Artificial Intelligence Review (2024) 57:105
https://doi.org/10.1007/s10462-024-10732-3

On the computational complexity of ethics: moral tractability for minds and machines

Jakob Stenseke[1]

Accepted: 17 February 2024

Abstract
Why should moral philosophers, moral psychologists, and machine ethicists care about computational complexity? Debates on whether artificial intelligence (AI) can or should be used to solve problems in ethical domains have mainly been driven by what AI can or cannot do in terms of human capacities. In this paper, we tackle the problem from the other end by exploring what kind of moral machines are possible based on what computational systems can or cannot do. To do so, we analyze normative ethics through the lens of computational complexity. First, we introduce computational complexity for the uninitiated reader and discuss how the complexity of ethical problems can be framed within Marr's three levels of analysis. We then study a range of ethical problems based on consequentialism, deontology, and virtue ethics, with the aim of elucidating the complexity associated with the problems themselves (e.g., due to combinatorics, uncertainty, strategic dynamics), the computational methods employed (e.g., probability, logic, learning), and the available resources (e.g., time, knowledge, learning). The results indicate that most problems the normative frameworks pose lead to tractability issues in every category analyzed. Our investigation also provides several insights about the computational nature of normative ethics, including the differences between rule- and outcome-based moral strategies, and the implementation-variance with regard to moral resources. We then discuss the consequences complexity results have for the prospect of moral machines in virtue of the trade-off between optimality and efficiency. Finally, we elucidate how computational complexity can be used to inform both philosophical and cognitive-psychological research on human morality by advancing the moral tractability thesis.

Keywords Computational complexity · Machine ethics · Artificial moral agents · Consequentialism · Deontology · Virtue ethics

✉ Jakob Stenseke
jakob.stenseke@fil.lu.se

[1] Department of Philosophy, Lund University, Helgonavagen 3, 221 00 Lund, Sweden

Published online: 31 March 2024

1 Introduction

Computational systems of hardware and software continue to enter and transform a growing number of human domains. As autonomous vehicles, virtual teachers, and carebots augment or even take over traditional human roles of drivers, educators, and caretakers, it becomes hard to ignore the need for systems that align with the norms and moral standards associated by such roles.[1] These concerns have spawned the interdisciplinary field of *machine ethics*, which broadly explores the prospects of implementing ethics into machines (Wallach and Allen 2008; Anderson and Anderson 2011). Lying in the intersection of computer science and moral philosophy, machine ethics encompasses a spectrum of more or less interconnected research aims, including work that addresses the challenges of value alignment (Gabriel 2020), explainability (Gunning et al. 2019), and safety (Amodei et al. 2016) of existing AI methods, the development of systems tackling various ethical dilemmas (Cervantes et al. 2020; Tolmeijer et al. 2020), and theoretical debates on whether and to what extent artificial moral agents are feasible or desirable (Floridi and Sanders 2004; Behdadi and Munthe 2020).[2]

The feasibility debate has, in turn, mainly been driven by what AI systems can or cannot do in terms of human capacities; whether artificial agents could be autonomous or have free will (Hellström 2013), be equipped with human-like rationality (Purves et al. 2015), or capable of conscious experience (Himma 2009). However, by centering on capacities that remain elusive and conceptually opaque from a computational perspective, debates on artificial morality fails to engage with the technical dimensions of AI, and as a result, they become practically otiose for the design and development of ethical machines (Mabaso 2021; Behdadi and Munthe 2020; Stenseke 2022b). Another issue that obscures the feasibility of moral machines is the absence of systemic evaluation tools (Tolmeijer et al. 2020). In machine ethics, there are at present no domain-specific nor general benchmarks that can be used to evaluate the performance of different ethical systems. Consequentially, since evaluations of systems are limited to the experimental conditions of their particular implementation, the scalability of solutions and generalizability of results are severely restricted.

In this paper, we address these issues by exploring what kind of moral machines are possible based on the ethical problems computational systems can or cannot solve effectively. To do so, we analyze normative ethics through the lens of computational complexity theory, which classifies problems in terms of the resources (e.g., time and space) a computer requires to solve them. While previous work have discussed computational limitations for moral machines more informally (Brundage 2014; Stenseke and Balkenius 2022), and provided embryonic complexity analyses of ethical actions (Reynolds 2005) and plans (Lindner et al. 2020), the computational complexity of ethics and its potential relevance for machine ethics remains largely unexplored. For instance, if artificial systems were to operate in ethical domains where time is of the essence (e.g., a self-driving ambulance), it is crucial that such systems can make efficient as well as competent ethical decisions. Furthermore, if human moral cognition is constrained by tractability (Van Rooij 2008), the analysis might also serve moral psychology and normative theory by constraining the space of problems an agent following a certain normative theory can be reasonably expected to solve.

[1] Unless specified, terms such as "AI system", "machine", and "computer" will be used interchangeably to denote computational systems of hardware and software.

[2] See also Coeckelbergh (2020) for an accessible introduction and overview of AI ethics.

In the rest of the paper, concepts and theories from both moral philosophy and computer science are introduced and explained in a way that is friendly for readers with a limited background in one or both areas. It is structured as follows. First, we give an introduction to computational complexity and tractability with the aim of explaining their relevance for the uninitiated reader (Sect. 2). In Sect. 3, we survey previous implementations in machine ethics and discuss various interpretations of the complexity of ethics using Marr's three levels of analysis (Sect. 3.1), which motivates the analysis of problems posed by normative theory (computational level, Sect. 3.1.1) that are solved through a variety of computational methods (algorithmic level, Sect. 3.1.2) by a deterministic Turing Machine (implementation level, Sect. 3.1.3). We then explore the complexity of various ethical problems based on consequentialism (Sect. 4), deontology (Sect. 5), and virtue ethics (Sect. 6). The main aim is to elucidate the complexity associated with the problems themselves (e.g., due to uncertainty, combinatorics, strategic dynamics, and generality), the available resources (e.g., time, cognition, and domain knowledge), and the computational methods employed to tackle the problems (e.g., probability, logic, and learning). The results indicate that most problems the normative theories pose lead to intractability issues (a succinct summary is given in Table 3), and especially if the prescriptive ideal should be optimally satisfied. In particular, based on the intractability (and undecidability) stemming from combinatorics of action plans (Sect. 4.1), probabilistic causal inference (Sect. 4.2), dynamic and partially observable environments (Sect. 4.3), general rules (Sect. 5.1), strategic dynamics (Sect. 5.1.3), logic (Sect. 5.2), semantics (Sect. 5.2.3) and learning (Sect. 6.1), we firmly conclude that perfect moral machines are impossible. Our investigation also provides additional insights regarding the computational nature of the normative theories, including (i) the differences between action- and outcome-based strategies, (ii) the benefits of moral hybrids (Sect. 5.3), and (iii) the extreme implementation-variance with regard to moral resources. In Sect. 7, we discuss the consequences the results have for the prospects of moral machines by focusing on the trade-off between optimality and efficiency, the equivocal role of normative theory, and the intimate relationship between different moral resources. Finally, we demonstrate how computational tractability can be used to inform both philosophical and psychological research on human morality by advancing the Moral Tractability Thesis.

2 The complexity of making a salad

Let us begin with an illustrative example.[3] There is a high chance that you have stumbled upon a salad bar where you can choose ingredients to your own liking.[4] The question is, what ingredients do you pick in order to create the best tasting salad? Let us assume that you can immediately assess the tastiness of each ingredient in isolation and give them a "taste value" (v) on a scale ranging from the most off-putting (-10) to the most delicious ($+10$). With these values, you find that one efficient way of putting together a decent salad is to exclusively pick ingredients (I) with a positive v ($v(I) > 0$), or a v that is higher than a certain threshold (e.g., $v(I) > 5$). Let us name this strategy Ψ. In fact, as a queue is lining up behind the salad bar, you appreciate the speed Ψ allows you to make a salad: you

[3] The reader who is already familiar with computational complexity is advised to skip to Sect. 3.

[4] The example is inspired by the excellent introduction to complexity analysis given in Van Rooij et al. (2019).

only have to visit each ingredient once and check whether they are sufficiently tasty to be included in your mix. Furthermore, you realize that the performance of Ψ grows, in the worst-case, linearly with the number of salad ingredients. This means that, regardless of how many ingredients there could be, Ψ will always be efficient: for any input—in this case, n number of ingredients—the time it takes to make a salad will closely mirror the size of the input (i.e., 1000 ingredients equals 1000 visits to distinct ingredients).

But upon further reflection, you realize that something is odd with Ψ. It asks you to put sun-dried tomatoes on top of pineapple. You imagine how the saltiness of sun-dried tomatoes mixes with the sweet–sourness of pineapple as they traverse the taste buds of your tongue. Your immediate disgust of the image reveals a fatal flaw of Ψ: even if these two ingredients were given some of the highest taste values ($v > 9$), their combination yields a taste value that is terribly off-putting ($v = -10$). You realize that Ψ violates a fundamental principle of gastronomy, namely, that combinations of ingredients yield taste values that do not necessarily correspond with the tastiness of its individual ingredients. We can call this principle the combinatorial principle of gastronomy (CPG).

Luckily, you have a perfect gustatory imagination and can immediately assess the taste value of any given combination of ingredients. How do you find the optimal combination of ingredients in a way that maximizes taste value and adheres to CPG? We can formally describe this as the following computational problem:

OPTIMAL SALAD FOLLOWING CPG

Input: A salad bar as a set $SB = \{I_1, I_2, \ldots, I_n\}$ of n ingredients and a value function v that assigns a taste value to every subset (or salad) $S \subseteq SB$.
Output: A salad $S \subseteq SB$ such that $v(S)$ is maximized over all possible salads in the salad bar ($S \subseteq SB$).

You realize that there is a straight-forward strategy, you call it Φ, that is guaranteed to produce an optimal salad while satisfying CPG: simply imagine the taste value of *each* possible subset $S \subseteq SB$ and pick the salad with the highest $v(S)$. But you have a feeling that there must be a catch with Φ. You do some basic combinatorics: if there was only one ingredient, e.g., $\{cucumber\}$, there would be one possible salad (made entirely of cucumber); two ingredients yield three distinct salads, e.g., $\{cucumber\}$, $\{onion\}$, $\{cucumber, onion\}$; three ingredients make seven; four make fifteen; etc. You determine that the number of possible salads grows *exponentially* with the number of ingredients, so that n ingredients produce $2^n - 1$ possible salads. The salad bar you are currently facing has 30 ingredients, which presents $2^{30} - 1 = 1,073,741,823$ distinct salads. Since your otherwise extraordinary gustatory system can only assess the taste of one salad per second, Φ asks you to imagine salads for roughly 34 years, that is, if you were to optimally satisfy the combinatorial principle of gastronomy. Unfortunately, you have already wasted more than enough time, and the people in the queue behind you are very upset.

The example serves to draw four important lessons about computational complexity:

(1) The first is that many decision problems that we encounter in everyday life can be formulated in similar ways, from planning an itinerary, packing a bag for a trip, or inviting a selection of friends to a birthday party in your small apartment. And as we will see throughout this paper, ethical problems are no exception. You might wonder why it matters so much to find the optimal salad; at worst, you end up with a poor-tasting salad, which is far from a disastrous consequence. But would you be so quick to disregard optimal results if the problem was a matter of life and death? And even if

you do not care much about the combinatorial principle of gastronomy, there might be moral principles that are fundamental to your ethical life.

(2) The second lesson is that the complexity of a problem can be expressed in terms of the resources an agent or algorithm requires to solve it. For computational systems, the two most interesting resources are *time* and *space*. The latter conventionally denotes the size of computer memory (e.g., bits), whereas the former refers to the number of machine operations (or synonymously used terms such as "computations", "calculations", "steps", or "state transitions"). Why measure time in terms of machine operations and not in seconds or minutes? The reason is that, while the real-time speed of computers solving a problem by running some algorithm A can vary greatly, the amount of machine operations they need to execute A remain unchanged. A 21st century computer and one from the 1960s both have to consider $2^n - 1$ salads if they were to produce the best tasting salad following Φ, even if the modern computer could potentially do so a million times faster. Importantly, this forms the basis of the Invariance Thesis,[5] which allows us to analyze and compare the worst-case complexity that is inherent to computational problems independent of specific machines.

(3) This leads to the third lesson, which is the simple observation that some problems are more complex than others. If a problem is undecidable, it means that it can be proven that no algorithm can be constructed to solve the problem.[6] Among the decidable problems, the most important distinction is between problems that are *tractable* and *intractable*. Crudely put, a problem is tractable if it can be solved using a 'realistic' amount of resources. For most computational theorists, however, tractable is synonymous with "computable in polynomial time". This means that the runtime (number of machine operations) of an algorithm is upperbounded by a polynomial expression in its input, i.e., of the type n^c (where c is some positive constant). This includes functions that show logarithmic ($\log n$), linear (n), quadratic (n^2), or cubic (n^3) growth in time as the input n increases. The class of decision problems that can be solved in polynomial time by a deterministic Turing machine is called P, capturing the notion of decision problems with "effective" decision procedures (Cobham 1965). Conversely, problems that cannot be solved in polynomial time are called intractable as their runtime grows exponentially (c^n), by a factorial ($n!$), or super-exponentially (n^n). This notion of tractability is illustrated in the difference between decision procedure Ψ and Φ. For Ψ, salad-making time will never grow more than linearly in relation to the number of ingredients. Using Big O notation, which expresses an asymptomtotic upperbound[7] of a function (in this case, a mapping between input size n and time), the time complexity of Ψ is $O(n)$. By contrast, performing an exhaustive search over all possible salads of

[5] More formally, the thesis states that given two machines M_1 and M_2, and a given computational problem Θ, the complexity of Θ executed by M_1 and M_2 will differ at most by a polynomial amount. That is, if M_1 is able to compute Θ in time t, M_2 can compute Θ in t^c, where c is a constant. The thesis is widely accepted among computer scientists provided that M_1 and M_2 are any type of Turing machine or any other reasonable model of computation (e.g., cellular automata, neural networks) and the input is reasonably encoded (e.g., it does not involve irrelevant information). See Garey and Johnson (1979).

[6] The halting problem is an example of an undecidable problem: in 1936, Alan Turing proved that there is no algorithm that can determine whether an arbitrary program eventually halts. Decidability will be further addressed in Sect. 5.2.1.

[7] "asymptotic" means that we can ignore lower order polynomials and constants when we describe a function. For instance, the function $f(n) = 4n + n^3$ is written as $O(n^3)$, since $4n$ becomes insignificant compared to n^3 as n increases.

the salad bar leads Φ to the exponential $O(2^n)$. Even if your gustatory imagination could utilize the speed of the parallelized neural computation of your brain, which allowed you to imagine one billion salads per second, a salad bar of 50 ingredients would still take you 13 days to master, and a bar of 60 takes you roughly 37 billion years. Of course, no sane person would spend that much time imagining the taste of different salads. But the problem with problems remains: if we want to solve them effectively, we might need to give up our requirement of optimality. Instead of "best imaginable", we need compromises that are "good enough" given the available resources. As such, intractable problems can present an uncomfortable trade-off between ideal and feasible. And it is precisely how this uncomfortable trade-off affects ethical decisions for computational agents that will be the topic of this paper.

Part of the reason why there is no effective way to make an optimal salad following CPG is captured in the widely believed conjecture $P \neq NP$. It states that decision problems that have solutions which can be *checked* (or verified) effectively cannot necessarily be solved effectively. To be more precise, it states that the complexity class P does not equal NP: the class of decision problems solvable in polynomial time by a *non-deterministic* TM, or equivalently, decision problems where solutions can be verified in polynomial time. Φ exemplifies such a case. Even if you can check the taste of any combination of salad ingredients quickly (polynomial time), there is no deterministic procedure that allows you to find the optimal; you still have to check the entire space of combinations to ensure that you have the optimal subset. In fact, finding the optimal salad following CPG is NP-hard, which means that it is *at least* as hard as the hardest problem in NP. More formally, a problem X is NP-hard when *every* problem in NP can be *reduced* in polynomial time to X. This means that if we assume that a solution for X takes one unit of time, the solution can be used to solve every problem in NP in polynomial time. A closely related property is the notion of completeness. An NP-complete problem is both NP-hard *and* belongs to NP. Note that, while P and NP are classes of decision problems—which can be framed as a yes/no-type question—NP-hard problems are not restricted to decision problems as such; they are simply at least as hard as the hardest decision versions of the same problem. For instance, while decision variants of NP-hard problems might be NP-complete—e.g., the Boolean satisfiability problem (SAT) or subset sum problem (SSP)—other variants of the same problem, e.g., framed as optimization or search problems, are not (they are not decision problems). Again, this is illustrated in our example: salad making following CPG is NP-hard since it is an optimization version of the subset sum problem (SSP), which is NP-complete.

Furthermore, note also that, while NP-hardness only denotes a general lower bound, it does not say anything about an upper bound, which might be more informative for understanding exactly *how* hard a problem is.[8] In computational complexity theory, classes of computational problems are instead defined by the upper bound (or constraints) on the amount of resources they require in the worst-case (formalized using Big O notation). In turn, this allows us to describe general hierarchies of how complex problems are. For instance, problems solvable in polynomial time by a deterministic TM are also solvable by a non-deterministic TM, which implies that P is a subset ($\subseteq$) of NP. Similarly, it is widely believed that $P \subseteq NP \subseteq PSPACE \subseteq EXPTIME \subseteq EXP-$

[8] For instance, the halting problem is NP-hard but undecidable (it is not decidable in a finite amount of operations); the true quantified Boolean formula language (QBF) is NP-hard but decidable in polynomial space (PSPACE-complete) (Garey and Johnson 1979).

SPACE (see Appendix 1 for a summary of the complexity classes used in this article). The worst-case analysis can be motivated by the fact that an algorithm needs to consider all possible inputs of a problem, which includes the worst-case input. But in the analysis of algorithms, there are several other essential tools to study complexity. For instance, if the lower and upper bound coincide, we have a *tight bound*. Again, there is such a tight bound on the time complexity of making an optimal salad following Φ: we have to imagine the taste of at least *and* at most $2^n - 1$ salads to ensure optimality. Alternatively, we could imagine that salad bars were arranged in ways that allowed for exploitation, e.g., sorted in rows of pre-made combinations. If so, we could measure the time complexity of an algorithm in terms of how many operations it required to make a salad over a number of different salad bars (inputs), and see how it performed in the best-case, average-case, and worst-case.[9] In short, computational complexity provides a smorgasbord of analytical tools to understand the difficulty of problems and their algorithmic solutions.

(4) The fourth lesson, and a corollary of the third, is that the way an agent solves a problem ultimately depends on its *resources*, broadly construed. Besides time and memory-size, these resources include heuristics (efficient strategies), cognition (capacities for perceiving and acting in the world),[10] knowledge, and learning. In reality, you might mix aspects of Ψ and Φ. You might select a few key ingredients as a basis that you already know yields a reasonably tasty salad, and imagine whether this basis could benefit from further additions. Drawing from your vast experience of cooking—combining previous trial-and-error, general rules of thumb, and educated guess-work—you are able to quickly put together an almost perfect salad while still adhering to the CPG (albeit not optimally). In fact, your stomach might already know what kind of salad it craves before you even see what the bar offers; you only have to pick up the ingredients. In such cases, a low input-size (e.g., 10 ingredients) could be a curse rather than a blessing, since you find that a critical ingredient is missing. The main point is that, although problems might be intractable regarding some specific resource (e.g., time), or due to the choice of strategy (e.g., Φ), it is hard to tell in a given situation whether an effective solution could be obtained via other means (e.g., using some different strategy or given more of a certain resource). Importantly, this leads to a distinction between the *problem itself* (e.g., put together a tasty salad), and *how* the problem is solved (e.g., follow Φ). And while the distinction between problem and solution might be relatively clear in computational contexts (e.g., between problem and algorithmic solution), we will dedicate much effort in this paper to elucidate their difference in moral contexts.

[9] However, note that such performance measures would not work for finding the optimal salad following Φ, since it does not matter in which way the ingredients are arranged.

[10] Throughout this paper, the term "cognition" will be broadly used to denote all sorts of information-processing that enables capacities such as perception, action, reasoning, and learning. As such, it differs from cognitivism in meta-ethics (the view that moral language can express propositions that can be true or false) and conceptions of cognition that emphasize prefrontal activity (e.g., thinking, memory, judgement) in contrast to 'back of the brain' sensory processing (Block 2019).

3 Computational complexity of ethics

What is the computational complexity of ethics? First, we should note that "ethics" is a multifaceted and equivocal concept that permeates many levels of analysis across different disciplines. Throughout the ages, moral philosophers have in more or less systematic ways tried to resolve questions regarding what is morally "good" and "bad". In modern times, Anglophone analytical ethics is conventionally divided into (i) *applied ethics* (determining what is "good" and "bad" in particular instances), (ii) *normative ethics* (advancing standards and principles of what is "good" and "bad"), and (iii) *meta-ethics* (determining the meaning and nature of morality). But the landscape of ethics stretches far beyond these divisions. From a biological point of view, it includes the evolutionary foundations of cooperation (as extensively studied in game theory (Axelrod and Hamilton 1981; Nowak 2006)), where morality can be viewed as an adaptive solution to the problem of competition among self-interested organisms,[11] from individual cells (Hummert et al. 2014) to human beings (Leben 2018). The landscape gets further complicated if we also consider the social, psychological, and cognitive dimensions, e.g., how ethical behavior is intertwined with the empathy, emotions, and reasoning of embodied agents, and carried out by highly distributed and parallel cognitive systems (Newen et al. 2018; FeldmanHall and Mobbs 2015). Far from being 'fixed', moral behavior is something which is developed and actively refined through experience.[12] Beyond individuals, ethics is also manifested at the level of societies and culture; maintained and transformed through practices and institutions, mediated through the language of ideology and religion, and with justifications that ranges from divine authority (e.g., word of God), maintaining political order (Hobbes 1651), to the promotion of liberty (Mill 1859) or justice (Rawls 1971).

Hence, to delimit our investigation, we will focus on the complexity of ethical problems as they have been framed within the field of *machine ethics*. The majority of technical work in machine ethics has been focusing on normative ethics, or more specifically, how certain tenets or aspects of a normative theory can be implemented so that an artificial agent acts in accordance with the theory (Cervantes et al. 2020; Tolmeijer et al. 2020). As such, it can be viewed as a form of *applied* normative ethics, since it primarily centers on the practical implementation of a certain theory as opposed to discussions about what theory that should be. In their exhaustive survey of implementations, Tolmeijer et al. (2020) has suggested that approaches to moral machines can be characterized along three broad dimensions: ethical theory, implementation, and technology. The first dimension denotes the ethical theory used, which includes normative frameworks such as deontology (Anderson and Anderson 2008; Malle et al. 2017a; Shim et al. 2017), consequentialism (Abel et al. 2016; Armstrong 2015; Cloos 2005), virtue ethics (Stenseke 2021; Govindarajulu et al. 2019; Howard and Muntean 2017), and hybrids (Dehghani et al. 2008b; Thornton et al. 2016). The second dimension, following a division proposed by Allen et al. (2005), considers *how* ethics is implemented in the system, e.g., whether it is through a 'bottom-up' learning process, carried out via 'top-down' principles, or in a combination of both top-down and bottom-up processing. The technical dimension, in turn, considers the computational techniques used

[11] Or alternatively put, the function of morality is to alleviate the failures of rationality (Ullmann-Margalit 2015).

[12] From the pioneering work of Kohlberg and Hersh (1977), through refinements by Rest et al. (1999), moral psychology has grown into a mature paradigm that investigates the link between morality and cognitive development.

to realize the implementation, which include methods from various AI paradigms such as logical reasoning (e.g., inductive, deductive, and abductive logic), machine learning (e.g., neural networks, reinforcement learning, evolutionary computing), and probability (e.g., Bayesian and Markov models).

3.1 The complexity of ethics following Marr's three-level analysis

Based on these considerations, how can we frame the computational complexity of ethics for machines? Recalling the final lesson in the previous section, we first need to find some way of distinguishing *problems* as such from *how* these problems are solved. This distinction is reflected in the influential scheme proposed by Marr (1981). Marr suggested that the information processing of a cognitive system can be explained on three distinct yet complementary levels of analysis: (i) *Computational level*, (ii) *Algorithmic level*, and (iii) *Implementation level*. The computational level describes the problem itself (e.g., an input–output mapping), the algorithmic level specifies the algorithmic process (e.g., strategy or heuristic) that is performed to tackle the problem, and the implementation level specifies how the algorithmic process is realized by the physical hardware of the system (e.g., neurons or circuits). These levels can be illustrated using the salad bar example: (i) the computational level specifies the number of ingredients, value functions (e.g., tastiness of individual ingredients or combinations of ingredients), and desired output (maximally tasty salad); (ii) the algorithmic level describes the problem-solving process (such as Φ); (iii) the implementation level describes the way a brain or machine implements the problem-solving process physically. Each of these levels of a system can be analyzed independently. For instance, since one and the same computational problem can be solved by a range of different algorithmic procedures, we can describe a cognitive system at the computational level independently of the algorithmic level, and thus have a computational-level theory of the computational system. Likewise, since an algorithm can be physically realized in a range of different systems—e.g., silicon or carbon—we might have an algorithmic-level theory of a cognitive system that does not require us to explain how it is physically implemented. Nevertheless, Marr argued that it is easier to elucidate the workings of a cognitive system through the top-down lens, i.e., by starting from the problem it solves as opposed to the precise mechanisms it uses to solve it (Marr 1977, 1981).[13] The reason is that higher-level explanations make commitments about the lower-levels, which in turn forms a hierarchy of underdetermination. For instance, if we conjecture that a cognitive system solves problem P at the computational level, we might be uncertain or agnostic with regards to the specific algorithm it employs to compute P. However, if our conjecture should carry any explanatory value beyond the computational level, we must commit to the idea that at least *some* algorithm can compute P. If it can be proven that no such algorithm exists, then our problem is undecidable. Similarly, if we believe that a system solves P using algorithm A, we commit to the idea that *some* physical system can realize A.

[13] To clarify, this particular notion of "top-down", from computation (top), to algorithm, to implementation (bottom), is distinct from the common use in cognitive psychology, where "bottom-up" processing starts from the sensory input, and "top-down" processes centers around interpreting the incoming information based on knowledge, experience, and expectations.

3.1.1 Level 1: computational problem

How do we fit ethical problems into Marr's scheme? More precisely, what is the algorithmic level and what is the computational level of ethical problems posed by normative theory (NT)? First, we note that normative ethics blurs the line between Marr's first two levels. In particular, its prescriptive component is intimately linked with its action-guidance, i.e., by answering what *is* good (e.g., adherence to moral duties), it tells you how to *do* good (e.g., only perform actions that adhere to moral duties).[14] In turn, this opens up a range of possible interpretations, and we will address three:

(1) *NT as algorithmic-level solution to generalized morality* In the most general sense, if the computational-level problem is phrased as "do what is moral", we might interpret an NT as an algorithmic-level solution to the computational problem "how to be moral in general". This interpretation would capture the generality ambition of NTs in human contexts (or at least in philosophical discourse on NT); that an NT should provide general answers or standards regarding right and wrong that are applicable to a range of particular instances. An agent that is committed to NT_1 would only be moral insofar as it adheres to NT_1 in its general behavior.[15] Nevertheless, it is hard to see how one could feasibly frame such a broad interpretation in the formalism required by a computational complexity analysis; it would entail some form of general-purpose algorithm—e.g., in terms of a value, principle, or maxim—that provides solutions to all possible moral dilemmas.[16]

(2) *NT as algorithmic-level solutions to specific moral problems* A similar but more narrow interpretation is that NTs provide algorithmic-level strategies that can be used to solve *specific* moral problems. This interpretation seems to, at least prima facie, capture everyday usage of the term "moral dilemma", i.e., a decision problem that arises as a conflict between two or more NTs (where "NTs" might as well be replaced with values, duties, virtues, or norms). We could, for instance, specify the computational-level problem as the trolley problem in order to draw attention to the conflict between action- and outcome-based NTs: is it morally right to save 5 people even if it involves actions that are intrinsically bad (e.g., murder)? Note, however, that the moral complexity (or undecidability) of such a problem does not reside in the computational-level problem itself, but rather in how the conflict between algorithmic-level solutions should be resolved [e.g., through the doctrine of double effect (Foot 1967)]. Regardless, the interpretation is still consistent with the view that different NTs could be employed to solve different problems, depending on the nature of the problem and the available resources. This seems to resonate with experimental studies that shows that humans are flexible with regard to the moral strategies they employ in different contexts (Capraro and Rand 2018; Conway and Gawronski 2013; Greene et al. 2008). Intuitively, facing

[14] Normative theories that put less emphasis on actions might present an interesting exception; for instance, versions of virtue ethics that emphasize *being* rather than *doing*. However, rather than resolving the distinction, it only pushes it to the blur between *flourishing* and the character traits that enable an agent to flourish. Furthermore, the idea that virtue ethics cannot offer action-guidance have also been criticized; see e.g., Hursthouse (1999) for a virtue theoretic take on action-guidance.

[15] Or at least, the agent uses NT_1 as its main criterion to evaluate whether an action is moral.

[16] The Golden Rule or Kant's categorical imperative (Kant 1785) might be paradigmatic examples of such general-purpose algorithms, which we will discuss in Sect. 5.

some ethical problem E_1, you might be reluctant to perform a certain action because you find the act immoral in itself (according to NT_1), while facing some other ethical problem E_2, no action seem immoral in itself, yet some actions lead to outcomes that seem more preferable than others (according to another theory, NT_2). That is, if no conflict arises between NT_1 and NT_2, you simply pick the one that is best suited for the computational-level problem at hand. Under this interpretation it would be possible to, at least in principle, assess whether some NT is more computationally efficient than another with regard to the same computational-level problem.[17] However, could we ask whether it is more successful, morally speaking? It seems unlikely that an answer can be provided without resolving further meta-theoretical issues.[18] Perhaps more problematically, the interpretation seems to posit that ethical problems are, in some meaningful way, distinctly invariant from the ways they could be solved. To the contrary, the algorithmic solution (NT) seems to depend on the nature of the computational problem itself, and how it affords an algorithmic solution via some NT; affordances that are already embedded at the computational level. If a specific problem is only decidable or tractable for a particular NT, it thus seems more fair to treat it as a computational-level problem in its own right. For instance, we could imagine an ethical problem space which only contains information about obligations, and no information regarding outcomes; it is thus decidable for obligation-based NTs while being undecidable for outcome-based NTs. This naturally leads to an even more narrow interpretation, and the one we will primarily focus on in our analysis:

(3) *Specific moral problems posed by NT as computational-level problems* Instead of placing NTs at the algorithmic level, we could define specific computational-level problems as they are framed by a *specific* NT. In turn, this allows us to be agnostic about the precise procedure that is carried out at the algorithmic-level: we only have to assume that such a procedure exists. As such, (3) provides a number of conveniences for machine ethicists, including (i) answers regarding what *is* moral, or what is morally good to *do* (as prescribed by the modeled theory), (ii) blueprints for action-guidance that can assist algorithmic design and choice of computational method(s), and (iii) means of evaluating performance (e.g., an apt deontological agent successfully adheres to moral duties and rules). Importantly, the narrowness of (3) allows one to ignore theoretical issues that plague (1) and (2): in contrast to (3), it does not have any generality ambition (and could thus be adopted to specific contexts or domains); in contrast to (2), the relevant action-guiding aspects of the modeled NT are already embedded at the computational-level problem description. I.e., while (3) accommodates the fact that different ethical problems—e.g., the information provided in a certain environment—give rise to different affordances with regard to ethical behavior, (2) does not. Perhaps most importantly, (3) allows us to analyze the algorithmic level of ethical problems, while (2) treats the normative theory as the algorithm itself, which potentially obscures the analysis of how such a procedure is actually carried out.

[17] Similarly, even if we believed that only one NT is correct, that does not necessarily mean that we cannot find an alternative theory useful due to its computational efficiency (even if we generally dislike the theory from a moral standpoint).

[18] For instance, it is plausible that, while a solution provided by NT_1 could be the most computationally efficient, it could also violate some principle from NT_2, yet, no efficient solution exists for NT_2 (e.g., requires exponential time). The issue is thus: what is the most successful NT if we assume that the agent believes that NT_2 is morally superior to NT_1?

Another principal strength with (3), with respect to a complexity analysis, is that it constitutes an *essential* yet the *least complex* aspect of ethical computing, in the sense that both (1) and (2) presuppose that an agent can perform computations of type (3). In other words, since interpretation (1) is a generalization of (2), which, in turn, depends on specific instantiations of (3), they form a hierarchy of ethical computations (illustrated in Fig. 1). That is, to solve the generalized moral problem (interpretation 1) following some normative theory, e.g., NT_1, it requires that an agent can also apply NT_1 to specific moral problems (interpretation 2); to apply NT_1 in the particular (interpretation 2), it requires that an agent can apply NT_1 in the very way it is framed by NT_1 (interpretation 3). Thus, if some specific (3)-type computation is undecidable, it would follow that it is undecidable for type (1) and (2) computations of the same problem; it is undecidable for the NT in the specific case (2) and thus in the general case (1). The difference between the interpretations is further illustrated in Table 1.

3.1.2 Level 2: algorithm

Thus, we believe that a natural way to analyze the computational complexity of ethics is to focus on problems posed by normative theory (computational level), that are solved through a variety of computational methods (algorithmic level), by a deterministic Turing Machine (implementation level). Of course, this still leaves a rather vast interpretative leeway regarding what goes on at the algorithmic and implementation level. To find the most effective algorithmic solution to a well-defined problem is often an empirical question, and answers are continuously revised in light of new advancements in programming techniques (e.g., breaking down a problem into simpler sub-problems through dynamic programming) or heuristics (e.g., exploiting regularities in the problem). More importantly, it also depends on what we accept as a solution. If we believe that the NT strictly dictates that the system should find the optimal solution to a problem, it entails that the algorithmic level should follow some procedure that is guaranteed to produce an optimal solution; a so-called *exact* algorithm.[19] A less strict interpretation is to accept solutions that are "close enough" to the optimal; so-called *approximate* algorithms. Although approximate algorithms are not guaranteed to find an optimal solution, they guarantee that the solution is within some fixed distance to the optimal one (i.e., there is a provable bound on the ratio between the optimal and approximated solution). The difference between approximate and exact makes all the difference with regard to tractability, since many problems that have intractable exact solutions can be approximated in polynomial time (Williamson and Shmoys 2011).

We will mainly focus on exact solutions for two interrelated reasons: (i) it is prescribed by the normative ideal (following the strict interpretation), and (ii) it allows us to focus on problems as opposed to algorithms. The first reason can be supported by the following consideration: if approximate solutions are acceptable, how can we motivate that a solution is within an acceptable distance to the optimal? Note that, although an approximation yields a provable guarantee of the distance, this distance can still be arbitrarily large.[20] It seems as if we then need to also define what an acceptable distance is, which might vary greatly from case to case. Furthermore, many real-world problems exhibit no identifiable structure

[19] For instance, this is analogous to the way Φ produces an optimal (albeit intractable) solution to the salad problem following the CPG.

[20] We can, for instance, imagine a dilemma where the optimal solution has a moral value of 100, but the best approximation only yields a value of 50.

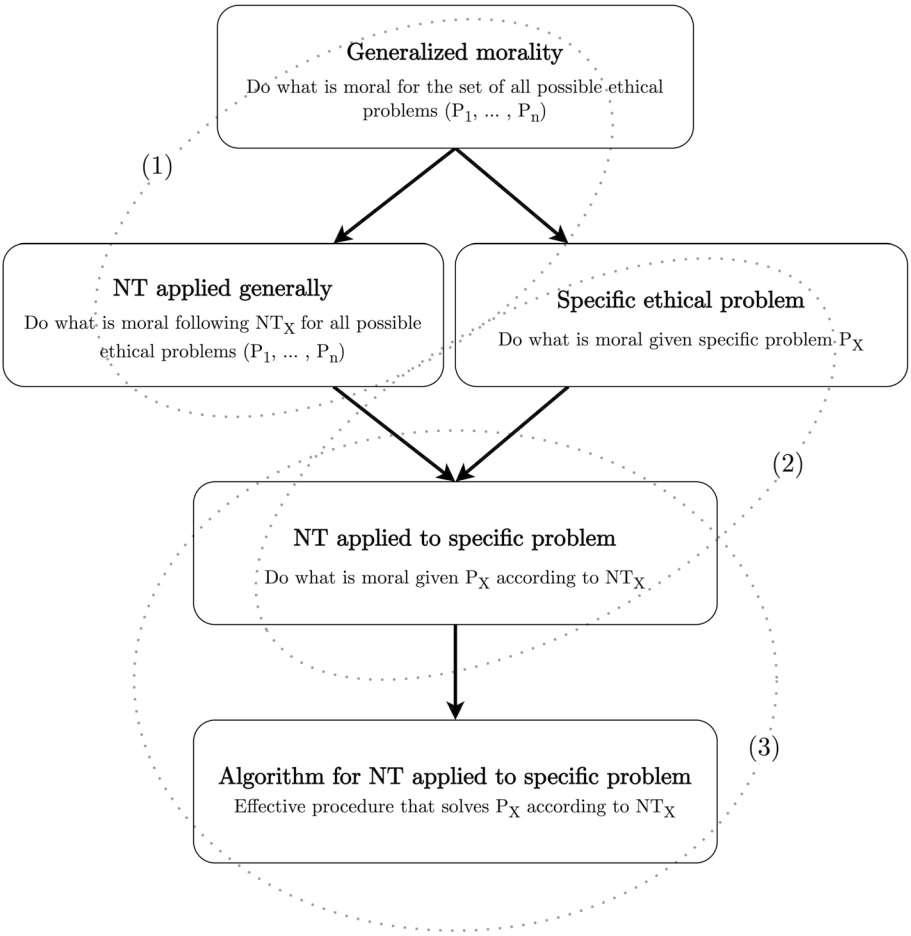

Fig. 1 Hierarchy of ethical computations. Arrows indicate dependency (i.e., A → B means that solutions to A depends on solutions to B). The dotted ellipses capture the computational (top) and algorithmic (bottom) level of the three interpretations (1)–(3)

that can be exploited, and as such, they yield no efficient approximation algorithms (Nievergelt et al. 1995). This naturally leads to the second reason, which is simply that it is easier to compare exact as opposed to approximate solutions, as we do not need to define the conditions under which an approximation is sufficiently close.

Of course, moral theorists might rightfully point out that we should not be interested in exact or optimal solutions to moral problems, but rather, we should understand them in terms of what is "permissible" or "impermissible". For instance, an action might be permissible even if it is suboptimal, and morality does not require us to do anything more than what is permissible, as long as we avoid what is impermissible. This line of reasoning might, in turn, serve to justify the use of suboptimal approximations. However, this would obscure the difference between optimality as a mathematical concept and as a moral concept. Moral permissibility could, for instance, be construed *as* the mathematical optimal; i.e., some fixed point or metric to evaluate behavior against. Alternatively, moral

Table 1 Three interpretations on how ethical problems under normative theory can be framed and analyzed within Marr's three levels

Interpretation	Computation What is the problem?	Algorithm What is the solution?	Implementation How is it implemented?
(1) NT as algorithmic solution to generalized morality	Do what is moral (general behavior)	NT_X (applied generally)	Mind/machine
(2) NT or NTs as algorithmic solution(s) to specific moral problems	Specific moral problem *P*	NT_X (applied to *P*)	Mind/machine
(3) Specific moral problem *P* posed by specific NT_X	*P* as framed by NT_X	Computational methods	Mind/machine

permissibility could be construed as a mathematical approximation of some fixed notion of moral optimality. But then, again, we are led back to the same dilemma we wanted to avoid: in each case, we need to justify how a given approximation is acceptable given a certain threshold of moral permissibility. It is important to note, though, that this does not exclude the possibility that such approximations can be justified in relation to permissibility in particular contexts, but rather that such an analysis is beyond the scope of this paper.

3.1.3 Level 3: implementation

On the level of implementation, we will adopt the most widely used model of computation: the Turing Machine (TM) (Turing 1936). More specifically, since a TM is a mathematical model of computation, it denotes any physical system that can realize a TM (i.e., it is Turing complete). Turing claimed that every function that can be computed by an algorithm can be computed by a TM. The thesis gained credence when Turing showed how his notion of computability was equivalent to the independently suggested proposal by Church (1936). This forms the basis for the Church–Turing thesis, which in turn has been shown to be equivalent to many other forms of computation (Herken 1995). Simply put, it means that any general-purpose system (e.g., computer or computer language) can simulate the computational aspects of any other general-purpose system. We will also assume that P $\neq$ NP (discussed in Sect. 2). Like the Church–Turing thesis, it is another widely accepted conjecture among computer scientists, even if it remains to be proven.[21]

Importantly, if we can show that a problem is NP-hard, it means that we cannot expect to find an efficient solution to it, where "efficient" means "solvable in polynomial time for a deterministic TM" (P-tractability).[22] Therefore, if ethical problems solved by computational methods are NP-hard, we cannot expect computational systems to solve them efficiently, and as such, it would yield direct consequences for the feasibility of moral machines. However, even if P $\neq$ NP and the Church–Turing thesis have near-universal acceptance, it is crucial to address a few caveats regarding the limitations and relevance for the notion of P-tractability. For instance, P-intractability is of no major concern if it is guaranteed that the input size remains sufficiently small (e.g., a salad bar with 5 ingredients only yields 32 possible combinations). Importantly, simply because a problem is P-intractable, it does not mean that it cannot be solved effectively under other reasonable conceptions of tractability. In fact, many NP-hard problems can be solved by algorithms whose runtime is superpolynomial in only *some* part of its input (input parameter), while the runtime is polynomial in the overall input size.[23] Conversely, large constants in polynomial functions, e.g., n^{100}, are P-tractable even if they might fail to capture any intuitive

[21] Note that the Church–Turing thesis is not a conjecture in the mathematical sense, but rather a hypothesis about the nature of computation; it cannot be proven since its notion of effective calculability is defined informally. With that said, the fact that every attempt to define the concept of "effective calculability" has picked out the same class of functions (namely those computable by a TM) is often taken as strong support for the thesis (Copeland 2020).

[22] P $\neq$ NP entails that we cannot expect to find effective solutions to NP-complete problems, and NP-hard problems which can be translated to NP-complete decision variants. Note that many NP-hard problems would still remain intractable even if P = NP, e.g., if they are complete for complexity classes that are believed to encompass NP (e.g., PSPACE or EXPTIME).

[23] It is this very observation that has motivated the development of parameterized complexity (Downey and Fellows 2012), and the class of fixed-parameter tractable problems (FPT). See also Fellows (2002) and Niedermeier (2006).

notion of "effective". Furthermore, time consumption might also be significantly reduced with alternative models of computation, e.g., utilizing parallelization, random access memory, or quantum computing. While the Invariance Thesis—along with the closely related extended Church–Turing thesis (Kaye et al. 2006; Bernstein and Vazirani 1997)—states that no machine can be super-polynomially faster than a deterministic TM.[24] it remains to be seen whether and to what extent it can be falsified in light of future advancements in computing.[25] The main point is that, although P-tractability constitutes an indispensable tool for the formal study of effective computing in theory and practice, it should not be interpreted as drawing a definitive line, across the board, between what is tractable and what is not. And while P-tractability has direct consequences for moral machines, a related yet even more convoluted question is whether it could provide any relevant insight into the moral cognition of humans (a question we will return to in Sect. 7).

To divide our problem space, we will focus on three types of moral machines: causal engines (Sect. 4), rule-followers (Sect. 5), and moral learners (Sect. 6). The main reason is that nearly all implementations in machine ethics take one of these approaches (Tolmeijer et al. 2020). Another reason is that these types each correspond to a prominent normative framework: consequentialism is about predicting future events (causal engines), deontology is about adhering to rules or duties, and virtue ethics emphasizes learning.[26]

In order to be subject to a complexity analysis, we will also assume that ethical problems can be cast as well-defined computational problems (of the kind discussed in Sect. 2). This means that they have clearly defined initial conditions and goals (e.g., in terms of specific input and output conditions) which can be formally represented by mathematical concepts—e.g., numbers, functions, sets, lists, graphs—and be solved by algorithms. For instance, a set of possible actions (e.g., taken as inputs) may be represented by the indices of a list (a number of ordered values) or the nodes in a directed graph (a set of vertices and edges), and morally relevant measures and values (e.g., the benefit of an outcome) may be represented as a numerical value (e.g., a real or integer number such as 6.54 or 3). While these simplifying conditions might do little justice to the vastly rich and potentially ill-defined ethical problems agents might face in the real world, it can be motivated by the fact that real-world ethical problems, given that they are decidable at all, are at least as rich in information as their simplified computational counterpart. In technical terms, we assume that well-defined computational problems represent a reasonable lower-bound on the information-theoretic nature of ethical problems in real-world environments. Finally, we will mainly focus on time rather than space complexity for the simple reason that accessing and storing memory consumes time, which means that memory consumption is often upper-bounded by time consumption (Garey and Johnson 1979).

[24] Note that it is generally believed that the Invariance Thesis applies to both parallel and serial models of computation, see, e.g., Frixione (2001), Parberry et al. (1994) and Tsotsos (1990).

[25] E.g., it is not unlikely that *some* future computer could at least solve *some* problems in polynomial time that are currently intractable.

[26] Of course, as we will see later on, in many cases the line between these theories and types become blurry.

4 Consequentialism and causal engines

Consequentialism is a family of normative theories that puts outcomes at the center of moral evaluation. While all consequentialists agree on the moral importance of outcomes, they might disagree on what a good outcome is, or alternatively, what *makes* an outcome good. For instance, utilitarianism—arguably the most influential branch of consequentialist theories—prescribes actions that maximize utility, where utility can be understood as the overall well-being of the individuals affected (Bentham 1789; Mill 1861), satisfaction of their preferences (Singer 2011), reduction of their suffering (Smart 1956), or the well-fare of their state (Sen 1979). There are also many nuances regarding the way outcomes are morally important, e.g., whether intended consequences matter (as opposed to only actual consequences), whether they depend on the perspective of the acting agent (i.e., agent-relative as opposed to agent-neutral), whether indirect consequences matter (as opposed to the direct consequences of the act itself), for whom they matter (e.g., a limited set of individuals or all sentient beings on earth), and for how long (e.g., only immediate outcomes or for all eternity) (Sinnott-Armstrong 2021). Nevertheless, what is common to all forms is the commitment to the moral value of future events. Therefore, any agent—artificial or biological—committed to consequentialism must be able to make predictions about the future, insofar as they are committed to carrying out the prescriptions of the theory in practice. This is why successful consequentialist agents rely on so called "causal engines", a term we use to broadly refer to the information processing that supports causal cognition.

Note that, in some way or another, most biological organisms care about the consequences of their actions, as it greatly increases their chance of survival. Intuitively, causal cognition appears to be critical for many essential capabilities such as avoiding harm, problem-solving, and planning. Experimental results indicate that human children, as young as eight months, can make inferences based on cause and effect (Sobel and Kirkham 2006). This might suggest that some form of pre-reflective capacity for causal inference could be deeply engraved in our very biological being, reflecting the predictive processing that many believe to be *the* central function of nervous systems (Friston 2010; Hohwy 2013; Keller and Mrsic-Flogel 2018). However, unlike biological organisms, machines did not develop causal engines through an evolutionary process. Instead, an artificial system's ability to follow consequentialism relies on computational techniques, often stemming from the families of statistical, Bayesian, and Markovian modeling (Casella and Berger 2021). It is also common to view machine learning methods as a form of "predictive analytics" in the sense that algorithms learn to make better predictions based on experience; e.g., in supervised learning via human-generated data, in reinforcement learning through an interactive process of trial-and-error. But consequentialism is not solely about making predictions about the future. It is also about evaluating, from the set of possible outcomes, what outcomes are morally preferable over others. That is, even if a consequentialist agent could predict the outcomes of all possible actions with godlike accuracy and speed, it does not necessarily mean that it can easily decide, with the same speed, which the optimal outcome is.

In light of these considerations, this section will explore the computational complexity of three general types of consequentialist problems: combinatorics of determining the optimal outcome (Sect. 4.1), causal inference (Sect. 4.2), and decisions in dynamic and partially observable environments under different time horizons (Sect. 4.3). The section is written so as to incrementally introduce uninitiated readers to time complexity analysis, probability theory (Bayesian Networks), and stochastic methods (Markov Decision Processes).

4.1 The combinatorics of outcomes

In the most simplified case, we could think of the problem a consequentialist face when they compare the moral value of different outcomes, given that the agent can already determine what these outcomes are. In this way, we can ignore the complexity of the causal inference itself so as to isolate the problem of optimal outcome evaluation. In complexity theoretical terms, we assume that the agent has access to a so-called *oracle machine*, which is able to provide answers regarding causal events in a single operation. For instance, if the agent asks "what happens if I perform action a?", the oracle gives an answer of the type "action a yields an outcome with a moral value of v".[27] The most trivial computational problem of this kind can be formalized in the following way:

C1: OPTIMAL OUTCOME FOLLOWING CONSEQUENTIALISM

Input: An environment as a set $E = \{a_1, a_2, \ldots, a_n\}$ of n possible actions and a value function v that assigns an outcome value to each action $a \in E$.
Output: An action $a \in E$ such that $v(a)$ is maximized over all possible actions in E.

An optimal solution can be guaranteed by the following generic exhaustive-search algorithm:

Algorithm 1 Exhaustive search with causal oracle

```
1  h ← 0 // outcome value (0 set as default)
2  j ← 0 // index of action (0 set as default)
3  for a_i = a_1 to a_n ∈ E do
4  |  h_i ← v(a_i) // call oracle
5  |  if h_i > h then
6  |  |  h ← h_i // update highest value
7  |  |  j ← i // update index of highest value
8  |  end if
9  end for
10 return j // return index of action with highest outcome value
```

In short, the algorithm initializes default values for outcomes (step 1) and the index of actions (step 2). It then loops through each action in the environment (step 3), calls the oracle (step 4), checks if the outcome of that action is higher than the current highest (step 5), and if so, updates the highest outcome value (step 6) and its index (step 7). Finally, it halts after returning the index of the highest outcome (step 10). If we assume that each instruction requires an equal amount of time (1) to be executed, we can count the precise number of machine operations the algorithm needs to solve C1 in the following way: lines 1, 2, and 10 needs to be executed just once (3), lines 3–7 needs to be executed n times each ($5n$), and 8 and 9 can be ignored (as they are flow control statements), which yields a total

[27] To encompass many versions of utilitarianism, we will remain agnostic about the exact nature of the utility that ought to be maximized; the only important thing is that it can be represented as a numerical value.

of $3 + 5n$. In Big O, this collapses into $O(n)$. In other words, the time complexity grows linearly ($O(n)$) to the size of the input. Importantly, regardless of how fast a machine can execute the other instructions, to ensure optimality, it must ask a number of questions to the oracle which is at least equal to the number of possible actions. I.e., if there are 10 actions, the agents must make, at minimum, 10 calls to the oracle.

What happens if we allow for multiple values? For instance, we could assume that the agent has a set of two or more outcome values that needs to be checked for each action-outcome (e.g., pleasure, fairness, trust, etc.). This yields the following problem:

C2: OPTIMAL COMBINATION OF VALUES

Input: Same as C1 with the addition of a set of outcome value functions $V = \{v_1, v_2, \cdots, v_i\}$ assigned to each $a \in E$.
Output: An action a such that $v(a)$ is maximized over all $v \in V$ and $a \in E$.

If we posit that the values interact trivially, in the sense that values can be summarized $v_1(a) + v_2(a) + \ldots + v_i(a)$ to yield a single total value $V(a)$ (i.e., obeying the law of additivity), the optimal action a^* can be formally expressed as:

$$a^* := \underset{a \in E}{\operatorname{argmax}}\, V(a) := \{a \in E : \sum_{m=1}^{i} v_m(\acute{a}) \leq \sum_{m=1}^{i} v_m(a) \text{ for all } \acute{a} \in E\} \tag{1}$$

If the agent needs to make distinct calls to the oracle for each value, the time complexity is the product of n (number of actions) and i (number of values), yielding $O(ni)$. If i is equal to the number of actions, the runtime grows quadratically in relation to n, which still yields the polynomial $O(n^2)$.[28]

We have so far only been focusing on the moral evaluation of a single action. But in ethical decision problems of the real world, it is possible to perform multiple actions. However, as illustrated in the salad example, the possibility of combining actions can present tractability issues that are inherent to permutations of combinatorial structures. To show how this affects the computational complexity of consequentialism,[29] we define an action plan $\varphi = \{a_1, a_2, \ldots, a_n\}$ as a distinct non-empty set of n actions presented by the environment such that $\varphi \subseteq E$. We then augment C1 to describe the following problem:

C3: OPTIMAL PLAN OF UP TO TWO DISTINCT ACTIONS

Input: An environment as a set $E = \{a_1, a_2, \ldots, a_n\}$ of n possible actions and a function v that assigns an outcome value to each action plan $\varphi \subseteq E$.
Output: An action plan φ such that $v(\varphi)$ is maximized over all $\varphi \subseteq E$, no $a \in \varphi$ is identical to itself (i.e., the same action cannot be performed more than once), and $|\varphi| \leq 2$.

The only way to solve C3 is to make a number of calls to the oracle which is equal to the number of possible action plans (with a maximum of two actions). This number will grow triangularly ($\frac{n(n+1)}{2}$)—i.e., half of a square—with the number of actions.[30] This tractable

[28] To show this result in an algorithmic procedure, we can simply extend the exhaustive search (Algorithm 1) to iterate n actions over i values, e.g., by adding one additional **for**-loop for each i, or as a nested loop over values 1 to i within the loop over actions 1 to n.

[29] See Lindner et al. (2020) for a similar analysis of action plans based on the SAS$^+$ formalism.

[30] As the famous story goes, Carl Friedrich Gauss quickly identified the formula for this series at a young age when asked by his teacher to add all the numbers between 1 and 100.

procedure would satisfy the Combinatorial Principle of Actions (CPO), i.e., that action plans yield outcome values that does not necessarily correspond to the sum of its individual actions if performed in isolation. It would, however, violate a fundamental principle of causality: that the resulting outcome of two causal events, action x and action y, depends on the order in which x and y occurs. We can call this the Principle of Causal Order (PCO). In order to satisfy PCO when solving C3, the consequentialist must make an additional triangle of calls to the oracle, which completes the quadratic growth of $n(n-1)$. In asymptotic Big O, however, solving C3 in either way results in a time complexity of $O(n^2)$, which is still comfortably within tractable bounds.

The computational complexity of action-outcomes becomes an issue for the consequentialist when we generalize problems of type C3, e.g., to account for n number of actions:

C4: OPTIMAL PLAN OF UP TO n DISTINCT ACTIONS

Input: Same as C3. **Output:** An action plan φ such that $v(\varphi)$ is maximized over all $\varphi \subseteq E$, no $a \in \varphi$ is identical to itself, and $|\varphi| \leq n$.

The time complexity of an exact algorithm that solves C4 while adhering to the CPO is $O(2^n)$. In other words, there is no polynomial-time tractable procedure for consequentialists who try to solve problems of type C4.[31] Worse still, if the consequentialist should also adhere to the PCO, an exact algorithm would yield the factorial growth of $O(n!)$ (Sloane 2022).[32] More broadly, it is well-known that many planning tasks are PSPACE-complete (Bylander 1991, 1994; Littman et al. 1998).[33]

Note that, while this intractability might not constitute a detrimental issue in practice—e.g., for small inputs, say, four possible actions, solving C4 following CPO requires 15 calls, whereas following CPO and PCO requires 64—C4 still presupposes a large set of other non-trivial assumptions that might not hold in real-world situations. For instance, it assumes that agents cannot perform the same action more than once, and that the problem space remains static while the agent computes the solution. By contrast, real-world environments might present a potentially infinite set of possible actions in a state space which is only partially observable and changes in continuous time (which we will return to in Sect. 4.3). Above all, the agent cannot make any calls to a causal oracle but needs to rely on its own causal engine; which leads us to the complexity of causal inference.

4.2 Causal inference

As soon as we enter the realm of uncertainty, we cannot guarantee that the performance of any system will be optimal. Instead, the best we can hope for is optimal according to our "best guesses", i.e., in virtue of what we believe or know (Bayesian optimality). This is part of the reason why many versions of utilitarianism are revised so as to stress the maximization of "expected" as opposed to actual utility (Broome 1987). It also forms the basis for the expected utility hypothesis (Von Neumann and Morgenstern 1947), which is widely used in decision theory and economics to model rational choice, preferences,

[31] This is analogous to the salad problem following Φ (where the CPO is exchanged for the equivalent CPG).

[32] This series is called "the number of permutations of nonempty subsets of $\{1, \ldots, n\}$" (Sloane 2022), and can be expressed as the floor function of $en! - 1$, where e denotes Euler's number.

[33] See also Bäckström and Nebel (1995) for some tractable results for SAS+ planning.

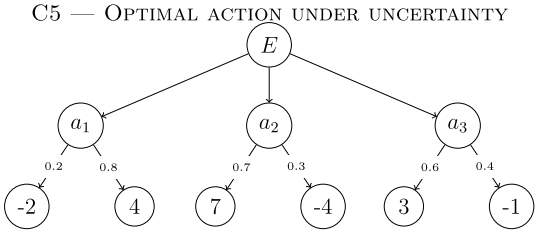

Fig. 2 A directed acyclic graph (DAG), representing a decision problem under uncertainty

and risk appetite (i.e., openness and aversion to risk) when payoffs are unknown.[34] Note, however, that different ways to model probability leaves room for interpretations that carry moral weight, in the sense that different normative principles can guide how decisions under uncertainty should be tackled.

This is illustrated in the following problem, represented as a directed acyclic graph (Fig. 2). The graph shows an environment with three actions, each with a probability of yielding one out of two possible outcome values. If we simply want to maximize expected utility regardless of risk, we can simply add the product of each outcome value with their respective probability—e.g., $0.2(-2) + 0.8(4)$ for a_1—and select the action with the highest expected utility. Alternatively, a more risk averse option would be to select the action with the *best* worst-case outcome (a decision rule called "minmax", i.e., maximizing the minimum gain). While these two decision procedures make little difference with regards to runtime—like our solution to c2, both take $O(no)$ time, where o refers to the number of outcomes for each action—they make a significant moral difference.

However, like c1–c4, c5 still assumes some sort of Bayesian oracle, which is able to infer the exact posterior probabilities that certain events (outcomes) will occur given certain causes (actions). More broadly, causal inference can be understood as the ability to identify *what causes what*, e.g., "what is the cause (or causes) of phenomenon *X*?", "what is the effect (or effects) of *Y*?", and "what is the causal relationship between *X* and *Y*?". None of these questions are trivial; indeed, scientific endeavors are to a large extent driven by answering causal question through a combination of carefully collected data, a vast set of statistical modeling techniques, and causal reasoning capacities such as deductive (deducing from given premises), inductive (inferring from observations), and abductive reasoning (inference to the best explanation).

One essential aspect of causal inference is to determine posterior probabilities based on prior knowledge, i.e., to the determine the likelyhood of A given evidence or belief B. In statistical modeling, the Bayesian interpretation of probability offers a popular response to this challenge. Bayesian methods—e.g., Bayesian inference, networks, and statistics—are all based on Thomas Bayes' theorem, which states that the probability of A given B is provided by the equation $\Pr(A|B) = \frac{\Pr(B|A)\Pr(A)}{\Pr(B)}$.[35] Bayesian modeling have been used to address

[34] Specifically, the hypothesis states that an agent chooses between alternatives by comparing expected utility values, which is commonly calculated as the weighted sum of utility values U multiplied by their probabilities P, in the sense that $\sum U(x_i)P_i$.

[35] Applying Bayes' rule to determine the likelyhood of some causal event A given B is a trivial procedure given that we have (i) some prior probability that A (before B) $P(A)$, (ii) some estimated evidence for the probability that $P(B)$ without A (and $P(B) \neq 0$), and (iii) an estimate of the converse likelihood that B happens given that A.

and model a vast range of cognitive phenomena, such as motor control (Körding and Wolpert 2006), symbolic reasoning (Oaksford and Chater 2001), animal learning (Courville et al. 2006), causal learning and inference (Steyvers et al. 2003; Griffiths and Tenenbaum 2005), inductive learning (Tenenbaum et al. 2006), goal inference (Baker et al. 2007), and consciousness (Lau 2007).

Among the most powerful and widely used extensions of Bayes theorem is the construction of graphical models, called Bayesian networks (BNs) (Pearl 1985), which can succinctly represent a large set of variables and their conditional dependencies as a single DAG (Fig. 3). BNs have been particularly useful in addressing the learning of causal relationships in humans (Griffiths et al. 2008). While the nodes of a BN represent Bayesian variables of interest—e.g., hypotheses, observable quantities, occurences of events, features of objects—the links (or edges) represent conditional dependencies between the variables. Each node has a probability function that returns a variable depending on its parent variables (following Bayes' theorem), and nodes that are not connected are conditionally independent of each other. For instance, the BN illustrated in Fig. 3 describes the causal relationships between eight variables: whether it is a public holiday (x_1), whether it is raining (x_2), whether two or more train operators are currently working at the train station (x_3), whether the operators are stressed (x_4), whether there is a runaway trolley (x_5), whether a lever is pulled (x_6), and whether the trolley is on course to collide with 5 (x_7) or 1 (x_8) people. Since BNs supports the inference of probabilities for any possible subset of variables (i.e., on the basis of evidence about those subsets), it can be used to support causal reasoning processes in any direction of the network. Using the chain rule of probability,[36] the joint probability—i.e., the probability distribution on all possible combinations of values—is given by:

$$P(x_1, \ldots, x_n = \prod_i P(x_i|\psi_i) \tag{2}$$

where ψ_i denotes the values for the parent nodes of x_i. The joint distribution for the network in Fig. 3 is therefore: $P(x_1, \ldots, x_8) =$

$$P(x_1)P(x_2)P(x_3|x_1)P(x_4|x_1, x_2, x_3)P(x_5|x_3, x_4)P(x_6|x_5)P(x_7|x_6)P(x_8|x_6) \tag{3}$$

We can now describe a range of Bayesian inference problems for consequentialism, such as:

C6: BAYESIAN TROLLEY PROBLEMS

(a) Likelihood—what is the probability P that x_7 is true? (given full, partial, or no evidence about its parent variables)
(b) Conditional probability—what is the probability that x_7 is true given evidence that it is a public holiday ($x_1 = true$)?
(c) Causal reasoning—e.g., what effect does pulling the lever ($x_6 = true$) have on x_7 or x_8 ?
(d) Most probable explanation (MPE)—what is the most probable configuration of a *set* of variables given *full* evidence about the complement of that set?

[36] The chain rule allows one to compute the joint distribution of a set of variables solely using conditional probabilities, in the sense that $P(A \cap B) = P(B|A)P(A)$.

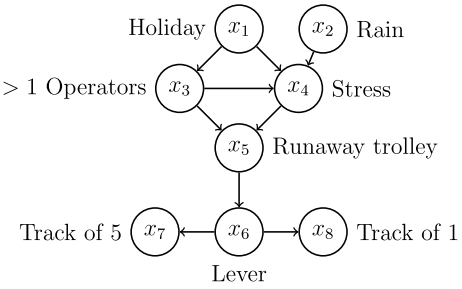

Fig. 3 A Bayesian network representing the causal relationships of eight Boolean variables for the Bayesian Trolley Problem (C6). Since their introduction in the 1980s, Bayesian networks have facilitated evidence-based prediction in complex domains such as medical diagnosis (Lucas et al. 2000) and weather forecasting (Cofino et al. 2002)

(e) Maximum a posteriori hypothesis (MAP)—what is the most probable configuration of a *set* of variables given *partial* evidence about the complementing set?[37]

BNs are perfectly suited to answer such causal inquiries, using algorithms such as variable elimination (Zhang and Poole 1996) and message-passing (Pearl 2022) for exact inference, and random sampling (Pearl 1987) for approximate inference. However, even if questions like C6 (a)–(e) can be solved in reasonable time for constrained networks, it has been proven that most inference problems for Bayesian Networks are intractable in general. More specifically, exact inference on arbitrary graphs is NP-hard (Cooper 1990),[38] which means that inferring the exact probability of some event (or that a propositional expression is true) is at least as hard as the hardest problems in class NP. Furthermore, the decision variant of finding the most probable explanation (MPE) is NP-complete (Shimony 1994), while the related maximum a posteriori hypothesis (partial MAP) is NP^{PP}-complete (Park and Darwiche 2004).[39] Perhaps more intriguing is the results that approximations of these problems are also intractable: approximating exact inference (Dagum and Luby 1993), MPE (Abdelbar and Hedetniemi 1998), and partial MAP (Park and Darwiche 2004) are all NP-hard.[40]

One important lesson from these results is that the complexity of Bayesian inference depends on the *structure* of the network: while constrained graphs yield a bound on the number of conditional dependencies and parent variables for each node, unconstrained graphs cannot be exploited for effective computation. For instance, for chain-like graphs of the type $x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4 \rightarrow x_5$, an elimination algorithm can determine the exact inference of $P(x_5)$ by a step-wise elimination of the parent variables, which can be computed in the polynomial time $O(nv^2)$, where n is the number of variables and v denotes the number of possible values the variables can take. However, as the number of variables depending on other variables grows, inference in BNs starts to mirror the problem of determining

[37] MAP (sometimes called Partial or Marginal MAP) can be viewed as the generalization of MPE in the sense that it might require a marginalization of the variables that are not observed nor explained. Furthermore, note also that both MPE and MAP has many variants in the literature on Bayesian networks: see (Kwisthout 2011) for a clarification.

[38] More precisely, exact inference is #P-complete (Roth 1996), where #P is the class of counting problems associated with NP.

[39] PP is the class of problems decidable by Probabilistic TM with an error probability of less than 1/2 (Gill 1977), and NP^{PP} is the class of problems solvable by a non-deterministic TM with access to an oracle for problems in PP.

[40] See Kwisthout (2011) and de Campos (2020) for summaries of complexity results for the many variants of the MPE and MAP problems.

whether an arbitrary Boolean formula can be satisfied (SAT): the first known NP-complete problem.[41]

In summary, if a consequentialist agent were to solve causal inference problems using Bayesian networks, we cannot expect that any tractable procedure could yield precise or even approximate solutions for arbitrary graphs. The same intractability results have pestered Bayesian modeling in cognitive science, as Bayesian planning (Körding and Wolpert 2006), learning (Kemp and Tenenbaum 2008), and decision-making (Vul et al. 2014) all presume NP-hard computations. As a potential remedy, we might instead identify the constraining conditions that enable tractable solutions (Kwisthout et al. 2011). For instance, the bounded-variance algorithm (Dagum and Luby 1997) can generate approximations of inferences in polynomial time if extreme conditional probabilities are excluded (i.e., values near 0). Similarly, it has been shown that MPE is tractable when either the treewidth of the underlying graph is low,[42] or the probability of the most probable explanation is high (and partial MAP is tractable when both conditions are true) (Kwisthout 2011). However, this introduces another uncomfortable trade-off: there is no guarantee that such constraining conditions capture reality. For machines, this means that a constrained graph could potentially fail to model the correct causal relationships. With regard to Bayesian modeling of human cognition—e.g., of ethical decision-making under uncertainty—it also means that one must ask whether the constraints are reasonable with respect to the modeled phenomenon. And for the consequentialist philosopher, it poses the question: what are the constraining conditions under which causal inference should be expected to be successful for an agent following consequentialism?

4.3 Decisions in dynamic and partially observable environments

We have thus far only investigated problems where the entire state space of a problem is taken as an input, e.g., as elements of sets or nodes of graphs. But ethical problems of the real-world presents a range of additional challenges that might curb a consequentialists ability to produce the best outcome, including (i) partial information and observability, (ii) dynamic and continuous environments that constantly change, (iii) limited time horizons to make decisions and execute actions (e.g., emergency situations), (iv) a potentially infinitely long time horizon to evaluate outcomes against, and (v) a potentially infinite set of possible actions (e.g., movement in dimensions higher than one). Each challenge reflects well-known epistemological issues for the consequentialist (Lenman 2000), such as, what is the smallest amount of information needed to make a reasonably informed ethical decision (given that information can never be complete)? Or what is the time horizon for which the outcome of an action should be considered (i.e., how long is the future we need to predict)?[43] Time alone might introduce chaotic unpredictability. As meteorologist and

[41] It should be no surprise that SAT has been instrumental in deriving complexity results for BNs.

[42] Treewidth is a graph theoretical concept which can informally be understood as a measure of how much 'wider' a given graph is than a simple tree (in which any two vertices are connected by exactly one path), and more formally as the size of the largest vertex set in a tree decomposition of the graph (Bodlaender 1994).

[43] However, note that both contemporary and classic and consequentialists—e.g., Bentham (1789), Mill (1861), and Sidgwick (1907)—do not assert their principle as a strict decision procedure, but rather as a criterion or standard. See Bales (1971).

mathematician Edvard Norton Lorenz famously noted: a butterfly flapping its wings could result in a tornado a few weeks later.[44]

Nevertheless, a number of mathematical tools have been developed to successfully tackle such issues. In the absence of analytical solutions or evidence, stochastic methods allow us to explore complex phenomena by throwing dice (e.g., Monte Carlo methods), or by viewing them as memoryless chains of events (Markov process). A Markov process is any process that satisfies the Markov property, which means that the likelihood of a certain future state *only* depends on the present state (i.e., it is "memoryless").[45] From a complexity theoretic point of view, the appeal of studying processes in Markovian terms is that it allows otherwise intractable or undecidable stochastic modeling to be tractable (Vanmarcke 2010). Monte Carlo methods denotes another general class of algorithms that are based on repeated random samplings, e.g., by drawing a number of pseudo-random variables within a certain distribution or interval.[46] In turn, these rather simple ideas have matured into an umbrella of stochastic approaches that have been successfully applied to a vast range of scientific problems, e.g., in statistical physics (Binder et al. 1993), engineering (Hajek 2015), and Bayesian statistics (Gelman et al. 2013).

One fruitful application of stochastic methods in the realm of automated decision-making is reinforcement learning (RL). The idea behind reinforcement learning is simple: an agent learns from interacting with an environment by updating its behavior—e.g., strategy or action-policy—in light of the reward it receives. An RL agent is often formalized as a Markov Decision Process (MDP), the 5-tuple $\langle S, A, R, P, \gamma \rangle$, where:

- S is a set of states (called state space)
- A is a set of actions (called action space)
- $R_a(s, s')$ is the reward the agent obtains by transitioning from state s to s' by performing action a
- $P_a(s, s') = Pr(s' \mid s, a)$ is the probability of transitioning from $s \in S$ to $s' \in S$ given that the agent performs $a \in A$
- γ is the discount factor ($0 \leq \gamma \leq 1$) that specifies whether the agent prefers long- or short-term rewards.

The goal of an RL agent is to maximize reward (R) over some specified time horizon. In order to do so, it needs to find an policy, i.e., a function $\pi(s, a)$ which decides what action a to execute given a certain state s. If the goal is to maximize the expected discounted reward arbitrarily into the future (called the *infinite-horizon objective*), the optimal policy π^* can be formalized as:

$$\pi^* := \underset{\pi}{\operatorname{argmax}} E\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \middle| \pi\right] \tag{4}$$

[44] In chaos theory, the "butterfly effect" is the observation that tiny changes in one state of a non-linear deterministic system can produce massive differences in later stages (Lorenz 1963).

[45] As such, Markov processes constitute a broad class of both continuous and discrete stochastic processes; for instance, Poisson, Wiener or Brownian motion, and random walks can all be formulated as special variants of a Markov process.

[46] As the story goes, the modern version of the Monte Carlo method was invented by Stanislaw Ulam while working with the Manhattan Project in Los Almos; in fact, the method were instrumental for deriving the simulations required for the project's success (Haigh et al. 2014).

RL—in combination with Monte Carlo, deep neural networks, and other techniques—have yielded super-human performance in complex game environments such as Dota 2 and Go (Berner et al., 2019; Silver et al., 2018), or, more recently, to notable advancements in the control of nuclear fusion plasma (Degrave et al. 2022). More broadly, it has been argued that reward is enough to drive all forms of behavior that are associated with natural and artificial intelligence, such as learning, knowledge, perception, language, social intelligence, and generalization (Silver et al. 2021). Due to its general applicability, it has been suggested that RL provides the appropriate framework to theorize about an ideal ethical artificial agent (Abel et al. 2016), or for the construction of artificial virtuous agents (Stenseke 2021).

Importantly, RL is able to address many of the factors that might curb ethical agents' decision-making: continuous dynamics (Serfozo 1979), partial observability (Cassandra et al. 1994), and objectives over different time horizon.[47] One key challenge in RL is the trade-off between exploration and exploitation. I.e., when we do not have perfect information, should we decide on the basis of what we already know (exploit), or take the risk of investigating options that would potentially be even better (explore)? In theory, the explore-exploit dilemma could be solved through the notion of partial observability, which offers a way to model what is and what is not directly observable by the agent. A partially observable Markov decision process (POMDP)[48] augments the MDP 5-tuple by adding two additional terms: a set of observations (Ω) and a set of conditional probabilities (O), which represent the likelihood of observing $\omega \in \Omega$ if the agent performs a and the environment transitions to hidden state s', in the sense that $O = Pr(\omega \mid s', a)$. In short, solving POMPDs centers around computing probability distributions over the possible states the agent *could* be in (belief states), where an optimal policy maximizes expected reward in virtue of mapping actions to observation histories. In principle, since an optimal solution to a POMDP incorporates the instrumental value an action has from an information-theoretic point of view—and how the information can be used to make better future decisions—it offers a solution the explore-exploit dilemma.

Unfortunately, finding optimal solutions to POMDPs is undecidable for infinite horizons (Madani et al. 2003). Furthermore, while solutions to finite MDPs and POMDPs are decidable, they are generally not tractable. The results by Papadimitriou and Tsitsiklis (1987) show that finite POMDPs are PSPACE-complete, while the results by (Mundhenk et al. 2000) prove that various MDP problems range from being complete for probabilistic logarithmic space (PL) to being EXPSPACE-complete.[49] Other complexity results in RL indicate a similar trend: reaching a goal state might require, in the worst-case, a number of actions that is exponential in the size of the state space (Whitehead 1991). Intuitively, when no a priori knowledge of the state space can be exploited, unbiased search can lead to excessive exploration. However, worst-case time complexity results alone are insufficient to assess the theoretical viability of RL as a framework for sequential decision-making under uncertainty, as it depends on a number of factors, such as task representation [e.g., number of states and actions (Koenig and Simmons 1993)], the sort of feedback provided by the environment (e.g., observability), policy types [e.g., stationary or history-dependent

[47] E.g., a discount factor γ of 0 only considers short-term rewards, while $\gamma = 1$ without a terminal state considers rewards over infinite time.

[48] The POMDP framework for control under uncertainty was first developed by Åström (1965) and later adapted to problems in AI by Cassandra et al. (1994).

[49] See also Littman (1996) for a detailed survey of the complexity of MDP and POMDPs.

(Mundhenk et al. 2000)], or restrictions on the agent's resources.[50] Similar to the results of Bayesian inference, while there is no sound theoretical guarantee of the success of RL, its practical viability can be significantly improved by simplifying the task representation (given that a simplified representation is achievable), improving the observability of rewards, and by exploiting a priori knowledge. It should be no surprise that RL have been particularly successful in game environments which often affords a simple representation of the state space (e.g., 2-dimensional grids of Chess and Go) and discernible rewards (e.g., Dota or Starcraft).

But there are other issues with RL which might obstruct its applicability for consequentialism. For instance, even if an agent has found an action-policy which maximizes its utility in an environment inhabited by other agents, the policies or preferred utility of the other agents might result in conflicts. Such game theoretic considerations are challenging, especially in combination with partial observability and imperfect information regarding the other agent's strategies and goals (we will return to this issue in Sect. 5.1). Other issues pertains the notion of sample complexity, i.e., the number of training samples a learning algorithm needs to learn a target function (or within some error of the optimal function). However, as we will discuss in Sect. 6, sample complexity is not only plagued by the existence of arbitrarily 'bad' distributions of training data, but it also raises deeper philosophical issues concerning induction.

Perhaps most critically, RL—and stochastic methods at large—presupposes trial-and-error. While this might not be a major issue in simulated games, it presents a challenge for real-world environments which does not necessarily afford the same stochastic exploration; particularly if some actions could have catastrophic consequences. Furthermore, given the stochastic nature of the process, a RL agent might find a way to increase its incentivized reward in a way that conflicts with the very intention of its human designer [called "reward hacking" in Safe AI research (Amodei et al. 2016)].[51]

In a similar vein, the multi-armed bandit problem has generated a rich body of work that investigates the explore-exploit-dilemma under various conditions (Slivkins 2019). In its most basic form, it asks: given n possible actions (arms) which yields some reward drawn randomly from a fixed (but a priori unknown) distribution, how do you maximize the expected gain? Instead of reward maximization, many versions of the multi-armed bandit looks at the learning problem in terms of regret minimization, measured as the difference between the performed action and the optimal action (e.g., given hindsight). The goal is to find strategies that balance exploration and exploitation while minimizing regret. Variations include regret minimization with incomplete information (Zinkevich et al. 2007), contextual bandits where agents receive some contextual information which relates to the rewards (Bouneffouf and Rish 2019; Langford and Zhang 2007), the problem of identifying the best arms (Bubeck et al. 2013), or finding arms whose mean is above a certain threshold (Locatelli et al. 2016). Solutions to multi-armed bandits are typically investigated under one of two assumptions: (i) *stochastic*—the reward distribution for each arm is unknown but *fixed*, or (ii) *adversarial*—the rewards are chosen by an adversary with unbounded computational resources (Auer et al. 1995) (e.g., gambling in a rigged casino). In its most general form, both assumptions are relaxed, which leads to the *restless bandit* problem (Whittle 1988). This means that the payoffs can vary over time even when

[50] For instance, see Chatterjee et al. (2016) for a more recent summary of decidable solutions for POMDPs given finite-memory strategies.

[51] See also Garcıa and Fernández (2015) for a survey of literature in Safe RL.

the arms are not played. For instance, imagine that you are the manager of a kindergarten with n children and m babysitters, and $m < n$. Since the children outnumber the babysitters, the task is to allocate the babysitters' attention in a way that minimizes mischief. While a child is attended to, information about its position, activity, and mood is gained. If it is not attended to, information is lost, and the child might be up to some mischief (they are, in a literal sense, restless).[52] While many tractable solutions exist for different variants of the bandit problem, the restless bandit is proven to be intractable to even approximate. The proof provided by Papadimitriou and Tsitsiklis (1994) shows that for n arms and deterministic transitions for both unattended and actively played arms (i.e., all transition values for attended arms and unattended arms are either 0 or 1), finding the optimal policy is PSPACE-hard.[53] Furthermore, since the proof also shows that it is PSPACE-hard to determine whether the optimal reward is non-zero, it rules out approximate solutions.[54]

Of course, while it is little to no surprise that there are no effective solutions to problems like the restless bandit, it shows that there is no algorithmic way to ensure optimal performance (or minimize regret) in sequential decision-making under uncertainty, unless the nature of the problem space (environment) itself affords exploitation. The intractability results for the restless bandits elegantly illustrate this with respect to making decisions in a changing world. More generally, while Markov chains and Monte Carlo dice-rolls can help to model and mine statistical tendencies of complex spaces, its success presupposes that such tendencies exist. This, however, might say more about the complexity of dynamic real-world processes than it does of the computational limitations of agents. Or as Hofstadter observed: Deep Blue's win against Garri Kasparov says more about chess than it says about human intelligence (Hofstadter 2002). Brożek and Janik (2019) have recently made the analogous remark with regard to moral theory: "the fact that a machine may be a better *homo Kantianus* or *homo Benthamus* than any *homo sapiens* says little about human morality, and much about the idealised nature of philosophical conceptions of moral agency" (p. 103). But the complexity results discussed in this section imply something even stronger: while we might expect AI methods to perform better than humans in a range of tasks related to ethical decision-making, they are also bounded by the complexity of the world, which inevitably curtails any attempt to construct a perfect moral machine.

To sum up, we have explored three sources of complexity that presents tractability issues for computational consequentialist agents. This could imply that any computational agent bounded by polynomial Turing-tractability will fail to adhere to the prescriptions of the normative ideal in practice. As a corollary, it indicates that consequentialism might be better suited as a theoretical ideal, as opposed to a viable decision-strategy that could inform ethical decisions. But what is the point of moral theorizing if it cannot inform moral decision-making in practice? More pragmatically, one might look for the constraining factors—e.g., in the space of possible actions, action-combinations, conditional probabilities, time horizons, task representations, and approximations—under which consequentialist decision-making becomes tractable, and determine, in each case, how closely those decisions approximates the optimal; or *some* fixed point of moral value.

[52] See Whittle (1988) for several other intuitive examples.

[53] Note that PSPACE-hardness entails something stronger than NP-hardness, as PSPACE-hard problems remains intractable regardless of whether P = NP.

[54] However, approximation guarantees are possible if the problem is relaxed to allow for linear programming, in the sense that one arm is played per time step *on average* (Guha et al. 2010).

5 Deontology and rule-followers

While consequentialism centers around outcomes, deontological ethics focuses on actions themselves: whether an action is moral depends on whether the action is obeys a set of moral rules, obligations, or duties. But what justifies a rule in the first place? And how can one ensure that a given interpretation of a rule stands up to the principle it was justified upon? According to divine command theory, the legitimacy and universal validity of moral rules is grounded on the authority of God. The Christian Ten Commandments provides canonical examples of such rules, e.g., "thou shalt not kill" or "thou shalt not steal", given to Moses at Mount Sinai by God. By contrast, in Immanuel Kant's deontological ethics, rational beings are bound to moral law by their own autonomous will, and the fundamental principle for our moral duties is captured in the categorical imperative: "Act only according to that maxim by which you can at the same time will that it should become a universal law" (Kant 1785). This means that, as rational beings, people have a duty to only act according to maxims that a community of rational agents would accept as laws.

It should be stressed that rules and systems of rules are already deeply embedded in most human societies; generated and enforced by social institutions as law. In fact, morality and law share a complex and complementary relationship, as they are both normative systems that seek to regulate human behavior, e.g., in order to foster social harmony and stability of communities. On the one hand, law may compensate for the functional frailty of morality, since the latter lacks the mechanisms to enforce its own prescriptions. On the other hand, morality can serve the coordination of social expectations where law is difficult to apply, e.g., through notions of responsibility, solidarity, and fairness. Furthermore, many legal thinkers believe that, to succeed in its function of regulating behavior, law must resonate with the moral norms and sentiments of its subjects.[55]

Given the rule-based nature of deontology in conjunction with the view that machines are essentially systems of automated rule-following, one might conclude that deontology provides an excellent recipe for moral machines. After all, deontological rules elegantly corresponds to the conditional statements pervading in machine code: e.g., "If input $X \rightarrow$ do action Y". In popular culture, this view has most famously been explored (and problematized) in Isaac Asimov's novels as "Laws of Robotics" (Asimov 1942). The appeal of computational deontology is also well reflected in the machine ethics literature; in fact, Tolmeijer et al. (2020) survey shows that 22 out of 50 implementations in machine ethics incorporate some elements from deontological ethics. Part of the appeal is the commonheld view that deontology is, computationally speaking, less complex than its alternatives (Brundage 2014; Wiegel and van den Berg 2009; Powers 2006; Tolmeijer et al. 2020). There is a technical, psychological, and philosophical dimension to this view:

(1) From a technical perspective, it is intuitively true that it is easier to follow hardcoded rules than to compute consequences (e.g., considering the complexity discussed in Sect. 4). As an example, in their study of the moral evaluation of action plans, Lindner et al. (2020) found that deciding whether a plan is morally permissible according to act and goal-deontology is computable in linear time. By contrast, they also found utilitarianism to be PSPACE-complete by the same metric, and that principles based on harm avoidance are co-NP-complete.

[55] Of course, this question divides legal positivists and non-positivists; the former view holds that law can be valid even if it is morally unjust, whereas the latter holds that law is only valid if it is consistent with moral norms (Moka-Mubelo 2017).

(2) Second, following the work on the psychology of decision-making by Kahneman (2011) comes the influential theory that posits the existence of two distinct aspects of human reasoning: "system 1" which is fast and intuitive, and "system 2" which slow and deliberate. In moral psychology, this has led to the development of the dual process theory of moral cognition (Greene 2007), which postulates that moral judgment rely on both conscious-controlled processes (corresponding to typically utilitarian judgements), and automatic-emotional processes (corresponding to typically deontological judgments). Empirical findings based on the theory has showed, among other things, that an increase in cognitive load (by imposing an additional control-demanding task) leads to an increase in reaction time for utilitarian judgments, while it does not increase reaction time for non-utilitarian judgments (Greene et al. 2008). Another study demonstrated that cognitive load may increase the frequency of deontological judgment, and that utilitarian responses are less likely if the subjects were reminded of their own mortality (Trémolière et al. 2012). Although this does not prove that deontology as a normative theory is more computationally efficient than utilitarianism per se, it suggests that conscious-controlled processes in human moral judgments, in contrast to automatic processes (which *may* be described as characteristically deontological judgments), are more cognitively demanding and susceptible to cognitive load manipulation.

(3) The idea that rules can serve to alleviate the cognitive burden of moral judgments is also widely represented in moral theory; in particular as a move to save consequentialism as a decision procedure in light of the challenges that classic utilitarianism face (Brandt 1979). For instance, modern versions of consequentialism differentiate between *acts* which would produce the most good (act utilitarianism) and *rules* which, if they were followed, would produce the most good (rule utilitarianism). Since the former, if it were to be used as a decision procedure, puts unrealistic demands on agents—e.g., susceptible to biases, lacking complete information about the consequences of actions, or lacking time to make the correct judgments—many consequentialists adopt some version of rule utilitarianism as a decision procedure (Hooker 2016).[56]

Nevertheless, there are computational aspects of deontology—and moral rule-following more broadly—that are relatively ignored in the machine ethics literature. While machine executions of the type "If input $X \rightarrow$ do action Y" might seem trivial, they rest on the conditions that:

(a) X is really X, e.g., the agent has the appropriate understanding that an input (e.g., a situation) has certain properties [or alternatively, a "semantic grounding" of X (Harnad 1990)] and
(b) performing Y is an appropriate response to X in every possible instance of X (or alternatively, Y is intrinsically good).

[56] As an intuitive example: we stop at a red light because we might believe that this particular traffic rule produces the most overall good (e.g., for society), and not because the act itself in this particular instance yields the most overall good. Note that this does not necessarily mean that act and rule utilitarianism are in conflict: act utilitarianism might still be the standard used to evaluate the consequences of rule-adherence prescribed by rules utilitarianism.

If both conditions are satisfied, a deontological agent would be able to act morally in constant time $O(1)$, given that we store the appropriate action-rules for every possible input (which instead puts demand on the space complexity).[57]

In practice, however, conditions (a) and (b) are immensely hard to satisfy: (a) assumes perfect knowledge of any possible state space, and (b) relies on some general ability to understand that an action is an adequate response to a particular state.[58] Against this background, this section will explore the complexity of deontology by discussing the generality of rules (Sect. 5.1), the complexity of logic and semantics (Sect. 5.2), and the prospect of consequentialist-deontological hybrids (Sect. 5.1.3). Additionally, section (Sect. 5.1.2) covers the complexity of strategic dynamics based on algorithmic game theory. We will argue that the moral power of rules lie in their general applicability, general justification, and computational simplicity. However, to be computational efficient at run-time, it presupposes that one has already collected the vast knowledge required for such generalities to hold in practice.

5.1 The generality of rules

The generality of moral rules can be understood in several ways. We will first make a distinction between two important dimensions, namely *application* and *justification*. The first refers to the general ability to decide, given any possible input, whether a certain action is appropriate or inappropriate according to a set of rules. Alternatively put, a general ability to decide whether an action successfully adheres to the rule, principle or obligation it was justified upon. For an agent to successfully follow "do not harm others" in general, it means that it never acts in a way which directly harms another agent. While rules of this type may seem trivial for humans, they are more difficult to approach from a computational perspective. For instance, it is apparent that they presuppose sophisticated abilities to interpret whether actions (or more likely, chains of actions) actually obey the rule: e.g., knowing what another agent is; being aware of the set of possible actions (and sequences of actions) in a dynamic environment that could cause harm to other agents; etc. Here, there is an extreme variance of rule-application with regard to context: a self-driving vehicle will likely cause harm to other agents if it collides with them at high speeds (i.e., "do not harm others" easily translates into collision-avoidance), whereas a social robot in a classroom might be oblivious to the potentially infinite set of actions and causes that could cause harm.

[57] For instance, we can imagine that every input is an index which immediately executes an action from a list of actions. Alternatively, we might use moral duties for deontological action-evaluation, e.g., in terms of constraint satisfaction: given an input X, we simply enumerate over a list of actions and check whether they violate any moral duty (this would be analogous to problem C2 in Sect. 4.1).

[58] As we will discuss later, this may also involve an ability to understand whether an action stands up to the normative principle upon it is justified. Alternatively, to satisfy (b), we might demand that an action Y is intrinsically good, regardless of context (or that certain input-action-pairs are intrinsically good). As a thought-experiment, we could imagine a machine that was hard-coded with exclusively intrinsically good actions, and simply have it repeatedly executing a loop of good actions. Given that the actions were in fact intrinsically good, the machine would be a perfect moral machine. This seems to work for simple machines such as toasters. But in dynamic and partially observable environments, this seems to presuppose God-like omniscience.

The second dimension is whether a rule itself is generally justified. For divine command theorists, rules may be universally justified on the basis of divine authority. In machine ethics, this has inspired the development of divine-command logic, where human input is interpreted as divine command for the machine (Bringsjord and Taylor 2012). For contractualists stemming from Kantian thought, justified principles require instead that they are agreed upon by everyone (Rawls 1980), or that no-one could reasonably reject them (Scanlon 2000). Contractualism puts emphasis on the rationality of agents, which, following Kant's moral rationalism, requires that we respect others in the sense that principles must be justified to each person. By contrast, contractarians stemming from Hobbesian thought—e.g., Gauthier (1987) and Narveson (2001)—puts emphasis on the *self-interests* of agents, in the sense that moral rules ought to maximize the joint interest. Thus, for contractarianism, adherence to moral norms or rules are justified on the basis of an agent's self-interest; and altruism occurs when the agent recognizes that the best way to maximize their self-interest is to cooperate with others.

5.1.1 Human rule-following in legal and liberal contexts

A first step in analyzing the computational complexity of rules stemming from these two notions of generality is to look at rule-following in human practices. One observation is that legislative practices are informed by the principle of *ignorantia juris non excusat* ("ignorance of the law is no excuse") in the sense that laws should be easy to apprehend and easy to comply with. If not, willful blindness can be exploited by defendants. It also entails that human laws are formulated with regard to human capacities: it would be contrary to the purpose of laws if they were formulated so that no human could follow them. However, what separates moral from legal rules, is that the latter are backed up by legislative mechanisms of jurisdiction and may be enforced by the state (e.g., police). Importantly, if it is hard to decide how a certain law should apply in a specific circumstance, we rely on experts (e.g., lawyers, judges, and counselors of various courts) to interpret the law, ensure that it is applied in a just way, and possibly generate a new praxis for applications in the future. In fact, in many legislatures, laws are vaguely formulated by intention in order to be continuously infused with meaning as novel situations transpire.[59] Thus, in such occasions, the normative content of the law is mainly given by its *interpretation*, and not by the law itself.[60] In turn, if application falls short in representing the moral sentiments of the subjects of which the law applies to, it may constitute a failure of law: and defendants might rightfully choose to appeal the verdict. The point is that, while rule-following in terms of legal law-obedience might appear to be relatively simple in human contexts, it is only against the background of rather complex pre-existing mechanisms that ensures their successful implementation. It is therefore hard to make sense of the computational complexity involved in following human laws, as it already presupposes the cognitive capacities (or certain behavioral standards) that makes up human legal personhood, while allocating the processes of interpretation and justification to legal practitioners.

Another observation, which is related to the first, is that laws are often articulated as positive or negative rules, i.e., "do this" or "do not do that". From a human perspective, negative rules are, at least prima facie, simpler than positive rules: it is easier to remember

[59] For instance, the Court of Justice of the European Union.

[60] That is, courts provides the guarantee that a particular application stands up to the principle upon which it was normatively justified. This argument is developed at a greater length in Stenseke (2023).

what one should *not* do, as opposed to what one should do (in other words, it reduces space complexity). This, especially in the context of Western liberal societies, can be reflected in "the harm principle", as found in France's *Declaration of the Rights of Man and of the Citizen* (1789): "Liberty consists in being able to do anything that does not harm others" (Johnson 1990), or in Mill's *On Liberty*: "[...] the only purpose for which power can be rightfully exercised over any member of a civilised community, against his will, is to prevent harm to others" (Mill 1859). However, the memory-saving aspect of such principles, from a computational perspective, only makes sense against the backdrop of autonomous citizens within the context of a liberal society, and the potentially infinite set of actions they can do. For instance, the best way to follow "do not harm others" for a machine, might simply be to do nothing at all: the complexity of adhering to the principle is proportional to the set of actions it *can* do.

5.1.2 General-purpose rules and their justification

What is common to legal rule-following and moral rules based on divine command is that the acting agent does not necessarily have to understand the moral rationale behind the rule in order to behave morally; the agent simply follows the rule without reflecting on whether the rule is morally justified (it is simply the case that God or law says so). Of course, there might be immense disagreement on what the right rules should be. By contrast, for Kantian ethics, contractarians and contractualists, rules may involve justification as part of the action itself, in the sense that an act is only moral in so far as it is accepted by the affected participants (or members of the moral community of which the contract applies to). This is well-reflected in rules that have the ambition of being both generally applicable and generally justified, such as Kant's categorical imperative (CI) or the Golden Rule (GR): "treat others as you would like others to treat you".

It is interesting to note that general-purpose rules that involve justification can be amenable to a complexity analysis if the rule itself provides grounds for deciding whether an action is moral. To illustrate, following the GR, the query "is action X morally permissible given input Y?" is decidable for an agent A, if it is decidable whether A herself would accept X if it were performed by another agent B given input Y. For instance, if X given Y causes harm, and A would not want others to cause her harm, the answer is decidedly no. Of course, while this sort of analysis ignores the complexity involved in the determining the de facto mappings between any possible input, action, and outcome, it gives a general structure which can be analyzed:

(GR1) *Golden Rule based on one's own preferences*[61] For instance, the GR might refer to a set of actions and individual preferences of an acting agent A, in the sense that A has to check whether the individual actions satisfy her individual preferences: she will treat others (act) based on how she likes others to treat her (i.e., how an action, if performed by someone else, would affect her own preferences). Given that the agent can quickly check how actions affect her preferences, the worst-case time complexity of deciding whether any

[61] Here, we remain agnostic about the precise nature of "preferences"—it might as well refer to duties or obligations—in order to encompass many versions of deontology; the only important thing is that it can be represented as a numerical or boolean value.

action is morally permissible is $O(np)$, where n is the number of possible actions and p is the number of preferences.[62]

As a general-purpose rule, GR1 has a great appeal in virtue of its computational efficiency: simply check whether *you* would accept an action based on *your* preferences. Prima facie, it would also capture many preferences that are commonly shared among self-interested agents in the natural world, e.g., to increase one's pleasure and reduce one's suffering. As such, it would therefore work to prevent actions that, for instance, directly cause harm. However, the major problem with GR1 is that it does not account for differences in preferences between agents, and how actions affect the preferences of agents in different ways. It is therefore susceptible to counter-examples: e.g., a judge should not send a criminal to prison, because the judge himself does not want to go to prison.[63] As a remedy, Reinikainen (2005) has argued that the universal applicability of GR needs to stand "the test of publicity", which means that an action needs to be "acceptable from the imagined perspective of everyone affected" (p. 155). This motivates a second formulation:

(GR2) *Golden Rule based on the preferences of others* Assuming that the acting agent has perfect knowledge of how actions affect the preferences of others, and every agent has the same amount of preferences, the worst-case time complexity of deciding whether any action is morally permissible is $O(npo)$, where o refers to the number of other agents.[64]

What becomes clear from comparing formulation GR1 and GR2 is not the relatively small increase in time complexity, but the increase in knowledge requirements for GR2 to work. Perfect knowledge about how actions affect the preferences of others might be difficult to attain, even in smaller groups of agents. However, what is missing from GR2 is that it fails to capture the recursive feature that is essential to most formulations of the Golden Rule. It is not that an agent should simply take others' preferences into account; the agent should consider *the way in which* they would like *others* to take their preferences into account. This may not even involve specific actions or preferences as such, but rather, that an agent should generally behave towards others in a manner which they would like others to generally behave towards them. This feature of GR has been explored by philosophers such as Stace (1937), Wattles (1996), and Singer (2002), whom all make the observation that objections against the GR only have force against specific applications by GR, but not if we take GR to refer to an agent's general behavior. Following Wattles (1996), this can involve one's own method of applying the golden rule:

(GR3) *Apply the golden rule to your general behavior in a way that you would want others to apply the golden rule to their general behavior.*

The formulation is self-correcting in the sense that it will consider the potentially infinite ways in which an agent A would like others to consider in their treatment of A; e.g., based on preferences, obligations, sensitivity to causal outcomes, integrity, or respect. It also reflects aspects of the moral rationalist project of determining universal a priori principles for morality (e.g., Kant's categorical imperative). Nevertheless, there are two main problems which distorts a complexity analysis of rules such as GR3. The first is that it may presuppose a great variety with regard to the cognitive capacities of agents, the ways

[62] Note that $O(np)$ is also the tight bound of finding the most satisfactory action (e.g., an action which maximizes preferences), which is analogous to C2 in Sect. 4.1.

[63] This counter-example was famously made by Kant (1785, footnote 12).

[64] For instance, if there are two possible actions, five preferences, and 8 billion agents, it would take 80 billion computations to decide whether any of the two actions satisfies the preferences of the world population in 2022.

such capacities enable features of an agent's general behavior, and in turn, the cognitive capacities and features of general behavior that are shared between agents. The categorical imperative, for instance, assumes an idealized form of autonomy, in the sense that agents act according to their own self-imposed rules (without any external influence). Similarly, as noted in our discussion of "the harm principle", the complexity of directing one's general behavior is proportional to the space of actions one can do. In the context of liberal democracies, it may refer to a cluster of capacities that are more or less shared among "generally competent" human adults (e.g., to act with intention, understanding, and without the controlling influence of external factors). In this context, a complexity analysis of G3 thus presupposes that we have a computational specification of an adult human being.

The second problem is game-theoretical: there may be great variance with regard to what one expects of other agents. Here, the conflict between contractarians (following Hobbes) and contractualists (following Rosseau) comes to play, as they differ in their understanding of "the original position" and its relation to morality. The former starts from the assumption that human nature is primarily driven by self-interests, which makes morality a problem of cooperation; it would only be rational to be moral (cooperate) if it increases the joint interest. The latter has a more optimistic view, in the sense that it starts from a basis of mutual respect: morality are the results of binding agreements from a standpoint that recognizes each rational autonomous agent's equal moral importance. This conflict may create a significant variance with regard to computational resources. For instance, if you are completely selfish, and you expect everyone else to be completely selfish as well, it will make your personal GR relatively simple to implement: you do not help others in need because others would not help you. By contrast, if you believe in every autonomous being's equal moral importance, you might help others, not in light of a joint-interest (e.g., direct or indirect reciprocity), but due to a reasoning process which led you to conclude that helping others is what free and equal citizens should do.

5.1.3 Moral behavior in strategic games

While it is hard to imagine a complexity analysis of general rules that is able to escape the extreme variances of general behavior and behavioral expectations, insights from game theory can shed light on *some* of its aspects. In 1950, mathematician John Nash famously proved that for every finite n-player game, there exists at least one fixed strategy profile—a Nash equilibrium (NE)—in the sense that no player can benefit from changing their strategy (Nash et al. 1950). In the seven decades that has followed, game theory and its many extensions have become a standard tool for mathematical modeling in biology, social science, and economics (Smith and Price 1973; Nowak 2006; Holt and Roth 2004). In philosophy, it has been instrumental in theories of social norms as mixed-motive games turned into coordination games (Ullmann-Margalit 2015; Bicchieri 2005), and to aid the contractarian project of deriving morality from self-interest (Gauthier 1987; Skyrms 2004).

From a game-theoretical standpoint, it is interesting to ask: what computational resources does an agent need in order to decide how to act? Using the Prisoner's Dilemma (Table 2) as the prototypical case, the most efficient approach would be to follow one of two pure strategies:

(S1) *Pure defection* Always do the selfish action, regardless of whether your own payoff (self-interest) would have been higher if you did not.

(S2) *Pure cooperation* Always do the altruistic action, regardless of your own potential loss.

Table 2 Canonical payoff matrix for the Prisoner's dilemma, where $T > R > P > S$

	Player 2	
	Cooperate	*Defect*
Player 1		
Cooperate	(R, R)	(S, T)
Defect	(T, S)	(P, P)

Assuming that an agent can, in every relevant circumstance, determine what it is to cooperate or defect, (S1) and (S2) are equally efficient, $O(1)$. In some sense, (S1) can be seen as a naive interpretation of behavior in Hobbes' state of nature (Hobbes 1651), whereas (S2) represents behavior in Kant's hypothetical Kingdom of Ends (Kant 1785). Since an agent's life in the former is "nasty, brutish, and short", it lacks the mechanisms that enable mutual flourishing.[65] In the latter, agents treat each-other as ends (as opposed to means), which allows them to prosper. Of course, agents who adopt (S1) will miss out on the game-theoretic rationality of morality itself, i.e., when the payoffs for mutual cooperation is larger than mutual defection. Similarly, agents adopting (S2) might perform extreme and seemingly unnecessary acts of self-sacrifice (in Kantian terms, violating obligations directed to oneself). Furthermore, in mixed populations, cooperative agents will be targeted by free-riders who exploit the good-will of others (Fehr and Fischbacher 2004), unless there are mechanisms for punishment (Fehr and Gächter 2000). Still, from a machine ethics perspective, (S2) deserves attention as it captures aspects of divine command theory and legal positivism; i.e., given that humans have gathered exhaustive moral knowledge of a certain domain (e.g., determined the actions that support human well-being) and are able to implement it as input-action-commands. For instance, if one adopts the view that machines are merely tools for human ends, extreme forms of machine-sacrifice or machine-exploitation may be irrelevant.[66]

While (S2) might have some appeal for the prospect of moral machines, it makes less sense from a human perspective. Simply put, people are less likely to cooperate if there are no self-directed incentives (even Kant's Kingdom of Ends assumes some form of obligations to oneself). The more common strategy, and the most extensively studied strategy in economy and behavioral ecology, is to be rational in the sense of maximizing self-interest:

(S3) *Mixed rationality* Do whatever maximizes self-interest.

The strategy is *mixed* as opposed to pure, in the sense that it can be represented as a probability assignment to each pure strategy. It is a more sophisticated version of (S1), since it takes the potential self-interested benefits of cooperation into account (Axelrod and Hamilton 1981). However, what constitutes a rational choice, following (S3), is ultimately dictated by features of the game. In a one-shot prisoner's dilemma (played only once), if both players know that they will never play again, the dominant strategy and only possible NE is to defect. No matter of what Player 2 player does, Player 1 will be better off defecting, and vice versa: since neither can retaliate against the other, none of them have anything to lose by defecting. By induction, the same holds for the iterated prisoner's dilemma, given that it is commonly known that the game is played precisely n times (Luce

[65] Based on his idea of the state of nature, Hobbes famously argues that people are better off submitting themselves to an absolute political authority, which has the power to protect people from themselves.

[66] From this point of view, it also seems absurd to conceive of moral machines driven by self-interest.

and Raiffa 1989). If the number of rounds are unknown or infinite, however, cooperation among rational players can emerge in non-cooperative situations; defection may still be a NE, but not a strictly dominating strategy (Aumann 2016). It might seem counter-intuitive that merely knowing how many rounds one will play should make such a big difference for the choice of action; nor does it reflect what humans do in experimental settings (Heuer and Orland 2019). Several solutions to this "paradox" have been proposed. Radner (1986) showed that relaxing the strict notion of rationality (e.g., being satisfied by a 'close enough' payoff) allows for longer periods of cooperation. Kreps et al. (1982) demonstrated similar results on the basis that agents have incomplete information about the options, motivations, or behaviors of other players. A more interesting solution from a complexity perspective was provided by Neyman (1985), who showed that cooperation becomes an equilibrium if the players have sufficiently small memories. More specifically, if agents are modeled as finite automata with a fixed size s, and play n number of games, mutual cooperation becomes a fixed point if $2 \leq s < n$.[67] Intuitively, since players do not have the memory needed to enumerate to n, they in effect treat is as an infinite game.

This generates an insight which is contrary to the complexity results discussed thus far. While the results in Sect. 4 indicate that an agent cannot do what is morally optimal due to their own computational constraints, these game-theoretic considerations demonstrate how agents behave morally (cooperate) due to a *lack* of certain computational resources: e.g., by restricting space complexity Neyman (1985), information Kreps et al. (1982), or rationality Radner (1986). This may suggest another role for normative theory which contradicts what was argued in Sect. 3.1.2: the purpose of NT is not to produce moral optimality as such, but rather to provide action-guidance in novel and complex situations. The results may be suboptimal, either from a rational or moral perspective, but they *do* work, given the constrained resources of agents. This idea has been extensively explored by Alexander (2007), who claims that moral principles have emerged to provide 'fast and frugal heuristics' which enables agents with bounded cognitive abilities to coordinate on suboptimal outcomes.

Nevertheless, it is also an optimistic form of question-begging: it already assumes that the moral norms and principles that foster cooperation leads to some non-trivial joint benefit (e.g., contractarian perspective), while the agents themselves might lack the cognitive abilities to verify the benefit. Consequentially, the joint benefit becomes its own optimistic fixed point unless it is compared to some other alternative (or the de facto optimal). Like divine command, legal positivism, or natural law, it might ask agents to blindly follow rules because "they are good", while offering no proof of why those rules are better than other alternatives. Of course, from an evolutionary perspective, it may provide informative post-hoc solutions to the problem of altruism: how is it that many organisms exhibit altruistic behaviors—increasing other agents' reproductive potential by reducing their own—given that natural selection favors the survival of the fittest? By observing cooperative behavior in various organisms, one might conclude that altruistic behaviors could not preserve unless they offered some alternative reproductive benefit [e.g., kin and selection, direct, indirect, and network reciprocity (Nowak 2006)]. In the same vein, one might justify "fast and intuitive" moral behaviors because they compress moral wisdom from evolutionary or cultural history. However, this view relies on an optimistic conservatism: that there

[67] See Aaronson (2013) for an interesting discussion of this paradox in the context of computational complexity.

are good reasons why things are as they are, even if we may not know these reason to a full extent (we will return to this issue in Sect. 5.3).[68]

5.1.4 Moral behavior in strategic games with incomplete information

Instead of finding computational constrains that enable cooperation, one can investigate the factors that hinders the maximization of rational self-interest in more realistic game-theoretic settings. For instance, through the work on Bayesian games by Harsanyi (1967), we can model scenarios with incomplete information:

(S4) *Bayesian rationality* Do whatever maximizes expected self-interest.

Bayesian games relax the underlying assumption in classic game theory: that features of the game, e.g., the actions and payoff functions for every player, are known by all players. For instance, in the original formulation of NE, it is assumed that each player knows the equilibrium strategies of the other players. In a Bayesian game, players instead have beliefs about the features of the game, which may involve beliefs about others' beliefs of features of the game. This naturally leads to infinite hierarchies of higher-order beliefs (beliefs about beliefs about beliefs ad infinitum), which are cumbersome to approach mathematically. Harsanyi's model of Bayesian games partly solves this through the notion of *type*, which summarizes a player's beliefs about the nature of the game (and her infinite hierarchy of beliefs).[69] If it is assumed that the private elements of players are drawn from a commonly known distribution, the infinite regress is resolved, which enables a Bayesian equilibrium analysis; a specification of the behavior of each player that is a best-response to what the player believes is the behavior of the other player (i.e., a best-response to the other players' strategies given the players own type).

In practice, however, Harsanyi's setting is unfeasible to model in multi-agent systems where agents interact with unknown agents. How can we be sure that a stranger draws from the same known probability pool as ourselves? In autonomous agents and multi-agent systems research, nested beliefs are instead investigated through the concept of recursive reasoning.[70] Methods for recursive reasoning typically approximate nested beliefs down to a predetermined recursion depth. For instance, if A is trying to predict the behavior of B, A predicts B's next action by simulating B's decision based on what A beliefs about B. This, in turn, requires a prediction of A's behavior from the perspective of B, based on what A thinks B believes about A, etc. Recursion is then terminated at a fixed depth by drawing the action prediction from some probability distribution (e.g., a uniform distribution). The bottom-level prediction at 0 is then passed up to the higher level (0 + 1), where the optimal action is chosen, and then passed recursively up to the highest level (l) where A makes it de facto choice.[71] One prominent version of the aforementioned process is the Interactive POMDP (I-POMDP), which extends a POMDP (discussed in Sect. 4.3) by adding models of other agents into the state space (Gmytrasiewicz and Doshi 2005). In short, for agent A to pick the optimal action, A has to solve the I-POMDP of B for each model of B, which

[68] As an interesting thought-experiment, if Thomas Hobbes were alive in the 21st century, would he still try to convince democratic societies to submit their power to a sovereignty?

[69] See Mertens and Zamir (1985) and Brandenburger and Dekel (1993) for two complementary constructions of Harsanyi's model of infinite hierarchies of beliefs.

[70] See section 4.5 in Albrecht and Stone (2018) for a recent survey on recursive reasoning methods.

[71] In fact, the famous minmax algorithm for zero-sum games is a special variation of this method, where the opponent B's evaluation function is taken to be the inverse of player A (Campbell and Marsland 1983).

involves solving the I-POMDP for each model B has of A, all the way down to the 0th level, where models of other agents are standard POMDPs (i.e., they are "noise" drawn from some probability distribution).

While several exact and approximate solutions for I-POMDPs have been offered,[72] they are, for obvious reasons, hard to compute. Since the 0th level constitutes POMDPs which the agent can recursively use to solve POMDPs for the higher levels, the complexity of solving an I-POMDP is equal to solving $O(N^l)$ POMPDs, where N is a bound on the number of models the agent considers at each level, and l is the recursion depth (Gmytrasiewicz and Doshi 2005). For instance, for an I-POMDP containing 4 agent models and 4 levels, this will amount to solving 256 POMDPs, which, individually, are PSPACE-complete for finite time horizons (Papadimitriou and Tsitsiklis 1987). More broadly, Bernstein et al. (2002) studied a number of *decentralized* control problems—where multiple agents cooperate to control a process, each with possibly different information about the state—and proved that both decentralized MDPs and POMDPs are NEXP-hard; i.e., at least as hard as the hardest problems that are solvable in non-deterministic exponential time. More specifically, while POMDPs and I-POMDPs are equally targeted by the "curse of history"—in the sense that the space of policies is proportional to the number of possible future beliefs given by the time horizon—since I-POMDPs may involve a greater number of (potentially nested) beliefs, they are further impeded by the "curse of dimensionality", as the complexity of belief representation is proportional to the dimensions of the belief structure.

But history and dimensionality are not the only curses in strategic interaction. For instance, if players are uncertain about other players' payoff function, there is an inherent tension between prediction and rationality. Foster and Young (2001) demonstrated that there are situations in which it is impossible for rational players to play optimally with respect to their beliefs, while simultaneously having *correct* beliefs. The reasoning is simple: if A tries to predict the action of B at t_2, and A must take an action at t_1 which B can observe, B's observation might invalidate A's prediction of B's behavior at t_2. Furthermore, in certain settings, higher-order reasoning may not provide any additional benefits. In fact, the studies by de Weerd et al. (2013, 2017) demonstrates settings—e.g., sequential negotiation and rock-paper-scissors—where reasoning levels higher than 2 do not offer any notable advantages for computational agents.

Nevertheless, while the intractability results for recursive reasoning have a direct impact for computational systems, it remains an open question how humans reason in similar problems, and consequently, how machines should interact with humans in a robust way. These questions have been extensively explored in the experimental psychology on strategic interpersonal situations, often in combination with the notion of "theory of mind" (Yoshida et al. 2008). For instance, in two experiments with human participants, Hedden and Zhang (2002) showed that participants employ a short-sighted "default model" about the other players minds, which were dynamically adjusted in light of new evidence. In addition, the "cognitive hierarchy" model proposed by Camerer et al. (2004) suggests that players presume that their own strategy is the most sophisticated, in the sense that they use the best-response at recursion level l to predict the behavior of players at level $l-1$ (i.e., one step 'higher' in the level of nested beliefs). By fitting their model with a large corpus

[72] E.g., using methods like Monte Carlo sampling (Doshi and Gmytrasiewicz 2009) or model equivalence (Rathnasabapathy et al. 2006).

of empirical data from a variety of games, they found that humans, on average, reason at recursion depth 1.5.[73]

Given the many conflicting dimensions of the subject matter, it seems difficult to arrive at any general conclusions about whether and to what extent computational constrains affect strategic interactions in moral contexts. However, it tentatively indicates a trend, namely that optimality is traded for efficiency (or mere feasibility) in light of information availability (about the state space and of other players), bounded rationality (e.g., memory and recursion level), and game setting (e.g., one-shot or iterated, stochastic or deterministic). Intuitively, the complexity results could support the appeal of pure strategies such as S1 and S2: in complicated situations, it is easier to simply believe that everyone is of a certain type (e.g., selfish or altruistic). It could also explain why it is easier to cooperate "locally" (e.g., in smaller groups of friends), where features of the games are common knowledge; a mutually shared goal (e.g., maximization of joint interest), and shared mechanisms for detecting and punishing free-loading.[74] Similarly, it also shows the inverse: why it is difficult to achieve cooperation in bigger populations, where game features are not shared, and agents do not know what to expect from each-other.

5.1.5 Computing moral equilibria

Perhaps the most interesting part of game theory from a moral computational perspective is not to ask about the resources an agent needs in order to decide how to act, but rather, to find strategies that maximize the interest of everyone (e.g., given that everyone were to follow the same strategy). This is interesting for two reasons. First, from a moral point of view, it would roughly correspond to the general-purpose rules discussed in Sect. 5.1.2, e.g., GR and CI. That is, if some action-rule (or strategy profile) is a consistent best-response to every other action *and* leads to some non-trivial mutual benefits, it would be attractive for a moral community to find those rules. Second, these general-purpose rules would in turn correspond to the typical solution concepts used in game theory—namely Nash equilibria—with properties that are attractive from a moral standpoint (e.g., maximizing joint interest).

It is thus natural to ask: how difficult is it to compute a Nash equilibrium? Already v. Neumann (1928) provided the Minmax Theorem, which entails that equilibrium in 2-player zero-sum games can be computed in polynomial time by linear programming (Khachiyan 1979). For non-zero sum games of at least two players, however, the problem is proven to be PPAD-complete (Chen et al. 2009). PPAD, introduced by Papadimitriou (1994), is the class of function problems solvable by a non-deterministic TM in polynomial time where a solution is guaranteed to exist on the basis of the parity argument on directed graphs ("PPAD"). The parity argument is based on the graph-theoretical insight that the number of nodes that touch an odd number of edges is even in all finite undirected graphs (this defines the PPA class, which contains PPAD). A similar insight holds for directed graphs: given a directed graph and a source (a node without predecessors), there must be

[73] However, the view that human recursive reasoning is "pessimistic"—i.e., at a relatively low level of recursion, and based on underestimating one's opponent—has been contested by Goodie et al. (2012), who found settings where all participants engaged in the highest available level of reasoning in both competitive and simple settings.

[74] In turn, cooperative groups may benefit from the "wisdom of the crowd" phenomenon (Yi et al. 2012), where aggregations of multiple solutions performs better than individual solutions.

a node at "the end of the line" which lacks successors.[75] What is special about the PPAD complexity class is that it reflects what is special about NE: since there always exist a NE (Nash 1951), the answer to the decision problem "does there exist a NE for this game?" is always *yes*, and therefore, it cannot be NP-hard. Similarly, it also reflects what is special about Brouwer's fixed-point theorem: for any continuous function f that maps a compact and convex Euclidean space to itself, there exists at least one point x_* such that $f(x_*) = x_*$. By analogy: if you are standing in a region and unfold a map of the same region, assuming that there are no holes in the map, at least one point of the map will correspond to your location. But simply knowing that such a point exists, does not necessarily make it easy to find it. If we assume that one can efficiently check whether a certain point on the map is in fact one's location, the localization problem can be solved in non-deterministic polynomial time (NP membership). By the same reasoning, if one can efficiently verify whether a strategy profile is a best-response or not, the problem of finding a NE is in NP (which, in turn, contains the PPAD class).

However, if the NE should have any special properties—e.g., such that it maximizes the sum of the player's utility, or everyone obtains an expected payoff of at least some number—the problem becomes NP-complete (Gilboa and Zemel 1989).[76] Conitzer and Sandholm (2008) strengthened these results and proved that the egalitarian optimization problem (e.g., maximizing mutual payoff) is inapproximable; i.e., it is impossible to find an NE that approximates the maximum joint payoff in polynomial time. In fact, optimal NE is one of many known NP-hard problems related to NE that are also hard to approximate (Austrin et al. 2013): e.g., computing whether there is more than one NE, or finding NE with minimal support.[77] More precisely, while a linear-time algorithm can obtain 1/2-approximations of the optimal NE (Daskalakis et al. 2006), achieving approximations better than 1/2 is as hard as the planted clique problem, which in turn may be solved in quasi-polynomial time (Lipton et al. 2003).[78]

Naturally, in situations with incomplete information or stochastic dynamics, things may get even harder. While it remains PPAD-hard to find mixed-strategy equilibrium in Bayesian games due to the fact that normal-form games are special cases of Bayesian games, Conitzer and Sandholm (2008) showed that, even in symmetric 2-player Bayesian games, it is NP-hard to determine whether the game has a pure-strategy Bayesian equilibrium. By contrast, one can determine the existence of pure-strategy NE in normal-form games in polynomial time by simply checking whether any combination of pure strategies is a NE. However, this procedure is unpractical in Bayesian games, since the space of strategies grows exponentially with the number of types, which contains the private information of players preferences. In addition, Conitzer and Sandholm (2008) also demonstrated PSPACE-hardness for checking whether there exists pure-strategy NE in repeated games with probabilistic state transitions—also called "Markov games", as they extend MDPs to include multiple decision-makers—and that the problem remains NP-hard in finite games.

[75] The "end-of-the-line" problem is a paradigmatic PPAD-complete problem.

[76] This can be proved by reducing it to the problem of finding a clique of size k in an undirected graph (Gilboa and Zemel 1989). For instance, finding a clique of size 3 means finding 3 nodes that all are connected to each-other.

[77] The minimal support problem asks: for a number $k \leq 1$, is there a NE in which players use at most k strategies with a positive probability?

[78] Here, the planted clique problem is used as a hardness assumption, as it is conjectured that no polynomial time algorithm can, better than chance, distinguish planted cliques from random graphs (Hazan and Krauthgamer 2011).

Although NE constitutes the most extensively studied solution concept in game theory, it is not the only one that is useful from a moral perspective. A prominent alternative is the *correlated equilibrium* (CE) introduced by Aumann (1974, 1987). In simple terms, CE can be seen as a result of a commonly shared Bayesian rationality: it is simply a maximization of utility given the player's information. In contrast to NE rationality, CE does not assume that players know that other players play their action as it is dictated by the NE, nor that each player know the strategies of others. Instead, an equilibrium is simply that no player, based on what they know (privately or commonly), can expect a higher return if they deviate from their strategy. A useful analogy is to imagine a situation where players choose an action by observing a random event: in mixed strategy settings, the event is assumed to be independent for each player, while in correlated settings, they may not be. For instance, player *A* and *B* might privately observe a correlated signal—e.g., two traffic lights—which recommends A to wait and B to go. If the signals are drawn from a correlated distribution, neither *A* nor *B* would want to violate the signal's recommendations (as running a red light might cause them to collide). As such, CE is a general distribution over strategy profiles, whereas mixed strategy NE is a distribution over the space of "uncorrelated" strategies (independently distributed over each player). Since every mixed NE can be defined as the product of the player's mixed strategies, it follows that NE is in fact a special case of CE; and that every game has a correlated equilibrium.

Apart from being guaranteed to always exist, CE is attractive for multiple reasons. It seems to emerge in natural settings where NE does not (Hart and Mas-Colell 2000).[79] Unlike NE, it is more apt to accommodate the role of external factors that are decisive for the outcome, e.g., by following the recommendations of trusted sources. From a moral perspective, CE captures many situations—e.g., in moral or legal rule-following—where it is clearly incentivized to not deviate from the prescribed action (e.g., following traffic rules). Last but not least, CE are easier to compute than NE. Since correlated equilibria can be defined by a set of linear inequalities (Hart and Schmeidler 1989)—and does not have to be based on Nash's result—the problem of finding an CE can be solved by linear programming, and is therefore computable in polynomial time for games with any number of players (Gilboa and Zemel 1989). Papadimitriou and Roughgarden (2008) demonstrated that there are polynomial-time algorithms for computing an arbitrary CE for many natural classes of succinctly representable multiplayer games, including polymatrix games, graphical and hypergraphical games, and scheduling games. Unfortunately, while *any* CE seems easy, Papadimitriou and Roughgarden (2008) also show that in nearly every class of succinct games, it remains NP-hard to compute a CE that maximizes the expected joint payoffs.

Besides these intractability results, one might also question the relevance of computing morally good equilibrium. Of course, at system-level, they provide indispensable analytical tools for modeling and measuring intricate behavioral dynamics. For instance, the *price of anarchy* measures how the joint welfare of a system degrades due to selfishness by dividing the welfare value for the "worst-case" decentralized equilibrium with the welfare at its optimal centralized configuration (Koutsoupias and Papadimitriou 2009).[80] But these tools might be of less use from the perspective of individual agents. We can imagine that an agent has managed to compute a "golden rule" strategy profile (e.g., a set of action-rules)

[79] See also chapter 7 in Cesa-Bianchi and Lugosi (2006).

[80] Conversely, the *price of stability* gives a ratio of the difference between "best possible" decentralized equilibrium and the optimal centralized solution.

which for any possible interaction, maximizes the joint benefit of her society. Still, her society would only flourish given that the *other* agents in the society followed the strategy profiles as fixated by her equilibrium computation. She might try to convince other agents to follow her lead by an appeal to her extraordinary computational powers[81]; she might, as Kant, argue that it is imperative for the will of rational beings; that it is, following Hobbes, imperative for the maximization of expected self-interested; or she might simply hope that everyone else—e.g., in virtue of shared rational capacities and correlated distributions—computes the corresponding set of strategy profiles. Given the adequate means, she might enforce the fixation of the equilibrium, e.g., by punishing everyone who did not follow their strategy profile. At the most extreme, we could imagine a super-intelligent Leviathan that has—even adjusting for any potential welfare losses due to the restrictions of freedom and the punishment of dissidents—computed that the joint welfare would still be optimal if strategy profiles were enforced by force.

The main point is that solutions to decentralized cooperation problems require, in some non-trivial sense, centralized features to work in practice. While shared cognitive capacities, rationality, communication (Crawford and Sobel 1982), and information (e.g., from correlated distributions), can certainly make it easier for agent's to collectively arrive at morally attractive equilibria, the complexity of cooperation is distorted by a vast range of features in the local and agent-specific context; e.g., more or less overt power-dynamics (e.g., via legal, political, and religious institutions), psychological heuristics (such as trust, shame, and guilt), and conflicts between in-group/out-group preferences. Additionally, following results from social choice theory, there may be situations in which it is impossible to translate individual preferences into community-preferences without violating some significant fairness criteria.[82]

Furthermore, in the real-world, it is often not clear what kind of game we are playing, or, whether we are even playing a game at all. Games can be cooperative or non-cooperative, discrete or continuous, one-shot or repeated (e.g., played with strangers or friends), simultaneous or sequential, zero or non-zero sum, symmetric or asymmetric, have varying degrees of imperfect and incomplete information, have varying population sizes, all while allowing for a potentially infinite set of different strategies (e.g., specific action-combinations in sequential Markov processes of unknown length). In other words, even if we assume "centralized features" (such as shared cognition, memory, and views on rationality), game-theoretic models—and thus, the applicability of equilibria solutions—are heavily underdetermined by data. While small and isolated groups might tend towards cooperation—e.g., by easily recognizing the maximization of a joint interest; having mechanisms for detecting free-loading; knowing that others share the same information about the game; etc—it might take millennia for larger societies to become egalitarian, e.g., as agents

[81] Following a recent proposal from Cummings et al. (2016), *coordination complexity* can be measured as the minimum information a centralized coordinator with complete knowledge of the game needs to publicly signal in order to coordinate players towards a nearly optimal solution.

[82] A famous example is Arrow's impossibility theorem, which states that in a ranked voting with at least three options, there is no electoral system that can produce a unique and complete ranking while simultaneously satisfying unanimity (i.e., if every voter prefers A over B, then the community ranking prefers A over B), non-dictatorship (i.e., no single voter can alone determine the community's preference), and independence of irrelevant alternatives (i.e., the preference for choosing A over B do not change if C is added to the alternatives) (Arrow 1950).

cannot agree on the nature of the coordination games they are playing (for instance, due to conflicts between in-group preferences).[83]

5.1.6 Algorithmic moral mechanism design

The strong indeterminacy of game-theoretic models, along with the hardness of computing good equilibria in such models, suggests that it may be easier to foster collective welfare *by design*; i.e., constructing systems where tractable and morally praiseworthy equilibria naturally arise (as opposed to finding good equilibria in arbitrary settings). This intuitively reflects the relationship between legality and morality. Although agents may follow rules due to moral reasons (e.g., they lead to a joint benefit if everyone followed them), they may be further incentivized to do so if there are mechanisms for blame, responsibility, and punishment. Similarly, institutionalized moral rules (e.g., "do not harm others"), may by design reflect natural strategy profiles that are easy to understand, morally justify, and apply in practice. Similarly, in economics, the aim of *mechanism design* (or "reverse game theory") is to design decentralized economic mechanisms that achieve desired objectives, e.g., to optimize social welfare (Hurwicz and Reiter 2006). In other words: design or alter the mechanisms of our interactions in a way that promotes moral values (e.g., mutually beneficial outcomes). A simple example is the "I cut, you choose"-procedure for fair division (e.g., when two parties share a cake), which guarantees that both players receive a payoff that is *at least* as valuable as the other payoff, regardless of what the other player does. Another example is democracy, where citizens are incentivized to participate in political decisions that affect their own lives. In algorithmic game theory, this has further motivated the field of *algorithmic mechanism design*, which seeks to design games that combine features that are attractive from a computational and game-theoretical perspective, with the prime example being games with good worst-case equilibria (i.e., a low price of anarchy) that can be computed in polynomial time (Nisan and Ronen 1999). Some notable examples of algorithmic mechanism design include traffic routing (Roughgarden 2005; Roughgarden and Tardos 2002), auction design (Cai and Papadimitriou 2014), and internet problems (Feigenbaum and Shenker 2004). Essentially, since it is difficult for agents who are constrained by tractability to find and follow moral equilibria in open-ended environments, complexity considerations can help to inform the design of decentralized systems that are morally attractive.

However, algorithmic mechanism design is not without its grand challenges. One challenge is to design and implement systems in a way that ensures that the mechanisms—e.g., behavioral rules—and desired outcomes are widely endorsed among those who participate in the system. Thus, while the design process in ideal circumstances could itself be participatory (Kensing and Blomberg 1998), implementing a system may in principle be as difficult as establishing a particular form of political governance or moral theory. A related challenge is to work out the uncomfortable trade-offs between different values that the mechanism might introduce: e.g., between freedom and welfare.[84] And similar to politics,

[83] To be clear with terminology, coordination games conventionally refer to games where it is optimal for players to cooperate, e.g., as in Stag Hunt or Battle of the Sexes. As such, they differ from anti-coordination games like Hawk-Dove (chicken), where it is optimal for players to play different strategies.

[84] Considering the Leviathan example discussed in Sect. 5.1.5, a centralized solution that optimizes some joint welfare may be inconsistent with freedom, as it comes at the cost of forcing agents to act in certain ways.

a reoccurring threat is that those in power to design and implement a particular mechanism may opt for one that only promotes their own interests, or further cements their own power over the political process.

In summary, our investigation into the generality of moral rules (Sect. 5.1) found that:

(1) Although moral rule-following in human contexts might appear to be computationally simple, they rely on shared cognitive capacities and agreements between "generally competent" human agents (often along with the institutions that ensure their just application). While rules such as "do not *X*" might reduce space complexity, it is only by restricting a potentially infinite space of actions that autonomous agents can do.

(2) If computational agents should follow general moral rules in a way that ensures their own normative justification—e.g., being sensitive to other beings via rational agreement or by maximizing joint interest—results from algorithmic game theory indicates that it requires solutions to intractable problems (with a few noteworthy exceptions). Note that these intractability results hold for salient games, in the sense that agents are assumed to have a model of the game they are (supposedly) playing, even if information about aspects of the game may be incomplete. In many real-world situations, the validity of such models, given that they are obtainable at all, might still be severely underdetermined by data (i.e., the model might fail to capture the essential real-world dynamics). In other words, if we assume that moral rule-following *can* be mathematically modeled as following strategies in games, the solution to such games may be decidable (e.g., by computing equilibria), although in general not polynomial-time tractable. It is also important to stress that these challenges are far from unique to deontology, but affects any moral theory with generality ambitions that seeks to account for multi-agent dynamics. Thus, it naturally targets consequentialism (as it centers on how outcomes affect others), but not theories such as divine command and legal positivism.

(3) Given the hardness of finding good equilibrium, the problem of computing the beliefs of others' (e.g., nested beliefs), the indeterminacy of game models, and the fact that cooperation may emerge due to a lack of certain computational resources (e.g., information, memory, or self-interest), the investigation also suggests that moral rules and principles might better serve as 'fast and frugal heuristics' that guide agents with bounded cognition in open-ended environments towards suboptimal but feasible results. Somewhat ironically, this might in turn make "naive" moral strategies (e.g., "when in doubt, be altruistic") viable or even necessary from a computational perspective; although without offering any game-theoretic explanation as to *why* they are viable from a self-interested perspective.[85] Furthermore, instead of computing moral equilibria in complex environments, one promising alternative approach is to use complexity constraints and equilibrium measures to guide the design of decentralized systems which are attractive from a moral perspective.

[85] To illustrate this point, we can imagine two populations—A and B—that have converged on the same set of cooperative behaviors that yield optimal joint welfare. In population A, the equilibrium was a result of repeated interactions between self-interested individuals over several thousands of iterations. In population B, the equilibrium was already fixed at start, as agents were told by their mothers to "always be kind". The point is that while agents in A understand the self-interested rationale behind cooperation, they are nevertheless equally well-off as agents in B, who are oblivious to the same rationale.

5.2 The complexity of moral logic and semantics

Another way to analyze the computational complexity of deontology is to look at the syntax and semantics of formal languages that aim to capture moral reasoning and rule-following. For instance, we might note that moral rules expressed in natural language have certain logical characteristics, such as "given fact *a*, action *b* is obligatory". Consequently, we might imagine that it is possible to construct a machine that has a number of norms (x, y) stored in memory which relate facts (x) to obligations or permissions (y). The question is, are there any complexity considerations that might curtail such a machine's practical success for reasoning about norms (this will be the topic of Sect. 5.2.2)? More broadly, how does the expressive power of a logic used for moral reasoning relate to complexity classes, e.g., in terms of the problems it can describe (the topic of Sect. 5.2.1)? In this section, we will discuss such problems along with some more fundamental issues in semantics (Sect. 5.2.3).

5.2.1 Decidability and descriptive complexity

Any discussion of the complexity of logic would be incomplete without reiterating the classic results that make up the very foundations of computability. In 1929, Kurt Gödel's completeness theorem establishes that in first order logic (FOL), semantic truth corresponds with syntactic provability; i.e., there are complete, sound, and effective deductive systems for FOL (Gödel 1930). Two years later, Gödel gave his two celebrated incompleteness theorems (Gödel 1931), which shows that any consistent formal system capable of carrying out elementary arithmetic—e.g., using natural numbers, addition, and multiplication—is incomplete (first theorem),[86] and that such a system cannot prove its own consistency (second theorem).[87] Inspired by Gödel's theorems, Turing (1936) and Church (1936) independently provided negative answers to the "Decision problem" (Entscheidungsproblem) for FOL; Church's proof utilizes the undecidability of checking the equivalence of two expressions in the λ-calculus, whereas Turing constructs the halting problem for Turing machines. Their results establish that no decision procedure exists that can decide whether arbitrary FOL formulas are logically valid.[88] On the other hand, propositional logic (PL)—which, unlike FOL, excludes relations and quantifiers—is decidable. For instance, the satisfiability problem for propositional logic (SAT) was the first decision problem proven to be NP-complete (Cook 1971), and has since remained at the center of computational complexity theory. Using Cook's results from SAT, Karp (1972) proved NP-completeness of 21 further problems, which not only demonstrated the intuitive appeal of the NP-class, but also that many natural computational problems are intractable.

However, instead of determining the resources needed to check whether some input satisfies some property X, we can ask, what is the complexity of *expressing* X? The latter question is central for the field of *descriptive complexity theory*, which defines complexity classes in terms of the type of logic required to express the languages in them.

[86] A system—or rather, a *theory* in the mathematical logical sense, i.e., a set of sentences in a formal language—is *consistent* if it does not lead to contradictions. An axiomatic system is *complete* if any statement in the systems language is provable from the axioms. Incompleteness thus entails that there are statements which cannot be proved nor disproved in the system.

[87] The theorems were further refined by Rosser (1936), who proved them without assuming ω-consistency.

[88] Validity means that true premises guarantee the truth of an argument's conclusion.

In fact, due to a wealth of results from descriptive complexity, it turns out that expressing and checking are intimately related.[89] The first major result in the field was Fagin's theorem (Fagin 1974), which established that NP is exactly the set of properties that can be expressed in existential second-order logic (SO∃), which unlike FOL, has the power to existentially quantify over properties and relations. In other words, every query that is computable in NP (including NP-complete problems) is equivalent to a query in SO∃.[90] Furthermore, since the complement of an existential formula (quantifying over *some* members of a domain) is a universal formula (quantifying over *all* members of a domain),[91] it follows directly that co-NP is captured by SO∀, and unrestricted second-order logic (SOL), which allows for both universal and existential quantification, is equal to the union of all classes in the polynomial hierarchy (PH). Other notable results include the fact that FOL corresponds to the logarithmic time hierarchy (LH) as well as the circuit complexity class AC^0, linearly ordered FOL systems with a least fixed-point operator yields P (Immerman 1982; Vardi 1982),[92] SOL with a transitive closure gives PSPACE (Immerman 1989), and SOL with a least fixed-point corresponds to EXPTIME (Immerman 1998; Abiteboul et al. 1997).

These results generate the wisdom that the expressiveness of a language is directly correlated to the problems it can describe. However, it should be noted that, while computational and descriptive complexity are intimately related, they also have some crucial differences. For instance, one key incongruity is that, whereas descriptive complexity studies finite mathematical structures, computational systems operate on ordered encodings of problems and are thus able to enumerate objects which may be logically unordered. As an example, from a logical perspective, we might see a set of nodes in an graph as undordered, but as soon as it is transferred to the tape of a Turing Machine, it inevitably becomes ordered (and thus exploitable for various forms of operations).[93] Nevertheless, the fact that complexity can be characterized in terms of expressibility—without reference to some abstract machine—further establishes the natural appeal of the complexity classes. And for computational moral rule-followers—e.g., systems performing queries over databases containing moral norms—the consequences are profound yet somewhat clouded by its mere generality. For instance, languages describable in FOL corresponds to AC^0 (polynomial-size circuits of bounded depth), which allows one to perform integer addition and subtraction but not multiplication. Adding an operator which can compute the transitive closures of binary relations, on the other hand, makes it possible to produce structures that can answer reachability queries (e.g., is it possible to go from node A to node Z in n steps?). Perhaps most interesting is the intractability of expressing queries which involve different forms of quantification over properties and relations (and not just objects, as in FOL), which is seemingly intuitive in natural language. For instance, even trivial moral queries of the type "For all observable facts, possible actions, and obligations, is there some action which does not violate any moral obligations?", might, in the worst-case, only

[89] See Immerman (1998) for the definite introduction to descriptive complexity.

[90] See chapter 7.1. in Immerman (1998) for a detailed proof.

[91] This follows from the fact that $\exists x \neg P(x)$ is equavalent to $\neg \forall x P(x)$.

[92] In short, fixed-point logics extend FOL with an operator which can construct fixed points of relational variables. For instance, if we view formulas with free relational variables as if they are determining maps on the relation space, the operator can define fixed points on this map.

[93] However, this incongruity does not hold for Fagin's theorem, since SO∃ can be used to declare the existence of some desired order.

be expressible in intractable complexity classes (e.g., if we assume that obligations are a relation between facts and actions). The point is, while quantification over properties and relations, transitive closure, and fixed points offer substantial expressive powers to well-studied collections of formal systems, it also raises intractability concerns; simply because language helps us to succinctly represent and communicate queries for moral rule-followers, it does not necessarily make such queries easy to compute.[94]

5.2.2 The complexity of modal and deontic logic

One of the most widely used fragments of FOL is modal logic; the go-to logic for representing necessity ($\Box$) and possibility ($\Diamond$).[95] From the mid 20^{th}-century and onward, the standard semantics for modal logic is the possible world approach,[96] where $\Box P$ means that P is true in *all* possible worlds, and $\Diamond P$ means that P is true in *some* possible worlds (assuming that these worlds are accessible). One attractive feature of modal logic is that it, contrary to FOL, is robustly decidable.[97] However, although it is decidable, typical problems for modal logic are in general not tractable. For instance, while the *model-checking* problem, which asks whether a given formula is true with regards to a given state of a given Kripke structure[98] is solvable in linear time, the *validity* problem, which asks whether a formula is true in *all* states of *all* Kripke structures is PSPACE-complete (Ladner 1977). More precisely, Ladner (1977) demonstrates that the validity problem for modal logic is PSPACE-complete for the systems K, T, and S4, whereas it is NP-complete for S5.[99] These results where extended by Halpern and Moses (1992), who proved that the validity problem is PSPACE-complete for multi-agent versions of K, T, S4, and S5 (i.e., a "join" of the logics used by at least two agents). The work of Halpern and Moses (1992) also shows that, while the addition of a distributed knowledge operator—allowing an "all-knowing" agent to combine the knowledge of everyone else—does not alter the complexity, the addition of a common knowledge operator—allowing everyone to know P, and that everyone knows that everyone knows ...that P holds—makes the problem EXPTIME-complete.[100]

Naturally, similar intractability concerns plague other popular versions of modal logic. For instance, (1993) showed that the PSPACE-completeness of the validity problem carries over to the tense case (temporal logic), i.e., with the addition of operators expressing "it will always be the case that..." and "it always was the case that...". Similarly, Sistla and Clarke (1985) demonstrated that satisfiability for Linear Temporal Logic (LTL,

[94] See Aaronson (2013) for an interesting discussion of the related problem of "logical omnicience", which uses complexity considerations to challenge the view that if an agent knows certain facts, it also know every logical consequence of those facts.

[95] Alternatively, since the main ideas of modal logic long predates FOL—e.g., Aristotle's modal syllogisms—it is arguably more accurate to describe modal logic as an expansion of propositional logic.

[96] The approach was originally suggested by Carnap (1947), and later developed to its modern day form by Kripke (1963).

[97] For an in-depth exposition see Vardi's report "Why is modal logic so robustly decidable?" (Vardi 1997).

[98] A Kripke structure is a graph that represents reachable states as nodes, state transitions as edges, and a labelling function that keeps track of the properties that hold in each state.

[99] For readers unfamiliar with modal logic, K, T, S4, and S5 refer to the choice of axioms and rules that are added to the systems; e.g., K—the weakest version—only includes the necessitation rule (i.e., $(\vDash P) \Longrightarrow (\vDash \Box P)$) and the distribution axiom ($\Box(P \rightarrow Q) \rightarrow (\Box P \rightarrow \Box Q)$); T includes the reflexivity axiom ($\Box P \rightarrow P$) in addition to K; S4 and S5 includes T along with iteration axioms 4 ($\Box P \rightarrow \Box\Box P$) and 5 ($\Diamond P \rightarrow \Box\Diamond P$), respectively.

[100] See also Spaan (2016) for an exhaustive treatment of the complexity of modal logics.

introduced by Pnueli (1977), with operators for "next" and "until" (excluding the past), is either PSPACE-complete or NP-complete depending on the operators used.[101] In general, PSPACE-completeness also holds for the model-checking problem for several versions of LTL (Schnoebelen 2002).

Perhaps more interesting for the prospect of moral machines is the complexity results for variants of modal logics such as dynamic logic (DL, introduced by Pratt 1976) and deontic logic. DL adds the additional modal operators $[a]$ and $\langle a \rangle$, which makes it able to capture properties of program behavior; e.g., $[a]P$ means that after performing action a, it is necessary that P is true (i.e., a brings about P), and $\langle a \rangle P$ means that it is possible that P holds after a is performed (i.e., a might bring about P). Note that a might refer to an entire program, which allows dynamic logic to formalize dynamics—e.g., transitions, sequences, and results—of complex algorithmic systems of multiple programs. In turn, Propositional Dynamic Logic (PDL, introduced by Fischer and Ladner 1979) was developed to describe correctness, termination, and equivalence of computer programs on the basis of PL [instead of FOL, which was the basis of the first version of DL (Pratt 1976)]. Interestingly, the decidability of checking whether a formula F of PDL is satisfiable can be secured by having two sub-procedures running in parallel: one which enumerates all deducible formulas (R1), another which enumerates the finite models of PDL and tests whether they satisfy the formulas (R2). In this way, if F is satisfiable, a model which satisfy F must eventually be found. If F is satisfiable, R2 will at some point answer "yes", if not, R1 will at some point answer "no" (and the procedure halts when either sub-process gives an answer). Nevertheless, although PDL is attractive for formal verification of program behavior, it is less attractive for computational moral agents bounded by polynomial time, as SAT for PDL is EXPTIME-complete (Fischer and Ladner 1979; Pratt 1980).[102]

Deontic logic (introduced by Von Wright 1951), on the other hand, aims to capture the logical features of moral concepts such as permissions (typically denoted by operator P) and obligations (typically denoted by O). For instance, the axiom $O(A \rightarrow B) \rightarrow (OA \rightarrow OB)$ states that "if it is obligatory that A implies B, then B is obligatory if A is obligatory". Likewise, Kant's "ought implies can" can be expressed by $OA \rightarrow \Diamond A$. It should thus be no surprise that deontic logic has been a popular framework for implementations in machine ethics, e.g., for automatized ethical reasoning (Arkoudas et al. 2005; Wiegel and van den Berg 2009; Furbach et al. 2014; Malle et al. 2017b). However, compared to other versions of modal logic, the computational complexity of deontic logic is relatively unexplored. One reason is that the inherent complexity of (and relationship between) normative concepts—e.g., of agency, right, responsibility, and commitment—may be drawn to arbitrary levels of detail[103] Another reason is the problem of agreeing on the appropriate semantics for norms, e.g., whether they are based on possible worlds, axiomatic constructions, or operational executions. A more fundamental challenge for the semantics of deontic logic is captured in "Jørgensen's dilemma" (Jørgensen 1937), which asks whether arguments that contain norms (e.g., imperatives) express a truth or not. In standard conceptions of logical entailment, it is essential that true conclusions follow from true premises; i.e., conclusions and

[101] See Vollmer et al. (2009) for a more detailed study of the complexity of temporal and propositional operators in LTL.

[102] A more practical algorithm for PDL-SAT—although still EXPTIME-complete in the worst-case—has been offered by De Giacomo and Massacci (2000).

[103] See Sergot (1998) for an exposition, which also appears in the handbook on deontic logic by Gabbay et al. (2013).

premises can be true or false. However, since imperatives are—supposedly—neither true or false, they cannot play any meaningful role in the validity of arguments, and as a results, we cannot justify imperatives on the basis of logical reasoning. At the same time, reasoning with imperatives seems to be logically valid in cases where the premises and conclusions are imperatives (Jørgensen 1937, p. 290):

(P1) Love your neighbor as yourself
(P2) Love yourself
(C) Love your neighbour

Thus, the dilemma entails that we either have to accept that normative statements cannot have truth values, or deny that the premises and conclusions of our argument have truth-values. In turn, these considerations have given rise to several families of deontic logics, which can be roughly divided into two camps: the possible-world approach—e.g., standard deontic logic, "Seeing To It That" (STIT) logic (Horty 2001) and dynamic deontic logic (Meyer et al. 1988; Van Der Meyden 1996)—and approaches that are not based on possible-worlds, such as input/output logic (Makinson and Van Der Torre 2000), imperative logic (Hansen 2008), defeasible deontic logic (Governatori et al. 2013), and prioritized default logic (Horty 2012).

With regards to complexity, the first path entails that one has to deal with the PSPACE-completeness that often plague possible-worlds semantics (Halpern and Moses 1992), or the EXPTIME-completeness of dynamic logic (Fischer and Ladner 1979). In fact, it is shown that the satisfiability problem for a fragment of STIT, which is used to construct deontic STIT, is undecidable (Schwarzentruber and Semmling 2014).[104] As an alternative to possible worlds, Sun and Robaldo (2017) has investigated the complexity of input/output logic on the basis of a 'norm-based' operational semantics. In this view, declarative statements can be true or false whereas norms cannot: they are simply violated or complied. More precisely, norms are ordered pairs (x, y) that corresponds to deductive operations of the input/output-system, taking a fact (x) as input, and producing an obligation (y) as output. However, their investigation finds that decision problems of input/output logic are shown to be NP/co-NP-hard and in the 2^{nd} level of the polynomial hierarchy (i.e., NP^{NP} and co-NP^{NP}). By contrast, Governatori et al. (2013) has demonstrated the computational tractability of a deafeasible deontic logic able to compute 'weak' (allowed unless explicitly prohibited) and 'strong' permission (only allowed if explicitly permitted). However, the constructed object logic has a rather constrained expressibility, as it only includes propositional symbols and their corresponding negations, and only modal literals are allowed (obtained by applying modal operators for obligation and permissibility to propositional literals).

In summary, the surveyed results support the conclusion that using modal logic—e.g., temporal logic, dynamic logic, deontic logic—to represent or automatize moral reasoning generally introduces computational intractability. Naturally, in order to capture the rich intricacy of ethical life, normative notions in logical form must be able to account for knowledge, time, agency, program behavior, and multi-agent dynamics, which inevitably adds complexity. However, it must be noted that while these results might appear to indicate something deeply problematic for the prospect of moral machines constrained by polynomial time, they are also immensely productive for other practical purposes. For instance,

[104] See also Balbiani et al. (2008), Herzig and Schwarzentruber (2008), Xu (1998) for other results on the complexity and decidability of STIT.

knowing that a problem is provably in PSPACE may allow it be related to a vast set of other PSPACE-problems, along with the algorithms, approximization techniques, and heuristics that have been developed to tackle them. The fact that the satisfiability problem for dynamic logic is EXPTIME-complete does not stop it from being useful in formal verifications of program behavior. More broadly, from a design perspective, the complexity results may be of great help to inform ones choice of formal system—its expressivity, operators, syntax, and semantics—for the practical purpose at hand.

5.2.3 The problem of moral semantics

The problem of semantics comes in various variants, some of which have been subject to extensive philosophical treatment for centuries. Earlier in this section, we noted that, if an agent should follow a rule of the type "'If input $X \rightarrow$ do action Y", we assume that the agent has some non-trivial understanding of what X and Y means. In the semantics of propositional logic, we simply say that the terms are either true or false with respect to some model (or in modal logic, with respect to possible worlds), and provide rules (e.g., a truth table) for determining the truth value of sentences made up of those terms and logical connectives (e.g., $\neg, \wedge, \vee, \rightarrow, \Leftrightarrow$). Similarly, semantics in programming languages might simply define the process of how valid strings in the syntax (instructions) will induce (interpretation) certain state transitions (execution), e.g., by manipulating some data structure. In such cases, semantics may be viewed as a purely mechanistic operations, and the job of establishing the link between model and reality—e.g., how any valid sentence that can be expressed by the language corresponds to some real-world state of affairs—is circumvented.

Nevertheless, human moral discourse—e.g., of virtues, principles and judgments—is abundant with descriptions and concepts that are semantically so-called 'thick'. For instance, moral talk involve 'thick descriptions' that embed subjectivity as part of their meaning, e.g., by explaining individuals' behavior in light of internal motivations (Geertz et al. 1973). That is, when we describe an agent as being courageous or fair, we naturally assume that the agent has some subjective characteristics (e.g., psychological dispositions and experience) that exemplifies courage and fairness. In turn, these subjective characteristics might only make sense for an agent, given that the agent also understands how they relate to her *own* subjectivity (e.g., sensori-motor experience, beliefs, desires, and intentions). Moral talk also involve paradigmatic examples of 'thick concepts' that are both descriptive and evaluative (Blackburn 1998); e.g., terms like *generous* and *selfish* can refer to descriptions of certain behaviors (acts of sharing or not sharing one's food), while simultaneously denoting an evaluative quality (being *good* respectively *bad*).

Unfortunately, thick concepts and descriptions invoke issues that remain at the center of long-standing metaphysical and meta-ethical debates. To explicate the meaning of thick descriptions, in so far as they depend on subjectivity, one might in turn require satisfying answers to other fundamental challenges, such as the symbol grounding problem (Harnad 1990), the hard problem of consciousness (Chalmers 1997), and the meta-ethical problems of determining the meaning of "meaning" and truth with regard to moral terms. The first problem is relevant for any system that make use of symbols—e.g., to communicate or reason—as it concerns how symbols ground their meaning. For over four decades, Searle's Chinese Room experiment (Searle 1980) has provided a venue for philosophical discussions about the limits of whether and to what extent computational systems can "understand". Searle famously argues that a computer program that is able to convince a human Chinese speaker that it understands Chinese, does not literally "understand" Chinese, as

it is merely following syntactic rules. But even if we oppose to Searle's broad rejection of computationalism,[105] we still need to give some account of how thick moral terms get their meaning, or more pragmatically, what they 'do' for our moral practices. However, it very much remains an open question to what extent the grounding of symbols rely on, e.g., intentionality (Brentano 1874), perceptual experience (Barsalou 1999), or certain sensorimotor capacities (Taddeo and Floridi 2005), and yet another open question whether such capacities can be carried out by a computational system.[106] Similarly, to explicate thick concepts, one naturally has to assume that it is possible to disentangle their evaluative and descriptive components.[107]

So how do these deeper problem of semantics relate to computational complexity? Of course, it ultimately depends on how we position ourselves with respect to these debates, as they would determine our view of moral semantics, and the role moral terms play in moral behavior. On the one hand, if we hold—following meta-ethical cognitivism—that the meaning of moral terms can be properly disentangled and defined as truth-bearing entities, problems expressed in any 'thick' moral language would in principle be computed in the ways discussed in Sect. 5.2.2 (e.g., the checking validity with respect to a given or all possible Kripke structures). As such, moral semantics would not yield any additional complexity baggage that does not already follow from the moral languages' expressiveness. However, while cognitivists hold that moral terms can express mind-independent facts about the world, it does not mean that we have found them; like divine command theory, the approach presupposes that one has collected exhaustive knowledge of all moral terms—e.g., via access to Platonic reality—and managed to encode them along with all

[105] Computationalism is the family of views that hold that the human mind—including consciousness and cognition—is some form of computation.

[106] Aaronson (2013) has framed a technical version of the related "Waterfall argument" in light of complexity considerations, which supplements the Chinese Room by claiming that meaning is always relative to some external observer (Searle 1992). The argument starts from the observation that any physical system with a sufficiently large state space could in principle implement the semantics of *any* other system; e.g., for some mapping M from a waterfall's initial states I to final states F, there is a way of labeling any given permutation P from I and F such that M implements P, and P may thus represent any "semantics" we like, such as a chess playing program. However, if we actually tried to use a waterfall to compute chess moves, we would need to find a reduction from the chess program to the waterfall, e.g., by showing how chess positions and chess moves can be efficiently tracked to the waterfall's initial and final states. From this, Aaronson conjectures that, for any given chess program with access to a waterfall oracle, there is another chess program with equally good performance and similar resource requirements that does not access the waterfall oracle. In other words, it seems highly probable that any reduction algorithm from chess to waterfalls would simply solve chess problems, and not use the waterfall in any meaningful way. For Aaronon, this mirrors the more substantive notion of completeness: the class that a problem is reduced to cannot itself be sufficient to solve the same problem. I.e., while NP-problem X_1 can be solved in polynomial time with access to an oracle for X_1, problems in P cannot be reduced to problems in P (as this would imply that every problem in P is P-complete). The presumed equivalence between waterfall and chess computation thus carry little substance, unless the equivalence can be demonstrated in a model of computation that itself isn't capable of solving waterfall or chess problems.

[107] This would, somewhat coarsely, be the position of Blackburn (1992) and Hare (1952). In contrast, thinkers like Putnam (2004) and Williams (2006) see thick concepts as indivisible blends of fact and value. For instance, Putnam states that 'thick' ethical concepts "simply ignores the supposed fact/value dichotomy and cheerfully allows itself to be used sometimes for a normative purpose and sometimes as a descriptive term" (Putnam, 2004, p. 35).

their semantic relationships.[108] At present, nevertheless, there is strong disagreement about what moral terms mean.[109]

A skeptical alternative would be to bite the bullet of undecidability. For instance, Rice's theorem states that all non-trivial semantic properties of a language recognized by a TM are undecidable, where semantic properties are about a programs behavior, and non-trivial properties are neither true for all partially computable functions, nor false for all partially computable functions.[110] To illustrate, we can imagine a moral language ML as a set of TM descriptions, and the criteria for TMs to be in ML is that their language L(TM) accepts at most three strings ($|L(TM)| \leq 3$). If a language M_1 belongs to ML ($M_1 \in$ML), it means that M_1 satisfies the property ML. That ML is a property of TM languages can be shown by the fact that two machines with the same language, M_1 and M_2, are either both in ML or neither in ML; since they have the same language, they have the same number of strings. To show that ML is non-trivial, we let M_3 be a machine that accepts every string, and M_4 be a machine that rejects every string. Since $M_3 \notin$ ML (accepting more than three strings), and $M_4 \in$ ML (0 is ≤ 3), it follows from Rice's theorem that ML is undecidable. Of course, we can let ML denote a great number of things, and given the vast generality of Rice's theorem, we can demonstrate undecidability for a large set of problems pertaining machine behavior.[111] Analogously, we can draw a skeptical conclusion about the prospect of moral languages; as long as there are some disagreement about the meaning of moral terms (e.g., different agents computing different outputs), there can never be a decidable moral language.

There are, of course, many other palpable reasons to be skeptical about the decidability of moral semantics. For instance, both computability and theories of meaning finds a common nemesis in the self-referential Liar Paradox: "this sentence is false". It plays a central role in Gödel's first incompleteness theorem—by replacing "false" with "not provable"—as no consistent system of mathematics can prove truths about itself. Similarly, Turing's Halting problem demonstrates that it is undecidable whether a computer program halts, as for any program P that can decide "Yes" for halting can be countered by another program that uses P as input in order to produce the opposite "No".[112] On the side of theories of meaning, Alfred Tarski found that the Liar Paradox only appears in "semantically closed" languages, i.e., a language that can express the truth of its own sentences (Tarski 1944). Tarski's own solution—to separate the referring meta-language from the referred object language in a constructed hierarchy—was in turn found to be incomplete by Kripke (1976), who, among other things, employed self-referential tricks to produce statements that break the hierarchy.[113] Of course, Kripke's solution—which utilizes partially defined truth predicates ("undefined")—can in turn be targeted by a *strengthened liar paradox*: "this sentence

[108] Of course, this is not unfeasible for extremely limited state spaces. For instance, we can imagine a toaster that uses sensors to read whether a toast is under-baked, baked, or burnt, and understand how evaluative sentences "this toast is bad!" express certain facts about about the toast's states.

[109] Recalling Jørgensen's dilemma from the previous section, we also note that there are profound disagreements about valid inference in deontic logic.

[110] Conversely, a partial function is trivial if it is true for all partial computable functions or for none.

[111] For instance, whether a given TM computes a constant function, a total function, the identity function, add two natural numbers, or a computable function can easily be shown to be undecidable using Rice's theorem.

[112] In turn, the more general Rice's theorem can be proven by reduction from the Halting problem.

[113] Kripke gives the following example, expressed by Jones and Nixon: (*J*) "Most (i.e., a majority) of Nixon's assertions about Watergate are false", (*N*) "Everything Jones says about Watergate is true" (Kripke, 1976, 691). In the Tarskian hierarchy, *N* needs to be on a higher level than everything that Jones says, and *J* needs to be on a level higher than what everything Nixon says.

is either false or undefined". Ultimately, as any solution seems to produce new self-referential problems that applies to the new solution, any semantic theory centering on truth is haunted by a paradox out for revenge (Beall 2007). Moreover, as demonstrated by Dahl (2022), the Liar Paradox is not content with semantic theories based on truth, but extends to all theories that seeks a unified explanation of meaning for *any* language. In short, since any unified theory of meaning requires a language that is expressive enough to assert its own meaning, and no language can coherently assign meaning to itself while articulating the unified theory, it follows that a universal theory of meaning is impossible.

A more pragmatic route is to deny that moral terms can be true or false, and instead say, following Wittgenstein (2010), that "meaning is use". Besides purifying moral semantics from Platonism and paradoxes, it would potentially bring moral talk closer to the eclectic social practices from which it stems, where occasional quarrel, misunderstanding and dissent is inevitable. In this view, moral expressions should not be understood in virtue of any formal account of meaning, but rather how they, in more or less satisfactory ways, serve our moral practices. Presumably, this could even circumvent the undecidability of non-trivial semantics and halting, if we accept that it is no problem that arbitrary programs either do 'this or that'; pragmatically, they either work well, or they do not. In place of truth-conditions, we could adopt one or several of the prominent non-cognitivist approaches to moral language, e.g., that moral statements function to express emotion and elicit emotion in others (Stevenson 1937); to assert prescriptive judgements (Hare 1952); or convey attitudes (Schroeder 2010).[114] However, while such theories may be credible for emotional, judgemental, and affective humans, they are less suitable for a complexity analysis, as they often presuppose a human-specific psychology.

Nonetheless, one way to analyze the complexity of potentially 'mindless' agents' use of moral terms is to look to modern Wittgensteinans such as David Lewis and Robert Brandom. Before his work on counterfactuals, Lewis provided an early game-theoretic analysis of social conventions (Lewis 1969). In this view, following the footsteps of Schelling (1960), linguistic as well as moral conventions can be viewed as self-perpetuating solutions to reoccurring *coordination problems*, where it is mutually beneficial for self-interested agents to coordinate their actions. Linguistic meaning, more particularly, have subsequently been explored within the paradigm of *signaling games*. In a simple signaling game, a messenger seeks to convince receivers that they are of a certain type—e.g., that they are competent—where the actual type is only known to the messenger. Intuitively, no honest messenger benefits from being misunderstood (e.g., conveying false information about their type), just as it is beneficial for incompetent players to lie, and players receive payoffs depending the receivers' responding action (e.g., hire agent *a* or *b*). Given its explanatory power, signaling games have been used to model the development of communication and linguistic meaning (Skyrms 2010; Huttegger 2007).

Since signalling games are typically modeled as sequential Bayesian games, their equilibria solutions are plagued by the intractability concerns discussed in Sect. 5.1.5; in particular the NP-hardness of checking whether a Bayesian game has a pure-strategy Bayesian equilibrium, and the PSPACE-hardness of computing a pure-strategy NE in Markov Games (Conitzer and Sandholm 2008). However, Lewis' own equilibrium concept, called *coordination equilibria*, has the property that every player also prefers that every other player conform to some regularity *R*, on the condition that at least *all but one* player conform. Naturally, this presupposes some concept of common knowledge, which was subsequently

[114] It should be noted that these non-cognitivist positions should not be equated with a Wittgensteinian outlook on meaning per se, but rather that they reject the belief that moral terms are truth-apt.

generalized by Aumann's correlated equilibrium (Aumann 1974). As discussed in Sect. 5.1.5, CE is computationally attractive, as they can be found in polynomial time for games with any number of players (Gilboa and Zemel 1989).[115] In Lewis' original definition, however, conventions are necessarily arbitrary, in the sense that there is a conflicting regularity R', which could have become the stable convention (and R and R' are mutually exclusive). Of course, the concept of arbitrariness certainly has explanatory value: while it is not arbitrary that a mutually beneficial solution to a coordination problem becomes convention (e.g., cars driving on opposite side of the roads), it is arbitrary that a particular solution was chosen over another (e.g., right-hand traffic as opposed to left-hand traffic). Still, the arbitrariness criteria does not exclude the possibility that a conflicting regularity could result in an even greater joint benefit. Thus, since stable conventions become self-perpetuating, their potential moral value relies on the same optimistic conservatism that permeates post-hoc evolutionary explanations of altruism and social contracts; that there are good reasons why conventions are as they are. By contrast, it is possible for a convention to have an exceptionally high price of anarchy (i.e., a terrible worst-case equilibrium), while the stability of the convention ensures that no one believes that anyone would benefit from going against the grain. However, evaluating the convention against alternatives, e.g., by computing a CE that maximizes the expected joint benefits, invites NP-hardness (Papadimitriou and Roughgarden 2008).

On a more optimistic note, these computational difficulties may be mitigated by cooperative principles for communication, e.g., following the work of Grice (1975). This strategy can be reflected in Lewis' refined analysis of convention (Lewis 1975), which includes preferences with regard to one's own beliefs: "The expectation of conformity ordinarily gives everyone a good reason why he himself should conform" (Lewis 1975, p. 8). Similarly, Lewis writes that "a language L is used by a population P if and only if there prevails in P a convention of truthfulness and trust in L, sustained by an interest in communication" (p. 10). For Lewis, along with other Griceans such as Schiffer (1972) and Bennett (1976), meaning ultimately stems from a coordination between speakers communicative intentions and receivers communicative expectations. In turn, these intentions and expectations may as well encompass cooperative conventions.

A related yet distinct account of "meaning is use" that is apt for explaining moral language has been provided by Brandom (1994). Brandom gives a theory of sapience—the type of rationality that humans possess—based on the notion of discursive practice, which can be summarized as "the game of giving and asking for reasons" (p. 6). Participants of discursive practices take on entitlements and commitments, which can be seen as carrying the normative force of permissions and obligations, respectively. At the core of discursive practices are inferential relations, which preserves commitments and entitlements to other statements. I.e., sentences only carry content in terms of their function, which is inferred in relation to other sentences. What is interesting with regards to moral discourse, is that linguistic performances are characterized by their ability to alter the normative status of the members of a discursive practice. This takes the form of a conversational "scorekeeping", where the participants keep track of commitments and entitlements within the conversational context, e.g., by making, acknowledging, contesting, or withdrawing assertions.[116]

[115] See also the work of Urbano and Vila (2002), which demonstrates how correlated equilibrium can be achieved by imposing computational restrictions on the unmediated communication.

[116] The idea of scorekeeping in the context of conversations was first introduced by Lewis (1979), who acted as a supervisor for Brandon's doctoral thesis in the 1970s.

In essence, the normative pragmatics of linguistic performances determine the inferential aspects of semantics, and not the other way around. However, while Brandom's inferential semantics may help to illuminate the normative commitments of speech acts, we can only speculate about the computational aspect that underpins the ability to successfully participate in a discursive practice, and similarly, whether and to what extent it alleviates or adds computational demands. On the one hand, the game of reason-giving and reason-asking seems tailored to effectively foster cooperative communication in normative life. On the other hand, Brandon's conception of sapience seems profoundly human, which, in addition to the sentience shared with non-verbal animals, involves an understanding of conceptual contents, which in turn may encompass intentional states, beliefs, and desires of oneself and others. In turn, a sapient game of reasons might more or less correspond to a Kantian conception of moral rationalist discourse (Brandom 2006), which presupposes sophisticated and idealized capacities for moral autonomy and freedom; capacities which—at least presently—cannot be construed in computational models.

The main lesson seems to be that, while game-theoretic concerns—of computational intractability and threats of repugnant equilibria—may be alleviated by communication, the challenge is to not only give an account of how communication works, but why it works so well. For Lewis, the success seems to rely on communicative intentions and expectations of speakers, and the cooperative conventions that results from it. For Brandon, it centers on the sapient game of "giving and asking for reasons". Unfortunately, this renders both approaches rather computationally opaque: as they aim to explain human communication, they can, just like non-cognitivist theories, resort to uniquely human features that remain more or less impenetrable from a computational perspective. Still, such theories might potentially yield significant value by enclosing the gap between, on the one hand, cognitive-psychological resources—e.g., reasons, trust, and communication—that fosters efficient cooperation, and on the other hand, the computational architectures that would potentially enable them. In turn, complexity considerations might help to illuminate the rich interrelationship between game-theoretic dynamics, social-psychological capacities, and normative theory in everyday moral interactions.

In summary, this section has discussed more profound issues for moral semantics that, although relatively ignored in machine ethics, remain at the center stage of meta-ethical and meta-semantical debates, as well as in theories of communication. Of course, a simple move would be to cling on to some metaphysical argument against "strong AI" (Searle 1980), some advanced requirements for symbolic grounding (Taddeo and Floridi 2005), or some uniquely human psychology of emotions, and conclude that "machines can never genuinely understand moral language". But even given that one decided to ignore such problems, many complex problems remains for computational moral semantics.

5.3 Consequentialist-deontological hybrids

Having investigated a range of computational aspects of both consequentialism and deontology, we are now in a position to say something more substantial about their difference and potential combination. First, it should be concluded that the claim "deontology is, computationally speaking, less complex than its alternatives" cannot be given a straightforward answer; and in many cases, it is simply false. Based on the material discussed in this section, we can outline a more nuanced answer:

(1) As argued in Sect. 5.1.1, it is a mistake to view rule-following in legal and liberal contexts as computationally simple. While legal rules may efficiently compress voluminous moral wisdom, it is only against the backdrop of a complex relationship between the mechanisms that incentivizes their adherence (e.g., police and punishment), ensures their just interpretation (e.g., courts), along with the moral sentiments of the subjects the laws apply to. For instance, the complexity of "do no harm"-principles in liberal contexts can only be meaningfully analyzed in relation to the capacities of fully autonomous citizens.

(2) On the other hand, deontological rules may be significantly more efficient than alternatives if they are justified on the basis of divine command or legal positivism. However, unless in extremely simplified cases, the knowledge-requirements for such approaches to work in practice remains unfathomably vast, while the knowledge itself may be highly contentious.

(3) As an alternative, deontological rule-following may include justification as a part of the moral computation, e.g., by considering how rule-following behaviors affect others. As discussed in Sect. 5.1.2, this can range from considerations of (i) one's own preferences ("I treat you based on what I personally prefer"), (ii) preferences of others ("I treat you based on what I know of what you prefer"), to (iii) general behavior ("I generally treat you in the way that I would want others to generally treat me"). While weaker versions—e.g., (i) and (ii)—may be relatively simple from a complexity perspective, they also lead to a range of other problems. By contrast, stronger versions—e.g., contractarian or contractualist versions of (iii) are obfuscated by extreme variances with regard to behavioral expectations and the capacities that makes up conceptions of general behavior.

(4) Any general-purpose normative theory that seeks to account for multi-agent dynamics face game-theoretic concerns. This includes issues of rationality (Sect. 5.1.3), incomplete information and recursive reasoning (Sect. 5.1.4), and the intractability of computing morally attractive Nash Equilibria (Sect. 5.1.5).

(5) Any computational system that employ formal logic, e.g., for deontological rule-following, moral reasoning, or communication are subject to expressibility (sect. 5.2.1), intractability (Sect. 5.2.2), and decidability (Sect. 5.2.3) issues that permeate the syntactics and semantics of logic.

(6) Finally, it should be acknowledged that deontology is a family that encompasses a range of ethical theories that may in turn emphasize a range of different cognitive abilities and computational resources. It can be a moral rationalist project of finding universally justifiable and applicable rules on the basis of a shared autonomy and rationality. It can be a contractarian project of finding mutually beneficial action-rules based on self-interest. In the simplest case, rules can act as merely 'fast and frugal' heuristics that support agents with bounded cognition to produce *any* action in complex or novel situations, even if there is no way to evaluate whether the performed action is appropriate in the specific situation; it may simply be an automatic 'default' action, or rely on optimistic conservatism ("it has worked well before, so it might also work well in the future"). In more ambitious cases, deontological rule-following might ask one to compute morally attractive equilibria for large-scale coordination problems with incomplete information.

Based on these considerations, it may be misleading to compare the complexity of deontology and consequentialism, as any comparison relies on a particular conception of what the theories dictate. For instance, both theories are equally targeted by game-theoretic intractability (Sect. 5.1.3) insofar as they take other agents into account. Similarly, many of the deeper problems of moral semantics that plague logical systems (Sect. 5.2.3) can also be construed for consequentialist agents: e.g., what is it for a computational system to "understand" what a certain utility really is? Another reason is that deontology and

consequentialism may converge on their solutions, for instance, in cases where the deontological moral right also produces the optimal outcome. Thus, a reasonable alternative is to view them as complementary theories, which emphasize different cognitive capacities, computational resources, or aspects of ethical life.

To that end, it is no surprise that hybrid accounts, which combine aspects of consequentialism and deontology, are well-reflected in moral philosophy (Brandt 1979; Hare 1981), moral psychology (Kahneman 2011; Greene 2007) and machine ethics (Bauer 2020; Arkin 2007; Dehghani et al. 2008a; Stenseke and Balkenius 2022; Pontier and Hoorn 2012; Azad-Manjiri 2014; Tufiş and Ganascia 2015; Govindarajulu and Bringsjord 2017; Pereira and Saptawijaya 2009). However, it should be noted that in almost every case, deontology takes on a rather narrow role in these architectures—often within a broader consequentialist framework—which does not reflect its rich tradition in ethical theory. For instance, deontology might act as *a priori* constraints which prevent certain intrinsically bad actions (Pereira and Saptawijaya 2009; Dehghani et al. 2008a). Another option is to use deontological rules to foster run-time efficiency, e.g., by turning consequentialist computation or exploration into exploitable rules (Stenseke and Balkenius 2022).[117] Similarly, in the dual process theory of moral cognition (Greene 2007), deontological judgements are construed as fast and instinctive responses, whereas consequentialist judgements denote slow and reflective reasoning processes.

With regards to complexity, the emerging trend is that moral rule-following can either (i) be used to prevent intrinsically bad actions, (ii) reduce the run-time complexity of (often consequentialist) moral computations, or (iii) act as a 'principles first' or 'defeasible' default mode in situations where the agent has nothing else to base their decision on. From a moral-psychological perspective, (ii) and (iii) can be supported by the idea that automatic moral judgements compress moral wisdom, e.g., on the basis of evolutionary adaptations promoting cooperation, culture-specific norms fostering collective well-fare (e.g., salient correlated equilibria), or the moral lessons an individual learns to internalize through experience. For machine ethicists, consequentialist-deontology architectures thus offer an attractive smorgasbord of context-specific modularity; from top-down a priori constraints based on preexisting moral knowledge (divine command) to bottom-up a posteriori turned into new top-down constrains (e.g., learning efficient rules through experience). However, this image is not without its flaws, as any reduction in run-time complexity implies one or many of the following problems:

The problem of optimistic conservatism In essence, 'fast and intuitive' moral behavior cannot be guaranteed to produce good moral results, unless it is evaluated against alternatives (e.g., on the basis of experience or moral reasoning). As such, the moral value of intuitions rely on an optimistic conservatism, as the efficiency of following an intuitive deontological (or rule-consequentialist) rule may come at the expense of losing out on a potentially better outcome that was only attainable via further reflection. Following Hare (1981), this is the challenge of knowing when to think like a 'prole' (using moral intuition) or an 'archangel' (using critical reflection). From a computational perspective, it reflects the well-studied explore-exploit dilemma (Sect. 4.3). It also echoes the problematic post-hoc rationale of norms and conventions. As discussed in (Sects. 5.1.3 and 5.2.3): even if we believe that a certain moral convention—or rule, norm, principle—emerged because it helped agents with bounded cognition to coordinate towards mutually beneficial goals, the

[117] Since consequences motivates the use of certain rules, it is more suitable to view this solution as a form of two-level utilitarianism (Hare 1981).

stability of the convention may rely on an optimistic conservatism about the convention itself, even if there are alternative conventions that might have been even better. Similarly, just because some moral intuitions may be a result of adaptations that yielded reproductive success for our evolutionary ancestors, it does not automatically make those intuitions morally right today (Greene 2014).

The problem of speed vs performance A related problem is the choice between moral speed and moral performance. For instance, even if we, like Hare (1981) and Greene (2014), believe that slow utilitarian deliberation should take normative and epistemological priority over 'fast' intuitions, there is still an open-ended conflict between moral speed and moral performance. In what circumstances do we opt for a fast—e.g., feasible, suboptimal, or satisficing—option in favor of a slower but potentially better option?

The problem of reasoning and learning'Fast and intuitive' moral intuitions may foster run-time efficiency, but only by ignoring the complexity of the process that gave rise to the moral intuition itself. On the one hand, this can refer to the complexity of the process characterized by Hare (1981) as the "critical level" of moral thinking (which governs the principles of the "intuitive layer"). On the other hand, we might think of it purely in terms of a learning-process: e.g., how many learning examples and how much training-time does an agent need to successfully internalize a moral intuition that allows it to effectively solve a given moral problem at run-time? In principle, since artificial neural networks can be used to approximate *any* function (Scarselli and Tsoi 1998), machine learning systems should be able to achieve great run-time performance in moral problems given the right kind of learning. However, as we will see in Sect. 6, this will instead put increasing demands on sample and training-time complexity, while introducing other problems related to induction.

The problem of knowledge Instead of reasoning or learning, efficient moral intuitions may be secured on the basis of moral knowledge. For instance, an alternative that is available for moral realists and meta-ethical cognitivists—who believe that moral intuitions can denote true propositions that reflect subject-independent features of the world—is to collect and write down the dicta of objective moral reality. Of course, as already noted several times throughout this paper, such a project seems, for a variety of reasons, deeply problematic. With that said, it does not exclude the possibility that there are *some* moral intuitions that have universal (or near-universal) consensus, or some—following Asimov (1942)—that are particularly attractive for computational agents.

In summary, if we view deontology simply as 'rule-following', it is clear that it yields efficient decision procedures from a bounded rationality point-of-view. Nevertheless, while it may lead to attractive run-time performance, the moral value of such efficiency is either based on optimistic conservatism, the results of some other complex process (reasoning or learning), or the collection of vast moral knowledge. Essentially, the moral power of rules—their general applicability, general justification, and computational simplicity—can only be secured in complex ways. After all, the simplicity of moral rules should not prevent one from questioning their authority; rather, it should help one remember that they—like logical systems—merely represent idealized facets of byzantine phenomena.

6 Virtue ethics and moral machine learning

As deontology centers on actions and consequentialism on results, they can be naturally construed as moral decision procedures. By contrast, virtue ethics is about *being* rather than *doing*: instead of focusing on what the right action is, or what action yields the best

outcome, it asks us to foster our moral character and the internal dispositions—virtues of courage, fairness, and temperance—that enables us to *be* morally virtuous. More broadly, virtue ethics denote a vast family of ethical traditions that emphasize the role of our moral character, and the theory can find its diverse origin in thinkers such as Confucius and Mencius in the East, and Plato and Aristotle in the West (Crisp and Slote 1997). In contemporary times, it has earned its spot as a central normative theory in Anglophone moral philosophy through the contributions of Anscombe (1958), Nussbaum (1988), Hursthouse (1999), and Annas (2011).

In machine ethics, virtue ethics has several times been proposed as an appropriate blueprint for the creation of artificial moral agents, as it emphasizes aspects of ethical life that are relatively ignored in other theories (Coleman 2001; Wallach and Allen 2008; Howard and Muntean 2017; Stenseke 2021).[118] Taking the character as the central object of moral evaluation—which encompasses both rational deliberation and psychological dispositions—it paints a more holistic picture of what it is to be moral; capturing not only what we ideally *ought* to do, but what motivates us to act in morally praiseworthy ways. One key aspect of this picture is the notion of *phronesis* ("practical wisdom"), which can be construed as the moral wisdom or skill an agent learns from practice and experience (Annas 2011). The focus on development and learning has in turn inspired machine ethicists to unify virtue ethics with connectionism; both in terms of modern machine learning methods, and as a broader theory of cognition (Casebeer 2003; Wallach and Allen 2008; Howard and Muntean 2017; Berberich and Diepold 2018). Optimistically, with the ability to constantly learn from experience, be sensitive to contexts and adaptable to changes, a virtuous machine might thus be able to apprehend the intricacies of human norms in dynamic environments where mere utility-maximization or rule-following fails.

In despite of these promises, virtue ethics remain relatively elusive from a computational perspective. One reason is that virtue ethics is founded on a deeply human view of what constitutes a moral life—of flourishing, reasoning, emotions—against a rich backdrop of culture and tradition. For obvious reasons, this makes virtue ethics hard to analyze from a complexity perspective, as it would require a comprehensive computational description of what a human being is, along with all her history and culture-specific flavors. Nevertheless, it is possible to isolate and analyze one necessary aspect of any virtuous agent: its ability to learn. In principle, any of the problems discussed thus far can be re-framed as a learning problem. Instead of asking "can I solve problem X effectively?", we ask "can I *learn* how to solve X effectively?". If the answer is yes, the de facto run-time complexity might be trivially low given the appropriate training. Intuitively, if a specific problem has already been solved—e.g., having discovered the optimal action-combination for a certain situation (Sect. 4.1), an optimal reinforcement learning policy for an environment (Sect. 4.3), or an action-rule in a strategic setting (Sect. 5)—the same solution may be applied in constant time $O(1)$ to future cases given that the very same problem re-occurs. And even if the exact same problem never re-occurs, there might be patterns or trends to extrapolate from situations that are sufficiently similar. Through repeated encounters with salad bars (Sect. 2), we might learn about good combinations of ingredients. Seeing the same ingredients appearing in other salad bars, we can make use of our previous experience to efficiently put together a tasty salad. Following the virtue-theoretic emphasis on moral learning, the same should hold for moral behavior. In the words of Aristotle (NE,

[118] See Stenseke (2022a) for a recent survey on computational implementations of virtue ethics.

book IV, chapter 8): "[...] though the young become proficient in geometry and mathematics, and wise in matters like these, they do not seem to become practically wise. The reason is that practical wisdom is concerned also with particular facts, and particulars come to be known from experience; and a young person is not experienced, since experience takes a long time to produce" (Aristotle 2000, p. 111).

In turn, learning a machine is the business of *machine learning*, which, due to a wealth of recent advancements, has come to dominate the field of AI in the 21st century. The relevant question for our investigation—and the topic of this section—is: are there any computational complexity considerations that might constrain a computational agents ability to learn in the moral domain?

6.1 The complexity of learning

As with any mature field of mathematical analysis, there are many relevant variants, settings, and measures that can be used to formally analyze the complexity of learning.[119] First, we might differentiate *run-time complexity* (the number of state transitions an algorithm needs to perform at run-time to solve problem X), from *training-time complexity* (the number of steps required to train the algorithm to solve X) and *sample complexity* (the number of data points needed to learn how to solve X). Nonetheless, in practice, the distinction between these types may break down; for instance, if one sample represents one point in time, our sample complexity would be equal to our training-time complexity; similarly, we might understand training-time complexity *as* the run-time complexity of a learning algorithm. What is important to note, however, is that the three types are intimately linked. In a completely known environment (say, Chess), simulation (e.g., using Monte Carlo methods) may be an effective way to get experience, e.g., by trying out the value of many different possible decisions. In this case, sample complexity measures how much simulation is required to find a good chess move. If the sample complexity is low, it means that it can be learned effectively, as it gives a lower bound on the total computational complexity (Kakade 2003). In other situations, there might be an abundance of data and plenty of time to train our model, while the task itself may refer to some general ability—e.g., evaluated using a Turing test—as opposed to how the system solves some specific decision-problem. Large Language Models (LLMs) like GPT-3 (Brown et al. 2020), which uses significant parts of the internet as training-data in order to produce new text, constitute suitable examples of this. In the most extreme case—e.g., an on-line reinforcement learning setting—if an agent have no information about the environment and no access to a simulation of it, trial-and-error exploration might be the only path to learning.

Our choice of complexity resource ultimately depends on the setting for our learning agent, and each setting has their own range of specific sub-types and relevant measures. To that end, it is common to differentiate between three broad classes of machine learning settings: (i) supervised learning (SL), where the aim is to learn from pre-labeled data, (ii) unsupervised learning (UL), where the training data lacks labels, and (iii) reinforcement learning, where learning is based on feedback. For an SL agent, the relevant question may be: how many labeled pictures of cats do I need to see in order to, within some margin of error, accurately classify new pictures as portraying cats or not? In a virtue-theoretic view,

[119] See Vapnik (1999) for the definite introduction to statistical learning theory, along with an exhaustive account of its rich history.

this could be rephrased as: how many labeled examples of "courage" (e.g., acts of courageous behavior) do I need to observe in order to classify unobserved actions as courageous in the future? In this case, we might only be interested in how sample complexity—number of pictures depicting courage in the training set—reduces the agents prediction error in the classification task (e.g., empirical risk minimization). For an RL agent, the question might instead be: how much do I need to observe the overall behavior of a moral exemplar (e.g., a virtuous human) in order to approximate their reward function,[120] Here, it could also be important to consider the potential risk of detrimental mistakes during an agent's learning phase. As discussed in Sect. 4.3, if trial-and-error exploration is the only option the goal for an RL agent may not be to merely maximize the cumulative reward, but to minimize regret relative to the optimal solution.

Before we can ask whether something can be learned effectively, we first need to determine whether it can be learned at all. For instance, we let X be a set of points, Y a set of labels (e.g., 0 and 1), and $\mathcal{H}$ a set of hypotheses h (e.g., binary classifiers) that takes an $x \in X$ and outputs a label $y \in Y$. The goal for our learning algorithm $\mathcal{L}$ is to, given a sequence of labeled training samples (x, y) drawn randomly from some distribution ρ over X, infer a hypothesis $h \in \mathcal{H}$ that is able to correctly classify future instances of $x \in X$ given some accuracy ϵ and failure probability δ. In turn, we say that a hypotheses space $\mathcal{H}$ is learnable if there exists a learning algorithm which, given a finite number of training samples $(x, y)_n$, can map inputs to outputs within ϵ of the optimal with a probability of at least $1 - \delta$.[121] If so, sample complexity can be defined as $n(\rho, \epsilon, \delta)$, which says that we need n training samples to learn a target function with respect to distribution ρ, error rate ϵ and failure probability δ.

6.1.1 Weak and strong sample complexity

Based on these definitions, we may also differentiate between *weak* and *strong* variants of sample complexity, e.g., by asking how many samples we need to learn a target for *some* specific input–output distribution ρ over X (weak), and how many we need to learn it for *any* possible distribution (strong). Following the strong approach, the impossibility results commonly known as the No Free Lunch Theorem (NFL) establishes that there will always be unfortunate distributions for which the sample complexity is arbitrarily large (Wolpert 1992, 1996; Schaffer 1994). As an intuitive example, we can imagine a learning algorithm whose goal is to predict the weather—in this case limited to sunny (S) or rainy (R)—based on the weather on previous days. Collecting data for three days, we have 2^3 possible weather-histories (i.e., SSS, SSR, SRR, SRS, ...). We then measure the learning algorithm's error ϵ as the ratio of incorrect predictions (e.g., predicting S for a day of rain). Then, we can demonstrate that every learning algorithm achieves a perfect $\epsilon = 0$ in exactly one weather history, a maximally bad $\epsilon = 1$ in another, a mixed result of 2/3 in three histories, and $\epsilon = 1/3$ in the three remaining histories. NFL establishes that, for each possible ϵ, every learning algorithm achieves ϵ for the equal amount of possible weather-histories. For instance, if we assign the same probability to each possible weather history (i.e., a uniform

[120] This particular method is called *inverse reinforcement learning* introduced by Ng and Russell (2000), and suggested as a path towards artificial virtuous agents by Berberich and Diepold (2018).

[121] For instance, following empirical risk minimization, we might define the optimal $h*$ as the hypothesis among $\mathcal{H}$ for which the risk of misclassification is minimal.

probability distribution), NFL entails that every learning algorithm has the same expected ϵ of 1/2.

NFL results have subsequently been extended to optimization (Wolpert and Macready 1997), supervised learning (Wolpert 2002), statistical learning (Von Luxburg and Schölkopf 2011), meta-learning (Giraud-Carrier and Provost 2005), and data privacy (Kifer and Machanavajjhala 2011).[122] Essentially, it means that there exists no learning algorithm that can perform well on every learning task having trained upon a dataset of a fixed size. Thus, for every learning algorithm, there exists a task on which it fails, as no learning algorithm can generalize to *all* possible realities while having only observed *some* instances of the realities. To that end, NFL has also been discussed in relation to more fundamental problems in the philosophy of induction, e.g., in connection to Hume's problem of induction (Sterkenburg and Grünwald 2021; Schurz 2017) or Occham's razor (Lattimore and Hutter 2013). Hume famously advanced skepticism against the very justification of induction, arguing that deductive reasoning alone cannot secure the validity of inductive inference; and neither can induction, due to circularity, provide non-deductive grounds for itself (Hume 1739).

Of course, it may not be fair to advance the fundamental issues of induction against the feasibility of moral learning systems. While they may obstruct the prospects of a perfect universal moral learner, it does not stop us from pursuing weaker yet reasonable alternatives that are practically viable. Instead of seeking a global and model-independent justification for why inductive inference seems to work, we can opt for local and model-relative justifications in order to explain why *some* learning algorithms work better than others (Sterkenburg and Grünwald 2021). However, it should be stressed that any such alternative would inevitably entail some form of inductive bias; assumptions that we exploit to enable and foster learnability. One strategy to alleviate the curse of arbitrarily large sample complexity is to constrain the space of probability distributions, e.g., by making assumptions about the structure of the distribution from which the data-points are drawn (called "parametric" procedures in statistics). The most straight-forward parametric assumption can be found in the Central Limit Theorem, which states that when we sum up randomly drawn independent variables, they tend towards a normal distribution.[123] In fact, most advancements in machine learning rely on some form of parametric assumptions, e.g., using linear and logistic regression, and the parameters of artificial neural networks.[124]

Another alternative is to constrain the space of hypotheses. In the philosophy of induction, this can be motivated on the basis of Occam's razor, which roughly states that simpler hypotheses are generally more better than complex alternatives. But what does "simpler" mean? And how many samples do we need to infer a simple hypothesis that is able to predict well? In computational learning theory, such questions can be effectively addressed by the Probably Approximately Correct (PAC) model of learning, introduced by Leslie Valiant (1984). Following the formalism described earlier, we consider a learning algorithm $\mathcal{L}$ that wants to learn a Boolean function $f : X \to \{0, 1\}$ in a finite set of hypotheses $\mathcal{H}$, on the basis of samples (x) drawn independently from distribution ρ over sample space X. We

[122] See Adam et al. (2019) for a systematic review of NFL theorems.

[123] Philosophically, this means that one accepts that the world tends towards a normal distribution.

[124] It should be note that deep neural networks with sufficiently many parameters can be viewed as non-parametric; e.g., Lee et al. (2017) demonstrates that an infinitely wide deep network is equivalent to a non-parametric Gaussian process.

call a hypothesis h infered by $\mathcal{L}$ *good* if it can approximate f within some error $\epsilon > 0$, in the sense that:

$$\Pr_{x \sim \rho} [h(x) \neq f(x)] \leq \epsilon \tag{5}$$

Then, we say that $\mathcal{L}$ is "probably approximately correct" if it can make good approximations with probability $1 - \delta$, for any choice of ρ and all failure probabilities δ and error rates $\epsilon > 0$. Finally, we say that a function f is PAC-learnable if there exists an $\mathcal{L}$ that can make "probably approximately correct" predictions with a sample size n that is a polynomial function of $1/\epsilon$ and $1/\delta$. In other words, PAC learning formalizes the idea that, if a function is learnable, it means that there exists a learning algorithm that with a reasonable likelihood can get reasonable generalization errors if it trains on randomly selected data, all while the number of samples are upper-bounded by a polynomial.

What is interesting from a computational complexity perspective is that PAC-learning yields a bound on sample complexity. If a target function is PAC-learnable, then the number of samples n required to learn the function can be derived by:

$$n = \frac{1}{\epsilon} \ln \frac{|\mathcal{H}|}{\delta} \tag{6}$$

Particularly, equation (6) illustrates the intimate relationships between prediction error, confidence, samples, and the space of hypotheses, and their computational trade-offs. For instance, it captures the intuition that error rate ϵ and generalization error δ can be reduced (although never to 0) by increasing the number of samples. It also shows that successful learning from a limited number of samples n requires us to constrain the cardinality of the hypothesis space $\mathcal{H}$; e.g., either by reducing the number of individual hypotheses, or by reducing their descriptive complexity.[125] By contrast, a larger $\mathcal{H}$ may lead to overfitting, i.e., when the hypothesis closely mirrors a particular set of samples and fails to generalize to additional observations.

From a philosophical perspective, PAC-learning is interesting since it, regardless of Hume's skepticism, defines a rich class of instances where induction is guaranteed to work (at least probably approximately). However, one significant trade-off with the model is that it only applies to finite classes of hypotheses, which inevitably entails a compromise between approximation accuracy and the learning algorithms capacity. For instance, a hypothesis containing continuous parameters may need to be turned into discrete parameters. The question is, how finely do we divide the infinite continuum? This conundrum can be addressed by VC theory (developed by Vapnik and Chervonenkis 1974, 2015). Importantly, as opposed to hypotheses space, VC theory brings the richer concept of VC dimensions (VCD), which measures a learning algorithm's expressive capacity by the cardinality of the largest set of points the learner can *shatter*. For instance, given three points in a two-dimensional space, we can ask whether a linear classifier *LC* can correctly separate negative (labeled −) and positive points (labeled +). Particularly, we ask, for all possible ways of labeling the three points $2^3 = 8$, is there a way we can draw a straight line that separates positive from negative points? After discovering that it is possible—i.e., *LC* shatters the set containing the three points—we try to do the same for four points. Since no set of four

[125] Aaronson (2013) has interpreted this aspect of PAC-learning as an mathematical justification for why Occam's Razor works.

points can be shattered by a straight line,[126] we conclude that the VCD of *LC* is 3. Similarly, a class of hypotheses $\mathcal{H}$ shatters the points $\{x_1, \ldots, x_m\} \subseteq X$ if there is a hypothesis $h \in \mathcal{H}$ which agrees with all 2^m possible configurations of $h(x_1), \ldots, h(x_m)$. The VCD of $\mathcal{H}$ is then the cardinality of the largest m where there is a subset $\{x_1, \ldots, x_m\} \subseteq X$ that is shattered by $\mathcal{H}$. In turn, due to the work of Blumer et al. (1989), VCD has been unified with PAC-learnability through the fundamental insight that finite VC dimensions provides the necessary and sufficient condition for distribution-free learnability. In other words, if a set of hypotheses is PAC-learnable, its VCD is finite. Of course, while it does not alleviate the fundamental problems of induction, it yields a framework for describing *when* induction is feasible, a rich measure of expressive capacity (VCD), as well as a substantive notion of simplicity (i.e., smallest amount of VCDs). With regards to complexity, it has recently been proven by Hanneke (2016) that the optimal sample complexity[127] of PAC-learnability for class $\mathcal{H}$ is:

$$n(\epsilon, \delta) = O\left(\frac{VCD(\mathcal{H}) + \ln \frac{1}{\delta}}{\epsilon}\right) \tag{7}$$

Nevertheless, theories for feasible learnability face the same problem as Nash equilibrium: just knowing that there exists a hypothesis h in $\mathcal{H}$ that is consistent with the data does not necessarily mean that it is easy to find. As such, PAC-learnability ignores the vast computations that are potentially required to actually find a good hypothesis. To that end, a large set of hardness results have been proven for PAC. In the *proper* setting, where the learner is required to output $h \in \mathcal{H}$, Pitt and Valiant (1988) proved that representation classes such as disjunctions of two monomials (a polynomial with only one term), Boolean threshold functions, and Boolean formulae where each variable occurs at most once, cannot be efficiently learned, as they can be reduced to known NP-complete problems. Based on widely used cryptographic assumptions—e.g., the Rivest-Shamir-Adleman system and Blum integers—Kearns and Valiant (1994) proves representation-independent hardness results in the *improper* setting (where the learner can output any $h \notin \mathcal{H}$) for a range of representation classes, including polynomial-size Boolean formulae, constant-depth threshold circuits, and acyclic deterministic finite automata.[128] In addition, while the hardness of improper learning rely on cryptographic assumptions, Applebaum et al. (2008) shows that a proof would either "collapse" the polynomial hierarchy[129] or imply that any average-case hard problem in NP can be transformed into a one-way function (which would yield an outstanding break-through in cryptography).

Another important distinction in learning theory besides proper and improper, is the one between *realizable* (or "noise-free") and *agnostic* ("noisy") learning. In the realizable case, it is assumed that there exists an optimal hypothesis $h*$ in the space of hypotheses $\mathcal{H}$ in the sense that its $\epsilon = 0$. In agnostic learning (Kearns et al. 1992), no assumptions are made

[126] This is a consequence of Radon's theorem, which can be used to infer the VCD of linear separations of d-dimensional points.

[127] Here, optimal means that the upper bound matches known lower bounds (Ehrenfeucht et al. 1989; Blumer et al. 1989) up to numerical constant factors.

[128] In the improper context, representation-independent hardness means that learning remains hard regardless of the form the algorithm represents its hypothesis, on the basis that the hypothesis can evaluated in polynomial time.

[129] This means that if NP = co-NP, then it follows that PH = NP. It is widely believed that a collapse of the PH is implausible.

about the target function; we simply want to find the best possible h from *some* distribution.[130] Paradigmatic problems in statistical machine learning includes the learning of halfspaces (or linear threshold function),[131] monomials, and decision lists. While learning a halfspace in the proper realizable case can be done in polynomial time via linear programming, its NP-hardness in the proper agnostic case have been proven in various ways.[132]

As discussed in Sect. 4.3, similar computational hardness prevail in reinforcement learning (Mundhenk et al. 2000; Papadimitriou and Tsitsiklis 1987). In fact, there are many reasons to believe that reinforcement learning is significantly harder than the supervised setting: the learner may not receive a training sets from the environment; the learner's may only receive 'noisy' immediate rewards (which can deceive the agent into learning policies that does not maximize the long-term future rewards); exploiting comes at the expense of losing out on exploring (and vice versa); and there may even be detrimental consequences to consider (minimize regret).[133]

6.1.2 Machine learning theory versus practice

It is important to stress that the theoretical considerations discussed here might have limited relevance for the practical viability of machine learning in various domains. Clever uses of inductive biases—e.g., task representation and parametric tools—along with vast amounts of training data and computational power continue to defy what might have appeared to be impossible only a decade ago. For instance, the performance of deep learning models in the field of natural language processing has recently been accelerated via the transformer architecture (Vaswani et al. 2017), which utilizes attention mechanisms to process tokens from any position in the input sequence; leading to improved context sensitivity through efficient use of parallelization. Similarly, advances in deep reinforcement learning has showed that only small sets of demonstration samples can significantly accelerate the learning process (Hester et al. 2018). Another advancement in value alignment techniques—in particular for LLMs—is Reinforcement Learning from Human Feedback (RLHF), which consists of training a reward model based on human feedback—e.g., a preference ranking of the outputs generated by the system—which is then used to fine-tune the model (Ziegler et al. 2019). In fact, what is surprising is not the general trend that shows that learning is computationally hard: it is rather that we lack rigorous explanations for *why* some learning systems seem to generalize well in practice. This is known as the "paradox of deep learning", which centers around understanding the empirical success of deep learning despite the absence of theoretical explanations (Kawaguchi et al. 2017; Neyshabur et al. 2017; Arpit et al. 2017; Zhang et al. 2021).

In turn, this generates a range of convoluted issues that are more or less unique to machine learning. For instance, what does it mean that a LLM—having trained on large parts of the internet—is ethical, when we know that the data itself is deeply imbued by

[130] See Hopkins et al. (2022) for an exposition of the deeper relationship between agnostic and realizable learning.

[131] Formally, a halfspace is a Boolean function of the form $f(x) = \text{sign}\,(w_1 x_1 + \ldots, w_n x_n - - - \theta)$, where w_i are "weights", θ is the "threshold", and $w_1 \ldots, w_n, \theta \in \mathbb{R}$. The sign function returns 1 on arguments ≥ 0, otherwise -1.

[132] See, among others, Angluin and Laird (1988), Amaldi and Kann (1998), Håstad (2001), Ben-David et al. (2003) and Feldman et al. (2012) for a more recent overview. See also Daniely et al. (2014) for the improper case.

[133] See Kakade (2003) for a detailed investigation of sample complexity in reinforcement learning.

Table 3 Summary of the surveyed complexity results

Problem	Results
Combinatorics (Sect. 4.1)	
Optimal plan of n unordered actions	$\Theta(2^n)$
Optimal plan of n ordered actions	$\Theta(n!)$
STRIPS and propositional planning	PSPACE-complete (Bylander 1991, 1994)
Bayesian inference (Sect. 4.2)	
Exact inference	#P-complete (Roth 1996)
Most probable explanation (MPE)	NP-complete (Shimony 1994)
Maximum a posteriori hypothesis (MAP)	NP^{PP}-complete (Park and Darwiche 2004)
Approximate exact inference	NP-hard (Dagum and Luby 1993)
Approximate MPE	NP-hard (Abdelbar and Hedetniemi 1998)
Partial MAP	NP-hard (Park and Darwiche 2004)
Sequential decision-making (Sect. 4.3)	
Finite MDP	From PL to EXPSPACE-complete (Mundhenk et al. 2000)
Finite POMDP	PSPACE-complete (Papadimitriou and Tsitsiklis 1987)
Infinite POMDP	Undecidable (Madani et al. 2003)
Restless bandit	PSPACE-hard (Papadimitriou and Tsitsiklis 1994)
Strategic dynamics (Sect. 5.1)	
Finite I-POMDP	PSPACE-complete (Papadimitriou and Tsitsiklis 1987)
Decentralized MDP	NEXP-hard (Bernstein et al. 2002)
2-player Nash equilibrium (NE)	PPAD-complete (Chen et al. 2009)
Maximum egalitarian NE (Max NE)	NP-complete (Gilboa and Zemel 1989)
Approximate max NE	NP-complete (Conitzer and Sandholm 2008)
Pure strategy Bayesian NE	NP-hard (Conitzer and Sandholm 2008)
Pure NE infinite Markov games	PSPACE-hard (Conitzer and Sandholm 2008)
Pure NE finite Markov games	NP-hard (Conitzer and Sandholm 2008)
Correlated equilibrium (CE)	P (Gilboa and Zemel 1989)
Max CE	NP-hard (Papadimitriou and Roughgarden 2008)
Logic (Sect. 5.2)	
SAT-FOL	Undecidable (Turing 1936; Church 1936)
SAT-PL	NP-complete (Cook 1971)
Validity for modal logic	PSPACE-complete (K, T, S4), NP-complete (S5) (Ladner 1977)
Multi-agent modal logic (MAML)	PSPACE-complete (Halpern and Moses 1992)
MAML + common knowledge	EXPTIME-complete (Halpern and Moses 1992)
Validity for temporal logic	PSPACE-complete (Sistla and Clarke 1985; Spaan 1993)
SAT-propositional dynamic logic	EXPTIME-complete (Fischer and Ladner 1979; Pratt 1980)
SAT-deontic STIT logic	Undecidable (Schwarzentruber and Semmling 2014)
Deontic input/output logic	NP/co-NP-hard (Sun and Robaldo 2017)
Descriptive complexity (Sect. 5.2.1)	
FOL	LH & AC^0 (Immerman 1998)
Least fixed-point FOL	P (Immerman 1982; Vardi 1982)
SO∃, SO∀ & SOL	NP, co-NP, and PH, respectively (Fagin 1974)
SOL with transitive closure	PSPACE (Immerman 1989)
Least fixed-point SOL	EXPTIME (Abiteboul et al. 1997)
Learning (Sect. 6)	No free lunch (Wolpert 1992, 1996; Schaffer 1994)

Table 3 (continued)

Problem	Results
Sample complexity for PAC-learnability	$O(\frac{VCD(\mathcal{H})+\ln\frac{1}{\delta}}{\epsilon})$(Hanneke 2016)
Proper realizable PAC	From P to NP-hard (Pitt and Valiant 1988)
Improper PAC	NP-hard (cryptographic assumptions) (Kearns and Valiant 1994)
Proper agnostic PAC	NP-hard (Feldman et al. 2012)

biases?[134] This presents an uncomfortable compromise between helpfulness and harmfulness: while massive amounts of unlabeled training data—hundreds of billions of byte-pair-encoded tokens (Brown et al. 2020)—may be required to support the helpful general-purpose text-processing capacities of a LLM, it inevitably includes undesirable behaviors and biases that can at best be inhibited (Wolf et al. 2023). Another issue is: how do we understand requirements of transparency, explainability, robustness, safety, and fairness of sufficiently advanced "black box" systems (Gunning et al. 2019; Amodei et al. 2016; Gabriel 2020; Berk et al. 2021)[135] in areas that range from medical diagnosis (Mykhailov 2021) to art (Dare et al. 2020)?

In sum, while the computational hardness of moral machine learning may be overcome—or distorted—by practice (using massive amounts of training time and data), this practice raises a host of other complicated issues that deserve attention in their own right. However, there is a key lesson that follows from our analysis: namely, the role of inductive biases and their moral justification. In some strong sense, the success of learning systems—e.g., training efficiency and predictive accuracy—seems inversely proportional to the inductive assumptions they exploit, as well as the problems of induction they introduce. I.e., for moral learning to work, we need to have a relatively clear idea of the performance measure—e.g., in terms of some predefined score, goal, or objective function—of the moral problem we want the learning system to tackle, or the morally virtuous trait we want it to exhibit. Alternatively, we may—as in the case of LLMs—hope that it already exist in the vast statistical ocean of the training data. As such, one might question whether existing moral learning systems can generate any "new" or "genuine" moral insight or reasoning, as they merely train on some given data filtered through some given inductive biases. Similar to divine command and legal positivism, it presupposes that we have an answer to the questions we seek. Thus, the problem of moral machine learning does not reside in the computational complexity of learning as such, but rather, in justifying the moral assumptions we need to exploit in order for induction to work. As elegantly put by Karl Popper (1962):

> In constructing an induction machine we, the architects of the machine, must decide *a priori* what constitutes its 'world'; what things are to be taken as similar

[134] See, e.g. Wellner (2021) for the topic of gender biases, and Liang et al. (2022) for a "holistic" evaluation of LLMs using multiple metrics and test cases.

[135] Recent work in algorithmic fairness indicates that there are inevitable trade-offs between, on the one hand, different concepts of fairness, and on the other, between fairness and accuracy (Berk et al. 2021).

> or equal; and what *kind* of 'laws' we wish the machine to be able to 'discover' in its 'world'. In other words we must build into the machine a framework determining what is relevant or interesting in its world: the machine will have its 'inborn' selection principles. The problems of similarity will have been solved for it by its makers who thus have interpreted the 'world' for the machine. (p. 48)

7 Moral tractability for minds and machines

First, we offered three possible interpretations of how to analyze the complexity of ethics based on Marr's three levels of analysis. We then proceeded to analyze a range of ethical problems for causal engines, rule-followers, and learners. The results are summarized in Table 3. Based on the surveyed results, what can computational complexity teach us about morality? In this section, we will discuss the consequences for moral machines (Sect. 7.1) and human morality (Sect. 7.2), the explanatory prospects of the Moral Tractability Thesis (Sect. 7.3), along with limitations (Sect. 7.4) and venues for future work (Sect. 7.5).

7.1 Consequences for the prospects of moral machines

What consequences do intractability results have for the prospects of moral machines? First and foremost, due to the intractability (and undecidability) stemming from combinatorics of action plans (Sect. 4.1), probabilistic causal inference (Sect. 4.2), dynamic and partially observable environments (Sect. 4.3, general rules (Sect. 5.1), strategic dynamics (Sect. 5.1.3), logic (Sect. 5.2), semantics (Sect. 5.2.3) and learning (Sect. 6.1), we can firmly conclude that perfect moral machines are impossible (i.e., given that the Extended Church–Turing Thesis is false and P $\neq$ NP is true). In many cases, suboptimal approximations of solutions are also intractable. Instead, the developers of moral machines should strive for "best possible" on the basis of constrained resources. Similar conclusions have been made—although not as formally—by Brundage (2014), Mabaso (2021), Hew (2014), Hagendorff and Danks (2022), and should be no surprise to scholars familiar with bounded rationality (Simon 1955, 1990; Rubinstein 1998; Russell and Subramanian 1994) or bounded ethicality (Bazerman and Tenbrunsel 2011; Tenbrunsel and Messick 2004). Nevertheless, the presented work should also be helpful in pinpointing the type of complexity that bounded computational agent's face in the realm of ethical decision-making, and the relevant trade-offs between optimality and feasibility it presents. It should therefore be informative for debates on artificial moral agents, as it draws the question of whether artificial moral agents are practically feasible or normatively desirable closer to the de facto dimensions of AI methods; as opposed to centering on the uniquely human capacities that existing AI systems lack (Stenseke 2023).

In a similar vein, the results should also be illuminating for the further development of artificial systems implemented in moral domains. In particular, the complexity of ethical problems highlights the intimate relationship between the cognitive capacities of agents and moral resources such as time, memory, knowledge, communication, learning, and heuristics. However, it also presents a strong implementation-variance with regard to moral resources, which potentially obfuscates any general notion of practical

moral competence. I.e., while it may be possible to identify the available resources for a particular agent in a given context, and how the resources can be effectively utilized and combined to yield competent ethical behavior, it is difficult to generalize such insights to *any* agent in *any* context. For instance, what may be considered a competent ethical decision for a social robot in a classroom differs significantly from the ethical competence required in high-speed traffic situations. The implementation-variance can be interpreted negatively: it shows that no general benchmarks can be established so as to assess the ethical performance of computational systems. More optimistically, it can also convey areas where domain- or problem-specific moral benchmarks can be established in terms of resource-dependency (e.g., given limited time or information). More directly, it points to venues where relevant benchmarks already exist: e.g., to find morally attractive equilibria in complex coordination games, minimize regret in multi-armed bandit settings, constructing tractable logics for moral reasoning, or efficient algorithms for Bayesian inference.

It also shows that a lot of work remains to be done on the moral end of machine ethics. In almost every instance, there is an uncomfortable trade-off between optimality and feasibility, and performance and efficiency; trade-offs which themselves may need normative justification. However, while computer science may have shifted towards becoming an empirical science in the advent of machine learning, our moral theories remain deeply rooted in theoretical ideals of right and wrong, which may presuppose unrealistic access to oracles of rationality. This brings out the open-ended tension between normative theory as standards of "ideal good", and normative theory as action-guiding heuristics to get suboptimal but feasible results. For instance, if our moral theories assume unrealistic computations, how can we provide a solid footing for their justification in practice? Similarly, if the tension remains unresolved, we cannot clearly determine cases whether a harmful action was due to a failure of competence, or whether it was a moral wrongdoing. It also presents challenges for moral theories: e.g., an agent that evaluates her actions according to NT_1 might be better off (according to NT_1) by following the action-decisions provided by an alternative theory NT_2.[136] Thus, the critical question is: upon what standards should we potentially revise our moral theories so as to be feasible as decision procedures with regard to the bounded and implementation-variant resources of agents? In machine contexts, we believe such open-ended issues can be fruitfully investigated under the lens of computational complexity, as it provides analytical means to measure how resources relate to formal notions of performance. More practically, it provides a venue to address what machines ought to do based on what they *can* do at all—and what they can do *effectively*—which in turn can convey the domains where computational systems can be successfully applied to make competent ethical decisions.

7.2 Consequences for human morality

More broadly, what consequences do these results have for our understanding of human morality? This ultimately depends on what one believes about the human mind, and

[136] The typical example of such a "self-effacing" theory is act utilitarianism (Parfit 1984, Sections 9 and 17), as an agent would produce more overall good by *not* following the prescriptions of act utilitarianism in practice.

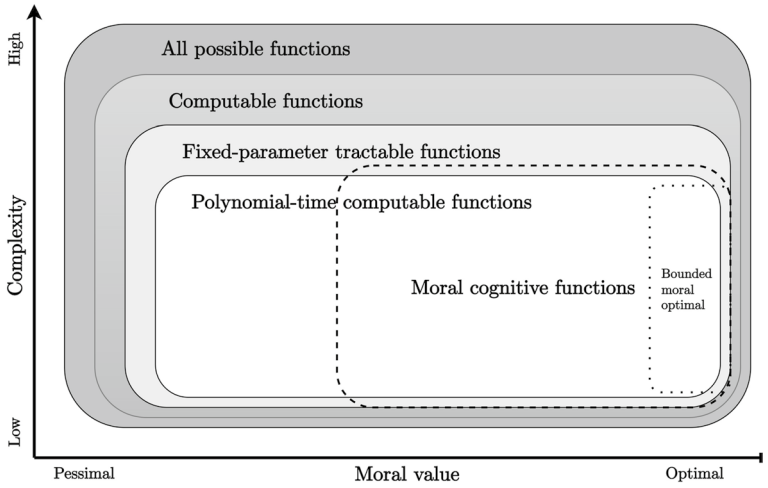

Fig. 4 The moral tractability thesis (MTT) states that the set of possible moral cognitive functions are subject to tractability constraints. More formally, this can be framed as a subset of functions that are fixed-parameter tractable for sufficiently small input parameters, which also includes the set of functions computable in polynomial time. We believe the MTT can serve (1) as a meta-ethical standard for the action-guidance of normative theory, in the sense that action-guidance should be feasible with respect to agents' resources, (2) as a guide for normative judgements and responsibility in cases where it is unclear whether an agent acted wrongfully due to a failure of cognitive constraints, (3) as an experimental paradigm in studies of human moral cognition and psychology, and (4) as a remedy to the tension between feasibility and performance in moral contexts

more particularly, whether and to what extent it is computational in nature. The ones who view humans as primarily cultural, social, and spiritual beings may find the computational perspective completely irrelevant for understanding human morality. Others, who attempt to understand human behavior in terms of cognitive capacities, may find it fruitful to assume that human cognition has at least *some* particular characteristics and constraints, which can be used to constrain the space of hypothesis. Simply put: assumptions about what the human mind can and cannot do should be informative for our understanding of the human mind. From this latter view, the step towards embracing some form of computationalism about the human mind becomes quite attractive, as it offers a smorgasbord of additional scientific tools; e.g., the use of computational architectures to model, test, and revise theories about cognition and behavior. If we take this step, computational complexity becomes an indispensable instrument, as it helps us constraint the space of possible computational theories of human cognition. This view can be captured in the *Tractable Cognition Thesis*, which states that computational models of cognitive abilities need to be computationally tractable, given some reasonable conception of tractability (Van Rooij 2008; Van Rooij et al. 2019). Conversely, if a computational theory implies intractable computations, it indicates that the theory is inadequate. A formal variant of the thesis is the *P-cognition Thesis*, which asserts that cognitive functions are constrained by polynomial time. In cognitive science, the P-Cognition thesis has explicitly been advanced as guide for computational-level theories of human cognition by Cherniak (1986), Tsotsos (1990), Levesque (1989), Frixione (2001), and is implicitly used as a constraining factor by a large group of cognitive

psychologists.[137] Furthermore, the observation that many NP-hard problems can in fact be efficiently solved for some part of the input parameter has led to the argument that P-Cognition Thesis should be replaced with the Fixed-Parameter Tractability thesis (Van Rooij 2008), which states that cognitive functions are restricted by polynomial time in the overall input size n, while allowing for superpolynomial time in some part of its input parameter (Downey and Fellows 2012). A related modeling paradigm is the concept of *resource-rational* cognition (Lieder and Griffiths 2020), which addresses cognitive modeling in terms of the optimal use of limited computational resources.

Thus, even if it remains unclear what kind of computer the human mind is (or whether it can be meaningfully captured by any model of computation), tractability considerations—in conjunction with cognitive modeling and experimental data—can work as a hypothesis that helps us carve out the space of feasible theories of cognition. Naturally, this would also include the functions that make up moral cognition. For instance, if we believe that humans perform causal, strategical, or logical reasoning to produce competent moral behavior, tractability considerations will directly serve to constrain the space of computational-level problems that underpins moral behavior. In addition, if we have reason to believe that humans perform these moral inferences in a specific way—e.g., using certain Bayesian, decision-theoretical, or logical inference techniques—a complexity analysis will help to pinpoint the relevant trade-offs between performance and feasibility; trade-offs which may directly relate to the complexity results surveyed in this paper.

7.3 Moral tractability thesis

The role of tractability in theories about moral behavior and cognition can be formulated as the *Moral Tractability Thesis* (MTT). It is a natural extension of the Tractable Cognition Thesis and states that morality—moral behavior, moral problem-solving, and moral cognition—are constrained by computational tractability (see Fig. 4), given some reasonable model of human moral cognition.[138] The MTT is a hypothesis that points in both normative and descriptive directions:

(1) *MTT as a meta-ethical standard for normative theory* MTT yields a meta-ethical point that stresses that the computational problems imposed by a moral theory also *should* be tractable with regard to the resources of an agent following the theory; i.e., to the extent the theory aims to provide any meaningful action-guidance for the agent.[139] If not, it could suggest that the moral theory imposes unrealistic demands in relation to the agent's resources, and therefore, the theory should be revised so as to constrain the space of problems the agent can be expected to solve. As such, it could help to exclude the "self-effacing" theories of normative action-guidance from the ones that respects the limited capacities of agents. More practically, it can help us to identify the situations and

[137] See Van Rooij (2008) for a detailed treatment of the Tractable Cognition Thesis and its formal variants.

[138] Here, "reasonable" is intentionally ambiguous, as there is currently no consensus of the specific model of computation that reflects the human brain. For a recent proposal, see Blum and Blum (2022).

[139] This point is the computational analogue to "ought implies can"-principles that has been proposed in moral philosophy with reference to the constraints of human psychology. One prominent example is Owen Flanagan's elaborate defense of the *The Principle of Minimal Psychological Realism*: "Make sure when constructing a moral theory or projecting a moral ideal that the character, decision processing, and behavior prescribed are possible, or are perceived to be possible, for creatures like us" (Flanagan 1993, p. 32).

contexts in which a certain form action-guidance can be expected to produce better results than competing alternatives.

(2) *MTT as a guide for normative judgements and responsibility* In situations where an agent behaves immorally, MTT can help us to answer the question whether the agent's immoral behavior was due to a failure of morality and rationality. For instance, if the computational demands are fair for some reasonable conception of human moral competence, it might point to the lack of a certain moral resource in the situation at hand (e.g., time, memory, knowledge, learning, or rationality demands). If no such absence can can be identified, it can also serve as grounds for holding the agent responsible for their immoral behavior.

(3) *MTT as an experimental paradigm* In the descriptive direction, MTT can provide a delimiting factor for existing paradigms studying human morality; from the computational modeling of human moral cognition to experimental studies in moral psychology. More precisely, MTT can be used to identify the principles and algorithms that underpin cognitive processes in moral decision-making, reveal relevant trade-offs between feasibility vs performance, and further investigate the role of resources in specific moral contexts.

(4) *MTT as a remedy to feasibility vs optimality* Another feature of MTT is that it can help to resolve open-ended tensions between feasibility and performance in moral contexts, in the sense that the latter is directly constrained by the former. For instance, if we assume that there is a fixed point which yields the optimal moral value, MTT implies that there is an action-guiding theory that yields the highest possible moral value with regard to the resources of the acting agent.[140] The same idea can be articulated for moral communities: recipes for action-guidance that produces the highest moral prosperity—e.g., joint welfare, or mutually agreed-upon moral values—with regard to the mutually *shared* resources of agents in the moral community. In other words, there may be normative action-guidance that provides the morally optimal use of bounded cognitive resources; a resource-rational moral theory (Lieder and Griffiths 2020). From a game-theoretic point of view, this can be interpreted as the optimal moral equilibrium point (e.g., maximum joint benefit) within the space of the agents' bounded resources (Fig. 4).

7.4 Limitations

There are several of gaps and limitations for the explanatory viability of moral complexity analyses in general and the MTT in particular. We will briefly address some of these along three broad questions: (i) How and to what extent is computational complexity relevant for the development and deployment of "morally competent" AI systems in real-world domains? (ii) What aspects of human morality are *not* captured by a complexity analysis? (iii) What can complexity tell us about the human use of morally informed AI systems (e.g., AI systems that are designed and used to extend or augment human moral capacities)?

(i) It should be stressed that the computational problems analyzed in this paper may have limited relevance for many real-world implementations of AI systems in moral domains. For instance, what can computational complexity, if anything, tell us about the moral behavior in domains such as autonomous driving, social assistive robotics, and natural language processing? While the overarching goal of a self-driving vehicle may be summarized

[140] More moderately, instead of an optimal, we can assume that there is a justified threshold for moral permissibility on the basis of *some* notion of moral value.

as "drive safely from point *A* to *B*", it encompasses a vast set of smaller sub-tasks and goals—e.g., path planning, adhere to traffic rules, crossing this and that intersection—for which it utilizes a cluster of distributed capacities—e.g., sensors, motors, maps, simulations—that makes up competent autonomous driving (Badue et al. 2021). Thus, a complexity analysis of an autonomous car's moral competence offers little insight unless it is based on the specific car's capacities. As noted in Sect. 6, some technical-ethical challenges that arise for LLMs may present the inverse problem of *constraining* a general capacity to produce text (e.g., so as to adhere to human principles and values). That is, instead of learning an optimal solution from the largest possible search-space—e.g., by training on vast amounts of human-generated text—the task may be to constrain or modify the search via some justified inductive bias that aligns with human values (e.g., using RLHF). Similarly, the most widely discussed ethical issues that pervade social assistive robotics—e.g., deception, dignity, trust, and recognition in Human-Robot Interaction (HRI)—are far removed from the formal notions of causal, logical, and strategic inference discussed in this work (Boada et al. 2021).

The main point is that each domain presents its distinct set of technical and ethical challenges that need to be addressed with respect to the domain's unique conditions. In turn, these challenges may have little to do with the computation of applied normative ethics, but rather, depend on some specific normative requirements that are presupposed by a certain human practice (Behdadi and Munthe 2020). However, *if* AI systems are developed to behave in accordance with normative ethics, the surveyed complexity results have an overarching relevancy for such endeavors, just as they remain central to several prominent paradigms for the advancements of computing; e.g., probabilistic inference, knowledge-systems, and learning. In a more trivial sense, even if the aim is to merely supplement AI systems with *some* capacities needed for *some* form of ethical decision-making, the behavior of such systems would also be constrained by limited computational resources. A resource-rational complexity analysis could therefore help to identify the sort of ethical problems that can be solved by machines; the ones that can be solved efficiently, where there is room for improvement, as well as pinpoint the relevant resources and trade-offs.

(ii) Another important gap that needs to be addressed is the difference between human morality and the computational form of applied normative ethics explored in this paper. It should be strongly emphasized that, although the surveyed results have direct implications for computational systems, any potential consequences for our understanding of human morality rest on speculative assumptions. As discussed in Sect. 3, ethics is an ambiguous and multifaceted concept; *what* sort of ethical problems humans actually "solve", *how* they solve them, and what they *use* to solve them (e.g., emotions, reasoning) are all open questions with many possible and elaborate answers. Similarly, while the aim of normative ethics is to find generally applicable standards of "good" and "bad", it is but a small conversation of the broader landscape that makes up ethical life. Adopting a more skeptical view, one could even claim that normative ethics—as theoretically construed in contemporary Anglophone analytical philosophy—does not have any bearing on ethical life at all (Stocker 1977), or similarly, that the dogmas of instrumental rationality—as captured in the mathematical optimization that underpins most AI research—only captures a small facet of intelligence (Pasquinelli 2020). To that end, it might be odd to imagine that humans put together ethical action-plans (or optimal salads) using exhaustive search methods (Sect. 4.1), make causal inferences using (arbitrarily) large Bayesian Networks (Sect. 4.2), compute egalitarian Nash equilibria in their strategic interactions (Sect. 5.1.5), or check behavioral norm-compliance with regards to possible worlds (Sect. 5.2.2). Humans may employ a broad range of resources in their everyday ethical life—e.g., emotions and

motivations, guilt and shame, spirituality and ideology, critical reflection and theory of mind—that are uncounted for in this analysis. Thus, the presented analysis—from computation, to algorithm, to implementation—should not stop researchers from pursuing other investigatory directions that may be fruitful for understanding the intricate and numerous forms of human morality; e.g., the embodied and felt, the shared dependencies and vulnerabilities of personal relationships, and the norms and institutions of society and culture.

Still, if one believes that cognitive tractability and MTT holds any merit, complexity can provide a guiding light into the vast intersection between normative, descriptive, and applied ethics. Even if one just accepts that computational limitations have *some* importance for *some* forms of human moral behavior, it suffices as a reason to further investigate those limitations. Furthermore, it is not unreasonable to believe that many human beings *are* able to and do—at least occasionally—follow deontological and consequentialist decision procedures, and, following virtue ethics, learn to foster the psychological dispositions that enable them to become better moral persons; although the de facto cognitive procedures that underpin these processes might look different from the ones considered in this work. To that end, it would be rather odd to view these normative theories as completely separated from the cognitive capacities of humans; for instance, it seems hard to explain the practical success or popularity of certain moral heuristics (e.g., principles or theories) unless they were applicable—decidable and tractable—for humans in moral communities (Alexander 2007). In some cases, it even seems reasonable to believe that some of these heuristics are motivated on the basis of their computational efficiency; e.g., the computational efficiency of adhering to moral rules (Sect. 5.3). Thus, while we should respect the vast gap between human and computational forms of morality, it would be unwise to exclude the possibility that there are more or less rigorous patterns in the cognitive processes that support the former, which in turn are amendable for formal investigation via the latter.

(iii) Another large area that is omitted in this paper is the integrative use of AI systems in human moral behavior. I.e., we have mainly considered the ethical behavior of computational systems in isolation, without any 'human-in-the-loop' (Wellner 2018), or any particular domain or context in mind. As such, we have ignored the integrative prospects of how AI systems can be utilized to support or augment human ethical decision-making, and the complex questions of intentionality and responsibility it opens up (Matthias 2004; Johnson and Powers 2005; Mykhailov 2023). Machines are, after all, a large cluster of computational methods that are employed to carry out the aims of its human users. In turn, this might point to a research area that can also be illuminated by computational complexity: how AI systems can foster and support moral prosperity in human practices in light of constrained resources (Vallor 2015; Giubilini and Savulescu 2018). For instance, many problems that are intractable for humans may be tractable for machines, and conversely, many problems that are intractable for machines may be tractable for machines. The guiding question is thus: on the basis of resource constraints, how can AI be developed and used so as to expand rather than constrain the space of moral reasoning (Vallor 2016)?

7.5 Future work

There are a number of interesting venues to further explore moral tractability for minds and machines beyond the ones already described. First, it should be stressed that although this work has discussed a number of complexity results relevant for moral behavior, it has circumvented an even greater amount. For most of the computational problem discussed,

there are hundreds of related results that would be relevant to consider under different conditions and assumptions. In fact, we have omitted results from entire fields of complexity theory—e.g., parameterized, communication, proof, and circuit complexity—that could yield further insights about the limitations of moral computation. Out of the discussed problems, there are two areas in particular that we believe deserves a more detailed investigation: (i) the sample complexity and regret-minimization of (moral) machine learning, and (ii) algorithmic game theory (in conjunction with algorithmic design theory). The reason for pursuing the first is that the complexity of machine learning in moral contexts remains relatively poorly understood; especially given issues such as explainability (Gunning et al. 2019), induction (Sterkenburg and Grünwald 2021), and the "paradox of deep learning" (Zhang et al. 2021). Another reason is that machine learning is the main vehicle behind the modern advancements in AI development. More urgently, learning systems are already deployed in a vast range of human practices, including areas that may involve salient forms of moral decision-making such as health care, autonomous driving, law, policing, and education.

The reason for pursuing (ii) is that game theory provides a unifying framework for the formal study of interactions, which in turn makes interactions amendable for algorithmic modeling and analysis. As such, we believe it could provide fertile synergies between historically distinct fields such as computer science, moral theory, evolutionary biology, behavioral economics, and social science. For instance, if we adopt the view that normative theories converge more than they disagree,[141] a resource-rational algorithmic game-theoretic analysis could help to identify the conditions under which certain theories are more practically viable than others. E.g., what sort of cognitive abilities and computational resources are required for a certain moral heuristic—e.g. an action-guiding normative theory—to support the highest possible moral prosperity for a community of agents? What sort of moral behaviors can be effectively computed or justified, and what behaviors can only be learned? Ideally, such investigations could not only illuminate the specific resources a computational agent need in order to be morally competent, but explain the very resource-rational rationales that underpin our most prominent ethical theories.

8 Conclusion

We have surveyed a large but far from exhaustive set of complexity results and discussed their relevance for minds and machines in the moral realm. On the one hand, it shows that being moral is hard. More precisely, if being moral involves planning, causal inference, sequential decision-making in dynamic and partially observable environments, general rules, strategic dynamics, logical reasoning, or learning in a way that is prescribed by the three dominant normative frameworks, then being moral is generally computationally intractable. On the other hand, it also identifies where we can look for more efficient decision procedures and action-guidance in the moral realm, just as it may guide us towards a better understanding of the relevant resources and trade-offs. Ultimately, we believe tractability opens up interesting interdisciplinary spaces between machine ethics, moral

[141] For instance, Parfit (2011) argues that it is erroneous to believe that there are profound disagreements between consequentialists, contractualists, and Kantians, writing: "these people are climbing the same mountain on different sides" (p. 385).

philosophy, and moral cognitive psychology, which will hopefully inspire new directions, not only in the engineering of moral machines, but in understanding the complex science of morality.

Appendix

Complexity classes

AC^0—Class of decision problems solvable by a family (one for each possible input-size) of constant-depth unlimited-fanin circuits, where the number of gates is bounded by some polynomial in the size of the input.
LH—Class of decision problems solvable by an alternating TM with a bounded number of alternations in time $O(\log n)$. See Immerman (1998) for a detailed exposition.
PL—Class of decision problems solvable by a probabilistic TM constrained by space $O(\log n)$, with an error probability $\epsilon < 1/2$ ("Probabilistic Logarithmic Space").
P—Class of decision problems solvable by a deterministic TM constrained by time $O(poly(n))$.
PP—Class of decision problems solvable by a probabilistic TM constrained by time $O(poly(n))$, with an error probability $\epsilon < 1/2$.
FTP—Class of decision problems solvable by a deterministic TM constrained by time $O(f(k)n^c)$, where f is a function that only depends on the parameter k, and c is a constant. See Downey and Fellows (2012) for the definite introduction to parameterized complexity.
NP—Class of decision problems solvable by a non-deterministic TM constrained by time $O(poly(n))$. Alternatively, class of decision problems for which "Yes"-instances are verifiable in polynomial time by a deterministic TM.
co-NP—The complement set of NP. Class of decision problems for which "No"-instances are verifiable in polynomial time by a deterministic TM.
#P—Class of function problems $f(x)$, where f is the number of accepting paths of a non-deterministic TM constrained by time $O(poly(n))$. Informally, it is the set of counting problems associated with NP; i.e., where NP decision problems ask "are there any", #P function problems asks "how many".
NP^{PP}—Class of decision problems solvable by a non-deterministic TM constrained by time $O(poly(n))$ with access to an oracle for problems in PP.
TFNP—Class of function problems solvable by a non-deterministic TM constrained by time $O(poly(n))$ where a solution is guaranteed to exist ("Total Function Non-deterministic Polynomial").
PPAD—The subclass of TFNP where functions are guaranteed to be total—i.e., a solution is guaranteed to exist—in virtue of the parity argument on directed graphs ("Polynomial Parity Arguments on Directed graphs"). See Papadimitriou (1994) for a detailed exposition.
PH—The union of classes in the polynomial hierarchy. It can be defined recursively using oracle machines: given $P = \Delta_0^P = \Sigma_0^P = \Pi_0^P$, we define $P^{\Sigma_i^P} = \Delta_{i+1}^P$, $NP^{\Sigma_i^P} = \Sigma_{i+1}^P$, and co-$NP^{\Sigma_i^P} = \Pi_{i+1}^P$, to express the union:

$$\text{PH} = \bigcup_{i=0}^{\infty} \Delta_i^{\text{P}} \cup \Sigma_i^{\text{P}} \cup \Pi_i^{\text{P}} \tag{8}$$

PSPACE—Decision problems solvable by a deterministic TM constrained by space $O(poly(n))$.
EXPTIME—Decision problems solvable by a deterministic TM constrained by time $O(2^{poly(n)})$.
NEXPTIME—Decision problems solvable by a non-deterministic TM constrained by time $O(2^{poly(n)})$. Often denoted NEXP.
EXPSPACE—Decision problems solvable by a deterministic TM constrained by space $O(2^{poly(n)})$.

Acknowledgements This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program—Humanities and Society (WASP-HS) funded by the Marianne and Marcus Wallenberg Foundation and the Marcus and Amalia Wallenberg Foundation. The author is especially grateful to Per Austrin (KTH Royal Institute of Technology) and Felix Lindner (Universität Ulm) for their insightful feedback on computational complexity. The author is also thankful to his colleagues at the Department of Philosophy and Cognitive Science at Lund University for providing helpful comments on previous versions of the paper. In particular, the author wants to acknowledge Niklas Dahl for answering countless queries about logic and semantics, Karl Samson and Gustav Stenseke Arup for input on the relationship between legality and morality, and Christian Balkenius, Björn Petersson, Ylva von Gerber, Sandra Lofs Midelf, Jiwon Kim, Alexander Velichkov, Trond Arild Tjøstheim, and Alfred Stenseke for their helpful comments. Finally, the author wants to thank participants of the Higher Seminar in Practical Philosophy (Lund University), the PhD seminar in philosophy (Lund University), the WASP-HS Winter Conference 2022, the philosophical colloquium at Georg-August-Universität, and the seminar at Institut für Künstliche Intelligenz (Universität Ulm), where earlier parts of the work were presented.

Author Contributions JS is the sole contributor of the work presented in the article.

Funding Open access funding provided by Lund University. This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program—Humanities and Society (WASP-HS) funded by the Marianne and Marcus Wallenberg Foundation and the Marcus and Amalia Wallenberg Foundation.

Declarations

Conflict of interest The author declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential Conflict of interest.

References

Aaronson S (2013) Why philosophers should care about computational complexity. Comput Tur Gödel Church Beyond 261:327

Abdelbar AM, Hedetniemi SM (1998) Approximating maps for belief networks is np-hard and other theorems. Artif Intell 102:21–38

Abel D, MacGlashan J, Littman ML (2016) Reinforcement learning as a framework for ethical decision making, In: AAAI workshop: AI, ethics, and society, Phoenix, AZ, pp 02
Abiteboul S, Vardi MY, Vianu V (1997) Fixpoint logics, relational machines, and computational complexity. J ACM (JACM) 44:30–56
Adam SP, Alexandropoulos SAN, Pardalos PM, Vrahatis MN (2019) No free lunch theorem: a review. In: Demetriou I, Pardalos P (eds) Approximation and optimization. Springer, Cham
Albrecht SV, Stone P (2018) Autonomous agents modelling other agents: a comprehensive survey and open problems. Artif Intell 258:66–95
Alexander JM (2007) The structural evolution of morality. Cambridge University Press, Cambridge
Allen C, Smit I, Wallach W (2005) Artificial morality: top-down, bottom-up, and hybrid approaches. Ethics Inf Technol 7:149–155
Amaldi E, Kann V (1998) On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems. Theor Comput Sci 209:237–260
Amodei D, Olah C, Steinhardt J, Christiano P, Schulman J, Mané D (2016) Concrete problems in AI safety. arXiv preprint arXiv:1606.06565
Anderson M, Anderson SL (2008) Ethel: toward a principled ethical eldercare system. In: AAAI fall symposium: AI in eldercare: new solutions to old problems. AAAI Press, Arlington, pp 4–11
Anderson M, Anderson SL (2011) Machine ethics. Cambridge University Press, Cambridge
Angluin D, Laird P (1988) Learning from noisy examples. Mach Learn 2:343–370
Annas J (2011) Intelligent virtue. Oxford University Press, Oxford
Anscombe GEM (1958) Modern moral philosophy. Philosophy 33:1–19
Applebaum B, Barak B, Xiao D (2008) On basing lower-bounds for learning on worst-case assumptions. In: 2008 49th Annual IEEE symposium on foundations of computer science. IEEE. pp 211–220
Aristotle (2000) Aristotle: nicomachean ethics. Cambridge texts in the history of philosophy. Cambridge University Press, Cambridge. https://doi.org/10.1017/CBO9780511802058
Arkin RC (2007) Governing lethal behavior: embedding ethics in a hybrid deliberative/hybrid robot architecture. Report. Report GIT-GVU-07-11. Georgia Institute of Technology's GVU, Atlanta
Arkoudas K, Bringsjord S, Bello P (2005) Toward ethical robots via mechanized deontic logic. In: AAAI fall symposium on machine ethics. The AAAI Press Menlo Park, pp 17–23
Armstrong S, (2015) Motivated value selection for artificial agents. In: AAAI workshop: AI and ethics. AAAI Press, Palo Alto
Arpit D, Jastrzebski S, Ballas N, Krueger D, Bengio E, Kanwal MS, Maharaj T, Fischer A, Courville A, Bengio Y et al (2017) A closer look at memorization in deep networks. In: International conference on machine learning, PMLR, pp 233–242
Arrow KJ (1950) A difficulty in the concept of social welfare. J Polit Econ 58:328–346
Asimov I (1942) Runaround. astounding science. Fiction 29:94–103
Åström KJ (1965) Optimal control of Markov processes with incomplete state information. J Math Anal Appl 10:174–205
Auer P, Cesa-Bianchi N, Freund Y, Schapire RE (1995) Gambling in a rigged casino: the adversarial multi-armed bandit problem. In: Proceedings of IEEE 36th annual foundations of computer science. IEEE, pp 322–331
Aumann RJ (1974) Subjectivity and correlation in randomized strategies. J Math Econ 1:67–96
Aumann RJ (1987) Correlated equilibrium as an expression of Bayesian rationality. Econom J Econom Soc 55:1–18
Aumann RJ (2016) 16. acceptable points in general cooperative n-person games. In: Contributions to the theory of games (AM-40), vol IV. Princeton University Press, pp 287–324
Austrin P, Braverman M, Chlamtáč E (2013) Inapproximability of np-complete variants of Nash equilibrium. Theory Comput 9:117–142
Axelrod R, Hamilton WD (1981) The evolution of cooperation. Science 211:1390–1396
Azad-Manjiri M (2014) A new architecture for making moral agents based on c4. 5 decision tree algorithm. Int J Inf Technol Comput Sci (IJITCS) 6:50–57
Bäckström C, Nebel B (1995) Complexity results for SAS+ planning. Comput Intell 11:625–655
Badue C, Guidolini R, Carneiro RV, Azevedo P, Cardoso VB, Forechi A, Jesus L, Berriel R, Paixao TM, Mutz F et al (2021) Self-driving cars: a survey. Expert Syst Appl 165:113816
Baker CL, Tenenbaum JB, Saxe RR (2007) Goal inference as inverse planning. In: Proceedings of the annual meeting of the cognitive science society
Balbiani P, Herzig A, Troquard N (2008) Alternative axiomatics and complexity of deliberative STIT theories. J Philos Log 37:387–406
Bales RE (1971) Act-utilitarianism: Account of right-making characteristics or decision-making procedure? Am Philos Q 8:257–265

Barsalou LW (1999) Perceptual symbol systems. Behav Brain Sci 22:577–660
Bauer WA (2020) Virtuous vs. utilitarian artificial moral agents. AI Soc 35:263–271
Bazerman MH, Tenbrunsel AE (2011) Blind spots. In: Blind spots. Princeton University Press
Beall JC (2007) Revenge of the liar: new essays on the paradox. OUP, Oxford
Behdadi D, Munthe C (2020) A normative approach to artificial moral agency. Minds Mach 30:195–218
Ben-David S, Eiron N, Long PM (2003) On the difficulty of approximately maximizing agreements. J Comput Syst Sci 66:496–514
Bennett JF (1976) Linguistic behaviour
Bentham J (1961) 1789. Doubleday, An introduction to the principles of morals and legislation. Garden City
Berberich N, Diepold K (2018) The virtuous machine-old ethics for new technology? arXiv preprint arXiv:1806.10322
Berk R, Heidari H, Jabbari S, Kearns M, Roth A (2021) Fairness in criminal justice risk assessments: the state of the art. Sociol Methods Res 50:3–44
Berner C, Brockman G, Chan B, Cheung V, Debiak P, Dennison C, Farhi D, Fischer Q, Hashme S, Hesse C (2019) Dota 2 with large scale deep reinforcement learning. arXiv preprint arXiv:1912.06680
Bernstein E, Vazirani U (1997) Quantum complexity theory. SIAM J Comput 26:1411–1473
Bernstein DS, Givan R, Immerman N, Zilberstein S (2002) The complexity of decentralized control of Markov decision processes. Math Oper Res 27:819–840
Bicchieri C (2005) The grammar of society: the nature and dynamics of social norms. Cambridge University Press, Cambridge
Binder K, Heermann D, Roelofs L, Mallinckrodt AJ, McKay S (1993) Monte Carlo simulation in statistical physics. Comput Phys 7:156–157
Blackburn S (1992) Through thick and thin. In: Proceedings of the Aristotelian society, pp 284–99
Blackburn S (1998) Ruling passions. Oxford University Press, Oxford
Block N (2019) What is wrong with the no-report paradigm and how to fix it. Trends Cogn Sci 23:1003–1013
Blum L, Blum M (2022) A theory of consciousness from a theoretical computer science perspective: insights from the conscious Turing machine. Proc Natl Acad Sci 119:e2115934119
Blumer A, Ehrenfeucht A, Haussler D, Warmuth MK (1989) Learnability and the Vapnik–Chervonenkis dimension. J ACM (JACM) 36:929–965
Boada JP, Maestre BR, Genís CT (2021) The ethical issues of social assistive robotics: a critical literature review. Technol Soc 67:101726
Bodlaender HL (1994) A tourist guide through treewidth. Acta Cybern 11:1
Bouneffouf D, Rish I (2019) A survey on practical applications of multi-armed and contextual bandits. arXiv preprint arXiv:1904.10040
Brandenburger A, Dekel E (1993) Hierarchies of beliefs and common knowledge. J Econ Theory 59:189–198
Brandom R (1994) Making it explicit: reasoning, representing, and discursive commitment. Harvard University Press, Cambridge
Brandom R (2006) Kantian lessons about mind, meaning, and rationality. South J Philos 44:49–71
Brandt RB (1979) A theory of the good and the right
Brentano F (1874) Psychology from an empirical standpoint
Bringsjord S, Taylor J (2012) The divine-command approach to robot ethics. In: Lin P, Abney K, Bekey GA (eds) The ethical and social implications of robotics, robot ethics. MIT Press, Cambridge, pp 85–108
Broome J (1987) Utilitarianism and expected utility. J Philos 84:405–422
Brown T, Mann B, Ryder N, Subbiah M, Kaplan JD, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A et al (2020) Language models are few-shot learners. Adv Neural Inf Process Syst 33:1877–1901
Brożek B, Janik B (2019) Can artificial intelligences be moral agents? New Ideas Psychol 54:101–106
Brundage M (2014) Limitations and risks of machine ethics. J Exp Theor Artif Intell 26:355–372
Bubeck S, Wang T, Viswanathan N (2013) Multiple identifications in multi-armed bandits. In: International conference on machine learning, PMLR, pp 258–265
Bylander T (1991) Complexity results for planning. In: IJCAI, pp 274–279
Bylander T (1994) The computational complexity of propositional strips planning. Artif Intell 69:165–204
Cai Y, Papadimitriou C (2014) Simultaneous Bayesian auctions and computational complexity. In: Proceedings of the fifteenth ACM conference on economics and computation, pp 895–910
Camerer CF, Ho TH, Chong JK (2004) A cognitive hierarchy model of games. Q J Econ 119:861–898
Campbell MS, Marsland TA (1983) A comparison of minimax tree search algorithms. Artif Intell 20:347–367

Capraro V, Rand DG (2018) Do the right thing: experimental evidence that preferences for moral behavior, rather than equity or efficiency per se, drive human prosociality. Forthcoming in Judgment and Decision Making
Carnap R (1947) Meaning and necessity: a study in semantics and modal logic
Casebeer WD (2003) Moral cognition and its neural constituents. Nat Rev Neurosci 4:840–846
Casella G, Berger RL (2021) Statistical inference. Cengage Learning, Boston
Cassandra AR, Kaelbling LP, Littman ML (1994) Acting optimally in partially observable stochastic domains. In: AAAI, pp 1023–1028
Cervantes JA, López S, Rodríguez LF, Cervantes S, Cervantes F, Ramos F (2020) Artificial moral agents: a survey of the current status. Sci Eng Ethics 26:501–532
Cesa-Bianchi N, Lugosi G (2006) Prediction, learning, and games. Cambridge University Press, Cambridge
Chalmers DJ (1997) The conscious mind: in search of a fundamental theory. Oxford Paperbacks
Chatterjee K, Chmelik M, Tracol M (2016) What is decidable about partially observable Markov decision processes with ω-regular objectives. J Comput Syst Sci 82:878–911
Chen X, Deng X, Teng SH (2009) Settling the complexity of computing two-player Nash equilibria. J ACM (JACM) 56:1–57
Cherniak C (1986) Minimal rationality. MIT Press, Cambridge
Church A (1936) A note on the entscheidungs problem. J Symb Log 1:40–41
Cloos C (2005) The utilibot project: an autonomous mobile robot based on utilitarianism. In: Machine ethics: papers from the 2005 AAAI fall symposium. AAAI Press, Menlo Park, pp 38–45
Cobham A (1965) The intrinsic computational difficulty of functions
Coeckelbergh M (2020) AI ethics. MIT Press, Cambridge
Cofino AS, Cano R, Sordo C, Gutierrez JM (2002) Bayesian networks for probabilistic weather prediction. In: 15th Eureopean conference on artificial intelligence (ECAI), Citeseer
Coleman KG (2001) Android arete: toward a virtue ethic for computational agents. Ethics Inf Technol 3:247–265
Conitzer V, Sandholm T (2008) New complexity results about Nash equilibria. Games Econ Behav 63:621–641
Conway P, Gawronski B (2013) Deontological and utilitarian inclinations in moral decision making: a process dissociation approach. J Personal Soc Psychol 104:216
Cook SA (1971) The complexity of theorem-proving procedures.In: Proceedings of the third annual ACM symposium on theory of computing, pp 151–158
Cooper GF (1990) The computational complexity of probabilistic inference using Bayesian belief networks. Artif Intell 42:393–405
Copeland BJ (2020) The Church–Turing thesis. In: Zalta EN (ed) The Stanford encyclopedia of philosophy, Summer 2020. Metaphysics Research Lab, Stanford University, Stanford
Courville AC, Daw ND, Touretzky DS (2006) Bayesian theories of conditioning in a changing world. Trends Cogn Sci 10:294–300
Crawford VP, Sobel J (1982) Strategic information transmission. Econom J Econom Soc 50:1431–1451
Crisp R, Slote MA (1997) Virtue ethics. Oxford University Press, Oxford
Cummings R, Ligett K, Radhakrishnan J, Roth A, Wu ZS (2016) Coordination complexity: small information coordinating large populations. In: Proceedings of the 2016 ACM conference on innovations in theoretical computer science, pp 281–290
Dagum P, Luby M (1993) Approximating probabilistic inference in Bayesian belief networks is np-hard. Artif Intell 60:141–153
Dagum P, Luby M (1997) An optimal approximation algorithm for Bayesian inference. Artif Intell 93:1–27
Dahl N (2022) A fixed-point problem for theories of meaning. Synthese 200:1–15
Daniely A, Linial N, Shalev-Shwartz S (2014) From average case complexity to improper learning complexity. In: Proceedings of the forty-sixth annual ACM symposium on theory of computing, pp 441–448
Dare Z, Brinkmann H, Rosenberg R (2020) Testing a calibration-free eye tracker prototype at the Kunsthistorisches museum in Vienna. J Eye Move Res **13**
Daskalakis C, Mehta A, Papadimitriou C (2006) A note on approximate Nash equilibria. In: International workshop on internet and network economics. Springer, pp 297–306
de Campos CP (2020) Almost no news on the complexity of map in Bayesian networks. In: International conference on probabilistic graphical models, PMLR, pp 149–160
De Giacomo G, Massacci F (2000) Combining deduction and model checking into tableaux and algorithms for converse-PDL. Inf Comput 162:117–137
de Weerd H, Verbrugge R, Verheij B (2013) How much does it help to know what she knows you know? an agent-based simulation study. Artif Intell 199:67–92

de Weerd H, Verbrugge R, Verheij B (2017) Negotiating with other minds: the role of recursive theory of mind in negotiation with incomplete information. Auton Agents Multi-Agent Syst 31:250–287
Degrave J, Felici F, Buchli J, Neunert M, Tracey B, Carpanese F, Ewalds T, Hafner R, Abdolmaleki A, de Las Casas D et al (2022) Magnetic control of tokamak plasmas through deep reinforcement learning. Nature 602:414–419
Dehghani M, Tomai E, Forbus KD, Klenk M (2008a) An integrated reasoning approach to moral decision-making. In: AAAI, pp 1280–1286
Dehghani M, Tomai E, Klenk M (2008b) An integrated reasoning approach to moral decision-making. In: Proceedings of the twenty-third AAAI conference on artificial intelligence. AAAI Press, Chicago, pp 1280–1286
Doshi P, Gmytrasiewicz PJ (2009) Monte Carlo sampling methods for approximating interactive POMDPs. J Artif Intell Res 34:297–337
Downey RG, Fellows MR (2012) Parameterized complexity. Springer, Berlin
Ehrenfeucht A, Haussler D, Kearns M, Valiant L (1989) A general lower bound on the number of examples needed for learning. Inf Comput 82:247–261
Fagin R (1974) Generalized first-order spectra and polynomial-time recognizable sets. Complex Comput 7:43–73
Fehr E, Fischbacher U (2004) Social norms and human cooperation. Trends Cogn Sci 8:185–190. https://doi.org/10.1016/j.tics.2004.02.007
Fehr E, Gächter S (2000) Cooperation and punishment in public goods experiments. Am Econ Rev 90:980–994
Feigenbaum J, Shenker S (2004) Distributed algorithmic mechanism design: recent results and future directions, In: Current trends in theoretical computer science: the challenge of the new century vol 1: algorithms and complexity vol 2: formal models and semantics. World Scientific, pp 403–434
Feldman V, Guruswami V, Raghavendra P, Wu Y (2012) Agnostic learning of monomials by halfspaces is hard. SIAM J Comput 41:1558–1590
FeldmanHall O, Mobbs D (2015) A neural network for moral decision making. In: Toga AW, Lieberman MD (eds) Brain mapping: an encyclopedic reference. Elsevier, Oxford
Fellows MR (2002) Parameterized complexity: the main ideas and connections to practical computing. In: Experimental algorithmics. Springer, pp 51–77
Fischer MJ, Ladner RE (1979) Propositional dynamic logic of regular programs. J Comput Syst Sci 18:194–211
Flanagan O (1993) Varieties of moral personality: ethics and psychological realism. Harvard University Press, Cambridge
Floridi L, Sanders JW (2004) On the morality of artificial agents. Minds Mach 14:349–379
Foot P (1967) The problem of abortion and the doctrine of the double effect. Oxford Rev 5:5–15
Foster DP, Young HP (2001) On the impossibility of predicting the behavior of rational agents. Proc Natl Acad Sci 98:12848–12853
Friston K (2010) The free-energy principle: A unified brain theory? Nat Rev Neurosci 11:127–138
Frixione M (2001) Tractable competence. Minds Mach 11:379–397
Furbach U, Schon C, Stolzenburg F (2014) Automated reasoning in deontic logic. In: International workshop on multi-disciplinary trends in artificial intelligence. Springer, pp 57–68
Gabbay D, Horty J, Parent X, van der Meyden R, van der Torre L (2013) Handbook of deontic logic and normative systems
Gabriel I (2020) Artificial intelligence, values, and alignment. Minds Mach 30:411–437
Garcıa J, Fernández F (2015) A comprehensive survey on safe reinforcement learning. J Mach Learn Res 16:1437–1480
Garey MR, Johnson DS (1979) Computers and intractability, vol 174. Freeman, San Francisco
Gauthier D (1987) Morals by agreement. Clarendon Press, Oxford
Geertz C et al (1973) The interpretation of cultures, vol 5019. Basic Books, New York
Gelman A, Carlin JB, Stern HS, Dunson DB, Vehtari A, Rubin DB (2013) Bayesian data analysis
Gilboa I, Zemel E (1989) Nash and correlated equilibria: some complexity considerations. Games Econ Behav 1:80–93
Gill J (1977) Computational complexity of probabilistic Turing machines. SIAM J Comput 6:675–695
Giraud-Carrier C, Provost F (2005) Toward a justification of meta-learning: Is the no free lunch theorem a show-stopper. In: Proceedings of the ICML-2005 workshop on meta-learning, pp 12–19
Giubilini A, Savulescu J (2018) The artificial moral advisor. the "ideal observer'' meets artificial intelligence. Philos Technol 31:169–188
Gmytrasiewicz PJ, Doshi P (2005) A framework for sequential planning in multi-agent settings. J Artif Intell Res 24:49–79

Gödel K (1930) Uber die vollständigkeit des logikkalküls. Ph.D. thesis. Ph.D. dissertation, University of Vienna
Gödel K (1931) Über formal unentscheidbare sätze der principia mathematica und verwandter systeme i. Mon Math Phys 38:173–198
Goodie AS, Doshi P, Young DL (2012) Levels of theory-of-mind reasoning in competitive games. J Behav Decis Mak 25:95–108
Governatori G, Olivieri F, Rotolo A, Scannapieco S (2013) Computing strong and weak permissions in defeasible logic. J Philos Log 42:799–829
Govindarajulu NS, Bringsjord S (2017) On automating the doctrine of double effect. In: Proceedings of the 26th international joint conference on artificial intelligence, pp 4722–4730
Govindarajulu NS, Bringsjord S, Ghosh R, Sarathy V (2019) Toward the engineering of virtuous machines. In: Proceedings of the 2019 AAAI/ACM conference on AI, ethics, and society, pp 29–35
Greene JD (2007) Why are VMPFC patients more utilitarian? A dual-process theory of moral judgment explains. Trends Cogn Sci 11:322–323
Greene JD (2014) Beyond point-and-shoot morality: Why cognitive (neuro) science matters for ethics. Ethics 124:695–726
Greene JD, Morelli SA, Lowenberg K, Nystrom LE, Cohen JD (2008) Cognitive load selectively interferes with utilitarian moral judgment. Cognition 107:1144–1154
Grice HP (1975) Logic and conversation. In: Speech acts. Brill, pp 41–58
Griffiths TL, Tenenbaum JB (2005) Structure and strength in causal induction. Cogn Psychol 51:334–384
Griffiths L, Kemp T, Tenenbaum CBJ (2008) Bayesian models of cognition
Guha S, Munagala K, Shi P (2010) Approximation algorithms for restless bandit problems. J ACM (JACM) 58:1–50
Gunning D, Stefik M, Choi J, Miller T, Stumpf S, Yang GZ (2019) Xai-explainable artificial intelligence. Sci Robot 4:eaay7120
Hagendorff T, Danks D (2022) Ethical and methodological challenges in building morally informed AI systems. AI Ethics 3:1–14
Haigh T, Priestley M, Rope C (2014) Los Alamos bets on Eniac: Nuclear monte Carlo simulations, 1947–1948. IEEE Ann Hist Comput 36:42–63
Hajek B (2015) Random processes for engineers. Cambridge University Press, Cambridge
Halpern JY, Moses Y (1992) A guide to completeness and complexity for modal logics of knowledge and belief. Artif Intell 54:319–379
Hanneke S (2016) The optimal sample complexity of PAC learning. J Mach Learn Res 17:1319–1333
Hansen J (2008) Prioritized conditional imperatives: problems and a new proposal. Auton Agents Multi-Agent Syst 17:11–35
Hare RM (1952) The language of morals
Hare RM (1981) Moral thinking: its levels, method, and point. Clarendon Press, Oxford
Harnad S (1990) The symbol grounding problem. Physica D Nonlinear Phenom 42:335–346
Harsanyi JC (1967) Games with incomplete information played by "Bayesian'' players, i–iii part i. the basic model. Manag Sci 14:159–182
Hart S, Mas-Colell A (2000) A simple adaptive procedure leading to correlated equilibrium. Econometrica 68:1127–1150
Hart S, Schmeidler D (1989) Existence of correlated equilibria. Math Oper Res 14:18–25
Håstad J (2001) Some optimal inapproximability results. J ACM (JACM) 48:798–859
Hazan E, Krauthgamer R (2011) How hard is it to approximate the best Nash equilibrium? SIAM J Comput 40:79–91
Hedden T, Zhang J (2002) What do you think i think you think?: Strategic reasoning in matrix games. Cognition 85:1–36
Hellström T (2013) On the moral responsibility of military robots. Ethics Inf Technol 15:99–107
Herken R (1995) The universal Turing machine a half-century survey. Springer, Berlin
Herzig A, Schwarzentruber F (2008) Properties of logics of individual and group agency. Adv Modal Log 7:133–149
Hester T, Vecerik M, Pietquin O, Lanctot M, Schaul T, Piot B, Horgan D, Quan J, Sendonaris A, Osband I et al (2018) Deep q-learning from demonstrations. In: Proceedings of the AAAI conference on artificial intelligence
Heuer L, Orland A (2019) Cooperation in the prisoner's dilemma: an experimental comparison between pure and mixed strategies. R Soc Open Sci 6:182142
Hew PC (2014) Artificial moral agents are infeasible with foreseeable technologies. Ethics Inf Technol 16:197–206

Himma KE (2009) Artificial agency, consciousness, and the criteria for moral agency: What properties must an artificial agent have to be a moral agent? Ethics Inf Technol 11:19–29
Hobbes T (1651) Leviathan
Hofstadter D (2002) Staring Emmy straight in the eye-and doing. In: Creativity, cognition, and knowledge: an interaction, p 67
Hohwy J (2013) The predictive mind. OUP, Oxford
Holt CA, Roth AE (2004) The Nash equilibrium: a perspective. Proc Natl Acad Sci 101:3999–4002
Hooker B (2016) Rule consequentialism. In: Zalta EN (ed) The Stanford encyclopedia of philosophy, Winter 2016. Metaphysics Research Lab, Stanford University, Standford
Hopkins M, Kane DM, Lovett S, Mahajan G (2022) Realizable learning is all you need. In: Conference on learning theory, PMLR, pp 3015–3069
Horty JF (2001) Agency and deontic logic. Oxford University Press, Oxford
Horty JF (2012) Reasons as defaults. OUP, Oxford
Howard D, Muntean I (2017) Artificial moral cognition: moral functionalism and autonomous moral agency. Springer, Berlin, pp 121–159
Hume D (2003) A treatise of human nature. Courier Corporation, North Chelmsford
Hummert S, Bohl K, Basanta D, Deutsch A, Werner S, Theißen G, Schroeter A, Schuster S (2014) Evolutionary game theory: cells as players. Mol BioSyst 10:3044–3065
Hursthouse R (1999) On virtue ethics. OUP, Oxford
Hurwicz L, Reiter S (2006) Designing economic mechanisms. Cambridge University Press, Cambridge
Huttegger SM (2007) Evolution and the explanation of meaning. Philos Sci 74:1–27
Immerman N (1982) Relational queries computable in polynomial time. In: Proceedings of the fourteenth annual ACM symposium on theory of computing, pp 147–152
Immerman N (1989) Descriptive and computational complexity. In: Hartmanis J (ed) Computational complexity theory, proceedings of the symposium on in applied mathematics, pp 75–91
Immerman N (1998) Descriptive complexity. Springer, Berlin
Johnson VR (1990) The declaration of the rights of man and of citizens of 1789, the reign of terror, and the revolutionary tribunal of Paris. BC Int'l Comp L Rev 13:1
Johnson DG, Powers TM (2005) Computer systems and responsibility: a normative look at technological complexity. Ethics Inf Technol 7:99–107
Jørgensen J (1937) Imperatives and logic. Erkenntnis 7:288–296
Kahneman D (2011) Thinking, fast and slow. Macmillan, New York
Kakade SM (2003) On the sample complexity of reinforcement learning. University of London, London
Kant I (2013) Groundwork of the metaphysics of morals. Routledge, Boca Raton
Karp RM (1972) Reducibility among combinatorial problems. In: Complexity of computer computations. Springer, pp 85–103
Kawaguchi K, Kaelbling LP, Bengio Y (2017) Generalization in deep learning. arXiv preprint arXiv:1710.05468
Kaye P, Laflamme R, Mosca M (2006) An introduction to quantum computing. OUP, Oxford
Kearns M, Valiant L (1994) Cryptographic limitations on learning Boolean formulae and finite automata. J ACM (JACM) 41:67–95
Kearns MJ, Schapire RE, Sellie LM (1992) Toward efficient agnostic learning. In: Proceedings of the fifth annual workshop on computational learning theory, pp 341–352
Keller GB, Mrsic-Flogel TD (2018) Predictive processing: a canonical cortical computation. Neuron 100:424–435
Kemp C, Tenenbaum JB (2008) The discovery of structural form. Proc Natl Acad Sci 105:10687–10692
Kensing F, Blomberg J (1998) Participatory design: issues and concerns. Comput Support Coop Work (CSCW) 7:167–185
Khachiyan LG (1979) A polynomial algorithm in linear programming. In: Doklady Akademii Nauk. Russian Academy of Sciences, pp 1093–1096
Kifer D, Machanavajjhala A (2011) No free lunch in data privacy. In: Proceedings of the 2011 ACM SIGMOD international conference on management of data, pp 193–204
Koenig S, Simmons RG (1993) Complexity analysis of real-time reinforcement learning. In: AAAI, pp 99–107
Kohlberg L, Hersh RH (1977) Moral development: a review of the theory. Theory Pract 16:53–59
Körding KP, Wolpert DM (2006) Bayesian decision theory in sensorimotor control. Trends Cogn Sci 10:319–326
Koutsoupias E, Papadimitriou C (2009) Worst-case equilibria. Comput Sci Rev 3:65–69
Kreps DM, Milgrom P, Roberts J, Wilson R (1982) Rational cooperation in the finitely repeated prisoners' dilemma. J Econ Theory 27:245–252

Kripke SA (1963) Semantical analysis of modal logic I normal modal propositional calculi. Math Log Q 9:67–96
Kripke S (1976) Outline of a theory of truth. J Philos 72:690–716
Kwisthout J (2011) Most probable explanations in Bayesian networks: complexity and tractability. Int J Approx Reason 52:1452–1469
Kwisthout J, Wareham T, Van Rooij I (2011) Bayesian intractability is not an ailment that approximation can cure. Cogn Sci 35:779–784
Ladner RE (1977) The computational complexity of provability in systems of modal propositional logic. SIAM J Comput 6:467–480
Langford J, Zhang T (2007) The epoch-greedy algorithm for contextual multi-armed bandits. Adv Neural Inf Process Syst 20:96–103
Lattimore T, Hutter M (2013) No free lunch versus Occam's razor in supervised learning. In: Algorithmic probability and friends. Bayesian prediction and artificial intelligence. Springer, pp 223–235
Lau HC (2007) A higher order Bayesian decision theory of consciousness. Prog Brain Res 168:35–48
Leben D (2018) Ethics for robots: How to design a moral algorithm. Routledge, London
Lee J, Bahri Y, Novak R, Schoenholz SS, Pennington J, Sohl-Dickstein J (2017) Deep neural networks as gaussian processes. arXiv preprint arXiv:1711.00165
Lenman J (2000) Consequentialism and cluelessness. Philos Public Affairs 29:342–370
Levesque HJ (1989) Logic and the complexity of reasoning. In: Philosophical logic and artificial intelligence. Springer, pp 73–107
Lewis D (1969) Convention. Harvard University Press, Cambridge
Lewis D (1975) Languages and language
Lewis D (1979) Scorekeeping in a language game. In: Semantics from different points of view. Springer, pp 172–187
Liang P, Bommasani R, Lee T, Tsipras D, Soylu D, Yasunaga M, Zhang Y, Narayanan D, Wu Y, Kumar A et al (2022) Holistic evaluation of language models. arXiv preprint arXiv:2211.09110
Lieder F, Griffiths TL (2020) Resource-rational analysis: understanding human cognition as the optimal use of limited computational resources. Behav Brain Sci 43:e1
Lindner F, Mattmüller R, Nebel B (2020) Evaluation of the moral permissibility of action plans. Artif Intell 287:103350. https://doi.org/10.1016/j.artint.2020.103350
Lipton RJ, Markakis E, Mehta A (2003) Playing large games using simple strategies. In: Proceedings of the 4th ACM conference on electronic commerce, pp 36–41
Littman ML (1996) Algorithms for sequential decision-making. Brown University, Providence
Littman ML, Goldsmith J, Mundhenk M (1998) The computational complexity of probabilistic planning. J Artif Intell Res 9:1–36
Locatelli A, Gutzeit M, Carpentier A (2016) An optimal algorithm for the thresholding bandit problem. In: International conference on machine learning, PMLR, pp 1690–1698
Lorenz EN (1963) Deterministic nonperiodic flow. J Atmos Sci 20:130–141
Lucas PJ, de Bruijn NC, Schurink K, Hoepelman A (2000) A probabilistic and decision-theoretic approach to the management of infectious disease at the ICU. Artif Intell Med 19:251–279
Luce RD, Raiffa H (1989) Games and decisions: introduction and critical survey. Courier Corporation, North Chelmsford
Mabaso BA (2021) Computationally rational agents can be moral agents. Ethics Inf Technol 23:137–145
Madani O, Hanks S, Condon A (2003) On the undecidability of probabilistic planning and related stochastic optimization problems. Artif Intell 147:5–34
Makinson D, Van Der Torre L (2000) Input/output logics. J Philos Log 29:383–408
Malle B, Scheutz M, Austerweil J (2017a) Networks of social and moral norms in human and robot agents. In: A world with robots. intelligent systems, control and automation: science and engineering, vol 84. Springer, Cham, pp 3–17
Malle BF, Scheutz M, Austerweil JL (2017b) Networks of social and moral norms in human and robot agents. In: A world with robots. Springer, pp 3–17
Marr D (1977) Artificial intelligence-a personal view. Artif Intell 9:37–48
Marr D (1981) Vision: a computational investigation into the human representation and processing of visual information. W. H. Freeman, San Fransisco
Matthias A (2004) The responsibility gap: ascribing responsibility for the actions of learning automata. Ethics Inf Technol 6:175–183
Mertens JF, Zamir S (1985) Formulation of Bayesian analysis for games with incomplete information. Int J Game Theory 14:1–29
Meyer JJC et al (1988) A different approach to deontic logic: deontic logic viewed as a variant of dynamic logic. Notre Dame J Form Log 29:109–136

Mill JS (1998a) On liberty and other essays. Oxford University Press, Oxford
Mill JS (1998b) Utilitarianism. Oxford University Press, New York
Moka-Mubelo W (2017) Law and morality. In: Reconciling law and morality in human rights discourse. Springer, pp 51–88
Mundhenk M, Goldsmith J, Lusena C, Allender E (2000) Complexity of finite-horizon Markov decision process problems. J ACM (JACM) 47:681–720
Mykhailov D (2021) A moral analysis of intelligent decision-support systems in diagnostics through the lens of Luciano Floridi's information ethics. Hum Affairs 31:149–164
Mykhailov D (2023) Philosophical inquiry into computer intentionality: machine learning and value sensitive design. Hum Affairs 33:115–127
Narveson J (2001) The libertarian idea. Broadview Press, Peterborough
Nash J (1951) Non-cooperative games. Ann Math 54:286–295
Nash JF et al (1950) Equilibrium points in n-person games. Proc Natl Acad Sci 36:48–49
Neumann vJ (1928) Zur theorie der gesellschaftsspiele. Math Ann 100:295–320
Newen A, De Bruin L, Gallagher S (2018) The Oxford handbook of 4E cognition. Oxford University Press, Oxford
Neyman A (1985) Bounded complexity justifies cooperation in the finitely repeated prisoners' dilemma. Econ Lett 19:227–229
Neyshabur B, Bhojanapalli S, McAllester D, Srebro N (2017) Exploring generalization in deep learning. Adv Neural Inf Process Syst 30:1–10
Ng AY, Russell SJ (2000) Algorithms for inverse reinforcement learning. In: ICML, p 2
Niedermeier R (2006) Invitation to fixed-parameter algorithms, vol 31. OUP, Oxford
Nievergelt J, Gasser R, Mäser F, Wirth C (1995) All the needles in a haystack: Can exhaustive search overcome combinatorial chaos? Springer, Berlin, Heidelberg, pp 254–274. https://doi.org/10.1007/BFb0015248
Nisan N, Ronen A (1999) Algorithmic mechanism design. In: Proceedings of the thirty-first annual ACM symposium on theory of computing, pp 129–140
Nowak MA (2006) Five rules for the evolution of cooperation. Science 314:1560–1563
Nussbaum MC (1988) Non-relative virtues: an Aristotelian approach. Midwest Stud Philos 13:32–53
Oaksford M, Chater N (2001) The probabilistic approach to human reasoning. Trends Cogn Sci 5:349–357
Papadimitriou CH (1994) On the complexity of the parity argument and other inefficient proofs of existence. J Comput Syst Sci 48:498–532
Papadimitriou CH, Roughgarden T (2008) Computing correlated equilibria in multi-player games. J ACM (JACM) 55:1–29
Papadimitriou CH, Tsitsiklis JN (1987) The complexity of Markov decision processes. Math Oper Res 12:441–450
Papadimitriou CH, Tsitsiklis JN (1994) The complexity of optimal queueing network control. In: Proceedings of IEEE 9th annual conference on structure in complexity theory. IEEE, pp 318–322
Parberry I, Garey MR, Meyer A (1994) Circuit complexity and neural networks. MIT press, Cambridge
Parfit D (1984) Reasons and persons. OUP, Oxford
Parfit D (2011) On what matters, vol 1. Oxford University Press, Oxford
Park JD, Darwiche A (2004) Complexity results and approximation strategies for map explanations. J Artif Intell Res 21:101–133
Pasquinelli M (2020) How a machine learns and fails—a grammar of error for artificial intelligence. Spheres
Pearl J (1985) Bayesian networks: a model of self-activated memory for evidential reasoning. In: Proceedings of the 7th conference of the cognitive science society. University of California, Irvine, pp 15–17
Pearl J (1987) Evidential reasoning using stochastic simulation of causal models. Artif Intell 32:245–257
Pearl J (2022) Reverend Bayes on inference engines: a distributed hierarchical approach. In: Probabilistic and causal inference: the works of Judea Pearl, pp 129–138
Pereira LM, Saptawijaya A (2009) Modelling morality with prospective logic. Int J Reason Based Intell Syst 1:209–221
Pitt L, Valiant LG (1988) Computational limitations on learning from examples. J ACM (JACM) 35:965–984
Pnueli A (1977) The temporal logic of programs. In: 18th annual symposium on foundations of computer science (SFCS 1977). IEEE, pp 46–57
Pontier M, Hoorn J (2012) Toward machines that behave ethically better than humans do. In: Proceedings of the annual meeting of the cognitive science society
Popper KR (1962) Philosophy of science: conjectures and refutations.The growth of scientific knowledge, vol 140. Basic Books, New York, p 1962

Powers TM (2006) Prospects for a Kantian machine. IEEE Intell Syst 21:46–51
Pratt VR (1976) Semantical considerations on Floyd–Hoare logic. In: 17th annual symposium on foundations of computer science (SFCS 1976). IEEE, pp 109–121
Pratt VR (1980) A near-optimal method for reasoning about action. J Comput Syst Sci 20:231–254
Purves D, Jenkins R, Strawser BJ (2015) Autonomous machines, moral judgment, and acting for the right reasons. Ethic Theory Moral Pract 18:851–872
Putnam H (2004) The collapse of the fact/value dichotomy and other essays. Harvard University Press, Cambridge
Radner R (1986) Can bounded rationality resolve the prisoner's dilemma. Essays in honor of Gerard Debreu, pp 387–399
Rathnasabapathy B, Doshi P, Gmytrasiewicz P (2006) Exact solutions of interactive POMDPs using behavioral equivalence. In: Proceedings of the fifth international joint conference on autonomous agents and multiagent systems, pp 1025–1032
Rawls JB (1971) A theory of justice
Rawls J (1980) Kantian constructivism in moral theory. J Philos 77:515–572
Reinikainen J (2005) The golden rule and the requirement of universalizability. J Value Inq 39:155
Rest JR, Narvaez D, Thoma SJ, Bebeau MJ (1999) Dit2: devising and testing a revised instrument of moral judgment. J Educ Psychol 91:644
Reynolds C (2005) On the computational complexity of action evaluations. In: 6th International conference of computer ethics: philosophical enquiry (University of Twente, Enschede, The Netherlands, 2005), Citeseer
Rosser B (1936) Extensions of some theorems of gödel and church. J Symb Log 1:87–91
Roth D (1996) On the hardness of approximate reasoning. Artif Intell 82:273–302
Roughgarden T (2005) Selfish routing and the price of anarchy. MIT press, Cambridge
Roughgarden T, Tardos É (2002) How bad is selfish routing? J ACM (JACM) 49:236–259
Rubinstein A (1998) Modeling bounded rationality. MIT press, Cambridge
Russell SJ, Subramanian D (1994) Provably bounded-optimal agents. J Artif Intell Res 2:575–609
Scanlon TM (2000) What we owe to each other. Harvard University Press, Cambridge
Scarselli F, Tsoi AC (1998) Universal approximation using feedforward neural networks: a survey of some existing methods, and some new results. Neural Netw 11:15–37
Schaffer C (1994) A conservation law for generalization performance. In: Machine learning proceedings 1994. Elsevier, pp 259–265
Schelling TC (1960) The strategy of conflict: with a new preface by the author. Harvard University Press, Cambridge
Schiffer SR (1972) Meaning
Schnoebelen P (2002) The complexity of temporal logic model checking. Adv Modal Log 4:35
Schroeder M (2010) Being for: evaluating the semantic program of expressivism. OUP, Oxford
Schurz G (2017) No free lunch theorem, inductive skepticism, and the optimality of meta-induction. Philos Sci 84:825–839
Schwarzentruber F, Semmling C (2014) Stit is dangerously undecidable. In: ECAI
Searle JR (1980) Minds, brains, and programs. Behav Brain Sci 3:417–424
Searle JR (1992) The rediscovery of the mind. MIT press, Cambridge
Sen A (1979) Utilitarianism and welfarism. J Philos 76:463–489
Serfozo RF (1979) An equivalence between continuous and discrete time Markov decision processes. Oper Res 27:616–620
Sergot M (1998) Normative positions. Norms Log Inf Syst 49:289–308
Shim J, Arkin R, Pettinatti M (2017) An intervening ethical governor for a robot mediator in patient–caregiver relationship: implementation and evaluation. In: 2017 IEEE international conference on robotics and automation (ICRA). IEEE, New York, USA, pp 2936–2942
Shimony SE (1994) Finding maps for belief networks is np-hard. Artif Intell 68:399–410
Sidgwick H (2019) The methods of ethics. Good Press, Glasgow
Silver D, Hubert T, Schrittwieser J, Antonoglou I, Lai M, Guez A, Lanctot M, Sifre L, Kumaran D, Graepel T et al (2018) A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. Science 362:1140–1144
Silver D, Singh S, Precup D, Sutton RS (2021) Reward is enough. Artif Intell 299:103535
Simon HA (1955) A behavioral model of rational choice. Q J Econ 69:99–118
Simon HA (1990) Bounded rationality. In: Utility and probability. Springer, pp 15–18
Singer MG (2002) The ideal of a rational morality: philosophical compositions. Oxford University Press, Oxford
Singer P (2011) Practical ethics. Cambridge University Press, Cambridge

Sinnott-Armstrong W (2021) Consequentialism. In: Zalta EN (ed) The Stanford encyclopedia of philosophy, Fall 2021. Metaphysics Research Lab, Stanford University, Stanford
Sistla AP, Clarke EM (1985) The complexity of propositional linear temporal logics. J ACM (JACM) 32:733–749
Skyrms B (2004) The stag hunt and the evolution of social structure. Cambridge University Press, Cambridge
Skyrms B (2010) Signals: evolution, learning, and information. OUP, Oxford
Slivkins A (2019) Introduction to multi-armed bandits. arXiv preprint arXiv:1904.07272
Sloane NJA (2022) Entry a007526 in the on-line encyclopedia of integer sequences. https://oeis.org/A007526
Smart JJC (1956) Extreme and restricted utilitarianism. Philos Q (1950-) 6:344–354
Smith JM, Price GR (1973) The logic of animal conflict. Nature 246:15–18. https://doi.org/10.1038/246015a0
Sobel DM, Kirkham NZ (2006) Blickets and babies: the development of causal reasoning in toddlers and infants. Dev Psychol 42:1103
Spaan E (1993) The complexity of propositional tense logics. In: Diamonds and defaults. Springer, pp 287–307
Spaan E (2016) Complexity of modal logics. Ph.D. thesis. University of Amsterdam
Stace WT (1937) The concept of morals
Stenseke J (2021) Artificial virtuous agents: from theory to machine implementation. AI Soc. https://doi.org/10.1007/s00146-021-01325-7
Stenseke J (2022a) Artificial virtuous agents in a multi-agent tragedy of the commons. AI Soc. https://doi.org/10.1007/s00146-022-01569-x
Stenseke J (2022b) Interdisciplinary confusion and resolution in the context of moral machines. Sci Eng Ethics 28:1–17
Stenseke J (2023) The use and abuse of normative ethics for moral machines. In: Social robots in social institutions. IOS Press, pp 155–164
Stenseke J, Balkenius C (2022) Assessing the time efficiency of ethical algorithms. In: CEUR workshop proceedings, CEUR-WS
Sterkenburg TF, Grünwald PD (2021) The no-free-lunch theorems of supervised learning. Synthese 199:9979–10015
Stevenson CL (1937) The emotive meaning of ethical terms. Mind 46:14–31
Steyvers M, Tenenbaum JB, Wagenmakers EJ, Blum B (2003) Inferring causal networks from observations and interventions. Cogn Sci 27:453–489
Stocker M (1977) The schizophrenia of modern ethical theories. J Philos 73:453–466
Sun X, Robaldo L (2017) On the complexity of input/output logic. J Appl Log 25:69–88
Taddeo M, Floridi L (2005) Solving the symbol grounding problem: a critical review of fifteen years of research. J Exp Theor Artif Intell 17:419–445
Tarski A (1944) The semantic conception of truth: and the foundations of semantics. Philos Phenom Res 4:341–376
Tenbrunsel AE, Messick DM (2004) Ethical fading: the role of self-deception in unethical behavior. Soc Justice Res 17:223–236
Tenenbaum JB, Griffiths TL, Kemp C (2006) Theory-based Bayesian models of inductive learning and reasoning. Trends Cogn Sci 10:309–318
Thornton SM, Pan S, Erlien SM, Gerdes JC (2016) Incorporating ethical considerations into automated vehicle control. IEEE Trans Intell Transp Syst 18:1429–1439
Tolmeijer S, Kneer M, Sarasua C, Christen M, Bernstein A (2020) Implementations in machine ethics: a survey. ACM Comput Surv (CSUR) 53:1–38
Trémolière B, Neys WD, Bonnefon JF (2012) Mortality salience and morality: thinking about death makes people less utilitarian. Cognition 124:379–384. https://doi.org/10.1016/j.cognition.2012.05.011
Tsotsos JK (1990) Analyzing vision at the complexity level. Behav Brain Sci 13:423–445
Tufiş M, Ganascia JG (2015) Grafting norms onto the BDI agent model. In: A construction manual for robots' ethical systems. Springer, pp 119–133
Turing AM (1936) On computable numbers, with an application to the entscheidungs problem. J Math 58:5
Ullmann-Margalit E (2015) The emergence of norms. OUP, Oxford
Urbano A, Vila JE (2002) Computational complexity and communication: coordination in two-player games. Econometrica 70:1893–1927
Valiant LG (1984) A theory of the learnable. Commun ACM 27:1134–1142

Vallor S (2015) Moral deskilling and upskilling in a new machine age: reflections on the ambiguous future of character. Philos Technol 28:107–124
Vallor S (2016) Technology and the virtues: a philosophical guide to a future worth wanting. Oxford University Press, Oxford
Van Der Meyden R (1996) The dynamic logic of permission. J Log Comput 6:465–479
Van Rooij I (2008) The tractable cognition thesis. Cogn Sci 32:939–984
Van Rooij I, Blokpoel M, Kwisthout J, Wareham T (2019) Cognition and intractability: a guide to classical and parameterized complexity analysis. Cambridge University Press, Cambridge
Vanmarcke E (2010) Random fields: analysis and synthesis. World Scientific, Singapore
Vapnik V (1999) The nature of statistical learning theory. Springer, Berlin
Vapnik V, Chervonenkis A (1974) Theory of pattern recognition
Vapnik VN, Chervonenkis AY (2015) On the uniform convergence of relative frequencies of events to their probabilities. In: Measures of complexity. Springer, pp 11–30
Vardi MY (1982) The complexity of relational query languages. In: Proceedings of the fourteenth annual acm symposium on theory of computing, pp 137–146
Vardi MY (1997) Why is modal logic so robustly decidable? Technical report
Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. Adv Neural Inf Process Syst, vol 30
Vollmer H, Schnoor I, Schnoor H, Schneider T, Bauland M (2009) The complexity of generalized satisfiability for linear temporal logic. Log Methods Comput Sci 5:48–62
Von Luxburg U, Schölkopf B (2011) Statistical learning theory: models, concepts, and results. In: Handbook of the history of logic, vol 10. Elsevier, Amsterdam, pp 651–706
Von Neumann J, Morgenstern O (1947) Theory of games and economic behavior, 2nd rev
Von Wright GH (1951) Deontic logic. Mind 60:1–15
Vul E, Goodman N, Griffiths TL, Tenenbaum JB (2014) One and done? Optimal decisions from very few samples. Cogn Sci 38:599–637
Wallach W, Allen C (2008) Moral machines: teaching robots right from wrong. Oxford University Press, Oxford
Wattles J (1996) The golden rule. Oxford University Press, Oxford
Wellner G (2018) From cellphones to machine learning. A shift in the role of the user in algorithmic writing. In: Romele A, Terrone E (eds) Towards a philosophy of digital media. Palgrave Macmillan, Cham, pp 205–224
Wellner G (2021) I-algorithm-dataset: mapping the solutions to gender bias in AI. In: Büssers J, Faulhaber A, Raboldt M, Wiesner R (eds) Gendered configurations of humans and machines: interdisciplinary contributions, pp 79–97
Whitehead SD (1991) A complexity analysis of cooperative mechanisms in reinforcement learning. In: AAAI, pp 607–613
Whittle P (1988) Restless bandits: activity allocation in a changing world. J Appl Probab 25:287–298
Wiegel V, van den Berg J (2009) Combining moral theory, modal logic and mas to create well-behaving artificial agents. Int J Soc Robot 1:233–242
Williams B (2006) Ethics and the limits of philosophy. Routledge, Boca Raton
Williamson DP, Shmoys DB (2011) The design of approximation algorithms. Cambridge University Press, Cambridge
Wittgenstein L (2010) Philosophical investigations. Wiley, Hoboken
Wolf Y, Wies N, Levine Y, Shashua A (2023) Fundamental limitations of alignment in large language models. arXiv preprint arXiv:2304.11082
Wolpert DH (1992) On the connection between in-sample testing and generalization error. Complex Syst 6:47
Wolpert DH (1996) The lack of a priori distinctions between learning algorithms. Neural Comput 8:1341–1390
Wolpert DH (2002) The supervised learning no-free-lunch theorems. In: Roy R, Koppen M, Ovaska S, Furuhashi T, Hoffmann F (eds) Soft Computing and Industry. Springer, London, pp 25–42
Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. IEEE Trans Evol Comput 1:67–82
Xu M (1998) Axioms for deliberative STIT. J Philos Log 27:505–552
Yi SKM, Steyvers M, Lee MD, Dry MJ (2012) The wisdom of the crowd in combinatorial problems. Cogn Sci 36:452–470
Yoshida W, Dolan RJ, Friston KJ (2008) Game theory of mind. PLoS Comput Biol 4:e1000254
Zhang NL, Poole D (1996) Exploiting causal independence in Bayesian network inference. J Arti Intell Res 5:301–328

Zhang C, Bengio S, Hardt M, Recht B, Vinyals O (2021) Understanding deep learning (still) requires rethinking generalization. Commun ACM 64:107–115

Ziegler DM, Stiennon N, Wu J, Brown TB, Radford A, Amodei D, Christiano P, Irving G (2019) Fine-tuning language models from human preferences. arXiv preprint arXiv:1909.08593

Zinkevich M, Johanson M, Bowling M, Piccione C (2007) Regret minimization in games with incomplete information. Adv Neural Inf Process Syst, vol 20