



OPEN ACCESS

EDITED BY

Eckehard Schöll,
Technical University of Berlin, Germany

REVIEWED BY

Viktor Jirsa,
Aix-Marseille Université, France
Klaus Obermayer,
Technical University of Berlin, Germany

*CORRESPONDENCE

Karl Friston,
✉ k.friston@ucl.ac.uk

RECEIVED 03 November 2024

ACCEPTED 02 April 2025

PUBLISHED 18 June 2025

CITATION

Friston K, Heins C, Verbelen T, Da Costa L,
Salvatori T, Markovic D, Tschantz A, Koudahl M,
Buckley C and Parr T (2025) From pixels to
planning: scale-free active inference.
Front. Netw. Physiol. 5:1521963.
doi: 10.3389/fnetp.2025.1521963

COPYRIGHT

© 2025 Friston, Heins, Verbelen, Da Costa,
Salvatori, Markovic, Tschantz, Koudahl, Buckley
and Parr. This is an open-access article
distributed under the terms of the [Creative
Commons Attribution License \(CC BY\)](#). The use,
distribution or reproduction in other forums is
permitted, provided the original author(s) and
the copyright owner(s) are credited and that the
original publication in this journal is cited, in
accordance with accepted academic practice.
No use, distribution or reproduction is
permitted which does not comply with these
terms.

From pixels to planning: scale-free active inference

Karl Friston^{1,2*}, Conor Heins², Tim Verbelen²,
Lancelot Da Costa^{1,2}, Tommaso Salvatori², Dimitrije Markovic^{2,3},
Alexander Tschantz², Magnus Koudahl², Christopher Buckley^{2,4}
and Thomas Parr⁵

¹Queen Square Institute of Neurology, University College London, London, United Kingdom, ²VERSES
Research Lab, Los Angeles, CA, United States, ³Cognitive Computational Neuroscience, Technische
Universität, Dresden, Germany, ⁴Department of Informatics, University of Sussex, Brighton, United Kingdom,
⁵Nuffield Department of Clinical Neurosciences, University of Oxford, Oxford, United Kingdom

This paper describes a discrete state-space model and accompanying methods for generative modeling. This model generalizes partially observed Markov decision processes to include paths as latent variables, rendering it suitable for active inference and learning in a dynamic setting. Specifically, we consider deep or hierarchical forms using the renormalization group. The ensuing *renormalizing generative models* (RGM) can be regarded as discrete homologs of deep convolutional neural networks or continuous state-space models in generalized coordinates of motion. By construction, these scale-invariant models can be used to learn compositionality over space and time, furnishing models of paths or orbits: that is, events of increasing temporal depth and itinerancy. This technical note illustrates the automatic discovery, learning, and deployment of RGMs using a series of applications. We start with image classification and then consider the compression and generation of movies and music. Finally, we apply the same variational principles to the learning of Atari-like games.

KEYWORDS

active inference, active learning, Bayesian model selection, renormalization group, compression, structure learning, network-physiology

Introduction

This paper considers the use of discrete state-space models as generative models for classification, compression, generation, prediction, and planning. The inversion of such models can be read as inferring the latent causes of observable outcomes or content. When endowed with the consequences of action, they can be used for planning as inference (Attias, 2003; Botvinick and Toussaint, 2012; Da Costa et al., 2020). This allows one to cast any classification, prediction, or planning problem as an inference problem that, under active inference, reduces to maximizing model evidence. However, applications of active inference have been largely limited to small-scale problems. In this paper, we consider one solution to the implicit scaling problem, namely, the use of scale-free generative models and the renormalization group (Cardy, 2015; Cugliandolo and Lecomte, 2017; Hu et al., 2020; Lin et al., 2017; Schwabl, 2002; Vidal, 2007; Watson et al., 2022). The contribution of this paper is to consider *discrete* models that are renormalizable in state space *and time*.

Specifically, this paper explores the use of generalized Markov decision processes as discrete models appropriate for compressing data, generating content, or planning. The generalization in question rests on equipping a standard (partially observed) Markov decision process with random variables called *paths*. This affords an expressive model of dynamics, in which transitions among

states are conditioned on paths, which themselves can have lawful transitions. This generalization becomes particularly important when composing Markovian processes in a deep or hierarchical architecture; for example, Friston et al. (2017b). This follows from the use of states at one level to generate the initial conditions and paths at a lower level. In effect, this means that states generate paths, which generate states, which generate paths, and so on, furnishing trajectories with deep, semi-Markovian structure; for example, Marković et al. (2022). Their recursive aspect speaks to a definitive feature of the generative models considered in this paper—renormalizability.

Intuitively, renormalizability rests on a renormalization group (RG) operator, which takes a description of the system at hand (e.g., an action and partition function) and returns a coarse-grained version that retains the properties of interest while discarding irrelevant details (Cardy, 2015; Schwabl, 2002; Watson et al., 2022). For excellent overviews of the renormalization group in machine learning, please see Hu et al. (2020) and Mehta and Schwab (2014). See also Lin et al. (2017), who appeal to the renormalization group to formalize the claim that “when the statistical process generating the data is of a certain hierarchical form prevalent in physics and machine learning, a deep neural network can be more efficient than a shallow one.” Crucially, any random dynamical system with sparse coupling and an implicit Markov blanket partition (Sakthivadivel, 2022) is renormalizable (Friston, 2019). Therefore, any generative model that recapitulates “the statistical process generating the data” must be renormalizable.

In what follows, we illustrate applications of the same *renormalizing generative model* (RGM) in several settings. The universality of this model calls on the apparatus of the renormalization group. In brief, its deep structure ensures that each level can be renormalized to furnish the level above. The renormalization group requires that the functional form of the dynamics (e.g., belief updating) is conserved over levels or scales. This is assured in variational inference, in the sense that the inference process itself can be cast as pursuing a path of least action (Friston K. et al., 2023), where action is the path integral of variational free energy: c.f., Mehta and Schwab (2014). The only things that change between levels are the parameters of the requisite action (e.g., sufficient statistics of various probability distributions). The relationship between the parameters at one level and the next rests on an RG operator that entails a grouping and dimension reduction, that is, coarse-graining or scaling transformation. By choosing the right kind of RG operator, one can effectively dissolve the scaling problem. In short, by ensuring each successive level of a deep generative model is renormalizable, one can, in principle, generate data at any scale and implicitly infer or learn the causes of those data. The notion of scale invariance is closely related to universality, licensing the notion of a universal generative model.

Instances of the renormalization group abound in natural and machine learning: for example, the cortical visual hierarchy in the brain, with progressive enlargement of spatiotemporal receptive fields as one ascends hierarchical levels or scales (Angelucci and Bullier, 2003; Hasson et al., 2008; Zeki and Shipp, 1988). The same kind of architectures associated with deep convolutional neural networks could almost be definitive of deep models and learning (Hu et al., 2020; Lin et al., 2017). Here, we pay special attention to the implications of universality and scale invariance for specifying the structural form of generative models and illustrate the ensuing efficiency when deployed in some typical use cases.

This paper comprises four sections. The first rehearses the variational procedures or methods used in active inference, learning, and selection, with a special focus on the selection of hierarchical model

structures that can be renormalized. In brief, active *inference*, *learning*, and *selection* speak to the distinct sets of unknown variables that constitute a generative model, namely, latent *states*, *parameters*, and *structure*. On this view, model inversion corresponds to (Bayesian) belief updating at each of these levels by minimizing *variational free energy*, that is, maximizing an evidence lower bound (Winn and Bishop, 2005). The active part of inference, learning, and selection arises operationally through selecting or choosing those actions that minimize *expected free energy*, which can be decomposed in a number of ways that subsume commonly used objective functions in statistics and machine learning (Da Costa et al., 2020). The dénouement of this section considers the structural form of renormalizing architectures, illustrated by successive elaborations in the remaining sections.

The second section starts with a simple application to models of static images that can be read as a form of image compression; that is, maximizing model evidence via minimizing model complexity through the compression afforded by successive block-spin transformations (Vidal, 2007). The implicit sample efficiency is showcased by application to the MNIST digit classification problem. This application foregrounds the representational nature of the generative model, moving from a *place-coded* representation at lower levels to an *object-centered* representation (i.e., digit classes) at the highest level. This differs significantly from approaches to generative modeling in which images depend on several objects that can be placed in different parts of a scene (Henniges, Turner et al., 2014)—in other words, that factorize the highest level into “what” and “where” things are. This is relevant to the way in which occlusions are handled. The renormalized models we propose deal with occlusions in terms of local patterns, and then of local patterns of local patterns, and so on. This means one could generate images of one object partially occluding another by exploiting the statistics of pixels at and around the occluding edge of the proximal object. The benefit of the approach we outline here is that one does not need to know, or even discover (Blei, Ng et al., 2003), the number of objects in a scene to be able to generate it. The downside is that one could not query, from the final model, what would happen were one to change the ordering of objects in the scene such that a previously occluded object becomes the occluder. The next section uses the same methods to illustrate renormalization over time, that is, modeling paths or sequences of increasing temporal depth at successively higher levels. This application can be regarded as a form of video compression illustrated using short movie files that can be used to recognize sequences of events or generate sequences in response to a prompt. In contrast to the object-centered representations afforded by application to static images, this section speaks to *event-based* compression suitable for classifying or generating visual or auditory scenes. The next section leverages the ability to classify or generate sequences by applying the same methods to sound files, illustrated using birdsong and music. The final section turns to planning and agency by using an RGM to learn and play Atari-like games. This application involves equipping the generative model with the capacity to act, namely, to realize the predicted consequences of action, where these predictions are based on a fast form of structure learning, effectively evincing a one-shot learning of expert play.

Although the focus on renormalization inherits from the physics of universal phenomena (Haken, 1983; Haken, 1996; Haken and Portugali, 2016; Schwabl, 2002; Vidal, 2007), we highlight the biomimetic aspects of inference and learning that emerge under these models. The implication here is that natural intelligence may have evolved renormalizing structures simply because the world features universal

phenomena, such as scale invariance. This is not a machine learning paper because the objective in active inference is to maximize model evidence. Therefore, we refrain from benchmarking any of the examples in terms of performance or accuracy. However, it should be self-evident that the methods on offer are generally more sample efficient than extant machine learning schemes¹.

Active inference, learning, and selection

This section overviews the model used in the numerical studies of subsequent sections. This model generalizes a partially observed Markov decision process (POMDP) by equipping it with random variables called *paths* that “pick out” dynamics or transitions among latent states. These models are designed to be composed hierarchically in a way that introduces a separation of temporal scales.

Generative models

Active inference rests on a *generative model* of observable outcomes. This model is used to infer the most likely causes of outcomes in terms of expected states of the world. These states (and paths) are latent or *hidden* because they can only be inferred through observations. Some paths are controllable, in that they can be realized through action. Therefore, certain observations depend on action (e.g., where one is looking), which requires the generative model to entertain expectations about outcomes under different combinations of actions (i.e., policies)². These expectations are optimized by minimizing *variational free energy*. Crucially, the prior probability of a policy depends on its *expected free energy*. Having evaluated the expected free energy of each policy, the most likely action is selected, and the implicit perception-action cycle continues (Parr et al., 2022).

Figure 1 provides an overview of the generative model considered in this paper. It is structured such that it highlights three general motifs in the factorization of a probabilistic generative model. These are shown as three panels arranged in a single row, each with a title underneath indicating the type of factorization. The three panels use a graphical formulation for the models, which are supplemented with mathematical descriptions of their structures. Outcomes at any time depend on hidden *states*, while transitions among hidden states depend on *paths*. Note that paths are random variables that may or may not depend on action. The resulting POMDP is specified by a set of tensors. The first set **A** maps from hidden states to outcome modalities, for example, exteroceptive (e.g., visual) or proprioceptive (e.g., eye position) *modalities*. These parameters encode the likelihood of an outcome given their hidden causes. The second set **B** encodes transitions among the hidden states of a *factor* under a particular path. Factors correspond to different kinds of

causes, such as the location *versus* the class of an object. The remaining tensors encode prior beliefs about paths **C** and initial conditions **D** and **E**. The tensors are generally parameterized as Dirichlet distributions, whose sufficient statistics are concentration parameters or *Dirichlet counts*, which count the number of times a particular combination of states and outcomes has been inferred. We will focus on learning the likelihood model, encoded by Dirichlet counts, **a**.

The generative model in Figure 1 means that outcomes are generated as follows: first, a policy is selected using a softmax function of expected free energy. Sequences of hidden states are generated using the probability transitions specified by the selected combination of paths (i.e., policy). Finally, these hidden states generate outcomes in one or more modalities. *Inference* about hidden states (i.e., state estimation) corresponds to inverting a generative model, given a sequence of outcomes, while *learning* corresponds to updating model parameters. The requisite expectations constitute the sufficient statistics (**s, u, a**) of approximate posterior beliefs $Q(s, u, a) = Q_s(s)Q_u(u)Q_a(a)$. The implicit factorization of this approximate posterior effectively partitions model inversion into inference, planning, and learning.

Variational free energy and inference

In variational Bayesian inference (a.k.a. approximate Bayesian inference), model inversion entails the minimization of variational free energy with respect to the sufficient statistics of approximate posterior beliefs. This can be expressed as follows, where, for clarity, we will deal with a single factor, such that the policy (i.e., a combination of paths) becomes the path, $\pi = u$. Omitting dependencies on previous states, we have for model *m*:

$$\begin{aligned} Q(s_t, u_t, a) &= \operatorname{argmin}_Q F \\ F &= \mathbb{E}_Q [\underbrace{\ln Q(s_t, u_t, a)}_{\text{posterior}} - \underbrace{\ln P(o_t | s_t, u_t, a)}_{\text{likelihood}} - \underbrace{\ln P(s_t, u_t, a)}_{\text{prior}}] \\ &= \underbrace{D_{KL}[Q(s_t, u_t, a) \| P(s_t, u_t, a | o_t)]}_{\text{divergence}} - \underbrace{\ln P(o_t)}_{\text{evidence}} \\ &= \underbrace{D_{KL}[Q(s_t, u_t, a) \| P(s_t, u_t, a)]}_{\text{complexity}} - \underbrace{\mathbb{E}_Q [\ln P(o_t | s_t, u_t, a)]}_{\text{accuracy}} \end{aligned} \quad (1)$$

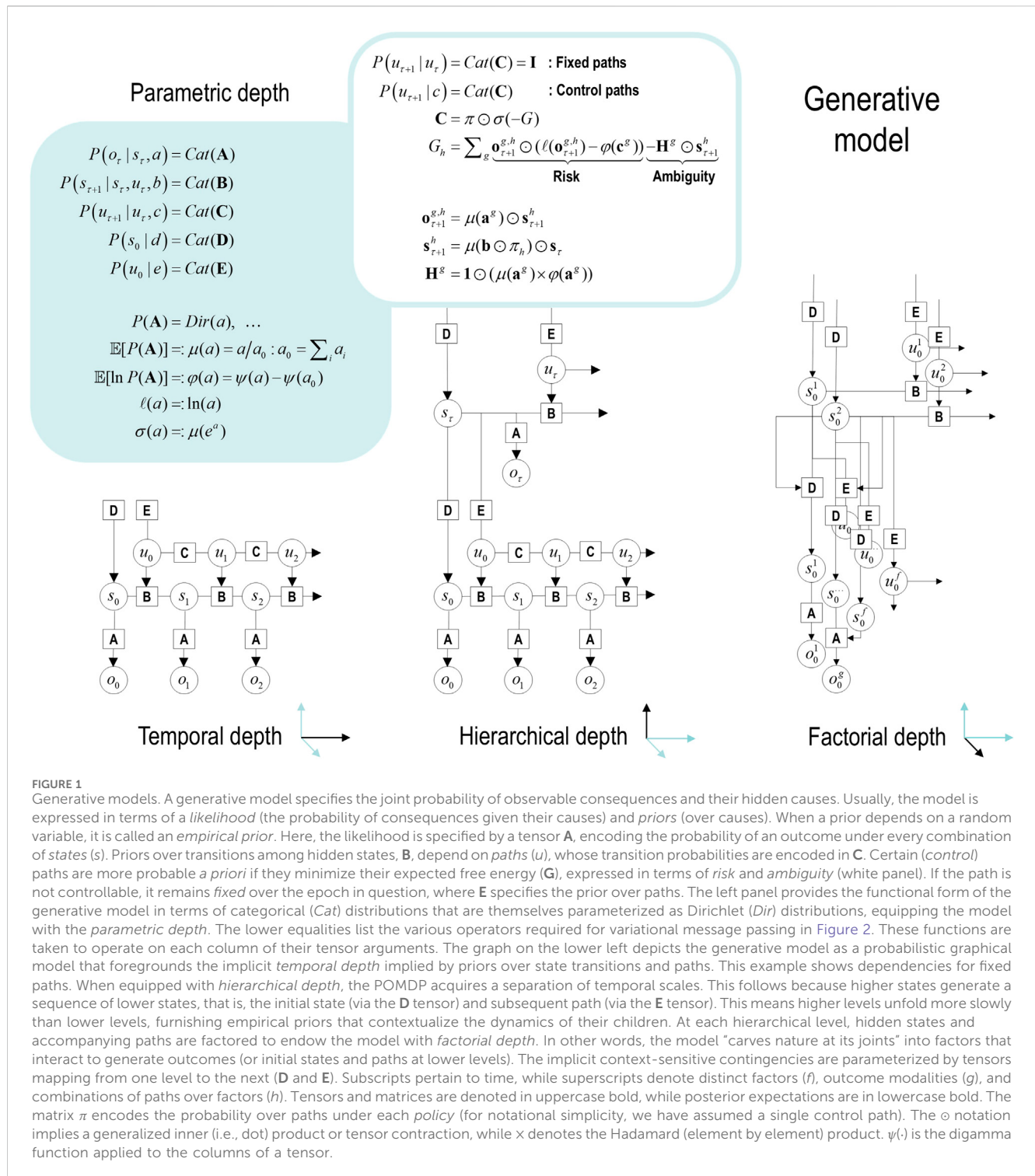
Because the (KL) divergences cannot be less than 0, the penultimate equality means that free energy is 0 when the (approximate) posterior is the true posterior. At this point, the free energy becomes the negative log evidence for the generative model (Beal, 2003). This means minimizing free energy is equivalent to maximizing model evidence.

Planning emerges under active inference by placing priors over (controllable) paths to minimize expected free energy (Friston et al., 2015):

$$\begin{aligned} G(u) &= \mathbb{E}_{Q_u} [\ln Q(s_{t+1}, a | u) - \ln Q(s_{t+1}, a | o_{t+1}, u) - \ln P(o_{t+1} | c)] \\ &= - \underbrace{\mathbb{E}_{Q_u} [\ln Q(a | s_{t+1}, o_{t+1}, u) - \ln Q(a | s_{t+1}, u)]}_{\text{expected information gain (learning)}} \\ &\quad - \underbrace{\mathbb{E}_{Q_u} [\ln Q(s_{t+1} | o_{t+1}, u) - \ln Q(s_{t+1} | u)]}_{\text{expected information gain (inference)}} - \underbrace{\mathbb{E}_{Q_u} [\ln P(o_{t+1} | c)]}_{\text{expected cost}} \\ &= - \underbrace{\mathbb{E}_{Q_u} [D_{KL}[Q(a | s_{t+1}, o_{t+1}, u) \| Q(a | s_{t+1}, u)]]}_{\text{novelty}} \\ &\quad + \underbrace{D_{KL}[Q(o_{t+1} | u) \| P(o_{t+1} | c)]}_{\text{risk}} - \underbrace{\mathbb{E}_{Q_u} [\ln Q(o_{t+1} | s_{t+1}, u)]}_{\text{ambiguity}} \end{aligned} \quad (2)$$

¹ To avoid overburdening the main text, some details of the numerical studies have been placed in figure legends.

² In this setting, a policy is not a sequence of actions but a combination of paths, where each hidden factor has an associated state and path. This means there are, potentially, as many policies as there are combinations of paths.



Here, $Q_u = Q(o_{\tau+1}, s_{\tau+1}, a | u) = P(o_{\tau+1}, s_{\tau+1}, a | u, o_0, \dots, o_\tau) = P(o_{\tau+1} | s_{\tau+1}, a) Q(s_{\tau+1}, a | u)$ is the posterior predictive distribution over parameters, hidden states, and outcomes at the next time step, under a particular path. Note that the expectation is over *observations in the future* that become random variables, hence, *expected* free energy. This means

that preferred outcomes that subtend expected cost and risk are prior beliefs that constrain the implicit planning as inference (Attias, 2003; Botvinick and Toussaint, 2012; Van Dijk and Polani, 2013).

One can also express the prior over the parameters in terms of an expected free energy, where marginalizing over paths:

Message passing on factor graphs

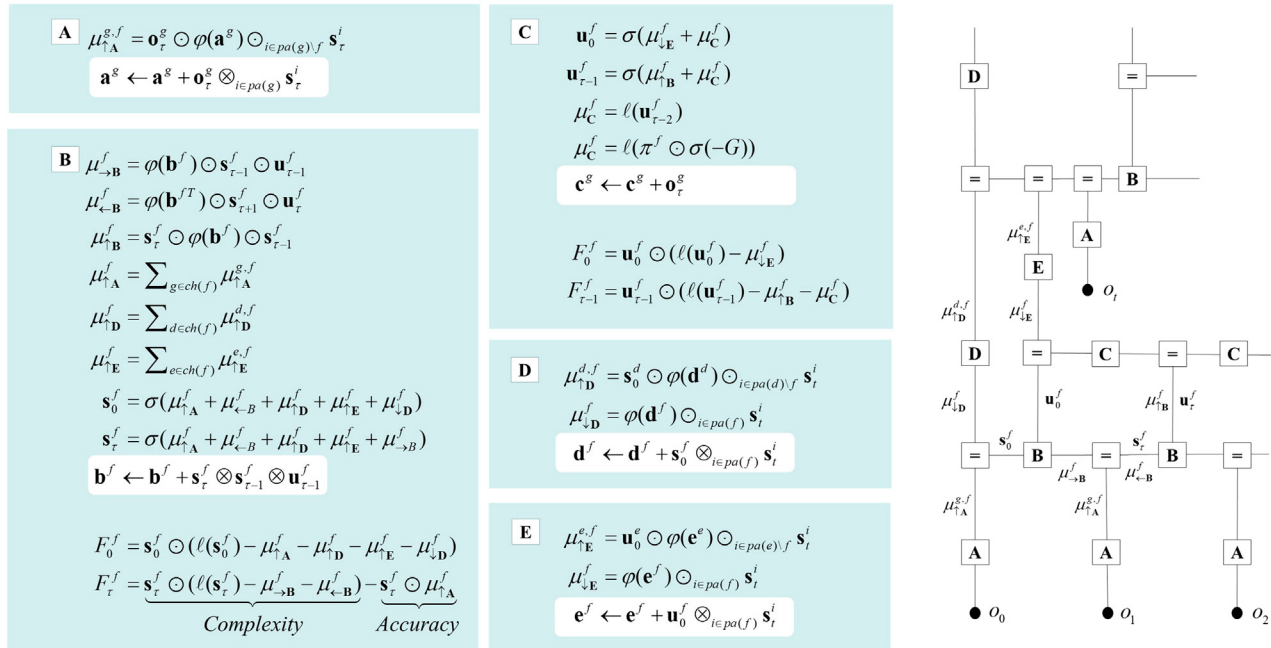


FIGURE 2 Belief updating and variational message passing: the right panel presents the generative model as a factor graph, where the nodes (square boxes) correspond to the factors of the generative model (labeled with the associated tensors). The edges connect factors that share dependencies on random variables. The leaves of filled circles correspond to known variables, such as observations ($\mathbf{o}$). This representation is useful because it scaffolds the message passing—over the edges of the factor graph—that underwrites inference and planning. The functional forms of these messages are shown in the left-hand panels, where the panels labels (A–E) indicate the corresponding tensors in the factor graph on the right. For example, the expected path in the first equality of panel (C) is a softmax function of two messages. The first is a descending message $\mu_{\downarrow E}^f$ from (E) that inherits from expectations about hidden states at the level above. The second is the log-likelihood of the path based on expected free energy, G . This message depends on Dirichlet counts scoring preferred outcomes—that is, prior constraints on modality g —encoded in $\mathbf{c}^g$: see Figure 1 and Equation 2. The two expressions for $\mu_{\downarrow C}^f$ correspond to fixed and control paths, respectively. The updates in the lighter panels correspond to learning, that is, updating Bayesian beliefs about parameters. Similar functional forms for the remaining messages can be derived by direct calculation. The $\odot$ notation implies a generalized inner product or tensor contraction, while $\otimes$ denotes an outer product. $ch(\cdot)$ and $pa(\cdot)$ return the children and parents of latent variables.

$$P(a) = \sigma(-G(a))$$

$$\begin{aligned} G(a) &= \mathbb{E}_{Q_a} [\ln P(s|a) - \ln P(s | \mathbf{o}, a) - \ln P(\mathbf{o}|c)] \\ &= -\underbrace{\mathbb{E}_{Q_a} [\ln P(s|\mathbf{o}, a) - \ln P(s|a)]}_{\text{expected information gain}} - \underbrace{\mathbb{E}_{Q_a} [\ln P(\mathbf{o}|c)]}_{\text{expected cost}}, \\ &= -\underbrace{\mathbb{E}_{Q_a} [D_{KL}[P(\mathbf{o}, s|a) \| P(\mathbf{o}|a)P(s|a)]]}_{\text{mutual information}} - \underbrace{\mathbb{E}_{Q_a} [\ln P(\mathbf{o}|c)]}_{\text{expected cost}} \end{aligned} \quad (3)$$

where $Q_a = P(\mathbf{o}|s, a)P(s|a) = P(\mathbf{o}, s|a)$ is the joint distribution over outcomes and hidden states, encoded by Dirichlet parameters, and σ is the softmax (normalized exponential) function. Here and throughout, we leave the functional dependence of G on c implicit for simplicity of notation. This is only a slight simplification, in that we will treat c as if it is a fixed value and not something we expect to vary. Note that the Dirichlet parameters encode the mutual information, in the sense that they implicitly encode the joint distribution over outcomes and their hidden causes. When normalizing each column of the a tensor, we recover the likelihood distribution (as in Figure 1). However, we could normalize over every element to recover a joint distribution [as in Equation 5 later].

Expected free energy can be regarded as a universal objective function that augments mutual information with expected costs or constraints. Constraints parameterized by c reflect the fact that we are dealing with open systems with characteristic outcomes, $\mathbf{o}$. This can be read as an expression of the constrained maximum entropy principle (Ramstead et al., 2022). Alternatively, it can be read as a constrained principle of maximum mutual information or minimum redundancy (Ay et al., 2008; Barlow, 1961; Linsker, 1990; Olshausen and Field, 1996). In machine learning, this kind of objective function underwrites disentanglement (Higgins et al., 2021; Sanchez et al., 2019) and generally leads to sparse representations (Gros, 2009; Olshausen and Field, 1996; Sakthivadivel, 2022; Tipping, 2001).

There are many special cases of minimizing expected free energy. For example, maximizing expected information gain maximizes (expected) Bayesian surprise (Itti and Baldi, 2009), in accordance with the principles of optimal experimental design (Lindley, 1956). This resolution of uncertainty is related to artificial curiosity (Schmidhuber, 1991; Still and Precup, 2012) and speaks to the value of information (Howard, 1966). Expected complexity or risk is the same quantity minimized in risk-sensitive or KL control (Klyubin et al., 2005; van den Broek et al., 2010) and underpins (free energy)

formulations of bounded rationality based on complexity costs (Braun et al., 2011; Ortega and Braun, 2013) and related schemes in machine learning; for example, Bayesian reinforcement learning (Ghavamzadeh et al., 2016). Finally, minimizing expected cost subsumes Bayesian decision theory (Berger, 2011).

Active inference

In variational inference and learning, sufficient statistics encoding posterior expectations are updated to minimize variational free energy. Figure 2 illustrates these updates in the form of variational message passing (Dauwels, 2007; Friston et al., 2017a; Winn and Bishop, 2005). For example, expectations about hidden states are a softmax function of messages that are linear combinations of other expectations and observations.

$$\begin{aligned} \mathbf{s}_\tau^f &= \sigma(\mu_{\mathbf{A}}^f + \mu_{\mathbf{B}}^f + \mu_{\mathbf{C}}^f + \dots) \\ \mu_{\mathbf{A}}^f &= \sum_{g \in \text{ch}(f)} \mu_{\mathbf{A}}^{g,f} \\ \mu_{\mathbf{A}}^{g,f} &= \mathbf{o}_\tau^g \odot \varphi(\mathbf{a}^g) \odot_{i \in \text{pa}(g)} \mathbf{s}_\tau^i \end{aligned} \quad (4)$$

Here, the ascending messages from the likelihood factor are a linear mixture³ of expected states and observations, weighted by (digamma) functions of the Dirichlet counts that correspond to the parameters of the likelihood model (c.f., connection weights). The definitions of the terms that appear in Equation 4 are summarized in Figure 2 and its caption. The expressions in Figure 2 are effectively the fixed points (i.e., minima) of variational free energy. This means that message passing corresponds to a fixed-point iteration scheme that inherits the same convergence proofs of coordinate descent (Beal, 2003; Dauwels, 2007; Winn and Bishop, 2005)⁴.

Active learning

Active learning has a specific meaning in this paper. It implies that the updating of Dirichlet counts depends on expected free energy, namely, the mutual information encoded by the tensors: see Equation 1. This means that an update is selected in proportion to the expected information gain. Consider two actions: to update or not to update. From Figure 2, we have (dropping the modality superscript for clarity):

$$\Delta \mathbf{a} = \mathbf{o}_\tau \otimes_{i \in \text{pa}} \mathbf{s}_\tau^i$$

$$\mathbb{E}[\mathcal{P}(o, s|a)] = \bar{a} = : \frac{\mathbf{a}}{\Sigma(\mathbf{a})} \quad (5)$$

$$\mathbb{E}_Q[a|u = u_o] = \mathbf{a}|u_o = \mathbf{a}$$

$$\mathbb{E}_Q[a|u = u_1] = \mathbf{a}|u_1 = \mathbf{a} + \Delta \mathbf{a}$$

Here, $\Sigma(\mathbf{a}) =: 1 \odot \mathbf{a} \odot_{i \in \text{pa}} 1$ is the sum of all tensor elements. The expression on the right-hand side of the first line of Equation 5 indicates the outer product between beliefs about an outcome and the state factor i that is the parent factor of that outcome. The prior probability of committing to an update is given by the expected free energy of the respective Dirichlet parameters, which scores the expected information gain (i.e., mutual information) and cost⁵:

$$\begin{aligned} P(u) &= \sigma(-\alpha \cdot G(\mathbf{a}|u)) \\ G(\mathbf{a}) &= -\mathbb{E}_{Q_a} [D_{KL}[Q(s, o|\mathbf{a}) \| Q(s|\mathbf{a})P(o|\mathbf{a})]] - \mathbb{E}_{Q_a} [\ln P(o|c)] \\ &= \underbrace{(\mathbf{1} \odot \bar{\mathbf{a}}) \odot \ell(\mathbf{1} \odot \bar{\mathbf{a}}) + (\bar{\mathbf{a}} \odot \mathbf{1}) \odot \ell(\bar{\mathbf{a}} \odot \mathbf{1})}_{\text{mutual information}} - \underbrace{\mathbf{1} \odot (\bar{\mathbf{a}} \times \ell(\bar{\mathbf{a}})) \odot \mathbf{1} - \varphi(c) \odot (\bar{\mathbf{a}} \odot \mathbf{1})}_{\text{expected cost}} \end{aligned} \quad (6)$$

This prior over the updates furnishes a Bayesian model average of the likelihood parameters, effectively marginalizing over update policies:

$$\begin{aligned} \mathbb{E}_Q[a_{\tau+1}^g] &= P(u_o) \mathbb{E}_Q[a_\tau^g | u = u_o] + P(u_1) \mathbb{E}_Q[a_\tau^g | u = u_1] \\ &\Rightarrow \\ \mathbf{a}_{\tau+1}^g &= P(u_o) \mathbf{a}_\tau^g + P(u_1) (\mathbf{a}_\tau^g + \Delta \mathbf{a}_\tau^g) \\ &= \mathbf{a}_\tau^g + P(u_1) \Delta \mathbf{a}_\tau^g \end{aligned} \quad (7)$$

In Equation 6, α plays the role of a hyperprior that determines the sensitivity to expected free energy. When this precision parameter is large, the Bayesian model average above becomes Bayesian model selection; that is, either the update is selected, or it is not.

This kind of active learning rests on treating an update as an action that is licensed if expected free energy decreases. A complementary perspective on this selective updating is that it instantiates a kind of Maxwell's demon, selecting only those updates that maximize (constrained) mutual information. Exactly the same idea can be applied to model selection, leading to active selection.

Active selection

In contrast to learning that optimizes *posteriors* over parameters, Bayesian model selection or structure learning (Tenenbaum et al., 2011; Tervo et al., 2016; Tomasello, 2016) can be framed as optimizing the *priors* over model parameters. In this view, model selection can be implemented efficiently using Bayesian model reduction, which starts with an expressive model and removes redundant parameters. Crucially, Bayesian model reduction can be applied to posterior beliefs after the data have been assimilated. In other words, Bayesian model reduction is a *post hoc* optimization that refines current beliefs

³ The notation implies a sum-product operator, that is, the dot or inner product that sums over one dimension of a numeric array or tensor. In this paper, these sum-product operators are applied to a vector $\mathbf{a}$ and a tensor $\mathbf{A}$, where $\mathbf{a} \odot \mathbf{A}$ implies the sum of products is taken over the leading dimension, while $\mathbf{A} \odot \mathbf{a}$ implies the sum is taken over a trailing dimension. For example, $\mathbf{1} \odot \mathbf{A}$ is the sum over rows, and $\mathbf{A} \odot \mathbf{1}$ is the sum over columns, where $\mathbf{1}$ is a vector of ones, and $\mathbf{A}$ is a matrix. This notation replaces the Einstein summation notation to avoid visual clutter.

⁴ Figure 2 and Equation 4 provide both the forward and backward messages in time, characteristic of forward-backward (i.e., Bayesian smoothing) schemes. In practice—and in the examples below—it is sometimes simpler to omit backward messages, in which case the forward messages can be replaced with exact updates.

⁵ For simplicity, we assume tensors have been formatted as matrices by vectorizing the second and trailing dimensions.

based on alternative models that may provide potentially simpler explanations (Friston and Penny, 2011).

Bayesian model reduction is a generalization of ubiquitous procedures in statistics (Savage, 1954). In the present setting, it reduces to something remarkably simple: by applying Bayes rules to parent and reduced models, it is straightforward to show that the change in variational free energy can be expressed in terms of posterior Dirichlet counts $\mathbf{a}$, prior counts $\mathbf{a}'$, and the prior counts that define a reduced model $\mathbf{a}'$. Using B to denote the beta function, we have (Friston et al., 2018):

$$\begin{aligned}\Delta F &= \ln P(o|\mathbf{a}) - \ln P(o|\mathbf{a}') \\ &= \ln B(\mathbf{a}) + \ln B(\mathbf{a}') - \ln B(\mathbf{a} + \mathbf{a}' - \mathbf{a}). \quad (8) \\ \mathbf{a}' &= \mathbf{a} + \mathbf{a}' - \mathbf{a}\end{aligned}$$

In Equation 8, $\mathbf{a}'$ corresponds to the posterior that one would have obtained under the reduced priors. Please see Friston et al. (2020) and Smith et al. (2020) for worked examples in epidemiology and neuroscience, respectively.

The alternative to Bayesian model reduction is the bottom-up growth of models to accommodate new data or content. If one considers the selection of one (parent) model over another (augmented) model as an action, then one can score the prior plausibility of a model in terms of its expected free energy so that the difference in expected free energy furnishes a log prior over models that can be combined with the (variational free energy bound on) log marginal likelihoods to score their posterior probability. This can be expressed in terms of a log Bayes factor (i.e., odds ratio) comparing the likelihood of two models, given some observation, o :

$$\begin{aligned}\ln \frac{P(m|o)}{P(m'|o)} &= \ln \frac{P(o|m)P(m)}{P(o|m')P(m')} = \Delta F + \Delta G \\ \Delta F &= \ln P(o|m) - \ln P(o|m') \\ \Delta G &= \ln P(m) - \ln P(m') = G(\mathbf{a}|m) - G(\mathbf{a}'|m')\end{aligned} \quad (9)$$

Here, $\mathbf{a}$ and $\mathbf{a}'$ denote the posterior expectations of parameters under a parent m and augmented model m' , respectively. The difference in expected free energy reflects the information gain in selecting one model over the other, following Equation 6. One can now retain or reject the parent model, depending on whether the log odds ratio is greater than or less than 0, respectively. Active model selection, therefore, finds structures with precise or unambiguous likelihood mappings. When assimilating new (e.g., training) data, one can simply equip the model with a new latent cause to explain each (unique) observation when, and only when, expected free energy decreases (Friston et al., 2023b). This affords a fast kind of structure learning. Before illustrating the above procedures, we now consider a particular structural form that characterizes the generative models used in the illustrative applications.

Renormalizing generative models

Renormalizability is feasible under the models in Figure 1. This follows because the dynamics constitute a coordinate descent on

variational free energy, leading to paths of least action, namely, a path integral of variational free energy (Friston K. et al., 2023). However, we also require renormalizing transformations of model parameters from one hierarchical level to the next. These scale transformations entail a coarse graining that generally induces a separation of temporal scales, such that the dynamics—here, belief updating—slow down as one ascends levels or scales. The implicit RG flow rests on the inclusion of dynamics in the generative model.

In discrete time, the inclusion of paths means that a succession of states at any given level can be generated by specifying the initial state and successive transitions encoded by the slice of the transition tensor specified by a path. Crucially, the initial state and path can be generated from the superordinate state, which has its own dynamics and associated path. This structure can be read in a number of ways. It can be regarded as a discrete version of switching dynamical systems (Linderman et al., 2016; Olier et al., 2013), in which the switching variables (i.e., paths) change at a slower timescale than the dynamics or paths at the scale being switched. In the limit of continuous time, the composition of implicit RG operators means that one can model changes or switches in velocity—that is, acceleration—and, at the next level, changes in acceleration—that is, jerk, and so on. In continuous state-space models, this reduces to working in generalized coordinates of motion (Friston et al., 2010; Kerr and Graham, 2000).

A more intuitive view of the latitude afforded by temporal renormalization is that successively higher levels encode sequences of sequences and, implicitly, compositions of events or episodes. In other words, at a deep level, one state can generate sequences of sequences of sequences, thereby destroying the Markovian properties of content generated at the lowest level. It is this deep structure that has been leveraged in applications of active inference under continuous models of song and speech; see, for example, Friston and Kiebel (2009) or Yildiz et al. (2013). We will see the discrete homologs of the ensuing semi-Markovian processes later.

In addition to the renormalization over time, we must also consider renormalization over state space. The example in Figure 3 illustrates a graphical model in which groups of states at a lower level are generated by a single state at the higher level. In the next section, we will associate states at the lowest level with the value of pixels and successive block transformations (i.e., tessellations) with image compression. An important aspect of these models is that the states at any level never share children in the lower level. This renders latent factors at every level conditionally independent. Conditional independence follows from the fact that the Markov blanket of any given state comprises its parents, children, and parents of children; however, its children have no co-parents, rendering hidden factors D-separated when conditioned on the initial states (and paths) of the level below. In turn, this has the practical implication that the likelihood mappings that link different levels or scales (i.e., the $\mathbf{D}$ and $\mathbf{E}$ tensors in Figure 2) are low-dimensional matrices at every level of the hierarchy. Of course, if one can factorize the states at a given level into distinct state factors, these low-dimensional matrices become lower-dimensional tensors. This means that this scheme is general in the sense that one could construct matrices of the sort described above from tensorial distributions simply by taking Kronecker tensor products of state factors and reorganizing the tensors to matrices. However, it is not general in the sense that it relies on state factors at each level that predict nonoverlapping local (spatiotemporal) regions of the states at the level below.

Models of this sort can be regarded as generating local dependencies within each group of states at the lower level, whereas global (between-group) dependencies are modeled at higher levels. They feature a characteristic progression from local to global as one ascends the hierarchy: c.f., Hochstein and Ahissar (2002), providing an efficient way to model data that show strong local dependencies over space and time⁶.

In the setting of discrete models, one can also regard an RGM as an expressive way of modeling nonlinearities in the generation of data, much in the spirit of deep neural networks with nonlinear activation or rectification functions. In the limit of precise likelihood mappings, this can be viewed as the composition of logical operators. For example, if the first two elements of the leading dimension of a $\mathbf{D}$ likelihood tensor were nonzero, this means that the parent state can generate the first OR second child's initial state. Conversely, nonzero elements of two likelihood tensors generate a particular state of one child AND another child. Heuristically, one can see why an RGM can compose operators to represent or generate content that has a compositional structure. The inversion of an RGM could be construed as a simple form of abductive (e.g., modal) logic. In the ensuing example, this perspective is leveraged to learn the composition of local image features subtending global objects.

Image compression and compositionality

We start with the simplest example of renormalizing generative models appropriate for image classification and recognition. The aim is to automatically assemble an RGM to classify and generate the image content to which it is exposed. In other words, we seek procedures for automatic structure learning followed by learning the parameters of the ensuing structure, which can then be deployed to classify or generate the kind of content on which it was trained.

The first step is to quantize continuous pixel values for an image of any given size. Effectively, this involves mapping continuous pixel values to some discrete state space, whose structure can be learned automatically by recursive applications of a blocking transformation. Note that this procedure gives us the first (and lowest) level of a hierarchical model and is not required to map between higher levels as they will all deal in categorical variables. In this example, we group local pixels by tiling (i.e., tessellating) an image, partitioning it into little squares with a spin-block

transformation (Vidal, 2007)⁷. Each group is then subject to singular value decomposition, given a training set of images, to identify an orthogonal (spatial) basis set of singular vectors. This grouping is followed by a reduction operator that retains singular variates with large singular values (here, the first 32 principal vectors based on groups of 4×4 pixels). Under linear transformations, this is guaranteed to maximize the mutual information in accordance with Equation 3 (given the absence of prior constraints). One way to see this is to assume a precise conditional distribution of a continuous variable I given observations that can be expressed in terms of linear weights inside a Dirac delta and identify the weights that maximize the mutual information (using ρ to indicate a mean-centered sample distribution):

$$\begin{aligned} p(I|o) &= \delta(o - W \cdot I) \Rightarrow D_{KL}[\rho(I)p(o|I) \parallel \rho(I)p(o)] \\ &= -E_{\rho(o)}[\ln \rho(o)] \approx \frac{1}{2} \ln \left| (W^T U S)^2 \right| \\ &\Rightarrow U \propto \arg \max_W \left| (W^T U S)^2 \right| \end{aligned} \quad (10)$$

$$E_{\rho(o)}[I \odot I^T] = U S^2 U^T, [I_1, I_2, \dots] = U S V^T$$

The conditional entropy of the delta distribution tends to zero and is, therefore, omitted on the second line of Equation 10. The approximate equality rests on the assumption that the sample distribution for o is approximately Gaussian and uses a singular value decomposition of the concatenated samples, where the diagonal elements of S are singular values. To maximize the entropy of the marginal of o , the weights must be chosen to be proportional to the left singular vectors (columns of U) or, equivalently, the eigenvectors of the sample covariance.

The set of singular variates for each group specifies the pattern for any given image at the corresponding location. The continuous variates can then be quantized to a discrete number of levels (here, seven) to provide a discrete representation of each block. This corresponds to the first RG operator (a.k.a. blocking transformation). Given a partition of the image into quantized blocks, we now apply a second block transformation into groups of four nearest neighbors. This reduces the number of blocks by a factor of two in each image dimension. One then repeats this procedure until only one group remains at the highest level or scale.

Each application of the block transformation creates a likelihood mapping ($\mathbf{D}$) from the states at a higher level to the lower level. In other words, the state of a latent factor at any level generates the states of a group at lower levels (or quantized singular variates for pixels within a group at the image level). The requisite likelihood matrices can be assembled using a fast form of structure learning based on Equation 9. This equation says that the likelihood mapping (in the absence of any constraints) should have the maximum mutual information. This is assured if each successive column of the likelihood matrix is unique. In turn, this means we can automatically assemble the requisite likelihood

⁶ A more thorough treatment of the implicit RG operator or blocking transformation would rest on characterizing conditional independencies within and between groups in terms of their Markov blankets (Friston, K.J., Fagerholm, E.D., Zarghami, T.S., Parr, T., Hipólito, I., Magrou, L., Razi, A., 2021c. Parcels and particles: Markov blankets in the brain. *Network Neuroscience* 5, 211–251). However, for systems with strictly local conditional dependencies (e.g., Markov random fields), the Markov blanket of any element constitutes its nearest neighbors. This motivates the use of spin-block transformations, such as those described by Kaluza and Meyer-Ortmanns (2010). On the role of frustration in excitable systems. *Chaos* 20, 043111, Vidal, G. (2007). Entanglement renormalization. *Physical Review Letters* 99.

⁷ In practice, we use overlapping groups, where the singular value decomposition is applied following weighting by a radial (Gaussian) basis function whose standard deviation is the distance between group centers.

Renormalising Generative model

Likelihoods or empirical priors
 $\mathbf{A}^n = \{\mathbf{D}_1^n, \mathbf{E}_1^n, \mathbf{D}_2^n, \mathbf{E}_2^n, \dots\}$

Dynamics or transition priors
 $\mathbf{B}^n = \{\mathbf{B}_1^n, \mathbf{B}_2^n, \dots\}$

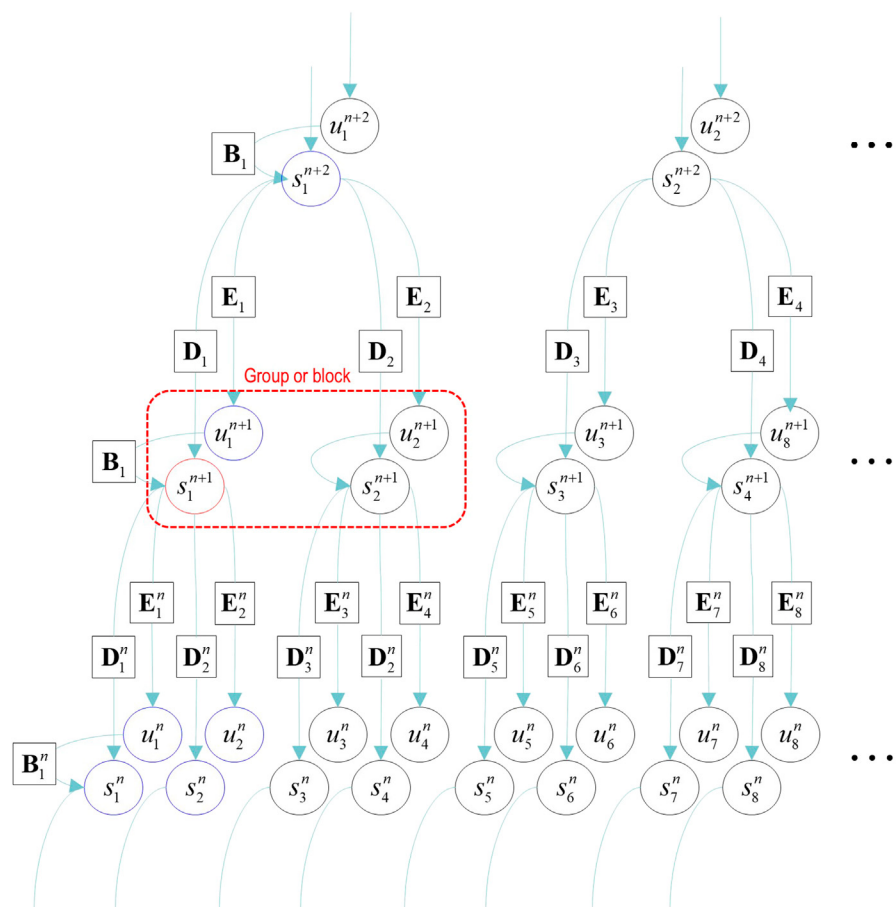


FIGURE 3

Renormalizing generative model. This graphical model illustrates the architecture of renormalizing generative models (temporal renormalization has been omitted for clarity). In these models, the latent states at any given level generate the initial conditions and paths of [groups of] states at the lower level (red box). This means that the trajectory over a finite number of timesteps at the lower level is generated by a higher state. This entails a separation of temporal scales and implicit renormalization, such that higher states only change after a fixed number of lower state transitions. This kind of model can be specified in terms of (i) transition tensors ($\mathbf{B}$) at each level, encoding transitions under each discrete path, and (ii) likelihood mappings between levels, corresponding to the $\mathbf{D}$ and $\mathbf{E}$ tensors of previous figures. These can be treated as subtensors of likelihood mappings $\mathbf{A}^n = \{\mathbf{D}_1^n, \mathbf{E}_1^n, \mathbf{D}_2^n, \mathbf{E}_2^n, \dots\}$ that furnish empirical priors over the states (and paths) at each level (n). Because each state (and path) has only one parent, the ensuing Markov blanket (blue circles) of each state (red circle) ensures conditional independence among latent factors. In summary, a renormalizing generative model (RGM) is a hypergraph, parameterized by two sorts of ($\mathbf{A}$ and $\mathbf{B}$) tensors, in which the children of states at any level bipartition into initial states and paths. In this example, the blocking transformation groups pairs of states (and paths).

mappings by appending *unique* instances of quantized singular variates (encoded as one-hot vectors) in a set of training images. This results in one-hot likelihood arrays for each group that share the same parent at the higher level:

$$\begin{aligned}
 \mathbf{A}^\ell &= \mathbf{D}_1^\ell \\
 \mathbf{A}^{\ell-1} &= \overbrace{\{\mathbf{D}_1^{\ell-1}, \mathbf{D}_2^{\ell-1}, \mathbf{D}_3^{\ell-1}, \mathbf{D}_4^{\ell-1}\}}^{pa(\mathbf{D}^{\ell-1})=1} \\
 &\vdots \\
 \mathbf{A}^{n+1} &= \overbrace{\{\mathbf{D}_1^{n+1}, \mathbf{D}_2^{n+1}, \mathbf{D}_3^{n+1}, \mathbf{D}_4^{n+1}\}}^{pa(\mathbf{D}^{n+1})=1} \overbrace{\{\mathbf{D}_5^{n+1}, \mathbf{D}_6^{n+1}, \mathbf{D}_7^{n+1}, \mathbf{D}_8^{n+1}\}}^{pa(\mathbf{D}^{n+1})=2}, \dots \\
 \mathbf{A}^n &= \overbrace{\{\mathbf{D}_1^n, \mathbf{D}_2^n, \mathbf{D}_3^n, \mathbf{D}_4^n\}}^{pa(\mathbf{D}^n)=1} \overbrace{\{\mathbf{D}_5^n, \mathbf{D}_6^n, \mathbf{D}_7^n, \mathbf{D}_8^n\}}^{pa(\mathbf{D}^n)=2} \overbrace{\{\mathbf{D}_9^n, \mathbf{D}_{10}^n, \mathbf{D}_{11}^n, \mathbf{D}_{12}^n\}}^{pa(\mathbf{D}^n)=3}, \dots \\
 &\vdots
 \end{aligned}
 \tag{11}$$

We have dropped $\mathbf{E}^n = [1, 1, \dots]$ in Equation 11 because there is only one path in the absence of dynamics. In this case, the penultimate level comprises the likelihood mappings that generate each quadrant of the image in terms of quadrants of quadrants, much like a discrete wavelet decomposition. The ultimate level corresponds to priors over the (initial) states or class of image. In short, structure learning emerges from the recursive application of blocking transformations of some training images. This is a special case of fast structure learning (see Algorithm 1), described in detail in the next section.

At the first glance, this procedure may appear to generate likelihood mappings ($\mathbf{D}$ matrices) of increasing size because the combinatorics of increasingly large groups could explode at higher levels. However, by training on a small number of images, one upper bounds the number of states at each level. This follows from the fact that there can be no more columns of the likelihood matrix than there are unique images in the

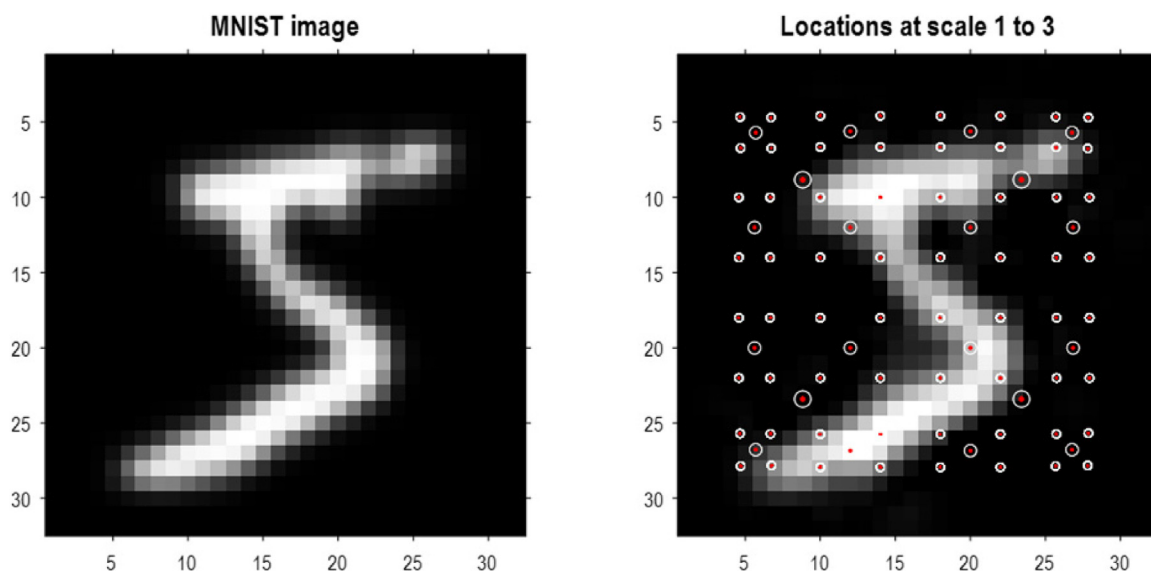


FIGURE 4

Quantizing images. The left panel shows an example of an MNIST image after resizing to 32 pixels $\times$ 32 pixels, following histogram equalization. The image in the right panel corresponds to the image in pixel space generated by quantized singular variates, used to form a linear mixture of singular vectors over groups of pixels. The centers of the (4 $\times$ 4) groups of pixels are indicated by the small red dots (encircled in white). In this example, the singular variates could take seven discrete values centered on zero for a maximum of 16 singular vectors. At subsequent levels, (2 $\times$ 2) groups of groups are combined via grouping or blocking operators. The centroids of these groups (of groups) at the three successive scales are shown with successively larger red dots. At the third scale, there are four groups corresponding to the quadrants of the original image.

training set. In other words, one can effectively encode a finite number of images without any loss of information such that RGM inversion corresponds to lossless compression.

To generalize to a more expansive training set, one can populate the likelihood mappings with small concentration parameters and use active learning to recover an optimal lossy compression. According to Equation 7, active learning simply means accumulating appropriate Dirichlet counts in the likelihood mappings until the mutual information converges to its maximum. This offers a principled way to terminate the ingestion of training data, after which there can be no further improvement in expected free energy or mutual information.

A worked example

To demonstrate the above methods, they were applied to the MNIST digit classification problem (LeCun and Cortes, 2005). MNIST images were preprocessed by up-sampling to 32 pixels $\times$ 32 pixels, smoothing, and histogram equalization⁸. In addition, they were converted into a format suitable for video processing with three (TrueColor or RGB) channels. An exemplar image is shown in Figure 4 (left panel). The blocking transformation to discrete state space is

illustrated in the right panel, which shows the reconstructed image (i.e., local mixtures of singular vectors weighted by discrete singular variates). The centroids of each group are shown with small red dots, where each group comprises pixels within a radius of four pixels.

Based on the prior that there can be a dozen ways of writing any given number, the first 13 (Baker's dozen) images of each digit class were used for fast structure learning. This produced an RGM with four levels. The centroids of the ensuing groups of increasing size are shown as successively larger red dots in Figure 4. At the penultimate level, this grouping is into quadrants. In this application, we equipped the last level, covering all pixel locations, with a likelihood mapping between the known digit class (i.e., label) and compressed representations at the penultimate level.

The likelihood mappings are shown in Figure 5 to illustrate the ensuing structure. The lower row shows concatenated likelihood mappings at each level. The upper row reproduces these mappings after transposing to illustrate how states at one level generate the states of groups at the subordinate level. For example, at level 1, we have 16 groups of pixels whose states are generated by four groups at level 2. Similarly, the four groups at level 2 are generated by one group at level 3. Level 4 implements our prior knowledge about digit classes, effectively providing a mapping from digit class to the (13 $\times$ 10) exemplar images that have been compressed in a lossless fashion.

Note that the graphics in Figure 5 represent concatenated likelihood matrices. In other words, each block of the likelihood matrices generates multiple outcomes, namely, combinations of outputs for several groups. This illustrative concatenation conceals the fact that each likelihood (i.e., **A**, **D**, or **E** tensor) is a

⁸ The digits were downloaded from <https://lucidar.me/en/matlab/load-mnist-database-of-handwritten-digits-in-matlab/> and pre-processed by smoothing with a Gaussian convolution kernel (of two pixels width).

$$\mathbf{A}^n = \{\mathbf{D}_1^n, \mathbf{E}_1^n, \mathbf{D}_2^n, \mathbf{E}_2^n, \dots\}$$

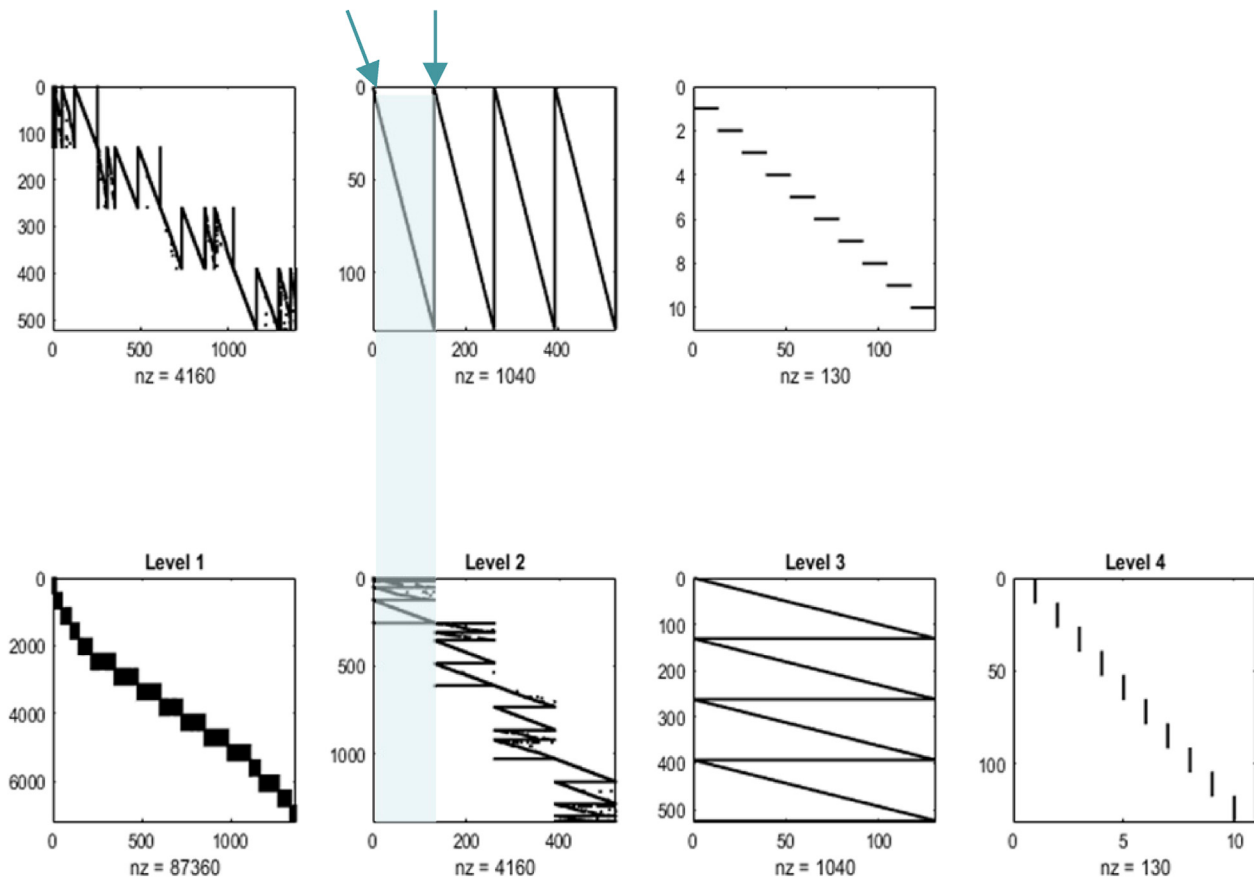
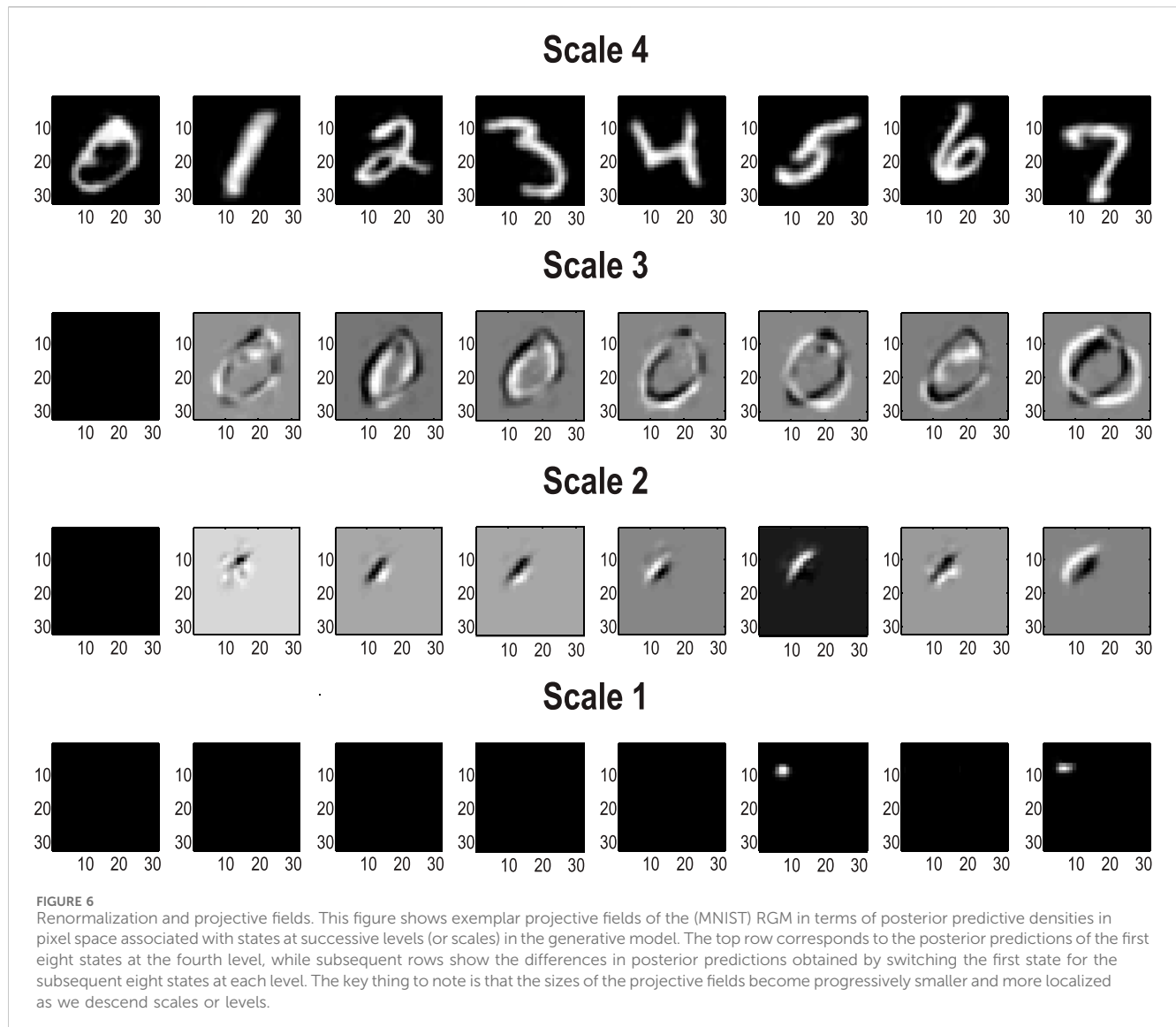


FIGURE 5

Renormalizing likelihoods. This figure is a schematic representation of the composite likelihood mappings (comprising $\mathbf{D}$ and $\mathbf{E}$) among the levels of an RGM, following fast structure learning. In each of these graphics, black indicates a nonzero element, and white indicates a zero element. In the lower row of matrices, the columns of the matrices are the alternative possible values of states at that level, concatenated for all state factors. The rows are the possible values for states (or observations) at the level below, similarly concatenated. By the third level, each latent state can generate an entire image via recursive application of sparse, block-diagonal matrices, where “nz” counts the number of nonzero elements. In this example, the model has been equipped with a fourth level, mapping from 10 digit classes to 130 latent states at the third level (encoding 13 images of 10 digits). These likelihood mappings (that mediate empirical priors) are assembled automatically during structure learning by accumulating unique combinations of (recursively grouped) states at subordinate levels. The upper row reproduces the matrices of the lower row after transposition to illustrate the dimension reduction inherent in the grouping of states. Each transposed matrix is shifted one to the left relative to the lower row. This means the states generated by the upper matrices are represented in the columns, aligning with the states in the conditioning sets in the columns of the matrices below. For example, the thousand or so states at the second level generate over 6,000 states at the first, which specify the mixture of singular vectors required to generate an image. Similarly, the 500 or so states at level 3 generate empirical priors over a partition of level 2 states into four subsets or groups (the first is highlighted in cyan), and so on. Crucially, by construction, the children of states at any level constitute a partition, such that every child is included in exactly one subset. This means that states at any level have only one parent, rendering the subsets of the partition at the higher level conditionally independent. In other words, there are no conditionally dependent co-parents. This enables efficient sum-product operations during model inversion because one must only compute dot products of subtensors (i.e., small matrices) specified by the parents of a group. Note that the matrices in this figure are not simple likelihood mappings: they are concatenated likelihood mappings from all hidden states at one level to all hidden states (and paths) at the subordinate level, where the sum to one constraint is applied to the states (or paths) each child could be in.

relatively small matrix. The implicit sparsity of these likelihoods inherits from the blocking transformations based on a partition at each level. If we had imposed the additional constraint (i.e., structural prior) that these likelihood mappings are conserved identically over groups, then one would have the discrete homolog of a convolutional neural network with an implicit weight sharing. Here, we did not impose this prior constraint because different groups of pixels show systematic, location-dependent differences. From a biomimetic perspective,

this can be likened to differences in the size of receptive fields between central (i.e., foveal) and peripheral visual fields that speak to the principles of maximum mutual information and minimum redundancy (Barlow, 1961; Linsker, 1990; Olshausen and Field, 1996; Simoncelli and Olshausen, 2001). Technically, this is reflected in the fact that the peripheral groups of pixels showed little or no variation over exemplar images. This means that only a small number of singular variates were retained in the periphery and explains why the size of the groups at the first level



increases toward the center of the image (see the lower left panel of Figure 5).

Pursuing the biomimetic theme, Figure 6 illustrates the encoding of images at successively deeper levels of the RGM. The top row (Scale 4) shows the images generated under the first eight levels of the corresponding latent factor. By construction, this factor has 10 levels corresponding to prior knowledge about the class of each digit. The subsequent rows illustrate the projective fields of particular states at lower levels. These fields can be regarded as the changes in predicted stimuli due to changes in the representations (i.e., the complement of receptive fields). In Figure 6, the first state of every factor of every level was selected to provide a baseline image. This was then subtracted from the image generated by subsequent states of the first factor. The purpose of these characterizations was to show that changing posterior expectations at the highest levels (Scales 3 and 4) produces changes everywhere in the image. In other words, these are coarse-grained global representations of the *object* represented. Conversely, at subordinate levels (Scales

1 and 2), the projective fields are restricted to local parts of the image. The form of these projective fields is remarkably similar to that seen in the visual system, namely, compact, simple receptive fields in the early visual cortex and complex fields (e.g., with a center-surround structure) of greater spatial extent at higher levels in the visual hierarchy (Angelucci and Bullier, 2003; Zeki and Shipp, 1988). It is noteworthy that this kind of functional specialization is an emergent property of a renormalizing generative model.

From a statistical or Gestalt perspective, the progressive enlargement of projective fields can also be viewed in terms of composition and the transformation from local, place-coded representations to global object-centered representations; for example, Hochstein and Ahissar (2002) and Kersten et al. (2004). This is again reminiscent of biomimetic architectures in the sense that elemental image features are encoded at the lower levels of a visual hierarchy while objects and classes (e.g., faces) are encoded at much higher levels with successive levels comprising more extended but compressed representations from lower-level representations

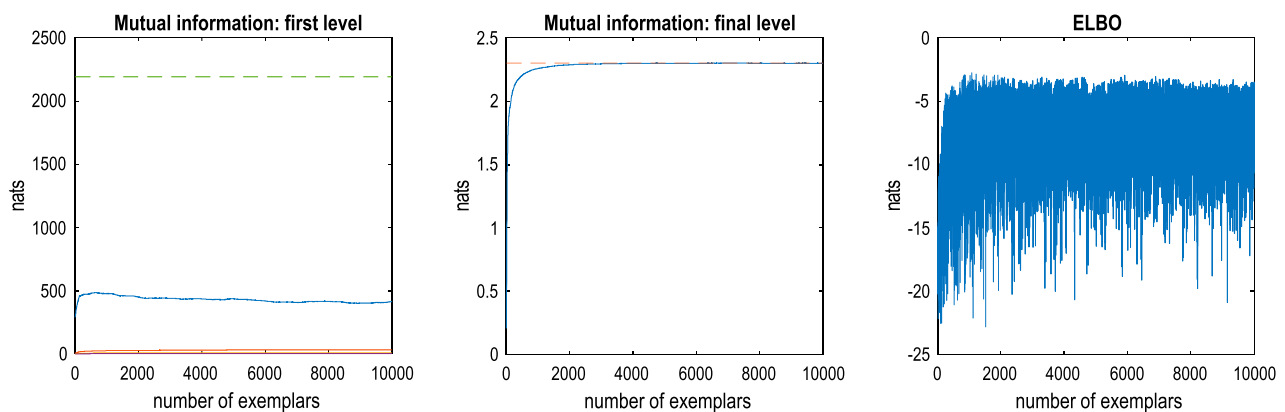


FIGURE 7

Active learning. This figure reports the assimilation or active learning of the training MNIST dataset. In this example, images were assimilated if, and only if, they increased the expected free energy of the RGM. In the absence of prior preferences or constraints, this ensures minimum information loss by underwriting informative likelihood mappings at each level (i.e., maximizing mutual information). The left panel reports the mutual information as a function of ingesting 10,000 training images. The left panel reports the mutual information at the first level (blue line) and intermediate levels. The middle panel reports the mutual information at the final (fourth) level. The dashed lines correspond to the maximum mutual information that could be encoded by the likelihood mappings. The right panel shows the corresponding evidence lower bound (negative variational free energy), scored by inferring the latent states (digit class) generating each image. The fluctuations here reflect the fact that some images are more easily explained than others under this model. As the model improves, there are progressively fewer images with a very low evidence lower bound (ELBO): that is, -16 natural units or fewer.

(Alpers and Gerdes, 2007; Kersten et al., 2004; Zeki and Shipp, 1988).

The foregoing illustrates the generation of content following structure learning of a lossless sort. We now turn to inference and classification. This rests on optimizing the parameters of the RGM structure to maximize the marginal likelihood of some training data. Figure 5 illustrates the results of this active learning during exposure to the first 10,000 training images of the MNIST dataset. This training proceeded by (i) populating all the likelihood mappings with small concentration parameters⁹, (ii) equipping the highest level of the model with precise priors ($\mathbf{D}^4$) corresponding to the class labels, and (iii) accumulating Dirichlet parameters according to Equation (7), with $\alpha = 512$.

This kind of evidence accumulation converges when the expected free energy asymptotes. This convergence is illustrated in the middle panel of Figure 7, which plots the mutual information at the final level as a function of training exemplars. The broken red line corresponds to the upper bound on mutual information afforded by the fact that there are 10 classes or hidden states at the final level. The left panel shows the equivalent (summed) mutual information of lower-level likelihood mappings, while the right panel shows the accompanying variational free energy following each exemplar. One can see that there is an initial period of fast learning that asymptotes in terms of mutual information and (negative) variational free energy with little further improvement after approximately 5,000 samples. Note that the expected free energy provides a convergence criterion that quantifies the

number of training exemplars required to maximize model evidence (on average).

Figure 8 reports the capacity of the ensuing model to correctly infer or classify the class of each the subsequent 1,000 (unseen test) images. This figure provides a nuanced report of classification performance, which reflects the fact that we have two kinds of inference at hand. These rest on (i) the posterior distribution over digit classes and (ii) the marginal likelihood that the image is classifiable under the model. This means that one can assess the accuracy of classification conditioned on whether a particular image was classifiable, enabling a more comprehensive characterization of performance in terms of sensitivity and specificity. The format adopted in Figure 8 plots the classification accuracy as a function of the ELBO (right panel) and the distribution of ELBOs for correctly and incorrectly classified images (left panel). The thing to note here is that classification accuracy increases with the marginal likelihood that each image belongs to the class of numbers. For example, if we split the test data into two halves, the half with the highest marginal likelihood was classified with 99.8% accuracy, while the classification accuracy for the entire test set was only 95.1%.

Images with a low marginal likelihood can be regarded as ambiguous or difficult to classify because they have a small likelihood of being sampled from the class of digits. Figure 9 provides some examples that speak to the potential importance of scoring the validity of classification (Bach et al., 2015).

Summary

This section illustrates the use of renormalization procedures for learning the structure of a generative model for object recognition

⁹ Adding a concentration parameter of $1/16$ to the one-hot likelihood arrays produced by fast structure learning and renormalizing.

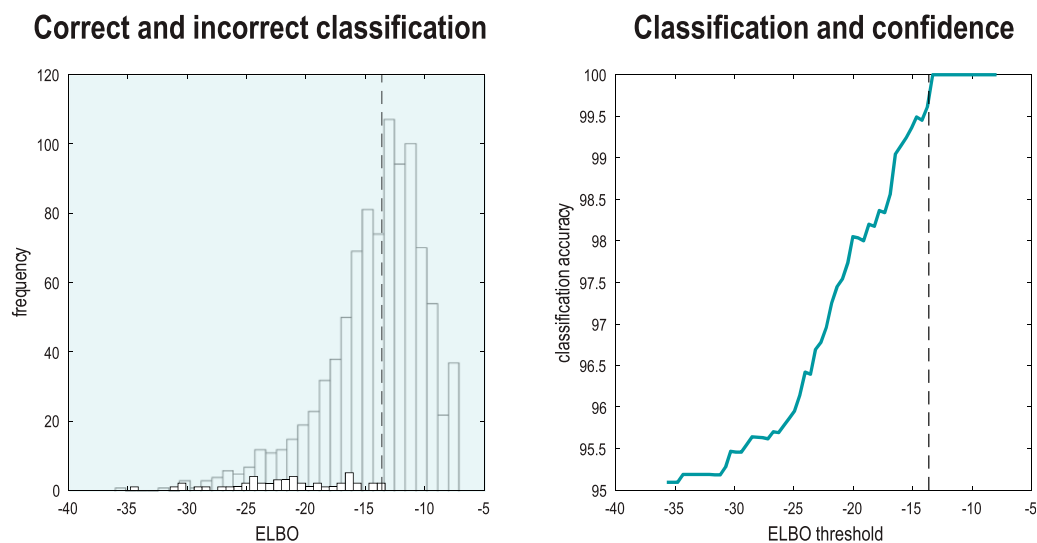


FIGURE 8

Classification and confidence. This figure reports classification performance following the learning described in Figure 7. Because inverting a generative model corresponds to inference, recognition, or classification, one can evaluate the posterior over latent causes—here, digit class—and the marginal likelihood (i.e., model evidence) of an image while accommodating uncertainty about its class. This means that one can score the probability that each image was caused by any digit class in terms of the ELBO. The distribution of the ELBO over the 10,000 training images is shown as a histogram in the left panel (for correctly classified images). The smaller histogram (foregrounded) shows the distribution of log-likelihoods for the subset of images that were classified incorrectly. Having access to the marginal likelihood means that one can express classification accuracy as a function of the (marginal) likelihood the image was generated by a digit. The ensuing classification accuracy is shown in the right panel as a function of a threshold (c.f., Occam's window) on the ELBO or evidence that each image was generated by a digit. The vertical dashed lines show the median ELBO (−13.85 nats). Classification accuracy for all images was only 95.1%. However, the accuracy rises to 99.8% following a median split based on their marginal likelihoods.

and generation in pixel space. The protocol uses a small number of exemplar images to learn a renormalizing structure appropriate for lossless compression. The ensuing structure was then generalized by active learning, that is, learning the likelihood mappings that parameterize the block transformations required to compress images sampled from a larger cohort. This active learning ensures high mutual information between the scale-invariant mapping from pixels to objects or digit classes. Finally, the RGM was used to classify test images by inferring the most likely digit class.

It is interesting to compare this approach to learning and recognition with the complementary schemes in machine learning. First, the supervision in active inference rests on supplying a generative model with prior beliefs about the causes of content. This contrasts with the use of class labels in some objective function for learning. In active inference, the objective function is a variational bound on the log evidence or marginal likelihood. Committing to this kind of (universal) objective function enables one to infer the most likely cause (e.g., digit class) of any content and whether it was generated by any cause (e.g., digit class), *per se*.

In classification problems of this sort, test accuracy is generally used to score how well a generative model or classification scheme performs. This is similar to the use of cross-validation accuracy based on a predictive posterior. The key intuition here is that test and cross-validation accuracy can be read as proxies for model evidence (MacKay, 2003). This follows because log evidence corresponds to accuracy minus complexity: see Equation 2. However, when we apply the posterior predictive density to evaluate the expected log-likelihood of test data, the complexity term vanishes because there is no further updating of model parameters. This means, on

average, the log evidence and test or cross-validation accuracy are equivalent (provided the training and test data are sampled from the same distribution). Turning this on its head, models with the highest evidence generalize in the sense that they furnish the highest predictive validity or cross validation (i.e., test) accuracy. One might argue that the only difference between variational procedures and conventional machine learning is that variational procedures evaluate the ELBO explicitly (under the assumed functional form for the posteriors), whereas generic machine learning uses a series of devices to preclude overfitting, such as regularization, mini-batching, and other stochastic schemes. See Sengupta and Friston (2018) for further discussion.

This speaks to the sample efficiency of variational approaches that elude batching and stochastic procedures. For example, the variational procedures above attained state-of-the-art classification accuracy on a self-selected subset of test data after seeing 10,000 training images. Each training image was seen once, with *continual learning* (and no notion of batching). Furthermore, the number of training images actually used for learning was substantially smaller¹⁰ than 10,000 because active learning admits only those informative images that reduce expected free energy. This Maxwell's demon aspect of selecting the right kind of data for learning will be a recurrent theme in subsequent sections.

Finally, the requisite generative model was self-specifying, given some exemplar data. In other words, the hierarchical depth

¹⁰ In these numerical studies, approximately 20% of training images were accumulated during active learning.

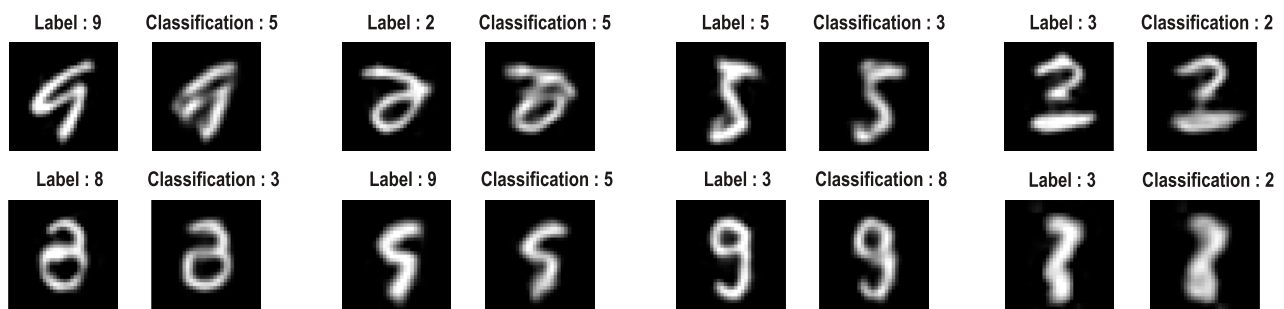


FIGURE 9

Classification failures. This figure provides examples of incorrect classification of images with a small marginal likelihood. Each pair of images presents the training image with its label and the corresponding posterior prediction in pixel space and accompanying maximum *a posteriori* classification.

and size of the requisite tensors were learned automatically within a few seconds on a personal computer. In the next section, we pursue the notion of efficiency and compression in the context of time-series and state-space generative models that are renormalized over time.

Video compression and generative AI

This section generalizes the renormalizing procedures of the previous section to include dynamics for recognizing and generating ordered sequences of images. Procedurally, this simply involves specifying a scale transformation in time and installing unique state transitions into the prior transition tensors at each level of the RGM. In this setting, structure learning reduces to quantizing images in space, color, and time to produce time-color-pixel *voxels*. Unique transitions among neighboring voxels are then recorded in **B** tensors, where each unique voxel state is recorded in the column of the corresponding **A** matrix of each group of voxels. The time series of quantized images is then partitioned into segments of equal lengths (e.g., pairs), and the ensuing segments are inverted (using the **A** and **B** tensors) to evaluate posterior estimates over the initial state and path of each segment¹¹. This produces a new sequence that is coarse-grained in time. Neighboring states (and paths) are then grouped together, and the process is repeated to populate the likelihood mappings (**D** and **E**) from parents at the higher level. By repeating this process, one ends up with a representation of an image sequence in terms of paths through the states of a single factor at the highest level. Each state in this factor generates the initial state (and path) of a group at the lower level, and so on recursively until an image sequence is generated.

Algorithm 1 provides a pseudocode description of the procedure, where R denotes the number of frames constituting a video event and j labels successive blocks in terms of unique instances. Here, the renormalization implicit in fast structure learning is expressed in terms of the Dirichlet parameters of the requisite tensors: $\mathbf{a}'' = \{\mathbf{d}_1'', \mathbf{e}_1'', \mathbf{d}_2'', \mathbf{e}_2'', \dots\}$.

// reshape and partition video frames

$$\{I_1, I_2, \dots\} \leftarrow I \in \mathbb{R}^{pixels \times pixels \times colours \times frames}$$

$$I_b \in \mathbb{R}^{(block \times colours \times R) \times T} : T = \frac{frames}{R}$$

for every b

$$[U_b, S_b, V_b] = SVD(I_b) \quad // \text{ singular value decomposition}$$

$$U_b \leftarrow U_b \odot S_b \in \mathbb{R} \quad // \text{ scale singular variates}$$

$$\mathbf{o}_b^0 \leftarrow Bin(U_b) \in (0, 1) \quad // \text{ and quantise into bins}$$

end for

// recursive renormalisation

while $\#(\mathbf{o}'') > 1$

$$\{\mathbf{o}_1'', \mathbf{o}_2'', \dots\} \leftarrow \mathbf{o}'' \quad // \text{ partition into spatial blocks}$$

$$\mathbf{o}_b'' = \{\mathbf{o}_{b_1}'', \mathbf{o}_{b_2}'', \dots\} : \mathbf{o}_{b_g}'' \in (0, 1)$$

for every b // accumulate unique blocks

$$j = unique(\mathbf{o}_b'') \in \mathbb{N}^{1 \times T} \quad // \text{ find unique blocks}$$

for every τ

$$\mathbf{a}_{j(\tau)}^{n,g} = \mathbf{o}_{b_g,\tau}'' : pa(g) = f = j(\tau)$$

$$\mathbf{b}_{j(\tau+1),j(\tau),u}^{n,f} = 1, \quad \text{s.t.} \quad \mathbf{b}_{j(\tau+1),j(\tau),i < u}^{n,f} = 0$$

end for

end for

$$\{\mathbf{o}_1'', \mathbf{o}_2'', \dots\} \leftarrow \mathbf{o}'' \quad // \text{ partition into temporal blocks}$$

$$\mathbf{o}_\tau'' \in (0, 1)$$

for every τ // accumulate unique events

$$\mathbf{s}_\tau'' = P(s_0 | \mathbf{o}_\tau'', \mathbf{a}'', \mathbf{b}'') \quad // \text{ initial states}$$

$$\mathbf{u}_\tau'' = P(u | \mathbf{o}_\tau'', \mathbf{a}'', \mathbf{b}'') \quad // \text{ and paths}$$

$$\mathbf{o}_\tau^{n+1} \leftarrow (\mathbf{s}_\tau''; \mathbf{u}_\tau'') \quad // \text{ concatenate}$$

end for

end while

Algorithm 1. Fast structure learning.

By partitioning each sequence into nonoverlapping pairs during the timescale transformation at successive levels, one effectively halves the length of the sequence when ascending from one level to the next.

¹¹ Model inversion is used so that this fast structure learning can be applied to categorical probability distributions over quantized singular variates, such as when supplied with event-related averages of quantized images.

From the perspective of generating sequences, this means that a state at one level generates two successive states at the lower level by generating the initial state and the path to the second state. The ensuing RGM acquires a definitive feature of such models, namely, a separation of temporal scales, in which there are two (or more) belief updates for every update at the level above. This means that higher levels encode sequences of sequences of sequences that can be regarded as *episodes* of successive *events*. From the perspective of recognition or classification, an image sequence is successively compressed with a coarse graining or blocking over space and time into a succession of events. This furnishes an *event-based* representation that generalizes the *object-based* representations of the preceding section. Note that this kind of RGM compresses images or scenes that may involve multiple objects or, indeed, causes that may not have the attribute of objecthood, such as textures and backgrounds.

During inference, the separation of temporal scales manifests a particular kind of reactive message passing (Bagaev and de Vries, 2021; Hewitt et al., 1973). Because each level generates a small sequence (here pairs), the variational message passing depicted in Figure 2 is scheduled as follows: at the highest level, priors are passed to the level below to generate two iterations. After two belief updates, the ascending messages are returned to the higher level to form a posterior over events. However, before the lower level can respond, it must query its lower level, waiting for two iterations before updating its beliefs, and so on, down to the first level. This means any given level receives messages or requests from the level above and responds to those requests at a slower rate than it exchanges messages with its subordinates. In short, lower levels update their beliefs more quickly, in a way that rests on an asymmetry in the frequency of hierarchical message passing—an asymmetry that characterizes message passing in real neuronal networks (Bastos et al., 2015; Hasson et al., 2008; Kiebel et al., 2008; Pefkou et al., 2017).

A worked example

To illustrate the basic architecture of this RGM, we used a short video sequence of a dove flapping her wings. The original video¹² was down-sampled to 32 frames, where each frame comprised 128 pixels $\times$ 128 pixels with three TrueColor channels. The first scaling transformation from images to discrete state space used the same nearest-neighbor block transformation as in the previous section; however, here, we blocked each image into 4×4 blocks, generating an RGM with two hierarchical levels. Crucially, singular value decomposition operated on successive pairs of images (c.f., spatiotemporal receptive fields). This simply involved reshaping the tensors for each image block to concatenate color and time before applying a singular-value decomposition. The ensuing singular vectors, therefore, span three colors and two time points, compressing the video into 64 time-color-pixel *voxels*. The remaining scaling transformations constructed the requisite likelihood and transition matrices as described in Algorithm 1.

Figure 10 shows the first frame from the original video and the reconstructed frame following compression using the same format

as Figure 4. Figure 11 illustrates the generation of a movie with 128 frames or 64 voxels from a structure learned from compressing a video of 64 frames (a cycle through two flaps of the wings). The RGM has compressed each cycle into eight events, repeated four times during generation. The upper right panel shows the discovered transitions among these events. In this instance, we have a simple *orbit* or closed path where the last event transitions to the first.

The upper left panel depicts the posterior distribution over states at the highest level in image format; here, it shows four cycles. These latent states then provide empirical priors over the initial states of the four image quadrants at the subordinate level (via D). The accompanying predictive posterior over paths (via E) shows that each of the four paths was constant over time, thereby generating predictive posteriors over the requisite states at the first level (i.e., singular variates) and, ultimately, the posterior predictions in pixel space. The first and last generated images are shown in the lower row.

Figure 11 illustrates the ability of the RGM to generate video content. Conversely, Figure 12 illustrates the inference or recognition of images presented as (partial) stimuli using the same format. The upper panels show the predictive posteriors over states (on the left) and paths (on the right), respectively. The images below correspond to the predictions in pixel space at the first time points and at the last time point. The corresponding stimuli are shown in the lower row of images. This illustration of model inversion speaks to some key biomimetic aspects.

First, by generalizing scale transformations to both space and time, we induce nontrivial posteriors over paths or dynamics. The separation into predictive posteriors over states and paths has a clear homology with the segregation of processing in the visual cortical hierarchy in the brain (Ungerleider and Mishkin, 1982). This is often cast in terms of a distinction between dorsal and ventral streams, in which the dorsal stream is concerned with “where” things are in the visual scene and how they move or can be moved (Goodale et al., 2004). In contrast, the ventral stream is responsible for encoding “what” is causing visual impressions. Furthermore, higher or deeper levels of the visual hierarchy show slower stimulus-bound responses than lower levels: for example, Hasson et al. (2008). This is an emergent property of the (perceptual) inference demonstrated in the above example in terms of the successive slowing of belief updating at higher levels.

In Figure 12, the stimuli presented to the RGM were restricted to the upper left quadrant. In neurobiology, this would be like presenting a moving bar in a restricted portion of the visual field (Livingstone and Hubel, 1988). Despite this partial stimulus, the top-down predictions quickly evince a form of pattern completion, effectively seeing what was not actually presented. Indeed, by the sixth frame (i.e., third voxel), the posterior predictions have filled in the missing content. This predictive capacity is reminiscent of functional magnetic resonance imaging studies of predictive processing in human subjects, in which one can record predictive activity in the visual cortex that is induced by only providing one quadrant of the visual stimulus (Muckli et al., 2015).

Paths, orbits, and attractors

In the preceding example, the sequence of images constitutes a closed path, namely, a simple orbit. This can be regarded as a

¹² From <https://pixabay.com/videos/search/birds/>

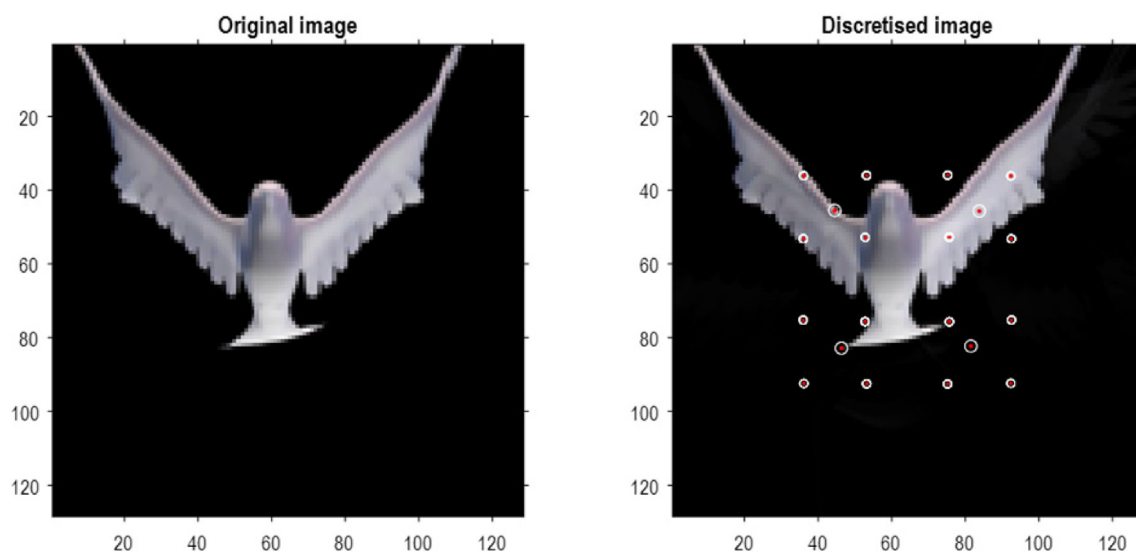


FIGURE 10

A dove in flight. This figure shows a frame from a movie of a (digital) dove flapping her wings. The left panel is a TrueColor (128 pixels $\times$ 128 pixels, RGB) image used for structure learning, while the right panel shows the corresponding posterior prediction following discretization. This example used a tessellation of the pixels into 32 voxels $\times$ 32 voxels, with a temporal resampling of $R = 2$: that is, successive pairs of (32 pixels $\times$ 32 pixels) image patches were grouped together for singular value decomposition. Singular variates took nine discrete values (centered on zero) for a maximum of 32 singular vectors. The locations of the image patches are shown with small red dots (encircled in white). The larger dots correspond to the centroids of blocks, following the first block transformation at the second level of the ensuing RGM.

quantized representation of a periodic attractor. Here, we use the same procedures to compress and generate stochastic chaos, using images generated by a Lorenz system (Friston K. et al., 2021; Lorenz, 1963; Ma et al., 2014; Poland, 1993). The aim of this example is to show how active learning after structure learning furnishes a generative model of chaotic orbits. In this instance, the dynamics or transitions are learned to accommodate switches among paths that acquire a probabilistic aspect due to random fluctuations and exponential divergence of trajectories.

Figure 13 shows the time series used to generate training images. The upper right panel shows a sequence of hidden states over 1,024 time bins, generated by solving stochastic differential equations based on a Lorenz attractor. The upper left panel shows the random fluctuations (i.e., state noise) used in generating these states in terms of an arbitrary mixture of hidden states and stochastic fluctuations on their motion (i.e., prediction and error). The ensuing hidden states were used to generate an image sequence in which the motion of a white circle traced out the trajectory of the first two hidden states (indicated with golden dotted lines). The first half of the resulting sequence was then used for structure learning. A training image and its reconstitution following discretization are shown in the lower panels of Figure 13. The encircled red dots show the location of the groups of pixels at successive scales.

Structure learning was based on the first 512 images. Following this, the model was exposed to the subsequent 512 images to enable learning of the transition dynamics via the accumulation of appropriate Dirichlet parameters. The resulting transitions are shown in the top left of Figure 14. The resulting model has represented this dynamical system with 64 events (i.e., states at the highest level) with switching among certain events that

recapitulate the stochastic switching of trajectories on the underlying Lorenz attractor. Figure 14 shows the predictive posteriors over states at successive levels and predicted images in response to a stimulus. This stimulus was a 128-image sequence from the training set. Following this initial “prompt,” the stimulus was rendered imprecise for a further 128 images and then removed completely for the remainder of the simulation period.

Figure 15 shows the stimuli and posterior predictions induced with this stimulation protocol. Maximum-intensity projections over the horizontal dimension were concatenated to render the fluctuations of the first hidden (Lorenz) state easily visible (compare Figure 15 with Figure 14). This format shows that the first 128 images have been recognized veridically; namely, the paths through renormalized latent state spaces have been correctly identified. When the stimulus is rendered imprecise (the dark region in the lower panel), the posterior predictions continue to produce plausible chaotic dynamics until the stimulus is removed altogether at 256 time bins. At this point, the RGM generates its own outcomes based on the learned generative model, which correctly infers the latent causes of this self-generated content. In other words, the second sequence of stimuli in the lower panel of Figure 15 is generated by the model’s *discrete* representation of chaotic events, as opposed to the first period, in which they were generated by solving *continuous* stochastic differential equations.

These numerical studies speak to two interesting points. First, the RGM can be regarded as the discrete homolog of switching linear and nonlinear dynamical systems; for example, Linderman et al. (2016), Olier et al. (2013), Rabinovich et al. (2010), and Tani and Nolfi (1999). However, in this quantized setting, there is no distinction between linear and nonlinear. This follows because

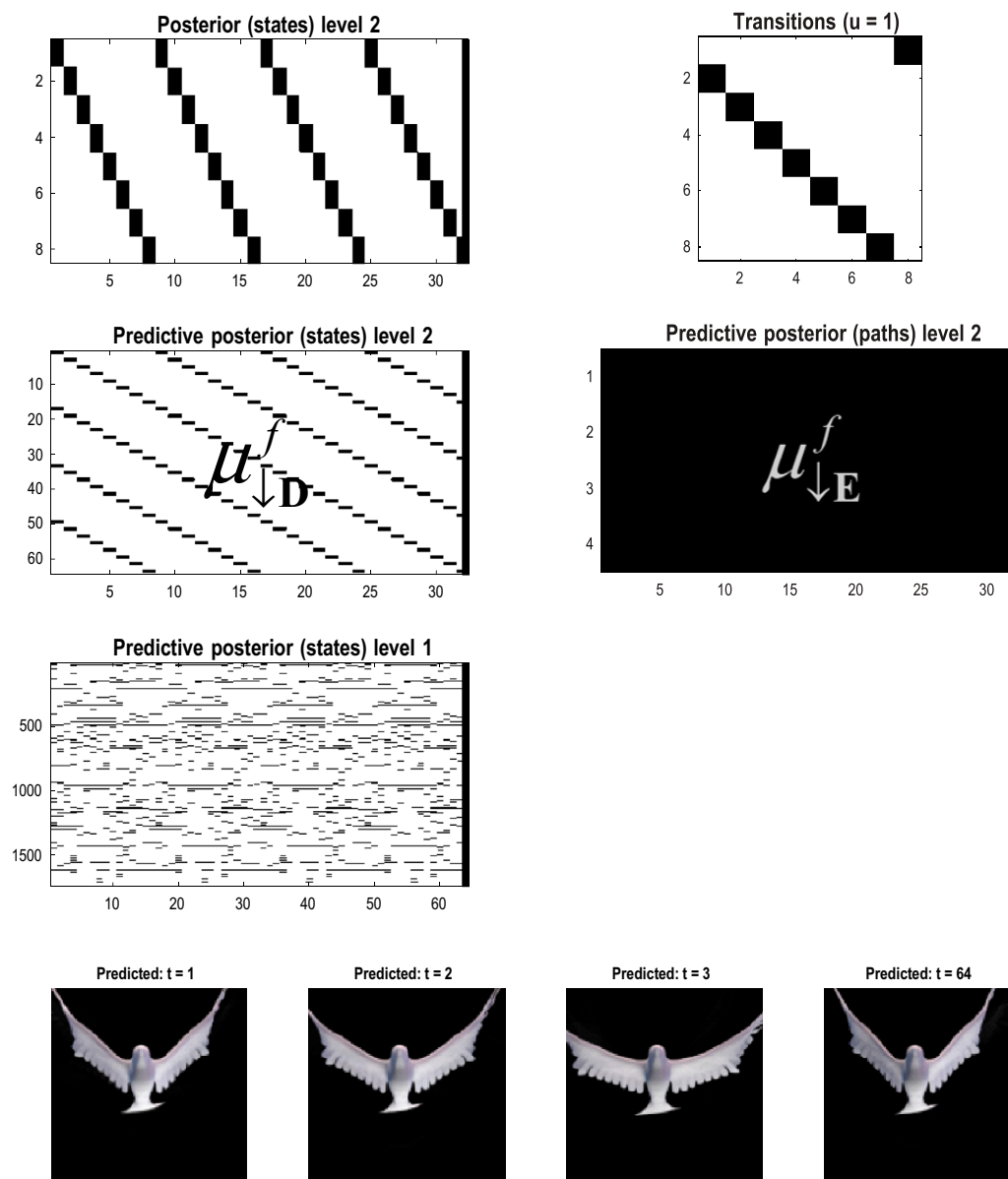


FIGURE 11

Generating movies. Following structure learning based on two cycles of wing flapping (i.e., 64 frames or 32 time-color-pixel voxels), an RGM was used to generate posterior predictions over 128 video frames, namely, four flaps. Each panel plots probabilities, with white representing zero and black representing one. The “Posterior” plots have an x-axis that represents time, with coarser steps at higher levels. The rows along the Y-axis are the different states we might occupy. The “Transitions” plot has columns representing the state we come from and rows representing that we go to. The posterior predictions are the messages passed down hierarchical levels (see Figure 2). The structure learned under this RGM compressed each cycle into eight events. The format of this figure will be used in subsequent examples: the upper right panel shows the discovered transitions among (high-level) events. In this instance, we have an orbit where the last state transitions to the first. The upper left panel depicts the posterior distribution over states at the highest level in image format; here, showing four cycles. These latent states then provide empirical priors over 64 initial states of the four image quadrants at the subordinate level, depicted in the predictive posterior panel below. The accompanying predictive posterior over paths at this level (on the right) shows that each of the four paths was constant over time, thereby generating predictive posteriors over the requisite states at the first level (i.e., singular variates) and, ultimately, the posterior predictions in pixel space. The first and last generated images are shown in the lower row.

all metric state spaces have been tessellated and coarse-grained. This means that the only metric space is the statistical manifold (simplex) associated with probability distributions over discrete events. In principle, these discrete models can handle any degree of nonlinearity in continuous state-space models.

The second issue is a perspective on embedding (Deyle and Sugihara, 2011; Klimovskaia et al., 2020; Ørstavik and Stark, 1998). One might ask how the RGM can recover chaotic dynamics from a sequence of static

images. The answer lies in Takens’ embedding theorem (Deyle and Sugihara, 2011; Takens, 1980), which means that any (chaotic) attractor can be reconstructed from a time-delay embedding, which is implicit in the temporal RG operators used in renormalization. Effectively, we started with an infinite dimensional system (due to the inclusion of random fluctuations) and projected a realization into high-dimensional pixel space. Because the resampling discretized successive pairs of images, we are effectively summarizing the state and the velocity at each (group

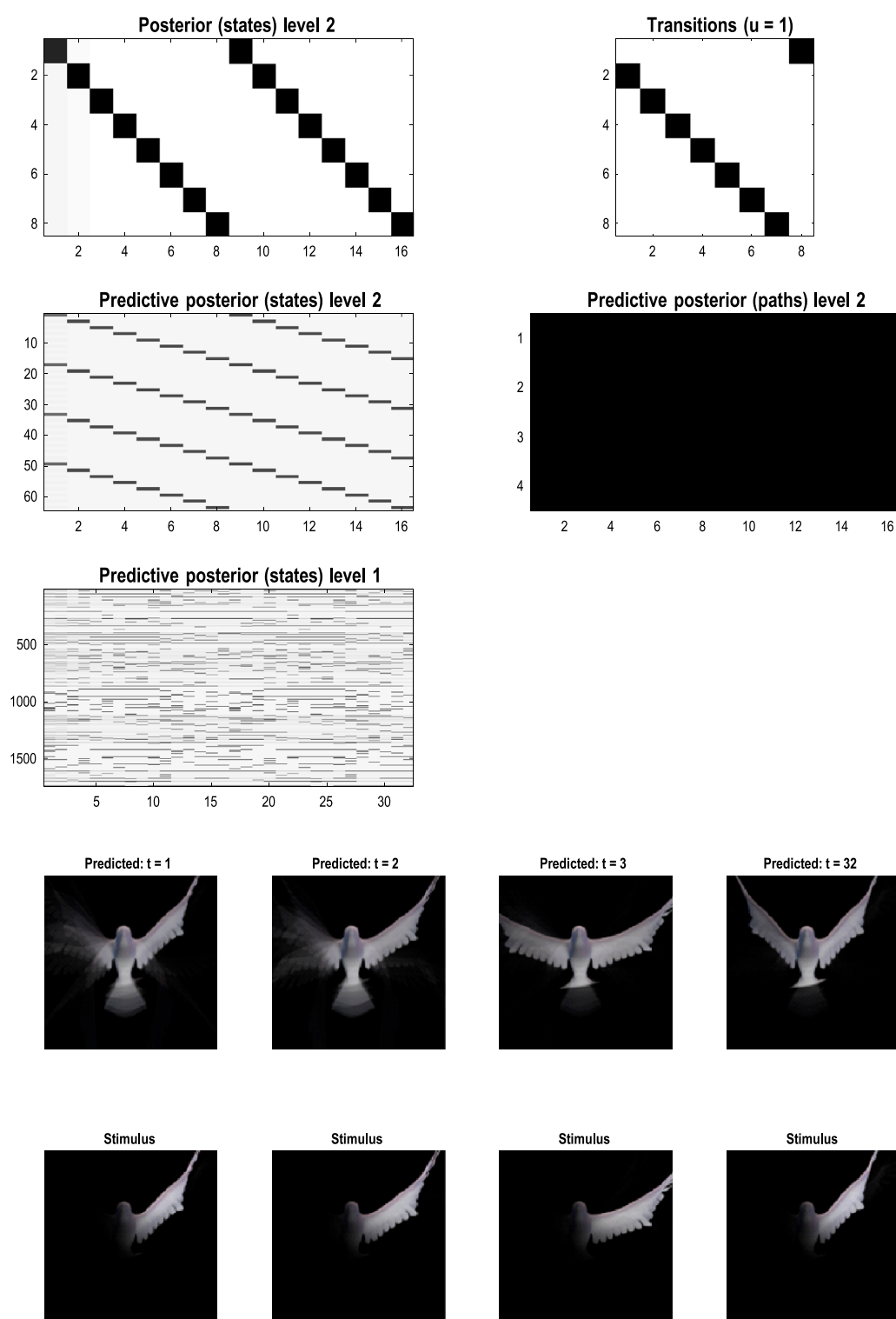


FIGURE 12

Image completion. This figure reproduces the previous figure but presents the model with a partial stimulus in the upper right quadrant. The likelihood mappings were equipped with small concentration parameters (of $1/32$) to model any uncertainty around the events installed during structure learning. The lower rows show the posterior predictions (upper row) and stimulus (lower row) for the first and last timeframes. The key thing to take from this figure is that by the sixth video frame or third voxel ($t = 3$), the posterior predictive density has correctly filled in the missing quadrants and continues to predict the stimuli veridically by treating the missing data as imprecise or uninformative.

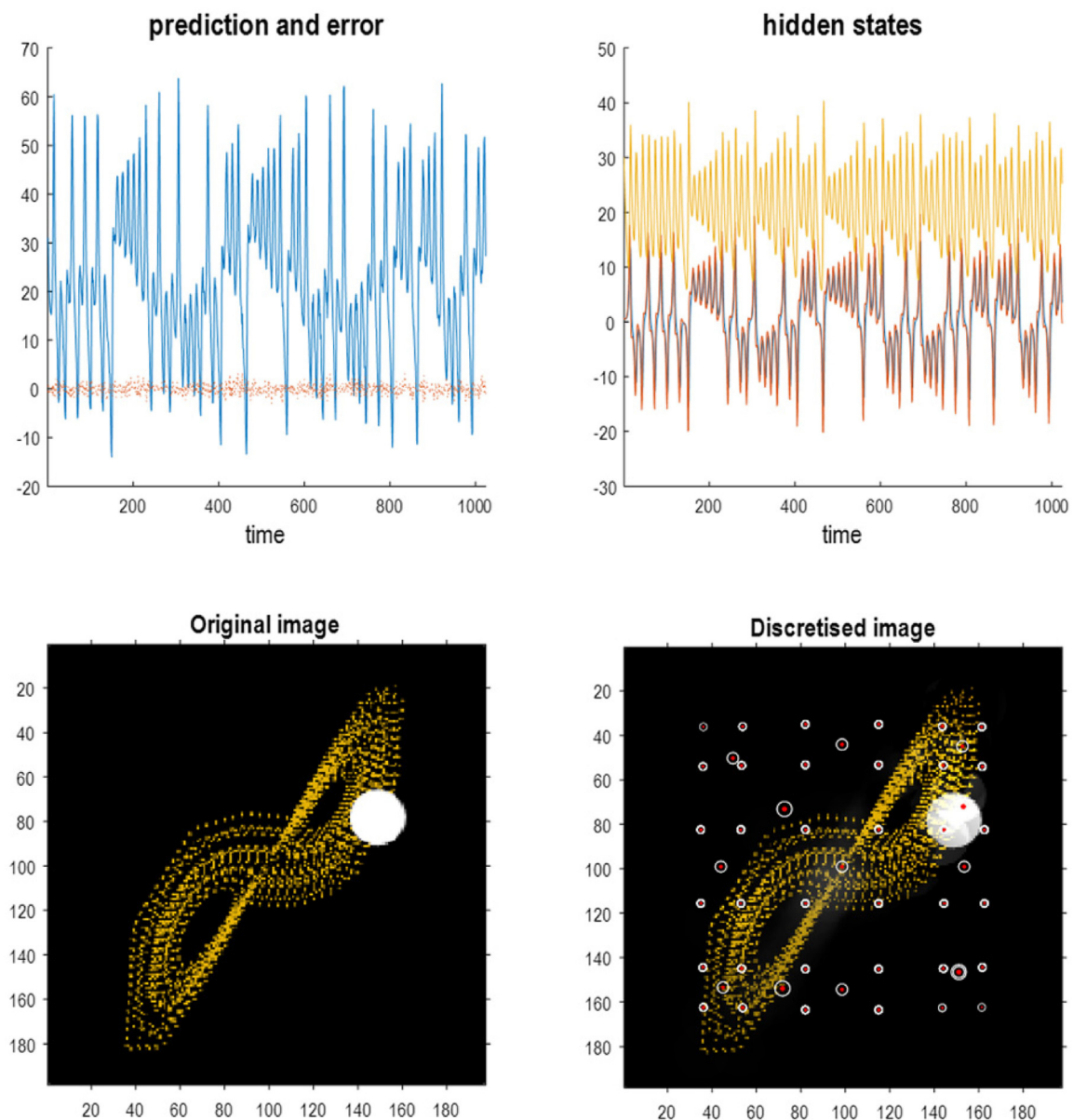


FIGURE 13

Stochastic chaos. This figure summarizes the quantization of images generated from stochastic differential equations based on the Lorenz system.

The upper panels report the solution in terms of the three hidden states of a Lorenz system (right upper panel) and the contribution of random fluctuations, innovations, or state noise (left upper panel). This contribution is characterized in terms of an arbitrary linear mixture of the hidden states and the prediction errors induced by random fluctuations (red line). The hidden states were used to generate an image in which the position of a white ball was specified by the first two hidden states. The ensuing trajectory was used to populate the image with gold dots. One can envisage the ensuing sequence of video frames as depicting a white particle flowing in a medium whose convection is described using the Lorenz equations of motion. (Strictly speaking, the equations pertain to the eigenmodes of convection). The lower left panel shows an exemplar video frame in a TrueColor (198 pixels $\times$ 198 pixels) image. The lower right panel shows the reconstructed image generated from its quantized representation. Following the format of Figures 4, 10, the encircled red dots show the centroids of subsequent groups. In this example, the image was tessellated into (32 $\times$ 32) pixel groups with singular variates taking five discrete values for a maximum of 16 singular vectors. As previously mentioned, the temporal resampling considered successive pairs. The resulting three-level RGM is illustrated in the subsequent figure.

of) pixels at the first level of the RGM, which subsequently embeds pairs of pairs during renormalization. At the highest level, we are left with a probabilistic representation of chaotic dynamics on a 64-dimensional simplex, which retains the essential structure of the random dynamical attractor. Next, we will consider a more natural random dynamical attractor afforded by the motion of natural kinds.

A natural extension

Figure 16 illustrates the spatial blocking of a more natural video—a short sequence (from pixabay.com/video/search/birds) in which a robin alights on a branch, feeds, preens itself for a few seconds, and then flies away. In this example, each block comprises 16 pixels $\times$ 16 pixels,

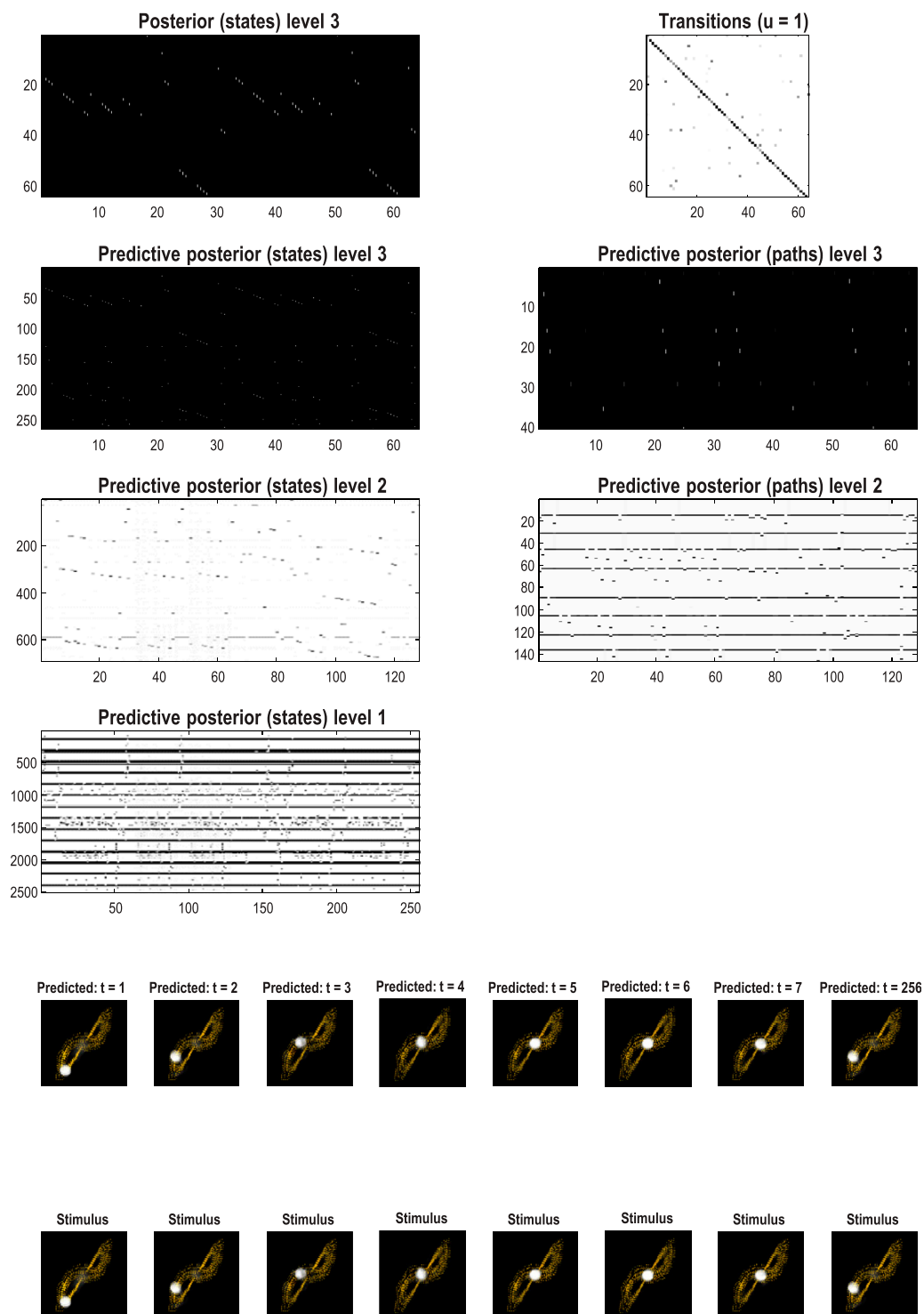


FIGURE 14

Quantized stochastic chaos. This figure uses the same format as Figure 12 to illustrate the learned transitions following fast structure learning and subsequent active learning, based on the first and second half of the image sequence depicted in Figure 13. In this example, the dynamics are summarized in terms of 64 events that pursue stochastic orbits under the discovered probability transition matrix shown on the upper left. The lower panels show the posterior predictions in pixel space and the accompanying stimuli presented for the first quarter of the simulated recognition and generation illustrated in Figure 15.

tessellating (128) video frames cropped to 128 pixels $\times$ 128 pixels. During fast structure learning, this video was compressed via three renormalizations under a three-level RGM. Figure 17 illustrates the posterior predictions of each level on exposure to a segment of the

original movie. Note, in comparison with the previous example, the dynamics are more itinerant. This reflects the fact that this (naturalistic) video has more frequent transitions, as the robin repeated stereotyped behavioral repertoires.

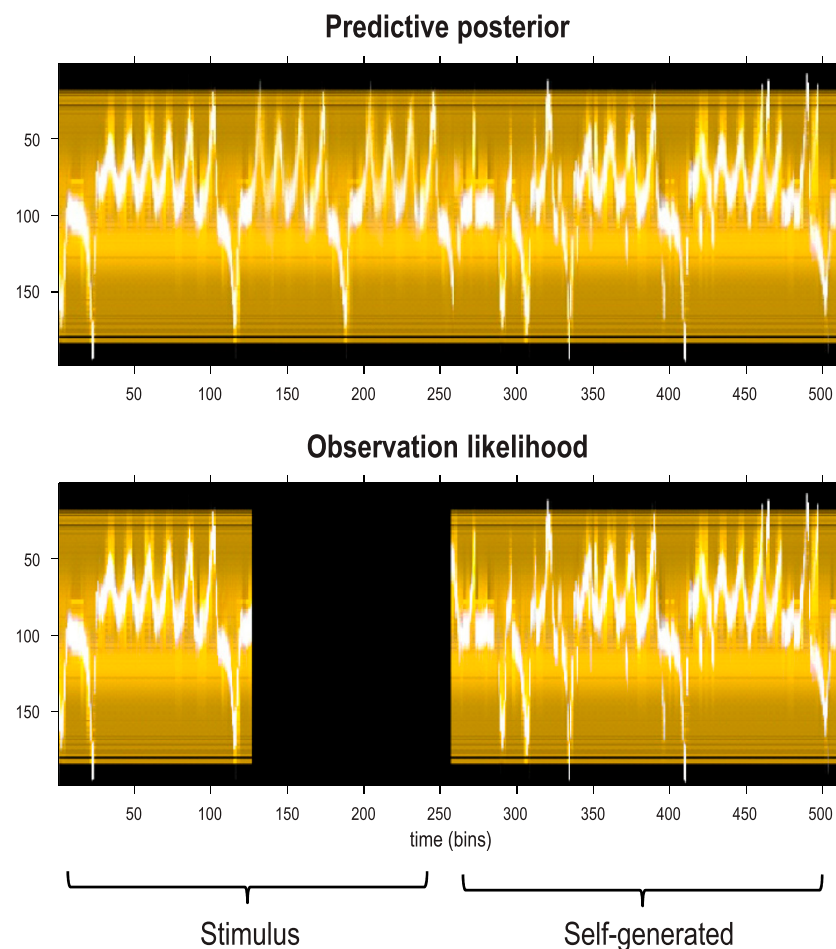


FIGURE 15

Generating stochastic chaos. This figure illustrates the sequence of images predicted (and presented) based on the posterior predictive distributions of the previous figure. Here, maximum intensity projections of each frame have been concatenated to show the video as a single image (i.e., as if each image were viewed from the side). The upper panel shows the predictive posterior in pixel space, while the lower panel reports the stimulus presented to the model. Crucially, the first quarter of the stimulation used images generated from the Lorenz system, while the second half of the stimulus was self-generated, namely, sampled from the learned generative model. The intervening dark regime (second quarter) denotes a period in which the input was rendered imprecise (i.e., presented with poor illumination or with the eyes shut). Despite this imprecise input, the posterior predictions continue to generate plausible and chaotic dynamics until they become entrained by self-generated observations. Here, the simulations lasted for 512 frames (i.e., time bins).

The lower panels of Figure 17 show that the posterior predictions in pixel space were quickly entrained by the stimulus or observations to the extent that the bird's movements were correctly predicted despite withholding frames during movement initiation. This ability to “fill in the gaps” reflects the “filling in” of quadrants in the previous (dove) example. However, here the filling-in is over time, as opposed to pixel space. This ability to generate sequences is reminiscent of generative AI, in which we can regard the initial frames as a “prompt” that is sufficient to generate a succession of future frames. However, in this setting, this predictive capacity does not rest on an autoregressive model of the kind used in transformers (i.e., mapping past content to future content); rather, there is an explicit generative model of trajectories or paths due to the separation of temporal scales which are, effectively, entrained by observations or content. In the next section, we pursue the temporality of this class of models by applying the same procedures to birdsong and music.

Sound compression and generative AI

In this section, we focus on an exemplar application to auditory streams or sound files. In this setting, pixels (i.e., picture elements) are replaced by voxels (volume elements) over frequency and time. These constitute a time-frequency representation of a time series, such as a continuous wavelet transform (CWT). Here, time series are mapped to time-frequency space using (Morlet) wavelet transforms, while an inverse transform converts CWT representations to a linear sound file for playing. Renormalizing generative models for sound is simpler than for video content because there is only one metric dimension (i.e., frequency) that accompanies time. Using a (spin) block transformation to coarse-grain over frequencies reflects the fact or assumption that neighboring frequencies fluctuate in a correlated fashion.

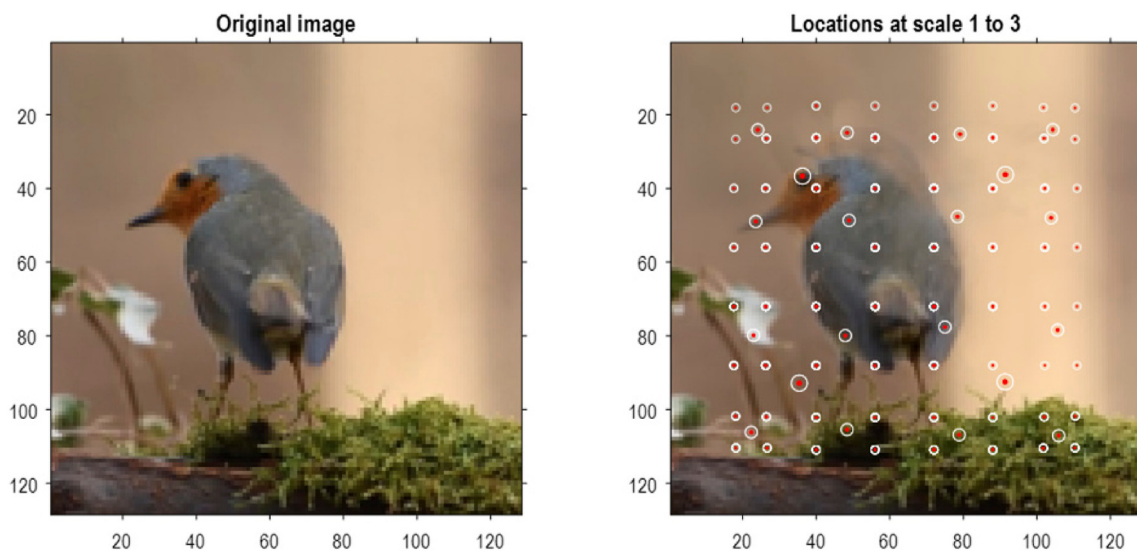


FIGURE 16

Natural kinds. This figure illustrates a single video frame from a short movie of a bird feeding and preening. This movie sequence comprised 128 frames of (128 × 128) TrueColor images. Following the format of previous figures, the left panel shows an original image, and the right panel shows the corresponding image generated from discrete singular variates. In this example, the singular variants took 17 discrete values for a maximum of 32 singular vectors. As previously, the temporal scaling was $R = 2$; that is, pairs of video frames were grouped together to constitute time-color-pixel voxels. The locations of successively grouped voxels are shown with encircled red dots engendering the three-level RGM reported in Figure 17.

A worked example

We will first look at a simple example using sound recordings¹³ of a crossbill bird and then look at a more deeply structured, semi-Markovian process, namely, jazz music. Figure 18 shows the content modeled using the same procedures above after quantizing into 64 frequency bins, with a maximum of 16 singular vectors for each group of (1,024 × 4) time-frequency voxels. The corresponding time series is shown in the upper panel and sampled from the CWT in the lower panel. In this example, which lasts for approximately 8 seconds, the bird makes two calls, each comprising a short sequence of chirps. The second call is longer, with a crescendo of increasingly broadband chirps.

After fast structure learning, this recording engendered an RGM with four levels, with the highest-level states encoding an event of approximately one second, where episodes included intervening periods of a low-frequency background rumble (visible as a band of low-frequency, continuous power in the CWT). This RGM can now be used to generate arbitrarily long sequences of calls. An example is shown in Figure 19, over approximately 32 s (or 256 time-frequency voxels). The upper right panel shows that the time series has been compressed into a succession of eight episodes that follow each other systematically until the last, after which the model has uniform beliefs about the next event. This is reflected in the generated birdsong, in which the RGM completed a sequence of eight events, repeated the last event, and then selected the sixth event, and so on. This resulted in alternating calls, as seen in the sonogram in the lower panel.

We could have continued generating calls indefinitely, where each successive episode would be longer or shorter, depending on the point of “re-entry.” There are clearly many other variations on this theme; for example, we could have implemented a transition from every event to the first event so that sequences terminate at varying times or terminate on particular events (e.g., as in language). The question is now: would a different sequence be predicted in the presence of auditory input?

To illustrate the dual role of generation and recognition, under these models, we repeated the simulation above but presented auditory input, starting with the second call. See Figure 20A. In this instance, the posterior over the initial episode now correctly begins during the onset of the second call and faithfully predicts the auditory input for the duration of its presentation (here, a couple of seconds). After this initial period of stimulation, the auditory input was removed (by delivering outcomes with an imprecise, uniform probability distribution). The RGM effectively treats the stimulus as a “prompt” and pursues its itinerant cycle of song generation.

The preceding illustration shows that content is assimilated and predicted (i.e., generated) online without the need for caching. This follows because the Bayesian belief updating has, by design, a scheduling that eludes backward message passing in time (a.k.a., Bayesian smoothing). This kind of reactive message passing can be conceived of as follows: the highest level sends a prior message to the level below. The level below reacts to this top-down message by performing two updates before returning a likelihood message requested by the higher level. The higher level forms a posterior belief on the basis of the likelihood message and updates its posterior belief about the current and next state, which it then provides as an empirical prior to the level below, and so on. As noted in previous sections, any level can only update its beliefs after waiting for its subordinate levels to accumulate two messages. In short, the posterior beliefs at any given level are retrospective beliefs after accumulating evidence from lower levels over

¹³ Copyright-free files were obtained from the following website: <https://sound-effects.bbcrewind.co.uk/>

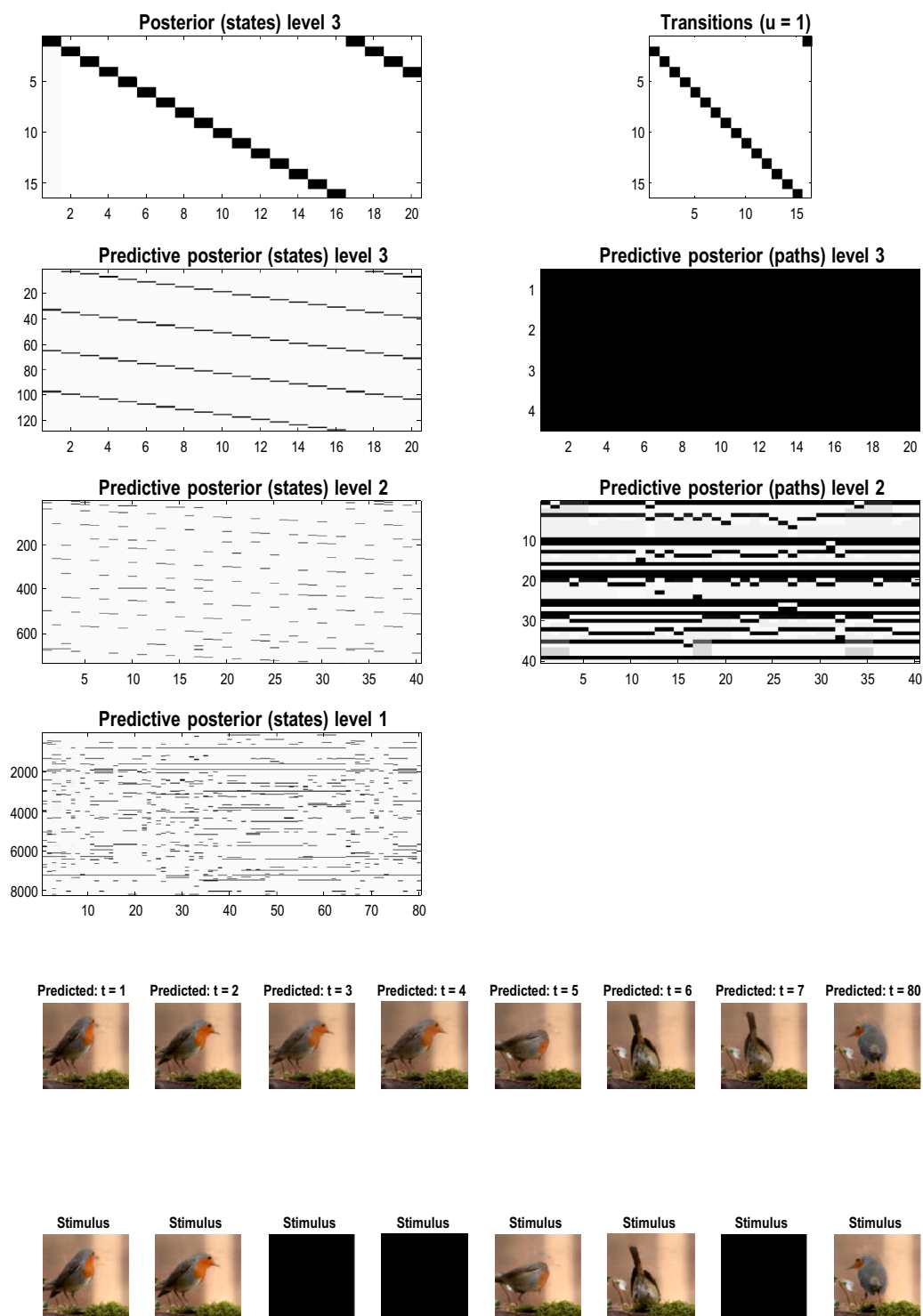


FIGURE 17

Now you see it. Now you do not. This figure uses the same format as Figure 12 to illustrate the recognition and generation of a short movie of a robin comprising 160 video frames (where the last 32 frames are a repeat of the first 32). The reason there are only 80 timesteps on the x-axis for the level 1 predictive posteriors is that these predictions only pertain to the first of a pair of video frames. The implicit loop has been summarized as a simple orbit through 16 events. In this example, a stimulus was presented for the first four frames and then removed for the subsequent four frames. Despite the absence of precise stimuli, the posterior predictions veridically track the motion of the bird during the missing stimulus.

the timescale in question. The ensuing predictive posteriors are then used as priors to provide contextual constraints on the initial states and paths of lower levels and likelihoods for the states at the level above. In

the MATLAB implementation of these demonstration routines, this reactive message passing emerges from the nested composition of function calls. It is interesting to consider how the requisite

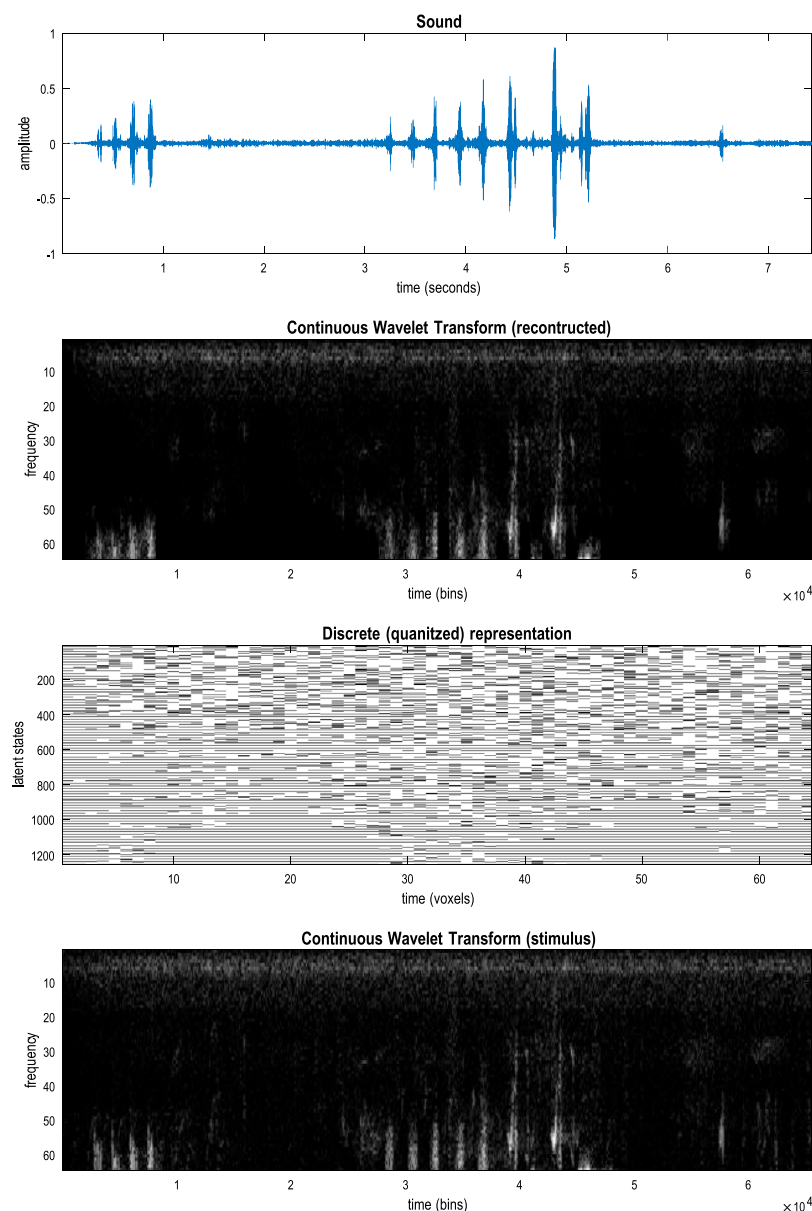


FIGURE 18

Sound images. This figure shows the training data for structure learning. The lower panel shows the continuous wavelet transform (CWT) of a recording of two crossbill bird calls, where each call comprises a crescendo of short chirps. The CWT is shown as an image of time-frequency responses, that is, the spectral power from 1 to 64 frequency bins, as it evolves over time. Strictly speaking, this is not a continuous wavelet transform because the Gaussian envelope of the Morlet wavelets was fixed at 32 m. As such, this is effectively a short-term Fourier transform between 40 Hz and 4,000 Hz (appropriate for the frequency range of human hearing). The second panel shows the equivalent representation generated from the quantized representation in the third panel. The discrete representation is shown as an image of the probability distributions over singular variates associated with the singular vectors (i.e., time-frequency basis sets) used for discretization. The upper panel shows a sound file generated from the reconstructed CWT.

scheduling would be implemented with reactive message passing (Bagaev and de Vries, 2021); c.f., the actor model (Hewitt et al., 1973) (Keith Duggar; personal communication).

From songs to music

One might ask if RGMs could be applied to language, perhaps in the spirit of hierarchical Dirichlet process models, as was done by MacKay and Peto (2008) and Teh et al. (2006). We will not

address this here but provide an application to music to illustrate the generalized synchrony that accompanies communicative exchange (Friston and Frith, 2015). Figure 20B shows the sound file used for fast structure learning. In this example, a short (2 min) sound file of a jazz pianist was discretized to 64 time-frequency voxels (between 40 Hz and 4,000 Hz in 32 bins). The ensuing generation of piano music is shown in Figure 21A, where the RGM has compressed the sound file into 16 events, each corresponding to a bar of music. In this example, the model was exposed to a succession of music segments using

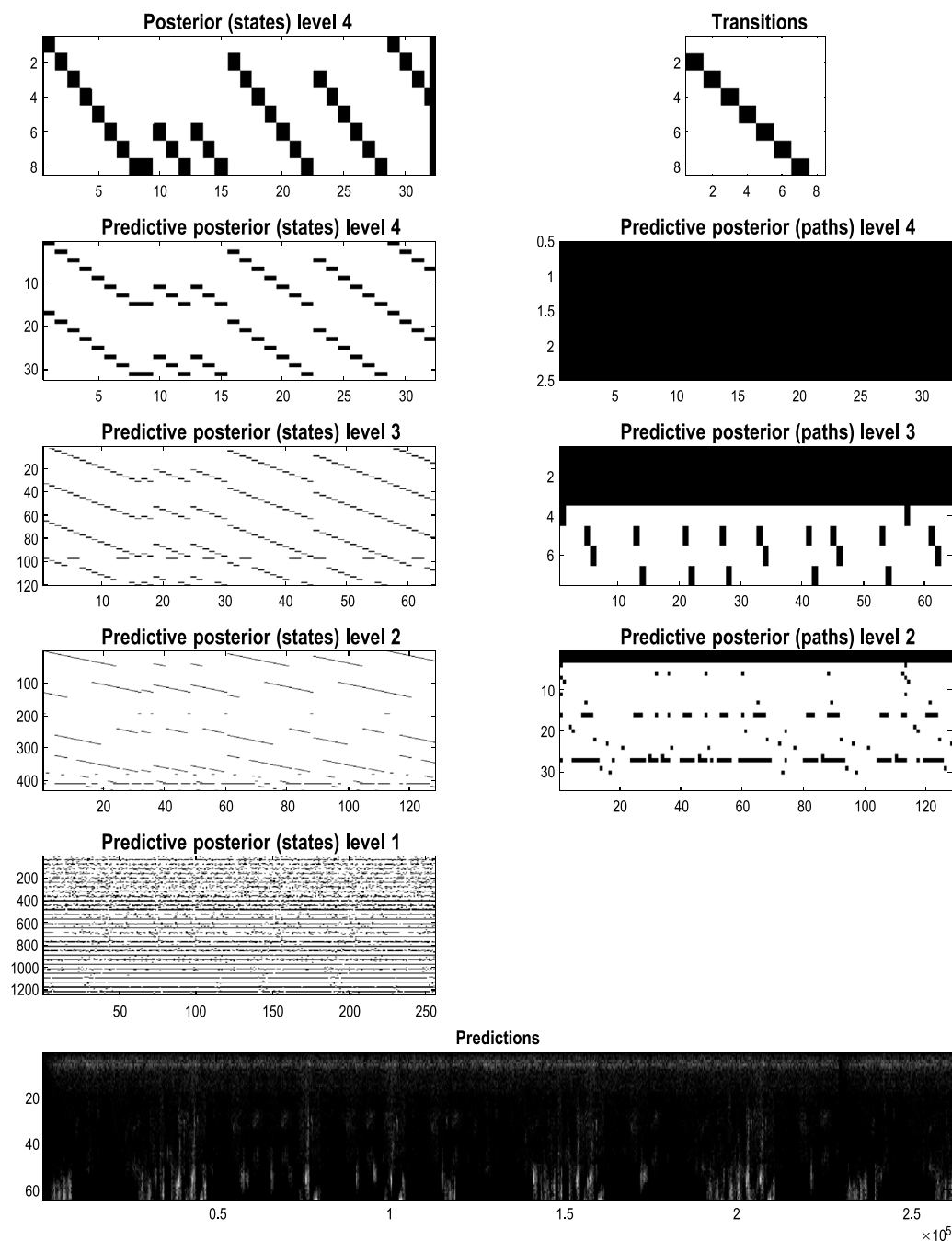


FIGURE 19

Renormalizing birdsong. This figure uses the same format as Figure 17 but displays the posterior predictions as a continuous wavelet transform (i.e., a time–frequency representation of the spectral power at each point in time). In this example, time–frequency voxels covered four frequency bins and 1,024 time bins, corresponding to approximately 100 m (at a sampling rate of 8,820 Hz). Quantization of the ensuing time–frequency voxels used singular variates with five discrete values for a maximum of 16 singular vectors. This example reports the generation of birdsong after compression to a sequence of eight events. Here, the final event could be followed by any preceding event with equal probability. This follows because there was no recurrence of events in the training data used for structure learning. As a consequence, during generation, events cascade to the eighth event and then transition to a preceding event stochastically. The upper left panel shows the resulting succession of events that produce the posterior predictive sequence of bird calls in the lower panel. When played, the resulting sound file is indistinguishable from a bird emitting a variety of stereotypical calls in a quasi-random sequence.

active learning—Equation 7—after fast structure learning (Algorithm 1). In virtue of the time series presented, it learned that the last event (i.e., musical bar) was followed by

the first or eighth bar of music. The RGM now generates extended piano play, selecting successive 8-bar sequences in accord with the learned statistics of variation at this timescale.

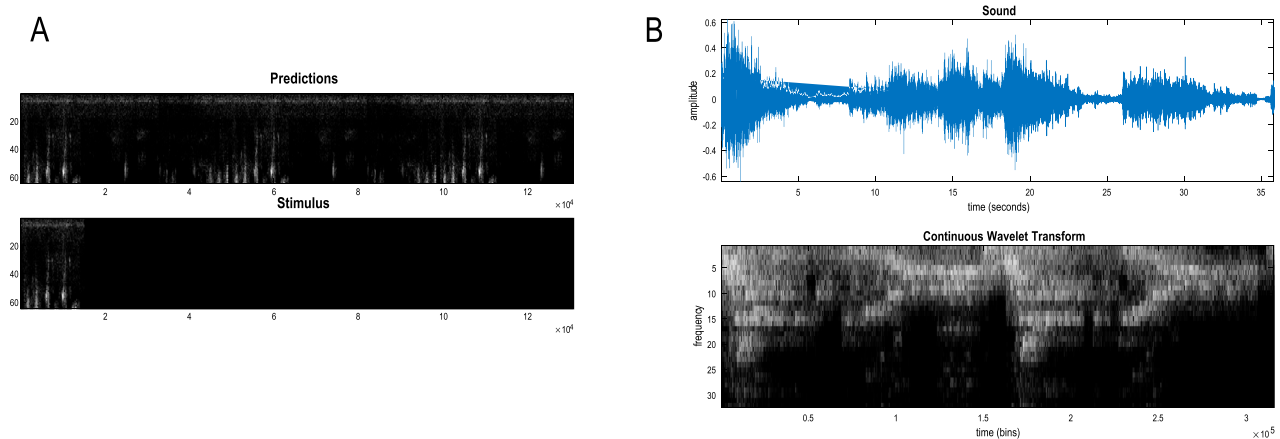


FIGURE 20

(A) Song recognition. These time–frequency representations reproduce the lower panel of Figure 19 but in the context of an initial stimulus or “prompt,” corresponding to the second call in the training data. These posterior predictions show that the model immediately identified the call and its phase and then continued to generate predictions in accord with its model of successive auditory events. In this example, the stimulus was rendered imprecise (i.e., inaudible) after 16 of the 128 time-frequency voxels were generated. **(B)** Jazz music. This panel follows the same format as Figure 18; however, here, the recording is of 36 s of jazz piano, comprising approximately 16 bars. The continuous wavelet transform was quantized into 32 frequency bins between 40 Hz and 4,000 Hz (with a fixed Gaussian envelope of 8 m). The (Nyquist) sample rate was twice the highest frequency considered. The time–frequency representation was quantized using time–frequency voxels of four neighboring frequencies and time bins covering approximately 500 m, corresponding to 1/4 of a musical bar. Following renormalization, musical events at the highest (third) level have a duration of 2.24 s; that is, a bar of music.

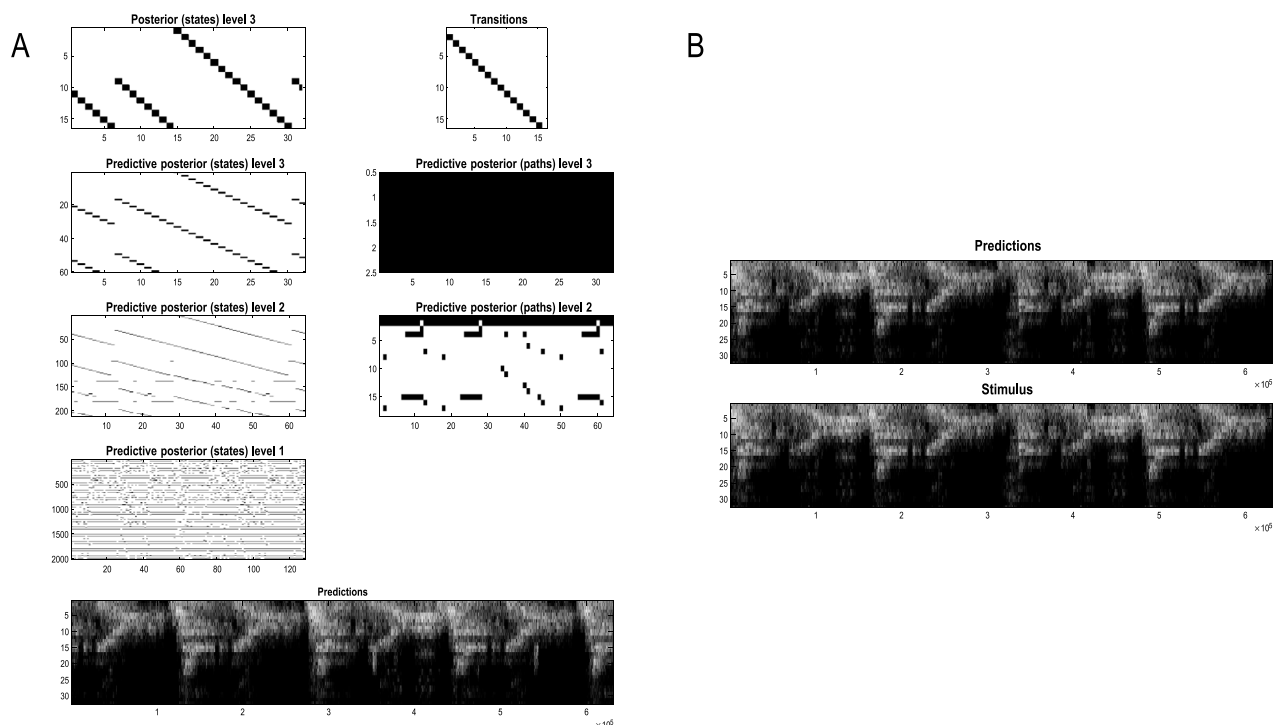


FIGURE 21

(A) Generating music. This figure follows the same format as Figure 19. In this example, each event corresponds to a bar of music, and the simulation reports the generation of 32 bars under the learned transitions shown on the upper right. Here, the model has learned to generate 8 bars of music until the final bar, after which it re-enters at the first or ninth event to pursue its path through event space. In other words, the model generates stochastically alternating 8-bar musical sequences. This particular generative behavior is a simple consequence of what it has heard during structure learning and subsequent active learning. **(B)** Musical accompaniment. This panel illustrates the synchronous entrainment of posterior predictions by stimuli. This entrainment can be read as a generalized synchrony (a.k.a., synchronization of chaos) under a shared musical narrative or generative model.

Figure 21B shows the same simulation during the presentation of the original sound file. The key thing to note is that the predictions are almost instantaneously synchronized with the heard music. If we read the predictions as the sounds that would be generated by an agent and the stimuli as the music generated by an accompanist, we can regard this simulation as a discrete analog of the generalized synchronization that emerges in dyadic interactions under a shared generative model; that is, singing from the same hymn sheet (Friston and Frith, 2015).

From pixels to planning

In this final section, we turn to the deployment of RGMs for planning as inference (Attias, 2003; Botvinick and Toussaint, 2012; Da Costa et al., 2020) and, implicitly, their use as synthetic agents and decision-makers under uncertainty. This application is relatively straightforward under active inference due to its roots in the free energy principle and implicit biomimetic commitments. From the perspective of the free energy principle, agents are read as certain kinds of self-organizing systems that possess characteristic states, which characterize the kind of agent or system they are. The very existence of this attracting set or manifold (a.k.a., pullback attractor) means that one can describe or simulate self-organization as a variational principle of least action (Friston K. et al., 2023). Application of this principle underwrites the variational belief updating of Figure 2.

The notion of an attracting set is useful here because it foregrounds the role of constraints, in the sense there are certain states outside the attracting set that an agent is unlikely to be found in. This means the attracting set can be described in terms of prior preferences, whose logarithm can be interpreted as value (for the agent in question). The negative logarithm corresponds to self-information, providing an information-theoretic account of self-organization in terms of value-pointing dynamics. From the current perspective, self-information is the quantity bounded by variational free energy.

From the perspective of reinforcement learning, one can regard sparse rewards as specifying unstable fixed points on the pullback attractor. This means that if we wanted to learn the structure of a generative model of a particular (expert) agent, we simply must compress the paths of least action that link rewarding states. Equipped with such a model, action, under active inference, simply realizes the predictions under such a model, thereby realizing expert play or Bayes optimal decision-making under uncertainty; see Equation 3.

From a biomimetic perspective, this replaces the notion of motor commands with motor predictions that are fulfilled by peripheral motor reflexes (Adams et al., 2013; Friston, 2011). This view of motor (and autonomic) control inherits from the partitioning of states under the free energy principle. In brief, internal states—for example, of a neuronal network—are separated from external states—for example, of the body or extrapersonal space—by control and sensory states that, together, constitute blanket states. This leads to a form of active inference that can be regarded as control as inference (c.f., model predictive control) in which internal states generate predictions or setpoints that are realized reflexively by active states: c.f., Kappen et al. (2012). Crucially, both internal and control states can be cast as minimizing variational free energy. For internal states, this manifests as inference and learning. For control

states, this reduces to minimizing proprioceptive prediction errors¹⁴ in motor control (Friston et al., 2011) or telemetry prediction errors in drones or other artifacts (Lanillos et al., 2021).

This kind of active (control as) inference can be usefully contrasted with the optimization of state-action policies in reinforcement learning. Figure 22 illustrates this in terms of the differences in computational architectures, which focus on planning as inference (i.e., sequential policy optimization) and learning to plan (i.e., learning state-action policies), respectively.

Planning as inference

Much of reinforcement learning—for example, deep RL—is concerned with the difficult problem of learning state-action policies through learning the parameters of a neural network that maximizes expected reward, where sparse rewards are only available after some suitable sequence of actions (Kaiser et al., 2019). Model-free approaches (such as the one illustrated in Figure 22) learn a mapping from states to actions, while model-based schemes learn a generative model to finesse action selection (Theodorou et al., 2010).

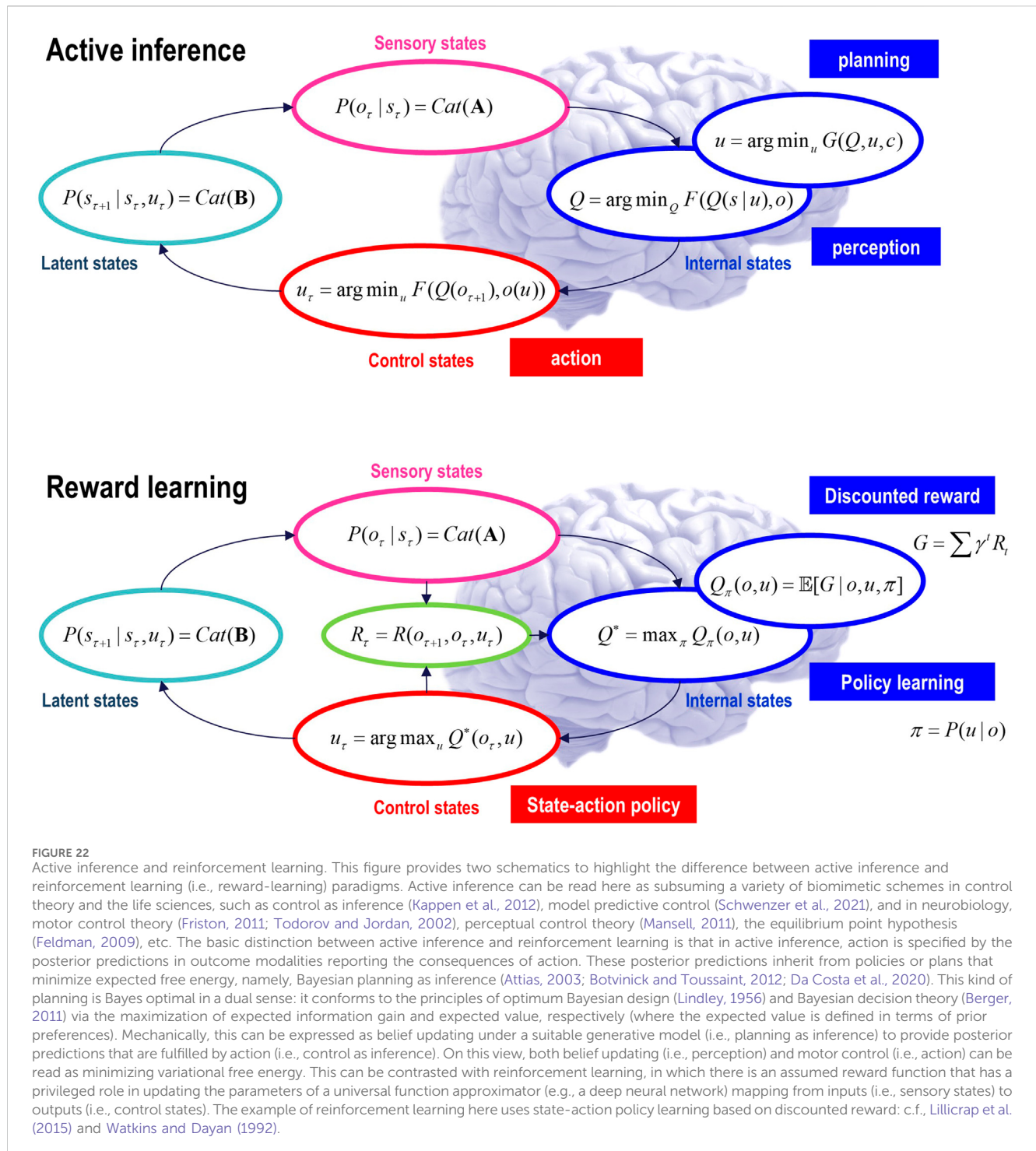
Active model selection offers a different perspective on the requisite learning: instead of updating model parameters following a reward, one can simply update the model *per se*. In other words, one can learn a compressed representation of events that intervene between one reward and the next. This casts reinforcement learning as a structure learning problem, in which the structure is only updated if unique events do not entail a cost: that is, if the training sequence ends with a preferred (rewarded) outcome. Equation 3 formalizes this notion in terms of expected free energy—that is, expected information gain and cost—where selection (c.f., Maxwell's demon) is effectively applied to data used for structure learning, as in Equation 9.

Procedurally, this looks very much like conventional reinforcement learning; however, it is simpler and more efficient because it learns a compressed representation of, and only of, paths that link rewarded states. Another way of looking at this is as a “smart data selection.” Effectively, this leads to the automatic selection of model structures that can only recognize and predict rewarding episodes, thereby eluding subsequent parameter learning. This rests on the fact that one does not need to learn how to maximize reward if one has a generative model that can only predict paths that lead to reward and thereby specify the next action. In what follows, we will illustrate this approach in sequential policy optimization and unpack some of its corollaries.

Games and attractors

To illustrate the use of the RGM for planning as inference, this section uses simple Atari-like games to show how a model of expert play self-assembles, given a sequence of outcomes under random actions. We illustrate the details using a simple game and

¹⁴ Where prediction errors can be read as variational free energy gradients.



then apply the same procedures to a slightly more challenging game.

The simple game in question was a game of Pong, in which the paths of a ball were coarse-grained to 12×9 blocks of 32×32 RGB pixels. A total of 1,024 frames of random play were selected that (i) started from a previously rewarded outcome, (ii) ended in a subsequent hit, and (iii) did not contain any misses. In short, we used rewards for, and only for, data selection. The training frames were selected from 21,280 frames generated under random play. The

sequence of training frames was renormalized to create an RGM. This fast structure learning took approximately 18 s on a personal computer. The resulting generative model is, effectively, a predictor of expert play because it has only compressed paths that intervene between rewarded outcomes. Figure 23 (lower panels) illustrates these high-dimensional orbits by plotting the paths in the first few principal dimensions of the underlying statistical manifold. The rewarded states (hitting the ball) are encircled in red, while the blue lines show the multiplicity of paths that connect rewarded outcomes.

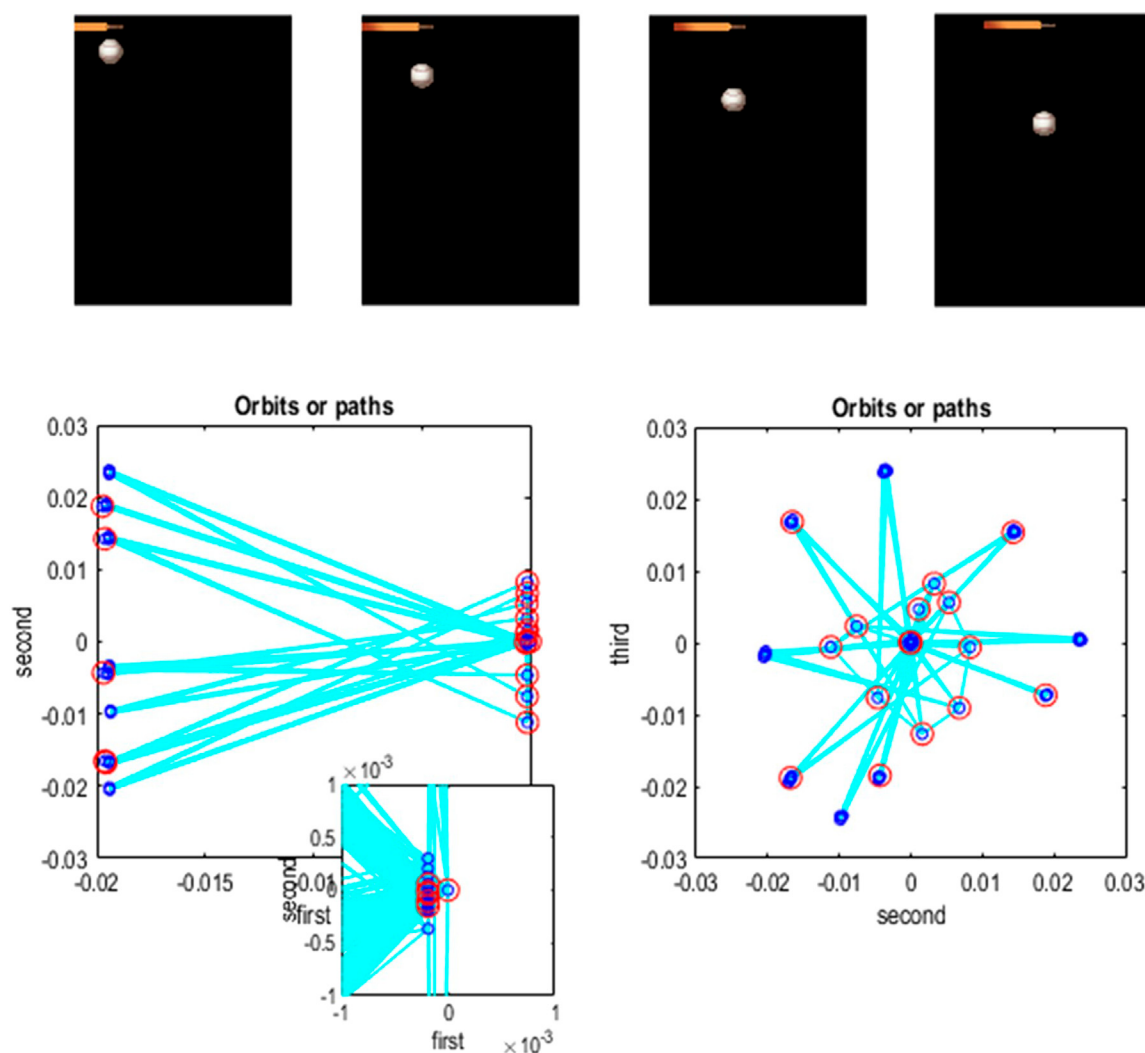


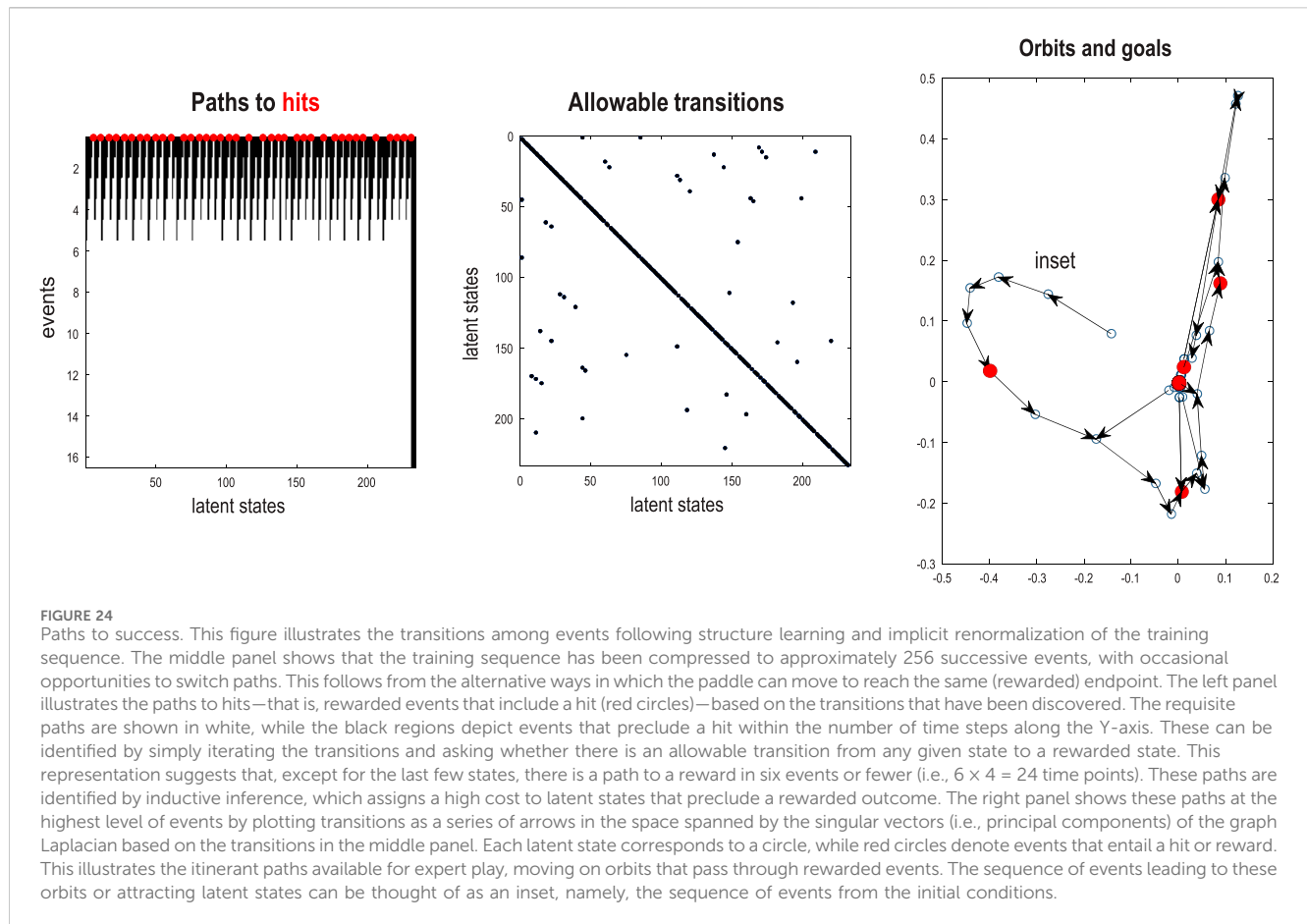
FIGURE 23

Policies and attractors. The upper panels show the first four frames generated by a game engine simulating a simple version of Pong. In this game, the paddle must return a ball that bounces around inside a rectangular box. These images were generated from discrete factors generating (32×32) groups of TrueColor pixels where each (12×9) factor (i.e., location) could be in five states, corresponding to three parts of the paddle, a ball, or background. In the game engine or generative process, the ball simply bounced around with constant momentum, moving from one location to the next at every time step. The paddle could move in either direction by one location or stay still. A training set of such images (in discretized space) was generated by concatenating sequences of random play that intervened between ball hits. When expressed in terms of probability distributions over quantized states, the ensuing trajectory corresponds to an orbit on a high-dimensional statistical manifold (i.e., simplex). The lower panels illustrate the itinerant nature of this orbit in the space spanned by the first two pairs of singular vectors of the associated time series. The inset provides a magnification of the orbit near the origin of the projection. This inset speaks to a self-similar aspect of the transitions among unique points in this quantized (probabilistic) representation. The number of points corresponds to the unique combination of image features in the training set. These points constitute an attracting set that can be learned under an RGM. Rewarded states or configurations are circled in red, illustrating the fact that there are several paths available for getting from one sparse reward to the next (via unrewarded states).

Figure 24 shows how these orbits have been assimilated into the RGM in terms of the transitions among events at the highest (third) hierarchical level, where each event lasts for $4 = 2^{(3-1)}$ time steps at the first level. The RGM compressed all such events encountered during structure learning into slightly less than the total number of events experienced: $233 < 256 = 1,024/4$. The central panel shows unique transitions among events, where each event is followed by one and only one subsequent event. There are three such unique transition matrices. This means that each event can be followed by up to three other events. This

reflects the plurality of paths among rewarding states in Figure 23. In other words, there are several paths from one reward to the next.

Using these allowable transitions, one can identify the events that precede a rewarding event. By repeating this process recursively and retaining proceeding events, one can work backward in time and identify at which point there exists a path to a rewarding event, defined as an event that contains a hit at the first level. The left panel of Figure 24 illustrates this analysis graphically by showing whether reaching a rewarding state within a certain number of events is



possible. In this example, a rewarding event is accessible in six events or fewer from all but the final latent states.

One can leverage this characterization of event-based encoding to implement *inductive inference* (Friston et al., 2023c), namely, an efficient form of active inference for discrete state-space models under precise beliefs about transitions that is reminiscent of intentional behavior (Tschacher and Haken, 2007). In brief, this involves recursive applications of the backward transitions to goal states to identify plausible paths into the future, namely, avoiding the black states in the left panel of Figure 24. This kind of inductive planning effectively defines the paths of least action (i.e., expected free energy) from any given state to our intended or goal state. We will illustrate inductive inference under the RGM above to reproduce expert behavior.

Learning expert behavior

To illustrate inductive (planning as) inference, we equipped the RGM with the prior belief that transitions among the final states (i.e., events) were under its control. This means, *a priori*, it believes that the next event will be on a path to a preferred or rewarded event. These empirical priors are then propagated down the hierarchy to predict the next outcome and drive action, which is selected to minimize variational free energy: see Figure 22 and Equation 1. Because action can only change outcomes, this minimization reduces to maximizing the predictive accuracy of outcomes under

allowable actions. In general, this predictive accuracy pertains to proprioceptive or telemetry data reporting the state of the motor plant (Adams et al., 2013; Friston et al., 2011; Lanillos et al., 2021). Here, for simplicity, we used the upper row of visual inputs reporting the location of the paddle. Note the subtlety of this implementation of planning as inference: from the agent's perspective, it does not explicitly represent action or, indeed, anything that can be overtly controlled. At the highest level, it is simply predicting a path to preferred outcomes, which, unsurprisingly from its point of view, unfolds as anticipated (via control as inference). Anthropomorphically, the agent does not know that its own action realized these predictions or that this realization was operating at a faster timescale than its plans were being conceived. It simply imagines its own future, unaware it is the author of its sensorium.

At this point, one might ask why inductive inference is necessary, given that the RGM can only generate expert play. The answer is that there is a plurality of paths to the next rewarding event. By enabling inductive inference by specifying events that entail a reward, one can be assured that the paths of least action are selected. One could stop here and illustrate the performance of inductive inference following structure learning. In other words, in principle, one can solve reward-learning problems quickly and efficiently using Bayesian model selection and inductive inference without recourse to any learning *per se*. However, we can recall the benefits of continual (active) learning in previous sections, which enable the model to generalize to outcomes that have not been

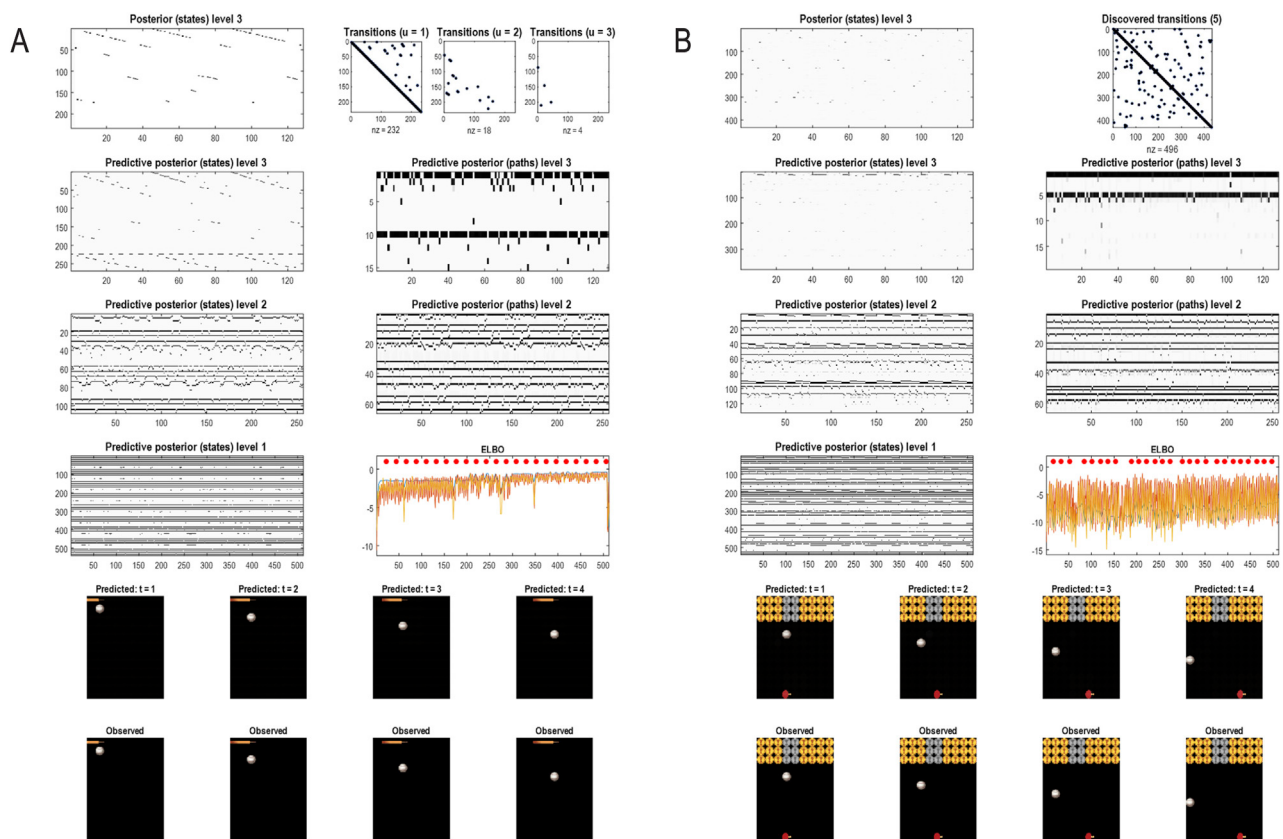


FIGURE 25

(A) Learning expert play. This figure summarizes the results of fast structure learning after exposure to the selected training set. A sequence of 1,024 frames was sampled selectively from 21,280 frames generated with random paddle movement. The ensuing sequence was learned in approximately 18 s on a standard PC. This figure follows the format of previous figures, showing three discovered transitions among certain events, corresponding to the alternative ways in which the paddle moved in the training sequence. This sequence has been summarized in terms of transitions among 233 events. The panel labeled ELBO reports the (negative) variational free energy during continual learning of 512 frames of self-generated play. The red dots correspond to (rewarded) hits, while the colored lines report the ELBO at each level of the RGM. Because the model can only recognize, predict, and thereby realize expert play, the implicit agent never misses the ball (in this example). However, it can learn to become more confident in its (realized) predictions, as evinced by a gradual increase in the ELBO. (B) Breakout. Here, we repeated the analysis using a slightly more complicated game based on Breakout and doubled the number of training frames to 2,048. In this version of Breakout, a reward is recorded whenever the ball hits a row of targets (see lower panels). The row of targets is then removed to expose the underlying row. If a golden target on the final row is hit, the game is reset. Whenever the agent misses a ball or on reset, the paddle is reset to the center, and the ball appears at a fixed height and randomly selected horizontal location around the center. In this example, expert play is confounded by “sticky action,” which means that the movement of the paddle diverges occasionally from the predicted movement. However, the agent recovers quickly and resumes expert play following each miss. This rests on waiting for a recognizable event that is within the attracting set that leads to a reward. As in the previous example, there is a slow increase in confidence with accumulating Dirichlet counts in the likelihood mappings of the RGM. Note that because this game has many more configurations than the previous game of Pong, there are more paths among events; here, there are five such paths (which are shown as discovered transitions by summing over the path dimension of the transition tensor at the final level).

encountered before, namely, outcomes that are generated by a degree of unpredictability or randomness.

Figure 25A shows the result of continual learning over 512 exchanges, using the same format as Figure 17. Crucially, we introduced stochasticity into action selection to confound the otherwise expert play the RGM would exhibit. This was modeled by sampling from the posterior over action, as opposed to selecting the most likely action. In the active inference literature, this is enabled by a “shaky hand” parameter (Parr et al., 2022, p177), in other words, introducing a probabilistic mapping between the agent’s prescribed movement and the actual execution. This mirrors “sticky action” in machine learning benchmarks (Machado et al., 2017). Furthermore, we rendered the likelihood mappings slightly uncertain by adding a small concentration parameter of $(1/128)$ to the Dirichlet

distributions. In principle, the agent should now be able to learn precise dynamics under inductive inference. In turn, this should increase the precision of posterior predictions underwriting action selection, enabling more confident play as the agent becomes more experienced. Figure 25A (the panel labeled ELBO) suggests that this “self-confidence” asymptotes after approximately 400 frames of gameplay.

The upper panels of Figure 25A show the posterior predictions emanating from each of the three levels pertaining to states (left column) and paths (right column). Note that there are no posterior predictions over paths at the first level, which is simply generating images in pixel space. The corresponding images for the first four timesteps are shown in the lower panels in terms of predictions and the stimuli or frames supplied by the game engine. Performance in

terms of hits (red dots) and negative variational free energy (ELBO) are shown in the middle panel on the right. Note that the free energy at each successive level fluctuates more slowly because of the renormalization in time.

The final game

Figure 25B shows the results of the same simulation using a slightly more complicated game based on Breakout. In this simplified game, a row of targets was deleted if, and only if, the targets were hit by a returned ball. In this example, the number of frames selected for structure learning under random action was doubled to 2,048. Otherwise, the active selection, learning, and (inductive) inference were the same as above.

In this example, unlike the Pong example, the game is reset. This allowed the introduction of further stochasticity by replacing the ball to the left or right of the initial location at random. However, these alternative paths from rewarded states are accommodated within the RGM, provided they are experienced as part of the selected training data. Note that this example introduces a slow change in the nature of events as rows of targets are progressively eliminated. In other words, there are fast dynamics as the ball is returned after hit and a slower succession of episodes as rows of targets are eliminated. This can be seen in the upper left panel of Figure 25B, which reports the predictive posteriors from the highest (third) level. Here, certain episodes are repeated regularly but infrequently as the agent revisits the same context whenever there is a reset.

The additional complexity and stochasticity afforded by this game confound as the agent occasionally misses the ball (reported by the gaps in the red dots in the panel labeled ELBO). However, the agent can recover, provided it recognizes an event that leads to a reward. As in the previous illustration, Dirichlet counts are slowly accumulated under largely expert play, as reflected in the slow increases in the ELBO as learning progresses.

Conclusion

This paper has showcased several applications of a generative model for discrete state spaces based on the renormalization group. The applications used to explain and illustrate the accompanying belief updating use exactly the same routines and principles, namely, selecting, learning, and inverting generative models via the minimization of variational and expected free energy. The appeal to the apparatus of the renormalization group is relatively straightforward in this setting. This is because active inference is an application of the free energy principle, which itself is a variational principle of least action, whose functional form is conserved over scales or hierarchical levels (Friston, 2019). One can leverage this to provide a belief updating process that is renormalizable over both space and time. The spatial aspect has been foregrounded by application to content or observations in pixel space, motivating the use of spin-block transformations that have proved successful in related applications in physics and machine learning (Hu et al., 2020; Vidal, 2007). Having said this, the restriction to spin-block transformations and pixel spaces could

also be regarded as a limitation. In principle, any outcome space could be a candidate for coarse graining, provided some tessellation or partition procedure is at hand and can be appropriately motivated in terms of conditional independencies: see Friston et al. (2021b) for a worked example in the context of Markov blankets and the renormalization group.

RGMs with quantized paths may or may not be useful in certain application domains. The applications above illustrate the simplicity and efficiency with which the structure of these models can be learned. Furthermore, their discrete nature lends the requisite belief updating a straightforward form; resting largely on sum-product operators, which can be exact in the sense of Bayesian inference. In turn, this enables efficient forms of planning, such as inductive inference (Friston et al., 2023c). The downside of these models is that their parameterization may not be appropriate for complex system modeling and associated scenario or intervention modeling that is usually predicated on continuous state-space models. For example, dynamic causal models are generally parameterized in terms of rate constants in the context of ordinary or stochastic differential equations. This means that there is a biophysical interpretation to the model parameters that can be lost following discretization or, indeed, discrete switching among dynamical systems. Having said this, it is always possible to convert a continuous state-space model into a master equation, specifying transitions among discrete states: for example, compartmental models of the sort used in biochemistry and epidemiology (Seifert, 2005; Wang, 2009; Wolkenhauer et al., 2004). This speaks to the possibility of learning discrete state-space models using renormalization procedures and then replacing the Dirichlet parameterization with forms parameterized in terms of rate constants and accompanying probability fluxes among compartments or states.

Further limitations of the approach described above rest on the efficiency of the accompanying methods. This efficiency is both a gift and a burden. Clearly, in terms of sample efficiency and compression, the schemes described above are designed to outperform conventional learning schemes. This follows because the model selection procedure is constructive, as opposed to the reductive approach of Bayesian model reduction and related procedures, enabling the self-assembly of a minimally expressive model to recognize, generate, and predict the content at hand. This should be contrasted with the reductive approaches that start with a large, overly expressive model that is then reduced or pruned, minimizing complexity to recover predictive validity and generalization.

However, in committing to a minimally complex, maximally efficient model structure, one is necessarily committed to the kind of data used for model selection and learning. This means such models are necessarily brittle in the sense that they will not recognize or respond to events they have not previously encountered. A useful heuristic here is learning to ride a bike: in this kind of procedural learning, we learn from a starting position and accumulate successive cycles of peddling until we avoid falling completely. In other words, we retain only those experiences characteristic of successful bike riding, until we can ride fluently. However, we would not be able to ride a bike starting from any arbitrary state, for example, having fallen over. We would have to return to a known starting position to recover our flow. Flow is meant in manifold senses here: the fluency associated with being in the flow (Parvizi-

Wayne et al., 2024), control flow (Fields et al., 2023a; b), dynamical flow (Huys et al., 2014; Rabinovich et al., 2012) and, of course, RG flow (Hu et al., 2020).

The approach to renormalized generative modeling outlined in this paper owes much to the work of Hermann Haken and his many contributions to the theoretical context in which the work reported in this paper arose. We conclude here by drawing some specific parallels. First, Haken's work included a focus on the notion of self-organization (Haken, 2006) and the centrality of information—in the Shannon sense—in determining principles for self-organizing systems. Our use of informational (variational free energy) principles in the self-organization of generative models closely follows these ideas. A compelling point of contact between Haken and colleagues' treatment of the free energy principle and the work of Jaynes (Haken and Portugali, 2021b) is provided by the duality between the path integral formulation of the free energy principle (Friston K. et al., 2023) and the principle of maximum caliber (MaxCal) or maximum path entropy principle, introduced in an article entitled: "The Minimum Entropy Production Principle" for non-equilibrium statistical mechanics (Jaynes, 1980). All of these treatments foreground the importance of probability densities over paths or trajectories that underwrite the dynamics of systems in open exchange with their environment or each other. Second, Haken's emphasis on phase transitions as an important element of his synergetic theory (Haken, 1977; Haken et al., 1985; Haken and Portugali, 2021a) has an analogy with our treatment in terms of categorical systems, switching between alternative states. This is perhaps most obvious in the treatment of chaotic systems like the Lorenz attractor above, in which the RGM discovers stochastic switches between alternative phases of the system. Finally, this paper leans heavily into Haken's notion that many physical systems can be described in terms of collaborative subsystems with levels of organization that can be described at different temporal resolutions. In a sense, our paper could be summarized by the idea that *"rather complex phenomena brought about by the cooperation of many subsystems can be understood and described by a few simple concepts"* (Haken, 1975).

Dedication

Patterns become functional because they consume in a most efficient manner the gradients which cause their evolution, thereby making synergetic pattern formation appear "intentional" (Tschacher and Haken, 2007). pp.1.

It is an honor to dedicate this paper to the memory of Hermann Haken—a scientist and gentleman whose intellect and kindness have left an immeasurable legacy. I have many fond memories of interactions with Hermann Haken (Friston, 2017), especially at the Herbstakademien organized by Wolfgang Tschacher, and subsequent (handwritten) correspondence (now in my file of treasures). The work reported below inherits directly from his foundational work (Haken, 1983) and his guidance in applying the free energy principle to self-organization in far-from-equilibrium systems; for example, Haken and Portugali (2016) and Haken and Portugali (2021b): beautifully articulated as

consuming the free energy gradients that create them (Tschacher and Haken, 2007).

Specifically, this paper addresses the issue of scaling in applications of the free energy principle to active inference. In brief, the free energy principle in question licenses a functionalist account of self-organization in terms of inference and learning under a generative model of the sensed world variously referred to as self-evidencing or Bayesian mechanics (Hohwy, 2016; Ramstead et al., 2022). This account rests on the definition of the states of things (i.e., particles) in terms of their (Markov) boundaries, leading to a recursive definition of things as particles of particles, where the states of a particle at one scale supply the (Markov) boundary conditions for the emergence of particles at the scale above. It is this emergence that rests on the synergetics of Hermann Haken (Haken, 1983), right down to the use of eigenfunctions of (Markov) boundary states to supply slow, unstable macroscopic states for superordinate scales (c.f., order parameters) (Friston, 2019). The ensuing cause-effect structure could be regarded as pre-physics, in the sense that quantum, statistical, classical, and Bayesian mechanics can be derived as limit cases. So, why is this important for the current paper?

The current work is a pragmatic application of active inference to generative modeling that deals with scaling issues by appeal to the scale invariance afforded by the recursive application of the slaving principle (Haken, 1996) under the rubric of the renormalization group (Friston K., 2019). In short, if the world behaves according to Haken, then any generative model appropriate for that world will feature the same scale invariance.

Karl J Friston.

Author contributions

KF: conceptualization, formal analysis, writing – original draft, and writing – review and editing. CH: conceptualization, formal analysis, and writing – review and editing. TV: conceptualization, formal analysis, and writing – review and editing. LD: conceptualization, formal analysis, and writing – review and editing. TS: conceptualization, formal analysis, and writing – review and editing. DM: conceptualization, formal analysis, and writing – review and editing. AT: conceptualization, formal analysis, and writing – review and editing. MK: conceptualization, formal analysis, and writing – review and editing. CB: conceptualization, formal analysis, and writing – review and editing. TP: conceptualization, formal analysis, and writing – review and editing.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. KF was supported by funding for the Wellcome Centre for Human Neuroimaging (Ref: 205103/Z/16/Z), a Canada-UK Artificial Intelligence Initiative (Ref: ES/T01279X/1), and the European Union's Horizon 2020 Framework Programme for Research and Innovation under the Specific Grant Agreement No. 945539

(Human Brain Project SGA3). TP is supported by an NIHR Academic Clinical Fellowship (ref: ACF-2023-13-013).

Conflict of interest

Authors KF, CH, TV, LC, TS DM, AT, MK and CB were employed by company VERSES Research Lab.

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

The authors KF, DM and TP declared that they were an editorial board member of Frontiers, at the time of submission. This had no impact on the peer review process and the final decision.

References

- Adams, R. A., Shipp, S., and Friston, K. J. (2013). Predictions not commands: active inference in the motor system. *Brain Struct. Funct.* 218, 611–643. doi:10.1007/s00429-012-0475-5
- Alpers, G. W., and Gerdes, A. B. M. (2007). Here is looking at you: emotional faces predominate in binocular rivalry. *Emotion* 7, 495–506. doi:10.1037/1528-3542.7.3.495
- Angelucci, A., and Bullier, J. (2003). Reaching beyond the classical receptive field of V1 neurons: horizontal or feedback axons? *J. physiology* 97, 141–154. doi:10.1016/j.jphysparis.2003.09.001
- Attias, H. (2003). Planning by probabilistic inference. *Proc. 9th Int. Workshop Artif. Intell. Statistics*.
- Ay, N., Bertschinger, N., Der, R., Guttler, F., and Olbrich, E. (2008). Predictive information and explorative behavior of autonomous robots. *Eur. Phys. J. B* 63, 329–339. doi:10.1140/epjb/e2008-00175-0
- Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K. R., and Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *Plos One* 10, e0130140. doi:10.1371/journal.pone.0130140
- Bagaev, D., and de Vries, B. (2021). Reactive message passing for scalable bayesian inference. *arXiv:2112.2023*, 13251–13326. doi:10.1155/2023/6601690
- Barlow, H. (1961). “Possible principles underlying the transformations of sensory messages,” in *Sensory communication*. Editor W. Rosenblith (Cambridge, MA: MIT Press), 217–234.
- Bastos, A. M., Vezoli, J., Bosman, C. A., Schoffelen, J. M., Oostenveld, R., Dowdall, J. R., et al. (2015). Visual areas exert feedforward and feedback influences through distinct frequency channels. *Neuron* 85, 390–401. doi:10.1016/j.neuron.2014.12.018
- Beal, M. J. (2003). *Variational algorithms for approximate bayesian inference*. University College London. PhD. Thesis.
- Berger, J. O. (2011). *Statistical decision theory and Bayesian analysis*. New York; London: Springer.
- Botvinick, M., and Toussaint, M. (2012). Planning as inference. *Trends Cognitive Sci.* 16, 485–488. doi:10.1016/j.tics.2012.08.006
- Braun, D. A., Ortega, P. A., Theodorou, E., and Schaal, S. (2011). “Path integral control and bounded rationality, adaptive dynamic programming and reinforcement learning (ADPRL),” in *2011 IEEE symposium on. Paris: IEEE*, 202–209.
- Cardy, J. L. (2015). Scaling and renormalization in statistical physics.
- Cugliandolo, L. F., and Lecomte, V. (2017). Rules of calculus in the path integral representation of white noise Langevin equations: the Onsager-Machlup approach. *J. Phys. a-Mathematical Theor.* 50, 345001. doi:10.1088/1751-8121/aa7dd6
- Da Costa, L., Parr, T., Sajid, N., Veselic, S., Neacsu, V., and Friston, K. (2020). Active inference on discrete state-spaces: a synthesis. *J. Math. Psychol.* 99, 102447. doi:10.1016/j.jmp.2020.102447
- Dauwels, J. (2007). “On variational message passing on factor graphs,” in *2007 IEEE international symposium on information theory*, 2546–2550.
- Deyle, E. R., and Sugihara, G. (2011). Generalized theorems for nonlinear state space reconstruction. *PLoS One* 6, e18295. doi:10.1371/journal.pone.0018295
- Feldman, A. G. (2009). New insights into action-perception coupling. *Exp. Brain Res.* 194, 39–58. doi:10.1007/s00221-008-1667-3
- Fields, C., Fabrocini, F., Friston, K., Glazebrook, J. F., Hazan, H., Levin, M., et al. (2023a). Control flow in active inference systems—Part I: classical and quantum formulations of active inference. *IEEE Trans. Mol. Biol. Multi-Scale Commun.* 9, 235–245. doi:10.1109/tmbmc.2023.3272150
- Fields, C., Fabrocini, F., Friston, K., Glazebrook, J. F., Hazan, H., Levin, M., et al. (2023b). Control flow in active inference systems—Part II: tensor networks as general models of control flow. *IEEE Trans. Mol. Biol. Multi-Scale Commun.* 9, 246–256. doi:10.1109/tmbmc.2023.3272158
- Friston, K. (2011). What is optimal about motor control? *Neuron* 72, 488–498. doi:10.1016/j.neuron.2011.10.018
- Friston, K. (2017). “Epistemic patterns – a tribute to Hermann haken,” in *Synergetik als Ordner. Die strukturierende Wirkung der interdisziplinären Ideen Hermann Hakens*. Editors J. Kriz and W. Tschacher (Pabst), 143–146.
- Friston, K. (2019). A free energy principle for a particular physics, eprint arXiv: 1906.10184.
- Friston, K., Da Costa, L., Sakthivadivel, D. A. R., Heins, C., Pavliotis, G. A., Ramstead, M., et al. (2023a). Path integrals, particular kinds, and strange things. *Phys. Life Rev.* 47, 35–62. doi:10.1016/j.plrev.2023.08.016
- Friston, K., Heins, C., Ueltzhoffer, K., Da Costa, L., and Parr, T. (2021a). Stochastic chaos and Markov blankets. *Entropy (Basel)* 23, 1220. doi:10.3390/e23091220
- Friston, K., and Kiebel, S. (2009). Predictive coding under the free-energy principle. *Philosophical Trans. R. Soc. Lond. Ser. B, Biol. Sci.* 364, 1211–1221. doi:10.1098/rstb.2008.0300
- Friston, K., Mattout, J., and Kilner, J. (2011). Action understanding and active inference. *Biol. Cybern.* 104, 137–160. doi:10.1007/s00422-011-0424-z
- Friston, K., Parr, T., and Zeidman, P. (2018). Bayesian model reduction. *arXiv Prepr. arXiv:1805.07092*.
- Friston, K., and Penny, W. (2011). Post hoc Bayesian model selection. *Neuroimage* 56, 2089–2099. doi:10.1016/j.neuroimage.2011.03.062
- Friston, K., Rigoli, F., Ognibene, D., Mathys, C., Fitzgerald, T., and Pezzulo, G. (2015). Active inference and epistemic value. *Cogn. Neurosci.* 6, 187–214. doi:10.1080/17588928.2015.1020053
- Friston, K. J., Da Costa, L., Tschantz, A., Kiefer, A., Salvatori, T., Neacsu, V., et al. (2023b). “Supervised structure learning,” in *arXiv:2311*.
- Friston, K. J., Fagerholm, E. D., Zarghami, T. S., Parr, T., Hipolito, I., Magrou, L., et al. (2021c). Parcels and particles: Markov blankets in the brain. *Netw. Neurosci. Camb. Mass.* 5, 211–251. doi:10.1162/netn_a_00175
- Friston, K. J., and Frith, C. D. (2015). Active inference, communication and hermeneutics. *Cortex; a J. devoted study Nerv. Syst. Behav.* 68, 129–143. doi:10.1016/j.cortex.2015.03.025
- Friston, K. J., Parr, T., and de Vries, B. (2017a). The graphical brain: belief propagation and active inference. *Netw. Neurosci. Camb. Mass.* 1, 381–414. doi:10.1162/NETN_a_00018
- Friston, K. J., Parr, T., Zeidman, P., Razi, A., Flandin, G., Daunizeau, J., et al. (2020). Second waves, social distancing, and the spread of COVID-19 across the USA. *Wellcome Open Res.* 5, 103. doi:10.12688/wellcomeopenres.15986.3
- Friston, K. J., Rosch, R., Parr, T., Price, C., and Bowman, H. (2017b). Deep temporal models and active inference. *Neurosci. Biobehav. Rev.* 77, 388–402. doi:10.1016/j.neubiorev.2017.04.009
- Friston, K. J., Salvatori, T., Isomura, T., Tschantz, A., Kiefer, A., Verbelen, T., et al. (2023c). Active inference and intentional behaviour. *arXiv:2312.07547*.
- Friston, K. J., Stephan, K., Li, B. J., and Daunizeau, J. (2010). Generalised filtering. *Math. Problems Eng.* 2010, 621670. doi:10.1155/2010/621670

Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

Publisher’s note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

- Ghavamzadeh, M., Mannor, S., Pineau, J., and Tamar, A. (2016). Bayesian reinforcement learning: a survey. *arXiv arXiv:1609.04436*.
- Goodale, M. A., Westwood, D. A., and Milner, A. D. (2004). Two distinct modes of control for object-directed action. *Prog. Brain Res.* 144, 131–144. doi:10.1016/s0079-6123(03)14409-3
- Gros, C. (2009). Cognitive computation with autonomously active neural networks: an emerging field. *Cogn. Comput.* 1, 77–90. doi:10.1007/s12559-008-9000-9
- Haken, H. (1975). Cooperative phenomena in systems far from thermal equilibrium and in nonphysical systems. *Rev. Mod. Phys.* 47, 67–121. doi:10.1103/revmodphys.47.67
- Haken, H. (1977). Synergetics. *Phys. Bull.* 28, 412–414. doi:10.1088/0031-9112/28/9/027
- Haken, H. (1983). “Synergetics: an introduction. Non-equilibrium phase transition and self-organization in physics,” in *Chemistry and biology*. Berlin: Springer Verlag.
- Haken, H. (1996). Slaving principle revisited. *Phys. D. Nonlinear Phenom.* 97, 95–103. doi:10.1016/0167-2789(96)00080-2
- Haken, H. (2006). *Information and self-organization: a macroscopic approach to complex systems*. Springer Science and Business Media.
- Haken, H., Kelso, J. A. S., and Bunz, H. (1985). A theoretical model of phase transitions in human hand movements. *Biol. Cybern.* 51, 347–356. doi:10.1007/BF00336922
- Haken, H., and Portugali, J. (2016). Information and selforganization: a unifying approach and applications. *Entropy* 18, 197. doi:10.3390/e18060197
- Haken, H., and Portugali, J. (2021a). Information and self-organization II: steady state and phase transition. *Entropy* 23, 707. doi:10.3390/e23060707
- Haken, H., and Portugali, J. (2021b). “Relationships. Bayes, Friston, Jaynes and synergetics 2nd foundation, synergetic cities: information, steady state and phase transition: implications to urban scaling,” in *Smart cities and planning*. Cham: Springer International Publishing, 85–104.
- Hasson, U., Yang, E., Vallines, I., Heeger, D. J., and Rubin, N. (2008). A hierarchy of temporal receptive windows in human cortex. *J. Neurosci.* 28, 2539–2550. doi:10.1523/JNEUROSCI.5487-07.2008
- Hewitt, C., Bishop, P., and Steiger, R. (1973). “A universal modular ACTOR formalism for artificial intelligence,” in *Proceedings of the 3rd international joint conference on Artificial intelligence*. Stanford, USA: Morgan Kaufmann Publishers Inc., 235–245.
- Higgins, I., Chang, L., Langston, V., Hassabis, D., Summerfield, C., Tsao, D., et al. (2021). Unsupervised deep learning identifies semantic disentanglement in single inferotemporal face patch neurons. *Nat. Commun.* 12, 6456. doi:10.1038/s41467-021-26751-5
- Hochstein, S., and Ahissar, M. (2002). View from the top: hierarchies and reverse hierarchies in the visual system. *Neuron* 36, 791–804. doi:10.1016/s0896-6273(02)01091-7
- Hohwy, J. (2016). The self-evidencing brain. *Nous* 50, 259–285. doi:10.1111/nous.12062
- Howard, R. (1966). Information value theory. *IEEE Trans. Syst. Sci. Cybern.* SSC-2, 22–26. doi:10.1109/tssc.1966.300074
- Hu, H.-y., Wu, D., You, Y.-Z., Olshausen, B. A., and Chen, Y. (2020). RG-Flow: a hierarchical and explainable flow model based on renormalization group and sparse prior. *Mach. Learn. Sci. Technol.* 3, 035009. doi:10.1088/2632-2153/ac8393
- Huys, R., Perdikis, D., and Jirsa, V. K. (2014). Functional architectures and structured flows on manifolds: a dynamical framework for motor behavior. *Psychol. Rev.* 121, 302–336. doi:10.1037/a0037014
- Itti, L., and Baldi, P. (2009). Bayesian surprise attracts human attention. *Vis. Res.* 49, 1295–1306. doi:10.1016/j.visres.2008.09.007
- Jaynes, E. T. (1980). The minimum entropy production principle. *Annu. Rev. Phys. Chem.* 31, 579–601. doi:10.1146/annurev.pc.31.100180.003051
- Kaiser, L., Babaeizadeh, M., Milos, P., Osinski, B., Campbell, R. H., Czechowski, K., et al. (2019). Model-based reinforcement learning for Atari, p. arXiv:1903.00374.
- Kaluza, P., and Meyer-Ortmanns, H. (2010). On the role of frustration in excitable systems. *Chaos* 20, 043111. doi:10.1063/1.3491342
- Kappen, H. J., Gomez, V., and Opper, M. (2012). Optimal control as a graphical model inference problem. *Mach. Learn.* 87, 159–182. doi:10.1007/s10994-012-5278-7
- Kerr, W. C., and Graham, A. J. (2000). Generalized phase space version of Langevin equations and associated Fokker-Planck equations. *Eur. Phys. J. B* 15, 305–311. doi:10.1007/s100510051129
- Kersten, D., Mamassian, P., and Yuille, A. (2004). Object perception as Bayesian inference. *Annu. Rev. Psychol.* 55, 271–304. doi:10.1146/annurev.psych.55.090902.142005
- Kiebel, S. J., Daunizeau, J., and Friston, K. J. (2008). A hierarchy of time-scales and the brain. *PLoS Comput. Biol.* 4, e1000209. doi:10.1371/journal.pcbi.1000209
- Klimovskaia, A., Lopez-Paz, D., Bottou, L., and Nickel, M. (2020). Poincaré maps for analyzing complex hierarchies in single-cell data. *Nat. Commun.* 11, 2966. doi:10.1038/s41467-020-16822-4
- Klyubin, A. S., Polani, D., and Nehaniv, C. L. (2005). Empowerment: a universal agent-centric measure of control. *Proc. CEC* 1, 128–135. doi:10.1109/cec.2005.1554676
- Lanillos, P., Meo, C., Pezzato, C., Meera, A. A., Baioumy, M., Ohata, W., et al. (2021). Active inference in robotics and artificial agents: survey and challenges. *arXiv:2112.01871*.
- LeCun, Y., and Cortes, C. (2005). The mnist database of handwritten digits.
- Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., et al. (2015). Continuous control with deep reinforcement learning. *arXiv e-prints*. arXiv:1509.02971.
- Lin, H. W., Tegmark, M., and Rolnick, D. (2017). Why does deep and cheap learning work so well? *J. Stat. Phys.* 168, 1223–1247. doi:10.1007/s10955-017-1836-5
- Linderman, S. W., Miller, A. C., Adams, R. P., Blei, D. M., Paninski, L., and Johnson, M. J. (2016). Recurrent switching linear dynamical systems. *eprint arXiv:1610.08466*, 1–15.
- Lindley, D. V. (1956). On a measure of the information provided by an experiment. *Ann. Math. Statistics* 27, 986–1005. doi:10.1214/aoms/1177728069
- Linsker, R. (1990). Perceptual neural organization - some approaches based on network models and information-theory. *Annu. Rev. Neurosci.* 13, 257–281. doi:10.1146/annurev.ne.13.030190.001353
- Livingstone, M., and Hubel, D. (1988). Segregation of form, color, movement, and depth: anatomy, physiology, and perception. *Science* 240, 740–749. doi:10.1126/science.3283936
- Lorenz, E. N. (1963). Deterministic nonperiodic flow. *J. Atmos. Sci.* 20, 130–141. doi:10.1175/1520-0469(1963)020<0130:dnf>2.0.co;2
- Ma, Y., Tan, Q. J., Yuan, R. S., Yuan, B., and Ao, P. (2014). Potential function in a continuous dissipative chaotic system: decomposition scheme and role of strange attractor. *Int. J. Bifurcation Chaos* 24, 1450015. doi:10.1142/s0218127414500151
- Machado, M. C., Bellemare, M. G., Talvitie, E., Veness, J., Hausknecht, M., and Bowling, M. (2017). Revisiting the arcade learning environment: evaluation protocols and open problems for general agents, p. arXiv:1709.06009.
- MacKay, D. J. C. (2003). *Information theory, inference and learning algorithms*. Cambridge: Cambridge University Press.
- MacKay, D. J. C., and Peto, L. C. B. (2008). A hierarchical Dirichlet language model. *Nat. Lang. Eng.* 1, 289–308. doi:10.1017/s1351324900000218
- Mansell, W. (2011). Control of perception should be operationalized as a fundamental property of the nervous system. *Top. Cogn. Sci.* 3, 257–261. doi:10.1111/j.1756-8765.2011.01140.x
- Marković, D., Reiter, A. M. F., and Kiebel, S. J. (2022). Revealing human sensitivity to a latent temporal structure of changes. *Front. Behav. Neurosci.* 16, 962494. doi:10.3389/fnbeh.2022.962494
- Mehta, P., and Schwab, D. J. (2014). *An exact mapping between the variational renormalization group and deep learning*, 3831. arXiv:1410.
- Muckli, L., De Martino, F., Vizioli, L., Petro, L. S., Smith, F. W., Ugurbil, K., et al. (2015). Contextual feedback to superficial layers of V1. *Curr. Biol.* 25, 2690–2695. doi:10.1016/j.cub.2015.08.057
- Olier, I., Trujillo-Barreto, N. J., and El-Deredey, W. (2013). A switching multi-scale dynamical network model of EEG/MEG. *NeuroImage* 83, 262–287. doi:10.1016/j.neuroimage.2013.04.046
- Olshausen, B. A., and Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature* 381, 607–609. doi:10.1038/381607a0
- Ørstavik, S., and Stark, J. (1998). Reconstruction and cross-prediction in coupled map lattices using spatio-temporal embedding techniques. *Phys. Lett. A* 247, 145–160. doi:10.1016/s0375-9601(98)00541-6
- Ortega, P. A., and Braun, D. A. (2013). Thermodynamics as a theory of decision-making with information-processing costs. *Proc. R. Soc. A* 469, 20120683. doi:10.1098/rspa.2012.0683
- Parr, T., Pezzulo, G., and Friston, K. J. (2022). *Active inference: the free energy principle in mind, brain, and behavior*. Cambridge: MIT Press.
- Parvizi-Wayne, D., Sandved-Smith, L., Pitliya, R. J., Limanowski, J., Tufft, M. R. A., and Friston, K. J. (2024). Forgetting ourselves in flow: an active inference account of flow states and how we experience ourselves within them. *Front. Psychol.* 15, 1354719. doi:10.3389/fpsyg.2024.1354719
- Pefkou, M., Arnal, L. H., Fontolan, L., and Giraud, A. L. (2017). θ -Band and β -band neural activity reflects independent syllable tracking and comprehension of time-compressed speech. *J. Neurosci.* 37, 7930–7938. doi:10.1523/JNEUROSCI.2882-16.2017
- Poland, D. (1993). Cooperative catalysis and chemical chaos - a chemical-model for the Lorenz equations. *Phys. D.* 65, 86–99. doi:10.1016/0167-2789(93)90006-m
- Rabinovich, M. I., Afraimovich, V. S., Bick, C., and Varona, P. (2012). Information flow dynamics in the brain. *Phys. Life Rev.* 9, 51–73. doi:10.1016/j.plrev.2011.11.002
- Rabinovich, M. I., Afraimovich, V. S., and Varona, P. (2010). Heteroclinic binding. *Int. J.* 25, 433–442. doi:10.1080/14689367.2010.515396

- Ramstead, M. J. D., Sakthivadivel, D. A. R., Heins, C., Koudahl, M., Millidge, B., Da Costa, L., et al. (2022). On bayesian mechanics: a physics of and by beliefs, arXiv:2205.11543.
- Sakthivadivel, D. A. R. (2022). "Weak Markov blankets in high-dimensional," in *Sparsely-coupled random dynamical systems*. arXiv:2207.
- Sanchez, E. H., Serrurier, M., and Ortner, M. (2019). Learning disentangled representations via mutual information estimation, p. arXiv:1912.03915.
- Savage, L. J. (1954). *The foundations of statistics*. New York: Wiley.
- Schmidhuber, J. (1991). Curious model-building control systems. *Proc. Int. Jt. Conf. Neural Netw. Singap.* 2, 1458–1463 vol.2. doi:10.1109/ijcnn.1991.170605
- Schwabl, F. (2002). *Phase transitions, scale invariance, renormalization group theory, and percolation, statistical mechanics*. Berlin Heidelberg, Berlin, Heidelberg: Springer, 327–404.
- Schwenzer, M., Ay, M., Bergs, T., and Abel, D. (2021). Review on model predictive control: an engineering perspective. *Int. J. Adv. Manuf. Technol.* 117, 1327–1349. doi:10.1007/s00170-021-07682-3
- Seifert, U. (2005). Entropy production along a stochastic trajectory and an integral fluctuation theorem. *Phys. Rev. Lett.* 95, 040602. doi:10.1103/PhysRevLett.95.040602
- Sengupta, B., and Friston, K. (2018). How robust are deep neural networks? arXiv:1804.11313.
- Simoncelli, E. P., and Olshausen, B. A. (2001). Natural image statistics and neural representation. *Annu. Rev. Neurosci.* 24, 1193–1216. doi:10.1146/annurev.neuro.24.1.1193
- Smith, R., Schwartenbeck, P., Parr, T., and Friston, K. J. (2020). An active inference approach to modeling structure learning: concept learning as an example case. *Front. Comput. Neurosci.* 14, 41. doi:10.3389/fncom.2020.00041
- Still, S., and Precup, D. (2012). An information-theoretic approach to curiosity-driven reinforcement learning. *Theory Biosci. = Theor. den Biowiss.* 131, 139–148. doi:10.1007/s12064-011-0142-z
- Takens, F. (1980). *Detecting strange attractors in turbulence*. Groningen: Rijksuniversiteit Groningen. Mathematisch Instituut.
- Tani, J., and Nolfi, S. (1999). Learning to perceive the world as articulated: an approach for hierarchical learning in sensory-motor systems. *Neural Netw.* 12, 1131–1141. doi:10.1016/s0893-6080(99)00060-x
- Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2006). Hierarchical dirichlet processes. *J. Am. Stat. Assoc.* 101, 1566–1581. doi:10.1198/016214506000000302
- Tenenbaum, J. B., Kemp, C., Griffiths, T. L., and Goodman, N. D. (2011). How to grow a mind: statistics, structure, and abstraction. *Science* 331, 1279–1285. doi:10.1126/science.1192788
- Tervo, D. G. R., Tenenbaum, J. B., and Gershman, S. J. (2016). Toward the neural implementation of structure learning. *Curr. Opin. Neurobiol.* 37, 99–105. doi:10.1016/j.conb.2016.01.014
- Theodorou, E. A., Buchli, J., and Schaal, S. (2010). A generalized path integral control approach to reinforcement learning. *J. Mach. Learn. Res.* 11, 3137–3181.
- Tipping, M. E. (2001). Sparse Bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.* 1, 211–244.
- Todorov, E., and Jordan, M. I. (2002). Optimal feedback control as a theory of motor coordination. *Nat. Neurosci.* 5, 1226–1235. doi:10.1038/nn963
- Tomasello, M. (2016). Cultural learning redux. *Child. Dev.* 87, 643–653. doi:10.1111/cdev.12499
- Tschacher, W., and Haken, H. (2007). Intentionality in non-equilibrium systems? The functional aspects of self-organized pattern formation. *New Ideas Psychol.* 25, 1–15. doi:10.1016/j.newideapsych.2006.09.002
- Ungerleider, L. G., and Mishkin, M. (1982). "Two cortical visual systems," in *Analysis of visual behavior*. Editors D. Ingle, M. A. Goodale, and R. J. W. Mansfield (Cambridge, MA: MIT Press), 549–586.
- van den Broek, J. L., Wiegerinck, W. A. J. J., and Kappen, H. J. (2010). Risk-sensitive path integral control. *UAI* 6, 1–8.
- Van Dijk, S. G., and Polani, D. (2013). Informational constraints-driven organization in goal-directed behavior. *Adv. Complex Syst.* 16, 1350016. doi:10.1142/s0219525913500161
- Vidal, G. (2007). Entanglement renormalization. *Phys. Rev. Lett.* 99, 220405. doi:10.1103/PhysRevLett.99.220405
- Wang, X. M. (2009). *From Dirac notation to probability bracket notation: time evolution and path integral under wick rotations*. ArXiv e-prints.
- Watkins, C. J. C. H., and Dayan, P. (1992). Q-Learning. *Mach. Learn.* 8, 279–292. doi:10.1023/a:1022676722315
- Watson, J. D., Onorati, E., and Cubitt, T. S. (2022). Uncomputably complex renormalisation group flows. *Nat. Commun.* 13, 7618. doi:10.1038/s41467-022-35179-4
- Winn, J., and Bishop, C. M. (2005). Variational message passing. *J. Mach. Learn. Res.* 6, 661–694.
- Wolkenhauer, O., Ullah, M., Kolch, W., and Cho, K. H. (2004). Modeling and simulation of intracellular dynamics: choosing an appropriate framework. *IEEE Trans. Nanobioscience* 3, 200–207. doi:10.1109/tnb.2004.833694
- Yildiz, I. B., von Kriegstein, K., and Kiebel, S. J. (2013). From birdsong to human speech recognition: bayesian inference on a hierarchy of nonlinear dynamical systems. *PLoS Comput. Biol.* 9, e1003219. doi:10.1371/journal.pcbi.1003219
- Zeki, S., and Shipp, S. (1988). The functional logic of cortical connections. *Nature* 335, 311–317. doi:10.1038/335311a0