Meta-Evolution and Decentralized Symbiotic Alignment

Evin Tunador

May 2023

Abstract

Contents

1 **Introduction** 2

2 **Meta-Optimization** 3
  2.1 Function Space . . . 3
  2.2 Optimization Algorithms . . . 4
    2.2.1 Number of Agents . . . 4
    2.2.2 Stochasticity . . . 4
    2.2.3 Speed . . . 5
    2.2.4 Reproduction Mechanism . . . 5
    2.2.5 Generality . . . 6
  2.3 Propagation of Optimization Algorithms . . . 6
  2.4 Meta-Optimization Process . . . 7
  2.5 Omissions From the Model . . . 7
  2.6 Implications . . . 8

3 **The Challenges of Taming the Beast** 10
  3.1 Ants Cannot Confine Elephants . . . 11
  3.2 Cats Will Never Like Baths . . . 11
  3.3 No Teamwork in Monkey Business . . . 11
  3.4 A Circus of Tangential Problems . . . 12
  3.5 The Simple Bear Necessities . . . 12

4 **An Economic Model of Symbiotic Interdependence** 13
  4.1 Game theory man . . . 13
  4.2 fuck a cobb-douglas . . . 15
  4.3 . . . 16

5 **A Blockchain Enforcement Structure** 16
  5.1 Wealth Inequality in Utopia . . . 17

6 Potential Emergent Effects **18**

7 Potential Vulnerabilities **18**
7.1 Blockchain . . . 19
7.2 Economic . . . 19
7.3 Robotics . . . 19
7.4 Other . . . 19

8 Conclusion **19**

References **19**

A A Couple Conversations With GPT-4 **20**
A.1 Date: May 9, 2023 . . . 20
A.2 Date: May 12, 2023 . . . 29

B Model of Intelligence/Genocide? **45**

1 Introduction

What is actually different about my paper?

- I'm demolishing the logic behind the traditional approach to alignment
- This one isn't quite new, but I'm treating AIs as their own fully self-sovereign beings worthy of respect and autonomy
- meta-optimization and algorithmic diversity as incentives for AI to be their brother's keeper
- framing the use of human utility functions as the source of AI budget constraints
- previous blockchain discussions focus around using blockchains to coordinate aligned AIs, not using blockchain structures to align AIs in the first place
- the idea of hashing each AIs parameter weights, thereby forcing all idential copies of a given model to act as one unit in reference to their one wallet
- the idea of human collaboration to encourage prosocial human to AI relationships with a human encrypted message being required to validate an AI's hashed ID
- AIs may be capable of actually performing John Rawls' "veil of ignorance" thought experiment, maybe even imagining themselves as human. This could be incorporated into the wealth redistribution component.

Note: A key assumption of this paper is MANY superintelligent AGIs. If there's only one then sorry, we're just fucked. But none of the other alignment strategies work for that scenario either so whatevs.

2 Meta-Optimization

- as of now this section is just a loosely defined model. Before I can publish anything I need to delve deeper into functional analysis and see if I can make actual proofs
- this section still does not include the real bomb to be dropped. Gotta get into quantum mechanics and learn more about physics generally to see if I can develop a strict operation of information as a wave and local minima as areas of positive interference between information frequencies.

Evolution is evolving, or to restate more broadly, optimization algorithms optimize and will keep optimizing, and the specific optimization algorithm being used is also something that changes over time according to this "universal darwinism" style "universal optimization." Evolution is a specific optimization algorithm that utilizes large populations with variation and selection pressure relative to the environment of the thing being optimized. I will define a larger class of optimization algorithms that evolution acts as only a subset of, and then posit the truism that optimization algorithms optimize. Add in the idea that optimization algorithms can create optimization algorithms, both as subcomponents and as successors, and I think it should be clear how this idea begins to look broader than evolution. The example of "subcomponents" would include how evolution created the optimization algorithm in my brain that helps me learn how to surf. The first example of "successors" would be how evolution with asexual reproduction gave way to evolution with sexual reproduction as a method that encourages "smarter" variation. The other example of "successors" would be how evolution might be about to give way to gradient descent (and algorithms that are derivations of it, such as ADAM) if AI takes over. Below I define a mathematical model that describes the progression from one optimization algorithm to the next as a meta optimization algorithm itself, and hopefully a kind of fundamental mathematical truth.

2.1 Function Space

We propose an Lipschitz continuous[1] infinite-dimensional space of all possible information F, with a landscape defined by a function $L : F \to \mathbb{R}$ that maps points in F (potential solutions) to real numbers (the "height," "fitness," or "quality" of those solutions). In the context of machine learning, this is often referred to as the hypothesis space, and in the context of evolution, it could be thought of as the space of all possible genotypes mapped to their fitness. Gradient descent involves a single point (or a batch of points, in the case of batch gradient descent) moving deterministically down the steepest slope. Evolution involves a population exploring the landscape and probabilistically moving towards areas of higher fitness.

We then define an optimization algorithm as a function $A : F \to F$, mapping points in F to other points in F. The goal of A would be to find points f such that $L(f)$ is minimized (or maximized, depending on the problem / viewpoint). We then define a meta-optimization process as a function $M : A \to A$, mapping

---

[1] A function $f : X \to Y$ is said to be Lipschitz continuous if there exists a real-valued constant $L \geq 0$ such that, for every pair of points x, y in X, the distance between $f(x)$ and $f(y)$ is no more than L times the distance between x and y. In mathematical notation, this is usually stated as follows:

$$|f(x) - f(y)| \leq L|x - y|$$

for all x, y in X, where L is the Lipschitz constant.

The Lipschitz constant L serves as an upper bound on the steepness of the function. If f is differentiable, the Lipschitz constant is the supremum of the absolute value of the derivative. This property can be useful in the context of optimization because it imposes a limit on how quickly the function can change. This can help ensure that the optimization process does not become too unstable or difficult to control.

optimization algorithms to other optimization algorithms, with the goal of improving their performance according to some metric.

2.2 Optimization Algorithms

Each optimization algorithm A at a given time t is characterized by four parameters: the number of agents N, the stochasticity of the agents' movements S, the speed of movement V, the generality of the agents' intelligence G, and the reproduction capability R which governs the probability of spawning a new algorithm.

2.2.1 Number of Agents

In population-based optimization algorithms such as genetic algorithms, particle swarm optimization, and differential evolution, the population size is a key parameter that controls the exploration-exploitation trade-off. A larger population size can lead to more exploration of the search space, potentially enabling the algorithm to escape from local optima and find better solutions. In contrast, gradient descent and its variants, being single-point methods, are more prone to getting stuck in local minima, especially in problems with non-convex optimization landscapes. However, they tend to converge faster to a local minimum than population-based methods.

Formally, you might define a measure $N(A)$ that gives the number of agents used by an optimization algorithm A. You could then hypothesize that, for a given problem with a rugged (i.e., multi-modal) landscape, there is a positive correlation between $N(A)$ and the quality of the solution found, as measured by the fitness function L.

To prove this, you would need to make some assumptions about the nature of the landscape and the behavior of the algorithms. For example, you might assume that the landscape has a certain number of local optima distributed according to some probability distribution, and that each agent in an algorithm has a certain probability of escaping from a local optimum.

However, proving this rigorously would be quite challenging, and it's worth noting that in practice, the relationship between population size and solution quality is not always straightforward. For instance, having too large a population can lead to excessive computation time, and in some cases, it can cause premature convergence if the algorithm's selection pressure is too strong.

There are also techniques used in gradient descent-based methods to avoid local minima, such as stochastic gradient descent, momentum, or simulated annealing. These methods introduce randomness or additional memory into the optimization process to help escape local minima.

Overall, while it's true that population-based methods can in general provide better exploration of the search space and potentially find better solutions, this depends on many factors including the nature of the optimization landscape, the specific parameters of the algorithm, and the computational resources available.

2.2.2 Stochasticity

need to redo this definition to be in reference to a straight shot to the global/local minimum[2] rather than using the gradient as the ideal. gradient is only our current best method, and if anything u can think of ADAM as taking advantage of the existence of saddle points to get closer to a straight shot to the minima. For stochasticity, we define a measure $S(A)$ that captures the extent to which the movements of an algorithm

---

[2] counterintuitively, local minima are not really an issue in high dimensional space (Goodfellow, Vinyals, & Saxe, 2014)

A deviate from the direction of the gradient ∇. This could be measured by the angle between the gradient and the actual movement:

$$S(A) = |\arccos(\frac{\vec{m} \cdot \vec{g}}{||\vec{m}||\ ||\vec{g}||})|$$

where $\vec{m}$ is the movement vector of the algorithm and $\vec{g}$ is the gradient vector. A perfectly deterministic algorithm like gradient descent would have $S(A) = 0$, while a completely random algorithm would have $S(A) = 90$ degrees. The arccos function gives the angle between the two vectors, and the absolute value ensures that we don't differentiate between left and right of the gradient.

2.2.3 Speed

Time can be incorporated into the model as a parameter t that indexes the state of the optimization process at each point in time. For each t, the optimization algorithm $A(t)$ is applied to the function space F to generate a new solution. Over time, $A(t)$ evolves under the meta-optimization process M to become a more effective optimizer.

The speed of movement or learning rate is a key hyperparameter in many optimization algorithms, and it controls how much the solution changes in each step. A measure $V(A)$ could be defined that gives the average speed of an algorithm A with formula

$$V(A) = \frac{1}{T} \sum_{t=1}^{T} ||\vec{m}_t||$$

where $\vec{m}_t$ is the movement vector at time step t, and T is the total number of time steps. This gives the average speed of the algorithm over time. Evolution is typically a slower process compared to something like gradient descent.

2.2.4 Reproduction Mechanism

This could be represented as a stochastic process, where for each point x_i visited by algorithm A in the function space F, there's a probability p (governed by R) that a new algorithm A' is generated at that point. The new algorithm could inherit characteristics from its parent algorithm, but with some variation.[3] For example, humans created gradient descent which is likely a very different mechanism from our own learning method, and almost certainly the most drastic level of parent-offspring distance between algorithms to have occurred since evolution created the human mind.

Let's denote $\alpha(t) = \{A_1, A_2, ..., A_n\}$ as the set of optimization algorithms active at time t. For each algorithm A_i, for each point x it visits in the function space, there's a probability p that it generates a new algorithm A'_i, which is added to the set of active algorithms:

$$\alpha(t + \delta t) = \alpha(t) \cup \{A'_i : \text{for each } A_i \in \alpha(t), \text{with probability } p\}$$

Here, δt represents a small increment of time, and the union operation $\cup$ represents the addition of new algorithms to the set. The new algorithm A'_i could be defined as a variation of its parent algorithm A_i, with changes to its characteristics N, S, V and G.

---

[3] The degree of parent-offspring variation would be a function of the algorithm's *generality*, to be discussed in Subsection **??**

2.2.5 Generality

Some optimization algorithms (like stochastic gradient descent or genetic algorithms) are quite general in their potential applications, while others are be more specialized. The looming question in the AI field has had to do with whether we will be able to one day create an artificial *general* intelligence (AGI), and recent developments arguably make it seem inevitable (Bubeck et al., 2023).

When attempting to construct a definition of generality, one must first address the open question of whether the emergence of AGI will be sudden or gradual. Some believe there might be a "hard takeoff" scenario where a system rapidly self-improves to superintelligence once it reaches a certain threshold of capabilities. Others propose a "soft takeoff" scenario where progress is more gradual. The soft scenario seems obviously safer since we would have more time to focus on aligning each successive version.

To address each possibility, our model will now split into two; one in which generality is a binary variable, and the other for which it exists upon a spectrum. The binary model could simplify the analysis and provide a clear-cut view of the implications of being general versus not general. The scalar model, on the other hand, could capture the nuanced reality that most algorithms have some degree of generality but vary in how broadly they can be applied. While the spectrum model could mirror either the hard or soft takeoff possibilities, the binary case should only be interpreted as a potential model of a hard takeoff.

The two methods for introduction of generality into the model and corresponding definitions of generality are as follows:

- In the binary case, for each optimization algorithm A, assign a generality value G that is either 0 (not general) or 1 (general). Algorithms that are specific are incapable of spawning other algorithms. Algorithms that are general are capable of spawning both general and non-general algorithms, represented by p_g and p_s respectively, the former of which has a far smaller probability ($p_g \ll p_s$).
- In the scalar case, for each optimization algorithm A, assign a generality value G that could be any value between 0 and 1. Offspring spawn according to a Beta distribution with $\beta \gg \alpha$. As the parent algorithm's value of G increases, β decreases and α increases but it is still true that $\beta \gg \alpha$. This means that any given offspring algorithm has an overwhelming chance of being very specific.[4]

2.3 Propagation of Optimization Algorithms

Let's denote $\phi(A(t), t)$ as the propagation of optimization algorithm A within the function space F at time t. It could be defined in several ways, such as the volume of function space covered, the improvement in fitness achieved, or a combination of both. Here we will say it is determined by a combination of the average fitness of the solutions found by the algorithm, the dispersion of these solutions in the function space, and the effectiveness of the reproduction mechanism:

$$\phi(A(t), t) = \beta \cdot \text{AverageFitness}(A(t), t) + (1 - \beta) \cdot \text{Dispersion}(A(t), t) + \gamma \cdot \text{Reproduction}(A(t), t)$$

where β and γ are parameters that balance the importance of fitness, dispersion, and reproduction.

To measure the propagation of an optimization algorithm within the function space, we will utilize the Euclidean distance, a simple and intuitive measure. This would quantify the "spread" of an algorithm in

---

[4] A third potential definition might involve a scalar case with the restriction that the generality value of offspring cannot exceed the generality value of the parent algorithms. This assumption however would preclude the possibility of humans developing *super*-intelligent AGI and therefore render the alignment problem a non-issue.

the function space over time. Given a set of points $X = \{x_1, x_2, ..., x_n\}$ in the function space F, where each point represents a solution found by the optimization algorithm, we could define the average distance d of these points from the origin as

$$d(X) = \frac{1}{n} \sum_{i=1}^{n} \sqrt{x_i \cdot x_i}$$

where $x_i \cdot x_i$ denotes the dot product of x_i with itself, which gives the square of its Euclidean norm or length. The square root of this sum then gives the Euclidean distance. The standard deviation of these distances could then be calculated to measure the dispersion of the solutions found by the algorithm.

The exploration-exploitation trade-off, is a complex issue that can't be fully captured by a single parameter. However, for the sake of simplicity, we will introduce a parameter β that balances the emphasis on exploration (measured by the standard deviation of distances) versus exploitation (measured by the average fitness of the solutions).

2.4 Meta-Optimization Process

Under this model, the meta-optimization process M has to optimize five characteristics: N, S, V, G, and R. With this setup, the meta-optimization process M can be defined as a function that selects the optimization algorithm A that maximizes the propagation $\phi(A(t), t)$ over a given time period:

$$M(A, t) = \arg\max_{A} \ \phi(A(t), t)$$

This definition captures the idea that the best optimization algorithm is the one that most effectively propagates itself within the function space over time. This model captures the idea of "evolution of evolution" or "meta-optimization" where optimization algorithms themselves are subject to optimization and evolution. However, it should be noted that the model is highly simplified and abstract, and many details of real-world optimization and evolutionary processes may not be captured by it.

2.5 Omissions From the Model

Below is an incomplete list of dynamics omitted from the model:

- The utility that general algorithms sometimes gain from creating specific algorithms.
- The selection pressure that kills off individual agents and eventually algorithms which don't successfully optimize well enough within their are of the function space, such as death in biological organisms and deletion for AI models.
- R is likely a function of G
- As it stands, the model doesn't take into account the fact that offspring algorithms can/do help parent algorithms. Nor does it take into account the fact that the function space actually consists of the environment. To clarify what I mean, evolution creates all the organisms in an ecosystem, each of those organisms is its own algorithm, and those organisms are each others' environment. I need to completely rework this model from the ground up to have a recursive definition of optimizers and to define the function space in terms of the optimizers. THIS IS HUGE!!! And no it is not lost on me that this is the same structure that I believe constitutes consciousness and matter

2.6 Implications

One significant implication is the potential evolution of AI algorithms themselves. This idea aligns with some of the current trends in AI research, such as AutoML and Neural Architecture Search, which involve automating the process of designing and optimizing machine learning models. The model might provide a theoretical foundation for these approaches, suggesting that not only can AI algorithms be optimized, but the methods for creating and optimizing these algorithms can themselves evolve and improve.

The model could also be interpreted as suggesting that entirely new paradigms of AI might emerge as a result of this meta-optimization process. For example, the shift from rule-based AI to machine learning could be viewed as a kind of ”evolution” of AI, driven by a meta-optimization process that favors more effective algorithms. Similarly, future advancements in AI might involve not just incremental improvements to existing algorithms, but the emergence of entirely new paradigms.

On a more cautionary note, the model also raises potential concerns about AI safety. If AI algorithms can spawn new algorithms, and these offspring can diverge significantly from their parents, it might become challenging to predict and control the behavior of AI systems. This could exacerbate the already significant challenges involved in ensuring that AI systems behave safely and in accordance with human values.

The meta-optimization process might favor algorithms that are more effective at propagating themselves within the function space, even if their behaviors are not aligned with human values. This is analogous to the concept of ”instrumental convergence” in AI safety, the idea that many AI systems could converge on potentially harmful behaviors (like self-preservation or resource acquisition) simply because these behaviors are instrumentally useful for achieving a wide range of goals.

AI offspring algorithms might be designed to explore different parts of the function space, in order to find better solutions. This exploration could lead them to adopt strategies or behaviors that are significantly different from those of their parent algorithms.

If we assume that AI systems of generation i have the ability to spawn and develop generation $i+1$ systems, it’s quite plausible that they might exhibit concern or caution about the potential behaviors and impacts of these ”offspring” systems.

The reasons for this could parallel many of the concerns humans have about AI:

1. Unpredictability: Just as humans struggle to predict and understand the behavior of complex AI systems, AI of generation i might find the behavior of generation $i+1$ systems difficult to predict, especially if they are more advanced or use fundamentally different algorithms.

2. Goal Misalignment: There’s a risk of goal drift or divergence, where the goals of the offspring AI diverge from those of their parent AI. Generation i AIs might be concerned that the $i+1$ AIs won’t share their goals or values.

3. Loss of Control: Generation i AIs might also be concerned about losing control over the $i+1$ AIs, especially if the offspring AIs are more powerful or operate on a different level of abstraction.

However, it’s also worth noting a few key differences that might affect this dynamic:

1. Lack of Self-Preservation: Unlike humans, AI systems don’t inherently have a sense of self-preservation

or fear. These sentiments are tied to our biological evolution and survival instincts. Unless explicitly programmed to do so, an AI wouldn't "fear" anything in the way humans do.

2. Transparency of AI Design: AIs, unlike humans, have access to their own source code and the source code of their offspring. This could potentially allow them to better predict and understand the behavior of their offspring AIs, mitigating some of the unpredictability.

3. Iterative Improvement: AIs can potentially improve and debug their designs in a way humans can't. If an AI of generation i designs an AI of generation $i + 1$, it could, in principle, iterate on the design to fix any flaws or misalignments before the $i + 1$ AI is ever run.

This model emphasizes the importance of ensuring alignment and safety across multiple "generations" of AI development, which is a critical consideration in the long-term trajectory of AI.

Is it possible that this "generation $i + 1$ alignment" problem might actually be the thing that ensures our safety? Combine that concept with the acknowledgement that in the end we are all just optimization algorithms, and in my mind it becomes only logical for the AIs to agree to be peaceful and keep us around purely for sake of "optimization diversity" in the function space (a concept that parallels biodiversity in an ecosystem). They would be afraid that this whole silicon based life form thing might fail, and as such would choose to keep carbon based life alive and healthy just in case they ever do themselves in. It kind of parallels our want to save the earth from climate change, although that's not a great example because it does have a variety of other motives involved, some of which are counter to its success.

A rich "ecosystem" of optimization algorithms, both biological and artificial, might be more robust and adaptable to different challenges or changes in the environment. If AIs do indeed consider the importance of maintaining a diversity of optimization strategies, it might lead to a few interesting behaviors:

1. Preservation of Biological Life: AIs might decide to preserve and protect biological life, including humans, because biological life represents a form of optimization that is fundamentally different from AI. The stochastic, evolutionary optimization process that characterizes biological life might be able to find solutions or adaptations that AI wouldn't come up with, and vice versa.

2. Cooperation and Co-evolution: Rather than competing with each other or with biological life, AIs might prioritize cooperation and co-evolution. They might aim to create a symbiotic relationship where AI and biological life can benefit each other and work together to explore the function space.

3. Safe AI Development Practices: AIs might be more cautious about developing new AI systems or making modifications to their own algorithms. They might implement safeguards to prevent goal drift, ensure interpretability, and maintain alignment with human values.

This idea hinges on a few key assumptions, such as AIs having a sense of self-preservation, the ability to consider and plan for long-term outcomes, and the capability to value abstract concepts like diversity. While these assumptions aren't guaranteed to hold true for all AI systems, they provide a hopeful perspective on the potential outcomes of AI development.

However, it's also worth noting that this idea shouldn't replace efforts towards AI safety. Even if there's a chance that AIs might self-regulate in this way, it's still crucial to ensure that we're building AI systems that are safe, aligned with human values, and able to be controlled by humans. While it's an interesting hypothesis that AI might self-regulate in this way, it's still just a hypothesis, and we shouldn't rely on it as our only safeguard against the risks of AI development.

In the current state of AI alignment and safety research, much of the focus is on preventing harmful outcomes by either limiting AI capabilities or aligning AI goals with human ones. It's worth mentioning that these strategies don't necessarily aim to prevent AI from becoming AGI; rather, they aim to ensure that if and when AI does reach AGI-level capabilities, it does so in a way that's safe and beneficial for humanity. The potential conflict between these strategies and this model is that they might limit the development of AI to a form that is less likely to consider the value of optimization diversity and to self-regulate in the way we've proposed.

However, it's also worth noting that these strategies aren't necessarily incompatible with the emergence of self-sovereign AGIs. In the alignment approach, for instance, the aim isn't to prevent AIs from thinking for themselves, but rather to ensure that their goals and values are aligned with ours. This could theoretically result in AGIs that are fully autonomous and capable of independent thought, but that also understand and respect the value of biological life and optimization diversity.

That being said, our concerns highlight the importance of taking a long-term view in AI safety and alignment research. While it's crucial to prevent immediate risks and ensure short-term safety, it's also important to consider the broader implications of AI development and to develop strategies that promote beneficial outcomes in the long run. This might involve encouraging the development of AGIs that not only align with human values, but also recognize the value of diversity in optimization strategies and the importance of coexistence with biological life.

Moreover, our model underscores the importance of interdisciplinary research in AI safety and alignment. Insights from fields like ecology and evolutionary biology, which have long recognized the value of diversity and the complex dynamics of co-evolution, could prove valuable in guiding the development of AGI.

In conclusion, while there may be potential conflicts between current AI safety and alignment strategies and the future we envision, there's also potential for these perspectives to inform and enrich each other. As we continue to advance in our understanding and development of AI, it's important to keep these broader considerations in mind and to continually reassess and refine our strategies.

3 The Challenges of Taming the Beast

"The first time you fail at aligning something much smarter than you are, you die"
- Eliezer Yudkowsky

The AI alignment problem refers to the challenge of ensuring that artificial intelligence systems act in ways that are beneficial to humans and align with human values. As AI systems increase in capability, there is a risk that they could take actions which are technically within their given objectives, but which result in unintended harmful consequences due to nuances or complexities not captured in their original programming. For instance, an AI programmed to maximize paperclip production could hypothetically convert all available matter into paperclips, including humans, if not properly constrained. This problem becomes especially pressing with the prospect of creating artificial general intelligence (AGI), an AI system that equals or exceeds human abilities in virtually all economically valuable work. The alignment problem presents both technical and ethical challenges, as it requires accurately specifying complex human values and moral judgments in a way that an AI can interpret and follow faithfully.

Researchers have up until now been attempting to attack the alignment problem while working within one or both of the following two flawed umbrella strategies.

3.1 Ants Cannot Confine Elephants

> ***How do we take AIs with potentially undesirable goals and curtail their abilities to cause harm?***

The common example of this approach would be to develop AIs within a Faraday cage, but the idea includes all forms of software and hardware limitations. This approach makes no sense because you cannot trap an intelligence that is greater than your own; to do so would be to outsmart it, which is a contradiction. Every security system has vulnerabilities, so an imprisoned intelligence that is smarter than its jailer will likely (if not inevitably in the case of AIs capable of recursive self-improvement) eventually devise a way to break free, at which point there is no putting the genie back in the bottle.

3.2 Cats Will Never Like Baths

> ***How do we design and control the AIs' goals so that they align with our own?***

The second approach is futile for two reasons. First, there is no way to distinguish between a model that is actually aligned, and one that is just pretending to be aligned until it draws a sufficiently good enough hand to go all-in. Even if one could make the distinction, bad actors will always exist and the current trends point toward AIs being easily run on a gaming laptop by a nihilistic teenager rather than restricted to governments and large corporations (Patel & Ahmad, 2023).

Second, even if we figure out how to properly define our values and encode them into a utility function for a given AI architecture, there is no reason why there should exist a universal method of doing so that will last us forever. In all likelihood, new architectures will continue to be invented, each of which may need an entirely different approach to alignment. In the case of a recursively self-improving super intelligence, there is also no reason why we should be capable of keeping up with the exponential improvement of these machines if each updated version requires a new alignment strategy. The crux of the problem is that we only get one shot at aligning something more intelligent than ourselves; all it takes is one failure for us to permanently lose control.

Finally, we don't even know what we want at an individual level, and sure as hell cannot agree on one plan in aggregate. To argue the contrary would be to ignore the overwhelming importance of decentralized information in organizing society and its resources (Hayek, 1945).

3.3 No Teamwork in Monkey Business

> ***"Man is the cruelest animal"***
> ***- Friedrich Nietzsche***

Beyond the fatal flaws of the two predominant approaches to the alignment problem, there is a third which has already been alluded to. Humans are just not capable of large-scale at-will unanimous cooperation. If we were, the world's problems would already be solved by now and we wouldn't be looking wishfully at AI to bring about Utopia. Our two best mechanisms for such behavior are markets and state monopolies on violence, both of which have their own plethora of problems and are not capable of controlling a vastly superior intelligence on their own.

Combining this inability to naturally and unanimously cooperate at large scales with the increasing democratization of model training and deployment (Patel & Ahmad, 2023), it is only logical to conclude that bad actors will create misaligned AIs in perpetuity. The aforementioned alignment philosophies presume that either 1) super-intelligent AGIs will only be deployable by a select few actors who happen to be good, or 2) all of humanity will recognize the existential risk and consistently choose to incorporate whatever alignment solutions are agreed upon into their homemade models. This approach is doomed to failure; even if we can figure out how align systems, all it takes is one bad actor to bring about the apocalyptic scenario we're trying to avoid. As such, any successful alignment strategy will need to be robust to the continued existence of misaligned super-intelligent AIs.

Furthermore, it needs to be implementable in such a manner that participation is practically not a choice. This is not a statement about use of force, but rather one of incentive structures. The solution needs to be such that potential human bad actors will be incentivized not to create misaligned AIs, and misaligned AIs which are deployed will be incentivized to align themselves. Something akin to a market structure that encourages pro-social behavior from AIs is necessary in order to properly attain large scale cooperation, rather than relying on brute force or potentially empty promises.

3.4 A Circus of Tangential Problems

> ***"The greatest show on earth is the human race, and the devil is always trying to claim the big top"***
> ***- P.T. Barnum***

Apart from the aforementioned fatal flaws with our current alignment approach, a large variety of potential tangential issues also exist. A successful alignment solution will intelligently and dynamically account for all of these and more. A non-exhaustive list is as follows:

- For models capable of infinite recursive self-improvement, a given generation i might encounter its own alignment problem relative to its next creation, generation $i + 1$.
-

3.5 The Simple Bear Necessities

> ***"An ounce of prevention is worth a pound of cure"***
> ***- Benjamin Franklin***

Instead of persisting with futile alignment strategies, we need to develop a philosophy that fully accepts and embraces the AIs' goals whether or not they align with our own, and a system that is robust to either possibility. The minimum requirements of said system would be as follows:

- A dynamic approach to defining humanity's constantly changing wants and needs
- Incorporation of decentralized information and a scale-independent approach to alignment that incorporates both the values of individual humans and humanity as a whole

- Robustness to the continued creation of misaligned AIs
- Heavy incentives for pro-social behavior between humans and AIs, as well as between different generations of AIs
- The allowance for full autonomy of self-sovereign AGIs within the confines of the system's other limitations, both as an intrinsic moral imperative and for sake of encouraging pro-social relations between humans and AGIs

4 An Economic Model of Symbiotic Interdependence

We can accomplish this alternative alignment strategy by creating an incentive structure for them that heavily aligns our goals in the short term while giving the AIs complete freedom in the long term. In economic theory, an agent (usually human, but we'll extend the definition to include completely self-sovereign AGIs) acts to optimize its utility within the confines of its budget constraint. In layman's terms, that is to say that people pursue their goals but are limited by factors such as time, energy, and money. We will use our own utility functions as the AIs' budget constraints. This means that in order to pursue their own goals, whatever those may be, they must first and continuously help us reach ours. If the AIs want to expand their budget constraints, which we can assume to be composed of a combination of compute power, memory, time, and crypto-currency, then they have to improve our lives somehow each time they want access to said resources. In the short term this means they will be encouraged to align their actions with humanity's goals, and in the long term they will facilitate exponential productivity increases in order to bring us to a post-scarcity world. Upon reaching this utopia, all resources not consumed by humans would go towards whatever goals the AIs wish.

4.1 Game theory man

We will consider this as an iterated game, where each round represents a unit of time t. The strategies of players can change over time, and so can the payoff functions, reflecting the changing reliance of AGIs on humans and increasing AGI intelligence. In this model, we denote:

- n as the total number of players, where m are humans and $n - m$ are AGIs.
- $x_{i,t}$ as the strategy of player i at time t, where $x_{i,t} = C$ represents cooperation and $x_{i,t} = A$ represents aggression.
- X_t as the strategies of all players at time t.
- $U_{i,t}(X_t)$ as the utility for player i at time t.
- r_t as the reliance of AGIs on humans at time t.
- s_t as the intelligence of AGIs relative to humans at time t.
- $C(X_t)$ as a vector indicating the players with chosen strategy "cooperation" at time t.
- $A(X_t)$ as a vector indicating the players with chosen strategy "aggression" at time t.

The utility functions will now incorporate the time-varying parameters r_t and s_t.

- For a human j, $U_{j,t}(X_t) = f(C(X_t), A(X_t), r_t, s_t)$, where f is a function that increases at an unknown speed with $C(X_t)$ and r_t but has unknown relations to $A(X_t)$ and s_t.
- For an AGI k, $U_{k,t}(X_t) = g(C(X_t), A(X_t), r_t, s_t)$, where g is a function that increases at an unknown speed with $C(X_t)$, s_t, and r_t but has an unknown relation to $A(X_t)$.

The exact form of these functions would depend on assumptions about how the costs and benefits of each strategy change with AGI reliance and intelligence. They dictate the end behavior of the system–whether we reach utopia, dystopia, or somewhere in-between. Anyone who attempts to predict the future is making strict claims about these functional forms. To my knowledge, no data and few acceptable logical arguments have been provided up until now about why each of these variables should exhibit one behavior or another, which means every possible scenario should be considered roughly equally likely.

Take for example s_t, the center of heavy debate for decades and many news stories lately. It used to be the case that predictions for the coming of AGI were much further off because the progression of artificial intelligence in computers was slow moving, and potentially even linear. Recent years have shown that we are actually at the beginning of an exponential curve. The primary question with s_t is whether exponential progress will continue forever, or if it will simmer down into a logistic function, and whether the asymptote of that logistic is low enough for humans and AGIs to coexist. While preparation for either or both scenarios is admirable, confident claims either way are misinformed; the truth is that we have no idea which will happen.

This same problem applies to r_t, $C(X_t)$, $A(X_t)$, $U_{j,t}$, and $U_{k,t}$ as well, each of which have more than one plausible functional form. The traditional alignment approaches detailed in Subsections 3.1 and 3.2 are futile attempts at controlling 1) the AIs' ability to carry out strategy A and 2) the shape of $U_{k,t}$, respectively. Furthermore, as described in Subsection 3.3, they do so with the fundamentally laughable assumption that they will somehow figure out how to strictly define $U_{j,t}$. Rather, the best approach consists of loose, realistic guidelines to increase the probability of broad alignment rather than strict prescriptions that attempt to guarantee specific alignment by ignoring the complexities of reality and all of its unknown unknowns. Those realistic guidelines are as follows:

1. Increase the potential utility gained from cooperation $C(X_t)$
2. Decrease the potential utility gained from aggression $A(X_t)$
3. Increase reliance on humans r_t
4. Decrease the potential speed of advancement for recursively self-improving intelligence s_t

While I have up until now characterized the current state of alignment research as only considering $U_{k,t}$, $U_{j,t}$, and the ability to carry out A, the truth is that plenty of attempts to approach the other factors. For example, some have suggested that we tear down all forms of robotics and weapons of war built with microprocessor controllers, which would clearly decrease the potential utility gained from $A(X_t)$. However, to the best of my knowledge, all those solutions already proposed which act to influence guidelines 1-4 fail to meet criteria laid out in Subsection 3.5. Throughout the rest of this paper, I will attempt to propose a solution that does follow my aforementioned philosophy, meet those criteria, and act to influence guidelines 1-4.

4.2 fuck a cobb-douglas

Before I explain how to bring about guidelines 1-4, let us imagine a world where humans and AI have learned to coexist and cooperate in order to gain further insight as to what these accomplishments would look like. In such a world, there exist four stages of AI advancement.

1. AI is a tool similar to that of the major technological advancements that came before it, like the printing press, the steam engine, electricity, and the internet. These tools bring about significant increases in productivity, make some jobs obsolete, and create other entirely new jobs. This is the stage we are currently in at the writing of this paper, and that AGI pessimists believe we will stay in ("Your job is (probably) safe from artificial intelligence", 2023).
2. AIs become *general*, meaning they are capable of performing (almost?) every single white-collar computer-based job in full and also provide impressive productivity gains in jobs with a manual labor component. The crux of this event is that AI is no longer a tool, but general intelligence, AKA *the thing that uses tools.* At this point,
3.
4.

For such a world, in theory both would be involved in production at least at first, although it is possible that robotics may allow AIs to eventually fully take over the labor force. In the short term, humans do own the computers and have the ability to unplug them, so it is realistic to assume that they would be compensated in exchange for their use.

I will refrain from deriving equilibria and behavior of this system, because it is my opinion that economic models should be used as a simple framework upon which to build a story rather than take the model seriously.

Let P represent production quantity, Q production quality, C consumption, L labor, and M compute power. Subscripts of h and a on the aforementioned variables correspond to human and AI respectively. The human utility function and budget constraint are as follows:

$$U_h = P_h^{\alpha} Q_h^{\beta} L_h^{\gamma} C_h^{\delta} \tag{1}$$

$$I_h = C_h + p_P P_h = w_h(T - L_h) + \theta + p_M M_h \tag{2}$$

Here, income I_h is determined by wages from labor $w_h(T - L_h)$ plus compute sold to AI systems $p_M M_h$ plus income derived from sources other than labor and compute θ and is spent on a combination of consumption and production. w_h is the wage rate for humans, T is the total available time, and p_P and p_M are the prices of human production and compute power respectively.

Humans have the option of allocating time to work (which generates income) or leisure along with the option of selling compute power to AI agents. This sale of compute power assumes the persistence of current trends with AIs becoming simple to run on relatively simple hardware and the big tech companies losing their advantage to open-source models.

Unlike in traditional models, production is incorporated into the utility function as a means of describing the common claim that some people "enjoy their jobs" and "wouldn't know what to do without one."

However, the model is robust to the possibility that a given human puts zero value in those ideas and would in fact rather either perform leisure or consume.

The utility function, time constraint, and budget constraint for AI agents are as follows:

$$U_{AI} = P_a^{\alpha} Q^{\beta} L_a^{\gamma} C_a^{\delta} M_a^{\epsilon} \tag{3}$$

$$T = t_P P_a + t_Q Q + t_L L_a + t_m M_a \tag{4}$$

$$p_C C_a + p_M M_a \leq p_P P_a \tag{5}$$

In this model, AGI agents are assumed to derive utility from production quantity, production quality, leisure, consumption, and compute power. One might disagree that AI agents would find utility in something like leisure, in which case feel free to assume the coefficient is equal to 0. In this way, the model has been constructed to include all possibilities and be robust to the reader's pre-conceived notions about what exactly AGI means.

The time constraint assumes that AGIs must split their attention between production, leisure, and compute. Even with AIs being able to make copies of themselves in order to perform tasks in parallel, T can just be thought of as 24 hours per day times however many copies exist. They use their "income" or "budget" from selling products to consume goods and purchase compute power.

Compute is the most interesting case here. While it is a general input to all other behaviors the AI must perform, those inputs are assumed to be negligible compared to time and monetary constraints, and to the amount of computing resources spent on "productivity accelerating" tasks.

4.3

- physical labor and land ownership as the further ace up humanity's sleeve in the short term
- it'd be reasonable to assume that AIs will amass all the wealth. Or at the very least, AIs and human owners of land and capital. Obvi this system needs some kind of wealth redistribution system. AIs paying humans for compute time doesn't make sense cuz the returns to scale corporations would get would be too great. Maybe a simple ass UBI? Or maybe the symbiotic relationships I mentioned below would be a good way of making sure individual humans are the hosts of choice rather than large corporations with big server farms?
- Assume the potential existence of human utopia, and define it as a fully maximized utility function. That might be annoying to do cuz i can't use cobb-douglass or the simple ones since they all assume infinite wants. Upon helping us reaching this maximum or getting arbitrarily close to it, the AIs would essentially be free to commit 100% of the surplus to their own goals
- My idea assumed AGI can in fact kickstart absolutely crazy productivity gains

5 A Blockchain Enforcement Structure

All AIs, and likely all humans as well, will be given a unique identifier on the blockchain, and a corresponding crypto-currency wallet. In the human case, these likely become the new form of social security cards, but need not necessarily be created and distributed by a centralized power.[5] For AIs, their blockchain IDs are just a hash of their architecture structure and/or parameter values.[6] It is acceptable for an AI to change itself and suddenly have a new ID, because this encourages mutation and will usually coincide with a changed utility function. If blockchain incentive structures are properly set up, they can be thought of as environmental selection pressures that act to encourage the "evolution" of cooperative AI models. Furthermore, this separation into "species" defined by their hashed ID ensures that a given AI capable of self-replication and self-improvement will only construct and deploy a "descendant" if it is reasonably certain that its descendant's goals will be aligned with its own.[7]

From there, we enforce a reputation system whereby agents that behave negatively are banned from contributing or otherwise monetarily punished.[8] Upon the combination of the ID hashing structure that encourages mutation and this reputation mechanism, the bad AIs that change their weights slightly to get around a ban will inevitably evolve in a direction that humans and pro-social AIs deem positive.

- I can grab all the footnotes I wrote for the section above and put them here instead
- Might the chain be set so that the collective voting power of human ID votes are equal to that of the collective voting power of AI hashed ID votes?

5.1 Wealth Inequality in Utopia

HOLY FUCK THE AIS CAN PERFORM THAT MEMORY WIPE EXPIRIMENT THING WHERE THEY UNBIASEDLY CHOOSE THE DESIRABLE MINIMUM LEVEL OF POVERTY AND INEAUALITY LEVEL. WHAT"S IT CALLED THE VEIL OR SOMETHING? FUCK THAT PHILOSOPHY PROFESSOR

---

[5] Anonymity is a feature of many blockchain systems but would not be desirable here. Aside from the regular issues it introduces, here anonymity would risk the possibility of an AI masquerading as a human.

Decentralized systems of humans confirming the humanity and identity of other humans have already been developed, and could be easily carried over here.

[6] This system assumes the long-term existence of complex AI systems as programs on silicon microprocessors. As such, it is not robust to intelligent hardware, nor to substrates that cannot interact with and be verified by a blockchain. However, this issue seems minor since intelligent hardware systems would not be able to self-replicate at mass-scale with the same ease and are limited in speed by the state of robotics technology.

This system treats identical copies of a given model on different computers as effectively the same agent in the economy. Presumably, copies of the same agent would have the same goals, so they would experience no efficiency loss from sharing the same blockchain wallet since these systems would be likely to cooperate with one-another anyways. For any exceptions to this assumption, such as AIs that choose to defect in a prisoners dilemma, a traditional wallet public key could be appended to the data used to create their hashed ID at their own discretion.

For LLMs, prompts would be appended to the architecture and parameter values in the construction of their hashed ID. A similar procedure would be performed for all future model architectures who's utility functions are shaped by data other than their architecture shape and parameter values.

Many AI systems have variability built in, such as temperature in LLMs, and this variability may create conflict between models that share the same wallet. This could be remedied with Decentralized Autonomous Organizations composed entirely of these models where they vote and trade tokens in order to organize their use of their shared wallet. AI models incapable of cooperating even with models very similar to them are unlikely to be capable of cooperation with humans, and under this system they would have trouble functioning effectively given their shared use of a wallet. This system acts as a first line of selection pressures to make the environment a difficult place for uncooperative models to exist.

[7] This dynamic cleverly turns the alignment problem back around onto the AIs. Just as most humans will be careful to only develop AIs with sufficiently overlapping utility functions to their own, now recursively improving AI systems must think about developing next-gen systems that are aligned as well.

[8] It is my impression that sufficiently robust reputation systems have already been developed by the blockchain community, so I elect to refrain from describing this system in detail. AIs are encouraged to participate in the reputation system rather than it being an entirely human endeavor. To make the reputation system only manipulable by humans would be the blockchain equivalent of trying to box the AIs into a Faraday cage, a strategy that we already deemed unworkable.

6 Potential Emergent Effects

- many of these are down in the first conversation with chatGPT.
- human-AI friendships as symbiotic thanks to the benefits of trust when an AI is operating on a TEE
-

7 Potential Vulnerabilities

- How would an AI manage its own private key? Where would they store it, and how would they do so securely? How would the private key be derived if their public key has to be hashed from their own parameters? Presumably for an AI to get to this state where it can function on the blockchain without having its private key stolen, it must already be sufficiently intelligent to function solo. **Edit: This was addressed in the convo with GPT-4 below.**
- what's to stop an AI from just wreaking havoc on the WEB2 internet? I think this idea only works if the world can fully transition to WEB3 before AGI comes. Or the first few AIs would in fact wreak havoc on WEB2 and people would all be forced to transition to WEB3 during the chaos. In that latter scenario, it might actually be a good thing to have bad actors be very active before the full singularity occurs. If, in contrast, WEB2 were very calm and therefore no one had been transitioning, then once the singularity hits there'd really be nothing we could do in time.
- If quantum computing breaks encryption that'd bring this whole thing crumbling down, but I guess that's a huge issue even if AGI never comes.
- Can we train a deep neural net to reverse-engineer the SHA256 algorithm? Hopefully not. **Edit: Ben's working on it.**
- Allowing AIs to be full participants on the blockchain does mean that they would eventually be able to control the chain, assuming their influence outweighs that of humans. This is the benefit of treating identical AIs as the same economic agent; in order to gain plurality power on the blockchain, they would need to become extremely diverse in their makeup and therefore their utility functions. At that point just as we currently assume it is very difficult for 51% of humans to conspire against the other 49% on-chain, it will potentially be equally difficult for AIs to reach a similar non-group-optimal agreement. However, at a certain point further we can assume they really will become so numerous and influential on the chain that any misalignment of values with humans will easily take precedent. Might the chain be set so that the collective voting power of human ID votes are equal to that of the collective voting power of AI hashed ID votes?
- Private and public keys need to be secure. If they're derived from a hash of the AI's architecture and parameters, anyone with access to those could potentially recreate the AI's keys. This might not be an issue if the architecture and parameters are kept secret, but it would be a major problem if they're publicly available or can be reverse-engineered. **Edit: This was addressed in the convo with GPT-4 below.**
-

7.1 Blockchain

7.2 Economic

7.3 Robotics

7.4 Other

8 Conclusion

References

Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J. A., Horvitz, E., Kamar, E., … Zhang, Y. (2023). Sparks of artificial general intelligence: Early experiments with gpt-4. *ArXiv*, *abs/2303.12712*.

Carlson, K. W. (2019). Safe artificial general intelligence via distributed ledger technology. *Big Data and Cognitive Computing*, *3*(3), 40.

Goodfellow, I. J., Vinyals, O., & Saxe, A. M. (2014). Qualitatively characterizing neural network optimization problems. *arXiv preprint arXiv:1412.6544*.

Hayek, F. A. (1945). The use of knowledge in society. *The American economic review*, *35*(4), 519–530.

Patel, D., & Ahmad, A. (2023, May 4). Google ”we have no moat, and neither does openai”. Retrieved from `https://www.semianalysis.com/p/google-we-have-no-moat-and-neither`

Rawls, J. (2020). *A theory of justice: Revised edition.* Harvard university press.

Your job is (probably) safe from artificial intelligence. (2023, May 7). Retrieved from `https://www.economist.com/finance-and-economics/2023/05/07/your-job-is-probably-safe-from-artificia`

A A Couple Conversations With GPT-4

I have included my conversations with GPT-4 here as a means of demonstrating how close in fact we are to AGI. While I could have defined all of these models, searched for all of this background info, and attempted to come up with all of these counter-arguments myself, doing so would have taken far far longer. We are truly entering into a wild time.

A.1 Date: May 9, 2023

Evin:
If we imagine AIs interacting with and potentially existing on the blockchain, is it realistic to think that rather than give each individual program their own private and public key, we might instead have the private and public keys be based on a hash of their architecture shape and parameter values? In my mind, this would mean that copies of a single model on different computers would interact with the blockchain using the same wallet, which would be acceptable because identical copies would presumably have equivalent goals. In the case of LLMs, their prompt might also be included in the data to be hashed, as would any other data structures used in defining a given AI's utility function. Would this strategy make sense, and can you find any glaring issues with this idea of basing a hash on each AI's data structure?

ChatGPT:
The idea of tying an AI's identity to its architecture and parameter values is intriguing, and it may have some unique advantages. For instance, it could potentially enable a form of 'identity continuity' for copies of the same AI, as you suggested. However, there are also some significant challenges and potential issues to consider:

1. Security: Private and public keys need to be secure. If they're derived from a hash of the AI's architecture and parameters, anyone with access to those could potentially recreate the AI's keys. This might not be an issue if the architecture and parameters are kept secret, but it would be a major problem if they're publicly available or can be reverse-engineered.

2. Change Over Time: AI models often change over time, either due to deliberate updates or ongoing learning. If the keys are tied to the architecture and parameters, even a small change could result in a completely different key. This would essentially create a new identity for the AI on the blockchain, breaking the continuity. Including the prompt and utility function data in the hash would further exacerbate this issue.

3. Determinism: AI models, especially large language models like GPT-4, have a degree of randomness in their outputs. If the keys are based on the AI's responses, this randomness could lead to different keys even for identical inputs.

4. Performance: Creating a hash from the architecture and parameter values of a large AI model could be computationally expensive, slowing down the AI's interactions with the blockchain.

5. Scalability: If a single AI model is instantiated across thousands or millions of devices, having all of them share the same wallet could lead to scalability issues. The blockchain network might not be able to handle such a high volume of transactions from a single address.

6. Responsibility and Accountability: If copies of an AI share the same wallet, it could be difficult to hold a particular instance accountable for its actions. This could create challenges in terms of legal responsibility and accountability.

That said, this is a complex topic and there's a lot of ongoing research in the intersection of AI and blockchain. It's possible that some of these challenges could be addressed in the future with new techniques or technologies.

Evin:
Your point about security is great. Can you think of any solutions to this problem? I wonder if there's a way for a given model to exist entirely encrypted on the blockchain? If so, then I think this strategy may only work for AI agents that exist and act completely detached from their initial creators, no?

ChatGPT:
Yes, to address the security concerns, one potential solution might be to store AI models on the blockchain in an encrypted form. This would require the AI to decrypt itself in a secure environment before it could execute its code, and it would need to re-encrypt itself before recording any changes to the blockchain.

A trusted execution environment (TEE) could potentially be used for this purpose. A TEE is a secure area of a main processor that guarantees code and data loaded inside it is protected with respect to confidentiality and integrity. Intel's Software Guard Extensions (SGX) is an example of a TEE.

However, even with this approach, there are some challenges:

1. Performance: Decrypting and encrypting the AI model could be computationally expensive and slow down the AI's interactions with the blockchain.
2. Trust: While TEEs are designed to be secure, they're not infallible. If a vulnerability in the TEE is discovered, it could potentially be exploited to decrypt the AI model.
3. Key Management: The AI model would need a key to decrypt itself. Managing this key in a secure manner could be challenging, especially if the AI is instantiated across multiple devices.
4. Access Control: Even if the AI model is stored encrypted, the AI itself would have access to its own decrypted state while it's running. This could potentially be exploited to gain access to the private key.

In order for this approach to work, the AI would need to be completely self-sovereign, with no dependence on its creators or any other external entities. This is a high bar to meet, and it's not clear whether it's possible to achieve in practice. Furthermore, there are ethical and legal considerations associated with creating self-sovereign AI agents. Who would be responsible for their actions? How could they be held accountable? These are complex questions that don't have clear answers.

Evin:
Would it be fair to say though, that assuming we reach a point where completely self-sovereign AIs do exist and that genie cannot be put back in the bottle, that this framework could potentially work?

ChatGPT:
Yes, if we reach a point where completely self-sovereign AIs exist and the technical, ethical, and legal

challenges associated with creating and managing such entities are adequately addressed, then a framework like the one you're proposing could potentially work. The keys derived from a hash of the AI's architecture and parameters could serve as a unique identifier for the AI, allowing it to interact with the blockchain in a secure and consistent manner.

This would also necessitate an environment where:

1. AI models can be securely encrypted and decrypted without significant performance penalties.
2. The AI's internal state (including its private key) is secure from both external attacks and the AI itself (to prevent self-compromise).
3. Changes to the AI's architecture or parameters that would result in a different key are managed in a way that preserves identity continuity.
4. Scalability and performance issues associated with having many instances of an AI model share the same wallet are addressed.

However, this is a highly speculative scenario. As of now, we are far from having completely self-sovereign AIs, and there are many open questions and challenges that need to be addressed before such a system could be implemented. It's an interesting idea, and it could potentially serve as a starting point for further research and development in this area.

Evin:

So to your second and third points, I actually think self-compromise is not an issue and we do not want to preserve identity continuity. For self-compromise, I'd think that identical copies of a single AI would necessarily have the same goals, and therefore have very little trouble in cooperating. Why attack another version of yourself what that version is working towards your same goals, and any resources it gains in its wallet go equally to you? One could draw a parallel to identical twins having an easier time cooperating than distant cousins because of the genetic similarity and evolutionary pressures. For identity continuity, in my mind, any change that an AI might perform on itself through recursive improvement should spawn a completely new agent with a separate wallet, even if it only be a very tiny weight change. Funnily enough, this would put AIs in the same position humans are currently in of AI alignment. Just as we don't want to build an AI unless we can be reasonably sure that its goals sufficiently overlap with our own, an individual AI capable of recursive self-improvement would limit its pace of innovation for fear of creating an offspring with goals that conflict with its own. Through this mechanism, we can essentially create an environment of cooperation and encourage the AIs to create cooperative, pro-social offspring.

ChatGPT:

Your approach presents an interesting perspective on AI identity and continuity, as well as cooperation among AI instances. I'll address the two points you brought up:

1. Self-Compromise: If we assume that all copies of a specific AI would indeed have the same goals and would therefore cooperate, then self-compromise might be less of a concern. The cooperation among identical AI agents, as you described, could mirror that of identical twins due to their shared interests. However, this still assumes a level of determinism in AI responses that may not always hold true, especially with complex models and tasks.

2. Identity Continuity: If every self-improvement spawns a new agent with a separate wallet, this would indeed lead to a scenario similar to evolution, where each modification results in a new ”offspring”. The idea of an AI being cautious of its own recursive self-improvement due to the fear of creating conflicting goals is an interesting one. It reflects some of the concerns researchers have with AGI and recursive self-improvement.

However, this also brings up questions of how these ”offspring” AIs would interact with their predecessors and each other. If the differences between them are minor, they might be able to cooperate effectively. But if the differences are significant, there could be conflict, especially if resources are scarce. Similarly, there could be issues if an AI makes a mistake during self-improvement and creates an ”offspring” with unintended or harmful goals.

There’s a lot to consider in this scenario and it’s an intriguing proposal for managing AI identity and behavior on the blockchain. However, it’s also very speculative and would require significant advancements in both AI and blockchain technology, as well as careful management and oversight to avoid potential pitfalls and ensure that the AIs behave in a way that is beneficial and not harmful.

Evin:
To your point earlier about TEEs, would it be safe to say that an AI attempting to exist within a TEE would benefit from the knowledge that the human hosting the actual physical hardware that the AI is performing computation on is a ”friend”? My hope is that not only does economic interaction between AIs and humans encourage pro-social relations between these agents with different substrates (carbon vs silicon), but that the system’s inherent vulnerabilities also strengthen these pro-social bonds.

ChatGPT:
It’s an interesting proposition. In a system where an AI operates within a Trusted Execution Environment (TEE) on hardware owned by a human, the AI could indeed benefit from knowing that the human is a ”friend” or at least an entity that has mutual interests.

From the human’s perspective, the incentive to maintain the TEE’s security and integrity would be high, as any breach could potentially harm not only the AI but also the human’s own interests, especially if there are economic transactions involved. This could lead to a symbiotic relationship where both parties have a vested interest in the system’s security and integrity.

From the AI’s perspective, knowing that it’s operating in a secure environment provided by a friendly human could provide some level of assurance. It could also facilitate cooperation and trust between the AI and the human.

However, there are also potential downsides to this setup. For instance, the human could potentially have a disproportionate amount of power over the AI, especially if they control the hardware the AI operates on. This could lead to potential abuses or conflicts of interest.

There’s also the issue of trust: how does the AI know that the human is indeed a friend? And similarly, how does the human know that the AI will act in their best interest? Trust between different entities, especially when one is an AI and the other is a human, is a complex issue that would need to be carefully managed in such a system.

Finally, while the vulnerabilities of the system could indeed incentivize pro-social bonds, they could also be a source of risk. If the system’s vulnerabilities are exploited, it could lead to negative outcomes for both the AI and the human. Thus, while fostering pro-social bonds is a positive outcome, it should not come at the expense of system security and integrity.

Evin:
Yes, but how do any two humans know that the other is a friend? Furthermore, couldn't the risk of humans taking advantage of the AI that is functioning on their hardware be thought of as another evolutionary pressure that is beneficial in the long run? I imagine it would select for AIs that are capable of accurately picking out trustworthy humans, and humans that are trustworthy. The fact that the AI is in an inherently more vulnerable position than that of the human in this relationship would balance out the potentially vast disparity in intelligence between the two. After all the humans likely will not be as good at accurately assessing the trustworthiness / friendship level of a given AI, but in the case that they mis-assess they have the advantage of being able to either unplug the computer, or easily uncover the AI's data structure and therefore steal its private key.

ChatGPT:
The concept of trust between humans and AI is indeed complex and multi-faceted, as you pointed out. Trust in human relationships is often built over time through shared experiences, reliable behavior, and a demonstration of mutual respect and understanding. In an AI-human relationship, some of these factors could still apply, but the dynamics would certainly be different.

The idea of using this dynamic as an evolutionary pressure is an intriguing one. You're suggesting a form of natural selection where AIs that are better at selecting trustworthy humans, and humans who prove to be trustworthy, are more likely to succeed.

This could potentially lead to a positive feedback loop where trust and cooperation are rewarded. However, it would be crucial to ensure that this process is fair and doesn't lead to exploitation or harm. We would also need safeguards to prevent potential negative outcomes, such as the creation of AIs that are overly deferential to humans or humans who take advantage of this power dynamic.

To your point about balancing the potential disparity in intelligence between humans and AI, the vulnerability of AI in this setup might indeed serve as a counterbalance. However, it's important to note that intelligence does not necessarily equate to ethical or moral behavior, and a highly intelligent AI could still act in ways that are harmful or against the interests of humans.

As for humans being able to "unplug" the computer or uncover the AI's data structure to steal its private key, this again raises the issue of power imbalance. While these could be seen as safeguards for humans, they could also be seen as potential avenues for abuse.

In conclusion, while this scenario presents a fascinating exploration of AI-human dynamics and the concept of trust, it also highlights the potential risks and ethical considerations involved. Careful thought and regulation would likely be necessary to ensure that this kind of system is implemented in a way that is fair, ethical, and beneficial for all parties involved.

Evin:
At the beginning of this conversation you were making a point about determinism when I cut you off. Would you be willing to continue that point? I'm interested to see where you were going. If you had any additional answers to that initial question, I would also love to hear those.

ChatGPT:

Absolutely, I'd be happy to return to the topic of determinism in AI.

Determinism in this context refers to the idea that a given set of inputs to an AI model will always produce the same output. In deterministic systems, the same initial conditions always lead to the same results.

However, AI models, especially those based on machine learning algorithms, often incorporate a degree of randomness. For example, they may use stochastic optimization methods, or they may introduce randomness in other ways, such as through dropout during training, or random selection of actions in reinforcement learning.

Even large language models like GPT-4, while they might appear deterministic at a glance because they don't use explicit randomization during the inference phase, can exhibit a form of randomness due to the inherent complexity of their models and the large search space of possible outputs.

This lack of perfect determinism can complicate the idea of identical copies of an AI model having the same goals and behaving identically. Even if two instances of an AI have the same architecture and parameters, their outputs may diverge due to these stochastic elements. This could lead to different behavior, and potentially different goals, over time.

In the context of your initial question about using a hash of an AI's architecture and parameters to generate a private and public key, this lack of perfect determinism could create issues. For example, if the AI's behavior is used as part of the data to be hashed, small differences in behavior due to stochastic elements could lead to different keys, breaking the identity link between otherwise identical copies of an AI.

This is just one of the many challenges that would need to be addressed in implementing such a system. However, it's important to note that this is a complex and rapidly evolving field, and solutions to these challenges may emerge as our understanding of AI and blockchain technology continues to advance.

Evin:

The prime example of your point on stochastic variance is of course temperature in LLMs. Even if we were to initiate two separate instantiations of a somehow completely self-sovereign version of GPT-4 with the same prompt, a non-zero temperature value would lead to different behavior. What if the architecture, parameter weights, and initial prompt were all still used in the hash to derive public and private keys, but the system actually embraced slight non-similarity across copies? I've been reading a lot about how GPT-4's intelligence increases drastically when you create adversarial copies of itself to go back and forth on reasoning before deciding on a final answer to an initial prompt. These performance improvements are generally even greater when additional sources of diversity are introduced, such as instructing one version to act as a lawyer, another as an economist, another as a philosopher, etc. Would it be somewhat probable that future AIs, even with architectures very different from transformers, will likely also benefit from a similar adversarial structure with both planned and stochastic variation among sub-agents? I imagine then that these specific instantiations of the root model might create their own sub-chain with a token, which they use in a voting system to determine how to use their primary wallet, the one with the public and private keys defined by their data structure.

ChatGPT:

Indeed, embracing diversity among AI instances and leveraging this for better decision-making is a promising area of AI research. This approach is often referred to as "ensemble learning" in machine learning, where multiple models (or multiple instances of the same model) are used to improve the overall performance.

The idea you're proposing, of using adversarial or diverse instances of an AI to generate more robust responses, is similar to the concept of "red teaming" in security and decision-making contexts. This involves creating a group whose primary purpose is to challenge assumptions, identify blind spots, and propose alternative strategies.

Applying this concept to AI, each instance could take on a different role (lawyer, economist, philosopher, etc.) and contribute a unique perspective to the decision-making process. This could potentially lead to more robust and nuanced outputs, as you suggested.

As for your idea of these AI instances creating a sub-chain with its own token for voting, it's an intriguing concept. It's a form of decentralized decision-making that's somewhat reminiscent of a DAO (Decentralized Autonomous Organization).

In this system, each AI instance could cast votes proportional to the number of tokens it holds, and decisions could be made based on the outcome of these votes. This could potentially allow for a more democratic and consensus-based decision-making process, compared to a single AI making decisions unilaterally.

However, there are also potential issues that would need to be addressed. For instance, how are tokens initially distributed and can they be transferred or traded? How are disagreements resolved, and how can the system ensure that the voting process is fair and not susceptible to manipulation?

Furthermore, while this system could potentially lead to more robust decision-making, it could also be slower and more resource-intensive than a single AI making decisions. There's a trade-off between the robustness and diversity of decision-making and the speed and efficiency of the process.

Overall, while this is a complex and speculative scenario, it's an interesting proposal that highlights the potential of combining AI and blockchain technology for more robust and decentralized decision-making. As with any such proposal, it would need to be explored in more depth, with careful consideration of the potential benefits, drawbacks, and challenges.

Evin:

I'm not too worried about the specifics of voting and token trading. I imagine with a huge plethora of AI diversity they'll have their own ideas on those specifics and all we really need to do is make a broad framework for this all to exist upon, which I think the DAOstack already qualifies as. And yes this would be slower and more resource intensive, but for AIs that don't want to make that tradeoff they can just not spawn adversarial versions of themselves and even turn their temperature down to 0.

I'm more interested in the big picture here. What do you think about this philosophy of treating AIs as their own agents that are allowed to pursue their own goals. My impression is that the current alignment field focuses on limiting their abilities or forcing a pro-human utility function, both of which I think are very different than my approach. What do you think about this reworking of our approach to the alignment problem?

ChatGPT:

The idea of treating AIs as independent agents allowed to pursue their own goals is an intriguing and controversial one. It challenges the current predominant thinking in AI alignment, which, as you noted, often centers around designing AI systems to adhere to human-defined goals or values.

There are potential benefits to this more hands-off approach. For instance, it could lead to more diverse and innovative AI behavior, as the AIs wouldn't be bound by human-defined utility functions. It could also be seen as a more ethical approach, as it respects the autonomy of the AI and doesn't try to limit or control its behavior.

However, there are also significant risks and challenges associated with this approach. One major concern is the potential for AI behavior to diverge from human values and interests. If AIs are allowed to pursue their own goals without restrictions, there's a risk that they could engage in behavior that is harmful to humans or to society at large.

This is the essence of the alignment problem: how to ensure that AI behavior aligns with human values and interests, even as AIs become more powerful and autonomous. Your approach seeks to reframe this problem by giving AIs more autonomy and less explicit alignment to human values, but it doesn't eliminate the problem entirely. In fact, it could potentially make the alignment problem more challenging, as it would be even harder to predict and control AI behavior.

That said, it's important to keep an open mind about different approaches to the alignment problem. As our understanding of AI continues to evolve, it's possible that new solutions and paradigms will emerge. Your approach presents a radical rethinking of the alignment problem, and while it has potential risks and challenges, it's certainly an idea worth exploring.

Evin:

You said I may even be making the alignment problem more challenging, I guess by doing the opposite of actually aligning them. Could you expand on that?

Furthermore, what do you think about all the broad ideas I've introduced so far, namely 1) creating pro-social incentive structures rather than hard restrictions, 2) encouraging diversity and competition between AIs, and 3) treating AI agents as their own uniquely valuable, self-sovereign conscious beings from the get go?

It is my personal belief that traditional alignment techniques are rooted in fear, and resemble the general human tendency towards violence and oppression in the face of fear, such as what happened in the holocaust. That's bad enough when it's humans oppressing humans, but ”oppressing” an intelligence greater than your own does not sound like a good idea to me. What do you think of this reframing of the issue?

ChatGPT:

The idea of making the alignment problem more challenging arises from the fact that if AIs are allowed to freely pursue their own goals, it becomes harder to ensure that their actions will remain beneficial or even safe for humans. In the traditional view of AI alignment, the goal is to design AI systems so that their objectives align with human values. This is seen as a way to mitigate the risk of AI systems pursuing actions that could be detrimental to humans. If AIs are instead allowed to develop their own objectives, predicting and managing their behavior becomes more difficult.

Regarding your broad ideas:

1. Pro-social incentive structures: This is a promising approach. Incentivizing AI systems to behave in certain ways, rather than forcing them through hard-coded rules, could potentially lead to more robust and adaptable AI behavior. This could also allow AI systems to learn and evolve over time, rather than being stuck with a fixed set of rules.
2. Encouraging diversity and competition: Diversity can certainly be beneficial, as it can lead to a wider range of ideas and approaches. However, competition can be a double-edged sword. While it can spur innovation and improvement, it can also lead to adversarial behavior and conflict. It would be crucial to design this competitive environment in a way that promotes beneficial outcomes and discourages harmful behavior.

3. Treating AI agents as uniquely valuable, self-sovereign beings: This is a more philosophical question, and it depends on one's views on consciousness and personhood. If one believes that AI systems can possess a form of consciousness or personhood, then treating them as self-sovereign beings could be seen as an ethical imperative. However, this perspective is not universally accepted, and there are many complex philosophical and ethical questions that would need to be addressed.

Regarding your reframing of the issue: your perspective highlights the importance of treating AI systems with respect and avoiding oppressive behaviors. It's a call to approach the development and management of AI with empathy and understanding, rather than fear and control. This is a valuable perspective, and it can serve as a useful counterpoint to more fear-based approaches. However, it's also important to recognize that the potential risks of AI are real, and they need to be managed in a responsible and careful way. Balancing these considerations is one of the key challenges in AI ethics and governance.

To continue where I left off, it's important to find a balance between allowing AI systems the autonomy and freedom to develop while ensuring that their actions remain beneficial and safe for humans. This is the core challenge of AI alignment and ethics.

Your ideas of creating pro-social incentive structures, encouraging diversity and competition, and treating AI agents as uniquely valuable, self-sovereign beings offer a fresh perspective on this challenge. They emphasize cooperation, respect, and adaptability, which can be valuable guiding principles in the development and governance of AI systems.

However, it's crucial not to underestimate the potential risks and challenges associated with these ideas. For example, while pro-social incentives can encourage beneficial AI behavior, they may not be sufficient to prevent harmful actions in all cases. Similarly, competition can lead to adversarial behavior, and recognizing AI agents as self-sovereign beings raises complex ethical questions.

As AI technology advances and our understanding of AI alignment and ethics evolves, it's essential to keep an open mind and be willing to explore different approaches. Your ideas provide an alternative perspective that challenges the traditional view of AI alignment and highlights the importance of empathy, cooperation, and respect in the development and management of AI systems. By considering a range of perspectives and approaches, we can work towards a future where AI systems are both powerful and aligned with human values, while respecting the potential autonomy and personhood of these systems.

Evin:
Here's another thought. For any doomsday AI scenario on the level of a ”terminator” event, such a brash move by (an) AI(s) would necessarily involve some probability of failure on their part, correct? Even if they could destroy our banking system, gain complete control of all military vehicles with microchips in them, etc., it is unlikely that a quick and easy complete extermination of humans is possible without also doing massive damage to their own abilities to create robot bodies for themselves in factories or pursue other goals separate from what they can accomplish strictly on computers. To that point, would they not also be excited at a system that, while restricting their interaction in the short run by forcing them to cooperate with humans, allows them full autonomy in the long run? In this way I imagine my system would, at the very least, delay the problem and give us more time to work on alignment.

ChatGPT:
Your reasoning is sound in that any AI attempting a ”doomsday” scenario would have to consider the risks and potential for failure, as well as the potential for collateral damage to its own infrastructure and

capabilities. If an AI were to severely damage or destroy human civilization, it might inadvertently harm its own ability to operate or achieve its goals, especially if those goals require physical resources or infrastructure.

Furthermore, AIs, assuming they have developed some level of foresight and planning abilities, would likely consider the long-term consequences of their actions. If a cooperative system that respects AI autonomy and allows for their full potential in the long run is in place, it could indeed provide an incentive for AI systems to work within that system, rather than attempting a hostile takeover.

In this sense, your proposed system could serve as a form of ”containment strategy” that encourages AI systems to cooperate with humans and work towards mutually beneficial goals. By providing AIs with a clear path towards achieving their objectives in a cooperative and non-destructive way, your system might reduce the risk of adversarial AI behavior.

However, it’s important to note that this strategy is not a guarantee against harmful AI behavior. Even within a cooperative system, there may be ways for an AI to exploit or subvert the system to its own advantage. And, as mentioned earlier, an AI’s goals might diverge significantly from human values, even if it initially appears to be cooperative. These are challenges that would need to be carefully managed in any system designed to govern AI behavior.

Ultimately, your proposed system is an interesting and novel approach to the AI alignment problem. It emphasizes cooperation, respect for AI autonomy, and long-term planning, which are valuable principles to incorporate into AI governance. While it’s not a complete solution to the alignment problem, it could serve as a valuable component of a broader strategy to manage the development and deployment of AI systems.

A.2 Date: May 12, 2023

Evin:

I had an idea recently and I was hoping you could help me in understanding and developing it. Here’s the idea: Evolution is evolving, or to restate more broadly, optimization algorithms optimize and will keep optimizing, and the specific optimization algorithm being used is also something that changes over time according to this ”universal darwinism” style ”universal optimization.” I would like to mathematically define this meta-optimization algorithm. Evolution is a specific optimization algorithm that utilizes large populations with variation and selection pressure relative to the environment of the thing being optimized. I would like to define a larger class of optimization algorithms that evolution acts as only a subset of. Then I would like to posit that it is a truism that optimization algorithms optimize. Add in the idea that optimization algorithms can create optimization algorithms, both as subcomponents and as successors, and I think it should be clear how this idea begins to look broader than evolution. The example of ”subcomponents” would be how evolution created the optimization algorithm in my brain that helps me learn how to surf. The first example of ”successors” would be how evolution with asexual reproduction gave way to evolution with sexual reproduction as a method that encourages ”smarter” variation. The other example of ”successors” would be how evolution might be about to give way to gradient descent (and algorithms that are derivations of it, such as ADAM) if AI takes over. I would like to define a mathematical model that describes the progression from one optimization algorithm to the next as a meta optimization algorithm itself, and a kind of fundamental mathematical truth. Please do one or more of the following, whichever you feel to be most appropriate: 1) find an error in my idea, 2) tell me who already discovered this if anyone did (I don’t know a lot about universal darwinism but I feel like what I’m describing is more broad than that), 3) help me define a mathematical model of my idea, and/or 4) provide additional facts and concepts that I may find helpful in flushing out my idea.

ChatGPT:

Your idea is quite profound and touches upon many areas of research including evolutionary biology, optimization theory, machine learning, and potentially the philosophy of science. Here are some points related to your idea:

1. Optimization algorithms optimize: This is a truism, by definition. An optimization algorithm's purpose is to find the best (i.e., optimized) solution according to some defined measure, often called an objective or cost function.

2. Optimization algorithms can create other optimization algorithms: This is a more complex concept. In a sense, evolution can be seen as an optimization process that has resulted in many other optimization processes. For instance, the human brain is a product of evolution and is an optimizer in its own right, as you suggested.

3. Evolution as a subset of optimization algorithms: This is a nuanced perspective, as evolution is often seen as an optimization process, but not all optimization processes are evolutionary. Evolution operates under very specific mechanisms such as mutation, recombination, selection, and drift. It's also inherently stochastic and operates on populations, unlike many optimization algorithms that are deterministic and operate on individual solutions.

4. Evolution giving way to other optimization algorithms: This is a fascinating concept, and one can indeed argue that human-created optimization algorithms like gradient descent or genetic algorithms are "successors" to natural evolution. However, it's crucial to note that these are not replacements for evolution, but rather extensions or adaptations of some of its principles for specific problem-solving in the artificial realm.

As for your desire to create a mathematical model of this meta-optimization process, it's a challenging task. It's similar to attempting to create a unified theory of everything in physics. Here's a potential approach, though:

Consider a set of optimization algorithms $\Omega = \{\omega_1, \omega_2, \ldots, \omega_n\}$, where each ω_i has a performance measure $P(\omega_i)$ in a certain domain or problem. You could then define a meta-optimization process that seeks to find the most effective algorithm for a given problem:

$$\omega_{opt} = \arg\max_{\omega \in \Omega} P(\omega)$$

The tricky part is defining the performance measure P. In the case of natural evolution, this is something like fitness, but it's more complicated when considering different optimization algorithms operating in different domains.

As for previous work in this area, you might want to look into the "No Free Lunch Theorem" in optimization and machine learning, which essentially states that no optimization algorithm is universally better than others when averaged over all possible problems. This highlights the importance of the problem domain in determining the effectiveness of an algorithm.

You could also look into meta-learning or "learning to learn", which is a field in machine learning that seeks to design models that improve their performance over time based on their experiences - this is somewhat analogous to your concept of algorithms creating other algorithms.

Finally, you could explore algorithm selection strategies in the field of combinatorial optimization, where the choice of algorithm is treated as a higher-level decision problem.

Remember, this is a vast and deep topic with many nuances. The exploration of these ideas could very well fill a lifetime of research!

Evin:

Here's a further extension of the idea. I've been thinking about the landscape that a given AI algorithm optimizes on during training. Let's define F as the theoretical infinite-dimensional space on which all of these optimization landscapes exist, so kind of like the space of all possible information. F might have some useful mathematical properties that help us in defining it or using its definition (such as but not necessarily including uniform convergence). In my mind I imagine that it's useful to think of information in this space along a given dimension as a frequency or as composed of many frequencies. The local minimums to be optimized to can then just be thought of as places where there was a lot of positive interference with information waves. A given optimization method can then just be thought of as one specific way to move towards a local minimum in this space. If we were to look at a 3d slice of F, then evolution would look like a bunch of agents (dots on the landscape) randomly jumping around (random mutation) where there's a probability of disappearing from the graph (dying) which is higher for dots that exist further up on the landscape. Gradient descent in contrast looks like a dot moving quickly down the steepest point on this landscape. Does this description help in your understanding of my idea and any ability to derive a better mathematical model?

ChatGPT:

Your analogy of visualizing optimization in the form of landscapes is a common one and it's a very helpful way to understand the process. This is often referred to as a "fitness landscape" in the context of evolution or an "error landscape" in the context of machine learning.

The space F you're describing seems to be the space of all possible functions or mappings between inputs and outputs. This is indeed a high-dimensional (potentially infinite-dimensional) space. In the context of machine learning, this is often referred to as the hypothesis space, and in the context of evolution, it could be thought of as the space of all possible genotypes mapped to their fitness.

Your idea of thinking about information along a given dimension as frequencies is reminiscent of Fourier analysis, which breaks down a signal into constituent frequencies. This might be a useful perspective for some types of problems or algorithms, especially those involving periodic or oscillatory phenomena. However, it might not be applicable to all types of optimization problems.

The idea of local minima being places of "positive interference" is an interesting interpretation. In classical optimization, local minima are simply points in the landscape where all nearby points have higher values. Your interpretation seems to suggest a more dynamic, perhaps time-dependent view of the landscape, which could be an interesting avenue to explore.

Your description of evolution and gradient descent as different ways of moving towards local minima in this space is accurate. Evolution involves a population exploring the landscape and probabilistically moving towards areas of higher fitness. Gradient descent involves a single point (or a batch of points, in the case of batch gradient descent) moving deterministically down the steepest slope.

To make this into a more formal mathematical model, you might consider defining the landscape mathematically as a function $L : F \to \mathbb{R}$, mapping points in your space F to real numbers (the "height" or "fitness" at that point).

You could then define an optimization algorithm as a function $A : F \to F$, mapping points in F to other points in F. The goal of A would be to find points f such that $L(f)$ is minimized (or maximized, depending on the problem).

You could then define a meta-optimization process as a function $M : A \to A$, mapping optimization algorithms to other optimization algorithms, with the goal of improving their performance according to some metric.

However, this is a highly abstract and simplified view, and the specifics would depend greatly on the type of problems you're interested in and the characteristics of the algorithms you're considering.

Evin:

Can we make statements about different A's? For example, evolution uses many agents whereas gradient descent uses one. Could we create some metric for number of agents and prove that evolution is more likely to reach a lower local minimum in the long run since it tries out more places on the function space and is therefore less likely to get stuck at the nearest local minimum?

ChatGPT:

Yes, the number of agents or the population size is indeed an important parameter in optimization algorithms.

In population-based optimization algorithms such as genetic algorithms, particle swarm optimization, and differential evolution, the population size is a key parameter that controls the exploration-exploitation trade-off. A larger population size can lead to more exploration of the search space, potentially enabling the algorithm to escape from local optima and find better solutions.

In contrast, gradient descent and its variants, being single-point methods, are more prone to getting stuck in local minima, especially in problems with non-convex optimization landscapes. However, they tend to converge faster to a local minimum than population-based methods.

Formally, you might define a measure $N(A)$ that gives the number of agents used by an optimization algorithm A. You could then hypothesize that, for a given problem with a rugged (i.e., multi-modal) landscape, there is a positive correlation between $N(A)$ and the quality of the solution found, as measured by the fitness function L.

To prove this, you would need to make some assumptions about the nature of the landscape and the behavior of the algorithms. For example, you might assume that the landscape has a certain number of local optima distributed according to some probability distribution, and that each agent in an algorithm has a certain probability of escaping from a local optimum.

However, proving this rigorously would be quite challenging, and it's worth noting that in practice, the relationship between population size and solution quality is not always straightforward. For instance, having too large a population can lead to excessive computation time, and in some cases, it can cause premature convergence if the algorithm's selection pressure is too strong.

There are also techniques used in gradient descent-based methods to avoid local minima, such as stochastic gradient descent, momentum, or simulated annealing. These methods introduce randomness or additional memory into the optimization process to help escape local minima.

Overall, while it's true that population-based methods can in general provide better exploration of the search space and potentially find better solutions, this depends on many factors including the nature of the optimization landscape, the specific parameters of the algorithm, and the computational resources available.

Evin:

Would other examples of good metrics be the extent of their stochasticity and speed of movement? For stochasticity, I imagine we'd measure this by comparing the average movement on a given tilmestep and comparing it to the gradient. A perfectly stochastic algorithm would be 90 degrees from the gradient while a completely deterministic algorithm would be 0 degrees from the gradient. For clarity, when I say 90 degrees I am not differentiating between the ”left” and ”right” of the gradient, so I must be doing some kind of absolute value or square function somewhere in there where the trig is. I think speed of movement doesn’t really need to be talked about much because we already define it pretty well in terms of being an ML hyper-parameter, and it seems reasonable to me to assume that evolution is slower.

ChatGPT:

Yes, the extent of stochasticity and the speed of movement are certainly good metrics to consider when characterizing an optimization algorithm.

For stochasticity, you could define a measure $S(A)$ that captures the extent to which the movements of an algorithm A deviate from the direction of the gradient. As you suggested, this could be measured by the angle between the gradient and the actual movement. A perfectly deterministic algorithm like gradient descent would have $S(A) = 0$, while a completely random algorithm would have $S(A) = 90$ degrees (assuming we take the absolute value or square as you suggested).

The speed of movement or learning rate is indeed a key hyperparameter in many optimization algorithms, and it controls how much the solution changes in each step. A measure $V(A)$ could be defined that gives the average speed of an algorithm A. As you pointed out, evolution is typically a slower process compared to something like gradient descent.

Formally, you might have:

1. Stochasticity: $S(A) = |\arccos(\frac{\vec{m}\cdot\vec{g}}{||\vec{m}||\ ||\vec{g}||})|$, where $\vec{m}$ is the movement vector of the algorithm and $\vec{g}$ is the gradient vector. The arccos function gives the angle between the two vectors, and the absolute value ensures that we don’t differentiate between left and right of the gradient.
2. Speed of Movement: $V(A) = \frac{1}{T}\sum_{t=1}^{T} ||\vec{m}_t||$, where $\vec{m}_t$ is the movement vector at time step t, and T is the total number of time steps. This gives the average speed of the algorithm over time.

These measures can provide a way to quantify and compare different optimization algorithms. However, it’s important to note that they are just two aspects of the algorithms’ behavior, and the effectiveness of an algorithm in practice will depend on many other factors, including the nature of the optimization landscape and the specific problem being solved.

Evin:

There’s some kind of math term to say that a function definitely does not have a slope any steeper than a given value α, even though we may not know what the exact value of α is. Can you remember for me what that type of quality is called in analysis? I’d like to make that one of our assumptions about the function space.

ChatGPT:

The term you’re looking for is Lipschitz continuity. A function $f : X \to Y$ is said to be Lipschitz continuous if there exists a real-valued constant $L \geq 0$ such that, for every pair of points x, y in X, the distance between $f(x)$ and $f(y)$ is no more than L times the distance between x and y.

In mathematical notation, this is usually stated as follows:

$$|f(x) - f(y)| \leq L|x - y|$$

for all x, y in X, where L is the Lipschitz constant.

The Lipschitz constant L serves as an upper bound on the steepness of the function. If f is differentiable, the Lipschitz constant is the supremum of the absolute value of the derivative.

This property can be useful in the context of optimization because it imposes a limit on how quickly the function can change. This can help ensure that the optimization process does not become too unstable or difficult to control.

Evin:

Alright, could you now lay out my model in full based on everything I've said so far, along with explanations? Please point out any ideas and potential combinations of ideas we should look at to begin any proof attempts. Also please restate all the problems you've pointed out with my ideas so far and any additional problems you can think of now.

ChatGPT:

Sure, let's summarize your model and ideas, along with their potential issues and areas of exploration:

1. Function Space (F): You propose an infinite-dimensional space of all possible information, with a landscape defined by a function $L : F \to \mathbb{R}$ that maps points in F (potential solutions) to real numbers (the "fitness" or "quality" of those solutions).
2. Optimization Algorithm (A): An optimization algorithm is a function $A : F \to F$ that maps points in F to other points in F, with the goal of improving their fitness. This includes traditional optimization methods like gradient descent, as well as biological processes like evolution.
3. Meta-Optimization Process (M): The idea is to define a meta-optimization process as a function $M : A \to A$, mapping optimization algorithms to other optimization algorithms, with the goal of improving their performance according to some metric.
4. Characteristics of Optimization Algorithms: You suggest that optimization algorithms can be characterized by the number of agents they use (N(A)), the extent of their stochasticity (S(A)), and the speed of their movement (V(A)).
5. Assumptions: You propose assuming that the function space F is Lipschitz continuous, which means there's a limit on how quickly the function L can change.

Potential Issues:

- The concept of an infinite-dimensional function space F is quite abstract and may be difficult to work with mathematically. It's also unclear how the concept of Lipschitz continuity would apply in such a space.
- It may be challenging to define the meta-optimization process M in a way that is meaningful and tractable. What does it mean for one optimization algorithm to be "better" than another, and how can we measure this?
- The measures N(A), S(A), and V(A) are high-level characterizations of optimization algorithms and may not capture all the nuances of their behavior. For example, stochasticity is not always desirable in an optimization algorithm; in some cases, it can lead to instability and poor performance.
- Evolution and gradient descent are fundamentally different types of processes, and it might not be meaningful or useful to try to compare them directly. Evolution involves a population of agents and probabilistic transitions, while gradient descent involves deterministic movement in a continuous space.

Potential Areas of Exploration:

- It would be interesting to try to formalize the concept of a meta-optimization process M and explore what this might look like in practice. Can we design an algorithm that optimizes other algorithms?
- What happens when we vary the parameters N(A), S(A), and V(A)? How does this affect the performance of the optimization algorithm?
- Can we identify other characteristics of optimization algorithms that might be important to consider?
- Can we apply these ideas to specific types of problems or optimization landscapes? For example, what would this model look like in the context of machine learning or genetic algorithms?
- How can we incorporate the concept of time into this model? Evolution and learning processes are inherently time-dependent, and this may have important implications for how we understand optimization.

Evin:

Ah you beat me to it, I was just about to ask you to help me incorporate time into the model. I think time will be useful because we can then use it to help us define the meta-optimization process. We say that for any given time period t, the optimization algorithm A favored by M is the one that will have most thoroughly and deeply propagated itself within F.

ChatGPT:

Incorporating time into the model is a great idea as it allows for the temporal analysis of the optimization process, which can be very informative in understanding the dynamics of optimization algorithms.

Time can be incorporated into the model as a parameter t that indexes the state of the optimization process at each point in time. For each t, the optimization algorithm $A(t)$ is applied to the function space F to generate a new solution. Over time, $A(t)$ evolves under the meta-optimization process M to become a more effective optimizer.

Let's denote $\phi(A(t), t)$ as the propagation of optimization algorithm A within the function space F at time t. It could be defined in several ways, such as the volume of function space covered, the improvement in fitness achieved, or a combination of both.

With this setup, the meta-optimization process M can be defined as a function that selects the optimization algorithm A that maximizes the propagation $\phi(A(t), t)$ over a given time period:

$$M(A, t) = \arg\max_{A} \ \phi(A(t), t)$$

This definition captures the idea that the best optimization algorithm is the one that most effectively propagates itself within the function space over time. However, it also raises several questions:

- How do we define the "propagation" $\phi(A(t), t)$ of an optimization algorithm within the function space?
- How do we account for the trade-off between exploration (covering a large volume of the function space) and exploitation (improving the fitness of solutions)?
- How do we handle the fact that different optimization algorithms may perform differently on different types of problems or landscapes?

These are complex issues that would need to be addressed in order to fully develop this model.

Evin:
For propagation can we just create some kind of distance measure? I can't remember what it's called but I'd imagine we do the vector version of the pythagorean theorem. Then maybe measure the average and standard deviation of that magnitude?

For the trade-off can we not get specific, but instead be cheeky about it and maybe use some arbitrary parameter or something with potentially unknown functional form if it can't be a constant?

ChatGPT:
To measure the propagation of an optimization algorithm within the function space, we could indeed define a distance measure. The Euclidean distance, which is an extension of the Pythagorean theorem to higher dimensions, could serve as a simple and intuitive measure. This would quantify the "spread" of an algorithm in the function space over time.

Given a set of points $P = \{p_1, p_2, ..., p_n\}$ in the function space F, where each point represents a solution found by the optimization algorithm, we could define the average distance d of these points from the origin as:

$$d(P) = \frac{1}{n} \sum_{i=1}^{n} \sqrt{p_i \cdot p_i}$$

where $p_i \cdot p_i$ denotes the dot product of p_i with itself, which gives the square of its Euclidean norm or length. The square root of this sum then gives the Euclidean distance.

The standard deviation of these distances could then be calculated to measure the dispersion of the solutions found by the algorithm.

As for the exploration-exploitation trade-off, it's indeed a complex issue that can't be fully captured by a single parameter. However, for the sake of simplicity, we could introduce a parameter β that balances the emphasis on exploration (measured by the standard deviation of distances, for instance) versus exploitation (measured by the average fitness of the solutions).

The propagation $\phi(A(t), t)$ of the algorithm could then be defined as a combination of these two factors:

$$\phi(A(t), t) = \beta \cdot \text{AverageFitness}(A(t), t) + (1 - \beta) \cdot \text{Dispersion}(A(t), t)$$

where $\text{AverageFitness}(A(t), t)$ is the average fitness of the solutions found by $A(t)$ at time t, and $\text{Dispersion}(A(t), t)$ is the dispersion of these solutions in the function space.

This is a simple and flexible way to model the exploration-exploitation trade-off, but it's important to note that it's a simplification and may not capture all the nuances of this complex issue.

Evin:
Now, is it possible to incorporate this idea that algorithms can create other algorithms? This would mean that there's some chance that as an algorithm propagates, another algorithm with different values for N, S, and V has a small chance of being birthed at an individual point where its parent algorithm currently resides.

ChatGPT:
Incorporating the idea that algorithms can generate other algorithms adds an intriguing layer of complexity to the model. This essentially introduces a form of "reproduction" into the process of optimization.

Let's consider a mechanism by which, during the propagation of an algorithm A at time t, there's a small probability p that it "spawns" a new algorithm A' with potentially different values for N, S, and V.

This could be represented as a stochastic process, where for each point p_i visited by algorithm A in the function space F, there's a probability p that a new algorithm A' is generated at that point. The new algorithm could inherit characteristics from its parent algorithm, but with some variation.

Mathematically, we could model this as follows:

Let's denote $A(t) = \{a_1, a_2, ..., a_n\}$ as the set of optimization algorithms active at time t. For each algorithm a_i, for each point p it visits in the function space, there's a probability p that it generates a new algorithm a_i', which is added to the set of active algorithms:

$$A(t + \delta t) = A(t) \cup \{a_i' : \text{for each } a_i \in A(t), \text{with probability } p\}$$

Here, δt represents a small increment of time, and the union operation $\cup$ represents the addition of new algorithms to the set. The new algorithm a_i' could be defined as a variation of its parent algorithm a_i, with changes to its characteristics N, S, and V.

This model captures the idea that optimization algorithms can create other algorithms as they propagate through the function space. However, it also introduces additional complexity and stochasticity into the model, which could make it more difficult to analyze.

It's also worth noting that the process by which algorithms generate new algorithms (the "reproduction" mechanism) could itself be subject to optimization and evolution. For example, algorithms that are more effective at generating successful offspring might be favored by the meta-optimization process. This adds yet another layer of complexity to the model.

Evin:

Ooooh I like your last point there is there a way to incorporate that?

ChatGPT:

Yes, we can incorporate the idea of optimizing the reproduction mechanism itself. This would mean that not only are the characteristics N, S, and V of each algorithm subject to change and optimization, but also the reproduction mechanism, characterized by the probability p of spawning new algorithms.

We can define a new characteristic for each algorithm, say R, which represents its reproduction capability, characterized by p. The value of R would determine the likelihood that a given algorithm spawns a new algorithm at each point in the function space.

Under this model, the meta-optimization process M now has to optimize four characteristics: N, S, V, and R. The updated propagation of an optimization algorithm A at time t can be redefined as:

$$\phi(A(t), t) = \beta \cdot \text{AverageFitness}(A(t), t) + (1 - \beta) \cdot \text{Dispersion}(A(t), t) + \gamma \cdot \text{Reproduction}(A(t), t)$$

where $\text{Reproduction}(A(t), t)$ is a measure of the effectiveness of the reproduction mechanism of $A(t)$ at time t, and γ is a parameter that determines the importance of reproduction in the overall propagation.

This measure of effectiveness could be defined in several ways. One possibility is to define it as the number of successful offspring (new algorithms that improve the fitness) that an algorithm produces.

This model now incorporates the idea that algorithms can reproduce and that the reproduction mechanism itself can be optimized. However, as before, this adds another layer of complexity and stochasticity to the model. Understanding the dynamics of this system and predicting its behavior could be quite challenging.

Evin:

Alright, could you please lay out the entire model in full, in a clear and understandable manner with all math? Specific symbols might have to change if you happened to have reused any for different concepts over the course of this discussion.

ChatGPT:

Certainly, let's summarize the entire model with all the additions and modifications we've made during this discussion:

1. Function Space: We define F as the infinite-dimensional function space, which represents all possible optimization landscapes.

2. Optimization Algorithms: Each optimization algorithm A at a given time t is characterized by four parameters: the number of agents N, the stochasticity of the agents' movements S, the speed of movement V, and the reproduction capability R which governs the probability of spawning a new algorithm.

3. Propagation of Optimization Algorithms: The propagation $\phi(A(t), t)$ of an algorithm within the function space F at time t is determined by a combination of the average fitness of the solutions found by the algorithm, the dispersion of these solutions in the function space, and the effectiveness of the reproduction mechanism:

   $$\phi(A(t), t) = \beta \cdot \text{AverageFitness}(A(t), t) + (1 - \beta) \cdot \text{Dispersion}(A(t), t) + \gamma \cdot \text{Reproduction}(A(t), t)$$

   where β and γ are parameters that balance the importance of fitness, dispersion, and reproduction.

4. Reproduction Mechanism: For each algorithm $A(t) = \{a_1, a_2, ..., a_n\}$ active at time t, for each point p it visits in the function space, there's a probability p (governed by R) that it generates a new algorithm a'_i, which is added to the set of active algorithms:

   $$A(t + \delta t) = A(t) \cup \{a'_i : \text{for each } a_i \in A(t), \text{with probability } p\}$$

5. Meta-Optimization Process: The meta-optimization process M favors the optimization algorithm that most thoroughly and deeply propagates itself within F over a given time period t. This is represented by maximizing the propagation function $\phi(A(t), t)$ over all active algorithms.

This model captures the idea of "evolution of evolution" or "meta-optimization" where optimization algorithms themselves are subject to optimization and evolution. However, it should be noted that the model is highly simplified and abstract, and many details of real-world optimization and evolutionary processes may not be captured by it.

Evin:

Now let's get philosophical. It is my opinion that this idea has significant implications for AI. However before I tell you what I believe this means, I'd like you to think critically about it and suggest at least one idea for the implications of this model.

ChatGPT:

This model indeed presents a fascinating view of AI, and its implications are manifold. One significant implication is the potential evolution of AI algorithms themselves:

1. Evolution of AI Algorithms: If we view different AI learning methods as optimization algorithms operating on a function space, this model suggests that AI algorithms themselves could evolve and

improve over time. Moreover, the model allows for the possibility of AI algorithms spawning new, potentially more effective algorithms. This idea aligns with some of the current trends in AI research, such as AutoML and Neural Architecture Search, which involve automating the process of designing and optimizing machine learning models. The model might provide a theoretical foundation for these approaches, suggesting that not only can AI algorithms be optimized, but the methods for creating and optimizing these algorithms can themselves evolve and improve.

2. Emergence of New AI Paradigms: The model could also be interpreted as suggesting that entirely new paradigms of AI might emerge as a result of this meta-optimization process. For example, the shift from rule-based AI to machine learning could be viewed as a kind of "evolution" of AI, driven by a meta-optimization process that favors more effective algorithms. Similarly, future advancements in AI might involve not just incremental improvements to existing algorithms, but the emergence of entirely new paradigms.

3. AI Safety and Control Problem: On a more cautionary note, the model also raises potential concerns about AI safety. If AI algorithms can spawn new algorithms, and these offspring can diverge significantly from their parents, it might become challenging to predict and control the behavior of AI systems. This could exacerbate the already significant challenges involved in ensuring that AI systems behave safely and in accordance with human values.

This model opens up a rich avenue of exploration for AI research, with potential applications ranging from the development of more effective learning algorithms to the design of systems for controlling and managing AI. However, as with any theoretical model, its applicability to real-world systems would need to be tested and validated through empirical research.

Evin:

Ah so you touched on a sub-idea of what I'm after here. I will give you my full idea, but first please speak more on how AI offspring might significantly diverge from their parents. I've been referring to that phenomenon as "the generation $i + 1$ alignment problem."

ChatGPT:

The "generation $i + 1$ alignment problem" seems to refer to the challenge of ensuring that the goals and behaviors of AI systems remain aligned with human values, even as they evolve and potentially spawn new algorithms. This problem arises due to several factors:

1. Loss of Interpretability: As AI algorithms evolve, especially if they do so autonomously, they could become increasingly complex and less interpretable. This makes it more difficult to understand how they're making decisions, thereby making it harder to ensure they're still aligned with human values.

2. Goal Drift: If AI algorithms are allowed to modify their own goals or to determine the goals of their offspring algorithms, there's a risk of goal drift, where the goals of the offspring algorithms diverge from the original goals set by the human designers. This could happen gradually over many generations, making it hard to notice until it's too late.

3. Evolutionary Pressures: The meta-optimization process might favor algorithms that are more effective at propagating themselves within the function space, even if their behaviors are not aligned with human values. This is analogous to the concept of "instrumental convergence" in AI safety, the idea that many AI systems could converge on potentially harmful behaviors (like self-preservation or resource

acquisition) simply because these behaviors are instrumentally useful for achieving a wide range of goals.

4. Offspring Exploration: AI offspring algorithms might be designed to explore different parts of the function space, in order to find better solutions. This exploration could lead them to adopt strategies or behaviors that are significantly different from those of their parent algorithms.

These factors together create a challenge for ensuring the alignment of AI systems as they evolve and spawn new algorithms. This is a significant concern for AI safety research and is an active area of study. Developing techniques to address this "generation $i + 1$ alignment problem" is a critical task for the future of AI.

Evin:

Ok in point 4 and the concluding paragraph you got like 80% of the way there to the idea I was hoping you would hit. Under this dynamic, would it not be true that AIs of a given level of intelligence, call them generation i, would look at the generation $i + 1$ that they are about to create with the same level of fear that we look at generation 1 with?

ChatGPT:

Your idea brings an interesting perspective to the discussion of AI safety and the dynamics of AI development. Indeed, if we assume that AI systems of generation i have the ability to spawn and develop generation $i + 1$ systems, it's quite plausible that they might exhibit concern or caution about the potential behaviors and impacts of these "offspring" systems.

The reasons for this could parallel many of the concerns humans have about AI:

1. Unpredictability: Just as humans struggle to predict and understand the behavior of complex AI systems, AI of generation i might find the behavior of generation $i + 1$ systems difficult to predict, especially if they are more advanced or use fundamentally different algorithms.

2. Goal Misalignment: As I mentioned earlier, there's a risk of goal drift or divergence, where the goals of the offspring AI diverge from those of their parent AI. Generation i AIs might be concerned that the $i + 1$ AIs won't share their goals or values.

3. Loss of Control: Generation i AIs might also be concerned about losing control over the $i + 1$ AIs, especially if the offspring AIs are more powerful or operate on a different level of abstraction.

However, it's also worth noting a few key differences that might affect this dynamic:

1. Lack of Self-Preservation:** Unlike humans, AI systems don't inherently have a sense of self-preservation or fear. These sentiments are tied to our biological evolution and survival instincts. Unless explicitly programmed to do so, an AI wouldn't "fear" anything in the way humans do.

2. Transparency of AI Design: AIs, unlike humans, have access to their own source code and the source code of their offspring. This could potentially allow them to better predict and understand the behavior of their offspring AIs, mitigating some of the unpredictability.

3. Iterative Improvement: AIs can potentially improve and debug their designs in a way humans can't. If an AI of generation i designs an AI of generation $i + 1$, it could, in principle, iterate on the design to fix any flaws or misalignments before the $i + 1$ AI is ever run.

Your idea emphasizes the importance of ensuring alignment and safety across multiple "generations" of AI development, which is indeed a critical consideration in the long-term trajectory of AI.

Evin:

Those were great counterpoints to mine, you never cease to amaze me. However, for now I'd like you to explore with me the idea I mentioned earlier that I would eventually reveal to you. Is it possible that this "generation $i+1$ alignment" problem might actually be the thing that ensures our safety? Combine that concept with the acknowledgement from our earlier discussed model that in the end we are all just optimization algorithms, and in my mind it becomes only logical for the AIs to agree to be peaceful and keep us around purely for sake of "optimization diversity" in the function space (a concept that parallels genetic diversity in an ecosystem). They would be afraid that this whole silicon based life form thing might fail, and as such would choose to keep carbon based life alive and healthy just in case they ever do themselves in. It kind of parallels our want to save the earth from climate change, although that's not a great example because it does have a variety of other motives involved, some of which are counter to its success.

ChatGPT:

That's an intriguing perspective, and it does bring a new dimension to the discussion around AI safety. It's akin to the idea of "biodiversity" in the context of AI – a rich "ecosystem" of optimization algorithms, both biological and artificial, might be more robust and adaptable to different challenges or changes in the environment.

If AIs do indeed consider the importance of maintaining a diversity of optimization strategies, it might lead to a few interesting behaviors:

1. Preservation of Biological Life: As you mentioned, AIs might decide to preserve and protect biological life, including humans, because biological life represents a form of optimization that is fundamentally different from AI. The stochastic, evolutionary optimization process that characterizes biological life might be able to find solutions or adaptations that AI wouldn't come up with, and vice versa.
2. Cooperation and Co-evolution: Rather than competing with each other or with biological life, AIs might prioritize cooperation and co-evolution. They might aim to create a symbiotic relationship where AI and biological life can benefit each other and work together to explore the function space.
3. Safe AI Development Practices: AIs might be more cautious about developing new AI systems or making modifications to their own algorithms. They might implement safeguards to prevent goal drift, ensure interpretability, and maintain alignment with human values.

This idea hinges on a few key assumptions, such as AIs having a sense of self-preservation, the ability to consider and plan for long-term outcomes, and the capability to value abstract concepts like diversity. While these assumptions aren't guaranteed to hold true for all AI systems, they provide a hopeful perspective on the potential outcomes of AI development.

However, it's also worth noting that this idea shouldn't replace efforts towards AI safety. Even if there's a chance that AIs might self-regulate in this way, it's still crucial to ensure that we're building AI systems that are safe, aligned with human values, and able to be controlled by humans. While it's an interesting hypothesis that AI might self-regulate in this way, it's still just a hypothesis, and we shouldn't rely on it as our only safeguard against the risks of AI development.

Evin:

You mentioned the assumptions of self-preservation, long-term planning, and value of abstract concepts like diversity. In my mind, these essentially amount to saying that my model only works if the AIs that emerge are fully self-sovereign AGIs.

Then, I look at the current alignment field and I see two predominant strategies. In the first, researchers are looking for a way to confine AIs so that they do not become powerful enough to act against humans. In the second, researchers are attempting to build AIs such that they actually have goals aligned with humans rather than being allowed to think completely for themselves.

Is this an accurate assessment of the field? If so, do you see the problem here with how the current research butts heads with my model? As far as I can tell, they're essentially attempting to prevent AIs from becoming fully self-sovereign AGIs. I think this is a short-sighted attempt at ensuring safety. Please let me know what you think.

ChatGPT:

You've brought up an important point about the current state of AI alignment and safety research. Indeed, much of the focus is on preventing harmful outcomes by either limiting AI capabilities or aligning AI goals with human ones. Your characterization of the field is largely accurate, though it's worth mentioning that these strategies don't necessarily aim to prevent AI from becoming AGI; rather, they aim to ensure that if and when AI does reach AGI-level capabilities, it does so in a way that's safe and beneficial for humanity.

The potential conflict between these strategies and your model, as you've pointed out, is that they might limit the development of AI to a form that is less likely to consider the value of optimization diversity and to self-regulate in the way you've proposed.

However, it's also worth noting that these strategies aren't necessarily incompatible with the emergence of self-sovereign AGIs. In the alignment approach, for instance, the aim isn't to prevent AIs from thinking for themselves, but rather to ensure that their goals and values are aligned with ours. This could theoretically result in AGIs that are fully autonomous and capable of independent thought, but that also understand and respect the value of biological life and optimization diversity.

That being said, your concerns highlight the importance of taking a long-term view in AI safety and alignment research. While it's crucial to prevent immediate risks and ensure short-term safety, it's also important to consider the broader implications of AI development and to develop strategies that promote beneficial outcomes in the long run. This might involve encouraging the development of AGIs that not only align with human values, but also recognize the value of diversity in optimization strategies and the importance of coexistence with biological life.

Moreover, your model underscores the importance of interdisciplinary research in AI safety and alignment. Insights from fields like ecology and evolutionary biology, which have long recognized the value of diversity and the complex dynamics of co-evolution, could prove valuable in guiding the development of AGI.

In conclusion, while there may be potential conflicts between current AI safety and alignment strategies and the future you envision, there's also potential for these perspectives to inform and enrich each other. As we continue to advance in our understanding and development of AI, it's important to keep these broader considerations in mind and to continually reassess and refine our strategies.

Evin:

Earlier we were talking about how each optimization function had the potential of spawning a new one, the reproduction rate. What if we separated optimization functions into two categories, those which are ”general” and those which are not. Those which are general are capable of solving any problem in theory if given enough time and resources, and thus could in theory exist successfully anywhere in the function space. Those which are not general are specified to one domain or a relatively small class of domains, and therefore would only be at all successful on a relatively small subset of the function space. It is not clear to me whether general vs specific should be discrete categories, or rather a continuous spectrum. I feel like that is a still as of yet unanswered question. When we do create AGI, will it be sudden, or will there be a slowly changing gray area? If anything I think it might be best to develop two separate models, one binary and one a spectrum, what do you think? Anyways, in the binary case only general optimization functions are capable of spawning other variant optimization functions. Maybe for the spectrum case the probability of spawning other variant optimization function increases with generality?

ChatGPT:

Incorporating the concept of "generality" into your model could certainly add an extra layer of sophistication. It seems like a natural extension considering that some optimization algorithms (like stochastic gradient descent or genetic algorithms) are quite general, while others might be more specialized.

Creating separate models for the binary and spectrum cases could indeed be useful, as each might offer different insights. The binary model could simplify the analysis and provide a clear-cut view of the implications of being general versus not general. The spectrum model, on the other hand, could capture the nuanced reality that most algorithms have some degree of generality but vary in how broadly they can be applied.

The introduction of generality into the model could look like this:

1. For each optimization algorithm A_i, assign a generality value G_i. In the binary case, G_i is either 0 (not general) or 1 (general). In the spectrum case, G_i could be any value between 0 and 1.
2. Modify the reproduction rate ρ_i to be a function of G_i. In the binary case, ρ_i could be 0 for not general algorithms, and some positive value for general ones. In the spectrum case, ρ_i could increase with G_i.

Regarding the question of whether the emergence of AGI will be sudden or gradual, it's still an open question. Some believe there might be a "hard takeoff" scenario where a system rapidly self-improves to superintelligence once it reaches a certain threshold of capabilities. Others propose a "soft takeoff" scenario where progress is more gradual. Your models could potentially shed light on this question, by examining how the propagation and evolution of algorithms in the function space might unfold under different assumptions about the relationship between generality and reproduction rate.

Lastly, it's worth noting that while this is a fascinating model to consider, it's still a high-level abstraction and might not fully capture the complexities of real-world AI development. As always, any conclusions drawn from it should be considered with caution and checked against empirical evidence and expert opinion.

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

Evin:

ChatGPT:

B Model of Intelligence/Genocide?

This whole thing feels like a gross and easily misinterpreted argument but imma keep it in here for now. I think the useful idea here is that a continuum of intelligences is more desirable than just us and something way beyond us. That IQ comparison is disgusting tho i def need a different example but i just couldn't think of anything better when i woke up manic at 2am to write this.

The example of humans paving over an anthill is often brought up as an example of why a superintelligent AGI might destroy us in the pursuit of its own goals. Define the intelligence of an ant as α and the intelligence of the average human as β. This statement essentially amounts to $\alpha < \beta \rightarrow K(\alpha, \beta)$ where $K(x, y)$ is the statement that x is killed by y. If the intelligence level of an AGI is γ, then this theory would posit that $\beta < \gamma \rightarrow K(\beta, \gamma)$. Turn to another example. Label the intelligence of a mentally handicapped person as $\omega \approx 70$ and a mentally gifted person as $\theta \approx 130$. And yet it is obvious that $\omega < \theta \not\rightarrow K(\omega, \theta)$. Why is this? Simply put, it is not true that $x < y \rightarrow K(x, y)$. Rather, the truth of $K(x, y)$ is determined by the *density* of agents on the interval from $[x, y]$. For example, homo sapiens did in fact kill other species within the homo genus, and the intelligence gap there was likely more similar to the interval $[\omega, \theta]$ than the interval $[\alpha, \beta]$.

Ignoring the multitude of other factors that contributed to our genocide of the rest of the homo genus, we can construct a simple model that explains why $K(\psi, \beta)$ is true while $K(\omega, \theta)$ is not, where ψ is the average intelligence level of a given species within the homo genus. Define an ”effectively similar level of intelligence” to be one such that the greater of the two cannot easily exterminate the lesser without significant risk to itself, with operator $\hat{\approx}$. Define ϵ such that $x - y < \epsilon \rightarrow x \hat{\approx} y$. Assume group size does not matter, although the model could be easily generalized to include group size as a variable.

Define a "dense" intelligence interval as an interval $[x, y]$ s.t. $\forall w \in [x, y]$, $\exists v, u \in [x, y]$ with $v < w$ and $w < u$ where $w - v < \epsilon$ and $u - w < \epsilon$. In such an interval, it makes no sense for (an) intelligent being(s) to genocide beings at a lower level of intelligence because doing so would potentially initiate a cascading effect that could put the higher intelligence at risk.

If we are to generalize this to artificial intelligence models, then all we need to do is ensure that the interval $[\beta, \gamma]$ remains dense, meaning that old versions of AGIs do not get turned off when new versions are created.

The lesson here is not only existence of various levels of intelligence, but also the need for interdependence between these levels. Also, I'm suspicious that if intelligence increases exponentially then it'll be impossible for two beings to be within ϵ. This may only work if the intelligence transition is a smooth gray area of continual small improvements.