Towards a Generic Behaviour Modelling Interface

Emiel Ubink, M. Sc., Wouter Lotens, PhD., Arend Woering, PhD.

emiel.ubink@tno.nl

ABSTRACT

NATO Task Group 128 identifies that behaviour in military simulations is often too mechanistic in nature. In this paper we will argue that behaviour in simulations can be improved if the two central aspects of behaviour modelling, i.e. action selection *and* performance modelling*, are not addressed as separate issues, but as interacting instead. We will describe how the interaction between these aspects may be improved by the introduction of an interfacing layer, consisting of human capabilities. Although capability based modelling is not new, it is usually limited to cognition and information processing. It is our view that the use of capabilities in HBR should be expanded to include perception, motor skills and physiology. We will also describe the CHAOS behaviour model, that uses capabilities as a behaviour modelling interface. CHAOS can therefore be seen as an illustration and proof of concept of the ideas proposed in this paper.*

1.0 INTRODUCTION

NATO Task Group 128 identifies the following problem in relation to Human Behaviour Representation (HBR) for military simulations: "*Most simulations handle military units as if they are robots, carefully representing only mechanical qualities*" [1]. One of the reasons for this lack of realism may be found in the fact that *action selection* (which actions are executed in a certain situation) and *performance modelling* (how well are the selected actions executed) are often treated as independent processes. In this paper we will argue that these two aspects of HBR should not be addressed independently. We will describe how the interaction between these aspects may be improved by the introduction of an interfacing layer, consisting of *capabilities*.

1.1 Action Selection and Performance Modelling

Many methods for action selection have been developed, mainly by the artificial intelligence (AI) community. These methods include, among others, neural networks and reinforcement learning, expert systems and BDI agents. For an extensive overview of these methods, see Russel and Norvig [2] and Woolridge [3]. Research in the field of human factors (HF) has resulted in a multitude of models that can be used to represent the human condition. The variables from these models, that describe the human state, can in turn be used with so called performance shaping functions (PSF), to predict performance on some task, ranging from the effects of heat on physical performance to the effects of multitasking on cognitive performance. For an overview of (cognitive) behaviour and performance modelling, see Pew and Mavor [4], Ness et al. [5].

It can be concluded then that many methods and models exist that address action selection or performance modelling. However, expected performance of actions may influence which action is preferred, whereas a combination of selected actions can influence the performance on these actions (e.g. effects of multi-tasking). A behaviour representation scheme should therefore address performance shaping and action selection as interacting processes. But, when taking into account that the factors that influence action selection or performance can range from equipment to environment and from physiology to psychology, the resulting requirements for HBR can become quite complicated. To deal with this complexity, a structure is needed that can act as an interface between performance shaping and action selection modules. We will argue in this paper that such an interface can be created by defining human capabilities, such as *attention* or *aerobic capacity*, that are relevant to the behavior that is to be represented.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
OCT 2010	**N/A**	**-**

4. TITLE AND SUBTITLE	
Towards a Generic Behaviour Modelling Interface	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **NATO**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S)
	11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES

See also ADA564696. Human Modelling for Military Application (Applications militaires de la modelisation humaine). RTO-MP-HFM-202

14. ABSTRACT

NATO Task Group 128 identifies that behaviour in military simulations is often too mechanistic in nature. In this paper we will argue that behaviour in simulations can be improved if the two central aspects of behaviour modelling, i.e. action selection and performance modelling, are not addressed as separate issues, but as interacting instead. We will describe how the interaction between these aspects may be improved by the introduction of an interfacing layer, consisting of human capabilities. Although capability based modelling is not new, it is usually limited to cognition and information processing. It is our view that the use of capabilities in HBR should be expanded to include perception, motor skills and physiology. We will also describe the CHAOS behaviour model, that uses capabilities as a behaviour modelling interface. CHAOS can therefore be seen as an illustration and proof of concept of the ideas proposed in this paper.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**SAR**	**10**	

1.2 Relation to Existing HBR Research

The idea of using capabilities as an integrated part of HBR is not new. In many cognitive architectures (see Pew and Mavor [4] for an overview), capabilities (often referred to as *resources*) such as attention or working memory, are used. One of the best examples of capability based modelling is probably the Multiple Resource Theory by Wickens [6], a framework for predicting effects on performance when multiple tasks are concurrently executed. Although many examples of capability-based HBR exist, all these examples seem to be limited to the modelling of cognition and information processing tasks.

It is our view that the use of capabilities in HBR should be expanded to include not only information processing, but also perception, motor skills and physiology. Also, capabilities should be used to integrate action selection and performance modelling. Using capabilities in such a broad sense will require that several difficult questions are addressed. For instance: which capabilities should be defined? How can we quantify these capabilities? How much of the capability is required for a task and how should the capabilities be distributed between different tasks?

These questions will be addressed in this paper. We will also describe the CHAOS behaviour model, that uses capabilities as a behaviour modelling interface. CHAOS can therefore be seen as an illustration and proof of concept of the ideas proposed in this paper.

1.3 Nomenclature

Before we can continue, a note on the use of the words “action” and “task” in this paper is required. In AI it is common to use the word “action” (e.g. action selection), where in the HF community the word “task” is preferred (e.g. task performance). Therefore, in this paper “action” will be used in relation to action selection and “task” in relation to performance modelling. Although these words may have different connotations in their respective domains, these differences are not really relevant to the points made in this paper, so it is assumed that they are interchangeable. “Goal directed activity” may be used as a working definition for both terms in this paper.

2.0 A BEHAVIOUR MODELLING INTERFACE

In the previous section we argued that an interface is needed that connects performance shaping to action selection methods. In other words: we need to combine a description of the human state, including performance effects, with the actions or behaviours that may be executed. A logical candidate to be used as such an interface would be a collection of human *capabilities,* since capabilities affect performance and also determine which actions are feasible. In this section we will describe exactly how performance and action selection can interact with capabilities and what the benefits of such an approach would be.

2.1 Performance Shaping

A common approach to human performance research is the following: determine the effect of some manipulation *m* (body temperature, anxiety, training status, having decision support, etc.) on the performance on some task *t* (running on treadmill, flying a jet fighter, monitoring a display, etc.). The resulting data can be used to construct a model (or performance shaping function, or PSF) that predicts the performance on task *t* (or similar tasks), given a certain body temperature, anxiety level, training level, or type of decision support. For many applications this is a very useful approach. However, if we want to create a constructive simulation of military operations, we need models of human behaviour that can integrate multiple PSF’s into a behaviour selection module and that is when things start to get difficult. Three problems arise in this case.

The first problem is that it is difficult to combine PSF's that affect the same task. Suppose the task is to move quickly from A to B by foot, and we have a PSF that predicts the effect of body temperature on walking and another PSF that predicts the effect of exhaustion on walking. The question that arises then is how performance should be modified if both body temperature and exhaustion are at performance-affecting levels? The second problem is that it is cumbersome to model task performance of tasks for which no dedicated PSF's are available. For instance: we may want to model the performance of walking and riding a bicycle, but we only have PSF's that can be used to predict walking speed. How should we then apply these PSF's to the (in some aspects similar) task of riding a bicycle? The third problem has to do with expanding the simulation with new tasks and PSF's. Since PSF's are connected *directly* to task performance, adding new PSF's or new tasks to the simulation can be cumbersome. New tasks should be connected to all relevant PSF's that are already present in the simulation. When adding a new PSF, the implementation of all tasks that are to be influenced by the new PSF, needs to be reconsidered. So in both cases existing implementations of tasks and PSF's need to be checked and possibly changed.

2.1.1 Performance Shaping and Capabilities

Now the question is if an interfacing layer of capabilities could help to solve the problems identified above. The introduction of such an intermediate layer would mean that performance shaping is split into two stages. In the first stage, capabilities are affected, based on low level model variables describing the human state. In the second stage, task performance is determined, based on the capability levels that were affected in stage one. So instead of PSF's, we would need "Capability Shaping Functions" (CSF's) for stage one and "Task Performance Functions" (TPF's) for stage two (Figure 1). Now how does this help us with the three problems we identified?

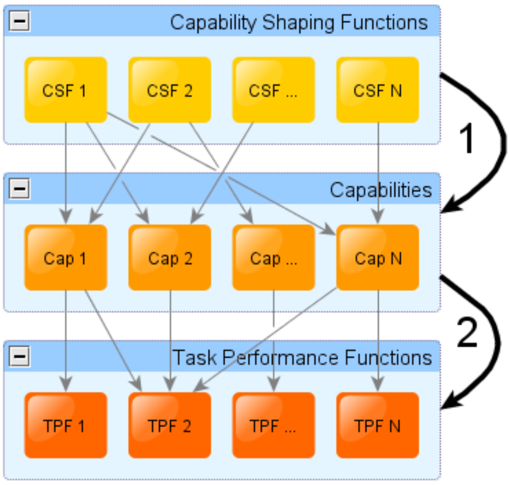

Figure 1: Performance shaping in two steps. In the first step "Capability Shaping Functions" affect the middle layer of capabilities, in the second step task performance is determined by "Task Performance Functions".

The first problem (how to combine PSF's that affect the same task) is, in the scheme represented in Figure 1, simply dealt with by connecting CSF's that have similar effects to the same capabilities. For instance: heat stress and exhaustion both impair physical performance. This could be modelled by introducing a capability *aerobic capacity* that is influenced (in this case impaired) by two CSF's that represent heat stress and exhaustion. Any tasks that depend on aerobic capacity will then *automatically* be affected by both CSF's. Of course, we still need data on the combined effects of heat stress and exhaustion, but this was the case anyway. Introducing capabilities as an interface has at least simplified the implementation side of things. And even if no data exists (which is unfortunately not uncommon for combined performance effects), we can still use common sense to predict what the combined effect on the capabilities would be. It would probably be more difficult to assess the combined effect on task performance directly, since task performance may depend on several other factors as well (such as, in the case of walking, the load that is carried).

Some instances of the second problem (predicting performance for tasks for which no dedicated PSF's are available) may be partially solved by the introduction of capabilities. By using capabilities as an interface, any task that depends on (a subset of) these capabilities, will be affected by all CSF's that affect those capabilities. So in our example: heat stress and exhaustion will both impair aerobic capacity, which in turn will affect the performance on walking and riding a bike (and similar physical tasks). Of course, other performance effects relevant to bike riding would still need to be added, but we have at least used some of the existing CSF's to model bike riding performance.

The third problem (the problem of adding new tasks or PSF's) is solved almost completely by the introduction of an interfacing layer of capabilities. Since tasks are connected to capabilities and not directly to PSF's, adding a new task is "simply" a matter of connecting the task to the capabilities in the system. There is no need to change the implementation of any of the existing tasks or CSF's. This allows then for *pluggable* architectures, in which tasks can be plugged in or removed from the system without affecting the integrity of the system as a whole. The same goes for adding new PSF's, or in this case CSF's: new CSF's can be connected to the capabilities and all tasks that depend on those capabilities will automatically be influenced by the newly added CSF, without any reimplementation of the tasks. Note that some effort may need to go into considering the effects on capabilities of the new CSF in combination with other, existing CSF's.

2.2 Action Selection

As was explained in section 1, HBR is not only a matter of performance modelling, but also of action selection. Usually the process of action selection starts with the identification and prioritization of goals and sub-goals. Goals will be derived from the assigned task, or from internal stimuli, or stimuli in the environment that require to be acted upon. This process of goal identification and prioritization will not be discussed here, but will be addressed in the description of the CHAOS behaviour model (section 3.1.2). For the sake of simplicity it is assumed here that a set of (sub-) goals with an appropriate prioritization exists. So provided that a set of prioritized (sub-) goals exists, a common approach to selecting an appropriate action is to select the action that gives the greatest chance of achieving the (sub-) goal with the highest priority, i.e. the action with the greatest expected *utility*. Once an action is selected to be executed, the performance on that action may be "shaped" by PSF's.

So, the usual approach is to first select the action and then "shape" the performance. But to determine the actual utility of an action, it would be better to also take the expected performance into account, since actual utility will increase with performance. So, for example, if two actions are comparable with respect to achieving a goal, the action which is most likely to succeed should of course be favoured. A similar argument holds for multi-tasking, i.e. when multiple goals are pursued simultaneously. Some combinations of goals may result in a set of conflicting actions. If that is the case, performance should be degraded (e.g. use of mobile phone while driving) or some actions should not be executed at all (e.g. walking and riding a bicycle at the same time). It turns out then that, for HBR, action selection is not merely a matter of comparing expected utilities. The effects of expected performance on utility should also be accounted for, as well as the effects of multitasking.

2.2.1 Action Selection and Capabilities

When using an HBR system that is built around a set of capabilities, it is relatively easy to integrate expected performance and multi-tasking effects into the action selection process. The first requirement can be met by having CSF's affect capabilities *before* actions are selected. It is then possible to incorporate expected performance into the action selection process, since from the affected capabilities the expected performance can be determined. The second requirement can be met if, with each selected action, the required capabilities are seized. This would mean that fewer capabilities are available for any actions that are considered next. This would result in suboptimal performance on tasks with lower priority (e.g. driving while using mobile phone) and would prevent conflicting tasks to be performed simultaneously (e.g. simultaneously walking and riding a bicycle).

2.3 Identifying Capabilities

In the previous sections is discussed how capabilities can be used to integrate performance modelling and action selection. An important question that comes to mind then is: how do we know which capabilities should be defined and used? This of course completely depends on the domain and level of detail that is needed. The generic and pragmatic answer to this question is therefore that those capabilities should be defined that allow the behaviours and stressors in the simulation to be “connected”. This is of course a very generic answer. Luckily, we can get some good clues for useful capabilities by looking at stressors and tasks.

In the example of physical performance given above, where heat stress and exhaustion would both impair physical performance, a capability “aerobic capacity” was postulated. This capability actually represents the common denominator of tasks and stressors related to physical performance; it allows to “connect” all stressors that impair physical performance to all tasks that require physical performance and is as such a perfect candidate capability for an interface of capabilities.

A capability can also be used to represent the conflict between behaviours. Some actions simply can not be executed simultaneously; we already saw the example of riding a bicycle and walking. If for a group of actions such a conflict is identified, this conflict could be modelled by the introduction of a capability. We could in this case define a capability “gross motor ability” and declare that it is needed for any action requiring gross motor skills. This capability could then be used in the simulation to prevent that multiple actions, that require gross motor skills, are executed simultaneously.

3.0 THE CHAOS BEHAVIOUR MODEL

We have been arguing that the use of capabilities in HBR offers important advantages, and that capabilities should not be limited to the modelling of cognitive processes or multi-tasking. Since the proof of the pudding is in the eating, we will in this section describe the CHAOS behaviour model, a computer implementation of these ideas.

3.1 The CHAOS Behaviour Model

The CHAOS (Capability-based Human-performance Architecture for Operational Simulations) behaviour model [7], is inspired on the pandemonium model of letter recognition as proposed by Selfridge [8]. Selfridge’s model consists of demons that are “shrieking” for attention in an arena called the pandemonium. Where in Selfridge’s pandemonium the demons represent different stages of feature recognition, the demons in CHAOS represent either *behaviours* (actions or tasks) or *stressors*. In CHAOS, the shrieking level of a behaviour demon represents its importance or priority and the shrieking level of a stress demon is a representation of the stress level. CHAOS contains, besides demons, also capabilities that play a central role in the performance modelling and action selection mechanisms.

The central algorithm in CHAOS is basically a four-step procedure that is repeated each time-step:

1) **Capability levels are initialized.** The status of the capabilities may be affected as a result of the previous iteration, so the capabilities are set to their initial state. Also, effects that traits may have

on capabilities can be effectuated in this step.

2) **Shrieking levels are updated.** Some demons have a fixed shrieking level, for instance, demons that represent an assigned task. A fixed shrieking level then represents the importance of that task in the scenario. Demons with a fixed shrieking level will ignore this step. Other demons have dynamic shrieking levels. These demons will in most cases represent some form of stress and they can adjust their shrieking level according to the level of stress.

3) **Stress-demons affect capabilities.** In CHAOS, the CSF's from Figure 1 are actually demons that represent some form of stress. In this step these stress demons can affect (increase or decrease) capability levels, according to their shrieking level, i.e. according to the stress level. This could be viewed as the transformation of stress into strain.

4) **Demons take actions.** In this step demons that represent behaviour are requested to take actions, starting with the demon that is shrieking loudest. This demon will determine if the capabilities it requires are available. If so, it will seize these capabilities and will execute its behaviour. If not, it will do nothing. Then the next demon in line (the loudest demon of the rest), checks if the capabilities it requires, are available. If so, it will seize these capabilities and execute its behaviour. This process continues until the last demon has had a chance to execute its behaviour.

This is, in a nutshell, how the CHAOS model works. In the next sections we will elaborate on performance modelling and action selection in CHAOS.

3.1.1 Performance Modelling in CHAOS

Performance modelling in CHAOS is split into two stages, similar to the process depicted in Figure 1. In CHAOS, the CSF's from Figure 1 are actually demons that represent some form of stress. These demons will start to "shriek" louder as the stress level rises and they will affect the capability levels according to their shrieking level. For example: suppose we want to model thermal stress, i.e. stress related to the thermal condition of a person. In that case we can implement a "thermal stress demon" that monitors the person's body temperature, calculated in a thermo-physiological model. This demon will start shrieking as soon as the temperature exceeds a certain threshold, and will shriek louder as body temperature rises. To be more precise, the demon would "know" in what range the variable affects performance, i.e. is considered stressful, as depicted in Figure 2. The stress level would then be reflected by the demon's shrieking level. This demon could then simulate the effects of thermal stress on performance (strain), by impairing the capabilities of the system (in step 3 of the algorithm), depending on its shrieking level.

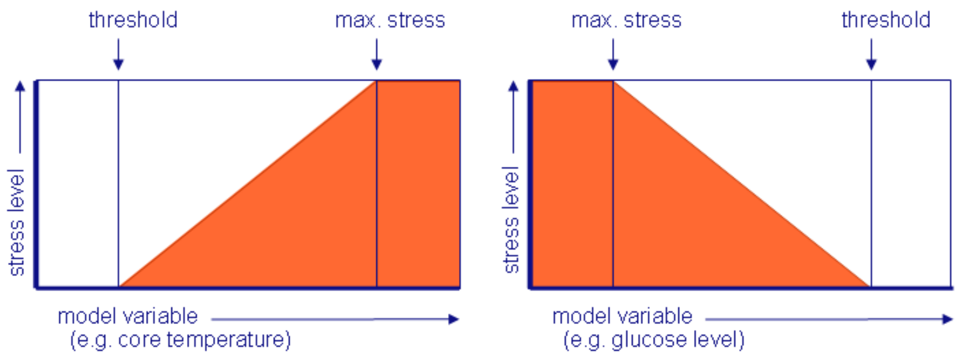

Figure 2: Representing stress in CHAOS. A stress-demon is connected to a model variable, such as body temperature. The stress-demon knows the value at which the variable will start affecting performance (stress-threshold) and also knows at which value the stress is considered maximal. The stress level is interpolated in between. Note that the

stress level can also be reversed to model, for instance, glucose depletion.

The second stage is the stage of determining task performance as a function of available capabilities (the TPF's in Figure 1). The TPF's in CHAOS are integrated into the demons that represent behaviour. In step 4 of the CHAOS algorithm, each demon is allowed to execute its behaviour. However, it must first check if the capabilities it requires are available and sufficient. There are three options here. The first option is that the capabilities are not sufficient, in which case the demon will do nothing. The second option is that the required capabilities are ample, in which case the demon can execute its behaviour optimally. The third option is that there are enough capabilities to execute the behaviour, but only at a suboptimal level. In that case, the demon also plays the role of "Task Performance Function" and has to map the available capabilities to a suboptimal performance level.

So, stress demons can affect the performance by affecting the capabilities behaviour demons require. The effects of multi-tasking can be modelled similarly in CHAOS. When a demon executes its behaviour, it will also seize the capabilities that are needed for that behaviour. This will leave fewer capabilities for the next demons in line that are shrieking less loudly. This may result in those demons only being able to execute their behaviours sub-optimally, if at all. It is also possible that a demon does not *require* a certain capability but still *affects* that capability by executing its behaviour. A good example is the effect of running on tasks that require fine motor skills. Fine motor skills are not *needed* for running but are still *affected by* running. To model this in CHAOS, a "running demon" would, when executing its running behaviour, not only seize the capabilities it needs, but also impair the fine-motor capability. This would mean that the performance of demons that are next in line and that require fine-motor skills, is degraded.

3.1.2 Action Selection in CHAOS

Step 2 in the algorithm described above is the first step in the action selection process. Demons can start shrieking (or can change their shrieking level if they were already shrieking) in reaction to changes in the environment or changes in models that represent the human state. For example: assume a soldier who is taking part in a social patrol suddenly detects enemy troops nearby. A demon that represents *threat perception* will in that case probably start to shriek very loudly. If the situation is threatening enough, this shrieking will be louder than the shrieking of the demon responsible for the social patrol. This means that, in step 4 of the algorithm, the threat demon will be the first to execute its behaviour and seize the capabilities it requires. This behaviour will probably involve taking cover. Since it is not possible to simultaneously take cover and perform a social patrol, a capability is needed that can be used to express this incompatibility. A candidate capability for this could be *gross motor skills*. If this capability is required by both demons, i.e. the demon that is responsible for reacting to threat and the demon that is responsible for carrying out the social patrol, then the result will be that, in case of a threatening situation, the soldier will stop patrolling and take cover instead, which is of course exactly the behaviour we are after.

So the prioritization of behaviours is done in a decentralized manner in CHAOS, by demons that "read" the situation and start shrieking accordingly. As shown, action selection is not merely a matter of prioritizing behaviours, but also a matter of distributing capabilities. The loudest (most important) demon can take the first pick from the capabilities. This will prevent that other, less important, demons can execute conflicting behaviours. However, non-conflicting behaviours, requiring *other* (still available) capabilities, may still be executed.

3.2 Applications of CHAOS

The CHAOS behaviour model is currently used in three applications, all developed by The Netherlands Organization for Applied Scientific Research (TNO): SCOPE, BRIGADE and the Driver Model Library. SCOPE is a simulation of dismounted soldier operations and is used for research in the field of operational performance. In SCOPE [7], CHAOS is used to model soldier behaviour and performance. SCOPE contains, besides CHAOS, models related to visual detection, situational awareness (SA), physical work, thermal stress, threat perception and weapons effects. BRIGADE is in many aspects similar to SCOPE, but is focused on fire-fighter operations and incorporates, among other things, a fire model and a detailed building representation. Finally, the Driver Model Library (DML) [9] is a cross platform plug-in for traffic simulations. In DML, CHAOS is used to model driving tasks and driver performance.

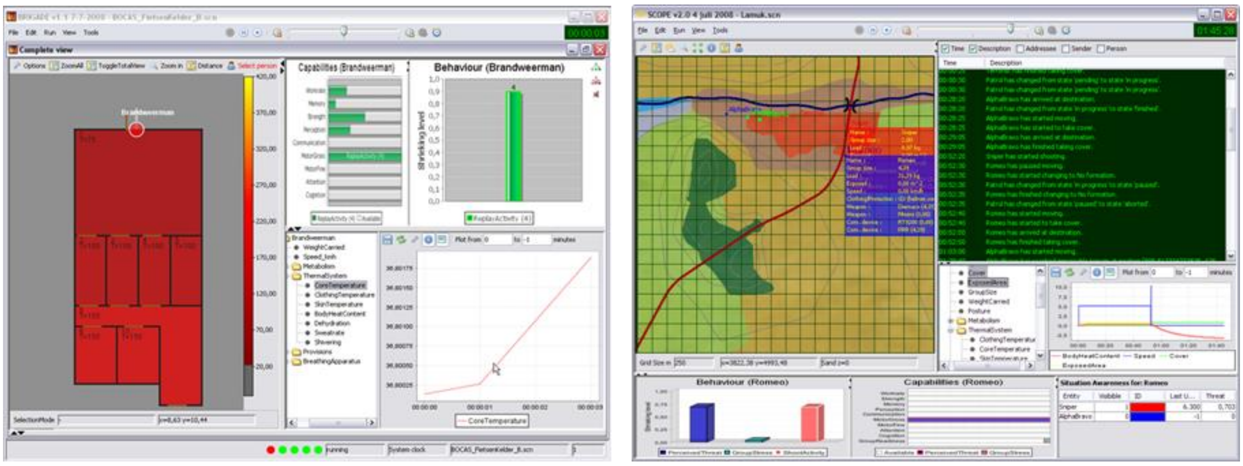

Figure 3: Screenshots of the BRIGADE (left) and SCOPE (right) simulation environments, that both employ CHAOS for behaviour modelling.

4.0 DISCUSSION

The CHAOS behaviour model shows that capabilities can be used to integrate performance shaping and action selection. The implementations of CHAOS in SCOPE and BRIGADE show, to some extent, that capability based modelling can be extended to include not only information processing but also perception, motor skills and physiology. However, as was already noted in the Introduction, several difficult questions related to capability based modelling remain to be answered.

One of these questions is which capabilities should be defined. This is a difficult question since it is not likely that a single, generic set of capabilities can be found that is applicable to each application. However, it is likely that simulations that are similar in respect to domain, level of detail and human factors, will require similar capabilities, so suitable sets of capabilities may be defined for these simulations. In section 2.3 is discussed how useful clues on capabilities can be found by looking at similarities between stressors and tasks.

Another important question is how capabilities can be quantified. For capabilities that are mainly used to model behavioural conflicts, (e.g. *gross motor skills*), quantification is not an issue if the capability is used completely or not at all. For other capabilities, quantification will be necessary. Quantification means that stressors that affect the capability have to define how the capability is impaired at certain stress levels. Tasks that require the capability need to define how performance is affected at certain capability levels. How this quantification should be achieved, and what its validity will be, will depend greatly on the level of correspondence of the capability to actual, quantifiable human factors. For instance, the capability *aerobic capacity* may be represented as oxygen uptake, a commonly used physiological variable. Quantification of other, more abstract capabilities will be more difficult and may therefore need to be estimated.

Although capability based modelling poses some difficult questions, it also offers significant advantages.

An interfacing layer of capabilities can be used to create *pluggable* HBR architectures, in which behaviour and performance models can be plugged in or removed from the HBR system (e.g. the demons in CHAOS), without affecting the integrity of the system as a whole. We have shown that capabilities can also be used for performance modelling and action selection and that they allow for the integration of these processes. And last but not least: capabilities may also provide to be an "interface" between HF research and modelling and simulation. As capabilities are defined that can be studied in HF research and be used for HBR, more human factors can accurately be integrated into military simulations. If HF research is focused more on capabilities than on directly manipulating task performance, the resulting data and models will be more generic and thus easier to integrate into (and exchange between) HBR systems. This all can contribute to increased realism and less mechanistic behaviour in HBR for military simulations.

REFERENCES

[1] HFM 128, *Human Behaviour Representation in Constructive Modelling*. NATO report, NATO website (http://www.rta.nato.int), 2007.

[2] Russell, S.J. and Norvig, P., *Artificial Intelligence: A Modern Approach.* Prentice Hall, Upper Saddle River, NJ, USA, 2003.

[3] Woolridge, M. *Reasoning about Rational Agents.* The MIT Press, Cambridge, Massachussetts/London, England, 2000.

[4] Pew, R.W., Mavor, A. S. (editors), *Modeling Human and Organizational Behavior: Application to Military Simulations*, National Academy Press, Washington, D.C. 1998.

[5] Ness, J.W., Tepe, V., Ritzer, D.R. (editors), *The science and simulation of human performance*, Elsevier, Oxford, England, 2004.

[6] Wickens, C.D., *Multiple Resources and Mental Workload.* Human Factors: The Journal of the Human Factors and Ergonomics Society June 2008 vol. 50 no. 3 449-455

[7] Ubink, E., Aldershoff, F., Lotens W., Woering A., *Behavior modeling through CHAOS for simulation of dismounted soldier operations,* Proceedings SPIE, Vol. 6965, 696504, 2008.

[8] Selfridge, O.G., *Pandemonium: a Paradigm for Learning*, Proc. Symposium on Mechanization of Thought Processes: National Physical Laboratory. London: HM Stationery Office, 1959.

[9] Absil, N.E., Pauwelussen, J., *Driver Model Library, Driver behaviour and decision making framework*, Proceedings of 2010 IEEE International Conference on Systems, Man, and Cybernetics, to be published.