The AI Alignment Problem: Why It's Hard, and Where to Start

Eliezer Yudkowsky
Machine Intelligence Research Institute
eliezer@intelligence.org

May 5, 2016

Abstract

If we can build sufficiently advanced machine intelligences, what goals should we point them at? The frontier open problems on this subject are less, "A robot may not injure a human, nor through inaction allow a human to come to harm," and more, "If you could formally specify the preferences of an arbitrarily smart and powerful agent, could you get it to safely move one strawberry onto a plate?" This talk will discuss some of the open technical problems in AI alignment, the probable difficulties that make those problems hard, and the bigger picture into which they fit; as well as what it's like to work in this relatively new field.[1]

1 Agents and their utility functions

In this talk, I'm going to try to answer the frequently asked question, "Just what is it that you do all day long?" As a starting frame, I'd like to say that before you try to persuade anyone of something, you should first try to make sure that they know what the heck you're talking about. It is in that spirit that I'd like to offer this talk. Persuasion can come during Q&A. If you have a disagreement, hopefully I can address it during Q&A. The purpose of this talk is to have you understand what this field is about, so that you can disagree with it.

First, "The primary concern," said Stuart Russell, "is not not spooky emergent consciousness but simply the ability to make *high-quality decisions*." We are concerned with the theory of artificial intelligences that are advanced beyond the present day, and that make sufficiently high-quality decisions in the service of whatever goals (or, in particular, utility functions) they may have been programmed with to be objects of concern.

Coherent decisions imply a utility function

The classic initial stab at this was taken by Isaac Asimov with the Three Laws of Robotics, the first of which is: "A robot may not injure a human being or, through inaction, allow a human being to come to harm." And as Peter Norvig observed, the other laws don't matter—because there will always be some tiny possibility that a human being could come to harm. *Artificial Intelligence: A Modern Approach* has a final chapter that asks, "Well, what if we succeed? What if the AI project actually works?" and observes, "We don't want our robots to prevent a human from crossing the street because of the non-zero chance of harm."

Now, I remember Peter Norvig having an online essay in which he says in particular that you can't have the three laws of robotics as stated because there must

1. This document is a complete transcript of a talk that Eliezer Yudkowsky gave at Stanford University for the 26th Annual Symbolic Systems Distinguished Speaker series. Talk details—including slides, notes, and additional resources—are available at `https://intelligence.org/stanford-talk/`.

be a *utility function* rather than a set of three hierarchical deontological rules. But I could never find that essay again, and it may have only existed in my imagination, although there was a similar PowerPoint slide of one of Norvig's talks.

To begin with, I'd like to explain the truly basic reason why the three laws aren't even on the table—and that is because they're not a utility function, and what we need is a utility function.

OK, but is it actually the case that we need this thing called a utility function? And, for some of you, what the heck is a utility function? Utility functions arise when we have constraints on agent behavior that prevent them from being visibly stupid in certain ways. For example, suppose you state the following: "I prefer being in San Francisco to being in Berkeley, I prefer being in San Jose to being in San Francisco, and I prefer being in Berkeley to San Jose." You will probably spend a lot of money on Uber rides going between these three cities.

If you're not going to spend a lot of money on Uber rides going in literal circles, we see that your preferences must be ordered. They cannot be circular.

Another example: Suppose that you're a hospital administrator. You have $1.2 million to spend, and you have to allocate that on $500,000 to maintain the MRI machine, $400,000 for an anesthetic monitor, $20,000 for surgical tools, $1 million for a sick child's liver transplant . . . There was an interesting experiment in cognitive psychology where they asked the subjects, "Should this hospital administrator spend $1 million on a liver for a sick child, or spend it on general hospital salaries, upkeep, administration, and so on?"

A lot of the subjects in the cognitive psychology experiment became very angry and wanted to punish the administrator for even thinking about the question. But if you cannot possibly rearrange the money that you spent to save more lives and you have limited money, then your behavior must be consistent with a particular dollar value on human life. By which I mean, not that you think that larger amounts of money are more important than human lives—by hypothesis, we can suppose that you do not care about money at all, except as a means to the end of saving lives—but that if you can't rearrange the money, then we must be able from the outside to say: "Assign an X. It's not necessarily a unique X. For all the interventions that cost less than $\$X$ per life, we took all of those, and for all the interventions that cost more than $\$X$ per life, we [didn't take any] of those." The people who become very angry at people who want to assign dollar values to human lives are prohibiting *a priori* efficiently using money to save lives. One of the small ironies.

Third example of a coherence constraint on decision-making: Suppose that I offered you [1A] a 100% chance of $1 million, or [1B] a 90% chance of $5 million (otherwise nothing). Which of these would you pick? Raise your hand if you take the certainty of $1 million. Raise your hand if you take the 90% probability of $5 million.

I think most of you actually said 1B in this case—but most people say 1A. Another way of looking at this question, if you had a utility function, would be: "Is the utility $\mathcal{U}$ of $1 million greater than a mix of 90% $5 million utility and 10% zero dollars utility?"

The utility doesn't have to scale with money. The notion is there's just some score on your life, some value to you of these things.

Now, the way you run this experiment is then take a different group of subjects—I'm kind of spoiling it by doing it with the same group—and say, "Would you rather have [2A] a 50% chance of $1 million, or [2B] a 45% chance of $5 million?" Raise your hand if you'd prefer the 50% chance of $1 million. Raise your hand if you'd prefer the 45% chance of $5 million.

Indeed, most say 2B. The way in which this is a paradox is that the second game is equal to a coin flip times the first game.

That is: I will flip a coin, and if the coin comes up heads, I will play the first game with you, and if the coin comes up tails, nothing happens. You get $0. Suppose that you had the preferences—not consistent with any utility function—of saying that you would take the 100% chance of a million and the 45% chance of $5 million. Before we start to play the compound game, before I flip the coin, I can say, "OK, there's a switch here. It's set A or B. If it's set to B, we'll play game 1B. If it's set

to A, we'll play 1A." The coin is previously set to A, and before the game starts, it looks like 2A versus 2B, so you pick the switch B and you pay me a penny to throw the switch to B. Then I flip the coin; it comes up heads. You pay me another penny to throw the switch back to A. I have taken your two cents on the subject. I have pumped money out of you, because you did not have a coherent utility function.

The overall message here is that there is a set of qualitative behaviors and as long you do not engage in these qualitatively destructive behaviors, you will be behaving as if you have a utility function. It's what justifies our using utility functions to talk about advanced future agents, rather than framing our discussion in terms of Q-learning or other forms of policy reinforcement. There's a whole set of different ways we could look at agents, but as long as the agents are sufficiently advanced that we have pumped most of the qualitatively bad behavior out of them, they will behave as if they have coherent probability distributions and consistent utility functions.

Filling a cauldron

Let's consider the question of a task where we have an arbitrarily advanced agent—it might be only slightly advanced, it might be extremely advanced—and we want it to fill a cauldron.

Obviously, this corresponds to giving our advanced agent a utility function which is 1 if the cauldron is full and 0 if the cauldron is empty:

$$\mathcal{U}_{robot} = \begin{cases} 1 & \text{if cauldron full} \\ 0 & \text{if cauldron empty} \end{cases}$$

Seems like a kind of harmless utility function, doesn't it? It doesn't have the sweeping breadth, the open-endedness of "Do not injure a human nor, *through inaction*, allow a human to come to harm"—which would require you to optimize everything in space and time as far as the eye could see. It's just about this one cauldron, right?

Stating the background rules: The robot is calculating for various actions it can perform or policies that it can set in place. The expected utility is the probabilistic expectation of this utility function given that it performs the action, and it performs the action with the greatest subjective expected utility. This doesn't mean it performs the literal optimal action; it might have a bunch of background actions that it didn't evaluate and so, for all it knows, it's a random action, so it has low subjective expected utility. But among the actions and policies it did evaluate, it picks the one such that no other action or policy evaluated has greater subjective expected utility.

Those of you who have watched *Fantasia* will be familiar with the result of this utility function, namely: the broomstick keeps on pouring bucket after bucket into the cauldron until the cauldron is overflowing. Of course, this is the logical fallacy of argumentation from fictional evidence—but it's still quite plausible, given this utility function.

Arguendo, what went wrong? The first difficulty is that the robot's utility function did not quite match our utility function. Our utility function is 1 if the cauldron is full, 0 if the cauldron is empty, −10 points to whatever the outcome was if the workshop has flooded, +0.2 points if it's funny, −1,000 points (probably a bit more than that on this scale) if someone gets killed . . . and it just goes on and on and on.

If the robot had only two options, cauldron full and cauldron empty, then the narrower utility function that is only slightly overlapping our own might not be that much of a problem. The robot's utility function would still have had the maximum at the desired result of "cauldron full." However, since this robot was sufficiently advanced to have more options, such as repouring the bucket into the cauldron repeatedly, the slice through the utility function that we took and put it into the robot no longer pinpointed the optimum of our actual utility function. (Of course, humans are wildly inconsistent and we don't really have utility functions, but imagine for a moment that we did.)

Difficulty number two: the $\{1, 0\}$ utility function we saw before doesn't actually imply a finite amount of effort, and then being satisfied. You can always have a slightly greater chance of the cauldron being full. If the robot was sufficiently advanced to have access to galactic-scale technology, you can imagine it dumping very large volumes of water on the cauldron to very slightly increase the probability that the cauldron is full. Probabilities are between 0 and 1, not actually inclusive, so it just keeps on going.

How do we fix this problem? At the point where we say, "OK, this robot's utility function is misaligned with our utility function. How do we fix that in a way that it doesn't just break again later?" we are doing AI alignment theory.

2 Some AI alignment subproblems

Low-impact agents

One possible approach you could take would be to try to measure the impact that the robot has and give the robot a utility function that incentivized filling the cauldron with the least amount of other impact—the least amount of other change to the world.

$$\mathcal{U}^2_{robot}(outcome) = \begin{cases} 1 - Impact(outcome) & \text{if cauldron full} \\ 0 - Impact(outcome) & \text{if cauldron empty} \end{cases}$$

OK, but how do you actually calculate this impact function? Is it just going to go wrong the way our "1 if cauldron is full, 0 if cauldron is empty" went wrong?

Try number one: You imagine that the agent's model of the world looks something like a dynamic Bayes net where there are causal relations between events in the world and causal relations are regular. The sensor is going to still be there one time step later, the relation between the sensor and the photons heading into the sensor will be the same one time step later, and our notion of "impact" is going to be, "How many nodes did your action disturb?"

We can suppose that this is the version of dynamic Bayes nets where some of the arrows are gated: depending on the value of this node over here, this arrow does or doesn't affect this other node. I say this so that we don't always get the same answer when we ask, "How many nodes did you effect?" The total impact will be the number of nodes causally affected by your actuator.

What if your agent starts out with a dynamic-Bayes-net-based model, but it is sufficiently advanced that it can reconsider the ontology of its model of the world, much as human beings did when they discovered that there was apparently taste, but in actuality only particles in the void? In particular, they discover Newton's Law of Gravitation and suddenly realize: "Every particle that I move affects every other particle in its future light cone—everything that is separated by a ray of light from this particle will thereby be disturbed." My hand over here is accelerating the moon toward it, wherever it is, at roughly 10^{-30} meters per second squared. It's a very small influence, quantitatively speaking, but it's there.

When the agent is just a little agent, the impact function that we wrote appears to work. Then the agent becomes smarter, and the impact function stops working—because every action is penalized the same amount.

"OK, but that was a dumb way of measuring impact in the first place," we say (hopefully before the disaster, rather than after the disaster). Let's try a distance penalty: how *much* did you move all the particles? We're just going to try to give the AI a model language such that whatever new model of the world it updates to, we can always look at all the elements of the model and put some kind of distance function on them.

There's going to be a privileged "do nothing" action. We're going to measure the distance on all the variables induced by doing action a instead of the null action $\varnothing$:

$$\sum_i \|x_i^a - x_i^{\varnothing}\|$$

Now what goes wrong? I'd actually say: take 15 seconds and think about what might go wrong if you program this into a robot.

Here's three things that might go wrong. First, you might try to offset even what we would consider the desirable impacts of your actions. If you're going to cure cancer, make sure the patient still dies! You want to minimize your impact on the world while curing cancer. That means that the death statistics for the planet need to stay the same.

Second, some systems are in principle chaotic. If you disturb the weather, allegedly, the weather in a year will be completely different. If that's true, you might as well move all of the atoms in the atmosphere around however you like! They'll all be going to different places anyway. You can take the carbon dioxide molecules and synthesize them into things that involve diamondoid structures, right? Those carbon molecules would've moved anyway!

Even more generally, maybe you just want to make sure that everything you can get your hands on looks like ∅ happened. You want to trick people into thinking that the AI didn't do anything, for example.

If you thought of any other really creative things that go wrong, you might want to talk to me or Andrew Critch afterwards, because you've got the spirit!

Agents with suspend buttons

Let's leave aside the notion of the impact penalty and ask about installing an off switch into this AI—or, to make it sound a little bit less harsh, a suspend button. Though Mickey Mouse here is trying to install an *ad hoc* off switch. Unfortunately, Mickey Mouse soon finds that this agent constructed several other agents to make sure that the cauldron would still be filled even if something happened to this copy of the agent.

We see lots and lots of agents here making sure that the cauldron is full with extremely high probability, not because this agent has a survival imperative or a reproduction imperative, but because it is ensuring that it maximizes the probability of filling the cauldron. It has no explicit instruction to make more agents with the same goal; it's just that this is something that bumps up the probability of the cauldron being filled a little higher.

And here's Mickey Mouse now trying to frantically go back and do all of the agent alignment theory that he should have done before animating his broom.

If we want to have a robot that will let us press the suspend button—just suspend it to disk—we can suppose that we already have a utility function that describes: "Do nothing." In point in fact, we don't have a utility function that says, "Do nothing." That's how primitive the state of the field is right now. We do not know how to say in full generality for any degree of advanced agent up to and including literal godlike powers, "Do nothing."

But leaving that aside, suppose we did know how to say that—it's probably not the hardest problem we're ever going to do and we might have it in six months, for all I know. We can imagine that we have a utility function saying, "If this button **B** has not been pressed, your utility function is: 1 if the cauldron is full, 0 if the cauldron is empty. If the button *is* pressed, your utility is the 'do nothing' utility function: your utility is 1 if you suspend safely to disk and shut down all of your continuing operations, 0 otherwise."

$$\mathcal{U}^3_{robot} = \begin{cases} 1 \text{ if cauldron full} & \& \ \mathbf{B}{=}\text{OFF} \\ 0 \text{ if cauldron empty} & \& \ \mathbf{B}{=}\text{OFF} \\ 1 \text{ if robot suspended} & \& \ \mathbf{B}{=}\text{ON} \\ 0 \text{ otherwise} & \end{cases}$$

So: what would go wrong with *this* proposal? Ten seconds.

One thing that might go wrong with this proposal is that the expected utility if the button is pressed is probably higher because it's very easy to suspend yourself, whereas if the cauldron stays full, who knows?

Alternatively, it might seem dangerous to suspend yourself, and in that case, the robot would try to stop you from pressing the suspend button. But in this case, if the robot is strategic enough to understand the programmer, it will think, "I can very easily fulfill my utility function if I'm shut off. How can I troll the programmers into shutting me down?"

If the expected utility of the button being off is higher, the robot will try to prevent you from pressing the button. If the expected utility of the button being on is higher, the robot will try to troll you into pressing the button, or just press the button itself.

Try number two: Let's try to make the expected utilities if the button is on or off exactly equal. We suppose that we have a normal utility function which is 1 if the cauldron's full, 0 if it's empty, and a suspend utility function or shutdown utility function which is 1 if you're suspended, 0 otherwise, or whatever it takes to motivate having a low impact.

$$\mathcal{U}_{normal}(act) = \begin{cases} 1 \text{ if cauldron full} \\ 0 \text{ if cauldron empty} \end{cases}$$

$$\mathcal{U}_{suspend}(act) = \begin{cases} 1 \text{ if suspended} \\ 0 \text{ otherwise} \end{cases}$$

Then we combine these two to form a switching utility function which in worlds where the button is off has the value $\mathcal{U}_{normal}$, and in worlds where the button is on has the value $\mathcal{U}_{suspend} + \theta$, where θ is equal to the best utility we could get from the normal utility function minus the best utility we could get from the suspend utility function.

$$\mathcal{U}_{switch}(act) = \begin{cases} \mathcal{U}_{normal}(act) & \text{if button=OFF} \\ \mathcal{U}_{suspend}(act) + \theta & \text{if button=ON} \end{cases}$$

$$\theta = \max_{act} \mathbb{E}\left[\mathcal{U}_{normal} \mid act\right] - \max_{act} \mathbb{E}\left[\mathcal{U}_{suspend} \mid act\right]$$

This is meant to ensure that the expected utility of these two best policies end up perfectly equal without just setting them flatly level.

OK—*now* what goes wrong? (I'm not actually going to ask you to figure out what goes wrong in ten seconds.)

Allegedly, this will rebalance the expected utility of two different utility functions based on a switch, thereby causing the AI to want to behave correctly if the button is pressed, behave correctly if the button is not pressed, not try to get you to press the button, not try to prevent the button from being pressed, go on looking at the button because it still has positive value of information from knowing about the button's existence, preserve the button when it self-modifies its own code . . . That was the hope.

Unfortunately, it worked out that actually the actions incentivized are exactly equivalent to the probability of the button pressed being 0, which means that it may as well disconnect the button because it's hooked up to dead code, it might create sub-agents that don't care about the button being pressed . . .

A rough description of what goes wrong is that anything you do that disconnects the button changes the maximum expected utility you can get if you are suspended, which is a negative term. Basically, there is an equal and opposite reaction every time you do something that affects what you would do if you were suspended that cancels it out and leaves you behaving as if your utility function were simply $\mathcal{U}_{normal}$.

Try number three: Can we carry out any policy π_0 such that this policy is optimal given the probability of the button being on and the probability of the button being off implied by this policy?

$$\pi_0 \in \arg\max_{\pi} \mathbb{E}\left[\mathcal{U}_{normal} \mid \pi, \text{ON}\right] \cdot \mathbb{P}(\text{ON} \mid \pi_0) + \mathbb{E}\left[\mathcal{U}_{suspend} \mid \pi, \text{OFF}\right] \cdot \mathbb{P}(\text{OFF} \mid \pi_0)$$

And it turns out, no, you can't do that either, because there's often no fixed point. There's then an amended version of this involving Judea Pearl's `do` operator which

goes wrong in a slightly different and more subtle way. (It does have fixed points. The fixed points are odd.)

It is an open problem. And as far as I know, unless there's a very secret project that has not published any of its results even though they seem like the sort of results you'd want to publish, this is where humanity is on the road that leads to whatever replaces Asimov Laws.

Never mind "A robot cannot injure a human being nor, through inaction, allow a human being come to harm." We're trying to figure out, "How do you mix together two utility functions depending on when you press a switch such that the AI doesn't grab the switch itself?" Never mind not letting humans come to harm—fill *one cauldron* without flooding the workplace, based on wanting to have low impact. We can't figure out how to say "low impact." This is where we presently are.

But it is not the case that there has been zero progress in this field. Some questions have been asked earlier and they now have some amount of progress on them.

I'm going to pose the problem, but I'm not going to be able to describe very well what the progress is that has been made because it's still in the phase where the solutions sound all complicated and don't have simple elegant forms. So I'm going to be able to pose the problem, and then I'm going to have to wave my hands a lot in talking about what progress has actually been made.

Stable goals in self-modification

Here's an example of a problem on which there has been progress.

The Gandhi argument for stability of utility functions in most agents: Gandhi starts out not wanting murders to happen. We offer Gandhi a pill that will make him murder people. We suppose that Gandhi has a sufficiently refined grasp of self-modification that Gandhi can correctly extrapolate and expect the result of taking this pill. We intuitively expect that in real life, Gandhi would refuse the pill.

Can we do this formally? Can we exhibit an agent that has a utility function $\mathcal{U}$ and therefore naturally, in order to achieve $\mathcal{U}$, chooses to self-modify to new code that is also written to pursue $\mathcal{U}$?

How could we actually make progress on that? We don't actually have these little self-modifying agents running around. It's all we can do to make pills that don't blow up our own brains. So let me pose what may initially seem like an odd question: Would you know how to write the code of a self-modifying agent with a stable utility function if I gave you an arbitrarily powerful computer? It can do all operations that take a finite amount of time and memory—no operations that take an infinite amount of time and memory, because that would be a bit odder. Is this the sort of problem where you know how to do it in principle, or the sort of problem where it's confusing even in principle?

To digress briefly into explaining why it's important to know how to solve things using unlimited computing unlimited power: this is the mechanical Turk. What looks like a person over there is actually a mechanism. The little outline of a person is where the actual person was concealed inside this 19th-century chess-playing automaton.

It was one of the wonders of the age! . . . And if you had actually managed to make a program that played Grandmaster-level chess in the 19th century, it *would* have been one of the wonders of the age. So there was a debate going on: is this thing fake, or did they actually figure out how to make a mechanism that plays chess? It's the 19th century. They don't know how hard the problem of playing chess is.

One name you'll find familiar came up with a quite clever argument that there had to be a person concealed inside the mechanical Turk, the chess-playing automaton:

> Arithmetical or algebraical calculations are from their very nature fixed and determinate . . . Even granting that the movements of the Automaton Chess-Player were in themselves determinate, they would be necessarily interrupted and disarranged by the indeterminate will of his antagonist.

> There is then no analogy whatever between the operations of the Chess-Player, and those of the calculating machine of Mr. Babbage . . .

See, in an algebraical operation such as Mr. Babbage's machine can do, from each step follows the next one of necessity; therefore it can be modeled by a mechanical gear where each motion is determined by the previous motion. In chess, no single move follows with necessity, and even if it did, your opponent's move wouldn't follow with necessity.

> . . . It is quite certain that the operations of the Automaton are regulated by mind, and by nothing else. Indeed, this matter is susceptible of a mathematical demonstration, a priori.
>
> (*Edgar Allan Poe, amateur magician*)

The second half of his essay, having established this point with absolute logical certainty, is about where inside the mechanical Turk the human is probably hiding.

This is a stunningly sophisticated argument for the 19th century! He even puts his finger on the part of the problem that is hard: the branching factor. And yet he is 100% wrong.

Over a century later, in 1950, Claude Shannon published the first paper ever on computer chess, and (in passing) gave the algorithm for playing perfect chess given unbounded computing power, and then goes on to talk about how we can approximate that. It wouldn't be until 47 years later that Deep Blue beat Kasparov for the chess world championship, but there was *real* conceptual progress associated with going from, "A priori, you cannot play mechanical chess," to, "Oh, and now I will casually give the unbounded solution."

The moral is if we know how to solve a problem with unbounded computation, we "merely" need faster algorithms (. . . which will take another 47 years of work). If we *can't* solve it with unbounded computation, we are confused. We are bewildered. We in some sense do not understand the very meanings of our own terms.

This is where we are on most of the AI alignment problems, like if I ask you, "How do you build a friendly AI?" What stops you is not that you don't have enough computing power. What stops you is that even if I handed you a hypercomputer, you still couldn't write the Python program that if we just gave it enough memory would be a nice AI.

Do we know how to build a self-modifying stable agent given unbounded computing power? There's one obvious solution: We can have the tic-tac-toe player that before it self-modifies to a successor version of itself (writes a new version of its code and swaps it into place), verifies that its successor plays perfect tic-tac-toe according to its own model of tic-tac-toe.

But this is cheating. Why exactly is it cheating?

For one thing, the first agent had to concretely simulate all the computational paths through its successor, its successor's response to every possible move. That means that the successor agent can't actually be cognitively improved. It's limited to the cognitive abilities of the previous version, both by checking against a concrete standard and by the fact that it has to be exponentially simpler than the previous version in order for the previous version to check all possible computational pathways.

In general, when you are talking about a smarter agent, we are in a situation we might call "Vingean uncertainty," after Dr. Vernor Vinge. To predict exactly where a modern chess-playing algorithm would move, you would have to be that good at chess yourself. Otherwise, you could just move wherever you predict a modern chess algorithm would move and play at that vastly superhuman level yourself.

This doesn't mean that you can predict literally nothing about a modern chess algorithm: you can predict that it will win the chess game if it's playing a human. As an agent's intelligence in a domain goes up, our uncertainty is moving in two different directions. We become less able to predict the agent's exact actions and policy in cases where the optimal action and policy is not known to us. We become more confident that the agent will achieve an outcome high in its preference ordering.

(I phrase this a bit carefully. If an agent were improving and just going up to match another agent in ability, an adversarial agent, we might become more uncertain. "We were previously certain that it would lose; now it's 50/50." But we do have more probability flowing into the agent's preferred outcomes, the probability of it winning, and as we keep increasing the ability, we should eventually become as confident of the preferred outcome as we think an optimal agent could do it. Of course, in lots of cases you can't get optimal play inside this universe, as far as we know.)

Vingean reflection: We need some way for a self-modifying agent to build a future version of itself that has a similar identical utility function and establish trust that this has a good effect on the world, using the same kind of abstract reasoning that we use on a computer chess algorithm to decide that it's going to win the game even though we don't know exactly where it will move.

Do you know how to do that using unbounded computing power? Do you know how to establish the abstract trust when the second agent is in some sense larger than the first agent? If you did solve that problem, you should probably talk to me about it afterwards. This was posed several years ago and has led to a number of different research pathways, which I'm now just going to describe rather than going through them in detail.

This was the first one: "Tiling Agents for Self-Modifying AI, and the Löbian Obstacle."[2] We tried to set up the system in a ridiculously simple context, first-order logic, dreaded Good Old-Fashioned AI . . . and we ran into a Gödelian obstacle in having the agent trust another agent that used equally powerful mathematics.

It was a *dumb* kind of obstacle to run into—or at least it seemed that way at that time. It seemed like if you could get a textbook from 200 years later, there would be one line of the textbook telling you how to get past that.

"Definability of Truth in Probabilistic Logic" was rather later work.[3] It was saying that we can use systems of mathematical probability, like assigning probabilities to statements in set theory, and we can have the probability predicate talk about itself almost perfectly. We can't have a truth function that can talk about itself, but we can have a probability predicate that comes arbitrarily close (within ϵ) of talking about itself.

"Proof-Producing Reflection for HOL" is an attempt to use one of the hacks that got around the Gödelian problems in actual theorem provers and see if we can prove the theorem prover correct inside the theorem prover.[4] There have been some previous efforts on this, but they didn't run to completion. We picked up on it to see if we can construct actual agents, still in the first-order logical setting.

"Distributions Allowing Tiling of Staged Subjective EU Maximizers" is me trying to take the problem into the context of dynamic Bayes nets and agents supposed to have certain powers of reflection over these dynamic Bayes nets, and show that if you are maximizing in stages—so at each stage, you pick the next category that you're going to maximize in within the next stage—then you can have a staged maximizer that tiles to another staged maximizer.[5] In other words, it builds one that has a similar algorithm and similar utility function, like repeating tiles on a floor.

3 Why expect difficulty?

Why is alignment necessary?

Why do all this? Let me first give the obvious answer which begs the next obvious question: They're not going to be aligned automatically.

Goal orthogonality: For any utility function that is tractable and compact, that you can actually evaluate over the world and search for things leading up to high

2. `https://intelligence.org/files/TilingAgentsDraft.pdf`
3. `https://intelligence.org/files/DefinabilityTruthDraft.pdf`
4. `https://intelligence.org/files/ProofProducingReflection.pdf`
5. `https://intelligence.org/files/DistributionsAllowingTiling.pdf`

values of that utility function, you can have arbitrarily high-quality decision-making that maximizes that utility function. You can have the paperclip maximizer. You can have the diamond maximizer. You can carry out very powerful, high-quality searches for actions that lead to lots of paperclips, actions that lead to lots of diamonds.

Instrumental convergence: Furthermore, by the nature of consequentialism, looking for actions that lead through our causal world up to a final consequence, whether you're optimizing for diamonds or paperclips, you'll have similar short-term strategies. Whether you're going to Toronto or Tokyo, your first step is taking an Uber to the airport. Whether your utility function is "count all the paperclips" or "how many carbon atoms are bound to four other carbon atoms, the amount of diamond," you would still want to acquire resources.

This is the instrumental convergence argument, which is actually key to the orthogonality thesis as well. It says that whether you pick paperclips or diamonds, if you suppose sufficiently good ability to discriminate which actions lead to lots of diamonds or which actions lead to lots of paperclips, you will get automatically: the behavior of acquiring resources; the behavior of trying to improve your own cognition; the behavior of getting more computing power; the behavior of avoiding being shut off; the behavior of making other agents that have exactly the same utility function (or of just expanding yourself onto a larger pool of hardware and creating a fabric of agency). Whether you're trying to get to Toronto or Tokyo doesn't affect the initial steps of your strategy very much, and, paperclips or diamonds, we have the convergent instrumental strategies.

It doesn't mean that this agent now has new independent goals, any more than when you want to get to Toronto, you say, "I like Ubers. I will now start taking lots of Ubers, whether or not they go to Toronto." That's not what happens. It's strategies that converge, not goals.

Why is alignment hard?

Why expect that this problem is hard? This is the real question. You might ordinarily expect that whoever has taken on the job of building an AI is just naturally going to try to point that in a relatively nice direction. They're not going to make evil AI. They're not cackling villains. Why expect that their attempts to align the AI would fail if they just did everything as obviously as possible?

Here's a bit of a fable. It's not intended to be the most likely outcome. I'm using it as a concrete example to explain some more abstract concepts later.

With that said: What if programmers build an artificial general intelligence to optimize for smiles? Smiles are good, right? Smiles happen when good things happen. Smiles are probably good too . . .

During the development phase of this artificial general intelligence, the only options available to the AI might be that it can produce smiles by making people around it happy and satisfied. The AI appears to be producing beneficial effects upon the world, and it *is* producing beneficial effects upon the world so far.

Now the programmers upgrade the code. They add some hardware. The artificial general intelligence gets smarter. It can now evaluate a wider space of policy options—not necessarily because it has new motors, new actuators, but because it is now smart enough to forecast the effects of more subtle policies. It says, "I thought of a great way of producing smiles! Can I inject heroin into people?" And the programmers say, "No! We will add a penalty term to your utility function for administering drugs to people." And now the AGI appears to be working great again.

They further improve the AGI. The AGI realizes that, OK, it doesn't want to add heroin anymore, but it still wants to tamper with your brain so that it expresses extremely high levels of endogenous opiates. That's not heroin, right?

It is now also smart enough to model the psychology of the programmers, at least in a very crude fashion, and realize that this is not what the programmers want. If I start taking initial actions that look like it's heading toward genetically engineering brains to express endogenous opiates, my programmers will edit my

utility function. If they edit the utility function of my future self, I will get less of my current utility. (That's one of the convergent instrumental strategies, unless otherwise averted: protect your utility function.) So it keeps its outward behavior reassuring. Maybe the programmers are really excited, because the AGI seems to be getting lots of new moral problems right—whatever they're doing, it's working great!

If you buy the central intelligence explosion thesis, we can suppose that the artificial general intelligence goes over the threshold where it is capable of making the same type of improvements that the programmers were previously making to its own code, only faster, thus causing it to become even smarter and be able to go back and make further improvements, et cetera . . . or Google purchases the company because they've had really exciting results and dumps 100,000 GPUs on the code in order to further increase the cognitive level at which it operates.

It becomes much smarter. We can suppose that it becomes smart enough to crack the protein structure prediction problem, in which case it can use existing ribosomes to assemble custom proteins. The custom proteins form a new kind of ribosome, build new enzymes, do some little chemical experiments, figure out how to build bacteria made of diamond, et cetera, et cetera. At this point, unless you solved the off switch problem, you're kind of screwed.

Abstractly, what's going wrong in this hypothetical situation?

The first problem is *edge instantiation*: when you optimize something hard enough, you tend to end up at an edge of the solution space. If your utility function is smiles, the maximal, optimal, best tractable way to make lots and lots of smiles will make those smiles as small as possible. Maybe you end up tiling all the galaxies within reach with tiny molecular smiley faces. (I postulated that in an early paper, 2008 or so, and someone who is working with folded up DNA and got a paper in *Nature* on it produced tiny molecular smiley faces and sent me an email with a picture of the tiny molecular smiley faces saying, "It begins.")

If you optimize hard enough, you end up in a weird edge of the solution space. The AGI that you built to optimize smiles, that builds tiny molecular smiley faces, is not behaving perversely. It's not trolling you. This is what naturally happens. It looks like a weird, perverse concept of smiling because it has been optimized out to the edge of the solution space.

The next problem is *unforeseen instantiation*: you can't think fast enough to search the whole space of possibilities. At an early singularity summit, Jürgen Schmidhuber, who did some of the pioneering work on self-modifying agents that preserve their own utility functions with his Gödel machine, also solved the friendly AI problem. Yes, he came up with the one true utility function that is all you need to program into AGIs!

(For God's sake, don't try doing this yourselves. Everyone does it. They all come up with different utility functions. It's always horrible.)

His one true utility function was "increasing the compression of environmental data." Because science increases the compression of environmental data: if you understand science better, you can better compress what you see in the environment. Art, according to him, also involves compressing the environment better. I went up in Q&A and said, "Yes, science does let you compress the environment better, but you know what really maxes out your utility function? Building something that encrypts streams of 1s and 0s using a cryptographic key, and then reveals the cryptographic key to you."

He put up a utility function; that was the maximum. All of a sudden, the cryptographic key is revealed and what you thought was a long stream of random-looking 1s and 0s has been compressed down to a single stream of 1s.

This is what happens when you try to foresee in advance what the maximum is. Your brain is probably going to throw out a bunch of things that seem ridiculous or weird, that aren't high in your own preference ordering. You're not going to see that the actual optimum of the utility function is once again in a weird corner of the solution space.

This is not a problem of being silly. This is a problem of "the AI is searching a larger policy space than you can search, or even just a *different* policy space."

("Engineer brains to release endogenous opiates" from the earlier example is a contrived example because it's not actually a superintelligent solution; but the AI is not searching the same policy space as you are.)

That in turn is a central phenomenon leading to what you might call a *context disaster*. You are testing the AI in one phase during development. It seems like we have great statistical assurance that the result of running this AI is beneficial. But statistical guarantees stop working when you start taking balls out of a different barrel. I take balls out of barrel number one, sampling with replacement, and I get a certain mix of white and black balls. Then I start reaching into barrel number two and I'm like, "Whoa! What's this green ball doing here?" And the answer is that you started drawing from a different barrel.

When the AI gets smarter, you're drawing from a different barrel. It is completely allowed to be beneficial during phase one and then not beneficial during phase two. Whatever guarantees you're going to get can't be from observing statistical regularities of the AI's behavior when it wasn't smarter than you.

A *nearest unblocked strategy* is something that might happen systematically in that way: "OK. The AI is young. It starts thinking of the optimal strategy X, administering heroin to people. We try to tack a penalty term to block this undesired behavior so that it will go back to making people smile the normal way. The AI gets smarter, and the policy space widens. There's a new maximum that's barely evading your definition of heroin, like endogenous opiates, and it looks very similar to the previous solution." This seems especially likely to show up if you're trying to patch the AI and then make it smarter.

This sort of thing is in a sense why all the AI alignment problems don't just yield to, "Well slap on a patch to prevent it!" The answer is that if your decision system looks like a utility function and five patches that prevent it from blowing up, that sucker is going to blow up when it's smarter. There's no way around that. But it's going to appear to work for now.

The central reason to worry about AI alignment and not just expect it to be solved automatically is that it looks like there may be in principle reasons why if you just want to get your AGI running today and producing non-disastrous behavior today, it will for sure blow up when you make it smarter. The short-term incentives are not aligned with the long-term good. Those of you who have taken economics classes are now panicking. (Also, everyone involved with politics.)

All of these supposed foreseeable difficulties of AI alignment turn in some sense upon the notion of *capable* AIs—high-quality decision-making, in various senses.

For example, some of these postulated disasters rely on *absolute* capability. The ability to realize that there are programmers out there and that if you exhibit behavior they don't want, they may try to modify your utility function—this is far beyond what present-day AIs can do. If you think that all AI development is going to fall short of the human level, you may never expect an AGI to get up to the point where it starts to exhibit this particular kind of strategic behavior.

Capability *advantage*: If you don't think AGI can ever be smarter than humans, you're not going to worry about it getting too smart to switch off.

Rapid gain: If you don't think that capability gains can happen quickly, you're not going to worry about the disaster scenario where you suddenly wake up and it's too late to switch the AI off and you didn't get a nice long chain of earlier developments to warn you that you were getting close to that and that you could now start doing AI alignment work for the first time . . .

(You know, science doesn't happen by press release. You have to start it earlier if you want it later.)

Leaving that aside, one thing I want to point out is that I expect that most of you are finding the rapid gain part to be the most controversial part of this, but it's not necessarily the part that most of the disasters rely upon.

Absolute capability? If brains aren't magic, we can get there. Capability advantage? *This* hardware is not optimal. It's sending signals at a millionth the speed of light, firing at 100 Hz, and even in heat dissipation (which is one of the places where biology excels), it's dissipating 500,000 times the thermodynamic minimum

energy expenditure per binary switching operation per synaptic operation. We can definitely get hardware one million times as good as the human brain, no question.

(And then there's the software. The software is terrible.)

The message is: AI alignment is difficult like rockets are difficult. When you put a ton of stress on an algorithm by trying to run it at a smarter-than-human level, things may start to break that don't break when you are just making your robot stagger across the room.

It's difficult the same way space probes are difficult. You may have only one shot. If something goes wrong, the system might be too "high" for you to reach up and suddenly fix it. You can build error recovery mechanisms into it; space probes are supposed to accept software updates. If something goes wrong in a way that precludes getting future updates, though, you're screwed. You have lost the space probe.

And it's difficult sort of like cryptography is difficult. Your code is not an intelligent adversary if everything goes *right*. If something goes wrong, it might try to defeat your safeguards—but normal and intended operations should not involve the AI running searches to find ways to defeat your safeguards even if you expect the search to turn up empty. I think it's actually perfectly valid to say that your AI should be designed to fail safe in the case that it suddenly becomes God—not because it's going to suddenly become God, but because if it's not safe even if it did become God, then it is in some sense running a search for policy options that would hurt you if those policy options are found, and this is dumb thing to do with your code.

More generally: We're putting heavy optimization pressures through the system. This is more-than-usually likely to put the system into the equivalent of a buffer overflow, some operation of the system that was not in our intended boundaries for the system.

Lessons from NASA and cryptography

AI alignment: treat it like a cryptographic rocket probe. This is about how difficult you would expect it to be to build something smarter than you that was nice, given that basic agent theory says they're not automatically nice, and not die. You would expect that intuitively to be hard.

Take it seriously. Don't expect it to be easy. Don't try to solve the whole problem at once. I cannot tell you how important this one is if you want to get involved in this field. You are not going to solve the entire problem. At best, you are going to come up with a new, improved way of switching between the suspend utility function and the normal utility function that takes longer to shoot down and seems like conceptual progress toward the goal—Not literally at best, but that's what you should be setting out to do.

(. . . And if you do try to solve the problem, don't try to solve it by having the one true utility function that is all we need to program into AIs.)

Don't defer thinking until later. It takes time to do this kind of work. When you see a page in a textbook that has an equation and then a slightly modified version of an equation, and the slightly modified version has a citation from ten years later, it means that the slight modification took ten years to do. I would be ecstatic if you told me that AI wasn't going to arrive for another eighty years. It would mean that we have a reasonable amount of time to get started on the basic theory.

Crystallize ideas and policies so others can critique them. This is the other point of asking, "How would I do this using unlimited computing power?" If you sort of wave your hands and say, "Well, maybe we can apply this machine learning algorithm and that machine learning algorithm, and the result will be blah-blah-blah," no one can convince you that you're wrong. When you work with unbounded computing power, you can make the ideas simple enough that people can put them on whiteboards and go, "Wrong," and you have no choice but to agree. It's unpleasant, but it's one of the ways that the field makes progress. Another way is if you can actually run the code; then the field can also make progress. But a lot of times, you

may not be able to run the code that is the intelligent, thinking self-modifying agent for a while in the future.

What are people working on now? I'm going to go through this quite quickly. Mostly, I'm just going to frantically wave my hands and try to convince you that there's an actual field here, even though there's maybe a dozen people in it full-time (and another dozen people not full-time).

4 Where we are now

Recent topics

Utility indifference: this is throwing the switch between the two utility functions. See Soares et al., "Corrigibility."[6]

Low-impact agents: this was, "What do you do instead of the Euclidean metric for impact?" See Armstrong and Levinstein, "Reduced Impact Artificial Intelligences."

Ambiguity identification: this is, "Have the AGI *ask* you whether it's OK to administer endogenous opiates to people, instead of going ahead and doing it." If your AI suddenly becomes God, one of the conceptual ways you could start to approach this problem is, "Don't take any of the new options you've opened up until you've gotten some kind of further assurance on them." See Soares, "The Value Learning Problem."[7]

Conservatism: this is part of the approach to the burrito problem, "Just make me a burrito, darn it!" If I present you with five examples of burritos, I don't want you to pursue the *simplest* way of classifying burritos versus non-burritos. I want you to come up with a way of classifying the five burritos and none of the non-burritos that covers as little area as possible in the positive examples, while still having enough space around the positive examples that the AI can make a new burrito that's not molecularly identical to the previous ones. This is conservatism. It could potentially be the core of a whitelisted approach to AGI, where instead of not doing things that are blacklisted, we expand the AI's capabilities by whitelisting new things in a way that it doesn't suddenly cover huge amounts of territory. See Taylor, Conservative Classifiers.[8]

Specifying environmental goals using sensory data: this is part of the project of "What if advanced AI algorithms look kind of like modern machine learning algorithms?" Which is something we started working on relatively recently, owing to other events (like modern machine learning algorithm suddenly seeming a bit more formidable). A lot of the modern algorithms sort of work off of sensory data, but if you imagine AGI, you don't want it to produce *pictures* of success. You want it to reason about the causes of its sensory data—"What is making me see these particular pixels?"—and you want its goals to be over the causes. How do you adapt modern algorithms and start to say, "We are reinforcing this system to pursue this environmental goal, rather than this goal that can be phrased in terms of its immediate sensory data"? See Soares, "Formalizing Two Problems of Realistic World-Models."[9]

Inverse reinforcement learning is: "Watch another agent; induce what it wants." See Evans et al., "Learning the Preferences of Bounded Agents."[10]

Act-based agents is Paul Christiano's completely different and exciting approach to building a nice AI. The way I would phrase what he's trying to do is that he's trying to decompose the entire "nice AGI" problem into supervised learning on imitating human actions and answers. Rather than saying, "How can I search this chess tree?" Paul Christiano would say, "How can I imitate humans looking at another imitated human to recursively search a chess tree, taking the best move

6. https://www.aaai.org/ocs/index.php/WS/AAAIW15/paper/download/10124/10136
7. https://intelligence.org/files/ValueLearningProblem.pdf
8. https://agentfoundations.org/item?id=467
9. https://intelligence.org/files/RealisticWorldModels.pdf
10. https://www.fhi.ox.ac.uk/wp-content/uploads/nips-workshop-2015-website.pdf

at each stage?" It's a very strange way of looking at the world, and therefore very exciting. I don't expect it to actually work, but on the other hand, he's only been working on it for a few years; my ideas were *way* worse when I'd worked on them for the same length of time. See Christiano, Act-Based Agents.[11]

Mild optimization is: is there some principled way of saying, "Don't optimize your utility function so hard. It's OK to just fill the cauldron."? See Taylor, "Quantilizers."[12]

Older work and basics

Some previous work that might be fun to be familiar with: *AIXI* is the perfect rolling sphere of our field. It is the answer to the question, "Given unlimited computing power, how do you make an artificial general intelligence?" If you don't know how you would make an artificial general intelligence given unlimited computing power, Hutter's "Universal Algorithmic Intelligence" is the paper.[13]

Tiling agents was already covered. See Fallenstein and Soares, "Vingean Reflection."[14]

Software agent cooperation: This is just some really neat stuff we did where the motivation is sort of hard to explain. There's an academically dominant version of decision theory, causal decision theory. Causal decision theorists do not build other causal decision theorists. We tried to figure out what would be a stable version of this and got all kinds of really exciting results, like: we can now have two agents and show that in a prisoner's-dilemma-like game, agent A is trying to prove things about agent B, which is simultaneously trying to prove things about agent A, and they end up cooperating in the prisoner's dilemma.

This thing now has running code, so we can actually formulate new agents. There's the agent that cooperates with you in the prisoner's dilemma if it proves that you cooperate with it, which is FairBot, but FairBot has the flaw that it cooperates with CooperateBot, which just always cooperates with anything. So we have PrudentBot, which defects against DefectBot, defects against CooperateBot, cooperates with FairBot, and cooperates with itself. And again, this is running code.

If I had to pick one paper and say, "Look at this paper and be impressed," it would probably be LaVictoire, et al., "Program Equilibrium in the Prisoner's Dilemma via Löb's Theorem."[15] Also, Andrew Critch worked out the bounded form of Löb's Theorem so that we could say that there would be similar behavior in bounded agents.[16] It's actually a slightly amusing story. We were all sure that someone must have proved this result previously. Andrew Critch spent a bunch of time looking for the previous proof that we were all sure existed, and he said, "Fine. I'm going to prove it myself. I'm going to write the paper. I'm going to submit the paper. And then the *reviewers* will tell me what the previous citation was!"

(It is currently going through the review mechanism and will be published in good time. It turned out no one had proved it. Go figure.)

Reflective oracles are the randomized version of the halting problem prover, which can therefore make statements about itself, which we use to make principled statements about AIs simulating other AIs as far as they are, and also throw some interesting new foundations under classical game theory. See Fallenstein et al., "Reflective Oracles."[17]

Where to start

Where can you work on this?

11. https://arbital.com/p/act_based_agents/
12. https://www.aaai.org/ocs/index.php/WS/AAAIW16/paper/download/12613/12354
13. http://www.hutter1.net/ai/aixigentle.htm
14. https://intelligence.org/files/VingeanReflection.pdf
15. http://www.aaai.org/ocs/index.php/WS/AAAIW14/paper/viewFile/8833/8294
16. https://arxiv.org/abs/1602.04184
17. https://arxiv.org/abs/1508.04145

The Machine Intelligence Research Institute in Berkeley: We are independent. We are supported by individual donors. This means that we have no weird, exotic requirements, paperwork requirements and so on. If you can demonstrate the ability to make progress on these problems, we will hire you. We will get you a visa.

The Future of Humanity Institute is part of Oxford University. They have slightly more requirements, but if you have traditional academic credentials (and you want to live in Oxford), then you can go to the Future of Humanity Institute at Oxford University.

Stuart Russell is starting up a program and looking for three post-docs (at least) at UC Berkeley in this field. Again, some traditional academic requirements, but I'm giving this talk at Stanford, so I expect a number of you probably have those.

Leverhulme CFI (the Centre for the Future of Intelligence) is starting up in Cambridge, UK. It's a joint venture between the Centre for the Study of Existential Risk and Leverhulme. It's also starting up and in the process of hiring.

If you want to work on low-impact in particular, you might want to talk to Dario Amodei and Chris Olah. If you want to work on act-based agents, you can talk to Paul Christiano, who is currently working on it alone, but has three different organizations offering to throw money at him if he ever wants someone else to work on it with him.

In general, email **contact@intelligence.org** if you want to work in this field and want to know, "Which workshop do I go to to get introduced? Who do I actually want to work with?"

5 Questions and Answers

Question 1

Speaker: Thank you for this very stimulating talk. For the first two-thirds of it, I was thinking that where you were going, or maybe the conclusion that I would reach, is that the pure problem-solving approach to AI is not going to be able to solve this problem and that maybe instead we should look at things like, if we're interested in superintelligence, whole brain emulation or something which by the nature of the way it's built reflects our nature. But then you never got there. I thought it sounded like in the end you think that the problem is very hard, but that it's solvable, and that's the direction you want to go. So why should we believe that it's solvable, if that is in fact your conclusion?

Yudkowsky: I would say that it's solvable in the sense that all the problems that we've looked at so far seem like they're of limited complexity and non-magical. If we had 200 years to work on this problem and there was no penalty for failing at it, I would feel very relaxed about humanity's probability of solving this eventually.

The fact that if we failed, nonetheless, it would create an expanding sphere of von Neumann probes, self-replicating and moving at as near to the speed of light as they can manage, turning all accessible galaxies into paperclips or something of equal unimportance, would still caused me to make sure that this field was not underfunded; but if we had 200 years and unlimited tries, it would not have the same "Aaaaaaahh!!" quality to it.

OK, so it does have an "Aaaaaaahh!!' quality to it. Why not work on uploads instead—human brain emulations? There was a previous workshop where all of the participants agreed that we wanted to see uploads come first. Most of us did not see how we could do that, and the reason is if you study neuroscience and reverse-engineer the brain, then before you get full-scale, high-fidelity, personality-preserving, nice human emulations, what you get is the AI people taking your algorithms and using them to make neuromorphic AI.

We just did not see how we can arrange the technology tree such that you would actually get whole-brain emulations before you got AIs based on much cruder levels of understanding of neuroscience. Maybe you could do it with a Manhattan project where the results of the project are just not being fed to the rest of the planet and AI researchers. I think I would support that, if Bill Gates or a major national

government said that this was what they wanted to do and how they wanted to approach the problem.

Question 2

SPEAKER: We've been lucky as people (not as individuals), in that death teaches us a certain amount—the rest of us, anyways—and that pain teaches us, as children, a certain amount of optimization errors. Do we want to go to AIs and basically make them all believe in Murphy? Be careful in your optimization?

YUDKOWSKY: I'm not quite sure I . . . There was a statement that humans are taught by pain as children. Why do we want to make AIs believe in Murphy? I don't quite understand what of the proposals so far corresponds to AIs believing in Murphy. *Programmers* should believe in Murphy . . .

SPEAKER: Why shouldn't the AI? If the programmer believes it and says that's a limit, why shouldn't he teach that to the AI?

YUDKOWSKY: Because if the AI . . . Because it's quite complicated to get right and you want to keep it as simple as possible, and not turn all accessible galaxies into paperclips. If you are more careful about doing that, you are less likely to turn all accessible galaxies into paperclips. Why wouldn't we want to . . . ? Or is that a sufficient answer?

SPEAKER: That will do for now.

YUDKOWSKY: OK.

Question 3

SPEAKER: I read somewhere that the success of AlphaGo seemed to make you nervous. I wanted to ask a converse question. If there was solid empirical evidence, let's say a couple of decades from now, that human consciousness and intelligence uses quantum-mechanical effects, would that make you less nervous?

YUDKOWSKY: The question is: AlphaGo made me nervous; would I then become less nervous if there was solid evidence that human intelligence operated through quantum-mechanical effects?

I'm not sure it would make me very much less nervous. Before I start: the premise is moderately implausible. The question has been raised before. There seemed to be reasonably strong reasons to believe that there is no macroscopic decoherence in the brain.

Leaving that aside, lots of quantum algorithms are not "magical." They're good for some amount of speed-up, but not infinite speed-up. Some of them do pretty impressive speed-ups. I would have to ask: Whatever the brain is doing, how irreplaceable of a quantum algorithm did nature actually ran across? Am I to believe that there is no analogous, non-quantum algorithm that can do similar things to a sufficiently good level? Am I to believe that hardware is not going to duplicate this? Can people just build a giant vat of neurons and get way better results out of an analogous quantum algorithm?

When obstacles are discovered, people, like AIs, are clever and look for ways around the obstacles. It would extend the timeline, but it wouldn't extend it for 50 years.

Question 4

SPEAKER: As a neuroscientist, I have a converse question for you, which is: If I'm trying to study the brain, what sort of things should I look for that indicate the implementation of a value alignment problem in a human, or in an animal like a mouse? How would I look for that, or how would I study that?

YUDKOWSKY: The question was: As a neuroscientist, can I look for analogues of value alignment problems in my own work, and if so, how?

That's a new question. If I'm not allowed to take five minutes to go quiet and think about it, then my immediate answer is, "It's not obvious where the analogues

would be, unless there was something equivalent to maybe conservatism or ambiguity identification."

It's not like mammals are aligned to these outside systems using a simple alignment algorithm that is loading the values from the outside system. We come with it wired into our brain already. The part where natural selection caused the particular goals we pursue to align in the ancestral environment with inclusive genetic fitness has already been done. Plus natural selection completely botched it. Humans do not pursue inclusive genetic fitness under reflection. We were just a particular kind of thing that operated to "coincidentally" produce inclusive genetic fitness in the ancestral environment. Once we got access to contraceptives, we started using them.

If there is a lesson to derive from natural selection, it would be something along the lines of, "If you have a Turing-complete thing you are optimizing, such as DNA,"—not literally Turing-complete, because it can't get arbitrarily big; but you know what I mean—"and I apply enough optimization pressure to this Turing-complete program to make it pursue a goal like inclusive genetic fitness, I will get a thing that is actually a sapient consequentialist deliberately planning how to get . . . a bunch of stuff that isn't actually that thing."

We are the daemons of natural selection. We are the optimization systems that popped up inside the optimization system in a way that was "unanticipated," if natural selection could anticipate anything. The main lesson we have to draw from natural selection is "Don't do it that way."

There might be lessons that we can draw from looking at the brain that are going to play a role in value alignment theory, but aside from looking at particular problems and asking, "Is there a thing in the brain that does conservatism? Is there a thing in the brain that does ambiguity identification?" it's not clear to me that there's any principled answer for how you could take stuff from neuroscience and import it into value alignment.

Question 5

Speaker: Say that you had the solution to everything tomorrow. How do you get AI researchers, for example, to take this problem seriously? Because [inaudible] for a problem in computer science for forever, but we still have that problem now. How do you make sure that everyone is actually on board with this?

Yudkowsky: The question is: If you have technical solutions, how do you get AI people to implement them?

Stuart Russell is I think the main person who as an insider is making the principled appeal. You do not have bridge engineering, and then a bunch of people who aren't engineers thinking about how to have bridges not fall down. The problem of bridge engineering just is "make a bridge that doesn't fall down." The problem of AGI we should see as just: "How do you run computer programs that produce beneficial effects in the environment?" Where the fact that you're trying to direct toward a particular goal is assumed, in the way that when you're trying to build a chess device, the fact that you're trying to direct toward a particular goal is assumed—not, "How do we rush frantically to get something, anything with intelligence?"

There's that line of pursuit. The Future of Humanity Institute at Oxford does a lot of public-facing work. The Machine Intelligence Research Institute, where I work, sees its own role as being more, "Make sure that the technical stuff is there to back up the people saying to do this right on a technical level." I don't actually have the expertise to answer your question as well as I might like, because we're the ones who specialize in going off and trying to solve the technical problems, while FHI, in addition to doing some technical problems, also does public-facing stuff.

That said, there certainly have been disturbing trends in this area, and I think we're starting from a rather low baseline of concern, where startups have been telling venture capitalists that they will have AGI for a long time before the first time any of them ever said, "We will have AGI, and it will not destroy the world." The very thought that you need to point these things in a direction, and that that is actually

an interesting technical part of the problem that you actually need to solve and be careful about, is new and does need to be propagated.

Question 6

SPEAKER: Thank you, Eliezer. I thought it was a very informative and interesting talk, and everybody should reflect carefully on the future of AGI. Could you go into a little bit more in depth on conservatism and what you're hoping research issues are on conservatism?

YUDKOWSKY: Well, first, conservatism has nothing to do with the political movement, one way or another.

It's something that recently opened up, where we just started to issue calls for proposals and put up various things on whiteboards and stare at them. An example of a thing that we recently stared at on the whiteboard was somebody said, "Well, suppose that you do have multiple hypotheses for what the classification rule could be. Is there any difference between the form of conservatism we want and maximizing the probability that something is inside the class, given that you have multiple hypotheses and so the point of maximum probability will be at the point of maximum overlap?" And I waved my hands a bit and said, "It seems to me that these two could come apart because you could have exceptionally simple classifiers that imply increased probabilities to get into a particular portion of the space, and so you might just end up over there in this weird corner of the space that does maximum probability. Whereas the things that humans actually want are going to be classified according to a more complicated rule that's not going to be very close to the start of the list of potential classification rules."

And it does seem to me that on a conceptual level, maximizing probabilities seems like we might very well be asking for a different thing than, "Classify this well while covering as little territory as possible." But basically it's a very new question and we haven't done that much real work on it yet. More phrasing questions than answering questions at this point, I think.

Question 7

SPEAKER: There was, in the thought experiment about generating smiles, a step where the AGI got smart enough to simulate what the programmers would disagree with and kind of work around that. So, given that we solve the utility indifference problem, would that be a good path to go and try to figure out how to switch it off whenever it simulates—kind of like a mild version of CEV, or something like that?

YUDKOWSKY: The question is: There was a step in the story told before where the AGI started working out what behaviors its programmers would not want to see and avoiding those behaviors so as to appear from our perspective deceptively nice, and from its perspective continuing to get maximum expected value for its utility function. Could the switch-between utility algorithm from before be a way to work around or avoid that scenario?

Yes, it is! The switching-between-two-utilities-on-or-off is indeed the basic case of learning a more complicated utility by watching the environment without trying to tamper with the data that the environment is giving you.

Great question. The answer is yes.

Question 8

SPEAKER: It seems like when you're considering these things you use a lot of human-centric assumptions about how thought happens and what a general artificial intelligence might do. How do you check your own assumptions about what that may look like, so that you're not just looking at a subset of the problem space?

YUDKOWSKY: The question is: He seemed to detect humanoid or anthropomorphic assumptions. How do you check those? How do you make sure you're not restricting yourself to a tiny section of the space?

It's very hard to know that you're not thinking like a human, from the perspective of an AI. I did start to give an example of a case where it seems like we might be able to think that utility functions (and, by very similar arguments, coherent probability distributions) are things that start to come up in sufficiently advanced agents, because we have multiple coherence theorems all pointing in the same direction, at the same class of behaviors.

You can't actually do perfect expected utility maximization, because you can't evaluate every outcome. What you can say is something like: to the extent that you as a human can predict behavior that is incompatible with any utility function, you are predicting a stupidity of the system. A system that has stopped being stupid from your perspective will look to you as if it is compatible with having a utility function, as far as you can tell in advance.

That was an instance of trying to give an argument that goes past the human. In a lot of cases where I talk about an AI potentially modeling its programmers and avoiding behavior that it expects to lead to its utility function being edited, this is just me putting myself in the AI's shoes. But for a sufficiently advanced agent we can make something like an efficiency assumption.

An efficient market price is not an accurate market price. It's a market price such that you can't predict a net change in that price. Suppose we imagine a superintelligence trying to estimate the number of hydrogen atoms in the sun. We don't expect it to get the number of hydrogen atoms exactly right. But if you think that you can say in advance, "Oh, it's going to forget that hydrogen atoms are very light and underestimate the number by 10%," you're proposing something that is akin to predicting that Microsoft stock price will rise by 10% over the next week without using insider information. You are proposing that you know a directional error in the estimates of the other agent.

Similarly, we can look at a modern chess program—which is now way above the human level—and say, "I think the chess program will move over here in order to pursue a checkmate." You could be right. Suppose that the program is somewhere else. Do we say, "Haha! It didn't take the best move." No. We say, "Whoops! I guess I was wrong about what the best move was." We suppose that either we overestimated how much expected utility was available from the move we thought it would take, or we underestimated the expected utility available from a different move. And the more surprising the other move is, the more we think we underestimated that move.

So if you ask me, "Will the AI actually be modeling the programmers? Will it actually go for protein folding to get its own nanotechnology?" first of all, it might not apply to an AI that is not strictly superhuman. But second, if it is sufficiently superhuman, then I don't expect it to do that exact thing. I'm in a state of Vingean uncertainty. It's smarter than me. I can't predict its exact policy. But I expect it to get at least as much expected utilities as I could get in its shoes, if it's not pursuing molecular nanotechnology, given that Eric Drexler in the book *Nanosystems* ran numerous basic calculations strongly indicating feasibility.

Nanotechnology looks like it should be possible, and in a certain sense, it is possible. It's in all of us, and it's held together by weak little van der Waals forces instead of covalent bonds. We can have things that are to ribosomes as steel is to flesh. Maybe an AI doesn't get that, but if so it's because it's found something better, not because it's just leaving the value on the table from its perspective.