



Simuleringsmodell CARABAS för integration i MOSART

Peter Follo, Gustav Haapalahti, Torleif Martin, Lars Ulander

FOI är en huvudsakligen uppdragsfinansierad myndighet under Försvarsdepartementet. Kärnverksamheten är forskning, metod- och teknikutveckling till nytta för försvar och säkerhet. Organisationen har cirka 1350 anställda varav ungefär 950 är forskare. Detta gör organisationen till Sveriges största forskningsinstitut. FOI ger kunderna tillgång till ledande expertis inom ett stort antal tillämpningsområden såsom säkerhetspolitiska studier och analyser inom försvar och säkerhet, bedömningen av olika typer av hot, system för ledning och hantering av kriser, skydd mot hantering av farliga ämnen, IT-säkerhet och nya sensorers möjligheter.



FOI
Totalförsvarets forskningsinstitut
Sensorteknik
Box 1165
581 11 Linköping

Tel: 013-37 80 00
Fax: 013-37 81 00

www.foi.se

Peter Follo, Gustav Haapalahti, Torleif Martin, Lars Ulander

Simuleringsmodell CARABAS för integration i MOSART

Utgivare FOI - Totalförsvarets forskningsinstitut Sensorteknik Box 1165 581 11 Linköping	Rapportnummer, ISRN FOI-R--1660--SE	Klassificering Teknisk rapport
	Forskningsområde 4. Ledning, informationsteknik och sensorer	
	Månad, år Maj 2005	Projektnummer E3029
	Delområde 42 Spaningssensorer	
	Delområde 2	
Författare/redaktör Peter Follo Gustav Haapalahti Torleif Martin Lars Ulander	Projektledare Lars Ulander	
	Godkänd av Björn Larsson	
	Uppdragsgivare/kundbeteckning FM	
	Tekniskt och/eller vetenskapligt ansvarig Lars Ulander	
Rapportens titel Simuleringsmodell CARABAS för integration i MOSART		
Sammanfattning Dokumentet beskriver en syntetisk aperturradarmodell (SAR) anpassad för VHF-bandet (CARABAS) implementerad under 2004. Simulatorens är tidsstegad och använder sig utav ett antal olika lokala modeller i kombination med superposition för att generera radarrådata från en stor yta. Scenen som kan simuleras består av trädstammar samt enskilda spridare, såsom fordon, på dielektrisk mark. Genereringen av SAR-bilderna sker med tidigare framtagna algoritmer. Via MOSART fås ett användargränssnitt till modellen.		
Nyckelord modellering, simulering, radar, SAR, CARABAS, SEMARK, MOSART		
Övriga bibliografiska uppgifter	Språk Svenska	
ISSN 1650-1942	Antal sidor: 32 s.	
Distribution enligt missiv	Pris: Enligt prislista	

Issuing organization FOI – Swedish Defence Research Agency Sensor Technology P.O. Box 1165 SE-581 11 Linköping	Report number, ISRN FOI-R--1660--SE	Report type Technical report
	Programme Areas 4. C4ISTAR	
	Month year May 2005	Project no. E3029
	Subcategories 42 Above water Surveillance, Target acquisition and Reconnaissance	
	Subcategories 2	
Author/s (editor/s) Peter Follo Gustav Haapalahti Torleif Martin Lars Ulander	Project manager Lars Ulander	
	Approved by Björn Larsson	
	Sponsoring agency Swedish Armed Forces	
	Scientifically and technically responsible Lars Ulander	
Report title (In translation) Simulation model CARABAS for MOSART integration		
Abstract This document describes a synthetic aperture radar (SAR) model adopted for the VHF band (CARABAS) and implemented during 2004. The simulator is time stepped and uses several local models in combination with the superposition principle to generate radar raw data from a large area. The scene possible to simulate contains tree stems and individual scatterers, e.g. vehicles, on dielectric ground. SAR image formation is performed with previously developed algorithms. Through MOSART a user interface is provided for the model.		
Keywords modelling, simulation, radar, SAR, CARABAS, SEMARK, MOSART		
Further bibliographic information	Language Swedish	
ISSN 1650-1942	Pages 32 p.	
Price acc. to pricelist		

INNEHÅLLSFÖRTECKNING

SIMULERINGSMODELL CARABAS FÖR INTEGRATION I MOSART.....	3
BAKGRUND.....	4
ANVÄNDNINGSMOMRÅDEN.....	5
VAD KAN SIMULERAS.....	6
BEGRÄNSNINGAR.....	8
START-STOPP APPROXIMATION.....	8
SUPERPOSITION.....	8
ÖVRIGA BEGRÄNSNINGAR.....	13
SIMULATORNS ARBETSCYKEL.....	14
MÅLMODELLER.....	16
KLOTTERMODELL.....	16
IMPLEMENTATION.....	17
MÅL.....	18
HJÄLPKLASSER.....	20
SENSORBÄRARE.....	23
RADAR (STYRENHET).....	23
INTEGRATION AV CARABAS-MODELLEN TILL MOSART.....	25
PLATTFORMSFEDERATEN.....	26
RADARFEDERATEN.....	26
PROCESSERINGSFEDERATEN.....	27
APPENDIX A KRAVLISTA – ÅTERKOPPLING.....	28
PLATTFORM.....	29
RADAR.....	30
SENSORMODELL.....	31

FOI-R--1660--SE

SIMULERINGSMODELL **CARABAS** FÖR INTEGRATION I **MOSART**

Detta dokument beskriver den radarsimulator som har implementerats under 2004 för CARABAS:s räkning. För ytterligare detaljer hänvisas till referensmanualen [1].

BAKGRUND

För att kunna utvärdera sensorer samt sensorer i samverkan har försvarsmakten ett behov av sensormodeller, bland annat för lågfrekvent SAR samt för SAR på mikrovågor. Vid en inledande inventering visade det sig att FOI ej har tillgång till några väl fungerande SAR-modeller varför olika sätt att införskaffa sådana undersöktes. Möjligheter som studerades var att köpa kommersiella modeller, vetenskapligt utbyte med andra institutioner samt att på egen hand utveckla modellerna.

Slutsatsen blev att det ej gick att uppbringa någon lämplig systemmodell för lågfrekvent SAR men att flera möjligheter återstår vad det gäller en SAR modell för mikrovågor. Det bedömdes också viktigt att kompetensen på detta område skulle finnas och kunna vidareutvecklas inom FOI. Det beslutades därför att FOI på egen hand tar fram en sensormodell för lågfrekvent SAR emedan modellen för de högre frekvensbanden lämpligen införskaffas genom ett kommersiellt inköp eller utbyte med annan institution.

Under 2004 lades tonvikten på att modellera en lågfrekvent SAR inom frekvensbandet 20 – 80 MHz. Den slutgiltiga ambitionen är dock en modell för både VHF och UHF. Syftet är att ta fram en radarmodell för att studera fenomenologin för lågfrekvent SAR. I samarbete med projekt SEMARK studeras också prestanda hos samverkande sensorer.

Fokus har varit att ta fram en radarsimulator som efterliknar CARABAS men den underliggande modellen klarar dessutom av multistatiska radarsystem med godtyckligt antal antennelement.

ANVÄNDNINGSSOMRÅDEN

Tack vare den väl dokumenterade omgivningen inom delar av markstridsskolan Kvarns övningsområde kan man ta fram simulerat radardata och jämföra detta med verkliga mätningar av området. Genom att jämföra det simulerade datat med det verkliga fås en ökad förståelse för den underliggande elektromagnetiska interaktionen och fenomenologin.

Simulatorn ger också möjlighet att undersöka prestanda för ännu ej existerande lågfrekventa radarsystem såsom bi- och multistatiska system samt funktionaliteten GMTI (Ground Moving Target Indication). Den kan även fungera som hjälpmedel för att ta fram signalbehandlingsalgoritmer för dessa ännu ej tillgängliga radarsystem. Via MOSART [2] nås SEMARK:s andra sensormodeller vilket möjliggör simuleringar av multisensorsystem och där dess prestanda och egenskaper kan undersökas och uppskattas.

Simulatorn kan också fungera som ett underlag för att ta fram andra realistiska men mer specialiserade realtidssimulatorer som kan användas i exempelvis ledningssystemsimulatorer.

VAD KAN SIMULERAS

Ett system med godtyckligt antal antenner och plattformar som är tidsstegat och tidssynkroniserat. Plattformarna kan ha godtyckliga flygbanor. Spaningsavstånd, stråkbredd och vågform kan bestämmas från puls till puls. Även på plattformen rörliga antenner kan simuleras. Illustration 2 visar en bistatisk geometri.

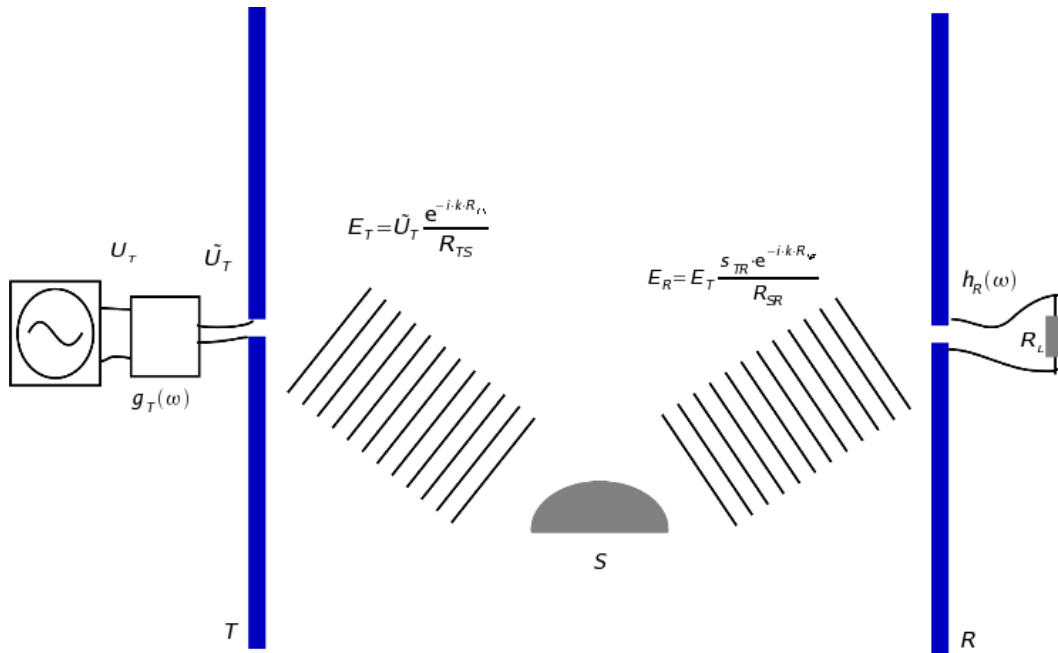
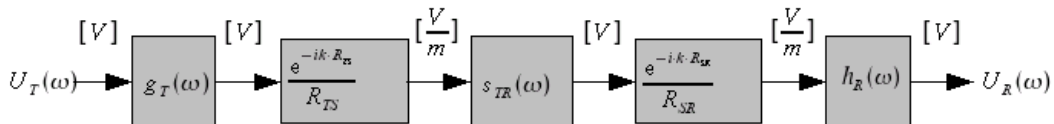


Illustration 1: Bistatiska geometrier kan simuleras.

Modellen implementerar följande system för E-fältet.



Den teoretiska signalen $U_T(\omega)$ filtreras genom den sändande antennens överföringsfunktion $g_T(\omega)$ som i spridarens riktning bildar ett E -fält enligt;

$$E_T(\omega) = g_T(\omega) \frac{U_T \cdot e^{-i \cdot k \cdot R_{TS}}}{R_{TS}}$$

Spridaren reflekterar signalen mot mottagande antennen enligt;

$$E_R(\omega) = E_T(\omega) \cdot s_{TR}(\omega) \frac{e^{-i \cdot k \cdot R_{SR}}}{R_{SR}}$$

där $s_{TR}(\omega) = \frac{\text{spridande } E\text{-fält} \cdot R_0}{\text{inkommande } E\text{-fält}}$ och R_0 är enhetssfären. I mottagande antenn exciteras följande potential;

$$U_R(\omega) = E_R(\omega) \cdot h_R(\omega)$$

där h_R är mottagande antensns effektiva längd.

Enligt radarekvationen är mottagen effekt i det bistatiska fallet;

$$P_R = \frac{P_T \cdot \lambda^2 \cdot G_T \cdot G_R \cdot \sigma}{(4 \cdot \pi)^3 \cdot R_T^2 \cdot R_R^2 \cdot L}$$

där P_T är utsänd effekt, λ är våglängden, $G_k, k=\{T, R\}$ är antennförstärkningen, σ radarmålearean, $R_k, k=\{T, R\}$ avstånd mellan spridare och sändande respektive mottagande antenn och L är förlustfaktorn för systemet. Uttryckt i beteckningarna ovan med optimalt anpassad mottagare blir;

$$P_R = \frac{U_R^2}{4 \cdot Z_R} = \frac{P_T \cdot Z_T}{4 \cdot Z_R} \frac{|g_T(\omega)|^2 \cdot |h_R(\omega)|^2 \cdot |s_{TR}(\omega)|^2}{R_T^2 \cdot R_R^2 \cdot L}$$

Man kan visa att [3];

$$g(\omega) = \frac{i \cdot \omega \cdot \mu_0}{4 \cdot \pi \cdot Z} \cdot h(\omega)$$

där Z är impedansen i antennen. Då blir mottagen effekt;

$$P_R = P_T \frac{Z_R \cdot Z_T \lambda^2 \cdot |g_T|^2 \cdot |g_R|^2 \cdot |s_{TR}|^2}{c^2 \cdot \mu_0^2 \cdot R_T^2 \cdot R_R^2 \cdot L}$$

Utförligare beskrivning av bistatisk SAR ges i FOI-rapporten *Bistatic SAR Pilot Study: Simulations using FDTD* [4].

BEGRÄNSNINGAR

START-STOPP APPROXIMATION

Simulatorn implementerar i dagsläget start-stopp approximationen, dvs radarn förflyttar sig, stannar, samlar in data varefter radarn förflyttar sig till nästa position och stannar igen. En mer kontinuerlig modell är under framtagning.

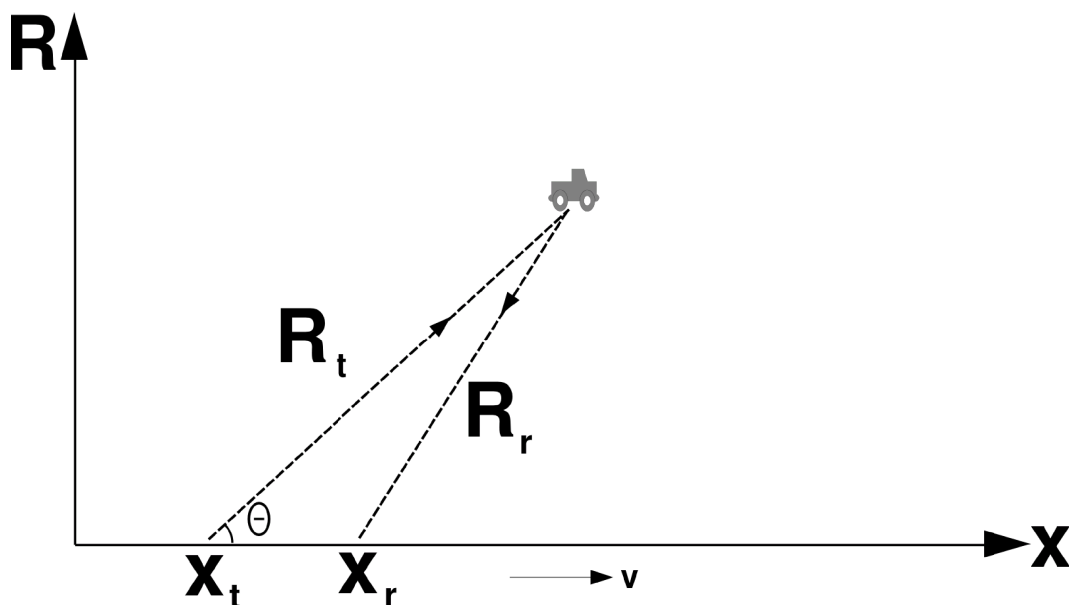


Illustration 2: Geometri för skattning av felet i start-stopp approximationen i det mono-statiska fallet. Hastigheten antas konstant mellan sändning och mottagning.

Antag att radarn rör sig med konstant hastighet v i en rakbana, felet i signalväg kan då skattas med formeln (för beteckningar se illustration 3);

$$\frac{R}{2R_t} = \frac{1}{(1 - v^2/c^2)} \left(1 - \frac{v}{c} \cos \theta\right) \quad \text{där} \quad R = R_t + R_r \quad \text{och} \quad c \text{ ljushastigheten.}$$

I fallet CARABAS med en platform som flyger med ca 127 m/s och med ett maximalt spaningsavstånd på 12 km blir felet i signalväg mindre än 6 mm.

Relativistiska effekter har ej tagits med i beräkningen.

SUPERPOSITION

Alla spridare i omgivningen modelleras var för sig och responsen ackumuleras till den av radarn mottagna signalen, dvs interaktionen mellan spridande objekt modelleras inte.

För VHF-bandet bedöms denna interaktion i normalfallet vara försumbar, se illustrationerna 12-6. Illustrationernaⁱ visar den elektriska fältstyrkan i fjärrfältet normerat med avståndet för tre exemplifierande fall. Man kan konstatera att med horisontellt infallande och spridande våg är överensstämmelsen god med superpositionsantagandet. I vertikalfallet däremot skiljer sig kurvorna åt betydligt mer men den totala energin över alla frekvenser är ändå i samma storleksordning varför det är främst sidloberna som påverkas, något som anses acceptabelt i sammanhanget. Träden står placerade enligt illustration 12 med trädet längst till

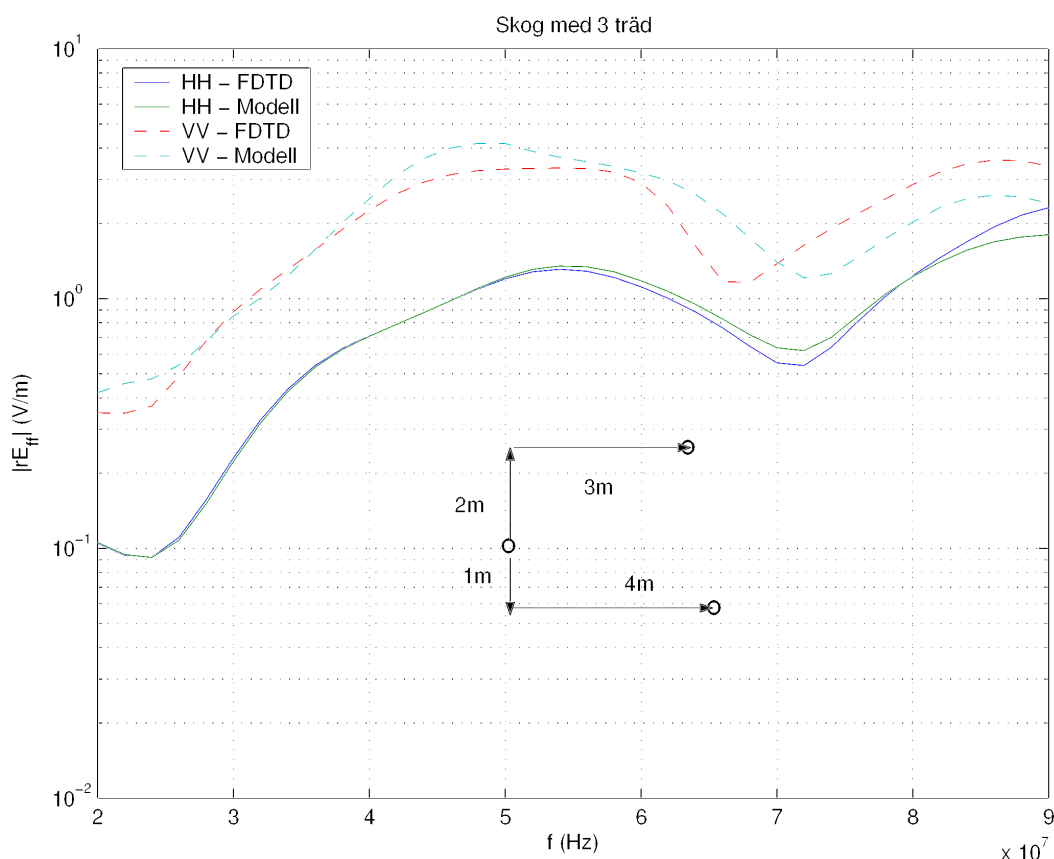


Illustration 3: Monostatiskt. Den inbördes skillnaden mellan de heldragna respektive streckade linjerna ger en upp-fattning om felets storlek vid super-position för de två polarisationskanalerna. Trädhöjden är 15 m. Infallsvinkeln är 60 grader. är 180 grader.

vänster i origo, ϕ räknas positiv från x-axeln som går åt höger, θ räknas positiv från z axeln mot xy-planet

ⁱ Trädmodellen som används i dessa illustrationer är framtagen av Torleif Martin vid Institutionen för mikrovågsteknik på FOI och är en parametriserad bistatisk trädmodell, ej att förväxla med den trädmodell som används i radarmodellen.

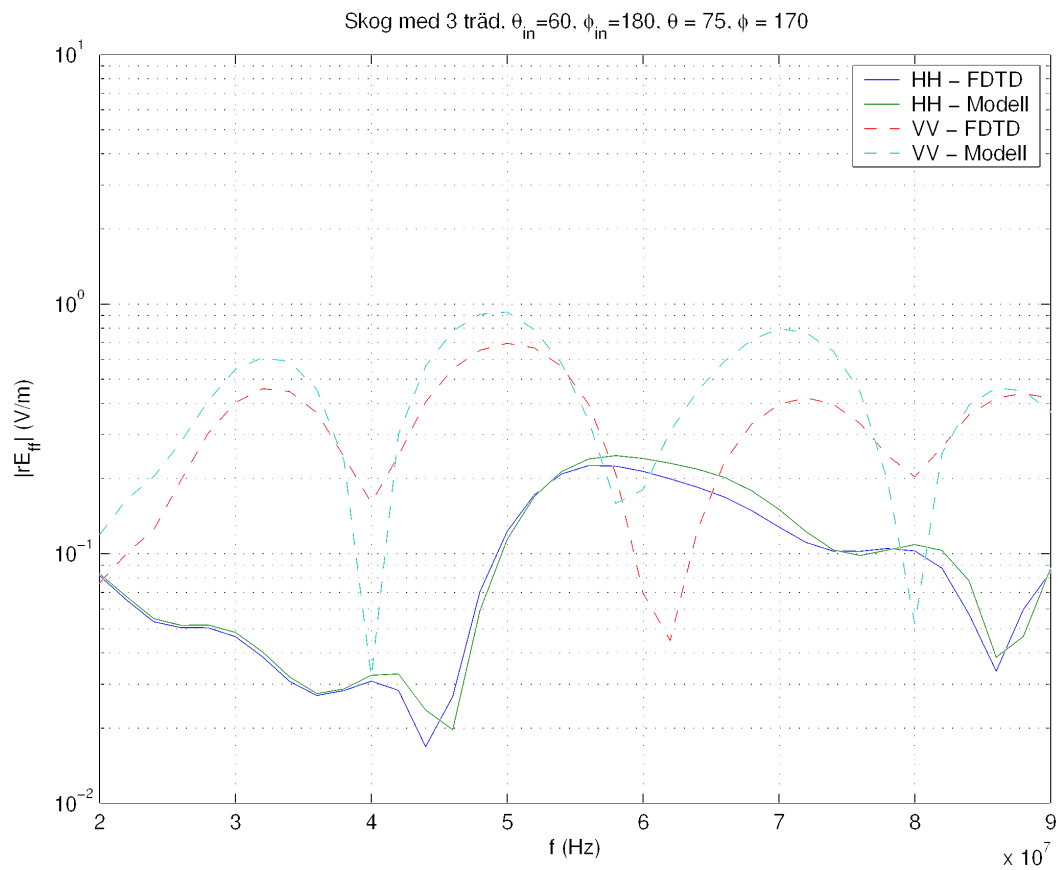


Illustration 4: Bistatiskt. Infallande vinkel är 60 grader, spridande vinkel är 75 grader. Skillnaden i azimuth är 10 grader. Träden står uppställda som i illustration 11.

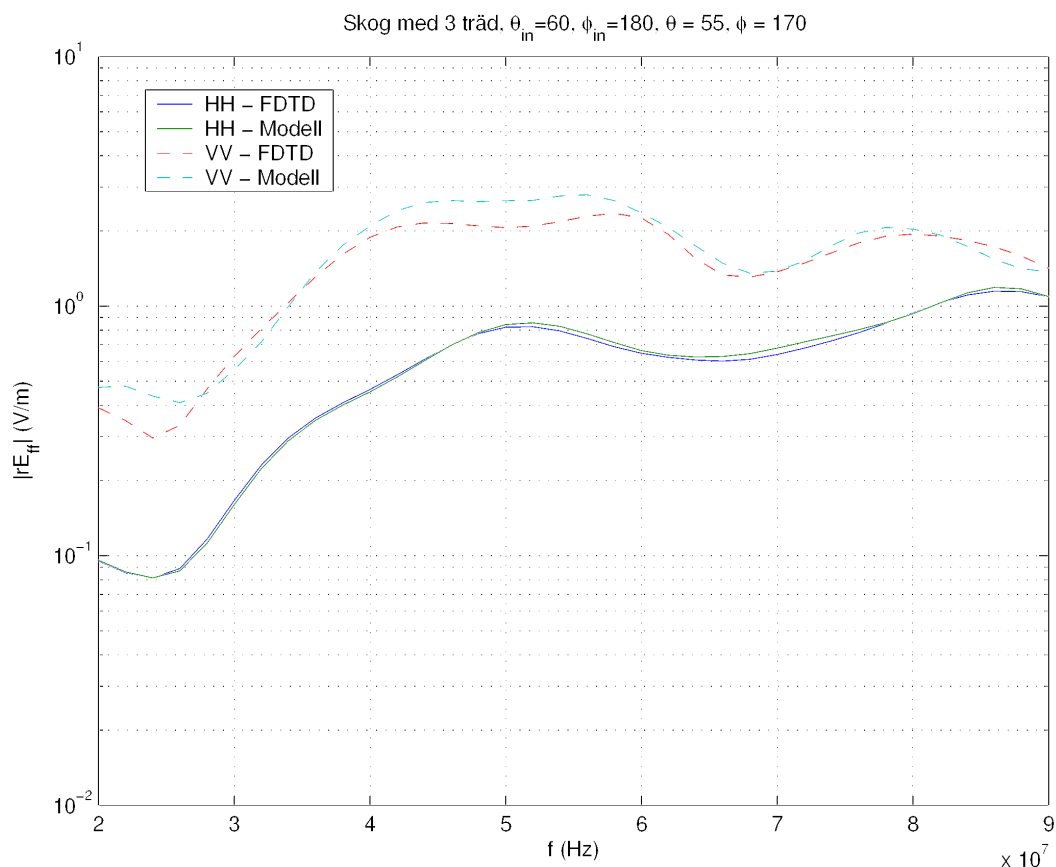


Illustration 5: Bistatiskt. Infallande vinkel är 60 grader, spridande vinkel är 55 grader. Skillnaden i azimuth är 10 grader. Träden står uppställda som i illustration 11.

En modell som tar hänsyn till interaktionen mellan spridarna bedöms bli mycket komplex och beräkningstung med hänsyn taget till den mängd spridare vi vill kunna simulera. Dessutom kräver en sådan modell ett annat angreppssätt då fjärrfältsapproximationen ej längre kan antas vara giltig. Illustration 7 visar interaktionen mellan tre spridare för en retur från en spridare S_0 .

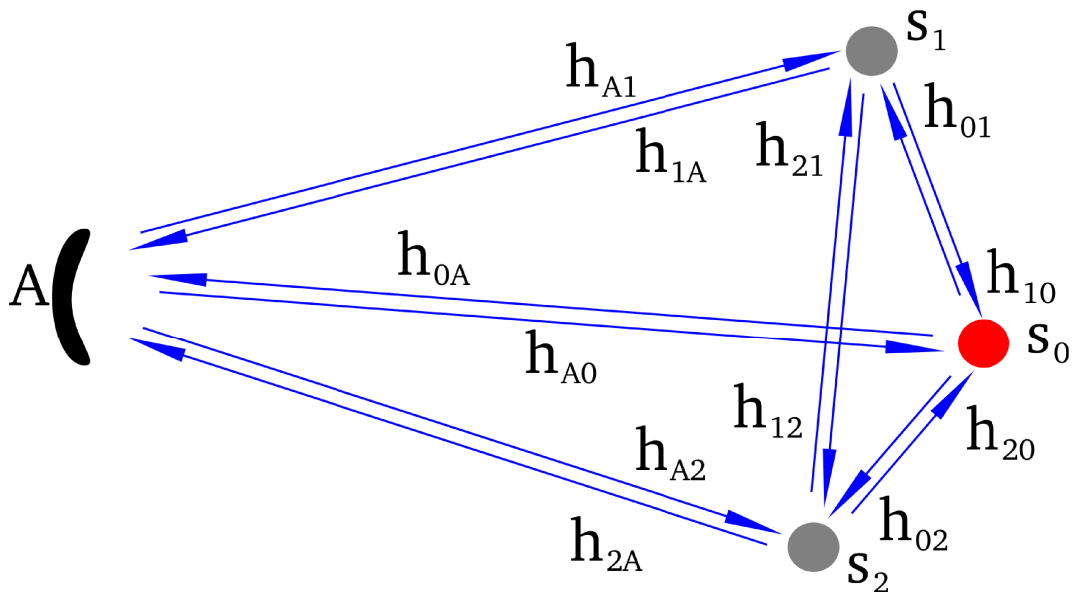


Illustration 6: Den mottagna responsen från en spridare i antennen. Spridaren interagerar med andra spridare i närheten. Varje överföringsfunktion interagerar även med marken.

Returen kan uttryckas som;

$$\begin{aligned}
 h_{s_0} &= h_1 + h_2 + h_3 + \dots \text{där} \\
 h_1 &= h_{A0} * h_{0A} \text{ (Enkelstuds)} \\
 h_2 &= h_{A1} * h_{10} * h_{0A} + h_{A2} * h_{20} * h_{0A} \text{ (Dubbelstuds)} \\
 h_3 &= h_{A1} * h_{12} * h_{20} * h_{0A} + h_{A2} * h_{21} * h_{10} * h_{0A} + \dots \\
 &+ h_{A0} * h_{01} * h_{10} * h_{0A} + h_{A0} * h_{02} * h_{20} * h_{0A} \text{ (Trippelstuds)} \\
 &\dots
 \end{aligned}$$

där varje överföringsfunktion h_{xy} även omfattar interaktion med marken. För varje spridare ökas dimensionaliteten ytterligare. Med superpositionsantagandet förenklas detta till;

$$h_{s_0} = h_1 = h_{A0} * h_{0A}$$

I de fall man misstänker att spridarna interagerar med varandra på ett sådant sätt att det inte är försumbart kan man undersöka detta med exempelvis FDTD-metoder. Dessa är dock extremt beräkningstunga varför det simulerade området torde bli mycket litet. Ett tänkbart scenario där FDTD-beräkningar kan krävas är då ett fordon står mycket nära en stor sten. Man kan då modellera denna uppställning separat och betrakta både fordon och sten som ett spridande objekt.

ÖVRIGA BEGRÄNSNINGAR

Direktsignaler, dvs en sändande antens signal som direkt eller via markstuds når en mottagande antenn, beaktas inte i nuläget.

Beräkningsbördan för modellen skalar linjärt med aperturlängd, antal spridare samt stråkbredd.

I dagsläget begränsar sig spridardatabasen till en trädmodell samt ett antal fordonstyper men fler typer av spridare är under framtagning, exempelvis kraftledning. Hus modelleras inte i dagsläget.

Modellen använder sig utav fjärrfältsapproximation.

SIMULATORNS ARBETSCYKEL

Radarmodellen är tänkt att på systemnivå simulera ett radarsystem, dvs det data modellen genererar ska kunna processas med de ordinarie signalbehandlingsalgoritmer CARABAS använder sig utav idag.

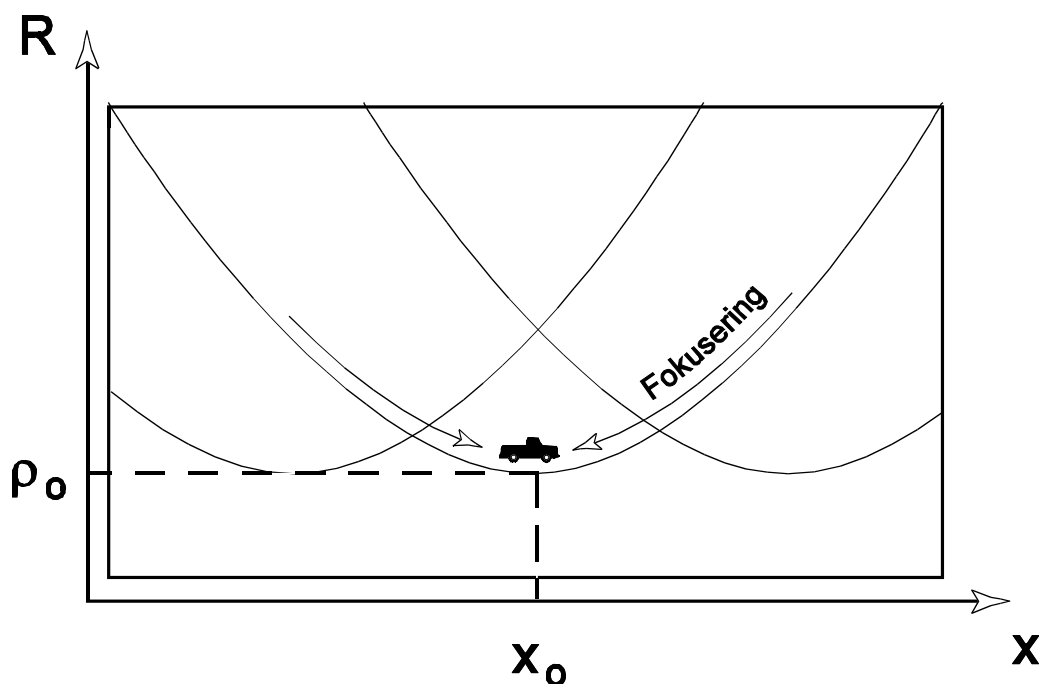


Illustration 7: Monostatiska radarekon från stillastående objekt bildar hyperbelkurvor i radardata förutsatt att flygplanet rör sig utefter en rakbana.

Modellen simulerar puls för puls så kallat hyperbeldata, se illustration 8. Radarekon från stillastående objekt bildar hyperbelkurvor i radardata förutsatt att flygplanet rör sig utefter en rakbana i x-led. I varje position x utsänds en puls och radarekot registreras som funktion av fördröjningstid. Kortaste avståndet ρ_0 mellan antenn och objekt fås i tvärsikten när $x = X_0$.

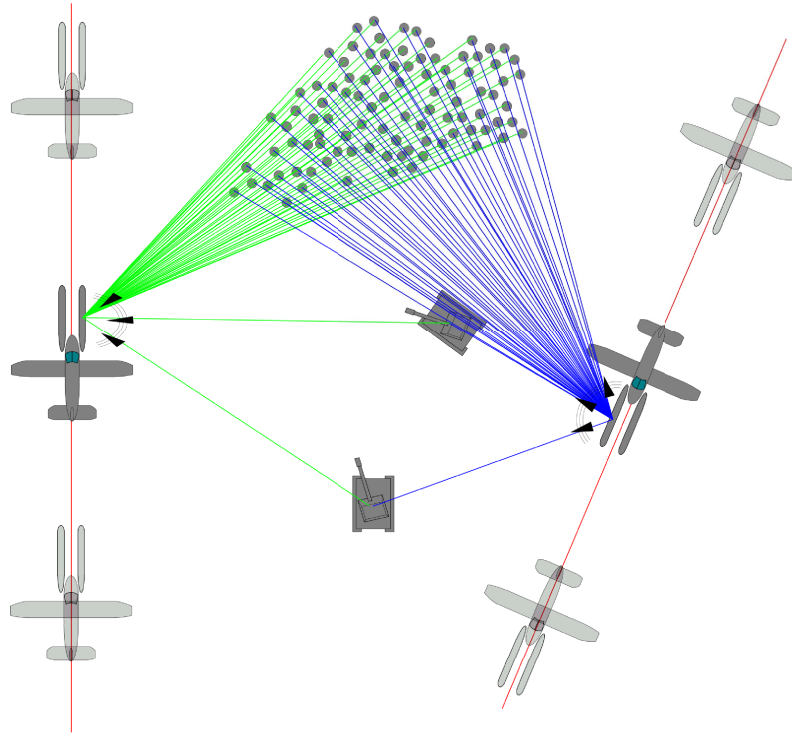


Illustration 8: Radar-simulatorens ackumulerar responsen från alla spridare puls för puls.

För varje puls ackumuleras radarekon från spridare enligt;

$$R(\omega) = \sum_{t=1}^T \sum_{i=1}^N S_{i,t,r}(\omega) e^{-i(R_t+R_r) \cdot k} * H_{t,i}(\omega) * H_{r,i}(\omega) * E_t(\omega)$$

där T är antalet sändande antenner, N är antalet spridare, $S_{i,t,r}(\omega)$ är spridaren i 's överföringsfunktion från sändande antenn till mottagande antenn, $H_{t,i}(\omega)$ och $H_{r,i}(\omega)$ är sändande respektive mottagande antensns överföringsfunktion i spridaren i 's riktning. $E_t(\omega)$ är utsänd signal från antenn t .

Antennen hämtar sitt fascentras globala position och orientering utifrån sensorbäarens position korrigerat för antennens momentana position och orientering relativt denna.

Radardata omvandlas till SAR-bild genom att i det monostatiska rakbanefallet transformera ("fokusera") hyperbelkurvorna till punkter.

Detta koordineras med hjälp av radarstyrenheten med dess radarstyrprogram.

Radarstyrprogrammet består av ett antal instruktioner som anger vilka antenner som är mottagande respektive sändande, vilka signalformer som skall användas samt i vilka filer impulssvaret skall sparas på disk.

Simulatorn arbetar enligt följande princip. Styrenheten hämtar nuvarande instruktion från radarstyrprogrammet. För varje mottagande antenn ackumuleras svaret från varje spridare som de utsända signalerna orsakat. Det ackumulerade svaret pulskomprimeras och sparas på disk. Därefter stegas programräknaren upp till nästa instruktion.

MÅLMODELLER

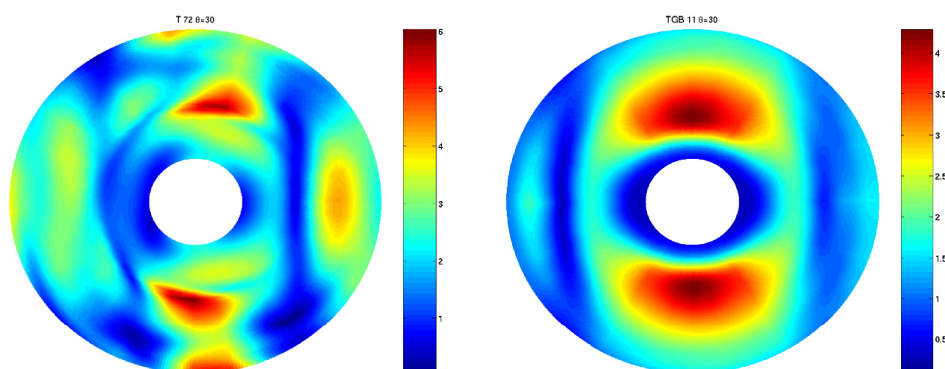
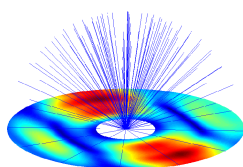


Illustration 9: Strålnings-diagram för två olika måltypen med fix infalls-vinkel

Illustration 10: Varje mål har en halvsfär av överförings-funktioner



Målmodellerna är ett antal små lokala modeller anpassade till frekvensbandet. Modellerna är beräknade med FDTD-metoder [5] utvecklade för frekvensbandet. Målmodellerna använder sig utav samma angreppssätt som radarantennerna, signaturerna förberäknas och lagras i binär form. Ett utsnitt ur en sådan binärfil visas i illustration 10 som återger den beräknade radarsignaturen ett varv runt målet för två olika fordonstyper, en T72 (stridsvagn) och en TGB 11 (terrängbil). Infallsvinkeln är 30 grader.

Varje mål modelleras som en punkt med en halvsfär av överföringsfunktioner (se illustration 11).

KLOTTERMODELL

I frekvensbandet 20-80 MHz består markklottret i huvudsak av den elektromagnetiska interaktionen med träd. En bistatisk trädmodell utvecklad på Chalmers [6] integrerades i radarsimulatorn. Det visade sig dock att koden ej var lämplig för massiva simuleringar varför en enkel monostatisk punktmålsmodell används istället (omarbetning av Chalmers-modellen till mer operativ kod pågår). Styrkan på punktmålet är proportionell mot trädhöjden upphöjt till fyra [7] [8].

IMPLEMENTATION

Radarsimulatore är indelad i tre huvuddelar, sensorbärare, omgivning samt radarantennerna och innehåller en gemensam styrenhet. Dessa tre delar intera-

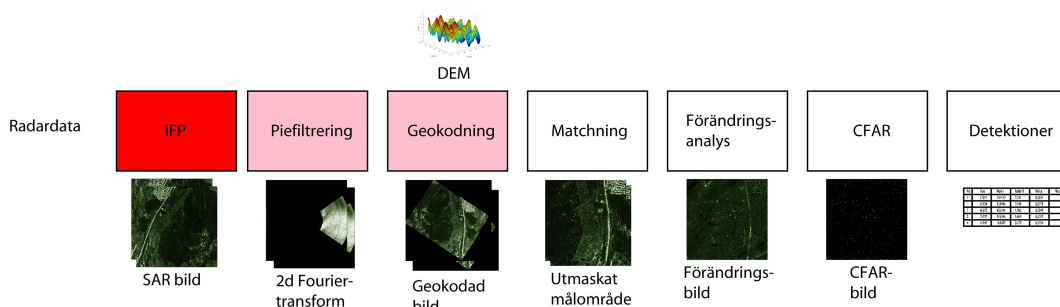


Illustration 11: Schematisk skiss över CARABAS processerings-kedja.

gerar med varandra och producerar en fil med radardata anpassad för den ordinarie processkedjan [9].

Utöver dessa tre delar som utgör kärnan av simulatore finns kopplingar till MOSART som använder sig utav HLAⁱ. HLA är distribuerat vilket gör att de olika delarna kan exekveras på olika datorer. Modellen är dock inte helt distribuerad då det mellan omgivningen och radarantennerna utbyts stora mängder information, vilket skulle lida av stora prestandaförluster om detta gick över HLA. Tack vare integrationen med MOSART fås ett grafiskt användargränssnitt till simulatore.

Hela simulatore är implementerad i C/C++.

ⁱ HLA, High Level Architecture, är en standard för distribuerade simuleringar

MÅL

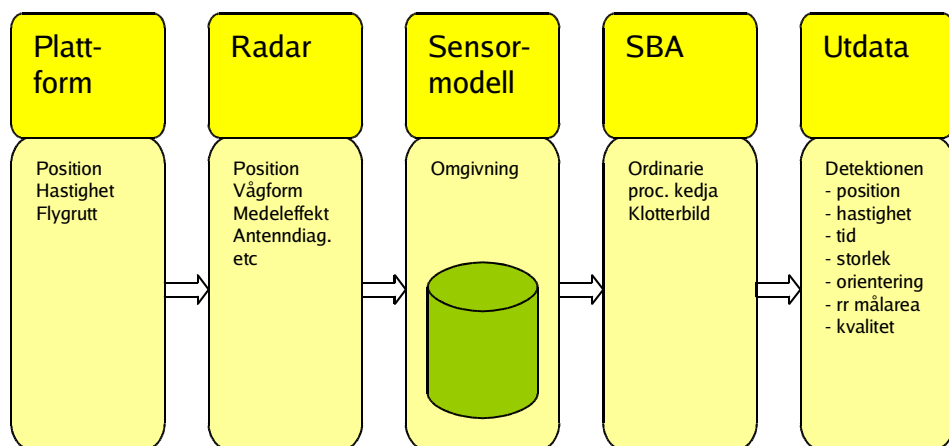


Illustration 12: Övergripande block-schema för ett mono-statiskt system. Figuren är tagen ur FOI-memo-808 [10].

Den drivande specifikationen för radarmodellen finns beskriven i FOI-memo-808 Kravspecifikation Radarmodell [10] och kan sammanfattas i krav på sensorbäraren (plattform), styrenheten för radarn (radar), sensormodellen, signalbehandlingsalgoritmerna (SBA) samt utdata, se illustration 13. Där ställs också krav på omvärldsmodellen.

Nedan sammanfattas och återkopplas de ställda kraven med nuvarande status för respektive modul. I Appendix A Kravlista – Återkoppling finns kravlistan från [10] med tillägg av nuvarande status för respektive krav.

Målet är att ta fram en modell som levererar tillräckligt realistiska sensordata för att kunna utvärdera sensorer i samverkan samt att kunna studera elektromagnetiska fenomen i SAR-bilder.

PLATTFORM

Plattformen skall representera en MALE (Medium Altitude Long Endurance) UAV (Unmanned Aerial Vehicle) som flyger enligt en brytpunktsbana. Brytpunktsbanan anges i en fil av XML-format eller enligt ett binärformat [1]. Dess uppgift är att vid varje given tidpunkt leverera en position och riktning där radarantennen befinner sig. Väder, vind eller plattformens verkliga dynamik modelleras inte. I det multistatiska fallet finns det flera plattformsmoduler.

Plattformen kan leverera sin position, orientering, hastighet, acceleration och vinkelhastighet som utdata. Dessa kan lagras i en XML-fil eller som ett binärformat.

Till skillnad från det ursprungliga kravet bär inte plattformen på information om vilka sensorer som är fästade på den utan sensorerna bär på informationen om på vilken plattform de sitter. Anledningen till denna förändring var att det blev naturligare att implementera modellen på det sättet.

I övrigt är kraven uppfyllda.

RADAR

Modulen beskriver de fysiska enheterna som radarn består utav såsom hur radarantennerna är orienterade, antenndiagram etc. Här kan man modellera dynamiken i den mekaniska konstruktionen. Varje radarsystem har en radarmodul som koordinerar signalerna.

Modulen kan läsa in radarparametrar såsom spaningsavstånd, stråkbredd, PRF, etc från ett XML-dokument.

Modulen innehåller även en signalbank som används vid genereringen av det pulskomprimerade radardatat.

Radarn består av två delar; styrenhet och styrprogram. Styrenheten koordinerar utsända och spridda signaler enligt instruktionerna i styrprogrammet och given mätgeometri, exempelvis spaningsavstånd och stråkbredd.

Vissa av de parametrar som enligt specifikationen skulle finnas här har flyttats till en speciell CARABAS-modul då de egentligen inte var radarparametrar utan snarare kunde hänföras till signalbehandlingsparameterar.

I övrigt uppfylls kraven.

SENSORMODELL

Sensormodellens uppgift är att generera hyperbeldata utifrån de parametrar som är givna från radarmodulen. Till sin hjälp har modellen en databas över topografin i omgivningen samt mål-/spridarmodeller. Sensormodellen tar hand om interaktionen mellan signaler och omvärld. FOI-memo-808 [10] beskriver kraven för en lågfrekvens radarsimulator upp till 1 GHz men under 2004 var målet endast att uppnå kraven för en CARABAS-modell varför nuvarande status är enligt följande;

- Modellen är endast giltig inom VHF bandet under 80 MHz
- Fortfarande Start-Stop approximation

1 GHz kraven är dock beaktade och implementationen är gjord så att den skall vara lätt att utöka.

Ursprungligen skulle denna modul ta hand om tidstämplingen av det genererade datat, denna funktion har flyttats över till styrenheten. Här skulle även antenndiagramen lagras men detta har flyttats till radarmodulen.

SIGNALBEHANDLINGSSALGORITM

Kravet var att den nuvarande ordinarie signalbehandlingsalgoritmen skulle klara att processa fram bilderna vilket den också gör. Signalbehandlingskedjan behöver dock startas för hand.

UTDATA

I väntan på en tydlig mottagare av utdata har inte mycket gjorts på detta område. De data som skulle levererats enligt specifikationen, dvs. position, hastighet, orientering, storlek, radarmålarea samt tidpunkt finns dock tillgängliga i den mån signalbehandlingsalgoritmenⁱ klarar leverera dem.

OMVÄRLDSMODELL

Från början var det tänkt att radarsimulatorens skulle arbeta mot lantmäteriets kartor och höjddatabas. Det framkom dock i ett tidigt skede att SEMARK kunde ta fram en mycket mer detaljerad och korrekt modell utgående från tidigare insamlade lasermätningar över ett delområde inom Kvarn. Modellarbetet har därför fokuserats till detta område.

HJÄLPKLASSER

För att underlätta programmeringen utvecklades ett antal hjälpklasser, då dessa tillsammans i någon mån beskriver vad simulatorens har för kapacitet sammanfattas de nedan.

KLASSEN `se::foi::follo::mvector`

En klass för vektorberäkningar. Klassen implementerar elementvis addition, subtraktion, multiplikation samt division. Den använder sig utav fftw 3.0.1 [11] för fouriertransform. Komplexkonjugat, skalärprodukt samt kryssprodukt är också implementerade. Det finns även operationer som arbetar på quaternionerⁱⁱ [12], vilka används för transformation mellan olika koordinatsystem. Klassen ärver från C++ standardvektorklass.

KLASSEN `se::foi::follo::mmatrix`

Klassen ärver från `se::foi::follo::mvector` men implementerar även transponering samt 2-dimensionell indexering, dessutom finns funktioner för att beräkna egenvärden för symmetriska reellvärda matriser.

ⁱ Signalbehandlingsalgoritmerna för CARABAS levererar för närvarande ingen skattning av orienteringen.

ⁱⁱ Quaternioner är en utvidgning av de komplexa talen $C = a + b \cdot i$ till $Q = a + b \cdot i + c \cdot j + D \cdot k$.

KLASSEN se::foi::radarmodel::exnihilo

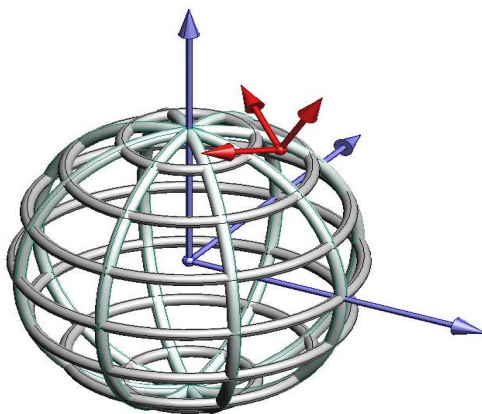


Illustration 13: Koordinatsystem

Exnihilo - ur intet, den mest grundläggande klassen i radarmodellen. Klassen representerar en punkt i universum med egenskaperna position, orientering, hastighet, acceleration samt vinkelhastighet. Positionen sätts i ett jordcentrerat – jordfixerat kartesiskt koordinatsystem. Alla uppdateringar av position, orientering samt hastighet beräknas i detta kartesiska koordinatsystem enligt;

$$\mathbf{x} = \begin{pmatrix} p \\ \dot{p} \\ \Phi \end{pmatrix}, \mathbf{u} = \begin{pmatrix} \dot{\mathbf{v}}_b \\ \dot{\Phi} \end{pmatrix}$$

$$\dot{\mathbf{x}} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \mathbf{x} + \begin{pmatrix} 0 & 0 \\ \Phi(\dot{\mathbf{v}}_b) & 0 \\ 0 & 1 \end{pmatrix} \mathbf{u}$$

där

$$\mathbf{x} = \begin{pmatrix} x & y & z & \dot{x}_b & \dot{y}_b & \dot{z}_b & \omega_{bx} & \omega_{by} & \omega_{bz} \end{pmatrix}^T$$

$\Phi(\mathbf{v})$ transformerar en vektor $\mathbf{v}$ från objektets koordinatsystem (rött) till det globala koordinatsystemet (blått) i illustration 14.

Ett objekt av denna typ kan även länkas samman med ett annat objekt av samma typ. Egenskaperna sätts då relativt det andra objektet. Detta används till exempel då antennerna sätts samman på sensorplattformen.

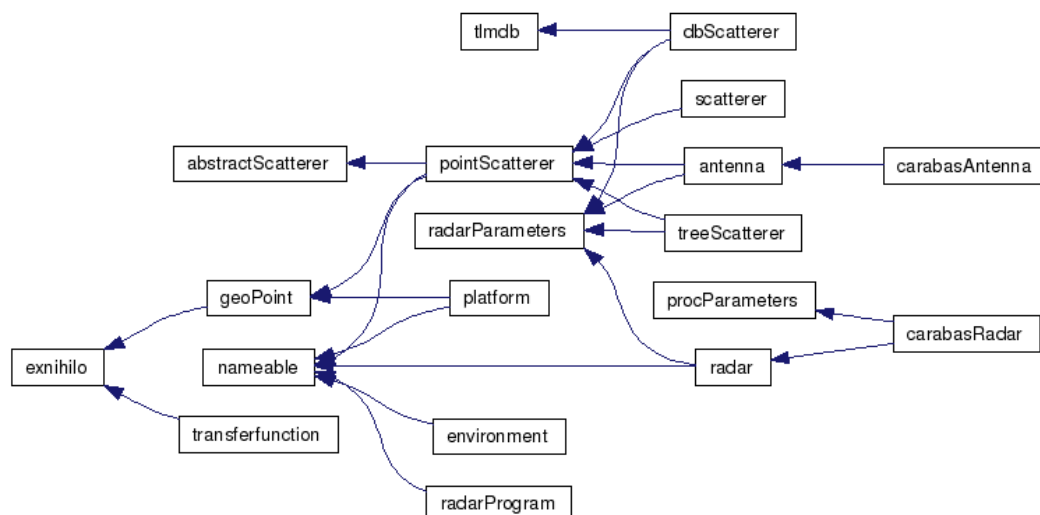


Illustration 14: Klassdiagram för radar-modellen. Klassen exnihilos roll i radarmodellen ses vara mycket central.

KLASSEN `se::foi::radarmodel::transferfunction`

En komplex polariserad överföringsfunktion. Denna tar hand om och ackumulerar signaler och överföringsfunktioner. Klassen tar även hänsyn till att överföringsfunktionerna har olika orientering genom att projicera inkommande fälts axlar på sina egna, se illustration 16 och 17.

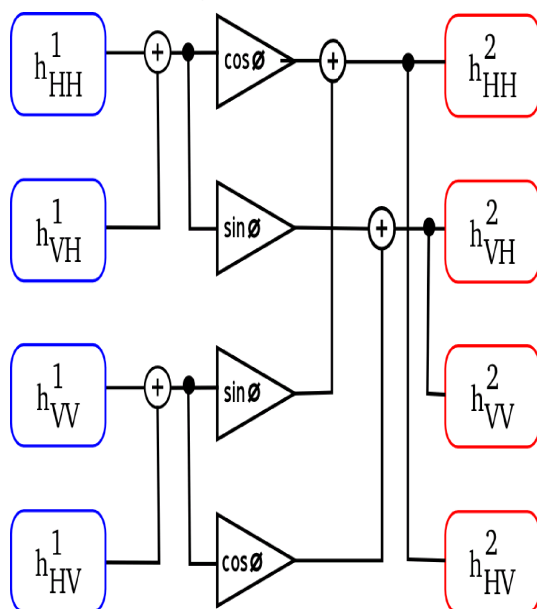


Illustration 15: Blockshema över en polariserad överföringsfunktion

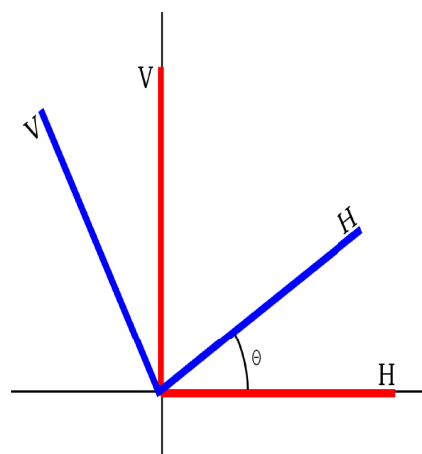


Illustration 16: Blå linjer är inkommande fälts axlar, röda är egna axlar.

Två överföringsfunktioner f och g ackumuleras (fållas) till h med multiplikations operatoren enligt;

$$\begin{aligned}
H_{HH} &= G_{HH} \cdot ((F_{HH} + F_{VH}) \cos \phi + (F_{VV} + F_{HV}) \sin \phi) \\
H_{VH} &= G_{VH} \cdot ((F_{HH} + F_{VH}) \sin \phi + (F_{VV} + F_{HV}) \cos \phi) \\
H_{VV} &= G_{VV} \cdot ((F_{HH} + F_{VH}) \sin \phi + (F_{VV} + F_{HV}) \cos \phi) \\
H_{HV} &= G_{HV} \cdot ((F_{HH} + F_{VH}) \cos \phi + (F_{VV} + F_{HV}) \sin \phi)
\end{aligned}$$

där F_{XX} , G_{XX} och H_{XX} är f , g och h 's respektive polarisationskomponents fouriertransform.

SENSORBÄRARE

Sensorbäraren kan ges parametrarna position, hastighet, orientering, acceleration samt vinkelacceleration. Den kan hantera de geografiska referenssystemetn wgs84 samt rikets nät. Via en XML-fil kan en brytpunktsbana läsas in. Varje brytpunkt innehåller en geografisk koordinat samt en tidpunkt då plattformen skall befinna sig där. Plattformen ärver från klassen *exnihilo* via en klass, *geoPoint*, som tar hand om koordinattransformationer mellan exempelvis wgs84 till det globala kartesiska koordinatsystemet. Ett objekt tillhörande denna klass kan även logga sina inre tillstånd i ett binärformat [13] kompatibelt med den navigationsutrustning som används av CARABAS/LORA idag.

RADAR (STYRENHET)

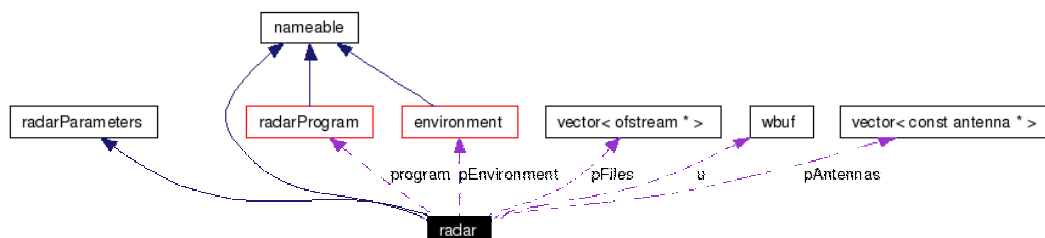


Illustration 17: Klassdiagram för radarn.

Styrenheten består av pekare (referenser till) antenner, miljö samt ett radarstyrprogram med tillhörande signalbank. Enheten innehåller också information om PRI, spaningsavstånd och sampelhastighet. Illustration 18 visar styrenhetens klassdiagram.

Signalbanken innehåller de signaler som radarsystemet sänder ut och styrprogrammet koordinerar vilken antenn som skall sända enligt instruktionerna i styrprogrammet. En instruktion i radarprogrammet består av följande;

- Vilka sändande antenner som skall användas
- Vilken signal respektive sändande antenn skall använda
- I vilken fil navigationsdata från respektive sändande antenn skall lagras

- Vilka antenner som är mottagande
- I vilken fil radarresponsen från respektive mottagande antenn skall lagras
- I vilken fil navigationsdata från respektive mottagande antenn skall lagras

Antennernas antenndiagram förberäknas eller uppmäts och lagras i en binärfil unik för varje enskild antenntyp. Binärfilen innehåller antennens överföringsfunktion samplad i en sfär kring fascentrum. Enklare typer av antenner finns också. I dagsläget används en isotropisk antennmodell då mätdata från CARABAS-antennen ännu ej konverterats till binärfilsformatet. Det är också möjligt att implementera matematiska modeller för antennerna.

INTEGRATION AV CARABAS-MODELLEN TILL MOSART

Kopplingen av CARABAS-modellen till MOSART är gjord i C++ och använder sig av MOSART:s API. MOSART är en implementation av simuleringsstandarden HLA [14].

Det finns tre federater [2] i CARABAS/MOSART integrationen. De kallas plattformsfederat, radarfederat och processeringsfederat. Plattformsfederaten kontrollerar flygplanets och antennernas rörelser. Radarfederaten kontrollerar själva radarsimulatoren. Processeringsfederaten tar emot information från radarfe-

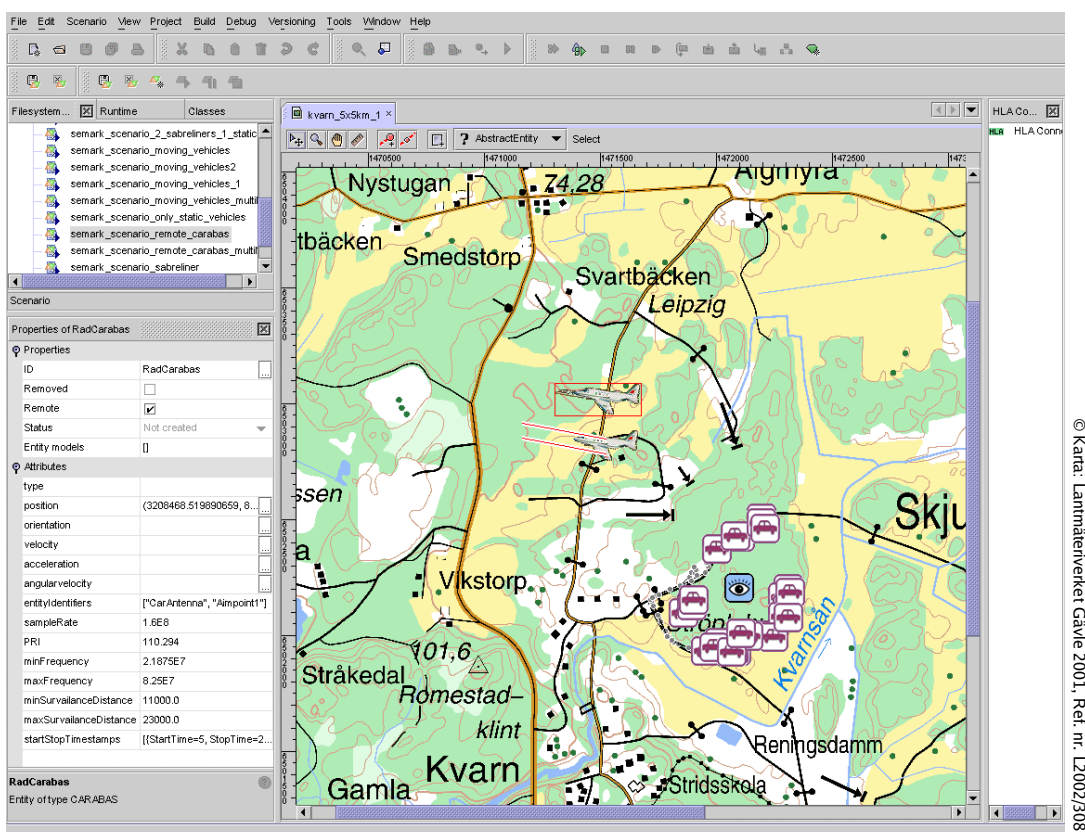


Illustration 18: Användargränssnitt
deraten och processerar fram radarbilder.

Scenariot för en simulering skapas i programmet NetScene som är en del av MOSART, se illustration 19. Användaren ställer ut fordon, flygplan och andra objekt i en scen och matar in de parametrar som behövs. Kontrollprogrammet för hela simuleringen är också ett MOSART-program och heter *HLA Federation Manager*. Det används för att koppla ihop alla HLA federater och sedan kontrollera simuleringsförloppet.

PLATTFORMSFEDERATEN

Plattformsfederaten sköter interaktionen mellan flygplansmodellen i CARABAS-simulatorens och flygplansklassen i HLA/MOSART som heter SabreLiner. Antennmodellen i CARABAS-simulatorens kopplas också till HLA/MOSART klassen RadarAntenna.

Plattformsfederaten är en server i den meningen att den kopplas upp mot en RTIⁱ och sedan lyssnar på *remote create* interaktioner som skickas ut från en scenarioreditor eller liknande. När en *remote create* interaktion tas emot för något av de HLA-objekt som federaten hanterar så skapar federaten HLA-objektet samt ett likvärdigt CARABAS-simulatorobjekt. I detta fall är det NetScene-federaten som skickar *remote create* interaktioner till plattformsfederaten som därefter skapar HLA- och simulator- plattformsobjektet. Dessa objekt kopplas ihop så att en uppdatering i simulatorobjektet medför samma uppdatering i HLA-objektet.

Då ett SabreLiner objekt skapas med hjälp av en *remote create* interaktion så finns det en parameter som heter *XmlData* som ska sättas. Den klassas som internattribut av HLA, dvs. attributet är inte ett HLA attribut utan är en parameter som bara plattformsfederaten hanterar. *XmlData* parametern innehåller information om den flygrutt plattformen skall flyga.

RADARFEDERATEN

Radarfederaten sköter CARABAS-simulatorns pulsgenerering genom att hantera simulatorns CARABAS-objekt samt HLA/MOSART klassen CARABAS. Även integrationen av olika fordonstyper från HLA/MOSART in i CARABAS-simulatorns miljö hanteras.

Radarfederaten är en server på samma sätt som plattformsfederaten är. Den tar emot *remote create* interaktioner och skapar CARABAS-HLA-objekt samt CARABAS-simulatorobjekt. Genom att prenumerera på *SabreLiner*, *RadarAntenna* och *aimpoint* objekt från HLA så fås information om plattformens position, hastighet, riktning. Även information om antennerna och positionen på en aimpoint tas emot.

Varje radarobjekt har start och stopptider som bestämmer när radarregistreringen ska startas och när den ska stoppas. Då radarn stoppas skickas information ut om var de skapade pulsfilerna ligger. Informationen skickas som en HLA interaktion som kan tas emot av en federat som är intresserad av datat. HLA interaktionen heter *CARABASLevel1DataAvailable* och innehåller filnamn för puls-komprimerat data, navigeringsdata samt parameterfil med processeringsinformation.

ⁱ Real Time Infrastructure – Till exempel mjukvaran *pitch* utvecklad för Försvarmakten.

PROCESSERINGSFEDERATEN

Processeringsfederaten sköter själva genereringen av SAR-bilder från puls-komprimerat data. Då en HLA interaktion som innehåller pulsfilinformation tas emot från radarfederaten så startas en bildgenereringsprocess. Bilden som skapas är en så kallad slant-range bild ej geokodad, se illustration . Bilden skickas i sin tur ut som en HLA interaktion som andra federater kan ta emot.

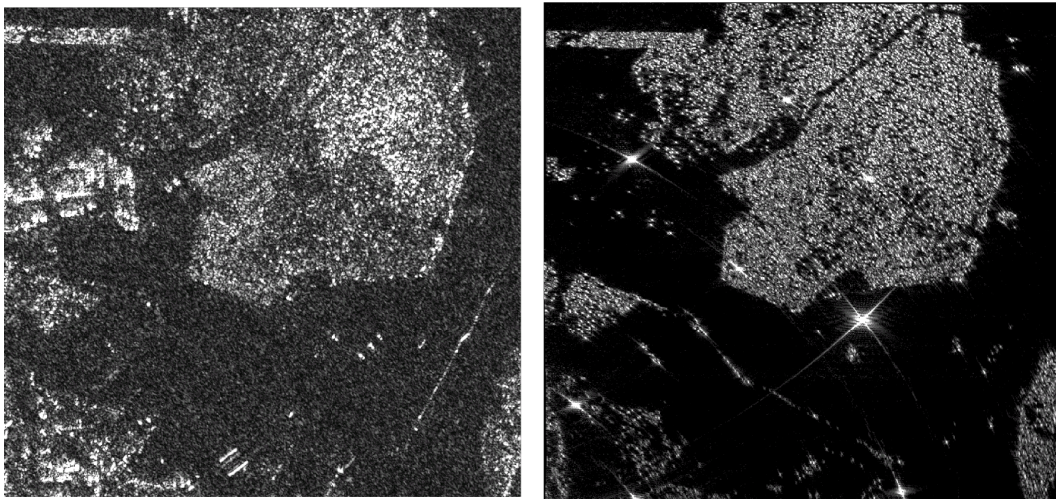


Illustration 19: *Till vänster*; En CARABAS-bild över Kvarn *Till höger*; Motsvarande simulerad bild. Bilderna har geokodats i efterhand för att jämförelsen skall bli enklare.

APPENDIX A KRAVLISTA – ÅTERKOPPLING

Listan kommer från FOI-memo-808 [10] där kraven på radarsimulatorens sammanställdes innan implementeringen av den påbörjades. Listan är indelad i krav på plattform, radar samt sensormodell.

PLATTFORM

1. Giltig inom Sverige under 20000 m höjd – **Ja**
2. Simulera en flygning enligt en brytpunktsbana – **Ja**
3. Läs in en brytpunktsbana med georefererade koordinater från ett XML-dokument – **Ja**
4. Spara navigationsdata i ett binärformat kompatibelt med SBA:n – **Ja**

R_{ADAR}

5. Läs in radarparametrar från ett XML-dokument – **Ja**
6. Läs in styrparametrar från ett XML-dokument – **Ja**

SENSORMODELL

7. Anpassad för frekvensbandet 20 – 800 MHz – **Endast 20 – 80 MHz**
8. Läsa in antenndiagrammen från en binärfil – **Ja**
9. Läsa från omvärldsdatenⁱ vilken innehåller information om – **Ja**
 - a. Topografin – **Ja**
 - b. Träd – **Ja**
 - c. Stolpar – **Finns ej**
 - d. Kabel- och teleledningar – **Ja**
 - e. Stängsel – **Finns ej**
 - f. Byggnader – **Nej**
 - g. Diskontinuiteter i marktopografin – **Nej**
 - i. Stora stenar av typen flyttblock – **Finns ej**
 - ii. Stora diken – **Finns ej**
 - h. Överhörningar – **Ja**
10. Läsa radarsignaturer från en måldatabas anpassad för ett givet scenario – **Ja**
11. Läsa radarsignaturer från en databas över byggnader anpassad för ett givet scenario – **Nej**
12. Beakta störande sändare – **Ja**

ⁱ Det är inte säkert att omvärldsdaten innehåller all information

- [1] Peter Follo (2005) *Reference manual Radar model 2004*, FOI-D--0212--SE
- [2] Törne A, Tobias Horney, Mikael Brännström, Pontus Hörling, Lars Tydén (2004) *Projektutnyttjande MOSART*, FOI-memo-1128
- [3] Warren L Stutzman, Gary A. Thiele (1981) *Antenna theory and design*, John Wile & Sons, ISBN 0-471-04458-X
- [4] Torleif Martin, Lars Ulander (2003) *Bistatic SAR pilot study: Simulations using FDTD*, FOI-R—0839--SE, ISSN 1650-1942
- [5] Torleif Martin, Lars M.H. Ulander (2000) *VHF/UHF synthetic aperture radar image simulations of a vehicle using FDTD*, FOA-R—00-01765-408—SE, ISSN 1104-9154
- [6] Gary Smith (2000) *Radar remote sensing of forests using CARABAS and ERS. Doktorsavhandling*, Chalmers, ISBN 91-7197-982-4
- [7] Gary Smith, Lars M.H. Ulander (2000) *A Model Relating VHF-Band Backscatter to Forest Stem Volume*, *IEEE Trans. Geosci. Remote Sensing*, Vol. 38, No. 2, pp. 728-740
- [8] Gary Smith, Lars M.H. Ulander, Xianyun Luo (in press) *Low Frequency Scattering of Coniferous Forests on Sloping Terrain*, *IEEE Trans. Geosci. Remote Sensing*
- [9] Lars M.H. Ulander, Björn Flood, Peter Follo, Per-Olov Frölind, Anders Gustavsson, Tommy Jonsson, Björn Larsson, Mikael Lundberg, William Pierson, Gunnar Stenström (2003) *Flight Campaign Vidse 2002 CARABAS-II Change Detection Analysis*, FOI-R--1001--SE, ISSN 1650-1942
- [10] Peter Follo, Lars M.H. Ulander (2004) *Kravspecifikation radarmodell LORA*, FOI-memo-808
- [11] <http://www.fftw.org>
- [12] Daniel Murdin (2000) *Standby Attitude Estimation, Position Location and Navigation Symposium*, IEEE 2000, pp. 429-436
- [13] Applanix (2000), *POS/AV V4 Ethernet & Disk Logging ICD (External Release)*, PUBS-ICD-000002
- [14] IEEE(2002), *IEEE Standard for Distributed Interactive Simulation – Application Protocols*, IEEE std 1278.1-1995(R2002)