Reuse and arbitration in diverse *Societies of Mind*

Ciarán O'Leary

School of Computing, Dublin Institute of Technology
Kevin St., Dublin 8, Ireland
`www.comp.dit.ie/coleary`

Abstract. One popular multi-disciplinary view on the architecture of the minds of natural creatures is of a mind sub-divided and organized into a set of specialised and semi-autonomous agents, each with insufficient intelligence to drive the whole creature alone, but with an ability to work with other agents in what has generally become known as a *society of mind*. Though such models are popular in AI, building truly diverse minds is difficult since it must require the integration of the work of multiple independent authors. Since reuse is not a key concern for AI developers, at least to the degree that it is considered important by software engineers, non-intrusive mechanisms are required to reuse software and integrate agents into societies. One such attempt, using a common interface and simple arbitration is described here.

1 Introduction

The term *Society of Mind*, originally coined by Minsky [18], has come to refer to any type of modular intelligence where multiple, specialised, semi-independent, individually mindless agents contribute to a single, whole intelligence which emerges through their individual expression and collective competition and cooperation.

Agents in such societies employ heterogeneous algorithms, specialised for the goal they are addressing or the task they are designed to carry out. Within the society however, all agents generally present a standard interface to each-other which masks the differences between the algorithms neatly encapsulated inside the agent.

The popularity of this model in artificial intelligence research is due to both the psychological plausibility [7, 10, 23] and the demands placed on the engineer of complex artificial intelligences. From an engineering perspective, it is widely accepted that the modular approach to systems development facilitates incremental development, software reuse, increased understanding of components and improved maintenance [17]. Given the strengths of the modular, or object oriented approach to software engineering, many advances have been made in the development of object oriented patterns, standards, platforms and languages [8, 9, 25].

Outside of the field of multi-agent systems (MAS) [31], particularly Internet agents, the AI community has paid little attention to the standardisation efforts taking place in software engineering. Specifically, those areas of AI closely aligned with the Society of Mind (SOM) approach have much to gain from standardisation of agent interfaces, although little work has been done in the area.

The most important benefit of standardisation of SOM agents is the ability to take an agent from one society and easily integrate it into another society, effectively *reuse* the agent (and consequently its algorithm) elsewhere. This simplifies the task of the AI researcher who wants to develop systems with truly diverse components, as the details of the implementation are hidden behind an easily understood, standard public interface.

This paper describes a first attempt at developing such a standard interface. We review a number of different approaches to developing agents for an SOM and show the key commonalities that can be easily captured in a standard interface. In brief, all agents in an SOM can provide the following information:

1. An expression of whether or not it is interested in the current state of the world i.e. does it want control of the body.
2. An action, or course of action to pursue, or to avoid.
3. An estimate of the duration of the course of action.
4. An estimate of how urgently it requires control.

In section 2 we describe a standard *Society of Mind*. There are many differences between the numerous implementations described in the published literature over the past decades, so our description is at a relatively high level. The remaining sections discuss the World-Wide-Mind project, and various modular approaches to AI, before describing our attempts to standardise an interface that can be used by diverse agents in an SOM.

2 A *Society of Mind*

The term *Society of Mind* was first introduced in Marvin Minsky's famous book of that title in the mid-1980s [18]. Minsky proposed that the mind is far too complex to be described by a single, clean set of rules, rather the mind must be composed of a large, interconnected set of mindless, focused, diverse specialists, too useless on their own to take command of the creature they were hosted by, but valuable citizens nonetheless in a society of agents. Two of the main points in the proposal, modularity and diversity, had received earlier treatment from Minsky and had served as an alternative view on the ongoing debate between the symbolic and connectionist approaches to AI [19].

In the Society of Mind, and in subsequent work [20], Minsky defined a variety of types of agents, each involved in the management, selection, admission and censorship of other agents who are selfishly trying to express themselves and exploit others. The theory was relatively high level and served as a foundation for much work in AI, with many authors describing their architectures as *Societies of Mind* [26], but Minsky's proposal never turned into an actual implementation. Work directly influenced by this theory is, however, currently being undertaken under Minsky's own supervision, where the focus is on cognitive diversity, or supporting multiple "ways to think" [27].

2.1 Characteristics

Some of the most important points of the Society of Mind theory are:

1. **Diversity**. This is the single most crucial dimension to the theory. No complex, adaptive mind could be limited to a single approach. Animals display a variety of behaviours, both innate and learned, use different types of memory for different problems, and learn different solutions in different ways. No single set of rules, akin to Newton's Laws in physics, could capture this diversity.
2. **Specialists**. Individual agents in the Society of Mind are specialists, capable of a single small contribution to the society. Some agents specialise in managing other agents by turning them on, or suppressing or censoring their output.
3. **Communication**. No single communication language could prevail throughout the society. Individual agents are not sophisticated enough to be able to speak the same language as all other agents in order to communicate, rather agents exploit or use each other by becoming activated at the same time.
4. **Lack of Centralised Control**: Variously referred to as the *homunculus* problem, or the Cartesian Theatre [10], the Society of Mind rejects the notion of a single centralised agent who is responsible for the management of the entire society.
5. **Redundancy**: Given the diversity of the agents in the society as well as the lack of centralized control, there are a variety of ways to think about, or approach any problem, so the society can continue to function in the absence of any of its agents.

3 SOM Implementations

In this section we describe some AI implementations that align closely with Society of Mind theory, in particular the five key points identified in section 2.

3.1 The Subsumption Architecture

Brooks *subsumption architecture* (SA) [3] represented a new departure for AI. Occurring around the same time as the publication of the Society of Mind, its introduction challenged much established practise in robotics and more generally in AI, as the first implementation of a behaviour based system. Much attention in behaviour based robotics, and more generally in behaviour based AI focussed on the removal of knowledge representation and reasoning [4] as core components of an intelligent system, but equally important was the idea of dividing an intelligent system into individual components that assumed responsibility for taking sensory input directly from the environment, and producing behaviour by directly influencing the actuators of the system. The components of the system, termed behaviours, were organised into layers where it was assumed that behaviours in the same layer would not conflict with each other. Behaviours in higher layers could over-ride the inputs to, or outputs from, behaviours at lower layers. In this way conflict was averted since higher layer behaviours always took precedence.

3.2 Behaviour Based AI

The past two decades have seen considerable advances in the behaviour based approach to AI [1, 16]. Various implementations have violated the early restrictions on knowledge representation by incorporating state into the behaviours [6]. Of particular interest however, are the wide variety of conflict resolution strategies that have been tried and tested [24]. One of the first deviations from the SA was Steel's architecture [28], where the outputs from the behaviours were summed to produce the cumulative output of the whole system. In his system all output was realised, and the complex behaviour of the robot emerged.

Maes' network of competencies [14] marries together a connectionist approach to intelligent systems with symbolic representation. Competencies represented activation controlled agents which triggered when a set of preconditions were met. Activation spread between competencies along links established according to conditions and lists associated with each competency. The action of the system was determined according to the activation levels, so like Brooks' and Steels' architecture, there was no centralised action selection or arbitration.

Tyrrell's [30] extension of roboticists Rosenblatt and Payton's architecture provided action selection by passing activation within a hierarchy of nodes, where internal nodes represented neither primitive actions nor whole agents. Leaf nodes represented primitive actions, where the action with the highest level of activation was chosen by the system.

Bryson's *Behaviour Oriented Design* (BOD) [6] is a methodology based on Object Oriented Design [9], with a centralised action selection mechanism. Behaviours are designed in an iterative fashion according to the requirements of the system, and encapsulate all the perceptive and action producing state and behaviour required. Action selection is centralised in a dedicated module, which identifies prioritised drive collections, competencies and action patterns from which it selects a behaviour to execute.

3.3 Modular Reinforcement Learners

Reinforcement learning (RL) [29] is a technique that allows an agent develop its own model of action in an environment by trying out actions according to a policy which is updated as the agent receives rewards and punishments. It removes the need for hand coding courses of action into the agent, but is limited by the memory requirements for large state-action spaces. A frequently used technique for addressing the problem of large state spaces is the division of a single agent into a society of agents, each member of which takes responsibility for learning sub-sections of the state space, or learning about individual goals or sub-goals. In behaviour based AI it has been used to both learn how to behave, and how to co-ordinate behaviour [15], often separately but sometimes together [11]. An advantage of combining learning within the individual agent with learning co-ordination in the society is that the reinforcement values learned internally by the individual agents can be propagated to higher levels where they can be considered as expressions of *how good* an agent considers an action to be.

Hierarchical RL [2] has become popular in recent years, where learning occurs simultaneously at multiple levels. In some cases the agent is permitted to follow a course of action without interruption, whereas in other cases decisions are made at higher levels, or collectively among the agents using their reinforcement values. This covers instances of both centralised and decentralised action selection.

3.4 Action Selection

Action selection is an SOM is the problem of choosing, at any point in time, the best behaviour to execute, or the best agent to listen to, or ultimately the best action for the body (animal, robot) to take. An agent in an SOM is any component that can proactively choose an action, or course of action, and suggest it, or try to execute it. In the extensive literature on the subject, these agents are variously referred to as behaviours, layers, systems, modules, competences, drives, beings, demons and homunculi among others. The key concern for us is the type of information that these agents make available to each-other in order for action selection to take place in the society as a whole.

Tyrrell [30] and Maes [14] both treat the action selection problem, by providing lists of the characteristics of effective action selection. A summarised list is given here:

1. Action selection must try to satisfy all the goals of the creature.
2. It must persist with a goal until completion, unless there is a much greater benefit for switching to another goal. This must be balanced with the requirement to be opportunistic and reactive to allow goals to be quickly spotted and satisfied where possible.
3. It must be highly adaptive to changing environments.
4. It must be able to choose *compromise candidates*, or actions that can satisfy more than one goal.

In general it is assumed that a single agent in an SOM would address a single goal, although this is not necessarily always the case. When constructing an SOM from diverse agents, it is important that each agent can provide sufficient information so that the action selection mechanism, or more generally, the arbitration scheme, can attempt to satisfy the criteria listed.

4 World-Wide-Mind

The work described in this paper is taking place as part of the World-Wide-Mind (WWM) project [12, 13], a project which attempts to use the World-Wide-Web as a mechanism for supporting the scaling up of AI research. Using WWM technology, authors of new algorithms, or agents, can make their software available online as a web service which can be interacted with remotely. Third parties can then build agents which reuse existing agents by incorporating them into societies and arbitrating between their action choices. Communication between agents takes place by exchanging simple XML over HTTP, meaning that the technical knowledge required of authors is reduced to a familiarity with some basic web technologies such as CGI or Java Servlets.

At its most basic level, WWM agents are queried by client software at each time-step. The purpose of the client is to take data from a single agent, and present it to a body which can use the data to execute an action or behaviour in a real or simulated world. Societies of agents are constructed by providing a high level agent that arbitrates between a set of pre-existing agents – effectively carrying out action selection akin to standard behaviour based architectures. Both the world and the agent are available online as web services, thus making them available to all web users. All the components in a society could be developed independently by multiple authors, thus facilitating the type of diversity that is rare in a single research group.

5 Interface

The interface that is presented by each of the minds, or agents, at the WWM entry level just requires each agent to provide an action at each time-step. This requires that the higher level arbitration does not have a great deal of information which it can use to perform action selection, thus requiring the author of the arbitration mechanism to either perform a detailed analysis of the performance of each agent separately [22], or contact the author of the individual agent to establish how it is making its decision. Without this, it is difficult to perform the type of action selection described in section 3.4.

The interface we describe in this section can be used by agent developers to provide more sophisticated information to arbitration mechanisms, which can in turn perform better action selection.

When queried at each time-step, an agent will return as much of the following information as possible:

For each action that it is interested in, either to take or avoid (if it is not interested in taking or avoiding any actions i.e. it does not trigger, it returns nothing), it returns an action tuple made up of <a, g, p, t, d, w> where

- a is the action proposed.
- g is the main goal currently being pursued which can take a null value.
- p is the priority (between 0 and 1). All actions proposed by an agent should be prioritised relative to each-other. If only one action is being proposed, or if all actions are of equal priority then no value is required.
- t is a boolean value which is used to capture whether or not the agent is seeking to take the action (set to true), or avoid the action (false).
- d is the distance to goal. If this takes a value of 0 then the proposed action should result in the agent achieving goal g. Otherwise the agent is indicating that there is no point selecting this action now, if its actions for the next d steps are not selected.

- w is the maximum waiting time. Using this value, an agent can state that it is willing to let another agent have a go once it can get control within w steps.

Using these tuples, agents can express the following:

1. Whether or not they are interested in the competition
2. A group of actions that they would be equally satisfied with.
3. The beginning of an action pattern, or sequence.
4. The achievement of a goal.
5. The urgency of its action choice.

6 Experiment

The problem world used for our experiment is a well known environment used in animat (animal/robot) research [32]. Tyrrell's SE [30] models a small animal in a heavily populated dynamic environment. As well as the animal, the world contains fruit, cereal food and water (any of which can be toxic), prey, two types of predator, other animals, cover where the animal can hide, shade where the animal can cool down, dangerous places where the animal can be injured or killed, landmarks the animal can use for navigation, a den where the animal can sleep, and creatures with which the animal can mate. The animal can choose from a set of 35 different actions (9 looking to improve perception, 16 moving, 4 eating or drinking, 2 courting or mating, cleaning, sleeping, resting and freezing to avoid detection by predators). The animal can only choose one action, but must over time satisfy each of its goals (cleaning, obtaining food, obtaining water, temperature regulation, predator avoidance, vigilance, hazard avoidance, irrelevant animal avoidance, sleeping at night, staying close to cover, not getting lost, reproduction). Failure to satisfy each of its goals to some degree will negatively affect the animal's health resulting in its death. The measure of the success in the SE is the number of times the animal successfully mates, but the animal must ensure it lives long enough to be presented with mating opportunities.

6.1 A Society of Mind for Tyrrell's Simulated Environment

Five agents (*hunter*, *lookout*, *maintenance*, *mater* and *navigator*) were implemented using simple motivational based algorithms, hidden behind the interface described above. Arbitration could have been conducted according to any number of algorithms, but to demonstrate how information from the action tuples can be employed in arbitration, we designed a priority based algorithm, described here.

Each of the agents is given a *level*. At every time step the actions of the highest level agent are considered first. If any action is suggested which has a waiting time of 0, it is selected. If more than one action fits this category then actions from the next level are used to break the tie. If no action at the top level has a waiting time of 0 then the second level is considered. Where numerous actions are of equal priority, action choices from the top level down are considered. If no action is selected after going through all agents then the process is repeated for a higher waiting time. This simple algorithm does not make use of all of the information available from the agents but it is able to make good choices with the limited information it uses.

```
agents = { lookout, mater, maintenance, hunter, navigator }
wait_max = 0
while true
  for each agent in agents
    for each action in agent
      if waiting <= wait_max
        if no other <= wait_max return action
        else examine other levels return action
    end for_each
  end for_each
  wait_max = wait_max + 1
end while
```

6.2 Results

Tyrrell implemented five different minds for the creature in his simulated environment. Results are shown here for the two algorithms that achieved the best results in four versions of his world. (ER&P is the Extended Rosenblatt and Payton algorithm, a monolithic algorithm based on the work of the two named roboticists. Drives is a simple ethologically inspired motivation based algorithm)

Version	Standard	Version 1	Version 2	Version 3
ER&P	8.09	3.61	8.16	13.38
Drives	6.44	3.29	6.41	8.78

These values are the average number of times the animal mated over 1,650 runs in the world. Tyrrell's code is freely available online, so each of his algorithms was re-tested along with the *five agent society of mind* giving the following results.

Version	Standard	Version 1	Version 2	Version 3
ER&P	7.74	3.55	7.88	13.03
Drives	7.11	3.55	7.14	8.95
SOM	6.86	3.54	6.35	8.3

Bryson [5] produced a modified version of Tyrrell's world where food was made more scarce – in which she tested her own POSH algorithm as well as Tyrrell's ER&P. Results for her algorithm and the society of mind in this world are below.

Version	Standard	Version 1	Version 2	Version 3
ER&P	4.77	2.46	4.56	12.53

POSH	8.17	3.56	10.79	10.74
SOM	4.18	2.47	4.6	6.98

The society of mind was able to perform as well as other minds in some worlds. The difficulty of reusing agents not specifically designed to work together remains but with well designed agents, arbitration can be improved once sufficient information is available.

7 Summary

The importance of diversity in intelligent systems is well understood. For certain AI problems, diversity is a key requirement so mechanisms must be provided that facilitate the smooth integration of diverse components. It is important when diverse components are being integrated, that a balance is reached between the need to provide a standard interface to each of the components, and the need to express information about how decisions were made internally in the component, or agent.

We have described how a standard interface is provided for developing societies of mind. We have built on early work on the World-Wide-Mind project by showing that a more sophisticated set of parameters at the interface to an agent can capture information that is useful for arbitration. We also provide a straightforward priority based arbitration mechanism that was used to select between agents and their action choices.

Designing for reuse and integrating reusable components can lead to novel and interesting combinations of agents in societies - resulting in novel and interesting behaviour in robots and other artificial intelligences. It also presents opportunities for diverse and distributed groups of people to collaborate on the AI problem. Given the scale of the problem being dealt with, perhaps reuse and integration is the only way to keep our eyes on the prize [21].

References

1. Arkin, R.: Behavior-Based Robotics. MIT Press, Cambridge, MA, (1998)
2. Barto A. and Mahadevan S.: Recent Advances in Hierarchical Reinforcement Learning, Discrete Event Dynamic Systems: Theory and Applications, 13, 41-77 (2003)
3. Brooks, R.: Robust layered control system for a mobile robot. IEEE Rob. & Auto. (1986)
4. Brooks, R.: Intelligence without Representation, Artificial Intelligence 47, (1991)
5. Bryson, J.: The Study of Sequential and Hierarchical Organisation of Behaviour via Artificial Mechanisms of Action Selection, MPhil: University of Edinburgh (2000)
6. Bryson, J., Intelligence by Design: Principles of Modularity and Coordination for Engineering Complex Adaptive Agents, PhD Thesis, MIT, 2001
7. Buss, D.:, Evolutionary Psychology: New Science of the Mind, Allyn & Bacon (2003)
8. Capretz L.F.: A Brief History of the Object-Oriented Approach, ACM SIGSOFT Software Engineering Notes, 28(2), March 2003
9. Coad, P. and Yourdon, E.: Object Oriented Analysis (2nd Edition), Prentice Hall (1990)
10. Dennett, D.: Darwin's Dangerous Idea, Simon and Schuster, (1995)

11. Humphrys, M.: Action Selection methods using Reinforcement Learning, PhD Thesis, University of Cambridge (1997)
12. Humphrys, M.: The World-Wide-Mind: Draft Proposal, Dublin City University, School of Computing, Tech Report CA-0301 computing.dcu.ie/~humphrys/WWM/ (2001)
13. Humphrys, M. and O'Leary, C.: Constructing complex minds through multiple authors, 7th International Conference on the Simulation of Adaptive Behavior (SAB-02). (2002)
14. Maes, P.: How to do the right thing. A.I. Memo 1180, MIT, Cambridge, MA (1989)
15. Martinson, E.: Stoytchev, A. and Arkin, R.: Robot Behavioral Selection Using Q-learning, IEEE/RSJ Conference on Intelligent Robots and Systems (IROS), (2002)
16. Mataric, Maja.: Behavior-Based Robotics as a Tool for Synthesis of Artificial Behavior and Analysis of Natural Behavior, Trends in Cognitive Science, Mar 1998, 82-87 (1998)
17. Mili, H., Mili, A., Yacoub, S., Addy, E.: Reuse Based Software Engineering: Techniques, Organizations, and Measurement, Wiley (2002)
18. Minsky, M.: The Society of Mind, Simon and Schuster (1985)
19. Minsky, M.: Logical vs. Analogical or Symbolic vs. Connectionist or Neat vs. Scruffy, Artificial Intelligence at MIT., Expanding Frontiers, Patrick H. Winston (Ed.). (1990)
20. Minsky, M.: The Emotion Machine, online at web.media.mit.edu/~minsky/ (2005)
21. Nilsson, N.J.: Eye on the Prize, AI Magazine 16(2):9-17, Summer (1995)
22. O'Leary, C.: Humphrys, M. and Walshe, R., Constructing an animat mind using 505 agents from 234 different authors, 8th Int. Conf. Simulation of Adaptive Behavior (2004)
23. Pinker, S.: How the Mind Works, W. W. Norton & Company (1999)
24. Pirjanian, P.: Behavior Coordination Mechanisms - State of the Art, Ph.D., USC (1999)
25. Schmidt, D., Stal, M., Rohnert, H. and Buschmann, F.: Pattern-Oriented Software Architecture: Patterns for Concurrent and Networked Objects, Wiley and Sons (2000)
26. Singh, P.: Examining the Society of Mind. Computing and Informatics, (2003)
27. Singh, P. and Minsky, M.: An architecture for cognitive diversity. Visions of Mind, Darryl Davis (ed.), London: Idea Group Inc. (2005)
28. Steels, L.: A Case Study in the Behavior Oriented Design of Autonomous Agents, Proc. 3rd Int. Conf. on Simulation of Adaptive Behavior (SAB-94) (1994)
29. Sutton, R. and Barto, A.: Reinforcement Learning: An Introduction, MIT Press, (1998)
30. Tyrrell, T.: Computational Mechanisms for Action Selection, PhD Thesis, Edinburgh, Centre for Cognitive Science (1993)
31. Weiss, G.: Multiagent Systems: Modern Approach to Distributed AI. MIT Press (1999)
32. Wilson, S.: The animat path to AI, 1st Int. Conf. on Sim. of Adaptive Behavior (1990)