Chapter 25
Virtual Soar-Agent Implementations:
Examples, Issues, and Speculations

Jeremy Owen Turner
Simon Fraser University, Canada

ABSTRACT

This chapter provides a brief overview of those virtual agent implementations directly inspired by the cognitive architecture: Soar. This chapter will take a qualitative approach to discussing examples of virtual Soar-agents. Finally, this chapter will speculate on the future of Soar virtual characters. The goals of this chapter are sixfold. The first goal is to explain why cognitive architectures are becoming increasingly important to virtual agent design(s). The second goal is to convey why this chapter focuses exclusively on virtual agents that utilize the Soar architecture. The third goal is to explore some of Soar's technical details. The fourth goal is to showcase a few diverse examples where Soar is beginning to have a design impact on virtual agents. The fifth goal addresses Soar's limitations – when applied to agent design in virtual environments. The final goal speculates on ways Soar can be expanded for virtual agent design(s) in the future.

INTRODUCTION

Cognitive architectures have been around for decades.[1] Many of the more canonical architectures such as CLARION (CLARION webpage, n.d.) and Soar (Soar webpage, n.d.) have been continually updated over the years for agent implementation by external developers, academics and hobbyists. However, most virtual agents (e.g. bots, Non-player-characters [NPCs]) created up to the present day, still do not directly utilize cognitive architectures for interaction with the agent's embodied environment. Instead, such agents often depend on cognition-less systems that may include the use of: finite-state machines, subsumption-level (i.e. reactive) algorithms (Brooks, 1991), hierarchical search interaction trees, heuristic search, and brute-force techniques, or some combination thereof. In recent years, computing power has exponentially increased and this has allowed for a serious reconsideration of cognitive architecture

DOI: 10.4018/978-1-7998-0951-7.ch025

implementations. This new technological climate is beginning to encourage tractable innovations in artificial general intelligence (AGI) generally, and experimentation with cognitive architectures in particular. Academics and developers are increasingly interfacing cognitive architectures with virtual agents for more deliberation and cognitive potential in their interaction design.

This chapter provides an opportunity to qualitatively discuss and tally some of the state-of-the-art of those virtual agents that employ a particular canonical cognitive architecture, Soar (State Operator And Result)[2] (invented by Laird, Newell & Rosenbloom, ca. 1983). There are numerous cognitive architectures existing today (e.g. ACT-R, CLARION, RASCALS etc.) but due to size and other limitations explained below, Soar will be the focus of this chapter.

The thematic goals of this chapter are sixfold. The first goal is to explain what a cognitive architecture is and why they are becoming increasingly important to virtual agent design(s). The second goal is to convey why this particular chapter focuses exclusively on virtual agents that utilize the Soar architecture. The third goal is to explore some technical details that are uniquely characteristic of Soar's handling of cognition. The fourth goal is to showcase - via a few diverse examples - the continued legacy and design impact Soar is beginning to have on virtual agents. The fifth goal is to address Soar's limitations – when applied to NPC/bot design in virtual environments. The final goal is to speculate on additional ways Soar can be expanded for virtual agent design in the near future.

CAVEAT: VIRTUAL AGENTS MIGHT NOT EVEN REQUIRE COGNITIVE ARCHITECTURES

Design motivations behind whether or not to deploy a cognitive architecture often varies according to whether the virtual agent is meant to academically evaluate embodied and/or extended cognition or instead, to implement a virtual character (NPC) for social and/or ludic entertainment purposes. With these contextual contingencies in mind, most contemporary virtual agents are not required to be cognizant of the visual inputs beyond the raw numerical data underlying each object and path. Basically, a virtual agent can successfully interact and navigate its environment without being directly aware of it in any introspective way. An agent can react to numbers alone. For example, a bit-mapped image or any other virtual object can be represented simply as a stream of numeric representations (e.g. binary digits). These numbers provide the agent with coordinates, proximity-relations, and other manually coded identification data that allow the agent to respond in a completely reactive way. In many video-games, for example, an NPC can appear to be intelligent simply by mindlessly reacting to the changes in numerical inputs afforded by the dynamic changes occurring via game-play interaction. However, cognitive architectural enhancement might be useful for any agent-based cognitive task involving reasoning and mnemonic assistance. Otherwise, a cognitive architecture may only be useful in this numerically-reduced case if we wish to imbue the agent with the ability to contemplate its environment as an "[...] idealized version of the underlying reality" (Best et al., 2006, p. 186). This underlying numerical reality might be modeled to enable embodied, extended, situated, and distributed (EESD) cognition so that virtual agents can actively "[...] structure or manipulate their environments" to "simplify, sequence, or otherwise, support cognitive processes" (Smart & Sycara, 2015, p. 3837).

Related Content