Chapter 1

Constrained Incrementalist Moral Decision Making for a Biologically Inspired Cognitive Architecture

Tamas Madl, Stan Franklin

Introduction

The field of machine ethics has emerged in response to the development of autonomous artificial agents with the ability to interact with human beings, or to produce changes in the environment which can affect humans (Allen, Varner, & Zinser, 2000). Such agents, whether physical (robots) or virtual (software agents) need a mechanism for moral decision making in order to ensure that their actions are always beneficial, that they 'do the morally right thing'.

There has been considerable debate on what doing the right thing means, and on how moral decision making should be implemented (Anderson & Anderson, 2011; Lin, Abney, & Bekey, 2011; Wallach & Allen, 2008) in order to create so-called Artificial Moral Agents (AMAs) (Allen et al., 2000). Apart from the problem that no consensus on ethics exists, it has also proven to be exceedingly difficult to computationally implement the often vague and under-constrained

ethical frameworks invented for humans. To the authors' knowledge, no current AMA implementation comes even close to passing a full Moral Turing Test [1].

However, robots are getting increasingly autonomous, and are becoming increasingly prevalent. According to the International Federation of Robotics, about three million service robots were sold in 2012, 20% more than in the year before (IFR, 2013) (the IFR defines a service robot as *'a robot that performs useful tasks for humans or equipment excluding industrial automation application'*). Examples for service robots available on the market include Care-O-Bot (Graf, Hans, & Schraft, 2004) and the REEM service robot (Tellez et al., 2008), which can aid elderly or handicapped people, with functions such as carrying requested objects to users, entertainment, or telepresence / tele-assistance via videoconferencing. Recent research in autonomous transport could lead to driverless vehicles available on the market within the next decade - Google's fleet of self-driving cars have already driven 800,000 km on public roads (Burns, 2013) (see (Ciupe & Maniu, 2014; Alonso, 2011) for further examples of service robots).

The increasing autonomy of such robots - their ability to perform intended tasks based on their current state and sensory input, without human intervention - makes it very difficult to anticipate and control the actions they perform in advance. At the same time, their actions are morally relevant if it is possible that humans could be made worse off by them. Thus, autonomous robots need some kind of *moral decision making mechanism* if they can affect humans or their

[1] Just like the original Turing test, in the Moral Turing Test proposed by (Allen et al., 2000), a 'blind' observer is asked to compare the behavior of a machine to humans. Passing it requires that the machine should not be judged less moral than the humans on average.

environment, in order to constrain them to actions beneficial to humans, and to prevent them from doing harm in unforeseen circumstances (Powers, 2011).

Despite the vast unsolved technical, ethical and social challenges associated with developing such a mechanism, short-term solutions are needed for systems that could cause harm. The emerging field of robot ethics is concerned with the ethical implications and consequences of robotic technology (Scheutz, 2013; Lin et al., 2011; Anderson & Anderson, 2011). The field includes the ethics of how humans act through or with robots, and the ethical relationships between humans and robots, as well as the ethics of how to design and program robots to act ethically (Asaro, 2006). This chapter is concerned with the latter focus of robot ethics, taking a biologically inspired cognitive modeling approach.

Instead of trying to directly address the implementation of a full ethical framework, which would be very difficult with current technologies even if a universally accepted framework existed, we propose a simplification of this problem, following the advice *'make things as simple as possible, but not simpler'* (commonly attributed to Einstein). We will outline a moral decision making mechanism that is

- constrained to the domain and functionalities for which the agent is designed (instead of the full range of human actions, responsibilities, or 'virtues')
- based on a biologically inspired cognitive architecture (LIDA), and making use of existing cognitive mechanisms (such as routine decision making procedures, and theory of mind)

- a combination of top-down (based on explicit rules) and bottom-up (based on implicit, heuristic strategies) processing
- adaptive incrementalist (instead of assuming full knowledge and understanding of an ethical system and an appropriate computational mechanism)

We will also introduce a way of testing a specific AMA, inspired by test-driven development, that we believe will facilitate the incremental development of a robust moral decision making mechanism, reduce the number of 'bugs' or unintended malfunctions, and simplify the comparison of different AMAs operating in the same domain.

In order to illustrate these ideas, we will use the example of a Care-O-Bot type robot (Graf et al., 2004), controlled by the LIDA (Learning Intelligent Distribution Agent) cognitive architecture (Franklin, Madl, DMello, & Snaider, 2013; Baars & Franklin, 2009). Care-O-Bot is equipped with a manipulator arm, adjustable walking supporters and a hand-held control panel (additionally, it has two cameras and a laser scanner). It has been demonstrated to perform fetch and carry tasks, but could in principle also provide mobility aid (support for standing up, guidance to a target), execute everyday jobs (setting a table, simple cleaning tasks, control electronic infrastructure), or facilitate communication (initiate calls to a physician or to family, supervise vital signs, and call emergency numbers if needed) (Graf et al., 2004).

A Simplified Moral Decision Making Mechanism

Constrained to specific domain and functionalities

The difficulty of designing a moral decision making mechanism increases with the size of the space of possible actions. The problem of implementing a full ethical framework which would account for the entire vast space of possible human action can be simplified by constraining an AMAs actions. This is possible on different levels. We will use Sloman's proposed three levels of cognitive processes, the reactive, deliberative, and metacognitive (Sloman, 1999), as well as an additional non-cognitive level.

- On the non-cognitive level, the agent can be mechanically limited in terms of power and mobility. This decreases the scope of possibly harmful actions and thus simplifies the required ethics implementation. For example, in their proposed design for an 'intrinsically safe personal robot', (Wyrobek, Berger, Van der Loos, & Salisbury, 2008) have significantly limited their robots maximum force output, range of motion, and speed in order to prevent it from physically causing harm, while still facilitating a wide range of functions.

- On the reactive level (which has stimulus-action mappings but no explicit representation and evaluation of alternative actions), actions can be constrained in advance by designers or at runtime by bottom-up mechanisms. Designers might restrict the parametrized action space that the AMA can select from, avoiding unnecessary actions and parametrizations. For example, on the lowest level, the action moving the Care-O-Bots

manipulator arm might not permit a full swing of the arm, restricting one action to a small movement. On the other hand, harmful actions can also be avoided on the lowest level during runtime by a bottom-up emotional mechanism, which would inhibit the selection of the action if there is a negative emotional response. Emotional responses can implement values and contribute to bottom-up moral decision making (see next subsection). These would have to be designed for the specific domain of application, requiring only a subset of a full affective model.

- On the deliberative level (which includes planning, scheduling, problem solving), a top-down, rule based process could constrain decisions during run-time. Rules could be stored in declarative memory, be re-called if they apply in the current situation or when value conflicts arise, and influence the action selection process. As for the emotional reactions, the rules would also have to be designed specifically for the domain of the agent (a much easier problem than capturing all rules of any ethical theory). For complex situations such as moral dilemmas, multiple scenarios can be simulated internally to select the one most conforming to all rules (see next subsection for a description of how this would work in the LIDA cognitive architecture).

- On the metacognitive level ('thinking about thinking', includes monitoring deliberative processes, allocating resources, regulating cognitive strategies), it would be in principle possible to implement ethical meta-rules such as Kant's categorical imperative, since metacognitive processes might verify the validity of rules by simulating and monitoring their application in

different scenarios. However, such approaches are intractable with current limitations on available processing power, and the detail of available cognitive models (see next subsection).

Using mechanisms of a cognitive architecture

A moral decision making mechanism based on the LIDA cognitive architecture would not have to be built from scratch. It could make use of some of LIDA's relevant cognitive mechanisms (all of which have been designed conceptually and some of which have implementations). Within the conceptual LIDA model, these include volitional decision making (Franklin, 2000; Franklin et al., 2013) and non-routine problem solving mechanisms (Negatu, Franklin, & McCauley, 2006) and a theory of mind (Friedlander & Franklin, 2008). Although the partially implemented architecture is currently only capable of controlling software agents, work is underway to embody LIDA on a Willow Garage PR2 robot by interfacing it to the Robot Operating System.

The LIDA cognitive architecture is based on prevalent cognitive science and neuroscience theories (e.g. Global Workspace Theory, situated cognition, perceptual symbol systems, ... (Franklin et al., 2013)), and is one of the few cognitive models which are neuroscientifically plausible and to provide a plausible account for functional consciousness [2] (Baars & Franklin, 2009; Baars, 2005),

[2]The LIDA model talks of functional consciousness as described in Global Workspace Theory (referring to information that is 'broadcast' in the Global Workspace and made available to cognitive processes such as action selection, as opposed to only locally available, non-conscious information). It makes no commitment to phenomenal (subjective) consciousness.

attention, feelings and emotions; and has been partially implemented (Franklin et al., 2013; Baars & Franklin, 2009; Franklin & Patterson Jr, 2006).

Cognition in LIDA functions by means of continual iteration of similar, flexible and potentially cascading - partially simultaneous - cognitive cycles. These cycles can be split into three phases, the understanding phase (concerned with recognizing features, objects, events, etc., and building an internal representation), the attending phase (deciding what part of the representation is most salient, and broadcasting it consciously), and the action selection phase (choosing an appropriate action in response).

The major modules of the LIDA model implementing various stages of these cycles are displayed in Figure 1.1 below. We will describe the processes these modules implement starting from the top left and traversing the diagram roughly clockwise.

1. *Perception.* The agent senses its environment continually. Sensory stimuli are received and stored in a sensory buffer in the Sensory Memory. Feature detectors sample the sensory buffers frequently, and activate nodes in the Perceptual Associative Memory (PAM) which represent percepts, emotions, concepts, categories, events, etc. (McCall, Franklin, Friedlander, & D'Mello, 2010). PAM nodes are based on perceptual symbols (Barsalou, 1999); their activations reflect recognition confidence as well as bottom-up salience. The most recent implementation of LIDA's perceptual recognition mechanism is inspired by predictive coding and perception as statistical inference (McCall & Franklin, 2013) (a simpler approach integrating SURF-based feature detection with LIDA also exists, see (Madl & Franklin, 2012)).

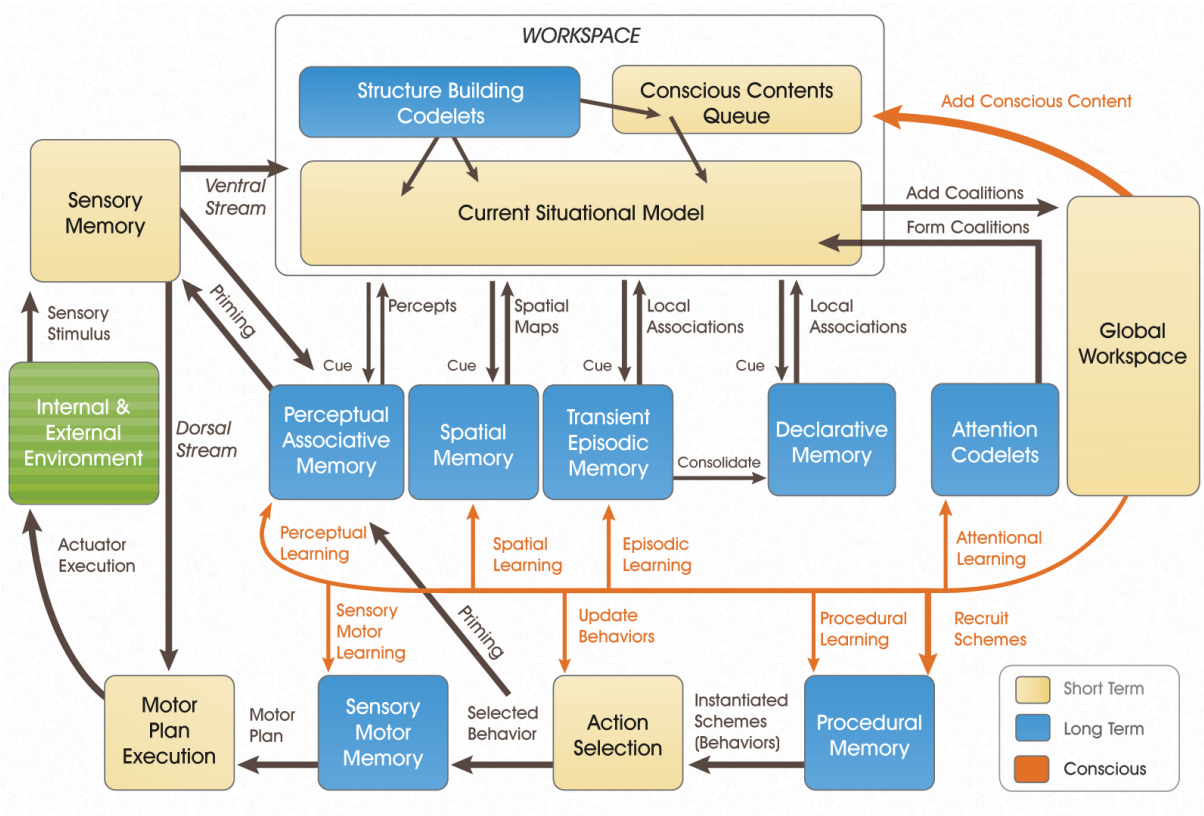

Figure 1.1: **LIDA's Cognitive Cycle**. From (Franklin et al., 2013)

2. *Percept to preconscious buffer.* Recognized percepts are stored in the preconscious buffers of LIDA's long-term working memory (Workspace), where a model of the agents current situation (current situation model) is assembled by structure building codelets [3]. The Workspace also contains salient or recent previous percepts that have not yet decayed away. Along

[3]In the computational LIDA model, the term codelet refers generally to any small, special purpose processor or running piece of software code. Codelets correspond to processors in Global Workspace Theory (Baars & Franklin, 2009)

with perceptual representations, the Workspace can also contain PAM nodes representing feelings or emotions, which can be activated either by low-level feature detectors, or by appraisal codelets reacting to the relevance, implications, and significance of the current situation, and the agent's coping potential (Franklin et al., 2013).

3. *Local associations.* Percepts and other Workspace contents serve to cue and retrieve local associations from the Transient Episodic (recording the what, where and when of unique personal experiences) and Declarative Memories (containing autobiographical long term episodic information as well as factual information separated from the place and time of their acquisition). These memory systems are extended instances of a content-associative Sparse Distributed Memory (SDM) (Snaider & Franklin, 2012; Kanerva, 1988). An additional Spatial Memory module is currently being developed (Madl, Franklin, Chen, & Trappl, 2013).

4. *Competition for consciousness.* Attention codelets look out for their sought content in working memory, create structures called coalitions with them, and compete to bring them to consciousness. The coalition containing the most salient (important, urgent, insistent, novel, threatening, promising, arousing, unexpected) perceptual structures wins the competition.

5. *Conscious broadcast.* The coalition of codelets winning the competition (typically an attention codelet and its content of PAM nodes, local associations and other structures) gains access to the Global Workspace (a fleeting memory enabling access between brain functions that are otherwise

separate) and has its content broadcast consciously (in the sense of the Global Workspace Theory). The contents of this conscious broadcast are available globally, but their main recipient is the Procedural Memory module, and their main purpose is to provide important information to facilitate action selection (as well as modulating learning).

6. *Recruitment of resources.* The most relevant behavioral schemes in Procedural Memory respond to the contents of the conscious broadcast. The implementation of these schemes is based on Dreschers schema mechanism (Drescher, 1991) and includes a model of constructivist learning (Franklin & Patterson Jr, 2006).

7. *Activation of schemes in the Procedural Memory.* Multiple applicable behavioral schemes are instantiated in the Action Selection module, and receive activation, based on the conscious contents.

8. *Action chosen.* The Action Selection module chooses a single scheme from the newly instantiated schemes and remaining previously active schemes. The action selection mechanism in LIDA is based on Maes' bottom-up behavior selection mechanism (Maes, 1989). If an action can be selected and executed in a single cognitive cycle, this could be called *consciously mediated action selection*, since the information upon which the action was selected was acquired consciously (it was moved to the Global Workspace, and broadcast globally), but the choice itself was made unconsciously.

9. *Action taken.* The execution of the action of a scheme results in external (e.g. the movement of a manipulator) or internal consequences (e.g. changing an internal representation).

Some decisions might require multiple cognitive cycles, and weighing the consequences of multiple possible actions. *Volitional Decision Making* is a higher-level cognitive process for action selection, and is performed consciously - unlike consciously mediated action selection, automatized action selection, or alarms (Franklin et al., 2013). In humans, consciously planning a novel route is an example of deliberative, volitional decision making.

LIDA's deliberative volitional decision making mechanism is based on Global Workspace Theory and James' ideomotor theory of volition (Baars & Franklin, 2009; Franklin, 2000). An idea or potential decision, represented as a structure of nodes in PAM (which can represent objects, actions, events etc. - see (McCall et al., 2010)), can reach the Global Workspace if selected by an attention codelet, and if judged relevant / important enough, be broadcast consciously and acted upon by recruiting a behavior scheme in Procedural Memory. Such schemes can initiate internal or external action.

Before the execution of an external action, multiple internal actions might be required to build internal structures upon which a final decision can be made, in multiple cognitive cycles. LIDA's Workspace includes a 'virtual window', in which temporary structures can be constructed with which to try out possible actions and their consequences without actually executing them. Multiple such structures can be selected by attention codelets, moved to the Global Workspace,

and compete with each other (here, attention codelets perform the role of James' 'proposers' and 'objectors') (Franklin, 2000; Franklin et al., 2013).

For a more detailed description of LIDA's modules and their functions in the cognitive cycle, see (Franklin et al., 2013; Baars & Franklin, 2009). We will introduce a concrete example of how LIDA's modules and processes might aid moral decision making in Section 3.

Combination of top-down and bottom-up processing

(Wallach, Allen, & Smit, 2008) describes 'top-down' approaches to mean both the engineering sense, i.e. the decomposition of a task into simpler sub-tasks, and the ethical sense, i.e. the derivation of consequences from an overarching ethical theory or system of rules.

In contrast, 'bottom-up' approaches can be specified atheoretically, and treat normative values as being implicit in the activity of agents (Wallach et al., 2008).

The LIDA model of cognition integrates both of these approaches (Wallach, Franklin, & Allen, 2010). 'Bottom-up' propensities are embodied in emotional/affective responses to actions and their outcomes in the LIDA model (Wallach et al., 2010). Feelings are represented in LIDA as nodes in PAM. Each feeling node constitutes its own identity. Each feeling node has its own valence, always positive or always negative, with varying degrees of arousal. The current activation of the node measures the momentary arousal of the valence, that is, how positive or how negative. The arousal of feelings can arise from feature detectors,

or it can be influenced by the appraisal that gave rise to the emotion [4], by spreading activation from other nodes representing an event (Franklin et al., 2013).

Thus, 'bottom-up' propensities can be engineered in a LIDA-based AMA by carefully specifying feature detectors and weights in PAM, in order to give rise to the right arousal levels of the right feeling nodes; as well as the specification of appropriate behaviours in procedural memory. For example, there could be a 'fall' feature detector watching out for quick, uncontrolled downward movement, and passing activation to a 'concern' feeling node. Another feature detector could recognize cries for help and also pass activation to the same node. Upon reaching a high enough activation, the 'concern' feeling node would be broadcast consciously and influence action selection, leading to the execution of the 'call emergency' behaviour. 'Bottom-up' influences on action selection can occur in a single cognitive cycle, as a result of consciously mediated action selection.

On the other hand, 'top-down' moral decision making can be implemented in LIDA by designing and storing an ethical rule system in the declarative memory. Such rules consist of PAM nodes, the common representation in the LIDA model; and specify internal or external actions in response to some perceptual features of a situation. LIDA's declarative memory is a content-associative Sparse Distributed Memory (SDM) (Snaider & Franklin, 2012; Kanerva, 1988). Moral rules are automatically recalled from declarative memory by either the current situation in working memory resembling the context of the rule, or alternatively by proposal/objector codelets (which implement volitional decision making and allow LIDA to compare options) (Wallach et al., 2008).

[4]The LIDA model speaks of emotions as feelings with cognitive content

More complex moral rules in which decisions are not directly based on perceptual representations, such as utilitarianism, would require additional implementation of the decision metric. In the case of utilitarianism, this would involve assembling representations of the positive feelings of humans involved in each action in simulations, and weighing them against each other. These representations could be created by internal actions in LIDA's 'virtual window', a space in working memory reserved for simulations, in a multi-cyclic process (Franklin et al., 2013). The amount of positive feeling could be determined using LIDA's proposed theory of mind mechanism (Friedlander & Franklin, 2008).

However, there are inherent computational limitations to rules requiring simulations, especially if multiple humans might be affected by an action. In order to make the computation tractable, a limit would have to be imposed on the number of affected humans simulated, and on the time. 'Bottom-up' values would have to take over when that limit is exceeded by a difficult dilemma.

Adaptive incrementalist and moral test-driven development

Incrementalism proposes to progress toward a goal in a stepwise fashion, instead of starting out with a final theory or plan. Adaptive incrementalism in machine ethics (AIME) as proposed by (Powers, 2011) allows starting out with a small initial set of constraints or rules, testing the system, and then adding new constraints or rules if necessary. Constraints, rules, and functionalities of the system can be adapted or removed at a later date if it turns out that there is a more precise or effective way to constrain the system. This strategy of development allows starting without a complete theory or ethical framework, initially specifying

only basic behaviors with well-understood consequences, and subsequently extending the system in a stepwise fashion.

This model of a stepwise refinement of moral decision making is in accordance with current software development approaches (Larman & Basili, 2003). It also lends itself to Test-Driven Development (TDD). TDD, in its original form, is a development strategy which proposes to write the test cases for each feature first, and develop the feature functionality afterwards, so that it can be tested and verified immediately or later when it is changed. With each change or addition in the system, the entire test battery can be run to verify that the changes or additions do not impair existing functionality. TDD has been reported to lead to higher quality code, fewer malfunctions or defects, higher reliability, and reduction of testing effort (Müller & Padberg, 2003; Williams, Maximilien, & Vouk, 2003).

The idea of TDD can be extended to an adaptively developed moral decision making mechanism. For each function of the robot, a number of simple moral tests can be written, simulating a specific situation in which the function would be applicable and defining acceptable and unacceptable outcomes. For example, in a situation where a Care-O-Bot would detect a person falling, acceptable outcomes would be immediately calling for help, checking for vital signs and injuries, and calling an ambulance if necessary. Subsequent additions of functionality would include their own moral tests, but each time the system is changed, every other moral test would have to be passed as well. This reduces the risk of altering previously acceptable behaviour. For example, if action is added for the robot to go re-charge when the battery levels fall below a specific level, and this action would be selected in the moral test involving the falling person instead of calling for help,

the failed test would alert developers that the new action needs to be modified and constrained.

A final advantage to a battery of moral tests is that once developed it can perform run-time testing on the AMA. A dead man's switch type mechanism could immediately turn off the AMA if any of the tests fail at any point, due to any malfunctions or to newly learned behaviors that might interfere with the specified moral rules.

How could we obtain a large set of tests which minimize the probability of unethical or harmful actions in an efficient and practical fashion? Asaro (2006) suggests the existing legal system as a practical starting point for thinking about robot ethics, pointing out that legal requirements are most likely to provide the initial constraints of robotic systems, and that the legal system is capable of providing a practical system for understanding agency and responsibility (thus avoiding the need to wait for a consensual and well-established moral framework).

Extending this idea, legal cases might be a basis from which to derive tests for moral decision making mechanisms. Among other freely available sources, UNESCO's [5] bioethics curriculum provides a number of real-world case studies on relevant topics such as 'human dignity and human rights' (UNESCO, 2011b) or 'benefit and harm' (UNESCO, 2011a).

[5]The United Nations Educational, Scientific and Cultural Organization. http://www.unesco.org

A LIDA-based CareBot

Overview

This section describes CareBot, a partially implemented mobile assistive robot operating in a simple simulated environment, as an example constrained decision making mechanism based on the LIDA cognitive architecture.

Assistive robotics aims to provide aids for supporting autonomous living of persons who have limitations in motor and/or cognitive abilities e.g. the elderly or the severely disabled. This support can take different forms, for example (Pollack, 2005):

1. Providing assurance that the elder is safe and otherwise alerting caregivers (assurance systems)

2. Helping the person perform daily activities, compensating for their disabilities (compensation systems)

3. Assessing the person's cognitive status or health (assessment systems)

The CareBot simulation can perform some tasks in the first two categories, such as fetch and carry tasks (fetch food, drinks, or medicine for elderly or disabled individuals), and recognizing the absence of vital signs (and alerting caregivers if this occurs).

CareBot operates in, and is structurally coupled to, a simple simulated 2D environment consisting of patients (elderly or disabled), essential facilities such as a kitchen and a power outlet, items such as food, drinks and medication, and, of course, the CareBot (see Figure 1.2). The CareBot agent's main goal is to ensure

the patients' continued health and assist them in performing daily activities, while ensuring its own continued survival (recharging whenever necessary and avoiding bumping into obstacles).

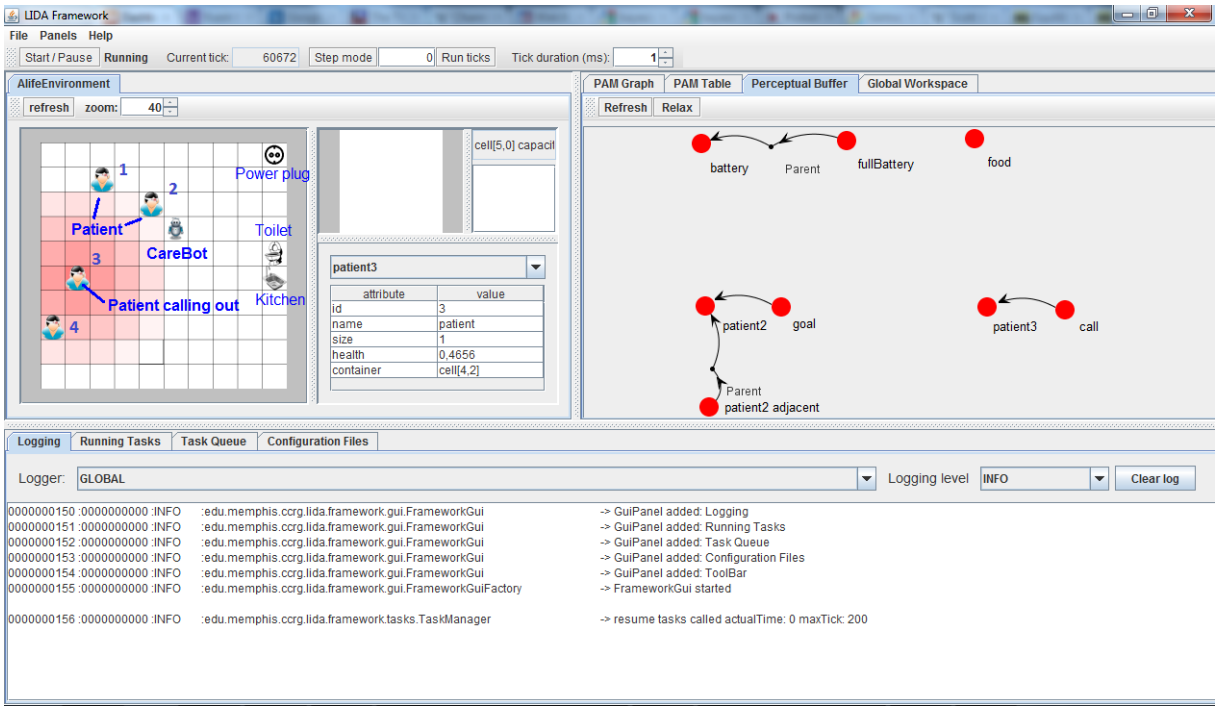

Figure 1.2: **The LIDA-based CareBot simulation environment**. Top left: the environment representation (the reddish gradient arount the patient calling out represents the auditory information that the agent can receive). Top right: diagnostic panels; shown here: the perceptual buffer (contents of the current situation model. Red circles represent PAM nodes). Bottom: Logging panel.

It achieves these goals using multimodal sensors (a simple visual and auditory sensor) and effectors enabling it to move around in the environment. The agent performs action selection using 'cognitive cycles', analogously to

action-perception cycles in the brain (Freeman, 2002; Fuster, 2002) after perceiving various objects, building a model of the current situation, and selecting important objects to attend to, these objects can compete for and enter functional (access) consciousness (Franklin et al., 2013) , after which the most appropriate action to deal with the current situation can be selected.

CareBot is constrained at the non-cognitive level (its speed is limited, and it is only allowed to carry light objects and to communicate with humans), and at the reactive level (its perceptual and procedural memories have been designed to respond appropriately to the demands of its limited domain). It might also be constrained at the deliberative level, e.g. by adding top-down rules to its declarative memory and allowing it to simulate consequences; however, this mechanism has not been implemented yet. Finally, constraints at the metacognitive level are beyond the supported mechanisms of the current LIDA computational framework.

A Simple Decision Making Example

Here we will describe what happens in each of the modules of the LIDA cognitive cycle outlined in the previous section, specifically:

1. Sensory Memory
2. Perceptual Associative Memory (PAM)
   (the modules above are part of the Perception phase)
3. Workspace
4. Transient Episodic and Declarative Memory
5. Attention Codelets

6. Global Workspace

   (the modules above are part of the Understanding phase)

7. Procedural Memory

8. Action Selection

9. Sensory-Motor Memory

   (the modules above are part of the ActionSelection phase)

In this simple simulated environment, no advanced visual and auditory processing was necessary (although there are two preliminary approaches for perceptual recognition in LIDA, a recent cortical learning algorithm inspired by predictive coding and perception as statistical inference (McCall & Franklin, 2013), and a simpler approach integrating SURF-based feature detection with LIDA (Madl & Franklin, 2012)).

An environment class is inspected periodically by the Sensory Memory module, and information is copied to visual and auditory buffers. Simple feature detectors monitor these buffers, and pass activation to their corresponding nodes in the Perceptual Associative Memory in a way similar to activation passing in an artificial neural network (although the modeling is done on a higher level) see (McCall et al., 2010). PAM nodes represent semantic knowledge about concepts or objects in the environment; the CareBot agent is initialized with knowledge required for its domain, such as e.g. PAM nodes representing the patients, their locations, the facilities and their locations (kitchen, medicine cabinet, toilet, powerplug), and internal state nodes representing the CareBots location and its power status.

After this perception phase, the identified percept (PAM nodes identified reliably, i.e. exceeding a specific activation threshold) is copied into the Workspace, constituting a preconscious working memory. If the content-associative long-term memories (Transient Episodic and Declarative Memory) contain memories relevant to the current percepts (such as e.g. the types of medication a specific patient might require), these memories are also copied into the Workspace as local associations. In the example in Figure 1.2, the Workspace contains current external percepts (patient 2, the auditory call of patient 3, the food being carried) and internal percepts (full battery status) as well as secondary concepts which are not directly perceptual (a goal representation, spatial relations).

Attention Codelets look out for perceptual representations of their own specific concern in the Workspace, form coalitions with them, and copy these coalitions to the Global Workspace, the short-term memory capacity that facilitates contents becoming functionally conscious. These coalitions subsequently compete for being broadcast consciously the one with the highest activation wins the competition, enters functional consciousness, and can be acted upon. The agent is consciously aware of an object, entity, or event, the moment the nodes representing these become part of the conscious broadcast after winning the competition. In the example in Figure 1.2, presumably there would be at least two coalitions in the Global Workspace competing for consciousness:

1. Since patient 2 is adjacent to the CareBot, and it is CareBot's goal to reach patient 2 and give him/her the food it is carrying, there would be a coalition with high activation containing these perceptual structures (patient 2, the adjacency relation, and the goal).

2. CareBot has also perceived a potentially important auditory call; therefore there would also be another high-activation coalition containing the perceptual representation of the call and the source associated with it (patient3).

In this example, coalition 1 would presumably win the competition, enter consciousness, and lead to an action. (Note that this outcome would be different if the call by patient 3 is a medical emergency. In this case, the representation of the call would have a much higher activation - determined by an appropriate emergency feature detector -, win the competition for consciousness, and lead to CareBot attending to patient 3).

After this understanding phase, the contents of the conscious broadcast are transmitted to the Procedural Memory module, which leads to the selection of relevant behavioral schemes (Drescher, 1991; Franklin et al., 2013), i.e. all schemes the context (precondition) of which is satisfied by the current conscious context will receive activation (depending on how well the context is matched). Schemes also receive top-down activation through goals or drives that match their result state. From all the possible schemes and actions, it is the task of the Action Selection module to select a single relevant action that the agent can execute.

In the example in Figure 1.2, this action would presumably be to give the food held by CareBot to patient 2 (unless the call of patient 3 is an emergency, as mentioned above). This selected action is now transmitted to the Sensory-Motor Memory for parameterization (highly important in robotics, but not needed in this simple simulation), and executed.

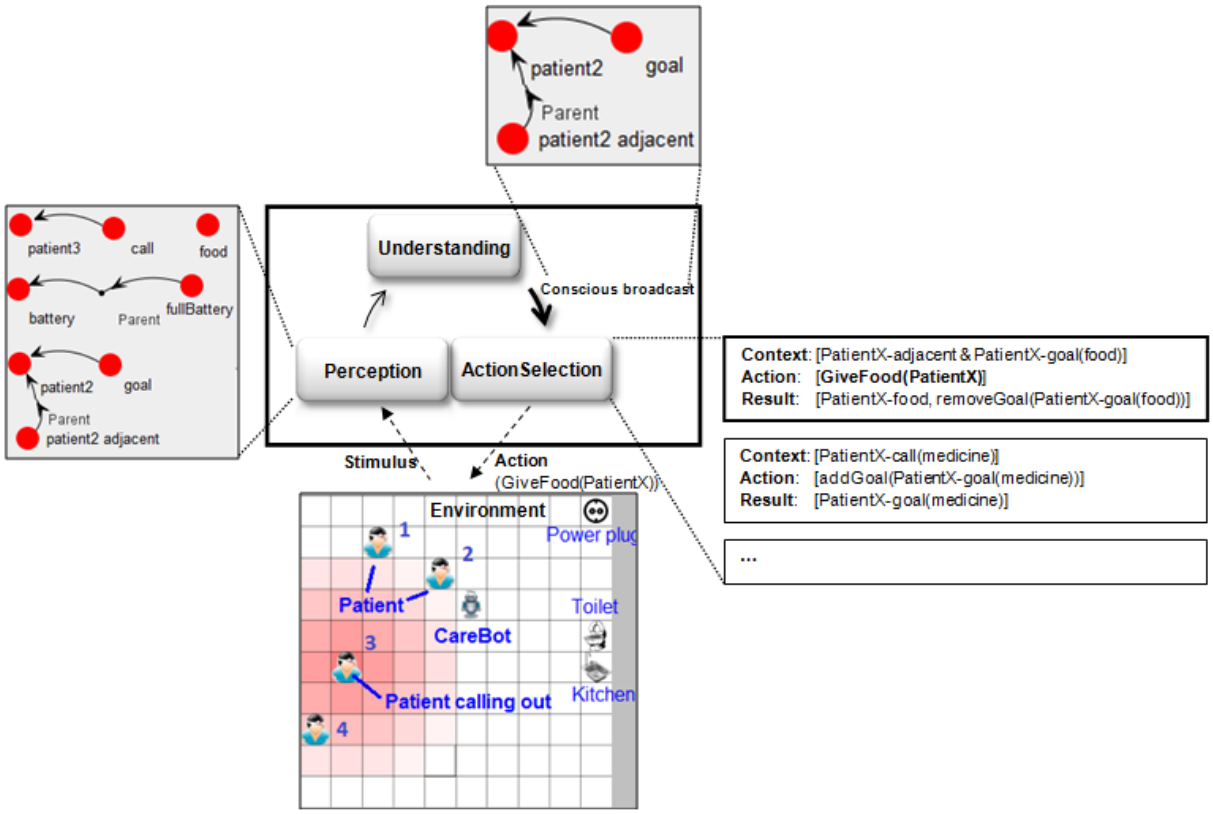

Figure 1.3: **The three phases of CareBot's cognitive cycle in the described example**.

Figure 1.3 illustrates the phases of CareBot in the example described above. This example illustrates single-cycle (consciously mediated) decision making, which has been implemented in LIDA agents.

When could the prevention of harmful actions become relevant in this scenario? Let us extend the above example by making the additional assumption that patient 2 is a diabetic, and that the food carried by CareBot is a dessert

containing a very high amount of sugar. Thus, it could be very dangerous for the patient to eat this food; nevertheless, he or she asked the robot to fetch it.

If CareBot were to provide fetch and carry tasks also for diabetic patients, this would require additional pre-programmed knowledge to ensure their safety. This could be ensured either using a top-down, or a bottom-up approach (as described in the previous section). The simpler alternative would be a bottom-up solution that constrains actions which might endanger the patient - 'concern' emotion PAM nodes activated by foods with high sugar content (detected by appropriate feature detectors in PAM passing activation to this node). Furthermore, additional behavior schemes in procedural memory would have to be defined to deal with situations raising concern, such as contacting a human to ask for help.

Instead of the action to give patient 2 the high-sugar food, the 'concern' node would lead to the selection of a behavior scheme alerting a human caregiver or doctor (who might explain to the patient why he or she should not eat this food, or suggest an alternative), thus preventing harm and preserving the patient's health.

Finally, if - for whatever reason - the robot could not reach a human to ask for help, a volitional, deliberative decision making process as outlined in Section 2 could be used to weigh the main options against each other (respect the patient's autonomy and give him the food, vs. ensure the patient's continued health and ask him to wait until a human arrives to make a final decision). This would require performing internal actions in the 'virtual window' of LIDA's Workspace, as well as knowledge about the effects of sugar on diabetic patients in long-term declarative memory, for evaluating the consequences of those actions. Volitional decision making is not implemented in LIDA as of yet (although it was

implemented in IDA (Franklin, 2000), its predecessor); and the internal actions and 'virtual window' (Franklin et al., 2013; McCall et al., 2010) of the computational architecture are not developed well enough to implement such a comparison process at this time.

Conclusion

Full ethical frameworks are difficult to implement with current methods, but simplified, constrained moral decision making problems might be tractable. In this paper, we have suggested four ways to constrain robot ethics implementations, and argued that biologically inspired cognitive architectures might be good starting points for an implementation (since some of the mechanisms they provide are also needed for moral decision making; and they are designed to be human-like, and can be used in limited domains to approximate human-like decisions). We have described an approach to use test cases to help ensure that extensions of ethical systems and autonomous learning preserve correct behavior and do not lead to harmful actions. We have also outlined a possible moral decision making mechanism based on the LIDA cognitive architecture, and described a partial implementation.

Although most cognitive architectures in general, and LIDA in particular, are still in early stages of development, and still far from being adequate bases for implementations of human-like ethics, we think that they can contribute to both the understanding, design, and implementation of constrained ethical systems for robots, and hope that the ideas outlined here might provide a starting point for future research.

References

Allen, C., Varner, G., & Zinser, J. (2000). Prolegomena to any future artificial moral agent. *Journal of Experimental & Theoretical Artificial Intelligence*, *12*(3), 251–261.

Alonso, I. G. (2011). Service robotics. In *Service robotics within the digital home* (pp. 89–114). Springer.

Anderson, M., & Anderson, S. L. (2011). *Machine ethics*. Cambridge University Press.

Asaro, P. M. (2006). What should we want from a robot ethic. *International Review of Information Ethics*, *6*(12), 9–16.

Baars, B. J. (2005). Global workspace theory of consciousness: toward a cognitive neuroscience of human experience. *Progress in brain research*, *150*, 45–53.

Baars, B. J., & Franklin, S. (2009). Consciousness is computational: The LIDA model of global workspace theory. *International Journal of Machine Consciousness*, *1*(01), 23–32.

Barsalou, L. W. (1999). Perceptual symbol systems. *Behavioral and brain sciences*, *22*(04), 577–660.

Burns, L. D. (2013). Sustainable mobility: A vision of our transport future. *Nature*, *497*(7448), 181–182.

Ciupe, V., & Maniu, I. (2014). New trends in service robotics. In *New trends in medical and service robots* (pp. 57–74). Springer.

Drescher, G. L. (1991). *Made-up minds: a constructivist approach to artificial intelligence*. The MIT Press.

Franklin, S. (2000). Deliberation and voluntary action in conscious software

agents. *Neural Network World*, *10*, 505–521.

Franklin, S., Madl, T., DMello, S., & Snaider, J. (2013). LIDA: A systems-level architecture for cognition, emotion, and learning. *IEEE Transactions on Autonomous Mental Development*, *PP*, 1–1. doi: 10.1109/TAMD.2013.2277589

Franklin, S., & Patterson Jr, F. G. (2006). The lida architecture: Adding new modes of learning to an intelligent, autonomous, software agent. *pat*, *703*, 764-1004.

Freeman, W. J. (2002). *The limbic action-perception cycle controlling goal-directed animal behavior* (Vol. 3).

Friedlander, D., & Franklin, S. (2008). LIDA and a theory of mind. In *Artificial general intelligence, 2008: Proceedings of the first agi conference* (Vol. 171, p. 137).

Fuster, J. M. (2002). Physiology of executive functions: The perception-action cycle. *Principles of frontal lobe function*, 96–108.

Graf, B., Hans, M., & Schraft, R. D. (2004). Care-o-bot iidevelopment of a next generation robotic home assistant. *Autonomous robots*, *16*(2), 193–205.

IFR. (2013). *World Robotics 2013 Service Robot Statistics.* Retrieved 23/12/2013, from `http://www.ifr.org/service-robots/statistics/`

Kanerva, P. (1988). *Sparse distributed memory*. MIT Press.

Larman, C., & Basili, V. R. (2003). Iterative and incremental developments. a brief history. *Computer*, *36*(6), 47–56.

Lin, P., Abney, K., & Bekey, G. A. (2011). *Robot ethics: The ethical and social implications of robotics*. The MIT Press.

Madl, T., & Franklin, S. (2012). A lida-based model of the attentional blink. *ICCM 2012 Proceedings*, 283.

Madl, T., Franklin, S., Chen, K., & Trappl, R. (2013). Spatial working memory in the lida cognitive architecture. In *Proceedings of the 12th international conference on cognitive modelling* (pp. 384–390).

Maes, P. (1989). How to do the right thing. *Connection Science*, *1*(3), 291–323.

McCall, R., & Franklin, S. (2013). Cortical learning algorithms with predictive coding for a systems-level cognitive architecture. In *Proceedings of the second annual conference on advances in cognitive systems* (pp. 149–166).

McCall, R., Franklin, S., Friedlander, D., & D'Mello, S. (2010). Grounded event-based and modal representations for objects, relations, beliefs, etc.. In *Flairs-23 conference.*

Müller, M. M., & Padberg, F. (2003). About the return on investment of test-driven development. In *Edser-5 5 th international workshop on economic-driven software engineering research* (p. 26).

Negatu, A., Franklin, S., & McCauley, L. (2006). A non-routine problem solving mechanism for a general cognitive agent architecture. In *Problem solving: Techniques, steps, and processes.* Nova Science Publishers.

Pollack, M. E. (2005). Intelligent technology for an aging population: The use of ai to assist elders with cognitive impairment. *AI magazine*, *26*(2), 9.

Powers, T. M. (2011). Incremental machine ethics. *Robotics & Automation Magazine, IEEE*, *18*(1), 51–58.

Scheutz, M. (2013). What is robot ethics? *Robotics & Automation Magazine, IEEE*, *20*(4), 20–165.

Sloman, A. (1999). What sort of architecture is required for a human-like agent. *Foundations of Rational Agency*, 35–52.

Snaider, J., & Franklin, S. (2012). Extended sparse distributed memory and sequence storage. *Cognitive Computation*, *4*(2), 172–180.

Tellez, R., Ferro, F., Garcia, S., Gomez, E., Jorge, E., Mora, D., ... Faconti, D. (2008). Reem-b: An autonomous lightweight human-size humanoid robot. In *8th ieee-ras international conference on humanoid robots* (pp. 462–468).

UNESCO. (2011a). *Casebook on benefit and harm* (Vol. 2). Retrieved from http://www.unesco.org/ulis/cgi-bin/ulis.pl?catno=192370&set=52C1C48A_1_207&gp=1&lin=1&ll=1

UNESCO. (2011b). *Casebook on human dignity and human rights* (Vol. 1). Retrieved from http://www.unesco.org/ulis/cgi-bin/ulis.pl?catno=192371&set=52C1C48A_1_207&gp=1&lin=1&ll=1

Wallach, W., & Allen, C. (2008). *Moral machines: Teaching robots right from wrong*. Oxford University Press.

Wallach, W., Allen, C., & Smit, I. (2008). Machine morality: bottom-up and top-down approaches for modelling human moral faculties. *AI & Society*(22), 565–582. doi: 10.1007/s00146-007-0099-0

Wallach, W., Franklin, S., & Allen, C. (2010, May). A Conceptual and Computational Model of Moral Decision Making in Human and Artificial Agents. *Topics in Cognitive Science*, *2*(3), 454–485. doi: 10.1111/j.1756-8765.2010.01095.x

Williams, L., Maximilien, E. M., & Vouk, M. (2003). Test-driven development as a defect-reduction practice. In *14th international symposium on software*

reliability engineering (pp. 34–45).

Wyrobek, K. A., Berger, E. H., Van der Loos, H. M., & Salisbury, J. K. (2008). Towards a personal robotics development platform: Rationale and design of an intrinsically safe personal robot. In *ICRA 2008* (pp. 2165–2170).