A Cognitive Modelling Approach To Capturing the Context of Complex Behaviour in Gameplay

Manolya Kavakli and Jason R. Thorne, ***School of Information Technology, Charles Sturt University, Bathurst, NSW 2795, Australia,*** mkavakli, jthorne@csu.edu.au

Abstract-- **The goal of this paper is to present a cognitive modelling approach to capturing the context of complex behaviour in gameplay. This cognitive model can be used for the implementation of goal-directed agents in computer games. In this paper, first, we explore the role of context in artificial systems and postulate that contextual mechanisms may be active mental processes intermixed with others. Considering cognition as a complex system, we investigate semiotics by analysing the verbal protocols of game players during game-play. We discuss the cognitive actions involved in game-play and see the interdependency between them. At the end, we focus on modelling cognitive processes to implement artificial agents in computer games.**

Index Terms-- **Games, Intelligent Systems, Cognitive Science, Natural Languages.**

I. CONTEXT OF COGNITION AS A COMPLEX SYSTEM

Virtual environments and virtual reality applications are characterized by human operators interacting with dynamic world models of truly realistic sophistication and complexity [1]. Current research in large-scale virtual environments can link hundreds of people and artificial agents with interactive 3D graphics, massive terrain databases, global hypermedia, and scientific datasets [2]. Related work on teleoperation of robots and devices in remote or hazardous locations further extends the capabilities of human-machine interaction in synthetic computer-generated environments. In this paper, we focus on goal-directed agents as an active area of research driven by artificial intelligence.

We suggest using the term AR (Autonomous Reaction) to define so-called AI (Artificial Intelligence) in computer games. The term AI is used in the games industry for situation where no artificial learning is involved and where computer generated agents react predictably to a given set of conditions. In the present case, the agents do not process complex cognitive behaviour, but autonomously react in certain conditions. Our hypothesis is that it is possible to extend the view of autonomously reactive agents beyond the classical information processing paradigm into a new perspective that takes into account the cognitive, behavioural, and social levels of human activities.

Rappaport [3] addresses cognition as a complex system, highly interactive, where a sort of weaving together of inputs contributes significantly to defining the course of human thoughts and reasoning.

ISBN (1-86467-114-9)

Endowing information infrastructures with the characteristic of being intelligent, consists in establishing a form of "cognitive resonance" between the systems and the minds of their users [3]. The cognitive resonance should be attained by making the intelligent infostructures actively participate in the construction of the context.

> "The cognitive mind is grounded in a set of biological structures that interact with one another and with the external world. Indeed, each functional subsystem does not depend only upon its relations to the others but also on the properties of the whole. An understanding of cognition would result from the sum of the subsystems it involves. It is this basic interdependence between the parts and the whole that makes cognition a **complex system.** This circularity presents a fundamental challenge in the formalization of artificial systems. One answer to this challenge is to build systems that are extensions to the human mind." [3]

The goal of this paper is to explore the possibility of using protocol analysis as a basis for capturing the context of complex behaviour. We suggest to collect verbal protocols of game players during game-play, analyse and code the cognitive actions by using a specific coding scheme, and use this cognitive model to implement artificial agents to interact with the player in a more complex manner.

II. PATHS OF MIND

Representations are actual emerging properties of the mind as a complex system and viewed as expressions of knowledge, the result of a constructive process coding at a higher level the more elementary mechanisms of thoughts. Speech, text, facial expressions, body languages, and certainly other unexplored internal states of the mind all contribute to the representational level. Context and representations are interdependent; their interactions determine both continuity and changes in behaviour [3]. Whether they are symbolic, as with the use of rules, logics and plans, or neuromimetic, as with the use of neural networks, representations in artificial systems do not have to be either complete or clear; in fact, they are more likely to be partial, subjective and ill-defined.

According to Rappaport [3], the mind may follow at times more deterministic paths, based, for example, on learned strategies or well defined problem solving techniques, while at other times it follows less predictable avenues based on the presence of contextual information. It is quite likely that both types of mechanisms are constantly at play in human activities. Introduction of contextual information in an otherwise closed system turns it into complex one. Since complexity is related to a form of continuing constructivism, the context is not only necessary to make sense of our actions relative to environment, but it may also be the source of subjectivity, imaginary attitudes and creativity. Capturing the context is critical to make progress towards the goal of cognitively resonant systems. In this paper, we will investigate the verbal expressions of game players during game-play to capture the context.

III. COMPLEX BEHAVIOUR & AR

Most of the computer games involve game-controlled human or human-like characters in game-play. The AR technology gives agents the ability to react autonomously to environmental stimuli and communicate with other agents in the game to act as a multi-agent system or team.

How can we develop game characters? Animation of human-like characters must eventually be derived from processes that decide-upon the behaviour agents. Thus, the animation becomes the result of the behaviour. In animation, there are two levels of motion that we have to take into account: low-level and high-level. Low-level motion focuses on moving from point A to point B. On the other hand, high level motion focuses on how we know at a point in the game to issue the command to the character: move from point A to point B [4].

How do we motivate the actions of the character to achieve low-level motion? Human-character games are conventionally designed by using pre-calculated sequences which are selected during the game by user intersection. We select and control low-level sequences by means of a low-level module that is capable of appropriately interpreting any high-level commands from an AR module.

How these low-level sequences are generated is the subject of interactive control including different types of interactions such as user-object interaction, object-object interaction, camera-object interaction, and user-scene interaction. In this paper, we are concerned with how a character behaves rather than how the geometry is controlled to execute a behaviour or motion.

Characters need to have a conceptual understanding of the game world and be able to perceive changes in it if they are to react in a human-like manner [4]. At this point, we are face to face with the complexity of emergent behaviour – behaviour that emerges from a program which does not apparently contain explicit instructions that specify the behaviour.

There are two main approaches to implementing behaviour [4]:

A. Deterministic approach

This is an established approach mainly based on variations of the finite state machine (FSM) model. In this approach, the programmer attempts to define the behaviour of the agent by building an FSM which can be viewed simplistically as a look up table. In other words, the programmer determines exactly what is going to happen and writes high-level code accordingly.

B. Non-deterministic approach

This is a new goal based approach. In this approach, instead of trying to account for all possible modes of behaviour, the programmer defines goals and ways in which these can be achieved. The behaviour is determined at run-time. The programmer cannot predict exactly what is going to happen. However, behind the scenes tree-searching strategies determine exactly what is going to happen. Complex behaviour is easier to set up in non-deterministic way. However the run-time computational cost is much higher.

IV. IMPLEMENTING AGENTS IN COMPUTER GAMES

In cyberspace -the digital landscape of the information superhighway- the term avatar generally describes various representations of real people in a computer-generated environment [5]. Avatars are not agents themselves, but they are visual embodiments of agents. Agents are generally thought of as autonomous or semi autonomous pieces of software that may assume some visual embodiment so that we can engage them in a social manner. Agent is a generic term in AI to describe software that perceives a world, thinks, then affects actions on the world [4].

The world may be real – in the case of robotics –or virtual in the case of computer games. In computer games an agent will usually be a character – an opponent or an adviser. We can look upon an agent in a game as a separate software entity communicating with the game world and the player. The game character associated with this agent sees the game world, through a synthetic vision facility and acts on the world according to its AI, moving virtual objects, avoiding obstacles, and so on. Thus an agent can be seen as a module whose decision cycle is executed in (simulated) parallel with the game cycle.

A reactive agent is an entity that immediately maps percepts into actions. A percept is the unit of information supplied to the agent as input. A reactive agent only considers current percepts – it has no memory. Goal-based agents, on the other hand, consider the consequence of actions, rather than simply reacting to the current state of the world. The agent has to consider what will happen if it performs action X. Thus the agent needs to have knowledge or be supplied with goals and strategies that can operate on the current world state to see if and how these goals can be reached. For example, search strategies and path planning strategies are algorithms that result in action sequences that try to achieve the agents' goals.

A game that contains many agents is like many processors all executing in parallel [4]. To implement human-like agents in a game, we need to investigate not only the parallel processes in game-play but also the parallel processes in humans' activities during game-play. The research studies on the structure of concurrent cognitive actions of human beings [6] [7] may give us new ideas about how to implement parallel cognitive processing in human-like agents in computer games.

V. COGNITIVE MODELLING OF BEHAVIOUR

Research in behavioral modeling is making progress towards self-animating characters that react appropriately to perceived environmental stimuli [8]. A combination of procedural animation and behavioral scripting is used in a research project at New York University's Media Research Lab [5]. Procedural animation is a toolset for generating realtime interactive animated human and nonhuman figures. Behavioral scripting is a system for creating synthetic personalities with realistic moods, goals and behaviours and with believable responses to changing situations and stimuli. Behavioral scripting allows to describe behavioral scripts from which to create more complex sets of actions and behaviours. Scripts may be used to trigger other scripts and/or actions, allowing virtual world designers to easily construct hierarchies of behaviour.

It has remained difficult, however to instruct these autonomous characters. Cognitive models go beyond behavioral models, in that they govern what a character knows, how that knowledge is acquired, and how it can be used to plan actions. Cognitive models are applicable to instructing the new breed of highly autonomous, quasi-intelligent characters that are beginning to find use in interactive computer games [8].

Funge et al.[9] argue that a general model of any system requires a reasoning level above the behavioural or reactive level. This is built up by the game designer who gives knowledge to the agents by defining a set of axioms that constitute the way in which agents interact with each other and the environment. Characters possess knowledge concerning their goals and how they may be achieved. The system's behaviour becomes emergent – it arises out of the interaction of the agent's behavioural rules. Such a model is an explicit statement of how a player expects the world to behave.

Funge [8] decomposes cognitive modeling into two related sub-tasks: domain knowledge specification and character instruction. Domain knowledge specification involves administering knowledge to the character about its world and how that world can change. Character instruction involves telling the character to try to behave in a certain way within its world in order to achieve specific goals. There are two broad possibilities to instruct a character on how to behave: predefined behaviour and goal-directed behaviour. The distinction between the two is based on whether the character can non-deterministically select actions or not. This involves remembering the other choices, even after making a selection.

We use the term reactive behaviour, when a character's behaviour is based solely on its perception of the current situation. This means that the character has no memory of previous situations it has encountered. On the contrary, the first step in goal-directed behaviour is to come up with a way to define a cognitive character's goals. Funge [10] developed an intuitive high level interaction language CML (Cognitive Modeling Language) whose syntax employs descriptive keywords, but which has a clear and precise mapping to the underlying formalism. In this paper, we suggest a different approach based on protocol analysis to define the context in goal-directed behaviour.

In computer games, the game characters are not necessarily expected to be as complex as actual humans. However, they are expected to behave like humans at certain tasks. Therefore, we believe that knowledge acquisition strategies used in expert system development can be applied to development of game characters in computer games. We suggest using the content-oriented concurrent protocol analysis method to investigate the context in complex cognitive actions of humans. We believe that this method is applicable to creating human-like characters in computer games, assuming game-play as a complex problem solving activity.

We view cognitive processing as information processing in the context of the problem-solving theory of Newell and Simon [11]. The general theory of human-problem solving specifies a short-term memory (STM) with a fixed, small capacity of chunks or symbols, and a long term memory (LTM) of large or infinite capacity. Kavakli and Gero [6] [7] have shown the evidence on this assumption by structuring concurrent cognitive processes of human designers.

VI. CODING COMPLEX COGNITIVE ACTIONS

We have developed a coding scheme to analyze play behaviour. This coding scheme is adapted from the coding scheme developed by Suwa and Tversky [12] who classified the contents of what humans see, attend to, and think of in complex creative activities based on visual recognition, into four information categories: representations and their perceptual features, spatial relations, functional thoughts, and knowledge. Suwa and Tversky applied this coding scheme to verbal protocols of the designers to gather clues for visual recognition process. We will apply the coding scheme to verbal protocols of game players. The steps of the cognitive modeling process can be categorized as follow:

- Data collection by videotaping
- Transcription of verbal protocols
- Segmentation
- Coding

We collect verbal protocols as a concurrent report during the gameplay, videotaping players in a game session. These protocols are divided into segments, indexed and coded according to the information categories. In the coding scheme, different modes of player's actions are coded for each segment. There are four modes of cognitive actions in our version of the coding scheme as there are in the coding scheme developed by Suwa et al. [13]: physical, perceptual, functional, and conceptual.

In this hypothetical game, assume that the task of the human player is to find where the treasure is burried in competition with an AR agent. In the beginning of the game, human and artificial players are given a map including some symbolic key elements and keywords that serve as clues for problem solving. These key elements are a ford, mountain, cave, forest, river, bridge, etc. In this case, the range of cognitive actions varies according to the cognitive tasks. Tables 1, 2, 3 and 4 show the possible subcategories and codes of the cognitive actions we need to analyze in this hypothetical adventure game.

Table I. Codes of physical actions

Phy-actions: Physical Actions	
Dcl: dig a certain location	Bs: Block sword Ts: Thrust sword
Gs: get a spade	P: Point to a key element
Rm: Read the map including elements	R: Run
Wf: Walk forward Wb: Walk backward	Ju: Jump up Jd: Jump down Jf: Jump forward
Tl: Turn left Tr: Turn right	Fwp:Fight with people Fwa: Fight with animals
C: Climb	Moa: Move over an area
Sp: Speak	Mh: Move hand
L: Look at a feature, object, or a scene	St: Stand

Table II. Codes of P-actions

P-actions:perceptual actions		
Psg: discover a space as a ground	Pp: discover a way to the treasure	Prn: attend to a new relation
Posg: discover an old space as a ground	Prp: discover a spatial or organizational relation	Pfm: attend to the feature on the map
Por: mention or revisit a relation	Pw: discover a weakness of an enemy	Pfn: attend to the feature of an enemy
	Pt: discover a tool	Ps: attend to a scene

Table III. Codes of F-actions

F-actions:Functional actions		
Fn: associate a feature with a new function	Fo: continuing or revisited thought of a function	Frei: reinterpretation of a function

Table IV. Codes of G-actions

G-actions: Goals	*Subcategories of G1 type goals:*
G1: to cross a barrier	G4: to find the treasure
G2: to overcome an advisory	G5: goals directed by the use of explicit knowledge or past cases (strategies)
G3: to get help from an AR agent	

Our purpose is to define the context of complex behaviour in gameplay and simulate this process in the implementation of autonomously reactive agents in computer games. Table 5 shows a sample for the context behind a verbal representation, indicating the relations between different codes of cognitive actions. In this sample, the segment numbered 248 in the verbal protocol includes three sentences associated with the same concept : “I have to cross the river. I see crocodiles. I am going to fight with them.” In this case, the symbols visually represented on the screen are Sym1 (river) and Sym2 (crocodiles). The player notices both symbols by looking at them (L1 and L2). The physical actions that the avatar should execute in this case are coded as Jd (jump down), Ts (thrust sword), and Fwa (fight with animals). These physical actions all depend on a group of perceptual actions. Perception of crocodiles (Pfn) is associated with looking at the symbol crocodiles. The player perceives a spatial relation (Prn1) between the crocodiles and the river by looking at these symbols. The player also discovers (Psg) an area on the other side of the river. The area at which the player stands (Posg) has already been discovered before. These discoveries lead the player to perceive another relation (Prn2) between the river and its two sides. The function of the Sym2 (crocodiles) is associated with the enemy and recorded as a functional action (Fn). In this case, the player has two goals: to cross the river and to fight with the crocodiles. The first goal triggers a discovery (Prn2) and the second one triggers a functional action (Fn). Thus, the context behind these three sentences can be defined by this coding scheme.

By using statistical methods we can investigate the correlations between cognitive actions and capture parallel processing actions in complex behaviour in gameplay. By structuring these concurrent cognitive actions, it is possible to find the chunks in short-term memory, and to produce information processing charts and matrices. Following this, matrix analytic methods can be used to find out if there are certain paths followed in this data set. Thus, it is possible to develop a model based on the paths and processes found in these matrices.

Table V. A coded segment

Transcription Segment no: 248	I have to cross the river. I see crocodiles. I am going to fight with them.				
Action type	index	object	Description (where, of what, among what?)	Dependency	
				index	On what
Symbol					
Sym1	new	River			
Sym2	new	Crocodiles			
Look at					
L1	new	Sym1			
L2	new	Sym2			
Move					
Jd				new	Prn1
Ts				new	Pfn
Fwa				new	Ts
Perceptual					
Pfn	new		Crocodiles	new	L2
Prn1	new	l-relation	spatial rel: river & crocodiles		L1 & L2
Psg	new		other side of the river		
Posg	old		the side that the player stands at		
Prn2	new	l-relation	spatial rel: two sides divided by the river	new/old	Psg & Posg
Functional					
Fn	new		Crocodiles: enemy		Pfn
Goals					
type	content			Source Seg/type	Trigger what?
G1	To cross the river				Prn2
G2	To fight with crocodiles				Fn

VII. CONCLUSION

In this paper, we presented a cognitive modelling approach to capturing the context of complex behaviour in gameplay. We explored the context of cognition as a complex system, summarised different approaches to implementing behaviour, defined the characteristics of reactive and goal-directed agents, and demonstrated a coding scheme to analyse the context of complex cognitive actions in gameplay. The coding scheme adopts characteristics of semiotics and natural language processing, but is applied to visual cognition in computer games.

The method we suggest may be applicable for the implementation of a visual and verbal recognition system to be used in multi-agent communication in computer games. The next step is to test this coding scheme in a series of experimental studies.

In the protocol analysis technique we outlined in this paper, goals either trigger or are triggered by other cognitive actions expressed by the verbal representation of the player - which in this case, the speech. We believe that this technique may be used for modelling cognitive processes of goal-directed agents. To design computer games and to implement human-like characters in them, we need to know more about cognitive processes of human beings and investigate not only how humans behave, but also the nature of the game-play itself.

REFERENCES

[1] Durlach, N.I., Mavor, A.S. (eds) 1995: Virtual Reality: Scientific and Technological Challenges, National Research Council, National academy Press, Washington DC.

[2] Brutzman, D, 1998: Graphics Internetworking: Bottlenecks and Breakthroughs, In: Dodsworth, C., Digital Illusion, Entertaining the future with high technology, Addison-Wesley, Sydney, pp. 61-97

[3] Rappaport, A.T., 1997: Context, Cognition, and the Future of Intelligent Infostructures, In: Feltovich, Ford, Hoffman, Expertise in Context, AAAI Press, Menlo Park California, pp. 523-539

[4] Watt, A., Policarpo, F., 2001: 3D Games, Real-time Rendering and Software Technology, Addison-Wesley, Sydney

[5] Goldberg, A., 1998: Avatars and Agents, or Life Among the Indigenous Peoples of Cyberspace, In: Dodsworth, C., Digital Illusion, Entertaining the future with high technology, Addison-Wesley, Sydney, pp. 161-180

[6] Kavakli, M., Gero, J.S., 2002: The structure of concurrent cognitive actions: A case study on novice and expert designers, *Design Studies,* Vol 23/1, 25-40

[7] Kavakli, M., Gero, J.S., 2001: Sketching as mental imagery processing, *Design Studies*, Vol 22/4, 347-364

[8] Funge, J.: 1999, AI for Games and Animation: A Cognitive Modelling Approach, Gamasutra, http://www.gamasutra.com/features/19991206/funge_pfv.htm.

[9] Funge, J., Tu, X., Terzopoulos, D., 1999: Cognitive Modelling: Knowledge Reasoning and planning for intelligent characters, Proceedings of SIGGRAPH'99, pp. 29-38

[10] Funge, J.: 1999, AI for Games and Animation: A Cognitive Modelling Approach, A.K.Peters. Natick, MA.

[11] Newell, A., and H. A. Simon. 1972: Human problem solving. Englewood Cliffs, NJ: Prentice Hall.

[12] Suwa, M. and Tversky, B.: 1996, What architects see in their design sketches: implications for design tools, *Human Factors in Computing Systems: CHI'96*, ACM, New York: 191-192.

[13] Suwa, M., Gero, J.S., and Purcell, T.: 1998a, Macroscopic analysis of design processes based on a scheme for coding designers' cognitive actions, *Design Studies* **19**(4): 455-483.