The Mind of the Noble Ape in Three Simulations

Tom Barbalet

Abstract The Noble Ape Simulation offers an account of the mind as something that can be observed, measured, and ultimately simulated through external effects. This version of the applied mind is not created through a single method but through layering three simulations relating to information chemistry, social constraints, and evolving narrative. As examples, additional simulation elements in Noble Ape are presented to offer the simulation methodology of Noble Ape. This chapter, rather than being a theoretical critique, is intended as a project report relating to three distinct yet interoperating simulated models of the mind. These are presented both as individual simulations and also the simulations' interactions. This produces a novel account of the applied mind. The methods used in creating such an applied mind provide an interesting insight into the possible origin of mind through pragmatic application rather than conjecture.

Keywords Artificial life • Simulation • Theories of mind • Robotics • Social robotics • Cognitive science • Cognitive simulation • Intelligent agents • Open source • Linguistics • Computational linguistics • Philosophy of mind • Philosophy of language

1 Background

I started Noble Ape at 19 years of age in 1996 in Australia. As Noble Ape is open source, there have been numerous contributing developers including engineers from Apple, Intel, and Cern. A substantial component of the Noble Ape Simulation discussed in this chapter has come from Bob Mottram, an industrial roboticist based in the United Kingdom. This chapter would not be possible without the dedicated

T. Barbalet (✉)
Noble Ape, Campbell, CA, USA
e-mail: tom@nobleape.com

L. Swan (ed.), *Origins of Mind*, Biosemiotics 8, DOI 10.1007/978-94-007-5419-5_20,

work Mottram has donated to Noble Ape. Through the work effort described in this chapter, Mottram worked remotely and often contributed source code independently. This is one of the charming idiosyncrasies of open source development. Multiple participants can work on the same piece of software for distinct purposes at the same time with minimal communication.

Noble Ape can be thought of as a series of different simulations:

- A landscape simulation that creates a large environment
- A biological simulation that models the underlying biology in the environment
- A weather simulation that creates the meteorological aspects of the environment
- Three independent but intertwined agent simulations:
  - A cognitive simulation
  - A social simulation
  - A narrative engine

The latter three are the primary topic of this chapter. The weather and biological simulations will also be discussed as they offer a connection with the cognitive simulation and the account of the broader methodological perspective of the project.

2 Artificial Life

Noble Ape is considered an artificial life project. Artificial life does not have an exact disciplinary definition. It covers a variety of different kinds of software, hardware, and chemistry that look to show *life as it could be* (Langton 1997). Artificial life is an idea that predates computation and exists in its most basic form in thought experiments about life – speculative life, if you will. Concepts in artificial life can be found as early as Hobbes' Leviathan (1651).

Computation has moved the field from thought experiments into a variety of different approaches including evolutionary computing, intelligent agents, genetic algorithms, applied genetic programming, and cellular automata. The field was broadly defined through a number of popular surveyings (Emmeche 1991; Levy 1992) and authors who developed their own early artificial life simulations (Dawkins 1987).

While the early artificial life simulations were relatively simple and similar to other kinds of software, artificial life software that has been in development for more than a decade is in a comparably advanced state. Modern computing, specifically continuous development adapting to modern multi-core processors, has advanced the capabilities of artificial life software. Noble Ape has been able to give back into this cycle too through its use by Apple and Intel to optimize processor power (Barbalet 2009).

It is important to recognize this chapter in this light. The work presented here relates to software that can be obtained both in source and executable form free of charge for additional scrutiny. The descriptions offered here are not speculative but relate to software that, although it may appear whimsical, has been of great practical benefit.

3 Motivations

Noble Ape was created with a basic hope: through sweat equity it would be possible to create a philosophically rich simulation of the mind. The problem was divided into two parts. An environment needed to be created that would have the depth and interest for these simulated minds to flourish. Also, perhaps far more difficult, the simulated minds would need to show a degree of tenacity to be a compelling representation of real-world cognitive dilemmas.

At the time of initial development, I knew of no peers in this kind of project. I later learned about the work of Larry Yaeger with Polyworld (1994). The distinction between Noble Ape and Polyworld was that Noble Ape did not have a neural network as the intelligence in silico. Initially Noble Ape relied on the cognitive simulation described in this chapter.

The early development of Noble Ape was a youthful opposition to dominant and failing ideas that went against my own experience. As a student of philosophy, I was frequently told that computer simulations offered no insight into the mind. I was presented with straw arguments relating to buggy software and failed robot experiments far from the work I read about at MIT in a similar time frame (Kirsh 1991).

The misguided view of software intelligence as failing and sub-utilitarian was in stark contrast to my own experiences in creating software (Barbalet 1997a). In my early teens, I developed computer games with compelling simulated comrades and enemies. Through my late teens, I wrote heuristic antiviral software that detected both known computer viruses and also predicted computer viruses from heuristic analysis of known symptoms and projected symptoms. Prior to Noble Ape, while I wrote antiviral software, I also wrote compiler software (software that took English language readable code and translated it into machine code) that was based on some of the dynamic and adaptive methods I saw used in computer viruses. My compiler software was intentionally non-malicious as it related to transforming abstract information without the infrastructure to be transmitted from machine to machine. These compiled models of adaptive intelligence seem distant from the poor accounts of software intelligence I was provided with in my philosophy studies.

My choice of study in mathematics, physics, and philosophy was a primary indication of my general level of distain for computer science with flawed neural networks and obsessive historical self-induced paradoxes (similar to the philosophy I found). As the early Noble Ape development showed (Barbalet 1997b), I was fixated on finding solutions to the origin of mind and a means of simulating the mind. Computer science and, as I found through my studies, philosophy were not going to provide the answers or even the direction for this insight.

I felt very strongly that trying to find a biological mirror of the mind in software failed to identify how little was known about the relating biology. In fact these attempts to simulate the mind through biologically inspired neural networks appeared to confirm the skeptical philosophical view that was omnipresent in my philosophy education. The early development of Noble Ape, in particular the biological simulation and the cognitive simulation, were intentionally developed in stark contrast to the failed but commonly accepted means of simulating both biology through representative biological models in software and attempts to simulate the mind through neural networks.

The energy and anger of youth tends to taper. The practical nature of maintaining a development like Noble Ape required progressive compromise. It is important to note the development moved from being distinctly radical to relatively mainstream not through a movement in the project but through a movement of the thinking on simulations that contained intelligent agents.

Part of the normalization of Noble Ape came through its utilitarian use. Within 7 years of the project starting, it was embraced by a generation of engineers at Apple and 2 years following another group of engineers at Intel (Barbalet 2005a). Through this period, limited additional work could be done on the simulation. As the primary maintainer, roughly 5 years were spent updating the project to the changes required by the Apple and Intel engineers.

It was this normative maintenance culture that appealed to Bob Mottram. The cognitive simulation (unique and original to the project), the social simulation (based in social robotics), and the narrative engine (based in early artificial life simulation) are combined in the Noble Ape development. This combination of simulations within a unified project represents my pluralist and utilitarian philosophical views on the origin of mind. The project also identifies the only productive way these models can be used is in concert: not in contrast or competition. It is also important to note that the latter two contributions for the simulations of mind in Noble Ape probably would not have been accepted in the early history of the development.

Moreover, it is perfectly feasible that additional simulations will be added through the continued developmental history of Noble Ape. It is also quite possible that the simulation models used could be unified. This should provide further philosophical insight as the method used to reduce these simulations should also provide finer conceptual structure to the origin of mind.

4 Biological Simulation

The biological simulation was the first new software developed for Noble Ape. Noble Ape was created rapidly as it was primarily a combination of existing projects I created. The landscape and visualization came from earlier landscape graphics environments I created (Barbalet 2004), and the cognitive simulation came from earlier agar (petri dish) simulations I created. The early development was undertaken on first-generation personal computing (PC-XT and PC-AT computers and 68000 Macintosh computers). For the scale of landscape being simulated, even macro population simulation (Volterra 1931) would have been too computationally intensive.

At the time, I was studying physics. It appeared the easiest way to minimize the exertion of processing power was to model the biological simulation on quantum mechanics (computationally, if not conceptually). The use of quantum mechanics in the biological simulation can be explained relatively simply. Take a point on the landscape and perform a summation of probabilities. These probabilities can be offered in a thought experiment. What would it take for a particular biological species to exist at that point? The landscape is a wave function. It is a continuous two-dimensional planar function. There are various properties of the landscape. The landscape at a particular point has an area associated with it. It has a height above some arbitrary level – a height above sea level, for example. It has a water value associated with it that relates to its proximity to saltwater or freshwater. There is a moving sunlight operator that represents how the simulated sun is hitting the point at a particular time. There's a total sunlight operator that is taken over all time. Also a salt operator that represents salt water or ground salt.

The height is the underlying quantum mechanics wave function, and these operators (height above sea level, area, moving sunlight, total sunlight, water, and salt) are applied to the wave function to give a value.

At any given point, there is a probability density that something will be there. This only becomes actuality when a noise map is put on the probability density. This cuts the probability density and shows where something actually is rather than a probability of its being there. Rather than creating a huge biological system including every part and a wide variety of other interactions, the biological simulation just interrogates the environment at a particular place and calculates the operators that are applicable. If the Noble Apes are foraging for food, the simulation can get the various operators that converge on whether the Noble Apes are interested in berries or whatever food is available and can interrogate the environment directly rather than having a large biological simulation.

Using a plant as an example, consider the surface area needed. Surface area is a point relative term based on a flat plane having little surface area, approaching a near infinite surface area as the landscape moves to a cliff. A tree can't grow well on a cliff, so the surface area has certain importance. There are various plants that thrive at particular heights. Water is also an important factor. Moving sunlight is less important, but total sunlight is critical, and depending on whether the plant likes or dislikes salt is a factor too. Insects may dislike being in direct sunlight so the moving sunlight indicates where some insects may not want to be.

A noise map is used to intersect the probability function coming from the operators acting on the wave function. The change on the noise map depends on whether the biology is a plant or an animal. If it is a plant, there needs to be reproducibility at a specific point, whereas if it is an animal, it needs to change over time. The plant noise map is static, whereas the animal noise map has periodic transitions.

The biological simulation provides a good example of the pragmatism that has been a defining factor in the creation of simulations for Noble Ape. A specific need for great detail and a limitation of processing power created a biological simulation that may not express all the components for a detailed biological understanding but produces enough biodiversity to provide a detailed simulation environment and simulated diet for the Noble Apes.

5 Weather Simulation

Added to Noble Ape in 2000, the weather simulation can be summarized as a water vapor simulation with a hard ceiling. The water vapor moves over the landscape. As there is increased pressure, the clouds form and rainfall occurs at the highest pressure points. The weather simulation is calculated at half the resolution of the landscape. This is due to the time to calculate the underlying weather. This calculation was heavily optimized to make it as fast as possible.

The weather simulation is less scalable than the biological simulation. It not only has been maintained through the functional purpose of providing accurate and diverse weather conditions to the simulation inhabitants but it also closely resembles the initial two-dimensional cognitive simulation. The weather simulation still has shared mathematics with the three-dimensional cognitive simulation.

There is a somewhat tongue-in-cheek grand unified simulation theory that the weather simulation and the cognitive simulation could have greater shared mathematical elements. The cognitive simulation was the subject of substantial optimization by engineers at both Apple and Intel for their respective processing hardware (Barbalet 2009). If it was possible to find connective mathematical elements, and have these elements optimized through modern processing hardware, both the weather and the cognitive simulations would see substantial speed improvements.

6 The Cognitive Simulation

The cognitive simulation predates a majority of the development of Noble Ape. It comes from my early simulation of agar (petri dish) bacterial growth. Through developing these simulations, I came to the idea that bacterial growth could represent information transfer. As the bacteria grew through the agar, the movement into corresponding cells was similar to information being transferred to the surrounding cells (Barbalet 2009). The mathematics for bacterial growth in agar and the final mathematics for the Noble Ape cognitive simulation were quite different, but they had mathematical similarities. Both were represented by competing equations: one associated with the movement through space and one associated with the movement through time. In the cognitive simulation, these two competing equations were labeled *desire* in terms of the traversing through space and *fear* for reacting through time (Barbalet 1997b). The original cognitive simulation was a two-dimensional simulation in a 128×128 cell space. The sensors (that pushed sense information into the cognitive simulation) were at one end, and the actuators (that took information from the simulation to produce action) were at the other end. The sensors' noise and excitement would ripple through to the actuators through the agar-like substrate accordingly.

In the two-dimensional simulation, the information flow has characteristics that were very similar to those of the weather simulation; however, it had a strong bias in linear movement providing just a single dimension of information transfer.

I moved to a three-dimensional model with the same underlying mathematics in a smaller area ($32 \times 32 \times 32$ cells). This added the ability for information to transfer in all three dimensions rather than the scanning two dimensions that ultimately led to a single productive dimension that related to the time transfer of the information.

In the current version of the cognitive simulation, Mottram changed the code slightly, so the sensors and actuators are once again equidistantly spaced. The addition of a third dimension gives a fixed processing length and an ability for the information to intermingle.

What the cognitive simulation presents is a description of the mind in a pre-language and pre-social state. It is the idea of the mind as survival organ. The mind must guide the agent to food and away from danger. Society, as it is represented in such a mind, is purely a fear negator and potentially also the guide toward feeding and procreating areas. The cognitive simulation provides a primitive survival model of the mind.

The cognitive simulation describes not only the process but also the information vessel where there are sensors and actuators that are passing information through the vessel. The properties of the vessel explain how the information is retarded and propagated. The sensors are firing information, and actuators are reacting to this information. The space between the sensors and actuators in the vessel is the mathematical space described by fear and desire. Conceptually the vessel description of the cognitive simulation has only one flaw. The space of the cognitive simulation wraps around. The x-axis wraps into itself as does the y-axis and the z-axis of the simulation space. This provides an additional property where the nearest sensor and actuator connection may be through the axis origin. The contribution of sensor information into the cognitive simulation may be maintained through multiple traverses through the cognitive simulation space. These rippling waves of information transfer are negated through both desire properties and also the ability for sensors to provide strobing feedback that can stabilize returning information signals.

Desire is reinforcing for actuator responses. Rather than reacting violently to information that is being put through the sensors, desire reinforces this information, slightly retarding it through the spatial mathematics it employs. The agent does not react so fearfully. In contrast, fear amplifies the sensor signals and causes more reactive movement when this information is received by the actuators. Both fear and desire coexist in the cognitive simulation to counterbalance these competing properties.

The cognitive simulation size for the Noble Apes has remained the same since it moved to three dimensions. Those size constraints should be expanded for some interesting effects. With the additional simulations of the mind described in this chapter, in particular the narrative engine, a $64 \times 64 \times 64$ cell to even a $256 \times 256 \times 256$ cell cognitive simulation would greatly benefit the broader agent model.

There are a number of other species that exist in the Noble Ape environment. The Noble Apes have a primary role because they are sentient human-like creatures. There are felines, birds, and smaller mammals. These species would benefit from having simple cognitive simulations that are similar to the Noble Apes. The weighting between fear and desire as well as the size of the cognitive simulation could be altered. Consider a feline having a cognitive simulation of $8 \times 8 \times 8$ cells. Rather than having a heavy fear weighting to the cognitive simulation, the simulated feline would have a stronger weighting to desire as they are the primary predator in the environment. They have little need for fear and are more governed by their general desires.

7 Social Simulation

Noble Apes with just the cognitive simulation were not particularly social. They were a reactive and fearful group of simulated agents. Mottram came to the Noble Ape Simulation with a background in social robotics, in particular a strong interest in the work of Cynthia Breazeal at MIT (2002). Mottram's initial feedback having reviewed the simulation was that there needed to be a set of social factors and constraints hardcoded into the simulation.

Mottram saw grooming as an important primate social behavior that was absent from the Noble Apes. He set about implementing something comparable to grooming as he realized that grooming served both a utilitarian function (the removal of parasites) and also a psychological function (of determining and reinforcing status and bonding relationships between individuals). In keeping with the theme of nobility, Mottram added an *honor* value that was indicated of the social status of each individual in the group. He also added a value indicating the number of parasites carried by each Noble Ape together with a simple mathematical model of parasite reproduction, energy cost to the Noble Ape, and their transmissibility between Apes.

Mottram hardcoded interactions that would create a simple economy based on social status. When one Noble Ape was groomed by another, they spent some of their honor value, while the groomer acquired a corresponding amount of honor for performing the service of removing parasites (and hence reducing energy depletion). The honor value might then be later used to bias mating decisions. Mottram also started adapting some of the genetic aspects of the simulation and created ideas of families, social groups, and clans.

Although in the initial implementation of this grooming-based economy of status, Noble Apes were not explicitly aware of their own honor value or that of others; the later addition of the narrative engine permitted them to become aware of this factor.

If the Noble Apes had a self-aware notion of their own honor, then it would change their interactions and the simulation would digress into an honor optimization algorithm. Honor was heavily muted in things that the Noble Apes could access. Primarily it just had unexplained effects when, for example, they were meeting other Noble Apes or they were squabbling. This simulated honor contained elements of luck based on probabilistic outcomes.

Mottram also explicitly hardcoded for social drives (Breazeal 2002). The hunger drive represented a biological quantity but also represented an interaction with food.

The social drive represented an interaction with other entities and had various feedbacks associated with social interaction. The fatigue drive related to tiredness, an overabundance of swimming, and a variety of other factors. The sex drive also contained elements of social interaction and genetic predetermined preferences. These drives, like honor, were represented as a single variable each.

8 Social Graph

In addition to social variables, Mottram and I worked together to produce a social graph. The social graph described a spatial map where the relationship of each Noble Ape in space is represented by their social connections and time is represented in simulated time. The social graph could be considered another simulation in and of itself. It is foreseeable in the future development that the social graph becomes a fully independent simulation.

The social graph interaction produced a very rich graphical view of Noble Ape society. Social groups of Noble Apes appear in cloud-like formations through the social graph. Each individual Noble Ape only has a social group of six other Noble Apes. Although six others may seem extremely small, the larger families and genetic groups maintain hardcoded connections. The Noble Ape will be able to implicitly recognize kin, but it may not have the same memory of this hardcoded kin as it has of an individual in its social group memory. The six Noble Apes in the social group memory of each Ape magnified over the population total produce a rich social environment that is represented as a rich graphical environment.

This graphical view illustrates dramatically the friends and enemies of each Noble Ape. Moreover, conditions of social ejection are shown graphically. Some conditions of Noble Ape squabbling eject one or two Noble Apes out of a family or clan group. There are choices the other Apes need to make about whether or not they want to interact socially with the socially ejected party.

The social graph tracks a variety of smaller things, but it can be used in a spatial graph setting as well. The difficulty in understanding simulations like Noble Ape is that they are just so rich. Vast numbers of interactions occur. Any additional abstraction that can convey meaning is greatly beneficial. The social graph provides this ability to see an aspect of the simulation that would have been very difficult to do through observing the simulation over time and interpolating through the information presented.

The social graph highlighted two properties of the social simulation that had been observed through simulation space interactions, but the profound effect on Noble Ape society was not properly understood until the social graph identified them explicitly.

The first property highlighted was that social relations can be asymmetric. This is identified in the social graph interaction where Noble Apes make mistakes. Information is forgotten by certain Apes at a faster rate and remembered by others. There are bitter Noble Apes who have had negative interactions that they haven't forgotten. Other Noble Apes forget these interactions and get on with their foraging. There is also implicit confusion the way the family groups are described. Some of the Noble Apes think that certain other Apes are in one family group, and some of the Apes think they're in another family group due to implicit mistakes in group meetings and information presented to the Noble Apes in conversation with other Noble Apes. The notion of primary truth is not there. It is relative and muddled. In code, the same event or idea is represented by something that is not referential to a single thing but in fact is completely uniquely represented per Noble Ape. As the Noble Ape replays these events through narrative either internal (in their own thinking) or external (telling any Noble Ape who will listen), it is possible for the Noble Ape's own description of the thing being discussed to change through the narrative process.

The second property highlighted through the social graph was the role that squabbling plays in the Noble Ape interactions. There is a wide variety of extremes associated with squabbling. Squabbling is a very broad description of anything from gesturing and shouting to noncontact swipes and aggressive posturing to violent blows and murder in some rare circumstances. As the Noble Apes get closer, more interaction can occur. Mottram hardcoded these interactions offering honor as the defining factor but also utilizing the level of social animosity the Noble Apes held to one another. As noted, Noble Apes implicitly have very small social groups in their recall social memory. For this reason, if a Noble Ape has a dispute with another Ape, this interaction may replace other Apes that they periodically meet and this replacement may make the Noble Apes more susceptible to creating a sometimes artificial nemesis.

9 Narrative Engine

The social simulation provides an underlying social structure that is relatively easy to understand both in short-term interactions and long-term trends primarily because it is heavily hardcoded. Each interaction has a specific condition and a coded response.

Through extended discussions with futurist linguist, Heron Stone, the challenge was made that Noble Ape should be able to simulate the linguistic phenomenon Stone advocated: every aspect of modern human existence appears to be based on an executed language program (Barbalet and Stone 2011). An internal narrative (thought) similar to the external narrative (speech) governs modern existence and should be able to be simulated through Noble Ape. While the idea of thought as language was not new, the ability to construct an internal and external narrative engine that literally drove the Noble Ape interactions was a challenge.

Up until this point, Noble Ape communications in the simulation were very basic. There was screaming and shouting and gesturing, but there was nothing that described the rich internal narrative that could capture things like belief or even things like social dance.

There is a variety of things captured by language both implicitly and explicitly. The challenge was to create a narrative engine where the Apes could have both internal dialogue (language-structured thought) and an external dialogue (language-structured speech).

Mottram and I came to this challenge at the same time. There was a shared interest in Corewar (Shock and Hupp 1982) and in the artificial life simulations like Tierra (Ray 1991). Corewar provided a thorough treatment of early stable byte-code languages. Byte-codes mean literally small atomic blocks of computer executable code. Stable byte-code languages had the benefit that although code could be modified (and the effects of these code changes could be dramatic for only a single change of the code), the actual code remained execution stable. The narrative engine for Noble Ape would have to be execution stable. Execution unstable in contrast would mean there would be byte-codes that could *crash* the Noble Ape's language, creating a fatal or irrecoverable error.

The narrative engine commands captured five kinds of things: data, sensors, actuators, operators, and conditionals. The data maintained data elements that were not executed but stored. Sensors captured a variety of simulated external senses of the Noble Apes. Actuators captured the abstracted movement of the Noble Apes. Operators covered both logical and arithmetic operators. Conditionals covered casual logic.

The original narrative engine implementation offered by Mottram had the limitation of just a single narrative. The Noble Apes had this narrative both internally and communicated this narrative externally and it existed as a single entity. I noted that this method did not capture radicalization or an ability to exist in a society and hold independent beliefs (Barbalet and Stone 2011). It was critical to have an internal and an external narrative. These two narratives needed to be quite distinct.

In the current narrative engine, each Noble Ape has an external and an internal narrative that is a stream of byte-codes. When Noble Apes meet and converse, they are running a shared program that alters their own byte-code. This is happening in parallel with their conversing companion. External narrative is exchanged and altered in parallel; this creates a conversation.

When the Noble Ape is not in conversation, the same process is going on but rather than the external narrative being run with another external narrative, the internal narrative of the Noble Ape converses with the external narrative and vice versa. The Noble Apes literally talk to themselves without uttering a simulated sound.

Mottram tied the movement or the physical action of the Noble Ape to the internal narrative. This is an ongoing point of development discourse as I contend the internal narrative should be totally private. At the same time, I concede that the spoken external narrative is not the best place to gather movement from. This mapping of movement from the internal narrative also lends a simulated weight to saying one thing but doing another.

Mottram and I had distinctly different views on the initial conditions of the narratives. My view was that the narrative byte-code should have an even and random probability of occurring in the initial internal and external narrative states. Mottram held the view that the byte-code should be genetically weighted and also contain a distinctly higher ratio of sensors to all other narrative engine types similar to the sensory wonder of a baby. The random case produced faster productive narratives both internally and externally. The genetically ordered with heavy sensor predetermined method produced more natural timescales in terms of productive and mature narrative creation.

10 Narrative Engine as Narrative Generator

The narrative engine-generated byte-code is alien when compared to the English language. It is relatively unintelligible to even those familiar with the byte-code syntax. As with the social graph to understand the social simulation, there is a need for an equivalent technology to turn the Noble Ape narrative byte-code into a human-readable form.

I wrote a scripting language to compliment Noble Ape called ApeScript (Barbalet 2005b). Rather than describing a piece of software, ApeScript creates a programming model for writing a single time-cycle of Noble Ape interaction. Nontrivially, ApeScript can cover more than just a single time-cycle of interaction, but the time-cycle (a simulated minute) is the unit of execution in simulation. ApeScript is created to cover a series of possible situations where the actual circumstances leading into the execution of the ApeScript code define which paths in the ApeScript code will be executed.

The same conditions are in place for the narrative engine byte-code. It is based in the same unit of time and has roughly the same possibilities of code paths.

At the time of writing, the initial work has been performed to translate the byte-code into ApeScript. Curiously the combined ApeScript and byte-code translation is a subset (or intersection) of both languages. It produces a robust syntax that translates both ways. ApeScript is not English, and this final translation is outside the time frame of this chapter; however, it is a direction the development needs to go to provide the following possibility.

The ability to provide a detailed description of the Noble Ape external and internal narratives would provide a compelling additional element. As with the social graph, it would give immediate feedback to a great level of detail on exactly what was happening with Noble Ape societies from an individual up to a community. If the ability to provide bidirectional translation is maintained (as the intersection of ApeScript and the byte-code narrative provides), the ability to inject English language programming back into the simulation environment is possible. Assuming that the English language programming is an intersecting set of wild English (Barbalet and Stone 2011), ApeScript and the narrative byte-code, it may not appear as fluidly readable as wild English but it would provide an ability to add a wide variety of concepts external to the simulation that would have to otherwise be grown organically through the simulation interactions or artificially hardcoded.

11 In Concert

The cognitive simulation, the social simulation, and the narrative engine are not independent simulations. Each takes from elements of the external simulation environment, and each has its own dependencies. All three simulations can be turned off allowing only one or two remaining simulations to run and interact or none of these simulations to run, to test other aspects of the Noble Ape Simulation environment. For clarity, the interactions that are explicitly hardcoded are nullified in this context.

The shared external simulation space should not be discounted in this analysis. It may appear that the cognitive simulation has the most ethereal connection to the external simulation environment. This is not the case. From the early origins of Noble Ape, the connection between movement and the forced feedback loop from the external simulation back into the cognitive simulation resulting in movement ensures the external simulation is the most important contributor to the cognitive simulation (Barbalet 1997b).

The narrative engine is the mediator between the cognitive simulation and the social simulation. Prior to the narrative engine, the Noble Apes existed as reactive agents with additional surprises through social interactions. The movement to hard-code more behaviors created a reinforcement of certain behaviors.

The narrative engine allows for the possibility of future undoing of this hardcoding. It should be possible for all the elements of the hardcoded social simulation to be removed and potentially suggested to the narrative engine. This would allow the Noble Apes to truly evolve their own social norms where concepts like honor are socially agreed upon and also open to individual and historical misinterpretation.

It is possible for the cognitive simulation to hybridize with the narrative engine as well. Consider if the narrative engine byte-codes were communicated through the cognitive simulation substrate. In this regard, the simulations discussed could all resolve to a single system and still maintain their functionality with the potential addition of new behaviors that could not have been explicitly hardcoded.

12 Noble Apes and Humans

This chapter offers a nontechnical surveying of the Noble Ape Simulation to show fundamentally that software can be a useful analytical tool for philosophy. Rather than discussing specific philosophical dilemmas posed by different philosophical models of the mind to determine the possible origin of mind, this chapter has offered a pragmatic surveying of the strengths and weaknesses of simulation methods used to model aspects of the mind as it is externally represented. This has been done intentionally to avoid implicit and oftentimes artificial paradoxes these philosophical models present. As should be clearly demonstrated through Noble Ape, three or more views of the mind can coexist in productive agents.

The connection to origin described here is relatively simple. From basic reactive chemistry through early social needs to language-dominated primates, the origin of the mind can be reduced to basic reactive chemistry; however, this is not a unique solution. There is a multiplicity of solutions.

The solution outside chemistry is equally plausible. It is perfectly credible that a mind could come from computation like the narrative engine, and that this mind would have distinct but valid origins. The narrative engine mind does not have to come through computation either. The origin of language could force the mind as an internal representation of external conversations.

Similarly the mind could come through arbitrary social constraints that force the need for a mind on the entity within the social environment. The mind would exist just as much from the society as it does from the individual.

For coherence, I will continue to write simulation software that coexists rather than finding apparent artificial paradoxes. An artificial mind, whatever its origin, is a terrible thing to waste.

References

Barbalet, T. S. (1997a). *The original manuals of Noble Ape*. Raleigh: Lulu.

Barbalet, T. S. (1997b). Noble Ape Philosophic. *Noble Ape Website*. Retrieved February 10, 2012, from http://www.nobleape.com/man/philosophic.html

Barbalet, T. S. (2004). Noble Ape simulation. *IEEE Computer Graphics and Applications, 24(2)* (pp. 6–12). Los Alamitos: IEEE Computer Society.

Barbalet, T. S. (2005a). Apple's CHUD tools, Intel and Noble Ape. *Noble Ape Website*. Retrieved February 10, 2012, from http://www.nobleape.com/docs/on_apple.html

Barbalet, T. S. (2005b). ApeScript notes. *Noble Ape Website*. Retrieved February 10, 2012, from http://www.nobleape.com/man/apescript_notes.html

Barbalet, T. S. (2009). Noble Ape's cognitive simulation: From agar to dreaming and beyond. In R. Chiong (Ed.), *Nature-inspired informatics for intelligent applications and knowledge discovery: Implications in business, science, and engineering*. Hershey: IGI Global Information Science Reference.

Barbalet, T.S., & Stone, H. (2011). *Stone Ape Podcast*. Retrieved February 10, 2012, from http://www.nobleape.com/stone/

Breazeal, C. L. (2002). *Designing sociable robots (Intelligent robotics and autonomous agents)*. Cambridge, MA: MIT Press.

Dawkins, R. (1987). *The blind watchmaker*. New York: Norton.

Emmeche, C. (1991). *The garden in the machine*. Princeton: Princeton University Press.

Kirsh, D. (1991). Today the earwig, tomorrow man? *Artificial Intelligence, 47*, 161–184.

Hobbes, T. (1651). *Leviathan*. Retrieved February 10, 2012, from http://archive.org/details/hobbessleviathan00hobbuoft

Langton, C. G. (1997). *Artificial life: An overview (Complex adaptive systems)*. Cambridge, MA: MIT Press.

Levy, S. (1992). *Artificial life: A report from the frontier where computers meet biology*. New York: Pantheon.

Ray, T. S. (1991). Evolution and optimization of digital organisms. In K. R. Billingsley et al. (Eds.), *Scientific excellence in supercomputing: The IBM 1990 contest prize papers* (pp. 489–531). Athens: The Baldwin Press.

Shock, J., & Hupp, J. (1982, March). The worms programs – Early experiences with a distributed computation. *Communications of the ACM, 25*(3), 172–180.

Volterra, V. (1931). Variations and fluctuations of the number of individuals in animal species living together. In R. N. Chapman (Ed.), *Animal ecology* (pp. 409–448). New York: McGraw-Hill.

Yaeger, L. S. (1994). Computational genetics, physiology, metabolism, neural systems, learning, vision, and behavior or PolyWorld: Life in a new context. In C. Langton (Ed.), *Proceedings of the artificial life III conference* (pp. 263–298). Reading: Addison-Wesley.