Available online at www.sciencedirect.com

ScienceDirect

Cognitive Systems Research 50 (2018) 1–9

Cognitive Systems RESEARCH

www.elsevier.com/locate/cogsys

The extended ramp model: A biomimetic model of behaviour arbitration for lightweight cognitive architectures

Swen E. Gaudl [a,*], Joanna J. Bryson [b]

[a] *Falmouth University, Metamakers Institute, Penryn Campus, Treliever Road, Penryn TR10 9FE, UK*
[b] *Department of Computer Science, University of Bath, Claverton Down, Bath BA2 7AY, UK*

Received 31 March 2017; received in revised form 19 October 2017; accepted 22 February 2018
Available online 16 March 2018

Abstract

In this article, we present an idea for a more intuitive, low-cost, adjustable mechanism for behaviour control and management. One focus of current development in virtual agents, robotics and digital games is on increasingly complex and realistic systems that more accurately simulate intelligence found in nature. This development introduces a multitude of control parameters creating high computational costs. The resulting complexity limits the applicability of AI systems. One solution to this problem it to focus on smaller, more manageable, and flexible systems which can be simultaneously created, instantiated, and controlled. Here we introduce a biologically inspired systems-engineering approach for enriching behaviour arbitration with a low computational overhead. We focus on an easy way to control the maintenance, inhibition and alternation of high-level behaviours (goals) in cases where static priorities are undesirable. The models we consider here are biomimetic, based on neuro-cognitive research findings from dopaminic cells responsible for controlling goal switching and maintenance in the mammalian brain. The most promising model we find is applicable to selection problems with multiple conflicting goals. It utilizes a ramp function to control the execution and inhibition of behaviours more accurately than previous mechanisms, allowing an additional layer of control on existing behaviour prioritization systems.

Keywords: Reactive planning; Action selection; BBAI; POSH; BOD; Bio-inspiration; Modelling; Behaviour design; Curiosity

2010 MSC: 00-01; 99-00

1. Introduction: Behavior arbitration and lightweight cognitive architectures

The mechanism described in this article addresses the issue of responsive and flexible action selection for behaviour-based AI (Brooks, 1986; Bryson, 2000) or similar modular approaches to cognitive systems, where the system contains a set of potentially conflicting rival goals. We focus on intelligence for limited-performance systems, e.g. subsystems in a larger system which cannot simultaneously exploit the same resource. These systems face resource constraints such that they are not able to or not intended to use a fully-fledged cognitive architecture such as SOAR (Laird, Newell, & Rosenbloom, 1987) or ACT-R (Anderson et al., 2004). Examples of resource restrictions include limited CPU cycles, low power consumption or restricted memory size. For both clarity on the type of problem we face and inspiration for its solution we look to nature. Arbitrating conflicting goals is an essential skill in animal behaviour as it heavily influences the fitness of an entity.

---

[*] Corresponding author.
E-mail addresses: swen.gaudl@gmail.com (S.E. Gaudl), jjb@bath.ac.uk (J.J. Bryson).
URL: http://www.swen.fairrats.eu (S.E. Gaudl).

https://doi.org/10.1016/j.cogsys.2018.02.001
1389-0417/

Take for example an antelope at a waterhole. The antelope lives in a hostile environment and water is a valuable resource because it is both needed and sparse. Inside the waterhole there is a group of crocodiles waiting for an animal to get close enough to be eaten. As the antelope needs to drink it has now two highly prominent and conflicting goals—survival by avoiding the predator and survival by drinking from the waterhole. If the antelope is not able to solve that situation by selecting one it would simply die in front of the waterhole.

Consider also the situation when the antelope is drinking and a predator emerges close to it. It is already pursuing an important goal—drinking to sustain living—yet it needs to make a decision as quickly as possible to escape the predator without hesitating or reverting back to the drinking behaviour, at least in the near term. Resource constraints in technology can be different, but still create conundrums.

Looking more closely at digital games, we find similar problems of control in a very different context. It is quite common to allow the AI only to occupy a small number of cycles per frame as most of the resources are needed for visual representations. It is arguable whether this is a correct choice, but it can be expected to be a given fact in most commercial products. Including a full-fledged cognitive architecture to control multiple cognitive agents in such an environment is in most cases not desirable as the cognitive architecture requires both more runtime resources and more work to design. The commercial game “Assassins Creed Unity” for example, where the player explores Paris during the French Revolution, features hundreds of agents moving around the city. In addition, designing the cognitive agents themselves is more time consuming than writing scripted agents, and extra time spent on the design the player might never encounter is not seen as worthwhile for the game developer. Game agents have in most cases the specific task to follow a specific designed role they are assigned to by the author. This creates a crafted experience similar to an actor in a theatre play. The agent is not to act entirely freely. However, large hand-authored behaviour-based systems or expert systems tend to be difficult to maintain as there are lots of transitions between behaviours. In addition, once a game is conceived there is often considerable economic motivation to bring it to market quickly. Consequently, the main interest of game AI designers and engineers is to have flexible, modular tools for creating template agents, and then to modify those to create the desired character outcomes.

In robotic applications the need for easily-modifiable, light-weight but robust behaviour is driven not only by military research or humanoid robotics but also by commercial applications such as crop analysis and land scouting done by unmanned aerial systems (UAS). Again, in these applications, a large fully cognitive system is rarely needed but a lightweight, flexible and modular architecture to control basic functionality is essential. The AI for such systems is only responsible for a limited but important subspace of the problem at hand, e.g. to secure the safety of the robot and its surrounding operating environment. These underlying basic functionalities need to be fast and reliable requiring only minimal input. Brooks (1991) presents a solution to that problem with the SUBSUMPTION architecture (Brooks, 1991) which allows the development of reactive agents utilising their embodiment. The benefit of reactive agents becomes visible when they utilise their embodiment, it can act as a form of memory or state, this provided a breakthrough in managing constraint resources (Nolfi, 2002). In nature, a solution to faster action sequences can be found when habitual behaviours are employed which are learnt, fast, robust and require nearly no cognition.

The work presented here is motivated by an analysis of existing agent architectures and agent modelling environments for autonomous agents in digital games (Grow, Gaudl, Gomes, Mateas, & Wardrip-Fruin, 2014) and robotics and trying to find ways of aiding existing systems in their arbitration process. Existing cognitive approaches such as SOAR, ACT-R, LIDA (D’Mello, Franklin, Ramamurthy, & Baars, 2006) and CRAM (Beetz, Mösenlechner, Tenorth, & Rühr, 2012) are extremely powerful, allowing the creation of sophisticated agents. However, due to the high complexity and steep learning curve they are seldom used outside of academic demonstrations and simplified problem spaces. Even where they are used, they are used primarily in communities strongly linked to an academic environment, such as military war games. When full cognitive reasoners or large expert systems are not needed or applicable, lightweight architectures and models such as BEHAVIOR-ORIENTED DESIGN (BOD) (Bryson, 2001), Agile Behaviour Design (Gaudl, 2016) and BEHAVIOR TREE (BT) (Champandard, 2017) or purpose specific architectures such as Pogamut (Gemrot et al., 2009), A Behavior Language (ABL) (Mateas & Stern, 2002), and FAtiMA Modular (Dias, Mascarenhas, & Paiva, 2014) can be used. Purpose-specific architectures offer an optimized workflow for specific settings, reducing development time. Other academic systems such as the MOASIC model (Haruno, Wolpert, & Kawato, 2006) offer a way to learn the required connections between actions and sensors. Later work such as (Oudeyer, 2004) uses intrinsic motivation to learn behaviours by altering the environment and exploring it (Oudeyer, 2004). Both of the latter systems require relatively long training sessions and are also less intuitive for later hand-tuning. However, they offer an automated way to achieve good behaviour arbitration. In games, editorial control of agents is important. The model that is presented in this work can also be adjusted using automated approaches such as gradient descent but the main focus is on providing an augmentation for existing systems and that initially requires no tuning. Lightweight systems due to their lower additional computational cost and lower learning curve are generally more favoured in non-academic application. These systems have to date been used most widely in the computer games industry, a sub-