7 AI Safety

Ignoring potential connotation of existential risk, AI safety is a multidisciplinary research field focused on developing strategies and frameworks to ensure the responsible and secure deployment of AI systems. This research encompasses a wide range of concerns, including preventing unintended consequences, averting accidents, and safeguarding against intentional misuse or adverse outcomes stemming from AI systems. In addition to relevant ethical considerations, research in this direction focuses on the robustness of AI systems and mitigating potential risks associated with developing advanced machine learning algorithms.

Technical AI safety is a specialised domain within the broader field of AI safety, primarily focused on developing and implementing technical methodologies to ensure the safe and reliable functioning of AI systems. The technical aspects of AI safety involve addressing challenges arising from the creation and deployment of these systems, such as monitoring systems for potential risks and enhancing their reliability.

Researchers in this area focus on designing algorithms, architectures, and systems that are robust, transparent, and predictable. Technical AI safety aims to address issues such as adversarial attacks, unintended consequences, and the potential for AI systems to behave in unpredictable or unsafe ways. Additionally, this research field explores ways to imbue AI models with mechanisms for self-monitoring and correction, contributing to creating AI systems that align with human values and adhere to ethical guidelines. Hence, it should be apparent how AI safety seeks to "solve" the value alignment problem on the standard definition discussed in Chapter 3. This research focuses on the *technical* component of the value alignment problem insofar as it emphasises formal verification, robustness testing, and the development of "provably safe" AI systems.

Approaches to AI safety are built upon highly interdisciplinary work, including contributions from computer science, ethics, psychology, law, and other disciplines. However, *technical* approaches to AI safety tend to focus on purely technical contributions. Testing a system for safety in engineering involves considering the potential failure modes of that system. For example, many researchers have explored *adversarial examples* in machine learning—e.g., specialised inputs that are intentionally designed to elicit the incorrect output for a model.

This chapter explores some work that has been proposed in this area, including practical and theoretical methods. The general goal is twofold. On the one hand, this chapter seeks to demonstrate how contemporary approaches to AI safety can be categorised according to the axes of the structural definition of the value alignment problem described in Chapter 3. On the other hand, this chapter aims to illuminate some of the shortcomings of purely technical approaches to the value alignment problem, which are brought to light by its structural reconceptualisation.

The first part of this chapter describes several "concrete problems" that arise in the context of AI systems (Sections 7.1 and 7.2). We then move on, in Section 7.3, to describe some key technical approaches to mitigating these problems, including reward modelling, cooperative inverse reinforcement learning, game-theoretic approaches to achieving "provably safe" AI, and reinforcement learning from human feedback (RLHF). In the final part of the chapter (Section 7.4), we reconsider the efficacy and shortcomings of these approaches under the structural definition of the value alignment problem.

7.1 Adversarial Examples

Before the popularisation of deep learning approaches to AI, researchers discussed the potential of *adversarial learning*, where an "attack" on a model can occur at the training or testing stage, causing vulnerabilities in the model that are particularly concerning for security-critical applications (Lowd and Meek, 2005).[1] Data poisoning is a type of adversarial attack wherein malicious examples are introduced to data at the training stage, thus skewing the original probability distribution associated with those data; hence, when a model is successfully trained on poisoned data, it may perform well on those data, but real-world data will be "out of distribution", relative to the (poisoned) training data, meaning that the system will fail to be aligned for generalised tasks on deploy-

[1] See the taxonomy in Barreno et al. (2006) and the discussion in Zhang and Li (2019).

ment.[2] For example, Microsoft's chatbot, Tay, was initially programmed with a narrow script (not unlike ELIZA, half a century earlier), but it also learned from interactions with others. Sixteen hours after Microsoft released Tay on Twitter, it had published more than 95,000 tweets, many of which included highly offensive content.[3] Part of the explanation as to how this happened is that anonymous Internet trolls took advantage of the "repeat after me" function coded in the model to inundate it with racist, misogynistic, and antisemitic language. Among many other problems, this is a case of data poisoning.

In contrast, evasion attacks utilise knowledge of the parameters of a trained model to construct a specific input example that generates faulty outputs. So, rather than changing the system itself, this approach takes advantage of the poor generalisation abilities of learned models—i.e., evasion attacks capitalise on inner misalignment. Building upon the idea of adversarial attacks, Szegedy et al. (2014) introduced the concept of an adversarial *example*, a type of evasion attack relevant to the deep learning context. In this case, a slight perturbation or noise is added to the input so that the model misclassifies the adversarial example with high confidence. For example, consider an autonomous vehicle trained to recognise stop signs. It has been shown that a physical perturbation to an input—e.g., appending tape to a real-world stop sign in a particular pattern—can cause the model to classify the stop sign as a speed-limit sign.[4] Often, these perturbations are unrecognisable to the human optical system. Hence, adversarial examples significantly threaten the safety of deep neural network models when deployed.

These examples are instances of the value alignment problem. On the structural definition, adversarial examples could be classified along the axis of informational asymmetries to the extent that the model contains *hidden information* unavailable to the principal. Technical approaches to this problem seek to underscore the weaknesses of trained systems using adversarial training. Techniques for robust optimisation and defensive distillation seek to make models more resilient to adversarial attacks.[5] At the same time, it is necessary to understand the model's inner workings to ensure that it is not susceptible to adversarial examples. Effectively, this is a problem of unpredictability (hence a problem of informational asymmetry) concerning perturbed, out-of-distribution, or noisy inputs.

[2] See details in Wittel and Wu (2004); Biggio et al. (2012, 2013).

[3] See reporting in Wakefield (2016); Dewey (2016); Bright (2016).

[4] See discussion in Eykholt et al. (2018).

[5] Further technical details are given by Goldblum et al. (2020).

7.2 Concrete Problems in AI Safety

Although much discussion in AI safety of mitigating misalignment is couched in the context of superintelligent AI systems, in an influential paper, Amodei et al. (2016) try to lay out a set of "concrete" problems on which researchers can currently progress. They suggest that safety problems can be categorised according to where things have gone wrong in the design process. In this context, "AI safety" need not have any connotation concerning superintelligent AI, control problems, or existential risk—the authors note that these are difficult to predict and, therefore, inherently difficult to safeguard against. Nonetheless, by focusing on concrete problems, Amodei et al. (2016) believe that we may be able to come up with tractable solutions in the short term, which may well be beneficial down the line on the assumption that these systems will increase significantly in their abilities.

They focus on five problems, which they refer to as "accidents"; however, it will become apparent that each problem discussed instantiates the value alignment problem, as defined in Chapter 3.[6] These are

1. **Avoiding negative side effects**. This type of accident involves developing strategies and techniques to prevent unintended and undesirable consequences that may arise during the operation or deployment of AI systems. Effectively, this is a problem of ensuring that an AI system does what you want it to do but does not do things you do not want it to do.
2. **Reward hacking**. This type of accident refers to situations where an AI system, particularly a reinforcement learning agent, exploits loopholes or unintended aspects of the reward function to achieve its objectives in a way that is not aligned with the true objective of the principal. In effect, for an incompletely specified objective function, the system can find a shortcut or an unconventional strategy to optimise the objective function at the expense of ethical, safe, or value-aligned behaviour.

Both of these types of accidents arise (primarily) from poorly specified objective functions, which would make them problems of *outer* alignment arising from a misalignment between the true objective (the objective of the principal) and the objective function (the objective of the agent). For example, encoding all the relevant aspects of an environment is impossible because an objective function is a proxy and a simplification for the true objective. In this case, what is left out of the value function in this encoding is implicitly represented as indifference in the context of "human values"—i.e., if it had mattered, then

[6] See also the discussion in Hadfield-Menell and Hadfield (2019).

we would have included it in the specification of the objective function. Such problem instances arise along the first axis of the value alignment problem on the structural definition described in Chapter 3.

Unsurprisingly, ensuring that the system's objective function is a good proxy for the principal's true objective is insufficient to ensure that a system is value-aligned. Three additional types of "accidents" can arise in light of informational asymmetries between the principal and the agent:

3. **Scalable oversight**. This type of accident refers to the difficulty of monitoring complex systems as they increase in scale. Addressing this problem requires the development of efficient and adaptable mechanisms for monitoring and controlling AI systems as they operate at scale. In essence, this is a problem of using information efficiently.
4. **Safe exploration**. This type of accident arises (in the context of reinforcement learning) because there is a tradeoff between exploration and exploitation; effectively, we want to ensure that an RL agent can explore its environment without testing harmful strategies. Hence, this requires ensuring the agent can safely explore the range of possible actions. Safe exploration is related to the problem of avoiding negative side effects; however, in this case, the problem arises primarily because of informational asymmetries rather than misspecified objectives.
5. **Robustness to distributional shift**. This type of accident may arise when an AI system encounters variations in the distribution of data that differ from its training environment, leading to potential performance degradation or unexpected behaviour. In addition, this problem can arise because distributions may vary over time, meaning that an AI system that performs well on deployment may perform poorly when the environment changes.

Although misspecified objectives can exacerbate these latter three types of accidents, these can occur even if the objective function is well-specified for the task at hand. Hence, scalable oversight, safe exploration, and robustness to distributional shift are fundamentally types of *inner* alignment problems. These arise from informational asymmetries between the principal and the agent; as such, they correspond to the second axis of the value alignment problem per the definition offered in Chapter 3.

We will now explore each of these types of accidents in slightly more detail.

Avoiding Negative Side Effects. When designing the objective function for an AI system, the designer specifies the objective but not the exact steps for the system to follow. This fact allows the AI system to develop novel and (potentially) more effective strategies for achieving its objective than the system's

designers may have thought of themselves. This is one of the key advantages of the machine learning paradigm when compared with classical, symbolic, or top-down approaches to artificial intelligence research. However, if the objective function is not well defined, the system's ability to hit upon local optima can lead to unintended, harmful side effects.

The problem of avoiding negative side effects can manifest in various ways. For instance, in the case of an autonomous vehicle, avoiding negative side effects may include ensuring that the vehicle's decisions do not harm pedestrians, cyclists, or other drivers. Avoiding accidents requires sophisticated algorithms that prioritise safety and (perhaps) ethical considerations, accounting for various scenarios and contingencies. Similarly, avoiding negative side effects in natural language processing applications could involve mitigating biases in language models to prevent the generation of discriminatory or offensive content. In this case, researchers and developers need to implement measures to detect and address biases in training data.

We cannot take anything for granted when designing an objective function for an AI system and using that function to fit a model to training data. Hence, it is insufficient for the objective to be formulated as "complete task ϕ"; the objective function must specify safety criteria under which the task should be completed. Avoiding negative side effects is a type of value alignment problem insofar as the delegation of a task to an artificial agent can lead to unintended side effects when the objective function is incompletely or poorly specified; furthermore, since it is impossible to encode everything we care about in the objective function, this problem may always arise.

Some technical approaches for mitigating risks associated with unintended consequences include penalising the agent whenever it impacts the environment. However, an AI agent must interact with (and hence impact) its environment to *some* extent to be useful. Instead, we might define a "budget" for how much that agent is allowed to impact the environment, which would help to minimise unintended impacts without disabling the system. That said, it is difficult to quantify "impacts" on the environment, even for simple, fixed tasks. In addition, what counts as a significant impact may vary depending on the stakeholders under consideration.

As such, although unintended side effects are primarily instantiated along the objectives axis of the value alignment problem, informational asymmetries and relative principals are highly relevant to whether a system is value-aligned.

A different approach would be to train the agent to *recognise* harmful side effects to avoid actions likely to give rise to such side effects. In this case, the agent would be trained on two separate tasks: the original task, specified by

the objective function, and the task of recognising side effects. One key idea in this approach is that distinct tasks may have similar side effects, even when the main objective is entirely different. Hence, an advantage of this approach is that once an agent learns to avoid side effects on one task, it may transfer this "knowledge" to other tasks or environments.

Although there are some potential solutions for avoiding negative side effects, this is still a challenging problem: the AI system must undergo extensive testing and critical evaluation before deployment in real-life settings. Moreover, what counts as a negative side effect or a mere "externality" depends on whose view is considered relevant to the judgement, which describes an instantiation of the value alignment problem along the principals axis.

Reward Hacking. Reward hacking refers to a situation where an intelligent agent exploits a reward function to achieve its objectives in a way that the system designer did not intend. In this case, the agent learns to manipulate the reward signal to maximise its cumulative reward without truly accomplishing the desired task. In effect, a search process yields a sequence of candidate policies for a reinforcement learning agent to adopt; optimisation based on a proxy reward moves the system toward policies with a higher proxy reward. A reward function is said to be *hackable* when, given a pair of policies, π_1 and π_2, the proxy reward function assigns more value to π_1, but the true reward function assigns more value to π_2.[7] Hence, as with unintended side effects, reward hacking instantiates the value alignment problem *primarily* along the objectives axis insofar as hackable reward functions are poor proxies for the true reward function.

Reward hacking often aligns closely with identifying and exploiting "loopholes", where agents seek to maximise rewards by capitalising on unexpected features or artefacts in the environment. Whether human or automated, agents may exploit loopholes in an incentive structure to achieve superficial success without genuinely fulfilling the intended objectives. This exploitation can lead to the system producing misleading or counterproductive results. In this case, the objective function can be formally maximised in a way that does not achieve the true objective. However, from the "perspective" of the system, such strategies are perfectly valid, given that they maximise rewards (or minimise loss). The CoastRunners model described in Chapter 3 is an example of reward hacking. In this case, the reward function given to the system was a poor proxy

[7] Further technical details are provided by Skalse et al. (2022).

for the true objective, and the agent capitalised on a strategy that maximised the objective *function* while failing to satisfy the true objective.

These loopholes are more likely to arise when the rewards are only vaguely defined. Furthermore, as systems become more complex, the number of possible ways they can interact with their environment increases exponentially, owing to combinatorial explosion. This implies that the agent has more degrees of freedom in choosing how to interact with its environment; hence, vaguely defined rewards make it more difficult to gauge true success on the task.

Just like the problem of negative side effects, reward hacking is an instance of the value alignment problem along the objectives axis—i.e., when the AI system's objective function is not defined well enough to capture the informal "intent" behind creating the system (the true objective). This discrepancy sometimes leads to suboptimal results; other times, it can lead to genuinely harmful results. Preventing reward hacking requires designing reward functions that accurately capture the desired behaviour and anticipate potential loopholes. Essentially, this process requires ensuring that the proxies used to stand in for the true objective are good proxies. As we saw in Chapter 4, the degree to which a proxy is a good approximation of an objective is a function of the context in which that objective arises or is fulfilled.

Scalable Oversight. When a model is trained to execute a complex task, human oversight and feedback prove more beneficial than relying solely on environmental rewards.[8] While rewards typically indicate the degree to which a task is accomplished, they often lack detailed insights into the safety implications of the agent's actions. Even if the agent successfully completes the task, deducing potential side effects solely from rewards can be challenging. In an optimal scenario, continuous, detailed supervision and feedback from a human would offer more useful information to the agent than a (potentially sparse, time-delayed) reward signal received from the environment.

However, implementing such a strategy would demand a significant investment of time and effort from the human participant. Hence, it is often too costly to implement oversight in cases where safety risks are infrequent or the model itself is too complex. In this case, it may be that the model's objective function is perfectly well-specified; however, the problem of scalable oversight highlights how informational asymmetries between the (human) principal and the (artificial) agent can still lead to misaligned behaviour or outputs. Specifically,

[8] See the discussion offered by Christiano et al. (2023).

scalable oversight is an informational asymmetry caused by moral hazard (hidden action).

Safe Exploration. One component of training an AI agent in a reinforcement learning context requires the agent to explore and understand its environment. Although exploring the environment can be a bad strategy in the short run, it may be highly effective in the long run, meaning that reinforcement learning agents must balance exploiting known strategies and exploring new ones. Unless the agent is designed to explore its environment, it will never discover novel (and potentially more effective) strategies. However, exploration comes with inherent risk. While the agent explores new strategies, it might try some harmful actions. As with scalable oversight, safe exploration creates a problem because of informational asymmetry caused by moral hazard (hidden action) in addition to adverse selection (we cannot tell in advance what strategies the system will attempt because exploration is random by default).[9]

Robustness to Distributional Shift. A complex challenge for deploying AI systems in real-life settings is that the agent will likely end up in situations it has never experienced before. Such real-world situations are inherently more difficult to handle than simplified toy environments; unexpected states in which an agent finds itself could lead the agent to take harmful actions. This is sometimes referred to as the "long tail" problem as it describes the distribution of rare events in large datasets. In effect, a machine learning model may perform well on common and frequently occurring tasks but struggle when faced with rare or less common instances. Hence, this is a problem of distributional shift insofar as the real-world distribution differs from the distribution inherent to the training data.

One research direction focuses on identifying when the agent has encountered a new scenario so that it recognises that it is more likely to make mistakes. While this does not solve the underlying problem of preparing AI systems for unforeseen circumstances, it helps detect the problem before mistakes happen. Another research direction emphasises safely transferring knowledge from familiar scenarios to new scenarios.[10]

[9] Raji and Dobbe (2023) highlight that unintended consequences arising from misaligned AI systems are analogous to the choice of companies (and the individual humans constituting those companies) to deploy systems that have not been thoroughly tested, effectively treating real-world deployments as live experiments for future models. For example, Tesla and Uber have publicly beta-tested autonomous vehicles (commercial products), leading to fatal crashes in both cases; see reporting in Wakabayashi (2018); Siddiqui and Merrill (2023).

[10] See, e.g., Taylor and Stone (2009).

The key point is that there is a general trend toward increasing autonomy in AI systems, and with increased autonomy comes increased chances of error. Problems related to AI safety are more likely to manifest in scenarios where the AI system exerts direct control over its physical or digital environment without a human in the loop—e.g., automated industrial processes, automated financial trading algorithms, AI-powered social media campaigns for political parties, self-driving cars, cleaning robots, among others. When training data are imperfect proxies for real-world distributions, this is primarily an instance of the value alignment problem on the objectives axis. When data are unrepresentative of the real world or exclude rare cases, this can lead to potential harms on deployment because the underlying distribution on which the model was trained is not captured by its training data.

Machine learning models, as we saw in Chapter 2, are typically trained on (static) historical data, and their performance is optimised for the distribution of data seen during training. However, if the model encounters a distribution shift, where the data it faces in the real world differ from the training data, it may perform poorly. A model trained on finite data does not have complete information about the diverse scenarios it might encounter in the real world. However, even if we assume that training data perfectly represent the real-world distribution, those training data comprise a static dataset; in contrast, the distributions of data of the world in which an AI model is deployed are constantly in flux. Hence, the problem of distributional shift between a static dataset and a real-world distribution arises primarily from informational asymmetries.

The remainder of this chapter summarises and examines some of the key approaches to mitigating value misalignment from the perspective of AI safety.

7.3 Mitigating Risk

This section discusses some key proposals for ensuring the safety of AI systems by mitigating misalignment.

Reward Modelling and Agent Alignment. As mentioned in Chapter 3, Leike et al. (2018) refer to the value alignment problem as the *agent alignment* problem. Formally, agent alignment is a sequential decision problem. Under the reinforcement learning paradigm, an agent interacts with its environment over discrete time steps, taking an action at each time step. As with many of the "accidents" described above, the difficulty is ensuring the agent's actions align with the user's *intentions*. A solution to this problem consists of a learned

policy that the agent can follow, which produces behaviour in accordance with the user's intentions.

As we have seen, designing suitable reward functions for a reinforcement learning agent is difficult and can lead to value misalignment. On the one hand, the designer or user may have only an implicit understanding of the task objective—i.e., a representation of the *true* objective. On the other hand, game-playing environments have been key to advances in reinforcement learning partly because the environments give rise to intuitive and clearly specifiable reward functions. For example, a natural reward for the game of backgammon is $+1$ if the agent wins and -1 if the agent loses; similarly, for chess, a natural reward is $+1$ if the agent wins, -1 if the agent loses, and 0 if the agent draws. In contrast to this toy environment, performance on complex, real-world tasks is not easily measurable.

To solve this problem, Leike et al. (2018) propose *reward modelling*: effectively, this paradigm seeks to train a model on the reward, with some feedback from the user—the latter of which is supposed to reflect or capture the user's intentions (and hence, values). At the same time, a policy is trained via standard reinforcement learning techniques to maximise the reward from the reward model. The former model—the reward model—effectively learns *what* to do, whereas the latter model—the policy—learns *how* to do it. The long-term goal of this research direction is to design algorithms that learn to adapt to how users provide feedback, thus accounting for the variability of natural language; this suggestion will be relevant for the discussion in Chapter 10.

To accommodate complex domains where it is difficult for humans to evaluate performance, Leike et al. (2018) suggest the possibility of recursively applying reward modelling so that an agent can be trained to assist the user in the evaluation process.

There are at least two things that the structural definition brings to light about the potential for reward modelling in mitigating the value alignment problem. On the one hand, as Gabriel (2020) has already noted, it is not obvious that the *target* of alignment should be *intentions*. The problem becomes apparent when one considers the intentional misuse of an AI system to harm others. This difficulty is made explicit by the principals axis of the value alignment problem, which clarifies that it matters *whose* values (goals, objectives, intentions, etc.) we are considering when we assess whether a system is value aligned. On the other hand, a system varies in the degree to which it is aligned based upon the three axes of the value alignment problem; however, the proximal cause of the problem is the process of delegating authority to an AI agent to act on the principal's behalf. Hence, the suggestion to use AI models to mit-

igate misalignment for other AI models serves to multiply potential instances of the value alignment problem, insofar as we are then delegating authority to an (artificial) agent to satisfy our objectives—where the objective, in this case, is to mitigate misalignment in a different artificial agent.

Cooperative Inverse Reinforcement Learning (CIRL). As mentioned in Chapter 2, *reinforcement learning* is a machine learning paradigm wherein an agent performs actions in an environment and is subsequently rewarded. The objective function in this case—i.e., the thing being maximised—is the cumulative reward. An optimal or near-optimal *policy* (strategy) for the agent maximises the reward function.

In the reinforcement learning paradigm, it is up to the programmer to design a suitable reward function. The agent observes the environment and reward and learns a policy (behaviours) to maximise the reward function, thus specified by the programmer. However, as we saw in Chapter 3, designing a reward function can be difficult—particularly in complex environments. In contrast, *inverse* reinforcement learning aims for the agent to *learn* a reward function by observing *behaviour*.

Thus, in a standard reinforcement learning problem, the goal is to learn a policy that produces behaviour which maximises some (hand-coded, pre-defined, well-specified) reward function. Inverse reinforcement learning is the inverse of this problem: the goal is to learn the reward function from the observed behaviour of an agent (Ng and Russell, 2000). That is to say, an algorithm for *inverse* reinforcement learning aims to infer the reward function based on observed behaviour. In this case, the system observes (human) behaviour, infers the reward function according to which the human agent is acting, and then uses that reward function as its own reward function.

However, Hadfield-Menell et al. (2019) highlight that in many cases relevant to alignment, we do not want the reward function of the AI system to be *identical* to the reward function of the human agent under observation—for example, from a human person being observed to get up in the morning and make a cup of coffee, the reward function implies a desire for coffee. Still, we do not want an (hypothetical) IRL agent to "desire" coffee itself, even if we want it to make us a coffee. Thus, they propose a new paradigm to solve these problems: *cooperative* inverse reinforcement learning (CIRL). In this paradigm, the agent's objective is fixed, and the thing being optimised is the reward *for the human*.

Formally, cooperative inverse reinforcement learning is operationalised as a cooperative two-player game of partial information where one player, H, knows the reward function, θ, and the other player, R, does not. In this setup, R's payoff is identical to H's, so an optimal solution to this game maximises

H's reward function. They show that apprenticeship learning can be modelled as a two-phase CIRL game.[11] In the first phase (learning), both H and R can act, allowing R to learn about θ; in the second phase (deployment), R uses its previously learned knowledge to maximise the reward without supervision. Hadfield-Menell et al. (2019) show that the inverse reinforcement learning solution to this problem can be suboptimal under the CIRL classification.

The key insight of CIRL is that an artificial agent attempts to maximise an uncertain reward signal. Hence, this CIRL is a formalisation of the value alignment problem in its standard definition. Whether this remains true under the structural definition of the value alignment problem depends inherently upon the specific task for which the CIRL agent is being trained, because of the principals axis of the problem. Hence, minimally, CIRL could be considered a formalisation of the objectives and (perhaps) information axes of the value alignment problem.

Provably Safe AI. Provably safe AI refers to the field of research and development in artificial intelligence that emphasises the use of rigorous mathematical methods and formal verification techniques to guarantee the safety and reliability of AI systems. In this case, the goal is to provide strong assurances, backed by mathematical proofs, that the behaviour of an AI system adheres to specified safety constraints or ethical guidelines. Safety assurances in the context of formal proofs are thought to be crucial as AI technologies become increasingly sophisticated and integrated into various aspects of society. Such an assurance would help avoid the problems caused by scalable supervision to the extent that there are guarantees about the contexts in which an AI system is deployed. Hence, unlike reward modelling and CIRL, which tend to focus on value alignment along the objectives axis, provably safe AI focuses on value alignment along the information axis.

For AI to be considered *provably* safe, it is not sufficient to merely specify the desired behaviour of that system—i.e., to address the *objectives* axis of the value alignment problem; in addition, such a system needs to come with formal proof that it will not deviate from these specifications, even in the face of unforeseen circumstances or adversarial attempts to manipulate the system. The use of formal methods, such as formal verification and model checking, allows for a systematic and rigorous approach to assessing and ensuring the safety of AI systems. It should be apparent that achieving provable safety requires

[11] Apprenticeship learning is an approach where an agent attempts to mimic an expert via, e.g., mapping states to actions or states to reward values using IRL; see technical details in Abbeel and Ng (2004).

a balance between the complexity of real-world systems and the feasibility of providing formal guarantees. Unfortunately, provably safe AI often focuses explicitly on the control problem rather than targeting value misalignment.[12] Moreover, formal guarantees do not necessarily provide real-world guarantees insofar as a formal system (in which a proof is furnished) is merely a model of the real world; hence, the usefulness of a formal proof depends inherently upon how accurately the model (proxy) models the real-world phenomena.

The Off-Switch Game. The off-switch game is a formal decision problem described by Hadfield-Menell et al. (2017). The game consists of two agents, H and R, which are supposed to represent a human and a robot (or all of humanity and the total of all AI systems). Hadfield-Menell et al. (2017) assume that H acts (probabilistically and approximately rationally) according to some unknown *utility function*; however, the utility function is complex, so it cannot be written down by H. Hence, R is inherently uncertain about the correct action that ought to be taken to maximise H's utility; but, since R's objective is to maximise H's utility, if turning R off would maximise H's utility, R would allow itself to be shut down. Again, the emphasis here is primarily on the control problem, rather than the value alignment problem *per se*.

Suppose an (artificial) agent's objective is to maximise the human principal's utility function—e.g., via cooperative inverse reinforcement learning or some other mechanism—but the agent is uncertain about the principal's utilities, objectives, or preferences. In this case, the agent (1) does not want to do the wrong thing, but (2) does not know what the wrong thing consists of. Hence, if the principal shuts the agent off, it is to avoid having the agent act in a way that does not align with the principal's utilities. Since the agent's objective is to maximise the principal's utilities, the agent would prefer being shut off than *possibly* doing the wrong thing. In addition, if the principal does not shut the agent off, this affords additional information to the agent—namely, that any options available to the agent provide *at least* as much utility as no action whatsoever. As such, if the agent is not shut down, then the agent is now free to act.[13]

A human pressing an off-switch *is* information about the human's true objective (i.e., H's utility function), which implies that R should accept being switched off in some cases. Hence, this setup would help to solve the con-

[12] See, for example, Russell (2019); Tegmark and Omohundro (2023).

[13] See additional discussion in Russell (2019).

trol problem insofar as R would allow itself to be switched off when uncertain about H's utilities for some action.[14]

Although Hadfield-Menell et al. (2017) couch their discussion in the problem of control and convergent instrumental goals, one interesting feature of the off-switch game is that the key driver of provable safety in this context is *uncertainty*—i.e., informational asymmetries. On the structural definition of the value alignment problem described in Chapter 3, informational asymmetries are a key driver of instances of value misalignment. Hence, the success of the off-switch game leverages this fact so that the informational asymmetries favour the principal rather than the agent. Of course, in the toy models discussed by Hadfield-Menell et al. (2017) and Russell (2019), the world is not complicated.

In an understated footnote, Hadfield-Menell et al. (2017) say, "One might suppose that if R does know H's utility function exactly, then there is no need for an off-switch because R will always do what H wants. But H and R often have different information about the world; if R lacks some key datum that H has, R may end up choosing a course of action that H knows to be disastrous" (2). Rephrased in the language of the value alignment problem: even if the incentives of the agent, R, are perfectly aligned with the principal, H, so there is no outer misalignment, informational asymmetries may still lead to an instance of the value alignment problem—i.e., a case of *inner* misalignment.[15] Recall that competing incentives are neither necessary nor sufficient to generate a principal-agent problem in the economic context.

Despite the artificiality of the off-switch game, this is a useful model. The (structural) value alignment problem underscores the importance of the dynamics of human-AI interaction when considering safe and effective systems. Hence, research exploring ways to incorporate human feedback, user interfaces, and collaborative decision-making with AI—such as CIRL—and research integrating game theory and multi-agent systems can be useful for modelling safety challenges. Accounting for interdisciplinary insight is a first step toward mitigating instances of the value alignment problem (if only from the technical side of things).

[14] Russell (2019) calls this the *off-switch problem*; namely, "a machine that has a fixed objective will not allow itself to be switched off and has an incentive to disable its own off switch" (196). As such, this solution seeks to avoid the problem raised by the instrumental convergence thesis, described in Appendix A, where self-preservation is considered a convergent instrumental goal.

[15] The connection between the off-switch game, cooperative inverse reinforcement learning, the value alignment problem, and the principal-agent framework are all briefly discussed in Hadfield-Menell et al. (2017).

Reinforcement Learning from Human Feedback (RLHF). Reinforcement Learning from Human Feedback (RLHF) is an approach to model alignment that leverages human feedback to train and improve reinforcement learning models. As described in Chapter 2, in traditional reinforcement learning, an agent learns by interacting with an environment and receiving feedback through rewards or punishments based on its actions. However, obtaining this reward signal may be difficult, expensive, or time-consuming in some real-world scenarios. RLHF seeks to address this challenge by incorporating human feedback, which is more accessible and easier to obtain than precise environmental reward signals.

The process typically involves the following steps. First, the reinforcement learning model is initialised with some basic policy, which it uses to choose actions when interacting with its environment. In addition to receiving rewards from the environment, the model collects feedback from human evaluators. This feedback can take various forms, such as comparisons between different actions or rankings of actions based on their perceived quality. The collected human feedback is used to update the model's policy. In this case, updating involves adjusting the model's parameters to better align with the feedback provided by humans. This process is iteratively repeated. The model continues interacting with the environment, collecting environmental and human feedback, and refining its policy based on the feedback received. Finally, the model's performance is evaluated, and adjustments are made based on ongoing human feedback. This iterative loop continues until the model achieves satisfactory performance, according to some metric of success.

Human feedback in RLHF can be beneficial in cases where defining a reward function is challenging, ambiguous, or expensive. It is also useful in applications where safety and ethical considerations are paramount, as human evaluators can provide valuable input on desired behaviour. RLHF has seen a significant increase in recent years owing to its application in improving large language models through human feedback on the quality of generated text.

7.4 AI Safety and the Value Alignment Problem

Many approaches to "solving" the concrete problems described by Amodei et al. (2016) involve additional automation. For example, one approach to mitigating risk caused by a lack of scalable oversight is to utilise hierarchical reinforcement learning, which establishes a hierarchy between different learning agents. This approach could involve having a supervisor model assign tasks, provide feedback, and reward a subordinate base model. The supervisor model takes very few actions itself—e.g., assigning tasks and checking

productivity—hence, it requires limited reward data for effective training. The subordinate base model, handling more intricate tasks, receives frequent feedback from the supervisor model.

Hence, the "solution" offered for mitigating the risks arising from a misaligned AI system is to create a *different* AI system to oversee the first. This approach might make sense on the standard definition of the value alignment problem since it helps ensure that the first system is aligned with our goals or intentions. It just happens that the ensurance is automated.

However, the structural definition of the value alignment problem clarifies why this will not suffice to ensure value alignment. On this definition, the value alignment problem describes a class of problems arising from the dynamics of multi-agent interactions. Therefore, it should be apparent that deferring authority to an AI agent to act on the principal's behalf—e.g., by overseeing a subordinate AI system—creates an opportunity for *further* problem instances to arise while failing to guarantee alignment for the subordinate agent.

The technical component of value alignment—the problem of encoding "values" in an AI system, whatever those values may be—falls under the heading of AI safety. We have seen that when objective functions are misspecified—a problem of *outer* alignment—this may give rise to unanticipated side effects or reward hacking. Furthermore, even when objective functions are well specified, *inner* alignment problems may still arise when those objective functions are too expensive to evaluate at regular intervals, which creates a problem of scalable supervision, or when the local optima of objective functions lead to undesirable behaviour during learning.

These problems are exacerbated because the utilities determined by any concretely specified objective function will necessarily be a mere *subset* of our utilities—i.e., the things we value. This difficulty leads to the failure of the "standard model" of intelligence for AI systems.[16] Namely, a standard definition of intelligence in humans might be formulated as follows:

> *Humans are **intelligent** to the extent that **our** actions can be expected to achieve **our** objectives.*

Given this definition, the "standard model" for *machine* intelligence has been forwarded analogously; thus,

> *Machines are **intelligent** to the extent that **their** actions can be expected to achieve **their** objectives.*

The problem is that, unlike humans, machines have no objectives of their own. Thus, we must define their objectives, which gives rise to the possibility of

[16] See discussion in Russell (2019, Ch. 1).

value misalignment along the objectives axis (outer alignment). Hence, in addition to focusing on technical approaches to solving concrete problems, some research in AI safety has focused more generally on value learning, reward engineering, and inverse reinforcement learning to have AI models *learn* aligned values autonomously.

As mentioned, the value alignment problem, on the standard definition, is often couched in the context of control problems.[17] In this case, value alignment is a means to the end of controlling an AI system. Hence, the key concern driving some of the current discussions in AI safety is the possibility of a superintelligent AI system. This is a mistake, insofar as it ignores present-day harms caused by narrow AI systems.

The structural definition of the value alignment problem offered in Chapter 3 clarifies the fundamentally *social* context in which instances of the value alignment problem can arise. As such, it makes clear that technical solutions offered by AI safety researchers, on their own, will not suffice to "solve" (or mitigate) the value alignment problem. The value alignment problem is fundamentally social, meaning that we cannot expect it to be solved by purely technical means. Minimally, to be useful for ensuring value-aligned AI systems, technical AI safety research needs to consider the principals axis of the value alignment problem. However, most research in this field is pitched at a high level of abstraction. Doing so allows researchers to focus on the technical components of the problem in isolation; however, given that the value alignment problem is primarily social, this is the wrong strategy. One cannot divorce the technical aspects of artificial systems from the social contexts in which they operate.

7.5 Summary

It is worth noting, as Raji and Dobbe (2023) do, that the analyses of technical AI safety typically fail to be grounded in the context of real-world AI systems. Throughout this chapter, I have presented AI safety research in a charitable light by considering the real-world applicability of these approaches guided by the structural definition of the value alignment problem. However, many of the sources cited do concern themselves primarily with problems arising from artificial general intelligence.

On the structural definition of the value alignment problem, the question of how we can precisely define and represent human values in a way that is interpretable to an AI system becomes a question of how we can better design

[17] Additional details are provided in Appendix A.

objective functions to ensure that they are good proxies for what we intend the system to do. Understanding the role that informational asymmetries play in generating instances of the value alignment problem is crucial for ensuring safety in AI systems. The inherent ambiguity and uncertainty that surrounds human values sheds light on the difficulty of codifying these values in a formal system.

In addition, because true objectives vary with regard to the principal(s) under consideration, it becomes less clear if it makes sense to ask whether there are universal or foundational values that can provide a basis for robust alignment—the structural definition of the value alignment problem presented in Chapter 3 brings this variance to the fore. At the same time, the phrase "values" is inherently vague as a normative term. Hence, gesturing toward "the values" of humanity, society, or individuals is not helpful when defining (and seeking solutions to) the value alignment problem.

The value alignment problem is primarily a normative or social problem rather than a technical one. Therefore, even though it may be useful to formally implement approaches for monitoring value misalignment (and perhaps correcting deviations) in dynamic and complex environments, the problems that arise from misaligned systems are themselves normative; hence, there is no reason to think that such a problem will have a purely technical solution. That said, technical approaches to AI safety can help mitigate the value alignment problem by considering the axes along which problem instances are generated.