

Chapter 32

Dynamic Causal Modeling for fMRI

32.1 Theoretical background

Dynamic Causal Modelling (DCM) is a method for making inferences about neural processes that underlie measured time series, e.g. fMRI data. The general idea is to estimate the parameters of a reasonably realistic neuronal system model such that the predicted blood oxygen level dependent (BOLD) signal, which results from converting the modeled neural dynamics into hemodynamic responses, corresponds as closely as possible to the observed BOLD time series. This section gives a short introduction to the theoretical background of DCM for fMRI; details can be found in [29]. Note that DCMs can be formulated, in principle, for any measurement technique. Depending on the spatio-temporal properties of a given measurement technique, one needs to define an adequate state equation and an observation model. See Fig 32.1 for a summary of the differences between DCM implementations for fMRI and Event-Related Potentials (ERPs).

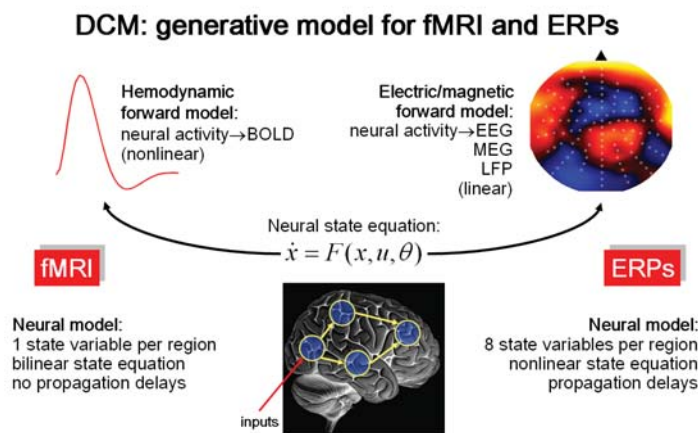


Figure 32.1: A schematic overview of the differences between the DCM implementations for fMRI and ERPs (as measured by EEG or MEG). Whereas the state equation of DCM for fMRI is bilinear and uses only a single state variable per region, that for ERPs is more complex and requires 8 state variables per region. Moreover, DCM for ERPs models the delays of activity propagation between areas. At the level of the observation model, DCM for fMRI is more complex than DCM for ERPs. While the former uses a non-linear model of the hemodynamic response that contains a cascade of differential equations with five state variables per region, the latter uses a simple linear model for predicting observed scalp data.

As in state-space models, two distinct levels constitute a DCM (see Figure 32.2). The hidden level, which cannot be directly observed using fMRI, represents a simple model of neural dynamics in a system of k coupled brain regions. Each system element i is represented by a single state variable z_i , and the dynamics of the system is described by the change of the neural state vector over time.

The neural state variables do not correspond directly to any common neurophysiological measurement (such as spiking rates or local field potentials) but represent a summary index of neural population dynamics in the respective regions. Importantly, DCM models how the neural dynamics are driven by external perturbations that result from experimentally controlled manipulations. These perturbations are described by means of external inputs u that enter the model in two different ways: they can elicit responses through direct influences on specific regions (“driving” inputs, e.g. evoked responses in early sensory areas) or they can change the strength of coupling among regions (“modulatory” inputs, e.g. during learning or attention).

Overall, DCM models the temporal evolution of the neural state vector, i.e. z , as a function of the current state, the inputs u and some parameters that define the functional architecture and interactions among brain regions at a neuronal level (n denotes “neural”):

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dots \\ \dot{z}_k \end{bmatrix} = \dot{z} = \frac{dz}{dt} = F(z, u, \theta^n) \quad (32.1)$$

In this neural state equation, the state z and the inputs u are time-dependent whereas the parameters are time-invariant. In DCM, F has the bilinear form

$$\dot{z} = Az + \sum_{j=1}^m u_j B_j z + Cu \quad (32.2)$$

The parameters of this bilinear neural state equation, $\theta^n = \{A, B_1, \dots, B_m, C\}$, can be expressed as partial derivatives of F :

$$\begin{aligned} A &= \frac{\partial F}{\partial z} = \frac{\partial \dot{z}}{\partial z} \\ B_j &= \frac{\partial^2 F}{\partial z \partial u_j} = \frac{\partial}{\partial u_j} \frac{\partial \dot{z}}{\partial z} \\ C &= \frac{\partial F}{\partial u} \end{aligned} \quad (32.3)$$

These parameter matrices describe the nature of the three causal components which underlie the modeled neural dynamics: (i) context-independent effective connectivity among brain regions, mediated by anatomical connections ($k \times k$ matrix A), (ii) context-dependent changes in effective connectivity induced by the j th input u_j ($k \times k$ matrices B_1, \dots, B_m), and (iii) direct inputs into the system that drive regional activity ($k \times m$ matrix C). As will be demonstrated below, the posterior distributions of these parameters can inform us about the impact that different mechanisms have on determining the dynamics of the model. Notably, the distinction between “driving” and “modulatory” is neurobiologically relevant: driving inputs exert their effects through direct synaptic responses in the target area, whereas modulatory inputs change synaptic responses in the target area in response to inputs from another area. This distinction represents an analogy, at the level of large neural populations, to the concept of driving and modulatory afferents in studies of single neurons.

DCM combines this model of neural dynamics with a biophysically plausible and experimentally validated hemodynamic model that describes the transformation of neuronal activity into a BOLD response. This so-called “Balloon model” was initially formulated by Buxton and colleagues and later extended by [31]. Briefly summarized, it consists of a set of differential equations that describe the relations between four hemodynamic state variables, using five parameters (θ^h). More specifically, changes in neural activity elicit a vasodilatory signal that leads to increases in blood flow and subsequently to changes in blood volume v and deoxyhemoglobin content q . The predicted BOLD signal y is a non-linear function of blood volume and deoxyhemoglobin content.

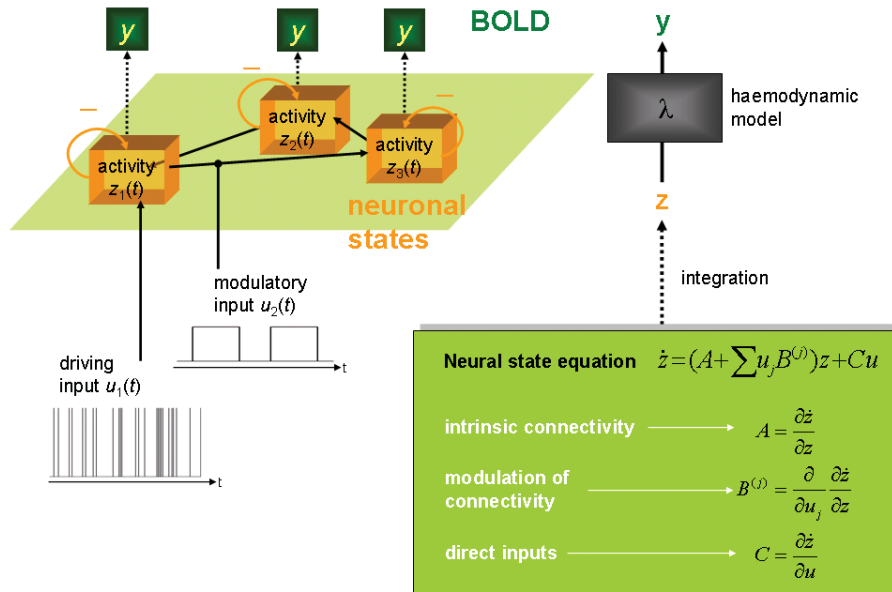


Figure 32.2: Schematic summary of the conceptual basis of DCM. The dynamics in a system of interacting neuronal populations (orange boxes), which are not directly observable by fMRI, is modeled using a bilinear state equation (grey box). Integrating the state equation gives predicted neural dynamics (z) that enter a model of the hemodynamic response (λ) to give predicted BOLD responses (y) (green boxes). The parameters at both neural and hemodynamic levels are adjusted such that the differences between predicted and measured BOLD series are minimized. Critically, the neural dynamics are determined by experimental manipulations. These enter the model in the form of “external” or “driving” inputs. Driving inputs (u_1 ; e.g. sensory stimuli) elicit local responses directly that are propagated through the system according to the intrinsic connections. The strengths of these connections can be changed by modulatory inputs (u_2 ; e.g. changes in cognitive set, attention, or learning).

Details of the hemodynamic model can be found in other publications [31]. By combining the neural and hemodynamic states into a joint state vector x and the neural and hemodynamic parameters into a joint parameter vector $\theta = [\theta^n, \theta^h]^T$, we obtain the full forward model that is defined by the neural and hemodynamic state equations

$$\begin{aligned} \dot{x} &= F(x, u, \theta) \\ y &= \lambda(x) \end{aligned} \quad (32.4)$$

For any given set of parameters θ and inputs u , the joint state equation can be integrated and passed through the output nonlinearity λ to give a predicted BOLD response $h(u, \theta)$. This can be extended to an observation model that includes observation error e and confounding effects X (e.g. scanner-related low-frequency drifts):

$$y = h(u, \theta) + X\beta + e \quad (32.5)$$

This formulation is the basis for estimating the neural and hemodynamic parameters from the measured BOLD data, using a fully Bayesian approach with empirical priors for the hemodynamic parameters and conservative shrinkage priors for the neural coupling parameters.

Details of the parameter estimation scheme, which rests on a Fisher scoring gradient ascent scheme with Levenburg-Marquardt regularisation, embedded in an expectation maximization (EM) algorithm, can be found in the original DCM publication (Friston et al. 2003). In brief, under Gaussian assumptions about the posterior distributions, this scheme returns the posterior

expectations $\eta_{\theta|y}$ and posterior covariance $C_{\theta|y}$ for the parameters as well as hyperparameters for the covariance of the observation noise, C_e .

After fitting the model to measured BOLD data, the posterior distributions of the parameters can be used to test hypotheses about the size and nature of effects at the neural level. Although inferences could be made about any of the parameters in the model, hypothesis testing usually concerns context-dependent changes in coupling (i.e. specific parameters from the B matrices; see Fig. 32.7). As will be demonstrated below, at the single-subject level, these inferences concern the question of how certain one can be that a particular parameter or, more generally, a contrast of parameters, $c^T \eta_{\theta|y}$, exceeds a particular threshold γ (e.g. zero).

Under the assumptions of the Laplace approximation, this is easy to test (Φ_N denotes the cumulative normal distribution):

$$p(c^T \eta_{\theta|y} > \gamma) = \Phi_N \left(\frac{c^T \eta_{\theta|y} - \gamma}{c^T C_{\theta|y} c} \right) \quad (32.6)$$

For example, for the special case $c^T \eta_{\theta|y} = \gamma$ the probability is $p(c^T \eta_{\theta|y} > \gamma) = 0.5$, i.e. it is equally likely that the parameter is smaller or larger than the chosen threshold γ . We conclude this section on the theoretical foundations of DCM by noting that the parameters can be understood as rate constants (units: $1/s = Hz$) of neural population responses that have an exponential nature. This is easily understood if one considers that the solution to a linear ordinary differential equation of the form $\dot{z} = Az$ is an exponential function (see Fig. 32.3).

Integration of a first-order linear differential equation gives an exponential function:

$$\frac{dz}{dt} = az \quad \longrightarrow \quad z(t) = z_0 \exp(at)$$

Coupling parameter a is inversely proportional to the half life τ of $z(t)$:

$$z(\tau) = 0.5z_0 \\ = z_0 \exp(a\tau)$$

$$\longrightarrow \quad a = \ln 2 / \tau$$

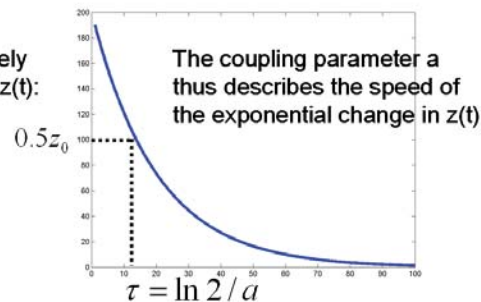


Figure 32.3: A short mathematical demonstration, using a simple linear first-order differential equation as an example, explaining why the coupling parameters in a DCM are inversely proportional to the half-life of the modelled neural responses and are therefore in units of $1/s = \text{Hertz}$.

32.2 Bayesian model selection

A generic problem encountered by any kind of modeling approach is the question of model selection: given some observed data, which of several alternative models is the optimal one? This problem is not trivial because the decision cannot be made solely by comparing the relative fit of the competing models. One also needs to take into account the relative complexity of the models as expressed, for example, by the number of free parameters in each model.

Model complexity is important to consider because there is a trade-off between model fit and generalizability (i.e. how well the model explains different data sets that were all generated from the same underlying process). As the number of free parameters is increased, model fit increases monotonically whereas beyond a certain point model generalizability decreases. The reason for

this is “overfitting”: an increasingly complex model will, at some point, start to fit noise that is specific to one data set and thus become less generalizable across multiple realizations of the same underlying generative process.

Therefore, the question “What is the optimal model?” can be reformulated more precisely as “What is the model that represents the best balance between fit and complexity?”. In a Bayesian context, the latter question can be addressed by comparing the evidence, $p(y|m)$, of different models. According to Bayes theorem

$$p(\theta|y, m) = \frac{p(y|\theta, m)p(\theta|m)}{p(y|m)} \quad (32.7)$$

the model evidence can be considered as a normalization constant for the product of the likelihood of the data and the prior probability of the parameters, therefore

$$p(y|m) = \int p(\theta|y, m)p(\theta|m)d\theta \quad (32.8)$$

Here, the number of free parameters (as well as the functional form) are considered by the integration. Unfortunately, this integral cannot usually be solved analytically, therefore an approximation to the model evidence is needed. One such approximation used by DCM, and many other models in SPM, is to make use of the Laplace approximation¹.

As shown in [70], this yields the following expression for the natural logarithm (\ln) of the model evidence ($\eta_{\theta|y}$ denotes the posterior mean, $C_{\theta|y}$ is the posterior covariance of the parameters, C_e is the error covariance, θ_p is the prior mean of the parameters, and C_p is the prior covariance):

$$\begin{aligned} \ln p(y|m) &= \text{accuracy}(m) - \text{complexity}(m) \\ &= \left[-\frac{1}{2} \ln |C_e| - \frac{1}{2} (y - h(u, \eta_{\theta|y}))^T C_e^{-1} (y - h(u, \eta_{\theta|y})) \right] \\ &\quad - \left[\frac{1}{2} \ln |C_p| - \frac{1}{2} \ln |C_{\theta|y}| + \frac{1}{2} (\eta_{\theta|y} - \theta_p)^T C_p^{-1} (\eta_{\theta|y} - \theta_p) \right] \end{aligned} \quad (32.9)$$

This expression properly reflects the requirement, as discussed above, that the optimal model should represent the best compromise between model fit (accuracy) and model complexity. The complexity term depends on the prior density, for example, the prior covariance of the intrinsic connections.

Two models can then be compared using the Bayes factor:

$$BF_{ij} = \frac{p(y|m_i)}{p(y|m_j)} \quad (32.10)$$

Given uniform priors over models, the posterior probability for model i is greater 0.95 if BF_{ij} is greater than twenty.

This results in a robust procedure for deciding between competing hypotheses represented by different DCMs. These hypotheses can concern any part of the structure of the modeled system, e.g. the pattern of intrinsic connections or which inputs affect the system and where they enter. Note, however, that this comparison is only valid if the data y are identical in all models. This means that in DCM for fMRI, where the data vector results from a concatenation of the time series of all areas in the model, only models can be compared that contain the same areas. Therefore, model selection cannot be used to address whether or not to include a particular area in the model. In contrast, in DCM for ERPs, the data measured at the sensor level are independent of how many neuronal sources are assumed in a given model. Here, model selection could also be used to decide which sources should be included.

32.3 Practical example

The following example refers to the “attention to visual motion” data set available from the SPM web site². This data set was obtained by Christian Buchel and is described in [13]. Note that the

¹This should perhaps more correctly be referred to as a fixed-form variational approximation, where the fixed form is chosen to be a Gaussian. The model evidence is approximated by the negative free energy, F .

²Attention to visual motion dataset: <http://www.fil.ion.ucl.ac.uk/spm/data/attention/>

data available from the web have been pre-processed using SPM99, therefore ensure `flip` is set to 1 in the `spm_flip_analyze_images.m` file before proceeding further.

Making a DCM requires two ingredients: (i) a design matrix and (ii) the time series, stored in VOI files. The regressors of the design matrix define the inputs for the DCM. Note that this means that the design matrix that is optimal for a given DCM is often somewhat different than the one for the corresponding GLM. DCM does not require the design matrix to be part of an estimated model, however. It just needs to be defined.

32.3.1 Defining the GLM

The present experiment consisted of 4 conditions: (i) fixation (F), (ii) static (S, non-moving dots), (iii) no attention (N, moving dots but no attention required), (iv) attention (A). The GLM analyses by Christian showed that activity in area V5 was not only enhanced by moving stimuli, but also by attention to motion. In the following, we will try to model this effect in V5, and explain it as a context-dependent modulation or “enabling” of V5 afferents, using a DCM. First, we need to set up the GLM analysis and extract our time series from the results. In this example, we want to use the same design matrix for GLM and DCM, therefore we recombine the above regressors to get the following three conditions:

1. **photoc**: this comprises all conditions with visual input, i.e. S, N, and A.
2. **motion**: this includes all conditions with moving dots, i.e. N and A.
3. **attention**: this includes the attention-to-motion (A) condition only.

Now we need to define and estimate the GLM. See chapters 8 and 9 on how to do this. Here are the relevant details for this data set that you need to set up the GLM (this information can also be found online³ note this web site describes the analysis for SPM2).

- The onsets for the conditions can be found in the file `factors.mat`. They are named `phot` (photoc), `mot` (motion) and `att` (attention) and are defined in scans (not seconds!). They are blocks of 10 TRs each.
- The TR is 3.22 seconds.
- There are 360 scans.

32.3.2 Extracting time series

Once you have specified and estimated the GLM, you should define t-contrasts that test for photoc, motion, and attention, respectively. These serve to locate areas that show effects due to visual stimulation (e.g. in V1), motion (e.g. V5) and attention (e.g. V5 and superior parietal cortex, SPC). Because V5 shows both motion and attention effects, it is useful to mask the motion-contrast inclusively with the attention-contrast when extracting time series for V5. You should also compute the usual “effects of interest” F-contrast, this is needed for mean-correcting the extracted time series (see below). Here is a step-by-step example for extracting the V5 time series:

1. Press “Results”.
2. Select the `SPM.mat` file.
3. Choose the t-contrast for the motion condition.
4. Mask with other contrasts: Yes
5. Choose the t-contrast for the attention condition.
6. Mask inclusively and choose a threshold of $p \leq 0.05$ uncorrected.

³http://ftp.fil.ion.ucl.ac.uk/spm/data/attention/README_GLM_DCM.txt

7. Select the local maximum of a blob that looks V5-ish, e.g. -45/-81/-9 (by overlaying the activations onto the normalised structural image you should be able to identify V5 more easily).
8. Press the “VOI” button.
9. Name of region: V5
10. Adjust data for: effects of interest (this mean-corrects the time series)
11. VOI definition: sphere
12. VOI radius(mm): e.g. 8 mm

SPM now computes the first principal component of the time series from all voxels included in the sphere. The result is stored (together with the original time series) in a file named `VOI_V5_1.mat` in the working directory. You can now proceed to select time series for V1 (using the “photic” contrast) and SPC (using the “attention” contrast). For this example, we selected -6/-84/-6 for V1 and -18/-57/66 for SPC.

32.3.3 Specifying and estimating the DCM

Now we have defined the inputs (via the design matrix) and the time series, we are ready to build the DCM. We will look at a simplified version of the model described in [29]. In our example here, we will model a hierarchically connected system comprising V1, V5 and SPC, i.e. reciprocal connections between V1-V5 and V5-SPC, but not between V1-SPC. We will assume that (i) V1 is driven by any kind of visual stimulation (direct input “photic”), (ii) motion-related responses in V5 can be explained through an increase in the influence of V1 onto V5 whenever the stimuli are moving (i.e. “motion” acts as modulatory input onto the $V1 \rightarrow V5$ connection) and (iii) attention enhances the influence of SPC onto V5 (i.e. “attention” acts as modulatory input onto the $SPC \rightarrow V5$ connection). This DCM is shown schematically in Figure 32.4, and can be made as follows:

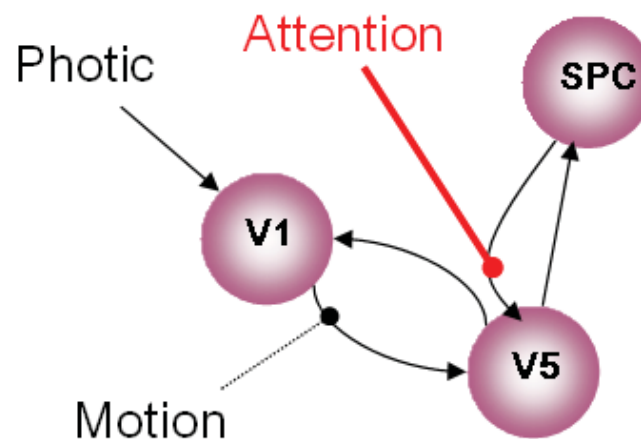


Figure 32.4: *DCM with attentional modulation of backwards connection. Dotted lines denote modulatory connections.*

1. Press the DCM button.
2. Choose “specify”.

3. Select the `SPM.mat` file you just created when specifying the GLM.
4. Name for `DCM_???.mat`: e.g. `mod_bwd` (for “attentional modulation of backward connection”)
5. Select all VOIs in order `VOI_V1_1`, `VOI_V5_1`, `VOI_SPC_1`
6. Include Photic: Yes
7. Include Motion: Yes
8. Include Attention: Yes
9. Define the following intrinsic connections: V1 to V5, V5 to V1, V5 to SPC, SPC to V5, i.e. a hierarchy with reciprocal connections between neighbouring areas. Note that the columns specify the source of the connection and the rows specify its target. Your connectivity matrix should look like the one in Fig. 32.5.
10. Specify Photic as a driving input into V1. See Fig. 32.6
11. Specify Motion to modulate the connection from V1 to V5. See Fig. 32.7
12. Specify Attention to modulate the connection from SPC to V5. See Fig. 32.8
13. Say No for “Nonlinear DCM?”
14. Specify slice timings for each area. This is a new option described in [54]. The default values are set to the last slice of the data, which was the default in the original DCM version. For sequential (as opposed to interleaved) data, this modelling option allows to use DCM in combination with any TR (slice timing differences). Here, we proceed with the default values.
15. Enter 0.04 for “Echo Time, TE[s]”.

A polite “Thank you” completes the model specification process.

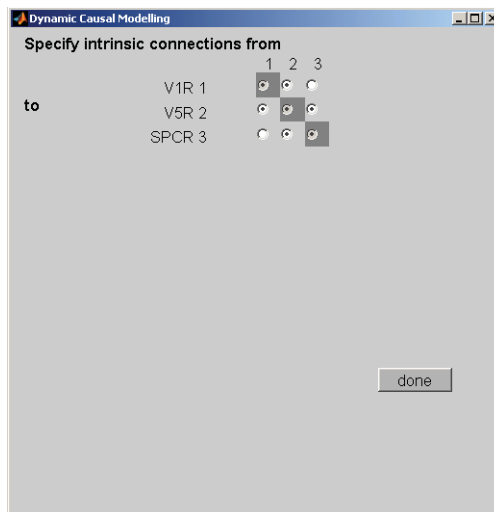


Figure 32.5: Filled circles define the structure of the intrinsic connections C such that eg. there are no connections from V1R to SPCR or from SPCR to V1R. See also Fig 32.4.

You can now estimate the model parameters, either by pressing the DCM button again and choosing “estimate”, or by typing `spm_dcm_estimate('DCM_mod_bwd')` from the MATLAB command line. Once this is completed, you can review the results as follows:

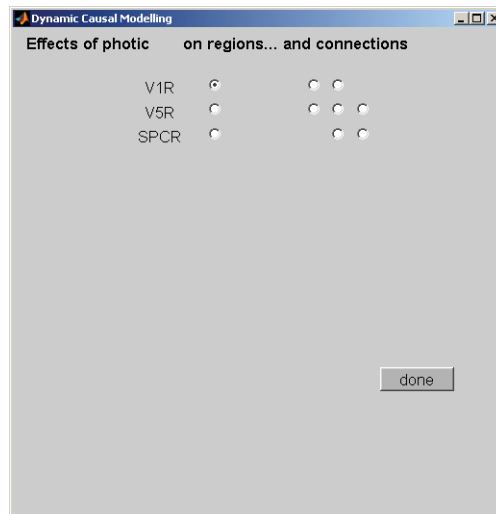


Figure 32.6: The filled circle specifies that the input ‘phot’ connects to region V1R. See also Fig 32.4.

1. Press the DCM button.
2. Choose “review”.
3. Select DCM_mod_bwd
4. Threshold: 0

Now you have multiple options, e.g. you can revisit the fit of the model (“Outputs”) or look at the parameter estimates for the intrinsic connections (“Intrinsic connections”) or for the parameters associated with the driving or modulatory inputs (“Effects of Photic”, “Effects of Motion”, “Effects of Attention”).

Also, you can use the “Contrasts” option to determine how confident you can be that a contrast of certain parameter estimates exceeds the threshold you chose in step 4. Of course, you can also explore the model results at the level of the MATLAB command line by loading the model and inspecting the parameter estimates directly. These can be found in DCM.A (intrinsic connections), DCM.B (modulatory inputs) and DCM.C (driving inputs).

32.3.4 Comparing models

Let us now specify an alternative model and compare it against the one that we defined and estimated above. The change that we are going to make is to assume that attention modulates the $V1 \rightarrow V5$ connection (as opposed to the $SPC \rightarrow V5$ connection in the previous model). For defining this model, you repeat all the steps from the above example, the only differences being that the model gets a new name (e.g. mod_fwd) and that attention now acts on the forward connection. This DCM is shown schematically in Figure 32.9. Once you have estimated this new model, you can perform a Bayesian model comparison as follows:

1. Press the “DCM” button.
2. Choose “compare”.
3. Number of models to compare: 2
4. Select the two models, e.g. in the order DCM_mod_bwd and DCM_mod_fwd.

The Graphics window, Fig. 32.10, now shows a bar plot of the model evidence. You can see that our second model is better than the first one.

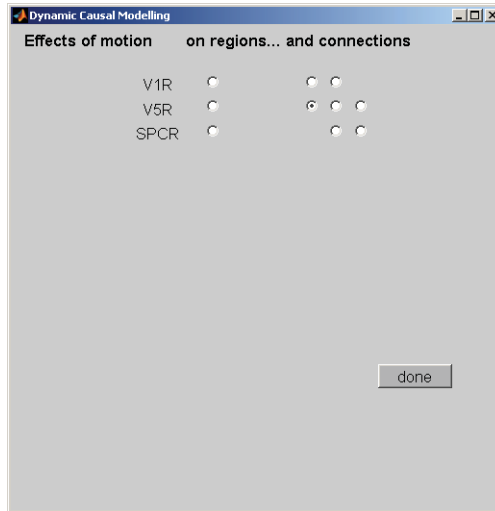


Figure 32.7: The filled circle indicates that the input variable ‘mot’ can modulate the connection from V1R to V5R. This specifies a ‘modulatory’ connection. See also Fig 32.4.

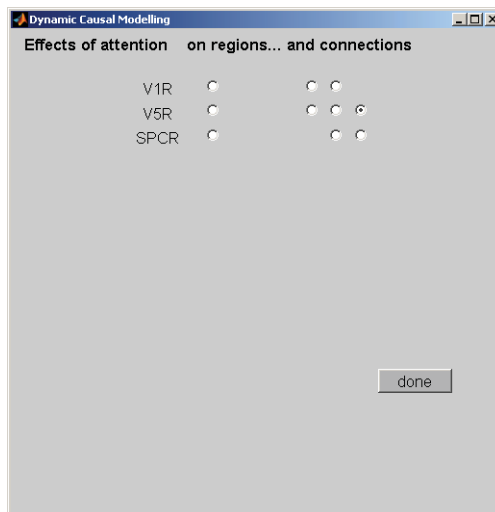


Figure 32.8: The filled circle indicates that attention can modulate the connection from SPCR to V5R. See also Fig 32.4.

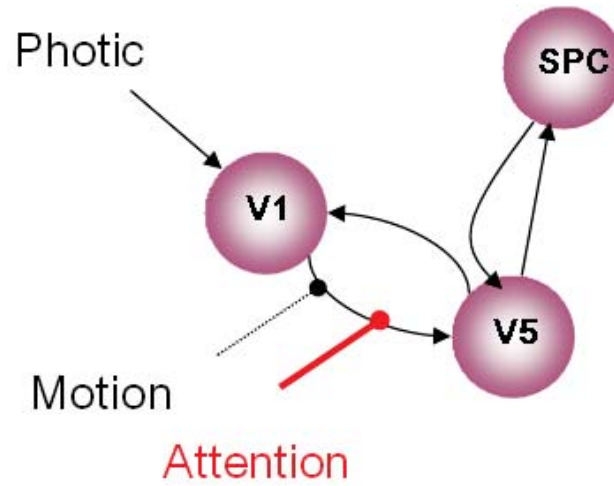


Figure 32.9: DCM with attentional modulation of forwards connection. Dotted lines denote modulatory connections.

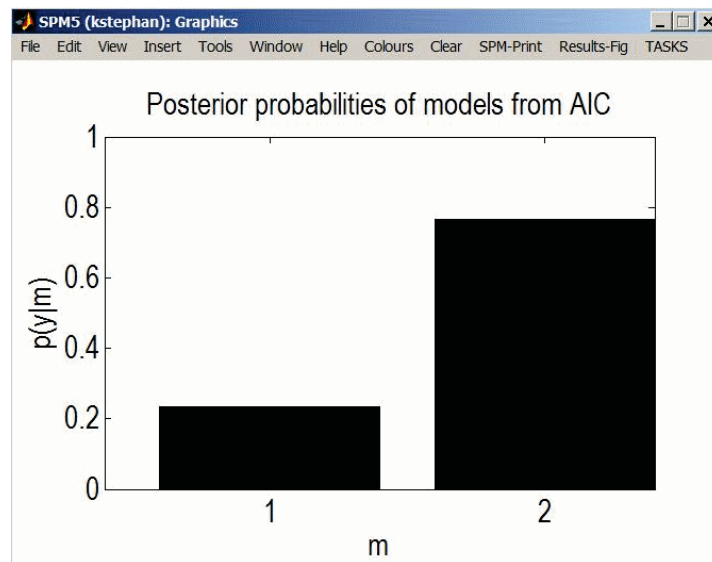


Figure 32.10: Model 2 (shown in Fig 32.9) is preferred to model 1 (shown in Fig 32.4).

