The Limits of Learning and Planning: Minimal Sufficient Information Transition Systems

Basak Sakcak, Vadim Weinstein, Steven M. LaValle[*]

Center of Ubiquitous Computing
Faculty of Information Technology and Electrical Engineering
University of Oulu, Finland

Abstract. In this paper, we view a policy or plan as a transition system over a space of information states that reflect a robot's or other observer's perspective based on limited sensing, memory, computation, and actuation. Regardless of whether policies are obtained by learning algorithms, planning algorithms, or human insight, we want to know the limits of feasibility for given robot hardware and tasks. Toward the quest to find the best policies, we establish in a general setting that minimal information transition systems (ITSs) exist up to reasonable equivalence assumptions, and are unique under some general conditions. We then apply the theory to generate new insights into several problems, including optimal sensor fusion/filtering, solving basic planning tasks, and finding minimal representations for feasible policies.

Keywords: planning, sensing uncertainty, information spaces, theoretical foundations

1 Introduction

Robotics increasingly appears as an application area for other fields. It is a frequent target for designing and testing machine learning algorithms, planning algorithms, sensor fusion methods, control laws, and so on. This may lead many to believe that robotics itself does not have its own, unique theoretical core (on this, we agree with Koditschek [6]). Thus, are we missing something? Surely robotics is not a pure algorithmic problem or pure nonlinear control problem. Could there be a theory that plays a similar role to Turing machines for computer science, or $\dot{x} = f(x, u)$ over differentiable manifolds for control theory, and yet is distinct from both? Can we formulate and potentially answer questions such as: Does a solution even exist to the problem? What are the minimal necessary components to solve it? What should the best learning approach imaginable
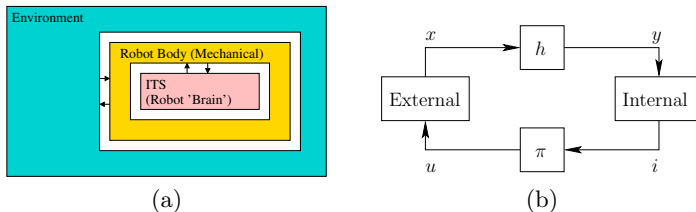

Fig. 1. (a)The *internal* robot brain is defined as an ITS that interacts with the *external* world (robot body and environment). (b)Coupled internal and external systems mathematically capture sensing, actuation, internal computation, and the external world.

produce as a representation? Such questions would be analogous to existence and uniqueness in control and dynamical systems, or decidability and complexity (especially Kolomogorov) in theoretical computer science.

This paper proposes a robotics theory that is built from the input-output relationships between a programmable mechanical system (robot) and its environment via sensing and actuation; see Figure 1(a). The key is to focus on necessary and sufficient conditions that a robot's internal processor (i.e., "brain") must maintain to solve required tasks, such as collision-free navigation or coverage. We assume that the robot hardware and actuation model are fixed, and that for a given set of tasks in a space of environments, we must determine the weakest amount of sensing, actuation, and computation that would be sufficient for solving tasks. We will call such conditions *minimal sufficient*: If you take away anything from a minimal sufficient system, the tasks will become unsolvable.

We introduce the notion of an *information transition system* (ITS) to formally model the robot's brain (as well as any other system observers). The "information" part of an ITS is inspired by von Neumann's definitions in the context of sequential games with hidden information (and *not* Shannon's later notion of information theory). This inspired the development of *information spaces* [8] (Chapter 11) as a foundation of planning with imperfect state information due to sensing uncertainty. The concept of *sufficient information mappings* appears therein. It is generalized in this paper, and the state space of each ITS will in fact be an information space.

In our work, the ITS and its underlying information space serve as the domain over which a plan or policy can be expressed and analyzed. Note that prior work in planning usually assumes that the space one plans over is fixed, as in a configuration space or state (phase) space based on the robot's mobility. Even the information spaces described in [8] remain fixed in the planning phase. A notable exception is by O'Kane and Shell [11], in which information spaces for passive filtering are reduced algorithmically, and is closely connected to this paper. All such spaces will be considered here as potential information spaces, and we intend to reduce or collapse them as much as possible in the development of an information-feedback plan. This is perhaps closer to the goals of machine learning, in which candidate representations are determined through optimization of discrepancies with respect to input-output data. In this paper, we in fact consider both model-free and model-based ITSs, in alignment with the choices commonly found in machine learning [2].

The robot's ITS is coupled to the physical world, which is itself modeled as a transition system. Note that the physical world model is not "given" to the robot; we will formalize notions of "who gets what" information in Section 3. The coupled system is inspired by neuroscience models (for example, [4]). Many of the concepts in this paper build upon [12], in which we recently proposed an enactivist-oriented model of cognition based on information spaces. By *enactivist* [5], it is meant that the necessary brain structures emerge from sensorimotor interaction and do not necessarily have predetermined, meaningful components as in the classical representationalist sense.

Section 2 provides a mathematical formulation of robot-environment interaction as transition systems. Section 3 develops notions of sufficiency and minimality over the space of possible ITSs. Section 4 applies the general concepts to address what it means to solve both passive (filtering) and active (planning/control) tasks minimally and Section 5 provides simple examples.

2 Mathematical Models of Robot-Environment Systems

Internal and external systems In this paper, we will consider a robot embedded in an environment and describe this system as two subsystems, named *internal* and *external*, connected through symmetric input-output relations. External refers to the system describing the physical world, and internal is the complement of it. This interaction is shown in Figure 1(b). In this sense, the states of the external and internal systems are similar to the use of the term in control theory and computer science, respectively.

External system corresponds to the totality of the environment and the robot body within it. Let X denote the set of states of this system; a state could be for example, the configuration of the robot in a known environment (or within a set of possible environments) or its phase. There are no restrictions on X; it may be discrete, an n-dimensional manifold, a function space, and so on. Next, let U be the set of control inputs (also referred to as actions) such that when applied at state $x \in X$ causes it to change according to a state transition function $f : X \times U \to X$. The set U can also be anything: a finite or infinite discrete set, a compact or non-compact manifold, and so on. Similarly, at each state x, $y = h(x)$ is the output in which $h : X \to Y$ is a state-based sensor mapping and Y is the set of all possible observations.

The internal system (robot's brain) observes the external system through a sensor mapping and interacts with it through a selection of actions with respect to a policy (alternatively, we can call it a plan or a strategy). Therefore, the input to the internal system is an observation and its output is an action. The states of this system correspond to the retained information gathered through the outcomes of actions in terms of sensor observations. To this end, the basis of our mathematical formulation of the internal is the notion of information space *(I-space)* [8]. We will use the term information state *(I-state)* to refer to the state of the internal system and denote it with ι, and $\mathcal{I}$ will denote the set of all I-states, that is, the *I-space*. Note that the notions I-space and I-state are not

exclusive to the internal system and we will use them in a more general setting in the following sections. Similar to the external system, the internal system evolves with each $y \in Y$ according to the information transition function $\phi : \mathcal{I} \times Y \to \mathcal{I}$. The output then corresponds to the control command given by an information feedback policy $\pi : \mathcal{I} \to U$.

Finally, we can write the coupled dynamical system composed of these two subsystems defined as external and internal as

$$
\begin{aligned}
x' &= f(x, \pi(\iota)) & y &= h(x) \\
\iota' &= \phi(\iota, h(x)) & u &= \pi(\iota).
\end{aligned}
\tag{1}
$$

Whereas the equations on the left side describe the evolution of this coupled system, the ones on the right show the respective outputs of each subsystem. Given an initial state $(x_1, \iota_1) \in X \times \mathcal{I}$, there exists a unique state-trajectory.

Suppose the system evolves in discrete stages. For the external system, starting from an initial state x_1, each stage k corresponds to applying an action u_k which then yields the next stage $k+1$ and the next state $x_{k+1} = f(x_k, u_k)$. As the system evolves through stages, $\tilde{x}_k = (x_1, x_2, \ldots, x_k)$, $\tilde{u}_{k-1} = (u_1, u_2, \ldots, u_{k-1})$, $\tilde{y}_k = (y_1, y_2, \ldots, y_k)$, correspond to the state, action and observation histories up to stage k, respectively. Note that applying the action u_k at stage k would result in a transition to state x_{k+1} and the corresponding sensor reading $y_{k+1} = h(x_{k+1})$. The same applies for the internal system, we can describe its evolution, starting from an initial I-state ι_0 following the state transition equation $\iota_k = \phi(\iota_{k-1}, y_k)$. At stage k, $\pi(\iota_k)$ would produce the action u_k. Note that the stage index of the I-state starts from 0, this corresponds to any prior information the internal system might have regarding the external; ι_1 is then obtained using ι_0 and y_1. Furthermore, $\pi(\iota_0) = u_0 = ()$ for all policies π, meaning that no action is outputted at this stage.

Generalizing to transition systems Without loss of generality, we can describe the internal and external subsystems as transition systems of the form (S, Λ, T) in which S is the set of states, Λ is the set of names for the outgoing transitions, and $T \subset S \times \Lambda \times S$ is a ternary relation describing the transitions. If for each $(s, \lambda) \in S \times \Lambda$ there is a unique $s' \in S$ such that $(s, \lambda, s') \in T$, then we will write this system as (S, Λ, τ) in which $\tau : S \times \Lambda \to S$ is a function, and call the system an *automaton*. This corresponds to a deterministic system. Note that our definition of an automaton differs from the one usually used in computer science in the sense that ours do not necessarily have a start state and a set of accepting states, and it is not necessarily finite. Suppose $\mathcal{T} : S \times \Lambda \to \mathrm{pow}(S)$, in which $\mathrm{pow}(\cdot)$ denotes the power set. Then, the transition system $(S, \Lambda, \mathcal{T})$ is a nondeterministic automaton.

In [12], we have used the notion of state-relabeled transition systems to model the internal and external systems. A state-relabeled transition system is the quintuple $(S, \Lambda, T, \sigma, L)$ in which $\sigma : S \to L$ is a labeling function and (S, Λ, T) is a transition system. Preimages of a labeling function σ induce a partitioning of the state space S. Let S/σ be the set of equivalence classes $[s]_\sigma$ induced by

σ such that $S/\sigma = \{[s]_\sigma \mid s \in S\}$ and $[s]_\sigma = \{s' \in S \mid \sigma(s') = \sigma(s)\}$. Then, we can define a new transition system $(S/\sigma, \Lambda, T/\sigma)$ called the *quotient* of (S, Λ, T) by σ, in which $T/\sigma = \{([s]_\sigma, \lambda, [s']_\sigma) \mid (s, \lambda, s') \in T\}$. Note that $(S/\sigma, \Lambda, T/\sigma)$ is a reduced version of (S, Λ, T). We might be interested in finding a labeling function σ such that the corresponding quotient transition system is as simple as possible while ensuring that it is still useful. In the following sections, we will provide motivations for a reduction and discuss in more detail the requirements on σ for the quotient system to be useful.

Considering the deterministic case and the description of an automaton given above, external and internal systems can be written as state-relabeled automata (X, U, f, h, Y) and $(\mathcal{I}, Y, \phi, \pi, U)$, respectively, in which h and π are considered as labeling functions. Interpreting the labels as the output of a transition system, coupled internal-external system can be described in terms of the state-relabeled transition systems formulation too such that output of one transition system is an input for another. Described this way, coupling of two transition systems result in unique paths in either automaton initialized at a particular state.

3   Sufficient Information Transition Systems

3.1   Information transition systems

In the general setting, an I-state corresponds to the available (stored) information at a certain stage with respect to the action and observation histories. Consequently, an I-space refers to the collection of all possible I-states. We will use the term information transition system *(ITS)* to refer to a transition system whose state space is an I-space. We have already used the notion of I-space while modeling the internal system representing the robot brain, which makes it an ITS. Here, we extend the notion of an ITS to include different perspectives from which the external and the coupled system is viewed. In particular, we identified three perspectives corresponding to 1) a plan executor which corresponds to the robot brain 2) a planner, and 3) an (independent) observer. With a slight abuse of previously introduced notation and terminology, we will use the term "internal" to refer to any system that is not the external and we will use $\mathcal{I}$ to denote a generic I-space. We describe an ITS in a robot-centric way such that an observation will refer to a sensor-reading, that is, y. However, an independent observer defined over the coupled system can observe, at stage k, both the action taken u_{k-1} and the corresponding sensor-reading y_k.

Recall that the information regarding the external is obtained through the sensor-mapping and any potential prior knowledge. Suppose that no policy is fixed over the I-space. Then, the corresponding internal system can be modeled as an ITS of the form $(\mathcal{I}, U \times Y, \phi)$, in which $\phi : \mathcal{I} \times (U \times Y) \to \mathcal{I}$ is a state (information) transition function, if it is deterministic. We will then use the term *deterministic information transition system* (DITS) to refer to them. Otherwise, it is called a *nondeterministic information transition system* (NITS) and described as $(\mathcal{I}, U \times Y, \Phi)$, in which $\Phi \subseteq \mathcal{I} \times (U \times Y) \times \mathcal{I}$ is the transition relation. This formulation corresponds to the perspectives other than the plan

executor such that it is possible to take any action from an I-state as it is not constrained by a policy, hence the outgoing transitions are determined by the elements of $U \times Y$.

A plan executor corresponds to the internal system (robot's brain) described in the previous section. The only information regarding the external is gained through manipulating the state of the external system through actions and obtaining the corresponding sensor readings. Recall the representation used in the previous section, that is, $(\mathcal{I}, Y, \phi)$, and $\pi : \mathcal{I} \to U$ a labeling function. This can be considered as a constrained version of the DITS described in the previous paragraph such that the transitions are restricted to those that can be realized under π. To show that, we augment the definition of internal system corresponding to the robot brain such that the transitions now also correspond to labels. Let $(\mathcal{I}, U \times Y, \Phi)$ be the augmented transition system describing the internal such that

$$\Phi = \{(\iota, (u, y), \iota') \in \mathcal{I} \times (U \times Y) \times \mathcal{I} \mid u = \pi(\iota) \wedge \iota' = \phi(\iota, y)\}, \qquad (2)$$

by construction, this augmented ITS is also deterministic[1]. Suppose $(\mathcal{I}, U \times Y, \Phi')$ is the DITS that is not constrained by a policy. Then, $\Phi \subseteq \Phi'$.

3.2   History information spaces

An I-space constitutes the state space of an ITS. Therefore, we describe the basic I-space named *history I-space* denoted as $\mathcal{I}_{hist}$. It will be used to derive other I-spaces as well. A *history I-state* at stage k corresponds to all the information that is gathered through sensing (and potentially also through actions) up to stage k assuming perfect memory. Let η_k denote the history I-state at stage k, that is $\eta_k = (\eta_0, \tilde{u}_{k-1}, \tilde{y}_k)$, in which η_0 is the initial condition. Recall that $\tilde{u}_0$ is assumed to be the null-tuple, hence, $\tilde{u}_k$ starts with u_1 for any $k > 1$.

Let $\mathcal{I}_0$ be the set of initial conditions whose description varies with the available prior information. We defer the descriptions of possible $\mathcal{I}_0$ to the following paragraph. The history information space at stage k is expressed as $\mathcal{I}_k = \mathcal{I}_0 \times \tilde{U}_{k-1} \times \tilde{Y}_k$. In general, the number of stages that the system will go through is not fixed. Therefore, we can define history I-space as the union over all $k \in \mathbb{N}$, that is, $\mathcal{I}_{hist} = \bigcup_{k \in \mathbb{N}} \mathcal{I}_k$. The DITS corresponding to $\mathcal{I}_{hist}$ becomes $(\mathcal{I}_{hist}, U \times Y, \phi_{hist})$, in which

$$\eta_k = \phi_{hist}(\eta_{k-1}, u_{k-1}, y_k) = \eta_{k-1} \frown u_{k-1} \frown y_k$$

---

[1] We could use the same approach for the external system too. In that case, let $(X, Y \times U, F)$ be this augmented transition system corresponding to the external, in which F is the set of transitions such that $F = \{(x, (y, u), x') \in X \times (Y \times U) \times X \mid y = h(x) \wedge x' = f(x, u)\}$. Further creating bipartite graphs (for either system) such that transitions from a state correspond either to an observation $y \in Y$ or to an action $u \in U$ allows us to describe the coupling as a form of intersecting two automata. However, because it is not central to this paper we will not elaborate on this topic.

and $\frown$ is the concatenation operation that adds an element at the end of a sequence.

We consider two categories of initial conditions depending on whether information regarding the state space X of the external system is available or not. Suppose X or any information regarding X is not given. Then, an I-state at stage k simply is $\eta_k = (\tilde{u}_{k-1}, \tilde{y}_k)$, that is, the concatenation of action and observation histories up till stage k. We call this type of history I-space, the *model-free history I-space*, and respectively call the corresponding ITS, *model-free history ITS*. In this case, we can treat η_0 as $\eta_0 = ()$. Thus, $\mathcal{I}_0 = \{()\}$. For the second category of initial conditions, full or partial information regarding X, against which the actions and observations can be interpreted, is given. We will then use the terms *model-based history I-space* and *model-based history ITS* to refer to the respective I-space and ITS. The initial condition η_0 could be (i) a known state $x_1 \in X$ such that $\mathcal{I}_0 = X$, (ii) a set of possible initial states $X_1 \subset X$ such that $\mathcal{I}_0 = \mathrm{pow}(X)$ or (iii) a probability distribution $P(x_1)$ over X such that $\mathcal{I}_0 \subseteq \mathcal{P}(X)$, in which $\mathcal{P}(X)$ is the set of all probability distributions over X.

3.3 Sufficient state-relabeling

In [12] we have introduced a notion of *sufficiency* that substantially generalizes the definition in Chapter 11 of [8] and is presented here for completeness.

Definition 1 (Sufficient state-relabeling). *Let (S, Λ, T) be a transition system. A labeling function $\sigma : S \to L$ defined over the states of a transition system is sufficient if and only if for all $s, q, s', q' \in S$ and all $\lambda \in \Lambda$, the following implication holds:*

$$\sigma(s) = \sigma(q) \wedge (s, \lambda, s') \in T \wedge (q, \lambda, q') \in T \implies \sigma(s') = \sigma(q').$$

If σ is defined over the states of an automaton (S, Λ, τ), then σ is sufficient iff for all $s, q \in S$ and all $\lambda \in \Lambda$, $\sigma(s) = \sigma(q)$ implies that $\sigma(\tau(s, \lambda)) = \sigma(\tau(q, \lambda))$.

Consider the stage-based evolution of external system (X, U, f, h, Y) with respect to the action (control input) sequence $\tilde{u}_{k-1} = (u_1, \ldots, u_{k-1})$. This corresponds to the state and observation histories till stage k, that are $\tilde{x}_k = (x_1, \ldots, x_k)$ and $\tilde{y}_k = (y_1, \ldots, y_k)$. Recall that applying u_k at stage k would result in a transition to x_{k+1} and the corresponding observation $y_{k+1} = h(x_{k+1})$. Hence, in this context, sufficiency of h implies that given the label $y_k = h(x_k)$ and the action u_k, it is possible to determine the label $y_{k+1} = h(x_{k+1})$. One interpretation of sufficiency of h is that the respective quotient system sufficiently represents the underlying system up to the induced equivalence classes. This notion is similar to minimal realization of a system, that is, the minimal state space description that models the given input-output measurements (see for example [7]). Second interpretation is in a predictive sense. Suppose the quotient system is known. Then, the label $y_{k+1} = h(x_{k+1})$ can be determined before the system gets to x_{k+1}, using the current label y_k and the action to be applied u_k. Furthermore, under a fixed policy, complete observation-trajectory can be determined from the initial observation by induction.

Now, consider an internal system with a labeling function $\kappa : \mathcal{I} \to \mathcal{I}'$, that is, $(\mathcal{I}, U \times Y, \phi, \kappa, \mathcal{I}')$, and its evolution with respect to the observation history $\tilde{y} = (y_1, \ldots, y_k)$. At stage k, the state of the automaton is ι_k and with (u_k, y_{k+1}) the system transitions to $\iota_{k+1} = \phi(\iota_k, u_k, y_{k+1})$. Sufficiency of κ implies that given $\kappa(\iota_k)$, u_k, and y_{k+1}, we can determine $\kappa(\iota_{k+1})$. This is equivalent to the definition introduced in Chapter 11 of [8] and makes it a special case for Definition 1.

3.4   Derived information transition systems

Even though it seems natural to rely on a history ITS, dimension of a history I-space increases linearly with each stage, making it impractical in most cases. Thus, we are interested in defining a reduced ITS that is more manageable.Furthermore, this would largely simplify the description of a policy for a planner or a plan executor.

Recall the quotient of a transition system by a a labeling function. We rewrite $(\mathcal{I}_{hist}, U \times Y, \phi_{hist})$ as $(\mathcal{I}_{hist}, U \times Y, \Phi_{hist})$, in which

$$\Phi_{hist} = \{(\eta, (u, y), \phi_{hist}(\eta, u, y)) \in \mathcal{I}_{hist} \times (U \times Y) \times \mathcal{I}_{hist}\}. \tag{3}$$

We can introduce an *information mapping* (I-map) $\kappa : \mathcal{I}_{hist} \to \mathcal{I}_{der}$ that categorizes the states of $\mathcal{I}_{hist}$ into equivalence classes through its preimages. In this case, κ serves as a labeling function and a reduction can be obtained in terms of the quotient of $(\mathcal{I}_{hist}, U \times Y, \Phi)$ by κ, that is, $(\mathcal{I}_{hist}/\kappa, U \times Y, \Phi/\kappa)$.

It is crucial that the derived ITS is a DITS so that the labels can be determined using only the derived ITS without making reference to the history ITS.Considering the quotient system derived from $(\mathcal{I}_{hist}, U \times Y, \phi)$, which is a DITS by definition, by κ, we can not always guarantee that the resulting ITS is deterministic. This depends on the I-map used for state-relabeling as stated in the following proposition.

Proposition 1. *For all non-empty U and Y, and for the corresponding $\mathcal{I}_{hist}$, there exists a labeling function κ such that the quotient of $(\mathcal{I}_{hist}, U \times Y, \phi)$ by κ, that is, $(\mathcal{I}_{hist}/\kappa, U \times Y, \Phi/\kappa)$, in which Φ is defined as in (3), is not a DITS.*

Proof. Let $\kappa : \mathcal{I}_{hist} \to \{l_1, l_2\}$ such that $\kappa^{-1}(l_1) = \{\eta_k = (\tilde{u}_{k-1}, \tilde{y}_k) \in \mathcal{I}_{hist} \mid \tilde{u}_{k-1} = (u_i)_{i=1,\ldots,k-1}, u_i = u, \forall i = 1, \ldots, k-1\}$ is the set of histories that correspond to applying the same action for $k-1$ times and $\kappa^{-1}(l_2)$ is its complement, that is, $\kappa^{-1}(l_2) = \mathcal{I}_{hist} \backslash \kappa^{-1}(l_1)$. Then, there exist sequences $\eta_{k-2} = (\tilde{u}_{k-3}, \tilde{y}_{k-2})$ and $\eta_{k-1} = (\tilde{u}_{k-2}, \tilde{y}_{k-1})$ such that $\eta_{k-2} = \eta_{k-1} ^\frown (u, y)$ and $\eta_k = \eta_{k-1} ^\frown (u, y)$ for which $\kappa(\eta_{k-2}) = \kappa(\eta_{k-1}) = l_2$ and $\kappa(\eta_k) = l_1$. Thus,

$$\{([\eta_{k-2}]_\kappa, (u, y), [\eta_{k-1}]_\kappa), ([\eta_{k-1}]_\kappa, (u, y), [\eta_k]_\kappa)\} \in \Phi/\kappa.$$

Since $[\eta_{k-2}]_\kappa = [\eta_{k-1}]_\kappa$ and $[\eta_{k-1}]_\kappa \neq [\eta_k]_\kappa$, the transition corresponding to $([\eta_{k-1}]_\kappa, (u, y))$ is not unique; thus, $(\mathcal{I}_{hist}/\kappa, U \times Y, \Phi/\kappa)$ is not deterministic. $\qquad \square$

Note that Proposition 1 holds also in the case of a generic ITS $(\mathcal{I}, U \times Y, \phi)$, with non-history I-states, if $\exists s, s', q, q' \in \mathcal{I}$ such that $\{(s, (u, y), s'), (q, (u, y), q')\} \in \Phi$, in which Φ is defined using ϕ as in (3). Then, any I-map κ such that $\kappa(s) = \kappa(q)$ and $\kappa(s') \neq \kappa(q')$ results in a quotient system that is not a DITS.

For the quotient system derived from $(\mathcal{I}_{hist}, U \times Y, \phi)$ to be a DITS depends on the sufficiency of κ. In [12] it is shown that the quotient of a transition system (S, Λ, T) by a labeling function σ is an automaton (recall our definition) if and only if (S, Λ, T) is full[2] and σ is sufficient. As ϕ_{hist} is a function with domain $\mathcal{I}_{hist} \times (U \times Y)$, it is full, then, the following follows from [12] as a special case.

Proposition 2. *Let $(\mathcal{I}_{hist}/\kappa, U \times Y, \Phi_{hist}/\kappa)$ be the quotient of $(\mathcal{I}_{hist}, U \times Y, \phi_{hist})$ by κ, in which Φ is defined as in (3), then $(\mathcal{I}_{hist}/\kappa, U \times Y, \Phi/\kappa)$ is a DITS if and only if κ is sufficient.*

For an I-map $\kappa : \mathcal{I}_{hist} \rightarrow \mathcal{I}_{der}$, $(\mathcal{I}_{hist}/\kappa, U \times Y, \Phi_{hist}/\kappa)$ is isomorphic to $(\mathcal{I}_{der}, U \times Y, \Phi_{der})$, in which $\Phi_{der} = \{(\kappa(\eta), (u, y), \kappa(\eta')) \mid (\eta, (u, y), \eta') \in \Phi_{hist}\}$ [12]. Thus, we can use the labels introduced by κ as the new (derived) I-space and the corresponding quotient system as the derived ITS. Suppose, κ is sufficient. Then, the derived ITS is a DITS, meaning that given the I-state $\iota_{k-1} \in \mathcal{I}_{der}$, and (u_{k-1}, y_k), $\iota_{k+1} \in \mathcal{I}_{der}$ can be uniquely determined. Consequently, we can write the derived ITS as $(\mathcal{I}_{der}, U \times Y, \phi_{der})$ in which $\phi_{der} : \mathcal{I}_{der} \times (U \times Y) \rightarrow \mathcal{I}_{der}$ is the new information transition function. Therefore, we no longer need to rely on the full histories and the history ITS and can rely solely on the derived ITS. This is shown in the first two rows of the following diagram:

$$
\begin{array}{ccccccccc}
\mathcal{I}_{hist} & \xrightarrow{u_1,y_2} & \mathcal{I}_{hist} & \xrightarrow{u_2,y_3} & \mathcal{I}_{hist} & \xrightarrow{u_3,y_4} & \mathcal{I}_{hist} & \xrightarrow{u_4,y_5} & \mathcal{I}_{hist} \cdots \\
\downarrow{\kappa} & & \downarrow{\kappa} & & \downarrow{\kappa} & & \downarrow{\kappa} & & \downarrow{\kappa} \\
\mathcal{I}_{der} & \xrightarrow{u_1,y_2} & \mathcal{I}_{der} & \xrightarrow{u_2,y_3} & \mathcal{I}_{der} & \xrightarrow{u_3,y_4} & \mathcal{I}_{der} & \xrightarrow{u_4,y_5} & \mathcal{I}_{der} \cdots \\
\downarrow{\kappa'} & & \downarrow{\kappa'} & & \downarrow{\kappa'} & & \downarrow{\kappa'} & & \downarrow{\kappa'} \\
\mathcal{I}_{min} & \xrightarrow{u_1,y_2} & \mathcal{I}_{min} & \xrightarrow{u_2,y_3} & \mathcal{I}_{min} & \xrightarrow{u_3,y_4} & \mathcal{I}_{min} & \xrightarrow{u_4,y_5} & \mathcal{I}_{min} \cdots \\
\downarrow{\kappa''} & & \downarrow{\kappa''} & & \downarrow{\kappa''} & & \downarrow{\kappa''} & & \downarrow{\kappa''} \\
\mathcal{I}_{task} & & \mathcal{I}_{task} & & \mathcal{I}_{task} & & \mathcal{I}_{task} & & \mathcal{I}_{task}.
\end{array}
\tag{4}
$$

Note that we can similarly define an I-map that maps any derived I-space to another. An example is given in (4) as the mappings $\kappa' : \mathcal{I}_{der} \rightarrow \mathcal{I}_{min}$ and $\kappa'' : \mathcal{I}_{min} \rightarrow \mathcal{I}_{task}$. The corresponding quotient system is deterministic for κ', indicating that it is sufficient. However, the quotient system by κ'' derived from $\mathcal{I}_{min}$ is not deterministic, hence, κ'' is not sufficient, as the next I-state can not be uniquely determined. Note that an I-map whose domain is $\mathcal{I}_{hist}$ can also be defined as composition of the mappings along the column of the diagram. For instance, $\kappa_{min} : \mathcal{I}_{hist} \rightarrow \mathcal{I}_{min}$ is the composition of κ and κ', that is, $\kappa_{min} = \kappa' \circ \kappa$ (same for $\kappa_{task} : \mathcal{I}_{hist} \rightarrow \mathcal{I}_{task}$).

---

[2] A transition system (S, Λ, T) is full, if $\forall s \in S, \lambda \in \Lambda$ there exists at least one $s' \in S$ with $(s, \lambda, s') \in T$.

3.5   Lattice of information transition systems

We fix $\mathcal{I}_{hist}$, which corresponds to fixing the set of initial states $\mathcal{I}_0$. Then, each I-map κ defined over $\mathcal{I}_{hist}$ induces a partition of $\mathcal{I}_{hist}$ through its preimages, denoted as $\mathcal{I}_{hist}/\kappa$. An I-map κ' is a refinement of κ, denoted as $\kappa' \succeq \kappa$, if $\forall A \in \mathcal{I}_{hist}/\kappa'$ there exists a $B \in \mathcal{I}_{hist}/\kappa$ such that $A \subseteq B$. Let $K(\mathcal{I}_{hist})$ denote the set of all partitions over $\mathcal{I}_{hist}$. Refinement induces a partial ordering since not all partitions of $\mathcal{I}_{hist}$ are comparable. The partial ordering given by refinements form a lattice of partitions over $\mathcal{I}_{hist}$, denoted as $(K(\mathcal{I}_{hist}), \succeq)$.

At the top of the lattice, there is the partition induced by an identity I-map (or equivalently, by a bijection), $\kappa_{id} : \mathcal{I}_{hist} \to \mathcal{I}_{hist}$, since all of its elements are singletons (all equivalence classes contain exactly one element), making it the maximally distinguishable case. Conversely, we can define a constant mapping $\kappa_{const} : \mathcal{I}_{hist} \to \mathcal{I}_{const}$ for which $\mathcal{I}_{hist}/\kappa_{const}$ is a singleton, that is, $\mathcal{I}_{const} = \{\iota_{const}\}$, which then will be at the bottom of the lattice. In turn, κ_{const} yields the minimally distinguishable case as all histories now belong to a single equivalence class. This idea is similar to the notion of the *sensor lattice* defined over the partitions of X [9,13]. Indeed, if we take $\mathcal{I}_0 = X$ and consider $\kappa_{est} : \mathcal{I}_{hist} \to X$, the ordering of partitions of $\mathcal{I}_{hist}$ such that $\mathcal{I}_{hist}/\kappa_{est}$ is the least upper bound gives out the sensor lattice.

As motivated in previous sections, we are interested in finding a sufficient I-map such that the quotient ITS derived from the history ITS is still deterministic. Notice that the constant I-map κ_{const} is sufficient by definition since for all $(u, y) \in U \times Y$, and all $\eta, \eta' \in \mathcal{I}_{hist}$, we have that $\kappa_{const}(\eta) = \kappa_{const}(\eta')$ and $\kappa_{const}(\phi_{hist}(\eta, (u, y))) = \kappa_{const}(\phi_{hist}(\eta', (u, y)))$. On the other hand, in certain cases it is crucial to differentiate certain histories from the others. This will become clear in the next section when we describe the notion of a task. Suppose κ is a labeling that partitions $\mathcal{I}_{hist}$ into equivalence classes that are of importance and suppose that κ is not sufficient. Then, we want to find a refinement of κ that is sufficient. This will serve as a lower bound on the lattice of partitions over $\mathcal{I}_{hist}$ since for any partition such that $\mathcal{I}_{hist}/\kappa$ is a refinement of it, the classes of histories that are deemed crucial will not be distinguished. The following defines the refinement of κ that ensures sufficiency and a minimal number of equivalence classes.

Definition 2. *Let $(\mathcal{I}_{hist}, U \times Y, \phi_{hist})$ be a history ITS and κ an I-map. A minimal sufficient refinement of κ is a sufficient I-map κ' such that there does not exist a sufficient I-map κ'' that satisfy $\kappa' \succ \kappa'' \succeq \kappa$.*

Remark 1. It is shown in [12] that the minimal sufficient refinement of κ defined over the states of an automaton (S, Λ, τ) is unique.

4   Solving Tasks Minimally

Definition of a task We now connect the general ITS concepts to the accomplishment of particular tasks. We have two categories: 1) *active*, which corresponds to planning and executing an information-feedback policy that forces a

desirable outcome in the environment, and 2) *passive*, which means only to observe the environment without being able to effect changes. Recall from Section 3.4 that there may be model-free or model-based formulations. In the model-free case, tasks are specified using a logical language over $\mathcal{I}_{hist}$ which will result in a labeling and derived I-space $\mathcal{I}_{task}$ and associated I-map κ_{task} that corresponds to the "resolution" at which the tasks are specified. Various logics are allowable, such as propositional or a temporal logic. The resulting sentences of the language involve combinations of predicates that may assign true or false values to subsets of $\mathcal{I}_{hist}$. Solving an active task (or tasks) requires that a sentence of interest becomes true during execution of the policy. This is called *satisfiability*. For example, the task may be to simply reach some goal set $G \subset \mathcal{I}_{hist}$, causing a predicate IN-GOAL($\mathcal{I}_{hist}$) to become satisfied (in other words, be true). Using linear temporal logic, more complex requirements, such as cycling through a finite sequence of subsets forever while avoiding others, can be specified [3].

Solving a passive task only requires maintaining whether a sentence is satisfied, rather than forcing an outcome; this corresponds to filtering. Whether the task is active or passive, if satisfiability is concerned with a single, fixed sentence, then a *task-induced labeling* (or *task labeling* for short), that is, κ_{task}, over $\mathcal{I}_{hist}$ assigns two labels: Those I-states that result in true and those that result in false. A task labeling may also be assigned for a set of possible sentences by assigning a label to each set of the common refinement of the partition of $\mathcal{I}_{hist}$ induced by each possible sentence. In the model-based case, tasks are instead specified using a language over X, and sentence satisfiability must be determined by an I-map that converts history I-states into expressions over X.

Problem families It is assumed that the state-relabeled transition system (X, U, f, h, Y) describing the external system is fixed, but it is unknown or partially known to the observer (a robot or other observer).

Filtering (passive case) requires maintaining the label of an I-state attributed by κ_{task}. Since κ_{task} is not necessarily sufficient, we can not guarantee that the quotient system by κ_{task} is a DITS (Propositions 1 and 2). Thus, relying solely on the quotient system by κ_{task}, we can not determine the class that the current history belongs to (see the last row in (4)) and, hence can not determine whether a sentence describing the task is satisfied (or which sentences are satisfied).

Suppose the sets U and Y are specified, and at each stage k, u_{k-1} is known and y_k is observed. The following describes the problem for a passive task given a state-relabeled (history) ITS $(\mathcal{I}_{hist}, U \times Y, \phi_{hist}, \kappa_{task}, \mathcal{I}_{task})$, in which $\kappa_{task} : \mathcal{I}_{hist} \to \mathcal{I}_{task}$ is a task labeling that is not sufficient, and $\mathcal{I}_{task}$ is the corresponding I-space.

Problem 1 (Find a sufficient I-space filter). Find a sufficient refinement of κ_{task}.

Note that $\mathcal{I}_{hist}/\kappa_{task}$ determines a lower bound on the partitioning of $\mathcal{I}_{hist}$ which is interpreted as the crucial information that can not be lost. Consequently, histories belonging to different equivalence classes with respect to κ_{task} must always be distinguished from each other. However, Problem 1 does not an

upper bound. At the limit, a bijection from $\mathcal{I}_{hist}$ is always a sufficient refinement of κ_{task}. As stated previously, using history ITS can create computational obstructions in solving problems. This motivates the following problem.

Problem 2 (Filter minimization). Find a minimal sufficient refinement of κ_{task}.

We now consider a basic planning problem, for which $\mathcal{I}_{task} = \{0, 1\}$, such that $\kappa_{task}^{-1}(1) \subset \mathcal{I}_{hist}$ is the set of histories that achieve the goal, and $\kappa_{task}^{-1}(0) \subset \mathcal{I}_{hist}$ is its complement. Most planning problems refer to finding a labeling function π such that, when used to label the states of the internal system, guarantees task accomplishment. Then π is called a *feasible* policy, which is defined in the following. Consider an external system (X, f, U, h, Y). Let $\mathcal{R}_X(\mathcal{I}_{task}) \subseteq X$ be the set of initial states for which there exist a k and histories $\tilde{x}_k$, $\tilde{u}_{k-1}$, and $\tilde{y}_k$, such that $x_{i+1} = f(x_i, u_i)$ and $y_i = h(x_i)$ for all $0 < i < k$, and $\eta_k \in \kappa_{task}^{-1}(1)$, in which η_k is the history I-state corresponding to $\tilde{u}_{k-1}$ and $\tilde{y}_k$.

Definition 3 (Feasible policy for $\mathcal{I}_{task}$). *Let $(\mathcal{I}, Y, \phi, \pi, U)$ and (X, f, U, h, Y) be the state-relabeled transition systems corresponding to internal and external systems, respectively. Labeling function π defined over $\mathcal{I}$ is a feasible policy for $\mathcal{I}_{task}$ if for all $x \in \mathcal{R}_X(\mathcal{I}_{task})$, the history corresponding to the coupled internal-external system initialized at (ι_0, x) belongs to $\kappa_{task}^{-1}(1)$.*

Most problems in the planning literature consider a fixed DITS and look for a feasible policy for $\mathcal{I}_{task}$. This yields the following problem. Typically, the I-space considered is X which corresponds to state estimation. Note that a DITS, in other words, the robot brain, can be seen as an I-space filter itself.

Problem 3 (Find a feasible policy). Given $(\mathcal{I}, Y, \phi)$, an internal system (robot brain), find a labeling function $\pi : \mathcal{I} \to U$ that is a feasible policy for $\mathcal{I}_{task}$.

We can further extend the planning problem to consider an unspecified internal system which refers to finding a DITS, and a policy such that the resulting histories satisfy the task description, that is, he problem of jointly finding an I-space-filter and a policy defined over its states. Recall from (2) that a policy constrains the transitions of an ITS to the ones that are realizable under that policy.

Problem 4 (Find a DITS and a feasible policy). Given $(\mathcal{I}_{hist}, U \times Y, \phi_{hist})$, for which $\kappa_{task} : \mathcal{I}_{hist} \to \mathcal{I}_{task}$ is a task labeling and $\mathcal{I}_{task}$ is the corresponding I-space, find a sufficient I-map $\kappa : \mathcal{I}_{hist} \to \mathcal{I}$ and a feasible policy π for $\mathcal{I}_{task}$ as a labeling function for the resulting quotient system by κ.

Note that κ_{task} can already be sufficient, so that it is the minimal sufficient refinement of itself, however, this does not necessarily imply the existence of a feasible policy defined over $\mathcal{I}_{task}$. Therefore, we are not looking for a refinement of κ_{task} while describing the DITS over which the policy is defined. On the other hand, we can still talk about a notion of minimality. Let $(\mathcal{I}, Y, \phi, \pi, U)$ be a state-relabeled DITS that solves Problem 4 (or similarly Problem 3) which is the quotient system of history ITS by κ, then, $(\mathcal{I}, Y, \phi, \pi, U)$ is *minimal for π* if there does not exist a sufficient I-map κ' with $\kappa \succ \kappa'$ for which there exists a π' for the quotient system by κ' that satisfies $\pi(\iota) = \pi'(\kappa'(\iota))$.
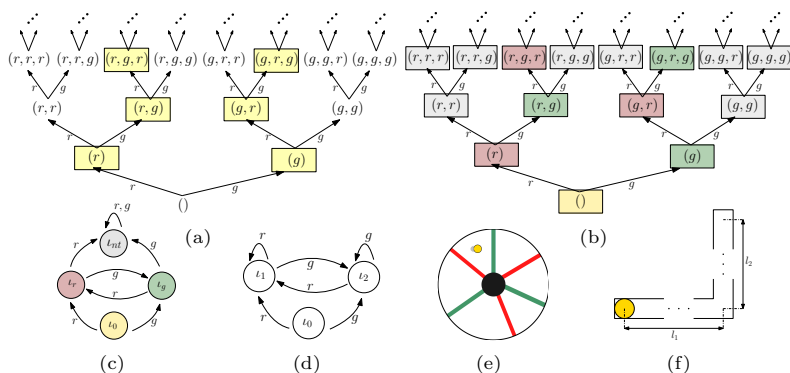
Fig. 2. (a) State-relabeled history ITS described in Example 1, and labeling function κ_{task} yellow colored states correspond to states that satisfy the task description. (b) Equivalence classes induced by κ'; the minimal sufficient refinement of κ_{task}. (c) Quotient of the history ITS by κ'. (d)DITS describing the internal system solving the planning problem described in Example 2. (e) Environment used in Examples 1,2, the obstacle (an open disk) is shown in black. (f) L-shaped corridor; $l_1, l_2 \leq l$.

Learning a sufficient ITS Although learning and planning overlap significantly, some unique issues arise in pure learning (see also [12]). This corresponds to the case when $\mathcal{I}_{task}$ is not initially given but needs to be revealed through interactions with the external system, that is, respective action and observation histories. It is assumed that whether the sentence (or sentences) describing the task is satisfied or not can be assessed at a particular history I-state.

We can address both filtering and planning problems defined previously within this context, considering model-free and model-based cases. In the model-free case, the task is to compute a minimal sufficient ITS that is consistent with the actions and observations. Variations include *lifelong learning*, in which there is a single, 'long' history I-state, or more standard learning in which the system can be restarted, resulting in multiple *trials*, each yielding a different history I-state. In the model-based case, partial specifications of X, f, and h may be given, and unknown parameters are estimated using the history I-state(s). Different results are generally obtained depending on what assumptions are allowed. For example, do identical history I-states imply identical state trajectories? If not, then set-based, nondeterministic models may be assumed, or even probabilistic models based on behavior observed over many trials and assumptions on probability measure, priors, statistical independence, and so on.

5 Illustrative Examples

In this section we provide some simple examples to show how the ideas presented in this paper apply to filtering and planning problems. All problems can be posed as well in a machine learning context for which $\mathcal{I}_{task}$ is not given but it is revealed through interactions between the internal and external as the input-output data.

Let $E \subseteq \mathbb{R}^2$ be a bounded planar environment (see Figure 2(e)) that is partitioned into regions separated by gates Each gate is either green or red whose color can be detected by the robot's color sensor and follows the rule that each region shares a boundary with exactly two gates; one green and one red. The set of possible observations are $Y = \{r, g\}$.

Example 1. This example considers a filtering problem from the perspective of an independent observer. Suppose the actions taken by the robot are not observable and the only information about the system is the history of readings coming from the robot's color sensor; for example, (r, r, r, g, r, g). Then, the history I-space is the set of all finite length sequences of elements of Y, that is, $\mathcal{I}_{hist} = Y^*$, which refers to the free monoid generated by the elements of Y (or the Kleene star of Y). Hence, the history ITS can be represented as an infinite binary tree. The task is to determine whether the robot crosses the gates consistently (in a clockwise or counterclockwise manner) or not. Hence, the preimages of $\kappa_{task} : \mathcal{I}_{hist} \to \mathcal{I}_{task}$ partition $\mathcal{I}_{hist}$ into two subsets: one which the condition is satisfied and the others. The labeling induced by κ_{task} is shown in Figure 2(a). Clearly, κ_{task} is not sufficient since there exist I-states η, η' such that $\kappa_{task}(\eta) = \kappa_{task}(\eta')$ and there exists a y for which $\kappa_{task}(\phi_{hist}(\eta, y)) \neq \kappa_{task}(\phi_{hist}(\eta', y))$; for example consider $\eta = (r, g)$, $\eta' = (r, g, r)$ and $y = g$. A sufficient refinement of κ_{task} can be obtained (equivalence classes shown in 2(b)), denote as κ', for which the quotient DITS is shown in Figure 2(d). Furthermore, κ' is a minimal sufficient refinement of κ_{task} since it follows from Proposition 2 that if a labeling is not minimal then there is a minimal one that is strictly coarser and is still sufficient. However, neither of the subsets that belong to $\mathcal{I}_{hist}/\kappa'$ can be merged, since merging ι_{nt} (colored gray) with anything else violates the condition that κ' is a refinement of κ_{task} and any pairwise merge of the others violate sufficiency.

Suppose the robot has a boundary detector, and executes a bouncing motion using the two motion primitives $U = \{u_1, u_2\}$, in which u_1 is move forward and u_2 is rotate in place. Let a basic motion be move forward and bounce off the walls, which can be implemented using the elements of U. We assume that the boundary detector and color sensor readings do not arrive simultaneously.

Example 2. We now consider a planning problem (that belongs to the class described in Problem 4) for which the goal is to ensure that the robot crosses the gates consistently. The history I-space of the planner is $\mathcal{I}_{hist} = (U \times Y)^*$ and the preimages of κ_{task} partition $\mathcal{I}_{hist}$ into two sets; the histories that satisfy the predicate and the ones that do not. Then, a DITS with only three states (see Figure 2(d)) can be derived using the mapping $\kappa : \mathcal{I}_{hist} \to \mathcal{I}$, in which $\mathcal{I} = \{i_0, i_1, i_2\}$ such that for $\pi(\iota_1)$ boundary with the red gate is set as a wall, $\pi(\iota_2)$ the boundary with the green gate is set as a wall, and for $\pi(\iota_0)$ no boundary with gates are considered as a wall. We assume that a bouncing motion can be determined using the motion primitives so that the resulting trajectory will strike every open interval in the boundary of every region infinitely often, with non-zero, non-tangential velocities [1].

Consider a robot in an L-shaped planar corridor (Figure 2(f)). Let $\mathcal{E}_l$ be the set of all such environments such that $l_1, l_2 \leq l$, in which l_1 and l_2 are the dimensions of the corridor bounded by l. We assume that the minimum length/width is larger than the robot radius, that is, 1. The state space X is defined as the set of all pairs (q, E_i), in which $(q_1, q_2) \in E_i$, and $E_i \in \mathcal{E}_l$. The action set is $U = \{0, 1\} \times \{0, 1\}$ which corresponds to moving one step in one of the 4 cardinal directions; if the boundary is reached, the state does not change. The robot has a sensor that reports 1 if the motion is blocked.

Example 3. Consider a model-based history ITS with $\eta_0 \subset X$ that specifies the initial position as $q_0 = (0, 0)$ but does not specify the environment so that it can be any $E_i \in \mathcal{E}_l$. Let $\mathcal{I}_{hist}$ be its set of states and $\kappa_{ndet} : \mathcal{I}_{hist} \to \text{pow}(X)$ be an I-map that maps η_k to a subset of $X_k \subseteq X$. Since (X, U, f, h, Y), and $X_0 = \eta_0$ are known, transitions for the quotient system can be described by induction as $X_{k+1} = \hat{X}_{k+1}(X_k, u_k) \cap H(y_{k+1})$, in which $\hat{X}_{k+1} = \bigcup_{x_k \in X_k} f(x_k, u_k)$ and $H(y_{k+1}) \subseteq X$ is the set of all states that could yield y_{k+1}. By construction, κ_{ndet} is sufficient. Suppose $\kappa_{task} : \mathcal{I}_{hist}/\kappa_{ndet} \to \mathcal{I}_{task}$ is a task labeling for localization that assigns each singleton a unique label and all the other subsets are labeled the same. Since the transition corresponding to $([X']_{\kappa_{task}}, (u, y))$ in which $X' \subseteq X$ is not a singleton can lead to multiple labels $[x']_{\kappa_{task}}$, in which x' is a singleton, κ_{task} is not sufficient. Furthermore, κ_{ndet} is a minimal sufficient refinement of κ_{task} because it is sufficient and because there does not exist a sufficient κ such that $\kappa_{ndet} \succ \kappa \succeq \kappa_{task}$. Suppose κ exists, that would mean some equivalence classes can be merged. However, this is not possible because merging any of the non-singleton subsets violates sufficiency (as shown for κ_{task}) and merging singletons with others violates that it is a refinement. A policy can be described over $\mathcal{I}_{hist}/\kappa_{ndet}$; $u = (1, 0)$ starting from X_0 until $y_k = 1$ is obtained and applying $u = (0, 1)$ starting from X_k until $y_n = 1$ is obtained, then it is found that $q = (k, n)$ and E is the corridor with $l_1 = k$, $l_2 = n$.

6   Discussion

We have introduced a mathematical framework for determining minimal feasible policies for robot systems, by comparing ITSs over I-spaces. The uniqueness and minimality results are quite general: X and U could be discrete, typical configuration spaces, or more exotic, such as the power set of all functions from an infinite-dimensional Hilbert space into an infinite-dimensional Banach space.

Nevertheless, there are opportunities to expand the general theory. For example, we assumed that u is both the output of a policy and the actuation stimulus in the physical world; more generally, we should introduce a mapping from an action symbol $\sigma \in \Sigma$ to a control function $\tilde{u} \in \tilde{U}$ so that plans are expressed as $\pi : \mathcal{I} \to \Sigma$ and each $\sigma = \pi(\iota)$ produces energy in the physical world via a mapping from Σ to $\tilde{U}$. Another extension is to consider stochastic models, that amounts to an ITS with a probabilistic I-space, and in which ways it ties to the representations used in the literature such as *predictive state*

representations (PSRs) [10]. Other potential extensions include continuous-time "transitions" and novel logics to consider satisfiability questions over spaces of robot systems and tasks. An interesting direction is to consider the hardware and actuation models are variables, and fix other model components.

A grand challenge remains: The results here are only a first step toward producing a more complete and unique theory of robotics that clearly characterizes the relationships between common tasks, robot systems, environments, and algorithms that perform filtering, planning, or learning. We should search for lattice structures that play a role similar to that of language class hierarchies in the theory of computation. This includes the structures of the current paper and the sensor lattices of [9,13]. Many existing filtering, planning, and learning methods can be formally characterized within this framework, which would provide insights into relative complexity, completeness, minimality, and time/space/energy tradeoffs.

References

1. L. Bobadilla, O. Sanchez, J. Czarnowski, K. Gossman, and S. M. LaValle. Controlling wild bodies using linear temporal logic. In *Proceedings Robotics: Science and Systems*, 2011.
2. Axel Brunnbauer, Luigi Berducci, Andreas Brandstätter, Mathias Lechner, Ramin M. Hasani, Daniela Rus, and Radu Grosu. Latent imagination facilitates zero-shot transfer in autonomous racing. https://arxiv.org/abs/2103.04909v2.
3. G. E. Fainekos, A. Girard, H. Kress-Gazit, and G. J. Pappas. Temporal logic motion planning for dynamic mobile robots. *Automatica*, 45(2):343–352, 2009.
4. K. Friston. The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, 11(2):127–138, 2010.
5. D. D. Hutto and E. Myin. *Radicalizing enactivism: Basic minds without content.* MIT Press, 2012.
6. D. E. Koditschek. What is robotics? Why do we need it and how can we get it? *Annual Review of Control, Robotics, and Autonomous Systems*, 4:1–33, May 2021.
7. Ü. Kotta, C. H. Moog, and M. Tõnso. Minimal realizations of nonlinear systems. *Automatica*, 95:207–212, 2018.
8. S. M. LaValle. *Planning Algorithms.* Cambridge University Press, Cambridge, U.K., 2006. Also available at http://lavalle.pl/planning/.
9. S. M. LaValle. *Sensing and Filtering: A Fresh Perspective Based on Preimages and Information Spaces*, volume 1:4 of *Foundations and Trends in Robotics Series.* Now Publishers, Delft, The Netherlands, 2012.
10. Michael Littman and Richard S Sutton. Predictive representations of state. *Advances in neural information processing systems*, 14, 2001.
11. J. M. O'Kane and D. A. Shell. Concise planning and filtering: Hardness and algorithms. *IEEE Transactions on Automation Science and Engineering*, 14(4):1666–1681, 2017.
12. V. Weinstein, B. Sakcak, and S. M. LaValle. An enactivist-inspired mathematical model of cognition. under review, 2022.
13. Y. Zhang and D. A. Shell. Lattices of sensors reconsidered when less information is preferred. In *IEEE International Conference on Robotics and Automation*, 2021.