

Sten-Åke Nilsson, Henrik Allberg, Mikael Lundin, Niklas Wallin, Martin Castor

Datorgenererade styrkor med mänskligt beteende

CGF-verksamhet vid FOI 2004

Sten-Åke Nilsson, Henrik Allberg, Mikael Lundin, Niklas Wallin, Martin Castor

Datorgenererade styrkor med mänskligt beteende

CGF-verksamhet vid FOI 2004

Utgivare Totalförsvarets Forskningsinstitut - FOI Systemteknik 172 90 Stockholm	Rapportnummer, ISRN FOI-R--1489--SE	Klassificering Användarrapport
	Forskningsområde 2. Operationsanalys, modellering och simulering	
	Månad, år November 2004	Projektnummer E6026
	Delområde 21 Modellering och simulering	
	Delområde 2	
Författare/redaktör Sten-Åke Nilsson Henrik Allberg Mikael Lundin Niklas Wallin Martin Castor	Projektledare Sten-Åke Nilsson	
	Godkänd av Monica Dahlén	
	Uppdragsgivare/kundbeteckning FM	
	Tekniskt och/eller vetenskapligt ansvarig	
Rapportens titel Datorgenererade styrkor med mänskligt beteende CGF-verksamhet vid FOI 2004		
Sammanfattning (högst 200 ord) FOI har sedan 2001 på uppdrag av Forsvarsmakten genomfört studier inom området datorgenererade aktörer. Syftet med projektet är att skapa kunskap om och en grund för utveckling av CGF och för uppbyggnad av ett komponentbaserat bibliotek med representativa kognitiva och beteendevetenskapliga modeller av aktörer vilka kan användas i olika simuleringsmiljöer. Tyngdpunkten för årets arbete har legat på fortsatt utveckling av biblioteksstruktur och uppbyggnad, framtagning av metodik för mätning av naturliga mänskliga rörelsescheman med åtföljande modellering och visualisering, genomgång med definition av krav på insamling av andra data (psykofysiologiska, beteendedata och andra kognitiva) som kan användas i CGF-modeller, deltagande i nationellt och internationellt samarbete och en i övrigt omfattande kontakt- och presentationsverksamhet.		
Nyckelord M&S, CGF, datorgenererade styrkor, simulering, kognitiv modellering HBR, MSI		
Övriga bibliografiska uppgifter	Språk Svenska	
ISSN 1650-1942	Antal sidor: 64 s.	
Distribution enligt missiv	Pris: Enligt prislista	

Issuing organization FOI – Swedish Defence Research Agency Systems Technology SE-172 90 Stockholm	Report number, ISRN FOI-R--1489--SE	Report type User report
	Programme Areas 2. Operational Research, Modelling and Simulation	
	Month year November 2004	Project no. E6026
	Subcategories 21 Modelling and Simulation	
	Subcategories 2	
Author/s (editor/s) Sten-Åke Nilsson Henrik Allberg Mikael Lundin Niklas Wallin Martin Castor	Project manager Sten-Åke Nilsson	
	Approved by	
	Sponsoring agency Swedish Armed Forces	
	Scientifically and technically responsible	
Report title (In translation) Computer Generated Forces and human behavior models at FOI 2004		
Abstract (not more than 200 words) <p>The Swedish Armed Forces has sponsored a study of CGF development since 2001, conducted at FOI. The main purpose of the project is to construct a framework to support CGF development and to maintain a component based library of HBR models to be used in existing and future simulations.</p> <p>The main effort 2004 within the project has been continued development of library structures, development of methods and techniques how to measure and deal with human behavior schemes, exploration of which claims one should do about collecting human factors metrics such as psychophysiological, mental, strengths and stressors. The project is a member of a recently started international working group on Human Behavior Representation in Constructive Simulations. The project also presented an interesting paper at the 2004 years conference, BRIMS in the USA.</p>		
Keywords M&S,CGF, Computer generated forces, simulation, cognitive modelling,HBR,Human Factors		
Further bibliographic information	Language Swedish	
ISSN 1650-1942	Pages 64 p.	
	Price acc. to pricelist	

Datorgenererade styrkor med mänskligt beteende CGF-verksamhet vid FOI 2004

Årsrapport

2004-12-15

Sammanfattning

FOI har sedan 2001 på uppdrag av försvarsmakten genomfört studier inom området datorgenererade aktörer. Syftet med projektet är att skapa kunskap om och en grund för utveckling av CGF och för uppbyggnad av ett komponentbaserat bibliotek med representativa kognitiva och beteendevetenskapliga modeller av aktörer vilka kan användas i olika simuleringsmiljöer.

Tyngdpunkten för årets arbete har legat dels på fortsatt utveckling av biblioteksstruktur och uppbyggnad, framtagning av metodik för mätning av naturliga mänskliga rörelsescheman med åtföljande modellering och visualisering, genomgång med definition av krav på insamling av andra data (psykofysiologiska, beteendedata och andra kognitiva) som kan användas i CGF-modeller, deltagande i nationellt och internationellt samarbete och en i övrigt omfattande kontakt- och presentationsverksamhet.

Innehållsförteckning

Projektet	9
Mål	9
Nytta för FM.....	9
Följande aktiviteter planerades för året:	10
Genomförande	10
Resultat från årets verksamhet	13
Utveckling av biblioteksuppbyggnad av CGFer	13
Kunskapsinhämtning och uppgiftsanalys vid SIB-övning	13
Utveckling av teknik och metodik för insamling av rörelsedata och visualisering av dessa med hjälp av 3D-teknik.....	15
Krav på insamling av mätdata att användas för CGF-modeller	15
Bidrag till BRIMS-konferensen	16
Reseberättelse från BRIMS-konferensen	17
Samarbete i en NATO/RTO grupp inom området	17
Nationellt samarbete och övriga presentationer	18
Summering av årets verksamhet inom CGF-projektet	19
Referenser.....	19
Bilaga 1, Motion caption.....	21
Bilaga 2, Krav på beteendedata till CGF-modeller	29
1. Bakgrund	31
2. Problemen.....	33
3. Mall för beteendedata för CGF-utveckling	37
4. Diskussion	38
5. Referenser.....	39
6. Appendix 1	41
Bilaga 3, Konferensbidrag.....	51
Bilaga 4, Reserapport BRIMS 2004.....	60

Förkortningar

ACT-R	Adaptive Control of Thought Rules
AI	Artificiell Intelligens
AI.Implant	AI-programvara
BRIMS	Behavior Representation in Modelling and Simulation
CGF	Computer Generated Forces
DMSO	Defence Modeling and Simulation Office.
EEG	Elektroencefalografi, Hjärnvågsmätning
FLSC	Flygvapnets LuftstridsSimuleringCenter
HBR	Human Behavior Representations
HFM	Human Factors and Medicine (NATO RTO Panel)
HLA	High Level Architecture
HTA	Hierarchical Task Analysis
M&S	Modellering och Simulering
MSI	Människa System Interaktion
NATO RTO	NATO Research and Technology Organisation
PMF's	Performance Moderator Functions
SAS	Studies, Analysis and Simulation (NATO RTO Panel)
SIB	Strid i bebyggelse
Soar	State, operator and result
UML	Unified Modeling Language

Projektet

Vid FOI, och Institutionen för Systemmodellering, startade under 2001 ett förstudieprojekt vid namnet ”Datorgenererade Styrkor, metoder och möjligheter”. Projektet, som utfördes på uppdrag från FM utgick ifrån FOA:s mångåriga verksamhet och erfarenhet inom beteendevetenskap, distribuerad simulering och konstruktion av komplexa modeller samt Försvarens behov vad gäller modellering av mänskligt beteende och datorgenererade styrkor.

Datorgenererade styrkor (Computer Generated Forces, CGF) är en term som används för att beskriva modeller och simuleringar som erbjuder representationer av militära styrkor, och har traditionellt använts för att befolka simuleringar med objekt och aktörer för att skapa en realistisk miljö.

Verksamheten under föregående år finns sammanfattad i årsrapporter, *Datorgenererade styrkor – Metoder och möjligheter*, Användarrapport FOI-R—0688—SE, December 2002 och *Datorgenererade styrkor med mänskliga egenskaper*, Metodrapport FOI-R—1076—SE, December 2003. Verksamheten under år 2004 är beskriven i föreliggande årsrapport.

Mål

Det långsiktiga och övergripande målet med projektet är att studera möjligheter och förutsättningar för CGF-utveckling, samt att bidra till att bygga upp ett försvarsmaktsgemensamt referensbibliotek med modeller av datorgenererade styrkor på olika nivåer. Samtidigt finns verksamhetsmålet att skapa gedigen kunskap inom området datorgenererade styrkor för att bl. a. kunna stödja modellerings- och simuleringens verksamheten inom Försvarens makt, FOI och industrin.

Nytta för FM

Genom att använda datorgenererade styrkor kan man minska kostnaden för både organisation, ledning och dirigering av enheter som behövs för en effektiv och meningsfull simulering. Det skulle ur försvarets synvinkel vara värdefullt att ha ett bibliotek av generiska styrkor som representerar olika militära och icke-militära enheter på olika aggregationsnivåer. Dessa skulle kunna användas som ”plug’n’play” moduler, som via ett givet gränssnitt kan kopplas till de simuleringar där de behövs. Träning, utbildning och analys är exempel på några användningsområden.

Området ”Human Behavior Representation” är tydligt utpekad som ett centralt område i DMSO:s Master Plan (Objective 4 i DMSO, 1995) och har blivit så omfattande att det årligen arrangeras speciella konferenser¹ inom ämnet. Följaktligen är det av största vikt att Sverige besitter kompetens inom detta område, dels för att kunna tillgodose våra egna behov och dels för att datorgenererade styrkor är svåra att komma åt inom den kommersiella marknaden, även om en del grundarkitekturer kan vara gratis. Om man t.ex. betraktar en applikation som TacAir-Soar,

¹ Conference on Behavior Representations in Modelling and Simulation, BRIMS (USA)

som innehåller 8000 regler för hur AI-kontrollerade flygplan skall agera enligt doktrin, är det uppenbart att sannolikheten att få tillgång till källkod är väldigt liten. Detta tyder på att det mesta inom detta område måste byggas upp inom landet och vi behöver ett försprång innan behoven blir akuta.

Andra intressanta pågående och kommande militära forskningsprojekt som skulle kunna vara betjänta av CGF:er, för att simulera militära och icke-militära enheter på olika nivåer, är LedsystX, Nätverkstrid, Strid i bebyggelse och IT-Krig/Framtida ledningssystem.

Följande aktiviteter planerades för året:

- Studera hur människors faktiska beteende kan utvinnas ur loggar från simuleringsanläggningar och överförs till modeller.
- Analysera vilka krav som skall ställas på data som kan samlas in vid experiment, simuleringar och från verkligheten, för att dessa data skall vara användbara vid bl. a utveckling av CGF-modeller.
- Påbörja design av en ”Psykofysiologisk beteendemotor” (dvs en ackrediterad databas innehållande olika typer fysiska, psykiska och prestationsdata etc över mänskligt beteende för användning i olika visuella markstridssimuleringar).
- Testa framtagna funktioner för att koppla befintliga arkitekturer till simuleringsmiljöer (SOAR-UNIX, SOAR-HLA)
- Komplettera framtagna riktlinjer för uppbyggnaden av ett referensbibliotek för kognitiva/beteendemodeller.
- Fortsätta studera olika former för kunskapsrepresentationer och deras för- och nackdelar (vari ingår jämförande arbeten och prototyping med hjälp av intressanta kognitiva arkitekturer och AI-system).
- Upprätthålla fortsatt diskussion med Försvarsmaktens simuleringsanläggningar och andra användare av datogenererade styrkor, t. ex Flygvapnets Luftstrids Simulerings Center (FLSC) och Markstridskolan om möjligheter att testa framtagna CGF-modeller.
- Besöka konferenser med målsättning att följa utvecklingen inom området men också att kunna presentera hos oss pågående verksamhet.

Genomförande

Projektgruppen har under 2004 bestått av sammanlagt sju forskare (två från avdelningen för Systemteknik och fem från MSI-institutionen vid avdelningen för Ledningssystem varav två i huvudsak verksamma vid FLSC). Följande aktiviteter har genomförts:

1. Fortsättning på arbetet med utveckling av biblioteksuppbyggnad av CGF:er
2. Kunskapsinhämtning och uppgiftsanalys vid SIB-övning
3. Utveckling av teknik och metodik för insamling av rörelsedata och visualisering av dessa med hjälp av 3D-teknik.
4. Presentation vid BRIMS-konferensen
5. Reseberättelse har skrivits från BRIMS-konferensen

6. Memo om krav på insamling av mätdata att användas för CGF-modeller har tagits fram
7. Samarbete i en NATO/RTO grupp inom området har startat
8. Kunskapsspridning genom presentation och demonstration vid M&S-dagen (årligen återkommande av FOI anordnad informationsdag för kunder och intresserade om M&S-verksamheter vid FOI)
9. Nationellt samarbete och övriga presentationer

Resultat från årets verksamhet

Utveckling av biblioteksutbyggnad av CGFer

Utvecklingen av riktlinjer och strukturer av komponentbaserad utbyggnad av bibliotek med kognitiva/beteendemodeller pågår. Biblioteket använder sig av det komponentbaserade ramverk som utvecklats i MERLIN-projektet. (Däri definieras struktur av bibliotek för plattformsmodeller, vapen, motmedel m.m.) Under året har en del plattformsgränssnitt samt administrativa funktioner tagits fram. De standardiserade gränssnitten gör att en agent kan agera i olika simuleringsmiljöer genom s.k. plug'n'play funktionalitet. En sammanfattande beskrivning av detta finns med redan i föregående års CGF-årsrapport, *Datorgenererade styrkor med mänskliga egenskaper* Metodrapport FOI-R--1076--SE, December 2003. De administrativa funktionerna förenklar arbetet att skapa och exekvera beteendemodeller i olika simuleringsapplikationer. Funktionaliteten har testats mot två olika flygsimulatorer med framgång.

Kunskapsinhämtning och uppgiftsanalys vid SIB-övning

Under två dagar i februari följde en grupp i projektet en stridsövning i centrala Norrköping. Syftet var att utvärdera olika former av kunskapsinhämtning av beteendedata och hur väl resultaten av dessa överensstämmer med varandra. En sammanställning av denna kunskapsinhämtning utgör ett underlag för en uppgiftsanalys. Detta är endast ett isolerat enkelt försök, vilket gör att de slutsatser och resultat som diskuteras inte kan anses som allmängiltiga. Det material som användes, gruppens erfarenhet, de övandes och övningsledarnas erfarenheter är exempel på faktorer som påverkar resultatet i hög grad.

Övningen var till för att låta värnpliktiga öva Strid i bebyggelse (SIB) genom att framrycka längsgator och bekämpa eventuella fiender. En pluton framryckte växelvis med sina stridsvagnar och med avsutten trupp. Speciella manövrer utfördes vid t.ex. minering, korsningar, och sammanstötning.

För att utveckla en beteendemodell och därefter värdera modellen krävs det att man noggrant ser till behovet, dvs. hur modellen kommer att användas. Eftersom kunskapsinhämtningen ligger till grund för modellen måste även kunskapsinhämtningsmetoderna värderas utifrån den modell man vill skapa. När det gäller modellering av beteende så har strid i bebyggelse en väldigt hög komplexitet, vilket gör det extra svårt. Gruppen valde därför att i kravspecifikationen beskriva en beteendemodell av en vagnchef i en stridsvagn som utför enkel framryckning. Modellen skall kunna användas i träningssyfte och kunna delta i simuleringar/simulatorer med andra beteendemodeller och/eller mänskliga operatörer. En mänsklig operatör skall uppleva träningen som om det vore med enbart mänskliga operatörer. Dels skall modellen kunna kommunicera med stridsvagnscheferna i den egna plutonen och dels med den interna besättningen (förare och skytt). Utifrån dessa krav samt uppgiftsanalys utvärderades metoderna för kunskapsinhämtning.

Kunskapsinhämtning via dokumentation

Gruppen började med att sätta sig in de SIB-moment som senare skulle övas med att gå igenom utbildningsmaterialet "Utbildningsanvisning SIB 02-03". Utifrån detta material erhöles en uppfattning om hur t.ex. framrykning längs väg ska genomföras. En enkel uppgiftsanalys skapades med detta material.

Kunskapsinhämtning via instruktioner och intervjuer

Innan övningens början gjordes intervjuer med de övande om hur de agerar i de olika situationerna. Därefter deltog gruppen i utbildningsledarnas genomgångar och övningsledarna gick igenom de olika momenten på ungefär samma sätt som utbildningsmaterialet är upplagt. Gruppen gjorde en förnyad uppgiftsanalys.

Kunskapsinhämtning via observation

Under övningen följde gruppen med bland de övande för att observera hur de olika momenten utfördes, samtidigt som de hade möjlighet att ställa kompletterande frågor till de övande samt utbildningsledarna. Övningen filmades även med videokamera. Uppgiftsanalys gjordes ånyo.

Analys

Efter övningen gjordes en analys och jämförelse av de olika metoderna.

1) Kunskapsinhämtning via dokumentation

Studiematerialet gav en överblick över hur momenten skall genomföras. Materialet kändes dock inte tillräckligt för att kunna skapa en uppgiftsanalys för den modell som eftersöktes. I många fall fick man göra egna antaganden och gissningar. Många situationer saknades, t.ex. hantering av minering.

2) Kunskapsinhämtning via instruktioner och intervjuer

Utbildningsledarnas genomgång av momenten var ett bra komplement till dokumentationen. Intervjuerna av besättningen kändes dock mindre viktig i detta skede, men gav information om hur man arbetar internt i stridsvagnen, något som saknades i dokumentation och ej heller behandlades av övningsledarna. Intervjuer av övningsledarna under övningen gav en hel del information om "konstiga" situationer och hur de ska lösas.

3) Kunskapsinhämtning via observation

Genom att observera övningarna blev det snabbt klart hur komplext beteendet är trots det till synes enkla scenariot. Brist på kommunikation, missuppfattningar och obeslutsamhet var mycket vanligare än väntat. Detta ledde till att det blev mycket stillastående, "felaktiga" manövrer och till slut onödigt många förluster. Att utifrån enbart observation skapa en uppgiftsanalys är praktiskt taget omöjligt om man inte sedan tidigare har bra förståelse för hur och varför vissa situationer uppkommer och hur de ska lösas. Enligt den konceptuella modell som utformades, var dock observation nödvändig för att kunna modellera de "brister" som förekom ("krigets dimma") eftersom simulatorm skulle innehålla en realistisk träningsmiljö. Det är svårt att få grepp om dessa faktorer utan att vara på plats och observera vad som faktiskt händer och varför. Det är dock

viktigt att man får de övandes synpunkter på övningens resultat för att kunna skaffa sig en bättre bild av problematiken, varför någon sorts "after action review" är nödvändig.

Slutsats

Alla tre metoderna för kunskapsinhämtning upplevdes som viktiga för att kunna genomföra uppgiftsanalysen. En "after action review" med genomgång av situationerna med de övande och övningsledarna hade varit önskvärd. Utifrån kraven rekommenderar därför gruppen ordningen:

- 1) Studera dokumentation, manualer, etc (ger grundläggande kunskaper för gruppen)
- 2) Delta i övningsledarnas genomgångar
- 3) Observera övningarna, gärna tillsammans med övningsledare eller annan expert
- 4) Intervjua de övande tillsammans med expert i en "after action review"

Utveckling av teknik och metodik för insamling av rörelsedata och visualisering av dessa med hjälp av 3D-teknik.

En av huvudverksamheterna i projektet under året har varit att använda existerande teknik för mätdatainsamlingar av naturliga och individuella fysiska rörelsemönster. Detta är en populär teknik hos spelindustrin men även inom medicin och sjukvård. Däremot har det hittills fattats metodik och i någon mån teknik att på ett rationellt sätt använda sig av dessa mätdata för att kunna använda som motor för datorgenererade aktörer med naturliga rörelsemönster. En tanke med denna verksamhet förutom att utveckla denna metodik är att göra mätdatainsamlingar för en mängd typiska rörelser, samla dem i bibliotek och genom att beräkna smidiga övergångar mellan typrörelser kunna styra datorgenererade aktörer i virtuella världar beroende på situation och tillämpning. En koppling mellan denna verksamhet och en verksamhet som presenterats vid BRIMS 2004, se bilaga 4 nedan, är uppenbara. Vår tanke är också att i likhet med denna presentation på sikt kunna kombinera teknik för rörelsefunktioner med styrning av intelligenta och kognitiva funktioner för simulering av den avstutne soldaten.

Framgångarna med detta arbete så här långt finns kortfattat beskrivna i bilaga 1, vilken också finns som ett fristående memo, "Motion Capture", FOI Memo 963.

För denna verksamhet har kunskap inhämtats bl a genom ett besök på Game Developer Conference, GDC i San José, USA, 22-26 mars 2004. För projektets räkning bevakades en rad intressanta områden såsom AI för CGF:er, användning av motion capture, integrering av fysik för CGF:er och design i allmänhet av system som innehåller datorgenererade aktörer.

Krav på insamling av mätdata att användas för CGF-modeller

Projektet har haft god tillgång på data inspelade både från verkligheten och från simuleringar av mångahanda slag genom samarbete bl a med FOI-projektet "MSI Metodutveckling". Ett stort intresse finns i projektet att kunna använda dessa data och på bekvämt sätt kunna föra in dem i de modeller man vill producera och använda. Detta har dock visat sig vara svårare än vad vi först trodde. Orsakerna till detta är flerfaldiga. För det första är sammanhanget och situationen där

mätdata är hämtat inte alltid tillräckligt bra beskriven. För det andra visar det sig ofta fattas data för att kunna definiera tillräckligt bra modellerade kognitiva funktioner. Detta är ett generellt problem när man vill bygga kognitiva modeller. Det är över huvud taget svårt att både bygga och bekräfta kognitiva modeller med de mätdata som finns. Det beror delvis på att det finns väldigt litet mätdata från såna funktioner och dels på att utfallsrummet är så stort. För det tredje är det svårt att matcha existerande mätdata med de modeller som är uppbyggda med ett typiskt mönster med mål och undermål som ofta är den struktur som stöds av kända arkitektuer som Soar och ACT-R och som passar in i en programvarustruktur med if-then-else satser. För det fjärde så är det bara att konstatera att mätdata inte har samlats in med tanke på att de skulle kunna användas för CGF-utveckling.

Hur dessa frågor behandlats och försöken att hitta en metod för mätdatainsamling och krav för denna redovisas i FOI-Memo 1016, som tagits fram inom projektets ram.

Däri beskrivs de problem som ofta uppstår vid utvecklingen av datorgenererade styrkor där beteendedata används för att få mer realistiska modeller. Med beteendedata avses i rapporten både uppgiftsanalyser och data om prestation och belastning.

Problemen kan grupperas under tre områden. Det första är det konceptuella området där frågor som vad modellen skall kunna klara av, om den skall vara mest beteende- eller mest processinriktad, måste besvaras. Processinriktning innebär att funktioner för människans kognitiva processer beskrivs. Det andra området handlar om hur man skapar och bygger upp kunskap om hur modellen skall fungera utifrån analyser av och mätdatainsamlingar från verkligheten. Det tredje området innehåller frågor om verifiering och validering av beteendemodeller.

Som ett steg på vägen att bemästra dessa problem presenteras en mall för insamling av beteendedata som kan användas vid CGF-utveckling och enklare och tydligare kopplas till regelutvecklingen.

Se vidare bilaga 2.

Bidrag till BRIMS-konferensen

Utifrån arbete i projektet under 2003 har ett konferensbidrag/presentation tagits fram av projektet under början av 2004, skickats in till BRIMS-sekretariatet och blivit antaget för presentation vid BRIMS-konferensen i maj (BRIMS, Behavior Representation in Modeling and Simulation). Konferensbidraget bygger på ett examensarbete som utfördes i projektet under 2003 och hade inriktningen beteendemoderatorer, Performance Moderator Functions (PMF). Konferensbidraget beskriver arbetet med att modellera en stridsvagnsbesättning alltifrån den konceptuella modellen med uppgiftsanalys till teorier om mental arbetsbelastning och hur denna påverkar prestationen och beteendet. Modellerna är implementerade i den kognitiva modelleringsarkitekturen Soar och exekveringen av modellerna visar intressanta skillnader i beteendet beroende på arbetsbelastning och stress. Se mer i bilaga 3.

Reseberättelse från BRIMS-konferensen

En medarbetare i projektet deltog i konferensen om datorgenererade styrkor i USA. Reseberättelsen beskriver dels några verktyg och hjälpmedel som är intressanta, bl a PMFServ och REDUX, dels några paneldebatter kring design patterns, modellering av folkmassors beteende och modellering av naturligt språk. En intressant koppling till vårt projekt kan ses i en presentation av Gita Sukthankar et al på Carnegie Mellon University vilka har skapat en modell för resonering om kroppens fysik. Den utgår från motion-capture data och skapar sedan en kostnadsfunktion för vilken typ av rörelse som ska göras. På så sätt ser projektgruppen det som resonering om sin egen kropp, dvs en beräkningsmodul mellan Soar's "högre" resonering och den fysiska modellen. De motiverar detta med att begränsningarna till beteenden sitter i hur man resonerar om sin kropp snarare än de fysiska begränsningarna. Se mer i bilaga 4.

En slutsats som kan dras från besöket och medverkan vid konferensen är att vårt arbete ligger väl framme vid forskningsfronten inom området.

Samarbete i en NATO/RTO grupp inom området

Projektet har sedan ett par års tid haft kontakt med en NATO RTO Exploratory Group i Europa. Denna grupp har länge haft för avsikt att starta en arbetsgrupp inom området Datorgenererade styrkor vilket också kom till stånd lagom till sommaren 2004.

Under 2004 har projektmedarbetarna deltagit vid två möten med gruppen NATO RTO HFM 128 - Representation of Human Behavior in Constructive Simulation. Förberedelsearbete pågick under 2003 och gruppens första möte hölls 30 mars 2004 vid TNO i Soesterberg. FOI var värd för gruppens andra möte som hölls i Stockholm 13-15 september 2004.

Gruppens deltagare (se tabell nedan) representerar en mycken erfaren samling forskare inom området. Detta framgår t.ex. av att en av gruppens medlemmar är huvudarrangör för BRIMS konferensen under 2005.

Namn	Organisation	Land	Kommentar
Laurel Allender	US Army Research Labs	USA	Arrangör av BRIMS 2005
Andy Belyavin	QinetiQ	Storbritannien	
Brad Cain	DRDC	Kanada	
Martin Castor	FOI	Sverige	
Wolf Käppler	FGAN	Tyskland	
Peter Kwantes	DRDC	Kanada	
Woter Lotens	TNO	Nederländerna	Ordförande
Mikael Lundin	FOI	Sverige	
Sten-Åke Nilsson	FOI	Sverige	
Gina Thomas-Meyers	US Air Force Research Labs	USA	
Niklas Wallin	FOI	Sverige	

Tabell 1. Deltagare i stockholmsmötet av NATO RTO HFM 128 - Representation of Human Behavior in Constructive Simulation

Verksamhet

Gruppens målsättning är att inom NATO RTO (inkl. PfP-länder som Sverige) samverka och dela erfarenheter kring kognitiv modellering och användandet av datorgenererade styrkor och agenter för olika militära tillämpningar som t.ex. träning, taktikutveckling, materielanskaffning och ”mission rehearsal”. Gruppen kommer att författa ett större dokument som beskriver riktlinjer för och erfarenheter från området.

Gruppens nästa möte sker i samband med BRIMS 2005, då den också troligen kommer att hålla ett seminarium. Inför nästa möte finns också gemensamma och enskilda skrivuppgifter för gruppens medlemmar.

En NATO RTO-grupp är normalt aktiv under tre år och denna grupps verksamhet avses också pågå under tre år från starten sommaren 2004.

Nationellt samarbete och övriga presentationer

- CGF-projektet deltog aktivt vid FOI M&S-dag vid IVA den 4 februari med presentation av principer för modelluppbyggnad samt med demonstration av enkla AI-funktioner byggda på Soar-arkitektur.
- Under några möten med personal vid SMART-lab på FMV har diskuterats koppling av en CGF(Soar)-modell till programvaran FbSim. Detta är avsett som en bra lösning på problemet med att förse FbSim med agenter som uppträder mer dynamiskt och mänskligt. Mötena har haft målet att ta fram ett scenario där man vill kunna demonstrera hur mänskligt beslutsfattande kan modelleras vid en terroristaktion. Tanken med denna samverkan mellan FMV och FOI var att vi ville se hur man kan kombinera programvaror som redan finns för att åstadkomma en förbättrad simulering framför allt genom en mer omfattande modellering av mänskligt beteende och beslutsfattande. Arbetet är inte slutfört p.g.a. begränsat tidsutrymme för alla parter.
- Under ett möte med personal från utbildningssystem Leopard 2, FMV, har vi diskuterat möjligheterna att förse besättningsträningsanläggningar BTA och TBTA med agenter som uppträder dynamiskt med korrekta rörelsemönster.
- En föreläsning om verktyget/arkitekturen Soar har vi genomfört med efterföljande laboration för studenter vid Linköpings Universitet, IDA-institutionen. Soar är den kognitiva arkitektur vi lärt oss mest om under projektiden.
- En 3D-manikin simulator har förevisats för Markstridsskolan, MSS, i Skövde. Denna simulerar enskild persons uppträdande såväl som gruppers taktiska uppträdande i olika miljöer och kan utnyttjas i existerande simulatorer för att representera svenska soldater, poliser, brandmän, terrorister etc.

Summering av årets verksamhet inom CGF-projektet

Kunskapsspridning och kontaktverksamhet har blivit mer omfattande än vad som var planerat. Det är i sig mycket glädjande att se att vårt arbete ligger väl framme vid forskningsfronten, att samarbetsformer har hittats med FMV och att NATO RTO-gruppen inom Human Behavior Representation i Europa äntligen kommer igång med en arbetsgrupp och att vi är med. Detta har dock haft till följd att vi under året inte har hunnit med att studera och testa andra arkitekturer och kunskapsrepresentationer.

Referenser

Castor M., Wallin N., Nilsson S-Å., Moradi F., *Datorgenererade styrkor – Metoder och möjligheter* Användarrapport FOI-R--0688-SE, December 2002

Wallin N., Lundin M., Nilsson S-Å., Hasewinkel H., *Datorgenererade styrkor med mänskliga egenskaper* Metodrapport FOI-R--1076-SE, December 2003

Bilaga 1, Motion caption



FOI Memo 963
Juli 2004

Projekt nr: E6026
Organisation: 780
Forskningsområde: 2. Operationsanalys, modellering och simulering
Delområde: 21. Modellering och simulering
Verksamhetsgren: 5. Uppdragsfinansierad verksamhet
Godkänd av: Erland Svensson

Motion caption

Henrik Allberg

Bakgrund och Syfte: Relevanta modeller av människan, såväl fysiska som psykiska, eftersträvas för att ge såväl ett korrekt rörelsemönster (fysiskt) som ett korrekt mänskligt beteende (psykiskt). Detta för att korrekt simulera en person i en simulerad omvärld som anpassar sig efter uppgift, terräng, situation etc. Att samla in psykofysiologiska data, att ha som underlag för dessa simuleringar är därför av intresse. Insamlad data kan t.ex. vara puls, pulsvariation och rörelsemönster. Att korrekt avbilda personers rörelsemönster är viktigt vid simuleringar av ”digitala personer” som befolkar simulerade världar.

Ett rörelsemönster som återger en persons verkliga rörelser med hög noggrannhet och god överensstämmelse med verkligheten, i en simulerad omvärld, eftersträvas för den simulerade personen. Så liten skillnad som möjligt eftersträvas mellan verklighet och modell för att en betraktare t.ex. en försökspersons slutsatser och beslut i en träningsituation, inte skall påverkas av den simulerade världen.

Då avståndet mellan verkliga rörelser och animeringar i simulerade världar krymper, ges möjlighet till bättre utbildnings- och träningsimulatorer, vilket i förlängningen ger bättre prestations- utvärderingsverktyg. Därför finns det ett behov av förbättrade metoder och system för inspelning av rörelser s.k. Motion Caption.

Inledning problemställning

Hur skall man fånga och återge ett rörelsemönster hos en person?

En möjlighet är att modellera hela rörelseschemat. Ett skelett byggs upp med kända anatomiska avstånd. Leder som sammanlänkar benen i skelettet byggs upp, med kända rörelsefriheter i 3D-

rymden. För detta skelett byggs sedan ett rörelseschema upp. Denna lösning medför att varje rörelse måste modelleras exakt in i minsta detalj. Ett komplett rörelseschema byggs sedan upp genom att modellera varje rörelse. Notera att varje rörelse måste modelleras från grunden. En alternativ metod till att modellera rörelser från grunden är att spela in rörelser med så kallad ”Motion Caption” -teknik. Denna teknik använder sensorer för att spela in en persons rörelser.

Motion Caption

Ett rörelsemönster för en person är komplexare än för ett fordon, där ett fåtal parametrar kan modellera fordonets dynamiska egenskaper. Samtidigt har fordon av samma typ och modell samma dynamiska egenskaper varför bara en uppsättning parametrar behöver mätas upp för fordonet. Dessa parametrar kan sedan användas i simuleringar för flera fordon. Människan däremot är individuell i sitt rörelsemönster och kan inte generaliseras till endast ett rörelsemönster. Att bygga upp ett referensbibliotek för dessa rörelser är därför av intresse.

Krav på Motion Caption utrustning

För att spela in rörelser bör ett antal grundkrav vara uppfyllda:

- Datainsamling skall alltså så långt som möjligt ske oberoende av omgivningen och skall t.ex. vara möjlig att genomföra i fordon där mycket svåra mätförhållanden råder.
- Inspelning av rörelser skall således kunna göras oberoende av vilka förhållanden som råder vid inspelningstillfället. Tex. ska ljusförhållanden, elektromagnetiska störningar och typ av rörelsemönster, (statiskt eller aktivt), inte påverka inspelningen av rörelsen.

Den utrustning som motsvarade kraven ovan bäst var Measurands Shaperecorder. Den centrala komponenten i utrustningen är Measurands ShapeTape™, en fiberoptisk 3D böj- och vridbar sensorarray. Den optiska fibern har behandlats så att den förlorar ljus proportionellt mot böjningen av densamma. Dvs. det ljus som passerar genom sensorn integreras över längden på densamma. Med denna sensor kan information om såväl position som orientering inhämtas också när sensorn bara är i partiell kontakt med mätobjektet, dvs. mätsystemet kan räkna ut relativa koordinater för leder i skelettet och därmed kompensera för offsets i systemet. En perfekt placering av sensorerna är därför inte nödvändig. Fiberoptik är inte känslig för elektromagnetiska fält i omgivningen vilket möjliggör mätningar även i störcänsliga miljöer t.ex. inne i fordon. Utrustningen är heller inte känslig för vilka ljusförhållanden som råder. Därför valdes denna utrustning för datainsamling. Dessutom finns två gyron som mäter huvudets och bäckenets rörelser och position.

Länkmodell

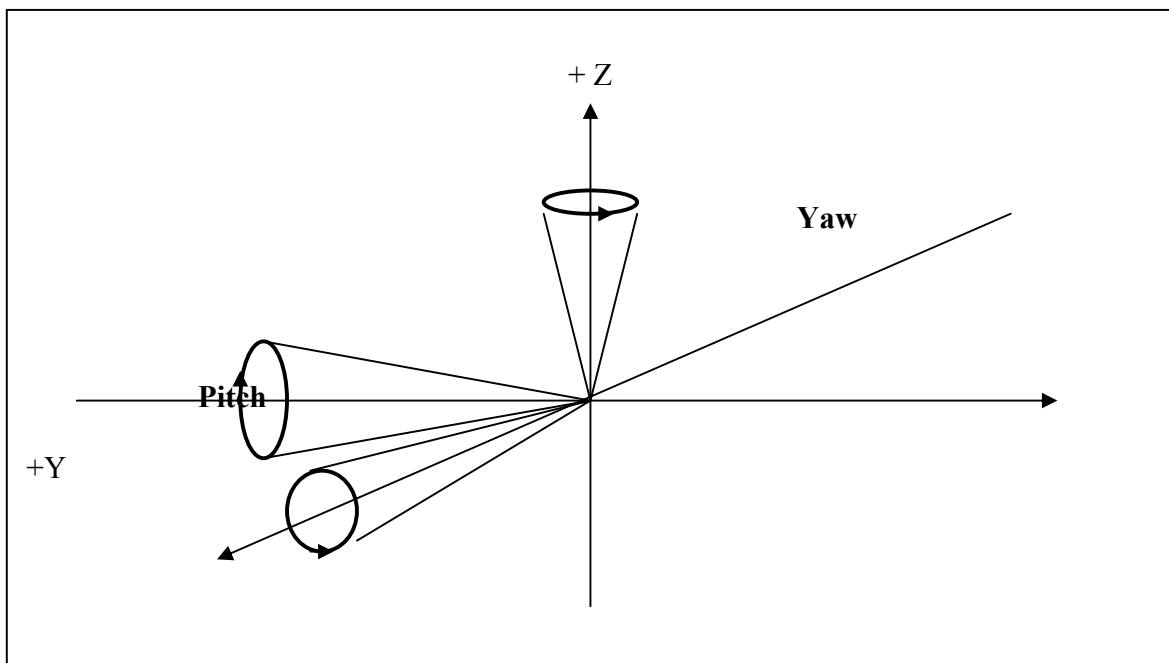
En länkmodell är en modell som bygger upp skelettet som en serie av länkar. Ett ben byggs t.ex. upp av lårben, vadben, fot och tå. På samma sätt byggs armen upp av överarm, underarm, hand och finger. Hela skelettet byggs sedan upp på motsvarande sätt genom att länka samman skelettets olika delar med leder och definiera vilka frihetsgrader respektive led har. En knäled kan bara röra sig runt en och samma axel medan en höftled kan röra sig runt tre axlar (Figur 1).

För att få överensstämmelse mellan mätobjekt och simulerad modell anpassas det modellerade skelettet till försökspersonens anatomi. Avstånden mellan varje led på mätobjektet mäts upp och den modellerade personen anpassas därefter.



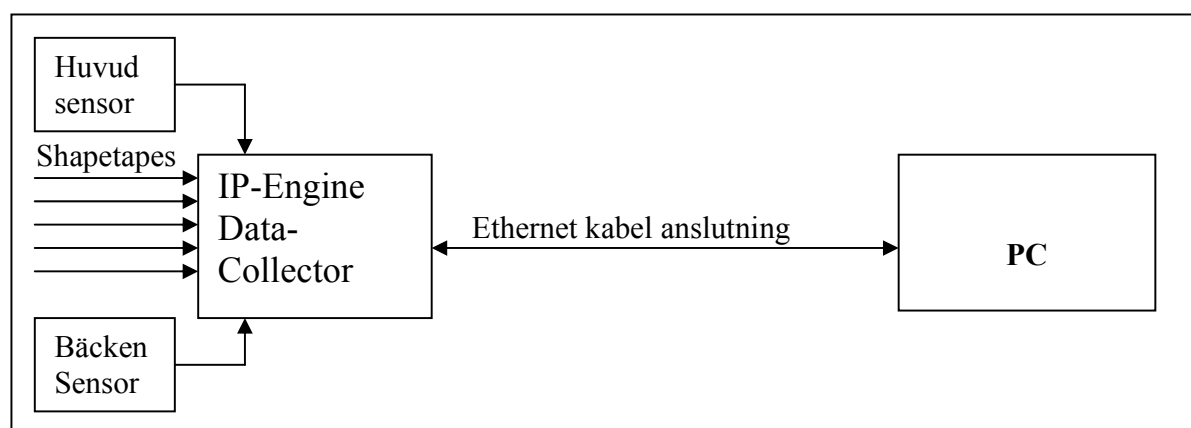
Figur 1. Modellerat skelett [3].

Rörelserna för varje led definieras i 3D-rymden genom att avstånd mellan lederna bestäms och rotation kring X-, Y-, och Z-axlarna i modellen bestäms. Rotationsriktningarna definieras som Yaw-Rotation kring +Z axeln, Pitch-rotation kring -Z-Axeln och Roll-rotation kring X-Axeln. Hur stor rörelsefrihet en led har beror på vilken led som definieras. (Figur 2)



Figur 2. Rotationsriktningar.

Datinsamling: Datinsamling genomfördes som ett metodförsök där principerna för motion caption prövades. En person utrustades med Shaperecorder motion caption utrustning (Figur 4) ShapeTapes monterades på armar, ben och rygg. Gyron monterades på huvud och bäcken. Sensorerna anslöts sen till en IP-Engine datainsamlare som samlade in inkommande data från Shapetapes och kommunicerade med den installerade Shaperecorder mjukvaran [1] på en PC. Kommunikation mellan mätutrustning och dator upprättades, varefter personen rörde sig i ett på förväg bestämt rörelsemönster, samtidigt som rörelserna registrerades av mjukvaran och sparades på ett i förväg valt filformat på en PC. (Figur 3)



Figur 3. ShapeRecorder mätuppkoppling.



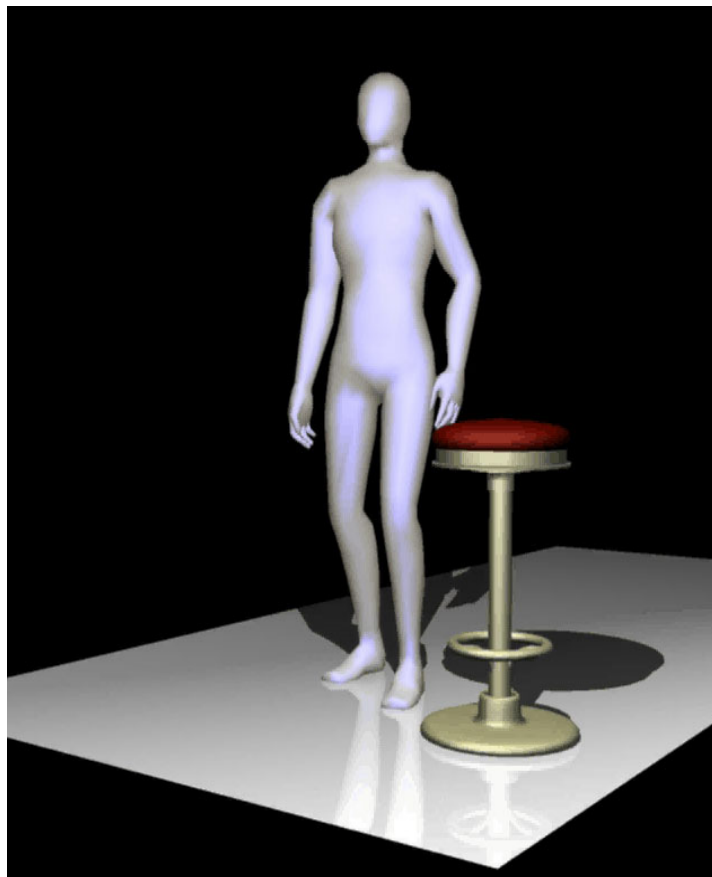
Figur 4 Monterad ShapeRecorder Motion Caption utrustning

Rensning av data

Idealt skulle en insamlad datamängd inte vara behäftad med några fel men så är ofta inte fallet. Drift av mätvärdena pga. felkalibrering och temperaturväxlingar kan förekomma. För att kompensera för dessa felkällor kan mätdata rensas upp. Denna upprensning går till så att offsets justeras för respektive led och data filtreras för att få det rörelsemönster som eftersträvas. T.ex. kan ett ryckigt rörelsemönster slätas ut för att få ett mjukare slutresultat. Den rensade rörelsedatan är sedan färdig att användas för att skapa rörelser för det tidigare anatomiskt anpassade simulerade skelettet.

Ytmodell

För att göra det möjligt att hänga kläder på den simulerade personen läggs ett skinn på skelettet. Det vill säga en ytmodell som begränsar den simulerade kroppens omfång skapas. På denna ytmodell kan sedan den textur, i det här fallet kläder, hängas på för att ge den simulerade personen önskvärda karakteristika. Figur 5 nedan visar ett exempel på hur en ytmodell av en kropp kan se ut.



Figur 5. Ytmodell av simulerad person.

Textur

Därefter textureras skinnet, dvs. 3-D modellen kläs på. I det här fallet har Uniform M/90, Stridssele, hjälm och kängor byggts upp med 3-D studio MAX [2] och fästs på 3-D modellens skinn varefter den färdiga soldaten placerats i en simulerad värld (Figur 6). Därmed är den fysiska delen av den simulerade personen klar.



Figur 6. Texturerad 3-D Modell placerad i simulerad värld.

Resultat och diskussion

Motion Caption är ingen separat lösning för 3-D animering av rörelser utan snarare ett komplement till den traditionella tekniken att modellera rörelser där varje rörelse byggs upp från grunden. Utrustningen måste noggrant kalibreras innan mätning och rörelsedata från mätningar måste rensas/filtreras för att ge en bra återgivning av rörelserna i en simulerad värld. Mätresultatet kan annars behäftas med mätfel som ökar efterarbetet av rörelsedata.

Motion Caption teknik minskar däremot tiden för att modellera rörelser och är ett bra verktyg för att snabbt bygga upp t.ex. ett rörelsebibliotek som kan användas i simulerade världar. Ur detta bibliotek kan sedan rörelsemönster för olika simulerade personer extraheras. Ur detta referensbibliotek kan olika kombinationer av rörelser sammanfogas till en stor mängd rörelsemönster anpassade till simulerade personer med olika funktioner. Tillståndsovergångar mellan rörelserna definieras av vilka tillståndsovergångar som är tillåtna och hur tillståndsovergångarna ska ske.

Vissa rörelser kan av mättekniska skäl inte mätas, t.ex. förstörande provning där sensorerna i mätutrustningen inte skulle klara av påfrestningarna under mätningen. Dessutom finns det etiska skäl att inte genomföra vissa försök. Dessa mätningar kan av förståeliga skäl inte genomföras utan modelleringar får ske från grunden utgående från bildmaterial och filmer.

Beroende på vad den simulerade modellen skall användas till kan den ges psykofysiologiska data och/eller AI, Artificiell intelligens. En kognitiv arkitektur t. ex Soar är lämplig att använda för att bygga denna AI-funktion och ytterligare förstärkt mänskligt uppträdande kan åstadkommas med hjälp av s.k. beteendemoderatorer som knyts till modellen [4].

Internet-referenser:

- [1] www.measurand.com (040601)
- [2] www.3dmax.com (040601)
- [3] www.kaydara.com (040601)

Referenser:

- [4] Lundin, M. (2004) Simulating the effects of mental workload on tactical and operational performance in tankcrew. LIU-KOGVET-D—04/xx—SE, Linköping Electronic Press, Sweden.

Bilaga 2, Krav på beteendedata till CGF-modeller



FOI Memo 1016
September 2004
Projektnamn:
Datorgenererade Styrkor
med mänskligt beteende
Delområde:
M & S
Kund: FM

Krav på beteendedata till CGF-modeller

Martin Castor & Niklas Wallin

Systemteknik
172 90 Stockholm

Sammanfattning

Denna rapport beskriver problem som ofta uppstår vid utvecklingen av datorgenererade styrkor där beteendedata används för att få mer realistiska modeller. Med beteendedata avses här både uppgiftsanalyser och data om prestation och belastning.

Problemen kan grupperas under tre områden. Det första är det konceptuella området där frågor som vad modellen skall kunna klara av, om den skall vara mest beteende- eller mest processinriktad, måste besvaras. Processinriktning innebär att funktioner för människans kognitiva processer beskrivs. Det andra området handlar om hur man skapar och bygger upp kunskap om hur modellen skall fungera utifrån analyser av och mätdatainsamlingar från verkligheten. Det tredje området innehåller frågor om verifiering och validering av beteendemodeller.

Som ett steg på vägen att bemästra dessa problem presenteras en mall för insamling av beteendedata som kan användas vid CGF-utveckling (Computer Generated Forces) för att beteendedata enklare och tydligare ska gå att koppla till regelutvecklingen.

1. Bakgrund

Datorgenererade egna och fientliga styrkor är något som bland annat det amerikanska DMSO (Defence Modeling and Simulation Office) under lång tid pekat ut som en central del i visionen om hur simuleringar ska användas i framtida militära sammanhang (se t.ex. Objective 4 i DMSO Modeling and Simulation Master Plan, 1995). Många termer används för att beskriva dessa datorgenererade styrkor och i denna rapport kommer termerna CGF (Computer Generated Forces) och intelligenta agenter att användas synonymt.

För att kunna implementera en tillfredställande CGF-tillämpning där de datorgenererade styrkorna uppför sig som de modellerade operatörerna och plattformarna tillräckligt bra så behövs en mängd komponenter som:

- Uppgiftsanalyser som beskriver beteendet.
- Experimentella data som beskriver belastning och prestation och vilka faktorer som påverkar dessa.
- En modelleringsarkitektur där innehållet i ovanstående två punkter kan beskrivas.
- En beräkningsmässigt och programmeringsmässigt effektiv arkitektur.
- Gränssnitt till simuleringsvärldar.
- Verktyg för uppdragsplanering.
- Mekanismer så att en operatör, t. ex en övningsledare, kan styra de datorgenererade styrkorna för att få fram de pedagogiska eller studiemässiga effekter som behövs.
- Verktyg för att underlätta uppbyggandet av modellerna.

Det är viktigt att inse att alla dessa byggstenar behövs för att få till stånd en effektiv CGF-utveckling. Denna rapport berör de två första punkterna.

1.1 Visionen

Målsättningen för utvecklingen av datorgenererade styrkor bör vara att det för varje intressant domän eller tillämpning finns agenter som har ”en kropp och sinnesorgan” i den simulerade världen (d.v.s. agenten är ”situated” och ”embodied”). De ska kunna interagera med övriga aktörer i världen såsom andra intelligenta agenter, plattformar som styrs av bemannade simulatorer och spelledare. Agenternas beteende ska efterlikna det beteende som verkliga operatörer i samma situation uppvisar.

Intelligenta agenter eller datorgenererade styrkor kan användas i många olika tillämpningar som träning, materielanskaffningsstudier och i planeringsverktyg. Simuleringens syfte medför olika höga krav på den kognitiva modelleringen (d.v.s. att agenternas uppfattning av världen och tänkande i någon mening liknar motsvarande processer hos människan). En simulering med datorgenererade styrkor som ser allt, räknar med alla faktorer i varje beslut, aldrig blir trötta och som fattar beslut på några millisekunder är användbar i färre situationer/studier än där de datorgenererade styrkorna innehåller någon sorts modell av relevanta kognitiva funktioner. Agenterna behöver dessutom vara implementerade i en arkitektur som medger att de dynamiskt kan ändra sin målsättning, planer och prioriteter. De ska t.ex. kunna möta ett uppdykande hot och ta hand om det för att sedan återgå till sitt ursprungliga mål samt samverka med andra agenter.

I träningsimulatorer som t.ex. FLSC (Flygvapnets Luftstrids Simulerings Center) och BTA 122 (Besättningsträningsanläggning Stridsvagn 122) kan man komma en bra bit på vägen att simulera fientliga styrkor med beteendemodeller, d.v.s. modeller som ger ett trovärdigt observerbart beteende. I simuleringar och modeller för avancerade studier av t.ex. informationshanteringen i det framtida nätverksbaserade försvaret (NBF) och beslutsstöd fordras mer detaljerade modeller. Exempel på innehåll som kommer att behövas är modeller av informationshantering och kommunikation samt även hur stress och uttröttning påverkar prestationen.

Med intelligenta agenter avses i denna rapport agenter som har ett tillräckligt komplext beteende för att fungera i komplexa situationer där de interagerar med andra agenter eller människor och en simulerad värld. Sådana agenter refereras ofta till som "human level agents" eller "knowledge intensive agents". Agenterna måste också på någon nivå resonera som mänskliga beslutsfattare för att beteenden ska kunna förklaras och valideras.

Kognitiv modellering är svårt och behöver stödjas med mycket kunskap och goda verktyg. Eftersom denna kunskapsuppbyggnad tar lång tid rekommenderar vi att Forsvarsmakten uthålligt fortsätter satsa resurser på detta.

1.2 Vad är en modell?

En modell är en förenklad representation på konceptuell nivå av (en del av) den riktiga världen och/eller hur den fungerar, som är tillräckligt detaljerad/korrekt för att man ska kunna använda modellen för att dra vissa slutsatser om den riktiga världen och/eller hur den fungerar. En modell består av komponenter och relationen mellan dessa komponenter, som oftast är någon sorts orsak-verkan relation. En modell kan ofta visualiseras i någon grafisk form.

Viktiga egenskaper hos en modell är:

- Den reducerar komplexiteten hos den riktiga världen.
- Den betonar vad som bedöms nödvändigt eller intressant.
- Relationen mellan modellens olika komponenter är tydlig.
- Modellen ska kunna användas för att upptäcka och studera hypoteser om hur den verkliga världen fungerar genom att lägga till eller ta bort komponenter, ändra i parametrar eller relationer mellan komponenter.

Modellutveckling innebär att utvecklaren måste avgöra vilka egenskaper hos den riktiga världen som ska vara med i modellen och vilka som inte ska vara med. Därför måste kriterier för vad som ska vara med upprättas. Ett krav på dessa egenskaper är att de ska gå att uttrycka i variabler som kan kvantifieras.

Gustafsson (2002) beskriver arbetet med att ta fram en modell i åtta steg:

- Välja vilka konceptuella begrepp/komponenter som ska ingå i modellen.
- Definiera vilka parametrar som i idealvärlden speglar dessa konceptuella begrepp/komponenter.
- Skriva om parameterlistan för att avspegla vilka parametrar man faktiskt får/har tillgång till information om.
- Utveckla en strategi för hur man ska omvandla "rå information" (t.ex. intervjuer eller observationer) till numeriska värden.

- Konstruktion av skalor för parametrarna.
- Vikta parametrarnas vikt/påverkan mot varandra i olika situationer.
- Använda utvald algoritm för att kunna generera ett beteende eller prediktion om beteende.
- Förbättra modellen.

Gustafsson tar inte upp problematiken med verifiering och validering. För att förstå problemen kring utvecklingen av datorgenererade styrkor är det viktigt att förstå att det idag inte finns vare sig tillräcklig kunskap om eller bra riktlinjer för någon av Gustafssons åtta punkter. Dessutom är det som alltid syftet med modell användningen som avgör kraven på vad modellerna ska klara av.

2. Problemen

Att utveckla datorgenererade styrkor och intelligenta agenter med realistiskt beteende är mycket svårt och komplext. Den inneboende icke-lineariteten och mängden möjliga mänskliga beteenden är i grunden orsaken till problemen vid utveckling och validering av modeller av mänskligt beteende. De mänskliga beteenden och regelbundenheter man vill beskriva spänner över vad Andersson (1992) kallar ”seven orders of magnitude” och täcker in allt från neurologiska fenomen som utspelar sig på tiotals millisekunder till sociala beteenden där hundratals timmar är en lämplig tidsskala att använda.

Ett grundläggande problem när man skapar datorgenererade styrkor och intelligenta agenter är därför att avgöra vilka kognitiva förmågor som agenterna behöver i olika situationer och tillämpningar. Att sedan avgöra hur detaljerade modeller man behöver för att tillfredställande efterlikna mänskligt beteende medför sedan också mycket svåra avvägningar. Eftersom denna typ av utveckling är mycket tidskrävande kommer bra modeller att kunna utvecklas endast för de absolut mest nödvändiga faktorerna och kognitiva funktionerna.

För att avgöra vilka delar av den mänskliga kognitionen och vilket beteende som behöver modelleras måste först en behovsanalys göras, d.v.s. vilken typ av frågor ska man studera och vilka krav det ställer på modellen. När man så bestämt vad som ska modelleras genomförs en uppgiftsanalys som är antingen beteendeorienterad (d.v.s. fokus på handlingar/beteenden som agenten utför och som kan observeras utifrån) eller mer processororienterad (d.v.s. tydligare fokus på kognitionen hos den operatör som modelleras). Här finns det ett antal olika intressefokus hos de personer som arbetar med utveckling av datorgenererade styrkor vilket beskrivs i Castor, Wallin, Nilsson och Moradi (2002). Detta påverkar hur detaljerade modeller man vill utveckla. En person som forskar kring kognitiv modellering har ofta andra krav på en modell än en utvecklare som ska ta fram beteendet på fienderna i ett dataspel, trots att de både säger sig hålla på med datorgenererade styrkor.

Bristen på existerande och tillfredställande modeller av belastning och prestation påpekas bl.a. i Castor m.fl. (2003) och är ett grundläggande problem för bl.a. CGF-utvecklingen, åtminstone om man vill ha modeller av faktiskt beteende. Det finns helt enkelt för lite data och för få konceptuella modeller. Därför är ofta de Performance Modifier Functions (PMF) (Silverman m.fl., 2001) eller Behavior Moderators (Pew & Mavor, 1998) som används grundade på

relativt lösa antaganden. Mer information om PMF:er finns bl.a. i Lundin (2004) och en omfattande lista på PMF:er ges i Silverman m.fl. (2001).

Inför en modelleringsuppgift kan man därför bli tvungen att utgå från de modeller man hittar i litteraturen, vilka sällan tagits fram för precis den plattform eller situation man intresserar sig för. Därför är det viktigt att information om modellens generaliserbarhet och ursprung följer med in i agentens dokumentation. Här behövs alltså någon sorts klassificeringssystem som visar hur giltig modellen är. Anta att vi vill ha en modell av hur en militär pilot visuellt av söker flygplanets presentationsytor och omvärlden. Modeller av följande typ skulle kunna hittas i litteraturen eller tas fram:

1. En påhittad modell av visuell avsökning utvecklad av en pilot som beskriver hur han/hon tror att piloter av söker skärmen.
2. En modell av visuell avsökning, men baserad på mätningar gjorda i ett kärnkraftsverk
3. En modell av visuell avsökning under flygning, men baserad på mätningar från civila flygplan.
4. En modell av visuell avsökning under flygning, baserad på mätningar från militära flygplan, men med annan typ eller mängd av presentationsmöjligheter.

Frågan är om alla, några eller ingen av dessa modeller är bättre än ingen modell. Och om man väljer att ta in någon av modellerna ska detta finnas med i kommentarerna av beteendet, så man senare i valideringsarbetet kan se vad som är ”hack” för att få agenten att fungera och vad som är baserat på någon sorts empirisk undersökning.

I NATO RTO-rapporten ”Human Behavior Representation (NATO RTO SAS017, 2001) görs en enkel uppdelning i fyra nivåer av modellers verklighetstrogenhet (fidelity):

- Mycket låg verklighetstrogenhet. Modellen beskrivs i tabellformat, typ Excel, med bedömningar av hur det som modellerats påverkas av olika faktorer.
- Låg verklighetstrogenhet. En ”svart låda” som tar indata och producerar utdata, men utan några krav på agentens kognitiva processer. Består ofta av matematiska modeller som har s.k. ”face validity”, dvs en expert har bedömt de som realistiska. Kallas ibland ”performance models” eller beteendemodeller.
- Medelgod verklighetstrogenhet. ”Svart låda” med ”face validity” och med viss explicit processmodellering.
- Hög verklighetstrogenhet. Processmodeller som explicit beskriver övergången mellan indata och utdata och efterliknande av hur mänskliga operatörer kommer fram till sina resultat. Kallas ibland ”functional models” eller kognitiva modeller.

NATO-gruppen som skrev rapporten diskuterar även vilken nivå av modeller som behövs för olika tillämpningar. Se tabell i Appendix 1.

2.1 ”Knowledge acquisition”

Professor John Laird är med stor säkerhet en av de tio personer i världen som har störst erfarenhet av CGF-utveckling och kognitiv modellering. Han har varit intensivt involverad i

utvecklingen av modelleringsarkitekturen SOAR och den kanske mest komplexa CGF-tillämpningen, TacAirSOAR, och många andra modeller. I Pearson och Laird (2004) anges att Lairds erfarenheter är att 75-90% av tiden och resurserna i ett mer komplext modelleringsprojekt går åt för att modelleringsgruppen ska bygga upp taktisk och uppdragsspecifik kunskap, dvs. vad det är den mänskliga operatören gör och hur. Denna kunskapsuppbyggnadsfas av ett modelleringsprojekt kallas ofta "knowledge acquisition" eller "knowledge elicitation".

Pearson och Laird beskriver hur en typisk kunskapsuppbyggnadsfas ser ut:

1. Utvecklingsgruppen går igenom litteratur som beskriver relevanta delar av domänen och de kognitiva funktioner som ska modelleras.
2. Genom intervjuer med en domänexpert, t.ex. en pilot eller bataljonsstabsmedlem, får utvecklingsgruppen en översikt över domänen och specifika beskrivningar av scenarier, beslut och handlingar som de datorgenererade styrkorna måste kunna hantera.
3. En prototyp av kunskapsdatabas/regelsamling utvecklas
4. En första utvärdering av regelsamlingen görs av domänexperten.
5. Fortsatta utvecklingscykler där kunskapsingenjörerna lägger till fler beteenden som kontrolleras av domänexperten.
6. Validering av den färdiga modellen av flera domänexperter.
7. Fortsatta tillägg och underhåll av modellen för att kunna klara av nya uppgifter.

Pearson och Laird anger att steg 3, 5 och 7 oftast är de dyraste faserna av utvecklingen, när modelleringsgruppen försöker omsätta vad domänexperterna berättar till en strukturerad och heltäckande regelsamling.

Kunskapsuppbyggnaden är alltså oftast en tidskrävande och kostsam process och i NATO RTO-rapporten "Human Behavior Representation" (NATO RTO SAS017, 2001) beskrivs ett antal anledningar till detta (ordningen nedan innebär ingen rangordning):

1. Experter är ofta omedvetna om vilken kunskap de använder för att lösa aktuella problem. Mycket av deras kunskap är också svår att verbalisera. Detta gör det svårt för modelleringsgruppen som försöker beskriva deras kunskap.
2. De metoder som används under kunskapsuppbyggnaden är ofta inte så effektiva och resultaten är ofta svåra att återanvända. Modelleringsgruppen kan oftast enbart använda intervjuer för att låta domänexperten beskriva världen och resultaten kan vara omständliga och inte direkt passa in i modelleringsarbetet och regelutvecklingen.
3. Aktiviteterna hos de operatörer som beskrivs är ofta mycket kontext- och uppgiftsspecifika. Det gör att reglerna för det beteende som beskrivs i stor utsträckning är knutna till existerande system och uppgifter på en specifik plattform. Beteendet med en annan plattform är inte nödvändigtvis detsamma och det innebär att kunskapsuppbyggnadsfasen måste göras om.

4. Domänexperterna har ofta svårt att engagera sig så lång tid som behövs eftersom t.ex. piloter har många andra uppgifter utöver CGF-utveckling. Ibland kan domänexperter också vara motvilliga att delta eftersom de upplever att arbetet med att utveckla en intelligent agent som utför arbetet kan göra dem överflödiga i framtiden.
5. Frågan om vem som egentligen är en domänexpert är ofta inte heller helt lätt att besvara. Ofta finns det olika meningar bland tillgängliga experter om vad som är rätt beteende. Ibland kan det t.o.m. vara bättre att använda en domänkunnig person som ännu inte helt bemästrat domänen eftersom en sådan person kan ha lättare att verbalisera vad han/hon gör.

2.2 Verifiering och validering

Som Juarez-Espinoza & Gonzalez (2004) påpekar är det inte tillräckligt med att bygga en modell av *en* beslutsfattare. En agent baserad på modellen måste också producera data som kan valideras mot andra empiriska data. Det måste också finnas strukturerade metoder för hur denna validering ska gå till.

Wallace (2003) pekar ut ett antal anledningar till varför det kan vara mycket svårt att avgöra om en avvikelse från ett förväntade beteendet är något som måste korrigeras:

- En skillnad mellan två beteenden betyder inte nödvändigtvis att det ena är fel. Agenter med den komplexitetsnivå som är aktuell här opererar i en verklighetstrogen simulering och denna typ av simuleringar innehåller en stor mängd potentiella lösningar för att nå existerande mål. Till exempel så behöver inte en observation att en expert och en nybörjare väljer olika handlingar för att nå samma mål betyda att det är fel i regelsamlingen.
- De kriterier som man använder för att avgöra om ett beteende är rätt kan förändras under problemlösningen och är beroende på vilka överordnade mål agenten har. Komplexa uppgifter kan innehålla många enkla problem. Varje delproblem kan ses som en uppgift i sig och därför kan kriteriet för att avgöra vad som är rätt beteende vara unikt inom ramen för det övergripande problemet. Det här betyder att de metoder man använder för att upptäcka fel i beteenden måste vara tillräckligt anpassningsbara för att vara användbara i olika typer av situationer och scenarier.
- Den aktuella kontexten när agenten väljer beteenden påverkar vad som är rätt val. Till exempel kan det finnas en stor mängd tillgängliga handlingar när en pilotagent ska välja handlingar i en strid där bara några få flygplan är inblandade. Men om agenten är en del av ett större koordinerat anfall kan det kraftigt begränsa antalet lämpliga alternativ. De hänsyn som agenten måste ta till kontexten måste därför programmeras in och det ökar mängden regler som måste utvecklas.
- I icke-deterministiska situationer har handlingar inte alltid den effekt som avses. Alltså kan en handling som var ”rätt” ändå resultera i ett oväntat utfall. För att kunna klara sig i dessa situationer måste agenterna ha tillgång till ett flexibelt och varierat regelbibliotek. Detta gör felsökning av regelsamlingen ännu svårare. En svår avvägning måste göras av hur många situationer agenten ska kunna klara och hur mycket energi som kommer krävas för att validera beteendet.

3. Mall för beteendedata för CGF-utveckling

Som det noteras i Archer m.fl. (2003) så finns det väldigt få CGF-tillämpningar som använder data som beskriver faktiskt mänskligt beteende. I stort sett alla CGF:ers beteende baserar sig på gällande doktriner och hur domänexperterna anser att beteendet borde se ut. Ofta är det just det beteende som är rätt enligt doktrinen som man vill att agenterna ska uppvisa, men man måste då vara medveten om att det inte är en simulation av hur människor i kritiska situationer faktiskt beter sig.

Anledningen till att data om faktiskt mänskligt beteende inte används mer är dels att grunddata och konceptuella modeller av hur det fungerar fortfarande saknas, men också för att de data som samlas in inte är specifikt insamlade för CGF-utveckling. Agentutvecklaren har därför mycket svårt att koppla olika begrepp som mental arbetsbelastning, rädsla och utmattning till specifika beteenden och förändringar i agentens uppträdande.

En modell är alltid en förenkling av verkligheten och eftersom mänskligt beteende är så komplext måste CGF-modeller vara kraftiga förenklingar av hur människor beter sig. Mycket kan sägas om hur kraftiga förenklingar man kan tillåta sig att göra men för området kognitiv modellering argumenterar John Anderson i sin "Rules of the Mind" (1993) övertygande att kognitiva funktioner kan beskrivas som målstyrt beteende på symbolisk nivå implementerat med produktionsregler.

Detta sätter vissa krav på insamlingen av beteendedata. Alla data som beskriver belastning och prestation och alla uppgiftsanalyser måste kunna omvandlas till (och helst redan vara beskrivna som) if-then satser som explicit beskriver tillståndsövergångar, konsekvenser och gränsvärden.

I ett försök att tydligare koppla beteendedata, både uppgiftsanalyser och data om belastning och prestation, till CGF-utveckling presenteras i Appendix 2 en mall som kan användas för att beskriva beteenden på taktisk nivå.

Mallen är baserad på uppgiftsanalysformalismen Hierarchical Task Analysis with decomposition tables (HTA) som beskrivs i Kirwan och Ainsworth (1992), en analys av vad en avancerad agent i en militär simulering bör hålla reda på (Avraamides & Ritter, 2002), information från MSIACs nätsidor (se nedan) samt projektets erfarenheter av prototyputveckling.

Bland Modeling and Simulation Information Analysis Centers (MSIAC, 2001) nätsidor ges följande rekommendationer om vad en kognitiv modell bör innehålla:

- Uppgifter som agenten måste kunna hantera.
- Objekt och egenskaper hos dessa objekt som agenten kan uppfatta.
- Objekt och egenskaper hos dessa objekt som agenten kan påverka genom sina handlingar.
- Situationer som identifierar när agenten måste förändra sitt beteende.
- Mål kopplade till varje uppgift.
- Situationer som identifierar när agenten måste ändra sina mål.

- Effekten av interna faktorer som kan påverka agentens beteende (t.ex. trötthet, stress, skador).
- Kunskap som agenten måste ha för att välja rätt handling i uppkomna situationer.
- Information om de olika typer av operatörer som simuleras och vilka roller de har.
- Skillnader i kunskap och personlighetsdrag som olika operatörer har.
- Kunskap som de simulerade personerna kan använda för att underlätta interaktionen mellan varandra och med verkliga personer.

Observera att den mall som presenteras är skriven för en taktisk grundmanöver för en enskild plattform och innehåller en del hypotetisk information (framför allt under avsnittet PMFs Influencing). För att dessutom beskriva hur dessa beteendepremitiver relaterar till varandra behövs någon uppgiftsanalysformalism som HTA eller eventuellt UML (Unified Modeling Language). Eventuellt behöver mallen utvecklas vidare om den ska användas för att beskriva t.ex. en stabsmedlems planerande av ett kompanis strid, men i princip bör den vara användbar. Mallen är på engelska för att kunna användas i det internationella samarbete som projektet är iblandat i, bl.a. NATO RTO-gruppen HFM 128.

Som kommenterats tidigare i inledningen råder det en stor brist på tillräckligt detaljerade modeller av de kognitiva funktioner som oftast är intressanta och en datainsamlare kan lätt reagera på mallen och säga att insamlade data inte kan beskriva beteenden med den precision som anges i mallen. Kognitiv modellering är dock till sin natur utan tolerans mot vaghet. I slutändan måste modellen eller uppgiftsanalysen explicit beskriva vad som ska hända i alla situationer och tillstånd (interna i agenter och externa i världen) som kan uppkomma och hur övergångarna mellan dessa tillstånd fungerar. Om detta inte beskrivs måste programmeraren som skriver agentens regler i slutändan göra en mängd ”fulhack” för att få modellen att snurra. Så även om en datainsamlare eller domänexperten inte heller riktigt vet hur det fungerar så är det bättre att de gör en gissning än att programmeraren ska göra det. Vikten av att reglerna som styr agenterna tas fram genom en iterativ process och i nära samarbete med domänexperten framgår här tydligt.

4. Diskussion

Enligt de diskussioner som rapportförfattarna deltagit i med NATO RTO-gruppen ”HFM 128 Representation of Human Behavior in Constructive Simulation” (personlig kommunikation NATO RTO HFM 128, 30 mars, 2004) så är de frågor som berörs i denna rapport de som utgör den besvärligaste flaskhalsen för CGF-utveckling i dag. Flera mycket erfarna gruppmedlemmar menar att området i stort sett stått stilla under flera år beroende på att det är så dyrt och komplext att beskriva alla beteenden som en komplex agent behöver.

Det finns alltså ett mycket stort behov att närmare studera och utveckla verktyg för att underlätta beskrivningen av nödvändiga beteenden och att kunna beskriva hur verkliga operatörer faktiskt agerar i de scenarier man försöker simulera. Även om de rena arkitekturfrågorna generellt sett inte är lösta så börjar det finns en stor mängd arkitekturer som passar olika syften och krav på simuleringarna. För en sammanfattning där huvuddelen av de idag aktuella arkitekturerna beskrivs, se Castor m.fl. (2002).

Några exempel på lösningar som är under utveckling för att underlätta regelutvecklingen är det användargränssnitt som finns i både SAABs TACSI (SAAB, 2004) och i Pitches Agent Factory (PITCH, 2003). Ett mycket intressant exempel är också det verktyg som Pearson & Laird (2004) beskrev på senaste BRIMS-konferensen. Verktuget REDUX som de presenterade kan användas för att grafisk beskriva beteenden som sedan ”automatiskt” ombildas till regler exekverbara i SOAR-arkitekturen. I artikeln beskriver Pearson och Laird ett exempel där domänexperter i REDUX i grafisk form kan beskriva hur soldater genomsöker och rensar byggnader vilket avsevärt underlättade utvecklingen av agenter med tillfredställande beteende.

Försvarsmakten bör satsa på att bygga upp referensbiblioteket för beteende och kognitiva modeller. För att underlätta uppbyggnaden av detta referensbibliotek bör FM titta närmare på verktyg som REDUX och implementationsoberoende representationsspråk, kanske det Agent UML som föreslås i Taylor & Wray (2004).

När grunden för CGF-utvecklingen är lagd, d.v.s. val av arkitektur, process för kunskapsuppbyggnadsfasen, process för grundläggande verifiering och validering kommer det att behövas särskilda modellutvecklingsprojekt, som i all annan utveckling av referensbibliotek. Om man vill ha modeller med någon sorts kognitiv validitet och därmed större chans att vara skalbara kommer det att kosta och kräva särskilda projekt, d.v.s. Projekt Utveckling Bataljonschefmodell eller Stridvagnsbesättningsmodell. Beställningarna på modellutveckling av jaktrobotXX till referensbiblioteket för robot och flygplansmodeller som FM byggt upp kan fungera som exempel. Väl värt att notera är dock att utvecklingen av dessa modeller tar lång tid och att det därför är hög tid att börja bygga upp detta referensbibliotek. Frågan är dock var beställningsansvaret för denna typ av modeller ska ligga inom Försvarsmakten.

5. Referenser

Anderson, J. (2002). Spanning seven orders of magnitude: a challenge for cognitive modelling. *Cognitive Science*, Volume 26, Issue 1, sid 85-112.

Archer, R., Archer, S., & Warwick, W. (2002). Synergy between Human Systems Integration and Training to Improve CGF Realism. I *Proceedings of BRIMS 2002*, sid 267-273.

Avraamides, M., & Ritter, F. (2002). Using Multidisciplinary Expert Evaluations to Test and Improve Cognitive Model Interfaces. I *Proceedings of BRIMS 2002*, sid 553-562.

Castor, M., Wallin, N., Nilsson S-Å., & Moradi, F. (2002). Datorgenererade styrkor - Metoder och möjligheter. FOI-R--0688--SE.

Castor, M., Hanson, E., Svensson, E., Nählinder, S., Le Blaye, P., MacLeod, I., Wright, N., Alfredson, J., Ågren, G., Berggren, P., Juppet, V., & Ohlsson, K. (2003). GARTEUR Handbook of Mental Workload Measurement. Final report from GARTEUR Flight Mechanics Action Group FM AG13. GARTEUR TP 145.

DMSO (1995). Modeling and Simulation Master Plan. Defence Modeling and Simulation Office.

Gustafson, S. (2002). Modeling the Terrorist Decision of a Closed State Regime. I Proceedings of BRIMS 2002, sid 465-472.

Juarez-Espinoza, O., & Gonzalez, C. (2004). Situation Awareness of Commanders: A Cognitive Model. I Proceedings of BRIMS 2004, sid 53-62.

Kirwan, B., & Ainsworth, L. (Eds.) (1992). A guide to task analysis. London: Taylor & Francis.

Lundin, M. (2004). Simulating the effects of mental workload on performance. LIU-KOGVET-D—04/xx—SE, Linköping Electronic Press, Sweden.

MSIAC (2001). Human Behavior Representation Validation. http://www.msiac.dmsomil/vva/Special_topics/HBR-validation/hbr-validation.htm. Besökt 2004-08-12.

NATO RTO SAS017 (2001). Human Behavior Representation. NATO RTO Technical Report 47, SAS Panel.

Pearson, D., & Laird, J. (2004). Redux: Example-Driven Diagrammatic Tools for Rapid Knowledge Acquisition. I Proceeding of BRIMS 2004, sid 185-192.

Pew, R., & Mavor, A. (1998). Modeling Human and Organizational Behavior: Application to Military Simulations. National Research Council.

Pitch Kunskapsutveckling AB (2003). Agent Factory. CGF programvara.

SAAB (2004). TACSI. CGF programvara.

Silverman, B., Johns, M., Shin, H., & Weaver, R. (2000). Performance Moderator Functions for Human Behavior Modeling in Military Simulations. <http://www.seas.upenn.edu/~barryg/PMFset.zip>. Besökt 2004-08-12.

Taylor, G., & Wray, R. (2004). Behavior Design Patterns: Engineering Human Behavior Models. I Proceeding of BRIMS 2004, sid 89-96.

Wallace, S. (2003). Validating Complex Agent Behavior. Doktorsavhandling vid University of Michigan.

6. Appendix 1.

Nedan presenteras en del av en tabell från NATO RTO rapporten Human Behavior Representation (NATO RTO SAS 017, sid 68). Här återges bara den del av tabellen som anger krav på beteendemodeller för enskilda plattformar eller förband på taktisk nivå, inte modellering av höga politiska nivåer.

<u>Tillämpning</u>	<u>Krav på modellens nivå</u>	<u>Kommentar</u>
Utbildning / Träning Individer	Blandat (Låg till Hög)	
Utbildning / Träning Grupper	Blandat (Låg till Hög)	
Övningar	Blandat (Låg till Hög)	Oklart vad rapportförfattarna menar med övning.
Materielanskaffning – Konceptgenerering	Mycket Låg	
Materielanskaffning – Val mellan koncept	Låg / Hög	
Materielanskaffning Konceptförädling	Hög	
Materielanskaffning – test och utvärdering	Hög	
Organisationsförändringar	Hög	
Uppdragsplanering	Blandat (Låg till Hög)	
Förövning	Blandat (Låg till Hög)	

“Mission Rehearsal”

Planering av logistik och underhålls-operationer (tex biståndsaktion)

Anges inte i den del av tabellen som återges här, men bör vara låg om det är aktuellt.

Planering på långsiktig strategisk nivå (tex förutsäga säkerhetspolitisk utveckling och krav)

Anges inte i den del av tabellen som återges här, men bör vara låg om det är aktuellt.

Planering på höga taktiska ledningsnivåer (tex ledning NATO koalition)

Låg

Informations Operationer / Asymmetrisk Krigföring

Hög

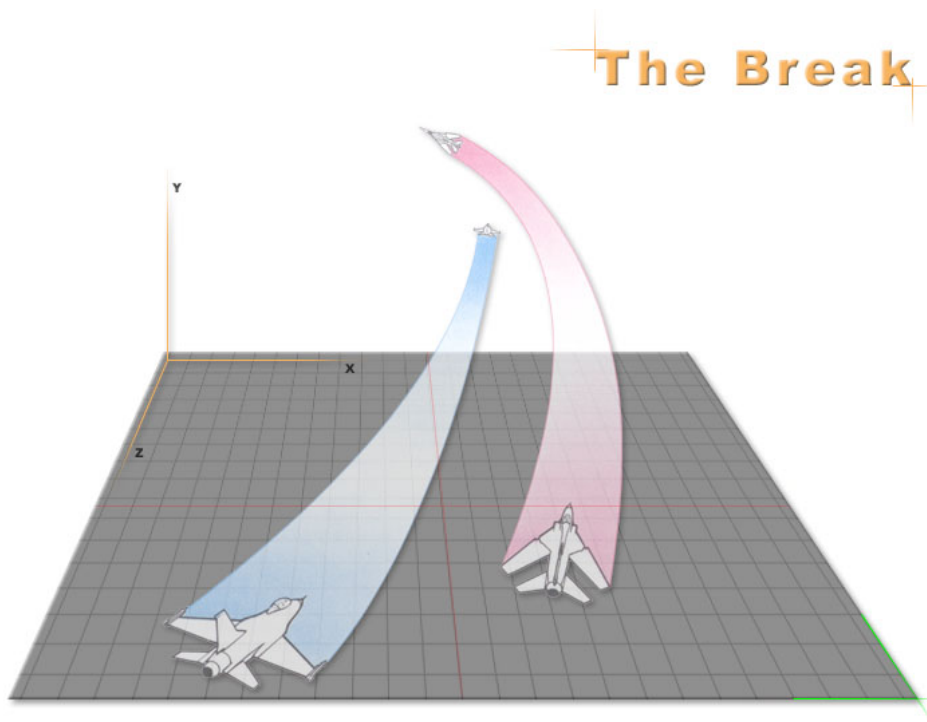
7. Appendix 2.

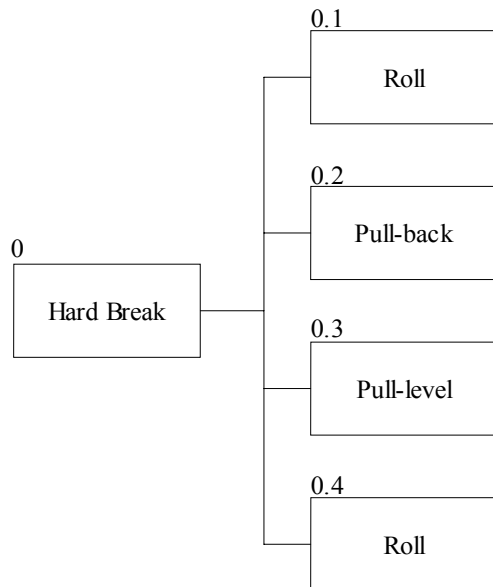
Tactical defensive manoeuvres**The Hard Break**

This manoeuvre is used when an attacker in within visual range combat (WVR) is first seen approaching or is already in the cone of vulnerability. Its purpose is twofold: to spoil the attacker's aim and to force him to overshoot. The break is always made towards the direction of attack; this generates "angle-off" as quickly as possible which makes the defender a difficult target. The attacker may be able to cut inside the turn but he is forced to pull lead. To do this he must tighten his turn, which increases his angle of attack. It is difficult for him to pull his nose around at high angles of attack to achieve a firing solution. The defender should also alter his plane of flight to make himself a more difficult target.

Two forms of break are possible, depending on the circumstances of the attack. The defender can use a maximum-rate sustained turn in which he does not lose speed, or the hardest possible turn in which he almost certainly does. The speed loss attendant on the hard turn aids his chances of forcing the attacker to overshoot, as does the smaller radius of turn, but oft-quoted maxims such as "speed is life" act as an inhibitor. If the break succeeds in forcing the attacker to overshoot, the next manoeuvre is the Scissors.

The break is a life-saving manoeuvre. It is used against an attacker who is about to achieve a firing position (or already has). It consists of a hard turn into the direction of attack, to generate angle-off as rapidly as possible to present the most difficult target.



Goal hierarchy**Plan 0:**

Do 0.0 until roll desired roll angle is achieved then do 0.1 until desired heading is achieved do 0.3 and 0.4 simultaneously until roll angle is 0°

Task decomposition

Task no	HBREAK-0 (Hard break)
Function	To keep the attacker out of the “cone of vulnerability” (COV) and to force the attacker to overshoot.
Context	May be initiated during defensive manoeuvres in a dogfight.
Validity	This manoeuvre has been developed from knowledge extracted from AF tactics manual XYZ. An expert pilot has inspected the rule set.
Decisions required	When the manoeuvre should be initiated (see Initiating events & Initiating decisions). Amount of roll and change of heading. When the manoeuvre should end.
Initiating events	<p>All of the following:</p> <ol style="list-style-type: none"> 1. Attacker is within COV <i>Attacker has a strong position of firing</i> 2. Attacker has a positive closing speed <i>Attacker is faster and will probably overshoot during a break, if</i>

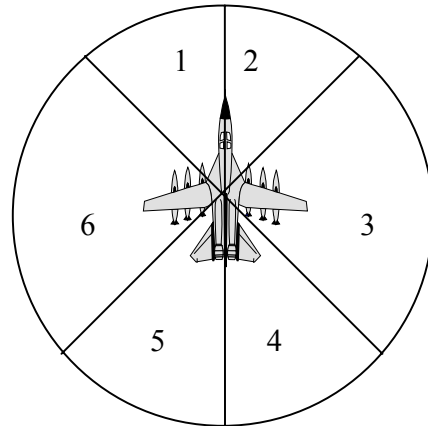
	<p><i>attacker is slower he will probably remain in COV</i></p> <ol style="list-style-type: none"> 3. Attacker DTG (Degrees To Go) is low <i>Attacker is not flying away from the defender</i> 4. Speed is relatively high <i>Defender should not end manoeuvre with too low speed</i>
Concluding events	<p>One of the following:</p> <ol style="list-style-type: none"> 1. Defender has achieved the break direction <i>The break has caused the aircraft to change direction</i> 2. Attacker is overshooting <i>Attacker has lost its attacking position</i>
Initial observations	<p>All of the following:</p> <ol style="list-style-type: none"> 1. Position of attacker <i>To decide if attacker is within COV</i> 2. Speed of attacker and defender <i>To calculate closing speed of attacker</i> 3. DTG of attacker <i>Needed to decide if attacker is flying nearly parallel</i> 4. Initial heading of defender <i>To set up a direction to complete the break</i>
Continuous observations	<p>All of the following:</p> <ol style="list-style-type: none"> 1. Current heading of defender <i>To decide when to finish break</i> 2. DTG of attacker <i>To decide if attacker is overshooting</i> 3. Speed of attacker <i>To decide if attacker is overshooting</i> 4. Position of attacker <i>To decide if attacker is overshooting</i>
Initial decisions	<p>All of the following:</p>

	<ol style="list-style-type: none"> 1. If attacker is within COV <i>(see implementation details below)</i> 2. If closing speed of attacker is positive <i>(see implementation details below)</i> 3. If DTG of attacker is low <i>(see implementation details below)</i> 4. Break angle <i>(see implementation details below)</i>
Continuous decisions	<p>All of the following:</p> <ol style="list-style-type: none"> 1. If attacker is overshooting <i>(see implementation details below)</i> 2. If break direction has been achieved <i>(see implementation details below)</i>
Initial Actions	<p>All of the following:</p> <ol style="list-style-type: none"> 1. Set up and remember break angle <i>To be able to exit break when direction has been achieved</i>
Continuous actions	Stick input. Variability in input is +- X degrees of break angle. Altitude deviation is +- Z meters.
Controls used	Stick and throttle.
Displays used	Head-up 100% to keep visual contact with attacker to estimate speed, direction, etc. (see implementation details).
Skills	Perception (to estimate opponents speed etc), manoeuvring & platform handling.
Experience needed	The hard break is a basic manoeuvre and is available to agents of all "experience levels".
PMFs influencing	The performance modifier function "Mental workload" affects the possibility to keep track of opponents. When heart rate exceeds valueX (approx 115) the chance to lose the opponent is 1% per second when the agent does not have eyes focused on the opponent. No working memory update of other targets than the target that was the reason for the break is performed during the manoeuvre. The PMF "Fatigue" affects the maximum G-force and duration of the break that is tolerated.
Priority	Highest priority, except missile or collision avoidance

Possible errors	Estimation errors can cause the break to fail
Time	Decisions 1-2s
Comm-unication	No communication needed
Preceding tasks	Any
Super-task	Any
Succeeding tasks	If attacker is decided to overshoot it is possible to start with scissors
Sub tasks	Roll, Pull-back
Concurrent tasks	None
Notes	

Implementation details

To be able to decide whether the attacker is within the Cone of Vulnerability (COV), the agent must first decide if the attacker is close. Estimates could be very close, close, far, very far depending on visual or radar information. The agent must also decide whether the attacker is behind. This could be done by dividing the space into different volumes. Typical values could be 30° angle of the tail (AOT) in each direction (region 4 and 5) and a distance of 10km. The same regions could be used as the current eye point of gaze.

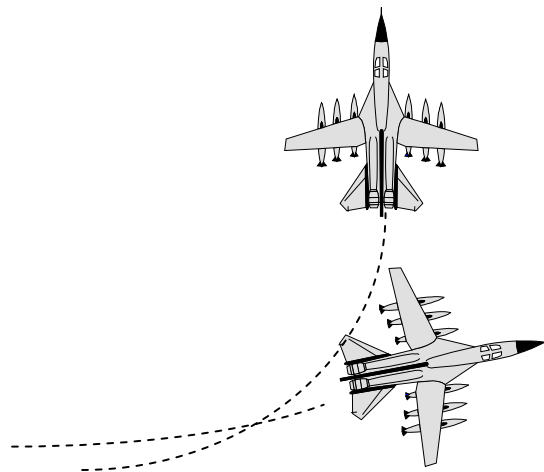


Closing speed could be a relative estimate between the speeds of both aircraft such as attacker is slightly faster or slower.

DTG of attacker could be calculated from the difference between the heading of both aircraft. For example, if difference is less than 60° (and in COV) the attacker should be regarded as an immediate threat.

Break direction should be towards the direction of attack. This could be decided by looking at the DTG of attacker. A relative break angle of 70°-90° could be used.

As in real life, to decide if the attacker is overshooting is very difficult. A definite decision could be made if the attacker is just behind, with a greater speed and a LOS difference of more than 70-90 degrees. It also depends very much on difference of manoeuvrability.



A break manoeuvre that causes the attacker to overshoot (unless his speed is much lower than the defender's)

Example code (Soar)

Operator proposal:

Proposes a break

```
sp {all*propose*break
  (state <s> ^name dog-fight
    ^bogey <bogey>
    ^self.heading <heading>)
  (<bogey> ^cov yes
    ^relative-heading < 60 ## DTG
    ^relative-speed > 0)
  -->
  (<s> ^operator <o> + =)
  (<o> ^name break
    ^result (+ <heading> 90))}
```

Suboperator proposal:

IF roll angle is not 90 and heading is not achieved
THEN propose roll

```
sp {break*propose*roll
  (state <s> ^name break
    ^top-state.input-link <il>)
  (<il> ^heading <heading>}
```

```

^roll <roll-angle> <> 90)
  (<s> ^result <> <heading>
    ^dir <direction>)
    -->
  (<s> ^operator <o>)
  (<o> ^name roll
    ^dir <direction>)}

```

IF roll angle is 90 and heading is not achieved
THEN propose pull-back

```

sp {break*propose*pull-back
  (state <s> ^name break
    ^top-state.input-link <il>)
  (<il> ^heading <heading>
    ^roll <roll-angle>)
    (<s> ^result <> <heading>
      ^dir <direction>)
      -->
    (<s> ^operator <o>)
    (<o> ^name joystick
      ^y-dir 5)}

```

IF roll angle is 90 AND heading is achieved AND joystick y-dir not 0
THEN propose pull-level

```

sp {break*propose*pull-level
  (state <s> ^name break
    ^top-state.input-link <il>)
  (<il> ^heading <heading>
    ^roll <roll-angle>)
    (<s> ^result <> <heading>
      ^dir <direction>)
      -->
    (<s> ^operator <o>)
    (<o> ^name joystick
      ^y-dir 0)}

```

IF roll angle is 90 and heading is achieved
THEN propose roll-level

```

sp {break*propose*roll-level
  (state <s> ^name break
    ^top-state.input-link <il>)
  (<il> ^heading <heading>
    ^roll <roll-angle> 90)

```

```
(<s> ^result <heading>
    ^dir <direction>)
-->
(<s> ^operator <o> + =)
  (<o> ^name roll
    ^x-dir 0)}
```

Bilaga 3, Konferensbidrag

Simulating the effects of mental workload on performance in tankcrew

Mikael Lundin
 Swedish Defence Research Institute (FOI)
 SE-172 90 Stockholm, SWEDEN
 +46 (0) 8 555 035 44
 mikael.lundin@foi.se

Keywords:

cognitive modeling, behavior moderators, mental workload, tankcrew, Soar

ABSTRACT: *Battle tankcrew must perform many diverse tasks in their missions: navigation, system control, enemy engagement, and communication. As human capabilities are limited, the crew members will find themselves in situations where taskload and other factors raise their mental workload severely. In turn, such stress has documented effects on quantitative and qualitative changes in task performance that could lead to mission failure.*

This paper describes the process of simulating tankcrew in a mission where mental workload is a key factor to the outcome of performance. Mental workload is represented as a behavior moderator, which can be manipulated to demonstrate and predict behavioral effects. The cognitive models of the tankcrew members are implemented as Soar agents, interacting with tanks in a simulated battlefield. With high empirical validity as a main objective, the data acquired for the model was collected from experiments with tankcrew, including observations and task analyses. Afterwards, the simulation was verified and face validated with two Subject Matter Experts. The paper summarizes a Master's Thesis [1] in cognitive science.

1. Introduction

1.1 Background

Computer generated forces (CGFs) have several uses in the military domain. Most obvious is the need for agents in training simulators, having the role of either enemy or allied force. Several goals must be met regarding the characteristics of CGFs. From a military perspective, a high degree of realism is a fundamental requirement or training may be detrimental.

One aspect of realism is that human operators are limited in their response to taskload. In situations where the number of tasks is exceeding their capabilities, mental workload will rise. One example of where this occurs is among battle tankcrew. Apart from the changes in taskload, different tank commanders show a range of different performances in effectiveness and efficiency.

The tank-commander's task is described by Gross et al [2] in the following terms:

“Main battle tank-commanders must perform a great number of tasks in the course of their duties: vehicle navigation, mission planning, surveillance, target acquisition, system monitoring and control, communications, and crew supervision. The speed and accuracy with which commanders execute these tasks can mean the difference between mission success and failure and ultimately between life and death”

As a background to the work in the thesis, a problem definition can be formulated around the following questions rather than in a single hypothesis:

1. Which tasks are relevant for successful performance of tankcrew and which tasks are necessary to model in an agent?
2. Of these tasks, which are affected by high mental workload and when being affected: how does the crew's behavior change?
3. Knowing (1) and (2), how can a valid behavior representation of the tank crew be demonstrated as agents in a simulated environment?

The purpose of the thesis was to explore the possibility of making computer simulation models to help answer the questions in the problem definition, and to demonstrate the findings in a realistic simulation environment. In specific, this means to simulate the effect of mental workload on performance in tankcrew. Such simulation requires modules for mental processes which have to be embedded in a surrounding where physical behavior can be demonstrated. Hence, the purpose is to create artificial agents representing the tankcrew, whose cognitive models are affected by mental workload, and to create a scenario where the behavior differences can be visualized in a simulation environment.

1.2 Structure of paper

While modeling approaches often are motivated by new methods or theories, this paper takes off in the domain to be modeled: the working situation for tankcrew. Throughout the work, it's been an objective to remain grounded in the domain and in a specific scenario, rather than making any general assumptions about the human mind.

This paper continues with *behavior moderators* and in specific *mental workload*, giving a theoretical background for the study, followed by the *method*, including data collection and agent architecture. Afterwards, the *results* are presented in terms of the conceptual model and implementation, before finally *discussing* the validity of the model and future extensions to it.

2. Behavior moderators

When modeling CGFs, a number of cognitive processes must be taken in consideration, such as perception, attention, memory, and decision-making. These processes have historically formed the main body in cognitive psychology and artificial intelligence. But they aren't isolated and unaffected by other factors. Particularly relevant for military settings where commanders, soldiers and operators perform tasks under stress and in extreme situations, is the investigation of those factors that affect cognitive processes: behavior moderators.

Internal variables such as eagerness, aggression, and fear, as well as external variables such as noise, heat, and vibrations, are examples of behavior moderators. These factors change the cognitive processes which in turn result in changes of behavior. Behavior moderators are part of a solution where the original cognitive models are kept intact. In order to satisfy

new demands of realism, the original models are enhanced by the moderators.

Another approach to behavior moderators is to represent individual differences. Although this perspective encompasses the whole set of cognitive processes (e.g. working memory speed), behavior moderators are typically variables where individuals differ. For example fear, anger, and anxiety at different levels as well as sensitivity towards external stressors are often what form a personality.

3 Mental workload

3.1 Concept and measurement

In short, high mental workload is a psychological concept representing the stress an operator experiences due to the inability to cope with a difficult task and overload of information. The precise definitions are more diverse. According to O'Donnell [3] "The term workload refers to that portion of the operator's limited capacity actually required to perform a particular task". Gopher and Donchin [4] defines mental workload as "the difference between the capacities of the information processing system that are required for task performance to satisfy performance expectations and the capacity available at any given time". Hart and Wickens [5] define it as "the effort invested by the human operator into task performance." Theoretically grounded for over 20 years ago, mental workload has been studied mostly in relation to situation awareness, pilot performance, and cockpit design.

The level of mental workload an operator experiences can be measured by a multitude of methods which are used together in order to assure the validity.

Subjective ratings, where the operator judges his or her mental workload directly on a specific scale or judges different aspects of the concept. In the latter case the judgements are collected in an index [6].

Psychophysiological measures, which require that the physical reaction is related to the operator's way of psychologically meet the demands of the task. These include measures of the operator's pulse, the variation of the pulse, EEG-activity, the blink frequency of the eye, pupil dilation, and endocrine activation.

Qualified observations by experiment leaders who, though subjective, rates the participants and doing so with an experience of a multitude of experiments and participants, and without the potential fallacies of the subjects themselves.

3.2 Workload effects in tankcrew

As mentioned in the introduction, mental workload is an important moderator of performance in tankcrew. The working situation is characterized by constant decision-making under time pressure in a hostile environment with sophisticated threats. The operator's working space is characterized by narrowness and a multitude of instruments (Figure 1).

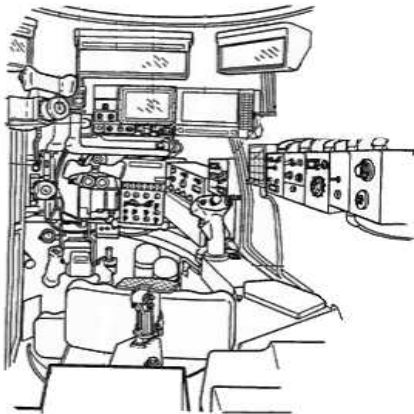


Figure 1 Working environment for the tank-commander in tank "Stridsvagn 122"

Given the complexity of the operator situation and the obvious sources for a high mental workload, how can good and bad performance be described? A certain performance must be put in relation to many surrounding factors, of which the performance of the enemy is crucial. Trying to fight an enemy is rational in some positions but irrational in others. Further the terrain and the task determine good behavior. Hence, being able to define and quantify a good performance is a difficult task.

In an interview with a SME, the following tasks were extracted where mental workload had a specific effect on performance:

- 1) While taking firing positions: A high mental workload will result in a less careful evaluation of the terrain, which leads to taking a worse firing position.
- 2) Without terrain reference points, two effects of an increase in mental workload follow when enemies are detected: The reaction time before the gunner has detected the target will increase, and the tank-commander misdirects the gunner by stating the direction from his own perspective
- 3) In the case where either the gunner or the tank-commander detects the target, and the mental workload is high, they forget to report the target before firing.

- 4) In the case where an enemy target has been fired at and hit, the crew which has a high mental workload keeps looking at this target, instead of keeping up observing.
- 5) The crew with high mental workload will fire at the closest target rather than evaluate the danger posed by all targets, and firing at the most dangerous.
- 6) A tank-commander with high mental workload misses putting the detected targets on the shared computer screen (LSS).

4. Method

4.1 Modeling approach

Given the theoretical insight about mental workload and its impact on performance, and the knowledge of how it applies to tankcrew, the aim was to build a model of a platoon whose crews are engaged in tasks where mental workload is critical to performance. Such model must include mental schemas for each of the individual tankcrew members, giving rise to the overall performance of the platoon.

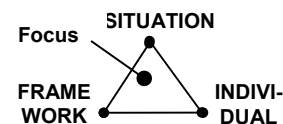


Figure 2 Dimension space of focus for cognitive/behavioral models

The triangle (Figure 2) illustrates the focus on behavior in a specific set of situations, rather than an "innate" human behavior representation (focus on individual). The model is generic to the extent the behaviors in each task are interoperable with other tasks. That is, the effects listed in the previous section are not stating something about innate human functioning. On the other hand, these behavioral effects may be possible to generalize to other situations as well (e.g urban battle or with helicopter targets). Further, the model mostly adheres to a specific framework (Soar) for the description of its tasks. The exception to this is the behavior moderator representing the level of mental workload, which naturally is modelled outside the framework.

4.2 Hierarchical Task Analysis

Hierarchical Task Analysis (HTA) is a task description technique which aims to collect a number of different goals and see how these correlate, in order to organize them in a common structure. HTA results in a hierarchy showing which tasks have to be performed, and in which order they must be performed, to reach goals that are higher in the hierarchy [8].

Based on the work of Lindström [9], two Hierarchical Task Analyses were the foundations the agent models. Beginning with “Critical Incident Technique” and later structured into a task hierarchy, Lindström’s analyses captures several important goals and sub-goals in the tankcrew’s working situation. The analyses also state the order of which tasks are to be performed, if not marked parallel (Figure 3).

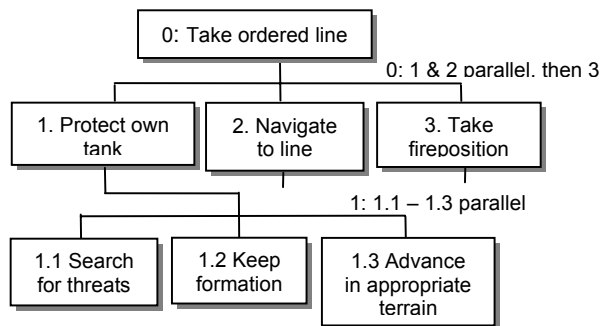


Figure 3 A snapshot from Lindström’s [8] HTA where the tasks for a tankcrew are structured in a hierarchy

4.3 Observations

The empirical data taken in consideration was recorded from an experiment on tankcrew in training simulators. Similar studies on pilots have shown strong correlations between measurements from simulators and real-world missions [6]. In the experiment, mental workload was measured on tank-commanders, gunners, and drivers, by using subjective and objective ratings as well as measures of heart rate and performance logs.

In this model only the data from the observation was used. The objective ratings were denoted by the experiment leaders and confirmed by simulation operators. By observing the crew, they monitored the experiment scenario and recorded during which tasks mental workload was rising and how behavior changed. This includes both dynamic states, as well as general tendencies.

4.4 Soar

The model was implemented in the symbolic architecture Soar [9]. This framework is based on production rules to create behavior. Two advantages have been noticed using a symbolic framework in collaboration with military commanders:

- It provides a logical description on a detailed level. Hence, the behaviors generated by the

model can be explained in more or less natural language for experts in the field. While a connectionist model in theory would be able to exhibit the very same behaviors, the internal structure would – even if accessible - be difficult to communicate in language with the military commanders and other SMEs.

- It’s compatible with Hierarchical Task Analysis. Based on goals and steps to reach these goals, Soar uses the same concepts as HTA to describe behavior. While not necessarily used on the same level of detail, the similarity in concept is an advantage while modeling.

Apart from these advantages and Soar’s success in providing reasoning abilities to the tankcrew representations, the Soar architecture embodies the principles of a candidate theory of human cognition [9].

5. Results

5.1 Conceptual model

The task analysis taking the form of a hierarchical tree was pruned so that 1) the tasks necessary for the agent to perform its mission, and 2) the tasks critical to different levels of mental workload, were kept. The conceptual models (Figure 4) were then designed as similar hierarchical trees of goals and sub-goals, describing cognitive and behavioral schemas, including those affected by mental workload. While the task analyses captured the goals of the tankcrew collectively, they did not describe important divisions of tasks among the crew members. For example, taskload on the tank-commander is notably higher than that of the tank driver. Other changes were made with respect to that the original HTA originated as descriptions of observed tasks, while the conceptual model aimed at being the schemas that resulted in the same tasks.

The effects of mental workload on performance can be seen as generated by selecting different paths in the goal tree depending on the workload level. With mental workload as single variable, it is included in the conditions that control the behavioral schemas (e.g. IF mental workload is low, THEN tell the gunner correct target position).

Mental workload is hence incorporated as a behavior moderator variable in the six situations (listed in section 3.2) where mental workload has an impact on performance. Further, workload is represented as a two-state value: HIGH or LOW. For example, when the tank-commander is to choose a firing position before engagement, mental workload affects the behavior.

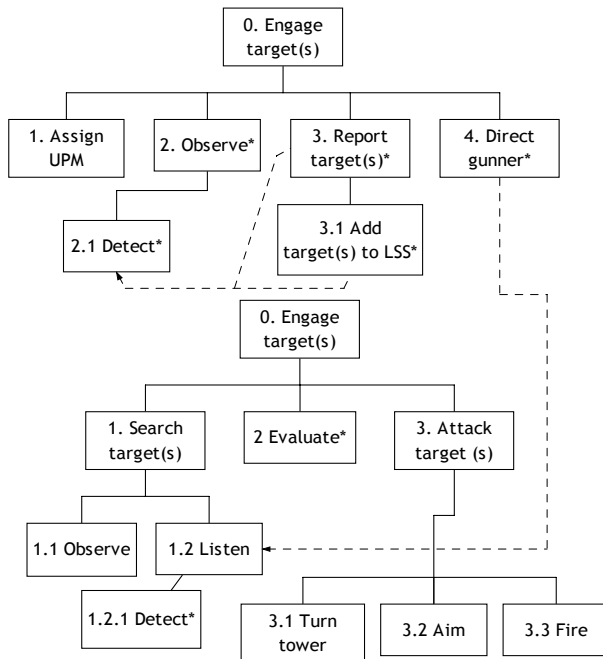


Figure 4 Second part of the conceptual model in the form of a hierarchy, representing the tasks from the fire-position to attack on targets. The upper and lower trees describe the behaviors of the tank-commander and the gunner respectively.

Then, LOW mental workload leads to a long evaluation of the terrain, protection, and field of fire, while HIGH mental workload leads to exclusively evaluation of distance to select the closest firing position. Although workload appeared as a continuum of levels in some of the empirical data, such accuracy was considered too difficult to model and validate.

5.2 Implementation

Each of the “boxes” in the conceptual model together with its conditions, are implemented with Soar production rules. Some boxes have a higher numbers of rules, representing more detailed behavioral schemas or more complex functionality, than other boxes. Figure 5 gives an example of two Soar rules, forming the agents’ long-term memory.

Implementation of some boxes are embedded in the functionality of Soar itself. For example, “Observe” and “Listen” is represented by Soar’s ability to fire any rule that match the current working memory state.

While the implementation and the simulation results are diverse in dimensions for each of the effects, they all describe a shift in behavior in a certain situation.

```

ext1
File Edit Search Soar Insert Template Runtime
# If we need a new turn and a turn
# operator isn't already proposed

sp {wp-nav*propose*decide-turn
  (state <s> ^name get-to-nextwp
    ^top-state.self <self>)
  (<self> -^wp-distance very-close
    -^wp-heading ok
    -^mission.orders.order.turn)
-->
  (<s> ^operator <o> + =)
  (<o> ^name decide-turn)
}

Line: 2 Modified

ext2
File Edit Search Soar Insert Template Runtime
# If we have HIGH mental workload,
# apply detection operator after 5 sec

sp {execute-mission*apply*detect-target
  (state |<s> ^name attack-target
    ^subgoal <sg>
    ^operator <o>
    ^delta > 5)
  (<o> ^name detect-target
    ^target <t>)
-->
  (<t> ^detected yes)
}

Line: 5 Modified

```

Figure 5 Examples of Soar production rules: The upper window contains one of a cluster of rules needed for way-point navigation. The lower window contains a single rule for detection of enemy targets where the effect of mental workload is represented by a 5 seconds delay for a stressed tank-commander. The time value is approximated.

This can be contrasted with more general claims of mental functioning, such as Jones’ et al [10] introduction of fatigue in TacAir-Soar. Although one of the effects of high mental workload for tankcrew is

delayed detection of targets, the phenomenon is modeled in the content of the architecture (i.e adding a timer to the detection rule). Jones, on the other hand, models fatigue by changing the very control structure: introducing a delay in Soar's decision-cycle, by which all rules are executed.

Table 1 Examples of in- and outputs of Soar agents representing different members of the tankcrew, and different types of variables.

Input	Agent	Type
Terrain node	Tank-commander	Percept
Target report	Gunner	Verbal message
Mental workload	Tank-commander	Behavior moderator

Output	Agent	Type
Target report	Tank-commander	Verbal message
Turn left	Driver	Action

The model can be viewed as a control structure which takes inputs (external percepts, verbal messages, behavior moderators) and generates outputs (motor behavior and verbal messages) (see Table 1). In Soar, in- and output always occurs on the ^io interface in working memory, independent of any specific problem space.

Each tankcrew member is incorporated into the system environment by linking the Soar models and the behavior moderator modules to a specification for the corresponding agent (Figure 5). This specification includes constant agent properties and behavioral

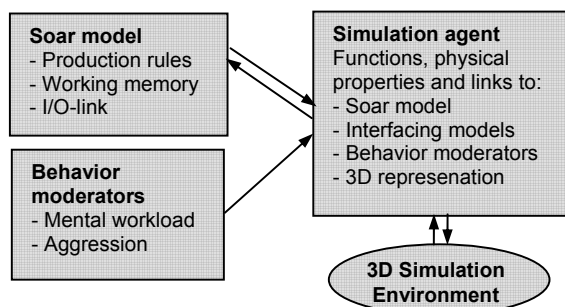


Figure 6 System integration of the Soar model and behavior moderators in the simulation environment. The mental workload is affecting the the Soar reasoning via links to the agent specification.

functions, as well as links to interfacing models (e.g vehicle dynamics- and operator interface models). It also includes a link to the 3D model which represents the tank in the simulation environment.

Mental workload level is represented in an external behavior moderator module which directly influences the reasoning in Soar, via the agent specification. In the current demonstration the workload level is fixed, meaning that tankcrew members have the same stresslevel throughout all critical tasks in the scenario. However, the level could easily be made dependent, either on simulated dependencies (e.g taskload) or system events (e.g. key pressed).

5.3 Demonstration scenario

The demonstration scenario is an assault mission for a tank platoon whose main objective is to engage and fight a set of enemy tanks. The mission can be described in the following general steps after orders have been given:

1. Navigate to a line of battle at the nearest hill as ordered
2. Choose a firing-position from where the enemy can be engaged
3. Engage and terminate threats when detected

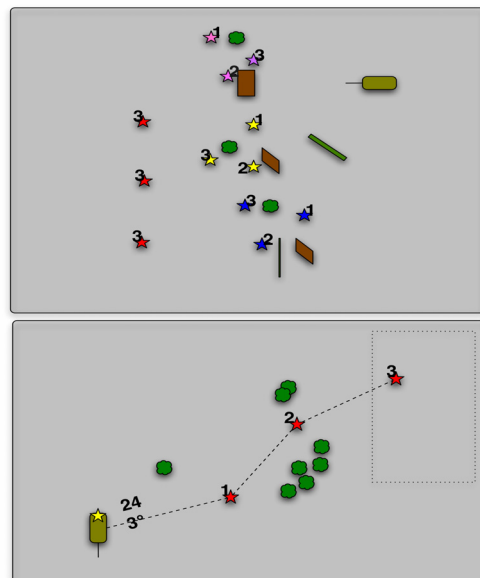


Figure 7 Birds-eye view of the simulation environment. Below represents the overall area for the mission with way-points marked as red stars. Above is the area around the last waypoint zoomed. The pink, yellow, and blue stars represents potential firing-positions.

The scenario takes place in a rather open terrain, where enemy forces are placed behind a hill. This is demonstrated in a 3D environment where the tank platoon navigates through the terrain. The cognitive models of the crew receive their inputs from the simulated world. Similarly, the behavior of the tanks is ultimately controlled by the crew, where the cognitive model of the platoon-commander is the highest executive.

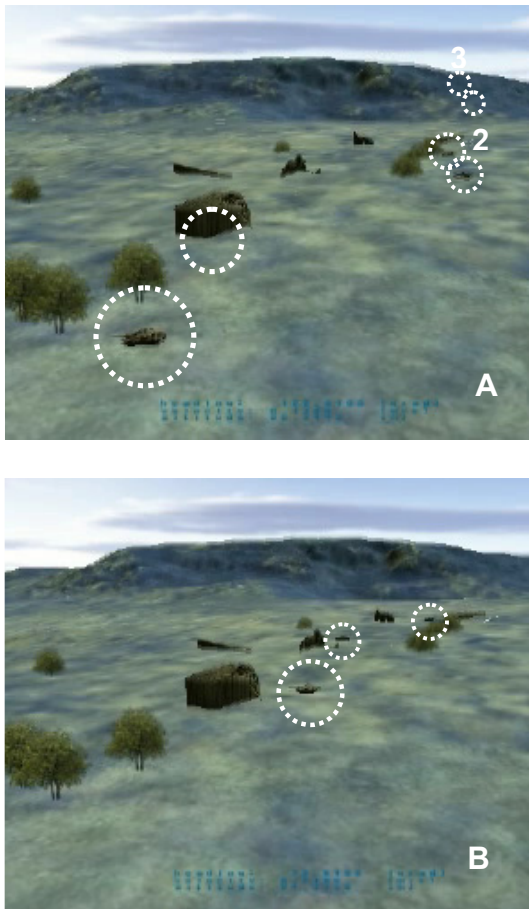


Figure 8 Example of simulation results: Entering from right in the images, the tank platoon is taking firing-positions (see upper part of Figure 7). The same situation is repeated with different values on the behavior moderator. In image (A), the crew has a high mental workload and chooses nearest possible firing-position. In image (B), the tankcrew's low mental workload leads to evaluation of the terrain according to the field manual. Consequently they choose better firing-positions.

6. Discussion

6.1 Validation and verification

In this work, the conceptual and implemented models were validated against external referents, while the implementation was verified against the conceptual model.

As described, the conceptual model is based on empirical data measured in a tank training simulator. The data comes in the form of objective ratings by a Subject Matter Expert (SME). Generally, the relationship of mental workload being negatively correlated to situation awareness, which in turn is positively correlated to tactical performance, had been established [11].

The six effects of mental workload mentioned in section 3.2 were not only observed by the SME, but also confirmed by the technical officers operating the tank simulator.

Several “desk checking tests“ during the running was then performed to verify the implementation against the conceptual model as well as assuring the continuity of the program flow:

- The simulation itself assured that no unexpected behaviors among the tanks occurred
- The simulation in combination with a “debug window” made sure that in each instance, the correct production rules were fired among the agents
- Stopping the program flow gave an instant view of the agents' working memory tree. The working memory content in these views where all correct with regards to the time when the simulation was stopped. The tree exposed:
 - o Which problem space is actual for the agent
 - o Which sensor information is available to the agent
 - o Which events have been remembered in order to maintain track of the order of tasks for the agent
- The “debug window” indicated that the communication between the agents was correct

The validity of the implementation is limited to face validation by SMEs, who observed the demonstration. As the model itself is based on empirical data from a particular experiment, any comparison with the same data would be incomplete. Instead, another SME (army officer from mechanized forces) was used to validate the implementation. In this checking, the results were positive.

It's difficult to recreate the real world circumstances from which the model was created to establish empirical validity. In this case, one reason is that the model does not state anything about the causes for mental workload but only the effects on performance. If mental workload could be recreated as a dependent variable, then having a number of real world scenarios where performance was observed in the same instances mental workload was manipulated, empirical validity of the model could be established.

The theoretical validity of the implementation is based on the established concepts of mental workload and the foundations of the framework in which the model is casted. As mentioned in section 2, the precise definitions on mental workload may differ, but the existence of the concept has been established for more than 20 years [9] and the theoretical validity of Soar has been under investigation for about the same amount of time.

6.2 Comparison with other models

MAMID developed by Hudlicka [12] and "SESAME-Soar" by Henninger and Jones [13], are similar attempts to interface behavior moderators with a modularized symbolic agent architecture.

While the model in this thesis is a task description for critical behaviors performed by a tankcrew, Hudlicka and Henninger's models are purely cognitive/affective models. While both MAMID and "SESAME-Soar" are demonstrated in a behavioral context, their focus is on the internal functioning of the cognitive and affective processes.

Further, MAMID and "SESAME-Soar" are developed with consideration to theoretical structures of the mind, but without empirical data from the field. No task analysis is mentioned in Hudlicka's and Henninger's work, meaning that even if the interaction between cognition and affect can be reliably established, there is no guarantee that the effects are relevant for the tasks demonstrated.

On the other hand, while the model in this paper only claims to explain the effects of mental workload on tactical performance in a specific scenario, the scope is much greater in both Hudlicka and Henninger's models for the following reasons:

- Both "SESAME-Soar" and MAMID have a complex interaction between the behavior moderators and performance.
- The number of mental phenomena explained in Hudlicka's framework includes: attention and

WM speed, extraversion, stability, aggression, anxiety, and positive and negative affect. The same list for Henninger is: clarity-confusion, pain-pleasure, arousal, stability-neuroticism, and intro-extroversion

- Both Hudlicka and Henninger attempt to create generic models. However, the validity of being "innately human" is still to be demonstrated by both models.

The implementation is based on the same framework as Henninger's model: Soar. In practise, the interaction with the behavior moderators in Soar, means to introduce new values on the *input* and *output* links of Soar's working memory. In contrast to the model in this thesis, Henninger's model also let the environment percepts pass through the moderating modules to the *input* link. The decision-making can be assumed similar in the models, meaning both use Soar's production rules to fire depending on set conditions.

6.3 Future works

While the model isn't plug-and-play for any context or application, slight modifications to new scenarios is assumed to be rather easy. For example, the addition of urban battle could be done by incorporating a new set of production rules in the same model, and environmental objects in simulation environment. A higher number of participating agents in the simulation could be introduced as well.

The model itself could be extended in many dimensions. More cognitive skills with higher fidelity such as detailed perception, planning and learning, are all potential areas for investigation. The integration of a "workload calculator" would be an interesting extension. By modeling the level of mental workload based on current taskload, the circle is closed with regards to interactivity between behavior and behavior moderator:

**Behavior (Tasks) → Behavior moderator (MW) →
Behavior (Tasks)**

Another interesting part extension would be to introduce the "cognitive variable" situation awareness, investigated in the work of Svensson [11], in the model. Hence the following interaction would form:

**Behavior (Tasks) ↔ Cognitive state (SA) ↔
Behavior moderator (MW)**

Finally, future extensions could include more dynamics in the workload level over a long time, as well as its local rise and decay. Also, filtering of the

workload level (and of other behavior moderator) could be introduced to represent sensitivity.

7. Summary

This paper describes the process of modeling the effects of mental workload on performance in tankcrew. Representing mental workload as a behavior moderator, it influences cognitive/behavioral tasks. The validity is accounted for by collecting the data from empirical results. The cognitive models are implemented in interactive Soar agents and demonstrated in a simulated battlefield scenario. Finally the results and their validity is discussed.

8. References

- [1] Lundin, M. (2004) *Simulating the effects of mental workload on performance*. LIU-KOGVET-D—04/xx—SE, Linköping Electronic Press, Sweden. (in print)
- [2] Gross, J., Ciappara, N., Smist, T., & Benson, P. (1998) *Evaluating the MIA2 tank-commanders interface: the battle of input devices*. Proceedings of the Human Factors and Ergonomic Society 42nd annual meeting.
- [3] O'Donnell, R.D. and Eggemeier, F.T. 1986, Workload assessment methodology, in K.R. Boff, L. Kaufman and J.P. Thomas (eds) *Handbook of perception and human performance*. Vol. II (John Wiley & Sons, New York),42-1 - 42-49.
- [4] Gopher, D. and Donchin E. 1986, Workload - an examination of the concept, in K.R. Boff, L. Kaufman and J.P. Thomas (eds) *Handbook of perception and human performance*. Vol. II. (John Wiley & Sons, New York),41-1-41-49.
- [5] Hart, S.G. and Wickens, C.D. 1990, Workload assessment and prediction, in H.R. Booher (ed) MANPRINT. *An approach to systems integration* (van Nostrand Reinhold, New York), 257-296.
- [6] Magnusson, S. Berggren P., Danielsson B., Svensson, E. (2001) *Dynamisk värdering av operatörsfunktion för framtida systemutveckling*. Metodrapport FOI-R--0430--SE, ISSN 1650-1942
- [7] Kirwan, B., & Ainsworth, L.K. (Eds.) (1992). *A guide to task analysis*. London: Taylor & Francis.
- [8] Lindström, P. (2002). *Prestationsmätt hos stridsvagnsbesättningar - Framtagande av mätverktyg med hjälp av uppgiftsanalys*. Examensarbete Mälardalens högskola, Teknik 2002-218
- [9] Newell, A. (1990) *Unified Theories of Cognition*. Harvard University Press, Cambridge, MA.
- [10] Jones, R. M., Neville, K. & Laird, J.E. (1998) *Modeling pilot fatigue with a synthetic behavior model*. Proceedings of the Seventh Conference on Computer Generated Forces and Behavior Representation. Orlando, FL.
- [11] Svensson, E. & Wilson, S. (1995). *Psychological and Psychophysiological Models of Pilot Performance for Systems Development and Mission Evaluation*. The International Journal of Aviation Psychology
- [12] Hudlicka, E. (2003). *Modeling Effects of Behavior Moderators on Performance: Evaluation of the MAMID Methodology and Architecture*, Proceedings of the 2003 Conference on Behavior Representation in Modeling and Simulation (BRIMS) May 12-15, 2003, Arizona
- [13] Henninger, A., Jones, R.M., and Chown, E. (2003). *Behaviors that Emerge from Emotion and Cognition: Implementation and Evaluation of a Symbolic-Connectionist Architecture*. Proceedings of the 2003 Autonomous Agents and Multiagent Systems (AAMAS 2003). Melbourne, Australia.

Author Biography

MIKAEL LUNDIN is a research engineer at the Department of Man-System Interaction at the Swedish Defence Research Agency (FOI). He holds a M.Sc in cognitive science from Linköping University, Sweden.

Bilaga 4, Reserapport BRIMS 2004

Reserapport BRIMS 2004

17-20 Maj, Arlington, VA

Kurser

PMFServ

Den första kursen gavs av Jason Cornwell och Barry Silverman vid University of Pennsylvania, och handlade om ramverket PMFServ. Mjukvaran baseras helt på beteendemoderatorer och är speciellt inriktad på faktorer som känslor, kulturella värderingar och fysiologisk stress. Den ska ses som ett komplement till Soar och AI-Implant.

I editeringsverktyget representeras varje moderator som en reservoar. Reservoar kan länkas samman för att integrera effekten av fler än en beteendemoderator. PMFServ har olika komponenter för PMFer: stress, perception, emotion, och beslutsfattande. Systemet bygger på en funktion med variabler, samt målträd.

Skrivet i Python och finns kopplingar till Unreal Tournament.

PMFServ kommer att sluta i en kommersiell version men man är ett par år därifrån. Man utvecklar API för interoperabilitet på alla nivåer.

Den antologi över beteendemoderatorer som Silverman gav ut visade sig vara sammansatt av hans studenter.

BRAHMS

Den andra kursen beskrev BRAHMS, utvecklat på NASA Ames Research Center av Maarten Sierhuis och Chin Seah. Det är ett multi-agent verktyg för att beskriva "work practise". Liknar delvis Soar, då det har produktionsregler och problemrymder (thoughtframes). BRAHMS är dock ett mer typat språk och har fundamentala primitiver som "fact" "belief" "activity" "object" "agent" osv. Allt är tydligt och logiskt definierat. Verktyget har även en geografimodul och tidshantering.

BRAHMS ger intryck av att vara en komplett utvecklingsmiljö.

Systemet följde med kursen som freeware. Utveckling går åt att integrera mer och mer av språket med Java. Har använts för att beskriva utforskning av Mars.

Paneldebatter

Design Patterns

Inom objektorienterad programmering beskriver design pattern användbara lösningar till problem. Dessa rör sig ofta på en abstrakt nivå och mynnar ut i den konceptuella modellens design. De används istället för att varje gång uppfinna bra lösningar som redan finns.

Diskussionen handlade om huruvida man kan använda design patterns inom beteendemodellering och isåfall på vilket sätt. Tre av fyra i panelen argumenterade för att man kan specificera upp beteendekomponenter i standardmallar. Deras fokus varierade från återanvändande av ”bitar av bra och generell kod” till ”bra och generella metoder att skriva kod” (program vs programmeringskonventioner). En av panelmedlemmarna motsatte sig dock en standard i nuläget, dels för att föreslagen mjukvarustandard inte innehöll alla viktiga aspekter som krävdes för mänsklig beteende, dels för att forskningen inte var mogen än att ”ruta in sig”.

Böcker som nämndes var Eric Gamma's ”Design Patterns” samt Christopher Alexander's ”The timeless way of building”. Den senare diskuterar designmönster från husarkitektur.

Förutom återanvändningen ser man design patterns som en potentiell metod att förmedla komponenter mellan modellerare (experter och noviser), mellan ”knowledge engineer” och SME, samt mellan arkitekturer.

Folkmassor

US Airforce Research Lab har satsat på ett större projekt kallat ”Stormchaser” (metafor ”social forecasting”). Det går ut på att empiriskt försöka identifiera de faktorer som påverkar folkmassors beteende. Idén bakom är att kunna skapa någon typ av datorverktyg till hjälp för kontrollstyrkor. Därmed ska de kunna undvika eskalering, skydda ”icke-stridande”, och bättre kunna utföra sina uppdrag. Genom att kombinera bakgrundsvariabler med situationsvariabler hos både folkmassa och kontrollgrupp vill man kunna förutse utfall och identifiera ”triggers”. Projektet är nystartat och de faktorer man tittat på är folkmassans:

- Tema
- Syfte
- Gruppstorlek
- Humör/Attityd
- Hot/Förmåga
- Tillhyggen
- Lokal polisstyrka
- Storlek på kontrollstyrka
- ”Rules of engagement”
- Kommunikationkanaler
- Kommunikationsbarriärer

- Informationsflöden före demonstrationer (ex. nyhetsändringar, info om demonstration)

I de studier man hittills gjort har det visat sig att vissa faktorer är avgörande (kontrollstyrkans beteende) medan andra är irrelevanta (Humör/Attityd) för sannolikheten att folkmassans beteende blir våldsamt.

Modellering av naturligt språk

Debatten var koncentrerad till en längre presentation av Wayne Zachary som beskrev var forskningen ligger idag. Det positiva var att en lång tids forskning till sist burit frukt och det idag går att göra realistiskt tal i begränsade domäner. Det negativa var att de system som till exempel telefonitjänster utnyttjar och program för att läsa upp dokument inte är lämpade.

Jonathan Gratch vid Army Research Labs spelade upp ett imponerande exempel på hur två soldater kommunicerade med syntetiskt tal. Systemet grundade sig på ett open-source system (Festival) för att skapa realistiskt tal.

Läsvärda artiklar

REDUX – Example-driven diagrammatic tools for rapid knowledge acquisition

REDUX utvecklas av Douglas Pearson och John Laird och är ett verktyg för att automatisera modelleringsprocessen. Det grundar sig i tanken att en SME tillsammans med modelleraren, eller SME själv, använder två-dimensionella diagram för att beskriva exempel på beteenden. Exempelen analyseras för inkonsistenser, eventuellt generaliseras, och genererar automatiskt kod. Förutom rörelsemönster som beskrivs i 2D finns ett antal menyer för mer abstrakta beteenden, mål, och tillstånd. I viss mån kan denna process baklänges generera diagram från kod.

Verktyget är i ett tidigt stadium av utvecklingen men ambitionen är hög. Man har skapat flygstridsmodeller på några dagar, även om exemplet berörde MOUT scenarier.

REDUX blev utvalt till "BRIMS selected reading list".

CaDaDis

CaDaDis är ett hjälpverktyg för att visualisera programflödet i kognitiva modeller (har testats för modeller i Soar, ACT-R, och iGEN). Två typer av diagram visades under presentationen: Aktuella tillstånd livslängd (millisekund eller decision-cycles) och flödet mellan aktuella operatorer. CaDaDis är en påbyggnad av SoarTech's VISTA.

Modeling Physical Variability for Synthetic MOUT Agents

Gita Sukthankar et al på Carnegie Mellon University har skapat en modell för resonering om kroppens fysik. Den utgår från motion-capture data och skapar sedan en kostnads-funktion för vilken typ av rörelse som ska göras. På så sätt ser projektgruppen det som resonering om sin egen kropp, dvs en beräkningsmodul mellan Soar's "högre"

resonerande och den fysiska modellen. De motiverar detta med att begränsningarna till beteenden sitter i hur man resonerar om sin kropp snarare än de fysiska begränsningarna.

Modeling Physical Variability for Synthetic MOUT Agents blev utvalt till "BRIMS selected reading list".

Using Human Behavior and Environmental Servers to Enhance the Fidelity of JSAF

Jennifer King vid Naval Research Labs presenterade projektet som gick ut på att skapa en federation av modeller som syftade till att påvisa hur röjdykare påverkas av externa beteendemoderatorer. Genom att modifiera en ubåtsmodell i JSAF skapades en individuell dykare. Vidare gjordes en uppgiftsanalys som sedan modellerades i MicroSaint. För att påverka dykarens beteende med avseende på externa faktorer (ex. vattenströmmar och salthalt) användes OASIS. Detta är ett ramverk för att simulera och predicera varierande miljödata, dvs temporala och spatiala effekter för rymd-, atmosfärs- och oceanografisk data. De olika modellerna kopplades samman i en HLA-federation.

Trots en avsevärt annorlunda domän var angreppssättet och metoden var påfallande lik "Simulating the Effects of Mental Workload on Performance in Tankcrew":

Uppgiftsanalys, sekvens och hierarkisk representation av uppgifter, samt beteendemoderatorer för mer realistiskt beteende.

Toward a working Taxonomy of Groups, Predicting Group Behavior from Profiles and Stereotypes

Bryan Karabaich och Eva Hudlicka har gjort en ansats till att förutsäga grupp beteenden. Taxonomin grundar sig på George's "Operational Codes" och Ajzen's "Theory of Planned Behavior". Variabler som tas upp är:

- Gruppens sammansättning: Etnisk, religiös, geografisk, socioekonomisk, professionell, politisk, militant, militär. Av intresse och gemensamt för alla är gruppens homogenitet samt kombinationer av sammansättningar.
- Gruppens filosofi: Globalt eller generellt mål, världssyn, specifika mål, upplevda effektiva och acceptabla metoder för att nå mål, idéer om gruppfilosofins spridning, chanser att uppnå mål, timing och planering, samt gruppens nivå av engagemang
- Gruppens struktur: Organisation, Ledarskapsstil (demokratisk, Laissez-Faire, autokratisk, eller diktatorisk), maktdynamik, kommunikationsdynamik, gruppdisciplin.

Utifrån dessa definitioner försöker sedan Hudlicka skapa stereotypa grupper, för att med hjälp av logiska regler förutsäga beteende. Genom att skapa profiler över grupper (där saknad kunskap kan härledas) och beräkna korrelationer kan tänkbara beteendemönster prediceras. En reflektion är att de givna parametrarna enbart formaliserar ganska självklar kunskap (ex IF past behavior includes violence THEN likelihood of violence = HIGH).

Mest framstående nordamerikanska aktörerna inom HBR

- BBN Technologies
- Charles River Analytics
- CHI Systems
- SoarTech Inc.
- Carnegie Mellon University
- University of Pennsylvania
- Pennsylvania State University