

How does use of simulation in computer science and robotics relate to the use of simulation in cognitive science? Are there ways in which simulation is used in robotics and computer science that could be applicable to cognitive modeling? Be sure to discuss multiple types of simulation, including probabilistic simulation, simulations in planning (e.g. Monte-Carlo Tree Search), physical simulation for computer graphics, and combinations thereof.

Introduction

Simulation has been used in many contexts in computer science, ranging from simulation for physically-based animation, to probabilistic simulation, to simulated planning for learning Markov Decision Processes (MDPs). These areas of computer science have developed algorithms that I believe can play key roles in cognitive modeling, particularly in the framework of resource-rational analysis (Griffiths, Lieder, & Goodman, 2015). First, I will describe how notions of simulation in physically-based animation are themselves approximating a computational-level solution, and that they are inherently biased towards a compatibility with human perception. Second, I will discuss how simulation in control theory applies to motor control, and how it relates to the idea of analysis-by-synthesis (Halle & Stevens, 1959, 1962) in cognitive science. Third, I will argue that contrary to recent claims, simulation is not incompatible with planning; and, that recent planning methods are an intriguing class of approximate algorithms that should be explored in resource-rational analysis, particularly in concert with metacognition. Finally, I will describe how simulation is used to approximate probabilistic inference.

Simulation in physically-based animation

In physically-based animation for computer graphics, the goal is generally to determine the solution to a differential equation. However, these differential equations typically do not have closed-form symbolic solutions and therefore must be simulated numerically. This leads to many choices about which representations and approximations to use in order to get a visually plausible animation. I see two parallels between physically-based animation

and cognitive modeling. The first is related in particular to the notion of resource-rational analysis (Griffiths et al., 2015), and the second is related to the preference in physically-based animation for *plausibility* over *accuracy*.

In a resource-rational analysis, one first determines what the computational-level solution to a problem is. The second step is to pick a class of *approximate* process-level solutions to the computational-level problem, and ask the question of how that approximate solution should be used in order to make optimal use of finite cognitive resources. Typically, resource-rational analysis focuses on approximate solutions to the intractable problem of performing Bayesian inference; however, this need not be the only computational-level problem. Physically-based animation deals with another intractable computational-level. In this case, the question is about how to compute what the value of a particular physical quantity will be (e.g., position, orientation, velocity, topology, etc.) at a future point in time, given the current value for that quantity (Witkin, Baraff, & Kass, 1997). However, as this is an intractable problem, the field of physically-based animation is based on developing approximate solutions to this problem (or, in the terminology of the field, finding tractable ways to model the physical system). Thus, many approximate solutions already exist to the computational-level problem of determining future physical quantities and are good candidates for a resource-rational analysis regarding how *people* reason about future physical quantities.

I will give two examples of potential classes of approximate solutions, both in the domain of liquid simulation. In physically-based animation, there are multiple ways that fluids and liquids may be simulated. One class are *grid-based* Eulerian methods for simulating large bodies of liquid such as the water in a swimming pool (Stam, 1999). This class allows for different granularities of discretization in terms of the size of the grid squares; this type of discretization is also similar to existing qualitative models of physical reasoning (Forbus, 2011) and thus might be a way to tie a computational-level analysis of physical reasoning to algorithmic-level models like qualitative simulation. A second class of models for liquid simulation are *particle-based* Lagrangian methods for simulating the detailed dynamics

of small quantities of liquid (Müller, Charypar, & Gross, 2003). Specifically, these types of particle methods better capture behaviors such as splashing and interaction with other objects. A well-known example is smooth particle hydrodynamics (SPH), which discretizes the liquid into particles, each representing a small volume of the liquid. Particle-based methods like SPH allow a modeler to choose the number of particles, thus providing another avenue by which the amount of computation can be varied. Indeed, this approach is already being explored by some researchers (Bates et al., CogSci 2015).

Methods in physically-based animation usually emphasize stability, plausibility, and efficiency over accuracy (Witkin et al., 1997; Stam, 1999; Müller, Dorsey, & McMillan, 2002; Guendelman, Bridson, & Fedkiw, 2003; Bridson, Marino, & Fedkiw, 2003; Müller et al., 2003). In particular, the keyword here is *plausibility*—what is plausible behavior if not something that satisfies human perceptual expectations? I would argue that methods for physically-based animation are already inherently biased towards approximations that are cognitively plausible, and are thus more applicable than the analogous solutions in engineering (which are more focused on accuracy and precision). To give an example, researchers in computer graphics have developed a method of model reduction in which the principle vibration modes of a deformable object are computed, similar to finding the principle components of a matrix (Nealen, Müller, Keiser, Boxerman, & Carlson, 2006). Physical behavior can then be simulated in those subspaces, revealing the ways that an object might deform “in general” (e.g., for a rectangular rod affixed to a wall, the modes might include bending left/right, bending up/down, twisting, etc.). Because human researchers agree that this algorithm produces perceptually plausible behavior, it is worth investigating whether such an algorithm could be used as a model for how people predict the behavior of deformable objects.

Simulation in control

One area in which the idea of simulation occurs in cognitive science is the notion of forward and inverse models in motor control (Kawato, 1999; Flanagan, Vetter, Johansson, & Wolpert, 2003). The idea is that the motor system uses forward models to predict sensations occurring from actions, i.e. $x_{t+1} = f(x_t, u_t)$, where x_t is the state at time t (which may include position, velocity, and sense data) and where u_t is the action taken at time t (which is usually interpreted as force). Inverse models, also known as controllers, determine what action should be taken in order to move from one state to the next, i.e. $u_t = g(x_t, x_{t+1})$. How are these models used, and how are they learned?

Another idea from cognitive science is that of *analysis-by-synthesis*. Originally proposed in the domain of speech recognition (Halle & Stevens, 1962, 1959), analysis-by-synthesis is the idea of learning about the world by synthesizing—or simulating—data from one’s generative knowledge about how the world works. This idea has been explored in multiple domains, including machine learning in the form of the Helmholtz machine (Dayan, Hinton, Neal, & Zemel, 1995), as well as visual perception (Yuille & Kersten, 2006). In the context of forward and inverse models in motor control, a hypothesis based on analysis-by-synthesis would suggest that the sensorimotor system first learns a generative forward model of $p(x_{t+1}, x_t, u_t)$ which is then used to learn a discriminative inverse model of $p(u_t | x_{t+1}, x_t)$. This would be consistent with the finding that forward models are acquired before control in motor learning (Flanagan et al., 2003).

Recent advances in robotics have experimented with models similar to those that would be predicted by analysis-by-synthesis. Paraschos, Rueckert, Peters, and Neumann, 2015 have developed a framework for learning *probabilistic motion primitives* (ProMPs) using Gaussian Process (GP) regression. For each motion primitive, they learn a joint distribution over states and controls, $p(x_t, u_t)$, where the states include additional information about velocity and sensory information. Using this generative distribution, they can flexibly combine motion primitives; condition on specific states, controls, or sensory information; and chain motion

primitives in sequence. Levine, Wagener, and Abbeel, 2015 also learn independent local forward models and controllers and then use the marginal distributions over $p(x_t, u_t)$ from the local models to simulate training data for a discriminative model $p(u_t|x_t)$ instantiated in a neural network. A robot trained with the method from Levine et al., 2015 was able to complete highly dexterous manipulations (such as putting a peg in a hole) or complex, multi-step tasks such as turning a screw with a wrench (Han, Levine, & Abbeel, 2015).

Simulation in planning

There are two notions of simulation in planning: in one case, planning might involve running simulations (for example, physical simulations) in order to determine the effects of actions; in the other case, the simulation is the online planning process itself. Recent criticism of the use of simulation has argued that simulation is infeasible in planning systems (Davis & Marcus, n.d., 2014) and I will discuss some recent research in opposition to that argument. I will also discuss simulated episodes in reinforcement learning as a resource-rational approximation to the general problem of reinforcement learning.

Davis and Marcus, n.d., 2014 argue that simulation (specifically, physical simulation) cannot in general be used for tasks beyond physical prediction. One area that they call out in particular is planning, suggesting that simulation is too precise and/or computationally intensive to be feasible in this domain. I offer two counterexamples to this claim. First, Aoude, Luders, Joseph, Roy, and How, 2013 simulate trajectories of moving obstacles using a combination of GP regression and rapidly-expanding random trees (RRTs), and demonstrate that planning can not only make use of this type of simulation, but that such information is crucial for making safety guarantees regarding how likely it is for a collision to occur. Second, Kitaev, Mordatch, Patil, and Abbeel, 2015 demonstrate how physical simulation of object dynamics is important for planning robot trajectories when contact with those objects is inevitable. They also show how physical simulation need not depend on Monte Carlo approximations; instead, they use a differentiable dynamics model that allows them

to perform the necessary optimization without the need for many Monte Carlo samples.¹

Turning to the more general problem of planning: how do people decide what actions to take? The framework specified Markov Decision Processes (MDPs) provide a computational-level solution to this question. Given a set of states S , actions A , and potentially unknown transitions $p(S'|A, S)$ and rewards $p(R|A, S)$, how should an agent act so as to maximize their expected long-term reward? There are many reinforcement learning algorithms that aim to solve this problem (Sutton & Barto, 1998), many of which can be run in simulation: that is, the agent does not take real actions, only simulated actions, and observes transitions and rewards according to an internal model of the environment.² While traditional RL algorithms have usually used point estimates of rewards and transitions, recent Bayes-adaptive methods have proposed maintaining full distributions over possible transition and reward models (Dearden, Friedman, & Andre, 1999). This has the benefit of allowing the agent to perform efficient inference in complex structured domains (Ross & Pineau, 2008), similar to how probabilistic models of cognition have showed how agents might quickly learn higher level structures such as overhypotheses (Kemp, Perfors, & Tenenbaum, 2007). Even more recently, an algorithm known as Monte-Carlo Tree Search (Browne et al., 2012) has been proposed as a more feasible way to perform inference in Bayes-adaptive MDPs, specifically in the formulation of the Bayes-Adaptive Monte Carlo Planning (BAMCP) algorithm (Guez, Silver, & Dayan, 2013). The BAMCP method suggests a class of approximation algorithms that could apply to how people solve the reinforcement learning problem (which it has been suggested that they do, e.g. Baker and Tenenbaum, 2014), in which the amount of computation can be adjusted through the number of simulations that are run. Moreover, there are interesting questions regarding the meta-level decision problem of how to optimally use simulation in the context of planning (Hay, Russell, Tolpin, & Shimony, 2012). While Hay

¹Kitaev et al., 2015 did not develop the differentiable physics engine that they use. The physics engine is called MuJoCo and was developed by Mordatch et al.

²Although reinforcement learning algorithms are frequently classified as *model-based* or *model-free*, a model-free algorithm can still be used in simulation provided the agent has separately learned a model of the environment.

et al., 2012 discussed how the meta-level decision problem applies to MCTS in general, the question has not been posed for the full Bayes-adaptive version (as far as I am aware).

Simulation in probabilistic inference

Probabilistic inference is perhaps the area of computer science that has already been most widely explored in cognitive science, as it is a fundamental component in probabilistic models of cognition (Tenenbaum, Kemp, Griffiths, & Goodman, 2011). Indeed, Griffiths et al., 2015 use probabilistic simulation as an example of the types of approximation algorithms that should be used in resource-rational analysis. For example, Lieder, Griffiths, and Goodman, 2012 perform a resource-rational analysis involving simulation via the Metropolis-Hastings algorithm in order to explain the anchoring bias (Tversky & Kahneman, 1974). Similarly, Abbott and Griffiths, 2011 use probabilistic simulation via the particle filtering algorithm to explain primacy and recency effects.

Another type of probabilistic simulation known as *Hamiltonian Monte-Carlo* (Neal, 2011) has parallels to the same types of approximations I discussed earlier in the section on physically-based animation. HMC must similarly solve an intractable differential equation, and does so using the Leapfrog method. The Leapfrog method is an *explicit* integration method, which means that it directly solves for $f(x + \delta x)$ (though it does so in a way that is more robust than Euler's method, which is the canonical explicit method). However, physically-based animation often relies on *implicit* time integration, in which a system of equations with $f(x + \delta x)$ on either side implicitly give the solution; however, the system of equations must further be solved to get the actual solution. Implicit methods tend to be more numerically stable than explicit solutions, and thus it is worth asking the question: could HMC be computed with implicit time integration and still maintain its desirable properties? If so, then changing the discretization of the time integration would be a way to manipulate the amount of computation that HMC requires beyond just the number of samples.

Simulation may also occur in probabilistic inference not through direct Monte-Carlo

sampling, but as a prior. Schulman, Lee, Ho, and Abbeel, [2013](#) make clever use of physical simulation to compute the M-step of the EM algorithm in order to track the topology of deformable objects such as rope, cloth, and foam. To enforce the likelihood term, they convert the likelihood into a potential and then apply that potential as a force in the physics simulation itself. The simulation is biased towards a low-energy state (the prior), and thus the result of stepping the simulation forward after applying the likelihood potential gives a posterior prediction of the current topology of the object. This type of way of thinking about physical simulation in particular offers a different perspective on the idea of physical reasoning being the result of Monte-Carlo sampling.

Conclusion

To conclude, I have outlined here several areas of computer science that have much to offer, particularly in terms of thinking about how different forms of simulation might be candidate approximation algorithms for use in resource-rational analysis.

References

- Abbott, J. T. & Griffiths, T. L. (2011). Exploring the influence of particle filter parameters on order effects in causal learning. In *Proceedings of the 33rd annual conference of the cognitive science society*. Retrieved from https://cocosci.berkeley.edu/josh/papers/cogsci2011_pfOrderEffects.pdf
- Aoude, G. S., Luders, B. D., Joseph, J. M., Roy, N., & How, J. P. (2013). Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns. *Autonomous Robots*, 35(1), 51–76. doi:10.1007/s10514-013-9334-3
- Baker, C. L. & Tenenbaum, J. B. (2014). Modeling Human Plan Recognition using Bayesian Theory of Mind. In *Plan, activity, and intent recognition* (Chap. 7, pp. 177–204). doi:10.1016/B978-0-12-398532-3.00007-5
- Bridson, R., Marino, S., & Fedkiw, R. (2003). Simulation of clothing with folds and wrinkles. *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 28–36. Retrieved from <http://dl.acm.org/citation.cfm?id=846281>
- Browne, C., Powley, E., Whitehouse, D., Lucas, S., Cowling, P. I., Rohlfshagen, P., ... Colton, S. (2012). A survey of Monte Carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in Games*, 4(1), 1–10. doi:10.1109/TCIAIG.2012.2186810
- Davis, E. & Marcus, G. F. (n.d.). The Scope and Limits of Simulation in Automated Reasoning. *Artificial Intelligence*. Retrieved from <http://www.cs.nyu.edu/faculty/davise/papers/SimulationSubmitAIJ.pdf>
- Davis, E. & Marcus, G. F. (2014). The Scope and Limits of Simulation in Cognitive Models. *arXiv:1506.04956 [cs.AI]*. arXiv: 1506.04956. Retrieved from <http://arxiv.org/abs/1506.04956>
- Dayan, P., Hinton, G. E., Neal, R. M., & Zemel, R. S. (1995). The Helmholtz machine. *Neural Computation*, 7(5), 889–904. doi:10.1162/neco.1995.7.5.889
- Dearden, R., Friedman, N., & Andre, D. (1999). Model based Bayesian exploration. *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, 150–159. Retrieved from <http://dl.acm.org/citation.cfm?id=2073814>
- Flanagan, R. R., Vetter, P., Johansson, R. S., & Wolpert, D. M. (2003). Prediction precedes control in motor learning. *Current Biology*, 13(2), 146–150. doi:10.1016/S0960-9822(03)00007-1
- Forbus, K. D. (2011). Qualitative modeling. *Wiley Interdisciplinary Reviews: Cognitive Science*, 2(4), 374–391. doi:10.1002/wcs.115
- Griffiths, T. L., Lieder, F., & Goodman, N. D. (2015). Rational Use of Cognitive Resources: Levels of Analysis Between the Computational and the Algorithmic. *Topics in Cognitive Science*, 7(2), 217–229. doi:10.1111/tops.12142
- Guendelman, E., Bridson, R., & Fedkiw, R. (2003). Nonconvex Rigid Bodies with Stacking. *ACM Transactions on Graphics*, 22(3). doi:10.1145/882262.882358
- Guez, A., Silver, D., & Dayan, P. (2013). Scalable and efficient Bayes-adaptive reinforcement learning based on Monte-Carlo tree search. *Journal of Artificial Intelligence Research*, 48, 841–883. doi:10.1613/jair.4117
- Halle, M. & Stevens, K. M. (1959). Analysis by synthesis. In W. Wathen-Dunn & L. E. Woods (Eds.), *Proceedings of the seminar of speech compression and processing* (Vol. 2, D7).

- Halle, M. & Stevens, K. M. (1962). Speech recognition: a model and a program for research. *IRE Transactions on Information Theory*, 8(2), 155–159. doi:[10.1109/TIT.1962.1057686](https://doi.org/10.1109/TIT.1962.1057686)
- Han, W., Levine, S., & Abbeel, P. (2015). Learning Compound Multi-Step Controllers under Unknown Dynamics. *Proceedings of the 28th IEEE/RSJ International Conference on Intelligent Robots and Systems*. Retrieved from http://rll.berkeley.edu/reset_controller/reset_controller.pdf
- Hay, N. J., Russell, S. J., Tolpin, D., & Shimony, S. E. (2012). Selecting Computations: Theory and Applications. *arXiv preprint arXiv:1207.5878v1 [cs.AI]*. arXiv: [1207.5879](https://arxiv.org/abs/1207.5879). Retrieved from <http://arxiv.org/abs/1207.5879>
- Kawato, M. (1999). Internal models for motor control and trajectory planning. *Current Opinions in Neurobiology*, 9(6), 718–727. doi:[10.1016/S0959-4388\(99\)00028-8](https://doi.org/10.1016/S0959-4388(99)00028-8)
- Kemp, C., Perfors, A., & Tenenbaum, J. B. (2007). Learning overhypotheses with hierarchical Bayesian models. *Developmental Science*, 10(3), 307–321. doi:[10.1111/j.1467-7687.2007.00585.x](https://doi.org/10.1111/j.1467-7687.2007.00585.x)
- Kitaev, N., Mordatch, I., Patil, S., & Abbeel, P. (2015). Physics-Based Trajectory Optimization for Grasping in Cluttered Environments. *Proceedings of the IEEE International Conference on Robotics and Automation*. Retrieved from <http://www.eecs.berkeley.edu/~pabbeel/papers/2015-ICRA-clutter.pdf>
- Levine, S., Wagener, N., & Abbeel, P. (2015). Learning Contact-Rich Manipulation Skills with Guided Policy Search. *Proceedings of the IEEE International Conference on Robotics and Automation*. arXiv: [1501.05611v1](https://arxiv.org/abs/1501.05611v1). Retrieved from <http://arxiv.org/abs/1501.05611v1>
- Lieder, F., Griffiths, T. L., & Goodman, N. D. (2012). Burn-in, bias, and the rationality of anchoring. *Advances in Neural Information Processing Systems*, 25. Retrieved from <http://papers.nips.cc/paper/4719-burn-in-bias-and-the-rationality-of-anchoring>
- Müller, M., Charypar, D., & Gross, M. (2003). Particle-Based Fluid Simulation for Interactive Applications. *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 154–159. Retrieved from <http://dl.acm.org/citation.cfm?id=846298>
- Müller, M., Dorsey, J., & McMillan, L. (2002). Stable Real-time Deformations. *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. doi:[10.1145/545261.545269](https://doi.org/10.1145/545261.545269)
- Neal, R. M. (2011). MCMC using Hamiltonian dynamics. In S. Brooks, A. Gelman, G. Jones, & X.-L. Meng (Eds.), *Handbook of markov chain monte carlo* (Chap. 5). Chapman and Hall. arXiv: [1206.1901](https://arxiv.org/abs/1206.1901). Retrieved from <http://arxiv.org/abs/1206.1901>
- Nealen, A., Müller, M., Keiser, R., Boxerman, E., & Carlson, M. (2006). Physically based deformable models in computer graphics. *Computer Graphics Forum*, 25(4), 809–836. doi:[10.1111/j.1467-8659.2006.01000.x](https://doi.org/10.1111/j.1467-8659.2006.01000.x)
- Paraschos, A., Rueckert, E., Peters, J., & Neumann, G. (2015). Model-Free Probabilistic Movement Primitives for Physical Interaction. *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems*. Retrieved from http://www.ausy.tu-darmstadt.de/uploads/Team/PubAlexParaschos/Paraschos_IROS_2015.pdf
- Ross, S. & Pineau, J. (2008). Model-based Bayesian Reinforcement Learning in Large Structured Domains. *Proceedings of the 24th Conference in Uncertainty in Artificial Intelligence*, 476–483. Retrieved from <http://arxiv.org/abs/1206.3281>

- Schulman, J., Lee, A. X., Ho, J., & Abbeel, P. (2013). Tracking deformable objects with point clouds. *Proceedings of the IEEE International Conference on Robotics and Automation*. doi:[10.1109/ICRA.2013.6630714](https://doi.org/10.1109/ICRA.2013.6630714)
- Stam, J. (1999). Stable Fluids. *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*. doi:[10.1145/311535.311548](https://doi.org/10.1145/311535.311548)
- Sutton, R. S. & Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press. Retrieved from <https://mitpress.mit.edu/books/reinforcement-learning>
- Tenenbaum, J. B., Kemp, C., Griffiths, T. L., & Goodman, N. D. (2011). How to grow a mind: statistics, structure, and abstraction. *Science*, *331*(6022), 1279–85. doi:[10.1126/science.1192788](https://doi.org/10.1126/science.1192788)
- Tversky, A. & Kahneman, D. (1974). Judgment under Uncertainty: Heuristics and Biases. *Science*, *185*(4157), 1124–1131. doi:[10.1126/science.185.4157.1124](https://doi.org/10.1126/science.185.4157.1124)
- Witkin, A., Baraff, D., & Kass, M. (1997). *An Introduction to Physically-Based Modeling (SIGGRAPH '97 Course Notes)*. Retrieved from <http://www.cs.cmu.edu/~baraff/pbm/pbm.html>
- Yuille, A. L. & Kersten, D. (2006). Vision as Bayesian inference: analysis by synthesis? *Trends in Cognitive Sciences*, *10*(7), 301–308. doi:[10.1016/j.tics.2006.05.002](https://doi.org/10.1016/j.tics.2006.05.002)