

# A Dynamic Bayesian Model of Homeostatic Control

Will Penny<sup>1</sup> and Klaas Stephan<sup>2</sup>

<sup>1</sup> Wellcome Trust Centre for Neuroimaging,  
University College London, 12 Queen Square, UK.

w.penny@ucl.ac.uk,

WWW home page: <http://www.fil.ion.ucl.ac.uk/~wpenny>

<sup>2</sup> Translational Neuromodeling Unit,  
Institute for Biomedical Engineering, University of Zurich and ETH Zurich.  
Wilfriedstrasse 6 CH-8032 Zurich

**Abstract.** This paper shows how a planning as inference framework with discrete latent states can be used to implement homeostatic control by providing an agent with multivariate autonomic set points as goals. Before receiving these goals the agent navigates according to the ‘Prior Dynamics’ which embody a cognitive map of the environment. Given the goals, optimal value functions are implicitly computed using a forward and backward message passing algorithm, which is then used to construct the ‘Posterior Dynamics’. We propose that this formalism provides a useful description of computations in the mammalian Hippocampus.

**Keywords:** Planning as Inference, Homeostasis, Hippocampus

## 1 Introduction

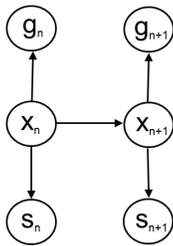
In previous work we have shown how an autonomous agent can be specified using a Hidden Markov Model, and that probabilistic inference in that model can be used to instantiate the operations of localisation and planning [12]. Importantly, both of these operations rely on the same underlying algorithm; belief propagation using forward and backward message passing. The only difference is that sensory inputs are upregulated during localisation whereas goal inputs are upregulated during planning.

The operations of localisation and planning are naturally addressed together as both require access to the same underlying environmental model or ‘cognitive map’. Our inference approach naturally allows uncertainty from localisation to be incorporated into planning. Moreover, localisation and planning are both thought to engage the hippocampus [11,7,14].

Our overall approach conforms to a ‘planning as inference’ perspective in which sequential decision making problems that have previously been the domain of Reinforcement Learning (RL) and dynamic programming, have been recast as problems of statistical inference [1]. More specifically, our HMM agent uses a cognitive map of its environment and its decisions are based on this model. This

is to be contrasted with the state-action mappings that are learnt in RL. The two approaches to decision making may more generally be referred to as model-based and model-free control [13] and are thought to have different neuronal substrates [4].

One interesting recent development in RL has been the replacement of scalar reward signals with homeostatic goals [9]. This incorporates the simple notion that an agent’s behavioral objective is dependent on its current autonomic state. Thus, food rewards are more important when an agent is hungry. In this paper we show how the HMM framework can be used to implement homeostatic control [3] by providing an agent with multivariate autonomic set points, rather than binary goals. Our overall approach thus provides a mechanism for model-based homeostatic control.



**Fig. 1.** The agent’s generative model. This corresponds to an HMM with two sets of observations; goals,  $g_n$ , and sensory inputs,  $s_n$ .

## 2 Methods

We consider a dynamical system evolving over time points  $n = 1..N$  with discrete latent states  $x_n$ , goals  $g_n$  and sensory states  $s_n$ . The overall generative model is shown in Figure 1 and is fully specified via the definition of three probability distributions (i) the state transition density  $p(x_{n+1}|x_n)$ , (ii) the sensory observation density  $p(s_n|x_n)$  and (iii) the goal observation density  $p(g_n|x_n)$ .

Inference is implemented in two separate phases (i) planning and (ii) localisation. In the planning phase, goals are provided and the posterior distribution over latent states is computed. At this time sensory states are either not provided or their influence on planning is eliminated, for example, by reducing sensory precision.

In the localisation phase, sensory observations are provided and the posterior distribution over latent states is computed. In this phase goals are either not provided or their influence on localisation is eliminated, for example, by reducing goal precision. This paper focusses on planning, as localisation has been dealt with in a previous publication [12].

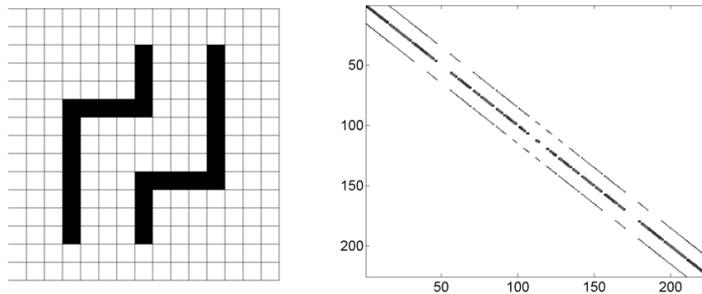
In what follows each of the  $k = 1..K$  discrete latent states is associated with a location in a 2D environment,  $l_k$ , and  $\mathbf{N}(x; \mu, \Sigma)$  denotes a multivariate Gaussian distribution over  $x$  with mean  $\mu$  and covariance  $\Sigma$ .

## 2.1 Prior Dynamics

In the planning phase the agent is given information about task goals. Prior to observing these goals the hidden states evolve according to Markovian dynamics

$$p(x_{n+1} = i | x_n = j) = A_{ij} \quad (1)$$

where  $A$  is a  $K \times K$  state transition matrix. Although high-dimensional this matrix is sparse, reflecting the spatial structure of an environment and allowed transitions within it, as shown in Figure 2. We refer to the above equation as describing the ‘Prior Dynamics’.



**Fig. 2.** Left Panel: The environment contains  $K = 15 \times 15 = 225$  discrete states, with black squares denoting forbidden locations. At each time step agents may move to cardinal neighbours or remain in the same position but cannot transit across edges of the domain (eg top to bottom). Right Panel: The prior dynamics are embodied in a state transition matrix  $A$  of dimension  $225 \times 225$  having a highly sparse structure reflecting the allowed transitions in an environment.

## 2.2 Probabilistic Goals

The probability of observing goal  $g_n$  given state  $x_n$  is given by the density  $p(g_n | x_n)$ . This formulation has many interesting properties. First, goals are inherently probabilistic; one specifies the likelihood of obtaining a goal at a given location. The simplest instantiation of this is the case of binary goals

$$p(g_n | x_n = k) = r_k \quad (2)$$

where  $r_k$  denotes the probability of reward at location  $k$ .

More interestingly, the goal signal  $g_n$  can be multivariate, allowing an agent to have multiple simultaneous goals. Here, the observed goal signal can be used to specify set points and thereby implement a system for instantiating homeostatic control. For example, if we have

$$p(g_n|x_n = k, a) = \mathbf{N}(g_n; a + a_k, C) \quad (3)$$

where  $g_n$  is the homeostatic set point (the multivariate goal),  $C$  encodes the allowed (co-)variance around that set point,  $a$  is the agent’s autonomic state, and  $a_k$  denotes the change in autonomic state (per unit time) that will accrue from visiting state  $k$ .

If the autonomic variables (eg. body water, glucose, temperature) have their own dynamics, the posterior dynamics,  $q$ , (see below) will need to be recomputed to account for this. The posterior dynamics could be updated at satiety, at every time point, or according to some other regime. In this paper, we choose to update them periodically, after a fixed number of time points. We consider linear autonomic dynamics

$$a_n = Ba_{n-1} + a_k \quad (4)$$

where  $B$  encodes the relevant time scales and  $a_k$  is the change accrued from visiting state  $k$ . This dynamical process is external to the HMM agent (loosely speaking, it is instantiated in the agent’s body).

### 2.3 Posterior Dynamics

We now consider an agent being in receipt of a goal signal  $g$ . In order to specify to the agent that this goal is to be reached within a ‘time horizon’ of  $N$  steps we set the sequence of observation variables  $g_n = g$  for  $n = 1..N$ . We denote this sequence as  $G_N = \{g_1, g_2, \dots, g_N\}$ .

The dynamics of the agent after receiving the goal signal, or the ‘Posterior Dynamics’, are defined as

$$\begin{aligned} q_{ij} &\equiv p(x_{n+1} = i|x_n = j, a, G_N) \\ &= \frac{p(x_{n+1} = i|x_n = j)p(x_n = j|a, G_N)}{\sum_{i'=1}^K p(x_{n+1} = i'|x_n = j)p(x_n = j|a, G_N)} \end{aligned} \quad (5)$$

Note also the dependence on the autonomic state,  $a$ . In this paper we set  $a$  to the autonomic state observed just prior to computing the posterior dynamics (but see Discussion). An agent following the posterior dynamics implements goal-directed navigation, whilst one following the prior dynamics merely obeys the physics of a given environment, and its motion within it.

### 2.4 State Posterior

The posterior dynamics constitute a reweighting of the prior dynamics by the density  $p(x_n = j|a, G_N)$ . This density can be computed using standard inference

algorithms such as the alpha-beta recursions [2]. This requires a forward sweep to compute

$$\alpha(x_n) = p(g_n|x_n, a) \sum_{x_{n-1}} p(x_n|x_{n-1})\alpha(x_{n-1}) \quad (6)$$

with  $\alpha(x_1 = k) = p(x_1 = k)p(g_1|x_1 = k, a)$ , and a backward sweep to compute

$$\beta(x_n) = \sum_{x_{n+1}} p(g_{n+1}|x_{n+1}, a)p(x_{n+1}|x_n)\beta(x_{n+1}) \quad (7)$$

with  $\beta(x_N = k) = 1$ . We then have

$$p(x_n = j|a, G_N) = \frac{\alpha(x_n = j)\beta(x_n = j)}{\sum_k \alpha(x_n = k)\beta(x_n = k)} \quad (8)$$

To avoid numerical underflow [2] we scale the forward and backward messages by  $\sum_k \alpha(x_n = k)$ . The forward ‘alpha’ recursions embody the prior distribution and provide a normalisation factor for the backward ‘beta’ recursions. It is also worth noting that both the alpha and beta recursions implicitly make use of prediction errors, as the Gaussian goal densities take on higher values with smaller prediction errors between the set point and predicted autonomic state. In equations 6 to 9,  $n$  is a virtual time index that organises the planning computations. We hypothesise that these are instantiated within a hippocampal ripple (see Discussion).

## 2.5 Trajectories

A known state at time  $n$  is equivalent to a probability distribution  $p(x_n)$  with unit mass at  $x_n = k$  and zero elsewhere. A probabilistic planning trajectory can then be found by integrating the posterior dynamics from this initial distribution. The probability mass at time point  $n + 1$  is

$$p(x_{n+1} = i) = \sum_{k=1}^K q(x_{n+1} = i|x_n = k)p(x_n = k) \quad (9)$$

The state density at subsequent time points can be computed as

$$p(x_{n+m} = i) = \sum_{k=1}^K q(x_{n+m} = i|x_{n+m-1} = k)p(x_{n+m-1}) \quad (10)$$

or in matrix form as

$$p(x_{n+m+1}) = Q^m p(x_{n+1}) \quad (11)$$

Iteration of this equation produces ‘goal-directed flows’ and individual paths to goal are produced by sampling from these flows.

## 2.6 KL Control

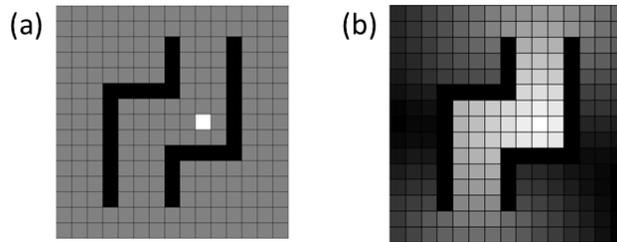
The state posterior can alternately be expressed as

$$p(x_n = j|a, G_N) = \frac{p(G_N|x_n = j, a)p(x_n = j)}{\sum_{j'=1}^K p(G_N|x_n = j', a)p(x_n = j')} \quad (12)$$

Given a uniform prior  $p(x_n = j)$  the equation for the Posterior Dynamics reduces to

$$q_{ij} = \frac{p(x_{n+1} = i|x_n = j)p(G_N|x_n = j, a)}{\sum_{i'=1}^K p(x_{n+1} = i'|x_n = j)p(G_N|x_n = j, a)} \quad (13)$$

Equation 13 corresponds to the ‘Active Dynamics’ of KL control, and  $\log p(G_N|x_n = j, a)$  to the ‘Optimal Value’ function [16]. The ‘Passive Dynamics’ of KL control then correspond to our ‘Prior Dynamics’. The scaling of the beta recursions in the HMM implementation (see above) is analagous to the normalisation used in the power method for computing the Optimal Value function [16].



**Fig. 3. Time to Goal** The figure shows the state posterior,  $p(x_1|G_N)$ , for four different times to goal (a)  $N = 1$ , (b)  $N = 1024$ . The goal location is  $[10, 8]$ . Under a uniform prior,  $p(x_1)$ , these plots correspond to the exponent of the Optimal Value function of KL control.

## 3 Results

This section refers to videos showing goal directed flows. These are available from [http://www.fil.ion.ucl.ac.uk/~wpenny/ica14\\_movies/](http://www.fil.ion.ucl.ac.uk/~wpenny/ica14_movies/).

### 3.1 Binary Goals

Figure 2 shows an example 2D environment and the state transition matrix,  $A$ , corresponding to the prior dynamics associated with it. Here  $A_{ij}$  has been set to  $1/N_j$  if a transition is allowed from  $j$  to  $i$ , with  $N_j$  being the number of allowable transitions from  $j$  (fewer next to boundaries and corners). This includes transitions from a state to itself. Transitions are not allowed to or from wall or edge locations (we set  $0/0 = 0$ , as per usual).

Figure 3 (a) now superimposes a binary goal onto this environment. The goal observation density  $p(g_n|x_n)$  is set so that the probability density is 1 at the goal (white square in Figure 3a) and zero elsewhere. We then computed the state posterior for different values for the time to goal  $N$ . This computation is implemented using the alpha-beta recursions in equations 6 to 8. For  $N = 1$  the state posterior has a single peak at the goal. The spatial gradient of the posterior at sites remote from the goal is zero for  $N = 1$ , but increases with  $N$ . This gradient informs the posterior dynamics via equation 5, allowing a path to be found from remote sites to the goal. The posterior dynamics,  $q$ , were then computed from equation 5 using  $N = 1024$  with the goal at  $[10, 8]$ . Note that for large  $N$ , we have  $p(x_1|G_N) \approx p(x_2|G_N) \approx p(x_3|G_N)$  etc., so we can simply use  $p(x_1|G_N)$  in equation 5.

The movie `known_15_1.avi` shows the state density evolving according to the posterior dynamics. The initial state density is a delta function with unit probability mass at location  $[15, 1]$  and zero elsewhere. The state density at subsequent time points has been computed using equation 11. We now keep the goal at the same location, hence do not change the posterior dynamics  $q$ , but move the initial position to  $[2, 8]$ . The evolution of the state density is shown in the movie `known_2_8.avi`.

Finally, we keep the goal at the same location but update the prior dynamics,  $A$ , to account for a small change to the environment. This hole in a wall appearing at location  $[6, 10]$  requires a change to only four elements of  $A$  (reciprocally between  $[6, 10]$  and  $[6, 9]$ , and  $[6, 10]$  and  $[6, 11]$ ). The posterior dynamics were recomputed based on this new prior and the movie `hole.avi` shows the goal-directed flow from position  $[2, 8]$ .

The above results show that changes in goal location are accommodated by recomputing the posterior dynamics,  $q$ . Small changes in the environment are readily accommodated by small changes in the prior dynamics (and updating  $q$ ). These nonstationarities are less gracefully accommodated in RL which requires extensive relearning of state-action mappings.

### 3.2 Homeostatic Goals

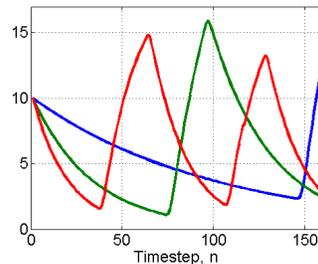
This simulation considers three autonomic variables reflecting the levels of body glucose, water and temperature. The set point is given by  $g = [10, 10, 10]$  with covariance  $C = 0.5I_3$ . Here the autonomic dynamics are set by specifying a diagonal transition matrix,  $B$ , with entries 0.99, 0.97, 0.95. These numbers reflect the rate at which body water, food and temperature levels are depleted. The changes in autonomic state (per unit time) afforded by visiting state  $k$  are set as follows

$$a_k(i) = \exp(-0.5\|s_k - \mu(i)\|^2) \quad (14)$$

where  $s_k$  is the location of the  $k$ th latent variable (place cell) and  $\mu(i)$  denote the spatial locations with maximal affordance for increases in water, glucose and temperature respectively. This implements affordances as a continuous but local function of space; other eg. discrete forms are of course possible. We set

$\mu(1) = [6, 12]^T$ ,  $\mu(2) = [10, 8]^T$  and  $\mu(3) = [10, 4]^T$ . Thus, for example, the agent receives a unit increase in glucose at location  $[10, 8]$  and a smaller amount at neighbouring locations.

The movie `auto.avi` shows the agent navigating according to the posterior dynamics, which are recomputed every 32 steps. This requires a forward and backward pass to compute the posterior state density, based on the autonomic state of the agent at that time point. Figure 5 shows a snapshot of this movie at time step  $n = 65$  and figure 4 plots the time series of autonomic state variables over all time points.



**Fig. 4. Autonomic Variables** *Body temperature (red), glucose (green) and water (blue). The state is initialised to the set point  $[10, 10, 10]$ . Increases correspond to the agent visiting locations which afford increases in body temperature, glucose, temperature and water respectively. Decreases reflect the time scales of depletion encoded in matrix  $B$ .*

## 4 Discussion

This paper has described a simple algorithm, based on inference in an HMM, that an agent can use for localisation, planning and homeostatic control. We propose that it provides a useful computational-level description of aspects of Hippocampal function and associated networks. There are several appealing features.

First, the use of a discrete rather than a continuous latent space allows multimodal posteriors to be supported using simple, exact inference. This is necessary for solving the problem of localisation, as shown in previous work [12]. This is to be contrasted with the approximate inference procedures (particle filtering etc.) required for nonlinear continuous latent space models [13].

Second, discretisation of an otherwise continuous state space is generally unworkable for generic control problems because of the curse of dimensionality. However, spatial navigation is inherently two dimensional so discretisation is tenable.



The resulting system is similar to, and inspired by, the free-energy principle which is founded on the concepts of active inference and homeostasis [6]. Our approach is marked out by its use of discrete latent variables and backwards message passing, and we have previously proposed that these messages are instantiated in ripple activity [13].

In this paper, planning is based on the agent’s autonomic state just prior to computation of the posterior dynamics. Its homeostatic control mechanism is therefore reactive rather than predictive. However, if the agent were also endowed with a predictive autonomic model (cf. equation 4) planning could be based on these predicted autonomic states [15].

## References

1. H. Attias. Planning by probabilistic inference. In C Bishop and B Frey, editors, *Proceedings of the 9th International Conference on Artificial Intelligence and Statistics*, 2003.
2. C.M. Bishop. *Pattern Recognition and Machine Learning*. Springer, New York, 2006.
3. W. Cannon. Organization For Physiological Homeostasis. *Physiol Rev.* 9: 399-431, 1929.
4. N. Daw, Y Niv, and P Dayan. Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. *Nat Neurosci*, 8(12):1704–1711, Dec 2005.
5. D Foster and M. Wilson. Hippocampal theta sequences. *Hippocampus*, 17(11):1093–1099, 2007.
6. K Friston, J Kilner, and L Harrison. A free energy principle for the brain. *J Physiol Paris*, 100(1-3):70–87, 2006.
7. D Hassabis and E Maguire. The construction system of the brain. *Philosophical Transactions of the Royal Society London B*, 364:1263–71, 2009.
8. A Johnson and A Redish. Hippocampal replay contributes to within session learning in a temporal difference reinforcement learning model. *Neural Netw*, 18(9):1163–1171, Nov 2005.
9. M Keramati and B Gutkin. A reinforcement learning theory for homeostatic regulation. In *Neural Information Processing Systems*, 2011.
10. R Muller and M Stead. Hippocampal place cells connected by Hebbian synapses can solve spatial problems. *Hippocampus*, 6:709–719, 1997.
11. J. O’Keefe and L Nadel. *The Hippocampus as a Cognitive Map*. Oxford University Press, 1978.
12. W. Penny. Simultaneous localisation and mapping. In *4th International Workshop on Cognitive Information Processing, Copenhagen, Denmark*, 2014.
13. W. Penny, P. Zeidman, and N. Burgess. Forward and Backward Inference in Spatial Cognition. *PLoS CB*, 9(12):e1003383, 2013.
14. B Pfeiffer and D Foster. Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, 497:74–79, 2013.
15. P. Sterling. Allostasis: a model of predictive regulation. *Physiology and Behaviour*, 106:5-15, 2012.
16. E. Todorov. Efficient computation of optimal actions. *Proceedings National Academy of Sciences*, 106(28):11478–11483, 2009.