Deriving Friston's DEM

Sebastian Bitzer Burak I. Yildiz

May 26, 2013

Abstract

In recent years Karl Friston has prominantly promoted his idea of free energy in the neurosciences. Although there it is highly received, many more technically minded scientists are a bit astranged, because his technical arguments are hard, or impossible, to follow.

Friston's free energy principle has two sides: 1) filtering for estimating hidden states in a hierarchical model and 2) control by performing actions which minimise the discrepancy between expectations and observations. In more general terms we can call these 1) perception and 2) action. Here, we want to shed some light on the technical details of 1). These have been presented in [1, 2] in the difficult Fristonian way. We will try to fill in the gaps while maintaining a consistent mathematical formulation. As this is partly an interpretation of the Fristonian descriptions we do not guarantee to represent his ideas completely, but we hope to provide a formulation which is more accessible to mathematically minded researchers. Also, this report should be seen as a work-in-progress where the date listed on the title page acts as a version number.

We neither want to praise nor discard Friston's ideas. Rather, we aim to provide a consistent clarification of the technical background which is as objective as possible.

Contents

$x \in \mathbb{R}$	scalar
$\mathbf{x} \in \mathbb{R}^{m \times 1}$	column vector
$x_i \in \mathbb{R}$	i-th element of vector $\mathbf{x}$
$\mathbf{X} \in \mathbb{R}^{m \times n}$	matrix
$\mathbf{x}_j = \mathbf{x}_{:,j} \in \mathbb{R}^{m \times 1}$	j-th column of matrix $\mathbf{X}$
$\mathbf{x}_{i,:} \in \mathbb{R}^{1 \times n}$	i-th row of matrix $\mathbf{X}$
$x_{ij} \in \mathbb{R}$	element in i-th row and j-th column of matrix $\mathbf{X}$
$\frac{\partial x}{\partial \mathbf{y}} = [\frac{\mathrm{d}x}{\mathrm{d}y_1}, \ldots, \frac{\mathrm{d}x}{\mathrm{d}y_m}]^\mathsf{T}$	derivative of scalar with respect to vector
$\frac{\partial \mathbf{x}}{\partial y} = [\frac{\mathrm{d}x_1}{\mathrm{d}y}, \ldots, \frac{\mathrm{d}x_m}{\mathrm{d}y}]$	derivative of vector with respect to scalar

Table 1: Matrix notation. Unless stated otherwise we use denominator notation for matrix derivatives (variable with respect to which we take the derivative is in the first dimension of the result).

1 Introduction and Notation

Throughout this report we try to maintain a maximally consistent notation. This is made difficult by the large amount of variables and their different, but interrelated meanings. Here we define the basic notation with reference to subsequent sections where the corresponding variables are first introduced. We use matrix notation as shown in Tab. 1, but also use subscripts to differentiate different variables which do not necessarily belong to a common matrix. It should be clear from the context which notation is used.

Following Friston we only consider Gaussian distributions and denote that the multidimensional random variable $\mathbf{x}$ is distributed according to a multivariate Gaussian distribution as $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ where $\boldsymbol{\mu}$ is the mean vector, $\boldsymbol{\Sigma}$ the covariance matrix and $\boldsymbol{\Pi} = \boldsymbol{\Sigma}^{-1}$ the precision. The corresponding probability density function is

$$p(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{\det(2\pi\boldsymbol{\Sigma})}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\mathsf{T} \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})\right) \tag{1}$$

which we may also write as

$$p(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{Z(\boldsymbol{\Sigma})} \exp\left(-\frac{1}{2}\boldsymbol{\epsilon}^\mathsf{T} \boldsymbol{\Pi} \boldsymbol{\epsilon}\right) \tag{2}$$

where we have defined the 'prediction error' $\boldsymbol{\epsilon} = \mathbf{x} - \boldsymbol{\mu}$.

In this report we consider hierarchical dynamic models represented in continuous time. However, most people find it easier to think of dynamic (probabilistic) models in discrete time. We therefore think that it is instructive to start with a discrete time representation and see how Friston's continuous time dynamic models relate to that. Fig. 1 depicts a discrete time hierarchical dynamic model and introduces some of the variables we consider. In particular, we denominate observed variables at the bottom of the hierarchy as $\mathbf{y}$. These only depend on the hidden, time-dependent state variables $\mathbf{z}^1$ in the first level which in turn depend on output variables of the 2nd level $\mathbf{v}^2$ which result from dynamics defined on state variables $\mathbf{z}^2$ and so on (see also Sec. 4.5).

Most of the theoretical concepts treated in this report do not depend on hierarchical models. Hence, we mostly consider only a single level with observations

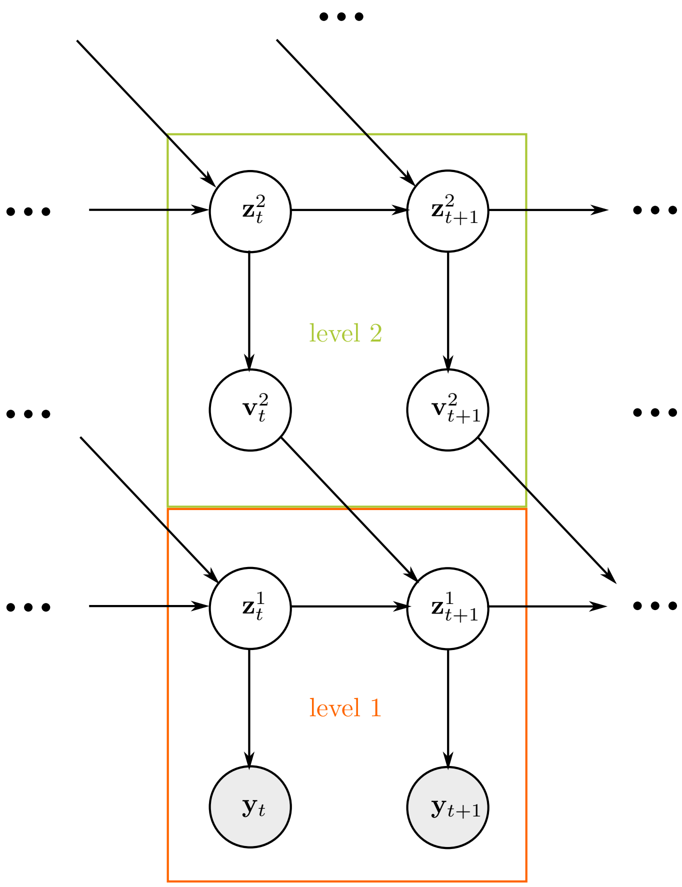

Figure 1: Incomplete 2-level graphical model in discrete time. Observed variables are shaded. Here we let the output of one level influence the dynamics on the level below only in the next time step, but it is equally conceivable that the effect from one level to the other is immediate which also corresponds to the situation in continuous time where time steps are infinitesimally small.

$\mathbf{y}$ and dynamic hidden states $\mathbf{z}$. For linear dynamical systems for which

$$\mathbf{z}_t = \mathbf{A}\mathbf{z}_{t-1} + \boldsymbol{\omega}_t, \tag{3}$$

where $\boldsymbol{\omega}_t$ is Gaussian noise, the Kalman filter can be applied to infer hidden states $\mathbf{z}$ from observations $\mathbf{y}$. Friston's DEM has the same aim, but it is formulated in continuous time, for nonlinear dynamical systems and additionally considers learning of the model parameters which we usually denote as $\boldsymbol{\theta}$.

This report is structured as follows: First, we introduce Friston's continuous time dynamic probabilistic models and explain his generalised coordinates. Then, we consider what Friston calls the 'internal action' of a model which is, roughly, the integrated joint data-hidden state log-probability whose gradients are used for state inference and learning. We then get to Friston's variational approximation which separates inference of hidden time-dependent states from that of constant parameters (learning) and introduce the Laplace approximation used to compute the expectations over the inferred state posteriors. Subsequently, we derive the optimisation problem used to solve parameter learning. In Sec. 8 we discuss the resulting filter algorithm and explain how generalised coordinates improve its performance. We close by considering implementational issues such as numerical integration and discretisation.

2 Discrete Time Dynamical Systems

We consider a multivariate, nonlinear, single level, discrete time dynamical system defined via the state space equations

$$\mathbf{z}_t = f(\mathbf{z}_{t-1}, \boldsymbol{\theta}_z) + \boldsymbol{\omega}_t^z \tag{4}$$
$$\mathbf{y}_t = g(\mathbf{z}_t, \boldsymbol{\theta}_y) + \boldsymbol{\omega}_t^y \tag{5}$$

where f, g are general, nonlinear functions of the hidden states $\mathbf{z}$ with parameters $\boldsymbol{\theta}_z, \boldsymbol{\theta}_y$ and the $\boldsymbol{\omega}$ are Gaussian noise variables centred at 0 and with covariance $\boldsymbol{\Sigma}_z, \boldsymbol{\Sigma}_y$.

The joint density over observations and hidden states is then defined as

$$p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta}) = p(\mathbf{y}_1|\mathbf{z}_1)p(\mathbf{z}_1)\prod_{t=2}^{T} p(\mathbf{y}_t|\mathbf{z}_t)p(\mathbf{z}_t|\mathbf{z}_{t-1}) \tag{6}$$

where we consider time points $t = 1 \dots T$, matrices $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_T]^\mathsf{T}$ and $\mathbf{Z} = [\mathbf{z}_1, \dots, \mathbf{z}_T]^\mathsf{T}$ collect the observed and state variables across time and the densities are Gaussian with $\mathbf{y}_t \sim \mathcal{N}(g(\mathbf{z}_t, \boldsymbol{\theta}_y), \boldsymbol{\Sigma}_y)$ and $\mathbf{z}_t \sim \mathcal{N}(f(\mathbf{z}_{t-1}, \boldsymbol{\theta}_z), \boldsymbol{\Sigma}_z)$. To complete the model, we add a Gaussian prior over parameters and obtain the complete joint density

$$p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\theta}) = p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta})p(\boldsymbol{\theta}). \tag{7}$$

The primary aim of inference in this system is learning: What are the parameters $\boldsymbol{\theta}_z, \boldsymbol{\theta}_y$ of an observed system in the world? But, using a variational approach [3], we will see that this entails also inferring the hidden states $\mathbf{Z}$ which corresponds to perception as it tries to estimate the current state of the real system in the world under the assumption that the real system is equal to our model. Both, parameters and hidden states, are therefore states which need to be inferred. The difference is only that parameters are stable over the observation period while hidden states change. Consequently, the resulting algorithm should provide online inference for hidden states while parameters only need to be updated after certain fixed time points (e.g., at time $t = T$).

2.1 Variational Approximation

We apply Variational Bayesian EM [3] to the state space model defined above. This means that we consider the log marginal likelihood of the model

$$\log p(\mathbf{Y}) = \log \iint p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\mathbf{Z}d\boldsymbol{\theta} \tag{8}$$

which we lower-bound using Jensen's inequality

$$\log p(\mathbf{Y}) \geq \iint q(\mathbf{Z}, \boldsymbol{\theta}) \log \frac{p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{q(\mathbf{Z}, \boldsymbol{\theta})} d\mathbf{Z}d\boldsymbol{\theta} = F(q). \tag{9}$$

We then let the variational density q factorise as $q(\mathbf{Z}, \boldsymbol{\theta}) = q_z(\mathbf{Z})q_\theta(\boldsymbol{\theta})$ which constitutes the actual approximation, because we know that $F(q) = \log p(\mathbf{Y})$, if $q(\mathbf{Z}, \boldsymbol{\theta}) = p(\mathbf{Z}, \boldsymbol{\theta}|\mathbf{Y})$. Consequently, the approximation is based on the assumption that the posterior factorises $p(\mathbf{Z}, \boldsymbol{\theta}|\mathbf{Y}) = p(\mathbf{Z}|\mathbf{Y})p(\boldsymbol{\theta}|\mathbf{Y})$. Applying variational calculus then yields

$$q_\theta(\boldsymbol{\theta}) \propto p(\boldsymbol{\theta}) \exp\left(\langle \log p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta})\rangle_{q_z}\right) \tag{10}$$

and

$$q_z(\mathbf{Z}) \propto \exp\left(\langle \log p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta})\rangle_{q_\theta}\right). \tag{11}$$

These two equations would need to be iterated in order to increase the lower bound on the log marginal likelihood $\log p(\mathbf{Y})$. Then, the variational density $q_z(\mathbf{Z})q_\theta(\boldsymbol{\theta})$ would come closer to the true posterior $p(\mathbf{Z}, \boldsymbol{\theta}|\mathbf{Y})$ (subject to the assumed conditional indepence of $\mathbf{Z}$ and $\boldsymbol{\theta}$ given $\mathbf{Y}$). However, we cannot actually evaluate the expectations inside the exponentials for the general models defined in Eqs. 4 and 5, i.e., we additionally need to approximate these expectations and give $q_z(\mathbf{Z}), q_\theta(\boldsymbol{\theta})$ a form we can handle. This approximation follows next.

2.2 Laplace approximation

With the Laplace approximation we assume that the variational densities q_z, q_θ are Gaussian with means $\boldsymbol{\mu}_z$, $\boldsymbol{\mu}_\theta$ and covariances $\boldsymbol{\Sigma}_{\mu_z}, \boldsymbol{\Sigma}_{\mu_\theta}$. In other words, we approximate the (variational) posteriors $p(\mathbf{Z}|\mathbf{Y})$ and $p(\boldsymbol{\theta}|\mathbf{Y})$ using their mean and covariance only. As a result, we only need to find modes of Eqs. 10, 11 and estimate the curvature around these modes which will provide means and covariances of the variational densities q_θ and q_z.

The Laplace approximation (e.g. [4, p. 255]) is based on a 2nd order Taylor series approximation around the mode of a considered density. We will use this approach twice: 1) to approximate the expectations using a given q_θ, q_z and 2) as the actual Laplace approximation to estimate the Gaussian form of q_θ, q_z.

Let us consider 1) first. We want to approximate the expectations $\langle \log p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta})\rangle_{q_z}$ and $\langle \log p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta})\rangle_{q_\theta}$. Following Friston we call $\mathrm{U} = \log p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta})$ the 'internal' or 'Gibbs energy'. When estimating q_θ in Eq. 10 we consider $\mathrm{U}(\mathbf{Z})$ to be a function of the hidden states such that the expectation becomes $\langle \mathrm{U}(\mathbf{Z}) \rangle_{q_z}$. Similarly, when estimating q_z in Eq. 11 we consider $\mathrm{U}(\boldsymbol{\theta})$ to be a function of the parameters such that the expectation becomes $\langle \mathrm{U}(\boldsymbol{\theta}) \rangle_{q_\theta}$. We proceed with the parameters and Taylor expand $\mathrm{U}(\boldsymbol{\theta})$ around the mean of q_θ:

$$\mathrm{U}(\boldsymbol{\theta}) \approx \mathrm{U}(\boldsymbol{\mu}_\theta) + (\boldsymbol{\theta} - \boldsymbol{\mu}_\theta)^\mathsf{T} \mathrm{U}_{\boldsymbol{\theta}}(\boldsymbol{\mu}_\theta) + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\mu}_\theta)^\mathsf{T} \mathrm{U}_{\boldsymbol{\theta\theta}}(\boldsymbol{\mu}_\theta)(\boldsymbol{\theta} - \boldsymbol{\mu}_\theta) \tag{12}$$

where $\mathrm{U}_{\boldsymbol{\theta}}(\boldsymbol{\mu}_\theta) = \frac{\partial \mathrm{U}}{\partial \boldsymbol{\theta}}(\boldsymbol{\mu}_\theta)$ is the gradient of U evaluated at $\boldsymbol{\mu}_\theta$ and $\mathrm{U}_{\boldsymbol{\theta\theta}}(\boldsymbol{\mu}_\theta)$ is the corresponding Hessian matrix.

We plug this into the expectation of U

$$\begin{aligned}
\langle \mathrm{U}(\boldsymbol{\theta}) \rangle_{q_\theta} &\approx \langle \mathrm{U}(\boldsymbol{\mu}_\theta) \rangle_{q_\theta} + \left\langle (\mathbf{x} - \boldsymbol{\mu}_\theta)^\mathsf{T} \mathrm{U}_{\boldsymbol{\theta}}(\boldsymbol{\mu}_\theta) \right\rangle_{q_\theta} \\
&\qquad + \left\langle \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_\theta)^\mathsf{T} \mathrm{U}_{\boldsymbol{\theta\theta}}(\boldsymbol{\mu}_\theta)(\mathbf{x} - \boldsymbol{\mu}_\theta) \right\rangle_{q_\theta} && (13) \\
&= \mathrm{U}(\boldsymbol{\mu}_\theta) + \left(\langle \boldsymbol{\theta} \rangle_{q_\theta} - \boldsymbol{\mu}_\theta \right)^\mathsf{T} \mathrm{U}_{\boldsymbol{\theta}}(\boldsymbol{\mu}_\theta) + \frac{1}{2} \mathrm{Tr} \left(\mathrm{U}_{\boldsymbol{\theta\theta}}(\boldsymbol{\mu}_\theta) \boldsymbol{\Sigma}_{\boldsymbol{\mu}_\theta} \right) && (14) \\
&= \mathrm{U}(\boldsymbol{\mu}_\theta) + \frac{1}{2} \mathrm{Tr} \left(\mathrm{U}_{\boldsymbol{\theta\theta}}(\boldsymbol{\mu}_\theta) \boldsymbol{\Sigma}_{\boldsymbol{\mu}_\theta} \right) && (15)
\end{aligned}$$

where we have solved the expectation of the quadratic term using standard results for the multivariate Gaussian distribution [5, eq. (357)]. To account for the structure of the state space model we need to do some additional work (see below) to apply this analysis also to the expectation $\langle \mathrm{U}(\mathbf{Z}) \rangle_{q_z}$, but the same principle will be used. First, we show how we can get q_z from the approximated $\langle \mathrm{U}(\boldsymbol{\theta}) \rangle_{q_\theta}$ which corresponds to 2) above.

We follow the standard approach described in [4, p. 255] by noting that Eq. 11 has the appropriate form

$$q_z(\mathbf{Z}) = \frac{1}{Z} \exp\left(\langle \mathrm{U}(\boldsymbol{\theta}) \rangle_{q_\theta} \right) = \frac{1}{Z} \exp\left(-E(\mathbf{Z}) \right) \tag{16}$$

where we have defined $E(\mathbf{Z}) = -\langle \mathrm{U}(\boldsymbol{\theta}) \rangle_{q_\theta}$ and Z is a normalisation constant. Then the mean $\boldsymbol{\mu}_z$ of q_z is a minimum of $E(\mathbf{Z})$ and the covariance $\boldsymbol{\Sigma}_{\boldsymbol{\mu}_z}$ is the curvature of $E(\mathbf{Z})$ around the found minimum: $\boldsymbol{\Sigma}_{\boldsymbol{\mu}_z} = \frac{\partial^2 E(\mathbf{Z})}{\partial \mathbf{Z} \partial \mathbf{Z}}(\boldsymbol{\mu}_z)$. At least, this is how it would be, if $\mathbf{Z}$ was a vector. Such a formulation can easily be found by simply redefining $\mathbf{Z}$ as the concatenation of all hidden state vectors $\mathbf{Z} = [\mathbf{z}_1^\mathsf{T}, \ldots, \mathbf{z}_T^\mathsf{T}]^\mathsf{T}$. We can compute the necessary gradients of $E(\mathbf{Z})$ when plugging in the approximation of $\langle \mathrm{U}(\boldsymbol{\theta}) \rangle_{q_\theta}$

$$\begin{aligned}
E(\mathbf{Z}) &\approx -\mathrm{U}(\boldsymbol{\mu}_\theta) - \frac{1}{2} \mathrm{Tr} \left(\mathrm{U}_{\boldsymbol{\theta\theta}}(\boldsymbol{\mu}_\theta) \boldsymbol{\Sigma}_{\boldsymbol{\mu}_\theta} \right) && (17) \\
&= -\log p(\mathbf{Y}, \mathbf{Z} | \boldsymbol{\mu}_\theta) - \frac{1}{2} \mathrm{Tr} \left(\left. \frac{\partial^2 \log p(\mathbf{Y}, \mathbf{Z} | \boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}} \right|_{\boldsymbol{\mu}_\theta} \boldsymbol{\Sigma}_{\boldsymbol{\mu}_\theta} \right) && (18) \\
&= \hat{E}(\mathbf{Z}). && (19)
\end{aligned}$$

We now need to consider the structure of $p(\mathbf{Y}, \mathbf{Z} | \boldsymbol{\theta})$ to find a minimum of $\hat{E}(\mathbf{Z})$ together with its associated curvature and obtain an online algorithm which can process observations sequentially as they come in.

2.3 Online Computation of the Variational Density of the Hidden States

In the previous section we have seen that we need to minimise $\hat{E}(\mathbf{Z})$ to find the mean of the variational density $q_z(\mathbf{Z})$. A major component of $\hat{E}(\mathbf{Z})$ is the joint data-hidden state density $p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\mu}_\theta)$ which is proportional to the true posterior $p(\mathbf{Z}|\mathbf{Y}, \boldsymbol{\mu}_\theta)$. The second term of Eq. 18 accounts for the uncertainty in the parameter estimates. When the variance of the parameters $\boldsymbol{\Sigma}_{\boldsymbol{\mu}_\theta}$ goes to zero, i.e., parameter estimates are very certain, such that the second term in Eq. 18 becomes zero, minimisation of $\hat{E}(\mathbf{Z})$ is equal to finding a mode of the true posterior $p(\mathbf{Z}|\mathbf{Y}, \boldsymbol{\mu}_\theta)$. In this situation, any algorithm which can compute (an approximation of) the posterior can be used and the variational approach is unnecessary.

2.3.1 Background: Filtering

In particular, we are interested in (online) filtering approaches which sequentially estimate $p(\mathbf{Z}_{1:t}|\mathbf{Y}_{1:t}, \boldsymbol{\mu}_\theta)$ for increasing t where the index $1:t$ indicates that variables at time points 1 to t are considered such that, e.g., $\mathbf{Z}_{1:t} = \{\mathbf{z}_1, \dots, \mathbf{z}_t\}$. There is a general recursion relating subsequent time points [see e.g. 6]

$$p(\mathbf{Z}_{1:t}|\mathbf{Y}_{1:t}, \boldsymbol{\mu}_\theta) = p(\mathbf{Z}_{1:t-1}|\mathbf{Y}_{1:t-1}, \boldsymbol{\mu}_\theta)\frac{p(\mathbf{z}_t|\mathbf{z}_{t-1}, \boldsymbol{\mu}_\theta)p(\mathbf{y}_t|\mathbf{z}_t, \boldsymbol{\mu}_\theta)}{p(\mathbf{y}_t|\mathbf{Y}_{1:t-1}, \boldsymbol{\mu}_\theta)} \tag{20}$$

where the model densities $p(\mathbf{z}_t|\mathbf{z}_{t-1}, \boldsymbol{\mu}_\theta), p(\mathbf{y}_t|\mathbf{z}_t, \boldsymbol{\mu}_\theta)$ are defined by Eqs. 4 and 5, respectively, and $p(\mathbf{y}_t|\mathbf{Y}_{1:t-1}, \boldsymbol{\mu}_\theta)$ is the density which predicts the next observation based on the model and the previously seen observations. It is the latter predictive density which often prevents computation of the exact posterior, because it requires integration over hidden states (we omit explicit conditioning on the expected parameters $\boldsymbol{\mu}_\theta$ for brevity):

$$p(\mathbf{y}_t|\mathbf{Y}_{1:t-1}) = \iint p(\mathbf{z}_{t-1}|\mathbf{Y}_{1:t-1})p(\mathbf{z}_t|\mathbf{z}_{t-1})p(\mathbf{y}_t|\mathbf{z}_t)d\mathbf{z}_{t-1}d\mathbf{z}_t \tag{21}$$

where $p(\mathbf{z}_t|\mathbf{Y}_{1:t})$ is the marginal posterior of the current hidden state $\mathbf{z}_t$ obtained from Eq. 20 by integrating the state history $\mathbf{Z}_{1:t-1}$ out of $p(\mathbf{Z}_{1:t}|\mathbf{Y}_{1:t})$. In tracking applications, estimation of the current state $\mathbf{z}_t$ typically is sufficient. Then, we can estimate the marginal posterior directly in a recursion with prediction step

$$p(\mathbf{z}_t|\mathbf{Y}_{1:t-1}) = \int p(\mathbf{z}_t|\mathbf{z}_{t-1})p(\mathbf{z}_{t-1}|\mathbf{Y}_{1:t-1})d\mathbf{z}_{t-1} \tag{22}$$

followed by an update step

$$p(\mathbf{z}_t|\mathbf{Y}_{1:t}) = \frac{p(\mathbf{y}_t|\mathbf{z}_t)p(\mathbf{z}_t|\mathbf{Y}_{1:t-1})}{p(\mathbf{y}_t|\mathbf{Y}_{1:t-1})}. \tag{23}$$

Filtering algorithms provide computationally feasible implementations of these equations by making suitable approximations, when analytic solutions cannot be found. The Kalman filter based algorithms mostly work with Eqs. 22, 23 while particle filters mostly approximate the recursion in Eq. 20 [6].

2.3.2 A Gradient Based Filter

In the case that the second term of Eq. 18 is nonzero, i.e., when there is uncertainty over the parameters expressed by a nonzero posterior covariance $\boldsymbol{\Sigma}_{\boldsymbol{\mu}_\theta}$, the standard filtering algorithms cannot be used to obtain a meaningful posterior over hidden states. Instead, we take the gradient based approach which minimises $\hat{E}(\mathbf{Z})$ (Eq. 18). The problem is that $\hat{E}(\mathbf{Z})$ is a function of all hidden states at all time points, but we want a sequential, online algorithm. To achieve this, we sacrifice our ability to infer the covariance of hidden state estimates over time. In particular, we let the variational density factorise such that

$$q_z(\mathbf{Z}) = \prod_{t=1}^{T} q_z(\mathbf{z}_t). \tag{24}$$

This approach is equivalent to standard filters (cf. Eqs. 22, 23) which only track the posterior distribution of the most recent hidden state. And equivalent to standard filters, we do take the dependence of a hidden state on previous hidden states into account, although we do not represent the covariance between hidden states in the final posterior. In the following we show that we can use prediction steps of standard filters and replace standard update steps by a gradient based update derived from $\hat{E}(\mathbf{Z})$.

The core function we need to consider is the joint data-hidden state density $p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\mu}_\theta)$ which, rolled out in time, is

$$p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\mu}_\theta) = p(\mathbf{z}_0|\boldsymbol{\mu}_\theta) \prod_{t=1}^{T} p(\mathbf{y}_t|\mathbf{z}_t, \boldsymbol{\mu}_\theta) p(\mathbf{z}_t|\mathbf{z}_{t-1}, \boldsymbol{\mu}_\theta). \tag{25}$$

It is our aim to minimise a function of this function with respect to the individual hidden states $\mathbf{z}_t$ as data points $\mathbf{y}_t, t \in 1, \dots, T$ arrive. Considering the first data point only we have

$$p(\mathbf{y}_1, \mathbf{z}_1, \mathbf{z}_0|\boldsymbol{\mu}_\theta) = p(\mathbf{z}_0|\boldsymbol{\mu}_\theta) p(\mathbf{y}_1|\mathbf{z}_1, \boldsymbol{\mu}_\theta) p(\mathbf{z}_1|\mathbf{z}_0, \boldsymbol{\mu}_\theta), \tag{26}$$

but we are not interested in the value of $\mathbf{z}_0$ over which we have defined the prior distribution $p(\mathbf{z}_0|\boldsymbol{\mu}_\theta)$. Hence, we would like to integrate $\mathbf{z}_0$ out of the equation

$$p(\mathbf{y}_1, \mathbf{z}_1|\boldsymbol{\mu}_\theta) = p(\mathbf{y}_1|\mathbf{z}_1, \boldsymbol{\mu}_\theta) \int p(\mathbf{z}_1|\mathbf{z}_0, \boldsymbol{\mu}_\theta) p(\mathbf{z}_0|\boldsymbol{\mu}_\theta) \mathrm{d}\mathbf{z}_0. \tag{27}$$

Because we consider nonlinear dynamics, the integral needs to be approximated, for example, by local linearisation (cf. extended Kalman filter), or by the unscented transform (cf. unscented Kalman filter).

Remark: In Friston's original proposal [1, 2] the prior $p(\mathbf{z}_0|\boldsymbol{\mu}_\theta)$ is approximated using a delta function such that the integral simplifies to

$$\int p(\mathbf{z}_1|\mathbf{z}_0, \boldsymbol{\mu}_\theta) p(\mathbf{z}_0|\boldsymbol{\mu}_\theta) \mathrm{d}\mathbf{z}_0 \approx p(\mathbf{z}_1|\boldsymbol{\mu}_{z_0}, \boldsymbol{\mu}_\theta) \tag{28}$$

where $\boldsymbol{\mu}_{z_0}$ is the value for $\mathbf{z}_0$ selected by the delta function, e.g., the mean of the original prior distribution. This is the simplest, computationally most efficient approximation of the integral you can make, but it ignores all uncertainty over the previous hidden state.

Whatever approximation is used for the integral, it provides us the approximated (Gaussian) density $\hat{p}(\mathbf{z}_1|\boldsymbol{\mu}_\theta)$ and Eq. 27 becomes

$$\hat{p}(\mathbf{y}_1, \mathbf{z}_1|\boldsymbol{\mu}_\theta) = p(\mathbf{y}_1|\mathbf{z}_1, \boldsymbol{\mu}_\theta)\hat{p}(\mathbf{z}_1|\boldsymbol{\mu}_\theta) \tag{29}$$

of which we now can take the derivative with respect to $\mathbf{z}_1$ to find the approximate posterior density $q_z(\mathbf{z}_1)$. This leads to an obvious iterative procedure in which the previous posterior $q_z(\mathbf{z}_{t-1})$ takes the place of the prior $p(\mathbf{z}_0|\boldsymbol{\mu}_\theta)$ for all subsequent hidden states $\mathbf{z}_t$. The computation of $\hat{p}(\mathbf{z}_t|\boldsymbol{\mu}_\theta)$, then, corresponds exactly to the prediction step in standard filters and the following gradient update to the standard update step.

Remark: The prediction step is incomplete, because the prediction density $\hat{p}(\mathbf{z}_t|\boldsymbol{\mu}_\theta)$ does not take the uncertainty over parameters into account. This can lead to distorted results, but I currently do not know of a good solution for including parameter uncertainty in the prediction density.

2.3.3 Gradient Updates

The prediction step provides us with an approximated joint (Gaussian) density $\hat{p}(\mathbf{y}_t, \mathbf{z}_t|\boldsymbol{\mu}_\theta)$ which is independent of previous hidden states. We can then define the corresponding energy function $\hat{E}(\mathbf{z}_t)$, which we want to minimise, as

$$\hat{E}(\mathbf{z}_t) = -\log \hat{p}(\mathbf{y}_t, \mathbf{z}_t|\boldsymbol{\mu}_\theta) - \frac{1}{2}\mathrm{Tr}\left(\left.\frac{\partial^2 \log \hat{p}(\mathbf{y}_t, \mathbf{z}_t|\boldsymbol{\mu}_\theta)}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}}\right|_{\boldsymbol{\mu}_\theta} \boldsymbol{\Sigma}_{\boldsymbol{\mu}_\theta} \right). \tag{30}$$

We initialise $\mathbf{z}_t$ to the mean $\boldsymbol{\mu}_{\hat{z}_t}$ of the prediction density $\hat{p}(\mathbf{z}_t|\boldsymbol{\mu}_\theta)$ computed in the prediction step. We now first consider the gradient defined by the first term, i.e.

$$-\left.\frac{\partial \log \hat{p}(\mathbf{y}_t, \mathbf{z}_t|\boldsymbol{\mu}_\theta)}{\partial \mathbf{z}_t}\right|_{\boldsymbol{\mu}_{\hat{z}_t}} = -\left.\frac{\partial \log p(\mathbf{y}_t|\mathbf{z}_t, \boldsymbol{\mu}_\theta)}{\partial \mathbf{z}_t}\right|_{\boldsymbol{\mu}_{\hat{z}_t}} - \left.\frac{\partial \log \hat{p}(\mathbf{z}_t|\boldsymbol{\mu}_\theta)}{\partial \mathbf{z}_t}\right|_{\boldsymbol{\mu}_{\hat{z}_t}}. \tag{31}$$

At least in the initial step, the gradient from the prediction density is 0:

$$\left.\frac{\partial \log \hat{p}(\mathbf{z}_t|\boldsymbol{\mu}_\theta)}{\partial \mathbf{z}_t}\right|_{\boldsymbol{\mu}_{\hat{z}_t}} = -\boldsymbol{\Sigma}_{\hat{z}_t}^{-1}(\mathbf{z}_t - \boldsymbol{\mu}_{\hat{z}_t})\big|_{\boldsymbol{\mu}_{\hat{z}_t}} = \mathbf{0}, \tag{32}$$

because $\hat{p}(\mathbf{z}_t|\boldsymbol{\mu}_\theta) \sim \mathcal{N}(\boldsymbol{\mu}_{\hat{z}_t}, \boldsymbol{\Sigma}_{\hat{z}_t})$. Consequently, it is clear that several gradient steps are necessary to reach an appropriate optimum of the objective function. If only a single gradient step is taken after each incoming observation, the hidden state uncertainty quantified by $\boldsymbol{\Sigma}_{\hat{z}_t}$ will never be taken into account. Friston's filter runs in continuous time, so there always is a new observation and no time to run intermediate gradient steps. So he needed to find another way of including hidden state uncertainty into the gradients. He did this with generalised state representations (see Sec. 2.3.4 below). It is currently unclear to me whether you need generalised states when you can run several intermediate gradient steps between incoming observations.

The gradient of the observation density is

$$\left.\frac{\partial \log p(\mathbf{y}_t|\mathbf{z}_t, \boldsymbol{\mu}_\theta)}{\partial \mathbf{z}_t}\right|_{\boldsymbol{\mu}_{\hat{z}_t}} = \left.\frac{\partial g(\mathbf{z}_t, \boldsymbol{\theta}_y)}{\partial \mathbf{z}_t}\boldsymbol{\Sigma}_y^{-1}(\mathbf{y}_t - g(\mathbf{z}_t, \boldsymbol{\theta}_y))\right|_{\boldsymbol{\mu}_{\hat{z}_t}}, \tag{33}$$

because $p(\mathbf{y}_t|\mathbf{z}_t, \boldsymbol{\mu}_\theta) \sim \mathcal{N}(g(\mathbf{z}_t, \boldsymbol{\theta}_y), \boldsymbol{\Sigma}_y)$.

Remark: How does this relate to extended Kalman filter updates?

next: gradient of parameter uncertainty term, perhaps first for delta approximation, then for unscented transform, then: discussion of gradient descent procedure (making a single step only, or making several steps until new data point arrives, efficiency can perhaps be shown experimentally)

2.3.4 Friston's Generalised State Representation

The variational approach presents us with a dilemma: On the one hand, it requires estimation of the variational posterior over hidden states at all time points given all data, i.e., the variational posterior distribution of hidden state $\mathbf{z}_t$ also needs to take data at time point $t + 1$ and later into account, calling for smoothing instead of filtering [cf. 3]. On the other hand, we would like to have a pure online algorithm which sequentially computes all necessary quantities as the data comes in, calling for filtering instead of smoothing. Our way out of this dilemma is to make another approximation which could be interpreted as local smoothing within a filtering procedure.

The benefit of smoothing is clear: By looking how observations change in the future you can get a more accurate estimate of a hidden state. That, in turn, may affect your estimate of the following hidden state demonstrating that hidden state estimates covary. Consequently, estimating the full variational posterior $q_z(\mathbf{Z}_{1:t})$, even when approximated by a simple Gaussian, requires to estimate covariances between hidden states at all time points. The size of the corresponding covariance matrix increases quadratically with time which is often infeasible in practice. But it appears to be a reasonable assumption that the covariance between posterior hidden state estimates should decrease over time such that the entries in the full covariance matrix should go towards 0 as you go away from the diagonal.

The approximation we use is based on this assumption and uses the simple trick of state generalisation: states at a given time t are augmented by states at previous and future times in the hope that the new augmented (generalised) states become conditionally independent. The idea can be demonstrated on transforming a second-order Markov dynamics into a first-order Markov dynamics: Assume that a state $\mathbf{z}_t$ depends on its two predecessors through a function $\mathbf{z}_t = f(\mathbf{z}_{t-2}, \mathbf{z}_{t-1})$, i.e., $\mathbf{z}_t$ depends on both $\mathbf{z}_{t-2}$ and $\mathbf{z}_{t-1}$. We then introduce a generalised state $\tilde{\mathbf{z}}_t = [\mathbf{z}_{t-1}^\mathsf{T}, \mathbf{z}_t^\mathsf{T}]^\mathsf{T}$ and an associated transition function $\tilde{f}(\tilde{\mathbf{z}}_{t-1}) = [\mathbf{z}_{t-1}^\mathsf{T}, f(\mathbf{z}_{t-2}, \mathbf{z}_{t-1})^\mathsf{T}]^\mathsf{T}$. The resulting generalised system now is first-order Markov, i.e., conditional on $\tilde{\mathbf{z}}_{t-1}$, $\tilde{\mathbf{z}}_t$ is independent of $\tilde{\mathbf{z}}_{t-2}$. We can now apply this idea to the states in our problem and define

$$\tilde{\mathbf{z}}_t = \begin{bmatrix} \mathbf{z}_{t-n} \\ \vdots \\ \mathbf{z}_t \\ \vdots \\ \mathbf{z}_{t+n} \end{bmatrix}, \tag{34}$$

where n is the embedding order of the generalised state. Then state transitions

(Eq. 4) simply become

$$\tilde{\mathbf{z}}_t = \tilde{f}(\tilde{\mathbf{z}}_{t-1}) + \tilde{\boldsymbol{\omega}}_t^z = \begin{bmatrix} \mathbf{z}_{t-n} \\ \vdots \\ \mathbf{z}_t \\ \vdots \\ f(\mathbf{z}_{t+n-1}) \end{bmatrix} + \tilde{\boldsymbol{\omega}}_t^z, \tag{35}$$

where $f(\cdot)$ is the state transition function of Eq. 4 and $\tilde{\boldsymbol{\omega}}_t^z$ is a generalised vector of noise variables which now also allows noise to be correlated across time points. In comparison with the simple example above we here do not gain a reduction of Markov order of the dynamic generative model, because the dynamics already was first-order Markov, but the generalised state representation now allows to represent posterior covariances between states at different time points, at least within the chosen time window.

Idea: variational posterior will then approximately factorise, i.e., $q(\tilde{\mathbf{Z}}_{1:T}) = \prod_t q(\tilde{\mathbf{z}}_t)$ (are we eventually only interested in the posteriors over $\mathbf{z}_t$, but not $\tilde{\mathbf{z}}_t$?) so that we can optimise it at each time point individually, but of course the dynamics needs to somehow be taken into account, how is done?

How will it be useful to represent posterior covariances, i.e., where will they be used?

The variational approach leads to the optimisation in which you find the posterior mode using gradient descent. The structure of that optimisation problem is the same with and without generalised representation. So what is the generalised representation good for? When you find a sequential way of doing this gradient descent, can there be a way of incorporating the posterior uncertainty similar to other filtering approaches, or is it fundamentally impossible to do so? What is the sequential gradient descent procedure?

3 Experiments

take a problem which has previously been solved with a variational filter (check Beal's thesis), apply the generalised filter and compare results, they should be worse for the generalised filter, but not much

then apply the variational filter on nonlinear synthetic learning problems to see whether it does sensible stuff

4 Nonlinear Gaussian Dynamic Models in Continuous Time

We have given the following, indirectly observed, stochastic dynamical system

$$\dot{\mathbf{z}} = f(\mathbf{z}, \boldsymbol{\theta}_z) + \boldsymbol{\omega}_z \tag{36}$$

$$\mathbf{y} = g(\mathbf{z}, \boldsymbol{\theta}_y) + \boldsymbol{\omega}_y \tag{37}$$

where f, g are general, nonlinear functions of the hidden states $\mathbf{z}$ with parameters $\boldsymbol{\theta}_z, \boldsymbol{\theta}_y$ and the $\boldsymbol{\omega}$ are Gaussian noise variables centred at 0 and with covariance $\boldsymbol{\Sigma}_z, \boldsymbol{\Sigma}_y$. We have here used Friston's own notation of stochastic differential equations which appears to be following the notation for discrete time systems. He never defines this formally. Our best guess is that Eq. 36 corresponds to a stochastic differential equation in Itō calculus:

$$d\mathbf{z} = f(\mathbf{z}, \boldsymbol{\theta}_z)dt + \mathbf{L}d\mathbf{w}$$

where $\mathbf{L}\mathbf{L}^\mathsf{T} = \boldsymbol{\Sigma}_z$ is the Cholesky decomposition of the covariance and $d\mathbf{w}$ is a vector of Wiener processes. We will now use the analogy to discrete time systems to gain an intuitive understanding of what kind of probabilistic model Friston assumes, what the aim of inference in this model is and what kind of approximations he makes. Because of our lack of in depth knowledge of Itō calculus we cannot comment on the corresponding Itō calculations or correctness of such an analysis in continuous time.

In discrete time the dynamic state equations would lead to the following probabilistic model at time $t = 1$

$$p(\mathbf{y}_1, \mathbf{z}_1) = p(\mathbf{y}_1|\mathbf{z}_1) \int p(\mathbf{z}_1|\mathbf{z}_0)p(\mathbf{z}_0)d\mathbf{z}_0 = p(\mathbf{y}_1|\mathbf{z}_1)p(\mathbf{z}_1) \tag{38}$$

where $p(\mathbf{z}_0)$ is a prior distribution over the initial hidden states at time $t = 0$.

Whatever function is used in the end to find the posterior mode, all corresponding gradients are computed from the gradients of $\bar{\mathrm{U}}$. So it is time to look at the structure of $\bar{\mathrm{U}}$. Notice that from here on subscripts of variables or functions are used solely to identify different variables or functions, or to index into matrices and vectors. Partial derivatives are written out in the standard format. While this clutters the formulae more, it helps us to make them conceptually clearer.

First, remember that (eq. 93)

$$\bar{\mathrm{U}}(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\theta}) = \log p(\boldsymbol{\theta}) + \int_{t=0}^{T} \log p(\mathbf{y}(t), \mathbf{z}(t)|\boldsymbol{\theta})dt.$$

The parameter prior, as all other model densities, is chosen to be Gaussian $p(\boldsymbol{\theta}) \sim \mathcal{N}(\boldsymbol{\eta}_\theta, \boldsymbol{\Psi}^\theta)$ such that

$$\log p(\boldsymbol{\theta}) = -\frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\eta}_\theta)^\mathsf{T} {\boldsymbol{\Psi}^{\theta}}^{-1}(\boldsymbol{\theta} - \boldsymbol{\eta}_\theta) + c \tag{39}$$

$$= -\frac{1}{2}\boldsymbol{\epsilon}_\theta^\mathsf{T} \boldsymbol{\Pi}^\theta \boldsymbol{\epsilon}_\theta + c \tag{40}$$

where we have lumped terms from the normalisation constant of the Gaussian distribution into the constant c and introduced Friston's prediction error - precision formulation in the second line.

The joint data - state density is a bit more difficult. The probabilistic definition is

$$p(\mathbf{y}(t), \mathbf{z}(t)|\vec{\mathbf{Y}}(t), \boldsymbol{\theta}) = p(\mathbf{y}(t)|\mathbf{z}(t), \boldsymbol{\theta})p(\mathbf{z}(t)|\vec{\mathbf{Y}}(t), \boldsymbol{\theta}) \tag{41}$$

where $\vec{\mathbf{Y}}(t)$ is all data observed previous to time t.

The observation density $p(\mathbf{y}(t)|\mathbf{z}(t), \boldsymbol{\theta})$ can be defined in a straightforward way based on an output nonlinearity with additive zero-mean Gaussian noise: $g(\mathbf{z}(t), \boldsymbol{\theta}) + \boldsymbol{\omega}_y$, $\boldsymbol{\omega}_y \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Psi}^y)$ such that

$$\log p(\mathbf{y}(t)|\mathbf{z}(t), \boldsymbol{\theta}) = -\frac{1}{2}(\mathbf{y}(t) - g(\mathbf{z}(t), \boldsymbol{\theta}))^\mathsf{T} {\boldsymbol{\Psi}^{y}}^{-1}(\mathbf{y}(t) - g(\mathbf{z}(t), \boldsymbol{\theta})) + c \tag{42}$$

$$= -\frac{1}{2}\boldsymbol{\epsilon}_y^\mathsf{T} \boldsymbol{\Pi}^y \boldsymbol{\epsilon}_y + c. \tag{43}$$

The prior $p(\mathbf{z}(t)|\vec{\mathbf{Y}}(t),\boldsymbol{\theta})$ is most generally defined as

$$p(\mathbf{z}(t)|\vec{\mathbf{Y}}(t),\boldsymbol{\theta}) = \left\langle p(\mathbf{z}(t),\vec{\mathbf{Z}}(t)|\vec{\mathbf{Y}}(t),\boldsymbol{\theta})\right\rangle_{\vec{\mathbf{Z}}(t)} \tag{44}$$

which states that the prior for the state at time t depends on the observations previous to t $\vec{\mathbf{Y}}(t)$ via averaging over all posterior state trajectories previous to t. Such a model is intractable. So we use a Markov assumption to make the current state only depend on the previous

$$\begin{aligned} p(\mathbf{z}(t)|\vec{\mathbf{Y}}(t),\boldsymbol{\theta}) &= \left\langle p(\mathbf{z}(t),\vec{\mathbf{z}}(t)|\vec{\mathbf{Y}}(t),\boldsymbol{\theta})\right\rangle_{\vec{\mathbf{z}}(t)} && (45)\\ &= \left\langle p(\mathbf{z}(t)|\vec{\mathbf{z}}(t),\boldsymbol{\theta})p(\vec{\mathbf{z}}(t)|\vec{\mathbf{Y}}(t),\boldsymbol{\theta})\right\rangle_{\vec{\mathbf{z}}(t)}. && (46)\end{aligned}$$

Well, at least this is what you would do in a discrete time setting where the current state is related to a previous state via $\mathbf{z}(t) = f(\vec{\mathbf{z}}(t),\boldsymbol{\theta})$. In the continuous time setting we only have the functional relationship

$$\dot{\mathbf{z}} = f(\mathbf{z},\boldsymbol{\theta}) \tag{47}$$

and we have to solve this differential equation to get $\mathbf{z}(t)$ for the observation density. Friston introduced generalised states to overcome this problem.

4.1 Generalised Coordinates

A generalised state represents the future trajectory of a state in form of an infinitely large vector containing its time derivatives

$$\tilde{\mathbf{z}}(t) = \begin{bmatrix} \mathbf{z}(t) \\ \frac{\partial \mathbf{z}(t)}{\partial t} \\ \frac{\partial^2 \mathbf{z}(t)}{\partial t \partial t} \\ \vdots \end{bmatrix} = \begin{bmatrix} \mathbf{z} \\ \dot{\mathbf{z}} \\ \ddot{\mathbf{z}} \\ \vdots \end{bmatrix}. \tag{48}$$

This can also be applied to functions which depend on time-varying variables

$$\tilde{\mathbf{f}}(\tilde{\mathbf{z}}(t),\boldsymbol{\theta}) = \begin{bmatrix} f(\mathbf{z}(t),\boldsymbol{\theta}) \\ \frac{\partial f(\mathbf{z}(t),\boldsymbol{\theta})}{\partial t} \\ \frac{\partial^2 f(\mathbf{z}(t),\boldsymbol{\theta})}{\partial t \partial t} \\ \vdots \end{bmatrix} \approx \begin{bmatrix} f(\mathbf{z}(t),\boldsymbol{\theta}) \\ \frac{\partial f(\mathbf{z}(t),\boldsymbol{\theta})}{\partial \mathbf{z}} \frac{\partial \mathbf{z}}{\partial t} \\ \frac{\partial f(\mathbf{z}(t),\boldsymbol{\theta})}{\partial \mathbf{z}} \frac{\partial^2 \mathbf{z}}{\partial t \partial t} \\ \vdots \end{bmatrix} = \begin{bmatrix} f(\mathbf{z}(t),\boldsymbol{\theta}) \\ f_{\mathbf{z}}\dot{\mathbf{z}} \\ f_{\mathbf{z}}\ddot{\mathbf{z}} \\ \vdots \end{bmatrix} \tag{49}$$

$$\tilde{\mathbf{g}}(\tilde{\mathbf{z}}(t),\boldsymbol{\theta}) = \begin{bmatrix} g(\mathbf{z}(t),\boldsymbol{\theta}) \\ \frac{\partial g(\mathbf{z}(t),\boldsymbol{\theta})}{\partial t} \\ \frac{\partial^2 g(\mathbf{z}(t),\boldsymbol{\theta})}{\partial t \partial t} \\ \vdots \end{bmatrix} \approx \begin{bmatrix} g(\mathbf{z}(t),\boldsymbol{\theta}) \\ \frac{\partial g(\mathbf{z}(t),\boldsymbol{\theta})}{\partial \mathbf{z}} \frac{\partial \mathbf{z}}{\partial t} \\ \frac{\partial g(\mathbf{z}(t),\boldsymbol{\theta})}{\partial \mathbf{z}} \frac{\partial^2 \mathbf{z}}{\partial t \partial t} \\ \vdots \end{bmatrix} = \begin{bmatrix} g(\mathbf{z}(t),\boldsymbol{\theta}) \\ g_{\mathbf{z}}\dot{\mathbf{z}} \\ g_{\mathbf{z}}\ddot{\mathbf{z}} \\ \vdots \end{bmatrix} \tag{50}$$

where we have assumed that f and g are locally linear with respect to $\mathbf{z}$.

4.2 Dynamic Model in Generalised Coordinates

It is then possible to redefine the observation density in generalised coordinates and obtain

$$\log p(\tilde{\mathbf{y}}(t)|\tilde{\mathbf{g}}(t),\boldsymbol{\theta}) = -\frac{1}{2}(\tilde{\mathbf{y}}(t) - \tilde{\mathbf{g}}(\tilde{\mathbf{z}}(t),\boldsymbol{\theta}))^{\mathsf{T}}\tilde{\boldsymbol{\Psi}}^{y^{-1}}(\tilde{\mathbf{y}}(t) - \tilde{\mathbf{g}}(\tilde{\mathbf{z}}(t),\boldsymbol{\theta})) + c \tag{51}$$

$$= -\frac{1}{2}\tilde{\boldsymbol{\epsilon}}_y^{\mathsf{T}}\tilde{\boldsymbol{\Pi}}^{y}\tilde{\boldsymbol{\epsilon}}_y + c. \tag{52}$$

Equivalently, we can now define the state transition probability in generalised coordinates based on the differential equation with additive Gaussian noise $\mathbf{D}\tilde{\mathbf{z}}(t) = \tilde{\mathbf{f}}(\tilde{\mathbf{z}}(t),\boldsymbol{\theta}) + \tilde{\boldsymbol{\omega}}_z$, $\tilde{\boldsymbol{\omega}}_z \sim \mathcal{N}(\mathbf{0}, \tilde{\boldsymbol{\Psi}}^{z})$ to get

$$\log p(\mathbf{D}\tilde{\mathbf{z}}(t)|\tilde{\mathbf{f}}(t),\boldsymbol{\theta}) = -\frac{1}{2}(\mathbf{D}\tilde{\mathbf{z}}(t) - \tilde{\mathbf{f}}(\tilde{\mathbf{z}}(t),\boldsymbol{\theta}))^{\mathsf{T}}\tilde{\boldsymbol{\Psi}}^{z^{-1}}(\tilde{\mathbf{z}}(t) - \tilde{\mathbf{f}}(\tilde{\mathbf{z}}(t),\boldsymbol{\theta})) + c \tag{53}$$

$$= -\frac{1}{2}\tilde{\boldsymbol{\epsilon}}_z^{\mathsf{T}}\tilde{\boldsymbol{\Pi}}^{z}\tilde{\boldsymbol{\epsilon}}_z + c \tag{54}$$

where $\mathbf{D}$ is a differential operator which shifts the corresponding generalised coordinates up by one coordinate such that

$$\mathbf{D}\tilde{\mathbf{z}} = \begin{bmatrix} \dot{\mathbf{z}} \\ \ddot{\mathbf{z}} \\ \vdots \end{bmatrix}. \tag{55}$$

We can now identify $p(\mathbf{D}\tilde{\mathbf{z}}(t)|\tilde{\mathbf{f}}(t),\boldsymbol{\theta})$ with $p(\mathbf{z}(t)|\vec{\mathbf{z}}(t),\boldsymbol{\theta})$ from eq. (46). Thus a full probabilistic model would integrate over $\tilde{\mathbf{f}}(t)$:

$$p(\mathbf{D}\tilde{\mathbf{z}}(t)|\vec{\mathbf{Y}}(t),\boldsymbol{\theta}) = \left\langle p(\mathbf{D}\tilde{\mathbf{z}}(t)|\tilde{\mathbf{f}}(t),\boldsymbol{\theta})\right\rangle_{p(\tilde{\mathbf{f}}(t)|\vec{\mathbf{Y}}(t),\boldsymbol{\theta})}. \tag{56}$$

Unfortunately, we don't know the distribution $p(\tilde{\mathbf{f}}(t)|\vec{\mathbf{Y}}(t),\boldsymbol{\theta})$. The variational approximation gives us a Gaussian $q(\tilde{\mathbf{z}}(t))$, but computing $\tilde{\mathbf{f}}(t) = \tilde{\mathbf{f}}(\tilde{\mathbf{z}}(t),\boldsymbol{\theta})$ via eq. (49) transforms $\tilde{\mathbf{z}}(t)$ nonlinearly. Friston uses the simplest possible approximation: a point estimate by (implicitly) setting $p(\tilde{\mathbf{f}}(t)|\vec{\mathbf{Y}}(t),\boldsymbol{\theta})$ to the Dirac delta function at the current estimate $\tilde{\boldsymbol{\mu}}_{\tilde{f}}(t)$ of $\tilde{\mathbf{f}}(t)$, i.e.,

$$p(\tilde{\mathbf{f}}(t)|\vec{\mathbf{Y}}(t),\boldsymbol{\theta}) \approx \delta(\tilde{\mathbf{f}}(t) - \tilde{\boldsymbol{\mu}}_{\tilde{f}}(t)) \tag{57}$$

such that

$$\left\langle p(\mathbf{D}\tilde{\mathbf{z}}(t)|\tilde{\mathbf{f}}(t),\boldsymbol{\theta})\right\rangle_{p(\tilde{\mathbf{f}}(t)|\vec{\mathbf{Y}}(t),\boldsymbol{\theta})} \approx p(\mathbf{D}\tilde{\mathbf{z}}(t)|\tilde{\boldsymbol{\mu}}_{\tilde{f}}(t),\boldsymbol{\theta}). \tag{58}$$

This approximation has an important impact on the inference. It means that state uncertainty is not propagated through time, i.e., the posterior covariances of the states only depend on the prior covariances $\tilde{\boldsymbol{\Psi}}^{z}$ and $\tilde{\boldsymbol{\Psi}}^{y}$ and on the corresponding functions f and g. On the one hand, this prevents the divergence of state uncertainty over time through accumulation of errors. On the other hand, it also prevents posterior uncertainty to decrease, for example, when the dynamics prescribed by the model can be followed accurately over an extended period of time. In any case, it makes the posterior covariances of the states purely instantaneous measures of uncertainty which do not necessarily reflect the true uncertainties accumulated over time.

4.3 Generalised Covariances

We used "generalised covariances" to define the Gaussian densities above, i.e., we introduced covariances for the representations in generalised coordinates. To this point, it is still unclear to us how Friston derived his formula for computing generalised covariances. However, this is central to the definition of his probabilistic model as it defines what kind of stochastic process he assumes. We now state what he suggests and comment on his equations as far as we understand them.

He defines generalised covariances as[1]

$$\tilde{\mathbf{\Psi}} = \mathbf{S}(\gamma) \otimes \mathbf{\Psi} \tag{59}$$

where $\mathbf{S}(\gamma)$ is a potentially infinitely large (depending on the number of considered generalised coordinates) matrix which translates an assumed covariance for a state $\mathbf{z}$ $\mathbf{\Psi}$ to the covariances for its generalised coordinates $\dot{\mathbf{z}}, \ddot{\mathbf{z}}, \ldots$ by scaling (and mixing) $\mathbf{\Psi}$. Friston states that the "temporal covariance" $\mathbf{S}(\gamma)$ can be written as

$$\mathbf{S}(\gamma) = \begin{bmatrix} 1 & 0 & \ddot{\rho}(0) & \cdots \\ 0 & -\ddot{\rho}(0) & 0 & \\ \ddot{\rho}(0) & 0 & \ddddot{\rho}(0) & \\ \vdots & & & \ddots \end{bmatrix} \tag{60}$$

where $\rho(\Delta t)$ is the autocorrelation function of the noise and $\ddot{\rho}(0)$ its second derivative with respect to time evaluated at $\Delta t = 0$. The origin of this relationship is a mystery to us. That ρ only has a single argument hints at the assumption of a stationary noise process, but the derivative of the corresponding autocorrelation function with respect to time should be 0 then, because it would not change with time. Friston further states that, if ρ is a Gaussian with variance γ, then

$$\mathbf{S}(\gamma) = \begin{bmatrix} 1 & 0 & -\frac{1}{2\gamma} & \cdots \\ 0 & \frac{1}{2\gamma} & 0 & \\ -\frac{1}{2\gamma} & 0 & \frac{3}{4\gamma^2} & \\ \vdots & & & \ddots \end{bmatrix}, \tag{61}$$

but the elements of $\mathbf{S}(\gamma)$ here are certainly not the derivatives of a Gaussian.

Consequently, we are tapping completely in the dark when trying to interpret generalised covariances. Why, for example, does a state only covary with every second of its generalised coordinates (cf. nonzero entries in $\mathbf{S}(\gamma)$)? We can only say for sure that the generalised covariance increases for higher order coordinates as long as γ is small enough. In his code, Friston sets $\gamma = 1/4$ by default which means that the generalised covariances quickly become so large for higher order coordinates that these coordinates become completely uninformative and considering more than 6, or so, is a waste of computing time.

[1] Friston actually defines generalised precisions instead of generalised covariances, but this turns out the same, because $(\mathbf{A} \otimes \mathbf{B})^{-1} = \mathbf{A}^{-1} \otimes \mathbf{B}^{-1}$, if $\mathbf{A}$ and $\mathbf{B}$ are invertible.

4.4 Internal Action of Dynamic Model

To summarise, we can now rewrite the internal action in terms of prediction errors and precisions

$$\begin{aligned}\bar{\mathrm{U}}(\mathbf{Y},\mathbf{Z},\boldsymbol{\theta}) &= \log p(\boldsymbol{\theta}) + \int_0^T \log p(\tilde{\mathbf{y}}(t),\tilde{\mathbf{z}}(t)|\boldsymbol{\theta})dt \\ &= C - \frac{1}{2}\left(\boldsymbol{\epsilon}_\theta^\mathsf{T}\boldsymbol{\Pi}^\theta\boldsymbol{\epsilon}_\theta + \int_0^T \tilde{\boldsymbol{\epsilon}}_y^\mathsf{T}(t)\tilde{\boldsymbol{\Pi}}^y\tilde{\boldsymbol{\epsilon}}_y(t) + \tilde{\boldsymbol{\epsilon}}_z^\mathsf{T}(t)\tilde{\boldsymbol{\Pi}}^z\tilde{\boldsymbol{\epsilon}}_z(t)dt\right) \qquad (62)\end{aligned}$$

with

$$\begin{aligned}C &= \frac{1}{2}\log\det(\boldsymbol{\Pi}^\theta) - \frac{n_\theta}{2}\log 2\pi \\ &\quad + \frac{1}{2}\int_0^T \log\det(\tilde{\boldsymbol{\Pi}}^y) + \log\det(\tilde{\boldsymbol{\Pi}}^z) - (n_y + n_z)\log 2\pi dt \qquad (63) \\ &= \frac{1}{2}\log\det(\boldsymbol{\Pi}^\theta) - \frac{n_\theta}{2}\log 2\pi \\ &\quad + \frac{T}{2}\left(\log\det(\tilde{\boldsymbol{\Pi}}^y) + \log\det(\tilde{\boldsymbol{\Pi}}^z) - (n_y + n_z)\log 2\pi\right) \qquad (64)\end{aligned}$$

which may potentially depend on $\boldsymbol{\theta}$ and

$$\begin{aligned}\boldsymbol{\epsilon}_\theta &= \boldsymbol{\theta} - \boldsymbol{\eta}_\theta \qquad (65) \\ \tilde{\boldsymbol{\epsilon}}_y(t) &= \tilde{\mathbf{y}}(t) - \tilde{\mathbf{g}}(\tilde{\mathbf{z}}(t),\boldsymbol{\theta}) \qquad (66) \\ \tilde{\boldsymbol{\epsilon}}_z(t) &= \mathbf{D}\tilde{\mathbf{z}}(t) - \tilde{\mathbf{f}}(\tilde{\mathbf{z}}(t),\boldsymbol{\theta}) \qquad (67)\end{aligned}$$

where we have equated $\tilde{\boldsymbol{\mu}}_{\tilde{f}} = \tilde{\mathbf{f}}$.

4.5 Hierarchy

So far we have only considered flat models, but a generalisation to deep, i.e., hierarchical, dynamical models is straightforward. To achieve this we make the model functions on one level depend on the output of the level above. In particular, the model becomes

$$\begin{array}{ll} \tilde{\mathbf{y}}(t) = \tilde{\mathbf{g}}^1(\tilde{\mathbf{z}}^1(t),\tilde{\mathbf{v}}^2(t),\boldsymbol{\theta}) + \tilde{\boldsymbol{\omega}}_y & \mathbf{D}\tilde{\mathbf{z}}^1(t) = \tilde{\mathbf{f}}^1(\tilde{\mathbf{z}}^1(t),\tilde{\mathbf{v}}^2(t),\boldsymbol{\theta}) + \tilde{\boldsymbol{\omega}}_z^1 \\ \tilde{\mathbf{v}}^2(t) = \tilde{\mathbf{g}}^2(\tilde{\mathbf{z}}^2(t),\tilde{\mathbf{v}}^3(t),\boldsymbol{\theta}) + \tilde{\boldsymbol{\omega}}_v^2 & \mathbf{D}\tilde{\mathbf{z}}^2(t) = \tilde{\mathbf{f}}^2(\tilde{\mathbf{z}}^2(t),\tilde{\mathbf{v}}^3(t),\boldsymbol{\theta}) + \tilde{\boldsymbol{\omega}}_z^2 \\ \qquad\vdots & \qquad\qquad\vdots \\ \tilde{\mathbf{v}}^l(t) = \tilde{\mathbf{g}}^l(\tilde{\mathbf{z}}^l(t),\tilde{\mathbf{v}}^{l+1}(t),\boldsymbol{\theta}) + \tilde{\boldsymbol{\omega}}_v^l & \mathbf{D}\tilde{\mathbf{z}}^l(t) = \tilde{\mathbf{f}}^l(\tilde{\mathbf{z}}^l(t),\tilde{\mathbf{v}}^{l+1}(t),\boldsymbol{\theta}) + \tilde{\boldsymbol{\omega}}_z^l \end{array} \qquad (68)$$

where we define $\tilde{\mathbf{y}}(t) = \tilde{\mathbf{v}}^1(t)$ and $\tilde{\mathbf{v}}^{L+1}(t)$ on the top level L is a causal variable which may influence the top level dynamics via direct external input. Note that both $\tilde{\mathbf{z}}^l(t)$ and $\tilde{\mathbf{v}}^l(t)$ are time-dependent state variables. We call them dynamic variables, $\tilde{\mathbf{z}}^l(t)$, and output variables, $\tilde{\mathbf{v}}^l(t)$, respectively. $\tilde{\mathbf{z}}^l(t)$ represents the state of the dynamics on level l while $\tilde{\mathbf{v}}^l(t)$ represents the output of level l which influences level $l-1$, or corresponds to the observations for $l = 1$.

Because the noise in each level is independent from the noise in the other levels, the internal action becomes[2]

$$\begin{aligned}\bar{\mathrm{U}}(\mathbf{V},\mathbf{Z},\boldsymbol{\theta}) = \log p(\boldsymbol{\theta}) + \int_0^T & \log p(\tilde{\mathbf{v}}^{L+1}(t)) \\ & + \sum_{l=1}^{L} \log p(\tilde{\mathbf{v}}^l(t), \tilde{\mathbf{z}}^l(t) | \tilde{\mathbf{v}}^{l+1}(t), \boldsymbol{\theta}) dt \end{aligned} \tag{69}$$

where we have introduced the prior probability $p(\tilde{\mathbf{v}}^{L+1}(t)) \sim \mathcal{N}(\tilde{\boldsymbol{\eta}}_{v^{L+1}}, \tilde{\boldsymbol{\Psi}}^{v^{L+1}})$. All other distributions remain Gaussian, too, such that we get

$$\begin{aligned}\bar{\mathrm{U}}(\mathbf{V},\mathbf{Z},\boldsymbol{\theta}) = C^h - \frac{1}{2}\boldsymbol{\epsilon}_\theta^\mathsf{T}\boldsymbol{\Pi}^\theta\boldsymbol{\epsilon}_\theta - \frac{1}{2}\int_0^T & \tilde{\boldsymbol{\epsilon}}_{v^{L+1}}^\mathsf{T}(t)\tilde{\boldsymbol{\Pi}}^{v^{L+1}}\tilde{\boldsymbol{\epsilon}}_{v^{L+1}}(t) \\ & + \sum_{l=1}^{L} \tilde{\boldsymbol{\epsilon}}_{v^l}^\mathsf{T}(t)\tilde{\boldsymbol{\Pi}}^{v^l}\tilde{\boldsymbol{\epsilon}}_{v^l}(t) + \tilde{\boldsymbol{\epsilon}}_{z^l}^\mathsf{T}(t)\tilde{\boldsymbol{\Pi}}^{z^l}\tilde{\boldsymbol{\epsilon}}_{z^l}(t) dt\end{aligned} \tag{70}$$

with

$$\begin{aligned}C^h = & \frac{1}{2}\log\det(\boldsymbol{\Pi}^\theta) + \frac{T}{2}\log\det(\tilde{\boldsymbol{\Pi}}^{v^{L+1}}) - \frac{n_\theta + T n_{v^{L+1}}}{2}\log 2\pi \\ & + \frac{T}{2}\left(\sum_{l=1}^{L}\log\det(\tilde{\boldsymbol{\Pi}}^{v^l}) + \log\det(\tilde{\boldsymbol{\Pi}}^{z^l}) - (n_{v^l} + n_{z^l})\log 2\pi\right)\end{aligned} \tag{71}$$

and

$$\boldsymbol{\epsilon}_\theta = \boldsymbol{\theta} - \boldsymbol{\eta}_\theta \tag{72}$$

$$\tilde{\boldsymbol{\epsilon}}_{v^{L+1}}(t) = \tilde{\mathbf{v}}^{L+1}(t) - \tilde{\boldsymbol{\eta}}_{v^{L+1}} \tag{73}$$

$$\tilde{\boldsymbol{\epsilon}}_{v^l}(t) = \tilde{\mathbf{v}}^l(t) - \tilde{\mathbf{g}}^l(\tilde{\mathbf{z}}^l(t), \tilde{\mathbf{v}}^{l+1}(t), \boldsymbol{\theta}) \tag{74}$$

$$\tilde{\boldsymbol{\epsilon}}_{z^l}(t) = \mathbf{D}\tilde{\mathbf{z}}^l(t) - \tilde{\mathbf{f}}^l(\tilde{\mathbf{z}}^l(t), \tilde{\mathbf{v}}^{l+1}(t), \boldsymbol{\theta}). \tag{75}$$

4.6 Gradients of the Internal Action

4.6.1 Gradients with respect to Parameters

The gradients with respect to parameters obviously depend on what the parameters control. We for now stick with parameterising the functions $\tilde{\mathbf{g}}^l$ and $\tilde{\mathbf{f}}^l$ and consider learning of prior covariances later. Based on [5, eq. (85)] we get from eq. (70)

$$\begin{aligned}\frac{\partial\bar{\mathrm{U}}(\mathbf{V},\mathbf{Z},\boldsymbol{\theta})}{\partial\boldsymbol{\theta}} = -\left(\frac{\partial\boldsymbol{\epsilon}_\theta}{\partial\boldsymbol{\theta}}\right)^\mathsf{T}\boldsymbol{\Pi}^\theta\boldsymbol{\epsilon}_\theta - \int_0^T \sum_{l=1}^{L} & \left(\frac{\partial\tilde{\boldsymbol{\epsilon}}_{v^l}(t)}{\partial\boldsymbol{\theta}}\right)^\mathsf{T}\tilde{\boldsymbol{\Pi}}^{v^l}\tilde{\boldsymbol{\epsilon}}_{v^l}(t) \\ & + \left(\frac{\partial\tilde{\boldsymbol{\epsilon}}_{z^l}(t)}{\partial\boldsymbol{\theta}}\right)^\mathsf{T}\tilde{\boldsymbol{\Pi}}^{z^l}\tilde{\boldsymbol{\epsilon}}_{z^l}(t)dt\end{aligned} \tag{76}$$

[2] We here indicate the dependencies of the internal action on both, dynamic and output, variables by replacing **Y** with **V**. This conceals a bit the dependence on the observations **Y**, but it's still correct as **Y** is only absorbed into **V** as first level output variables.

The second order gradients should be of the form (considering a single parameter θ_i only)

$$\begin{aligned}\frac{\partial^2 \frac{1}{2}\boldsymbol{\epsilon}_\theta^\mathsf{T}\boldsymbol{\Pi}^\theta\boldsymbol{\epsilon}_\theta}{\partial\boldsymbol{\theta}\partial\theta_i} &= \frac{\partial}{\partial\theta_i}\left[\left(\frac{\partial\boldsymbol{\epsilon}_\theta}{\partial\boldsymbol{\theta}}\right)^\mathsf{T}\boldsymbol{\Pi}^\theta\boldsymbol{\epsilon}_\theta\right] \\ &= \left(\frac{\partial^2\boldsymbol{\epsilon}_\theta}{\partial\boldsymbol{\theta}\partial\theta_i}\right)^\mathsf{T}\boldsymbol{\Pi}^\theta\boldsymbol{\epsilon}_\theta + \left(\frac{\partial\boldsymbol{\epsilon}_\theta}{\partial\boldsymbol{\theta}}\right)^\mathsf{T}\boldsymbol{\Pi}^\theta\frac{\partial\boldsymbol{\epsilon}_\theta}{\partial\theta_i}. \end{aligned} \tag{77}$$

However, Friston assumes local linearity such that the second order terms disappear and what remains is (now summarised for all parameters)

$$\frac{\partial^2 \frac{1}{2}\boldsymbol{\epsilon}_\theta^\mathsf{T}\boldsymbol{\Pi}^\theta\boldsymbol{\epsilon}_\theta}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}} \approx \left(\frac{\partial\boldsymbol{\epsilon}_\theta}{\partial\boldsymbol{\theta}}\right)^\mathsf{T}\boldsymbol{\Pi}^\theta\frac{\partial\boldsymbol{\epsilon}_\theta}{\partial\boldsymbol{\theta}}. \tag{78}$$

It would be interesting to know what the advantage is of assuming local linearity of the prediction errors compared to assuming local linearity of $\bar{\mathrm{U}}$ directly, i.e., why not simply ignore second order gradients of $\bar{\mathrm{U}}$ completely? We can test this in the implementation later on. Anyway, the second order gradients of $\bar{\mathrm{U}}$ become

$$\begin{aligned}\frac{\partial^2\bar{\mathrm{U}}(\mathbf{V},\mathbf{Z},\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}} \approx -\left(\frac{\partial\boldsymbol{\epsilon}_\theta}{\partial\boldsymbol{\theta}}\right)^\mathsf{T}\boldsymbol{\Pi}^\theta\frac{\partial\boldsymbol{\epsilon}_\theta}{\partial\boldsymbol{\theta}} - \int_0^T \sum_{l=1}^{L}\left(\frac{\partial\tilde{\boldsymbol{\epsilon}}_{v^l}(t)}{\partial\boldsymbol{\theta}}\right)^\mathsf{T}\tilde{\boldsymbol{\Pi}}^{v^l}\frac{\partial\tilde{\boldsymbol{\epsilon}}_{v^l}(t)}{\partial\boldsymbol{\theta}} \\ + \left(\frac{\partial\tilde{\boldsymbol{\epsilon}}_{z^l}(t)}{\partial\boldsymbol{\theta}}\right)^\mathsf{T}\tilde{\boldsymbol{\Pi}}^{z^l}\frac{\partial\tilde{\boldsymbol{\epsilon}}_{z^l}(t)}{\partial\boldsymbol{\theta}}dt \end{aligned} \tag{79}$$

Notice that any higher order gradients of $\bar{\mathrm{U}}$ with respect to parameters will be 0 under the local linearity assumption for the prediction errors.

4.6.2 Gradients with respect to States

The gradients of the variational energy, e.g., eqs. (137) and (138) were defined for all time-dependent variables. This means that we have to combine all states into a long state vector

$$\tilde{\mathbf{s}}(t) = \begin{bmatrix} \tilde{\mathbf{v}}^2(t) \\ \vdots \\ \tilde{\mathbf{v}}^{L+1}(t) \\ \tilde{\mathbf{z}}^1(t) \\ \vdots \\ \tilde{\mathbf{z}}^L(t) \end{bmatrix}. \tag{80}$$

Then the variational energy of the parameters from eq. (136) becomes

$$\bar{\mathrm{V}}(\mathbf{Y},\boldsymbol{\theta}) \approx \bar{\mathrm{U}}(\mathbf{Y},\tilde{\boldsymbol{\mu}}_\mathbf{S},\boldsymbol{\theta}) + \frac{1}{2}\int_0^T \mathrm{Tr}\left(\frac{\partial^2\mathrm{U}(\tilde{\mathbf{y}}(t),\tilde{\boldsymbol{\mu}}_{s(t)},\boldsymbol{\theta})}{\partial\tilde{\mathbf{s}}(t)\partial\tilde{\mathbf{s}}(t)}\tilde{\boldsymbol{\Sigma}}^{s(t)}\right)dt \tag{81}$$

where we have introduced generalised coordinates and just replaced $\tilde{\mathbf{z}}$ by $\tilde{\mathbf{s}}$ (equivalently for the gradients of the variational energy). We will consider the gradients of the internal action with respect to output, $\tilde{\mathbf{v}}^l(t)$, and dynamic variables, $\tilde{\mathbf{z}}^l(t)$, sequentially. Note that the gradient of the internal action with respect to any particular time-dependent state is equal to the corresponding gradient of the internal energy:

$$\frac{\partial \bar{\mathrm{U}}(\mathbf{V}, \mathbf{Z}, \boldsymbol{\theta})}{\partial \tilde{\mathbf{v}}^l(t)} = \frac{\partial \mathrm{U}(\tilde{\mathbf{y}}(t), \tilde{\mathbf{s}}(t), \boldsymbol{\theta})}{\partial \tilde{\mathbf{v}}^l(t)}. \tag{82}$$

We are only interested in taking gradients with respect to the *hidden* output variables of levels $l = \{2, \dots, L+1\}$ which are (from eq. 70)

$$\begin{aligned}\frac{\partial \mathrm{U}(\tilde{\mathbf{y}}(t), \tilde{\mathbf{s}}(t), \boldsymbol{\theta})}{\partial \tilde{\mathbf{v}}^l(t)} = &-\left(\frac{\partial \tilde{\boldsymbol{\epsilon}}_{v^l}(t)}{\partial \tilde{\mathbf{v}}^l(t)}\right)^{\mathsf{T}} \tilde{\boldsymbol{\Pi}}^{v^l} \tilde{\boldsymbol{\epsilon}}_{v^l}(t) - \left(\frac{\partial \tilde{\boldsymbol{\epsilon}}_{v^{l-1}}(t)}{\partial \tilde{\mathbf{v}}^l(t)}\right)^{\mathsf{T}} \tilde{\boldsymbol{\Pi}}^{v^{l-1}} \tilde{\boldsymbol{\epsilon}}_{v^{l-1}}(t) \\ &- \left(\frac{\partial \tilde{\boldsymbol{\epsilon}}_{z^{l-1}}(t)}{\partial \tilde{\mathbf{v}}^l(t)}\right)^{\mathsf{T}} \tilde{\boldsymbol{\Pi}}^{z^{l-1}} \tilde{\boldsymbol{\epsilon}}_{z^{l-1}}(t).\end{aligned} \tag{83}$$

This illustrates that these gradients are responsible for propagating information from lower to higher levels during dynamic inference.

For dynamic variables in all levels $l = \{1, \dots, L\}$ the gradients are

$$\frac{\partial \mathrm{U}(\tilde{\mathbf{y}}(t), \tilde{\mathbf{s}}(t), \boldsymbol{\theta})}{\partial \tilde{\mathbf{z}}^l(t)} = -\left(\frac{\partial \tilde{\boldsymbol{\epsilon}}_{v^l}(t)}{\partial \tilde{\mathbf{z}}^l(t)}\right)^{\mathsf{T}} \tilde{\boldsymbol{\Pi}}^{v^l} \tilde{\boldsymbol{\epsilon}}_{v^l}(t) - \left(\frac{\partial \tilde{\boldsymbol{\epsilon}}_{z^l}(t)}{\partial \tilde{\mathbf{z}}^l(t)}\right)^{\mathsf{T}} \tilde{\boldsymbol{\Pi}}^{z^l} \tilde{\boldsymbol{\epsilon}}_{z^l}(t). \tag{84}$$

Also here local linearity assumptions are used to approximate the second order gradients as

$$\frac{\partial^2 \mathrm{U}(\tilde{\mathbf{y}}(t), \tilde{\mathbf{s}}(t), \boldsymbol{\theta})}{\partial \tilde{\mathbf{v}}^l(t) \partial \tilde{\mathbf{v}}^k(t)} \approx \left\{ \begin{array}{ll} \begin{array}{c} -\left(\frac{\partial \tilde{\boldsymbol{\epsilon}}_{v^{l-1}}(t)}{\partial \tilde{\mathbf{v}}^l(t)}\right)^{\mathsf{T}} \tilde{\boldsymbol{\Pi}}^{v^{l-1}} \frac{\partial \tilde{\boldsymbol{\epsilon}}_{v^{l-1}}(t)}{\partial \tilde{\mathbf{v}}^k(t)} \\ -\left(\frac{\partial \tilde{\boldsymbol{\epsilon}}_{z^{l-1}}(t)}{\partial \tilde{\mathbf{v}}^l(t)}\right)^{\mathsf{T}} \tilde{\boldsymbol{\Pi}}^{z^{l-1}} \frac{\partial \tilde{\boldsymbol{\epsilon}}_{z^{l-1}}(t)}{\partial \tilde{\mathbf{v}}^k(t)} \end{array} & k = l-1 \\ \hline \begin{array}{c} -\left(\frac{\partial \tilde{\boldsymbol{\epsilon}}_{v^l}(t)}{\partial \tilde{\mathbf{v}}^l(t)}\right)^{\mathsf{T}} \tilde{\boldsymbol{\Pi}}^{v^l} \frac{\partial \tilde{\boldsymbol{\epsilon}}_{v^l}(t)}{\partial \tilde{\mathbf{v}}^k(t)} \\ -\left(\frac{\partial \tilde{\boldsymbol{\epsilon}}_{v^{l-1}}(t)}{\partial \tilde{\mathbf{v}}^l(t)}\right)^{\mathsf{T}} \tilde{\boldsymbol{\Pi}}^{v^{l-1}} \frac{\partial \tilde{\boldsymbol{\epsilon}}_{v^{l-1}}(t)}{\partial \tilde{\mathbf{v}}^k(t)} \\ -\left(\frac{\partial \tilde{\boldsymbol{\epsilon}}_{z^{l-1}}(t)}{\partial \tilde{\mathbf{v}}^l(t)}\right)^{\mathsf{T}} \tilde{\boldsymbol{\Pi}}^{z^{l-1}} \frac{\partial \tilde{\boldsymbol{\epsilon}}_{z^{l-1}}(t)}{\partial \tilde{\mathbf{v}}^k(t)} \end{array} & k = l \\ \hline -\left(\frac{\partial \tilde{\boldsymbol{\epsilon}}_{v^l}(t)}{\partial \tilde{\mathbf{v}}^l(t)}\right)^{\mathsf{T}} \tilde{\boldsymbol{\Pi}}^{v^l} \frac{\partial \tilde{\boldsymbol{\epsilon}}_{v^l}(t)}{\partial \tilde{\mathbf{v}}^k(t)} & k = l+1 \\ \hline 0 & \text{else} \end{array} \right. \tag{85}$$

$$\frac{\partial^2 \mathrm{U}(\tilde{\mathbf{y}}(t), \tilde{\mathbf{s}}(t), \boldsymbol{\theta})}{\partial \tilde{\mathbf{z}}^l(t) \partial \tilde{\mathbf{z}}^k(t)} \approx \left\{ \begin{array}{ll} \begin{array}{c} -\left(\frac{\partial \tilde{\boldsymbol{\epsilon}}_{v^l}(t)}{\partial \tilde{\mathbf{z}}^l(t)}\right)^{\mathsf{T}} \tilde{\boldsymbol{\Pi}}^{v^l} \frac{\partial \tilde{\boldsymbol{\epsilon}}_{v^l}(t)}{\partial \tilde{\mathbf{z}}^l(t)} \\ -\left(\frac{\partial \tilde{\boldsymbol{\epsilon}}_{z^l}(t)}{\partial \tilde{\mathbf{z}}^l(t)}\right)^{\mathsf{T}} \tilde{\boldsymbol{\Pi}}^{z^l} \frac{\partial \tilde{\boldsymbol{\epsilon}}_{z^l}(t)}{\partial \tilde{\mathbf{z}}^l(t)} \end{array} & k = l \\ \hline 0 & \text{else} \end{array} \right. \tag{86}$$

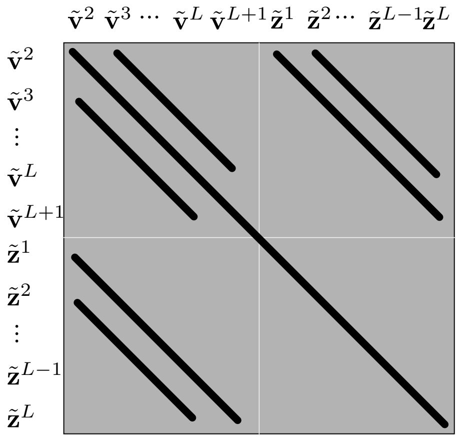

Figure 2: Schematic of sparsity structure of matrix of 2nd order gradients of internal energy with respect to states: $\frac{\partial^2 \mathrm{U}(\tilde{\mathbf{y}}(t),\tilde{\mathbf{s}}(t),\boldsymbol{\theta})}{\partial\tilde{\mathbf{s}}(t)\partial\tilde{\mathbf{s}}(t)}$. Labels at the sides correspond to $\tilde{\mathbf{v}}^l(t)$ and $\tilde{\mathbf{z}}^l(t)$, i.e., each of these elements stands for a vector and thus the elements of this schematic matrix are again matrices containing the gradients defined by eqs. (85-88).

$$\frac{\partial^2 \mathrm{U}(\tilde{\mathbf{y}}(t),\tilde{\mathbf{s}}(t),\boldsymbol{\theta})}{\partial\tilde{\mathbf{v}}^l(t)\partial\tilde{\mathbf{z}}^k(t)} \approx \begin{cases} -\left(\frac{\partial\tilde{\boldsymbol{\epsilon}}_{v^{l-1}}(t)}{\partial\tilde{\mathbf{v}}^l(t)}\right)^{\intercal} \tilde{\boldsymbol{\Pi}}^{v^{l-1}} \frac{\partial\tilde{\boldsymbol{\epsilon}}_{v^{l-1}}(t)}{\partial\tilde{\mathbf{z}}^k(t)} & \\ -\left(\frac{\partial\tilde{\boldsymbol{\epsilon}}_{z^{l-1}}(t)}{\partial\tilde{\mathbf{v}}^l(t)}\right)^{\intercal} \tilde{\boldsymbol{\Pi}}^{z^{l-1}} \frac{\partial\tilde{\boldsymbol{\epsilon}}_{z^{l-1}}(t)}{\partial\tilde{\mathbf{z}}^k(t)} & k = l-1 \\ -\left(\frac{\partial\tilde{\boldsymbol{\epsilon}}_{v^{l}}(t)}{\partial\tilde{\mathbf{v}}^l(t)}\right)^{\intercal} \tilde{\boldsymbol{\Pi}}^{v^{l}} \frac{\partial\tilde{\boldsymbol{\epsilon}}_{v^{l}}(t)}{\partial\tilde{\mathbf{z}}^k(t)} & k = l \\ 0 & \text{else} \end{cases} \tag{87}$$

$$\frac{\partial^2 \mathrm{U}(\tilde{\mathbf{y}}(t),\tilde{\mathbf{s}}(t),\boldsymbol{\theta})}{\partial\tilde{\mathbf{z}}^l(t)\partial\tilde{\mathbf{v}}^k(t)} \approx \begin{cases} -\left(\frac{\partial\tilde{\boldsymbol{\epsilon}}_{v^{l}}(t)}{\partial\tilde{\mathbf{z}}^l(t)}\right)^{\intercal} \tilde{\boldsymbol{\Pi}}^{v^{l}} \frac{\partial\tilde{\boldsymbol{\epsilon}}_{v^{l}}(t)}{\partial\tilde{\mathbf{v}}^k(t)} & k = l \\ -\left(\frac{\partial\tilde{\boldsymbol{\epsilon}}_{v^{l}}(t)}{\partial\tilde{\mathbf{z}}^l(t)}\right)^{\intercal} \tilde{\boldsymbol{\Pi}}^{v^{l}} \frac{\partial\tilde{\boldsymbol{\epsilon}}_{v^{l}}(t)}{\partial\tilde{\mathbf{v}}^k(t)} & \\ -\left(\frac{\partial\tilde{\boldsymbol{\epsilon}}_{z^{l}}(t)}{\partial\tilde{\mathbf{z}}^l(t)}\right)^{\intercal} \tilde{\boldsymbol{\Pi}}^{z^{l}} \frac{\partial\tilde{\boldsymbol{\epsilon}}_{z^{l}}(t)}{\partial\tilde{\mathbf{v}}^k(t)} & k = l+1 \\ 0 & \text{else} \end{cases} \tag{88}$$

As a result we get a (potentially very large) sparse matrix $\frac{\partial^2 \mathrm{U}(\tilde{\mathbf{y}}(t),\tilde{\mathbf{s}}(t),\boldsymbol{\theta})}{\partial\tilde{\mathbf{s}}(t)\partial\tilde{\mathbf{s}}(t)}$ with a structure as visualised in Fig. 2.

4.6.3 Mixed State-Parameter Gradients

5 The Free Energy

We differentiate between time-dependent (quickly varying) hidden variables (states) $\mathbf{z}$ and time-independent (slowly varying) hidden variables (parameters) $\boldsymbol{\theta}$.

$$\mathbf{x} = \left[\begin{array}{c} \mathbf{z} \\ \boldsymbol{\theta} \end{array} \right] \in \mathbb{R}^{n_x} \tag{89}$$

$$\begin{aligned} F(\mathbf{y}) &= \langle \log p(\mathbf{y}, \mathbf{x}) \rangle_{q(\mathbf{x})} - \langle \log q(\mathbf{x}) \rangle_{q(\mathbf{x})} \\ &= \langle \mathrm{U}(\mathbf{y}, \mathbf{x}) \rangle_{q(\mathbf{x})} + \mathrm{H}(q(\mathbf{x})) \end{aligned} \tag{90}$$

In the dynamic formulation, the free energy becomes free action and falls into time-dependent and time-independent parts:

$$\begin{aligned} \bar{F}(\mathbf{Y}) = &\int_0^T \langle \mathrm{U}(\mathbf{y}(t), \mathbf{z}(t), \boldsymbol{\theta}) \rangle_{q(\mathbf{z}(t))q(\boldsymbol{\theta})} \, dt + \langle \mathrm{U}(\boldsymbol{\theta}) \rangle_{q(\boldsymbol{\theta})} \\ &+ \int_0^T \mathrm{H}(q(\mathbf{z}(t))) dt + \mathrm{H}(q(\boldsymbol{\theta})) \\ = &\int_0^T \langle \mathrm{U}(\mathbf{y}(t), \mathbf{z}(t), \boldsymbol{\theta}) \rangle_{q(\mathbf{z}(t))q(\boldsymbol{\theta})} + \mathrm{H}(q(\mathbf{z}(t))) dt \\ &+ \langle \mathrm{U}(\boldsymbol{\theta}) \rangle_{q(\boldsymbol{\theta})} + \mathrm{H}(q(\boldsymbol{\theta})) \end{aligned} \tag{91}$$

This derives from the model:

$$p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\theta}) = p(\boldsymbol{\theta}) \prod_{t=0}^{T} p(\mathbf{y}(t), \mathbf{z}(t) | \boldsymbol{\theta}) \tag{92}$$

$$\begin{aligned} \bar{\mathrm{U}}(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\theta}) &= \log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\theta}) \\ &= \log p(\boldsymbol{\theta}) + \int_{t=0}^{T} \log p(\mathbf{y}(t), \mathbf{z}(t) | \boldsymbol{\theta}) dt \\ &= \mathrm{U}(\boldsymbol{\theta}) + \int_{t=0}^{T} \mathrm{U}(\mathbf{y}(t), \mathbf{z}(t), \boldsymbol{\theta}) dt \end{aligned} \tag{93}$$

where we have abused the matrix-vector notation by letting matrix formatting stand for a variable which collects values in the continuous time interval $[0, T]$ and assume that the product is defined over this continuous interval, too. The joint distribution is defined as $p(\mathbf{y}(t), \mathbf{z}(t)|\boldsymbol{\theta}) = p(\mathbf{y}(t)|\mathbf{z}(t), \boldsymbol{\theta})p(\mathbf{z}(t)|\boldsymbol{\theta})$. However, we have kept the notation as joint distribution as in [1], because the prior $p(\mathbf{z}(t)|\boldsymbol{\theta})$ is not explicitly defined there (see description of generalised filtering below).

6 Friston's Variational Laplace Approximation

6.1 Static Case

The aim is to simplify the free energy by assuming that $q(\mathbf{x})$ is Gaussian, i.e.,

$$q(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}). \tag{94}$$

Then, we can immediately look up the entropy of q [5, eq. (366)] as

$$\begin{aligned}\mathrm{H}(q(\mathbf{x})) &= \frac{1}{2}\left(\log[2\pi^{n_x}\det(\boldsymbol{\Sigma})] + n_x\right) \\ &= \frac{1}{2}\left(\log\det(\boldsymbol{\Sigma}) + n_x \log 2\pi e\right).\end{aligned} \tag{95}$$

The crucial trick is to approximate $\mathrm{U}(\mathbf{y}, \mathbf{x})$ (Friston calls it the internal or Gibb's energy) with a Taylor series expansion up to second order around the mean of q. Because the observations $\mathbf{y}$ are assumed to be fixed here, we drop $\mathbf{y}$ from the list of variables of U

$$\mathrm{U}(\mathbf{x}) \approx \mathrm{U}(\boldsymbol{\mu}) + (\mathbf{x} - \boldsymbol{\mu})^{\mathsf{T}}\mathrm{U}_{\mathbf{x}}(\boldsymbol{\mu}) + \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^{\mathsf{T}}\mathrm{U}_{\mathbf{xx}}(\boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu}) \tag{96}$$

where $\mathrm{U}_{\mathbf{x}} = \frac{\partial \mathrm{U}}{\partial \mathbf{x}}$ is the gradient of U and $\mathrm{U}_{\mathbf{xx}} = \frac{\partial^2 \mathrm{U}}{\partial \mathbf{x} \partial \mathbf{x}}$ is the Hessian of U which are evaluated at $\boldsymbol{\mu}$ here. We are actually interested in the expectation of U

$$\begin{aligned}\langle \mathrm{U}(\mathbf{x})\rangle_{q(\mathbf{x})} \approx{}& \langle \mathrm{U}(\boldsymbol{\mu})\rangle_{q(\mathbf{x})} + \left\langle (\mathbf{x} - \boldsymbol{\mu})^{\mathsf{T}}\mathrm{U}_{\mathbf{x}}(\boldsymbol{\mu})\right\rangle_{q(\mathbf{x})} \\ &+ \left\langle \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^{\mathsf{T}}\mathrm{U}_{\mathbf{xx}}(\boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})\right\rangle_{q(\mathbf{x})} \end{aligned}\tag{97}$$

$$= \mathrm{U}(\boldsymbol{\mu}) + \left(\langle \mathbf{x}\rangle_{q(\mathbf{x})} - \boldsymbol{\mu}\right)^{\mathsf{T}}\mathrm{U}_{\mathbf{x}}(\boldsymbol{\mu}) + \frac{1}{2}\mathrm{Tr}\left(\mathrm{U}_{\mathbf{xx}}(\boldsymbol{\mu})\boldsymbol{\Sigma}\right) \tag{98}$$

$$= \mathrm{U}(\boldsymbol{\mu}) + \frac{1}{2}\mathrm{Tr}\left(\mathrm{U}_{\mathbf{xx}}(\boldsymbol{\mu})\boldsymbol{\Sigma}\right) \tag{99}$$

where we have solved the expectation of the quadratic term using standard results for the multivariate Gaussian distribution [5, eq. (357)]. This result can be easily transferred to the case where q factorises

$$q(\mathbf{x}) = \prod_i q_i(\mathbf{x}^i) \tag{100}$$

where $\mathbf{x}^i \in \mathbb{R}^{n_{x_i}}$ is a subset of the hidden variables in our model, i.e., $\mathbf{x}^1$ could be $\mathbf{z}$ and $\mathbf{x}^2$ could be $\boldsymbol{\theta}$, but also further subdivisions can be used. Actually, we can still represent the factorised $q(\mathbf{x})$ with a Gaussian distribution over all variables. We just need to give $\boldsymbol{\Sigma}$ a suitable block-diagonal structure (assuming a suitable ordering of variables) which ensures the independence of the subsets of hidden variables $\mathbf{x}^i$. These blocks of $\boldsymbol{\Sigma}$ are $\boldsymbol{\Sigma}^i$ and it is easy to see then that

$$\langle \mathrm{U}(\mathbf{x})\rangle_{q(\mathbf{x})} \approx \mathrm{U}(\boldsymbol{\mu}) + \frac{1}{2}\sum_i \mathrm{Tr}\left(\mathrm{U}_{\mathbf{x}^i\mathbf{x}^i}(\boldsymbol{\mu})\boldsymbol{\Sigma}^i\right). \tag{101}$$

These considerations also apply to the entropy term. Consequently, we have approximated the free energy as

$$\begin{aligned}F(\mathbf{y}) &\approx \mathrm{U}(\boldsymbol{\mu}) + \frac{1}{2}\sum_i \mathrm{Tr}\left(\mathrm{U}_{\mathbf{x}^i\mathbf{x}^i}(\boldsymbol{\mu})\boldsymbol{\Sigma}^i\right) + \frac{1}{2}\sum_i \left(\log\det(\boldsymbol{\Sigma}^i) + n_{x^i}\log 2\pi e\right) \\ &= \mathcal{F}(\mathbf{y}).\end{aligned} \tag{102}$$

Note that this is based on an approximation of $\mathrm{U}(\mathbf{y}, \mathbf{x}) = \log p(\mathbf{y}|\mathbf{x})p(\mathbf{x}) = \log p(\mathbf{x}|\mathbf{y}) + c$ while usually the Laplace approximation is based on directly approximating the log-posterior $\log p(\mathbf{x}|\mathbf{y})$. However, you see that these only differ by a constant which leads to equivalent results.

The aim of variational inference is to find $q(\mathbf{x})$ which maximises $F(\mathbf{y})$ which in our case is approximated by $\mathcal{F}(\mathbf{y})$. As q is Gaussian we need to find its mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. Let us consider $\boldsymbol{\Sigma}$ first. The gradient of $\mathcal{F}(\mathbf{y})$ with respect to $\boldsymbol{\Sigma}$ is[3].

$$\mathcal{F}_{\boldsymbol{\Sigma}}(\mathbf{y}) = \frac{1}{2}\mathrm{U}_{\mathbf{xx}}(\boldsymbol{\mu}) + \frac{1}{2}\boldsymbol{\Sigma}^{-1} \tag{103}$$

where we have used [5, eq. (96)] and that $\frac{\partial \log\det(\boldsymbol{\Sigma})}{\partial \boldsymbol{\Sigma}} = \boldsymbol{\Sigma}^{-1}$ based on [5, eq. (43)]. Setting the gradient to 0 leads to

$$\boldsymbol{\Sigma}^{-1} = -\mathrm{U}_{\mathbf{xx}}(\boldsymbol{\mu}) \tag{104}$$

such that

$$\begin{aligned}
\mathcal{F}(\mathbf{y}) &= \mathrm{U}(\boldsymbol{\mu}) - \frac{1}{2}\mathrm{Tr}\left(\mathrm{U}_{\mathbf{xx}}(\boldsymbol{\mu})\mathrm{U}_{\mathbf{xx}}^{-1}(\boldsymbol{\mu})\right) + \frac{1}{2}\left(\log\det(-\mathrm{U}_{\mathbf{xx}}^{-1}(\boldsymbol{\mu})) + n_x \log 2\pi e\right) \\
&= \mathrm{U}(\boldsymbol{\mu}) - \frac{1}{2}n_x + \frac{1}{2}\left(\log\det(-\mathrm{U}_{\mathbf{xx}}^{-1}(\boldsymbol{\mu})) + n_x \log 2\pi e\right) \\
&= \mathrm{U}(\boldsymbol{\mu}) + \frac{1}{2}\left(\log\det(-\mathrm{U}_{\mathbf{xx}}^{-1}(\boldsymbol{\mu})) + n_x \log 2\pi\right) \qquad (105) \\
&= \mathrm{U}(\boldsymbol{\mu}) + \frac{1}{2}\sum_i \left(\log\det(-\mathrm{U}_{\mathbf{x}^i\mathbf{x}^i}^{-1}(\boldsymbol{\mu})) + n_{x^i} \log 2\pi\right) \qquad (106)
\end{aligned}$$

where we have reintroduced the block structure of $\boldsymbol{\Sigma}$ for different hidden variables in the last line.

6.2 Dynamic Case

We now apply these derivations in the dynamic context. We start from the free action as defined by Friston (repeated from eq. (91)):

$$\bar{F}(\mathbf{Y}) = \int_0^T \langle \mathrm{U}(\mathbf{y}(t), \mathbf{z}(t), \boldsymbol{\theta})\rangle_{q(\mathbf{z}(t))q(\boldsymbol{\theta})} + \mathrm{H}(q(\mathbf{z}(t)))dt + \langle \mathrm{U}(\boldsymbol{\theta})\rangle_{q(\boldsymbol{\theta})} + \mathrm{H}(q(\boldsymbol{\theta})).$$

The basic approach is to note that the parts of this equation are in the form of the static free energy above. Therefore, we can directly see from eq. (102) that (remember that $\mathrm{U}(\boldsymbol{\theta})$ is just the prior for $\boldsymbol{\theta}$ which is independent of any data)

$$\begin{aligned}
F^\theta &= \langle \mathrm{U}(\boldsymbol{\theta})\rangle_{q(\boldsymbol{\theta})} + \mathrm{H}(q(\boldsymbol{\theta})) \qquad (107) \\
&\approx \mathrm{U}(\boldsymbol{\mu}_\theta) + \frac{1}{2}\mathrm{Tr}\left(\mathrm{U}_{\boldsymbol{\theta\theta}}(\boldsymbol{\mu}_\theta)\boldsymbol{\Sigma}^\theta\right) + \frac{1}{2}\left(\log\det(\boldsymbol{\Sigma}^\theta) + n_\theta \log 2\pi e\right) \qquad (108) \\
&= \mathcal{F}^\theta. \qquad (109)
\end{aligned}$$

[3] For the sake of simplicity of presentation we here ignore the potential block structure of $\boldsymbol{\Sigma}$, but everything goes through equivalently for any $\boldsymbol{\Sigma}^i$

Similarly, we consider the part for the states inside the integral

$$
\begin{aligned}
F^{z}(\mathbf{y}(t)) &= \langle \mathrm{U}(\mathbf{y}(t), \mathbf{z}(t), \boldsymbol{\theta}) \rangle_{q(\mathbf{z}(t))q(\boldsymbol{\theta})} + \mathrm{H}(q(\mathbf{z}(t))) && (110) \\
&= \langle \mathrm{U}(\mathbf{y}(t), \mathbf{x}) \rangle_{q(\mathbf{x})} + \mathrm{H}(q(\mathbf{z}(t))) && (111) \\
&\approx \mathrm{U}(\mathbf{y}(t), \boldsymbol{\mu}) + \frac{1}{2}\mathrm{Tr}\left(\mathrm{U}_{\mathbf{xx}}(\mathbf{y}(t), \boldsymbol{\mu})\boldsymbol{\Sigma}\right) \\
&\quad + \frac{1}{2}\left(\log\det(\boldsymbol{\Sigma}^{z(t)}) + n_z \log 2\pi e\right) && (112) \\
&= \mathrm{U}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_{\theta}) + \frac{1}{2}\mathrm{Tr}\left(\mathrm{U}_{\mathbf{z}(t)\mathbf{z}(t)}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_{\theta})\boldsymbol{\Sigma}^{z(t)}\right) \\
&\quad + \frac{1}{2}\mathrm{Tr}\left(\mathrm{U}_{\boldsymbol{\theta\theta}}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_{\theta})\boldsymbol{\Sigma}^{\theta}\right) \\
&\quad + \frac{1}{2}\left(\log\det(\boldsymbol{\Sigma}^{z(t)}) + n_z \log 2\pi e\right) && (113) \\
&= \mathcal{F}^{z}(\mathbf{y}(t)) && (114)
\end{aligned}
$$

where we have used the same argument as for eq. (101) to separate the 2nd order gradients with respect to states and parameters based on their independence in

q. Putting things together we get

$$
\begin{aligned}
\bar{F}(\mathbf{Y}) &= \int_0^T F^z(\mathbf{y}(t))dt + F^\theta && (115)\\
&\approx \int_0^T \mathcal{F}^z(\mathbf{y}(t))dt + \mathcal{F}^\theta && (116)\\
&= \int_0^T \mathrm{U}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta) + \frac{1}{2}\mathrm{Tr}\left(\mathrm{U}_{\mathbf{z}(t)\mathbf{z}(t)}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta)\boldsymbol{\Sigma}^{z(t)}\right)\\
&\quad + \frac{1}{2}\mathrm{Tr}\left(\mathrm{U}_{\boldsymbol{\theta\theta}}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta)\boldsymbol{\Sigma}^\theta\right) + \frac{1}{2}\log\det(\boldsymbol{\Sigma}^{z(t)})dt\\
&\quad + \mathrm{U}(\boldsymbol{\mu}_\theta) + \frac{1}{2}\mathrm{Tr}\left(\mathrm{U}_{\boldsymbol{\theta\theta}}(\boldsymbol{\mu}_\theta)\boldsymbol{\Sigma}^\theta\right)\\
&\quad + \frac{1}{2}\left(\log\det(\boldsymbol{\Sigma}^\theta) + (Tn_z + n_\theta)\log 2\pi e\right) && (117)\\
&= \int_0^T \mathrm{U}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta)dt + \mathrm{U}(\boldsymbol{\mu}_\theta)\\
&\quad + \frac{1}{2}\int_0^T \mathrm{Tr}\left(\mathrm{U}_{\mathbf{z}(t)\mathbf{z}(t)}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta)\boldsymbol{\Sigma}^{z(t)}\right)dt\\
&\quad + \frac{1}{2}\int_0^T \mathrm{Tr}\left(\mathrm{U}_{\boldsymbol{\theta\theta}}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta)\boldsymbol{\Sigma}^\theta\right)dt + \frac{1}{2}\mathrm{Tr}\left(\mathrm{U}_{\boldsymbol{\theta\theta}}(\boldsymbol{\mu}_\theta)\boldsymbol{\Sigma}^\theta\right)\\
&\quad + \frac{1}{2}\left(\int_0^T \log\det(\boldsymbol{\Sigma}^{z(t)})dt + \log\det(\boldsymbol{\Sigma}^\theta) + (Tn_z + n_\theta)\log 2\pi e\right) && (118)\\
&= \int_0^T \mathrm{U}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta)dt + \mathrm{U}(\boldsymbol{\mu}_\theta)\\
&\quad + \frac{1}{2}\int_0^T \mathrm{Tr}\left(\mathrm{U}_{\mathbf{z}(t)\mathbf{z}(t)}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta)\boldsymbol{\Sigma}^{z(t)}\right)dt\\
&\quad + \frac{1}{2}\mathrm{Tr}\left(\left(\int_0^T \mathrm{U}_{\boldsymbol{\theta\theta}}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta)dt + \mathrm{U}_{\boldsymbol{\theta\theta}}(\boldsymbol{\mu}_\theta)\right)\boldsymbol{\Sigma}^\theta\right)\\
&\quad + \frac{1}{2}\left(\int_0^T \log\det(\boldsymbol{\Sigma}^{z(t)})dt + \log\det(\boldsymbol{\Sigma}^\theta) + (Tn_z + n_\theta)\log 2\pi e\right) && (119)\\
&= \bar{\mathcal{F}}(\mathbf{Y}). && (120)
\end{aligned}
$$

In analogy to eq. (104) we see that the posterior state covariances which maximise $\bar{\mathcal{F}}(\mathbf{Y})$ are given as

$$\boldsymbol{\Sigma}^{z(t)^{-1}} = -\mathrm{U}_{\mathbf{z}(t)\mathbf{z}(t)}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta). \tag{121}$$

Equivalently, we can derive the posterior parameter covariance:

$$\boldsymbol{\Sigma}^{\theta^{-1}} = -\left(\int_0^T \mathrm{U}_{\boldsymbol{\theta\theta}}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta)dt + \mathrm{U}_{\boldsymbol{\theta\theta}}(\boldsymbol{\mu}_\theta)\right). \tag{122}$$

Plugging in the result for the posterior state covariance we can simplify eq. (119) as above

$$\bar{\mathcal{F}}(\mathbf{Y}) = \int_0^T \mathrm{U}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta) dt + \mathrm{U}(\boldsymbol{\mu}_\theta) + \frac{1}{2}\left(\int_0^T \log\det(\mathbf{\Sigma}^{z(t)}) dt + \log\det(\mathbf{\Sigma}^\theta) + (Tn_z + n_\theta)\log 2\pi\right). \quad (123)$$

7 Finding the Posterior Mode of the Parameters

It is now tempting to implement the parameter learning by gradient ascent on $\bar{\mathcal{F}}(\mathbf{Y})$ with respect to the posterior mode of the parameters $\boldsymbol{\mu}_\theta$. The gradient of the approximated free action is (for simplicity of treatment we consider only a single parameter μ_{θ_i}):

$$\bar{\mathcal{F}}_{\mu_{\theta_i}}(\mathbf{Y}) = \bar{\mathrm{U}}_{\theta_i}(\mathbf{Y}, \boldsymbol{\mu}_\mathbf{Z}, \boldsymbol{\mu}_\theta) + \frac{1}{2}\int_0^T \mathrm{Tr}\left(\mathrm{U}_{\mathbf{z}(t)\mathbf{z}(t)\theta_i}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta)\mathbf{\Sigma}^{z(t)}\right) dt + \frac{1}{2}\mathrm{Tr}\left(\bar{\mathrm{U}}_{\boldsymbol{\theta}\boldsymbol{\theta}\theta_i}(\mathbf{Y}, \boldsymbol{\mu}_\mathbf{Z}, \boldsymbol{\mu}_\theta)\mathbf{\Sigma}^\theta\right). \quad (124)$$

We derived this gradient using

$$\frac{\partial \log\det(\mathbf{\Sigma})}{\partial x} = \frac{1}{\det(\mathbf{\Sigma})}\det(\mathbf{\Sigma})\mathrm{Tr}\left(\mathbf{\Sigma}^{-1}\frac{\partial \mathbf{\Sigma}}{\partial x}\right) \quad (125)$$

$$= \mathrm{Tr}\left(\mathbf{\Sigma}^{-1}\mathbf{\Sigma}\frac{\partial \mathbf{U}}{\partial x}\mathbf{\Sigma}\right) \quad (126)$$

$$= \mathrm{Tr}\left(\frac{\partial \mathbf{U}}{\partial x}\mathbf{\Sigma}\right) \quad (127)$$

together with $\mathbf{\Sigma} = -\mathbf{U}^{-1}$ based on [5, eqs. (41) and (53)]. The second order gradients with respect to θ_j can be derived using

$$\frac{\partial \frac{\partial \mathbf{U}}{\partial x}\mathbf{\Sigma}}{\partial y} = \frac{\partial^2 \mathbf{U}}{\partial x \partial y}\mathbf{\Sigma} - \frac{\partial \mathbf{U}}{\partial x}\frac{\partial \mathbf{U}^{-1}}{\partial y} \quad (128)$$

$$= \frac{\partial^2 \mathbf{U}}{\partial x \partial y}\mathbf{\Sigma} + \frac{\partial \mathbf{U}}{\partial x}\mathbf{U}^{-1}\frac{\partial \mathbf{U}}{\partial y}\mathbf{U}^{-1} \quad (129)$$

$$= \frac{\partial^2 \mathbf{U}}{\partial x \partial y}\mathbf{\Sigma} + \frac{\partial \mathbf{U}}{\partial x}\mathbf{\Sigma}\frac{\partial \mathbf{U}}{\partial y}\mathbf{\Sigma} \quad (130)$$

what leads to

$$\bar{\mathcal{F}}_{\mu_{\theta_i}\mu_{\theta_j}}(\mathbf{Y}) = \bar{\mathrm{U}}_{\theta_i\theta_j}(\mathbf{Y}, \boldsymbol{\mu}_\mathbf{Z}, \boldsymbol{\mu}_\theta) + \frac{1}{2}\int_0^T \mathrm{Tr}\left(\mathrm{U}_{\mathbf{z}(t)\mathbf{z}(t)\theta_i\theta_j}\mathbf{\Sigma}^{z(t)} - \mathrm{U}_{\mathbf{z}(t)\mathbf{z}(t)\theta_i}\mathbf{\Sigma}^{z(t)}\mathrm{U}_{\mathbf{z}(t)\mathbf{z}(t)\theta_j}\mathbf{\Sigma}^{z(t)}\right) dt + \frac{1}{2}\mathrm{Tr}\left(\bar{\mathrm{U}}_{\boldsymbol{\theta}\boldsymbol{\theta}\theta_i\theta_j}\mathbf{\Sigma}^\theta + \bar{\mathrm{U}}_{\boldsymbol{\theta}\boldsymbol{\theta}\theta_i}\mathbf{\Sigma}^\theta\bar{\mathrm{U}}_{\boldsymbol{\theta}\boldsymbol{\theta}\theta_j}\mathbf{\Sigma}^\theta\right) \quad (131)$$

where we have dropped the dependencies of U for convenience of writing. However, these gradients also depend on the posterior modes of the states $\boldsymbol{\mu}_\mathbf{Z}$ and maximisation of $\bar{\mathcal{F}}(\mathbf{Y})$, therefore, will have to iterate between maximising posterior modes of states and posterior modes of parameters.

It is clear now that we need third and fourth order gradients of U to maximise $\bar{\mathcal{F}}(\mathbf{Y})$ which makes this optimisation computationally very expensive. For example, to compute all elements of the matrix defined by $[\cdot]_{ij} = \text{Tr}\left(\bar{\text{U}}_{\boldsymbol{\theta\theta}\theta_i}\boldsymbol{\Sigma}^\theta\bar{\text{U}}_{\boldsymbol{\theta\theta}\theta_j}\boldsymbol{\Sigma}^\theta\right)$ is of complexity $O(n_\theta^6)$ where $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta \times 1}$ (when ignoring the cost of computing the gradients of $\bar{\text{U}}_{\boldsymbol{\theta\theta}}$). Similarly, computing all elements defined by the second line of eq. (131) has complexity $O(n_\theta^2 n_z^4)$. Computing the first order gradient $\bar{\mathcal{F}}_{\boldsymbol{\theta}}(\mathbf{Y})$ still has complexity $O(n_\theta n_z^2)$, or $O(n_\theta^3)$, respectively, depending on whether n_z or n_θ is larger.

Friston finds the posterior mode in a different way. Equivalently to the dynamic solution for the states he uses the variational result that $\bar{F}(\mathbf{Y})$ is maximised with respect to $q(\boldsymbol{\theta})$, while holding $q(\mathbf{Z})$ fixed, when

$$q(\boldsymbol{\theta}) = \frac{1}{Z^\theta}\exp(\bar{\text{V}}(\mathbf{Y}, \boldsymbol{\theta})) \tag{132}$$

where

$$\begin{aligned}
\bar{\text{V}}(\mathbf{Y}, \boldsymbol{\theta}) &= \left\langle \bar{\text{U}}(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\theta}) \right\rangle_{q(\mathbf{Z})} && (133)\\
&= \text{U}(\boldsymbol{\theta}) + \int_0^T \left\langle \text{U}(\mathbf{y}(t), \mathbf{z}(t), \boldsymbol{\theta}) \right\rangle_{q(\mathbf{z}(t))} dt && (134)\\
&\approx \text{U}(\boldsymbol{\theta}) + \int_0^T \text{U}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\theta}) dt \\
&\quad + \frac{1}{2}\int_0^T \text{Tr}\left(\text{U}_{\mathbf{z}(t)\mathbf{z}(t)}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\theta})\boldsymbol{\Sigma}^{z(t)}\right) dt && (135)\\
&= \bar{\text{U}}(\mathbf{Y}, \boldsymbol{\mu}_{\mathbf{Z}}, \boldsymbol{\theta}) + \frac{1}{2}\int_0^T \text{Tr}\left(\text{U}_{\mathbf{z}(t)\mathbf{z}(t)}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\theta})\boldsymbol{\Sigma}^{z(t)}\right) dt && (136)
\end{aligned}$$

where we have used the independence of the $q(\mathbf{z}(t))$ and the first line of eq. (113) (a result from the Laplace approximation). We only need to find the (rather "a") mode of $q(\boldsymbol{\theta})$ from which we already can compute $\boldsymbol{\Sigma}^\theta$ (provided we also have the modes $\boldsymbol{\mu}_{z(t)}$) which would complete our Laplace approximation. Also, it is clear from eq. (132) that a maximum of $q(\boldsymbol{\theta})$ (a mode) is equal to a maximum of $\bar{\text{V}}(\mathbf{Y}, \boldsymbol{\theta})$, because the exponential function is monotonically increasing. Hence, Friston suggests to maximise $\bar{\text{V}}(\mathbf{Y}, \boldsymbol{\theta})$ instead of directly maximising $\bar{\mathcal{F}}(\mathbf{Y})$. At this point it is instructive to remember that the interpretation of the two optimisations is different. We maximise $\bar{\mathcal{F}}(\mathbf{Y})$ directly with respect to the posterior mode $\boldsymbol{\mu}_\theta$ which was possible because of the Laplace approximation for both states and parameters. On the other hand, we maximise $\bar{\text{V}}(\mathbf{Y}, \boldsymbol{\theta})$ with respect to the parameters used in the model $\boldsymbol{\theta}$ in order to find a posterior mode $\boldsymbol{\mu}_\theta$ based on a Laplace approximation around the posterior mode of the states only. Notice that this approach makes no assumptions about the form of $q(\boldsymbol{\theta})$ while we assumed a Gaussian $q(\boldsymbol{\theta})$ in deriving $\bar{\mathcal{F}}(\mathbf{Y})$.

The first and second order gradients of $\bar{\text{V}}(\mathbf{Y}, \boldsymbol{\theta})$ are

$$\bar{\text{V}}_{\theta_i}(\mathbf{Y}, \boldsymbol{\theta}) = \bar{\text{U}}_{\theta_i}(\mathbf{Y}, \boldsymbol{\mu}_{\mathbf{Z}}, \boldsymbol{\theta}) + \frac{1}{2}\int_0^T \text{Tr}\left(\text{U}_{\mathbf{z}(t)\mathbf{z}(t)\theta_i}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\theta})\boldsymbol{\Sigma}^{z(t)}\right) dt \tag{137}$$

$$\bar{\mathrm{V}}_{\theta_i\theta_j}(\mathbf{Y},\boldsymbol{\theta}) = \bar{\mathrm{U}}_{\theta_i\theta_j}(\mathbf{Y},\boldsymbol{\mu}_{\mathbf{Z}},\boldsymbol{\theta}) + \frac{1}{2}\int_0^T \mathrm{Tr}\left(\mathrm{U}_{\mathbf{z}(t)\mathbf{z}(t)\theta_i\theta_j}(\mathbf{y}(t),\boldsymbol{\mu}_{z(t)},\boldsymbol{\theta})\boldsymbol{\Sigma}^{z(t)}\right) dt. \tag{138}$$

Therefore, this approach saves us the computation of $\frac{1}{2}\mathrm{Tr}\left(\bar{\mathrm{U}}_{\boldsymbol{\theta}\boldsymbol{\theta}\theta_i}(\mathbf{Y},\boldsymbol{\mu}_{\mathbf{Z}},\boldsymbol{\mu}_\theta)\boldsymbol{\Sigma}^\theta\right)$ in the first order gradients (compare eqs. 124 and 137) and also considerably simplifies computation of the second order gradients.

Shouldn't both approaches lead to a mode $\boldsymbol{\mu}_\theta$ which maximises $\bar{\mathcal{F}}(\mathbf{Y})$? Eventually, i.e., after iterating between optimising for $\boldsymbol{\mu}_\theta$ and optimising for $\boldsymbol{\mu}_{\mathbf{Z}}$, we guess, yes, but this is hard to see from these equations. We do note, though, that optimising $\bar{\mathcal{F}}(\mathbf{Y})$ with respect to $\boldsymbol{\mu}_\theta$ takes additionally the posterior uncertainty $\boldsymbol{\Sigma}^\theta$ of $\boldsymbol{\theta}$ (which is a function of $\boldsymbol{\mu}_\theta$) into account while this only enters in the computation of the posterior mode $\boldsymbol{\mu}_{\mathbf{Z}}$ in the variational approach. Potential advantages of this, e.g., in terms of better convergence properties, are unclear to us at this point.

The computational advantage of optimising $\bar{\mathrm{V}}(\mathbf{Y},\boldsymbol{\theta})$ instead of $\bar{\mathcal{F}}(\mathbf{Y})$ is greatest for the second order gradients where the complexity for computing all elements of $[\cdot]_{ij} = \mathrm{Tr}\left(\mathrm{U}_{\mathbf{z}(t)\mathbf{z}(t)\theta_i\theta_j}(\mathbf{y}(t),\boldsymbol{\mu}_{z(t)},\boldsymbol{\theta})\boldsymbol{\Sigma}^{z(t)}\right)$ is $O(n_\theta^2 n_z^2)$. Nevertheless, 4th order gradients of $\bar{\mathrm{U}}$ still need to be computed. We could radically reduce the computational costs further, if we ignore the mean-field terms of the variational approximation. Then, the gradients simply become

$$\bar{\mathrm{V}}_{\theta_i}(\mathbf{Y},\boldsymbol{\theta}) = \bar{\mathrm{U}}_{\theta_i}(\mathbf{Y},\boldsymbol{\mu}_{\mathbf{Z}},\boldsymbol{\theta}) \tag{139}$$

$$\bar{\mathrm{V}}_{\theta_i\theta_j}(\mathbf{Y},\boldsymbol{\theta}) = \bar{\mathrm{U}}_{\theta_i\theta_j}(\mathbf{Y},\boldsymbol{\mu}_{\mathbf{Z}},\boldsymbol{\theta}) \tag{140}$$

and we would have reduced our variational problem to a simple maximum a posteriori optimisation to find the mode of our approximated posterior density $q(\boldsymbol{\theta})$. In this case, the only computational complexity left would be the cost of computing the first and second order gradients of $\bar{\mathrm{U}}$. This is actually what would usually be understood as the Laplace approximation: find the MAP estimate for $\boldsymbol{\theta}$ and then simply put a Gaussian around it based on the local curvature of $\bar{\mathrm{U}}$ (cf. eq. (122)). Of course, when using this approach, we would ignore the uncertainty of the states and the result would not be a solution which optimises the approximated free action $\bar{\mathcal{F}}(\mathbf{Y})$ anymore. On the other hand, this may still be a viable approach when the uncertainties are low such that the mean-field terms contribute only little to the optimisation.

In conclusion, we have presented three approaches to finding an approximated posterior mode of parameters $\boldsymbol{\mu}_\theta$ which, for the benefit of decreasing computational costs, take decreasing amounts of information about posterior uncertainty of states and parameters into account.

8 Following the Posterior Mode of the States

We now come to the core of Friston's approach to filtering. We believe that his derivation based on an ensemble of particles is confusing and unnecessary. We will ignore it and take a more direct approach. The main idea is the following:

View finding the mode of a distribution as gradient ascent in a dynamical system, extend this to the case where the mode underlies a constant drift and show that a representation in generalised coordinates can estimate that drift.

Before, we have seen that the variational posterior is a function of the variational energy, i.e., holding $q(\boldsymbol{\theta})$ fixed, the approximated posterior for the states at time t is (TODO: this needs to be shown for $\mathbf{z}(t)$ cf. $\mathbf{Z}$)

$$q(\mathbf{z}(t)) = \frac{1}{Z^{z(t)}} \exp\left(\mathrm{V}(\mathbf{Y}, \mathbf{z}(t))\right). \tag{141}$$

Let us now replace the time-varying $\mathbf{z}(t)$ with a variable which is constant across time: $\mathbf{z}(t) = \mathbf{x}$

$$q(\mathbf{x}) = \frac{1}{Z^{x}} \exp\left(\mathrm{V}(\mathbf{Y}, \mathbf{x})\right). \tag{142}$$

Hence, a mode of $q(\mathbf{x})$ is also a maximum of $\mathrm{V}(\mathbf{Y}, \mathbf{x})$. One way of finding such a maximum is to do gradient ascent on V which may be formulated in continuous time as

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t'} = \mathbf{x}' = \frac{\partial \mathrm{V}(\mathbf{Y}, \mathbf{x})}{\partial \mathbf{x}}. \tag{143}$$

To make this explicit: this dynamical system will follow the gradient of V until it reaches a maximum of V (equal to $\boldsymbol{\mu}_{z(t)}$) at which $\mathbf{x}' = 0$, i.e., maxima of V are stable fixed points of this dynamical system while minima are unstable fixed points. Notice that this dynamical system operates in a different time frame t' than the real time t above. In other words, in the new time frame the posterior mode $\boldsymbol{\mu}_{z(t)}$ is constant. This is obviously wrong. So we introduce a drift variable $\mathbf{x}'^{*}$ which is constant in the frame of t' and is supposed to represent the change of the posterior mode in real time t

$$\mathbf{x}' = \frac{\partial \mathrm{V}(\mathbf{Y}, \mathbf{x})}{\partial \mathbf{x}} + \mathbf{x}'^{*}. \tag{144}$$

The difference between the time frames is only a theoretical construct, because they actually evolve simultaneously. Hence, when we are at a maximum of V with $\mathbf{x} = \boldsymbol{\mu}_{z(t)}$

$$\mathbf{x}' = \mathbf{x}'^{*}, \tag{145}$$

i.e., the change in $\mathbf{x}$ actually tracks the change in the posterior mode $\boldsymbol{\mu}_{z(t)}$, if we can ensure that $\mathbf{x}'^{*} = \dot{\boldsymbol{\mu}}_{z(t)}$. The trick now is to use generalised coordinates to simultaneously approximate $\boldsymbol{\mu}_{z(t)}$ and $\dot{\boldsymbol{\mu}}_{z(t)}$. We do this by setting up two (or more) simultaneous dynamical systems:

$$\mathbf{x}' = \frac{\partial}{\partial \mathbf{x}} \mathrm{V}(\mathbf{Y}, \mathbf{x}, \mathbf{x}', \dots) + \mathbf{x}'^{*} \tag{146}$$

$$\mathbf{x}'' = \frac{\partial}{\partial \mathbf{x}'} \mathrm{V}(\mathbf{Y}, \mathbf{x}, \mathbf{x}', \dots) + \mathbf{x}''^{*} \tag{147}$$

$$\vdots$$

where we define $\mathbf{x}'' = \mathrm{d}\mathbf{x}'/\mathrm{d}t''$ and denote the current state of the dynamical system defined by $\mathbf{x}''$ in eq. (147) as $\mathbf{x}'^{*}$ to differentiate it from $\mathbf{x}'$ in eq. (146) (similarly for higher order generalised coordinates). Now remember that (in our new notation) $\mathbf{x}^{*} = \mathbf{z}(t)$ and correspondingly $\mathbf{x}'^{*} = \dot{\mathbf{z}}(t)$. We also know that the dynamical systems for $\mathbf{x}'$ and $\mathbf{x}''$ converge to $\boldsymbol{\mu}_{z(t)}$ and $\dot{\boldsymbol{\mu}}_{z(t)}$, respectively, when the contribution from the higher order coordinates is ignored. Hence, $\mathbf{x}'^{*} = \dot{\boldsymbol{\mu}}_{z(t)}$ after convergence and will provide the drift of the posterior mode (which may be influenced by the drift of the drift of the posterior mode as estimated by the next higher coordinate, and so on).

It is possible to write these equations more compactly by using generalised coordinate notation:

$$\dot{\tilde{\mathbf{x}}} = \frac{\partial \mathrm{V}(\mathbf{Y}, \tilde{\mathbf{x}})}{\partial \tilde{\mathbf{x}}} + \mathbf{D}\tilde{\mathbf{x}} \tag{148}$$

where $\mathbf{D}$ is the matrix derivative operator which shifts coordinates up as defined in eq. (55). Note that all the dynamical systems now are formulated in a common time frame again. This does not change anything, because we only introduced the different time frames as an aid to understand the working of the combined system. But does the combined dynamical system now actually converge to $\tilde{\boldsymbol{\mu}}_{z(t)}$? Friston provides a linear stability analysis around the posterior mode $\tilde{\boldsymbol{\mu}}_{z(t)}$. This does not add much to what we already know from the previous considerations, but for completeness we repeat it anyway:

Make a first-order Taylor approximation of the variational energy gradient around the mode

$$\begin{aligned} \frac{\partial \mathrm{V}(\mathbf{Y}, \tilde{\mathbf{x}})}{\partial \tilde{\mathbf{x}}} &= \frac{\partial \mathrm{V}(\mathbf{Y}, \tilde{\boldsymbol{\mu}}_{z(t)})}{\partial \tilde{\mathbf{x}}} + (\tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}}_{z(t)})^{\mathsf{T}} \frac{\partial^2 \mathrm{V}(\mathbf{Y}, \tilde{\boldsymbol{\mu}}_{z(t)})}{\partial \tilde{\mathbf{x}} \partial \tilde{\mathbf{x}}} \\ &= \frac{\partial \mathrm{V}(\mathbf{Y}, \tilde{\boldsymbol{\mu}}_{z(t)})}{\partial \tilde{\mathbf{x}}} + \boldsymbol{\varepsilon}^{\mathsf{T}} \frac{\partial^2 \mathrm{V}(\mathbf{Y}, \tilde{\boldsymbol{\mu}}_{z(t)})}{\partial \tilde{\mathbf{x}} \partial \tilde{\mathbf{x}}} \\ &= \boldsymbol{\varepsilon}^{\mathsf{T}} \frac{\partial^2 \mathrm{V}(\mathbf{Y}, \tilde{\boldsymbol{\mu}}_{z(t)})}{\partial \tilde{\mathbf{x}} \partial \tilde{\mathbf{x}}} \end{aligned} \tag{149}$$

where we have used $\frac{\partial \mathrm{V}(\mathbf{Y},\tilde{\mathbf{x}})}{\partial \tilde{\mathbf{x}}} = 0$ for $\tilde{\mathbf{x}} = \tilde{\boldsymbol{\mu}}_{z(t)}$ on the last line. Plug into eq. (148)

$$\begin{aligned} \dot{\tilde{\mathbf{x}}} &= \boldsymbol{\varepsilon}^{\mathsf{T}} \frac{\partial^2 \mathrm{V}(\mathbf{Y}, \tilde{\boldsymbol{\mu}}_{z(t)})}{\partial \tilde{\mathbf{x}} \partial \tilde{\mathbf{x}}} + \mathbf{D}\tilde{\mathbf{x}} \\ &= \boldsymbol{\varepsilon}^{\mathsf{T}} \frac{\partial^2 \mathrm{V}(\mathbf{Y}, \tilde{\boldsymbol{\mu}}_{z(t)})}{\partial \tilde{\mathbf{x}} \partial \tilde{\mathbf{x}}} + \mathbf{D}(\boldsymbol{\varepsilon} + \tilde{\boldsymbol{\mu}}_{z(t)}). \end{aligned} \tag{150}$$

Use $\dot{\tilde{\boldsymbol{\mu}}}_{z(t)} = \mathbf{D}\tilde{\boldsymbol{\mu}}_{z(t)}$

$$\dot{\tilde{\mathbf{x}}} = \boldsymbol{\varepsilon}^{\mathsf{T}} \frac{\partial^2 \mathrm{V}(\mathbf{Y}, \tilde{\boldsymbol{\mu}}_{z(t)})}{\partial \tilde{\mathbf{x}} \partial \tilde{\mathbf{x}}} + \mathbf{D}\boldsymbol{\varepsilon} + \dot{\tilde{\boldsymbol{\mu}}}_{z(t)} \tag{151}$$

$$\dot{\boldsymbol{\varepsilon}} = \boldsymbol{\varepsilon}^{\mathsf{T}} \frac{\partial^2 \mathrm{V}(\mathbf{Y}, \tilde{\boldsymbol{\mu}}_{z(t)})}{\partial \tilde{\mathbf{x}} \partial \tilde{\mathbf{x}}} + \mathbf{D}\boldsymbol{\varepsilon} \tag{152}$$

$$= \left(\left[\frac{\partial^2 \mathrm{V}(\mathbf{Y}, \tilde{\boldsymbol{\mu}}_{z(t)})}{\partial \tilde{\mathbf{x}} \partial \tilde{\mathbf{x}}} \right]^{\mathsf{T}} + \mathbf{D} \right) \boldsymbol{\varepsilon} \tag{153}$$

$$= \left(\mathrm{V}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}^{\mathsf{T}} + \mathbf{D} \right) \boldsymbol{\varepsilon} \tag{154}$$

The resulting dynamical system of the difference between $\tilde{\mathbf{x}}$ and the posterior mode $\boldsymbol{\varepsilon} = \tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}}_{z(t)}$ is a simple linear system which converges to $\mathbf{0}$ exponentially fast as long as $\mathrm{V}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}^{\mathsf{T}} + \mathbf{D}$ is negative definite. We know that $\mathrm{V}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}$ has a negative effect, because we are close to a maximum of $\mathrm{V}(\mathbf{Y}, \tilde{\mathbf{x}})$ and the curvature there is negative. Hence, whether $\tilde{\mathbf{x}}$ converges to $\tilde{\boldsymbol{\mu}}_{z(t)}$ depends on whether $\mathrm{V}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}$ dominates over $\mathbf{D}$. Friston suggests to scale $\mathrm{V}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}$ by a constant to ensure that the contribution of $\mathrm{V}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}$ is sufficiently large, but notes that this was unnecessary in their experience. Overall this analysis provides little more information than our original insight that the dynamical system in eq. (143) converges to $\boldsymbol{\mu}_{z(t)}$.

In summary, the main filtering equation in Friston's DEM is eq. (148). For a static system the solution of the variational approximation ensures that the dynamical system defined by eq. (148) converges to a mode of the variational posterior distribution. In a dynamic setting, where this mode moves itself, the representation in generalised coordinates ensures that the whole movement of the mode is estimated. The differential term $\mathbf{D}\tilde{\mathbf{x}}$ is important, because it ensures that the estimate of the mode in one coordinate has the appropriate effect for tracking the mode in the next lower-order coordinate (e.g. the estimated velocity of the mode is taken into account when tracking the mode itself given an observation).

9 Discretisation

For an actual implementation we have to discretise the time integrals in this equation. One straightforward way of doing this is to simply use the time points at which we observed the data points $\mathbf{y}(t)$, but note that with the transformation into generalised coordinates the discretisation can be done independently from the observations and may, therefore, be made more precise, if necessary. Without loss of generality we from here assume that a discretisation step is the size of a time unit such that the approximated free action becomes

$$\begin{aligned}\bar{\mathcal{F}}(\mathbf{Y}) = \mathrm{U}(\boldsymbol{\mu}_\theta) &+ \sum_{t=0}^{T} \mathrm{U}(\mathbf{y}(t), \boldsymbol{\mu}_{z(t)}, \boldsymbol{\mu}_\theta) \\ &+ \frac{1}{2}\left(\log\det(\boldsymbol{\Sigma}^\theta) + (Tn_z + n_\theta)\log 2\pi + \sum_{t=0}^{T}\log\det(\boldsymbol{\Sigma}^{z(t)})\right). \qquad (155)\end{aligned}$$

References

[1] K. Friston, N. Trujillo-Barreto, and J. Daunizeau, "DEM: A variational treatment of dynamic systems," *NeuroImage*, vol. 41, no. 3, pp. 849 – 885, 2008.

[2] K. Friston, "Hierarchical models in the brain.," *PLoS Comput Biol*, vol. 4, p. e1000211, Nov 2008.

[3] M. J. Beal, *Variational algorithms for approximate Bayesian inference*. PhD thesis, University of London, 2003.

[4] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Adaptive Computation and Machine Learning, MIT Press, 2012.

[5] K. B. Petersen and M. S. Pedersen, "The matrix cookbook," tech. rep., Technical University of Denmark, oct 2008. Version 20081110.

[6] A. Doucet and A. M. Johansen, “A tutorial on particle filtering and smoothing: Fifteen years later,” in *Oxford Handbook of Nonlinear Filtering*, Oxford University Press, 2011.