

Bounded Rationality in Learning, Perception, Decision-Making, and Stochastic Games

Panagiotis Tsiotras

Abstract Current algorithms that try to emulate “human-like” decision-making and perception capabilities for autonomous systems are computationally very demanding. We propose to utilize bounded-rationality ideas for generating suitable hierarchical abstractions to handle demanding tasks under time and other resource constraints, when exact optimality/rationality may be elusive. These abstractions can be utilized in order to focus the available computational resources on the relevant part of the problem in a top-down, task-specific manner. The bounded rational paradigm for decision-making can also be extended to the case of stochastic games formulated in terms of competitive Markov decision processes in order to account for the presence of competing/adversarial agents.

1 The Autonomy Challenge

The science of autonomy (broadly interpreted here as a mix of control, artificial intelligence, machine learning, and robotics) has shown great promise in recent years, mainly owing to the success of statistical learning methods in achieving impressive results in a variety of pattern recognition and classification tasks. Nonetheless, most current algorithms for autonomous systems fail to address critical aspects of the problem that are ever present in many real-life applications: a) paucity of training data; b) limited computational resources; c) presence of adversarial agents. There is, therefore, a great need to better understand the way autonomous systems make decisions under the above constraints, especially if they are used to close perception-action (PA) feedback loops for fast “human-like” decision-making.

Future control and decision-making systems will have to strike a balance between *guaranteed performance* in the presence of uncertainty (a topic that is central

Panagiotis Tsiotras, David and Andrew Lewis Chair Professor
School Aerospace Engineering, Institute for Robotics and Intelligent Machines, Georgia Institute of Technology, Atlanta, GA 30332 e-mail: tsiotras@gatech.edu

to control theory), useful and *relevant information extraction* from incomplete or scarce data (a topic of estimation and information theory), dealing with *adversarial agents* (a topic of game theory), while, at the same time, ensure reduced *algorithmic complexity* so as to enable real-time (re)planning and learning (which is better addressed by computer science). This confluence of control, information theory, and computer/computational science and engineering is necessary for developing scalable computational methods for joint perception and planning in order to enable an autonomous agent extract task-relevant information from a multitude of data sources in a timely manner.

The Case of Actionable Data

Autonomous systems, supported by recent advances in artificial intelligence (AI) and machine learning (ML) – as well as the proliferation of new powerful computer hardware architectures – are poised to transform the current scientific paradigm for problem-solving by replacing many traditional model-based methodologies with data-intensive alternatives. Having access to a huge amount of data to sieve through, these computationally-intensive algorithms are capable of finding patterns that may elude human observers. Despite their great success however, these algorithms still suffer from unpredictability and often exhibit “child-like” behavior. They may confuse correlation for causality and may find “patterns” when none exist, becoming confused and perplexed, unable to come up with the correct answer even in very simple scenarios where the answer would have been clear to a human observer [52, 64]. This tendency of “theoretical statistical accuracy” but “demonstrated fragility in practice” makes these algorithms currently unsuitable for many safety-critical systems.

The main observation that may explain the current predicament with these data-driven algorithms is the realization that collected data is not necessarily information or knowledge. Extracting *actionable* information from data – and, similarly, deciding how to best represent this information – is the crux of the problem. So the key issue is not the absence of enough data, but rather the quality of the data and how to make sense out of it, especially if an adversary is capable of actively manipulating the data.

Not surprisingly, the answer to this fundamental question, namely, extracting the relevant information from the collected measurements, has been a long-standing open problem in the control, estimation, and information theory communities. A large part of control theory is devoted to constructing observers and estimators that recover the state from scarce measurements (e.g., output feedback), essentially reconstructing the whole history of the process, regardless of whether this history is needed for the control task. Today we find ourselves in a different landscape. Sensor technology has progressed so rapidly that we have ended up at the other extreme: there is too-much available data from a myriad of sources at various resolutions. And while the answer of finding the “relevant information” in the measurements was considered difficult

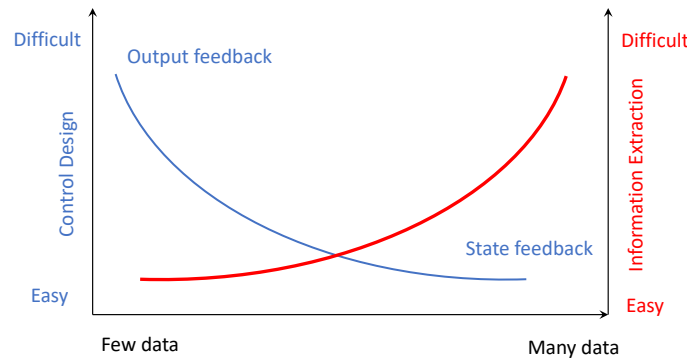


Fig. 1: The difficulty of high-confidence, robust control design and decision-making remains the same as it was many years ago, but for quite the opposite reasons. While before much of the effort was focused on finding the state of the system from few measurements (e.g., output feedback), today most of the difficulty lies in trying to extract meaningful information from a deluge of data.

many years ago because of lack of sufficient data (as in the output feedback case), it still remains difficult today, but for exactly the opposite reason: the availability of too much data (“data deluge”); see Figure 1.

The Curse of Optimality

Current models of decision-making follow the von Neumann-Morgenstern maximum expected utility (MEU) dictum [51], which assumes the existence of a utility function, the maximization of which is solely responsible for the ensuing actions by the agent. The main issue with optimal decision-making is the fact that finding *optimal* actions is very difficult, since it typically requires to search over *all* available alternatives, a problem that requires a prohibitive amount of computational resources, which have not been taken into account *during problem formulation*.

An elegant theory (namely, dynamic programming [11, 9]) exists to solve optimal decision-making, but its application in practice is stymied by the computational explosion resulting from the increased dimensionality of the search space. Most often than not, agents have limited attention, time and other resources (or they are just lazy) to come up with what is best for them. Furthermore, in many contexts what is “optimal” is not even known, so attempting to carefully optimize the (wrong) objective would be counter-productive. For instance, in planning motion through (or exploring) an unknown environment, one can try to maximize the expected reward, the computation of which requires a prior of the (yet to be seen) scene. Thus, the current plan is as good as the prior, which is unfalsifiable, and the expected reward is just that, i.e., “expected.” In other words, we claim that the often stated “curse of

dimensionality” bottleneck in decision-making is somewhat misleading, as it is not necessarily the dimensionality of the problem that defies trackable answers, but rather the fact that one tends to seek *the* optimal solution among a myriad of alternatives. By seeking approximate or suboptimal solutions, the problem dimensionality becomes secondary and the attention is turned to remedying the *curse of optimality*, instead.

2 How to Move Forward

We argue that the current data-intensive paradigm of perception and decision-making in robotics (and autonomous systems in general) that tends to operate on all collected data is neither relevant nor necessary for problems involving high-performance robots operating in real-time and at high-speed, in uncertain and complex environments, demonstrating “human-like” intelligence and agility. This point of view stems from the following key observations.

First, current, purely “computational” models of perception – especially those based solely on vision cues – essentially serve as surrogates of full-state feedback estimators in order to achieve specific control objectives, thus enforcing an artificial separation of perception and control. This separation – deeply rooted in the community by its wide applicability for the stabilization of linear systems (“separation principle”) – is questionable for most real-life problems involving autonomous agents operating in complex and unstructured environments, where information gathering (perception/sensing) is tightly coupled with motion (planning/control).

Second, human cognitive models provide strong evidence that the manner by which human (especially expert) agents perceive and (re)act to environmental stimuli, and process information, is quite different than the way most current perception techniques deal with this problem. For instance, because of experience, and prior familiarity of how the world “works”, a human has already a preconceived notion of “what lies ahead” and has a good compact representation of the environment. She certainly does not “reconstruct” the whole environment at every instant of time, and her approach to closing the perception/action loop has a significant anticipatory (non-causal) component [12].

Despite the great progress witnessed recently by data-driven approaches to learning (fueled mainly by convolutional/deep neural networks (CNNs)), these approaches are of limited applicability in capturing such priors or non-causal characteristics that are crucial for safety-critical control tasks. Traditional machine-learning approaches need a huge amount of (typically labelled) data which are often not available in practice. In addition, these learning algorithms are based on “similarity” measures of historically collected data; as a result, they are primarily suitable for interpolation but not for extrapolation. This reliance solely on data (while by-passing the guidelines of physical laws) often leads to inefficient and “brittle” learning [52].

Successful operation of autonomous agents in complex (and even adversarial) environments will require a novel approach to sensing, perception, and communication (and their interaction) that merge data-based with model-based methods in order to

enable continuous and reliable operation with limited computation resources and/or bandwidth (Figure 2).

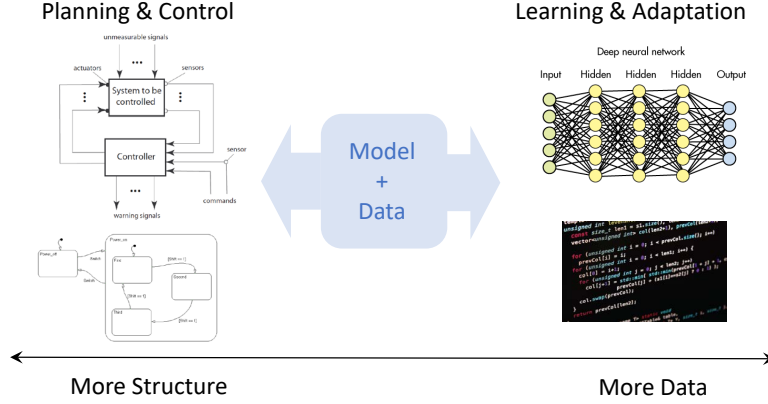


Fig. 2: The development of computationally efficient and resilient communication/learning/action loops to enable autonomous decision-making and control depends on the ability to devise suitable methodologies to merge data with models, depending on the amount of prior knowledge.

Drawing on the cognitive hierarchies of the human brain, we seek to develop algorithms that are aware of the (limited) available computational resources of the agent/decision-maker. The proposed algorithms are able to utilize prior information (e.g., from physics-based models) to operate on *actionable data* in a *timely* manner. The limitations of the available resources give rise to hierarchical abstractions on the action space which can be utilized to solve the problem at the “right” level of granularity, and thus avoid wasting valuable resources by reasoning and acting at levels of fidelity that are beyond what is required for the problem at hand.

The approach borrows from bounded rationality theory, according to which human beings tend to weight the benefits of reaching the “best” or “most informed” decision, versus the pitfalls associated with the time, energy and effort needed to collect all the information to reach that decision.

Bounded Rationality for Human-Like Decision-Making

Many surveys and laboratory experiments indicate that people often fail to conform to some of the basic tenets of rational (i.e., MEU) decision theory [54]. This is especially true when the environment is uncertain and/or complex, the task is challenging and is dominated by nonlinear phenomena, when the time is constrained, etc. As a result, decision-theory researchers, especially in economics [59, 4, 49, 57], have

proposed an alternative paradigm for rational behavior, one that is commensurate with the available knowledge and resources. According to this bounded rationality paradigm, human beings tend to weight the benefits of reaching the “best” or “most informed” decision, versus the pitfalls associated with the time, energy, and effort needed to collect all the information to reach that decision. Bounded rationality thus aims at formalizing the limited (*e.g., bounded*) *rationality decisions and actions* an intelligent agent is expected to take in those cases [30]. A bounded rational agent takes actions that meet the minimum requirements necessary to achieve a particular goal (i.e., those that meet the agent’s *aspiration level* [70]).

Another reason one may want to follow a bounded rational paradigm for decision-making stems from the desire to design autonomous robotic agents that can be trusted by humans. This implies that the robotic agent generates actions that are predictable/legible (and not necessarily optimal!) by the human. This point of view predicates that the intelligent agent needs to model the bounded-rational decision-making of the human (in essence inferring what constitutes predictable behavior for the human). Bounded rationality theory offers a convenient framework to formulate human-machine interaction.

Instead of focusing on maximizing expected utility (as in standard MDP formulations) the bounded rational approach suggests *maximizing expected free utility (or free energy)*, a quantity that encapsulates in a precise manner the effect of information and computational constraints of the decision-making process. This framework also intrinsically generates, *for free*, a *hierarchy of abstractions* for decision-making brought about by the severity of information-processing costs, with each abstraction tailored to the “right” level of granularity.

Hierarchical Abstractions for Scalability

A hallmark of human intelligence is arguably the ability to abstract, organize acquired knowledge, and process collected information in a hierarchical manner [45, 21, 31, 65]. The ability to organize collected data in hierarchies is so ingrained in the human brain that we take it for granted. Such hierarchies make it possible for us humans to make quick decisions even in complex situations we have never encountered before. Hierarchies can be utilized to focus the available resources in a top-down, task-specific manner [44, 27, 49]. Granted, these decisions may not be optimal but are made fast, respond to the most urgent or consequential stimuli, and are “good-enough” for the current situation. By moving up and down the hierarchy allows one to depart from pure optimality formulations and investigate (sub)optimal bounded-rational solutions.

Hierarchies are also a powerful source of heuristics [73] that can be used to shortcut exact computation of complex quantities. Most importantly, hierarchies are enablers of *attention* mechanisms. One often life-critical aspect of operation under strong time pressure is to be able to rapidly and efficiently focus computational resources onto the most relevant or important information in the environment, in

the context of the current situation. In primates, *attention acts as a fast heuristic* to guide the organism towards the most important sensory stimuli (e.g., obstacles in one's path, landmarks or other guiding signs, other objects/individuals). Unexpected events (or surprises) are dealt with quickly by focusing attention to the data and events at the right level of (temporal and spatial) granularity that is most relevant to the task(s) at hand [23, 42, 14]. In either case, attention provides a mechanism to shape or select incoming information under strict timing constraints. It is therefore clear that, at least in humans, attention does play a key role in identifying the “actionable information” in order to reduce computational complexity.

The idea of using hierarchical abstractions has been previously used by the author to develop efficient solutions to path-planning problems subject to computational constraints [19, 68, 18]. In those references hierarchies of multi-resolution abstractions were created to speed-up re-planning. Here we wish to generate abstractions in a more principled manner, as the result of a variational principle that captures the (limited) available computational resources of the agent. While in those references these abstractions were imposed externally (and perhaps rather axiomatically) by the designer, here they arise intrinsically and naturally as part of the problem formulation and the agent's capabilities.

3 Sequential Decision-Making Subject to Resource Constraints

Over the past several years considerable effort has been devoted to developing a framework that accounts for the available computational constraints during decision-making [27, 55, 44, 56, 49]. The idea is to account for the cost of computation not after the fact, as is traditionally the case, but from the very beginning, as part of the problem formulation. This stream of research has led to the development of *information-constrained MDPs* [44, 32, 27] that utilize concepts from information theory in order to model the computational limitations of the agent [53, 56]. The approach has also strong connections with the thermodynamic point of view of the theory of computation [10, 24, 55]. The key idea is to look the information flow $U \rightarrow X \rightarrow Y$ from the input to the output (random) variables as a suitable communication channel [67], and formulate the problem in the space of state (or output) distributions over actions.

The introduction of the added constraints (computational and otherwise) to the decision process is meant to model an agent that does not have access to unlimited resources, and hence will select actions that are not optimal in the traditional sense of maximizing value or utility, but rather in the sense of maximizing (expected) value or utility *for a given processing cost* [49, 67, 56]. Such a decision-maker, who alters its behavior based on its resources, is said to be a *bounded rational agent* [49, 59, 15, 40].

In this work, we apply this framework to develop decision-making algorithms for planning problems while also accounting for the information processing costs of computing the optimal actions for stochastic environments modeled in terms

of Markov Decision Processes (MDPs). This resource-limited MDP problem is formally posed as a constrained optimization problem, where the objective is to maximize the value function subject to a constraint that penalizes the difference between the resulting policy and a known (computationally cheap) prior policy. As it will be shown later on, this optimization problem involves the mutual information between states and control actions and leads to a direct analogy of the amount of information an action carries about the state. A perfectly rational decision-maker will solve the optimization problem producing a policy that is *state specific* since it has the resources to choose specific actions for each state. On the other hand, a bounded rational, resource-constrained agent does not have this luxury and hence prefers policies that are relatively generic and *state agnostic*. This classification of more informative versus less informative states leads in a natural way to a series of state space *abstractions*. Such abstractions can be utilized in order to focus the available computational resources in a top-down, task-specific manner. The details of the approach are discussed in greater detail next. Before proceeding with the details however, it is worth reviewing some of the basic theory of MDPs. The next section also help us establish the notation used in the rest of this work.

Standard Markov Decision Processes

We consider sequential decision-making problems in the presence of uncertainty. Markov Decision Processes (MDP) is the standard framework for optimal sequential decision-making under uncertainty [63]. We will use x_t to denote the system state and u_t the control input at time t , where $t = 0, 1, 2, \dots$. The system state is an element of the state set $\mathcal{X}$ (i.e., $x_t \in \mathcal{X}$) and the control input is selected from the set of admissible inputs $\mathcal{U}$ which, in general, depends on x_t so that $u_t \in \mathcal{U} = \mathcal{U}(x_t)$. The system dynamics is modeled via a collection of transition probabilities $P(x_{t+1}|x_t, u_t)$, which represent the probability that the agent, being at state x_t and selecting input u_t at time step t , will find itself at state x_{t+1} at the next instance of time. Each action results in a reward $R(x_{t+1}, x_t, u_t)$ that the agent receives for selecting input u_t while at state x_t and at time t and transitioning to state x_{t+1} at time $t + 1$.

The objective of the infinite-time MDP is to find a policy which maximizes the future discounted expected reward. By policy we mean a map that, for each state x_t , provides a probability distribution over actions, $\pi(u_t|x_t)$. Since we will restrict the discussion to infinite horizon problems, we assume that our policy is stationary, that is, π is independent of t [11].

More formally, we seek to maximize the objective function

$$V^\pi(x_0) \triangleq \lim_{T \rightarrow \infty} \mathbb{E}_T \left[\sum_{t=0}^{T-1} \gamma^t R(x_{t+1}, x_t, u_t) \right], \quad (1)$$

where $\gamma \in [0, 1)$ is the *discount factor*, x_0 is the initial state and u_t is selected according to the provided policy π at each time step. Equation (1) can be inter-

puted as the cost-to-go provided the system starts at state x_0 and executes the stationary policy π . The expectation in equation (1) is over all future trajectories $\tau = (x_0, u_0, x_1, u_1, \dots, x_T)$ of length T starting at x_0 . Under the Markov property assumption the probability of trajectory τ is given by

$$\Pr(u_0, x_1, u_1, x_2, \dots, x_T | x_0) = \prod_{t=0}^T \pi(u_t | x_t) P(x_{t+1} | x_t, u_t). \quad (2)$$

The objective is then to find the optimal distribution over admissible inputs at each state, in order to maximize the objective function in (1). Introducing the shorthand notation $x = x_t$, $u = u_t$, and $x' = x_{t+1}$, expanding (1) and making use of (2), we obtain

$$V^\pi(x) = \sum_{u \in \mathcal{U}} \pi(u|x) \sum_{x' \in \mathcal{X}} P(x'|x, u) \left[R(x', x, u) + \gamma V^\pi(x') \right]. \quad (3)$$

The optimal value function is

$$V^*(x) = \max_{\pi} V^\pi(x), \quad (4)$$

which can be computed by solving the *Bellman equation*

$$V^*(x) = \max_{u \in \mathcal{U}} \sum_{x' \in \mathcal{X}} P(x'|x, u) \left[R(x', x, u) + \gamma V^*(x') \right], \quad (5)$$

with the optimal policy given by $\pi^* = \operatorname{argmax}_{\pi} V^\pi(x)$.

We may also define the state-action value function (or Q-function) for any policy π as [63]

$$Q^\pi(x, u) \triangleq \sum_{x' \in \mathcal{X}} P(x'|x, u) \left[R(x', x, u) + \gamma V^\pi(x') \right]. \quad (6)$$

Similarly, the optimal state-action value function is defined as

$$Q^*(x, u) \triangleq \sum_{x' \in \mathcal{X}} P(x'|x, u) \left[R(x', x, u) + \gamma V^*(x') \right]. \quad (7)$$

In the case of a perfectly rational decision-maker, the agent will deterministically select the optimal action that maximizes equation (7). That is, the agent will act greedily with respect to $Q^*(x, u)$ by selecting $u^*(x) = \operatorname{argmax}_u Q^*(x, u)$. In this work, we distinguish between a general stochastic policy and one that is deterministic, by denoting optimal deterministic policies as $\Gamma^*(u|x)$ and all other optimal policies as $\pi^*(u|x)$. Then, a perfectly rational agent will choose the policy $\Gamma^*(u|x) = \delta(u - u^*(x))$, where $u^*(x) = \operatorname{argmax}_u Q^*(x, u)$.

Information-Limited Markov Decision Processes

The problem formulation presented above does not consider the “effort” required to find the optimal policy $\Gamma^*(u|x)$, which maximizes (1) [56, 67]. In order to model the effort required to obtain the (global) maximum, we provide the agent with a prior choice distribution $\rho(u|x)$ and then limit the amount by which the posterior policy $\pi(u|x)$ is permitted to differ from $\rho(u|x)$ [27, 67, 56]. The idea of course being that changing the prior policy to the posterior policy requires effort and resources, and the more one wants to shape the prior to the posterior, the more resources one needs to consume. One way to capture a resource-constrained agent is therefore to penalize the acceptable difference between the prior and posterior distributions.

The difference between the prior and posterior distributions is measured using the Kullback-Leibler (KL) divergence [17]

$$D_{\text{KL}}(\pi(u|x) \parallel \rho(u|x)) \triangleq \sum_{u \in \mathcal{U}} \pi(u|x) \log \frac{\pi(u|x)}{\rho(u|x)}. \quad (8)$$

The total discounted information cost for a decision-maker starting from any initial state x and following the policy $\pi(u|x)$ is then defined as [56]

$$D^\pi(x) \triangleq \lim_{T \rightarrow \infty} \mathbb{E}_\tau \left[\sum_{t=0}^{T-1} \gamma^t \log \frac{\pi(u_t|x_t)}{\rho(u_t|x_t)} \right], \quad (9)$$

where the expectation in equation (9) is with respect to the resulting trajectory, given by equation (2). The goal is now to not only maximize the value alone as in (4), but rather maximize the trade-off between the value and the information cost in (9) [56, 67, 15]. This is a constrained optimization problem, which can be solved using Lagrange multipliers. The corresponding cost is the free energy for a given policy $\pi(u|x)$, defined as

$$F^\pi(x; \beta) \triangleq V^\pi(x) - \frac{1}{\beta} D^\pi(x), \quad (10)$$

where $\beta > 0$. The objective is now to find the policy $\pi^*(u|x)$ which maximizes equation (10) instead of (4). Substituting known quantities and expanding terms, we obtain the mapping $\mathcal{B} : \mathbb{R}^{|X|} \rightarrow \mathbb{R}^{|X|}$ defined as

$$\begin{aligned} \mathcal{B}[F](x; \beta) \triangleq \max_{\pi(u|x)} \sum_{u \in \mathcal{U}} \pi(u|x) & \left[\sum_{x' \in \mathcal{X}} P(x'|x, u) R(x', x, u) - \frac{1}{\beta} \log \frac{\pi(u|x)}{\rho(u|x)} \right. \\ & \left. + \gamma \sum_{x' \in \mathcal{X}} P(x'|x, u) F(x'; \beta) \right]. \end{aligned} \quad (11)$$

The proof of how this mapping results from equation (10) can be found in [56]. The optimal value of the free energy $F^*(x; \beta)$ is the fixed point of the equation $F^* = \mathcal{B}[F^*]$. Similarly to (6), we define

$$Q_F(x, u; \beta) \triangleq \sum_{x' \in \mathcal{X}} P(x'|x, u) \left[R(x', x, u) + \gamma F(x'; \beta) \right], \quad (12)$$

and $Z(x; \beta) \triangleq \sum_{u \in \mathcal{U}} \rho(u|x) e^{\beta Q_F(x, u; \beta)}$. Equation (11) can be shown to be equivalent to

$$\mathcal{B}[F](x; \beta) = \frac{1}{\beta} \log Z(x; \beta). \quad (13)$$

This is iteratively applied until convergence to obtain the optimal free energy $F^*(x; \beta)$. For a given value of β the optimal policy for the infinite horizon problem is then

$$\pi^*(u|x) = \frac{\rho(u|x) e^{\beta Q_F^*(x, u; \beta)}}{Z^*(x; \beta)}, \quad (14)$$

where $Q_F^*(x, u; \beta)$ and $Z^*(x; \beta)$ are given by

$$\begin{aligned} Q_F^*(x, u; \beta) &= \sum_{x' \in \mathcal{X}} P(x'|x, u) \left[R(x', x, u) + \gamma F^*(x'; \beta) \right], \\ Z^*(x; \beta) &= \sum_{u \in \mathcal{U}} \rho(u|x) e^{\beta Q_F^*(x, u; \beta)}. \end{aligned} \quad (15)$$

Using this framework we are able to model a wide variety of agents with limited resources by varying the resource parameter β . Hence β can be viewed as a parameter which reflects the agent's computational abilities [32]. When $\beta \rightarrow 0$ the information term in equations (10) and (11) becomes dominant and the agent becomes primarily concerned with reducing the information cost $D^\pi(x)$. This leads to $\pi^* = \rho$ and the prior distribution is directly “copied” to the posterior policy, thus representing an agent who is not able to deviate from its initial control policy – indicative of a decision-maker with no computational resources [32]. When, on the other hand, $\beta \rightarrow \infty$ the information cost in equations (10) and (11) is effectively removed. Consequently, the agent focuses on finding a policy that maximizes $V^\pi(x)$. This means that the agent will behave more like its non-information limited counterpart. In this case we therefore expect to recover the non-information limited policy in the limit (i.e., $\pi^* \rightarrow \Gamma^*$), which is indicative of an agent that has unlimited computational resources [32].

In the previous analysis it was assumed that the prior distribution over actions $\rho(u|x)$ was given. Next, we address the question of whether we can find an optimal a priori action distribution $\rho(u|x)$ (or $\rho(u)$) so as to minimize the information cost across all states *on average*. To this end, note that the information-limited MDP does not modify $\rho(u|x)$ when searching for the optimal policy π^* . Instead, it will find the policy that best trades the value and information for a given $\rho(u|x)$. The problem now becomes one of finding both the posterior policy and prior action distribution so as to maximize equation (10) on average across all system states.

Averaging equation (10) over system states, and assuming for simplicity a state-agnostic prior distribution over actions ($\rho(u|x) = \rho(u)$), we obtain

$$\max_{\rho} \sum_{x \in \mathcal{X}} p(x) \left(\max_{\pi} \sum_{u \in \mathcal{U}} \pi(u|x) \left[Q_F(x, u; \beta) - \frac{1}{\beta} \log \frac{\pi(u|x)}{\rho(u)} \right] \right). \quad (16)$$

Optimizing equation (16) with respect to $\rho(u)$ is a convex optimization problem [27, 17], and is numerically solved by recursively iterating between the following relations until convergence [67, 27, 66]

$$\pi^*(u|x) = \frac{\rho^*(u) e^{\beta Q_F^*(x, u; \beta)}}{\sum_u \rho^*(u) e^{\beta Q_F^*(x, u; \beta)}}, \quad (17)$$

$$\rho^*(u) = \sum_x p(x) \pi^*(u|x). \quad (18)$$

Here $p(x)$ is the (fixed) probability distribution over states and π^* is found in the same manner described in the previous section. The proof of this result can be found in [27, 67]. It is worth mentioning that the iteration (17)-(18) is analogous to the Blahut-Arimoto (BA) algorithm in rate-distortion theory where the KL divergence plays the role of the “distortion function” [66, 27, 56, 67].

An agent utilizing the BA algorithm for path planning will find a policy that deviates in various amounts from the provided prior action distribution depending on the value of β .

The optimization problem (16) has an interesting interpretation in terms of the mutual information between states and actions. Recall that the mutual information between two random variables X and U is given by

$$I(X; U) \triangleq \sum_{x, u} p(x, u) \log \frac{p(x, u)}{p(x)p(u)}. \quad (19)$$

Rewrite now (16) as

$$\max_{\rho, \pi} \sum_{x, u} p(x) \pi(u|x) Q_F(x, u; \beta) - \frac{1}{\beta} \sum_x p(x) D_{\text{KL}}(\pi(u|x) \| \rho(u)), \quad (20)$$

or

$$\mathbb{E}_{\pi(u|x)} p(x) Q_F(x, u; \beta) - \frac{1}{\beta} I(X; U), \quad (21)$$

where we have used the fact that

$$\begin{aligned} \sum_x p(x) D_{\text{KL}}(\pi(u|x) \| \rho(u)) &= \sum_x p(x) \sum_u \pi(u|x) \log \frac{\pi(u|x)}{\rho(u)} \\ &= \sum_{x, u} p(x) \pi(u|x) \log \frac{\pi(u|x) p(x)}{\rho(u) p(x)} \\ &= \sum_{x, u} p(x, u) \log \frac{p(x, u)}{\rho(u) p(x)} = I(X, U). \end{aligned} \quad (22)$$

From (21) it can be immediately seen that for low values of β the agent becomes mainly concerned with minimizing the mutual information between states and actions. This results in a policy that is generic with respect to the state, leading to state abstractions as shown in [27]. That is, the agent will focus its efforts to find a single action that yields a high reward, on average, across the state space.

We demonstrate the previous algorithm by applying it to a path-planning problem in the 4×4 grid world shown in Figure 3.

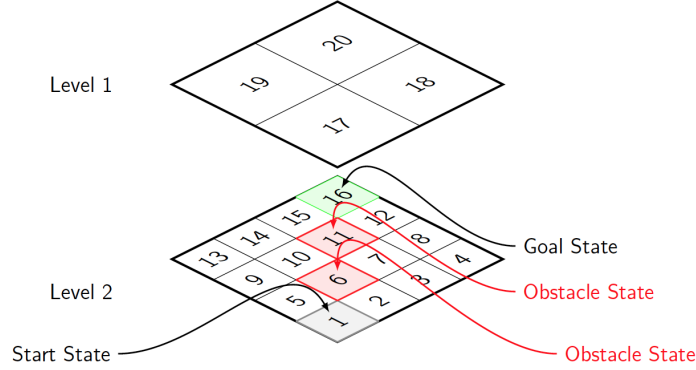


Fig. 3: Path-planning example with two levels of abstraction. Level 2 is the actual (“fine resolution”) level, whereas Level 1 is a coarser resolution level and has lower cardinality, thus planning at this level is much easier. © 2017 IEEE. Reprinted, with permission, from [44].

The agent has at its disposal the action set $\mathcal{U}(x_t) \subseteq \{\text{UP, DN, L, R, A-UP, A-DN}\}$. The first four actions (UP, DN, L, R) correspond to the standard actions “up-down-left-right” in a grid world with 4-cell connectivity. In addition to these actions, however, the agent has at its disposal the actions A-UP and A-DN. These do not correspond to physical movements of the agent, but rather encode changes in the “perceptual representation” of the environment that are motivated by the sensing and/or computational resources of the agent. By executing A-UP the agent reasons in the higher (coarser) abstraction level, while by executing A-DN the agent moves one level down (less coarse, more accurate) abstraction level (see Figure 3). For better accuracy, the agent would prefer, of course, to reason at the bottom (“finer”) resolution level but this may be expensive, and as a result the agent may be forced to reason at a coarse resolution level. These special actions can also be viewed, for example, as the agent’s ability to toggle a sensor suite consisting of two variable resolution sensors on and off. The agent must pay an increased price for navigating the grid world at the finest resolution level, due to the demand to process an increased amount of information. The actions A-UP and A-DN can thus also be viewed as actions that are taken when the agent has limited computational resources, in which case it will favor higher level (coarser) abstractions since these correspond to search spaces of lower cardinality and hence are presumably easier to navigate in.

Figures 4 and 5 display the policies obtained for various values of β for the Blahut-Arimoto information-limited (BA-IL) algorithm. Specifically, the figures show the distribution over actions for a given state along with the value of β and the distribution over states, $p(x)$. In each of these figures, the y-axis corresponds to the action space whereas the x-axis displays the possible system states, with state numbering consistent with Figure 3. Furthermore, the shading of a given state-action pair represents the probability that the action will be selected when the system finds itself in that state. Column n of these plots can therefore be considered a probability distribution over actions for the n^{th} state, where $n \in \{1, 2, \dots, 20\}$. Further details regarding the problem formulation and the reward structure for this paper can be found in [44].

It is important to note that the policy depicted in Figure 4(a) by Γ^* represents a rational agent. That is, the policy Γ^* would be followed by an agent who has sufficient processing capabilities to find the optimal actions and does so at the finest grid level, electing to not abstract to a coarser resolution. As the agent's computational abilities are reduced, it begins to favor planning at the coarser grid level while its resulting policy becomes increasingly suboptimal with respect to Γ^* .

Shown in Figure 4(b) is the prior action distribution, $\rho(u|x)$. The prior distribution, ρ , can be viewed as the policy the agent would follow in case it had no resources allowing it to find another, possibly better, policy. Thus, if the agent has insufficient resources to deliberate (limit as $\beta \rightarrow 0$), ρ becomes the posterior policy π , as seen when comparing Figures 4(b) and 5(b). This is also observed in Figure 5 as the

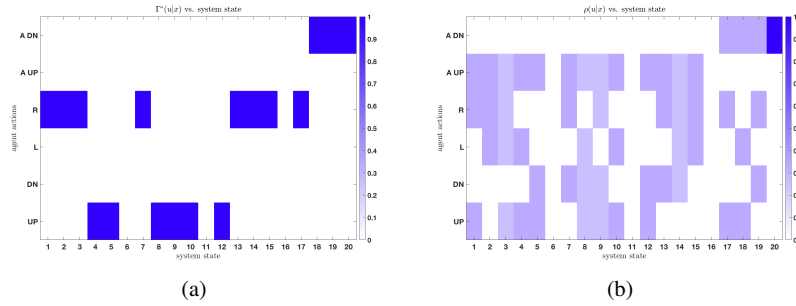


Fig. 4: Deterministic policy, $\Gamma^*(u|x)$ (left) and a priori action distribution, $\rho(u|x)$ (right). © 2017 IEEE. Reprinted, with permission, from [44].

resulting policies (π^*) approach Γ^* as β is increased. We also note that the policy becomes less deterministic as $\beta \rightarrow 0$. Furthermore, as β is reduced, abstractions emerge from the solution of the BA-IL algorithm. This can be seen in Figure 5(b) where we see that the agent elects to select the same action for many states. That is, the resulting optimal policy π^* from the BA-IL algorithm is relatively *generic* with respect to the state and thus the state and action spaces have low mutual information. Finally, it should be noted from Figure 5 that, as β is reduced, the agent prefers to

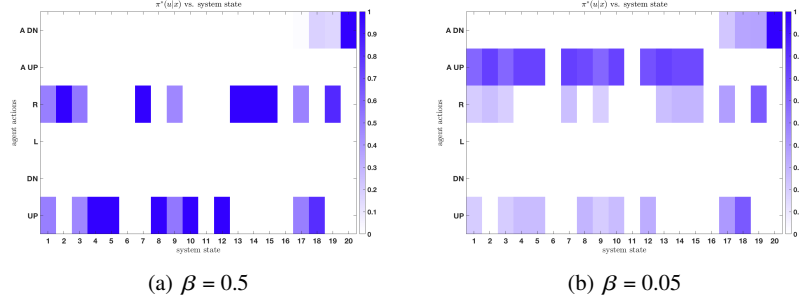


Fig. 5: $\pi^*(u|x)$ from the BA-IL algorithm for $\beta = 0.5$ and $\beta = 0.05$. © 2017 IEEE. Reprinted, with permission, from [44].

plan paths at the coarser grid level, indicative of the fact that less computationally complex solutions are sought at the cost of perfect optimality.

4 An Information-Theoretic Approach for Hierarchical Decision-Making

In the previous section we developed an information-theoretic approach to account for the computational resources in sequential decision-making problems modeled by MDPs using a hierarchy of abstractions of the state space. However, it was assumed that the hierarchy of abstractions was given a priori. In this section we develop a framework to obtain abstractions for decision-making where the abstractions emerge as a function of the agent's limited computational resources.

We consider the emergence of graph abstractions in the form of multi-resolution quadtree representations. Quadrees are a common tool utilized in the robotics community to reduce the complexity of environments in order to speed-up path-planning and ease internal storage requirements [35, 36, 8, 22, 50]. To this end, we assume that the environment $\mathcal{W} \subset \mathbb{R}^2$ (generalizable to $\mathbb{R}^d$) is given by a two-dimensional grid world where each grid element is a unit square (hypercube). We assume that there exists an integer $\ell > 0$ such that $\mathcal{W}$ is contained within a square (hypercube) of side length 2^ℓ . A tree representation $\mathcal{T} = (\mathcal{N}, \mathcal{E})$ of $\mathcal{W}$ consists of a set of nodes $\mathcal{N} = \mathcal{N}(\mathcal{T})$ and edges $\mathcal{E} = \mathcal{E}(\mathcal{T})$ describing the interconnections between the nodes in the tree [36]. We denote the set of all possible quadtree representations of maximum depth ℓ of $\mathcal{W}$ by $\mathcal{T}^Q$ and we let $\mathcal{T}_{\mathcal{W}} \in \mathcal{T}^Q$ denote the finest quadtree representation of $\mathcal{W}$; an example is shown in Figure 6.

Varying the abstraction granularity of $\mathcal{W}$ can be equivalently viewed as selecting various trees $\mathcal{T}_q$ in the space $\mathcal{T}^Q$. Our problem is then one of selecting a tree $\mathcal{T}_q \in \mathcal{T}^Q$ as a function of the agent's computational capabilities.

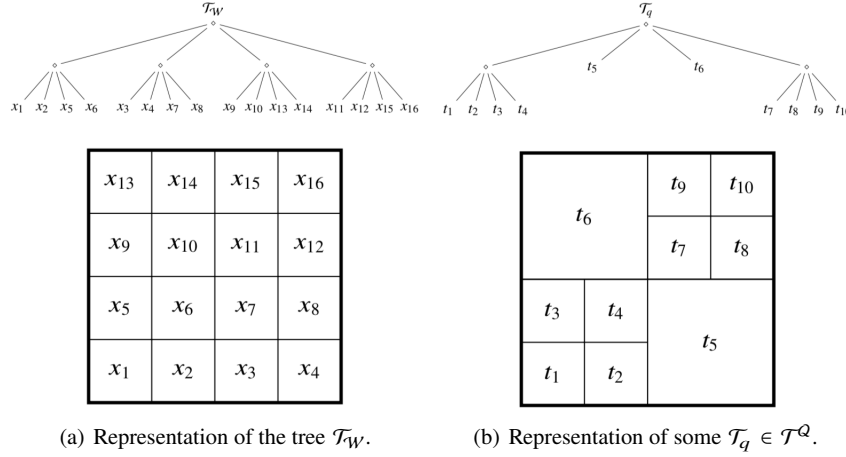


Fig. 6: Two different quadtree representations and the corresponding grid for a 4×4 environment.

Agglomerative Information Bottleneck for Quadtree Compression

Our objective is to find compressed representations in a given hierarchy of quadrees such that the compressed quadtree maintains as much as possible of the relevant information from the original quadtree. The task of obtaining optimal compressed representations of signals is addressed within the realm of information theory [66, 61, 60, 62, 17, 34]. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space with finite sample space Ω , σ -algebra $\mathcal{F}$, and probability measure $\mathbb{P} : \mathcal{F} \rightarrow [0, 1]$. Denote the set of real and positive real numbers as $\mathbb{R}$ and $\mathbb{R}_+$, respectively. Let $X : \Omega \rightarrow \mathbb{R}$ denote the random variable corresponding to the original, uncompressed, signal, where X takes values in the set $\Omega_X = \{x \in \mathbb{R} : X(\omega) = x, \omega \in \Omega\}$ and, for any $x \in \mathbb{R}$, let $p(x) = \mathbb{P}(\{\omega \in \Omega : X(\omega) = x\})$. Furthermore, let the random variable $T : \Omega \rightarrow \mathbb{R}$ denote the compressed representation of X , where T takes values in the set $\Omega_T = \{t \in \mathbb{R} : T(\omega) = t, \omega \in \Omega\}$. The level of compression between the random variables X and T is measured by the mutual information [66, 17]

$$I(T; X) \triangleq \sum_{t, x} p(t, x) \log \frac{p(t, x)}{p(t)p(x)}. \quad (23)$$

The goal is then to find a stochastic mapping (encoder), denoted by $p(t|x)$, which maps outcomes in the uncompressed space $x \in \Omega_X$, to outcomes in the compressed representation $t \in \Omega_T$ so as to minimize $I(T; X)$, while, at the same time, keep as much information encoded in another random variable Y . The variable Y represents information we are interested in preserving when forming the compressed representation T [66, 61].

In this work we will use the agglomerative information bottleneck (AIB) method [61] to form compressed representations of X , which is useful when deterministic clusters that retain predictive information regarding the relevant variable Y are desired. The AIB method is based on the information bottleneck (IB) approach [66] to solve for deterministic, or hard, encoders (i.e., $p(t|x) \in \{0, 1\}$ for all t, x). Specifically, we wish to use the AIB idea to find compressed versions of $\mathcal{T}_{\mathcal{W}}$. Concepts from AIB will prove useful in our formulation, since each tree $\mathcal{T}_q \in \mathcal{T}^Q$ encodes a hard (deterministic) abstraction of $\mathcal{W}$, where each leaf node of $\mathcal{T}_{\mathcal{W}}$ is aggregated to a specific leaf node of $\mathcal{T}_q$. That is, by viewing the uncompressed space (Ω_X) as the leaf nodes of $\mathcal{N}(\mathcal{T}_{\mathcal{W}})$ and the abstracted (compressed) space (Ω_T) as the leaf nodes of $\mathcal{N}(\mathcal{T}_q)$, the abstraction operation can be specified in terms of an encoder $p(t|x)$ where $p(t|x) \in \{0, 1\}$ for all t and x , where $p(t|x) = 1$ if a leaf node x of $\mathcal{N}(\mathcal{T}_{\mathcal{W}})$ is aggregated to a leaf node t of $\mathcal{N}(\mathcal{T}_q)$, and zero otherwise (see Figure 6). To better understand these connections, below we briefly review the AIB before presenting how it is applied to our problem.

The AIB considers the optimization problem

$$\max_{p(t|x)} \mathcal{L}_Y(p(t|x); \beta), \quad (24)$$

where the Lagrangian is defined as

$$\mathcal{L}_Y(p(t|x); \beta) \triangleq I(T; Y) - \frac{1}{\beta} I(T; X), \quad (25)$$

and the maximization is performed over deterministic distributions $p(t|x)$ for given $\beta > 0$ and $p(x, y)$ [61, 60]. The AIB works from bottom-up, starting with $T = X$ and with each consecutive iteration reduces the cardinality of T until $|\Omega_T| = 1$. Specifically, let T_m represent the abstracted space with m elements ($|\Omega_{T_m}| = m$) and let T_i represent the compressed space with $|\Omega_{T_i}| = i < m$ elements, where $i = m - 1$ and the number of merged elements is $n = 2$. We then merge elements $\{t'_1, \dots, t'_n\} \subseteq \Omega_{T_m}$ to a single element $t \in \Omega_{T_i}$ to obtain T_i . The merger cost is given by $\Delta \mathcal{L}_Y : 2^{\Omega_{T_m}} \times \mathbb{R}_+ \rightarrow \mathbb{R}$, which can be written as [60]

$$\Delta \mathcal{L}_Y(\{t'_1, \dots, t'_n\}; \beta) = p(t) \left[\text{JS}_{\Pi}(p(y|t'_1), \dots, p(y|t'_n)) - \frac{1}{\beta} H(\Pi) \right], \quad (26)$$

where $H(\Pi)$ is the Shannon entropy, where Π is given by

$$\Pi = [\Pi_1, \dots, \Pi_n]^T \triangleq \left[\frac{p(t'_1)}{p(t)}, \dots, \frac{p(t'_n)}{p(t)} \right]^T, \quad (27)$$

and $\text{JS}_{\Pi}(p_1, \dots, p_n)$ is the Jensen-Shannon (JS) divergence between the distributions $p_1, \dots, p_n$, with weights Π defined as [47]

$$\text{JS}_{\Pi}(p_1, \dots, p_n) \triangleq \sum_{s=1}^n \Pi_s \text{D}_{\text{KL}}(p_s, \bar{p}), \quad (28)$$

where, for each $y \in \Omega_Y$, $\bar{p}(y) = \sum_{s=1}^n \Pi_s p_s(y)$. Furthermore, we have that

$$p(t) = \sum_{s=1}^n p(t'_s), \quad p(y|t) = \sum_{s=1}^n \Pi_s p(y|t'_s), \quad (29)$$

which can be found by realizing that $p(t|x) \in \{0, 1\}$ for all $x \in \Omega_X$ and $t \in \Omega_T$ along with the Markov chain condition $T \leftrightarrow X \leftrightarrow Y$ [61, 60].

Optimal Compression of Quadtrees

In this section we show how to use the AIB ideas to provide a hierarchy of compressed representations of $\mathcal{T}_{\mathcal{W}}$ that depends on the parameter β . Similarly to (25), we can define the Lagrangian in the space of quadtrees as the mapping $L_Y : \mathcal{T}^Q \times \mathbb{R}_+ \rightarrow \mathbb{R}$, given by

$$L_Y(\mathcal{T}_q; \beta) \triangleq \mathcal{L}_Y(p^q(t|x); \beta), \quad (30)$$

where $\mathcal{L}_Y(p(t|x); \beta)$ is defined in (25). Then, for a given $\beta > 0$, we can search the space of quadtrees for the one that maximizes (30). This optimization problem is formally given by

$$\mathcal{T}_{q^*} = \operatorname{argmax}_{\mathcal{T}_q \in \mathcal{T}^Q} L_Y(\mathcal{T}_q; \beta). \quad (31)$$

By posing the optimization problem as in (31), we have implicitly incorporated the constraints on the mapping $p(t|x)$ in order for the resulting representation to be a quadtree depiction of the world $\mathcal{W}$. The resulting world representation is encoded by the mapping $p^{q^*}(t|x)$. That is, the leafs of $\mathcal{T}_{q^*}$ determine the optimal multi-resolution representation of $\mathcal{W}$ for the given β .

While the optimization problem given by (31) allows one to form an analogous problem to that in (24) over the space of trees, the drawback of this method is the need to exhaustively enumerate all feasible quadtrees which can represent the world space. In other words, (31) requires that $p^q(t|x)$ be provided for each $\mathcal{T}_q \in \mathcal{T}^Q$. Because of this, the problem becomes intractable for large grid sizes and thus requires reformulation to handle larger world maps. We propose a sequential characterization of (30), and we formulate an optimization problem requiring the generation of candidate solutions, which are all obtained by successive expansions starting from the root tree $\mathcal{T}_{q^0}$ to the current tree $\mathcal{T}_{q^m}$. The optimization problem that maximizes retained information while compressing $\mathcal{T}_{q^m}$ can then be reformulated as [43]

$$\max_m \max_{\mathcal{T}_{q^1}, \dots, \mathcal{T}_{q^m}} L_Y(\mathcal{T}_{q^0}; \beta) + \sum_{i=0}^{m-1} \Delta L_Y(\mathcal{T}_{q^i}, \mathcal{T}_{q^{i+1}}; \beta), \quad (32)$$

where

$$\Delta L_Y(\mathcal{T}_{q^i}, \mathcal{T}_{q^{i+1}}; \beta) \triangleq L_Y(\mathcal{T}_{q^{i+1}}; \beta) - L_Y(\mathcal{T}_{q^i}; \beta). \quad (33)$$

In this formulation, the constraint encoding that the resulting representation is a quadtree is handled implicitly by $\mathcal{T}_{q^i} \in \mathcal{T}^Q$. The additional maximization over m in (32) appears since the horizon of the problem is not known a-priori and is, instead, a free parameter in the optimization problem.

Next, we propose an efficient algorithm to solve the optimization problem in (32).

The Q-tree Search Algorithm

One way to solve (32) is to maximize $\Delta L_Y(\mathcal{T}_{q^i}, \mathcal{T}_{q^{i+1}}; \beta)$ myopically at each step. That is, provided that $\mathcal{T}_{q^{i+1}} \in \mathcal{T}^Q$ is a neighbor¹ of $\mathcal{T}_{q^i} \in \mathcal{T}^Q$, we consider the next tree $\mathcal{T}_{q^{i+1}}$ that maximizes the value of $\Delta L_Y(\mathcal{T}_{q^i}, \mathcal{T}_{q^{i+1}}; \beta)$, and we sequentially keep selecting trees ($\mathcal{T}_{q^{i+1}} \rightarrow \mathcal{T}_{q^{i+2}} \rightarrow \dots$) until no further improvement is possible. This greedy algorithm is simple to implement and requires little pre-processing but it does not, in general, find globally optimal solutions. This is owing to the short-sightedness of the algorithm and its inability to realize that poor expansions at the current step may lead to much higher-valued options in the future.

In order to improve performance we introduce the Q-tree search algorithm that takes into account future tree expansions. This is analogous to problems in reinforcement learning and dynamic programming, where an action-value function (Q -function) is introduced to incorporate the notion of cost-to-go for selecting among feasible actions in a given state [11, 63]. The idea behind introducing such a function is to incorporate future costs, thus allowing agents to take actions that are not the most optimal with respect to the current one-step cost, but have lower total cost due to expansions that are possible in the future.

To this end, we define the function

$$Q_Y(\mathcal{T}_{q^i}, \mathcal{T}_{q^{i+1}}; \beta) \triangleq \max \{ \Delta L_Y(\mathcal{T}_{q^i}, \mathcal{T}_{q^{i+1}}; \beta) + \sum_{\tau=1}^n Q_Y(\mathcal{T}_{q^{i+1}}, \mathcal{T}_{q_{\tau}^{i+2}}; \beta), 0 \}, \quad (34)$$

where $\mathcal{T}_{q^{i+1}}$ is a neighbor of $\mathcal{T}_{q^i}$ with higher leaf node cardinality and

$$Q_Y(\mathcal{T}_{q'}, \mathcal{T}_W; \beta) \triangleq \max \{ \Delta L_Y(\mathcal{T}_{q'}, \mathcal{T}_W; \beta), 0 \}, \quad (35)$$

for all $\mathcal{T}_{q'} \in \mathcal{T}^Q$ for which $\mathcal{T}_W \in \mathcal{T}^Q$ is a neighbor. The quadtrees $\mathcal{T}_{q_{\tau}^{i+2}}$, $\tau \in \{1, \dots, n\}$ are neighbors of $\mathcal{T}_{q^{i+1}}$ which are obtained by expanding each of the children of one of the leaf of $\mathcal{T}_{q^i}$. The next theorem formally establishes the optimality of solutions found by the Q-tree search algorithm.

¹ A quadtree $\mathcal{T}_{q'}$ is a neighbor of $\mathcal{T}_q$ if $\mathcal{T}_{q'}$ results from a $\mathcal{T}_q$ by a single node expansion of a leaf node of $\mathcal{T}_q$, where by “expansion” of a node we mean adding to the tree all its children. For instance, the two quadtrees in Figure 6 are not neighbors according to our definition, as more than one node expansion is needed to generate the quadtree shown on the right from the quadtree shown on the left in that figure.

Theorem 1 *Let $\mathcal{T}_{\bar{q}} \in \mathcal{T}^Q$ to be an optimal with respect to the cost $L_Y(\cdot; \beta)$. Assume, without loss of generality, that the Q-tree search algorithm is initialized at the tree $\mathcal{T}_{q^0} \in \mathcal{T}^Q$, where $\mathcal{T}_{q^0} \subseteq \mathcal{T}_{\bar{q}}$ and let $\mathcal{T}_{q^*} \in \mathcal{T}^Q$ be the solution returned by the Q-tree search algorithm. Then $\mathcal{T}_{q^*} = \mathcal{T}_{\bar{q}}$.*

Below we present a numerical example to demonstrate the emergence of abstractions in a grid-world setting using the Q-tree search algorithm with different values of β . To this end, consider the environment shown in Figure 7 having dimension 128×128 . We view this map as representing an environment where the intensity of the color indicates the probability that a given cell is occupied. In this view, the map in Figure 7 can be thought of as an occupancy grid where the original space, X , is considered to be the elementary cells shown in the figure. We wish to compress X

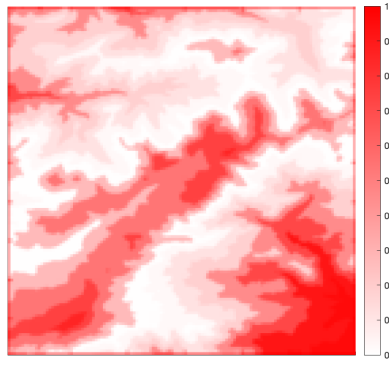


Fig. 7: 128×128 original map of environment. Shading of red indicates the probability that a cell is occupied.

to an abstract representation T (a quadtree), while preserving as much information regarding cell occupancy as possible. Thus, the relevant random variable Y is the probability of occupancy. Consequently, $\Omega_Y = \{0, 1\}$ where $y = 0$ corresponds to free space and $y = 1$ to occupied space. It is assumed that $p(x)$ is provided and $p(y|x)$ is given by the occupancy grid, where $p(x, y) = p(y|x)p(x)$. For simplicity, we assume that $p(x)$ is uniform, but the results can be easily extended to the case when $p(x)$ is not uniform.

Several environment depictions for various values of β obtained from the Q-tree search algorithm are shown in Figure 8. As seen in these figures, the solution returned by the Q-tree search algorithm approaches that of the original space as $\beta \rightarrow \infty$, with a spectrum of solutions obtained as β is varied. These figures also show that areas containing high information content, as specified by Y , are refined first while leaving the regions with less information content to be refined at a higher values of β .

We see that β resembles a sort of a “rationality parameter,” analogous to [44, 67, 27], where agents with low β are considered to be more resource limited, thus utilizing simpler, lower cardinality representations of the environment. Thus, once

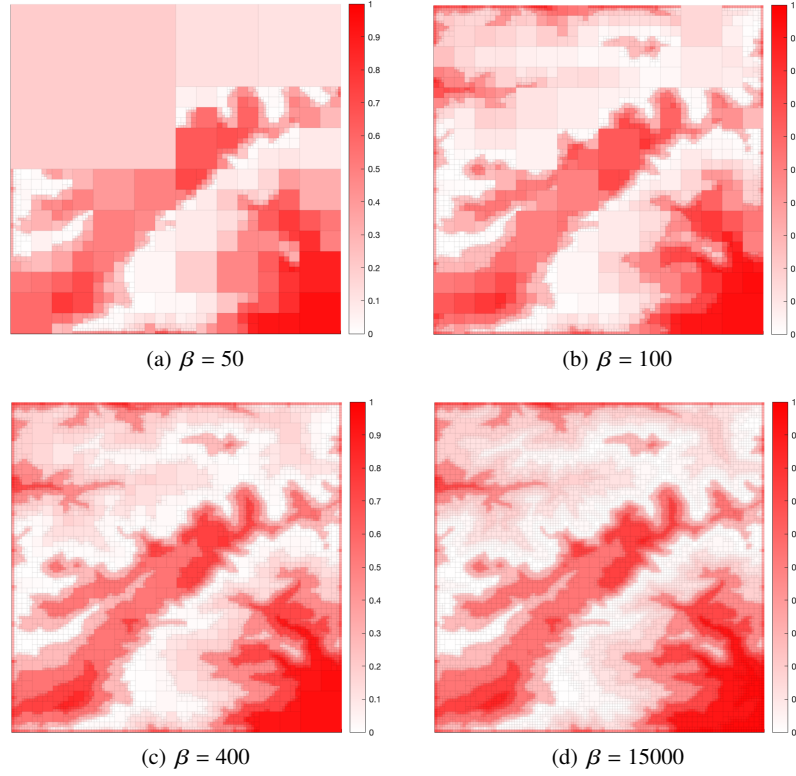


Fig. 8: Representations of the environment for different values of β [43].

the map is given, changing *only* the value of β gives rise to a variety of world representations of varying resolution.

5 Stochastic Games and Bounded Rationality

Decision-making is often a process that involves multiple entities, whose actions are beyond the subjective control of the decision-maker. Either the environment or other agents affect the outcome of decision-making and hence the agent's deliberation process needs to account for those unpredictable factors as well. A natural way to capture these externalities that are beyond the decision-maker's own control is via game theory. Stochastic games, in particular, is the standard framework to model situations involving multiple agents [25, 51]. However, the solution of stochastic games is often considered to be an intractable undertaking mainly because of the “rationality” and “common knowledge” assumptions imposed on all agents (also referred to

as “players” in the context of games). The manifestation of this axiomatic imposition of rationality is the celebrated Nash equilibrium (NE)². The Nash equilibrium involves a solution concept such that each player computes the best response strategy to the other players’ best response strategies, so that no player has the incentive to unilaterally deviate from its own strategy.

Given the strategies of its opponents, each player can compute the value function (and hence the best response) by solving a single-sided MDP using, say, value iteration (see (5)). However, in games the value function of each agent (or the value of the game for the zero-sum case) is coupled to the strategies of the other players. In order to compute the NE each player then needs to know the behavior (i.e., strategies) of the other player(s) and hence, implicitly, their value function. Solving for perfectly rational Nash equilibria thus leads to an infinite regress, as one player computes its own value function by inferring the value function of the other player, who, in turn, computes its own value function by inferring the first player’s value function, which requires inferring what the other player infers about the value function of the first player etc, at infinitum. This infinite regress is what makes the computation of Nash equilibria difficult in practice.

Although the Nash equilibrium definition seems natural, it has received a lot of criticism in practice since it is not at all clear as to why and under what conditions the players in an N-person game may be expected to play such an equilibrium [33]. Nash equilibria make sense only if one assumes that, for some reason, each player knows the strategies of the other players. But this assumption appears to be rather restrictive for many practical situations, especially if the players are not omnipotent (i.e., not perfectly rational). In addition, the calculation an NE (not surprisingly perhaps in the light of the previous discussion) has been shown to be NP-hard [20].

There are two aspects of the problem that challenge this pessimistic point of view to solving stochastic games. The first is the realization that, in practice, equilibria that imply some form of interdependence between agents are not only plausible but also more likely [2]. The basic tenet of simultaneous action by all players without any collaboration whatsoever, deeply engrained in the Nash equilibrium concept, is too stringent and may be relaxed, leading to more efficient computations [3, 28]. The second aspect (which is emphasized here) is a departure from imposing perfect rationality among all agents and, instead, take into consideration the different sophistication levels of the agents, their perceived knowledge of the environment and of the other agents, or their limited computational resources. Indeed, the assumption of perfect rationality is considered to be too strong and perhaps unrealistic for many applications [59, 29].

One way to resolve this dilemma is by adopting a bounded rationality framework. Several manifestations of bounded rationality exist in the literature, but here we adopt the one that makes use of the model of *cognitive hierarchy* (CH) [16] and *level-k thinking* [38], motivated by the so-called “theory of mind” [72, 26]. Under this notion, the agent no longer seeks the (Nash) equilibrium, but instead seeks a best response to some (believed or inferred) strategy of its opponent(s) according

² Since different agents may have different objectives, the concept of optimality is not suitable for games. Instead, all agents seek to find an *equilibrium*.

to an assumed level- k responding structure, thus restricting each player's depth of recursive inference. This imposes a level of sophistication for each agent, with a totally rational agent employing infinite regress. Less sophisticated (i.e., bounded rational) players use regress up to a finite depth. The level of regress of each player induces a "type" for each player (its level of sophistication) [72].

In order to demonstrate the main ideas behind bounded rational stochastic games, in this section we apply the theory to a pursuit-evasion game in a stochastic environment where both players are assumed to be bounded rational. The generalization to general non-zero sum multi-player games is conceptually straightforward, but the formulas can quickly become somewhat unwieldy.

Stochastic Pursuit-Evasion

In this section, we consider a two-player pursuit-evasion game (PEG) in a stochastic environment to demonstrate the theory. Pursuit-evasion games are a special class of dynamic games [41, 71, 39, 7], which capture a wide range of diverse behaviors amongst adversarial agents. An extensive amount of literature exists on the topic. Some notable examples include [5, 6, 39, 37, 7, 13, 58, 48, 69]. A *stochastic* pursuit-evasion game is a special case of a two-player ($N = 2$) zero-sum stochastic dynamic game for which the running reward is zero ($R(x, x', u) = 0$) and the objective is to maximize/minimize a *terminal cost/reward* $g(x_T)$ (typically the miss-distance) at some future time T where $x_T \in \mathcal{X}_a$ with $\mathcal{X}_a \subset \mathcal{X} \triangleq \mathcal{X}_1 \times \mathcal{X}_2$ the set of the absorbing states, that enforce game termination. Here $\mathcal{X}_i$ is the state space of agent $i \in \{1, 2\}$. The time T when the game ends is not known a priori, so the PEG is an infinite horizon game with no discount ($\gamma = 1$) and absorbing states.

In a PEG the maximizer player is called the Evader and the minimizer player is called the Pursuer. Let the policy (or strategy) of player $i = 1$ (the Evader) be denoted by π^1 and let the set of the Evader strategies be Π^1 . Similarly, let the policy (or strategy) of player $i = 2$ (the Pursuer) be denoted by π^2 and let the set of the Pursuers strategies be Π^2 . The expected reward-to-go function for agent i under the joint policy $\pi \triangleq (\pi^1, \pi^2) \in \Pi \triangleq \Pi^1 \times \Pi^2$ and initial state x is

$$V_\pi^i(x) \triangleq V_{\pi^1, \pi^2}^i(x) = \mathbb{E}_{x, \pi} [g^i(x(T))], \quad i = 1, 2, \quad (36)$$

where the terminal cost for a two-player, zero-sum game satisfies $g^1 = -g^2 = g$. Note that T depends, in general, on the joint strategy π . Both agents try to maximize their own expected reward functions by choosing their respective policies π^i . Specifically, for a given initial state x , the optimization problem can be written as

$$\max_{\pi^1 \in \Pi^1} V_{\pi^1, \pi^2}^1(x), \quad \max_{\pi^2 \in \Pi^2} V_{\pi^1, \pi^2}^2(x). \quad (37)$$

Since the game is zero-sum, we have that $V_{\pi^1, \pi^2}^1(x) = -V_{\pi^1, \pi^2}^2(x) = V_{\pi^1, \pi^2}(x)$ for all joint policies $(\pi^1, \pi^2) \in \Pi$ and all initial states $x \in \mathcal{X}$. As a result, the previous

optimization problem reduces to

$$\max_{\pi^1 \in \Pi^1} \min_{\pi^2 \in \Pi^2} V_{\pi^1, \pi^2}(x) = \min_{\pi^2 \in \Pi^2} \max_{\pi^1 \in \Pi^1} V_{\pi^1, \pi^2}(x). \quad (38)$$

Given the policy of the minimizing player π^2 , let us consider the MDP for the maximizer player. To calculate the value function at a state $x \in \mathcal{X}$, we solve the following Bellman equation:

$$V_{\pi^2}(x) \triangleq \max_{\pi^1 \in \Pi^1} \mathbb{E}_P[V_{\pi^1, \pi^2}^1(x')|x, \pi^2] = \max_{\pi^1 \in \Pi^1} \sum_{x' \in \mathcal{X}} P(x'|x, \pi^1(x), \pi^2(x)) V_{\pi^2}(x'). \quad (39)$$

where the subscript in V_{π^2} denotes the dependence on the opponent's strategy π^2 (compare with notation in (3)). For the minimizer player, the Bellman equation is similar but with an min operator instead of a max operator, that is,

$$V_{\pi^1}(x) \triangleq \max_{\pi^2 \in \Pi^2} \mathbb{E}_P[-V_{\pi^1, \pi^2}^1(x')|x, \pi^1] = \min_{\pi^2 \in \Pi^2} \sum_{x' \in \mathcal{X}} P(x'|x, \pi^1(x), \pi^2(x)) V_{\pi^1}(x'). \quad (40)$$

Let us assume that (39) can be solved given the opponent's policy, and similarly for (40). The result is the unique value function $V_{\pi^{1,*}, \pi^{2,*}}$ along with the optimal (Nash) policies $\pi^{1,*}$ and $\pi^{2,*}$.

Level-k Thinking

In the level- k framework, we construct iterative decision rules for the agents. The iterative process begins with “level-0” agents, who have no prior assumptions about their opponents or the structure of the game, and merely act randomly; i.e., they implement a random policy where each action is taken equally likely. The “level- k ” agents presume that their opponent(s) are of “level- $(k-1)$ ”, and they respond accordingly based on this assumption. The notion of rationality is captured through the parameter k ; the higher the value of k , the higher the level of rationality. Consequently, in order to capture the notion of bounded rationality [38, 46], each agent has a maximum level $k_{\max}^i$ beyond which it cannot operate (different agents may have different maximum levels). Under this framework, the computational complexity of finding the optimal policy for an agent is significantly reduced [46], since the game can be cast as a single-sided MDP problem [38] for each agent, which can be solved efficiently [25].

Let π^i denote the strategy of player $i \in \mathcal{I} = \{1, 2, \dots, N\}$, and let Π^i be the set of all such strategies. Let also $\pi^{-i} = (\pi^1, \pi^2, \dots, \pi^{i-1}, \pi^{i+1}, \dots, \pi^N)$ denote the collection of all the strategies of the opponents of player i . In what follows, the superscript $k \in \{0, 1, 2, \dots\}$ within a parenthesis will denote the level of rationality of the agent. The premise of level- k thinking is that, if $\pi^{-i, (k)}$ is a level- k strategy for the opponent(s) of agent i , the agent will choose its best response to $\pi^{-i, (k)}$, which is a level- $(k+1)$ strategy computed, for every state x , by

$$\pi^{i,(k+1)}(x) \in \operatorname{argmax}_{\pi^{i,(k+1)} \in \Pi^i} \sum_{x' \in \mathcal{X}} P(x'|x, \pi^{i,(k+1)}(x), \pi^{-i,(k)}(x)) V^{i,(k+1)}(x'), \quad (41)$$

which is precisely the best response of agent i to its opponent's strategy $\pi^{-i,(k)}$ (and hence it depends on $\pi^{-i,(k)}$), and where for notational simplicity we have set $V^{i,(k+1)} = V_{\pi^{-i,(k)}}^i$. We write, accordingly,

$$\pi^{i,(k+1)} \in \operatorname{BestResponse}(\pi^{-i,(k)}), \quad (42)$$

to represent the strategy of agent i obtained from (41). Note that the best response to an opponent's level- k strategy $\pi^{-i,(k)}$ is by definition the level- $(k+1)$ strategy $\pi^{i,(k+1)}$, which is not necessarily the NE strategy $\pi^{i,*}$. Recall that the set of strategies $(\pi^{1,*}, \pi^{2,*}, \dots, \pi^{N,*})$ is a Nash equilibrium if the following conditions hold

$$\pi^{i,*} \in \operatorname{BestResponse}(\pi^{-i,*}), \quad i \in \mathcal{I}, \quad (43)$$

with the corresponding value functions $V^{i,*}$ obtained from

$$\begin{aligned} V_{m+1}^{i,*}(x) &= \max_{\pi^i \in \Pi^i} \sum_{x' \in \mathcal{X}} P(x'|x, \pi^i(x), \pi^{-i,*}(x)) V_m^{i,*}(x'), \quad x \in \mathcal{X}, \\ V_m^{i,*}(x) &= g(x), \quad x \in \mathcal{X}_a, \end{aligned} \quad (44)$$

for $m = 1, 2, \dots$. If all players are rational, by induction of (42) they will all let $k \rightarrow \infty$ to ensure that $V^{i,(k)} \approx V^{i,(k+1)} = V^{i,(\infty)}$, since they all know that they should be using a strategy that is one level higher than the strategy of their opponents, and they also all know that if their opponents, being rational, will do the same. If computational constraints are present, which do not allow the agents to perform this induction till convergence to $V^{i,(\infty)}$, they will have to terminate the process at some finite order, say $k_{\max}$. As a result, in practice an agent with limited computational resources (e.g., a human³) can continue this process only up to a certain level $k_{\max}$. This process of building strategies level after level as in (41) therefore continues until the prescribed maximum rationality level $k_{\max}^i \geq 1$ of agent i is achieved.

In general, the level- k policy for agent i is calculated via (compare with (39) and (40))

$$V^{i,(k^i)}(x) = \max_{\pi^{i,(k^i)} \in \Pi^i} \sum_{x' \in \mathcal{X}} P(x'|x, \pi^{i,(k^i)}(x), \pi^{-i,(k^i-1)}(x)) V^{i,(k^i)}(x'), \quad (45)$$

$$\pi^{i,(k^i)}(x) \in \operatorname{argmax}_{\pi^{i,(k^i)} \in \Pi^i} \sum_{x' \in \mathcal{X}} P(x'|x, \pi^{i,(k^i)}(x), \pi^{-i,(k^i-1)}(x)) V^{i,(k^i)}(x'), \quad (46)$$

where $k^i \in \{1, 2, \dots, k_{\max}^i\}$ is the rationality level of agent i . Notice that, in order to calculate the strategy $\pi^{i,(k^i)}$, for $k^i \in \{1, \dots, k_{\max}^i\}$, agent i needs to calculate not only its own strategy $\pi^{i,(k^i)}$ but also its opponents' strategies $\pi^{-i,(k^i-1)}$, for all $k^i \in \{1, \dots, k_{\max}^i-1\}$.

³ Behavioral studies in economics seem to indicate that humans apply k -level reasoning only up to levels 3-4 at a maximum [1].

In general, there is no guarantee that the previous iterative level- k construction will converge without extra assumptions. For the case when $\mathcal{I} = \{1, 2\}$, however, we have the following result.

Lemma 1 *If $\pi^{i,(k+2)} = \pi^{i,(k)}$ for agent i at some level k , then the two agents reach a Nash equilibrium by applying the strategy pair $(\pi^{i,(k)}, \pi^{-i,(k+1)})$.*

Proof By the construction of level- k strategies in (41), we know that

$$\begin{aligned}\pi^{i,(k+2)} &\in \text{BestResponse}(\pi^{-i,(k+1)}), \\ \pi^{-i,(k+1)} &\in \text{BestResponse}(\pi^{i,(k)}).\end{aligned}$$

Since $\pi^{i,(k+2)} = \pi^{i,(k)}$, we have the following fixed point property:

$$\begin{aligned}\pi^{i,(k)} &\in \text{BestResponse}(\pi^{-i,(k+1)}), \\ \pi^{-i,(k+1)} &\in \text{BestResponse}(\pi^{i,(k)}).\end{aligned}$$

The strategy pair $(\pi^{i,(k)}, \pi^{-i,(k+1)})$ then corresponds to a Nash equilibrium, by definition. $\square$

Corollary 1 *Assume no constraint of the maximum level is imposed. The level- k iterative process terminates at some level $\bar{k}$ if the condition $\pi^{i,(\bar{k}+2)} = \pi^{i,(\bar{k})}$ is satisfied for some $i \in \{1, 2\}$.*

A Pursuit-Evasion Game in a Stochastic Environment

In order to apply the theory, we assume a PEG between two bounded-rational players (e.g., UAVs) in a stochastic environment (e.g., wind field). The game is taking place in a discrete world of size 18×18 , shown in Figure 9. The objective of the Evader is to reach one of the two evade states, while avoiding crashing into the obstacles and avoiding capture by the Pursuer. The Pursuer's objective is to capture the Evader before it reaches one of the evade states, while also not crashing into the obstacles. If the Pursuer and the Evader occupy the same cell capture ensues, and the Evader loses the game. As a result, the Evader wins the game if it reaches one of the two evade states alone (not co-occupied with the Pursuer). The terminal reward is assigned as follows

$$g(x) = \begin{cases} +1, & \text{if successful capture or if Evader crashes,} \\ -1, & \text{if successful evasion or if Pursuer crashes,} \\ 0, & \text{if both players crash.} \end{cases} \quad (47)$$

for all $x \in \mathcal{X}_a$.

The motion of both players is affected by a stochastic wind. The mean velocity of the wind is generated randomly, while the covariance is set to $\sigma_w = 0.4$ with no spatial correlation. Both agents have the same speed (assumed unity) and the same

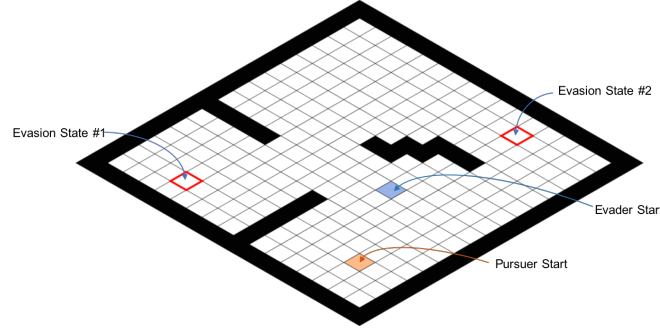


Fig. 9: Grid world for PEG.

action set, namely, $\mathcal{U}^1 = \mathcal{U}^2 = \{UP, DN, L, R\}$ corresponding to the four neighboring cells (4-connectivity). The starting positions of the agents, the evasion states, and the obstacles are shown in Figure 9.

We assume that the level-0 agents pick their actions such that they will not cause an immediate crash, while choosing their action uniformly randomly. We construct the level- k thinking hierarchy and calculate the strategies of both agents at different levels. We then simulate the resulted strategies at some selected level pairs over 1500 games. To illustrate the results, Tables 1 and 2 show the win percentages for each player of the selected scenarios. In Tables 1 and 2, we fix one agent to be level-2 while varying the level of the other agent. In both tables, the highest winning rates are attained at level 3. This result is expected, since level 3 strategies are by definition the best response to the level-2 strategies. It can also be observed that after level 5 the performance only varies slightly, since the strategies at higher levels do not differ much from each other (as they are converging to the NE).

Table 1: Pursuer win percentages against level-2 evader. Results from 1500 simulated games.

	$k^2 = 2$	$k^1 = 1$	$k^1 = 2$	$k^1 = 3$	$k^1 = 4$	$k^1 = 5$	$k^1 = 6$
Pursuer Wins		48.5	47.6	51.7	50.9	51.2	51.1
Due to Capture		36.4	38.2	45.3	43.1	42.7	42.9

Table 2: Evader win percentages against level-2 pursuer. Results from 1500 simulated games.

	$k^1 = 2$	$k^2 = 1$	$k^2 = 2$	$k^2 = 3$	$k^2 = 4$	$k^2 = 5$	$k^2 = 6$
Evader Wins		47.5	52.4	53.8	52.9	53.2	53.1
Due to Evasion		35.6	42.2	45.3	44.5	44.7	44.2

In Figure 10 two sample trajectories are illustrated. The first trajectory depicts a level-3 Evader against a level-2 Pursuer. One observes the “deceiving” behavior of

the more sophisticated Evader in this case: it first moves towards evasion state #2 (on the right), which tricks the level-2 Pursuer to also go right and take the shorter route beneath the obstacles to defend evade state #2. The Evader then suddenly turns left and goes to the evade state #1 (on the left). When the Evader reveals that its true intention is move towards evasion state #1, it is too late for the Pursuer to capture the Evader. The second trajectory depicts a successful capture. Different from the previous scenario, the level-3 Pursuer has perfect information about the strategy of the level-2 Evader, and as a result it can predict well what the Evader will do in the next time step. The Pursuer then simply acts accordingly to maximize the probability of successful capture.

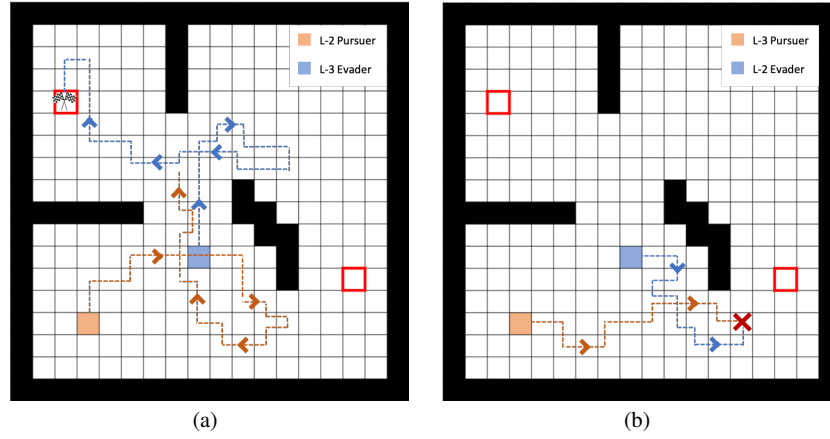


Fig. 10: Sample trajectories of agents playing fixed rationality levels. The first game is won by the evader; the second is won by the pursuer. (a) Level-2 pursuer vs. level-3 evader; (b) Level-3 pursuer vs. level-2 evader.

6 Conclusions

Future autonomous systems will integrate data-driven and model-based methods for perception, learning, planning, and control. Current algorithms in these areas are computationally intensive, a fact that hinders their use in many applications that require fast (re)planning and decision-making, especially under uncertain, time-varying, and complex environments. Enabling “human-like” planning (e.g., flexible, robust, resilient, in sync with changing situations) requires a different look, beyond the current practice which is heavily based on the assumption that the decision-maker is a rational agent who is a perfect optimizer. That is, in the majority of current theory assumes that the agent has ample computational and/or time resources to process

all information and compute the optimal solution unhindered. In many practical applications this is a strong and often unrealistic assumption. For example, biological systems are adaptive in nature and excel at finding solutions to complex problems that may not be globally optimal but are “good enough” given the constraints and/or available sensory information.

The lingering issue of how to best deal with the computational complexity for “human-like autonomy,” especially under strict timing constraints suggests an alternative approach that abandons the perfect optimality/rationality paradigm for a bounded-rationality one. Bounded rationality is supported by ample empirical evidence and captures better human decision-making than the more traditional maximum expected utility (MEU) paradigm.

The bounded-rational paradigm is especially relevant when the agent operates in uncertain, complex, dynamic environments under limited time, computational resources, or energy. It is a good option, in particular, when strong priors (heuristics) are available. A great source of heuristics is via abstractions that organize the collected information according to their relevance to the task at hand. In fact, a distinctive property of human intelligence is its ability to form abstractions by neglecting irrelevant information, allowing to separate structure from noise and allocate attention and processing capabilities tailored to the given resource constraints.

In this work we first provide an overview of the existing framework for bounded-rational decision-making based on the variational free-energy principle that allows the construction of state abstractions, which can then be utilized to solve sequential decision making problems subject to computational and informational constraints. We discuss the connection of the proposed approach with information-theoretic signal compression (i.e., the Information Bottleneck principle), and formulate a novel optimization problem to obtain tree-based abstractions for path-planning problems as a function of the agent’s computational resources. Finally, we investigate the effect of bounded rationality in pursuit-evasion stochastic games. Specifically, we show that limited-depth iterative best response strategies (e.g., the “theory of mind”) avoid the infinite regress implied for a Nash equilibrium and can provide implementable solutions to the problem at low computational complexity.

We believe that there are still many avenues open for investigation in this area. Task-aware and/or computation-aware algorithms for perception, planning, and learning with provable performance guarantees are indispensable for all autonomous systems. In particular, we argue that the bounded rational model is highly relevant for the development of the next generation of autonomous systems, having (expert) human-like perception and decision-making capabilities, especially when these autonomous agents need to interact with humans.

Acknowledgment

The author would like to acknowledge Daniel Larsson, Yue Guan, Daniel Braun, Pedro Ortega, and Dipankar Maity for many insightful discussions. Support for this

work has been provided by NSF award 1849130, by ONR awards N00014-13-0563, N00014-18-12375, N00014-18-1-2828, and by ARL under DCIST CRA W911NF-17-2-018.

References

1. Arad, A., Rubinstein, A.: The 11-20 money request game: A level-k reasoning study. *American Economic Review* **102**(7), 3561–3573 (2012)
2. Aumann, R.J.: Subjectivity and correlation in randomized strategies. *Journal of Mathematical Economics* **1**, 67–96 (1974)
3. Aumann, R.J.: Correlated equilibrium as an expression of Bayesian rationality. *Econometrica: Journal of the Econometric Society* **55**(1), 1–18 (1987)
4. Aumann, R.J.: Rationality and bounded rationality. *Games and Economic Behavior* **21**(1-2), 2–14 (1997)
5. Başar, T., Olsder, G.: *Dynamic Noncooperative Game Theory: Second Edition. Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (1999)
6. Balakrishnan, A.V.: *Stochastic Differential Systems I*. Springer-Verlag (1973)
7. Behn, R., Yu-Chi Ho: On a class of linear stochastic differential games. *IEEE Transactions on Automatic Control* **13**(3), 227–240 (1968). DOI 10.1109/TAC.1968.1098898
8. Behnke, S.: Local multiresolution path planning. *Lecture Notes in Artificial Intelligence* **3020**(1), 332–343 (2004)
9. Bellman, R.: *Dynamic Programming*. Princeton University Press, Princeton, NJ (1957)
10. Bennett, C.H.: The thermodynamics of computation - a review. *International Journal of Theoretical Physics* **21**(12), 905–940 (1982)
11. Bertsekas, D.P.: *Dynamic Programming and Optimal Control*, vol. 2, 4th edn. Athena Scientific (2012)
12. Bodis-Wollner, I.: Pre-emptive perception. *Perception* **37**, 462–478 (2008). DOI doi:10.1068/p5880
13. Bopardikar, S.D., Bullo, F., Hespanha, J.P.: On discrete-time pursuit-evasion games with sensing limitations. *IEEE Transactions on Robotics* **24**(6), 1429–1439 (2008)
14. Borji, A., Sihite, D., Itti, L.: Modeling task-driven visual attention. In: *Proceedings of the British Machine Vision Conference*. Dundee, Scotland (2011)
15. Braun, D.A., Ortega, P.A., Theodorou, E., Schaal, S.: Path integral control and bounded rationality. Paris, France (2011)
16. Camerer, C.F., Ho, T.H., Chong, J.K.: A cognitive hierarchy model of games. *The Quarterly Journal of Economics* **119**(3), 861–898 (2004)
17. Cover, T.M., Thomas, J.A.: *Elements of Information Theory*. John Wiley & Sons (2006)
18. Cowlagi, R., Tsiotras, P.: Hierarchical motion planning with kinodynamic feasibility guarantees. *IEEE Transactions on Robotics* **28**(2), 379–395 (2012)
19. Cowlagi, R., Tsiotras, P.: Multiresolution path- and motion planning for autonomous agents via wavelet-based cell decompositions. *IEEE Transactions on Systems Man and Cybernetics, Part B: Cybernetics* **42**(5), 1455–1469 (2012)
20. Daskalakis, C., Goldberg, P.W., Papadimitriou, C.H.: The complexity of computing a Nash equilibrium. *SIAM Journal on Computing* **39**(1), 195–259 (2009)
21. Deco, G., Heinke, D.: Attention and spatial resolution: A theoretical and experimental study of visual search in hierarchical patterns. *Perception* **36**, 335–354 (2007)
22. Einhorn, E., Schröter, C., Gross, H.M.: Finding the adequate resolution for grid mapping - cell sizes locally adapting on-the-fly. In: *IEEE Conference on Robotics and Automation*. Shanghai, China (2011)
23. Feldman, H., Friston, K.J.: Attention, uncertainty, and free-energy. *Frontier in Human Neuroscience* **4**(215) (2010)

24. Feynman, R.P.: Feynman Lectures on Computation. Addison-Wesley Longman Publishing Co., Inc. (1998)
25. Filar, J., Vrieze, K.: Competitive Markov Decision Processes. Springer-Verlag, Berlin, Heidelberg (1996)
26. Frith, U., Frith, C.D.: Development and neurophysiology of mentalizing. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences* **358**(1431), 459–473 (2003)
27. Genewein, T., Leibfried, F., Grau-Moya, J., Braun, D.A.: Bounded rationality, abstraction, and hierarchical decision-making: An information-theoretic optimality principle. *Frontiers in Robotics and AI* **2**, 27 (2015)
28. Gilboa, I., Zemel, E.: Nash and correlated equilibria: Some complexity considerations. *Games and Economic Behavior* **1**(1), 80–93 (1989)
29. Gilovich, T., Griffin, D., Kahneman, D.: *Heuristics and Biases: The Psychology of Intuitive Judgment*. Cambridge University Press (2002)
30. Goodrich, M.A., Stirling, W.C., Frost, R.L.: A theory of satisficing decisions and control. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* **28**(6), 763–779 (1998)
31. Grafton, S.T., Hamilton, A.: Evidence for a distributed hierarchy of action representation in the brain. *Human Movement Science* **26**(4), 590–616 (2007)
32. Grau-Moya, J., Leibfried, F., Genewein, T., Braun, D.A.: Planning with information-processing constraints and model uncertainty in Markov decision processes. In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 475–491. Springer (2016)
33. Harsanyi, J.C.: *Rational Behavior and Bargaining Equilibrium in Games and Situations*. Cambridge, UK (1977)
34. Hassanpour, S., Wübben, D., Dekorsy, A.: Overview and investigation of algorithms for the information bottleneck method. In: *International ITG Conference on Systems, Communications and Coding*, pp. 1–6. Hamburg, Germany (2017)
35. Hauer, F., Kundu, A., Rehg, J.M., Tsiotras, P.: Multi-scale perception and path planning on probabilistic obstacle maps. In: *IEEE International Conference on Robotics and Automation*, pp. 4210–4215. Seattle, WA (2015)
36. Hauer, F., Tsiotras, P.: Reduced complexity multi-scale path-planning on probabilistic maps. In: *IEEE Conference on Robotics and Automation*, pp. 83–88. Stockholm, Sweden (2016)
37. Hespanha, J.P., Hyoun Jin Kim, Sastry, S.: Multiple-agent probabilistic pursuit-evasion games. In: *Proceedings of the 38th IEEE Conference on Decision and Control*, vol. 3, pp. 2432–2437 vol.3 (1999)
38. Ho, T.H., Su, X.: A dynamic level-k model in sequential games. *Management Science* **59**(2), 452–469 (2013)
39. Ho, Y., Bryson, A., Baron, S.: Differential games and optimal pursuit-evasion strategies. *IEEE Transactions on Automatic Control* **10**(4), 385–389 (1965)
40. Horvitz, E., Zilberstein, S.: Computational tradeoffs under bounded resources. *Artificial Intelligence* **126**(1), 1–4 (2001)
41. Isaacs, R.: *Differential Games*. New York: John Wiley and Sons (1965)
42. Itti, L., Baldi, P.: Bayesian surprise attracts human attention. *Vision Research* **49**(10), 1295–1306 (2009)
43. Larsson, D., Maity, D., Tsiotras, P.: Q-search trees: An information-theoretic approach towards hierarchical abstractions for agents with computational limitations (2019). URL <https://arxiv.org/abs/1910.00063>
44. Larsson, D.T., Braun, D., Tsiotras, P.: Hierarchical state abstractions for decision-making problems with computational constraints. In: *56th IEEE Conference on Decision and Control*, pp. 1138–1143. Melbourne, Australia (2017)
45. Lee, T.S., Mumford, D.: Hierarchical Bayesian inference in the visual cortex. *Journal of Optical Society of America A* **20**(7), 1434–1448 (2003)
46. Li, J., Kendall, G., John, R.: Computing Nash equilibria and evolutionarily stable states of evolutionary games. *IEEE Transactions on Evolutionary Computation* **20**(3), 460–469 (2016)

47. Lin, J.: Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory* **37**(1), 145–151 (1991)
48. Lin, W., Qu, Z., Simaan, M.A.: Nash strategies for pursuit-evasion differential games involving limited observations. *IEEE Transactions on Aerospace and Electronic Systems* **51**(2), 1347–1356 (2015)
49. Lipman, B.L.: *Information Processing and Bounded Rationality: a Survey*, vol. 28, pp. 42–67. Wiley on behalf of the Canadian Economics Association (1995)
50. Nelson, E., Corah, M., Michael, N.: Environment model adaptation for mobile robot exploration. *Auton Robot* **42**, 257–272 (2015)
51. von Neumann, J., Morgenstern, O.: *Theory of Games and Economic Behavior*. Princeton University Press, Princeton, NJ (1945)
52. Nguyen, A., Yosinski, J., Clune, J.: Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 427–436. Boston, MA (2015)
53. Orega, P.A., Braun, D.A.: *Information, Utility and Bounded Rationality*, pp. 269–274. Springer-Verlag Berlin Heidelberg (2011)
54. Ortega, P., Stocker, A.: Human decision-making under limited time (2016). URL <http://arXiv:1610.01698v1>
55. Ortega, P.A., Braun, D.A.: Thermodynamics as a theory of decision-making with information-processing costs. In: *Proceedings of the Royal Society*. Royal Society (2013)
56. Rubin, J., Shamir, O., Tishby, N.: Trading value and information in MDPs. In: *Decision Making with Imperfect Decision Makers*, chap. 3, pp. 57 – 74. Springer-Verlag Heidelberg (2012)
57. Rubinstein, A.: *Modeling Bounded Rationality*. MIT press (1998)
58. Shinar, J., Gutman, S.: Three-dimensional optimal pursuit and evasion with bounded controls. *IEEE Transactions on Automatic Control* **25**(3), 492–496 (1980)
59. Simon, H.A.: *Models of Bounded Rationality*, vol. 3. MIT Press (1982)
60. Slonim, N.: The information bottleneck: Theory and applications. Ph.D. thesis, The Hebrew University (2002)
61. Slonim, N., Tishby, N.: Agglomerative information bottleneck. In: *Advances in Neural Information Processing*, pp. 617–623. Denver, CO (2000)
62. Strouse, D.J., Schwab, D.J.: The deterministic information bottleneck. *Neural Computation* **29**, 1611–1630 (2017)
63. Sutton, R.S., Barto, A.G.: *Reinforcement Learning: An Introduction*. The MIT Press (1998)
64. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks (2013). URL <https://arxiv.org/abs/1312.6199>
65. Taylor, P., Hobbs, J.N., Burroni, J., Siegelmann, H.T.: The global landscape of cognition: hierarchical aggregation as an organizational principle of human cortical networks and functions. *Scientific Reports* **5**, 18112 (2015)
66. Tishby, N., Pereira, F.C., Bialek, W.: The information bottleneck method. In: *The 37th Annual Allerton Conference on Communication, Control and Computing*, pp. 368–377. Monticello, IL (1999)
67. Tishby, N., Polani, D.: Information theory of decisions and actions. In: *Perception-Action Cycle: Model, Architectures and Hardware*, Springer Series in Cognitive and Neural Systems 1, chap. 19, pp. 601 – 636. Springer Science & Business Media (2011)
68. Tsiotras, P., Jung, D., Bakolas, E.: Multiresolution hierarchical path-planning for small UAVs using wavelet decompositions. *Journal of Intelligent and Robotic Systems* **66**(4), 505–522 (2012)
69. Vidal, R., Shakernia, O., Kim, H.J., Shim, D.H., Sastry, S.: Probabilistic pursuit-evasion games: theory, implementation, and experimental evaluation. *IEEE Transactions on Robotics and Automation* **18**(5), 662–669 (2002)
70. Wierzbicki, A.P.: A mathematical basis for satisficing decision making. *Mathematical Modelling* **3**(5), 391–405 (1982)
71. Willman, W.: Formal solutions for a class of stochastic pursuit-evasion games. *IEEE Transactions on Automatic Control* **14**(5), 504–509 (1969)

- 72. Yoshida, W., Dolan, R.J., Friston, K.J.: Game theory of mind. *PLOS Computational Biology* **4**(12), 1–14 (2008)
- 73. Zilberstein, S.: Satisficing and bounded optimality. In: *AAAI Spring Symposium on Satisficing Models*, pp. 91–94 (1998)