coneural
Center for Cognitive and Neural Studies

Thyrix: A simulator for articulated agents capable of manipulating objects

Răzvan V. Florian

Center for Cognitive and Neural Studies (Coneural)
Str. Saturn 24, 400504 Cluj-Napoca, Romania
www.coneural.org
florian@coneural.org

Technical Report Coneural-03-02
Version 1
August 15, 2003

Abstract

This paper describes a new simulator, Thyrix, designed for embodied cognitive science research. The simulator supports 2D agents having articulated arms, capable of manipulating objects from the environment and also of moving in the environment. The agents have sensors for vision, proprioception, and also tactile sensors distributed on the surface of their body. The simulator supports contact detection and resolution, and friction. It uses a simplified, Aristotelian mechanics, but can be easily extended to classical, Newtonian mechanics (and also to 3D). We present reasons for using such a simulator and describe design and implementation characteristics.

1 Introduction

There is an increasing awareness among the scientific community that genuine intelligence (adaptable, flexible, robust) can emerge only in a system that is embodied (i.e., has a body through which can interact with the external environment, using sensors and effectors), and is situated in an environment it can interact with (e.g., Steels & Brooks, 1995; Pfeifer & Scheier, 1999; Brooks, 1990, 1991; Bickhard, 1993; Varela, Thompson, & Rosch, 1992; Chiel & Beer, 1997; Ziemke, 2001; Florian, 2003). The essential implication of embodiment is the bidirectional, circular interaction between the body of the cognitive agent and the environment: some of the agent's actions change the state of the environment, thus changing also the influence of the environment on the agent (partly perceived through the sensors). This coupling permits the exploration by the agent of the structure of the environment and the discovery of structural invariants, through a process which depends on the sensorimotor capabilities of the agent and its goal. The agent can thus develop its own conceptualization of the environment, through self-organization and learning. The grounding of concepts on the sensorimotor interaction with the environment eliminates the problems of classical AI (lack of robustness; the lack of access to the semantic content of designer-provided symbols or categories; the confusion between the agent's perspective and the observer's perspective).

While embodiment generally implies a real physical body, like those of animals and robots, several studies (Quick, Dautenhahn, Nehaniv, & Roberts, 1999; Riegler, 2002; Oka et al., 2001) have argued that the importance of embodiment is not necessarily given by materiality, but by its special dynamic relation with the environment. This relation can also emerge in environments other than the material world, such as computational ones. The environment can be a simulated physical environment, or a genuinely computational one, such as the internet or an operating system. Simulated physical environments may be connected to the sensors and effectors of real physical agents, as in virtual reality, or may also simulate the body of the agent.

Embodied artificial intelligence research may thus be pursued using either physical robots or artificial simulated agents (animats). There are several advantages of using animats. It is much simpler to modify the body of a simulated agent than to modify a preexisting robot: it may require changing a few lines of code, versus many hours of engineering work. A simulated agent may be much cheaper to code, in comparison with the cost of a real robot. In simulation, one does not have to worry about charging the batteries. Common real robots have an autonomy of just several hours, when running on batteries. Simulated robots do not wear off, thus imposing recurrent costs on the experiment, neither break, which may result in unwanted interruptions of the experiments. In general, since the hardware considerations may be omitted, there is more time to focus on the conceptual issues.

Simulation of some simple environments, like in navigation experiments, may also be faster than real time. This makes simulation preferable for experiments where the cognitive system of the agent is generated with evolutive methods, where the behavior of generations of agents in the environment has to be tracked for long periods of time. Evolutive methods may also require the repositioning of the agent in the environment, when starting a new training epoch, which may need to be done manually for robots, but can be done automatically for animats.

There are also disadvantages of simulation. It is hard to simulate the dynamics of a physical robot and of an environment realistically, especially if the simulated agents have many degrees of freedom. In the real world, the dynamics is simply given by the laws of physics. A simulated environment is always simpler than the real world, with its infinite richness. This simplification is based on the designer's perspective of what features of the environment are important and what are negligible. On one hand, this limits the possible ontologies that the agent may develop. On the other hand, it may limit the capability of the agent to deal with the complexity of the real world.

However, if the purpose of the research is not the design of control systems that should also work in the real world, but rather the study of theoretical issues (e.g., sensorimotor integration, the self-organization of a neural system in interaction with an environment, the grounding of concepts on the sensorimotor interaction, paradigms for the emergence of representation in embodied neural systems), simulators are a useful tool. This paper presents a new simulator adapted for this purpose. It is a very simple (and thus, fast and convenient) simulator that allows the study of an agent capable of spatial movement and the manipulation of discrete objects. The simulator may thus be useful for studies of the emergence of the concept of object from the sensorimotor interaction, and also for studies involving navigation or spatial cognitive skills.

2 Purpose, design and implementation of the simulator

The software was designed to be a simple simulator (or even the simplest, if needed) in which agents are capable of navigating in an environment populated with solid objects, and of interacting with these objects using articulated arms. The agents can grasp some of the objects with their arms and move them relative to their body, carry them from place to place in the environment, explore them haptically and visually. The agents have sensors for vision, proprioception, and also tactile sensors distributed on the surface of their body. As effectors, agents can control the angles of their joints. In our experiments, agents also have a rotating "rocket" that allows them to move in the 2D space. In other experiments, agents could move in the environment by walking.

Simplicity was desired because it implies computational efficiency (rapidity in simulations) and stability in operation. A simple simulator can also be extended, as needed. Computational efficiency is very important if the simulator is used with evolutive methods, where many generations of agents have to be tested repeatedly in the environment.

Dimensionality of the space. The simplest physical environment that fits our purposes has a two-dimensional (2D) space. A 2D simulation is computationally much simpler / faster than a 3D one, but retains the needed features of the environment (spatial relations, the discreteness of the objects). The simulator was implemented in 2D, but was designed to be easily extendible to 3D.

Objects. The simulated environment may contain solid articulated agents and other solid objects. We used as solid primitives circles and capped rectangles (rectangles having two opposite sides capped with semicircles). These primitives were chosen for the computational simplicity of detecting the contact between. The code for contact detection may be easily extended to 3D, where the primitives would be spheres and capped cylinders.

The objects in the environment or the parts of the articulated agents may be composed of a single solid primitive, or of several primitives bonded together to form a composed object.

Dynamics. For computational efficiency, we have not used an environment implementing the Newtonian dynamics of the real world ($\vec{F} = m\vec{a}$), but an environment with a simplified, Aristotelian (quasistatic) dynamics, which obeys the law $\vec{F} = m\vec{v}$. The velocity of an object directly depends on the force applied to the object. There is thus no inertial movement: a body moves as long a force is applied to it, but it immediately stops if no force moves it. Objects cannot be thus thrown in our environment, they stop after the contact ceases, as in the real word they would stop if confronted with a large friction force. This may be a radical change from the laws of real physics. However, this difference is not essential from a cognitive point of view. Actually, most noneducated humans believe that the real world obeys Aristotelian laws: in elementary physics courses, children must be unteached the Aristotelian principles, sometimes with great pain (diSessa, 1982). Besides balls or other small objects thrown or kicked in the air or water, or on the surface of ice or other smooth surfaces, most other objects around us stop after one ceases pushing or pulling them, because of friction.

There are several computational advantages of using Aristotelian dynamics versus classical dynamics. The computation and integration of dynamical quantities are slightly simpler. We have no collisions to deal with, but only contacts. The treatment of friction is much simpler: dynamic friction in Aristotelian physics is equivalent with static-only friction in real (Newtonian) physics, and thus much easier to simulate computationally. Dynamic friction in real physics can yield configurations that are inconsistent or intractable computationally, and may require not only contact forces, but also contact impulses (besides collision impulses) (Baraff, 1991). The movement of objects in the environment are mainly initiated by the agents, so we can optimise the computations by not updating on each cycle objects that are out of the reach of the agents.

Contacts. The simulator detects and resolves the contact between the objects. The contact detection is integrated with the updating of the haptic sensors of the agents. The contact may be coupled with friction, if needed. For contact resolution, with or without friction, we have implemented a fast algorithm devised by Baraff (1994).

Articulations. The articulated agents have a body and one or more tree-shaped articulated limbs connected to the body. Each link of the articulated limb can rotate relative to the joint; the rotation angle can be limited to a predefined range.

The simulator does not allow loops in the structure of the articulated agent, because of the algorithm used. We have implemented a modified Featherstone-type algorithm (Featherstone, 1983, 1987), which allows the fast simulation of the dynamics of a chain of N articulated links in a computational time linearly proportional with N. Our implementation was based on the implementations described by McMillan (McMillan, 1994; McMillan, Orin, & McGhee, 1995b, 1995a, 1996). We have changed the algorithm to

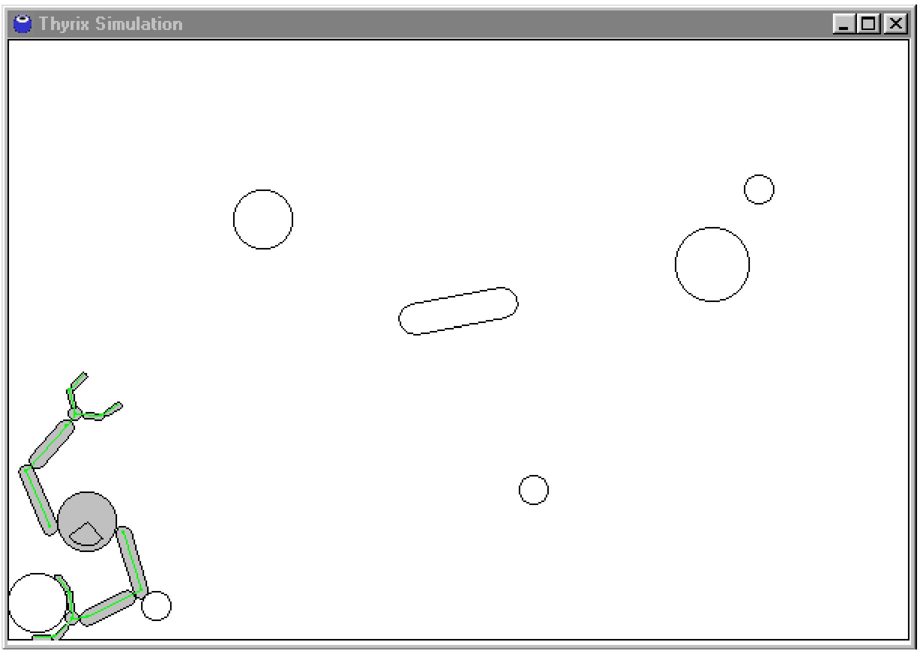

Figure 1: A screenshot of a Thyrix simulation. In the lower left corner, an articulated agent catches a circular object with its limb.

comply with the characteristics of our environment (2D, rather than 3D; and Aristotelian dynamics, rather than Newtonian). The algorithm uses the so-called "spatial notation", where the corresponding angular and linear components of velocity, acceleration and force are combined in a single vector. In 3D, these vectors are 6-dimensional, with 3 angular components and 3 linear components. In 2D, these vectors are 3-dimensional, with one angular component and 2 linear components. We have also changed the algorithm to allow contact resolution for the parts of the articulated body.

Performance. The simulator can easily attain a faster than real time performance on a common PC, even if the agents interact with several objects and tens of objects populate the environment.

Portability. The simulator has a graphical user interface developed with wxWindows[1], thus being easily ported to any major operating system (including Windows, Linux, and MacOS).

[1] http://www.wxwindows.org

3 Comparison with other existing simulators

Our simulator, named Thyrix, fills an important niche in the domain of environment-agent simulators for embodied artificial intelligence or artificial life research.

At one extreme, there exist several complex 3D simulators that can deal with articulated systems, contact and collision resolution and friction. Some of them are designed for professional simulation of robotic and other mechanical systems[2], other are designed as physics engines for professional computer games developers[3]. They are usually very costly, with prices of the order of thousands or of tens of thousands US dollars, but an open-source alternative also exists[4]. However, their features come at the cost of a high computational burden and even low stability, especially for the treatment of collisions, as some reviews show (Mueller, 2000; Lander & Hecker, 2000a, 2000b).

At another extreme, there exist many simple, 2D simulators that support Khepera-like robots having circular bodies[5]. However, they do not allow articulated limbs, and usually neither tactile sensors.

Between these extremes, very few simulators usable by embodied artificial intelligence researchers exist. In particular, there is no other 2D simulator sustaining articulated agents able to manipulate objects. Out of the existing simulators, very few support a dense array of tactile sensors on the bodies of the agents. Currently, Thyrix is the most economical (in terms of both computational needs and price) simulator that supports object manipulation.

4 Availability

The simulator is comercially available from Arxia (http://www.arxia.com/thyrix).

5 Conclusion

We have presented Thyrix, a simulator designed for embodied artificial intelligence research. The simulator supports 2D agents having articulated arms,

[2]Vortex, http://www.cm-labs.com/products/vortex/; RobotWorks, http://robotworks-eu.com/; Ropsim, http://www.camelot.dk/; Robotect, http://www.ophirtech.com/products/

[3]Havok, http://www.havok.com/; Hypermatter, http://www.hypermatter.com/

[4]OpenDE, http://opende.sourceforge.net/

[5]Yaks, http://r2d2.ida.his.se/; Evorobot, http://gral.ip.rm.cnr.it/evorobot/simulator.html; Kiks, http://www.kiks.net/; Mobotsim, http://www.mobotsoft.com/mobotsim.htm; BugWorks, http://www.bugworks.org/; The Rossum Project, http://sourceforge.net/projects/rossum/

capable of manipulating objects from the environment and also moving in the environment. The agents have sensors for vision, proprioception, and also tactile sensors distributed on the surface of their body. The simulator is a useful tool for research, having unique characteristics that recommend it in studies of agent-object interaction.

6 Acknowledgements

This work was supported by Arxia. We thank to Sorin Stan and Mihai Preda for help during the development of the software.

References

Baraff, D. (1991). Coping with friction for non-penetrating rigid body simulation. *Proceedings of SIGGRAPH 1991, Computer Graphics*, *25*, 31–40. (Available from: `http://www-2.cs.cmu.edu/~baraff/papers/sig91.pdf`)

Baraff, D. (1994). Fast contact force computation for nonpenetrating rigid bodies. In A. Glassner (Ed.), *Proceedings of SIGGRAPH 1994, Computer Graphics Proceedings, Annual Conference Series* (pp. 23–34). ACM Press. (Available from: `http://www-2.cs.cmu.edu/~baraff/papers/sig94.pdf`)

Bickhard, M. H. (1993). Representational content in humans and machines. *Journal of Experimental and Theoretical Artificial Intelligence*, *5*, 285–333. (Available from: `http://www.lehigh.edu/~mhb0/repconpage.html`)

Brooks, R. A. (1990). Elephants don't play chess. *Robotics and Autonomous Systems*, *6*, 3–15. (Available from: `http://www.ai.mit.edu/people/brooks/papers/elephants.pdf`)

Brooks, R. A. (1991). Intelligence without representation. *Artificial Intelligence Journal*, *47*, 139–159. (Available from: `http://www.ai.mit.edu/people/brooks/papers/representation.pdf`)

Chiel, H. J., & Beer, R. D. (1997). The brain has a body: Adaptive behavior emerges from interactions of nervous system, body and environment. *Trends in Neurosciences*, *20*, 553–557. (Available from: `http://vorlon.ces.cwru.edu/~beer/Papers/TINS.pdf`)

diSessa, A. (1982). Unlearning aristotelian physics: A study of knowledge based learning. *Cognitive Science*, *6 (2)*, 37–75.

Featherstone, R. (1983). The calculation of robot dynamics using articulated-body inertias. *The International Journal of Robotics Research*, *2*, 13–30.

Featherstone, R. (1987). *Robot dynamics algorithms.* Boston, MA: Kluwer Academic Publishers.

Florian, R. V. (2003). Autonomous artificial intelligent agents. *Technical Report Coneural-03-01.* (Available from: http://www.coneural.org/reports/Coneural-03-01.pdf)

Lander, J., & Hecker, C. (2000a). Product review of physics engines, part one: The stress tests. *Game Developer.* (Available from: http://www.gamasutra.com/features/20000913/lander_01.htm)

Lander, J., & Hecker, C. (2000b). Product review of physics engines, part two: The rest of the story. *Game Developer.* (Available from: http://www.gamasutra.com/features/20000920/lander_01.htm)

McMillan, S. (1994). *Computational dynamics for robotic systems on land and under water.* Columbus, OH: Ph.D. Thesis, The Ohio State University. (Available from: http://dynamechs.sourceforge.net/Papers/dissertation.tar.gz)

McMillan, S., Orin, D. E., & McGhee, R. B. (1995a). Dynamechs: An object oriented software package for efficient dynamic simulation of underwater robotic vehicles. In J. Yuh (Ed.), *Underwater robotic vehicles: Design and control* (pp. 73–98). TSI Press. (Available from: http://dynamechs.sourceforge.net/Papers/yuh_chapter.ps.gz)

McMillan, S., Orin, D. E., & McGhee, R. B. (1995b). Efficient dynamic simulation of an underwater vehicle with a robotic manipulator. *IEEE Transactions on Robotics and Automation*, 606–611. (Available from: http://dynamechs.sourceforge.net/Papers/ab_inertia.ps.gz)

McMillan, S., Orin, D. E., & McGhee, R. B. (1996). A computational framework for simulation of underwater robotic vehicle systems. *Journal of Autonomous Robots*, *3*, 253–268. (Available from: http://dynamechs.sourceforge.net/Papers/jar95.ps.gz)

Mueller, M. (2000). Physics-based simulation engines for artificial life simulations. (Available from: http://www.ifi.unizh.ch/groups/ailab/people/bongard/students/MuellerSem.pdf)

Oka, N., Morikawa, K., Komatsu, T., Suzuki, K., Hiraki, K., Ueda, K., & Omori, T. (2001). Embodiment without a physical body. In R. Pfeifer, G. Westermann, C. Breazeal, Y. Demiris, M. Lungarella, R. Nunez, &

L. Smith (Eds.), *Proceedings of the Workshop on Developmental Embodied Cognition, Edinburgh.* Edinburgh, Scotland. (Available from: http://www.cogsci.ed.ac.uk/~deco/posters/oka.pdf)

Pfeifer, R., & Scheier, C. (1999). *Understanding intelligence.* Cambridge, MA: MIT Press.

Quick, T., Dautenhahn, K., Nehaniv, C., & Roberts, G. (1999). The essence of embodiment: A framework for understanding and exploiting structural coupling between system and environment. *Proceedings of the Third International Conference on Computing Anticipatory Systems, Liege, Belgium. August 9-14, 1999 (CASYS'99).* (Available from: http://homepages.feis.herts.ac.uk/~comqkd/quick_casys99.ps)

Riegler, A. (2002). When is a system embodied? *Cognitive Systems Research*, *3*, 339–348. (Available from: http://pespmc1.vub.ac.be/riegler/papers/riegler02embodiment.pdf)

Steels, L., & Brooks, R. (Eds.). (1995). *The artificial life route to artificial intelligence: Building embodied, situated agents.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Varela, F. J., Thompson, E., & Rosch, E. (1992). *The embodied mind: Cognitive science and human experience.* Cambridge, MA: MIT Press.

Ziemke, T. (2001). The construction of 'reality' in the robot: Constructivist perspectives on situated artificial intelligence and adaptive robotics. *Foundations of Science*, *6*, 163–233. (Available from: http://researchindex.com/ziemke00construction.html)