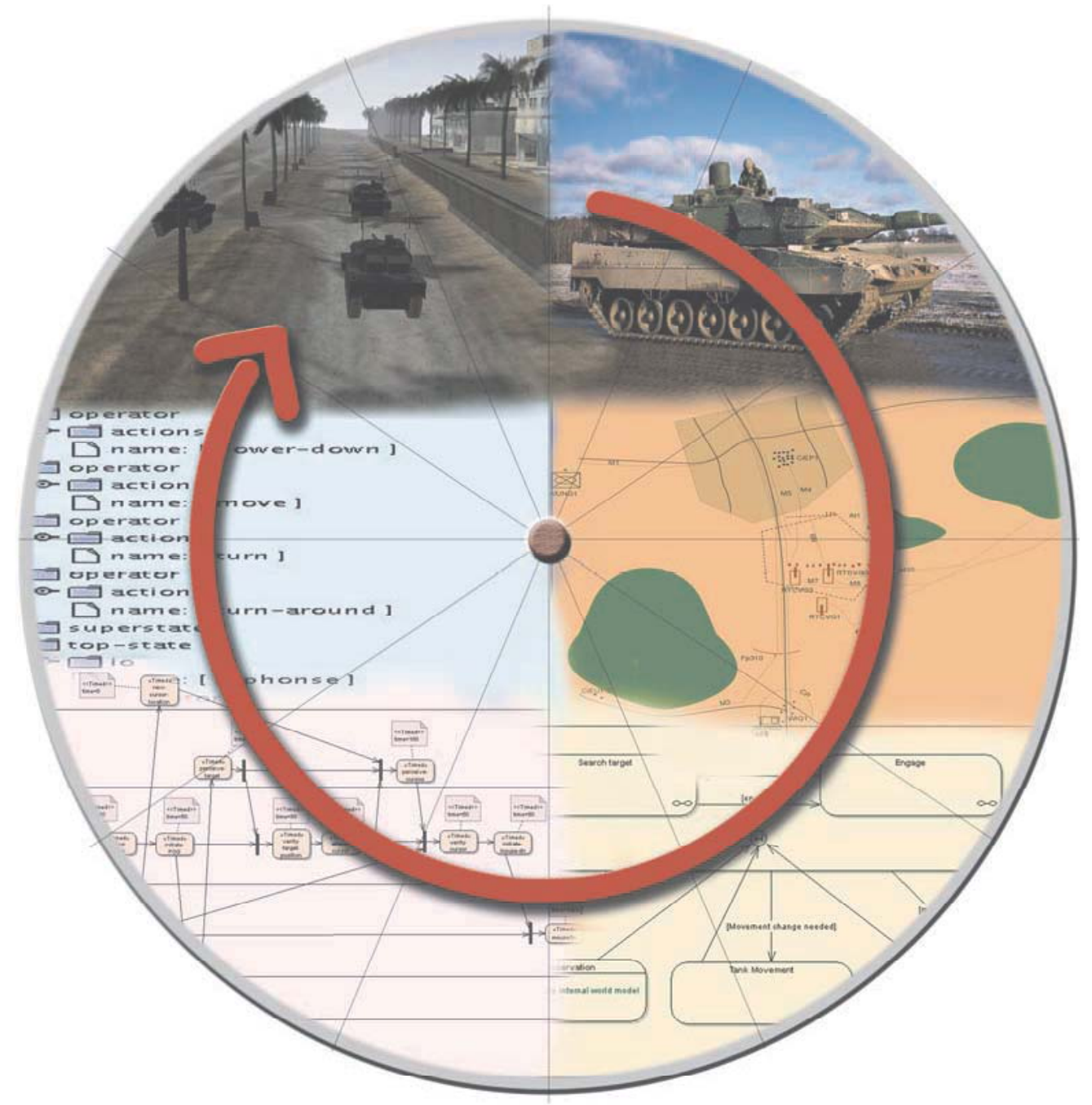


NIKLAS WALLIN, STEN-ÅKE NILSSON,  
MIKAEL LUNDIN, MARTIN CASTOR



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

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

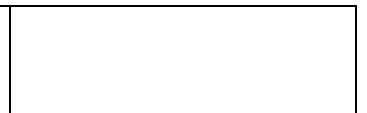
# Guidelines för utveckling och användning av modeller av mänskligt beteende

<b>Utgivare</b> FOI - Totalförsvarets forskningsinstitut Systemteknik 164 90 Stockholm	<b>Rapportnummer, ISRN</b> FOI-R--1837--SE	<b>Klassificering</b> Användarrapport
	<b>Forskningsområde</b> 2. Operationsanalys, modellering och simulering	
	<b>Månad, år</b> December 2005	<b>Projektnummer</b> E6026
	<b>Delområde</b> 21 Modellering och simulering	
	<b>Delområde 2</b>	
<b>Författare/redaktör</b> Niklas Wallin Sten-Åke Nilsson Mikael Lundin Martin Castor	<b>Projektledare</b> Sten-Åke Nilsson	
	<b>Godkänd av</b> Monica Dahlén	
	<b>Uppdragsgivare/kundbeteckning</b> FM	
	<b>Tekniskt och/eller vetenskapligt ansvarig</b>	
<b>Rapportens titel</b> Guidelines för utveckling och användning av modeller av mänskligt beteende		
<b>Sammanfattning</b> <p>Detta är en första utgåva av Guidelines för utveckling och användning av modeller av mänskligt beteende inom området datorgenererade militära aktörer (CGF). Förutom generell orientering om beteendemodeller och CGF:er ges förslag på innehåll i utvecklingsprocessen av sådana modeller samt många hänvisningar till viktig litteratur inom området. Denna guide följer i stort sett FEDEP som är en allmänt vedertagen modellutvecklingsprocess.</p> <p>Ett förslag på ett ramverk för utveckling och nyttjande av beteendemodeller inom FM har tagits fram och beskrivs. Ramverket bygger på tekniker för komponentbaserad mjukvaruutveckling där grundstenen är den komponentmodell som utvecklats av FOI och FMV och som bl.a. använts i Merlin. Tanken med en sådan komponentbaserad teknologi är att kunna skapa återanvändbara delmodeller som enkelt skall kunna sättas samman till större enheter. För att kunna åstadkomma detta måste delmodellerna erbjuda vissa standardiserade gränssnitt.</p>		
<b>Nyckelord</b> beteendemodellering, datorgenererade aktörer, CGF, M&S, arkitekturer, uppgiftsanalys, konceptuell modell, modellutvecklingsprocess, VV&A		
<b>Övriga bibliografiska uppgifter</b>	<b>Språk</b> Svenska	
<b>ISSN</b> 1650-1942	<b>Antal sidor:</b> 43 s.	
<b>Distribution enligt missiv</b>	<b>Pris:</b> Enligt prislista	

<b>Issuing organization</b> FOI – Swedish Defence Research Agency Systems Technology SE-164 90 Stockholm	<b>Report number, ISRN</b> FOI-R--1837--SE	<b>Report type</b> User report
	<b>Programme Areas</b> 2. Operational Research, Modelling and Simulation	
	<b>Month year</b> December 2005	<b>Project no.</b> E6026
	<b>Subcategories</b> 21 Modelling and Simulation	
	<b>Subcategories 2</b>	
<b>Author/s (editor/s)</b> Niklas Wallin Sten-Åke Nilsson Mikael Lundin Martin Castor	<b>Project manager</b> Sten-Åke Nilsson	
	<b>Approved by</b> Monica Dahlén	
	<b>Sponsoring agency</b> Swedish Armed Forces	
	<b>Scientifically and technically responsible</b>	
<b>Report title (In translation)</b> Guidelines for development and utilization of human behaviour models		
<b>Abstract</b> <p>This is the first edition of "Guidelines for development and utilization of human behaviour models within the field of Computer Generated Forces". The guide presents general aspects of behavior modeling, proposes an agenda for the development process of such models and holds references to other important projects and literature within the field. The guide adopts a development process similar to the FEDEP process.</p> <p>A framework for development and utilization of behavior models has been developed and is presented here. The framework uses the same techniques for component based software engineering as the Merlin architecture, developed by the Swedish Defence Material Administration and FOI. Thus, the framework facilitates reuse of models and aggregation of model components into larger assemblies with the use of standard interfaces.</p>		
<b>Keywords</b> HBR, CGF, computer generated aktors, architecture, task analysis, conceptual model, model development, VV&A		
<b>Further bibliographic information</b>	<b>Language</b> Swedish	
<b>ISSN</b> 1650-1942	<b>Pages</b> 43 p.	
	<b>Price acc. to pricelist</b>	

FOI-R--1837--SE  
ISSN 1650-1942

**Systemteknik, Ledningssystem**



# Guidelines för utveckling och användning av modeller av mänskligt beteende

## Innehåll:

<b>SAMMANFATTNING</b>	<b>7</b>
<b>1 INLEDNING</b>	<b>8</b>
1.1 OM PROJEKTET "DATORGENERERADE STYRKOR"	8
1.2 OM DETTA DOKUMENT	8
<b>2 OM CGF- OCH HBR-MODELLER</b>	<b>10</b>
2.1 AI	10
2.2 KOGNITIV MODELLERING	11
2.3 INTRESSANTA AKTIVITETER INOM OMRÅDET	12
<b>3 MODELLUTVECKLING</b>	<b>14</b>
3.1 UTVECKLING AV BETEENDEMODELLER	15
3.2 UTVECKLING AV SIMULERINGSMODELLER	16
<b>4 KRAVSPECIFIKATION</b>	<b>18</b>
<b>5 KONCEPTUELL MODELL</b>	<b>21</b>
5.1 KUNSKAPSANSKAFFNING (KNOWLEDGE ACQUISITION, KA)	21
5.2 VERKTYG FÖR KUNSKAPSANSKAFFNING	22
5.2.1 MANUALER, UTBILDNINGSMATERIAL	22
5.2.2 INTERVJUER OCH DISKUSSIONER MED DOMÄNEXPERTER	22
5.2.3 OBSERVATIONER, MÄTNINGAR	23
5.3 MALL FÖR BETEENDEDATA FÖR CGF-UTVECKLING	23
<b>6 DESIGN</b>	<b>26</b>
6.1 INTERAKTIONER	26
6.2 KUNSKAPSBEARBETNING (KE)	26
<b>7 UTVECKLING</b>	<b>29</b>
7.1 ARKITEKTURER	29
7.2 KOGNITIVA ARKITEKTURER	29
7.2.1 VANLIGA KOGNITIVA ARKITEKTURER	30
7.2.2 BETEENDEMODERATORER	30
7.3 AI-ARKITEKTURER	31
7.3.1 VANLIGA AI-ARKITEKTURER	31
7.4 CGF-ARKITEKTURER	31

7.4.1	VANLIGA CGF-ARKITEKTURER	32
<b>8</b>	<b>INTEGRATION</b>	<b>33</b>
<b>8.1</b>	<b>BIBLIOTEKSRAMVERK</b>	<b>33</b>
8.1.1	INTEGRATIONSVERKTYG	33
<b>9</b>	<b>VV&amp;A</b>	<b>36</b>
<b>9.1</b>	<b>DEFINITIONER</b>	<b>36</b>
<b>9.2</b>	<b>SEX VIKTIGA FASER UNDER EN VV&amp;A-PROCESS</b>	<b>37</b>
9.2.1	DEFINITION AV VV&A- KRITERIER.	37
9.2.2	VALIDERING AV DEN KONCEPTUELLA MODELLEN	37
9.2.3	VERIFIERING AV DESIGN	37
9.2.4	KODVERIFIERING	37
9.2.5	VERIFIERING OCH CERTIFIERING AV DATA	37
9.2.6	VALIDERING AV HELA TILLÄMPNINGEN	37
<b>9.3</b>	<b>ALTERNATIVA VV&amp;A-PROCESSER</b>	<b>38</b>
REVVÄ		38
SISO PDG (PRODUCT DEVELOPMENT GROUP) VV&A		38
ITOP (INTERNATIONAL TEST OPERATIONS PROCEDURE)		38
NATO MSG 19		38
<b>9.4</b>	<b>SPECIFIKA VV&amp;A-FRÅGOR SOM UPPKOMMER VID UTVECKLING AV MODELLER SOM AVBILDAR MÄNSKLIGT BETEENDE</b>	<b>38</b>
9.4.1	DEFINITION AV VV&A-KRITERIER	39
9.4.2	VALIDERING AV DEN KONCEPTUELLA MODELLEN	39
9.4.3	VERIFIERING AV DESIGN OCH IMPLEMENTATION	40
9.4.4	VERIFIERING OCH CERTIFIERING AV DATA	40
9.4.5	VALIDERING AV DEN KOMPLETTA MODELLEN	40
<b>9.5</b>	<b>OMRÅDET IDAG</b>	<b>41</b>
<b>10</b>	<b>BEGREPP OCH FÖRKORTNINGAR</b>	<b>43</b>

# Sammanfattning

Detta är en första utgåva av Guidelines för utveckling och användning av beteendemodeller inom området datorgenererade aktörer (CGF). Förutom generell orientering om beteendemodeller och CGF:er ges förslag på innehåll i utvecklingsprocessen av sådana modeller, både kortfattat med inledande text inom rutor i början av varje avsnitt och med text med många hänvisningar till viktig litteratur inom området. Denna guide följer i stort sett FEDEP som är en allmänt vedertagen modellutvecklingsprocess.

Grunden för datorgenererade aktörer och beteendemodeller är artificiell intelligens och kognitionsvetenskap. Beteendemodeller har ett brett spektrum alltifrån en modell som ger en trovärdig respons på ett enkelt stimuli (reaktiv modell) till avancerade modeller av hjärnans kognitiva funktioner, vilka kan generera komplexa beteenden.

Modellutveckling innebär att utvecklaren först måste avgöra vilka egenskaper hos den ”riktiga världen” (referenten) som ska vara med i modellen och vilka som inte ska vara med. Detta resulterar i en kravspecifikation utifrån vilken en konceptuell modell tas fram. En konceptuell modell är en övergripande beskrivning av den modellerade verkligheten. Den kan uttryckas i mer eller mindre formella språk.

I designfasen skall man utifrån de scenarier och den kunskap man utvecklat skapa en tillräckligt god grund för en exekverbar modell. En arkitektur för beteendemodeller är den delen av systemet som tillhandahåller och sköter de primitiva funktionerna hos en beteendemodell. En modellutvecklare kan med hjälp av dessa funktioner skapa egna modeller, oftast med hjälp av regler. En beteendemodell består därför av dels regler och dels arkitekturens tjänster för att generera beteendet.

I integrationsfasen skall beteendemodellen integreras med simuleringsramverket. Vid integrationen så bör man försöka se till att främja återanvändning av modellerna. Detta kan göras genom att man försöker skapa generella gränssnitt som kan anpassas till simuleringsmiljöernas modeller. Målet är att de modeller man utvecklar samlas i ett bibliotek av återanvändbara s.k. ”plug’n’play”-modeller.

Verifiering, validering och ackreditering (VV&A) av simuleringsmodeller är en process som syftar till att öka tillförlitligheten hos simuleringsmodeller och att ge användaren av en modell information om dess lämplighet för ett givet syfte samt om vilken tilltro han kan sätta till modellen. Denna process bidrar även till att fel och ofullständigheter ofta upptäcks tidigt under utvecklingsprocessen, vilket kan vara av betydande ekonomiskt intresse. VV&A-arbete bör således pågå som en parallell process i modellutvecklingsarbetet. Området VV&A av modeller rörande mänskligt beteende är idag dock behäftat med många svårigheter och man har inte nått samma trovärdighetsgrad som för modeller av fysiska system.

Beroende på att det pågår internationellt samarbete med liknande syfte som denna guide avser vi att inom två år uppdatera guiden med resultat från detta samarbete.



# 1 Inledning

Denna guide är ett resultat av ett fyraårigt forskningsarbete vid FOI inom området datorgenererade styrkor, CGF. Projektet har studerat grunder, tekniker och metoder för uppbyggnad och användning av CGF:er. Projektet har också tagit fram ett förslag till design av bibliotek för beteendemodeller och och därtill hörande verktyg som skall kunna användas inom försvarsmakten.

Forskningsarbetet inom projektet har i efterhand visat sig följa upplägget av den mest kända och använda framställningen inom området nämligen Pew and Mavor "Modeling Human and Organizational Behavior, Behavior Application to Military Simulations", [Pew and Mavor, 1998].

## 1.1 Om projektet "Datorgenererade styrkor"

Under 2002-2005 har ett projekt "Datorgenererade styrkor" bedrivits vid FOI. Syftet med projektet har varit att:

- Kartlägga området "Datorgenererade styrkor" och "Datorgenererade beteendemodeller" både nationellt och internationellt.
- Införskaffa kunskap inom området för att därigenom kunna bistå försvarsmakten i sådana frågor
- Skapa ett kontaktnät nationellt och internationellt
- Skapa ett ramverk för gemensamt nyttjande av beteendemodeller inom försvarsmakten.

## 1.2 Om detta dokument

Detta dokument är framtaget för att ge en kort introduktion till Datorgenererade beteendemodeller. Dokumentet är ett resultat av fyra års forskning inom området, och speglar de erfarenheter som projektet samlat från forskning, egenutveckling, seminarier, konferenser, internationella samarbeten, etc. Det är främst avsett för nya användare och utvecklare som snabbt vill sätta sig in i området. Rapporten fördjupar sig inte inom de olika delområdena utan hänvisar till litteratur som är lämplig för den som vill sätta sig mer in i ämnet.

Projektet deltar även i en NATO-grupp HFM-128 (Human Factors and Medicine), vars syfte är att hitta "best practices" inom beteendemodellering. Gruppen som etablerades 2004 kommer att arbeta under 3 år, och som resultat ge ut en rapport vars innehåll kommer att ingå i kommande utgåvor av denna rapport. Gruppen består av representanter från USA, Canada, Storbritannien, Nederländerna, Tyskland och Sverige. Gruppen presenterade sitt pågående arbete vid den senaste BRIMS-konferensen i i USA, Universal City, CA, maj 2005 [NATO HFM-128, 2005].

1996 fick National Research Council (NRC) ett uppdrag av Defense Modeling and Simulation Office (DMSO) i USA att göra en granskning av området tillämpad HBR vid militär simulering. En grupp (panel) sammanställde material vilket resulterade i en rapport [Pew and Mavor, 1998] som än idag anses vara bland de bästa inom området. Nedan är en introduktion till rapporten:

This report is the work of the Panel on Modeling Human Behavior and Command Decision Making: Representations for Military Simulations. The panel was established by the National Research Council (NRC) in 1996 in response to a request from the Defense Modeling and Simulation Office of the U.S. Department of Defense. The charge to the panel was to review the state of the art in human behavior representation as applied to military simulations, with emphasis on the challenging areas of cognitive, team, and organizational behavior.

The panel formed to meet these goals included experts in individual behavior, organizational behavior, decision making, human factors, computational modeling, and military simulations. The project extended over an 18-month period. At the end of the first phase, in February 1997, the panel published an interim report (Pew and Mavor, 1997) that argued for the need for models of human behavior, summarized a methodology for ensuring the development of useful models, and described selected psychological process models that have the potential to improve the realism with which human-influenced action is represented. In the second phase of the project, the panel conducted an in-depth analysis of the theoretical and applied research in human behavior modeling at the individual, unit, and command levels. The result of that analysis is presented in this final report.

This report is intended not only for policy makers in the Defense Modeling and Simulation Office and the military services, but also for the broader behavioural science community in the military, other government agencies, industry, and universities, whose modeling efforts can contribute to the development of more realistic and thus more useful military simulations.

Senare har det kommit tillägg till rapporten skriven av [Ritter et al., 2003]:

Selected recent developments and promising directions for improving the quality of models of human performance in synthetic environments are summarized, beginning with the potential uses and goals for behavioral models. The focus is on providing more complete performance, better integration of the models with synthetic environments and with each other (reusability), and improved usability of the models. Within this context, relevant, current work related to modeling is reviewed, such as cognitive modeling of emotion, advanced techniques for testing and building models of behavior, new cognitive architectures including hybrid architectures, and agent and Belief, Desires and Intentions (BDI) architectures. A list of projects with high payoff for modeling human performance in synthetic environments is provided as a conclusion.

Dessa rapporter refereras ofta till i denna guide.

## Referenser

- [NATO HFM-128, 2005] NATO RTO HFM-128 Research Task Group on Representation of Human Behavior in Constructive Simulation). Proceedings from BRIMS-conference Maj, 2005. SISO.
- [Pew and Mavor, 1998] Pew, R. W. and Mavor, A. S., Eds. (1998). *Modeling Human and Organizational Behavior Application to Military Simulations*. National Academy Press, Washington, D.C.
- [Ritter et al., 2003] Ritter, F. E., Shadbolt, N. R., Elliman, D., Young, R., Gobet, F., & Baxter, G. D. (2003). Techniques for modeling human and organizational behaviour in synthetic environments: A supplementary review. Wright-Patterson Air Force Base, OH: Human Systems Information Analysis Center.

## 2 Om CGF- och HBR-modeller

Grunden för datorgenererade aktörer och beteendemodeller är artificiell intelligens (AI) och kognitionsvetenskap och de bakomliggande vetenskaperna filosofi, matematik, psykologi samt datorvetenskap och lingvistik

Beteendemodeller har ett brett spektrum alltifrån modeller som ger en trovärdig respons på ett enkelt stimuli (reaktiv modell) till avancerade modeller av hjärnans kognitiva funktioner, vilka kan generera komplexa beteenden. Sådana mer avancerade beteendemodeller kallas också kognitiva modeller, "functional models" eller processmodeller.

Det finns särskilda fora och konferenser som tar upp området mänsklig beteenderepresentation (HBR) och närliggande frågor vid sidan av andra AI- och kognitionsvetarkonferenser.

Datorgenererade styrkor (CGF) har växt fram som en mycket värdefull komponent vid militära simuleringar för att befolka simulerade världar.

Datorgenererade styrkor är ett begrepp, som myntades under 1980-talet och som innefattar beteendemodeller som antingen representerar enskilda operatörer eller organisationer inom den militära domänen. På senare tid har man dock börjat generalisera begreppet till (datorgenererad) mänsklig beteenderepresentation (HBR), då många av problemen och lösningarna sträcker sig över många domäner och tillämpningar.

De främsta argumenten för att nyttja beteendemodeller inom M&S är att de gör det billigt och relativt enkelt att sätta samman simuleringar med många aktörer. Andra viktiga fördelar är att beteendemodeller kan exekveras med olika hastigheter, de kan exekveras upprepade gånger och de kan fås att upprepa sitt beteende, eventuellt med vissa variationer, vilket gör dem användbara bl.a. för Monte Carlo-simuleringar.

De främsta argumenten mot nyttjande av beteendemodeller är att det kan vara en lång och komplicerad process att ta fram tillräckligt bra modeller, och det kan ifrågasättas huruvida deras beteenden stämmer överens med deras mänskliga motsvarighet.

### 2.1 AI

Intelligens kan definieras som förmåga enligt ett kunskapsgraderat system ("a knowledge-level system"). Perfekt intelligens innebär att man (eller systemet som äger intelligensen) i varje ögonblick har förmåga att använda den kunskap man försetts med från början eller införskaffat under resans gång att behandla varje uppkommet problem. [Newell, 1982]

Artificiell intelligens är då följdriktigt tillämpningen hos ett konstgjort eller icke naturligt system vilket använder motsvarande kunskapsgrad för att uppnå sitt mål. En mer praktisk definition av AI är: *vetenskap om att få maskiner att utföra uppgifter som skulle kräva intelligens om människan hade utfört dem* [Minsky, 1963].

En dator har sedan länge kunnat räkna och sortera begrepp efter olika regler betydligt snabbare än människor men innehar trots denna egenskap inte intelligens om vi definierar den som ovan.

Sedan länge har man delat in verksamhet inom följande fyra delområden med olika definitioner på AI:

- Mänskligt tänkande – det kognitiva närmande- eller beskrivningssättet
- Mänskligt beteende – beskrivningssätt enligt Turing
- Rationellt tänkande – det logiska beskrivningssättet
- Rationellt beteende – beskrivningssättet enligt den rationelle agenten

## 2.2 Kognitiv modellering

Parallellt med AI-vetenskap har det tvärvetenskapliga området kognitionsvetenskap växt fram. Här försöker forskarna förena kunskap och metoder inom AI med ämnen som kognitiv psykologi, neurologi, filosofi och lingvistik.

*“Cognitive science brings together computer models from AI and experimental techniques from psychology to try to construct precise and testable theories of the workings of the human mind”*

*[Russell, Norvig, 1995].*

Kognition handlar om processerna i hjärnan: perception, igenkänning, resonerande funktioner, beslutsfattande, planerande och lärande. En ”Unified Theory” inom ett område är en teori som försöker förklara de mekanismer som kan definieras som frågeställningar inom detta område. Alla företeelser inom området skall kunna återskapas och förklaras av denna teori<sup>1</sup>. En ”Unified Theory of Cognition” (UTC) är ett försök att beskriva en teori om människans alla kognitiva funktioner, d.v.s. de interna processerna som försiggår i den mänskliga hjärnan.

Det är svårt att validera modellerna av dessa interna processer. Det enda man kan mäta som resultat från modellerna är vad som kommer ut i form av beteende beroende på vilken input som modellen matas med (kanske också i vissa fall beroende på inneboende kunskap). Experimentalpsykologin har genom experiment sökt kartlägga hur funktionerna fungerar generellt. AI-teorier har haft stor betydelse vid skapandet av dessa modeller inte minst vad gäller representationen av kunskap.

Några av målen med AI är att:

- Ta fram maskiner som utför mentalt arbete lika bra som människor eller bättre.
- Ta fram system som kan användas för beslutsstöd t.ex. expertsystem.
- Ta fram system som kan användas för problemlösning.
- Lära oss, förklara och få större kunskap om intelligenta processer.

Målet för kognitiv modellering är att bygga modeller av mänskliga mentala mekanismer. För både AI och kognitiv modellering gäller att modellen skall klara sin uppgift (vad den är avsedd för). Vid kognitiv modellering skall modellen dessutom bete sig mänskligt när den utför sin uppgift samt vara en representativ modell för hur processerna går till i hjärnan. För CGF:er har Laird listat följande önskade förmågor<sup>2</sup>:

---

<sup>1</sup> <http://ai.eecs.umich.edu/cogarch4/>

<sup>2</sup> <http://ai.eecs.umich.edu/soar/sitemaker/workshop/23>

- Interagera med en komplex värld.
- Reagera snabbt på förändringar i världen.
- Använda omfattande kunskap.
- Agera målmedvetet.
- Uppträda mänskligt.
- Resonera på meta-nivå (resonera om resonerande).
- Koordinera uppträdande och kommunicera med andra.
- Lära sig av framgångar och misstag.
- Integrera kunskap från tidigare uppgifter.

Problem med kognitiv modellering:

- Mentala processer är ofta undermedvetna.
- Beteenden styrs (störs) av många faktorer.
- Beteende är uppgiftsberoende, det är svårt att skapa modeller som gäller även för likartade uppgifter.
- Vissa processer som t.ex. problemlösande och resonerande är oftast mer beroende av förvärvad kunskap än av yttre förhållanden.
- Det är ont om data och det är svårt att få fram data som stöder modelleringsarbetet.

## 2.3 Intressanta aktiviteter inom området

### **BRIMS** (Behavior Representation in Modelling and Simulation)

Varje år hålls en konferens för forskning inom området. Tillämpningsområdena domineras av den militära domänen, vilket tydligare speglades i konferensens tidigare namn Computer Generated Forces. Namnbytet är ett försök att samla forskningsresultat även från andra domäner. Varje år väljs rekommenderade presentationer som ska visa på områdets utveckling.

### **Pew and Mavor**

Pew and Mavor sammaställde 1998 en arbetsgrupps ansträngningar att göra en state-of-the-art beskrivning av området som gällde för beteendemodellering både av grupp och organisation. Dokumentet innehåller också förslag på angelägna fortsatta forskningsinsatser inom området.

### **HFM-128**

NATO-arbetsgrupp som 2004 påbörjade ett treårigt samarbete för att ta fram ett dokument som överbryggar de teoretiska grunderna med den tillämpade användningen av HBR-modeller. Se också inledning ovan.

### **SAS-017**

The RTO Studies, Analysis and Simulation Panel (SAS) gjorde 2001 en sammanställning av möjligheter med HBR-modellering och behov av insatser inom området. [NATO SAS017, 2001].

### **AMBR**

AMBR projektet (Agent-based Modeling and Behavioral Representation) var ett mycket stort forskningsprogram, där flera olika arkitekturers (EPIC-Soar, ACT-R, iGen/COGNET, D-COG och D-OMAR) förmåga att modellera mänskligt beteende i ett enkelt flygledningsscenario jämfördes, dels med varandra och dels med mänskliga operatörer.

## Referenser

- [Minsky, 1963] Minsky, M. "Matter, Mind, and Models", 1963.
- [NATO SAS017, 2001] NATO RTO SAS017 (2001). Human Behavior Representation. NATO RTO Technical Report 47, SAS Panel.
- [Newell, 1982] Newell, A. (1982). The knowledge level. *Artificial Intelligence*. 18(1), 87-127.
- [Russell, Norvig, 1995]. Russel, S. J., Norvig, P. Artificial Intelligence. 1995. Prentice Hall.

### 3 Modellutveckling

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 relationer mellan dessa komponenter, som oftast är någon sorts orsak-verkan-relation. En modell kan ofta visualiseras i någon grafisk form.

Många av de tekniker som används för mjukvaruutveckling kan också användas för utveckling av beteendemodeller. Kravfångst, design, utveckling, tester, dokumentation, etc. är typiska rutiner vid mjukvaruutveckling.

Mer detaljerad information om processer, delprocesser och tekniker för simulerings- och mjukvaruutveckling finns i en mängd litteratur, t.ex [Law and Kelton, 1999].

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 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 betydelse 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 modellen användningen som avgör kraven på vad modellerna ska klara av.

### 3.1 Utveckling av beteendemodeller

Att utveckla datorgenererade styrkor och intelligenta agenter med realistiskt beteende är i allmänhet mycket svårt och komplext. Den inneboende icke-linjäritet 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 intressefokuser hos de personer som arbetar med utveckling av datorgenererade styrkor vilket beskrivs i [Castor et al., 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 adekvata modeller av belastning och prestation påpekas bl.a. i [Castor et al., 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 et al., 2001] eller Behavior Moderators [Pew and Mavor, 1998] som används grundade på relativt lösa antaganden. Mer information om PMF:er finns bl.a. i [Lundin, 2004] och en omfattande lista av PMF:er ges i [Silverman et al., 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 avsöker flygplanets presentationsytor och omvärlden. Anta vidare att följande modeller 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 avsöker 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.

### 3.2 Utveckling av simuleringsmodeller

Enligt tidigare definition så innebär utveckling av beteendemodeller också utveckling av simuleringsmodeller. Att utveckla en simuleringsmodell som kan exekveras på en dator innebär bl.a. att man utvecklar ett system av mjukvara. Många av de tekniker som används för mjukvaruutveckling kan således användas för utveckling av beteendemodeller. Kravfångst, design, utveckling, tester, dokumentation, etc. är typiska rutiner vid mjukvaruutveckling. Huruvida man vill använda sig av vattenfallsmodeller eller iterativa modeller är upp till utvecklaren. Iterativa modeller förespråkas i denna rapport.

Nedan presenteras ett förslag på utvecklingsprocess som är inspirerad av FEDEP. I följande kapitel kommer varje delprocess att behandlas, dock med inriktning på de tekniker som är fokuserade på utveckling av beteendemodeller.

- **Kravspecifikation**
- Hur och var ska modellen användas?
- **Konceptuell modell**
- Skapa lämpliga scenarion, kunskapsinhämtning, etc.
- **Design**
- Kunskapsbearbetning, uppgiftsanalys.
- **Utveckla modellen**
- Simuleringsramverk, programspråk
- **Integrera modellen**

Till dessa processer utförs också:

- **Test**
- **VV&A**
- **Dokumentation**

### Referenser

- [Andersson, 1992] Anderson, J. (2002). Spanning seven orders of magnitude: a challenge for cognitive modelling. *Cognitive Science*, Volume 26, Issue 1, sid 85-112.
- [Castor et al., 2002] Castor, M., Wallin, N., Nilsson S-Å., & Moradi, F. (2002). Datorgenererade styrkor - Metoder och möjligheter. FOI-R--0688--SE.
- [Castor et al., 2003] 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.

- [Gustafsson, 2002] Gustafson, S. (2002). Modeling the Terrorist Decision of a Closed State Regime. I Proceedings of BRIMS 2002, sid 465-472.
- [Law and Kelton, 1999] Law, A. M. and Kelton, W. D. 1999. Simulation Modeling and Analysis. McGraw – Hill Higher Education.
- [Lundin, 2004] Lundin, M. (2004). Simulating the effects of mental workload on performance. LIU-KOGVET-D—04/xx—SE, Linköping Electronic Press, Sweden.
- [Pew and Mavor, 1998] Pew, R. W. and Mavor, A. S., Eds. 1998. *Modeling Human and Organizational Behavior Application to Military Simulations*. National Academy Press, Washington, D.C.
- [Silverman et al., 2001] Silverman, B., Johns, M., Shin, H., & Weaver, R. (2000). Performance Mode-rator Functions for Human Behavior Modeling in Military Simulations.

## 4 Kravspecifikation

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.

Det fastställda syftet eller måldefinitionen för en modell och dess användning utgör grund för kravspecifikation. I FEDEP-processen tas kravspecifikationer fram under det andra steget, Konceptuell analys. Kravspecifikationen utgör sedan indata till utveckling av konceptuell modell men ger också underlag till VV&A-arbete genom att den bör resultera i både valideringskriterier för den konceptuella modellen och acceptanskriterier för den färdiga modellen.

Man kan dela upp krav i tre områden:

- Funktionella krav
- Krav på verklighetsöverensstämmelse
- Operationella krav.

De funktionella kraven talar om vad modellen ska kunna göra, d.v.s. vilka funktioner som skall finnas med. När väl de funktionella kraven är definierade återstår att definiera hur väl de modellerade funktionerna skall efterlikna verkligheten. Dessa är de svåraste kraven att definiera. Operationella krav är av typen att modellen skall kunna köras på en viss arbetsstation under ett visst operativsystem etc.

DMSO har i sin VV&A Recommended Practice Guide [RPG, Build 2.5] under Special topics mycket material bl. a om kravspecifikationer och hur de hänger ihop med VV&A-arbetet att läsa för den vetgirige och intresserade.

I [NATO SAS017, 2001] görs en enkel uppdelning i fyra nivåer av modellens 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”, d.v.s en expert har bedömt dem 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 efterliknar 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. Nedan presenteras en del av en tabell från [NATO SAS017, 2001], 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 / 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	
Organisations- förändringar	Hög	
Uppdragsplanering	Blandat (Låg till Hög)	
Förövning “Mission Rehearsal”	Blandat (Låg till Hög)	
Planering av logistik och underhållsopera- tioner (t.ex biståndsaktion)	Anges inte i den del av tabellen som återges här, men bör vara <b>låg</b> om det är aktuellt.	
Planering på långsiktig strategisk nivå (tex förutsäga säkerhets- politisk utveckling och krav)	Anges inte i den del av tabellen som återges här, men bör vara <b>låg</b> om det är aktuellt.	
Planering på höga taktiska ledningsnivåer (t.ex ledning NATO- koalition)	Låg	
Informationsopera- tioner / Asymmetrisk Krigföring	Hög	

## Referenser

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

[RPG, Build 2.5] DMSO VV&A Recommended Practice Guide (RPG), ([vva.dmsomil/](http://vva.dmsomil/)).

## 5 Konceptuell modell

En konceptuell modell är en övergripande beskrivning av den modellerade verkligheten. Den kan uttryckas i mer eller mindre formella språk. Att använda ett strikt formellt språk gör att missförstånd minimeras men gör ofta att modellen blir svår att begripa och kommunicera vilket man vill undvika.

Innan man skapar den konceptuella modellen bör man inhämta den kunskap som behövs inom området. För att begränsa kunskapsmängden bör man ställa upp ett antal scenarier eller situationer som modellen skall kunna agera i.

Kunskapsanskaffning kan göras på flera olika sätt som t.ex. intervjuer med s.k. domänexperter, manualer, utbildningsmaterial, observationer, eget sunt förnuft, etc.

### 5.1 Kunskapsanskaffning (Knowledge acquisition, KA)

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, d.v.s. 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.

## 5.2 Verktyg för kunskapsanskaffning

Kunskapsanskaffning kan göras på flera olika sätt som t.ex. intervjuer med domänexperter (SME, Subject Matter Expert), manualer, utbildningsmaterial, observationer, eget sunt förnuft, etc. Olika metoder har olika för- och nackdelar, och det är sällan som en enskild metod ensam kan utgöra grund för den konceptuella modellen [Nilsson et al., 2004].

### 5.2.1 Manualer, utbildningsmaterial

Utifrån manualer och utbildningsmaterial så brukar det gå ganska enkelt att hitta en grund för det beteende man vill skapa. Manualerna beskriver beteendet i en form som är lätt att förstå, undviker tvetydigheter, och förklarar också varför beteendet gäller. Problemet är att manualerna ofta beskriver beteendet på en ganska hög nivå, sällan tar upp specialfall eller variationer och tar inte hänsyn till mänskliga kognitiva begränsningar.

### 5.2.2 Intervjuer och diskussioner med domänexperter

Kunskapsuppbyggnaden är alltså oftast en tidskrävande och kostsam process och i [NATO 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.

Utöver dessa punkter kan nämnas Medelvärdesproblemet (Averaging problem). Olika domänexperter kan lösa problemen olika, t.ex. kan en pilot välja att göra en manöver efter 10s medan en annan gör manövern efter 20s. Ett sätt att sammanställa detta beteende är att ta medeltiden d.v.s. 15s för modellens beteende. Problemet är att

modellens beteende inte stämmer överrens med någon av piloterna och det kanske visar sig att ingen pilot någonsin väntar 15s.

### 5.2.3 Observationer, mätningar

Att observera ett beteende är det enda sättet där man faktiskt ser vad som egentligen händer. Man kan också få en bild av vilka problem och variationer som kan uppstå i en skarp situation. Det stora problemet är dock att det är svårt att förstå varför det händer, om man inte har tidigare domänkännedom. Det kan också vara svårt att veta vad man skall observera. Att t.ex. observera en trupps förflyttningar utan att även höra kommunikationen inom truppen, och utan att känna till ledarens intentioner kan leda till stor förvirring.

## 5.3 Mall för beteendedata för CGF-utveckling

Som det noteras i [Archer et al., 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 simulering 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" [Andersson, 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.

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.

I [Castor och Wallin, 2004] presenteras en mall som är skriven för en taktisk grundmanöver för en enskild plattform och innehåller förutom en tabell med beteendet nedbrutet i s.k. beteendepremitiver även en del hypotetisk information (framför allt under avsnittet PMFs Influencing). För att beskriva hur dessa beteendepremitiver relaterar till varandra behövs någon uppgiftsanalysformalism som HTA (Hierarchical Task Analysis) 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.

Som kommenterats tidigare råder det 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 mer eller mindre goda gissningar 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.

## Referenser

- [Andersson, 1993] Andersson, J., 1993, Rules of Mind (Chapter 1: Production Systems and the ACT-R Theory). Hillsdale, NJ: Erlbaum.
- [Archer et al., (2003)] 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.
- [Castor och Wallin, 2004] Castor, M. Wallin, N. Krav på beteendedata till CGF-modeller. FOI MEMO 1016, 2004.
- [MSIAC, 2001] MSIAC (2001). Human Behavior Representation Validation. [http://www.msiac.dmsomil/vva/Special\\_topics/HBR-validation/hbr-validation.htm](http://www.msiac.dmsomil/vva/Special_topics/HBR-validation/hbr-validation.htm). Besökt 2004-08-12.

- [NATO SAS017, 2001] NATO RTO SAS017 (2001). Human Behavior Representation. NATO RTO Technical Report 47, SAS Panel.
- [Nilsson et al., 2004] Nilsson, S-Å. Allberg, H. Lundin, M. Wallin, N. Castor, M. Datorgenererade styrkor med mänskligt beteende. FOI-R--1489—SE, 2004, Användarrapport.
- [Pearson och Laird, 2004] Pearson, D., & Laird, J. (2004). Redux: Example-Driven Diagrammatic Tools for Rapid Knowledge Acquisition. I Proceeding of BRIMS 2004, sid 185-192.

## 6 Design

I designfasen skall man utifrån de scenarier och den kunskap man utvecklat skapa en tillräckligt god grund för en exekverbar modell. Har man en riktigt god och tydlig design ska en utvecklare kunna transformera designen till en exekverbar modell utan att själv ha deltagit i de tidigare faserna. Det är vid utvecklingsfasen som eventuella brister i designen upptäcks. För att minimera eller helst eliminera dessa brukar en designmodell formaliseras med hjälp av någon sorts formellt språk eller notation. Traditionellt så brukar man inom mjukvaruutveckling använda UML eller liknande diagramorienterade beskrivningar. Inom kognitionsvetenskapen används bl.a. HTA och GOMS.

Val av utvecklingsmodell, t. ex vattenfallsmodell eller iterativ modellutveckling, kan komma att tillåta olika noggrannhet i designfasen.

### 6.1 Interaktioner

Ett viktigt steg är att identifiera vilka interaktioner beteendemodellen kan tänkas använda sig av. Interaktioner kan delas upp i sensorinformation, d.v.s. hur modellen får input från omvärlden samt handlingar, d.v.s. hur modellen ska kunna påverka omvärlden via s.k. aktuatorer. Om en beteendemodell agerar som en operatör i ett fordon så är operatörsplatsens visuella, audiella och taktila indikationer exempel på sensorer, medan styrspakar, reglage och knappar är exempel på aktuatorer.

Exempel på interaktioner är:

- Omvärldsinformation i form av avstånd, riktningar, klassificeringar av andra objekt (sensorer)
- Kommunikation med andra modeller (sensorer och aktuatorer)
- En modell kan även interagera med sig själv, vilket kan skapa interna interaktioner

Om man på ett tidigt stadium kan fastställa vilka interaktioner som skall ingå i modellen så blir det lättare dels för den som utvecklar beteendemodellen och dels för den som ska integrera modellen i en simuleringsmiljö då de får tydliga riktlinjer att arbeta efter. För vissa arkitekturer kan sena justeringar för modellens sensorer och aktuatorer påverka regeldatabasen lavinartat. Det kan även uppstå problem då t.ex. simuleringsmiljön inte kan erbjuda den sensorinformation som beteendemodellen kräver. Det kan också bli en avvägning om vad som anses vara färdig sensorinformation och vad som modellen själv ska beräkna.

### 6.2 Kunskapsbearbetning (KE)

Då man specificerat vilken information som en beteendemodell kan få via stimuli från omvärlden samt hur modellen själv kan påverka omvärlden, är det dags att formalisera de processer som ligger till grund för modellens beteende. Inom beteendemodellering

finns det flera mer eller mindre formella metoder för att bearbeta den kunskap som anskaffats. Exempel på uppgiftsanalysmetoder finns på "mentalmodels" hemsida<sup>3</sup>. Tanken är att man utifrån kunskapsanskaffningen och den konceptuella modellen skall skapa en implementationsoberoende designmodell.

### **HTA (Hierarchical Task Analysis)**

Hierarchical Task Analysis [Kirwan and Ainsworth, 1992] är en uppgiftsanalysmetod som används för att skapa beskrivningar av uppdrag och mänskliga verksamheter i termer av mål, operationer och planer.

Mål står för systemets önskade läge, t.ex. förstörelse av en bro eller ett luftrum utan fientliga flygplan. En operation är en beteendeenhet som exekveras för att uppnå ett mål. Denna beteendeenhet kan beskrivas i olika komplexitets- och varaktighetsnivåer. I en kontext kan en operation vara att patrullera på ett flygfält, medan en annan operation kan vara att röra en styrspak. Båda operationerna kan vara delar av samma mål, men vara beskrivna på olika detaljnivåer. En plan anger det villkor som bestämmer när varje operation skall utföras. Planer inkluderar ofta en rad handlingar som är kopplade till tid eller processvillkor.

Ibland behöver högnivåoperationer delas i ett antal deloperationer med egna delmål. Uppdelningsprocessen kan fortsätta tills önskad detaljnivå har uppnåtts. Många kognitiva teorier påstår att alla organismer ständigt försöker att uppnå ett antal mål som i sin tur är delar av högre mål och det var därför HTA tekniken ansågs vara en lämplig beskrivningsmetod. Genom att använda en effektiv dokumenteringsmetod, som både experter och modellutvecklare lätt kan begripa, förbättras både datainsamling och valideringsprocessen. Dokumentering skapar också ett lager mellan experter och implementation, vilket möjliggör att skapa olika modeller och implementationer från samma datamängd.

### **GOMS (Goals, Operators, Methods, Selection rules)**

En GOMS-modell är en beskrivning av den kunskap en användare måste ha för att kunna utföra uppgifter genom ett system [Kieras, 1994]. GOMS är alltså en beskrivning av de metoder som behövs för att uppfylla ett specificerat mål. Metoderna är uppbyggda av en serie operatorer och delmål så att dessa bildar en hierarkisk struktur. Om det finns flera metoder för att uppfylla ett mål finns det också regler som väljer ut en av metoderna (selection rules). I likhet med HTA kräver GOMS att analyseraren identifierar mål, delmål, operatorer och regler. Lika viktigt är att man identifierar sådant som inte skall tas med utifrån de krav som ställts på modellen.

GOMS finns i flera olika dialekter. Den kanske vanligaste dialekten för kognitiv modellering är "Natural" GOMS Language, NGOMSL. NGOMSL har en direkt relation till regelbaserade system, kräver inte någon kunskap om det underliggande regelsystemet. Till skillnad från t.ex. HTA så har GOMS en formell notation vilken kan direkt exekveras av en användare eller ett GOMS-interpretörande program. Ibland beskrivs GOMS som "sinnets programmeringsspråk" ("The programming language of the mind"). Detta medför bl.a. att man i NGOMSL inte kan skriva komplicerade villkorssatser eller komplexa algoritmer eftersom människan inte förmodas klara av sådan behandling i ett enda kognitivt steg. NGOMSL bygger också på teorier om människans minnesrymder, arbetsminne och långtidsminne, genom att stödja

---

<sup>3</sup> [http://mentalmodels.mitre.org/cog\\_eng/ce\\_methods\\_1.htm](http://mentalmodels.mitre.org/cog_eng/ce_methods_1.htm)

manipulation av dessa i modellbeskrivningen. Eftersom GOMS har en formell notation, kan det vara svårt att läsa av ”koden” för att försöka förstå syftet med modellerna. Därför bör en GOMS-beskrivning åtföljas av någon annan, icke exekverbar, dokumentationsform.

### **CMN-GOMS (Card Moran Newell GOMS)**

CMN-GOMS är en mer detaljerad version av GOMS och beskriver en uppgift i en hierarkisk form av metoder vilka var och en består av steg som exekveras i strikt sekventiell ordning.

### **UML (The Unified Modeling Language)**

UML består av en samling grafiska notationer som används för att beskriva mjukvarusystem. De officiella diagramtyperna är: Activity, Class, Communication, Component, Deployment, Object, Package, Sequence, State Machine, Use Case, Composite Structure, Interaction Overview and Timing diagrams. Att använda UML för att beskriva beteenden är inte så vanligt. Fördelen är dock att beteendemodellerna och de mjukvarusystem (simuleringsmiljöer) som de interagerar med får en enhetlig representation, samt att UML numera är standard i mjukvaruutvecklingsprojekt.

### **Referenser**

- [Kirwan and Ainsworth, 1992] Kirwan, B. & Ainsworth, L., 1992. *A guide to task analysis*. London: Taylor & Francis.
- [Kieras, 1994] Kieras, 1994. *A guide to GOMS Task Analysis*. University of Michigan.

## 7 Utveckling

### 7.1 Arkitekturer

En arkitektur för beteendemodeller är den delen av systemet som tillhandahåller och sköter de primitiva funktionerna hos en beteendemodell. En modellutvecklare kan med hjälp av dessa funktioner skapa egna modeller, oftast med hjälp av regler. En beteendemodell består därför av dels regler och dels arkitekturs tjänster för att generera beteendet. Vilka tjänster som finns, hur de fungerar, etc. bestämmer arkitekturs egenskaper. En beteendemodell som är implementerad i arkitekturen kan sägas ärva arkitekturs egenskaper då den i hög grad blir beroende men också begränsad av den underliggande arkitekturen.

Förutom det genererade beteendet så finns det en del andra kriterier som kan vara viktiga vid val av arkitektur:

- **Interoperabilitet**, hur lätt är det att interagera med simuleringsmiljöer (API:er, HLA)?
- **Användarvänlighet**, hur lätt är det att skapa beteenden (grafiska gränssnitt, färdiga funktioner, etc.)?
- **Spridning**, finns det många användare så är även många problem lösta
- **Resurskrav**, vad krävs i processorhastighet, minne, flera datorer, andra programvaror?
- **Analys**, går det att följa vad en modell gör, ser, tänker och har för intentioner?
- **Kostnad**, produktens pris, samt tid att bekanta sig med verktyget.
- **Konfiguration**, hur pass lätt är det att konfigurera, modifiera verktyget, finns tillgång till källkod?

Ofta går det att få en uppfattning om verktyget, genom att titta på hur andra projekt har lyckats, d.v.s. finns det liknande projekt som har använd verktyget. Idag finns en mängd olika arkitekturer som används inom olika områden, här följer en indelning av dessa och en kort presentation av dem vi har funnit vara de viktigaste:

### 7.2 Kognitiva arkitekturer

En kognitiv arkitektur är ett domänoberoende beräkningssystem som försöker att efterlikna mänsklig kognition genom att stödja t.ex. generell problemlösning, målorienterat beteende, inläring, reaktionstid, etc. En kognitiv arkitektur kan användas för att utveckla många olika typer av teorier eller kognitiva modeller. Då en kognitiv arkitektur till skillnad från en kognitiv modell försöker att tillhandahålla funktioner för att modellera "alla" kognitiva processer så bygger de ofta på någon teori som med ett antal mekanismer kan förklara dessa processer. En sådan teori brukar kallas "unified theory of cognition (UTC)". Soar och Act-R gör båda anspråk på att omfattas av denna teori. Eftersom en kognitiv arkitektur är realiserad som en programvara så kan arkitekturen ses som en större eller mindre förenkling av dess teori. En arkitektur måste dock tillföras domänspecifik information och förutsättningar för att kunna bilda en modell ur vilken man kan generera meningsfulla resultat.

Beteende = Innehåll x Arkitektur

Vanligtvis används en kognitiv arkitektur inom flera olika forskningsområden eftersom den kan ligga till grund för många olika typer av beteenden. Fördelen med att använda en kognitiv arkitektur är just att den är väldigt flexibel och att det eftersökta beteendet ofta går att realisera med en sådan arkitektur. En modell som realiserar ett symboliskt system med hjälp av en kognitiv arkitektur brukar ofta vara lättare att förstå, då sådana modeller ofta kan förklara varför de beter sig på ett visst sätt. Vill man studera mänskligt beteende i stället för optimalt beteende så är en kognitiv arkitektur att föredra.

Den ökade flexibiliteten gör dock att det krävs en hel del av den som implementerar beteendet. Beteendemodelleringen kan i vissa fall liknas vid programmering, där arkitekturen tillhandahåller ett programmeringsspråk. En kognitiv arkitektur har sällan inbyggd funktionalitet för t.ex. formationsbeteenden, navigering, kommunikation, etc. utan detta är något som en utvecklare får implementera själv eller eventuellt använda andra samverkande systems funktioner för.

### 7.2.1 Vanliga kognitiva arkitekturer

#### **ACT-R** (The Adaptive Control of Thought – Rational)

ACT-R är en hybridarkitektur, där ett symboliskt system med fakta och regler är realiserat på ett subsymboliskt system med t.ex. Bayesiansk inlärning. Brukar räknas som den arkitektur som har bäst kognitionsvetenskaplig validitet. Har använts i flera militära tillämpningar och även i spel (Unreal).

#### **Cognet/iGEN** (Cognition as a Network of Tasks)

*Chi Systems*

Cognet är en symbolisk arkitektur. Cognet/iGEN använder sig av s.k. cognitive tasks som kan aktiveras av och påverka ett arbetsminne. En perceptuell och en motorisk process används för att transformera externa händelser till symbolisk information i arbetsminnet.

#### **Soar** (State Operator And Result)

*University of Michigan/SoarTech*

Soar är en symbolisk kognitiv arkitektur skapad av Allen Newell, Paul Rosenbloom och John Laird. Soar är både en s.k. UTC samt en implementation av den teorin i form av programvara. Soar använder sig av symboler i form av arbetsminne samt produktionsregler ("if"- "then") i form av ett långtidsminne som kan manipulera symbolerna i arbetsminnet. Har använts i många olika simuleringar (TacAir), spel (Quake, Unreal) och tillsammans med olika CGF- och AI-arkitekturer (VR-Forces, AI-Implant, PMFServe).

### 7.2.2 Beteendemoderatorer

Vissa observationer av beteende saknar stöd i den kognitiva modell som en arkitektur representerar. Aggression, stress, rädsla, utmattning är typiska exempel på faktorer som påverkar den kognitiva processen. Hur dessa faktorer förändrar ett beteende kan variera kraftigt beroende på situation och individ, vilket gör det svårt att införliva dem i den

kognitiva arkitekturs modell. I stället behåller man den ursprungliga modellen och justerar dess processer för att uppnå en högre grad av realism.

**PMFServ** (Performance Modifier Function Server) är ett ramverk specialiserat för beteendemoderatorer och för framtagning av agenter med mänskliga egenskaper. Ramverket baseras på psykologiska teorier som är operationaliserade i PMFServs delkomponenter. Man kan säga att PMFServ fokuserar på de "mjuka faktorerna" som oftast ignoreras inom stridssimulering. Exempel på variabler som ramverket hanterar är yttre stressfaktorer, sorg, nervositet, påverkan av ryktesspridning, samt hur dessa påverkar beslutsfattande. Dessutom finns nätverk för att simulera fysiologiska variabler som utmattning och skador. PMFServ har använts för att simulera kulturella faktorer, gruppbeteenden, emotioner och personlighet.

## 7.3 AI-arkitekturer

En AI-arkitektur kan definieras som en mängd tekniker och algoritmer för att lösa och optimera vanliga problem såsom kortaste vägen, flockrörelser, etc. En AI-arkitektur har sällan en uniform representation utan består av skilda moduler för att lösa specifika men domänoberoende problem.

### 7.3.1 Vanliga AI-arkitekturer

#### **DirectIA** (Direct Intelligent Adaption)

*Mathematiques Appliquees*

DirectIA är ett franskutvecklat hjälpmedel för att skapa intelligenta agenter. Utvecklingsmiljön innefattar funktionalitet för perception, motivation, reaktivitet och kanske mest intressant: inlärning. Inlärning gör det möjligt för en agent att lära sig av misstag och framsteg, att imitera beteenden och att anpassa sin strategi utifrån förväntningar om fienden. Produktionsreglerna i DirectIA styr influenser av agentens tillstånd snarare än direkta beteenden. Tillståndet kan sedan generera olika beteenden i olika kontexter. Förutom tillståndsmaskinen påstås DirectIA av tillverkarna ha en biologisk grund.

#### **AI.Implant,**

*Engenuity (BioGraphic) Technologies*

AI.Implant är en verktygslåda för att generera syntetiska aktörer främst till datorspel och animationer, men även simuleringar. Verktuget fokuserar på att tillhandahålla funktioner för rörelseanimationer, "path-finding", flockrörelser, etc. AI.Implant har plugin moduler för t.ex. Maya och 3D Studio Max för att grafiskt kunna modellera beteenden.

## 7.4 CGF-arkitekturer

En CGF-arkitektur är en arkitektur som kan generera beteenden inom den militära domänen. Vissa CGF-arkitektur är specialiserade för t.ex. stridsflygdomänen med andra försöker skapa "generell" funktionalitet för olika militära ändamål. En CGF-arkitektur utnyttjar ofta AI-tekniker för att lösa problem men har sällan några referenser till mänskligt beteende. Ofta ingår scenariogenereringsverktyg, farkostmodeller, uppspelning av simuleringen, etc. Dessa egenskaper gäller även för de flesta simuleringsramverk. Ett simuleringsramverk med beteendemodeller som kan



agera i andra miljöer än sin egen kan räknas som en CGF-arkitektur. CGF-verktyg har sällan heller en modell av en operatör utan operatören och farkosten bildar en enhet. Detta gör att det inte går att t.ex. modellera enskilda operatörer i en farkost och hur dessa samverkar.

#### 7.4.1 Vanliga CGF-arkitekturer

##### **VR-Forces**

*MäK Technologies*

VR-Forces är ett simuleringsverktyg som kan generera och exekvera simuleringar på olika nivåer. Det finns grafiska gränssnitt för att styra s.k. semi-automated CGF:er genom att t.ex. klicka fram styrpunkter. Det finns också ett script-språk med vilket komplexare beteenden kan genereras. CGF:erna kan "exporteras" via HLA till andra simuleringar. VR-Forces har använts bl.a. tillsammans med kognitiva arkitekturer, där VR-Forces har behandlat enklare manövrar medan den kognitiva arkitekturen har använts för högre nivåer av resonerande.

##### **Strive**

*CAE*

I likhet med VR-Forces är Strive ett komplett simuleringsramverk, med tillhörande CGF-funktionalitet. HLA används för att exportera CGF:er. CGF:ernas beteende kan specificeras med hjälp av villkorssatser.

Andra kända CGF-arkitekturer är **OneSAF** och **Export CGF** (*Lockheed Martin*). Men om dessa saknar vi för närvarande aktuell information.

##### **TACSI**

*Saab Systems*

Inom flygstrid är Saabs TACSI (TACTical SIMulation Tool Suite) ett avancerat verktyg för taktikutveckling och operationsanalys. Det är tänkt att antingen nyttjas fristående eller som HLA-federat i en träningsimulator. Eftersom ramverket bygger på realistiska flygplansmodeller kan det även användas för studier av prestation. TACSI består av ett antal komponenter:

- En simuleringsmotor för att köra scenarier med visualiseringar.
- Ett modellverktyg för specificering av flygmodeller
- Ett scenarioverktyg som sätter utgångsvärdena för scenariot.
- Ett taktikverktyg med tillståndsmaskiner och produktionsregler

Det är främst taktikverktyget som är av intresse för beteendemodellering. I varje tillstånd kan logiska regler skapas för att styra modellernas beteenden. Nackdelen är att egna variabler och premisser inte kan skapas, utan villkoren är knutna till de inbyggda komponenterna.

## 8 Integration

I integrationsfasen är det dags att integrera beteendemodellen med simuleringsramverket, d.v.s. översätta simuleringsramverkets objektdata till beteendemodellens sensorer samt översätta modellens aktuatordata till simuleringsramverket. Det kan röra sig om att konvertera enheter till att faktiskt exekvera komplexa beräkningar, t.ex. då simuleringsmiljön inte direkt kan lämna ifrån sig viss information. För CGF-arkitekturer är mycket av integrationsarbetet redan gjort då de tillhandahåller en egen exekveringsmiljö och även HLA-stöd för att ”exportera beteendet” till andra miljöer.

Vid integrationen så bör man försöka se till att främja återanvändning av modellerna. Detta kan göras genom att man försöker skapa generella gränssnitt som kan anpassas till simuleringsmiljöernas modeller. Tanken är att en modell som är konstruerad att kunna flyga ett JAS-flygplan, ska kunna göra detta oavsett vilken simuleringsmiljö modellen befinner sig i. På så sätt kan man enkelt återanvända modellen för att agera i olika miljöer. De flesta CGF-arkitekturer har automatiskt stöd för HLA, där gränssnittsspecifikationen redan är gjord. Arbetar man efter ovanstående princip så kan de modeller man utvecklar samlas i ett bibliotek av återanvändbara s.k. ”plug’n’play”-modeller.

Under slutet av år 2002 utformades vid FOI ett förslag på ett ramverk för utveckling och nyttjande av beteendemodeller inom FM. Förslaget presenterades i projektets årsrapport för år 2002 [Castor et al., 2002]. Ramverket har sedan utvecklats vidare och använts för projektets prototyputveckling. Tanken är att man med ett sådant hjälpmedel ska kunna koncentrera sig på själva beteendemodelleringen, vilket ger en snabbare och mer effektiv modellutveckling.

### 8.1 Biblioteksramverk

Ramverket bygger på tekniker för komponentbaserad mjukvaruutveckling där grundstenen är den komponentmodell som utvecklats av FOI och FMV och som bl.a. använts i Merlin. Tanken med en sådan komponentbaserad teknologi är att kunna skapa återanvändbara delmodeller som enkelt skall kunna sättas samman till större enheter. För att kunna åstadkomma detta måste delmodellerna erbjuda vissa standardiserade gränssnitt.

Ramverket innehåller dessa gränssnitt, utvecklings och integrationstjänster samt den underliggande arkitektur som kan knyta ihop delmodellerna.

Allteftersom modellkomponenterna implementeras skall de införas i ett bibliotek så att de kan användas inom FM. Ett försvarsgemensamt bibliotek av både kompletta beteendemodeller och delmodeller kommer då att successivt byggas upp och revideras efter behov. Från modellbiblioteket kan man sedan hämta lämpliga modeller för användning i något simuleringsramverk.

#### 8.1.1 Integrationsverktyg

Integrationsverktygen skall göra det enklare att nyttja beteendemodeller eller agenter i befintliga simuleringsramverk, spel, etc. Målet är att ge modellerna s.k. plug’n’play funktionalitet i så hög grad som möjligt. Sådan funktionalitet återfinns bl.a. i webbläsare och designapplikationer i form av insticksprogram. Dessa insticksprogram

är direkt riktade mot de applikationer som de skall ingå i. I fallet med beteendemodellerna, vill man att de ska kunna integreras i en mängd kända och okända simuleringsramverk. Detta gör att man istället får anpassa simuleringsramverken till den arkitektur som beteendemodellerna utnyttjar. Anpassningen behövs i regel göras en enda gång och därefter kan man utnyttja en komplett plug'n'play funktionalitet.

Verktygen består av en uppsättning gränssnitt som skall utgöra beteendemodellernas syn på simuleringsramverkens omvärldsmodeller samt de medel som modellen har att påverka denna omvärld. Ett exempel är en pilotagent som skall agera i en flygsimulator och flyga en av ramverkets flygplansmodeller. Typiska gränssnitt skulle kunna förmedla planets indikatorer, d.v.s. agenten kan läsa av planets hastighet, planets alfavinkel, geografisk information, etc. Detta ger agenten en chans att observera den syntetiska omgivningen. Ett annat gränssnitt skulle kunna ge agenten en möjlighet att ange en ny riktning eller höjd för flygplansmodellen. Genom att kommunicera med detta gränssnitt kan agenten påverka omvärlden. Genom att utöka uppsättningen gränssnitt kan man se till att agenten får mer information om omvärlden samt flera möjligheter att påverka denna.

Antalet gränssnitt som en agent behöver beror helt på agentens uppgift. Även om en agent har tillgång till en mängd olika gränssnitt behöver inte det betyda att alla gränssnitt måste utnyttjas fullt ut eller alls. Precis som i det verkliga fallet beror utnyttjandet av gränssnitten helt på hur agenten resonerar och då vilka hjälpmedel den behöver.

Om en agent vill utnyttja ett gränssnitt som simuleringsramverket inte kan erbjuda bör dock simuleringen avbrytas då beteendet troligtvis kommer att försämrats drastiskt. Man vill således försöka hitta de gränssnitt som ger tillräcklig information och chans att påverka omvärlden för att en beteendemodell ska kunna utföra sin uppgift autonomt och trovärdigt.

Ett hårdare krav är att gränssnitten skall kunna erbjudas av "alla" simuleringsramverk. I pilotfallet innebär det att alla flygplansmodeller i olika ramverk skall kunna erbjuda samma gränssnitt. En agent kan således agera mot alla dessa flygplansmodeller, vilket innebär att agenten kan återanvändas i alla dessa ramverk. Problemet blir då att hitta dessa gemensamma gränssnitt, och därefter anpassa simuleringsramverkens modeller till dessa. Denna anpassning är ofta endast en syntaktisk översättning, då semantiken redan behandlats i utformandet av gränssnitten.

Som tidigare nämnts behövs gränssnitts Anpassningen av simuleringsramverken bara göras en enda gång, därefter kan man enkelt stoppa in färdiga beteendemodeller eller agenter som kan kommunicera via dessa gränssnitt. Beteendemodellerna kan återanvändas och bytas ut mot nya modeller. Val av beteendemodeller kan göras utan att behöva modifiera vare sig simuleringsramverken eller beteendemodellerna.

I verktygen ingår också en del hjälpmedel eller tjänster för att förenkla anpassningen av simuleringsramverken. Dessa anpassningstjänster hjälper till att översätta ramverkens interna modeller så att de kan erbjuda gränssnitten. Ett ramverk kanske representerar sina modellattribut med vektorer, medan en beteendemodell i de flesta fall resonerar om skalära storheter. Ett annat fall är då en agent vill styra en styrspak, medan modellen bara tar emot önskade riktningssignaler. Exempel på tjänster kan vara att konvertera vektorer till skalära storheter eller överföra styrspakskommandon till riktningssignaler.

## Referenser

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

## 9 VV&A

Verifiering, validering och ackreditering (VV&A) av simuleringsmodeller är en process som syftar till att öka tillförlitligheten hos simuleringsmodeller och att ge användaren av en modell information om dess lämplighet för ett givet syfte samt om vilken tilltro han kan sätta till modellen. Denna process bidrar även till att fel och ofullständigheter ofta upptäcks tidigt under utvecklingsprocessen, vilket kan vara av betydande ekonomiskt intresse [André et al.]. VV&A-arbete bör således pågå som en parallell process i modellutvecklingsarbetet.

Vid verifiering, validering och ackreditering av datorgenererade styrkor uppkommer ett antal specifika frågor.

För att kunna validera såväl den konceptuella modellen som den kompletta simuleringsmodellen krävs en referent av verkligheten mot vilken modellen skall valideras. Referenten kan baseras på information från domänexperter, empiriska observationer, experimentella data, validerade modeller av olika aspekter av mänskligt beteende eller andra informationskällor.

Den enklaste formen av validering är expertbedömningar både m.a.p. agentens militära och kognitiva agerande. Dessa kan med fördel användas i ett tidigt skede av valideringsprocessen, men även som ett komplement till övriga valideringsmetoder.

För att testa modellens kognitiva kvalitet finns en hel del kognitiva modeller och experimentella data med vilka jämförelse kan göras. Denna form av testning är lämpad för modeller som inkluderar effekter som t.ex. stress och sinnesrörelse.

Modeller som avbildar grupper av interagerande individer måste även innefatta socialt beteende. I detta avseende kan validering göras mot såväl modeller som beskriver sociala fenomen som experimentella observationer.

### 9.1 Definitioner

Kortfattat kan man säga att verifiering syftar till att besvara frågan ”Är modellen rätt enligt specifikationen”, medan validering skall belysa frågan ”Är modellen en rimlig approximation av verkligheten ”och ackreditering ”Är modellen godkänd för aktuellt syfte”.

Definitionerna nedan följer DoD Instruction (DoDI) 5000.61 och är de som används i DMSO:s Recommended Practice Guide. [RPG Build 2.5]:

*Verifiering* är en process som avgör om en modellimplementation och dess tillhörande data representerar utvecklarens idébeskrivning och specifikationer.

*Validering* är en process som avgör till vilken grad en modell och dess tillhörande data är en riktig representation av verkligheten ur perspektivet av modellens användning.

*Ackreditering* är ett officiellt bemyndigande att en modell, simulering eller federation av modeller eller simuleringar och dess tillhörande data är acceptabelt att användas för ett visst syfte.

*Certifiering* är en process som fastställer att data är lämpliga för en viss tillämpning.

## 9.2 Sex viktiga faser under en VV&A-process

I VV&A-processen kan följande sex viktiga faser urskiljas:

1. Definition av VV&A- kriterier
2. Validering av den konceptuella modellen
3. Verifiering av design
4. Kodverifiering
5. Verifiering och certifiering av data
6. Validering av hela tillämpningen

### 9.2.1 Definition av VV&A- kriterier.

Det är viktigt att man inför en modellutveckling definierar kriterier mot vilka VV&A-arbetet skall kunna värderas. Dessa kan vara av tre olika typer:

- Funktionella krav
- Krav på verklighetsöverensstämmelse
- Operationella krav.

De funktionella kraven talar om vad modellen ska kunna göra, d.v.s. vilka funktioner som skall finnas med. När väl de funktionella kraven är definierade återstår att definiera hur väl de modellerade funktionerna skall efterlikna verkligheten. Dessa är de svåraste kraven att definiera. Operationella krav är av typen att modellen skall kunna köras på en viss arbetsstation under ett visst operativsystem etc.

VV&A-kriterierna skall avspegla kundens krav och är helt beroende av modellens syfte.

### 9.2.2 Validering av den konceptuella modellen

Den konceptuella modellen är grunden för den fortsatta modellutvecklingen varför det är ytterst viktigt att den är ordentligt validerad innan man går vidare i modellutvecklingsarbetet. Man måste här försäkra sig om att de antaganden och algoritmer som ligger till grund för den konceptuella modellen kan förväntas vara en acceptabel representation av systemet för den tänkta applikationen.

### 9.2.3 Verifiering av design

När den konceptuella modellen är validerad gäller det att vid designen av datormodellen se till att den konceptuella modellen överförs till en datormodell på ett sådant sätt att simuleringsmekanismerna avspeglar intentionerna med systemet, d.v.s. designen måste ordentligt verifieras mot den konceptuella modellen.

### 9.2.4 Kodverifiering

När implementeringen av designen har resulterat i kod skall denna ordentligt verifieras, så att den är fri från programmeringsfel.

### 9.2.5 Verifiering och certifiering av data

Trovärdigheten för de resultat en modell producerar blir aldrig högre än trovärdigheten för de data modellen baserar sig på. Felaktiga data kan resultera i allvarliga fel i simuleringsmodellen, varför det är viktigt att se till att de data man använder är tillräckliga, riktiga och lämpliga för aktuell tillämpning.

### 9.2.6 Validering av hela tillämpningen

För att försäkra sig om att den kompletta modellen är tillräckligt noggrann för den aktuella tillämpningen jämförs den med känt eller förväntat uppträdande hos det system den skall representera.

### 9.3 Alternativa VV&A-processer

Eftersom VV&A-behovet har blivit allt mer uppenbart pågår arbete i flera internationella grupper att ta fram bättre processer för VV&A. Dessa arbeten syftar till att ta fram standarder för VV&A. Några exempel är:

#### **REVVA**

Förslag till en generisk modell för VV&A. REVVA innehåller en beskrivning både av processen, produkter och organisation i VV&A-arbetet. (WEAG THALES J. P. 11.20)

#### **SISO PDG (Product Development Group) VV&A**

Detta arbete beskriver vilka VV&A-aktiviteter som skall göras under FEDEP:s sju faser.

#### **ITOP (International Test Operations Procedure)**

Delar upp VV&A-arbetet i modell, data och användare och koncentrerar sig dessutom på att utveckla mallar som skall underlätta VV&A-arbetet och dokumentationen.

#### **NATO MSG 19**

Denna grupp arbetar med att utforma en RPG för VV&A-arbete för federationer.

### 9.4 Specifika VV&A-frågor som uppkommer vid utveckling av modeller som avbildar mänskligt beteende

Vid verifiering, validering och ackreditering av datorgenererade styrkor (Computer Generated Forces - CGF:er) uppkommer ett antal specifika frågor. Vi betraktar här endast den del av valideringen som berör modellering av mänskligt beteende (Human Behavior Representation – HBR:er), d.v.s den del som berör agenten. V&V-arbetet för resten av modellen följer samma principer som för modeller som ej hanterar mänskligt beteende.

För att kunna validera såväl den konceptuella modellen som den kompletta simuleringsmodellen krävs en referent av verkligheten mot vilken modellen skall valideras. Referenten kan baseras på information från domänexperter (Subject Matter Experts – SME:er), empiriska observationer, experimentella data, validerade modeller av olika aspekter av mänskligt beteende eller andra informationskällor. En referent måste uppfylla åtminstone tre grundläggande krav:

- Den måste finnas tillgänglig
- Användarna måste betrakta den som trovärdig
- Den måste var lika detaljerad som CGF:en

Den enklaste formen av validering är expertbedömningar både m.a.p. agentens militära och kognitiva agerande. Dessa kan med fördel användas i ett tidigt skede av valideringsprocessen, men även som ett komplement till övriga valideringsmetoder.

Subject Matter Experts (SME:er) har kunskap inom sin domän om vilket resultat som ett typiskt mänskligt beteende skulle producera och kan bedöma hur realistiskt detta är representerat i modellen. Detta går bra för en modell som baseras på enkla regler, men modeller med en mer abstrakt kunskapsrepresentation kan endast valideras genom att undersöka det observerbara uppträdandet som genereras vid exekvering av olika scenarier för en tillämpning. Icke-linjäriteten hos mänskligt beteende och den enorma mängden av möjliga utfall vid ett scenario gör det omöjligt att till fullo validera en modell på detta sätt. Detta tillsammans med den subjektiva naturen av denna typ av test

gör detta till den svagaste formen av validering [RPG, Build 2.5]. I de fall då kunskapsdatabasen är otillräcklig och kostnaderna för datainsamling för höga eller konsekvenser för människor och/eller miljö för allvarliga för att utföra verkliga experiment, kan expertbedömningar vara den enda utvägen.

För att testa modellens kognitiva kvalitet finns en hel del kognitiva modeller och experimentella data med vilka jämförelse kan göras. Denna form av testning är lämpad för modeller som inkluderar effekter som t.ex. stress och sinnesrörelse

Modeller som avbildar grupper av interagerande individer måste även innefatta socialt beteende. I detta avseende kan validering göras mot såväl modeller som beskriver sociala fenomen som experimentella observationer.

Finns tillgång till historiska data som insamlats under tidigare simuleringar för olika militära träningsändamål, och som är relevanta i detta sammanhang, kan utdata från modellen jämföras med dessa eller med andra validerade modeller. Många gånger kan det vara lämpligt att genomföra egna experimentella jämförelser.

Som för alla typer av modeller är dokumentationen viktig. Ackrediteringsunderlaget bör inkludera standarder för dokumentation som förklarar hur man kör och modifierar modellen och en plan för hur modellen skall underhållas och uppgraderas.

#### **9.4.1 Definition av VV&A-kriterier**

Vid definitionen av VV&A-kriterier gäller att definiera vilka mänskliga roller som skall representeras, vilka aspekter av mänskligt beteende som skall modelleras och hur väl agentens beteende måste efterlikna referentens beteende. Det är viktigt att kundens krav på modellen direkt kan kopplas till VV&A-kriterierna. I en enkel modell kan det räcka att modellera beslut och reaktionstid, men om modellen kräver en högre grad av trovärdighet måste även t.ex. psykologiska och sociologiska inflytanden modelleras. Önskvärt är att dessa krav kan formuleras tidigt, men för vissa ändamål kanske kraven på modellen inte kan specificeras i detalj från början, varför VV&A-kriterierna då får specificeras på en relativt hög nivå och förfinas under arbetets gång. Å andra sidan är mänskligt beteende mycket komplext och det gäller att endast modellera de delar som är relevanta för tillämpningen. Vid formulering av krav på verklighetsöverensstämmelse uppkommer också den ytterligare svårigheten att kunna mäta mänsklig prestation.

#### **9.4.2 Validering av den konceptuella modellen**

Den konceptuella modellen är kanske den största stötestenen och den viktigaste byggstenen vid utveckling av CGF:er. Den beskriver tre punkter som är väsentliga för utveckling och implementering av modellen [Castor et al., 2002]:

- kontexten i vilken CGF:erna byggs, t.ex. flygstridsdomänen
- information som beskriver vad som påverkar CGF:ens beteende och hur
- information om den totala simuleringen, t.ex. hur simuleringsenheter interagerar och kopplas ihop.

Den konceptuella modellen skall valideras med avseende på två olika aspekter:

- Är de regler som ligger till grund för agentens handlande riktiga enligt den militära doktrinen?
- Finns de för tillämpningen viktiga beteendefaktorerna med och är de rätt modellerade?



För att avgöra om de regler som ligger till grund för en agents beteende är riktiga enligt doktrinen, kan man utnyttja militära experter. För att validera om agentens beteende på ett korrekt sätt representerar mänskligt kognitivt beteende, kan experter på detta område utnyttjas eller i de fall det existerar tillämpliga modeller kan jämförelse med dessa göras.

#### **9.4.3 Verifiering av design och implementation**

V&V av designen och implementationen följer samma principer som för andra typer av modeller. De flesta simuleringar av en agents beteende är dock stokastiska och kräver upprepade körningar med liknande men icke identiska begynnelsevillkor för att verifiera att uppförandet är som utlovats. [Enciso, Gonzales, 2003] föreslår formella metoder för att verifiera modeller med avseende på specifikations- och implementationsfel.

#### **9.4.4 Verifiering och certifiering av data**

För V&A av CGF:er talar man snarare om kunskapsbaser än data. En av de viktigaste bristerna i nuvarande förmåga att validera modeller av mänskligt beteende är avsaknaden av tillgängliga databaser av konsistent och kvalitetssäkrad information rörande mänskligt beteende. Visserligen finns idag en hel del studier och data rörande mänskligt beteende, däremot saknas kunskap och erfarenhet om validering av detsamma. Vidare åligger det den individuella valideraren eller utvecklaren att lokalisera lämpliga valideringsdata och att sedan transformera datamängden till ett konsistent och meningsfullt underlag [RPG, Build 2.5].

För att kunna avgöra om data är tillräckliga, riktiga och lämpliga för den aktuella tillämpningen är det av största vikt att VV&A-kriterierna är ordentligt formulerade.

#### **9.4.5 Validering av den kompletta modellen**

Simuleringsresultatet skall valideras mot en referent. Beroende på VV&A-kriterierna och tillgängliga data kan olika metoder komplettera varandra.

Ett led i denna validering är expertutlåtanden. Personer med kunskap om det verkliga systemet kan bedöma om simuleringsresultaten ser rimliga ut. Turingtest är en annan informell valideringsmetod som innebär att experter ombeds skilja mellan två utdatauppsättningar. Den ena kommer från det verkliga systemet och den andra från simuleringsmodellen, då båda matats med samma indata. Kan experterna avgöra vilken som kommer från simuleringsmodellen får de beskriva skillnaderna, om de ej kan skilja på resultaten stärker detta tilltron till modellen.

Då man har tillgång till andra validerade modeller kan jämförelse göras med dessa. Jämförelser kan också göras med historiska data. Det kan också vara lämpligt att genomföra experimentella jämförelser genom att t.ex.

- Låta experter utarbeta ett antal scenarier (med facit) och se hur många modellen predikterar rätt. Om simuleringsmodellen är stokastisk måste ett antal simuleringar köras på varje scenario.
- Ge samma indata till en bemannad simulator som en CGF och jämföra utdata, eller jämföra beteendet hos en mänsklig beslutsfattare och en CGF som opererar i samma simuleringsmiljö.

En enstaka observation som visar att en referent och en agent utfört olika handlingar är inte tillräcklig information för att avgöra om ett fel föreligger. I de fall då modellen föreskriver att en agent kan ta ett felaktigt beslut på grund av för hög arbetsbelastning måste ett större antal försök med simuleringsmodellen genomföras. Vanligen finns

dessutom oftast inte endast ett korrekt uppträdande för ett uppdrag. CGF:er är ofta avsedda att operera i simuleringar med en hög verklighetstrogenhet och i verkliga miljöer där problem troligen har multipla lösningar. [Wallace, Laird, 2002] beskriver en metod för en automatisk jämförelse mellan en agents och en mänsklig operatörs beteende och identifiering av olika typer av avvikelser.

Även om de ingående delmodellerna tidigare är validerade är det ytterst viktigt att den kompletta modellen blir ordentligt validerad. Många CGF:er bygger på redan befintliga modeller av mänskligt beteende. Av [Pew and Mavor, 1998] framgår att många av de individuella micro-modellerna är baserade på validerade modeller, men det framgår inte att den övergripande har validerats. Största delen av valideringsarbetet med de validerade modellerna har baserats på subjektiva värderingar av SME:er och inte på verkliga mänskliga prestationsdata, varför de bör användas med försiktighet.

## 9.5 Området idag

Området VV&A av modeller rörande mänskligt beteende är idag behäftat med många svårigheter och man har inte nått samma trovärdighetsgrad som för modeller av fysiska system.

De valideringsmetoder vi använder idag bygger enligt ovan på expertutlåtanden, befintliga modeller, experimentella data, historiska data eller andra arkivkällor. Dessa är alla behäftade med svagheter:

- Expertutlåtanden är en enkel form av validering som har brister i form av sin subjektiva natur och det begränsade beteenderum som är möjligt att utforska.
- Experimentella data rörande mänskligt beteende i olika militära situationer är ofta mycket sparsamt förekommande och endast tillämpbara på begränsade situationer. Därtill kommer att kvaliteten på data ofta är bristfälligt kontrollerad och karakteriserad.
- Många omfattande kognitiva modeller existerar, men relativt få av dem är ordentligt validerade. Vidare baseras dessa ofta på mycket begränsade beteenderum.
- En mängd data rörande människans psykologi och sociala beteende finns, men är ofta svåråtkomliga och av osäker kvalitet.

Dessa problem tenderar att försvaga valideringsarbetet. Existerande kunskap rörande validering av kunskapsbaserade system erbjuder en del teori, tekniker, redskap och erfarenheter. Dessa resurser appliceras emellertid i första hand på regelbaserade expertsystem.

[Wallace 2003] pekar ut ett antal anledningar till varför det kan vara mycket svårt att avgöra om en avvikelse från ett förväntat 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.

## Referenser

- [RPG, Build 2.5] DMSO VV&A Recommended Practice Guide (RPG), (vva.dmsomil/)
- [Enciso, Gonzales, 2003] Enisco Lauto, Gonzales Avelino J.: *Verification of Human Performance through Formal Methods*. Proceedings of the Twelfth Conference on Computer-Generated-Forces and Behavior Representation. 2003
- [Pew and Mavor, 1998] Pew Richard W and Mavor Anne S: *Modeling Human and Organizational Behavior. Application to military simulations*, National Academy Press, Washington D.C. 1998
- [Castor et al., 2002] Castor Martin, Wallin Niklas, Nilsson Sten-Åke, Moradi Farshad: *Datorgenererade styrkor – Metoder och möjligheter*. FOI-R-0688—SE, Användarrapport, 2002.
- [André et al.] André Tommy, Bergsten Ulla, Sporre Lena: *Verifiering, validering och ackreditering av simuleringsmodeller – en förstudie*. FOI-R--99-01196.202--SE.
- [Wallace, Laird, 2002] Wallace, Laird: *Towards Automatic Knowledge Validation*. Proceedings of the Eleventh Conference on Computer-Generated-Forces and Behavior Representation 2002.
- [Wallace, 2003] Validating Complex Agent Behavior. Doktorsavhandling vid University of Michigan. 2003.

## 10 Begrepp och förkortningar

Beteendemodell	modeller som ger ett trovärdigt observerbart mänskligt beteende.
Kognitiv modell	modell som i någon eller stor utsträckning modellerar kognitiva funktioner. Kan också kallas "functional model" eller kognitiv processmodell.
CGF-arkitektur	arkitektur som kan generera beteenden.
Kognitiv arkitektur	beräkningssystem som försöker att efterlikna mänsklig kognition.
Intelligent agent	agent som har ett tillräckligt komplext beteende för att fungera i komplexa situationer där den interagerar med andra agenter eller människor och en simulerad värld.
ACT-R	Adaptive Control of Thought – Rational
AI	Artificiell intelligens
AMBR	Agent-based Modeling and Behavioral Representation
API	Application Programming Interface
CGF	Computer Generated Forces
Cognet	COgnition as a NEtwork of Tasks
DMSO	Defense Modeling and Simulation
EPIC	Executive-Process Interactive Control
FEDEP	Federation Development and Execution Process
GOMS	Goals, Operators, Methods, Selection rules
HBR	Human Behavior Representation
HFM	Human Factors and Medicine
HLA	High Level Architecture
HTA	Hierarchical Task Analysis
KA	Knowledge acquisition, kunskapsanskaffning
KE	Knowledge engineering, kunskapsbearbetning
M&S	Modeling & Simulation, Modellering och simulering
MSIAC	Modeling and Simulation Information Analysis Centers
NRC	National Research Council
OneSAF	Semi automated forces
PMF	Performance Modifier Functions
RPG	Recommended Practice Guide
RTO	Research and Technology Organisation
SAS	Studies, Analysis and Simulation
Soar	State, operator and result
Stage	Scenariogenereringsverktyg och system för realtidssimulering
Strive	CGF-arkitektur
TACSI	Arkitektur och system för taktisk simulering
UML	Unified Modeling Language
UTC	Unified Theory of Cognition
VR-Forces	CGF-arkitektur
VV&A	Verifiering, validering och ackreditering