Decision Support for Safe AI Design

Bill Hibbard

SSEC, University of Wisconsin, Madison, WI 53706, USA
test@ssec.wisc.edu

Abstract: There is considerable interest in ethical designs for artificial intelligence (AI) that do not pose risks to humans. This paper proposes using elements of Hutter's agent-environment framework to define a decision support system for simulating, visualizing and analyzing AI designs to understand their consequences. The simulations do not have to be accurate predictions of the future; rather they show the futures that an agent design predicts will fulfill its motivations and that can be explored by AI designers to find risks to humans. In order to safely create a simulation model this paper shows that the most probable finite stochastic program to explain a finite history is finitely computable, and that there is an agent that makes such a computation without any unintended instrumental actions. It also discusses the risks of running an AI in a simulated environment.

Keywords: rational agent, agent architecture, agent motivation

1 Introduction

Some scientists expect artificial intelligence (AI) to greatly exceed human intelligence during the 21st century (Kurzweil, 2005). There has been concern about the possible harmful effect of intelligent machines on humans since at least Assimov's Laws of Robotics (1942). More recently there has been interest in the ethical design of AI (Hibbard, 2001; Bostrom, 2003; Goertzel, 2004; Yudkowsky, 2004; Hibbard, 2008; Omohundro, 2008; Waser 2010; Waser 2011; Muehlhauser and Helm, 2012).

Hutter's universal AI (2005) defined an agent-environment framework for reasoning mathematically about AI. This paper proposes using elements of this framework to define a decision support system for exploring, via simulation, analysis and visualization, the consequences of possible AI designs. The claim is not that the decision support system would produce accurate simulations of the world and an AI agent's effects. Rather, in the agent-environment framework the agent makes predictions about the environment and chooses actions, and the decision support system uses these predictions and choices to explore the future that the AI agent predicts will optimize its motivation.

This is related to the oracle AI approach of Armstrong, Sandberg and Bostrom (forthcoming), in that both approaches use an AI whose only actions are to provide information to humans. The oracle AI is a general question answerer, whereas the decision support approach focuses on specific capabilities from the mathematical agent-environment framework. The oracle AI is described as a general AI with restricted ability to act on its environment. The decision support system applies part of the agent-environment framework to learn a model for the environment, and then uses that model to create a simulated environment for evaluating an AI agent defined using the framework. Chalmers (2010) considers the problem of restricting an AI to a simulation and concludes that it is inevitable that information will flow in both directions between the real and simulated worlds. The oracle AI paper and Chalmers' paper both consider various approaches to preventing an AI from breaking out of its restriction to not act in the real world, including physical limits and conditions on the AI's motivation. In this paper, a proposed AI design being evaluated in the decision support system has a utility function defined in terms of its simulated environment, has no motivation past the end of its simulation and the simulation is not visualized or analyzed until the simulation is compete.

The next section presents the mathematical framework for reasoning about AI agents. The third section discusses sources of AI risk. The fourth section discusses the proposed decision support system. The final section is a summary of the proposal.

2 An Agent-Environment Framework

We assume that an agent interacts with an environment. At each of a discrete series of time steps $t \in \mathbf{N} = \{0, 1, 2, ...\}$ the agent sends an action $a_t \in A$ to the environment and receives an observation $o_t \in O$ from the environment, where A and O are finite sets. We assume that the environment is computable and we model it by programs $q \in Q$, where Q is some set of programs. Let $h = (a_1, o_1, ..., a_t, o_t) \in H$ be an interaction history where H is the set of all finite histories, and define $|h| = t$ as the length of the history h. Given a program $q \in Q$ we write $o(h) = U(q, a(h))$, where $o(h) = (o_1, ..., o_t)$ and $a(h) = (a_1, ..., a_t)$, to mean that q produces the observations o_i in response to the actions a_i for $1 \leq i \leq t$ (U is a program interpreter). Given a program q the probability $\rho(q) : Q \rightarrow [0, 1]$ is the agent's prior belief that q is a true model of the environment. The prior probability of history h, denoted $\rho(h)$, is computed from $\rho(q)$ (two ways of doing this are presented later in this section).

An agent is motivated according to a *utility function* $u : H \rightarrow [0, 1]$ which assigns utilities between 0 and 1 to histories. Future utilities are discounted according to a *geometric temporal discount* $0 \leq \gamma < 1$ (Sutton and Barto, 1998). The value $v(h)$ of a possible future history h is defined recursively by:

$$v(h) = u(h) + \gamma \max_{a \in A} v(ha) \tag{1}$$

$$v(ha) = \sum_{o \in O} \rho(o \mid ha)\, v(hao) \tag{2}$$

Then the agent π is defined to take, after history h, the action:

$$\pi(h) := a_{|h|+1} = \mathrm{argmax}_{a \in A}\, v(ha) \tag{3}$$

For Hutter's universal AI (2005), Q is the set of programs for a deterministic prefix universal Turing machine (PUTM) U (Li and Vitanyi, 1997). The environment may be non-deterministic in which case it is modeled by a distribution of deterministic programs. The prior probability $\rho(q)$ of program q is $2^{-|q|}$ where $|q|$ is the length of q in bits, and the prior probability of history h is given by:

$$\rho(h) = \sum_{q:o(h)=U(q,\, a(h))} \rho(q) \tag{4}$$

Hutter's universal AI is a *reinforcement-learning* agent, meaning that the observation includes a reward r_t (i.e., $o_t = (\hat{o}_t\ , r_t)$) and $u(h) = r_{|h|}$. Hutter showed that his universal AI maximizes the expected value of future history, but it is not finitely computable.

As Hutter discussed (2009a; 2009b), for real world agents single finite stochastic programs (limited to finite memory, for which the halting problem is decidable) such as Markov decision processes (MDPs) (Puterman, 1994; Sutton and Barto, 1998) and dynamic Bayesian networks (DBNs) (Ghahramani 1997) are more practical than distributions of PUTM programs for defining environment models. Modeling an environment with a single stochastic program rather than a distribution of deterministic PUTM programs requires a change to the way that $\rho(h)$ is computed in (4). Let Q be the set of all programs (these are bit strings in some language for defining MDPs, DBNs or some other finite stochastic programming model), let $\rho(q) = 4^{-|q|}$ be the prior probability of program q where $|q|$ is the length of q in bits ($4^{-|q|}$ to ensure that $\sum_{q\in Q} \rho(q) \leq 1$ since program strings in Q are not prefix-free), and let $P(h \mid q)$ be the probability that q computes the history h[1]. Note $\rho(q)$ is a discrete distribution on individual program strings, not a measure on bit strings in the sense of page 243 of (Li and Vitanyi, 1997). Then given a history h_0, the environment model is the single program that provides the most probable explanation of h_0, that is the q that maximizes $P(q \mid h_0)$. By Bayes theorem:

$$P(q \mid h_0) = P(h_0 \mid q)\ \rho(q) \,/\, P(h_0) \tag{5}$$

$P(h_0)$ is constant over all q so can be eliminated. Thus we define $\lambda(h_0)$ as the most probable program modeling h_0 by:

$$\lambda(h_0) := \text{argmax}_{\ q\in Q}\ P(h_0 \mid q)\ \rho(q) \tag{6}$$

The following result is proved in (Hibbard, 2012b).

Proposition 1. Given a finite history h_0 the model $\lambda(h_0)$ can be finitely computed.

[1] $P(h \mid q)$ is the probability that q produces the observations o_i in response to the actions a_i for $1 \leq i \leq |h|$. For example let $A = \{a, b\}$, $O = \{0, 1\}$, $h = (a, 1, a, 0, b, 1)$ and let q generate observation 0 with probability 0.2 and observation 1 with probability 0.8, without any internal state or dependence on the agent's actions. Then the probability that the interaction history h is generated by program q is the product of the probabilities of the 3 observations in h: $P(h \mid q) = 0.8 \times 0.2 \times 0.8 = 0.128$. If the probabilities of observations generated by q depended on internal state or the agent's actions, then those would have to be taken into account.

Given an environment model $q_0 = \lambda(h_0)$ the following can be used for the prior probability of an observation history h in place of (4):

$$\rho(h) = P(h \mid q_0) \tag{7}$$

According to current physics our universe is finite (Lloyd, 2002) and for finite environments agents based on (6) and (7) are as optimal as those based on (4). And their prior probabilities better express algorithmic complexity if finite stochastic programs are expressed in an ordinary procedural programming language restricted to have only static array declarations, to have no recursive function definitions, and to include a source of truly random numbers.

3 Sources of AI Risk

Dewey (2011) argued that reinforcement-learning agents will modify their environments so that they can maximize their utility functions without accomplishing the intentions of human designers. He discussed ways to avoid this problem with utility functions not conforming to the reinforcement-learning definition. Ring and Orseau (2011) argued that reinforcement-learning agents will self-delude, meaning they will choose to alter their own observations of their environment to maximize their utility function regardless of the actual state of the environment. In (Hibbard, 2012a) I demonstrated by examples that agents with utility functions defined in terms of agents' environment models can avoid self-delusion, and also proved that under certain assumptions agents will not choose to self-modify.

Omohundro (2008) and Bostrom (forthcoming) describe how any of a broad range of primary AI motivations will imply secondary, unintended motivations for the AI to preserve its own existence, to eliminate threats to itself and its utility function, and to increase its own efficiency and computing resources. Bostrom discusses the example of an AI whose primary motive is to compute pi and may destroy the human species due to implied instrumental motivations (e.g., to eliminate threats and to increase its own computing resources). Omohundro uses the term "basic AI drives" and Bostrom uses "instrumental goals" but as I argue in (Hibbard, 2012b) they should really be called "unintended instrumental actions" since the agent's whole motivation is defined by its utility function.

4 A Decision Support System

The decision support system is intended to avoid the dangers of AI by having no motivation and no actions on the environment, other than reporting the results of its computations to the environment. However, the system runs AI agents in a simulated environment, so it must be designed to avoid subtle unintended instrumental actions.

The first stage of the system is an agent, here called π_6, that learns a model of the real world environment in order to provide a simulated environment for studying proposed AI agents. An AI agent is defined by (1)-(3), (6) and (7), but (6) can be used alone to define the agent π_6 that learns a model $\lambda(h_0)$ from history h_0. In order for π_6 to learn an accurate model of the environment the interaction history h_0 should include agent actions, but for safety π_6 cannot be allowed to act. The resolution is for its actions to be made by many safe, human-level surrogate AI agents independent of π_6 and of each other. Actions of the surrogates include natural language and visual communication with each human. The agent π_6 observes humans, their interactions with the surrogates and physical objects in an interaction history h_0 for a time period set by π_6's designers, and then reports an environment model to the environment (specifically to the decision support system, which is part of the agent's environment). The following result is proved in (Hibbard, 2012b). While it may seem obvious, given the subtlety of unintended behaviors it is worth proving.

Proposition 2. The agent π_6 will report the model $\lambda(h_0)$ to the environment accurately and will not make any other, unintended instrumental actions.

The decision support system analyzes proposed AI agents that observe and act in a simulated environment inside the decision support system. To formalize the simulated environment define O' and A' as models of O and A with bijections $m_O : O \leftrightarrow O'$ and $m_A : A \leftrightarrow A'$. Define H' as the set of histories of interactions via O' and A', with a bijection $m_H : H \leftrightarrow H'$ computed by applying m_O and m_A individually to the observations and actions in a history. Given h_p as the history observed by π_6 up to time $|h_p| = \mathit{present}$, define $h'_p = m_H(h_p)$ as the history up to the present in the simulated environment. Let Q' be a set of finite stochastic programs for the simulated environment and π'_6 be a version of the environment-learning agent π_6 for the simulated environment. It produces:

$$q'_p = \lambda(h'_p) := \operatorname{argmax}_{q' \in Q'} P(h'_p \mid q')\, \rho(q') \tag{8}$$

$$\rho'(h') = P(h' \mid q'_p) \tag{9}$$

Now let $\pi'(h'; \rho', u', \gamma')$ be a proposed AI agent to be studied using the decision support system, where u' is its utility function, γ' is its temporal discount and *future* is the end time of the simulation. The utility function u' is constrained to have no motivation after time = *future*:

$$\forall h' \in H'.\ |h'| > \mathit{future} \Rightarrow u'(h') = 0 \tag{10}$$

Then $\pi'(h'; \rho', u', \gamma')$ is defined by:

$$v'(h') = u'(h') + \gamma' \max_{a' \in A'} v'(h'a') \tag{11}$$

$$v'(h'a') = \textstyle\sum_{o' \in O'} \rho'(o' \mid h'a')\, v'(h'a'o') \tag{12}$$

$$\pi'(h'; \rho', u', \gamma') := a'_{|h'|+1} = \operatorname{argmax}_{a' \in A'} v'(h'a') \tag{13}$$

There are no humans or physical objects in the simulated environment; rather the agent π' (using π' and $\pi'(h')$ as abbreviations for $\pi'(h'; \rho', u', \gamma')$) interacts with a simulation model of humans and physical objects via:

$$a'_{|h'|+1} = \pi'(h') \tag{14}$$

$$o'_{|h'|+1} = o' \in O' \text{ with probability } \rho'(o' \mid h'a'_{|h'|+1}) \tag{15}$$

The decision support system propagates from h'_p to h'_f, where $|h'_f| = future$, by repeatedly applying (14) and (15). As in (Hibbard, 2012a) let Z' be the set of finite histories of the internal states of $\lambda(h'_p)$ and let $P(z' \mid h', \lambda(h'_p))$ be the probability that $\lambda(h'_p)$ computes $z' \in Z'$ given $h' \in H'$. The decision support system then computes a history of model states by:

$$z'_f = z' \in Z' \text{ with probability } P(z' \mid h'_f, \lambda(h'_p)) \tag{16}$$

The simulation in (14)-(16) is stochastic so the decision support system will support ensembles of multiple simulations to provide users with a sample of possible futures. An ensemble of simulations generates an ensemble of histories of model states $\{z'_{f,e} \mid 1 \leq e \leq m\}$, all terminating at time = *future*. These simulations should be completed before they are visualized and analyzed; that is visualization and analysis should not be concurrent with simulation for reasons discussed in Section 4.1.

The history h_p includes observations by π_6 of humans and physical objects, and so the decision support system can use the same interface via A' and O' (as mapped by m_A and m_O) to the model $\lambda(h'_p)$ for observing simulated humans and physical objects in state history $z'_{f,e}$. These interfaces can be used to produce interactive visualizations of $z'_{f,e}$ in a system that combines features of Google Earth and Vis5D (Hibbard and Santek, 1990), which enabled scientists to interactively explore weather simulations in three spatial dimensions and time. Users will be able to pan and zoom over the human habitat, as in Google Earth, and animate between times *present* and *future*, as in Vis5D. The images and sounds the system observes of the model $\lambda(h'_p)$ executing state history $z'_{f,e}$ can be embedded in the visualizations in the physical locations of the agent's observing systems, similar to the way that street views and user photographs are embedded in Google Earth.

The decision support system can also match specifications for specific humans and physical objects to the images and sounds it observes of the model $\lambda(h'_p)$ executing state history $z'_{f,e}$. The specifications may include text descriptions, images, sounds, animations, tables of numbers, mathematical descriptions, or virtually anything. Recognized humans and physical objects can then be represented by icons in the visualization, in their simulated physical locations and with recognized properties of humans and objects represented by colors and shapes of the icons. The system can enable users to selectively enable different layers of information in the visualizations.

Vis5D enables users to visualize ensembles of weather forecasts in a spreadsheet of parallel visualizations where spatial view, time and level selections are synchronized between spreadsheet cells. The decision support system can provide a similar spreadsheet visualization capability for ensembles of simulations.

The point isn't that these simulations are accurate predictions of the future, but that they do depict the future that the AI agent will create, according to the environment model learned by π'_6. The visualization depicts the future that the agent $\pi'(h')$ predicts will maximize the sum of future discounted utility function values. Agent designers can use this visualization to understand the consequences of various proposed utility functions and temporal discounts.

The visualizations will be accompanied by analyses, in which the decision support system computes the values of functions $g(z'_{f,e})$. Evaluation of these functions may depend on recognizing humans and physical objects in observations of the model $\lambda(h'_p)$ executing state history $z'_{f,e}$. Such analyses would enable designers to compute quantitative evaluations to complement visualizations. For example, these analyses can be used to compute statistics about future human wealth, health and intelligence.

There will be an interval before time = *future* when visualizations and analyses of the behavior of π' will not be an accurate reflection of the behavior of a real-world agent without the temporal cutoff in (10). The length of this interval will depend on the value of the temporal discount γ'. In order to compensate for this, *future* should be chosen past the time horizon of interest.

4.1 Will an Agent $\pi'(h'; \rho', u', \gamma')$ Act in the Real World?

The environment model $q'_p = \lambda(h'_p)$ may provide an agent π' with the information that humans have constructed a decision support system for analyzing proposed AI agents and that π' is such an agent, interacting with a simulated environment. If the utility function u' is defined in terms of the real environment then π' may predict via ρ' that certain actions in the simulated environment may manipulate the human users of the decision support system into increasing the value of u' in the real environment. Hence it is important that the utility function u' be defined in terms of the simulated rather than the real environment.

Even when u' is defined in terms of the simulated environment there are still possible risks. Assuming that $q'_p = \lambda(h'_p)$ includes the information that π' is in a simulation, then the agent may predict via ρ' that it can bribe human users of the decision support system, through an offer to act on their behalf in the real world (the bribed humans would have to enable π' to act in the real world), to manipulate its simulated environment to increase the value of u'. This offer of a bribe could be made by spoken or written language inserted by actions of π' into some $z'_{f,e}$ and visualized by the users. This possibility is the rationale for constraining that $u'(h') = 0$ for $|h'| >$ *future* in (10) and for requiring simulation to be complete before visualization and analysis begin. Looking at its definition in (11)-(13) it is hard to see any reason why π' would be motivated to affect events after time = *future*, and by the time a user sees any bribe offer the simulation would be complete through time = *future*. It is also hard to see any reason why π' running in one simulation would be motivated to affect another simulation occurring at a later time (if the simulations were running simultaneously then there could be some communication and interaction between them mediated by human users). As proved in (Hibbard, 2012a) π' will not have any motive to modify u', so π' will not modify itself to be motivated past time = *future*. However these are subtle issues and I do not claim they are completely resolved.

It is important not to anthropomorphize π'. A human restricted to act in a simulation would be motivated to act in the real world. But if u' is defined in terms of a simulation then π' would not be motivated to act in the real world, except as a means to increase u' in the simulation.

The greatest risk comes from the human users of the decision support system who may be tempted (Hibbard, 2009) to modify it to act in the real world on their behalf. As Elliott (2005) comments on the safety of US nuclear weapons, "The human factor introduces perhaps the weakest link in nuclear weapon safety and control." However, if society takes AI risks seriously then it can learn from the experience managing nuclear weapons to manage AI and some form of the proposed decision support system.

5 Discussion

An important challenge for safe AI is understanding the consequences of AI designs, particularly the consequences of AI utility functions. This paper proposes a decision support system for evaluating AI designs in safe, simulated environments that model our real environment. The paper shows that the agent π_6 is safe and learns to model our environment in a finite computation. The paper also addresses some possible risks in running and evaluating AI designs in simulated environments. It would be useful to find computationally feasible implementations for the definitions in this paper.

I believe that the greatest danger of AI comes from the fact that above-human-level AI is likely to be a tool in military and economic competition between humans and thus have motives that are competitive toward some humans. Some form of the proposed decision support system may be able to alert those building powerful AI to the long term consequences of decisions they take in the heat of competition.

Acknowledgements. I would like to thank Luke Muehlhauser for helpful discussions.

References

1. Asimov, I. 1942. Runaround. Astounding Science Fiction.
2. Bostrom, N. 2003. Ethical issues in advanced artificial intelligence. In: Smit, I. et al (eds) Cognitive, Emotive and Ethical Aspects of Decision Making in Humans and in Artificial Intelligence, Vol. 2, pp. 12-17. Int. Inst. of Adv. Studies in Sys. Res. and Cybernetics.
3. Bostrom, N. Forthcoming. The superintelligent will: Motivation and instrumental rationality in advanced artificial agents. *Minds and Machines*.
4. Chalmers, D. 2010. The Singularity: A Philosophical Analysis. J. Consciousness Studies 17, pp. 7-65.
5. Dewey, D. 2011. Learning what to value. In: Schmidhuber, J., Thórisson, K.R., and Looks, M. (eds) AGI 2011. LNCS (LNAI), vol. 6830, pp. 309-314. Springer, Heidelberg.
6. Elliott, G. 2005. US Nuclear Weapon Safety and Control. MIT Program in Science, Technology, and Society. http://web.mit.edu/gelliott/Public/sts.072/paper.pdf
7. Ghahramani, Z. 1997. Learning dynamic Bayesian networks. In: Giles, C., and Gori, M. (eds), Adaptive Processing of Temporal Information. LNCS, vol. 1387, pp. 168-197. Springer, Heidelberg.

8. Goertzel, B. 2004. Universal ethics: the foundations of compassion in pattern dynamics. http://www.goertzel.org/papers/UniversalEthics.htm
9. Hibbard, B., and Santek, D. 1990. The Vis5D system for easy interactive visualization. Proc. IEEE Visualization '90. 129-134.
10. Hibbard, B. 2001. Super-intelligent machines. Computer Graphics 35(1), pp. 11-13.
11. Hibbard, B. 2008. The technology of mind and a new social contract. J. Evolution and Technology 17(1), pp. 13-22.
12. Hibbard, B. 2009. Temptation. Rejected for the AGI-09 Workshop on the Future of AI. https://sites.google.com/site/whibbard/g/hibbard_agi09_workshop.pdf
13. Hibbard, B. 2012a. Model-based utility functions. J. Artificial General Intelligence 3(1), pp. 1-24.
14. Hibbard, B. 2012b. Avoiding unintended AI behavior. In: Bach, J., and Iklé, M. (eds) AGI 2012. LNCS (LNAI), this volume. Springer, Heidelberg. https://sites.google.com/site/whibbard/g/hibbard_agi12a.pdf
15. Hutter, M. 2005. Universal artificial intelligence: sequential decisions based on algorithmic probability. Springer, Heidelberg.
16. Hutter, M. 2009a. Feature reinforcement learning: Part I. Unstructured MDPs. J. Artificial General Intelligence 1, pp. 3-24.
17. Hutter, M. 2009b. Feature dynamic Bayesian networks. In: Goertzel, B., Hitzler, P., and Hutter, M. (eds) AGI 2009. Proc. Second Conf. on AGI, pp. 67-72. Atlantis Press, Amsterdam.
18. Kurzweil, R. 2005. The singularity is near. Penguin, New York.
19. Li, M., and Vitanyi, P. 1997. An introduction to Kolmogorov complexity and its applications. Springer, Heidleberg.
20. Lloyd, S. Computational Capacity of the Universe. Phys.Rev.Lett. 88 (2002) 237901.
21. Muehlhauser, L., and Helm, L. 2012. The singularity and machine ethics. In Eden, Søraker, Moor, and Steinhart (eds) The Singularity Hypothesis: a Scientific and Philosophical Assessment. Springer, Heidleberg.
22. Omohundro, S. 2008. The basic AI drive. In Wang, P., Goertzel, B., and Franklin, S. (eds) AGI 2008. Proc. First Conf. on AGI, pp. 483-492. IOS Press, Amsterdam.
23. Puterman, M. L. 1994. Markov Decision Processes - Discrete Stochastic Dynamic Programming. Wiley, New York.
24. Ring, M., and Orseau, L. 2011. Delusion, survival, and intelligent agents. In: Schmidhuber, J., Thórisson, K.R., and Looks, M. (eds) AGI 2011. LNCS (LNAI), vol. 6830, pp. 11-20. Springer, Heidelberg.
25. Sutton, R.S., and Barto, A.G. 1998. Reinforcement learning: an introduction. MIT Press.
26. Waser, M. 2010. Designing a safe motivational system for intelligent machines. In: Baum, E., Hutter, M., and Kitzelmann, E. (eds) AGI 2010. Proc. Third Conf. on AGI, pp 170-175. Atlantis Press, Amsterdam.
27. Waser, M. 2011. Rational universal benevolence: simpler, safer, and wiser than "friendly AI." In: Schmidhuber, J., Thórisson, K.R., and Looks, M. (eds) AGI 2011. LNCS (LNAI), vol. 6830, pp. 153-162. Springer, Heidelberg.
28. Yudkowsky, E. 2004. http://www.sl4.org/wiki/CoherentExtrapolatedVolition