen skal tilsvare en sannsynlighet mindre enn  $\chi_{num\_iter}$  slik at

$$T_{num\_iter} = \frac{-\overline{\Delta E}^+}{\ln \chi_{num\_iter}} \quad (6.11)$$

Merk at man ikke kan benytte samme verdi for  $\overline{\Delta E}^+$  som for den initielle temperaturen, siden det vil gi en større verdi for  $T_{num\_iter}$  enn om man beregnet  $\overline{\Delta E}^+$  over de siste iterasjonene. Problemet er at antall aksepterte perturberinger minker med temperaturen, slik at  $\overline{\Delta E}^+ \rightarrow 0$  når  $T \rightarrow 0$ . Ved å beregne  $\overline{\Delta E}^+$  over alle iterasjonene frem til nåværende tilstand vil man ta hensyn til dette, men det medfører en del ekstra beregninger.

En annen metode relaterer sannsynligheten for den maksimale kostøkningen til størrelsen på nabokonfigurasjonsrommet. Anta et lokalt minimum med kostverdi  $E_0$ . La  $E_1$  være nest høyeste kostnivå blant nabokonfigurasjonene og  $R$  antall nabokonfigurasjoner. For lave verdier av  $T$  gjelder [van Laarhoven and Aarst, 1987]:

$$\exp\left(-\frac{E_1 - E_0}{T}\right) < \frac{1}{R} \quad (6.12)$$

slik at

$$T_{num\_iter} < \frac{E_1 - E_0}{\ln R} \quad (6.13)$$

Ligning (6.12) sier at sannsynligheten for en perturbering, fra et lokalt minimum til nest høyeste nabo, skal være mindre enn den inverse størrelsen på naborommet.

En annen variant av (6.13) definerer sannsynligheten for den minste kostökningen blant nabotilstandene:

$$T_{\text{num\_iter}} = \Delta E_{\min} \quad (6.14)$$

Dette gir  $P(\Delta E_{\min}) = 0.37$ . For det samme eksempelet som for temperaturens initielle verdi så gir (6.13)  $T_{\text{num\_iter}} < 0.076$  og (6.14) gir  $T_{\text{num\_iter}} = 2 \cdot 10^{-6}$ . Ligning (6.14) gir altså veldig lang simuleringstid hvis den minste kostdifferansen mellom to nabover blir liten. Hvis man krever at den midlede kostökningen skal aksepteres med 0.1% sannsynlighet ved terminering, så gir (6.11)  $T_{\text{num\_iter}} = 0.012$ .

### Dekrementering av temperaturen

I termodynamikken må størkningsprosessen foregå gjennom likevektstilstander for å nå energi minimum. Derfor må dekrementeringen av temperaturen være tilstrekkelig liten slik at systemet kan perturberes tilbake i likevekt. Hvis man kun gjør én perturbing pr iterasjon og denne er trukket fra en gaussisk fordeling, så er konvergens mot globalt minimum sikret hvis man velger temperaturfunksjon [van Laarhoven and Aarst, 1987]:

$$T_i \propto \frac{T_0}{\log(i+1)} \quad , i = 1, 2, \dots \quad (6.15)$$

Dette er altfor sakte for de fleste praktiske anvendelser. Man velger derfor en temperaturfunksjon som faller raskere. I litteraturen kan man finne mange ulike temperaturfunksjoner og noen av disse er gjengitt her. En av de vanligste temperaturfunksjonene er konstant dekrementeringsrate for hver iterasjon:

$$T_i = \alpha^i T_0 \quad , i = 1, 2, \dots \quad , 0 < \alpha < 1 \quad (6.16)$$

Typiske verdier for  $\alpha$  ligger mellom 0.8 og 0.99.

En annen funksjon er konstant temperaturdifferanse:

$$T_i = \frac{\text{num\_iter} - i}{\text{num\_iter}} T_0 \quad , i = 1, 2, \dots \quad (6.17)$$

En tredje temperaturfunksjon som faller saktere enn de forrige er en omvendt proporsjonal funksjon:

$$T_i = \frac{T_0}{\beta i} \quad , i = 1, 2, \dots \quad (6.18)$$

Alle disse kjølingsfunksjonene er deterministisk bestemt. For å ivareta likevektstilstanden best mulig bør neste temperaturverdi bestemmes av tilstanden til prosessen.

Ut fra analogi til det termodynamiske uttrykket for spesifikk varmekapasitet vil forventet kostnivå i likevekt tilfredstille [van Laarhoven and Aarst, 1987]:

$$\frac{\partial}{\partial \ln T} \mathcal{E}\{E(T)\} = \frac{\sigma^2(T)}{T} \quad (6.19)$$

hvor  $E(T)$  er kostnivået ved temperatur  $T$  og  $\sigma^2(T)$  er variansen til  $E(T)$ . Tilnærming av (6.19) gir temperaturrelasjon:

$$T_{i+1} = T_i \exp \left[ \frac{T_i}{\sigma^2(T_i)} (\bar{E}(T_{i+1}) - \bar{E}(T_i)) \right] \quad (6.20)$$

Ved å kreve at

$$\bar{E}(T_{i+1}) - \bar{E}(T_i) = -\gamma \sigma(T_i) \quad , 0 < \gamma \leq 1 \quad (6.21)$$

får man

$$T_{i+1} = T_i \exp \left( \frac{-\gamma T_i}{\sigma(T_i)} \right) \quad , i = 1, 2, \dots \quad (6.22)$$

Denne temperaturrefasjonene er i størrelsesorden lik (6.16), men her er dekrementeringsfaktoren bestemt av tilstanden til systemet. Ulempen med tilstandsbestemte temperaturrefasjoner er at de øker prosesseringstiden til algoritmen. Derfor er det i mange tilfeller hensiktsmessig å definere en deterministisk temperaturrefasjon med tilnærmet det samme forløpet som den tilstandsbestemte.

### 6.3.2 Nabokonfigurasjonene

Nabokonfigurasjonene er de tilstandene man kan nå men en standard omorganisering av nåværende tilstand. Disse konfigurasjonene avgjør hvor raskt algoritmen nærmer seg det globale minimum.

#### Naborommet

Antall nabokonfigurasjoner  $R$  og deres spredning i kostverdi er et mål for den stokastiske friheten til algoritmen. I ligning (6.10) var  $R$  definert som antall tynnede arrayer hvor 1 av elementene var annerledes plassert. Snittet av antall arrayer for to nabotilstande definert ved (6.10) er gitt ved:

$$\begin{aligned} R_1 \cap R_2 &= \binom{M-N-1}{1} + N_{frie} - 1 \\ &= M - N - 2 + N_{frie} \end{aligned} \quad (6.23)$$

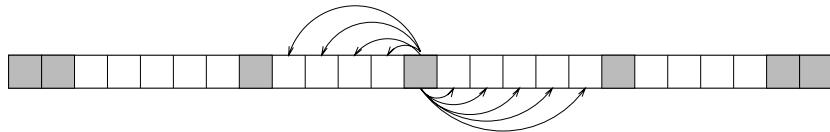
Tilskuddet av nye konfigurasjoner  $R_2 - (R_1 \cap R_2)$  er et mål for sannsynligheten for at algoritmen vil iterere seg ut av et lokalt minimum. Hvis verdien av  $R_2 - (R_1 \cap R_2)$  er liten vil konvergens mot det globale minimum være svært avhengig av initialkonfigurasjonen. Derimot hvis  $R_1 \cap R_2 \rightarrow 0$  er tilskuddet av nye konfigurasjoner veldig stort, og algoritmen vil hoppe mellom nær disjunkte løsningsmengder. Dette medfører at det er liten sannsynlighet for at algoritmen terminerer i det globale minimum.

For  $R$  definert ved (6.10) viste figur 6.3 spredningen i sidelobenivå for de  $R = 90$  nabotilstandene til et tynnet array med  $N = 7$  aktive elementer, hvor  $N_{frie} = 5$ , og aperture  $M = 25$ . Snittet av arrayer for to nabotilstander er  $R_1 \cap R_2 = 21$ . Anta i steden at  $R$  er gitt ved antall tynnede arrayer som har 2 elementer plassert annerledes. I så fall vil antall nabotilstander være gitt ved:

$$R \equiv (M - N)(M - N - 1)N_{frie} \quad (6.24)$$

For det samme eksempelet gir dette  $R = 1530$  og  $R_1 \cap R_2 = 315$ . Ved å endre to elementer vil algoritmen kunne velge to elementer som fører til kansellering i kostøkningen. Dermed kan man risikere at konvergenstiden øker. I senere simuleringer vil antall nabokonfigurasjoner være definert som arrayer som har ett element ulikt plassert, siden det gir en mer entydig endring og færre nabotilstander.

En variant av perturbering av 1 element anvendt på tynnede arrayer er benyttet i [Murino *et al.*, 1996], hvor antall nabotilstander er avhengig av elementet som perturberes. Perturberingselementet kan kun flyttes til de ledige posisjonene begrenset av dets nabøer, slik figur 6.4 illustrerer.

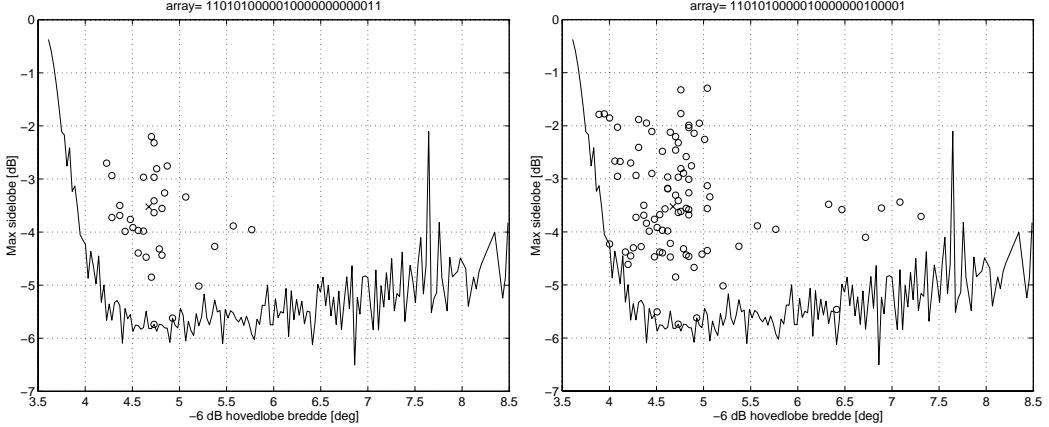


Figur 6.4: Mulige arrayposisjoner hvor perturberingselementet er begrenset av nabøene. Antall nabotilstander til dette arrayet er  $R = 36$ .

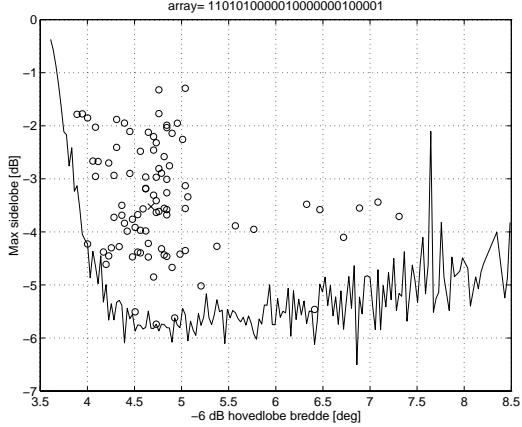
Dette gjør at antall nabokonfigurasjoner  $R$  til et tynnet array med  $N$  aktive elementer og aperture  $M$  er begrenset av:

$$M - N \leq R \leq 2(M - N) \quad (6.25)$$

For eksempelet med  $N = 7$  og  $M = 25$  gir dette  $R \in [18, 36]$ , mens  $R = 90$  uten begrensninger på nabøene. Figur 6.5 viser max sidelobenivå og hobedlobe bredde til de begrensede nabøene til det samme arrayet som i figur 6.3. For å gjøre sammenligningen enklere er en kopi av figur 6.3 gitt i figur 6.6. I figur 6.5 er



Figur 6.5: Max sidelobenivå og  $-6$  dB hovedlobebredde for de begrensede nabokonfigurasjonene til et stokastisk tynnet array (markert med  $X$ ) med  $N = 7$  elementer og aperture  $M = 25$  i forhold til de optimale verdiene.



Figur 6.6: Nabokonfigurasjonene til det samme arrayet som i figur 6.5 men uten begrensninger på perturberingene.

nabokonfigurasjonene som avviker mest i hovedlobebredde eliminert, i tillegg til konfigurasjoner med tilsvynelatende uniformt spredt sidelobenivå. Begrensningene som [Murino *et al.*, 1996] benytter gjør at nabotilstandene får en binnet begrensning, hvor binnelengden er gitt av perturberingselementet sine naboer. Siden binnelengden varierer vil ikke nabotilstandene ha helt de samme egenskapene man observerte i kapittel 4.3.1, men man kan se at det er mindre spredning på hovedlobebredden i figur 6.5 enn i figur 6.6.

### Antall perturberinger

Termisk likevekt oppnås når kostverdiene til perturberingene er Boltzmannfordelt. Anta at systemet er i termisk likevekt. Etter en dekrementering av temperaturen vil ikke likevekten være bevart, slik at man må gjøre perturberinger for å finne den konfigurasjonen som gjenopprettet likevektstilstanden. Ved terminering vil optimaliteten til den endelige konfigurasjonen hovedsakelig bestemmes av temperaturdekrementeringen og antall perturberinger. Det er derfor ønskelig at én av variablene er tilstandsbestemt. Man skiller derfor mellom to hovedkombinasjoner:

- Varialbelt antall pertuberinger bestemt av tilstanden til systemet og et deterministisk temperaturforløp.
- Antall perturberinger gitt a priori og tilstandsbestemt temperaturdekrementering.

Den siste strategien er ofte benyttet hvor antall perturberinger ( $num\_pert$  i figur 6.2) er uavhengig av temperaturen, men avhengig av størrelsen til problemet. Ved optimering av tynnede arrayer med  $N$  aktive elementer og  $M$  elementposisjoner vil altså:

$$num\_pert \propto \binom{M}{N} \quad (6.26)$$

I [Murino *et al.*, 1996] valgte man  $num\_pert = N$ .

Antall perturberinger er også avhengig av hvordan perturberingselementene velges. Ved høye temperaturer vil tilnærmet alle kostøkninger aksepteres, mens ved lave temperaturer aksepteres kun perturberinger som svarer til en negativ eller liten kostøkning. I [van Laarhoven and Aarst, 1987] refererer man til en metode som argumenterer for at store kostøkninger ved høye temperaturer og små kostøkninger ved lave temperaturer fører til raskere konvergens. Med en slik generatingsmetode vil færre perturberinger bli avvist, og algoritmen vil gjøre at grovsøk ved høye temperaturer mens den finere strukturen kommer frem ved lave temperaturer. For problemer med mange lokale minima, slik som elementplassering i et tynnet array, er det vanskelig å relatere perturbingelementet sin posisjon til kostøkingen. Dermed vil det være vanskelig å implementere en slik strategi for tynnede arrayer.

Fordelingsfunksjonen til perturberingene har betydning for hvor raskt temperaturen kan reduseres. Hvis pertureringen er trukket fra en gaussisk fordeling, så kan temperaturen dekrementeres i henhold til (6.15). For problemer hvor kostfunksjonen er reell og kontinuerlig, viser [Ingber, 1989] at perturberinger trukket fra en Cauchy fordeling vil gi en løsning som konvergerer mot globalt minimum, hvor temperaturen faller invers proporsjonalt eller i beste fall eksponentielt med antall iterasjoner. Max sidelobenivå har en diskret struktur, slik at antagelsen til [Ingber, 1989] holder ikke for tynnede arrayer. Fra teorien for stokastiske arrayer [Steinberg, 1976] vet man at det forventede strålingsdiagrammet til et stokastisk tynnet array er tilsvarende til strålingsdiagrammet til en kontinuerlig aperture vektet med fordelingsfunksjonen til det stokastiske arrayet. Fra teorien for vektingsfunksjoner (f.eks. [Oppenheim and Schafer, 1989, Harris, 1978]) vet man at ikke-uniforme vektingsfunksjoner kan gi bredere hovedlobe og lavere max sidelobe enn uniform vekting. For å finne konfigurasjoner med lavt max sidelobenivå, kan det altså være gunstig å trekke perturbingelementer fra en ikke-uniform fordeling. Hvis man velger en slik fremgangsmåte, vil man redusere den stokastiske friheten i algoritmen som en uniform fordeling gir.

# Kapittel 7

## Optimering av 1D arrayer med Simulated Annealing

Et lineært array med 101 elementer i aperturen tynnet til 25 aktive elementer har  $19146258135816088501224$  ( $1.9 \cdot 10^{22}$ ) mulige realisasjoner. Hvis man hadde skrevet ned hvert tynningsmønster på 1cm lange lapper, så ville samtlige realisasjoner gi en papirremse som gikk 2 billioner ganger rundt ekvator. Blant alle disse skal vi i dette kapittelet forsøke å finne *den ene* lappen som representerer arrayet med lavest max sidelobe.

### 7.1 Implementering

I kap. 6 ble det gitt 3 betingelser som et problem må oppfylle for å kunne minimeres med Simulated Annealing:

#### 7.1.1 Beskrive konfigurasjonene

Et lineært tynnet array med  $N$  aktive elementer og  $M$  posisjoner i aperturen med regulær avstand kan beskrives med en binær vektor. La verdien 1 markere posisjonen til de aktive elementene og 0 markere de ledige arrayposisjonene. For det tynnede arrayet i figur 7.1 får man:



Figur 7.1: Tynnet array hvor de mørke feltene viser de aktive elementene.

$$\text{array} = [101100010000010010 \dots \dots \dots 0101000001101101]$$

For å forenkle implementeringen er det gunstig å definere en vektor over arrayposisjonene til de  $N$  aktive elementene

$$konfig = [1 \ 3 \ 4 \ 8 \ 14 \ 17 \ \dots \ M-3 \ M-2 \ M]$$

og til de ledige posisjonene:

$$ledig = [2 \ 5 \ 6 \ 7 \ 9 \ 10 \ \dots \ M-7 \ M-4 \ M-1]$$

Enhver binær sekvens av lengde  $M$  med  $N$  1-ere, hvor første og siste verdi er 1, er altså et mulig tynningsmønster.

### 7.1.2 Stokastiske endringer

I kap. 6.3.2 ble en perturbering definert som endring av ett aktivt element. Posisjonen  $p_{ut}$  til perturberingselementet i vektoren *array* velges stokastisk:

$$p_{ut} = \text{random\_f}[konfig(2) \dots konfig(N-1)] \quad (7.1)$$

hvor *f* er en diskret sannsynlighets-tetthetsfunksjon. Den nye posisjonen til perturberingselementet i vektoren *array* velges stokastisk blant de ledige plassene:

$$p_{inn} = \text{random\_g}[ledig(1) \dots ledig(M-N)] \quad (7.2)$$

hvor *g* også er en diskret sannsynlighets-tetthetsfunksjon. Man kan så definere en perturbert utgave av vektoren *konfig*:

$$\begin{aligned} test\_konfig &\equiv konfig \\ test\_konfig(p_{ut}) &\equiv p_{inn} \end{aligned} \quad (7.3)$$

I samtlige simuleringer i denne oppgaven vil *f* og *g* være uniformt fordelt. Som tidligere nevnt kan dette medføre noe lenger simuleringstid, men man beholder den stokastiske friheten til algoritmen slik at algoritmen kan nå alle mulig løsnings.

### 7.1.3 Kostfunksjonen

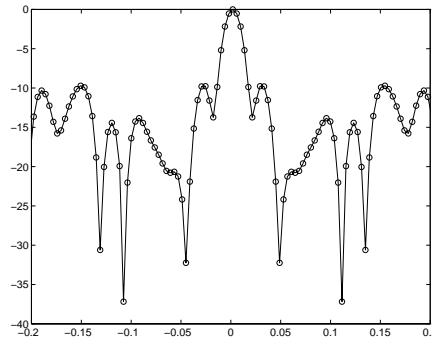
Definisjonen av kostfunksjonen bestemmer hvilke egenskaper ved det tynnede arrayet man ønsker å kontrollere. De vanligste kvalitetsmålene for tynnede arrayer er max sidelobenivå og hovedlobebredde. Fra lign. (3.16) er det normerte strålingsdiagrammet til et lineært tynnet array med  $N$  aktive elementer på et grid med  $M$  mulige posisjoner med avstand  $d = \lambda/2$  gitt ved

$$W(u) = \frac{1}{N} \sum_{n=0}^{M-1} w_n \exp(jn\pi u) \quad (7.4)$$

hvor  $w_n$  verdiene er gitt i vektoren *array*.  $W(u)$  er hermittisk og har periode  $u = 2$  slik at kostverdien  $E$  defineres som

$$E = \max |W(u)|^2 , u \in [u_0, 1] \quad (7.5)$$

hvor  $W(u)$  er gitt i lign. (7.4). Kostverdien er relatert til kvadratet av strålingsdiagrammet siden det gir mindre spredning på kostverdiene. Kostfunksjonen (7.5) er identisk med kostfunksjonen i [Murino *et al.*, 1996]. Den relative maksimale sidelobe amplituden i dB er gitt ved  $10 \log_{10} E$ . Vinkelen  $u$  samples i  $Npts$  kontrollpunkter over intervallet  $[0, 1]$ . Figur 7.2 viser et eksempel på tettheten til punktene for  $Npts = 256$ . I denne oppgaven benyttes  $Npts = 256$  for lineære



Figur 7.2: Utsnitt av strålingsdiagrammet til et tynnet array med  $N = 25$  og  $M = 101$ . Her er  $Npts = 256$  uniformt fordelt over intervallet  $[0, 1]$ .

arrayer. Dette gir en vinkelopløsning på  $\delta\phi = 0.22^\circ$ .

I de fleste tilfellene er det mer effektivt å implementere  $W(u)$  som en FFT av vektoren array enn ved doble for-løkker definert av lign. (7.4). Ved å generere en ny FFT for hver perturbering kaster man unødig mye data siden man kun endrer ett element hver gang. Hvis  $Npts$  kan uttrykkes som en potens av 2 så er en FFT en  $O(Npts \log_2 Npts)$  prosess. Derimot hvis dette ikke er oppfylt vil en FFT være av orden  $O(Npts^2)$ . Strålingsdiagrammet til et perturbert array kan uttrykkes som en  $O(4Npts)$  prosess, siden man kun fjerner og legger til ett element hver gang. La  $\mathbf{W}$  være en vektor over de  $Npts$   $W(u)$  verdiene slik at

$$\mathbf{W}^T = \frac{1}{N} \left[ \sum_{n=0}^{M-1} w_n e^{jn\pi u_0} \quad \sum_{n=0}^{M-1} w_n e^{jn\pi u_1} \quad \dots \quad \sum_{n=0}^{M-1} w_n e^{jn\pi u_{Npts-1}} \right] \quad (7.6)$$

$\mathbf{W}$  kan uttrykkes ved produktet

$$\frac{1}{N} \begin{bmatrix} e^{j0\pi u_0} & e^{j1\pi u_0} & e^{j2\pi u_0} & \dots & e^{j(M-1)\pi u_0} \\ e^{j0\pi u_1} & e^{j1\pi u_1} & & & \vdots \\ e^{j0\pi u_2} & & & & \vdots \\ \vdots & & & & \vdots \\ e^{j0\pi u_{Npts-1}} & \dots & \dots & \dots & e^{j(M-1)\pi u_{Npts-1}} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 1 \\ 1 \end{bmatrix} = \mathbf{W} \quad (7.7)$$

slik at

$$A \cdot \text{array}^T = \mathbf{W} \quad (7.8)$$

Altså vil vektoren  $\mathbf{W}$  til et tynnet array hvor posisjonene til de aktive elementene er gitt ved *konfig* være summen over søylene i  $A$  definert ved *konfig*. La vektoren  $\mathbf{W}_{test}$  være en perturbering av  $\mathbf{W}_0$  hvor man tar ut perturberingselementet  $p\_ut$  og setter inn elementet  $p\_inn$ .  $\mathbf{W}_{test}$  er da gitt ved:

$$\mathbf{W}_{test} = \mathbf{W}_0 - \begin{bmatrix} e^{j(p\_ut)\pi u_0} \\ e^{j(p\_ut)\pi u_1} \\ \vdots \\ e^{j(p\_ut)\pi u_{Npts-1}} \end{bmatrix} + \begin{bmatrix} e^{j(p\_inn)\pi u_0} \\ e^{j(p\_inn)\pi u_1} \\ \vdots \\ e^{j(p\_inn)\pi u_{Npts-1}} \end{bmatrix} \quad (7.9)$$

Både den FFT baserte metoden og matrise metoden er implementert som Matlab-rutiner og er gitt i tillegg C.2 og C.3 Hvis man benytter  $Npts = 2^8 = 256$  så vil hver perturbering bruke 6875 flops (floating point operations) med den FFT baserte algoritmen, mens bidragsmetoden kun bruker max 1024 flops (avhenger av  $u_0$ ). Altså en faktor 6.7 i forskjell, og hvis  $Npts$  ikke kan uttrykkes som en potens av 2 øker denne faktoren til 35.9 for  $Npts = 300$ . Noen av de kommeende simuleringene gjør flere millioner perturberinger og bruker flere dager, slik at bidragsmetoden blir svært nytting.

## 7.2 Resultater

I dette kapittelet presenteres resultatet av minimering av max sidelobenivået for utvalgte tynnede lineære arrayer. De fleste simuleringene er gjort for et tynnet array med  $M = 101$  mulige posisjoner og  $N = 25$  aktive elementer siden det er et av de vanligste referanseproblemene.

### 7.2.1 Et lite testproblem

Lokale søk algoritmer slik som Simulated Annealing har størst anvendelse på svært store problemer hvor eksakte metoder krever uendelig lang simuleringstid.

Små problemer er likevel interessante siden man kan sammenligne algoritmens løsninger med de optimale løsningene.

Tynningsproblemet med  $M = 25$  posisjoner i aperturen og  $N = 7$  aktive elementer har vært benyttet flere ganger i denne oppgaven. Det finnes som tidligere nevnt 33649 mulige tynningsmønstre for dette arrayet. Temperatur- og perturberingsparametre for simuleringene er gitt i tabell 7.1. Den initiale temperaturen  $T_0$

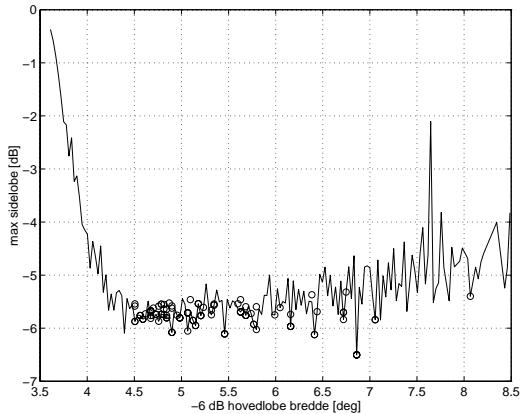
	Fig. 7.3 Mange pert.	Fig. 7.4 Middels antall pert.	Fig. 7.4 1 pert.
$T_0$	0.08	0.08	0.08
Temperatur-dekrementering	$T_i = T_0/i$	$T_i = T_0/i$	$T_i = T_0/i$
$num\_iter$	10	8	8
$num\_pert$	20	5	1
$u_0$	0.10	0.10	0.10

Tabell 7.1: Simuleringsparametre for testproblemet med  $M = 25$  og  $N = 7$ .

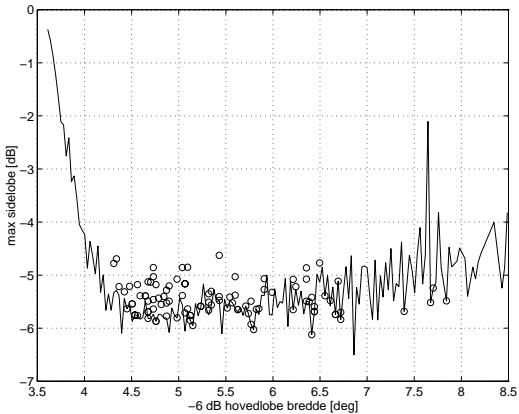
ble bestemt i henhold til ligning (6.9) hvor  $T_0$  er den minste anbefalte verdien. Dette tilsvarer en initiell akseptanse på 4/10. Den inversproporsjonale temperaturfunksjonen gitt i lign. (6.18) ble valgt uten teoretisk forankring. Fra lign. (3.18) er 1. nullpunkt til det fulle arrayet gitt ved  $u^0 = 2/M = 0.08$ . Optimeringsvinkelen  $u_0$  ble definert litt større enn  $u^0$ .

Produktet  $num\_iter \cdot num\_pert$  gir totalt antall perturberinger algoritmen genererer, med andre ord hvor mange nabokonfigurasjoner algoritmen undersøker. Merk at det er ingen garanti for at samme konfigurasjon ikke blir generert flere ganger, men sannsynligheten for dette er relativt liten siden tilskuddet av nye konfigurasjoner pr perturbing er 77%. For hver parameter-kombinasjon gitt i tabell 7.1 ble det gjort 100 simuleringer med ulik initiell konfigurasjon. Max sidelobenivå og  $-6$  dB hovedlobebredde for disse resultatene relativt til kurven gjennom de globale minimumsverdiene er plottet i figur 7.3, 7.4 og 7.5.

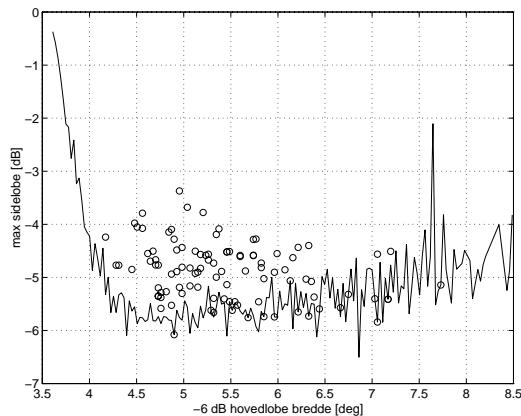
Felles for alle figurene er den store spredningen i hovedlobe breddde. Dette skyldes at  $-6$  dB hovedlobebredden ofte inkluderer 1. sidelobe pga. få elementer i arrayet. For arrayene i figur 7.3 er produktet av antall iterasjoner og antall perturberinger lik 200 (0.6% av totalt antall arrayer). Nesten samtlige av disse løsningene ligger på den optimale kurven. Arrayene i figur 7.4 ble funnet ved å undersøke 40 av nabokonfigurasjonene (0.1% av totalt antall arrayer). Her er ca. halvparten av resultatene optimale, mens resten ligger rett over den optimale kurven. I det siste tilfellet gjør algoritmen kun 1 perturbing for hver av de 8 iterasjonene (0.02% av totalt antall arrayer). Her ligger kun 18 av løsningene på den optimale kurven slik figur 7.5 viser. Resten av løsningene ligger spredt og strekker



Figur 7.3: Max sidelobe og  $-6$  dB hovedlobebredde for 100 simuleringer hvor algoritmen undersøker 200 nabokonfigurasjoner relativt til den optimale kurven for  $M = 25$  og  $N = 7$ .



Figur 7.4: Max sidelobe og  $-6$  dB hovedlobebredde for 100 simuleringer hvor algoritmen undersøker 40 nabokonfigurasjoner relativt til den optimale kurven for  $M = 25$  og  $N = 7$ .



Figur 7.5: Max sidelobe og  $-6$  dB hovedlobebredde for 100 simuleringer hvor algoritmen undersøker 8 nabokonfigurasjoner relativt til den optimale kurven for  $M = 25$  og  $N = 7$ .

seg opp mot  $-3.3$  dB.

Dette testproblemet illustrerer at algoritmen finner det globale minimum for et tilstrekkelig antall perturberinger. Parameteranlysen kan ikke generaliseres til større problemer pga. ulik frihetsgrad, men med bedre tilpassede parametre burde man forvente gode løsninger for større problemer også.

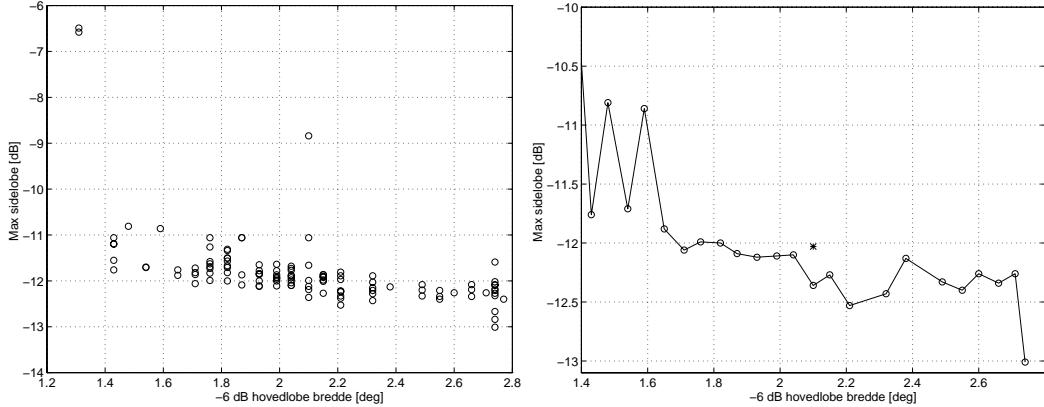
### 7.2.2 Et referanseproblem

Tynningsproblemet med aperture  $D = 50\lambda$  og  $N = 25$  aktive elementer er en gjenganger i flere artikler. I [Murino *et al.*, 1996] er de ulike metodene og deres resultater listet opp. Den første metoden stammer fra 60-tallet og benyttet en dynamisk programmering algoritme og hadde betingelser om symmetri. I flg. [Murino *et al.*, 1996] gav dette et max sidelobenivå på  $-8.8$  dB. Den beste metoden frem til i dag er gitt i [Murino *et al.*, 1996] og benyttet Simulated Annealing. Dette gav et max sidelobenivå på  $-12.07$  dB.

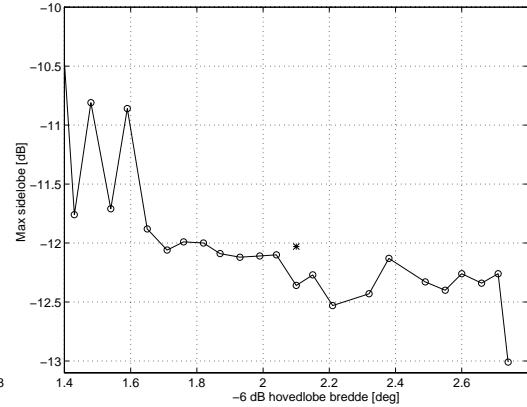
I denne oppgaven er det antatt at elementene i det tynnede arrayet har utstrekning  $\lambda/2$ , slik at strålingsdiagrammet blir beregnet over senteret til hvert element og arrayets aperture blir  $D = M \cdot \lambda/2$ . Det er den samme antagelsen som er benyttet i Ultrasim<sup>1</sup>. I [Murino *et al.*, 1996] derimot har man antatt punktelermer slik at for å implementere deres optimale løsning; 0 8.5 13 16 20 22 22.5 24 27 27.5 30.5 32 33 33.5 34.5 35 36 37 38.5 39 39.5 40.5 48.5 50 (posisjon i  $\lambda$ ) må man definere  $M = 101$  slik at  $D = 50.5\lambda$ . For å bestemme  $-6$  dB hovedlobebredde og max sidelobenivå med stor nøyaktighet benyttes en 4096-punkts FFT. Resultatet til [Murino *et al.*, 1996] gir da  $-12.04$  dB som max sidelobe og  $2.10^\circ$  hovedlobebredde. Av totalt 121 simuleringer med ulike parametre gav 10 av disse lavere max sidelobe for hovedlobebredder mellom  $1.71^\circ$  og  $2.10^\circ$ , hvor minste max sidelobe er  $-12.36$  dB. Max sidelobenivå og  $-6$  dB hovedlobebredde for de 121 simuleringene er gitt i figur 7.6 I figur 7.7 er det trukket en kurve gjennom minimumsverdiene for hver hovedlobebredde, hvor punktet markert med \* er den optimale løsningen i [Murino *et al.*, 1996]. Strålingsdiagrammet for de 10 arrayene med bedre max sidelobenivå enn [Murino *et al.*, 1996] er gitt i figur 7.8–7.17. De tekniske dataene til hvert array er gitt i tabell 7.2 og parameterverdiene til simuleringene er gitt i tabell 7.3.

---

<sup>1</sup>MATLAB toolbox for simulering av ultralyd bølger. Tilgjengelig i [Ult, ]



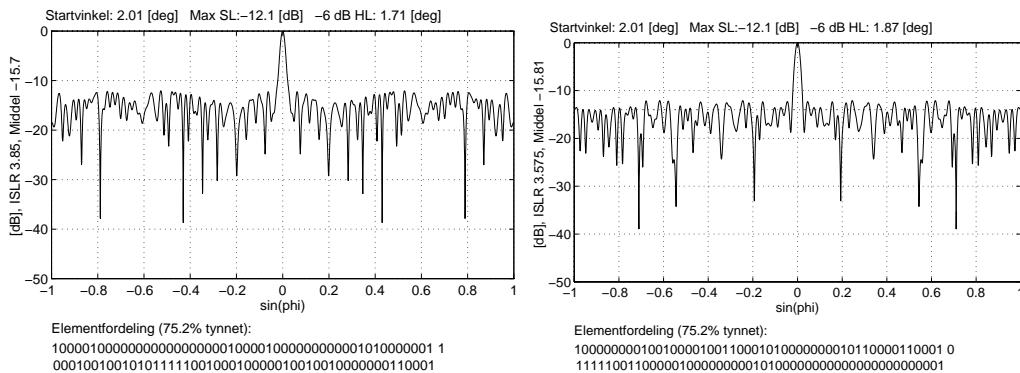
Figur 7.6: Max sidelobe og  $-6$  dB hovedløbubredder for 121 simuleringer for tynningsproblemet med  $N = 25$  aktive elementer og  $M = 101$  mulige elementposisjoner.



Figur 7.7: Minimumsverdiene for max sidelobe for hver av de observerte hovedløbubreddene. Punktet markert med \* er den optimale løsningen til [Murino *et al.*, 1996].

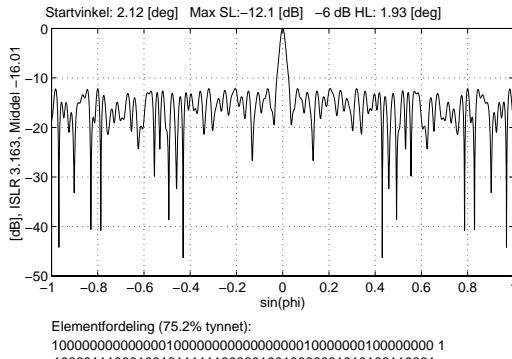
### De 10 beste løsningene

Figur 7.8–7.17 viser de 10 beste løsningene på tynningsproblemet med  $M = 101$  og  $N = 25$ .

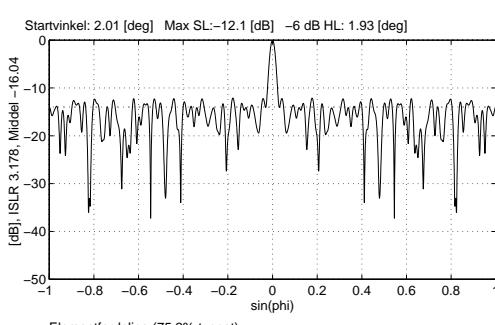


Figur 7.8: Resultatet er oppnådd ved å undersøke  $2.0 \cdot 10^5$  nabokonfigurasjoner.

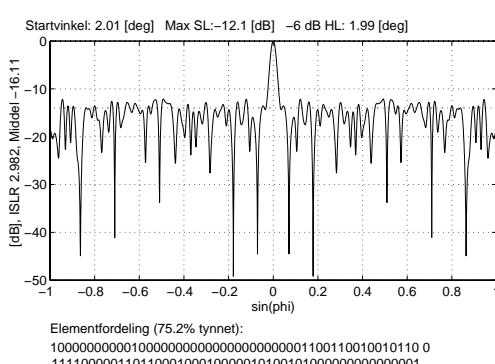
Figur 7.9: Resultatet er oppnådd ved å undersøke  $2.0 \cdot 10^7$  nabokonfigurasjoner.



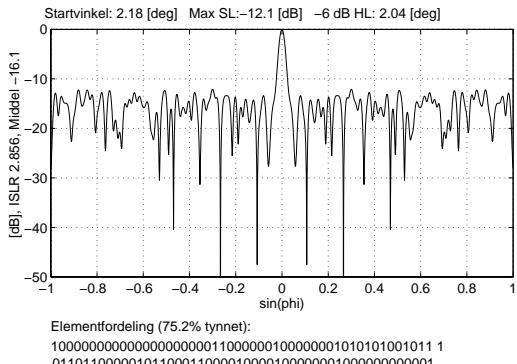
Figur 7.10: Resultatet er oppnådd ved å undersøke  $1.8 \cdot 10^5$  nabokonfigurasjoner.



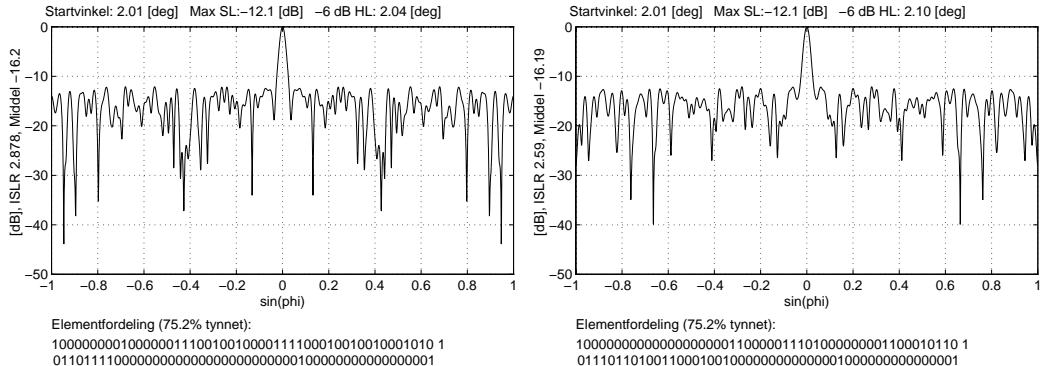
Figur 7.11: Resultatet er oppnådd ved å undersøke  $2.0 \cdot 10^7$  nabokonfigurasjoner.



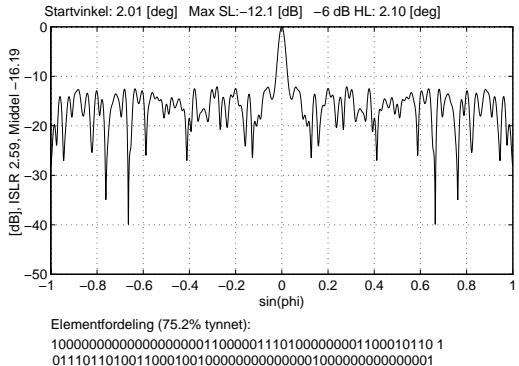
Figur 7.12: Resultatet er oppnådd ved å undersøke  $2.0 \cdot 10^7$  nabokonfigurasjoner.



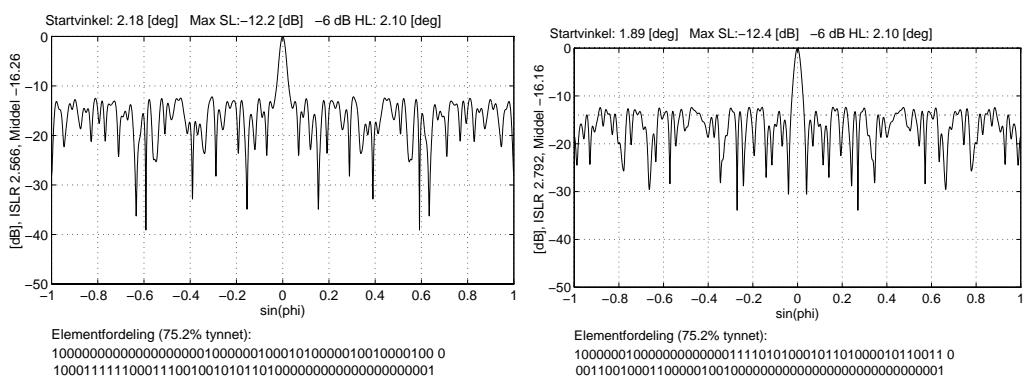
Figur 7.13: Resultatet er oppnådd ved å undersøke  $2.0 \cdot 10^5$  nabokonfigurasjoner.



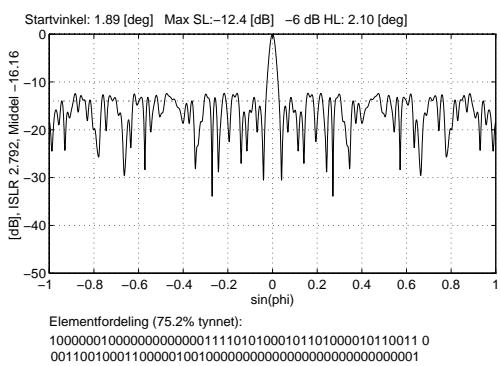
Figur 7.14: Resultatet er oppnådd ved å undersøke  $4.0 \cdot 10^5$  nabokonfigurasjoner.



Figur 7.15: Resultatet er oppnådd ved å undersøke  $4.0 \cdot 10^5$  nabokonfigurasjoner.



Figur 7.16: Resultatet er oppnådd ved å undersøke  $1.5 \cdot 10^6$  nabokonfigurasjoner.



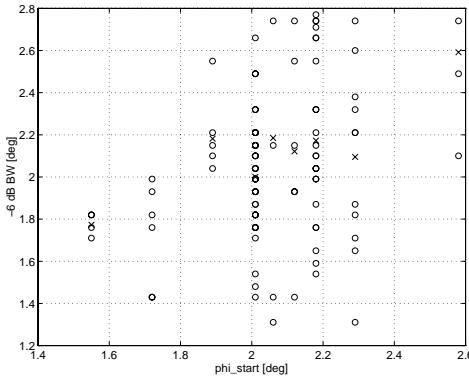
Figur 7.17: Den beste løsningen på tynningsproblemet med  $M = 101$  og  $N = 25$ . Resultatet er oppnådd ved å undersøke  $2.9 \cdot 10^7$  nabokonfigurasjoner.

	–6 dB HL [deg]	Max SL [dB]	Middel SL [dB]	Max SL – Middel(Max SL) [dB]
Fig. 7.8	1.71	–12.06	–15.70	1.62
Fig. 7.9	1.87	–12.09	–15.81	1.84
Fig. 7.10	1.93	–12.12	–16.01	1.65
Fig. 7.11	1.93	–12.11	–16.04	1.93
Fig. 7.12	1.99	–12.11	–16.12	1.70
Fig. 7.13	2.04	–12.10	–16.10	1.88
Fig. 7.14	2.04	–12.10	–16.20	1.91
Fig. 7.15	2.10	–12.12	–16.19	2.10
Fig. 7.16	2.10	–12.19	–16.26	1.88
Fig. 7.17	2.10	–12.36	–16.16	1.73

Tabell 7.2: –6 dB hovedlobebredde (HL), max sidelobe (SL), sidelobe verdiene i middel og avviket fra en Dolph-Chebyshev sidelobe fordeling for de 10 arrayene med lavest max sidelobe for  $N = 25$  og  $M = 101$ .

	$T_0$	$T_{num\_iter}$	Temp. funk.	# iter.	# pert.	$u_0$
Fig. 7.8	0.1	$2.5 \cdot 10^{-4}$	$T_i = \frac{T_0}{i}$	400	500	0.035
Fig. 7.9	0.05	$2.5 \cdot 10^{-4}$	$T_i = \frac{T_0}{0.1i}$	2000	10000	0.035
Fig. 7.10	0.1	$3.3 \cdot 10^{-4}$	$T_i = \frac{T_0}{i}$	300	600	0.037
Fig. 7.11	0.05	$2.5 \cdot 10^{-4}$	$T_i = \frac{T_0}{0.1i}$	2000	10000	0.035
Fig. 7.12	0.05	$2.5 \cdot 10^{-4}$	$T_i = \frac{T_0}{0.1i}$	2000	10000	0.035
Fig. 7.13	0.05	$2.5 \cdot 10^{-4}$	$T_i = \frac{T_0}{i}$	200	1000	0.038
Fig. 7.14	0.05	$2.5 \cdot 10^{-4}$	$T_i = \frac{T_0}{0.5i}$	400	1000	0.035
Fig. 7.15	0.05	$2.5 \cdot 10^{-4}$	$T_i = \frac{T_0}{0.5i}$	400	1000	0.035
Fig. 7.16	0.05	$1.0 \cdot 10^{-4}$	$T_{i+1} = T_i \exp\left(\frac{0.05T_i}{\sigma(T_i)}\right)$	381	4000	0.038
Fig. 7.17	0.05	$2.5 \cdot 10^{-4}$	$T_i = \frac{T_0}{0.7i}$	285	100000	0.033

Tabell 7.3: Parameterverdiene til simuleringene som gav de 10 laveste max sidelobenivåene for  $N = 25$  og  $M = 101$ .



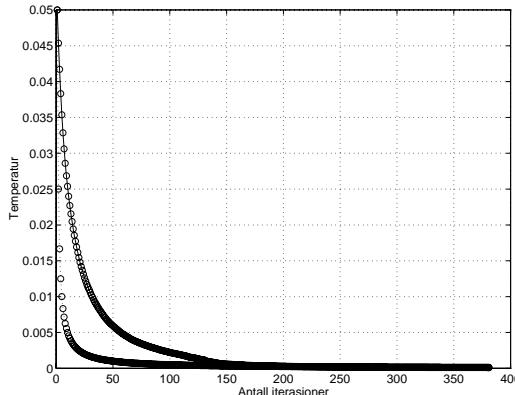
Figur 7.18:  $-6$  dB hovedlobebredde for ulik startvinkel  $\phi_0 = \sin^{-1} u_0$  for simuleringene for  $N = 25$  og  $M = 101$ . Punktene merket med X er middelverdien for hver  $\phi_0$

### Startvinkel

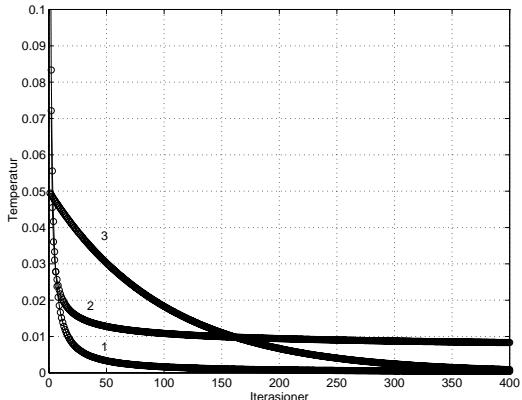
Startvinkelen  $u_0$  angir området hvor algoritmen leter etter den maksimale sidelobe verdien. Hvis  $u_0$  blir valgt for liten, så vil hovedloben bli inkludert i søker-området. Derimot, hvis  $u_0$  blir satt for stor, vil 1. sidelobe bli ekskludert fra søker-området. Parameteren  $u_0$  gir kun begrenset kontroll over hovedlobebredden. Dette kommer tydelig frem i figur 7.18 som viser  $-6$  dB hovedlobebredde for alle simuleringene for ulik startvinkel  $\phi_0$ .

### Temperaturparametere

Kun resultatet gitt i figur 7.16 er optimert med tilstandsbestemt temperaturfunksjon (lign. (6.22)). Simuleringer med ulik initialkonfigurasjon gav tilnærmet det samme tilstandbestemte temperaturforløpet. Derfor ble temperaturforløpet bestemt a posteriori for de andre simuleringene. Figur 7.19 viser det tilstandsbestemte forløpet til resultatet med lavest max sidelobe (figur 7.16) med  $\gamma = 0.05$  sammenlignet med den invers proporsjonale temperaturfunksjonen gitt i lign. (6.18) med  $\beta = 1$ . Den tilstandsbestemte funksjonen har en mye mindre dekrementering for høye temperaturer enn den invers proporsjonale funksjonen. Altså for samme antall iterasjoner vil den invers proporsjonale funksjonen tilbringe lengre tid ved lave temperaturer. Figur 7.20 sammenligner 3 deterministiske temperaturfunksjoner . Funksjonen markert med '2' gav det beste resultatet i [Murino *et al.*, 1996], mens funksjonen merket med '1' er den beste tilnærmingen til den tilstandsbestemte funksjonen i dette tilfellet.



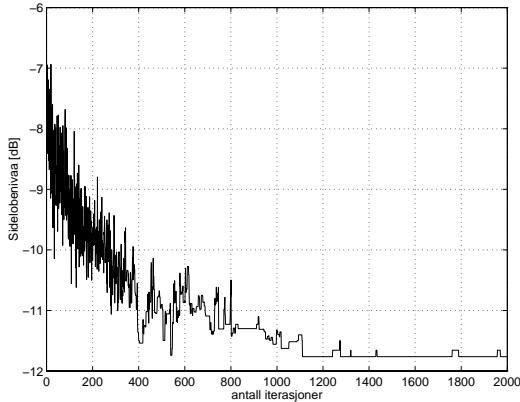
Figur 7.19: Øvre kurve: Tilstandsbestemt temperaturkurve gitt av lign. (6.22) med  $\gamma = 0.05$  for en simulering for  $N = 25$ ,  $M = 101$  og 4000 perturberinger pr temperaturverdi. Nedre kurve: Deterministisk temperaturkurve gitt av lign. (6.18) med  $\beta = 1$ .



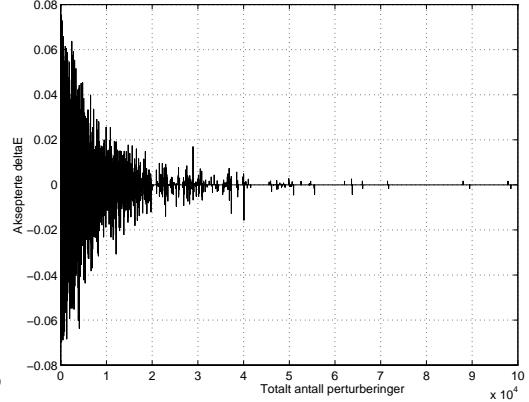
Figur 7.20: Sammenligning av 3 deterministiske temperaturfunksjoner hvor  $T_0 = 0.05$ .  
 Kurve 1:  $T_i = \frac{T_0}{\beta i}$ ,  $\beta = 0.3$   
 Kurve 2:  $T_i = \frac{T_0}{\log i}$   
 Kurve 3:  $T_i = \alpha^i T_0$ ,  $\alpha = 0.99$

## Antall perturberinger

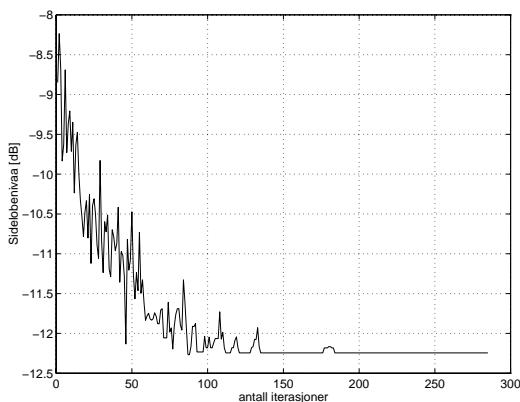
De fleste simuleringene ble gjort med den invers proporsjonale temperaturfunksjonen (lign. (6.18)) med skaleringsfaktor  $\beta$  mellom 0.1 og 1.5. Denne funksjonen reduserer temperaturen betydelig raskere enn den logaritmiske (lign. (6.15)), slik algoritmen må gjøre betydelig flere perturberinger pr iterasjon for å beholde likevektstilstanden. Figur 7.21 viser sidelobenivået for hver iterasjon for en simulering med mange iterasjoner og få perturberinger. Figur 7.22 viser de aksepterte kostendringene ved hver perturbing. Resultatet fra de 121 simuleringene viser at det motsatte er en bedre strategi. Ved å gjøre mange perturberinger og få iterasjoner undersøker algoritmen flere konfigurasjoner omkring frysepunktet. Resultatet i figur 7.23 er et eksempel på dette. For denne simuleringen ble det gjort 100000 perturberinger for hver av de 285 iterasjonene, og aksepraten er gitt i figur 7.24. Algoritmen undersøker altså 28.5 millioner konfigurasjoner, men det er likevel kun  $10^{-15}$  del av totalt antall konfigurasjoner. Simuleringen ble gjort med gjort med bidragsmetoden og brukte 67t på en SGI Power Challenge med 1 av 4 CPUer aktive. Dette resultatet har 0.32 dB lavere max sidelobe enn resultatet i [Murino *et al.*, 1996] for samme hovedlobebredde.



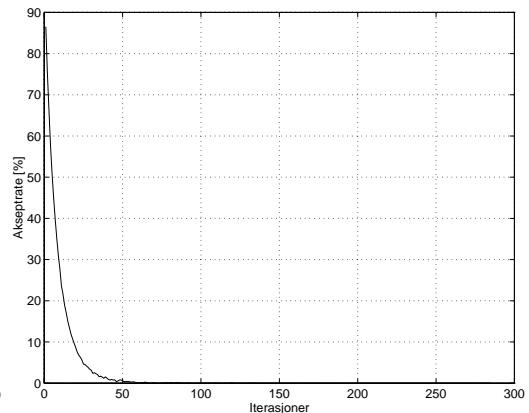
Figur 7.21: Sidelobenivået ved hver iterasjon for en simulering hvor  $num\_iter = 2000$ ,  $num\_pert = 50$  og temperaturen faller invers proporsjonalt med antall iterasjoner for  $N = 25$  og  $M = 101$ .



Figur 7.22: Aksepterte kostendringer ved hver perturbing for simuleringen i figur 7.21.



Figur 7.23: Sidelobenivået ved hver iterasjon for en simulering hvor  $num\_iter = 285$ ,  $num\_pert = 100000$  og temperaturen faller invers proporsjonalt med antall iterasjoner for  $N = 25$  og  $M = 101$ .



Figur 7.24: Raten av aksepterte kostendringer ved hver iterasjon for simuleringen i figur 7.23.

## 7.3 Konklusjon

Med utgangspunkt i algoritmen gitt i [Murino *et al.*, 1996], men med andre parametre, er det funnet 10 ulike arrayer med lavere max sidelobenivå uten å øke hovedlobebredden. Disse arrayene har 25 aktive elementer på et lineært grid med 101 posisjoner med avstand  $\lambda/2$ . Ved på la perturberingselementet plasseres fritt blant alle ledige posisjoner vil antall nabokonfigurasjoner være større en ved definisjonen i [Murino *et al.*, 1996]. Dette medfører at algoritmen får større frihetsgrad og er mindre avhengig av initiell konfigurasjon. Hvis antall perturberinger overstiger antall temperatur-dekrementeringer, vil algoritmen tilbringe lengre tid ved størningspunktet slik at sannsynligheten for å oppdage optimale konfigurasjoner øker.

Sannsynligheten for å oppdage optimale konfigurasjoner øker med produktet av antall iterasjoner og antall perturberinger. For et array med  $N = 7$  elementer i en aperture med  $M = 25$  posisjoner er det stor sannsynlighet for at alle løsningene være globalt optimale hvis algoritmen undersøker 0.6% av alle konfigurasjonene. For tynningsproblemet med  $N = 25$  og  $M = 101$  ble en av de 10 arrayene med lavest max sidelobe funnet ved å undersøke 200000 konfigurasjoner, noe som kun krever omkring 20 min på en SGI Power Challenge, mens for å finne arrayet med  $-12.36$  som max sidelobe undersøkte algoritmen 28.5 millioner konfigurasjoner. Det er lite trolig at dette er en globalt optimal løsning siden max sidelobe verdien avviker med 1.73 dB fra gjennomsnittet av alle toppene. Dette problemet har så stor frihetsgrad at man kan anta at det finnes arrayer med sidelobe amplitude nærmere en Dolph-Chebyshev fordeling. Dette arrayet har  $0.73^\circ$  bredere hovedlobe enn det fulle arrayet. Fra figur 5.10 og 5.11 så man at arrayer med bredere hovedlobe enn det fulle arrayet ( $5.45^\circ$  i figur 5.10 og 5.11) ikke tilhører klassen med minimum antall hull og redundanser, hvor sannsynligheten for optimalitet er relativt stor. Fra sorteringene i kapittel 5 fant man at de optimale arrayene hadde et avvik på 0.62dB–1.60dB fra en Dolph-Chebyshev fordeling. Derfor er det likevel rimelig å anta at arrayet med  $-12.36$  dB som max sidelobe amplitude er nær det globale minimum.

# Kapittel 8

## Optimering av 2D arrayer med Simulated Annealing

God Grant me the serenity  
To accept the problems that I cannot solve  
The persistence to solve the problems that I can  
And the wisdom to know the difference.

*T. C. Hu*

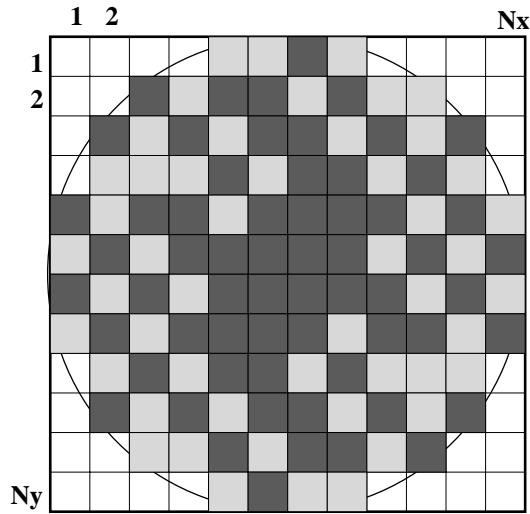
Et kvadratisk array med  $50 \times 50$  elementer i aperturen tynnet til 500 aktive elementer har  $4 \cdot 10^{542}$  mulig tynningsmønstre. Til sammenligning er det ca  $10^{80}$  atomer i universet. Optimering av 2D arrayer er altså en svært vanskelig oppgave.

### 8.1 Implementering

Algoritmen for 2D arrayer skiller seg ikke vesentlig fra 1D algoritmen. De viktigste forandringene er å beskrive konfigurasjonene i to dimensjoner, og definere en kostfunksjon deretter.

#### 8.1.1 Beskrive konfigurasjonene

For å kunne behandle ulike transducer geometrier på en effektiv måte antar man et underliggende kvadratisk grid. I denne oppgaven vil dette gridet ha avstand  $\lambda/2$  både i horisontal og vertikal retning. La  $N_x$  betegne antall posisjoner i horisontal retning og  $N_y$  antall posisjoner i vertikal retning. Ved å nummerere alle gridposisjonene for hver rad kan man definere en vektor *tillatt* for de mulige elementposisjonene. For en sirkulær aperture definert på et  $12 \times 12$  grid slik som i figur 8.1 får man:



Figur 8.1: Tillatte elementer i en sirkulær aperture hvor senteret til hvert element ligger innenfor sirkelen. De grå feltene markerer de tillatte posisjonene og de mørke feltene de aktive elementene.

$$tillatt = [ \ 5 \ 6 \ 7 \ 8 \ 15 \dots 22 \ 26 \dots 35 \ \dots ]$$

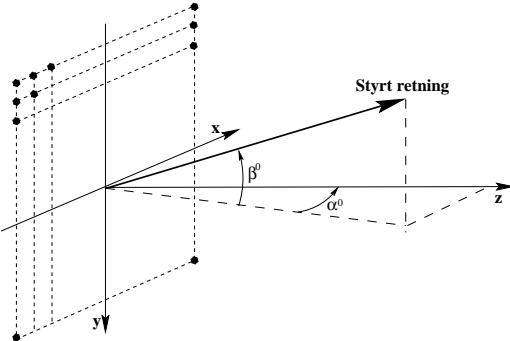
De aktive elementene i arrayet kan betegnes med 1-ere i et binært array. For arrayet i figur 8.1 får man:

$$array = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Man kan definere en vektor *konfig* over de aktive elementene og *ledig* over de komplimentære elementene i *tillatt*. Med denne implementeringen kan man betrakte 2D-arrayet som et generalisert 1D tilfelle. Altså kan man definere stokastiske endringer på samme måte som i kapittel 7.1.2.

### 8.1.2 Kostfunksjonen

La vinklene  $\alpha$  og  $\beta$  betegne azimuth vinkel i hhv. x og y retning, slik figur 8.2 viser. Dette er den samme vinkeldefinisjonen som i [Trucco, 1997]. Fra ligning



Figur 8.2: Arraygeometrien og styringsvinklene.

(3.10) er det normerte strålingsdiagrammet til et array med elementavstand  $\lambda/2$  gitt ved

$$W(u, v) = \frac{1}{M} \sum_{n=1}^M w_n e^{j\pi d(n_x u + n_y v)} \quad (8.1)$$

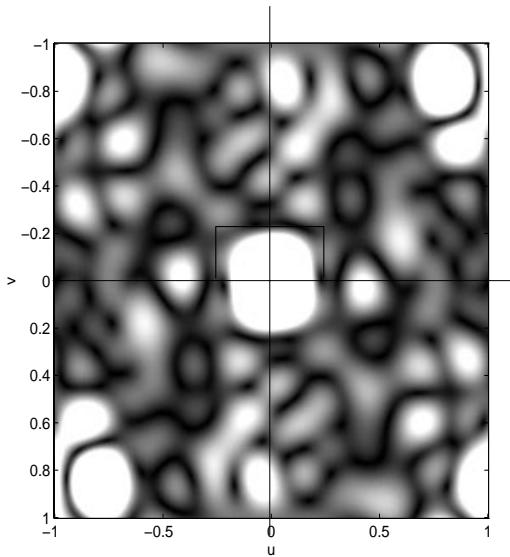
hvor  $u = \sin \alpha - \sin \alpha^0$ ,  $v = \sin \beta - \sin \beta^0$  og  $n = (n_y - 1)N_x + n_x$ .

$W(u, v)$  er hermittisk slik figur 8.3 illustrerer. Kostverdien  $E$  for 2D arrayer kan derfor defineres som

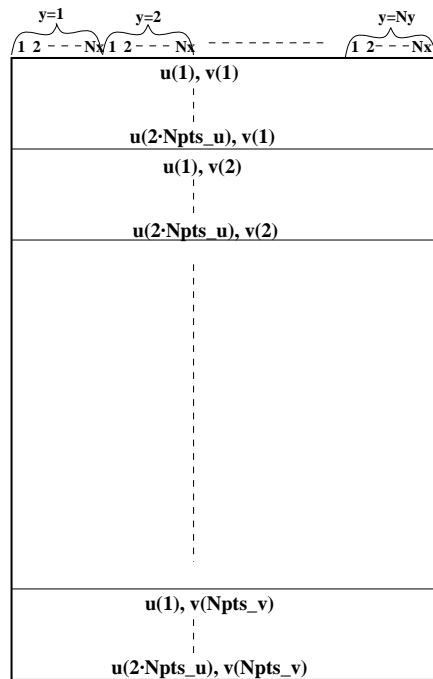
$$E = \max |W(u, v)|^2 \quad , v \in [0, 1], |u| \in \begin{cases} [0, 1] & v \geq v_0 \\ [u_0, 1] & v < v_0 \end{cases} \quad (8.2)$$

hvor hovedloben er tilnærmet med et rektangel slik figur 8.3 viser. La  $Npts\_u$  og  $Npts\_v$  betegne antall kontrollpunkter i hhv.  $u$  og  $v$  retning på intervallet  $[0, 1]$ . Dimensjonen til matrisen  $W$  blir da  $Npts\_v \times 2Npts\_u$ .  $W$  kan beregnes ved en 2D-FFT av den binære matrisen *array*. En 2D-FFT tilsvarer å gjøre en FFT på hver rad i matrisen *array* og deretter en FFT på hver kolonne i den resulterende matrisen.

Istedentfor å beregne en 2D-FFT for hver perturbering kan man beregne bidraget til hvert element slik som i kapittel 7. Anta vektoren  $\mathbf{W}$  som består av blokker hvor  $u$  er konstant. Tilsvarende defineres vektoren  $\mathbf{x}$  av radene i *array*. Matrisen  $A$  over bidragene til alle elementene i  $\mathbf{x}$  og over alle verdiene  $u$  og  $v$  får dimensjonen  $2Npts\_u Npts\_v \times N_x N_y$ . Hver søyle i  $A$  inneholder bidraget fra et element over alle  $u$  og  $v$  slik figur 8.4 viser. Vektoren  $\mathbf{W}$  kan nå uttrykkes ved



Figur 8.3: Intensitetsplott av strålingsdiagrammet i  $u, v$ -planet for arrayet gitt i figur 8.1.



Figur 8.4: Hvert element i *array* tilsvarer en søyle i matrisen  $A$ , som består av blokker hvor  $v$  er konstant og  $u$  varierer.

$$A\mathbf{x} = \mathbf{W} \quad (8.3)$$

slik at en perturbing av arrayet kan beregnes ved addisjon og subtraksjon av en øyle slik som i lign. (7.9). For å optimere 2D arrayer trenger man kun å modifisere 1D algoritmen ved å inkludere vektoren *tillatt* og bygge opp en ny *A* matrise. En slik Matlab-rutine er gitt i tillegg C.4.

## 8.2 Et lite array

I dette kapittelet presenteres optimeringsresultater for et  $12 \times 12$  array, som blir ansett som et lite array i medisinsk ultralyd sammenheng. Anta et array med 112 mulig elementposisjoner på et  $12 \times 12$  grid med elementavstand  $\lambda/2$  hvor elementposisjonen ligger innenfor en sirkel slik figur 8.1 viste. Det fulle sirkulære arrayet har  $13.66^\circ$  hovedlobebredde.

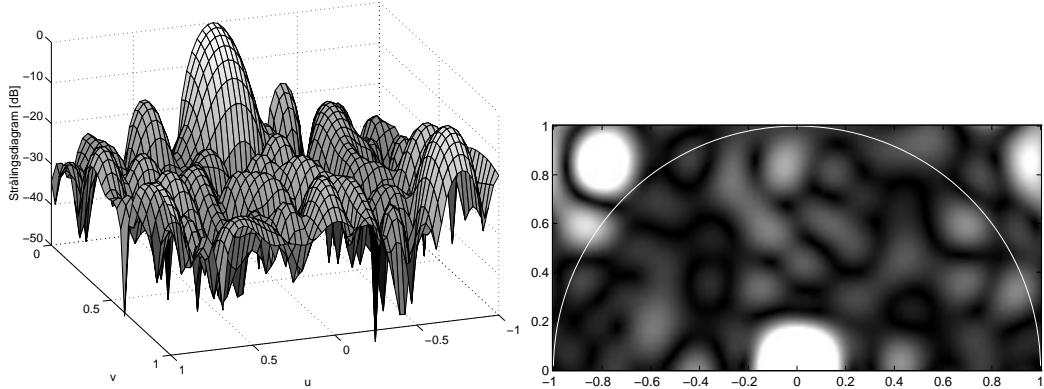
### 8.2.1 Tidligere resultater

Ved heltalls lineær programmering fant man i [Holm *et al.*, 1997, Elgtun, 1996] optimale løsninger for symmetriske arrayer. De optimale arrayene har max sidelobenivå mellom  $-20$  dB og  $-26$  dB for  $-6$  dB hovedlobebredde mellom  $14^\circ$  og  $19^\circ$ . Disse arrayene får andre egenskaper hvis de styres, siden optimeringsområdet kun dekker det synlige området i  $u, v$ -planet. Figur 8.5 viser strålingsdiagrammet til det optimale arrayet i [Holm *et al.*, 1997] evaluert over hele  $u, v$ -planet. Elementfordelingen til de 62 aktive elementene er tidligere vist i figur 8.1. Dette arrayet har  $17.2^\circ$  hovedlobebredde og  $-24.5$  dB som max sidelobe i det optimerete området. Figur 8.6 viser et intensitetsplott av strålingsdiagrammet og angir det optimerete området. Utenfor det optimerete området skjuler det seg en sidelobe med relativ amplitude  $-6.18$  dB. De gode verdiene i [Holm *et al.*, 1997] kan forklares ved at algoritmen har redusert sidelobene i det optimerete området og *tillatt* desto større sidelober utenfor det synlige området.

I [Trucco, 1997] optimerer man det samme arrayet med Simulated Annealing og tillater elementvektning og asymmetriske løsninger. Kostfunksjonen i [Trucco, 1997] skiller seg fra kostfunksjonene i [Murino *et al.*, 1996] ved at den setter a priori betingelser på strålingsdiagrammet og samtidig minimerer antall aktive elementer. Denne kostfunksjonen er gitt ved

$$E \equiv k_1 [W(u, v) - W_d(u, v)] + k_2 N \quad (8.4)$$

hvor  $W_d$  er det ønskede strålingsdiagrammet til de  $N$  vektede elementene og  $k_1, k_2$  er konstanter. Ved å definere  $W_d$  lik et konstant nivå på  $-24.5$  dB for sidelobene i hele  $u, v$ -planet, fant [Trucco, 1997] et vektet array med  $-24.5$  dB max sidelobe



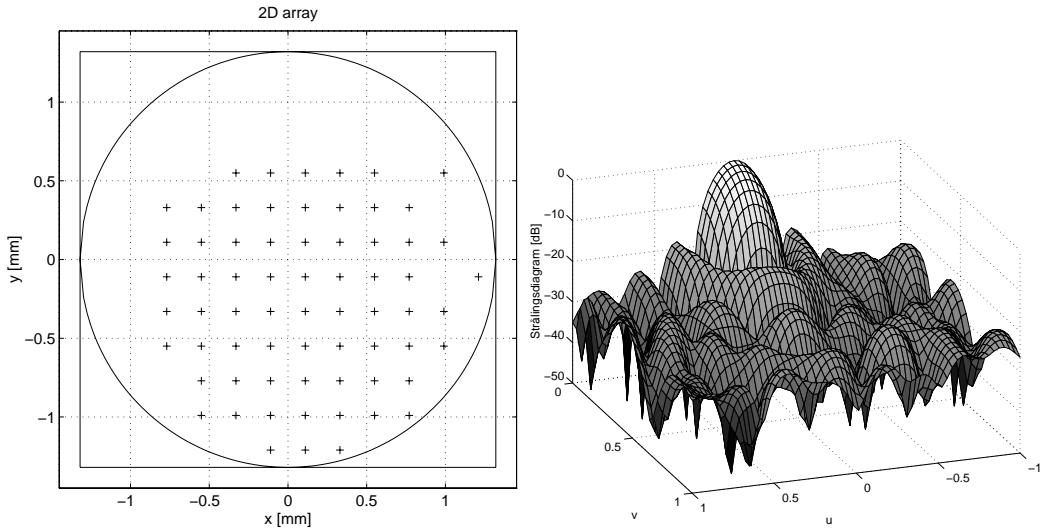
Figur 8.5: Strålingsdiagrammet til det optimale arrayet i [Holm *et al.*, 1997]. Hovedlobebredden er  $17.2^\circ$  og max sidelobe er  $-24.5$  dB i det optimerte området og  $-6.18$  dB i hele  $u, v$ -planet.

Figur 8.6: Intensitetsplott av strålingsdiagrammet i figur 8.5, hvor området innenfor sirkelen er det synlige området hvor strålingsdiagrammet er optimert.

og  $18.3^\circ$  hovedlobebredde. Posisjonen til de 67 aktive elementene er gitt i figur 8.7 og danner et fullt array med redusert størrelse. Strålingsdiagrammet til arrayet ved *uniform* vektning er gitt i figur 8.8. Max sidelobe er i dette tilfellet  $-16.33$  dB og hovedlobebredden  $17.8^\circ$ . Altså har dette arrayet max sidelobe verdi mellom verdiene til 1. sidelobe for et fullt sirkulært og rektangulært array. Til sammenligning er 1. sidelobe til et fullt sirkulært array  $-17$  dB og  $-13$  dB for et rektangulært array. I dette tilfellet er det altså vektingen som hovedsakelig bidrar til det lave sidelobenivået.

### 8.2.2 Nye resultater

I disse simuleringene krever man a priori  $N = 64$  aktive elementer. Dette gir totalt  $\binom{112}{64} \approx 1 \cdot 10^{32}$  mulige tynningsmønstre, som er en faktor  $10^9$  større enn 1D problemet i kapittel 7.2.2. Det ble gjort 22 simuleringer for dette problemet. Hovedlobebredden og max sidelobenivå til disse løsningene relativt til de refererte resultatene er gitt i figur 8.9. Et av de beste resultatene har max sidelobe  $-19.38$  dB og  $18.9^\circ$  hovedlobebredde. Strålingsdiagrammet til dette arrayet er gitt i figur 8.10 og 8.11. Et intensitetsplott av  $u, v$ -planet er gitt i figur 8.12. Posisjonen til de aktive elementene er gitt i figur 8.13. Algoritmen undersøkte kun 400 000 konfigurasjoner for å finne denne løsningen. De andre parametriene er:  $T_i = T_0/i$ ,  $T = 0.005$ ,  $num\_iter = 100$ ,  $num\_pert = 4000$ ,  $u_0 = v_0 = 0.25$  og  $Npts\_u = Npts\_v = 64$ . Kostforløpet og aksepraten til simuleringen er gitt i

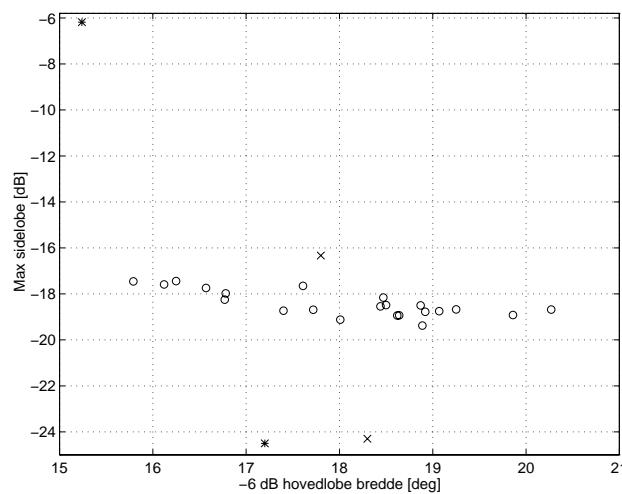


Figur 8.7: Elementposisjonene til de vektede elementene i [Trucco, 1997].

Figur 8.8: Strålingsdiagrammet til det optimale arrayet i [Trucco, 1997] ved uniform vektning. Hovedlobebredden er  $17.8^\circ$  og max sidelobe er  $-16.33$  dB.

figur 8.14.

Den mest regnekrevende simuleringen undersøkte to millioner konfigurasjoner uten at dette gav noe lavere sidelobenivå. Dette kan tyde på at løsningene i figur 8.9 er nær optimale. Disse resultatene viser at Simulated Annealing finner elementkonfigurasjoner med lavt max sidelobenivå for relativt små 2D arrayer uten å benytte elementvekting. Tidligere metoder har funnet arrayer med max sidelobe mindre enn  $-22$  dB, men disse arrayene kan enten ikke styres eller har vektede elementer.

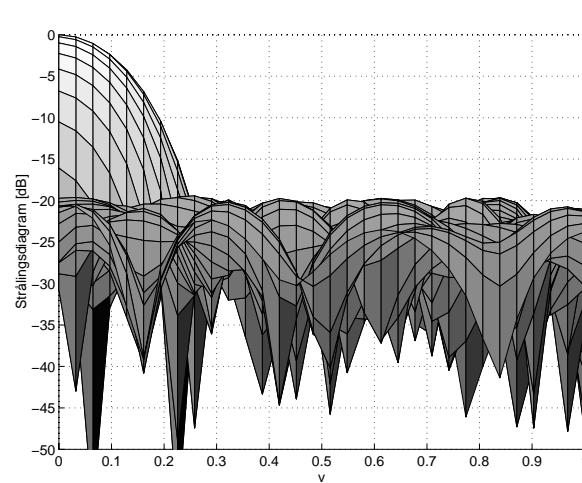


Figur 8.9: Max sidelobenivå og hovedlobebredde til de ulike simuleringene.

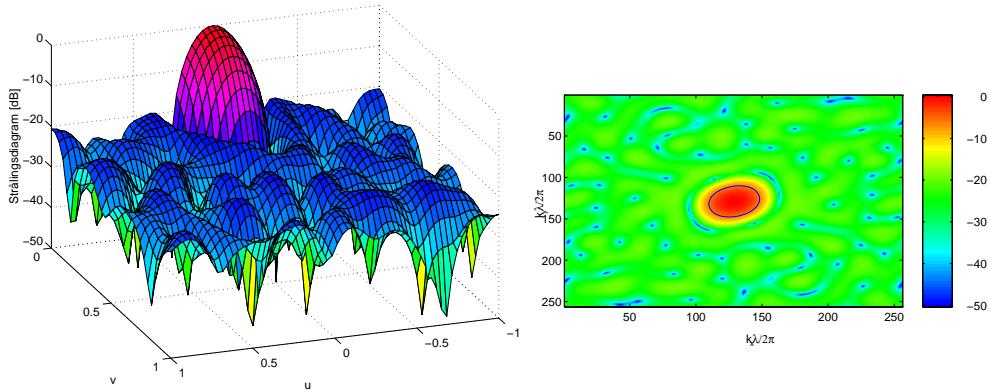
'o': Resultatet til de 22 simuleringene gjort i denne oppgaven.

'\*': Resultatene av LP optimeringen i [Holm *et al.*, 1997].

'x': Resultatene til den vektede Simulated Annealing optimeringen i [Trucco, 1997].

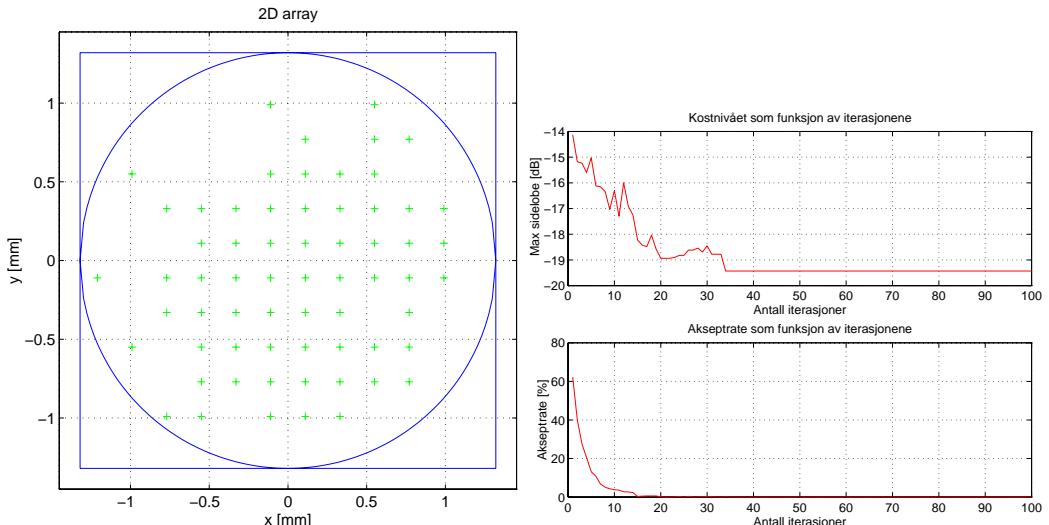


Figur 8.10: Strålingsdiagrammet til arrayet med lavest max sidelobe av de 22 simuleringene. Strålingsdiagrammet er her plottet som funksjon av  $v$ .



Figur 8.11: Strålingsdiagrammet til arrayet med lavest max sidelobe av de 22 simuleringene. Arrayet har  $-19.38$  dB max sidelobe og  $18.9^\circ$  hovedløbdebredde.

Figur 8.12: Intensitetsplott av  $u, v$ -planet til strålingsdiagrammet i figur 8.11. Aksene angir antall kontrollpunkter for  $u$  og  $v$  på  $[-1, 1]$ .



Figur 8.13: Elementfordelingen til arrayet med lavest max sidelobe av de 22 simuleringene.

Figur 8.14: Øverst: Kostfunksjonens utvikling til simuleringen som gav det laveste max sidelobenivået.

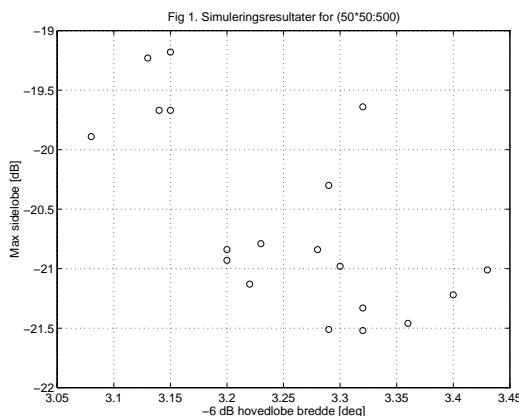
Nederst: Raten av aksepterte perturberinger ved hver temperatur til simuleringen som gav det laveste max sidelobenivået.

## 8.3 Et stort array

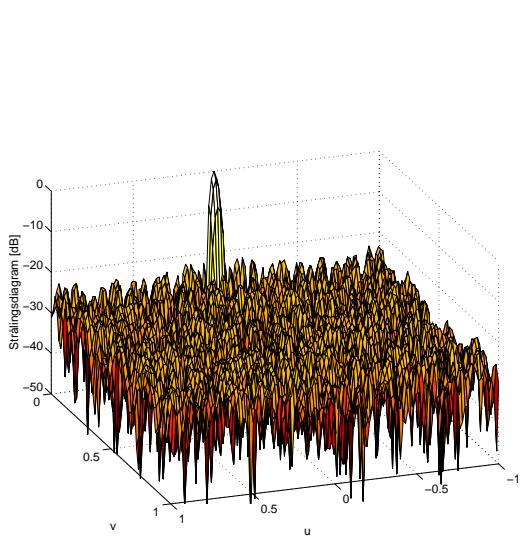
Et  $50 \times 50$  array tynnet til  $N = 500$  aktive elementer har et svimlende stort antall tynningsmønstre. Optimering av et slikt array, for å finne en løsning med akseptabelt lavt sidelobenivå, er et av målene til ESPRIT prosjektet [ESPRIT Project 22982 NICE, ]. Et slikt array muliggjør arrayprosessering med høy oppløsning over få kanaler hvor den utsendte bølgens retning kan styres elektronisk.

Det fulle arrayet har  $2.81^\circ$  hovedlobebredde. For stokastisk tynnede arrayer hevder [Steinberg, 1976] at forventningen til sidelobene vekk fra hovedloben for 2D arrayer ligger 3 dB over verdien for 1D arrayer. For dette optimeringsproblemet blir forventet sidelobenivå  $3 - 10\log_{10} N = -24.0$  dB. Forventet sidelobenivå for  $12 \times 12$  arrayet i forrige kapittel er  $-15.1$  dB, slik at den beste løsningen har en max sidelobe verdi 4 dB under den forventede middelverdien. Det samme resultatet kan man ikke forvente i dette tilfellet pga. ulik tynningsrate og kompleksitet.

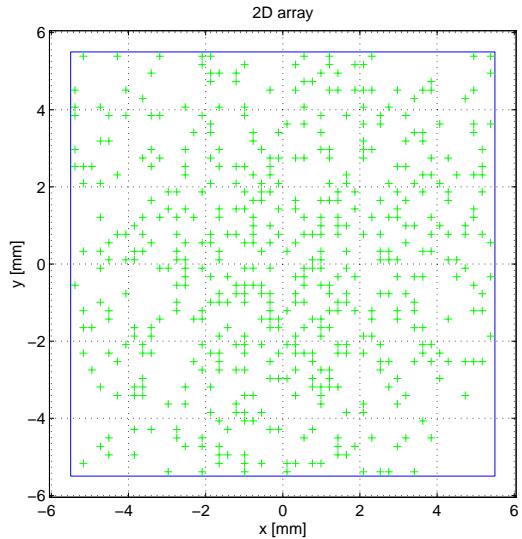
Totalt er det gjort 19 simuleringer for dette problemet. Max sidelobenivå og hovedlobebredde er gitt i figur 8.15. De beste løsningene har  $-21.5$  dB max sidelobe og omkring  $3.3^\circ$  hovedlobebredde. Strålingsdiagrammet til arrayet med lavest max sidelobe er gitt i figur 8.16 og elementfordelingen i figur 8.17. Parametrene til denne simuleringen er:  $T_i = \frac{T_0}{0.5i}$ ,  $T_0 = 8 \cdot 10^{-4}$ ,  $num\_iter = 500$ ,  $num\_pert = 5000$ ,  $u_0 = v_0 = 0.05$  og  $Npts_u = Npts_v = 64$ . Størkningsforløpet til simuleringen er gitt i figur 8.18. Algoritmen brukte 46t på en SGI Power Challenge med 1 av 4 CPUer aktive. Til denne simuleringen ble endringene i kostfunksjonen beregnet med bidragsmetoden gitt i lign. (8.3). Denne metoden krever flere hundre Mbytes RAM da  $A$  matrisen har dimensjon  $4096 \times 2500$ .



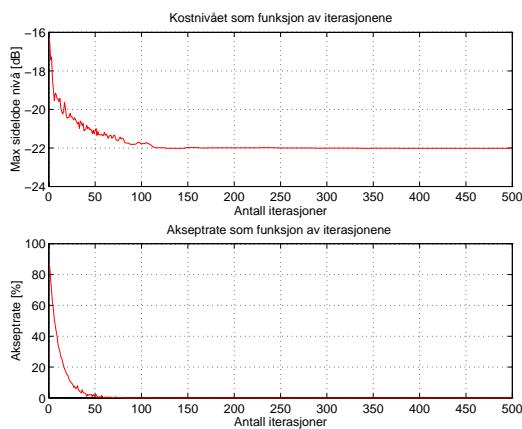
Figur 8.15: Max sidelobenivå og  $-6$  dB hovedlobebredde til resultatet av de 19 simuleringene.



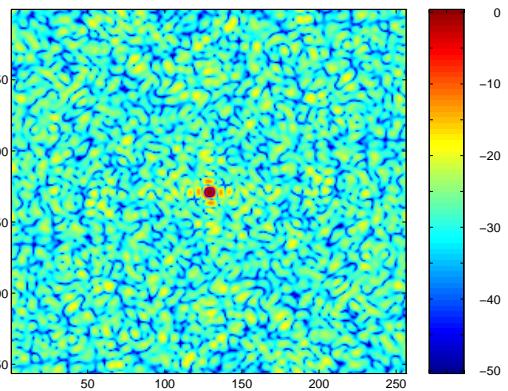
Figur 8.16: Strålingsdiagrammet til arrayet med lavest max sidelobe av de 19 simuleringene. Arrayet har  $-21.52 \text{ dB}$  max sidelobe og  $3.32^\circ$  hovedlobebredde.



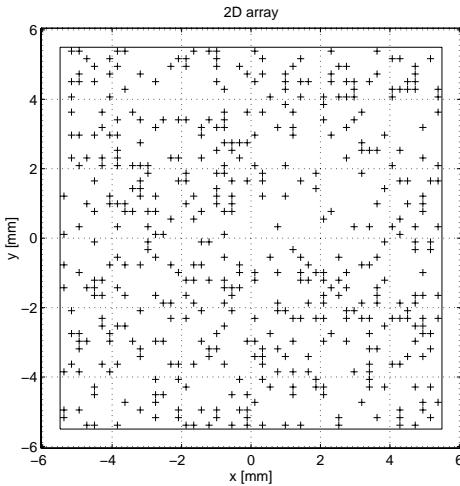
Figur 8.17: Elementfordelingen til arrayet med lavest max sidelobe.



Figur 8.18: Øverst: Kostfunksjonens utvikling til simuleringen som gav det laveste max sidelobenivået.  
Nederst: Raten av aksepterte perturberinger ved hver temperatur til simuleringen som gav det laveste max sidelobenivået.



Figur 8.19: Intensitetsplott av strålingsdiagrammet til et stokastisk tynnet array hvor elementposisjonene er trukket fra en uniform fordeling. Arrayet har  $2.88^\circ$  hovedlobebredde og  $-12.40 \text{ dB}$  max sidelobenivå.



Figur 8.20: Elementfordelingen til et stokastisk tynnet array hvor elementposisjonene er trukket fra en uniform fordeling.

Resultatene i figur 8.15 har langt dårligere max sidelobenivå enn  $-24$  dB. Grunnen til dette er hovedsakelig problemets størrelse. For det lineære tilfelle i kapittel 7.2.2 klarte algoritmen å redusere max sidelobenivået til det nest beste resultatet med at par tidels dB ved å øke antall undersøkte konfigurasjoner med faktor 100. I 2D tilfellet vil dette være mye vanskeligere pga. problemets størrelse. Figur 8.19 viser et intensitetsplot av strålingsdiagrammet til et stokastisk tynnet array, hvor de aktive elementene er trukket fra en uniform fordeling. Man kan se at første og andre sidelobe er betydelig større enn resten. Dette er også tilfelle for samtlige simuleringsresultater for dette problemet. I kapittel 4.3.1 så man at variansen til strålingsdiagrammet til stokastiske lineære arrayer, hvor elementposisjonene er trukket fra en uniform fordeling, når sitt første maksimum ved samme vinkel som første nullpunkt til det fulle arrayet. Med store 2D arrayer er det mulig å oppnå svært smale hovedlober, slik at første sidelobe kan opptre før variansen når sitt første maksimum. Dermed vil de første sidelobene i mindre grad la seg påvirke av elementplasseringen, slik at det ikke nødvendigvis eksisterer noen arrayer med max sidelobe under  $-24$  dB. Denne problemstillingen blir diskutert i [Iranpour *et al.*, ] hvor man benytter initielle konfigurasjoner trukket fra fordelinger som gir et lavt nivå for de første sidelobene. Med en genetisk algoritme har man i [Iranpour *et al.*, ] oppnådd resultater med ca 1dB lavere sidelobenivå enn resultatene i figur 8.15. Dermed kan man konkludere at Simulated Annealing algoritmen har gitt gode resultater innenfor de rammebeteglesene en uniform fordeling setter.

# Kapittel 9

## Oppsummering

### 9.1 Konklusjon

I denne oppgaven har egenskapene til tynnede arrayer blitt studert. Tynnede arrayer har en stor aperture med få aktive elementer. Forventet sidelobenivå for stokastisk tynnede 1D arrayer er  $-10 \log_{10} N$ , og for 2D arrayer ligger verdien 3 dB høyere. Max sidelobe verdiene varierer fra 0 dB (gitterlober) til verdier under middelverdien, men andelen av arrayer med lavt max sidelobenivå er svært liten. Derfor er det nødvendig å optimere elementplasseringen for å finne det arrayet med lavest max sidelobenivå. Optimering av max sidelobenivå er et kombinatorisk problem slik at eksakte metoder er utilstrekkelige for store arrayer. Dette har medført at man har forsøkt å legge betingelser på elementfordelingen for å redusere antall mulige løsninger. Ved å legge betingelser på coarrayet oppnår man en type arrayer med optimale oppløsningssegenskaper. Derimot er det veldig få som har innsett at disse arrayene også har optimale Fouriergenskaper. I kapittel 5 er nye coarray betingelser presentert. Få hull og redundanser i coarrayet gir strålingsdiagram med optimale/nær optimale Fouriergenskaper for tynnede arrayer generelt. Optimaliteten avhenger av plasseringen til redundansene og hullene. Redundansene bør forekomme ved små lag og hullene ved store lag, i tillegg må de være aperiodisk plassert.

I denne oppgaven er Simulated Annealing benyttet for å finne elementposisjonene som gir minimum max sidelobe uten å sette noen begrensninger på elementfordelingen. Algoritmen har gitt bedre resultater enn hva som finnes i litteraturen for 1D arrayer. I 2D tilfellet er det også presentert arrayer med flate sidelobeforlinger, men det er vanskelig å bedømme den optimale kvaliteten på disse da dette er et relativt nytt forskningsfelt med få referanser i litteraturen. Likevel har man sett at optimering av 2D arrayer er en svært vanskelig oppgave som krever enda bedre søkerutiner for å komme nærmere det globale minimum.

## 9.2 Forbedring av algoritme

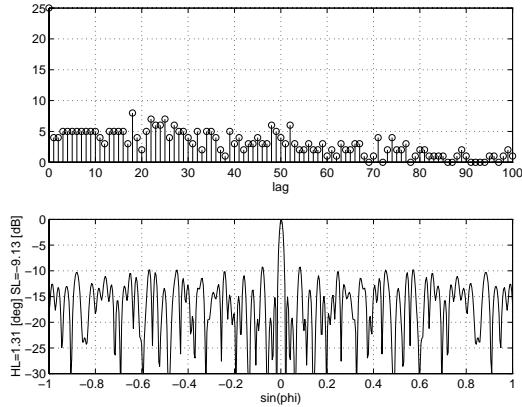
For å oppnå en algoritme som konvergerer raskere mot det globale minimum, må mest mulig informasjon om den optimale løsningen brukes i algoritmen. Fra kapittel 5 kjenner man betingelser som korrelasjonen til optimale arrayer oppfyller. Her presenteres to måter denne informasjonen kan brukes.

### 9.2.1 Komposisjon av optimale arrayer

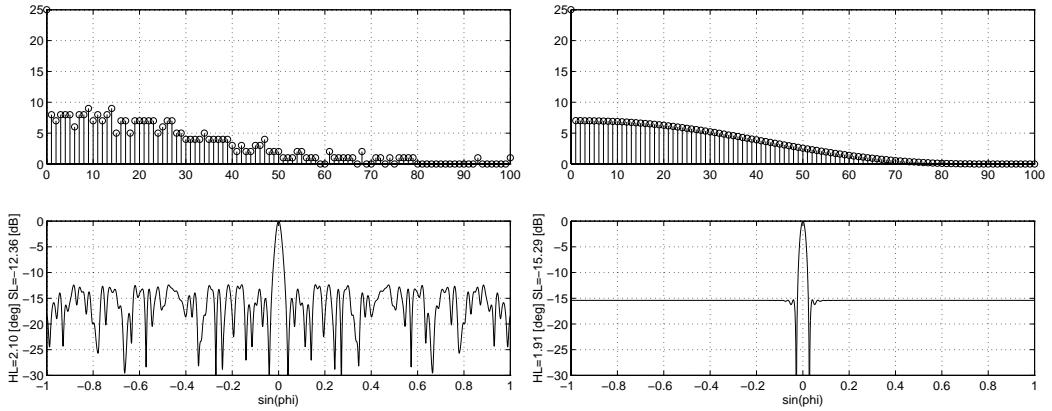
I [Dellas *et al.*, 1995] finnes en tabell over alle minimum hull arrayer for t.o.m.  $N = 19$  aktive elementer. Hvis man setter sammen *ulike* optimale arrayer til et stort array vil dette arrayet ha lignende egenskaper. For  $N = 7$  finnes det fem minimum hull arrayer. Hver for seg gir disse smale hovedlober og et tilnærmet flatt sidelobenivå. Hvis man benytter fire av disse arrayene får man et array med  $M = 101$  elementposisjoner og  $N = 25$  aktive elementer, hvor de aktive elementene kan være fordelt ved

```
array = [110010000010000000100001011000001000100000000100101 ...
          01100000010000010000100010100010010000100000000011]
```

hvor '1' markerer de aktive elementene. Coarrayet og strålingsdiagrammet til dette arrayet er gitt i figur 9.1. Dette arrayet har smalere hovedlobe enn det fulle arrayet og max sidelobe er  $-9.13$  dB. Hvis man sammenligner med de optimerte resultatene i figur 7.6 så har dette arrayet lavere max sidelobe for denne hovedlobebredden. Dette arrayet kan eventuelt benyttes for videre optimering.



Figur 9.1: Coarray og strålingsdiagram til et tynnet array med  $N = 25$  og  $M = 101$  satt sammen av 4 minimum hull arrayer for  $N = 7$ .



Figur 9.2: Coarray og strålingsdiagram til arrayet med lavest max sidelobe av simuleringene for  $N = 25$  og  $M = 101$ .

Figur 9.3: Optimalt coarray og tilhørende strålingsdiagram for et array med  $N = 25$  og  $M = 101$ .

### 9.2.2 Optimale coarrayer

Coarrayet kan optimeres med lineær programmering for å finne et coarray som gir en Dolph-Chebyshev sidelobefordeling. Dette coarrayet svarer ikke nødvendigvis til en fysisk elementfordeling, men definerer de globalt optimale egenskapene til coarrayet. I figur 9.2 er coarrayet og strålingsdiagrammet til arrayet med lavest max sidelobe i kapittel 7.2.2 gitt. Figur 9.3 viser det optimale coarrayet<sup>1</sup> med kontinuerlige verdier (svarer til vektede elementer) og det tilhørende strålingsdiagrammet. Dette strålingsdiagrammet har  $-15.29$  dB max sidelobe og litt smalere hovedlobe enn strålingsdiagrammet i figur 9.2. Den nedre grensen for max sideloben for dette problemet er altså  $-15.29$  dB. Formen til coarrayene i figur 9.2 og 9.3 er nokså like. Derfor kan man derfinere en søkerutine som genererer perturberingselementer som gir et coarray som konvergerer mot coarrayet i figur 9.3. Derved benytter man informasjon fra både korrelasjons- og Fourierdomenet slik at algoritmen har større sannsynlighet for å nå det globale minimum.

---

<sup>1</sup>Funnet med et lokalt LP-program skrevet av Skjalg Lepsøy

## **Tillegg A**

### **NORSIG artikkel**

The coarray of sparse arrays with minimum  
sidelobe level

Jon-Fredrik Hopperstad  
and Sverre Holm

To be published in Proc. NORSIG'98  
IEEE Nordic Signal Processing Symposium  
June 8–11, 1998  
Vigsø, Denmark



## Tillegg B

### Simula-kode for å generere alle mulige konfigurasjoner

```
BEGIN
    INTEGER ant,arraysize;
    Outtext("Antall elementer i tynnet array: ");
    Breakoutimage;
    Inimage;
    ant:=Inint;
    Outtext("Antall elementer i fullt array: ");
    Breakoutimage;
    Inimage;
    arrayszie:=Inint;
    BEGIN
        INTEGER ARRAY p(1:ant);
        BOOLEAN ARRAY brukt(1:arraysize);

        PROCEDURE gen(i);
        INTEGER i;
        BEGIN
            INTEGER elem;
            FOR elem:=p(i-1)+1 STEP 1 UNTIL arrayszie-1 DO
                IF NOT brukt(elem) THEN
                    BEGIN
                        brukt(elem):=TRUE;
                        p(i):=elem;
                        IF i<ant-1 THEN
                            gen(i+1)
                        ELSE

```

```

skriv_ut(p);
    bruktelem := FALSE;
END;
END gen;

PROCEDURE skriv_ut(p);
INTEGER ARRAY p;
BEGIN
    INTEGER k;
    FOR k:=1 STEP 1 UNTIL ant DO
        BEGIN
            fil.Outint(p(k), 0);
            fil.Outtext("  ");
        END;
        fil.Outimage;
    END skriv_ut;

TEXT filnavn;
REF(Outfile) fil;

Outtext("Filnavn for permutasjonene: ");
Breakoutimage;
Inimage;
filnavn:-Intext(70).Strip;

fil:-NEW Outfile(filnavn);
IF filnavn=/=NOTEXT AND THEN fil.Open(Blanks(200)) THEN
BEGIN
    p(1):=1;
    p(ant):=arraysize;
    gen(2);
    fil.Close;
END
ELSE
Error("Feil med fil.");
END;
END program;

```

# Tillegg C

## Matlab-kode

### C.1 Finne max sidelobenivå og hovedlobebredde

```
function [max_sl,h1]=finn_s1hl(allekonf,N_pts,arraysize)
%-----
% Function      : [max_sl,h1]=finn_s1hl(allekonf,N_pts,arraysize)
% Description   : Returns the peak sidelobe level and the -6 dB beamwidth of
%                  the arrays in allekonf.
%
% Language     : Matlab 4.2
% Written by   : Jon-Fredrik Hopperstad
%
% Version no.  : 1.0 JFH    28/5-97 Initial version.
%
% Input        : allekonf    ---> matrix with the positions for the active
%                  elements for each array in the rows
%                  N_pts       ---> number of controllpoints in the
%                  arraysize   ---> the length of the linear array
%
% Output       : max_sl     ---> the peak sidelobe levels for the arrays
%                  h1         ---> the -6 dB beamwidths for the arrays
%-----
```

```
[r,ant]=size(allekonf);
max_sl=zeros(r,1);
h1=zeros(r,1);
u=0:1/(N_pts-1):1;
Q=ant;

for k=1:r,
    konfig=allekonf(k,:);
    array=zeros(1,arraysize);
```

```

array(konfig)=ones(1,ant);
p=abs(fft(array,2*N_pts));
dB=20*log10(p./Q);
ind=1;
while dB(ind+1)<dB(ind) & dB(ind+1)>-6,
    ind=ind+1;
end;
if dB(ind+1)>dB(ind),           % sjekker om dette er et nullpkt.
    max_sl(k)=max(dB(ind:N_pts));
    while dB(ind+1)<-6,           % finner -6 dB pkt.
        ind=ind+1;
    end;
    if abs(-6-dB(ind+1))<abs(-6-dB(ind)), % finner indeksen nærmest -6 dB.
        ind=ind+1;
    end;
    h1(k)=asin(u(ind))/pi*180*2;
else                               % dette er -6 dB punktet.
    ;
    if abs(-6-dB(ind+1))<abs(-6-dB(ind)), % finner indeksen nærmest -6 dB.
        ind=ind+1;
    end;
    h1(k)=asin(u(ind))/pi*180*2;
    while dB(ind+1)<dB(ind),   % finner nullpkt.
        ind=ind+1;
    end;
    max_sl(k)=max(dB(ind:N_pts));
end;

end;

return

```

## C.2 Optimering av 1D arrayer med Simulated Annealing og FFT

```
%-----
% Function      : [konfig,kost,aksept,globalE,global_konf]=sa1Dfft(init, ...
%                  arraylength,T0,num_iter,num_pert,beta,u0,Npts)
%
% Description   : Optimizes THINNING of a 1D array transducer by
%                  minimizing the sidelobe level while keeping the thinning
%                  factor constant.
%
%                  Uses a Simulated Annealing algorithm with temperature
%                  reduction T_i=T0/(beta*i)
%
% Language     : Matlab 4.2
% Written by   : Jon-Fredrik Hopperstad
%
% Version no.  : 1.0 JFH 10/3-97 Initial version.
%                  1.1 JFH 23/5-97 Added parameter beta
%                  2.0 JFH 1/6-97 Included code for the objection function
%                  in this function.
%
% Input         : init          ---> vector with the initial active
%                  elementpositions
%                  arraylength  ---> number of elements in full aperture
%                  T0           ---> initial temperature
%                  num_iter     ---> number of temperature decrementation
%                  num_pert     ---> number of perturbations at each temperature
%                  beta          ---> temperature parameter
%                  u0            ---> start angle for optimization
%                  Npts          ---> number of controlpoints in 0<=u<=1
%
% Output        : konfig        ---> vector with active element positions
%                  when finished
%                  kost          ---> the costvalues at each iteration
%                  aksept        ---> the number of accepted perturbations at
%                  each iteration
%                  globalE       ---> the lowest cost-value
%                  global_konf   ---> konfig with the lowest cost-value
%-----
function[konfig,kost,aksept,globalE,global_konf]=sa1Dfft(init, ...
arraylength,T0,num_iter,num_pert,beta,u0,Npts)

ant=length(init);
array=zeros(1,arraylength);
array(init)=ones(size(init));
u=0:1/(Npts-1):1;
tmp=find(u>u_0);
```

```

u0=tmp(1)-1;
W=fft(array,2*Npts);
Energi=max((abs(W(u0:Npts)./ant).^2);

konfig=init;
globale=Energi;
global_konf=init;

for iter=1:num_iter;
    temp=T0/(beta*iter);
    for i=1:num_pert,
%-----Finner en nabokonfigurasjon-----
        ind=round((ant-3)*rand)+2;
        ut=konfig(ind);
        like=1;
        while ~isempty(like),
            inn=round((arraysize-3)*rand)+2;
            like=find(konfig==inn);
        end;
        testkonfig=konfig;
        testkonfig(ind)=inn;
%-----
%-----Beregner kostverdien til testkonfigurasjonen-----
        testarray=zeros(1,arraylength);
        array(testkonfig)=ones(size(testkonfig));
        W=fft(testarray,2*Npts);
        testEnergi=max((abs(W(u0:Npts)./ant).^2);
%-----
        deltaE=testEnergi-Energi;
        r=rand;
        if deltaE<0 | r<exp(-deltaE/temp),
            konfig=testkonfig;
        Energi=testenergi
            aksept(iter)=aksept(iter)+1;
            if Energi<globale,
                globale=Energi;
                global_konf=konfig;
            end;
        end;
        kost(iter)=10*log10(Energi);
    end;

    return;

```

### C.3 Optimering av 1D arrayer med Simulated Annealing og bidragsmetoden

```
%-----
% Function      : [konfig,kost,akseprate,globale,global_konf,num_iter]= ...
%                   sa1Dbest(init,arraylength,num_pert,T0,T_min,gamma,u0,Npts)
%
% Description   : Optimizes THINNING of a 1D array transducer by
%                   minimizing the sidelobe level while keeping the thinning
%                   factor constant.
%
%                   Uses a Simulated Annealing algorithm with temperature
%                   reduction T(i+1)=T(i)*exp(-gamma*T(i)/stdE);
%
% Language     : Matlab 4.2
% Written by   : Jon-Fredrik Hopperstad
%
% Version no.  : 1.0 JFH 16/5-97 Initial version.
%
% Input         : init          ---> vector with the initial active
%                   elementpositions
%                   arraylength  ---> number of elements in full aperture
%                   num_pert     ---> number of perturbations at each temperature
%                   T0           ---> initial temperature
%                   T_min        ---> freezing temperature
%                   gamma        ---> temperature parameter
%                   u0           ---> start angle for optimization
%                   Npts         ---> number of controlpoints in 0<=u<=1
%
% Output        : konfig        ---> vector with active element positions
%                   when finished
%                   kost          ---> the costvalues at each iteration
%                   akseprate    ---> the number of accepted perturbations at
%                   each iteration
%                   globale       ---> the lowest cost-value
%                   global_konf  ---> konfig with the lowest cost-value
%                   num_iter     ---> number of temperature decrements
%-----
function[konfig,kost,akseprate,globale,global_konf,num_iter]= ...
    sa1Dbest(init,arraylength,num_pert,T0,T_min,gamma,u0,Npts)

ant=length(init);
konfig=init;
disp(konfig)

%-----Beregner responsen for hele arrayet-----
u=0:1/(N_pts-1):1;
```

```

u=u';
A=exp(j*pi*u*[1:arraylength]);
%-----Beregner SL-nivå til init-----
p=sum(A(:,init)')';
tmp=find(u>u0);
u0=tmp(1)-1;
p=p(u0:N_pts);
Energi=max((abs(p)./ant).^2);
%-----globalE=Energi;
global_konf=init;
temp=T0;
iter=0;
while temp>T_min,
    iter=iter+1;
    akseprate(1,iter)=0;
    aksept=zeros(1,num_pert);
    ant_aksept=0;
    for i=1:num_pert,
        ind=round((ant-3)*rand)+2;
        ut=konfig(ind);
        like=1;
        while like~=[],
            inn=round((arraylength-3)*rand)+2;
            like=find(konfig==inn);
        end;
        testkonfig=konfig;
        testkonfig(ind)=inn;

        %-----SL-nivå til pert-----
        p_test=p-A(u0:N_pts,ut)+A(u0:N_pts,inn);
        testEnergi=max((abs(p_test)./ant).^2);
        %-----deltaE=testEnergi-Energi;
        r=rand;
        if deltaE<0 | r<exp(-deltaE/temp),
            konfig=testkonfig;
            Energi=testEnergi;
            p=p_test;
            ant_aksept=ant_aksept+1;
            aksept(ant_aksept)=Energi;
            akseprate(1,iter)=akseprate(1,iter)+1;
            if Energi<globale,
                globale=Energi;
                global_konf=konfig;
            end;
        end;
    end;
end;

```

```
    end;
end;
kost(iter)=10*log10(Energi);
ant_aksept=max(1,ant_aksept);
aksept=aksept(1:ant_aksept);
tmp=std(aksept);
if tmp~=0,
    stdE=tmp;
end;
temp=temp*exp(-gamma*temp/stdE);
end;

return;
```

## C.4 Optimering av 1D og 2D arrayer med Simulated Annealing og bidragsmetoden

```
%-----%
% Function      : [konfig,kost,aksept,globalE,global_konf]=sa1_2D(init, ...
%                   tillatt,T0,num_iter,num_pert,beta, ...
%                   arraysize,u0,v0,Npts_u,Npts_v)
% Description   : Optimizes THINNING of a 1D or 2D array transducer by
%                   minimizing the sidelobe level while keeping the thinning
%                   factor constant.
%
%                   Uses a Simulated Annealing algorithm.
%
% Language     : Matlab 4.2
% Written by   : Jon-Fredrik Hopperstad
%
% Version no.  : 1.0 JFH    9/9-97 Initial version.
%                   1.1 JFH    16/9-97 Added code for optimization of
%                   non-rectangular transducers.
%
% Input        : init       ----> vector with the initial active
%                   elementpositions
%                   tillatt    ----> vector with allowed element positions
%                   T0         ----> initial temperature
%                   num_iter   ----> number of temperature decrementation
%                   num_pert   ----> number of perturbations at each temperature
%                   beta        ----> temperature parameter
%                   arraysize  ----> size of rectangular grid
%                   u0,v0      ----> start angles for optimization
%                   Npts_u     ----> number of controlpoints in 0<=u<=1
%                   Npts_v     ----> number of controlpoints in 0<=v<=1
%
% Output       : konfig     ----> vector with active element positions
%                   when finished
%                   kost       ----> the costvalues at each iteration
%                   aksept    ----> the number of accepted perturbations at
%                   each iteration
%                   globalE    ----> the lowest cost-value
%                   global_konf  ----> konfig with the lowest cost-value
%-----%
function[konfig,kost,aksept,globalE,global_konf]=sa1_2D(init, ...
                           tillatt,T0,num_iter,num_pert,beta, ...
                           arraysize,u0,v0,Npts_u,Npts_v)

%-----Lag bidragsmatris A-----
% Lager strålingsdiagram-matrise A, hvor hver kolonne
% representerer hver posisjon i arrayet. Laget i
```

```

%      blokker hvor v er konstant.
u_h1=[-1:(1-u0)/(Npts_u-1):-u0 ... %ekskluderer
      u0:(1-u0)/(Npts_u-1):1]'; %hovedloben

u=[ -1:2/(2*Npts_u-1):1]';

v=[ 0:1/(Npts_v-1):1]';
Nu=length(u);
Nv=length(v);
A=zeros(Nu*Nv,arraysize(1)*arraysize(2));

for k=1:arraysize(2),
    for l=1:Nv,
        if v(l)<v0;
            A((l-1)*Nu+1:l*Nu,(k-1)*arraysize(1)+1:k*arraysize(1))=...
                exp(j*pi*(u_h1*[1:arraysize(1)]+v(l)*k));
        else
            A((l-1)*Nu+1:l*Nu,(k-1)*arraysize(1)+1:k*arraysize(1))=...
                exp(j*pi*(u*[1:arraysize(1)]+v(l)*k));
        end
    end
end
%-----.

ant=length(init);
ant_tillatt=length(tillatt);
konfig=init;
aksept=zeros(1,num_iter);
kost=zeros(1,num_iter);

%-----Beregner SL-nivå til init-----
W=sum(A(:,init)')';
Energi=max((abs(W)./ant).^2);
%%-----.

globalE=Energi;
global_konf=konfig;
for iter=1:num_iter;
    Temp=T0/(beta*iter);
    for i=1:num_pert,
        ind_ut=round((ant-1)*rand)+1;
        ut=konfig(ind_ut);
        like=1;
        while ~isempty(like),
            ind_inn=round((ant_tillatt-1)*rand)+1;
            inn=tillatt(ind_inn);
            like=find(konfig==inn);
        end;
        testkonfig=konfig;

```

```

testkonfig(ind_ut)=inn;

%-----SL-nivå til pert-----
W_test=W-A(:,ut)+A(:,inn);
testEnergi=max((abs(W_test)./ant).^2);
%-----

deltaE=testEnergi-Energi;
r=rand;
if deltaE<0 | r<exp(-deltaE/Temp),
    konfig=testkonfig;
    Energi=testEnergi;
    W=W_test;
    aksept(iter)=aksept(iter)+1;
    if Energi<globale,
        globale=Energi;
        global_konf=konfig;
    end;
end;
end;
kost(iter)=10*log10(Energi);
end;
end;

return;

```

# Bibliografi

- [Aarts and Lenstra, 1997] Emile Aarts and Jan Karel Lenstra. *Local search in combinatorial optimization*. Wiley-Interscience, 1997. ISBN 0-471-94822-5.
- [Adams, 1991] John W. Adams. A new optimal window. *IEEE Transactions on Signal Processing*, 39(8):1753–1769, August 1991.
- [Austeng *et al.*, 1997] Andreas Austeng, Sverre Holm, Peter K. Weber, Niels Aakvaag, and Kam-biz Iranpour. 1D and 2D algorithmically optimized sparse arrays. In *Proc. IEEE Ultrasonic Symp.*, Toronto, Canada, October 1997. [http://www.ifi.uio.no/~andrea/toronto\\_m.pdf](http://www.ifi.uio.no/~andrea/toronto_m.pdf).
- [Bedrosian, 1986] Samuel D. Bedrosian. Nonuniform linear arrays: Graph-theoretic approach to minimum redundancy. *Proceedings of the IEEE*, 74(7):1040–1043, July 1986.
- [Bloom and Golomb, 1977] Gary S. Bloom and Solomon W. Golomb. Application of numbered undirected graphs. *Proceedings of the IEEE*, 65(4):562–570, April 1977.
- [Boni *et al.*, 1994] Carlo Boni, Mario Richard, and Sergio Barbarossa. Optimal configuration and weighting of nonuniform arrays according to a maximum ISLR criterion. In *Proc. IEEE ICASSP-94*, volume V, pages 157–160, 1994.
- [Davis, 1989] L. Davis. *Genetic Algorithms and Simulated Annealing*. Morgan Kaufmann, 1989.
- [De Graaf and Johnson, 1984] Stuart R. De Graaf and Don H. Johnson. Optimal linear arrays for narrow-band beamforming. In *Proc. ICASSP-84*, volume 7, pages 40.8.1–40.8.4, San Diego, California, March 1984.
- [Dollas *et al.*, 1995] A. Dollas, Rankin W. T., and McCracken D. New algorithms for Golomb ruler derivation and proof of the 19 mark ruler. Technical report, Dept. of Electrical Engineering, Duke University, Durham, NC 27708-0291, November 1995. <http://www.ee.duke.edu/~wrakin/golomb/golomb.html>.
- [Elgtun, 1996] Bjørnar Elgtun. Optimization of responses from 2D arrays for medical ultrasound. Cand. Scient thesis, Institutt for Informatikk, UiO, May 1996. <http://www.ifi.uio.no/~ftp/publications/cand-scient-theses/BElgetun.ps>.
- [ESPRIT Project 22982 NICE, ] ESPRIT Project 22982 NICE. Real-time 3D ultrasound imaging system with advanced transducer arrays. <http://www.ifi.uio.no/~sverre/nice.html>.
- [Gardner, 1972] Martin Gardner. Mathematical games. *Scientific American*, pages 108–110, March 1972.
- [Harris, 1978] Fredric J. Harris. On the use of windows for harmonic analysis with the discrete Fourier transform. *Proc. IEEE*, 66:51–83, January 1978.

- [Hayward, 1994] Thomas J. Hayward. Optimization og hydrophone placement for acoustic arrays using simulated annealing. *J. Acoust. Soc. Am.*, 95(1):201–212, January 1994.
- [Hendricks, 1991] Walter J. Hendricks. The totally random versus the bin approach for random arrays. *IEEE Transactions on Antennas and Propagation*, 39(12):1757–1762, December 1991.
- [Holm *et al.*, 1997] Sverre Holm, Bjørnar Elgetun, and Geir Dahl. Properties of the beampattern of weight- and layout-optimized sparse arrays. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 44(5):983–991, September 1997.
- [Holm, 1995] Sverre Holm. Ultralydteknikker. Institutt for informatikk, Universitetet i Oslo, October 1995.
- [Holm, 1996] Sverre Holm. Minimum sidelobe energy versus minimum peak sidelobe level for sparse array optimization. In *Proc. NORSIG-96*, pages 227–230, Espoo, Finland, September 1996. <http://www.ifi.uio.no/~ftp/publications/preprints/SHolm-5.ps>.
- [Holm, 1997] Sverre Holm. Real-time 3D medical ultrasound –Signal processing challenges. In *Proceedings NORSIG-97*, pages 124–128, May 1997. <http://www.ifi.uio.no/~ftp/publications/preprints/SHolm-7.ps>.
- [Hopperstad and Holm, 1998] Jon-Fredrik Hopperstad and Sverre Holm. The coarray of sparse arrays with minimum sidelobe level. In *Proc. NORSIG-98*, Vigsø, Denmark, June 1998. To be published.
- [Ingber, 1989] Lester Ingber. Very fast Simulated Re-Annealing. *Journal Mathematical Computer Modelling*, 12(8):967–973, 1989.
- [Ingber, 1993] Lester Ingber. Simulated Annealing: Practice versus theory. *Journal Mathematical Computer Modelling*, 18(11):29–57, 1993.
- [Iranpour *et al.*, ] Kamran Iranpour, Sverre Holm, Andreas Austeng, and Skjalg Lepsøy. Improved 2D sparse array optimization using genetic algorithm. To be published.
- [Ishiguro, 1980] Masato Ishiguro. Minimum redundancy linear arrays for large number of antennas. *Radio Science*, 15(6):1163–1170, November 1980.
- [Johnson and Dudgeon, 1993] Don H. Johnson and Dan E. Dudgeon. *Array Signal Processing: Concepts and Techniques*. Prentice Hall, 1993. ISBN 0-13-048513-6.
- [Kirkpatrick *et al.*, 1983] S. Kirkpatrick, C.D. Gelatt Jr., and M.P. Vecchi. Optimization by Simulated Annealing. *Science*, 220(4598):671–680, May 1983.
- [Kittel and Kroemer, 1980] Charles Kittel and Herbert Kroemer. *Thermal Physics*. W. H. Freeman, 1980. ISBN 0-7167-1088-9.
- [Leech, 1956] J. Leech. On the representation of  $1, 2, \dots, n$  by differences. *Journal of the London Mathematical Society*, 31:160–169, 1956.
- [Linebarger *et al.*, 1993] Darel A. Linebarger, I. Hal Sudborough, and Ioannis G. Tollis. Difference bases and sparse sensor arrays. *IEEE Transactions on Information Theory*, 39(2):716–721, March 1993.
- [Lo and Lee, 1966] Y.T. Lo and S.W. Lee. A study of space-tapered arrays. *IEEE Transactions on Antennas and Propagation*, AP-14(1):22–30, January 1966.
- [Lockwood and Foster, 1996] Geoffrey R. Lockwood and F. Stuart Foster. Optimizing the radiation pattern of sparse periodic two-dimensional arrays. *IEEE Transactions of Ultrasonics, Ferroelectrics and Frequency Control*, 43(1):15–19, January 1996.

- [Lockwood *et al.*, 1996] Geoffrey R. Lockwood, Pai-Chi Li, Matthew O'Donnell, and F. Stuart Foster. Optimizing the radiation pattern of sparse periodic linear arrays. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 43(1):7–14, January 1996.
- [Miyashita, 1997] Toyokatsu Miyashita. High precision multifrequency acoustic radar with an unequally spaced frequency array designed by Simulated Annealing. *Japanese Journal of Applied Physics*, 36:3192–3196, 1997.
- [Moffet, 1968] Allan T. Moffet. Minimum-redundancy linear arrays. *IEEE Transactions on Antennas and Propagation*, AP-16(2):172–175, March 1968.
- [Murino *et al.*, 1996] Vittorio Murino, Andrea Trucco, and Carlo S. Regazzoni. Synthesis of unequally spaced arrays by Simulated Annealing. *IEEE Transactions on Signal Processing*, 44(1):119–123, January 1996.
- [Oppenheim and Schafer, 1989] Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing*. Prentice-Hall, 1989. ISBN 0-13-216771-9.
- [Rossouw *et al.*, 1997] M.J. Rossouw, J. Joubert, and D.A. McNamara. Thinned arrays using a modified minimum redundancy technique. *Electronic Letters*, 33(10):826–827, May 1997.
- [Ruf, 1993] Christopher S. Ruf. Numerical annealing of low-redundancy linear arrays. *IEEE Transactions on Antennas and Propagation*, 41(1):85–90, January 1993.
- [Smith *et al.*, 1991] Stephen W. Smith, Henry G. Pavy Jr., and Olaf T. von Ramm. High-speed ultrasound volumetric imaging system —Part 1: Transducer design and beam steering. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 38(2):100–108, March 1991.
- [Steinberg, 1976] Bernard D Steinberg. *Principles of Aperture and Array System Design*. Wiley, 1976. ISBN 0-471-82102-0.
- [Trucco and Repetto, 1996] Andrea Trucco and Francesco Repetto. A stochastic approach to optimizing the aperture and the number of elements of an aperiodic array. In *Oceans96*, pages 1510–1515, Fort Lauderdale, Florida, September 1996. MTS/IEEE.
- [Trucco, 1997] Andrea Trucco. Synthesis of aperiodic planar arrays by a stochastic approach. In *Proc. IEEE Oceans' 97*, Halifax Canada, October 1997.
- [Ult, ] MATLAB Toolbox for ultrasound field simulation. <http://www.ifi.uio.no/~ultrasim>.
- [van Laarhoven and Aarst, 1987] P.J. van Laarhoven and E.H.L. Aarst. *Simulated Annealing: Theory and applications*. Dordrecht, Reidel, 1987. ISBN 90-277-2513-6.
- [Vertatschitsch and Haykin, 1986] E. Vertatschitsch and S. Haykin. Nonredundant arrays. *Proceedings of the IEEE*, 74(1):217, January 1986.