Journal Title
XX(X):1–20
©The Author(s)
Reprints and permission:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

A Mathematical Characterization of Minimally Sufficient Robot Brains

Basak Sakcak, Vadim Weinstein, Kalle G. Timperi, and Steven M. LaValle[1]

Abstract

This paper addresses the lower limits of encoding and processing the information acquired through interactions between an internal system (robot algorithms or software) and an external system (robot body and its environment) in terms of action and observation histories. Both are modeled as transition systems. We want to know the weakest internal system that is sufficient for achieving passive (filtering) and active (planning) tasks. We introduce the notion of an *information transition system* (ITS) for the internal system which is 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. An ITS is viewed as a filter and a policy or plan is viewed as a function that labels the states of this ITS. Regardless of whether internal systems are obtained by learning algorithms, planning algorithms, or human insight, we want to know the limits of feasibility for given robot hardware and tasks. 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 modeling a system given input-output relations.

Keywords

Planning, transition systems, sensing uncertainty, sensor fusion, filtering, information spaces, machine learning, control theory, theoretical foundations

1 Introduction

Accomplishing a robot's tasks may involve designing or employing a combination of different parts: planning algorithms, sensor fusion or filtering methods, machine learning algorithms, and control laws. Given a problem expressed in terms of a well-defined task structure, the relationship between these different parts with each other, and their relation with the task is often ignored. Each part is developed rather in isolation, heavily motivated by the long lasting traditions in robotics. For example, navigating a mobile robot to a goal configuration is typically achieved by estimating the robot configuration in a known (or an unknown) map and applying a feedback policy that relies on the estimated configuration. Indeed, for some problems it may be possible that a simpler filtering approach that does not require estimating the full state could as well be sufficient to achieve the given task. This may lead many to believe that robotics itself does not have its own, unique theoretical core (on this, we agree with (Koditschek 2021)) and it appears as an application area for other fields; designing and testing machine learning algorithms, planning algorithms, sensor fusion methods, control laws, and so on. In our quest towards a theory that is unique to robotics and that plays a similar role to Turing machines for computer science, or $\dot{x} = f(x, u)$ over differentiable manifolds for control theory, we want to establish the limits of the intertwined notions of sensing, learning, filtering, and planning with respect to a given problem. We would like to have a general framework that allows researchers to formulate and potentially answer questions such as: Does a solution even exist to a given problem? What are the minimal necessary components to solve it? What should the best learning approach imaginable produce as a representation? Such questions would be analogous to existence and uniqueness in control and dynamical systems, or decidability and complexity (especially Kolmogorov) in theoretical computer science.

This paper proposes a mathematical robotics theory that is built from the input-output relationships between two (or more) coupled dynamical systems. For a programmable mechanical system (robot) embedded in an environment, the input-output relationships correspond to sensing and actuation between two coupled systems; an *internal system* (robot brain) and an *external system* (robot body and the environment). This relation is shown in Figure 1(a-b). We assume that the robot hardware is fixed, which means fixing the sensors and the actuators, and we focus on determining which necessary and sufficient conditions the internal system has to maintain for a task to be accomplished. In light of these conditions, we try to find a *minimal sufficient* internal system which corresponds to the weakest possible representation of

[01] Center for Ubiquitous Computing Faculty of Information Technology and Electrical Engineering, University of Oulu, Finland. (e-mail: firstname.lastname@oulu.fi)
[0]
Corresponding author:
Basak Sakcak, Center for Ubiquitous Computing Faculty of Information Technology and Electrical Engineering, University of Oulu, Finland.
[0] Email: basak.sakcak@oulu.fi

the acquired information through interactions; reducing the internal system any further makes the problem unsolvable.

At the core of our framework is the notion of an ITS which builds on the well-studied notion of transition systems. The *information* part of an ITS comes from *information spaces* (LaValle 2006, Chapter 11) which is developed 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. An internal system will be modeled as an ITS and its sufficiency and minimality with respect to a problem will be analyzed using this framework.

We categorize tasks into two classes: active and passive. Informally, a passive task corresponds to filtering and an active one corresponds to planning or control. In our work an ITS can be seen as a filter and together with its underlying information space serves as a domain over which a plan or a policy can be expressed and analyzed. We will consider a variety of information spaces, which also encompass robot configuration spaces or phase spaces, that are typically used for planning tasks. A distinction between *model-based* and *model-free* formulations will be considered too, in line with the choices commonly found in machine learning. We characterize the problems corresponding to the active and passive tasks and define notions of feasibility and minimality for *information transition systems* (ITSs) that solve these problems. In our approach to finding minimal sufficient ITSs, we will analyze the limits of reducing or collapsing the information spaces, until the lower limits of task feasibility are reached.

1.1 Previous work

Many of the concepts in this paper build upon (Weinstein et al. 2022), in which we recently proposed an enactivist-oriented model of cognition based on information spaces. By *enactivist* (Hutto and Myin 2012), 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 paradigm (Newell and Simon 1972). Despite introducing a general framework, the focus of (Weinstein et al. 2022) was on emerging equivalence classes of the environment states through sensorimotor interactions. In some sense, this corresponds to finding a (minimal) sufficient sensor or a representation for the external system. In this paper, we focus on the internal system which, in turn, refers to finding a minimal filter and/or a policy, given an appropriate task description.

Most approaches in filtering can be categorized into two classes: probabilistic and combinatorial. Probabilistic filters typically rely on Bayes' rule to propagate the obtained information (Thrun et al. 2005; Särkkä 2013). They have been extensively used in robotics; especially for state estimation, mapping, and localization (Dissanayake et al. 2001; Zhen et al. 2017). Combinatorial filters (Tovar et al. 2008; Kristek and Shell 2012) do not rely on probabilistic models but instead make use of nondeterministic (possibilistic) ones. The notion of information spaces (LaValle 2006, Chapter 11) provides a general formalization that encompasses both probabilistic and combinatorial filters. In our framework based on transition systems, the set of states of a transition system will indeed be an information space. Using the notion of information spaces, several works attempted to characterize different sensors defined over the same state space and compare their power in terms of gathered information (O'Kane and LaValle 2008; LaValle 2012; Zhang and Shell 2021). Despite providing an elegant and exact way of solving filtering problems, the space requirements for a combinatorial filter can be high. Considering a notion of minimality, some authors addressed algorithmic reduction of filters (Rahmani and O'Kane 2021; O'Kane and Shell 2017; Song and O'Kane 2012). Reducing combinatorial filters have roots in the theory of computation, where decomposition (Hartmanis 1960; Hartmanis and Stearns 1964) and minimization of finite automata (Moore and Mealy machines) has been a topic of active research since the 1950s.

A gap still remains between analyzing the requirements of filters or sensors for pure inference (passive filtering) and the ones needed for active tasks (planning and control) such that an *information-feedback* policy can be described. Various representations were used in the literature as a domain to define the policy. For most robotic planning problems, the domain of the policy or a plan is fixed; which is the robot configuration or the phase space (Majumdar and Tedrake 2017; Zhu and Alonso-Mora 2019). This corresponds to the assumption that the robot state can be fully observed or estimated with high accuracy. For problems that the state is not fully observable, *partially observable Markov decision processes* (POMDPs) (Kaelbling et al. 1998; Ross et al. 2008) and *belief spaces* (Vitus and Tomlin 2011; Agha-Mohammadi et al. 2014) have been considered for planning. Note that POMDP literature is mostly restricted to finite state and action spaces. There is a limited literature that studied the information requirements for active tasks which corresponds to determining the weakest notion of sensing or filtering that is sufficient to accomplish a task. A notable early work showed, especially for manipulation, that one can achieve certain tasks even in the absence of sensory observations (Erdmann and Mason 1988). In (Zhang and Shell 2020) the authors characterize all possible sensor abstractions that are sufficient to solve a planning problem. Minimality has been addressed for specific problems regarding mobile robot navigation in (Blum and Kozen 1978; Tovar et al. 2007). Closely related to our work, a language-theoretic formulation appears in (Saberifar et al. 2019), in which, Procrustean-graphs (p-graphs) were proposed as an abstraction to reason about interactions between a robot and its environment.

Obtaining a model from input-output relations that represents the underlying system has been a common interest for many fields ranging from control theory, machine learning, and robotics. Different approaches to this problem in the context of finite state automata were reviewed by (Pitt 1989). In diversity-based inference (DBI) for an input-output machine (Bainbridge 1977; Rivest and Schapire 1993, 1994), a model of the underlying system is constructed in terms of equivalence classes of *tests* which consist of one or more consecutive actions and observations. Its probabilistic counterpart, predictive state representation (PSR) (Littman and Sutton 2001; Boots et al. 2011), addresses the analogous

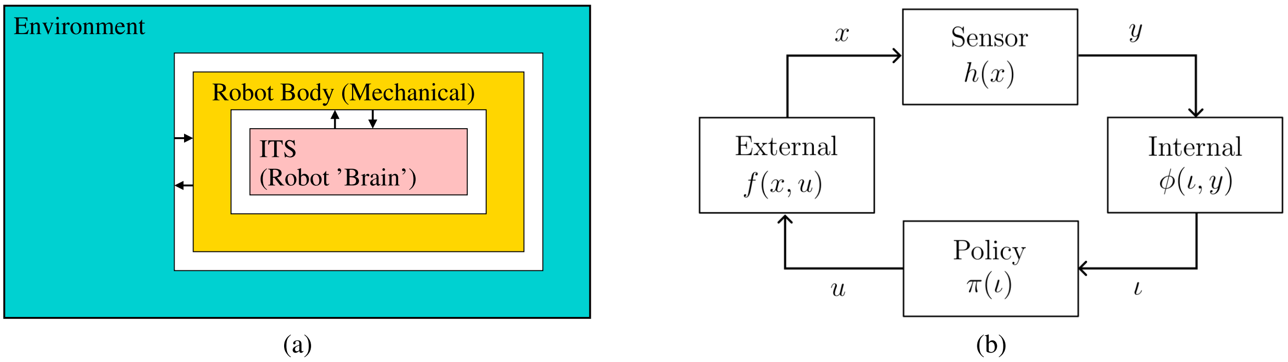


Figure 1. (a) The *internal* robot brain is defined as an ITS that interacts with the *external* world (robot body and environment). (b) Coupled internal and external systems mathematically capture sensing, actuation, internal computation, and the external world.

problem by considering (linear) combinations of prediction vectors, which represent probabilities for test success/failure. Other than relying solely on input-output relations, a parametric model (or a class of them) can be provided for learning a representation for the underlying system (Brunnbauer et al. 2022). We will also consider this distinction between model-based and model-free within our ITS framework.

1.2 Contributions

The main contribution of this paper is a novel mathematical framework for analyzing and distinguishing the interactions that emerge from a robotic system embedded in an environment. We introduce the notion of ITSs as a general way to characterize the internal system ("brain") of the robot. We then proceed to establish conditions for sufficiency and the existence of unique minimal ITSs in a very general setting. Intuitively, we establish how small the robot brain could possibly be for given goals or tasks. Anything less results in impossibility. The framework addresses both filtering and sensor fusion problems, which are passive in the sense of no controls are applied, and planning or control problems, which are active. We illustrate the scope of the framework by applying it to several problems that shed light on relationships to many existing concepts, including Kalman filters, predictive state representations, combinatorial filters, and planning over reduced information spaces.

Many of the concepts in this paper build upon (Weinstein et al. 2022) and (LaValle 2006, Chapter 11). The contributions of the present paper with respect to these works are listed in the following:

- The framework based on transition systems introduced in (Weinstein et al. 2022) is adapted into a robotic setting. We define the notions of internal and external systems within this context, together with concrete examples. Moreover, we formulate the disturbances affecting the external system model and the sensor mapping, within this framework.
- We formally define the notion of task description (distinguishing between finite and infinitary tasks, as well as between active and passive tasks), and filtering and planning problems.
- The notion of minimality for a transition system describing the internal system under a planning (control) task is new.
- The ITSs corresponding to some of the information spaces presented in (LaValle 2006, Chapter 11), which were based on intuitions, are shown to be minimal applying the proposed framework.
- We also formalize the model-based and the model-free information spaces using the notion of coupled internal-external systems.

This paper is an expanded version of (Sakcak et al. 2022).

1.3 Paper structure

The remaining of the paper is organized as follows. Section 2 provides a general mathematical formulation of robot-environment interaction as transition systems. We also introduce a notion of couplings which encode various types of interactions between an internal and an external system. Section 3 then develops central 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. Canonical problem families are presented that aim to capture typical problem settings in filtering and planning, from the perspective of minimal sufficient solutions. Section 5 illustrates how the theory can be applied using simple examples. Section 6 summarizes the contributions and identifies important directions for future work.

2 Mathematical Models of Robot-Environment Systems

2.1 Internal and external systems

In this paper, we consider a robot embedded in an environment and describe this system as two subsystems, named *internal* and *external*, connected through symmetric input-output relations. The external system describes the physical world, and the internal system describes the information processing "robot brain". With "robot brain" we refer to a centralized computational component that processes sensor observations and actions. The interaction between the internal and external systems is shown in Figure 1(b). The input to the internal system is the information reflecting the state of the external system, obtained through observations (that is the output of the external system). The output of the internal system is a control command that in turn corresponds to the control input of the external system, and causes its state to change. In this sense, the state of the external system is similar to the use of the term *state* in control theory and the state of the internal system is similar to the use of the term in computer science.

The external system corresponds to the totality of the physical environment, including the robot body. Let X denote the set of states of this system; it could be for example, the configuration of a robot (e.g., position and orientation of a mobile robot or joint configuration of a robotic manipulator) within a known environment (or within a set of possible environments), or it can be extended to include also the (higher order) derivatives of its configuration. See (LaValle 2012, Section 3.1) for possible state spaces of a mobile robot. Next, let U be the set of control inputs (also referred to as actions). When applied at a state $x \in X$, a control $u \in U$ causes x to change according to a state transition function $f\colon X \times U \to X$. Indeed, an action $u \in U$ refers to the control input to the system and corresponds to the stimuli created by a control command generated by a decision maker. Mathematically, *the external system* can now be expressed as the triple (X, U, f). The sets X and U can be finite or infinite discrete spaces, or they could be equipped with extra structure: they could be metric manifolds, vector spaces, compact, or non-compact topological spaces. In such cases, the function f may or may not be assumed to respect such structure: sometimes it is appropriate to assume continuity or measurability.

The internal system (robot brain) corresponds to the perspective of a decision maker. 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 system is the notion of an information space *(I-space)* presented in (LaValle 2006, Chapter 11). Let $\mathcal{I}$ be the set of these internal information states. We will use the term information state *(I-state)* to refer to elements of $\mathcal{I}$ and denote them by ι. Similar to the external system, the internal system evolves with each $y \in Y$ according to the *information transition function* $\phi\colon \mathcal{I} \times Y \to \mathcal{I}$. The *internal system* can now mathematically be described by the triple $(\mathcal{I}, Y, \phi)$.

The external (X, U, f) and internal $(\mathcal{I}, Y, \phi)$ can be *coupled* to each other to create a *coupled dynamical system*. This is achieved by introducing two coupling functions, that match outputs of one system to the inputs of the other and vice versa. For us, these are the *sensor mapping* $h\colon X \to Y$ and the *policy* $\pi\colon \mathcal{I} \to U$, see Figure 1. The function h labels the external states with sensory data, and π labels the internal information states with the actions. Note that π can be seen as an information feedback policy sending a control to the external. In the following parts of this paper, we will refer to π simply as the policy. Therefore, it will be a map from the states of an I-space (Sections 3.2 and 3.4 will present possible I-space descriptions) to the set of controls. This definition is more general than the use of policy in the robotics literature which typically refers to a state-feedback policy, that is, a map from the states of a deterministic or a probabilistic description of the external system.

Suppose the system evolves in discrete stages. Then, the coupled dynamical system can be written as

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

Here we use x' to refer to the next state, not the derivative of x. 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. The coupled system of internal and external described this way is an autonomous system, meaning that given an initial state $(\iota, x) \in \mathcal{I} \times X$ there exists a unique state trajectory.[1] We denote the function $(\iota, x) \mapsto (\iota', x')$ by $\phi *_{\pi,h} f$ which highlights that $\phi *_{\pi,h} f$ is a coupling of ϕ and f via the pair of coupling functions (π, h). Then, the coupled system is the pair

$$(\mathcal{I} \times X,\, \phi *_{\pi,h} f).$$

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, the tuples $\tilde{x}_k = (x_1, x_2, \ldots, x_k)$, $\tilde{u}_{k-1} = (u_1, u_2, \ldots, u_{k-1})$ and $\tilde{y}_k = (y_1, y_2, \ldots, y_k)$ correspond respectively to the state, action, and observation histories up to stage k, with $y_i = h(x_i)$ for $i \in \{1, \ldots, k\}$. 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, and following the state transition equation $\iota_k = \phi(\iota_{k-1}, y_k)$. At stage k, $\pi(\iota_k)$ produces the action u_k. Note that the stage index of the I-state starts from 0. In some cases, ι_0 can encode prior information regarding the external system and in others, it does not. We will consider this distinction more formally in Section 3.2. The next information state ι_1 is obtained using ι_0 and y_1. We assume that no control command (action) is outputted at stage 0, meaning that the control history starts with u_1. By convention, $\tilde{u}_0 = ()$ is an empty sequence.

2.2 Disturbances

The coupled internal-external systems formulation can be extended to include disturbances affecting the external system and the sensor. In particular, we can define two disturbance generating systems, with outputs $\theta \in \Theta$ and $\psi \in \Psi$, that are influencing the external system and the sensor, respectively (see Figure 2). Mathematically, the external system with disturbances is $(X, U \times \Theta, f)$, where $f\colon X \times (U \times \Theta) \to X$ is a state transition function for the external system under disturbances. Thus, the disturbances merely add a new dimension to the control parameters of the system. In the internal-external coupling, we also assume disturbances in the sensory mapping which takes the form

[1] In dynamical systems terminology, the coupled system is a closed, deterministic dynamical system. In the presence of extrinsic disturbances (see Section 2.2) the coupled system becomes a non-autonomous dynamical system.

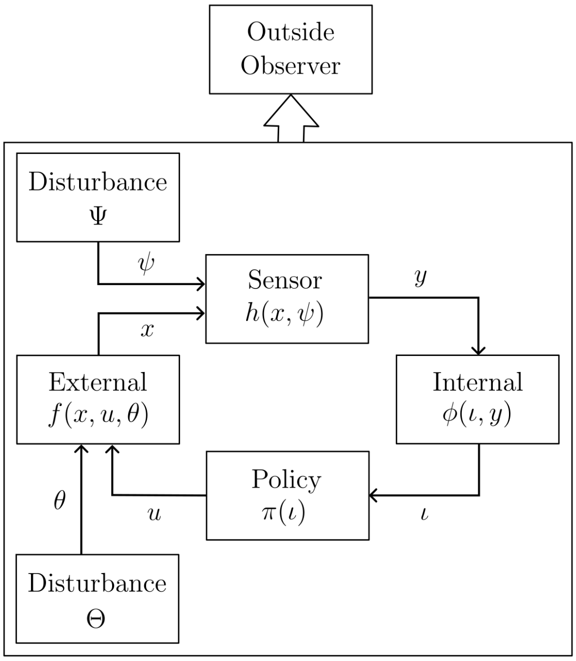


Figure 2. Disturbances may affect the external system and the sensor. Note that conditioned on the realization of these, the internal system and policy remain deterministic. An outside observer (planner / designer) may perceive the coupled system as a whole.

$h\colon X \times \Psi \to \mathcal{I}$. Then, the definition of the coupled internal-external system given in (1) is modified as follows:

$$\begin{aligned} \iota' &= \phi(\iota, y) && \text{in which } y = h(x,\psi), \\ x' &= f(x,u,\theta) && \text{in which } u = \pi(\iota'). \end{aligned} \tag{2}$$

Here, the other two functions ϕ (the information transition function) and π (policy) are as in Section 2.1.

Note that neither θ, which affects the state transition function of the external system, nor ψ, which affects the sensor mapping, is directly available to the internal system $(\mathcal{I}, Y, \phi)$, which is just as in Section 2.1. However, some information regarding the disturbances can be specified for an internal system that makes use of a model of the external. These could be encoded into the set $\mathcal{I}$ and the transition function ϕ. Then, the internal system has a non-trivial correlation with the disturbance, even though it is never directly perceived. We will consider the distinction between model-based and model-free systems in Section 3.5. Finally, the coupled system is mathematically a triple $(\mathcal{I} \times X, \Theta \times \Psi, g)$ where g is a function that, given a state $(\iota, x) \in \mathcal{I} \times X$ and a disturbance parameter $(\theta, \psi) \in \Theta \times \Psi$, outputs the next state in $\mathcal{I} \times X$, formally,

$$g\colon (\mathcal{I} \times X) \times (\Theta \times \Psi) \to \mathcal{I} \times X.$$

There are two possibilities for how the information regarding the disturbances can be specified: nondeterministic and probabilistic. In the nondeterministic case, the set Ψ and possibly a subset $\Psi(x) \subseteq \Psi$ is specified for all $x \in X$, in which $\Psi(x)$ represents the set of all ψ that can be realized for each x. In the probabilistic case, assuming that the disturbances are generated by a system that is Markovian, such that they do not depend on the previous stages, a probability distribution over Ψ can be specified for each x. This will be denoted as $P(\psi \mid x)$. The disturbances affecting the external system can be specified similarly to those that affect the sensor. In the nondeterministic case, the set $\Theta(x,u) \in \Theta$ is known for each $(x,u) \in X \times U$. In the probabilistic case, a probability distribution over Θ, that is, $P(\theta \mid x, u)$, can be specified for each (x,u).

2.3 Generalizing to transition systems

A *transition system* is a triple (S, Λ, T), in which, S and Λ are some sets (possibly equipped with some structure, for example, topology), and $T \subseteq S \times \Lambda \times S$ is a ternary relation. Here S is the set of states, Λ is the set of labels for transitions between elements of S, and a triple (s, λ, s') belongs to T if there is a transition from s to s' labeled by λ. A special case is when for each $(s,\lambda) \in S \times \Lambda$ there is a unique $s' \in S$ with $(s, \lambda, s') \in T$. Then, T defines a function $\tau\colon S \times \Lambda \to S$. These are called *deterministic transition systems*, and sometimes also *(open) dynamical systems*. An extensive analysis of those and their coupling is explored by (Spivak 2015).

All models in Sections 2.1 and 2.2 are deterministic transition systems: the external, the internal, the disturbed versions, and their couplings are all deterministic transition systems.[2] Note that if Λ is a singleton, the system (S, Λ, τ) is equivalent to a discrete time autonomous dynamical system. If (S, Λ, τ) is a deterministic transition system, S and Λ are finite, $s_0 \in S$, and $F \subseteq S$, then $(S, \Lambda, \tau, s_0, F)$ is a finite automaton as defined in (Sipser 2012, Definition 1.5). If not stated otherwise, we do not assume our systems to be finite. In Section 5.3 we explore connections between our theory with the theory of finite systems.

The following notion of state-relabeled transition systems was introduced in (Weinstein et al. 2022) to model the internal and external systems.

Definition 1. State-relabeled transition system. A state-relabeled transition system is the quintuple $(S, \Lambda, T, \sigma, L)$ in which $\sigma\colon S \to L$ is a labeling function and (S, Λ, T) is a transition system. The function σ is the *labeling function* and L is the set of labels.

A state-relabeled transition system is closely related to the *Moore machine* which is a state-relabeled transition system with a fixed initial state, a finite set of states, and finite sets of input and output alphabets (finite Λ and L) (Moore 1956).

In our framework, a labeling function σ serves two purposes; it enables a potential coupling by matching output of one system to the input of another, and acts as a categorization of the states of the system being labeled. Preimages of a labeling function σ induce a partition of the state space S into sets whose elements are indistinguishable through sensing. 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, using these equivalence

[2] Although the values of the disturbances, when present, are not known beforehand or deterministically, once the disturbed parts of the coupled system receive the value of the disturbance, they produce a uniquely determined output. Hence, these components and their couplings are deterministic transition systems.

classes, we can define a new transition system called the *quotient* of (S, Λ, T) by σ.

Definition 2. Quotient system. The quotient of (S, Λ, T) by σ is the transition system $(S/\sigma, \Lambda, T/\sigma)$, 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), in the sense that the map $s \mapsto [s]_\sigma$ is onto, but not necessarily one-to-one.[3] 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.

The external and internal systems can be written as state-relabeled deterministic transition systems (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, a coupled internal-external system can be described in terms of the state-relabeled transition systems formulation too, so that the output of one transition system is an input for another. Described this way, a coupling of two transition systems results in unique paths in either transition system, 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. An I-space is a collection of all possible I-states. We will use the acronym ITS to refer to an *information transition system*, that is, a transition system whose state space is an I-space.

We have already used the notion of an I-space when modeling the internal system representing the robot brain, which we view as an ITS. Here, we extend the notion of an ITS to include different perspectives from which the external and the coupled systems can be viewed. In particular, we identify three perspectives;

- a planner,
- a plan executor,
- and an (independent) observer.

With a slight abuse of previously introduced notation and terminology, we use the term *internal* to refer to any system that is not the external system and we use $\mathcal{I}$ to denote a generic I-space. We use the term *deterministic information transition system* (DITS) to refer to an ITS for which the transitions are governed by an information transition function so that they are deterministic. We denote these types of systems by $(\mathcal{I}, \Lambda, \phi)$, in which Λ is the edge labeling and $\phi\colon \mathcal{I} \times \Lambda \to \mathcal{I}$ is an information transition function. Otherwise, an ITS will be called a *nondeterministic information transition system* (NITS) and denoted by $(\mathcal{I}, \Lambda, \Phi)$, in which $\Phi \subseteq \mathcal{I} \times \Lambda \times \mathcal{I}$ is the transition relation.

Suppose $(\mathcal{I}, U \times Y, \Phi)$ is a NITS and $\pi\colon \mathcal{I} \to U$ a policy, and define

$$\Phi_\pi := \{(\iota, (u, y), \iota') \in \Phi \mid u = \pi(\iota)\} \subseteq \Phi. \qquad (3)$$

The transition system $(\mathcal{I}, U \times Y, \Phi_\pi)$ is called the *restriction* of $(\mathcal{I}, U \times Y, \Phi)$ by the policy π. [4] If $(\mathcal{I}, U \times Y, \phi)$ is a DITS, the *strong restriction* by $\pi\colon \mathcal{I} \to U$ is given by $(\mathcal{I}, Y, \phi_\pi)$, in which $\phi_\pi\colon \mathcal{I} \times Y \to \mathcal{I}$ and $\phi_\pi(\iota, y) = \phi(\iota, \pi(\iota), y)$. The strong restriction is obtained by first taking the restriction of ϕ treated as a subset of $\mathcal{I} \times (U \times Y) \times \mathcal{I}$ and then taking the projection of the resulting set onto $\mathcal{I} \times Y \times \mathcal{I}$.

Before any policy is fixed, a DITS of the form $(\mathcal{I}, U \times Y, \phi)$ corresponds to the *planner* perspective. Once the policy is fixed, the strong restriction $(\mathcal{I}, Y, \phi_\pi)$, which is just as the internal system was defined in Section 2, corresponds to the *plan executor*.

Example 1. *A binary toy model*. Consider the DITS $(\mathcal{I}, U \times Y, \phi)$ which corresponds to a planner perspective. Suppose $U = Y = \mathcal{I} = \{0, 1\}$ and let $\phi\colon \mathcal{I} \times (U \times Y) \to \mathcal{I}$ be defined by $\phi(\iota, (u, y)) = |y - u|$. Suppose a policy $\pi : \mathcal{I} \to U$ is fixed such that $\pi(\iota) = \iota$. Then, $(\mathcal{I}, Y, \phi_\pi)$, in which $\phi_\pi(\iota, y) = |y - \pi(\iota)|$, is the strong restriction of $(\mathcal{I}, U \times Y, \phi)$ by π. Furthermore, it corresponds to the plan executor.

In this paper, an observation will refer to a sensor reading y. However, when we discuss an (independent) observer described over the coupled system, the input to this observer system can be a function of any variable of the coupled system, for instance action, information state or the state of the external. If the coupled system is disturbed, the disturbances can be observed by the observer too.

3.2 History information spaces

The most fundamental I-space is the *history I-space*, which we denote by $\mathcal{I}_{hist}$. 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. In this sense, $\mathcal{I}_{hist}$ is the canonical I-space, and all the other I-spaces are derived from it. We denote the history I-states by the letter η to distinguish them from the states of other information spaces, which we typically denote by ι (recall the notation introduced in Section 2.1).

Let U and Y be the sets of possible actions and observations respectively. The elements of $\mathcal{I}_{hist}$ are finite sequences of alternating actions and observations which build upon some initial state $\eta_0 \in \mathcal{I}_{hist}$. Denote the set of possible initial states of $\mathcal{I}_{hist}$ by $\mathcal{I}_0$. Then the elements of $\mathcal{I}_{hist}$ are of the form

$$(\eta_0, \tilde{u}_{k-1}, \tilde{y}_k) := (\eta_0, y_1, u_1, y_2, u_2 \ldots, u_{k-1}, y_k) \qquad (4)$$

[3] Considering homomorphisms between transition systems goes beyond the present paper; a good source for the general theory is (Goranko and Otto 2007).

[4] Note that we can treat the external system symmetrically and given a non-deterministic transition system $(X, Y \times U, F)$ with $F \subseteq X \times (Y \times U) \times X$ and a sensor mapping $h\colon X \to Y$, we can define the restriction F_h in an analogous way as above. However, because it is not central to this paper we will not elaborate on this topic.

for $k \in \mathbb{N}$, in which $\eta_0 \in \mathcal{I}_0$, $u_i \in U$ and $y_i \in Y$ for all $i \leq k$. Additionally, denote

$$\eta_k := (\eta_0, \tilde{u}_{k-1}, \tilde{y}_k). \tag{5}$$

The notations (4) and (5) follow (LaValle 2006, Chapter 11). The lower index k refers to the *stage* of the state, or the length of the action-observation sequence. The convention here, as already mentioned in the end of Section 2.1, is that $\tilde{u}_0$ is assumed to be the null-tuple. Thus, $\eta_k = (\eta_0, \tilde{u}_{k-1}, \tilde{y}_k)$ is the I-state at stage k, which is achieved by iteratively concatenating the action-observation pairs (u_{i-1}, y_i) at the end of the sequence for $i \in \{1, \dots, k\}$ after the initial state η_0.

The description of initial conditions in the set $\mathcal{I}_0$ varies with the available prior information. We discuss these descriptions below. The history information space at stage k is the subset of $\mathcal{I}_{hist}$ which consists of elements of the form given by (4) for fixed k, and can be expressed as the product

$$\mathcal{I}_k := \mathcal{I}_0 \times \tilde{U}_{k-1} \times \tilde{Y}_k, \tag{6}$$

in which $\tilde{U}_{k-1} = U^{k-1}$ and $\tilde{Y}_k = Y^k$. In general, the number of stages that the system will go through is not fixed. Therefore we assume the history I-space to contain all finite action-observation sequences, that is, $\mathcal{I}_{hist} = \bigcup_{k \in \mathbb{N}} \mathcal{I}_k$. The DITS corresponding to $\mathcal{I}_{hist}$ is $(\mathcal{I}_{hist}, U \times Y, \phi_{hist})$, in which

$$\phi_{hist}(\eta, u, y) = \eta^\frown(u, y),$$

and $^\frown$ is the concatenation of two sequences. Note that the concatenation operation makes $(\mathcal{I}_{hist}, ^\frown)$ into a free monoid. The derived information transitions systems which will be introduced in Section 3.4 can be seen as quotients of this monoid by equivalence relations; sometimes these quotients can also be monoids, or even groups.

3.3 Sufficient state-relabeling

In (Weinstein et al. 2022) we have introduced a notion of *sufficiency* that generalizes the definition introduced in (LaValle 2006, Chapter 11) and is presented here for completeness.

Definition 3. Sufficient labeling function. Let (S, Λ, T) be a transition system. A labeling function $\sigma\colon S \to L$ defined over the states of a transition system is *sufficient* if and only if for all $s, t, s', t' \in S$ and all $\lambda \in \Lambda$, the following implication holds:

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

If σ is defined over the states of a deterministic transition system (S, Λ, τ), then σ is sufficient if and only if for all $s, t \in S$ and all $\lambda \in \Lambda$, $\sigma(s) = \sigma(t)$ implies that $\sigma(\tau(s, \lambda)) = \sigma(\tau(t, \lambda))$.

Consider the stage-based evolution of the state-relabeled deterministic transition system corresponding to the external system (X, U, f, h, Y) with respect to the action (control input) sequence $\tilde{u}_{k-1} = (u_1, \dots, u_{k-1})$. This corresponds to the state and observation histories till stage k, that are $\tilde{x}_k = (x_1, \dots, x_k)$ and $\tilde{y}_k = (y_1, \dots, y_k)$. 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 equivalence classes induced by h. This notion is similar to a minimal realization of a system, that is, the minimal state space description that models the given input-output measurements (see for example (Kotta et al. 2018)). Another 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, the complete observation trajectory can be determined from the initial observation by induction.

Now, consider an internal system with a labeling function $\kappa\colon \mathcal{I} \to \mathcal{I}'$, that is, $(\mathcal{I}, U \times Y, \phi, \kappa, \mathcal{I}')$, and its evolution with respect to the histories $\tilde{y} = (y_1, \dots, y_k)$ and $\tilde{u} = (u_1, \dots, u_{k-1})$. At stage k, the state of the DITS 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 (LaValle 2006, Chapter 11) and makes it a special case of Definition 3.

3.4 Derived information transition systems

Even though it seems natural to rely on a history ITS, the dimension of a history I-state increases linearly, and the size of the history I-space increases exponentially, as a function of the stage index, making it impractical in most cases. Thus, we are interested in defining a reduced ITS that is more manageable, due to for example lowered requirements for memory or computing power. 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 labeling function (see Definition 2). We rewrite $(\mathcal{I}_{hist}, U \times Y, \phi_{hist})$ as $(\mathcal{I}_{hist}, U \times Y, \Phi_{hist})$, in which

$$\Phi_{hist} = \Big\{ \big(\eta, (u, y), \eta'\big) \in \mathcal{I}_{hist} \times (U \times Y) \times \mathcal{I}_{hist} \mid \eta' = \phi_{hist}(\eta, u, y) \Big\}. \tag{7}$$

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

It is crucial that the derived ITS is a DITS so that the transition from the current label to the next can be determined using only the derived ITS, without making reference to the history ITS. The reason for this requirement is straightforward for an observer as the I-states correspond to what is inferred about the external system, given observation history (potentially accompanied by the action history). The same applies for the planner and the plan executor to be able to describe and execute a policy.

Considering the quotient system derived by κ from the DITS (by definition) $(\mathcal{I}_{hist}, U \times Y, \phi)$, we cannot always guarantee that the resulting ITS is deterministic. This depends on the I-map used for state-relabeling, as illustrated in the following proposition.

Proposition 1. Quotient of a history ITS may be a NITS. *For all non-empty U and Y, and for the corresponding $\mathcal{I}_{hist}$, there exists a labeling function κ such that the quotient $(\mathcal{I}_{hist}/\kappa, U \times Y, \Phi/\kappa)$ of $(\mathcal{I}_{hist}, U \times Y, \phi)$ by κ, in which Φ is defined as in (7), is not a DITS.*

Proof. Let $\kappa \colon \mathcal{I}_{hist} \to \{l_1, l_2\}$ and define $\kappa^{-1}(l_1) = \{\eta_k = (\tilde{u}_{k-1}, \tilde{y}_k) \in \mathcal{I}_{hist} \mid \tilde{u}_{k-1} = (u_i)_{i=1}^{k-1}, u_i = u$ for $1 \leq i \leq k-1$, and $k > 3\}$, and $\kappa^{-1}(l_2) = \mathcal{I}_{hist} \setminus \kappa^{-1}(l_1)$. Then $\kappa^{-1}(l_1)$ is the set of histories of length $k > 3$ which correspond to applying the same action u for $k-1$ times, and $\kappa^{-1}(l_2)$ is its complement. 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. □

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 there exist $s, s', q, q' \in \mathcal{I}$ such that $\{(s, (u, y), s'), (q, (u, y), q')\} \in \Phi$, in which Φ is defined using ϕ as in (7). 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.

Remark 1. *Whether the quotient system derived from $(\mathcal{I}_{hist}, U \times Y, \phi)$ is a DITS depends on the sufficiency of κ. In (Weinstein et al. 2022, Proposition 4.5) it is shown that the quotient of a transition system (S, Λ, T) by a labeling function σ is a deterministic transition system if and only if (S, Λ, T) is full*[5] *and σ is sufficient.*

As ϕ_{hist} is a function with domain $\mathcal{I}_{hist} \times (U \times Y)$, it is full, so the following is implied by (Weinstein et al. 2022) as a special case:

Proposition 2. A Quotient system is a DITS when the labeling is sufficient. *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 (7). Then $(\mathcal{I}_{hist}/\kappa, U \times Y, \Phi/\kappa)$ is a DITS if and only if κ is sufficient.*

Remark 2. *For an I-map $\kappa \colon \mathcal{I}_{hist} \to \mathcal{I}_{der}$, the quotient $(\mathcal{I}_{hist}/\kappa, U \times Y, \Phi_{hist}/\kappa)$ is isomorphic to $(\mathcal{I}_{der}, U \times Y, \Phi_{der})$, in which*

$$\Phi_{der} = \{(\kappa(\eta), (u, y), \kappa(\eta')) \mid (\eta, (u, y), \eta') \in \Phi_{hist}\}$$

(Weinstein et al. 2022, Proposition 2.37). Thus, we can use the labels introduced by an κ as the new (derived) I-space and the corresponding quotient system as the derived ITS.

Suppose an I-map κ is sufficient. Then, the derived ITS is a DITS, meaning that given an I-state ι_{k-1} in the derived space $\mathcal{I}_{der}$, and (u_{k-1}, y_k), $\iota_k \in \mathcal{I}_{der}$ can be uniquely determined. Consequently, we can write the derived ITS as $(\mathcal{I}_{der}, U \times Y, \phi_{der})$ in which

$$\phi_{der} \colon \mathcal{I}_{der} \times (U \times Y) \to \mathcal{I}_{der}$$

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:

$$\begin{tikzcd}
\mathcal{I}_{hist} \arrow[r, "u_1{,}y_2"] \arrow[d, "\kappa"] & \mathcal{I}_{hist} \arrow[r, "u_2{,}y_3"] \arrow[d, "\kappa"] & \mathcal{I}_{hist} \arrow[r, "u_3{,}y_4"] \arrow[d, "\kappa"] & \mathcal{I}_{hist} \arrow[d, "\kappa"] & \cdots \\
\mathcal{I}_{der} \arrow[r, "u_1{,}y_2"] \arrow[d, "\kappa'"] & \mathcal{I}_{der} \arrow[r, "u_2{,}y_3"] \arrow[d, "\kappa'"] & \mathcal{I}_{der} \arrow[r, "u_3{,}y_4"] \arrow[d, "\kappa'"] & \mathcal{I}_{der} \arrow[d, "\kappa'"] & \cdots \\
\mathcal{I}_{min} \arrow[r, "u_1{,}y_2"] \arrow[d, "\kappa''"] & \mathcal{I}_{min} \arrow[r, "u_2{,}y_3"] \arrow[d, "\kappa''"] & \mathcal{I}_{min} \arrow[r, "u_3{,}y_4"] \arrow[d, "\kappa''"] & \mathcal{I}_{min} \arrow[d, "\kappa''"] & \cdots \\
\mathcal{I}_{task} & \mathcal{I}_{task} & \mathcal{I}_{task} & \mathcal{I}_{task} &
\end{tikzcd} \tag{8}$$

Note that we can similarly define an I-map that maps any derived I-space to another. An example is given in (8) as the mappings $\kappa' \colon \mathcal{I}_{der} \to \mathcal{I}_{min}$ and $\kappa'' \colon \mathcal{I}_{min} \to \mathcal{I}_{task}$. In this example, κ' is sufficient, visible also from the commutativity of the respected square in the diagram. This implies that the quotient system derived by κ' is deterministic. On the other hand, κ'' is not sufficient, meaning that the derived ITS is not deterministic: given an element of $\mathcal{I}_{task}$ one cannot uniquely determine the next I-state using the derived ITS only. This is shown in (8) with the missing arrows at the respective row of $\mathcal{I}_{task}$. Hence, for κ'' the diagram does not commute. 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 example, $\kappa_{min} \colon \mathcal{I}_{hist} \to \mathcal{I}_{min}$ is the composition of κ and κ', that is, $\kappa_{min} = \kappa' \circ \kappa$ (same for $\kappa_{task} \colon \mathcal{I}_{hist} \to \mathcal{I}_{task}$).

3.5 Model-based and model-free

In machine learning, control, and robotics literature, methods are often categorized into *model-based* and *model-free* (or data-driven) ones. Informally, using our setup, a model-based scenario is one where the derived I-state is allowed to depend on knowledge about the external system, the sensor mapping, the initial state, and the disturbances acting on the external system or on the sensors (if there are any). The model-free scenario in contrast cannot depend on those, but can depend on data, which in our case is the history I-states, that is, the sequences of actions and observations.

In Section 2.1 we have defined internal-external coupled systems. Their coupling (1) produces an autonomous system $(\mathcal{I} \times X, \phi *_{\pi,h} f)$. However, we can choose not to consider either one of the coupling functions π and h, and be left with a system that still has a control parameter. For example, let $(\mathcal{I} \times X, U, \phi *_h f)$ be a system where the evolution of states can be written as

$$\begin{aligned} \iota' &= \phi(\iota, y) \qquad \text{in which } y = h(x), \\ x' &= f(x, u). \end{aligned} \tag{9}$$

Here, $u \in U$ is a control parameter on which the next state always depends. This system represents the coupled system

[5] 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 $(s, \lambda, s') \in T$.

before a policy π has been defined over the states of the internal.

Note that the internal system only has access to the current information state $\iota \in \mathcal{I}$, not to the external state $x \in X$. One can notationally express this perspective by evolving the internal system by an externally parametrized information transition function $\phi_{f,h}(\cdot\,;\,x)$, which maps the current I-state and action pair $(\iota, u) \in \mathcal{I} \times U$ to the next I-state $\iota' \in \mathcal{I}$. The maps h (which couples the external to the internal system) and f are subsumed into the global map $\phi_{f,h}$ which is additionally parametrized by the current state $x \in X$ of the external system. Thus, in accordance with (9), we define $\phi_{f,h}$ for each $(\iota, u) \in \mathcal{I} \times U$ and $x \in X$ by

$$\phi_{f,h}(\iota, u\,;\,x) := \phi(\iota, h(f(x,u))). \tag{10}$$

If the I-space in (9) is the history I-space, we can write (9) as $(\mathcal{I}_{hist} \times X, U, \phi *_h f)$, and its internal system perspective (10) becomes $(\mathcal{I}_{hist}, U, \phi_{f,h})$. We propose that a method of obtaining a derived I-space corresponding to an I-map κ is *model-based*, if κ is obtained as a function of $(\mathcal{I}_{hist} \times X, U, \phi *_h f)$, while it is *model-free*, if it is obtained as a function of it from the perspective of the internal system, that is, as a function of $(\mathcal{I}_{hist}, U, \phi_{f,h})$.

The distinction between model-based and model-free is also seen in the initial states η_0 of the history I-space. In model-based setups, typically η_0 is a subset of X, or a probability distribution over X while in model-free setups η_0 is an empty sequence. Examples 9 and 11 are examples of model-free and model-based I-spaces respectively.

Note that this formalization implies that model-free methods are a subset of model-based. This is because the internal perspective is itself a function of the entire coupled system, so anything that is a function of the internal perspective is by transitivity also a function of the entire coupled system. This matches the intuition that model-based are ones where more information is available. We leave the exploration of more aspects of this distinction and its formalization for future work.

We now present two examples that illustrate model-based and model-free derived ITSs.

Example 2. *Bayesian filter*. Suppose the initial history information state encodes a probability distribution over X such that $\eta_0 = P(x_1)$. We refer to the coupled system including the disturbances described in (2). A Markovian, probabilistic model of the disturbances is given in the form of conditional distributions $P(\psi \mid x)$ over Ψ, and $P(\theta \mid x, u)$ over Θ. In the former, conditioning takes place relative to external states $x \in X$, and in the latter relative to state-action pairs $(x, u) \in X \times U$. Using the definitions of f and h given in (2), $P(y_k \mid x_k)$ and $P(x_{k+1} \mid x_k, u_k)$ can be derived from $P(\psi_k \mid x_k)$ and $P(x_{k+1} \mid x_k, u_k)$ for all stages k.

Let $\mathcal{I}_{prob}$ be the set of all probability distributions defined over X and let $\mathcal{I}_{hist}$ be a history information space with $\mathcal{I}_0 = \mathcal{I}_{prob}$ such that η_0 is a probability distribution over X, that is $P(x_1)$. An ITS can be derived by $\kappa_{prob}\colon \mathcal{I}_{hist} \to \mathcal{I}_{prob}$ such that $\kappa_{prob}(\eta_k) = \iota_k = P(x_k \mid \eta_k)$. Note that we can write η_k as $\eta_k = \eta_{k-1}{}^\frown(u_{k-1}, y_k)$. The I-state $\iota_k = P(x_k \mid \eta_k)$ can be inductively computed from ι_{k-1} and (u_{k-1}, y_k) using marginalization and Bayes' rule starting from $\iota_1 = P(x_1 \mid y_1)$, in which $\eta_1 = y_1$. This corresponds to defining $\phi\colon \mathcal{I} \times (U \times Y) \to \mathcal{I}$ such that $\iota_k = \phi(\iota_{k-1}, (u_{k-1}, y_k))$. Then, $\kappa(\eta_{k-1}{}^\frown(u_{k-1}, y_k)) = \phi \circ \kappa(\eta_{k-1}) = P(x_k \mid \eta_k)$ which shows that κ_{prob} is sufficient. Hence, a Bayesian filter can be modeled as a derived DITS whose state space is $\mathcal{I}_{prob}$. Note that in this case, κ_{prob} is defined as a function of $(\mathcal{I}_{hist} \times X, U, \phi *_h f)$, making it model-based.

Note that the Kalman filter is a special case of a Bayesian filter when f and h are linear and the disturbances are Gaussian. These specifications imply that all the posterior distributions are Gaussian as well. Therefore, in this special case, the range of κ_{prob} is implicitly restricted to the set of all Gaussian distributions, denoted as $\mathcal{I}_{Gauss}$, such that $\kappa_{prob}\colon \mathcal{I}_{hist} \to \mathcal{I}_{Gauss} \subset \mathcal{I}_{prob}$. This restriction allows the I-state to simply encode only the mean and the covariance of a multivariate Gaussian distribution, that is, $\iota = (\hat{x}, \Sigma)$, in which $\hat{x}$ is the mean and Σ is the covariance matrix, without violating the sufficiency of κ_{prob}. An extension of the Kalman filter to nonlinear systems is the Extended Kalman Filter (EKF). In the case of EKF, the functions f and h are not linear. This violates the posterior distribution being Gaussian even if the disturbances are. However, the states of the EKF are defined as elements of $\mathcal{I}_{Gauss}$ and a state transition function $\phi\colon \mathcal{I}_{Gauss} \times (U \times Y) \to \mathcal{I}_{Gauss}$ is described that relies on linearizing f and h at each I-state. Note that even though the Kalman filter and the EKF share the same underlying I-space, the corresponding I-maps that derive these transition systems are different.

The following is an example of a model-free derived ITS.

Example 3. Moving average filter. Let $Y = \mathbb{R}$ and $\kappa_k : \mathcal{I}_k \to \mathbb{R}$, in which $\mathcal{I}_k$ is the set of k stage histories. A moving average filter (observation only) with a window size n can be derived from $\mathcal{I}_k$ as

$$(\tilde{u}_{k-1}, \tilde{y}_k) \mapsto \frac{1}{n} \sum_{i=k-n+1}^{k} y_i.$$

3.6 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 $\mathcal{I}_{hist}/\kappa$.

Definition 4. Refinement of an I-map. 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), $\kappa_{id}\colon \mathcal{I}_{hist} \to \mathcal{I}_{hist}$, 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 $\kappa_{const}\colon \mathcal{I}_{hist} \to \mathcal{I}_{const}$ 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 see (LaValle 2012; Zhang and Shell 2021). Indeed, if we take $\mathcal{I}_0 = X$ and consider $\kappa_{est} \colon \mathcal{I}_{hist} \to X$, 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 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 5. Minimal sufficient refinement. 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 satisfies $\kappa' \succ \kappa'' \succeq \kappa$.

Remark 3. *It is shown in (Weinstein et al. 2022, Theorem 4.19) that the minimal sufficient refinement of κ defined over the states of a deterministic transition system (S, Λ, τ) is unique up to relabeling, namely if $\kappa_{\min}$ and $\kappa'_{\min}$ are minimal sufficient refinements, then $\kappa_{\min} \succ \kappa'_{\min}$ and $\kappa'_{\min} \succ \kappa_{\min}$.*

4 Solving Tasks Minimally

4.1 Definition of a task

In this section, we formulate general planning and filtering tasks within the framework of information transition systems. We distinguish between two categories: 1) *active*, which entails planning and executing an information-feedback policy that forces a desirable outcome in the external system, and 2) *passive*, which refers to only observing the external system without being able to effect changes. We next describe active and passive tasks for the model-free and model-based I-space formulations, introduced in Section 3.5. In the model-free case, tasks are specified using a logical language over $\mathcal{I}_{hist}$. This results in a labeling, a derived I-space $\mathcal{I}_{task}$, and the associated I-map κ_{task}. Various logics are allowable, such as propositional, modal, or a temporal logic. The resulting sentences of the language involve combinations of predicates that assign true or false values to subsets of $\mathcal{I}_{hist}$. Solving an active task 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 $G \subseteq \mathcal{I}_{hist}$, causing a predicate IN-GOAL$(\mathcal{I}_{hist})$ to become satisfied (in other words, be true).

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 sentences. In this case, each sentence induces a partition of $\mathcal{I}_{hist}$, and the task labeling over $\mathcal{I}_{hist}$ assigns a label to each set in the common refinement of these partitions. 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.

Some naturally occurring robot tasks can only be described in terms of infinite sequences of actions and observations. These are called *infinitary tasks*. For example, cycling through a finite sequence of subsets of X indefinitely while avoiding others (Fainekos et al. 2009) can only be described in terms of infinite histories. For this task, whether the sentence of interest is satisfied cannot be determined based on a finite history of any given length. However, the histories that fail, that is, those for which the sentence of interest becomes false, can be defined in terms of finite histories (namely those that result in a state that needed to be avoided). Interested reader can refer to (Kress-Gazit et al. 2009) for examples based on linear temporal logic (LTL).

Infinitary tasks are defined on the set of infinite histories $\mathcal{I}^{\infty}_{hist}$ which consists of infinite sequences of the form $\bar{\eta} = (\eta_0, y_1, u_1, y_2, u_2, \cdots)$. These are the elements of the infinite Cartesian product

$$\mathcal{I}_0 \times (Y \times U) \times (Y \times U) \times \cdots = \mathcal{I}_0 \times \prod_{k=1}^{\infty} (Y \times U).$$

The preimages of an *infinitary task labeling* $\bar{\kappa} : \mathcal{I}^{\infty}_{hist} \to \mathcal{I}_{task}$ are subsets of $\mathcal{I}^{\infty}_{hist}$. Although the satisfiability of an infinitary task may depend on infinite sequences, these can nevertheless be characterised in terms of finite initial segments as follows. Any subset $H \subseteq \mathcal{I}^{\infty}_{hist}$ can be written as

$$H = I_0 \times \prod_{k=1}^{\infty} (Y_k \times U_k), \tag{11}$$

in which $I_0 \subseteq \mathcal{I}_0$, and $Y_k \subseteq Y$, $U_k \subseteq U$ for all $k \in \mathbb{N}$. For each $m \in \mathbb{N}$, we denote by $H(m)$ the collection of subsets of $\mathcal{I}^{\infty}_{hist}$ for which $Y_k = Y$ and $U_k = U$ for all $k > m$. In other words, such collections of histories are constrained only at a finite number of stages.

Now, let $\iota \in \mathcal{I}_{task}$ and suppose an equivalence class induced by the preimage $\bar{\kappa}^{-1}(\iota)$ is a (potentially infinite) union

$$\bar{\kappa}^{-1}(\iota) = \bigcup_{\alpha \in A} H_{\alpha}, \tag{12}$$

in which A is some index set and each H_{α} belongs to $H(m)$ for some m. Then, whether a particular history $\bar{\eta} \in \mathcal{I}^{\infty}_{hist}$ belongs to $\bar{\kappa}^{-1}(\iota)$ is determined by a finite number of stages in an initial segment of this history. In general, however, the length of these initial segments is not bounded from above.

To characterize infinitary tasks in terms of deciding their satisfiability, we rely on topology. Assume that some topology is defined for the sets $\mathcal{I}_0, Y$, and U. If these are finite sets, a natural choice is the discrete topology in which every singleton is an open set. For subsets of $\mathbb{R}^n$, a natural choice would be the restriction of the usual Euclidean metric. The base H° of the product topology in $\mathcal{I}_{hist}^\infty$ consists of those sets H for which $H \in H(m)$ for some $m \in \mathbb{N}$, and the sets I_0, Y_k, U_k in (11) satisfy that I_0 is open in $\mathcal{I}_0$, and Y_k, U_k are open in Y, U, respectively, for all $k \in \mathbb{N}$. All other open sets are obtained as arbitrary unions of sets $H \in H^\circ$. In particular, when the sets H_α in (12) satisfy $H_\alpha \in H^\circ$ for all $\alpha \in A$, the corresponding preimage $\bar{\kappa}^{-1}(\iota)$ is an open set. If the sets $\mathcal{I}_0, Y, U$ are compact, the space $\mathcal{I}_{hist}^\infty$ is compact in the product topology. This is the case for example when $\mathcal{I}$, Y, and U are finite sets with the discrete topology.

In the simplest nontrivial case, a task labeling $\bar{\kappa}$ concerns a single sentence. Then, $\mathcal{I}_{task} = \{0, 1\}$ so that the preimages of $\bar{\kappa}$ partition $\mathcal{I}_{hist}^\infty$ into the equivalence classes $\bar{\kappa}^{-1}(1)$, that is, the set of histories for which the sentence is satisfied, and $\bar{\kappa}^{-1}(0)$, which is the set of histories that result in false. We call $\bar{\kappa}^{-1}(1)$ and $\bar{\kappa}^{-1}(0)$ the *success set* and *fail set*, respectively. If the success set of a given task is open, we call this an *open task*. A *closed task* is one whose fail set is open, so that its success set is closed. It is possible for a task to be both open and closed, so that both the success and fail sets are both open and closed. We call such tasks *clopen*.

Due to the definition of the sets $H(m)$, the membership of a given history in an open set is determined by some finite initial segment of that history. Therefore, based on a finite length segment of a given history, we can determine its membership in the success set of an open task, in the fail set of a closed task, and both the success and fail sets for a clopen task. In these cases, task satisfiability can be defined in terms of the elements of $\mathcal{I}_{hist}$. We can thus transcribe an infinitary task labeling $\bar{\kappa} : \mathcal{I}_{hist}^\infty \rightarrow \mathcal{I}_{task}$ in the form of $\kappa : \mathcal{I}_{hist} \rightarrow \mathcal{I}_{task}$. This amounts to assigning to each finite history $\eta \in \mathcal{I}_{hist}$ some task label $\iota \in \mathcal{I}_{task}$ in such a way that this labeling expresses the success set of the corresponding infinitary task $\bar{\kappa}$. Recall that $\mathcal{I}_{hist} = \bigcup_{k \in \mathbb{N}} \mathcal{I}_k$, in which $\mathcal{I}_k$ are as in (6). Suppose $\bar{\kappa}^{-1}(\iota)$ is an open set for some $\iota \in \mathcal{I}_{task}$ so that $\bar{\kappa}^{-1}(\iota) = \bigcup_{\alpha \in A} H_\alpha$ as in (12). Then, for a finite history $\eta = (\eta_0, y_1, u_1, \ldots, u_{k-1}, y_k) \in \mathcal{I}_k$ we define $\kappa(\eta) = \iota$ if and only if there exists some $m \leq k$, some index $\alpha \in A$ and a collection of infinite histories

$$H(\eta) = \mathcal{I}_0 \times (Y_1 \times U_1) \times \cdots \times (Y_m \times U_m) \times \prod_{n=m+1}^{\infty} (Y \times U),$$

for which $H(\eta) \in H_\alpha \in H(m)$, $\eta_0 \in \mathcal{I}_0$ and $y_n \in Y_n$, $u_n \in U_n$ for all $1 \leq n \leq m$.

Below are examples of typical model-based task descriptions with their corresponding definitions in terms of infinite histories that can be expressed using a task-labeling over $\mathcal{I}_{hist}$.

Example 4. ***Reach state*** $x \in X$ ***from an initial state*** $x_0 \in X$. The success set of this task consists of those histories that correspond to the external system arriving to x in some finite time. If $H_m \subseteq \mathcal{I}_{hist}^\infty$ contains those histories in which x is visited for the first time at stage m, the success set of this task is $\bar{\kappa}^{-1}(1) = \bigcup_{m \in \mathbb{N}} H_m$. Assuming $H_m \in H^\circ$ for every $m \in \mathbb{N}$, this task is open.

Example 5. ***Never visit state*** $x \in X$. The fail set of this task consists of those histories that correspond to the external system arriving at x from some initial state x_0. Since this always happens in finite time (or not at all), the fail set can be expressed as the union $\bigcup_{m \in \mathbb{N}} H_m$, where H_m consists of all the histories in which state x is reached for the first time on stage m. Assuming $H_m \in H^\circ$ for all $m \in \mathbb{N}$, the task is therefore closed.

Example 6. ***Reach state*** x_1 ***while avoiding state*** x_2. The success set of this task may be written as the union, over $m \in \mathbb{N}$, of histories which reach x_1 for the first time after m stages, and did not visit x_2 during the first $m-1$ stages. Assuming these sets are open, the success set is thus open. With an analogous argument, it is seen that the fail set is also open.

An infinitary task is not necessarily either open or closed. One example of this are tasks that can be expressed as so-called G_δ sets (Dugundji 1978, Section 3.6), that is, infinite intersections of open sets (see Example 7).

Example 7. ***Revisit state*** $x \in X$ ***infinitely many times***. In this case, neither the success set or fail set of this task can be defined in terms of the sets $H(m)$, since no finite length history can rule out either success or failure. For each $m, k \in \mathbb{N}$, define $H_{m,k} = \{\bar{\eta} \in \mathcal{I}_{hist}^\infty \mid$ at stage m, the next visit to x happens after k stages$\}$. Then the success set is given by $\bar{\kappa}^{-1}(1) = \bigcap_{m \in \mathbb{N}} \bigcup_{k \in \mathbb{N}} H_{m,k}$ which is an infinite intersection of open sets if $H_{m,k} \in H^\circ$ for all m, k. This set is not generally open, but belongs to the broader class G_δ.

In this paper, we consider tasks that are expressed as a labeling over $\mathcal{I}_{hist}$ or those over $\mathcal{I}_{hist}^\infty$ that can be transcribed as one over $\mathcal{I}_{hist}$. Hence, the problem families that we will introduce in Section 4.2 would refer to these types of tasks. All the tasks in examples in Section 5 are either open or closed and thus are representable by either a finitary success or failure condition. More generally, if one defines tasks using any common version of temporal logic, the corresponding success sets are always going to be Borel, that is, members of the sigma algebra generated by open sets (Dugundji 1978, Section 3.6).

4.2 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 cannot guarantee that the quotient system by κ_{task} is a DITS (Propositions 1 and 2). This implies that relying solely on the quotient system by κ_{task}, we cannot determine the class that the current history belongs to (see the last row in (8)). Hence, we cannot 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, the action 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 $\kappa_{task} \colon \mathcal{I}_{hist} \to \mathcal{I}_{task}$ is a task labeling (that is not assumed to be 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 cannot be relinquished without losing predictability, or success guarantees. Consequently, histories belonging to different equivalence classes with respect to κ_{task} must always be distinguished from each other.

Example 8. *Goal recognition*. Suppose $\kappa_{task} \colon \mathcal{I}_{hist} \to \mathcal{I}_{task}$ is a labeling that partitions $\mathcal{I}_{hist}$ into two disjoint sets; $\mathcal{I}_{task} = \{\iota_G, \iota_{NG}\}$, in which ${\kappa_{task}}^{-1}(\iota_G)$ and ${\kappa_{task}}^{-1}(\iota_{NG})$ correspond to histories that lead to goal and the ones that do not, respectively. Suppose the goal is recognizable, meaning that, solely based on y_k, the value of $\kappa_{task}(\eta_k)$ is known, for all $\eta_k \in \mathcal{I}_{hist}$ and $k > 0$. Then, κ_{task} is trivially sufficient (also minimal). However, if the sensor mapping does not directly provide this information, then a refinement is needed to describe a sufficient filter that infers whether the goal is reached.

Notice that Problem 1 does not impose 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. Find a minimal sufficient I-filter. *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 formally defined in the following. Consider an external system (X, U, f, h, Y). Let $\mathcal{R}_X(\mathcal{I}_{task}) \subseteq X$ be the set of initial states for which there exist a $k \in \mathbb{N}$ 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 $\tilde{y}_k$. Informally, $\mathcal{R}_X(\mathcal{I}_{task})$ is the set of initial states of the external system for which there exists an action sequence such that the evolution of the external system under this action sequence results in histories that satisfy the task description. We will then call $\mathcal{R}_X(\mathcal{I}_{task})$ the *backward reachable set for $\mathcal{I}_{task}$*, analogously to the use of the same term in control theory.

Definition 6. Feasible policy for $\mathcal{I}_{task}$. Let $(\mathcal{I}, Y, \phi, \pi, U)$ and (X, U, f, h, Y) be the state-relabeled transition systems corresponding to internal and external systems, respectively. A labeling function $\pi : \mathcal{I} \to U$ is a feasible policy for $\mathcal{I}_{task}$ if for all x in the backward reachable set for $\mathcal{I}_{task}$, that is, $x \in \mathcal{R}_X(\mathcal{I}_{task})$, at least one history η_k corresponding to the coupled internal-external system (1) 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 makes the resulting π a *state-feedback policy*[6]. Note that a DITS, in other words, the robot brain, is an I-space filter itself.

Problem 3. Find a feasible policy. *Given $(\mathcal{I}, Y, \phi)$, find a labeling function $\pi \colon \mathcal{I} \to U$ that is a feasible policy for $\mathcal{I}_{task}$.*

We can further extend the planning problem to consider an unspecified internal system. This entails finding a DITS $(\mathcal{I}, Y, \phi)$ and a policy $\pi \colon \mathcal{I} \to U$ such that the resulting histories of the coupled system $(\mathcal{I} \times X, \phi *_{\pi,h} f)$ belong to ${\kappa_{task}}^{-1}(1)$, that is, they satisfy the task description. This is the problem of jointly finding an I-space-filter and a feasible policy defined over its states. Let $\mathcal{K}$ be the set of all I-maps defined over $\mathcal{I}_{hist}$. For $\kappa \in \mathcal{K}$, let Π_κ be the set of all policies (labeling functions) that can be defined over the states of $\mathcal{I}$ which is the image of the mapping $\kappa \colon \mathcal{I}_{hist} \to \mathcal{I}$.

Problem 4. Find a DITS and a feasible policy. *Find a pair $(\kappa, \pi) \in \{(\kappa, \pi) \mid \kappa \in \mathcal{K} \wedge \pi \in \Pi_\kappa\}$ such that κ is sufficient and π is a feasible policy for $\mathcal{I}_{task}$.*

Suppose $\kappa \colon \mathcal{I}_{hist} \to \mathcal{I}$ and assume (κ, π) is a solution to Problem 4. This corresponds to the DITS $(\mathcal{I}, Y, \phi_\pi)$ and a feasible policy $\pi \colon \mathcal{I} \to U$ such that $(\mathcal{I}, Y, \phi)$ is the derived ITS by κ and $(\mathcal{I}, Y, \phi_\pi)$ is the restriction of it by π.

We emphasize that finding a DITS for a planning problem differs from Problems 1 and 2 in the sense that we are not looking for a refinement of κ_{task}. The reason for this difference is because κ_{task} can already be sufficient, hence, it is the minimal sufficient refinement of itself. However, this does not necessarily imply the existence of a feasible policy defined over $\mathcal{I}_{task}$. For example, consider the κ_{task} described in Example 8 and a sensor mapping that reports whether the goal is reached or not. Even though κ_{task} is sufficient in this case, knowing when the goal is reached does not imply, in most cases, that a feasible policy exists as a labeling function for the quotient system by κ_{task}, that is, over the states of $\mathcal{I}_{task}$. On the other hand, we can still talk about a notion of minimality. This notion is defined in the following.

Definition 7. Minimal DITS for π. Let $\kappa \colon \mathcal{I}_{hist} \to \mathcal{I}$ and $\pi \colon \mathcal{I} \to U$ be a solution to Problem 4. Furthermore, let $(\mathcal{I}_{hist}, Y, \phi_{hist,\pi\circ\kappa})$ be the restriction of $(\mathcal{I}_{hist}, U \times Y, \phi_{hist})$ by $\pi \circ \kappa$. Denote by $(\mathcal{I}'_{hist}, Y, \phi_{hist,\pi\circ\kappa})$ the subgraph of $(\mathcal{I}_{hist}, Y, \phi_{hist,\pi\circ\kappa})$ from which the nodes that are not reachable from η_0 have been pruned. We restrict the domain of I-maps κ and κ' to $\mathcal{I}'_{hist}$. Then, $(\mathcal{I}, Y, \phi, \pi, U)$, determined by κ and π, is *minimal for* π if there does not exist a sufficient I-map κ' with $\kappa \succ \kappa'$ and a corresponding policy π' for the quotient system by κ' that satisfy $\pi \circ \kappa = \pi' \circ \kappa'$.

Informally, a minimal DITS for π implies that one cannot further reduce the quotient system by merging equivalence

[6] The term *state* in state-feedback policy refers to the states of the external system, as it is commonly used in the robotics literature.

classes induced by κ, while simultaneously ensuring that when coupled to the external system that is initialized at the same state, the coupling would result in the same observation and action histories as $(\mathcal{I}, Y, \phi, \pi, U)$.

There may be multiple pairs of (κ, π) that solve the same problem. Given two DITS, $(\mathcal{I}_{hist}/\kappa_1, Y, \Phi_{hist}/\kappa_1, \pi_1, U)$ and $(\mathcal{I}_{hist}/\kappa_2, Y, \Phi_{hist}/\kappa_2, \pi_2, U)$, a notion of equivalence can be determined if the preimages of $\pi_1 \circ \kappa_1$ and $\pi_2 \circ \kappa_2$ partition $\mathcal{I}_{hist}$ in the same way. We can say that $(\mathcal{I}_{hist}/\kappa_1, Y, \phi_{hist}, \pi_1, U)$ requires more histories to be distinguished if the partitioning induced by $\pi_1 \circ \kappa_1$ is a refinement over the partitioning induced by $\pi_2 \circ \kappa_2$.

Suppose a feasible policy $\pi \colon \mathcal{I}_{hist} \to U$ is defined over the states of the history ITS $(\mathcal{I}_{hist}, U \times Y, \phi_{hist})$. The restriction of the history ITS to the policy π is then $(\mathcal{I}_{hist}, Y, \phi_{hist_\pi})$ (recall the definition given in Section 3.1 of restriction of a DITS). This is a particular case that solves Problem 4 for which (κ, π) is the pair such that $\kappa \colon \mathcal{I}_{hist} \to \mathcal{I}_{hist}$ is a bijection. Let $(\mathcal{I}'_{hist}, Y, \phi_{hist_\pi})$ be the restriction by π from which the states that are not reachable from η_0 are pruned. Note that $\mathcal{I}'_{hist} \subseteq \mathcal{I}_{hist}$ is the set of histories that can be realized once the history ITS is restricted by the policy π. Restricting the domain of π to $\mathcal{I}'_{hist}$, we obtain a labeling function over the states of $(\mathcal{I}'_{hist}, Y, \phi_{hist_\pi})$ which determines the classes of histories that are distinguished by the actions selected under the policy π. To ensure that the same action histories are obtained when a derived DITS (quotient of the history ITS by κ') is coupled to the external system, the I-map κ' needs to be a refinement of π. Consequently, the following proposition establishes the connection between a policy π defined over $\mathcal{I}_{hist}$ and its respective minimal DITS.

Proposition 3. The minimal sufficient refinement of a feasible policy $\pi : \mathcal{I}_{hist} \to U$ determines its minimal DITS. *Let (κ, π) be a pair that solves Problem 4 such that $\kappa \colon \mathcal{I}_{hist} \to \mathcal{I}_{hist}$ is a bijection. Then, a minimal DITS for π is the DITS $(\mathcal{I}, Y, \phi)$ derived from $(\mathcal{I}'_{hist}, Y, \phi_{hist,\pi})$ by some minimal sufficient refinement κ' of π.*

Proof. Since κ' is a minimal sufficient refinement of π, it is sufficient and $\nexists \kappa''$ that satisfies $\kappa' \succeq \kappa'' \succeq \pi$. Since it is a refinement, every set in $\mathcal{I}'_{hist}/\kappa'$ is a subset of $\mathcal{I}'_{hist}/\pi$. Thus, we can find a $\pi' : \mathcal{I} \to U$ such that $\pi'(\kappa'(\eta)) = \pi(\eta)$. Then, by Definition 7, $(\mathcal{I}, Y, \phi)$ labelled with π' is a minimal DITS for π. □

4.3 Learning a sufficient ITS

Although learning and planning overlap significantly, some unique issues arise in pure learning (Weinstein et al. 2022). This corresponds to the case when $\kappa_{task} \colon \mathcal{I}_{hist} \to \mathcal{I}_{task}$ 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.

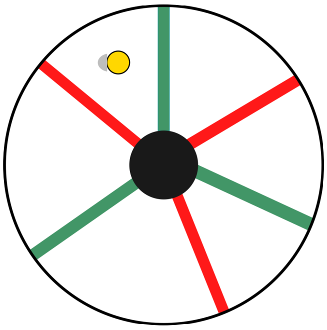


Figure 3. Environment used in Examples 9,10, the obstacle (an open disk) is shown in black.

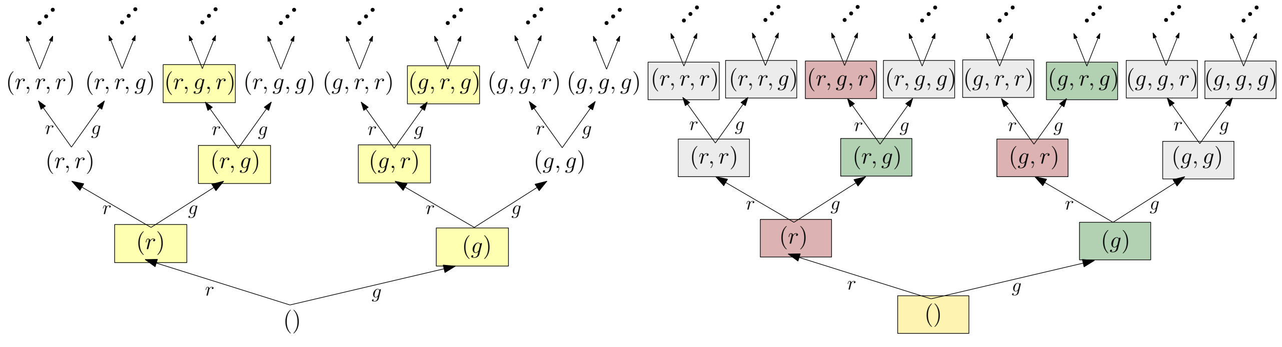
5 Applying the Theory

In this section we provide some simple filtering and planning problems and show how the ideas presented in the previous sections apply to these problems. All problems defined in the previous section can be posed in a learning context as well. Then, $\mathcal{I}_{task}$ is not given but it is revealed through interactions between the internal and external as the input-output data. Finally, we formulate as derived ITSs two established approaches, *diversity-based inference* and *predictive state representations*, for obtaining compact representations of the input-output (action-observation) relations for an unknown external system. These techniques illustrate the model-free approach to representing the internal-external coupling.

5.1 Red-green gates

This example is inspired by (Tovar et al. 2008). Let $E \subseteq \mathbb{R}^2$ be an annulus that is partitioned into non-empty regions separated by gates, see Figure 3. Each gate is either green or red. This 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 therefore $Y = \{r, g\}$. As in (Tovar et al. 2008) we assume that the robot's trajectory is in general position with respect to the gates, in the sense that it only crosses them transversally, and never goes through an intersection of two gates.

Example 9. *Consistent rotation filter*. 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



(a)

(b)

Figure 4. (a) State-relabeled history ITS described in Example 9, and the labeling function κ_{task}. States colored yellow are the ones that do not violate the task description.(b) Equivalence classes induced by κ'; the minimal sufficient refinement of κ_{task}.

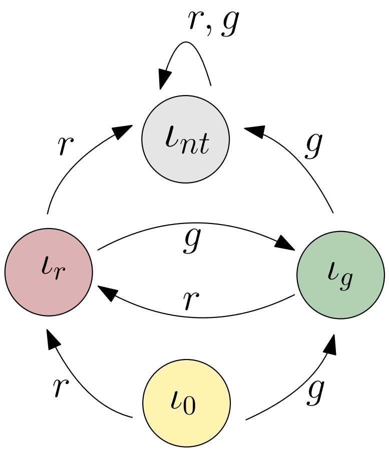


Figure 5. Quotient by κ' of the state-relabeled history ITS shown in Figure 4(b).

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. 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 (so far) and the others. The labeling induced by κ_{task} is shown in Figure 4(a).

Claim 1. *Task labeling κ_{task} defined in Example 9 is not sufficient.*

Proof. 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 $y = g$. This shows that κ_{task} violates Definition 3. □

We can obtain a sufficient refinement of κ_{task}, defined as $\kappa' : \mathcal{I}_{hist} \to \{\iota_0, \iota_r, \iota_g, \iota_{nt}\}$. The corresponding equivalence classes are shown in Figure 4(b). Its quotient DITS is shown in Figure 5.

Claim 2. *κ' as defined above is a minimal sufficient refinement of κ_{task}.*

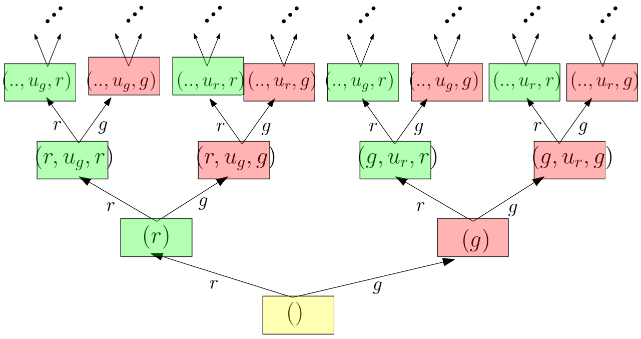


Figure 6. History ITS $(\mathcal{I}_{hist}, U \times Y, \phi_{hist})$ restricted by π, labelled with π. The histories where u_g is applied is colored green and where u_r is applied in red. The initial state η_0 is labeled with $()$.

Proof. 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 in Figure 5) 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 it is capable of executing a bouncing motion that involves *move forward* and *rotate in place*. The set of actions is defined as $U = \{u_r, u_g\}$, in which u_g represent a bouncing motion that allow the robot to traverse the green gate but not the red one, u_r allows it to traverse the red gate but not the green one. For all the actions, the robot also bounces off of the boundary. We assume that the boundary detector and color sensor readings do not arrive simultaneously, and that the resulting trajectory will strike every open interval in the boundary of every region infinitely often, with non-zero, non-tangential velocities (Bobadilla et al. 2011).

Example 10. ***Consistent rotation plan***. 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.

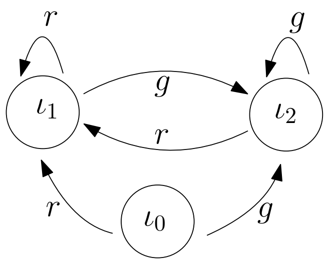


Figure 7. DITS describing the internal system solving the planning problem described in Example 10.

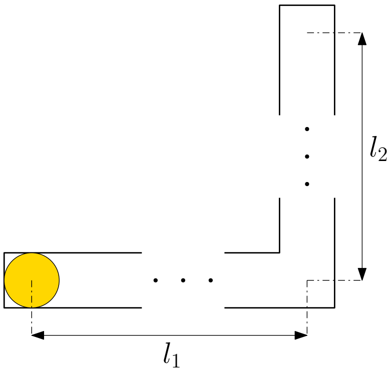


Figure 8. L-shaped corridor; $l_1, l_2 \leq l$. For any corridor, the robot starts at the left-most part of the corridor which corresponds to the coordinates $(0, 0)$.

A policy $\pi\colon \mathcal{I}_{hist} \to \mathcal{I}$ can be determined over the states of history ITS such that $\pi(\eta_0, \dots, y_k) = u_g$ if $y_k = r$ and $\pi(\eta_0, \dots, y_k) = u_r$ if $y_k = g$. Let $(\mathcal{I}'_{hist}, Y, \phi_{hist,\pi})$ be the restriction of history ITS by π such that the states that are not reachable from η_0 are pruned (see Figure 6). The labeling π defined over the states of $(\mathcal{I}'_{hist}, Y, \phi_{hist,\pi})$ is sufficient as can be seen from the inspection of the ITS given in Figure 6. Then, the following claim follows from Proposition 3.

Claim 3. *The minimal DITS for* π *is the quotient of* $(\mathcal{I}'_{hist}, Y, \phi_{hist,\pi})$ *by* π.

The quotient system, that is the minimal DITS, is shown in Figure 7. Let $\mathcal{I} = \{i_0, i_1, i_2\}$ be the states of this quotient ITS. The respective plan π' represented over the states of this minimal DITS is given as $\pi'(\iota_0) = ()$, $\pi'(\iota_1) = u_g$, $\pi'(\iota_2) = u_r$.

5.2 L-shaped corridor

Consider a robot in an inverted L-shaped planar corridor (Figure 8). 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 one which corresponds to moving one step towards right or up; if the boundary is reached, the state does not change. The robot has a sensor that reports 1 if the motion is blocked.

Example 11. ***L-shaped corridor***. Consider a model-based history ITS with $\eta_0 \subset X$ that specifies the initial position as $q_0 = (0, 0)$ which corresponds to the left-most bottom square of the mirrored L-shape (Figure 8) 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 let $\kappa_{ndet}\colon \mathcal{I}_{hist} \to \text{pow}(X)$ be an I-map that maps the history I-state η_k at stage 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 $X_{k+1} = \hat{X}(X_k, u_k) \cap H(y_{k+1})$, in which $\hat{X}(X, u) := \{f(x, u) \mid x \in X\}$, and $H(y) := h^{-1}(y) \subseteq X$ is the set of all states that could yield y. By construction, κ_{ndet} is sufficient. Suppose $\kappa_{task}\colon \mathcal{I}_{hist}/\kappa_{ndet} \to \mathcal{I}_{task}$ is a task labeling for localization that assigns each singleton a unique label and all the other subsets are labeled the same.

Claim 4. *Let* κ_{ndet} *and* κ_{task} *be the I-maps defined in Example 11. Then,* κ_{ndet} *is a minimal sufficient refinement of* κ_{task}.

Proof. Consider a subset $X' \subseteq X$ with cardinality $|X'| > 1$ and some $x' \in X$ as labels assigned by κ_{ndet}. The I-map κ_{task} is not sufficient because the transition corresponding to $([X']_{\kappa_{task}}, (u, y))$ can lead to multiple labels $[x']_{\kappa_{task}}$. By construction κ_{ndet} is sufficient. The I-map 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 to the contrary that a sufficient κ exists, which would mean that some equivalence classes could 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 $l_1 = k$, $l_2 = n$.

5.3 Diversity-based Inference (DBI) as a derived ITS

In this and the following section, we present DBI and its probabilistic counterpart PSR as deterministic ITSs. The core idea in DBI (Rivest and Schapire 1993, 1994) is to gather information about the environment through action-observation experiments. The environment is modeled as a finite state Moore machine (finite state automaton), formally defined as a 6-tuple $\mathcal{E} = (X, U, f, h, Y, x_0)$. This definition coincides with our definition of the external system, but also contains an *initial state* x_0.[7] Experiments on $\mathcal{E}$ are called *tests*. Each test $t = (\tilde{u}_m, y)$ consists of a finite action sequence $\tilde{u}_m := (u_1, \dots, u_m) \in U^m$, followed by an observation $y \in Y$. The test $t = (\tilde{u}_m, y)$ is said to *succeed* from state $x \in X$, if $\big(h \circ f^{\tilde{u}_m}\big)(x) = y$, where

$$f^{\tilde{u}_m}(x) := f\big(\cdots f\big(f(x, u_1), u_2\big) \dots, u_m\big). \tag{13}$$

By convention, if $m = 0$, then $f^{\tilde{u}_m}(x) = x$. Thus, for each test t there exists a *success function* $S_t\colon X \to \{0, 1\}$, for

[7] In (Rivest and Schapire 1993), the authors allow several different observations (predicate symbols) $p \in P$ to be observed at each state via a function $\gamma\colon X \times P \to \{0, 1\}$. We assume for simplicity that each $x \in X$ corresponds to a unique observation $y = h(x)$.

which $S_t(x) = 1$ if and only if t succeeds at x. In DBI, an equivalence relation $\sim_{\mathcal{T}}$ is defined in the set of tests $\mathcal{T} := \{(\tilde{u}_m, y) \mid \tilde{u}_m \in U^m,\ y \in Y,\ m \in \mathbb{N}\}$ by setting

$$t_1 \sim_{\mathcal{T}} t_2 \quad \Longleftrightarrow \quad S_{t_1}(x) = S_{t_2}(x), \quad \forall x \in X.$$

The cardinality $K := |P_{\mathcal{T}}|$ of the set of equivalence classes $P_{\mathcal{T}} = \{[t] \mid t \in \mathcal{T}\}$ is called the *diversity* of $\mathcal{E}$. The diversity of a finite state machine satisfies $K \le 2^{|X|} < \infty$. Each state $x \in X$ can thus be labeled by a finite *success vector*

$$\xi(x) := \big(S_1(x), \dots, S_K(x)\big) \tag{14}$$

in which, for $k \in \{1, \dots, K\}$, the functions $S_k := S_{t_k}$ are the success functions of tests $t_1, \dots, t_K$, whose respective equivalence classes $[t_1], \dots, [t_K]$ constitute the set $P_{\mathcal{T}}$.

Proposition 4. The success vector is a minimal sufficient refinement. *Let $\mathcal{E} = (X, U, f, h, Y, x_0)$ be a finite state Moore machine with diversity K. Then $\xi\colon X \to \{0,1\}^K$ defined in (14) is a minimal sufficient refinement of h.*

Proof. By setting $m = 0$ in (13), we see that ξ must be a refinement of h. Suppose $\xi(x_0) = \xi(x_1)$ for some x_0, x_1, and let $u \in U$ be arbitrary. We want to show that $\xi(f(x_0, u)) = \xi(f(x_1, u))$. Let $(\tilde{u}_m, y)$ be any test and let $\tilde{v}_{m+1} = u^\frown \tilde{u}_m$ denote the concatenation of u as a prefix to $\tilde{u}_m$. Then

$$h\big(f^{\tilde{u}_m}(f(x_0, u))\big) = h\big(f^{\tilde{v}_{m+1}}(x_0)\big) = \\ h\big(f^{\tilde{v}_{m+1}}(x_1)\big) = h\big(f^{\tilde{u}_m}(f(x_1, u)\big)$$

where the middle equality follows from the assumption that $\xi(x_0) = \xi(x_1)$. This means that all tests agree on $f(x_0, u)$ and $f(x_1, u)$, which implies that $\xi(f(x_0, u)) = \xi(f(x_1, u))$. Hence, ξ is sufficient.

Suppose ξ is not the minimal sufficient refinement of h. Let ξ' be a minimal sufficient refinement of h which always exists, see Remark 3. Thus, we have $\xi \succ \xi' \succeq h$, so there are $x_0, x_1 \in X$ with

$$\text{(i)}\ \xi'(x_0) = \xi'(x_1) \quad \text{and} \quad \text{(ii)}\ \xi(x_0) \ne \xi(x_1). \tag{15}$$

Since ξ' is sufficient, it follows from (15)(i) that $\xi'(f(x_0, u)) = \xi'(f(x_1, u))$ for all actions $u \in U$. Using the sufficiency of ξ' again m times, it follows that

$$\xi'\big(f^{\tilde{u}_m}(x_0)\big) = \xi'\big(f^{\tilde{u}_m}(x_1)\big) \tag{16}$$

for all finite action sequences $\tilde{u}_m \in U^m$. Now, since ξ' is a refinement of h, (16) implies that $h\big(f^{\tilde{u}_m}(x_0)\big) = h\big(f^{\tilde{u}_m}(x_1)\big)$ for all $\tilde{u}_m \in U^m$. By definition, this means that any test $(\tilde{u}_m, y)$ succeeds from x_0 if and only if it succeeds from x_1. This implies $\xi(x_0) = \xi(x_1)$, contradicting (15) (ii), and proves the claim. □

Now, denote the concatenation of an action $u_0 \in U$ and a test $t = (\tilde{u}_m, y) \in U^m \times Y$ by

$$u_0{}^\frown t := (u_0, u_1, \dots, u_m, y) \in U^{m+1} \times Y.$$

Since $S_t\big(f(x, u)\big) = S_{u^\frown t}(x)$ for all $t \in \mathcal{T}$ and $u \in U$, there exists for each $u \in U$ a well-defined mapping $g_u\colon P_{\mathcal{T}} \to P_{\mathcal{T}}$ given by $g_u\big([t]\big) := [u^\frown t]$. Furthermore, each g_u defines a mapping (not necessarily a permutation) $\alpha_u\colon \{1, \dots, K\} \to \{1, \dots, K\}$ by

$$\alpha_u(k) = n \quad \Longleftrightarrow \quad g_u\big([t_k]\big) = [t_n]. \tag{17}$$

Definition 8. Update graph. Let $\mathcal{E} = (X, U, f, h, Y, x_0)$ be a finite state Moore machine with diversity $K = |P_{\mathcal{T}}|$. Let $P_{\mathcal{T}} = \{[t_1], \dots, [t_K]\}$ be the set of test equivalence classes with representatives $t_1, \dots, t_K \in \mathcal{T}$, and let S_k be the success function of test t_k. Finally, let α_u be as in (17) above. The *update graph* of $\mathcal{E}$ is a state-relabeled deterministic transition system $\mathcal{G} := (S, U, \tau, \sigma, Y, s_0)$ where U and Y are as in $\mathcal{E}$, and

- $S := \{\big(S_1(x), \dots, S_K(x)\big) \in \{0,1\}^K \mid x \in X\}$,
- $\tau\big((s_1, \dots, s_K), u\big) := \big(s_{\alpha_u(1)}, \dots, s_{\alpha_u(K)}\big)$,
- $\sigma\big((s_1, \dots, s_K)\big) = h(x)$, where $x \in X$ is such that $s_k = S_k(x)$ for all $k \in \{1, \dots, K\}$, and
- $s_0 = (S_1(x_0), \dots, S_K(x_0))$.

A machine/environment $\mathcal{E}$ is said to be *reduced*, if for each state $x \in X$ there exist tests $t_1, t_2 \in \mathcal{T}$ for which $S_{t_1}(x) \ne S_{t_2}(x)$. It was shown in (Rivest and Schapire 1993, Theorem 3) that it is possible to simulate a reduced environment $\mathcal{E}$ by its update graph. We rephrase and prove this result in terms of isomorphisms of transition systems. Two Moore machines (X, U, f, h, Y, x_0) and (X', U, f', h', Y, x'_0) (both defined in terms of the same action and observation sets U and Y) are said to be *isomorphic*, if there exists a bijective map $g\colon X \to X'$ such that for all $x \in X$ and $u \in U$ we have $f'(g(x), u) = g(f(x, u))$, $g(x_0) = x'_0$, and $(h' \circ g)(x) = h(x)$. The following is essentially (Rivest and Schapire 1993, Theorem 3):

Proposition 5. Update graph representation of a Moore machine. *Let $\mathcal{E} = (X, U, f, h, Y, x_0)$ be a finite state Moore machine with a reduced state space X and let $\mathcal{G} := (S, U, \tau, \sigma, Y, s_0)$ be the update graph of $\mathcal{E}$. Then the function ξ from (14) is an isomorphism between $\mathcal{E}$ and $\mathcal{G}$.*

Proof. Let $P_{\mathcal{T}} = \{[t_1], \dots, [t_K]\}$ be the set of test equivalence classes in $\mathcal{E}$ with representatives $t_1, \dots, t_K \in \mathcal{T}$, and let S_k be the success function of test t_k. Recall the definition of ξ from (14).

Then $\xi\colon X \to S$ is onto by definition of the set S. To show injectivity, assume $\xi(x) = \xi(x')$ for some $x, x' \in X$, which means that $S_k(x) = S_k(x')$ for all $k = 1, \dots, K$. Since $\mathcal{E}$ is reduced, it suffices to show that $S_t(x) = S_t(x')$ for all tests $t \in \mathcal{T}$, because then $x = x'$. Since every test t satisfies $[t] = [t_k]$ for some k, the claim follows immediately.

We still need to show that for all $(x, u) \in X \times U$, the functions $f_u(x) := f(x, u)$ and $\tau_u(s) := \tau(s, u)$ satisfy $\big(\tau_u \circ \xi\big)(x) = \big(\xi \circ f_u\big)(x)$. According to Definition 8, each $x \in X$ and $u \in U$ satisfies $(\tau_u \circ \xi)(x) = \big(S_{\alpha_u(1)}(x), \dots, S_{\alpha_u(K)}(x)\big)$ where $\alpha_u(k) = n$ iff $[t_n] = [u^\frown t_k]$. Thus,

$$\begin{aligned}(\tau_u \circ \xi)(x) &= \big(S_{\alpha_u(1)}(x), \dots, S_{\alpha_u(K)}(x)\big) \\ &= \big(S_{u^\frown t_1}(x), \dots, S_{u^\frown t_K}(x)\big) \\ &= \big(S_1(f(x, u)), \dots, S_K(f(x, u))\big) \\ &= \xi\big(f(x, u)\big) = \big(\xi \circ f_u\big)(x).\end{aligned} \tag{18}$$

Since ξ is a bijection, the labeling function σ in Definition 8 is well defined, and satisfies $(\sigma \circ \xi)(x) = h(x)$ by definition. Finally, $\xi(x_0) = s_0$ by the definition of s_0. □

To simulate the environment $\mathcal{E}$ by $\mathcal{G}$, one needs to set the initial state $\hat{s}_0 = (s_1, \ldots, s_K) \in S$, which corresponds to the initial state $x_0 \in X$ for which $S_k(x_0) = s_k$ for all $k \in \{1, \ldots, K\}$.

If we remove the assumption that $\mathcal{E}$ is reduced, we can view the function $\xi\colon X \to S$ in (14) as a labeling that identifies those pairs of states that cannot be differentiated by any test.

Proposition 6. Update graph representation is a DITS. *Let $T_{\mathcal{E}} = (X, U, f, h, Y)$ be the transition system corresponding to the finite state Moore machine $\mathcal{E} = (X, U, f, h, Y, x_0)$, and let $\mathcal{G} := (S, U, \tau, \sigma, Y)$ be the update graph of $\mathcal{E}$. Then $\mathcal{G}$ is a DITS.*

Proof. According to Proposition 4, $\xi : X \to S$ in (14) is a sufficient labeling for $T_{\mathcal{E}}$. By definition, any two states (equivalence classes) $[x_1], [x_2]$ of the quotient system $T_{\mathcal{E}}/\xi$ satisfy $[x_1] \neq [x_2]$ if and only if $\xi(x_1) \neq \xi(x_2)$. This implies that $T_{\mathcal{E}}/\xi$ defines a reduced Moore machine which is isomorphic to $\mathcal{G}$ according to Proposition 5. The claim then follows from Remark 1, which states that quotients by sufficient labelings are deterministic transition systems. □

5.4 Predictive state representations (PSRs)

Predictive state representation (PSR) (Littman and Sutton 2001; James and Singh 2004), like its deterministic predecessor DBI, is based on the idea of performing tests on the environment. The difference is that PSR assumes a statistical description of the internal-external coupling, expressed via success probabilities of tests, conditioned on past histories. PSRs have been shown to be more general than POMDPs (Cassandra et al. 1997) in the sense that every POMDP model can be represented via the corresponding PSR.

Since the introduction of the original PSR, several variations of the concept have been proposed in connection to different learning algorithms that aim to discover the set of core tests and learn the associated prediction functions (James and Singh 2004; Boots et al. 2011, 2013). So-called TPSRs (Rosencrantz et al. 2004) are adaptations of the concept where, instead of maintaining vectors of probabilities over a finite set of core tests, a linear combination of a larger set of tests is maintained instead.

We focus on the original formulation of the PSR model and show how it can be expressed in our formalism as a DITS. Let $\eta_k := (\tilde{u}_{k-1}, \tilde{y}_k)$ denote the *history I-state* at stage k (including the kth observation, but not the kth action). In addition let $\tilde{y}_{k,m} := (y_k, \ldots, y_{k+m})$ and similarly $\tilde{u}_{k,m} := (u_k, \ldots, u_{k+m})$ for all $m \in \mathbb{N}$. In PSR, action-observation sequences $t = (\tilde{u}^t_{m-1}, \tilde{y}^t_m)$ are called *tests*. At stage $k \in \mathbb{N}$, a PSR model maintains a sufficient statistic for computing the conditional *success probabilities*

$$P_{\eta_k}(t) := P\big(\tilde{y}_{k,m} = \tilde{y}^t_m \mid \tilde{u}_{k,m-1} = \tilde{u}^t_{m-1}, \eta_k\big) \qquad (19)$$

for tests $t = \big(\tilde{u}^t_{m-1}, \tilde{y}^t_m\big)$ of arbitrary length $m \geq 1$.

The idea of PSR is to identify minimal *core sets of tests* $Q = (t_1, \ldots, t_m)$ which have the property that, given any test $t \notin Q$ and some history $\eta_k \in \mathcal{I}_{hist}$, it is possible to compute the success probability of t as $P_{\eta_k}(t) = f_t\big(Q(\eta_k)\big)$ where $Q(\eta_k) := \big(P_{\eta_k}(t_1), \ldots, P_{\eta_k}(t_m)\big)$ is the *prediction vector* for the set Q and f_t is the *prediction function* associated with t. In *linear* PSR, the space of admissible prediction functions is restricted to linear transformations (vectors) $r_t \in \mathcal{R}^{|Q|}$ so that $f_t(Q(\eta_k)) = r_t \cdot Q(\eta_k)$ for all t, η_k.

Formally, a PSR is a 5-tuple (U, Y, Q, F, m_0), where U is the set of actions, Y is the set of observations, Q is a *core set of tests*, F is the set of *prediction functions*, and $m_0 \in [0,1]^{|Q|}$ is the *initial prediction vector* after seeing the null history $\eta_0 = \varnothing$. A PSR model provides a complete (probabilistic) description of the action-observation dynamics, because the prediction vector $Q(\eta_k)$ can be updated with each new action-observation. For this, only the (finite number of) prediction functions $f_{(u,y)}$ and $f_{(u,y)\frown t_m}$ need to be known, corresponding to all possible action-observation pairs (u, y) and to concatenations of these with the core tests $t_m \in Q$. Then the update to $Q(\eta_{k+1})$, where $\eta_{k+1} = \eta_k \frown (u, y)$, is obtained through the function $\phi_{\text{PSR}}\colon [0,1]^m \times (U \times Y) \to [0,1]^m$ defined by

$$\phi_{\text{PSR}}\big(Q(\eta_k), (u,y)\big) := \Big(\phi_1\big(Q(\eta_k), (u,y)\big), \ldots, \phi_m\big(Q(\eta_k), (u,y)\big)\Big), \qquad (20)$$

where the functions $\phi_i\colon [0,1]^m \times (U \times Y) \to [0,1]$ are given for each $i \in \{1, \ldots, m\}$ by

$$\phi_i\big(Q(\eta_k), (u,y)\big) := P_{\eta_k \frown (u,y)}(t_i) = \frac{P_{\eta_k}\big((u,y)\frown t_i\big)}{P_{\eta_k}\big((u,y)\big)} = \frac{f_{(u,y)\frown t_i}\big(Q(\eta_k)\big)}{f_{(u,y)}\big(Q(\eta_k)\big)}.$$

Thus, a PSR with a core set of tests $Q = (t_1, \ldots, t_m)$ is a DITS $\big(\mathcal{I}_{\text{PSR}}, U \times Y, \phi_{\text{PSR}}\big)$, where $\mathcal{I}_{\text{PSR}} := \big\{Q(\eta_k) \mid \eta_k \in \mathcal{I}_{hist}\big\}$. The corresponding I-map $\kappa_{\text{PSR}}\colon \mathcal{I}_{hist} \to \mathcal{I}_{\text{PSR}}$ is given by $\kappa_{\text{PSR}}(\eta) := Q(\eta)$.

6 Conclusions and Future Work

This paper introduced a mathematical framework for determining minimal filters and minimal feasible policies by comparing ITSs over information spaces. The minimality results are quite general without imposing strong restrictions on the underlying dynamical system (external system). We show that a large class of problems can be posed and analyzed under this framework.

Nevertheless, there are several 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 $\pi\colon \mathcal{I} \to \Sigma$ and each $\sigma = \pi(\iota)$ produces energy in the physical world via a mapping from Σ to $\tilde{U}$.

It is also important to extend the models to continuous time. In this case, the sensing and action histories are time parameterized functions, rather than sequences. Sufficiency must be defined in terms information mappings that apply to any time slice from 0 to $t' < t$ for a history that runs from time 0 to t, rather than only over discrete time steps. Some ground work has already been done in (LaValle 2006).

Another direction is to consider the hardware and actuation models as variables, and fix other model components. This is similar to the class of problems related to co-design for which the design process of a robot given resource constraints (sensors and actuators) is sought to be automated (Shell et al. 2021; Censi 2016; Zardini et al. 2021).

In this paper, we considered the theoretical limits on the DITS necessary to express a policy defined over a history ITS. However, the problem of finding such a DITS remains as an open algorithmic challenge. Furthermore, we only considered feasible policies. An interesting direction is to analyze the information requirements for policies that are optimal with respect to a relevant objective and the trade-off between optimality and minimality. This will amount to an ordering (potentially a partial ordering) of policies in terms of (expected) cost and the minimal DITS to express such policy.

In an external-internal coupled system, the different components, I-map κ, the information transition function ϕ and the policy π share the total complexity (information content) of the internal information processing system. Of particular interest would be to explore the trade-offs between these components in terms of efficient encoding of data-structures and their successful decoding in terms of policies. Ultimately this could lead to fundamental characterisations of interaction system information content in the spirit of the minimum description length principle proposed in (Rissanen 1978).

The mathematical theory of coupling as presented in this paper is very general. Coupling in discrete dynamical systems, and of finite automata, are special cases of it, and even continuous systems can be seen as such. Connections to other work on coupling such as (Spivak 2015) is to be explored. Dynamic coupling has been proposed as a viable approach to a mathematical modelling of cognition from the enactivist perspective (Montebelli et al. 2008; Favela 2020). The existing literatutre on the latter uses bits and pieces of dynamical systems with sporadic applications in different areas of cognitive science, but a systematic unifying study is still to be seen, especially one that has meaningful ramifications to robotics and algorithmic design. An attempt to connect philosophical ideas with those of this paper was presented by the authors in (Weinstein et al. 2022).

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 (LaValle 2012; Zhang and Shell 2021). 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.

Funding

This work was supported by a European Research Council Advanced Grant (ERC AdG, ILLUSIVE: Foundations of Perception Engineering, 101020977), Academy of Finland (projects PERCEPT 322637, CHiMP 342556), and Business Finland (project HUMOR 3656/31/2019).

References

Agha-Mohammadi AA, Chakravorty S and Amato NM (2014) Firm: Sampling-based feedback motion-planning under motion uncertainty and imperfect measurements. *The International Journal of Robotics Research* 33(2): 268–304.

Bainbridge ES (1977) *The Fundamental Duality of System Theory*. Springer Netherlands, pp. 45–61.

Blum M and Kozen D (1978) On the power of the compass (or, why mazes are easier to search than graphs). In: *Proceedings Annual Symposium on Foundations of Computer Science*. pp. 132–142.

Bobadilla L, Sanchez O, Czarnowski J, Gossman K and LaValle SM (2011) Controlling wild bodies using linear temporal logic. In: *Proceedings Robotics: Science and Systems*.

Boots B, Gretton A and Gordon GJ (2013) Hilber space embeddings of predictive state representations. In: *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*. pp. 92–101.

Boots B, Siddiqi SM and Gordon GJ (2011) Closing the learning–planning loop with predictive state representations. *The International Journal of Robotics Research* 30(7): 954–966.

Brunnbauer A, Berducci L, Brandstátter A, Lechner M, Hasani R, Rus D and Grosu R (2022) Latent imagination facilitates zero-shot transfer in autonomous racing. In: *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 7513–7520.

Cassandra A, Littman ML and Zhang NL (1997) Incremental pruning: A simple, fast, exact algorithm for partially observable markov decision processes. In: *Proceedings of the Thirteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI)*.

Censi A (2016) A class of co-design problems with cyclic constraints and their solution. *IEEE Robotics and Automation Letters* 2(1): 96–103.

Dissanayake G, Newman P, Clark S, Durrant-Whyte HF and Csorba M (2001) A solution to the simultaneous localisation and map building (SLAM) problem. *IEEE Transactions on Robotics & Automation* 17(3): 229–241.

Dugundji J (1978) *Topology*. Boston: Allyn and Bacon, Inc.

Erdmann MA and Mason MT (1988) An exploration of sensorless manipulation. *IEEE Transactions on Robotics & Automation* 4(4): 369–379.

Fainekos GE, Girard A, Kress-Gazit H and Pappas GJ (2009) Temporal logic motion planning for dynamic mobile robots. *Automatica* 45(2): 343–352.

Favela LH (2020) Dynamical systems theory in cognitive science and neuroscience. *Philosophy Compass* 15(8). DOI:10.1111/phc3.12695.

Goranko V and Otto M (2007) 5 model theory of modal logic. In: Blackburn P, Van Benthem J and Wolter F (eds.) *Handbook of Modal Logic*, *Studies in Logic and Practical Reasoning*, volume 3. Elsevier, pp. 249–329. DOI:https://doi.org/10.1016/S1570-2464(07)80008-5.

Hartmanis J (1960) Symbolic analysis of a decomposition of information processing machines. *Information and Control* 3:

154–178.

Hartmanis J and Stearns RE (1964) Pair algebra and its application to automata theory. *Information and Control* 7: 485–507.

Hutto DD and Myin E (2012) *Radicalizing enactivism: Basic minds without content*. MIT Press.

James MR and Singh S (2004) Learning and discovery of predictive state representations in dynamical systems with reset. In: *Proceedings International Conference on Machine Learning*.

Kaelbling LP, Littman ML and Cassandra AR (1998) Planning and acting in partially observable stochastic domains. *Artificial Intelligence Journal* 101.

Koditschek DE (2021) 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.

Kotta Ü, Moog CH and Tõnso M (2018) Minimal realizations of nonlinear systems. *Automatica* 95: 207–212.

Kress-Gazit H, Fainekos GE and Pappas GJ (2009) Temporal-logic-based reactive mission and motion planning. *IEEE Transactions on Robotics* 25(6): 1370–1381. DOI:10.1109/TRO.2009.2030225.

Kristek SM and Shell DA (2012) Orienting deformable polygonal parts without sensors. In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 973–979.

LaValle SM (2006) *Planning Algorithms*. Cambridge, U.K.: Cambridge University Press. Also available at http://lavalle.pl/planning/.

LaValle SM (2012) *Sensing and Filtering: A Fresh Perspective Based on Preimages and Information Spaces*, *Foundations and Trends in Robotics Series*, volume 1:4. Delft, The Netherlands: Now Publishers.

Littman M and Sutton RS (2001) Predictive representations of state. *Advances in neural information processing systems* 14.

Majumdar A and Tedrake R (2017) Funnel libraries for real-time robust feedback motion planning. *The International Journal of Robotics Research* 36(8): 947–982.

Montebelli A, Herrera C and Ziemke T (2008) On cognition as dynamical coupling: An analysis of behavioral attractor dynamics. *Adaptive Behavior* 16(2-3): 182–195.

Moore EF (1956) Gedanken-experiments on sequential machines. In: Shannon C and McCarthy J (eds.) *Automata Studies*. Princeton, NJ: Princeton University Press, pp. 129–153.

Newell A and Simon HA (1972) *Human problem solving*. Prentice-Hall.

O'Kane JM and LaValle SM (2008) Comparing the power of robots. *The International Journal of Robotics Research* 27(1): 5–23.

O'Kane JM and Shell DA (2017) Concise planning and filtering: Hardness and algorithms. *IEEE Transactions on Automation Science and Engineering* 14(4): 1666–1681.

Pitt L (1989) Inductive inference, dfas, and computational complexity.

Rahmani H and O'Kane JM (2021) Equivalence notions for state-space minimization of combinatorial filters. *IEEE Transactions on Robotics* 37(6): 2117–2136. DOI:10.1109/TRO.2021.3070967.

Rissanen J (1978) Modeling by shortest data description. *Automatica* 14(5): 465–658.

Rivest RL and Schapire RE (1993) Inference of finite automata using homing sequences. *Information and Computation* 103: 299–347.

Rivest RL and Schapire RE (1994) Diversity-based inference of finite automata. *J. of the ACM* 41(3): 555–589.

Rosencrantz M, Gordon G and Thrun S (2004) Learning low dimensional predictive representations. In: *ICML*.

Ross S, Pineau J, Paquet S and Chaib-Draa B (2008) Online planning algorithms for pomdps. *Journal of Artificial Intelligence Research* 32: 663–704.

Saberifar FZ, Ghasemlou S, Shell DA and O'Kane JM (2019) Toward a language-theoretic foundation for planning and filtering. *The International Journal of Robotics Research* 38(2-3): 236–259. DOI:10.1177/0278364918801503. URL https://doi.org/10.1177/0278364918801503.

Sakcak B, Weinstein V and LaValle SM (2022) The limits of learning and planning: Minimal sufficient information transition systems. In: *2022 International Workshop on the Algorithmic Foundations of Robotics (WAFR)*.

Särkkä S (2013) *Bayesian filtering and smoothing*. 3. Cambridge university press.

Shell DA, O'Kane JM and Saberifar FZ (2021) On the design of minimal robots that can solve planning problems. *IEEE Transactions on Automation Science and Engineering* 18(3): 876–887. DOI:10.1109/TASE.2021.3050033.

Sipser M (2012) *Introduction to the Theory of Computation*. Cengage Learning.

Song Y and O'Kane JM (2012) Comparison of constrained geometric approximation strategies for planar information states. In: *2012 IEEE International Conference on Robotics and Automation*. IEEE, pp. 2135–2140.

Spivak DI (2015) The steady states of coupled dynamical systems compose according to matrix arithmetic. *arXiv preprint arXiv:1512.00802* .

Thrun S, Burgard W and Fox D (2005) *Probabilistic Robotics*. Cambridge, MA: MIT Press.

Tovar B, Cohen F and LaValle SM (2008) Sensor beams, obstacles, and possible paths. In: *Proceedings Workshop on Algorithmic Foundations of Robotics (WAFR)*.

Tovar B, Murrieta-Cid R and LaValle SM (2007) Distance-optimal navigation in an unknown environment without sensing distances. *IEEE Transactions on Robotics* 23(3): 506–518.

Vitus MP and Tomlin CJ (2011) Closed-loop belief space planning for linear, gaussian systems. In: *2011 IEEE International Conference on Robotics and Automation*. IEEE, pp. 2152–2159.

Weinstein V, Sakcak B and LaValle SM (2022) An enactivist-inspired mathematical model of cognition. *Frontiers in Neurorobotics* 16. DOI:10.3389/fnbot.2022.846982. URL https://www.frontiersin.org/articles/10.3389/fnbot.2022.846982.

Zardini G, Censi A and Frazzoli E (2021) Co-design of autonomous systems: From hardware selection to control synthesis. In: *2021 European Control Conference (ECC)*. pp. 682–689. DOI: 10.23919/ECC54610.2021.9654960.

Zhang Y and Shell DA (2020) Abstractions for computing all robotic sensors that suffice to solve a planning problem. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. pp. 8469–8475. DOI:10.1109/ICRA40945.2020.9196812.

Zhang Y and Shell DA (2021) Lattices of sensors reconsidered when less information is preferred. In: *IEEE International Conference on Robotics and Automation*.

Zhen W, Zeng S and Soberer S (2017) Robust localization and localizability estimation with a rotating laser scanner. In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. pp. 6240–6245. DOI:10.1109/ICRA.2017.7989739.

Zhu H and Alonso-Mora J (2019) Chance-constrained collision avoidance for mavs in dynamic environments. *IEEE Robotics and Automation Letters* 4(2): 776–783. DOI:10.1109/LRA.2019.2893494.