PAGI World: A Physically Realistic, General-Purpose Simulation Environment for Developmental AI Systems

John Licato[1] **and Selmer Bringsjord**[2]

Analogical Constructivism and Reasoning Lab[1]
Indiana University/Purdue University - Fort Wayne

Rensselaer AI and Reasoning Lab[2]
Rensselaer Polytechnic Institute

Abstract. There has long been a need for a simulation environment rich enough to support the development of an AI system sufficiently knowledgeable about physical causality to pass certain tests of Psychometric Artificial Intelligence (PAI) and Psychometric Artificial General Intelligence (PAGI). In this article, we present a simulation environment, PAGI World, which is: cross-platform (as it can be run on all major operating systems); open-source (and thus completely free of charge to use); able to work with AI systems written in almost any programming language; as agnostic as possible regarding which AI approach is used; and easy to set up and get started with. It is our hope that PAGI World will give rise to AI systems that develop truly rich knowledge and representation about how to interact with the world, and will allow AI researchers to test their already-developed systems without the additional overhead of developing a simulation environment of their own. After clarifying both PAI and PAGI, we summarize arguments that there is great need for a simulation environment like PAGI World. We present multiple examples of already-available PAI and PAGI tasks in PAGI World, covering a wide range of research areas of interest to the general-purpose AI community.

1 Introduction

Toward the end of his long and extremely distinguished career, Jean Piaget began to name and concretely describe some mechanisms he believed were responsible for the emergence of many features of mature cognition: formal reasoning, an understanding of causality, and analogical ability were among these features, along with many others [26, 25, 28]. Piaget had long suspected that these features and the concepts they relied on were constructed by the child using simpler schemas acquired through interaction with the physical world, at least since (Piaget and Inhelder 1958). Thus the role that the world plays in shaping the constructs and abilities of the child, which informs the related question of how much AI can progress without having a real-world-like environment, has been a cornerstone issue in AI for some time now [10, 11, 18].

But modeling these Piagetian beliefs is, to this day, an unmet goal that has existed at least since (Drescher 1991). Such modeling is a dream of computational cognitive modelers, but, perhaps more specifically, is a goal of the field of developmental AI. This is the field which attempts to show how, using an agent endowed with minimal innate capacities embedded in a sufficiently rich environment, higher-level cognitive abilities can emerge [17]. These abilities may include logico-mathematical reasoning, an understanding of causality, robust analogical reasoning, and others. Furthermore, work in developmental AI systems strives to show that the emergence of such abilities could be reflective of the way they develop in humans, whether this is in the pattern predicted by Piaget's stage theories or not.

This paper describes a task-centered, physically realistic simulation environment that we have developed to simultaneously address a set of challenges in evaluating AGIs. We motivate PAGI World in Section 2, introduce PAGI World in Section 3, and outline examples of a wide variety of tasks in PAGI World in Section 4.

2 Motivations

Here we summarize three categories of motivations for PAGI World, particularly of interest to those wishing to evaluate artificially-general intelligence (AGI). **C1** - **C6** specify conditions for a *sufficiently rich* simulation environment. The Tailorability Concern (Section 2.2) deals with the way in which an AGI acquires and constructs its knowledge. Finally, Section 2.3 puts forth our belief that an AGI's knowledge should be *expressive*, in the sense of logical expressivity.

2.1 Guerin's Conditions

Frank Guerin [17], in his recent survey of the developmental AI field, concluded that current systems were lacking in several key areas. Guerin then suggested that a major reason (arguably the most important reason) why the field has the shortcomings he described, is the absence of a suitable simulation environment. Current simulation environments used by developmental-AI projects were missing several key features, and Guerin described some conditions that would need to be met by simulation environments in order to address this problem. We refer to the most important of these conditions as **C1**, **C2**, and **C3**. A sufficiently rich simulation environment for developmental AI should, at a minimum:

C1 be rich enough to provide knowledge that would bootstrap understanding of concepts rooted in physical relationships; e.g.: inside vs. outside, large vs. strong, etc.

C2 allow for the modeling and acquisition of spatial knowledge, which Guerin notes is widely regarded to be a foundational domain of knowledge acquisition, through interaction with the world.
C3 support the creation and maintenance of knowledge the agent can verify itself.

[21] introduced a few additional conditions:

C4 be rich enough to provide much of the sensory-level information that an agent in the real world would have access to.
C5 allow for testing of a virtually unlimited variety of tasks, whether these are tasks testing low-level implicit knowledge, high-level explicit knowledge, or any of the other areas required by Psychometric Artificial General Intelligence (PAGI). Ideally, such a system would support the easy creation of new tasks and environments without requiring a massive programming effort.
C6 provide pragmatic features enabling tasks to be attempted by researchers using different types of systems and different theoretical approaches, thus enabling these different approaches to be directly compared with each other.

These conditions were elaborated on and defended in [21], so we will not do so here. A common theme running through all six conditions is that what is lacking from current microworlds is a physically realistic environment—one in which the agent can acquire, develop, and test its concepts. But the concerns raised by Guerin are not only of interest to the field of Developmental AI; in point of fact, *all* of AI can benefit by addressing them. For example, **C1** is extremely important for cognitive models of analogy, as they struggle to overcome what has been called the *Tailorability Concern* (TC)[16, 22].

2.2 The Tailorability Concern

TC, in essence, is the concern that models of analogy (though this can be applied to all cognitive architectures in general) work almost exclusively with manually constructed knowledge representations, using toy examples often *tailor-made* to display some limited-scope ability. Licato et al. ([22]) argue that overcoming TC is necessary to advance the fields of analogy and cognitive architectures, by developing a set of conditions that must be met in order to claim victory over TC:

> **$\mathbf{TCA_3}$** A computational system of analogy answers TC if and only if given no more than either
>
> - unstructured textual and/or visual data, or
> - a large, pre-existing database,
>
> and minimal input, it is able to consistently produce *useful* analogies and demonstrate stability through a variety of input forms and domains.

According to **$\mathbf{TCA_3}$**, then, good performance on the part of a cognitive agent on a sufficiently large knowledge-base from which source analogs could be drawn is required to answer TC. An agent interacting in the sort of microworld called for by Guerin might ideally be able to acquire such source analogs by simply interacting with its environment.

C1 and TC together require that the microworld itself is what provides the knowledge drawn upon to construct concepts of basic physical relationships, not manually constructed source analogs or fully explicit logical theories. **C2** expands on **C1** by requiring that this knowledge of physical relationships not be static, but rather should allow for an agent in the world to learn through interaction. The idea that children learn by initiating interactions with the world based on their (often incomplete) conceptions of reality—in a manner that resembles scientific experimentation—was championed by Piaget and later, Piaget-influenced work [2, 29, 27, 39].

Following **$\mathbf{TCA_3}$**, another formulation of the Tailorability Concern and recommendation for how to surpass it was also presented in [22]:

> **$\mathbf{TCA_4}$** A computational system $\mathcal{A}$ for analogy generation answers TC if and only if, given as input no more than either
>
> - unstructured textual and/or visual data, or
> - a vast, pre-existing database not significantly pre-engineered ahead of time by humans for any particular tests of $\mathcal{A}$,
>
> is—in keeping with aforementioned *Psychometric AI*—able to consistently generate analogies that enable $\mathcal{A}$ to perform *provably well* on precisely defined tests of cognitive ability and skill.

$\mathbf{TCA_4}$ ties TC to Artificial General Intelligence (AGI) by introducing the concept of *Psychometric AI* (PAI) [3, 9]. PAI sees good performance on well-established tests of intelligence as a solid indicator of progress in AI. Some may note that most intelligence tests fail to capture human-level skills such as creativity and real-time problem solving; therefore, related to PAI is Psychometric Artificial *General* Intelligence (PAGI) [6]. For example, one test of PAGI is Bringsjord and Licato's (2012) *Piaget-MacGyver Room*, in which an agent is inside a room with certain items and a task to be performed. The agent must achieve the task using some combination of the items in the room (or using none of them, if possible). Depending on the task, the solutions may require using the items in unusual ways, as viewers of the MacGyver television series may remember. We describe several example Piaget-MacGyver Rooms in PAGI World in Section 4.[1]

2.3 Expressivity

Even after satisfying TC, it would be difficult to claim an AGI is truly general-intelligent unless its knowledge satisfies a certain degree of *logical expressivity*. By this, we do not mean that an AGI must possess a Gödel-like mastery of formal logic. Rather, the term refers to the well-established hierarchy of expressivity in formal theories. For example, a formal theory equivalent in expressivity to first-order logic (FOL) can express anything that one equivalent to propositional calculus (PC) can express. The converse is not true; something like "all men are mortal" simply cannot be expressed in a quantifier-free logic like PC, since the ability to take any possible man m and apply the statement "all men are mortal" to deduce that m is mortal is only possible with machinery that treats quantified variables *as variables that can be quantified over*.

Logical expressivity, then, is a real restriction on any formal theory's ability to express properties.[2] But the use of terminology from mathematical logic should not obscure the more general fact that logical expressivity is really a restriction on *any* system whatsoever that

[1] Note that although we have adopted "PAGI World" as the name of our simulation environment in order to reflect the fact that it is designed to support many types of PAGI tests (including variants of the Piaget-MacGyver Room, as we describe below), PAI tests are just as easily implementable in PAGI World.

[2] See [33] for a good initial definition of what it means to express properties in formal theories.

can be described using rules for producing new behaviors, actions, knowledge, or structures. All AI systems in history are no exception; once we formally describe the set of rules that govern that AI system, those rules fall under some level of logical expressivity, and whatever that level is limits what that system can ultimately do.

Furthermore, FOL's expressivity is not enough for general intelligence. Humans routinely reason over sets, analogies, the beliefs, desires, and intentions of others, and so on. Such concepts require logics significantly more expressive than FOL: second-order logic, epistemic/modal logics, even third-order logic in some cases [7]. If an AGI is to truly be as general-purpose a reasoner as the typical human, a high level of expressivity is needed.

For the first time, we present here a conjecture[3] encapsulating this view:

> **AGI>FOL.** No system can claim to be an AGI unless its knowledge is at least more logically expressive than first-order logic.

A very high-level proof sketch of the above is as follows:

1. a concept $\mathscr{C}$ is accurately captured in a system $\mathcal{S}$ only if that system can, at a minimum, produce any actions, inferences, behaviors, or knowledge structures that would be expected of a system capturing $\mathscr{C}$.
2. A system cannot thus fully capture a concept $\mathscr{C}$ if its knowledge representation is below the level of logical expressivity required for $\mathscr{C}$.
3. There are many concepts required for AGI which are at a level of logical expressivity higher than FOL.
4. Thus, no artificial system with an expressivity at the level of FOL or lower can be an AGI.

We omit many details here, but the argument presented is at the core of a more encompassing argument for expressivity in AGI systems, currently under development. For our present purposes, suffice it to say that simulation environments which restrict the expressivity level of the knowledge of the agents which can use the environments to, or below that of FOL, can not hope to see the creation of fully general intelligence. PAGI World avoids that by placing no restrictions on the form of knowledge used by its artificial agents.

3 Introducing PAGI World

Condition **C6** is the most practicality-oriented, reflecting both Guerin's (2011) and our own inclination to believe that an effective way to compare AI and AGI methodologies would be to see how they perform on the same tasks, implemented on the same systems. But few such tasks and systems exist, and therefore before we describe PAGI World, it may be helpful to take a step back and look at our project in a broader view.

3.1 Why Isn't Such a System Already Available?

Given its potential benefit to the field as a whole, why does such an environment not currently exist, and do any of the roadblocks currently in the way affect the plausibility of our current project?

[3] The authors believe strongly in the truth of **AGI>FOL**, and ultimately hope to elevate it to the status of a theorem. However, as the present paper's scope permits only a loose proof sketch, we present it here as merely a conjecture.

3.1.1 Technical Hurdles

One potential roadblock is obvious: programming a realistic physics simulation is *hard*. Some of this difficulty is reduced by working with a 2D, rather than a 3D, environment. Although some software libraries have previously been available for 2D physics simulations, they have often been very language-specific and somewhat difficult to configure.

Secondly, even if one were to stick with a 2D physics library and commit to it, substantial development resources would be needed to enable the resulting simulation to run on more than one major operating system. Furthermore, even if *that* problem is somehow addressed, there is a vast diversity of languages that AI researchers prefer to use: Python, LISP (in various dialects, each with their own passionate proponents), C++, etc. All of these technical issues tend to reduce how willing researchers are to adopt particular simulation environments.

Fortunately, all of the above problems can be solved with a single design choice. Unity, a free game-development engine, has recently released a 2D feature set,[4] which comes with a 2D physics model that is extremely easy to work with. Furthermore, Unity allows for simultaneous compilation to all major operating systems, so that developers only have to write one version of the program, and it is trivial to release versions for Mac OS, Windows, and Linux. Because Unity produces self-contained executables, very little to no setup is required by the end users.

Finally, because Unity allows scripting in C#, we were able to write an interface for AI systems that communicates with PAGI World through TCP/IP sockets. This means that AI scripts can be written in *virtually any* programming language, so long as the language supports port communication.

3.1.2 Theoretical Hurdles

Unity conveniently helps to remove many of the technical roadblocks that have previously blocked the development of simulation environments that can be widely adopted. But there are also theoretical roadblocks; these are problems pertaining to the generality vs. work-required tradeoff. For example, if a simulation environment is too specifically tailored to a certain task, then not only can systems eventually be written to achieve that particular task and nothing else, but the simulation environment quickly becomes less useful once the task is solved. On the other hand, if the system is too general (e.g. if a researcher decides to start from scratch with nothing but Unity), then the researcher must devote too much time and energy to developing a new simulation environment for each project, rather than spending time on the AI itself.

PAGI World was designed with this tradeoff in mind. A *task* in PAGI World might be thought of as a Piaget-MacGyver Room with a configuration of objects. Users can, at run-time, open an object menu (Figure 1) and select from a variety of pre-defined world objects, such as walls made of different materials (and thus different weights, temperatures, and friction coefficients), smaller objects like food or poisonous items, functional items like buttons, water dispensers, switches, and more. The list of available world objects will frequently be expanding and new world objects will be importable into tasks without having to recreate tasks with each update. Perhaps most importantly, tasks can be saved and loaded, so that as new PAI/PAGI experiments are designed, new tasks can be created by

[4] In fact, the blog post making the announcement of the 2D feature set was dated November 12, 2013.

anyone. Section 4 illustrates the wide variety of tasks that can be created with such a system.

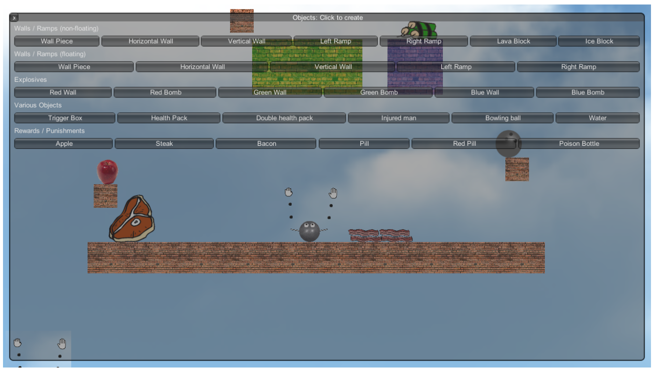

Figure 1: PAGI World With the Object Menu Visible

3.1.3 Other Simulation Environments

There have been some notable attempts to provide simulation environments for AI systems, particularly those inspired by the Developmental-AI approach. For example, in [12] Bruce created a Developmental-AI testbed by updating an older version created by Frank Guerin.

Although some of the present paper's authors are sympathetic to the power of Piagetian schemas and the AI systems derived from Piaget's theories, Bruce's system is tightly coupled with a particular cognitive architecture (presented in the same paper) that uses schema-based AI systems, whereas PAGI World, as we have said, is agnostic about what AI approach is used. It is unclear how easy or difficult it would be to adapt arbitrary cognitive architectures to work with their simulation environment.

They used the JBox2D library for their physics engine, which, according to [12], was poorly documented and difficult to work with (e.g., implementing a method to detect when the robot hand touched an object took markedly longer than they planned due to a lack of documentation for JBox2D). Although a newer version of JBox2D became available afterwards, implementing the new version requires the simulation programmer to manually update the relevant code, whereas updates to the Unity 2D physics engine automatically propagate to PAGI World, without any code changes on our part.

3.1.4 Drescher's Simulation

In [14], Drescher proposes an early microworld in which an agent, making use of a primitive form of Piagetian schemas, explores the world and learned about the objects with which it interacted. Although this was a promising start, after its initial success it was not developed further, nor was any significant effort made by other researchers to pick up on Drescher's work, as far as we are aware (only one small-scale re-implementation of Drescher's work exists, e.g. [13]).

Drescher's microworld consists of a 2D scene divided into a grid that limits the granularity of all other elements in the microworld. Inside this microworld are objects that take up discrete areas of the grid and contain visual and tactile properties, in the form of numerical vectors with arbitrarily chosen values.

Most importantly, the microworld contains a single robot-like agent with a single hand that can move in a 3-cell × 3-cell region relative to the part of the robot's body considered to be its "eye." If the hand object is adjacent to an object in the world (including the robot's own body), a four-dimensional vector containing tactile information is returned to the agent. The body has tactile sensors as well, though they do not return tactile information as detailed as that returned by the tactile sensors of the hand.

Visual information is available as well, in the form of a visual field whose position is defined relative to the robot's body. A smaller region within the visual field, called the *foveal region*, represents the area within the visual field where the robot is currently looking. The foveal region returns vectors representing visual information, and the cells in the visual field not in the foveal region also return visual information, but with lower detail.

Perhaps one of the most interesting features of Drescher's microworld is that the robot can only interact directly with the world by sending a set of predefined "built-in actions." Although the internal schema mechanism of the robot may learn to represent actions as richer and more complicated, ultimately what is sent to the simulation environment is always extremely low-level. Likewise, the information provided to the robot is always extremely low-level. The task of identifying and naming objects in the world—and even of knowing that objects in the world consistently exist!—is up to the learning mechanism the robot utilizes.

The fact that the learning and control system of the artificial agent can be developed almost completely independently of the features of the world itself, is one of the primary reasons why Drescher's microworld is appealing, and was selected as a starting point for PAGI World. PAGI World departs from, and has innovated beyond, Drescher's microworld in several key areas:

- **Agnosticism re. the AI method used**. Whereas Drescher's microworld was created for the sole purpose of testing his Piagetian schema-learning mechanism, we have designed the world, program, and interfaces so that as wide a variety as possible of AI techniques can be productively and easily used.
- **Optional mid-level input**. Related to the previous point, we realize that some researchers simply won't want to translate vector input for every piece of tactile or visual information they come across, and so we offer the option for the agent to directly receive the name of the object upon touching or viewing it.
- **Granularity**. The granularity of our world is dramatically finer; consider the increase in size of the visual field: Drescher's was an area of 7-×-7 cells with one visual sensor per cell. We have improved the visual area to span a 450-×-300 unit area, with each visual sensor spaced 15 units from its nearest neighbor (each unit roughly corresponds to a screen pixel).
- **Vision system**. In addition to having a wider visual field, ours has no foveal region, because the tasks we design require a visual field large enough to observe multiple objects at once. Certainly it is plausible that rapid eye movements can account for this ability in human beings, but our initial investigations found it to have too little theoretical benefit compared to how difficult it made working with the system.
- **Hands**. We have given the robot two hands instead of one, each with a similar range of motion, but with different distances (relative to the body) that each can reach. Although the simulation world is 2D, the hands exist on a separate layer that floats "above" objects in the world, analogously to a mouse cursor in any major operating system. The hands can grip and move objects they

are floating over (just as one might click and drag an object in Windows or MacOS), provided the objects are not too heavy or otherwise held down.

- **Realistic Physics**. Certainly a very important improvement we introduce is the aforementioned realistic physics provided by Unity 2D.
- **Focus on a wide breadth of tasks**. Although Drescher's microworld was a start in the right direction, we feel that it did not quite make enough of a push to be considered a simulation environment for AGI tasks, nor did it explicitly set out to be a testbed for the sort of tasks prescribed by Psychometric AI.

3.2 The Architecture of a PAGI World Setup

Figure 2 pictures the architecture of a typical PAGI World + AI controller pairing. As the figure illustrates, it is helpful to think of the processes controlled by the PAGI-World application to be the PAGI side, as opposed to the side which can be completely implemented externally, referred to as the AI side. The reflex and state machine described in Section 3.3 is controlled and managed on the PAGI side, but both states and reflexes can be dynamically modified through commands sent by the AI side.

All commands going from the AI side to the PAGI side, and all sensory information passing in the other direction, is passed via messages communicated through TCP/IP ports. Therefore, the AI-side can be written in any programming language that supports the creation, and decoding, of strings over TCP/IP. Although this flexibility sets PAGI World apart from many other alternatives, some may prefer an additional level of abstraction on the AI side, and for this reason we provide, and are continuing development on, a Python API called *pyPAGI*.

Tasks can be created, saved, and loaded using the GUI editor at run-time (Figure 1), but as suggested by Figure 2, they can also be somewhat configured by AI-side commands. This can be useful to modify the layout of the task dynamically in response to actions the AI agent takes (e.g., making an apple appear as a reward, or a bottle of poison as a punishment), or to load new tasks after successful task completion for automated batch processing of tasks.

3.3 Reflexes, DFAs, and the Implicit vs. Explicit Distinction

Although communication through TCP/IP ports is relatively quick, and the command system we have created is designed to be efficient, there are some actions that require extremely rapid, simple checks and responses. For example, holding an object in the air at a certain position relative to the body for an extended period of time may require many quick corrections. If the object starts to move down, more upward force should be applied. But if it moves too far up, downward force should be applied (or the amount of upward force should be reduced). In order to hold the object as still as possible, the amount of force applied would be based on its current and projected velocity and position. However, if the AI script requests this information, does a calculation to determine the amount of correction required, and sends back the command to adjust the amount of force, by the time this command is received by PAGI World and processed it may be inaccurate.

PAGI World fixes this problem by implementing *states* and *reflexes*. Reflexes and states can be set and modified through commands from the AI script, but they are actually checked and executed completely on the PAGI-World side, which allows for much faster reaction times. A reflex r consists of a tuple $(\mathbf{C},\mathbf{A})$, where $\mathbf{C}$ is a list of conditions and $\mathbf{A}$ is a list of actions. Each condition in $\mathbf{C}$ must be satisfied in order for reflex r to activate. These conditions can consist of sensory inequalities, for example: whether one of the tactile sensors detects a temperature above a certain amount, or whether the AI agent's body is moving above a certain velocity. If all of the conditions are met, then the actions are executed immediately. Furthermore, sensory inequalities can be specified as simple arithmetic functions of sensory values, so that a reflex can be fired if (to cite an arbitrary example) the horizontal component of the agent's body's velocity is at least twice the value of the vertical component of its velocity.

States can be activated and checked by reflexes. Essentially, this means that multiple deterministic finite automata (DFAs) can be stored and executed completely on the PAGI side. However, the expressivity of the conditions and actions within each reflex strictly restricts the system so that full Turing machines cannot be implemented on the PAGI side. This allows developers to implement two important categories of abilities generally regarded to be part of the human experience: explicit, and implicit. Recall that the explicit vs. implicit distinction divides the mind into explicit processes which are generally slow, deliberate, and easy to verbalize, versus implicit processes which are mostly quick, automatic, and not easily accessible to the conscious mind [35].

The implicit/explicit distinction [35], which roughly parallels the System 1/System 2 distinction of Kahneman [20] (but see [37] for a criticism of System 1 vs. 2), encompasses an extremely broad spectrum of explanations for human phenomena [34, 35, 36]. If a simulation environment restricts itself to AI controllers that rely on explicit or implicit processes exclusively, then it cannot hope to capture the breadth of tasks required to qualify a Psychometric Artificial *General* Intelligence. If PAGI tasks are meant to subsume all tasks solvable by neurobiologically normal human adults, then a simulation environment designed to capture PAGI tasks should also be able to test AI agents on their use of both explicit and implicit knowledge.

Although the PAGI side does not support all imaginable implicit processes (for example, some might believe that a Bayesian probabilistic approach or a Deep Neural Network is necessary to implement some implicit processes), the fact that multiple DFAs can be stored and executed in PAGI World's optimized code gives the user flexibility to capture a wide range of implicit processes. Furthermore, in keeping with the design principles of PAGI World, AI systems built on implicit processes can still be implemented fully on the AI side.

4 Some Example Tasks

We have designed some tasks to demonstrate the range of possibilities and showcase some of PAGI World's unique features.

4.1 Piaget-MacGyver Rooms

4.1.1 The Water Diversion Piaget-MacGyver Room

A prime example of a typical MacGyver task comes from Season 2, Episode 5 of the MacGyver television series. Angus MacGyver, the series' titular character, found a friend of his being threatened by a mountain lion. MacGyver, positioned at a ledge above both his friend and the mountain lion, reconfigured some rocks and a log so that he could guide a nearby stream of water in such a way that it created a small waterfall separating his friend and the mountain lion. The mountain lion ran away immediately.

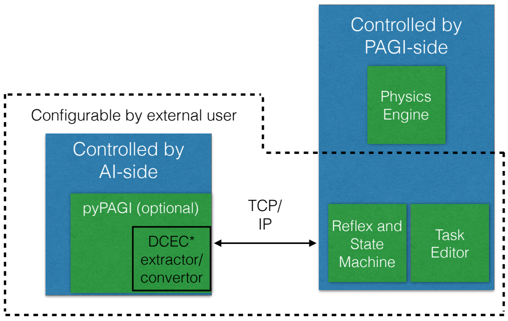

Figure 2: The architecture of an instance of PAGI World and an AI controller. Everything on the AI side can be written by AI researchers, as the interface with the PAGI side is handled through messages passed over TCP/IP sockets. A Python library, called *pyPAGI*, is also optionally available to assist researchers with common AI-side functionality, including encoding of PAGI-World knowledge in the Deontic Cognitive Event Calculus ($\mathcal{DCEC}^*$). The reflex/state machine and task editors can also be controlled through TCP/IP, though the task editor is additionally available through a WYSIWYG drag-and-drop interface.

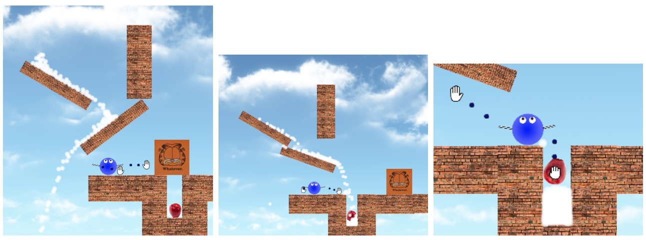

Figure 3: A sample Piaget-MacGyver Room in which the agent is expected to direct the flow of water in order to reach an apple.

Of course, other solutions may have been available. Perhaps MacGyver could have simply thrown rocks at the mountain lion, or fashioned a bow and arrow out of twigs, sharpened stones, and parts of his knapsack. But these different solutions would have come with their own unique advantages and disadvantages, and furthermore, to not lose sight of the PAGI-oriented question: Could an artificially intelligent agent figure out *any* of these solutions without having been specifically trained for that particular solution? PAGI problems such as the Piaget-MacGyver room challenge researchers to find answers to this question.

The ability to direct the flow of water opens up a wide range of tasks, which we can model in PAGI World. Using the Fluvio library for fluid dynamics, PAGI World can generate fluid-like particles. These particles have several realistic properties of fluids; for example, when poured into a cup-like container, an object placed in the filled container will either float or sink depending on its weight. The task in Figure 3 has water flowing down a system of angled brick walls. The bottom angled wall can be moved around a pivot in its center, so that the flow of water can be directed to the left or right. If it is directed to the right, the water will flow further down until it encounters an orange block. If the orange block is moved, the water will eventually flow into a pit which contains an apple, which the agent could previously see but not reach. The flow of water will eventually fill up the pit, causing the apple to float up to where the agent can reach it.

4.1.2 The Piagetian Balance-Beam Task

Some tasks that might be considered Piaget-MacGyver Rooms can come directly from classical Piagetian experiments. Inhelder and Piaget's (1958) Balance-Beam Task (BBT) [19] has been modeled many times using a variety of modeling techniques [41, 30, 32, 31, 38, 23]. In the BBT, a balancing beam with a set of weights are provided to a subject. The balancing beam has notches, hooks, or some other apparatus that allows the weights to be placed on the left or right sides of the balancing beam at predefined intervals. In most versions of the task, the values of the weights and the distances that the locations are from the center are made available to the subject. The task is normally to figure out some version of the torque rule, which relates the product of the value of a weight and its distance from the center. For example, the subject may be presented with a configuration of weights on the scale, and the subject is asked to predict whether the right or left side will tilt downwards or the scale will balance.

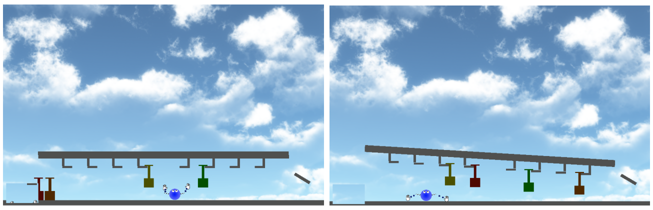

Figure 4: The Piagetian Balance Beam Task in PAGI World

We recreate the BBT in PAGI World as in Figure 4. A balance beam with several hooks on which weights can be hung is in the center of the screen. There is also a switch (on the bottom right side of the screen) that can be used to toggle the motion of the balance beam. That way, the weights can be hung on the balance beam and the beam will not move while the agent is reasoning about how the beam should tilt when the switch is toggled. This BBT is fully implemented and included in PAGI World (although no AI capable of solving this task has yet been developed).

One clear limitation of computationally modeling most Piagetian tasks is that you can't really communicate with the AI agent in natural language like you can with the children in Piagetian experiments. Although the current state of the art in natural-language processing and generation prohibits such communication at present, PAGI World offers tools to make it easier for researchers who are trying to achieve this benchmark. There is a way to "talk to" the AI agent through an input text box in PAGI World itself. Having the agent talk back, however, can be handled in three possible ways: through simple output handled completely by the code on the AI side, by sending a message to PAGI World that can be displayed in an output window (a console screen accessible through PAGI World), or by creating a speech bubble.

Speech bubbles can be created (Figure 5) by AI-side scripts. These speech bubbles are recognizable by PAGI guy's vision system, along with data such as the name of the speaker, the location of the box, the text written inside of it, and so on.

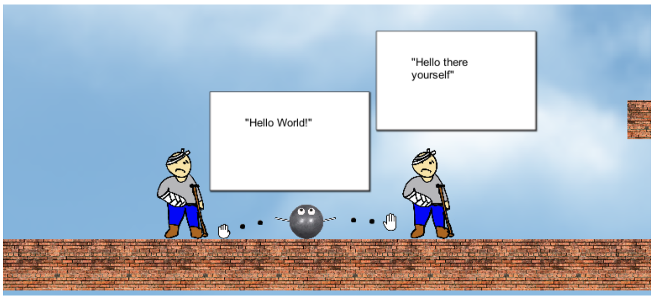

Figure 5: Speech bubbles can be created to simulate conversations, giving them a visual element that makes for understandable demonstrations.

4.2 Moral Reasoning

4.2.1 Resolving a Moral Dilemma

One interesting and possibly fertile source of PAGI-World tasks is the area of morality, specifically machine ethics [40], which is devoted to trying to engineer machines that have at least a degree of moral sensibility. Figure 6 depicts an example task in which two injured soldiers are equally distant from PAGI guy, and only a single health kit is available. PAGI World allows the visual sensors to detect whether a soldier is injured or healthy. If a health kit touches an injured soldier, the soldier will become healthy and the health kit will disappear.

Some might recognize this task as a variant of the *Buridan's Donkey* scenario, where a donkey unable to choose between two bales of hay, paralyzed by indecision, ends up starving to death. This task can be used, for example, to evaluate whether a reasoning system is sufficiently intelligent to avoid certain paralyzing situations, or to search for creative solutions where they exist. PAGI World provides two health kits: a small one, and a large one. The large health pack (as in Figure 6) can be broken in half, when enough force is applied to it, and each half can be given to one of the soldiers. We recently used this task to demonstrate that the search for creative solutions, and the ultimate decision to let one soldier remain injured if such a solution can not be found, can be carried out entirely within the framework provided by the Deontic Cognitive Event Calculus [5].

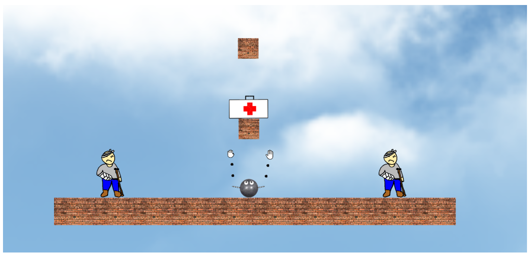

Figure 6: A task where an agent only has one health kit but two injured soldiers.

4.2.2 Reasoning Over Obligations to Detect Hijacking

Bello et al. in [1], use PAGI World to model a situation in which a robot capable of autonomously reasoning over moral obligations (represented by PAGI guy) is maliciously attacked and infected with a virus. The virus hijacks the robot's actuators, instructing them to perform the action of pushing a healthy soldier over a ledge, resulting into the soldier's falling into a lava pit below. By using the reasoning mechanisms provided by the Deontic Cognitive Event Calculus, Bello et al. are able to show that the robot can reason over its obligations and its apparent desire to perform the harmful action to the soldier, ultimately concluding that such an action is not morally permissible. Presumably, further reasoning might hypothesize the existence of the hijacking virus.

4.3 Low-Level Reasoning

4.3.1 Learning to Catch and to Avoid Painful Objects

The reflex and state system (Section 3.3) allows for *reflexes* to be created and configured from the AI side and stored on the PAGI side, where they can be activated automatically based on sensory values. Whenever such a reflex is activated, a message can optionally be sent to the AI side, naming the reflex which was just fired. This can allow for after-the-fact reasoning of automatic reflexes. In one example, a set of reflexes are configured so that the agent will quickly retract its hand after touching a hot object (temperature sensors are among the sensors lining the circumference of both hands). If the agent touches an object with a high temperature (the flames and the hot plates, in this example), the reflex will first fire and then the AI side can decide if further actions should be taken, for example by looking at the object which caused the reflex to fire, thus giving the agent a motivation to avoid going near hot objects in the future.

PAGI World also allows for objects in the simulation to have an objective utility value. These are called *endorphins*; objects with a positive endorphin value are those that the AI agent may want to pursue (e.g. food items), whereas negative endorphin values are those the agent should avoid. Most objects in PAGI World have an endorphin value of zero, and PAGI World itself does not ensure that certain endorphin-seeking behaviors are implemented by the agent.

Another example of the reflex system is catching an object. In the baseball-catching task (available online), a baseball is launched to soar right above the agent's head, and he has a time period of a little over a second in which his hands can reach the ball without moving his body. Of course, the agent can optionally move his body as well to reach for the ball if necessary. If the ball is caught, the agent will receive an endorphin bonus; otherwise, the ball will fall off of the ledge and no longer be accessible to the agent. Although TCP/IP communication is relatively quick, the microsecond timing required to estimate the speed and trajectory of the ball and move the hand in time to catch it is not easily (and probably not possibly) done with controls firmly on the AI side. Instead, reflexes must be used.

In addition to sensor values, reflexes can be configured to activate, deactivate, or check for *states*, which are simply string labels configured by the AI side. Because reflexes can optionally be set to fire only if certain states are active, followed by deactivation of current states and activation of new ones, reflexes can be used as transitions between states in a deterministic finite-automata machine. In addition to receiving notifications whenever such a reflex fires, the AI-side can poll the PAGI side to see which states and reflexes are currently active, so that it can change them if necessary.

4.4 High-Level Reasoning

4.4.1 Self-Awareness: Reasoning About the Self

Whether artificial agents can ever have self-awareness is a highly controversial topic; this is clear from the public discussion surrounding popular press reporting a recent experiment in robot self-awareness. Bringsjord et al. (2015) started with a puzzle devised by philosopher Luciano Floridi [15], in which three artificial agents, or robots, are given either a "dumbing" pill or a placebo. The dumbing pill disables their higher-level cognition (i.e. their ability to reason), and is given to two of the three robots. All three robots are then asked which pill they received. The two given the dumbing pill cannot reason, and thus remain silent (note their silence is not by choice, but because they fail to reason at all). The robot given the placebo ultimately concludes it cannot decide, whereupon it utters the phrase "I don't know." However, upon hearing (and, importantly, feeling) itself utter the phrase, it now has a new piece of knowledge with which to reason. Given this new piece of knowledge, along with an understanding of the rules of the current experiment (knowledge which is also initially given to the other two robots), it can then conclude that it was in fact, not given the dumbing pill.

PAGI World was used to create Floridi's task [8]. We created a task containing three agents[5], and on the AI-side, the agents were connected to an automated theorem prover reasoning in the Deontic Cognitive Event Calculus [5]. This calculus is a knowledge representation framework capable of expressing *de se* beliefs, i.e. a specific type of reasoning about the self [4]. Pills are given to each agent, implemented as endorphin-producing items. Placebos produce negative endorphins, and the dumbing pills produce positive endorphins. When a pill is absorbed by an agent, the endorphin value is sent to the AI-side, where the agent's reasoning system is either disabled or left untouched.

The experiment can then proceed. Commands "spoken" to the agents are typed in to a text box that PAGI World makes available (strings entered into this box are sent as messages to the AI-side). Statements the agents wish to output can be sent from the AI-side to PAGI-side as a special type of command, where they will be displayed on a debug screen.

It is important to here introduce a disclaimer: we do not claim such an experiment "proves" self-awareness in our artificial agent. Rather, this is just another example of the psychometric philosophy underlying PAGI World, according to which a series of psychometric tests are proposed (in this case, the tests given by Floridi (2005)), undertaken by AI researchers, and the cycle repeats. PAGI World makes the creation of such tests easier.

4.4.2 Analogico-Deductive Reasoning

Because tasks can be created dynamically through special commands sent from the AI-side, saved to file, and loaded, it is possible to train PAGI guy on tasks, then change the task and see how the previous training transfers. Such a setup is ideal in the testing of transfer learning, e.g. in instances of analogico-deductive reasoning (ADR). In ADR, an agent uses analogical reasoning to generate a hypothesis about some target domain (where the target domain is often unfamiliar in some way). The hypothesis is then subjected to deductive reasoning, so that it can either be proven false, or possibly shown to imply some method of experimental verification [23].

In [24], Marton et al. used PAGI World to demonstrate ADR in a real-time task. The agent was given the ability to assess its surroundings, and to solve a source task. In the source task, PAGI guy had to reach an apple by jumping over an obstacle (a raised brick). It was then faced with a different task, in which the obstacle was instead a gap in the ground. The AI-side was able to generate a hypothesis that the solution used in the source task, to jump over the obstacle, would work in the target task as well. It then carries out a proof using a deductive reasoner, and upon receiving confirmation from the prover, jumps over the gap to reach the apple. For more details, see [24].

[5] The ability to create multiple agents in PAGI World is a beta feature that is not available in the publicly available version. We expect this feature to be ready soon.

5 Conclusion and Future Work

The release of PAGI World is accompanied with a call to all AGI and human-level-AI researchers to finally examine the strengths and limits of their preferred approaches. PAGI World allows for researchers to very easily create tasks and microworlds in a 2D world with realistic physics, with no knowledge in how to program. PAGI World can interact with AI agents that are written in virtually any programming language, and the simulation can be run on any major operating system. We have very carefully designed PAGI World to have an extremely low technical barrier, so that many researchers can find common ground upon which to compare their different approaches.

PAGI World can also be used in educational applications. In the Spring of 2015, the present authors taught a course at RPI, which made use of PAGI World for homework assignments, projects, and to give students hands-on experience in implementing AI and cognitive modeling techniques. The experiment was quite a success, and we are currently further exploring PAGI World's possibilities for education.

The future of PAGI World is bright. We already have several AI systems in progress whose goals are to solve already-finished PAGI World tasks, and as development continues we hope to greatly increase the number of tasks which are available and the sophistication of the agents which solve those tasks. The library of future tasks, we hope, will diversify and reflect the broad spectrum of tasks which require human-like intelligence.

PAGI World downloads, documentation, example tasks, and source code can be downloaded from the RAIR Lab website at `http://rair.cogsci.rpi.edu/projects/pagi-world`.[6]

REFERENCES

[1] Paul Bello, John Licato, and Selmer Bringsjord, 'Constraints on Freely Chosen Action for Moral Robots: Consciousness and Control', in *Proceedings of RO-MAN 2015*, (2015).

[2] Mark H. Bickhard, 'The Interactivist Model', *Synthese*, **166**(3), 547–591, (July 2008).

[3] Selmer Bringsjord, 'Psychometric Artificial Intelligence', *Journal of Experimental and Theoretical Artificial Intelligence*, **23**(3), 271–277, (2011).

[4] Selmer Bringsjord and Naveen S. Govindarajulu, 'Toward a Modern Geography of Minds, Machines, and Math', *Philosophy and Theory of Artificial Intelligence*, **5**, 151–165, (2013).

[5] Selmer Bringsjord, Naveen S. Govindarajulu, Simon Ellis, Evan McCarty, and John Licato, 'Nuclear Deterrence and the Logic of Deliberative Mindreading', *Cognitive Systems Research*, **28**, 20–43, (2014).

[6] Selmer Bringsjord and John Licato, 'Psychometric Artificial General Intelligence: The Piaget-MacGyver Room', in *Theoretical Foundations of Artificial General Intelligence*, eds., Pei Wang and Ben Goertzel, 25–48, Atlantis Press, 8, square des Bouleaux, 75019 Paris, France, (2012).

[7] Selmer Bringsjord, John Licato, and Alexander Bringsjord, 'The Contemporary Craft of Creating Characters Meets Today's Cognitive Architectures: A Case Study in Expressivity', in *Integrating Cognitive Architectures into Virtual Character Design*, eds., Jeremy Turner, Michael

[6] Special thanks to a series of brilliant undergraduate students who contributed to earlier versions of PAGI World: Jack Cusick, Robert Dong, Kainoa Eastlack, Nick Marton, and Spencer Whitehead. This work was supported in part by ONR grant N000141310342, and Licato's AFOSR Young Investigator Research Program grant.

Nixon, Ulysses Bernardet, and Steve DiPaola, Advances in Computational Intelligence and Robotics (ACIR), Information Science Reference / IGI Global, (Forthcoming).
[8] Selmer Bringsjord, John Licato, Naveen S. Govindarajulu, Rikhiya Ghosh, and Atriya Sen, 'Real Robots that Pass Human Tests of Self-Consciousness', in *Proceedings of RO-MAN 2015*, (2015).
[9] Selmer Bringsjord and Bettina Schimanski, 'What is Artificial Intelligence? Psychometric AI as an Answer', in *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI)*, (2003).
[10] Rodney A. Brooks, 'Elephants Don't Play Chess', *Robotics and Autonomous Systems*, **6**, 3–15, (1990).
[11] Rodney A. Brooks, 'Intelligence Without Representation', *Artificial Intelligence*, **47**, 139–159, (1991).
[12] David Bruce, 'Cognitive Architecture and Testbed for a Developmental AI System'. Undergraduate Honours Thesis, 2010.
[13] Harold H. Chaput, Benjamin Kuipers, and Risto Miikkulainen, 'Constructivist Learning: A Neural Implementation of the Schema Mechanism', in *Proceedings of the Workshop on Self Organizing Maps (WSOM03)*. FCFM, (2003).
[14] Gary L. Drescher, *Made-Up Minds: A Constructivist Approach to Artificial Intelligence*, The MIT Press, 1991.
[15] Luciano Floridi, 'Consciousness, Agents and the Knowledge Game', *Minds and Machines*, **15**(3-4), 415–444, (2005).
[16] Dedre Gentner and Kenneth Forbus, 'Computational Models of Analogy', *Wiley Interdisciplinary Reviews: Cognitive Science*, **2**(3), 266–276, (2011).
[17] Frank Guerin, 'Learning Like a Baby: A Survey of Artificial Intelligence Approaches', *The Knowledge Engineering Review*, **26**(2), 209–236, (2011).
[18] S. Harnad, 'The Symbol Grounding Problem', *Physica D*, **42**, 335–346, (1990).
[19] Barbel Inhelder and Jean Piaget, *The Growth of Logical Thinking: From Childhood to Adolescence*, Basic Books, Inc, 1958.
[20] Daniel Kahneman, *Thinking, Fast and Slow*, Farrar, Straus and Girous, 2011.
[21] John Licato, *Analogical Constructivism: The Emergence of Reasoning Through Analogy and Action Schemas*, Ph.D. dissertation, Rensselaer Polytechnic Institute, Troy, NY, May 2015.
[22] John Licato, Selmer Bringsjord, and Naveen S. Govindarajulu, 'How Models of Creativity and Analogy Need to Answer the Tailorability Concern', in *Proceedings of the IJCAI 2013 Workshop on Computational Creativity, Concept Invention, and General Intelligence*, eds., Tarek Richard Besold, Kai-uwe Kühnberger, Marco Schorlemmer, and Alan Smaill, Beijing, China, (2013).
[23] John Licato, Selmer Bringsjord, and John E. Hummel, 'Exploring the Role of Analogico-Deductive Reasoning in the Balance-Beam Task', in *Rethinking Cognitive Development: Proceedings of the 42nd Annual Meeting of the Jean Piaget Society*, Toronto, Canada, (2012).
[24] Nick Marton, John Licato, and Selmer Bringsjord, 'Creating and Reasoning Over Scene Descriptions in a Physically Realistic Simulation', in *Proceedings of the 2015 Spring Simulation Multi-Conference*, (2015).
[25] Jean Piaget and Rolando Garcia, *Understanding Causality*, W.W. Norton and Company, Inc., New York, New York, USA, 1974.
[26] Jean Piaget and Barbel Inhelder, *The Growth of Logical Thinking*, Basic Books, Inc, New York, New York, USA, 1958.
[27] Jean Piaget, Jacques Montangero, and J.B. Billeter, *Studies in Reflecting Abstraction*, Psychology Press, 1976/2001.
[28] Jean Piaget, Jacques Montangero, and J.B. Billeter, 'The Formation of Analogies', in *Studies in Reflecting Abstraction*, ed., Robert Campbell, Psychology Press, (2001).
[29] Steven R Quartz and Terrence J. Sejnowski, 'The Neural Basis of Cognitive Development: A Constructivist Manifesto', *Behavioral and Brain Sciences*, **20**, 537–596, (1997).
[30] S. Sage and Pat Langley, 'Modeling Cognitive Development on the Balance Scale Task', in *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, ed., A. Bundy, volume 1, pp. 94–96, (1983).
[31] William C. Schmidt and C. X. Ling, 'A Decision-Tree Model of Balance Scale Development', *Machine Learning*, **24**, 203–230, (1996).
[32] Thomas R. Shultz, Denis Mareschal, and William C. Schmidt, 'Modeling Cognitive Development on Balance Scale Phenomena', *Machine Learning*, **16**, 57–86, (1994).
[33] Peter Smith, *An Introduction to Gödel's Theorems*, Cambridge University Press, Cambridge, UK, 2007.
[34] Ron Sun, 'From Implicit Skills to Explicit Knowledge: A Bottom-Up Model of Skill Learning', *Cognitive Science*, **25**(2), 203–244, (2001).
[35] Ron Sun, *Duality of the Mind: A Bottom Up Approach Toward Cognition*, Lawrence Erlbaum Associates, Mahwah, NJ, 2002.
[36] Ron Sun. Desiderata for Cognitive Architectures. Website, September 2003.
[37] Ron Sun, 'Interpreting Psychological Notions: A Dual-Process Computational Theory', *Journal of Applied Research in Memory and Cognition*, **In Press**, (2014).
[38] Geoffrey F. W. Turner and Hoben Thomas, 'Bridging the Gap between Theory and Model: A Reflection on the Balance Scale Task', *Journal of Experimental Child Psychology*, **81**, 466–481, (2002).
[39] Ernst von Glasersfeld, 'Abstraction, Re-Presentation, and Reflection: An Interpretation of Experience and of Piaget's Approach', in *Epistemological Foundations of Mathematical Experience*, ed., L. P. Steffe, 45–67, Springer, New York, New York, USA, (1991).
[40] Wendell Wallach and Colin Allen, *Moral Machines: Teaching Robots Right From Wrong*, Oxford University Press, Oxford, UK, 2008.
[41] Friedrich Wilkening and Norman H. Anderson, 'Comparison of Two Rule-Assessment Methodologies for Studying Cognitive Development and Knowledge Structure', *Psychological Bulletin*, **92**(1), 215–237, (1982).