Image-model coupling: a simple information theoretic perspective for image sequences

N.D. Smith[1,*], *C.N. Mitchell*[2], *C.J. Budd*[1]

[1] *Department of Mathematical Sciences, University of Bath, U.K., BA2 7AY*
[2] *Department of Electronic and Electrical Engineering, University of Bath, U.K., BA2 7AY*
[*] *Completed while with the Department of Electronic and Electrical Engineering, University of Bath*

Bath Institute For Complex Systems

Preprint 16/08 (2008)

http://www.bath.ac.uk/math-sci/BICS

Image-model coupling: a simple information theoretic perspective for image sequences

N.D. Smith[1,*], C.N. Mitchell[2], C.J. Budd[1]

[1] Department of Mathematical Sciences, University of Bath, U.K., BA2 7AY
[2] Department of Electronic and Electrical Engineering, University of Bath, U.K., BA2 7AY
[*] Completed while with the Department of Electronic and Electrical Engineering, University of Bath

October 2008
(updated on October 12, 2008)

Abstract: Images are widely used to visualise physical processes. Models may be developed which attempt to replicate those processes and their effects. The technique of coupling model output to images may be used to help understand the underlying physical processes, and better understand the limitations of the models. An information theoretic framework is presented for image-model coupling in the context of communication along a discrete channel. The physical process may be regarded as a transmitter of images and the model as part of a receiver which decodes or recognises those images. Image-model coupling may therefore be interpreted as image recognition. Of interest are physical processes which exhibit 'channel memory'. The response of such a system is not only dependent on the current values of driver variables, but also on the recent history of drivers and/or system state. Examples of such systems in geophysics include the ionosphere and Earth's climate. The discrete channel model is used to help derive expressions for matching images and model output, and analyse the coupling.

1 Introduction

Images describe what is present in a scene, and mathematical or empirical models may be developed which attempt to replicate the underlying processes which produce those images. Given an image, it is interesting to find the model input producing output which best matches the image. In this way useful knowledge concerning the mechanisms producing that image may be discerned. Such image-model coupling, in the context of sequences of images and systems with memory, is described below. The particular example application chosen is the coupling of images of the ionosphere and ionospheric models. This is a challenging problem because the ionosphere [8] is a nonlinear system with regard to the response of its electron content to its driver variables. Furthermore, the system has 'memory' since its response at a given time depends not only on the current drivers but also on the recent history.

There are various questions to consider. What objective function should be used to describe the match between an image sequence and model output? What techniques can be used to search the space of driver variables? More generally, how can competing models be compared in a quantitative manner? And how can the importance of driver variables, for replicating image sequences, be assessed?

To attempt to answer these questions in a consistent manner, image-model coupling is presented within a simple information theoretic context as transmission along a discrete channel. The simpler and more familiar discrete memoryless channel is a special case suitable for systems which do not exhibit channel memory. This approach clearly separates the true underlying real-world process producing the image sequence and the proposed model, and gives a framework to help identify assumptions in the proposed model and possible sources of coupling error. The objective function used for coupling is statistical in nature; with an increased availability of data, it should be possible to derive objective functions which improve the coupling. The context encourages the use of 'tools' drawn from information theory and statistical modelling. The paper is organised as follows. First, Section 2 introduces discrete channel models with and without channel memory; the true underlying process producing the image sequence is considered as a transmitter, and the proposed model is part of a receiver which interprets, or decodes, the image sequence. Section 3 then describes the receiver in more detail; the codebook, noise and state transition models, objective function for matching, and the search mechanism. In particular, the assumptions implicit in applying simple sum square error minimisation are detailed. Section 4 and Section 5 respectively introduce methods to compare alternative models, and assess the sensitivity of image sequences to different driver variables. Finally some discussion and conclusions follow in Section 6 and Section 7.

2 Communication channel framework

Image-model coupling may be viewed from an information theoretic perspective as communication via a discrete channel (e.g. see [11],[1], [9]). The true-world process generating the images is viewed as a transmitter, and the model which is used to interpret the images is part of the receiver. Some form of synchronous 'online' decoding is attractive for continuous communication, i.e. continuous online image-model coupling. In the context of ionospheric modelling, the true ionosphere acts as a transmitter and encodes physical driver variables as an image of the ionosphere, where the image is simply an often incomplete description of the state. The image may be in the form of electron densities or their line integrals. The ionospheric model is part of a decoder which attempts to recover the values of those driver variables. This channel approach allows us to derive an objective function for matching a temporal sequence of images with model output. The following analysis concerns systems such as the ionosphere which are not memoryless, so receivers which assume simple discrete memoryless channel models may not be accurate. However, for reasons of tractability, such receivers may be applied, though it is useful to understand the limitations and assumptions in so doing. In the following, 'TX' denotes the transmitter and 'RXq' the receiver for $q \in \{3, 2, 1\}$.

2.1 Transmitter (TX)

Figure 1 describes the true real-world process generating the images. The process is assumed driven. The driver variables of interest are recorded at time t as $\boldsymbol{u}_{\mathrm{TX}}(t) \in U_{\mathrm{TX}}$, where U_{TX} is a discrete, typically open set.[1] For the ionosphere, driver variables of interest may include measurements of solar or geomagnetic activity. In addition, there are latent driver variables $\boldsymbol{u}'_{\mathrm{TX}}(t) \in U'_{\mathrm{TX}}$ which are not measured and are typically not of direct interest to the modeller. Again, U'_{TX} is a discrete set. The current system is fully described by $\boldsymbol{z}_{\mathrm{TX}}(t) \in Z_{\mathrm{TX}}$ and $\boldsymbol{z}'_{\mathrm{TX}}(t) \in Z'_{\mathrm{TX}}$, which respectively describe those variables which form the image, and the complementary set required to complete the full description. For convenience, define $Z_{\mathrm{TX}} \subseteq Z$ and $Z'_{\mathrm{TX}} \subseteq Z'$. The real-world process may be viewed as a codebook which implements the deterministic mapping, for some fixed and known channel memory length $h_c \in \mathbb{N}$ (for channel memory, e.g. see [9]),

$$z_{\mathrm{TX}}(t) : (\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t), \boldsymbol{u}'_{\mathrm{TX}}(t-h_c,t), \boldsymbol{z}_{\mathrm{TX}}(t-h_c), \boldsymbol{z}'_{\mathrm{TX}}(t-h_c)) \quad \mapsto \quad \boldsymbol{z}_{\mathrm{TX}}(t), \tag{1}$$

where t is the timestep index and h_c is expressed in timesteps, and for example,

$$\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t) \quad \equiv \quad (\boldsymbol{u}_{\mathrm{TX}}(t-h_c), \boldsymbol{u}_{\mathrm{TX}}(t-h_c+1), \ldots, \boldsymbol{u}_{\mathrm{TX}}(t)).$$

[1] The constraint of discrete signals and sets rather than continuous analogues is necessary for a discrete channel model; although real-world processes are typically continuous, discretisation may be regarded as the result of sampling continuous signals or spaces into the machine precision of the recording, storage or computing device.

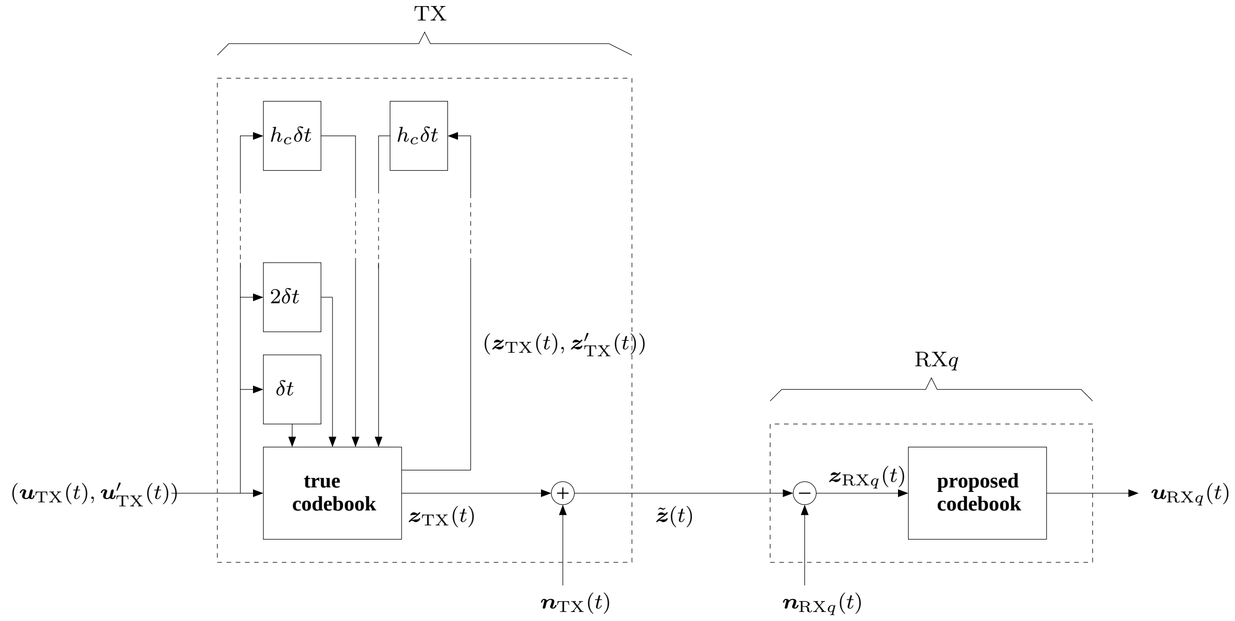
TX
$h_c\delta t$
$h_c\delta t$
$2\delta t$
δt
$(\boldsymbol{z}_{\mathrm{TX}}(t), \boldsymbol{z}'_{\mathrm{TX}}(t))$
RXq
$(\boldsymbol{u}_{\mathrm{TX}}(t), \boldsymbol{u}'_{\mathrm{TX}}(t))$
true codebook
$\boldsymbol{z}_{\mathrm{TX}}(t)$
$+$
$\tilde{\boldsymbol{z}}(t)$
$-$
$\boldsymbol{z}_{\mathrm{RX}q}(t)$
proposed codebook
$\boldsymbol{u}_{\mathrm{RX}q}(t)$
$\boldsymbol{n}_{\mathrm{TX}}(t)$
$\boldsymbol{n}_{\mathrm{RX}q}(t)$

Figure 1: Image-model coupling as a discrete channel model, where the true real-world process is an encoder, δt is the duration of a timestep when used to describe delay in a temporal buffer, and RX$q, q \in \{3, 2, 1\}$ denotes different receivers.

Similar abbreviations are used elsewhere for time-ordered temporal sequences of vectors. The mapping is assumed injective, so $\boldsymbol{z}_{\mathrm{TX}}(t)$ is fully and uniquely determined by the present and past driver variables, both known and latent, and an initial complete ionospheric description. Hence there are no variables beyond those in the domain of the mapping which cause stochastic variation in $\boldsymbol{z}_{\mathrm{TX}}(t)$. In the case of the ionosphere, the dependency on initialisation is required since the ionosphere is not memoryless; its state may evolve differently over time under the action of the same sequence of drivers depending on the initial distribution of plasma. The injective assumption is thought reasonable due to the inclusion of all driver variables and description variables in the domain.

There is also memory in the source. Letting $h_s \in \mathbb{N}_0$ denote[2] the length of source memory in timesteps, then it is convenient to define the state,

$$\boldsymbol{x}_{\mathrm{TX}}(t) \quad = \quad (\boldsymbol{u}_{\mathrm{TX}}(t-h, t), \boldsymbol{u}'_{\mathrm{TX}}(t-h, t), \boldsymbol{z}_{\mathrm{TX}}(t-h), \boldsymbol{z}'_{\mathrm{TX}}(t-h)), \tag{2}$$

where $h = \max[h_s - 1, h_c]$. The redundancy in this state allows the transmitter to be modelled as a hidden Markov process (see Section 3.4).

The actual image measured or recorded at time t is $\tilde{\boldsymbol{z}}(t) \in Z$. This image is related to the partial description $\boldsymbol{z}_{\mathrm{TX}}(t)$ by,

$$\tilde{\boldsymbol{z}}(t) \quad = \quad \boldsymbol{z}_{\mathrm{TX}}(t) + \boldsymbol{n}_{\mathrm{TX}}(t), \tag{3}$$

where $\boldsymbol{n}_{\mathrm{TX}}(t) \in Z$ is additive noise describing error in the measuring devices. Unfortunately, if an image is incomplete, it is sometimes necessary to complete the image using data assimilation or tomographic reconstruction. For the purposes of this analysis, such images are regarded as if directly imaged by a device, and the error in the reconstruction included into the noise process $\boldsymbol{n}_{\mathrm{TX}}(t)$. Both the true real-world process and noise source may be nonstationary; however for the identification and estimation of the statistical models described later, properties of stationarity and ergodicity [10] are convenient. The noise signal $\boldsymbol{n}_{\mathrm{TX}}(t)$ is not transmitted independently along the channel, only the image $\tilde{\boldsymbol{z}}(t)$. In practice, it is usual to consider a finite length sequence of images, for example T images $\tilde{\boldsymbol{z}}(1, T)$.

[2] The notation $\mathbb{N}_0$ denotes all positive integers and the zero.

2.2 Level 3 receiver (RX3)

With infinite knowledge, it is possible to construct a receiver which implements the reverse process to the transmitter. Hence the receiver first 'denoises' the image,

$$\boldsymbol{z}_{\mathrm{RX3}}(t) \quad = \quad \tilde{\boldsymbol{z}}(t) - \boldsymbol{n}_{\mathrm{RX3}}(t), \tag{4}$$

where $\boldsymbol{n}_{\mathrm{RX3}}(t) \in Z$, $\boldsymbol{z}_{\mathrm{RX3}}(t) \in Z_{\mathrm{RX3}}$, and Z_{RX3} is typically an open set denoting the range set of the receiver codebook. The codebook implements deterministic mappings of form,

$$z_{\mathrm{RX3}}(t) : (\boldsymbol{u}_{\mathrm{RX3}}(t-h_c, t), \boldsymbol{u}'_{\mathrm{RX3}}(t-h_c, t), \boldsymbol{z}_{\mathrm{RX3}}(t-h_c), \boldsymbol{z}'_{\mathrm{RX3}}(t-h_c)) \quad \mapsto \quad \boldsymbol{z}_{\mathrm{RX3}}(t), \tag{5}$$

and is used to implement the inverse mapping $\boldsymbol{z}_{\mathrm{RX3}}(t) \mapsto \boldsymbol{u}_{\mathrm{RX3}}(t)$. The codebook is typically implemented by a deterministic mathematical or empirical model. The noise source $\boldsymbol{n}_{\mathrm{RX3}}(t)$ should model the measurement noise in the imaging devices. This receiver is unrealisable, but is included since it permits decoding with the lowest possible error rate. Decoding is described more fully in Section 3.4. For clarity, the receiver is here called a level 3 receiver, where the higher the level, the deeper the conditional dependencies in the receiver codebook. For convenience, a receiver state is defined,

$$\boldsymbol{x}_{\mathrm{RX3}}(t) \quad = \quad (\boldsymbol{u}_{\mathrm{RX3}}(t-h, t), \boldsymbol{u}'_{\mathrm{RX3}}(t-h, t), \boldsymbol{z}_{\mathrm{RX3}}(t-h), \boldsymbol{z}'_{\mathrm{RX3}}(t-h)), \tag{6}$$

where $h = \max[h_s - 1, h_c]$.

2.3 Level 2 receiver (RX2)

This is identical to the level 3 receiver except for the codebook mappings and notation. Denoising is,

$$\boldsymbol{z}_{\mathrm{RX2}}(t) \quad = \quad \tilde{\boldsymbol{z}}(t) - \boldsymbol{n}_{\mathrm{RX2}}(t), \tag{7}$$

where $\boldsymbol{n}_{\mathrm{RX2}}(t) \in Z$ and $\boldsymbol{z}_{\mathrm{RX2}}(t) \in Z_{\mathrm{RX2}}$. The codebook then implements deterministic mappings of form,

$$z_{\mathrm{RX2}}(t) : (\boldsymbol{u}_{\mathrm{RX2}}(t-h_c, t), \boldsymbol{z}_{\mathrm{RX2}}(t-h_c, t-1)) \quad \mapsto \quad \boldsymbol{z}_{\mathrm{RX2}}(t), \tag{8}$$

where the unmeasured or unknown driver and description variables in U'_{TX} and Z'_{TX} have been omitted. The stochastic variation in $\tilde{\boldsymbol{z}}(t)$ due to these variables is instead incorporated into a more complicated noise source $\boldsymbol{n}_{\mathrm{RX2}}(t)$. The noise source no longer models the error in measurement devices alone, but also the stochastic variation due to the omitted variables. Again, define a state,

$$\boldsymbol{x}_{\mathrm{RX2}}(t) \quad = \quad (\boldsymbol{u}_{\mathrm{RX2}}(t-h, t), \boldsymbol{z}_{\mathrm{RX2}}(t-h, t-1)), \tag{9}$$

where $h = \max[h_s - 1, h_c]$.

2.4 Level 1 receiver (RX1)

This is identical to the level 3 and 2 receivers except again for the codebook mappings and notation. Denoising is,

$$\boldsymbol{z}_{\mathrm{RX1}}(t) \quad = \quad \tilde{\boldsymbol{z}}(t) - \boldsymbol{n}_{\mathrm{RX1}}(t), \tag{10}$$

where $\boldsymbol{n}_{\mathrm{RX1}}(t) \in Z$ and $\boldsymbol{z}_{\mathrm{RX1}}(t) \in Z_{\mathrm{RX1}}$. The codebook implements deterministic mappings of form,

$$z_{\mathrm{RX1}}(t) : \boldsymbol{u}_{\mathrm{RX1}}(t) \quad \mapsto \quad \boldsymbol{z}_{\mathrm{RX1}}(t). \tag{11}$$

Each codebook entry $\boldsymbol{z}_{\mathrm{RX1}}(t)$ may be regarded as 'typical' for its driver variables, in a similar manner to which the mean of a Gaussian distribution is typical of samples drawn from that Gaussian. The noise source $\boldsymbol{n}_{\mathrm{RX1}}(t)$ should now also describe the stochastic variation due to different histories of driver variables and initialisations. The present drivers form the state so $\boldsymbol{x}_{\mathrm{RX1}}(t) = \boldsymbol{u}_{\mathrm{RX1}}(t)$.

2.5 Perfect image-model coupling

In this analysis, perfect image-model coupling is the transmission of a sequence of driver variables, without loss, via the true real-world process as encoder, the images as the transmission medium, and the proposed model as decoder. Ideally $\boldsymbol{u}_{\mathrm{TX}}(t) = \boldsymbol{u}_{\mathrm{RX}q}(t)$. For the example of the level 3 receiver, perfect coupling presupposes that,

- the codebook mappings are identical, i.e. $z_{\mathrm{RX3}}^{-1}(t) \circ z_{\mathrm{TX}}(t) = I, \forall t$, where I is the identity map,
- the domain of the receiver codebook is identically descriptive, i.e. $U_{\mathrm{RX3}} = U_{\mathrm{TX}}$, and,
- the transmitter noise source is correctly modelled by the receiver noise source, i.e. $\boldsymbol{n}_{\mathrm{RX3}}(t) = \boldsymbol{n}_{\mathrm{TX}}(t), \forall t$.

The first two conditions assume the proposed model is correct, the third that the noise model is correct. Unfortunately, even under these conditions where the distribution of noise $\boldsymbol{n}_{\mathrm{RX3}}(t)$ is correct, the particular sample drawn from that noise distribution remains unknown. For this reason, even in the case of the level 3 receiver, perfect coupling may not be achievable. In many cases, the driver variables $\boldsymbol{u}_{\mathrm{TX}}(t)$ can only be recovered in the sense of maximum a-posteriori (MAP) or other estimates. For either of the level 3, 2 or 1 receivers, perfect coupling would only be possible if the transmitter and receiver were identical, and either noise samples were transmitted independently between the transmitter and receiver along a separate noiseless channel, or the widths of the noise distributions were always strictly less than the distances between neighbouring entries in the transmitter and receiver codebooks. Decoding is described more fully in Section 3.

3 Receiver

The purpose of the receiver is to decode the image sequence $\tilde{\boldsymbol{z}}(1,T))$ as a sequence of driver variables. The receiver at level q implements this by first decoding the most appropriate state sequence which minimises a scalar objective function,

$$\hat{\boldsymbol{x}}_{\mathrm{RX}q}(1,T)(\tilde{\boldsymbol{z}}(1,T)) \quad = \quad \underset{\boldsymbol{x}_{\mathrm{RX}q}(1,T)\in\otimes_{t=1}^{T}X_{\mathrm{RX}q}}{\operatorname{argmin}} \quad f_{\mathrm{RX}q}(\boldsymbol{x}_{\mathrm{RX}q}(1,T), \tilde{\boldsymbol{z}}(1,T)), \tag{12}$$

and then extracting the underlying consistent driver sequence $\hat{\boldsymbol{u}}_{\mathrm{RX}q}(1,T)(\tilde{\boldsymbol{z}}(1,T))$, i.e. implementing some extraction mapping,

$$\hat{\boldsymbol{x}}_{\mathrm{RX}q}(1,T)(\tilde{\boldsymbol{z}}(1,T)) \quad \mapsto \quad \hat{\boldsymbol{u}}_{\mathrm{RX}q}(1,T)(\tilde{\boldsymbol{z}}(1,T)). \tag{13}$$

This section explains how the receiver achieves this. The definition of the objective function requires the specification of (1) a codebook, both the mapping as implemented by a deterministic model and its domain, (2) a noise model, and (3) a state transition model. The decoder attempts to find the best match between each image and a member of the codebook. This requires careful selection of (4) the objective function to measure the 'goodness of match', and (5) a search mechanism, often heuristic, to navigate the codebook and find the member with maximum 'goodness-of-fit'. These components are described in the remainder of this section. It should be noted that the analysis is restricted to 'static models' where the principal driver variables do not vary with time. Otherwise modifications, particularly for the codebook and noise model, would be required.

3.1 Codebook

For the level q receiver, the codebook may be viewed as the set of deterministic mappings,

$$\mathcal{C}_{\mathrm{RX}q} \quad = \quad \{\boldsymbol{x}_{\mathrm{RX}q}(t) \mapsto \boldsymbol{z}_{\mathrm{RX}q}(t), \forall \boldsymbol{x}_{\mathrm{RX}q}(t) \in X_{\mathrm{RX}q}\}. \tag{14}$$

These may be implemented using lookup tables, but more typically by empirical or mathematical models. The codebook domain $X_{\mathrm{RX}q}$ may also be constrained by the choice of model. For example an empirical model may impose lower and upper bounds on its driver variables, which in turn constrain the codebook domain. The degree of quantisation effects the domain, whether it is at machine precision, or some greater level of quantisation. For example, if the decoder implements grid search, then the effective discrete codebook domain is still further reduced. Codebooks may be viewed as 'plug-and-play' modules. In ionospheric modelling, some codebooks may be more suitable for different tasks than others, e.g. some deterministic models are better at describing high latitude processes while others are more suitable for low latitude processes.

3.2 Noise model

Since the codebook is deterministic, stochastic variability must be introduced via a supplementary noise model. For the level q receiver, the noise process may be fully specified via a set of probability mass functions (PMFs),

$$\mathcal{N}_{\mathrm{RX}q} \quad = \quad \{P_{\mathrm{RX}q}(\boldsymbol{n}_{\mathrm{RX}q}(t)|\boldsymbol{x}_{\mathrm{RX}q}(t)), \forall \boldsymbol{n}_{\mathrm{RX}q}(t) \in Z, \forall \boldsymbol{x}_{\mathrm{RX}q}(t) \in X_{\mathrm{RX}q}\}. \tag{15}$$

In effect $\{\mathcal{C}_{\mathrm{RX}q}, \mathcal{N}_{\mathrm{RX}q}\}$ defines a stochastic version of the deterministic codebook. Indeed if the model involved in image-model coupling is stochastic, then there is no need to define $\mathcal{C}_{\mathrm{RX}q}$ and $\mathcal{N}_{\mathrm{RX}q}$ explicitly. Ideally the codebook and noise model should replicate the stochastic variation in the transmitter (i.e. the real-world process) so that $P_{\mathrm{RX}q}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{RX}q}(t)) = P_{\mathrm{TX}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{TX}}(t)), \forall \tilde{\boldsymbol{z}}(t)$ given $\boldsymbol{u}_{\mathrm{RX}q}(t) = \boldsymbol{u}_{\mathrm{TX}}(t)$. However this is a challenging task since considerable complexity is expected in the real-world variation, as described in Appendix A.

3.3 State transition model

Another supplementary model must be supplied governing and regularising the transitions between consecutive states. This may again be fully specified by a set of PMFs. For a level q receiver,

$$\mathcal{A}_{\mathrm{RX}q} \quad = \quad \{P_{\mathrm{RX}q}(\boldsymbol{x}_{\mathrm{RX}q}(t)|\boldsymbol{x}_{\mathrm{RX}q}(t-1)), \forall \boldsymbol{x}_{\mathrm{RX}q}(t) \in X_{\mathrm{RX}q}, \boldsymbol{x}_{\mathrm{RX}q}(t-1) \in X_{\mathrm{RX}q}\}. \tag{16}$$

This state transition model must assign zero mass to those transitions which do not respect a temporally consistent sequence for $\boldsymbol{u}_{\mathrm{RX}q}(1,T)$.

3.4 Objective function

The objective function should be derived from a decision theoretic perspective. For each sample $\tilde{\boldsymbol{z}}(1,T)$, the decision rule should seek to minimise the conditional risk [5],

$$R(\boldsymbol{x}_{\mathrm{RX}q}(1,T)|\tilde{\boldsymbol{z}}(1,T)) \quad = \sum_{\boldsymbol{y}_{\mathrm{RX}q}(1,T) \in \otimes_{i=1}^{T} X_{\mathrm{RX}q}} l(\boldsymbol{x}_{\mathrm{RX}q}(1,T), \boldsymbol{y}_{\mathrm{RX}q}(1,T)) P_{\mathrm{RX}q}(\boldsymbol{y}_{\mathrm{RX}q}(1,T)|\tilde{\boldsymbol{z}}(1,T)), \tag{17}$$

where $\boldsymbol{x}_{\mathrm{RX}q}(t), \boldsymbol{y}_{\mathrm{RX}q}(t) \in X_{\mathrm{RX}q}, \forall t \in [1,T]$. The scalar function $l(\cdot,\cdot)$ is the loss, and the conditional risk is expressed as the average loss over a posterior distribution in the receiver. An intuitive choice of loss function is the squared L2 norm of the difference between the two arguments. However, under such a regression-based loss, the conditional risk is expensive to compute particularly if samples must be drawn from the posterior. A simpler classification-based loss may instead be used [5],

$$l(\boldsymbol{x}_{\mathrm{RX}q}(1,T), \boldsymbol{y}_{\mathrm{RX}q}(1,T)) \quad = \quad \left\{ \begin{array}{lll} 0 & \text{if} & \boldsymbol{x}_{\mathrm{RX}q}(1,T) = \boldsymbol{y}_{\mathrm{RX}q}(1,T) \\ 1 & & \text{otherwise} \end{array} \right. . \tag{18}$$

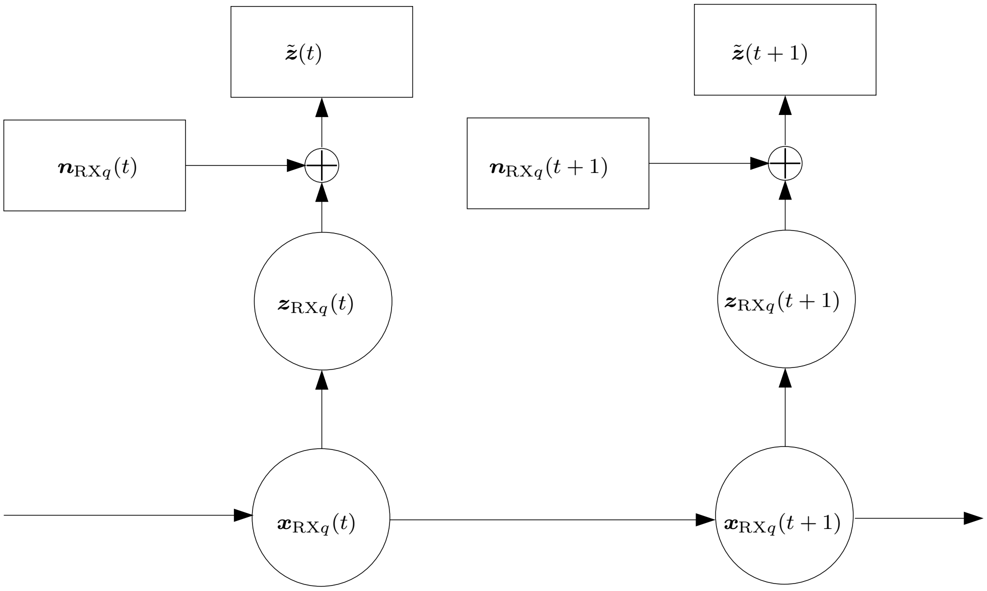
$\tilde{\boldsymbol{z}}(t)$
$\tilde{\boldsymbol{z}}(t+1)$
$\boldsymbol{n}_{\mathrm{RX}q}(t)$
$\boldsymbol{n}_{\mathrm{RX}q}(t+1)$
$\boldsymbol{z}_{\mathrm{RX}q}(t)$
$\boldsymbol{z}_{\mathrm{RX}q}(t+1)$
$\boldsymbol{x}_{\mathrm{RX}q}(t)$
$\boldsymbol{x}_{\mathrm{RX}q}(t+1)$

Figure 2: Temporal portion of hidden Markov model (HMM) for modelling the receiver RXq, $q \in \{3, 2, 1\}$.

This is preferable because it reduces computational cost since then,

$$R(\boldsymbol{x}_{\mathrm{RX}q}(1,T)|\tilde{\boldsymbol{z}}(1,T)) \quad = \quad 1 - P_{\mathrm{RX}q}(\boldsymbol{x}_{\mathrm{RX}q}(1,T)|\tilde{\boldsymbol{z}}(1,T)), \tag{19}$$

thereby avoiding the averaging operation. Hence an objective function may be expressed as follows, where logs are taken and unneccessary terms are discarded from the log posterior,

$$f_{\mathrm{RX}q}(\boldsymbol{x}_{\mathrm{RX}q}(1,T), \tilde{\boldsymbol{z}}(1,T)) \quad = \quad -\ln P_{\mathrm{RX}q}(\tilde{\boldsymbol{z}}(1,T)|\boldsymbol{x}_{\mathrm{RX}q}(1,T)) - \ln P_{\mathrm{RX}q}(\boldsymbol{x}_{\mathrm{RX}q}(1,T)). \tag{20}$$

The first of the two terms in the objective function models memory in the channel, the second memory in the state space (which includes memory in the driver source). Stochastic variation originates both in the channel and the source (e.g. see [9]). The first term is determined by the codebook $\mathcal{C}_{\mathrm{RX}q}$ and noise model $\mathcal{N}_{\mathrm{RX}q}$, the second by the state transition model $\mathcal{A}_{\mathrm{RX}q}$. The resultant decoder is the well-known maximum a-posteriori (MAP) decoder where the posterior acts as a measure of 'goodness-of-fit'. The objective function may be compared with those derived in the variational analysis of other applications such as weather prediction [4]. The remainder of this subsection considers how to obtain expressions for the objective function under different receivers, and details the assumptions implicit in decoding images using least sum squares minimisation.

3.4.1 Discrete channel with channel memory

All receivers RXq, $q \in \{3, 2, 1\}$ may be modelled as hidden Markov processes of the form shown in Figure 2. The noise process and state transition process are assumed stationary in time. The state contains all information about the past which can influence the future. Then the posterior is,

$$P_{\mathrm{RX}q}(\boldsymbol{x}_{\mathrm{RX}q}(1,T)|\tilde{\boldsymbol{z}}(1,T)) \quad = \quad \prod_{t=1}^{T} P_{\mathrm{RX}q}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{x}_{\mathrm{RX}q}(t)) P_{\mathrm{RX}q}(\boldsymbol{x}_{\mathrm{RX}q}(t)|\boldsymbol{x}_{\mathrm{RX}q}(t-1)), \tag{21}$$

where it is assumed that the state $\boldsymbol{x}_{\mathrm{RX}q}(0)$ is fully known, i.e. the history of driver variables and pre-noised images is known back to a timestep index of $-h$, if required. The objective function becomes,

$$f_{\mathrm{RX}q}(\boldsymbol{x}_{\mathrm{RX}q}(1,T), \tilde{\boldsymbol{z}}(1,T)) \quad = \quad -\sum_{t=1}^{T} \{\ln P_{\mathrm{RX}q}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{x}_{\mathrm{RX}q}(t)) + \ln P_{\mathrm{RX}q}(\boldsymbol{x}_{\mathrm{RX}q}(t)|\boldsymbol{x}_{\mathrm{RX}q}(t-1))\}. \tag{22}$$

Both level 3 and level 2 receivers assume channel memory and source memory of duration no greater than h and $h+1$ timesteps respectively. Of course, the length of these memories may effectively be reduced or eliminated by constraining the PMFs in $\mathcal{N}_{\mathrm{RX}q}$ and/or $\mathcal{A}_{\mathrm{RX}q}$. For example, channel memory may be eliminated in favour of source memory alone.

3.4.2 Discrete channel with no channel memory

The level 1 receiver assumes no channel memory, i.e. it assumes a discrete memoryless channel. This is well known and is worthy of further consideration. Since by definition $\boldsymbol{x}_{\mathrm{RX1}}(t) = \boldsymbol{u}_{\mathrm{RX1}}(t), \forall t$,

$$f_{\mathrm{RX1}}(\boldsymbol{x}_{\mathrm{RX1}}(1,T), \tilde{\boldsymbol{z}}(1,T)) \quad = \quad -\sum_{t=1}^{T}\{\ln P_{\mathrm{RX1}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{RX1}}(t)) + \ln P_{\mathrm{RX1}}(\boldsymbol{u}_{\mathrm{RX1}}(t)|\boldsymbol{u}_{\mathrm{RX1}}(t-1))\}. \quad (23)$$

The assumption that the image is conditionally independent of previous variables given the current driver variables is unrealistic for systems with channel memory. For the ionosphere, plasma accumulates with time. Its response, as illustrated by its description, under the action of a given set of drivers, will vary depending on its past driver values. However the discrete memoryless assumption is convenient and, at the risk of reduced modelling accuracy, the likelihood $P_{\mathrm{RX1}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{RX1}}(t))$ may be regarded as summarising all possible histories.

3.4.3 Discrete memoryless channel with spatially stationary Gaussian noise

The objective function for the discrete memoryless channel, as given in Equation 23, may be simplified to yield the well-known sum squares and weighted sum squares objective functions. It is useful to consider the additional constraints.

First, assume there is no source memory, i.e. there is conditional independence between successive values of driver variables. The conditional independence assumption is severe and unrealistic for real-world systems. Driver variables naturally change smoothly, albeit at a certain level of scale. For the ionosphere, on all but the shortest time scales, any measurement of incident solar radiation may appear discontinous during the arrival of a solar flare. However for the majority of time, this measurement varies smoothly between consecutive timesteps, and the state transition model should favour such smooth changes. Without these conditional dependencies, the receiver now models no memory, neither in the channel nor in the source. Also, it is sometimes convenient to assume that the mapping $\boldsymbol{u}_{\mathrm{RX1}}(t) \mapsto \boldsymbol{z}_{\mathrm{RX1}}(t), \forall t \in [1,T]$ is injective. The injective assumption is reasonable, particularly when a vector of few driver variables maps into a large image with many components. Then the objective function in Equation 23 becomes,

$$f'_{\mathrm{RX1}}(\boldsymbol{x}_{\mathrm{RX1}}(1,T), \tilde{\boldsymbol{z}}(1,T)) \quad = \quad -\sum_{t=1}^{T}\{\ln P_{\mathrm{RX1}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{RX1}}(t)) + \ln P_{\mathrm{RX1}}(\boldsymbol{u}_{\mathrm{RX1}}(t))\}. \quad (24)$$

The prior $P_{\mathrm{RX1}}(\boldsymbol{u}_{\mathrm{RX1}}(t))$ should reflect the frequency of occurrence of different driver variables, averaged over all possible previous driver sequences. In the example of the ionosphere, the prior may be calculated using measurements collected over a full solar cycle.

Next, consider that in addition to being temporally stationary, the noise process is invariant to the value of $\boldsymbol{z}_{\mathrm{RX1}}(t)$, i.e. it is spatially stationary. Hence $P_{\mathrm{RX1}}(\boldsymbol{n}_{\mathrm{RX1}}(t)|\boldsymbol{u}_{\mathrm{RX1}}(t)) = P_{\mathrm{RX1}}(\boldsymbol{n}_{\mathrm{RX1}}(t))$ simplifying the noise model $\mathcal{N}_{\mathrm{RX1}}$ considerably to the specification of a single PMF. This is unlikely for systems such as the ionosphere where the response, or state, varies nonlinearly with the drivers. Additionally, assume the noise model is a zero-mean discretised Gaussian so $P_{\mathrm{RX1}}(\boldsymbol{n}_{\mathrm{RX1}}(t)) = N(\boldsymbol{n}_{\mathrm{RX1}}(t); \boldsymbol{0}, \boldsymbol{R})$ where $\boldsymbol{R}$ is the covariance[3], and hence $P_{\mathrm{RX1}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{z}_{\mathrm{RX1}}(t)) = N(\tilde{\boldsymbol{z}}(t); \boldsymbol{z}_{\mathrm{RX1}}(t), \boldsymbol{R})$. Unfortunately, Gaussianity is unlikely for description variables, such as line integrals of electron content in the case of the ionosphere, which are naturally nonnegative and where noise variance is expected to increase with absolute value.

[3] Estimation of the covariance is not included in the variational problem, and must be performed apriori.

Finally, assume there is no prior preference in the driver variables so there is a uniform prior $P_{\mathrm{RX1}}(\boldsymbol{u}_{\mathrm{RX1}}(t))$ over U_{RX1}. Again this is unreasonable for many real-world systems. For the case of the ionosphere, quiet space weather occurs much more frequently than stormy space weather, and the prior over driver variables should reflect this. The objective function for the discrete memoryless channel in Equation 24 may now be simplified to the following, where irrelevant terms have been discarded,

$$f''_{\mathrm{RX1}}(\boldsymbol{x}_{\mathrm{RX1}}(1,T), \tilde{\boldsymbol{z}}(1,T)) \quad = \quad \sum_{t=1}^{T} \boldsymbol{z}_{\mathrm{RX1}}(t)^{\top} \boldsymbol{R}^{-1} (\frac{1}{2} \boldsymbol{z}_{\mathrm{RX1}}(t) - \tilde{\boldsymbol{z}}(t)). \tag{25}$$

The more negative the objective function, the better is the 'goodness-of-fit'. If the covariance $\boldsymbol{R}$ is assumed diagonal, minimisation is equivalent to the conventional least weighted sum squares solution. The diagonal elements, i.e. weights, represent the relative 'importance' of each component in the image. If each component is of equal importance, $\boldsymbol{R}$ may be set to Identity and the minimisation yields the least sum squares solution.

3.5 Search

Minimisation of the objective function over the full domain $\otimes_{t=1}^{T} X_{\mathrm{RX}q}$ of the receiver codebook requires a search mechanism. If the codebook mapping is implemented by some deterministic empirical or mathematical model, any objective function of type $f_{\mathrm{RX}q}(\cdot)$ and its derivative are rarely known analytically. Derivative-free optimisation techniques are then required. The simplest approach is full grid search. However computational cost increases exponentially with the number of driver variables, and the effective size of $\otimes_{t=1}^{T} X_{\mathrm{RX}q}$ must often be reduced. Alternative approaches include numerical approximation of gradients (e.g. see the algorithms in [13]) and sampling-based methods. An example of a sampling scheme is simulated annealing [16] which, although it converges to a global minimum, requires many evaluations of the objective function. A variant of simulated annealing, called fast annealing [16], may be applied since it is relatively easy to implement. To reduce computational cost, instant freezing may also be used, but the technique becomes sensitive to its initialisation and is not a global optimiser.

Fast annealing yields a sequence of samples which ideally converge to a global minimum. Samples are chosen according to a proposal distribution. A new sample is accepted if it yields a lower evaluation of the objective function, or otherwise only with a certain probability. Unlike conventional simulated annealing, the width of the proposal distribution shrinks with successive sampling. The rate of shrinkage must be controlled to avoid premature convergence. The proposal distribution should be heavy-tailed to increase the chance of convergence to other nearby, lower minima; however there is then always the risk of premature convergence in poorer neighbouring minima. For simplicity of notation, let $\boldsymbol{x} \equiv \boldsymbol{x}_{\mathrm{RX}q}(1,T)$, $X \equiv \otimes_{t=1}^{T} X_{\mathrm{RX}q}$ and $f(\cdot) \equiv f_{\mathrm{RX}q}(\cdot, \tilde{\boldsymbol{z}}(1,T))$.

Given an initial sample $\boldsymbol{x}_0$, one possible implementation of fast annealing for an objective funtion f is as follows.

1. Let $k := 0$ and $\boldsymbol{x}_{\mathrm{best}} := \boldsymbol{x}_0$.
2. Sample $\boldsymbol{x}_{k+1} \sim P(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k)$.
3. Evaluate $f(\boldsymbol{x}_{k+1})$.
4. Calculate temperature[4] $T_{k+1} \in \mathbb{R}_0^+$.
5. Let $\Delta f_{k+1} := f(\boldsymbol{x}_{k+1}) - f(\boldsymbol{x}_{\mathrm{best}})$. If $\Delta f_{k+1} < 0$, then accept the sample, otherwise accept it with a probability $\exp(-\Delta f_{k+1}/T_{k+1})$. If the sample is accepted, $\boldsymbol{x}_{\mathrm{best}} := \boldsymbol{x}_{k+1}$.
6. Let $k := k+1$ and repeat from Step 2 until a stopping criterion is fulfilled.

[4] The notation $\mathbb{R}^+$ and $\mathbb{R}_0^+$ denote positive real numbers, respectively without and with the extra inclusion of zero.

Examples of stopping criteria include reaching a maximum limit on the number of iterations, or reaching a maximum limit on the run of rejections (i.e. the number of consecutive failures to update $\boldsymbol{x}_{\text{best}}$). The temperatures are usually determined by a schedule, for example an exponentially decreasing schedule [16]. For instant freezing, $T_k = 0, \forall k > 0$; the algorithm above may be modified to exclude all references to temperature, and a new sample is only accepted if it yields a nonincreasing, or alternatively strictly a lower, value of the objective function.

Driver variables are discretised, but at often different levels, for consistency with the discrete channel model. For ionospheric models, the solar radiation parameter $F_{10.7}$ is continuous in the real world and may be discretised to 2 decimal places, whereas the geomagnetic parameter Ap is an integer. The sampling scheme must also restrict the range of samples between upper and lower bounds. Hence each component of $\boldsymbol{x}_{k+1}$ is sampled in turn according to its own specific probability mass function. Denoting $\boldsymbol{x} = (x^1, \ldots, x^d)$ and $x^i \in [x^i_{\text{min}}, x^i_{\text{max}}], i \in [1, d]$, the sampling scheme for each $\boldsymbol{x}_{k+1}$ in Step 2 above may be implemented as follows.

1. Let $i := 1$.
2. Sample $x^i_{k+1} \sim P_i(x^i_{k+1} - x^i_k; T^i_k)$ where $T^i_k \in \mathbb{R}^+$ controls the width of the mass function.
3. If $x^i_{k+1} > x^i_{\text{max}}$ or $x^i_{k+1} < x^i_{\text{min}}$, reject sample and return to Step 1.
4. If $i < d$, $i := i + 1$ and return to Step 2.

The rejection of the sample from any one component necessitates the rejection of samples from all previous components, otherwise the sampling scheme would be biased. Ignoring end effects due to quantisation, this ensures that the joint mass function within the lower and upper bounds would simply be a scaled version of the equivalent portion of the joint mass function if there were no limits or bounds. Here, the choice of heavy-tailed proposal distribution is a discretised version of the Cauchy distribution. Hence,

$$P_i(x^i_{k+1} - x^i_k; T^i_k) \quad = \quad \int_{Y^i_{k+1}} p_i(y^i_{k+1}; T^i_k) \mathrm{d}y^i_{k+1}, \tag{26}$$

where $y^i_{k+1} \in \mathbb{R}$, $Y^i_{k+1} = \{y^i_{k+1} : x^i_{k+1} = \text{quant}[y^i_{k+1} + x^i_k]\}$, quant[$\cdot$] is the quantisation mapping and,

$$p_i(y^i_{k+1}; T^i_k) \quad = \quad \frac{1}{\pi}\left(\frac{T^i_k}{(T^i_k)^2 + (y^i_{k+1})^2}\right). \tag{27}$$

Here $T^i_k \in \mathbb{R}^+$ controls the width of the proposal distribution and typically decreases with increasing k according to some schedule, for example exponential or geometric. In practice the sampling may be implemented[5] as follows (see 'Cauchy distribution" in [19]),

$$y^i_{k+1} \quad = \quad T^i_k \tan((c - \frac{1}{2})\pi), \tag{28}$$

where $c \in \mathbb{R}$ is sampled uniformly and randomly from its domain $c \in [0, 1]$.

There may be different schedules for different components x^i reflecting differences in dynamic range. To reduce the number of tuning parameters it is possible to define,

$$T^i_k \quad = \quad a_k(x^i_{\text{max}} - x^i_{\text{min}}), \forall i. \tag{29}$$

Only one schedule then needs specifiying, i.e. the schedule for the multiplicative factor a_k. An example is the following exponential schedule,

$$a_k \quad = \quad a_0 \exp\{-bk\}, \tag{30}$$

requiring (a_0, b), where $a_0 \in \mathbb{R}^+$ and $b \in \mathbb{R}^+$. The tuple (a_0, b) specifies the initial width and rate of width shrinkage of the proposal Cauchy distributions. It should be tuned to trade-off the quality of solution and the time and computational resources available.

[5] In practice, some modifications are required to ensure values at lower and upper bounds are assigned the same probability mass as midinterval values.

4 Evaluation

It is sometimes useful to compare different codebooks for image-model coupling. Codebooks should not be compared in isolation, but in the context of the accompanying noise model, state transition model, objective function and search mechanism. The error rate associated with each full receiver is the risk calculated as [5],

$$R(\mathrm{RX}q) = \sum_{\tilde{\boldsymbol{z}}(1,T)\in\otimes_{t=1}^{T}Z} R(\hat{\boldsymbol{x}}_{\mathrm{RX}q}(1,T)|\tilde{\boldsymbol{z}}(1,T))P_{\mathrm{TX}}(\tilde{\boldsymbol{z}}(1,T)), \tag{31}$$

where the conditional risk is as given in Equation 17. If each level q receiver is correct such that $\mathcal{M}_{\mathrm{RX}q} = \{\mathcal{C}_{\mathrm{RX}q}, \mathcal{N}_{\mathrm{RX}q}, \mathcal{A}_{\mathrm{RX}q}\}$ exactly replicates the statistical properties of the transmitter, then $R(\mathrm{RX3}) \leq R(\mathrm{RX2}) \leq R(\mathrm{RX1})$. This relationship does not necessarily hold for incorrect receivers.

However the error rate penalises the decoding of variables, with values which differ from those in the transmitter, with the same loss independent of the degree of difference. This may be misleading. A better 'measure' is the mutual information between the sequence of true driver variables $\boldsymbol{u}_{\mathrm{TX}}(1,T)$ and the sequence of received driver variables $\boldsymbol{u}_{\mathrm{RX}q}(1,T)$. Perfect coupling, or lossless transmission, maximises this measure. Any attempt to improve, learn or adapt models should aim to increase this measure. Using [11], the mutual information may be expressed as,

$$I(\boldsymbol{u}_{\mathrm{RX}q}(1,T);\boldsymbol{u}_{\mathrm{TX}}(1,T)) = H(\boldsymbol{u}_{\mathrm{TX}}(1,T)) - H(\boldsymbol{u}_{\mathrm{TX}}(1,T)|\boldsymbol{u}_{\mathrm{RX}q}(1,T)), \tag{32}$$

where the first and second terms on the right hand side are respectively source and conditional entropies. Since the source entropy is fixed, the following simpler 'measure' may instead be used. Using [11],

$$\begin{aligned} F(\mathcal{M}_{\mathrm{RX}q}) &= -H(\boldsymbol{u}_{\mathrm{TX}}(1,T)|\boldsymbol{u}_{\mathrm{RX}q}(1,T)) \\ &= \sum_{\boldsymbol{u}_{\mathrm{RX}q}(1,T)\in\otimes_{t=1}^{T}U_{\mathrm{RX}q}} P_{\mathrm{RX}q}(\boldsymbol{u}_{\mathrm{RX}q}(1,T)) \sum_{\boldsymbol{u}_{\mathrm{TX}}(1,T)\in\otimes_{t=1}^{T}U_{\mathrm{TX}}} P(\boldsymbol{u}_{\mathrm{TX}}(1,T)|\boldsymbol{u}_{\mathrm{RX}q}(1,T)) \\ &\quad \ln P(\boldsymbol{u}_{\mathrm{TX}}(1,T)|\boldsymbol{u}_{\mathrm{RX}q}(1,T)), \end{aligned} \tag{33}$$

where,

$$-H(\boldsymbol{u}_{\mathrm{TX}}(1,T)) \leq F(\mathcal{M}_{\mathrm{RX}q}) \leq 0. \tag{34}$$

In a decision theoretic context, maximising this function corresponds to minimising the risk subject to the loss function $l(\boldsymbol{u}_{\mathrm{TX}}(1,T), \boldsymbol{u}_{\mathrm{RX}q}(1,T)) = -\ln P(\boldsymbol{u}_{\mathrm{TX}}(1,T)|\boldsymbol{u}_{\mathrm{RX}q}(1,T))$. Although this loss function differs from that under which the MAP decoder is optimal, this mismatch between comparison and application/decoding simplifies the implementation of decoders. Rearranging and introducing the latent variables $\tilde{\boldsymbol{z}}(1,T)$,

$$\begin{aligned} F(\mathcal{M}_{\mathrm{RX}q}) &= \sum_{\boldsymbol{u}_{\mathrm{RX}q}(1,T)\in\otimes_{t=1}^{T}U_{\mathrm{RX}q}} \sum_{\tilde{\boldsymbol{z}}(1,T)\in\otimes_{t=1}^{T}Z} \sum_{\boldsymbol{u}_{\mathrm{TX}}(1,T)\in\otimes_{t=1}^{T}U_{\mathrm{TX}}} P(\boldsymbol{u}_{\mathrm{RX}q}(1,T), \tilde{\boldsymbol{z}}(1,T), \boldsymbol{u}_{\mathrm{TX}}(1,T)) \\ &\quad \ln P(\boldsymbol{u}_{\mathrm{TX}}(1,T)|\boldsymbol{u}_{\mathrm{RX}q}(1,T)). \end{aligned} \tag{35}$$

Making some conditional independence assumptions reasonable for such a communication channel,

$$\begin{aligned} F(\mathcal{M}_{\mathrm{RX}q}) &= \sum_{\boldsymbol{u}_{\mathrm{RX}q}(1,T)\in\otimes_{t=1}^{T}U_{\mathrm{RX}q}} \sum_{\tilde{\boldsymbol{z}}(1,T)\in\otimes_{t=1}^{T}Z} \sum_{\boldsymbol{u}_{\mathrm{TX}}(1,T)\in\otimes_{t=1}^{T}U_{\mathrm{TX}}} P_{\mathrm{RX}q}(\boldsymbol{u}_{\mathrm{RX}q}(1,T)|\tilde{\boldsymbol{z}}(1,T)) \\ &\quad P_{\mathrm{TX}}(\tilde{\boldsymbol{z}}(1,T)|\boldsymbol{u}_{\mathrm{TX}}(1,T))P_{\mathrm{TX}}(\boldsymbol{u}_{\mathrm{TX}}(1,T)) \ln P(\boldsymbol{u}_{\mathrm{TX}}(1,T)|\boldsymbol{u}_{\mathrm{RX}q}(1,T)). \end{aligned} \tag{36}$$

Unfortunately it is difficult to estimate the distributions defined on the transmitter variables. However the quantity may be approximated by assuming the transmitter and receiver codebooks are injective and the noise process $\boldsymbol{n}_{\mathrm{TX}}(t)$ is negligible. Then $\tilde{\boldsymbol{z}}(t) \approx \boldsymbol{z}_{\mathrm{TX}}(t), \forall t \in [1,T]$. Drawing ℓ samples of type $\boldsymbol{u}_{\mathrm{TX}}(1,T)$ according to the prior $P_{\mathrm{TX}}(\boldsymbol{u}_{\mathrm{TX}}(1,T))$ then,

$$F(\mathcal{M}_{\mathrm{RX}q}) \approx \frac{1}{\ell}\sum_{l=1}^{\ell} \sum_{\boldsymbol{u}_{\mathrm{RX}q}(1,T)\in\otimes_{t=1}^{T}U_{\mathrm{RX}q}} P_{\mathrm{RX}q}(\boldsymbol{u}_{\mathrm{RX}q}(1,T)|\tilde{\boldsymbol{z}}_l(1,T)) \ln P_{\mathrm{RX}q}(\tilde{\boldsymbol{z}}_l(1,T)|\boldsymbol{z}_{\mathrm{RX}q}(1,T)), \tag{37}$$

where $\tilde{\boldsymbol{z}}_l(1,T)$ is injectively mapped through the transmitter codebook from the ith sample drawn according to $P_{\mathrm{TX}}(\boldsymbol{u}_{\mathrm{TX}}(1,T))$. The decoder assumes all the probability mass is located at the decoded solution $\hat{\boldsymbol{z}}_{\mathrm{RX}q}(1,T)$ so,

$$F(\mathrm{RX}q) \quad \approx \quad \frac{1}{\ell}\sum_{l=1}^{\ell} P_{\mathrm{RX}q}(\hat{\boldsymbol{z}}_{\mathrm{RX}q}(1,T)|\tilde{\boldsymbol{z}}_l(1,T)) \ln P_{\mathrm{RX}q}(\tilde{\boldsymbol{z}}_l(1,T)|\hat{\boldsymbol{z}}_{\mathrm{RX}q}(1,T)), \tag{38}$$

which yields an expression for the whole receiver RXq. Since the expression is only dependent on the receiver, the quality of the estimate depends on the veracity of the receiver's components including the effectiveness of its search algorithm in finding the global minimum. Unfortunately, the distributions conditional only on driver variables may not be directly available from the decoder, and must be obtained by suitable marginalisation of unwanted components in the state vectors. In the context of ionospheric modelling, each of the ℓ samples may be a sequence collected from a different day's data. When ℓ is small, alternative measures which penalise model complexity should also be considered, for example stochastic complexity (e.g. see [6]). It is difficult to evaluate a model based on a single sample, i.e. when $\ell = 1$, though useful insight is still possible.

For the level 1 receiver, as in Section 3.4, assume the noise process for the memoryless channel is zero-mean discretised Gaussian and stationary in time and space. Furthermore, assume that there is no source memory and that the prior $P_{\mathrm{RX1}}(\hat{\boldsymbol{z}}_{\mathrm{RX1}}(t))$ is uniform. Then,

$$F(\mathrm{RX1}) \quad \approx \quad \frac{1}{\ell}\sum_{l=1}^{\ell}[\prod_{t=1}^{T} P_{\mathrm{RX1}}(\hat{\boldsymbol{z}}_{\mathrm{RX1}}(t)|\tilde{\boldsymbol{z}}_l(t))] \sum_{t=1}^{T} \ln P_{\mathrm{RX1}}(\tilde{\boldsymbol{z}}_l(t)|\hat{\boldsymbol{z}}_{\mathrm{RX1}}(t)), \tag{39}$$

where,

$$P_{\mathrm{RX1}}(\hat{\boldsymbol{z}}_{\mathrm{RX1}}(t)|\tilde{\boldsymbol{z}}_l(t)) \quad = \quad \frac{P_{\mathrm{RX1}}(\tilde{\boldsymbol{z}}_l(t)|\hat{\boldsymbol{z}}_{\mathrm{RX1}}(t))}{\sum_{\boldsymbol{z}_{\mathrm{RX1}} \in Z_{\mathrm{RX1}}} P_{\mathrm{RX1}}(\tilde{\boldsymbol{z}}_l(t)|\boldsymbol{z}_{\mathrm{RX1}})}, \tag{40}$$

and $P_{\mathrm{RX1}}(\tilde{\boldsymbol{z}}_i(t)|\hat{\boldsymbol{z}}_{\mathrm{RX1}}(t)) = N(\tilde{\boldsymbol{z}}_i(t); \hat{\boldsymbol{z}}_{\mathrm{RX1}}(t), \boldsymbol{R})$.

5 Sensitivity

Sensitivity of a sequence of true images $\boldsymbol{z}_{\mathrm{TX}}(1,T)$ to the different drivers in $\boldsymbol{u}_{\mathrm{TX}}(1,T)$ is of scientific interest. For example, during a geomagnetic storm, what drivers are most influential in producing the particular electron content patterns in the ionosphere? If the true real-world process is nonlinear, sensitivity must typically be evaluated at each sequence of driver variables of interest. Using the Fisher information (see "Fisher information" in [19]),

$$J_{ij}(\boldsymbol{u}_{\mathrm{TX}}(1,T); \mathcal{M}_{\mathrm{TX}}) \quad = \quad \sum_{\boldsymbol{z}_{\mathrm{TX}}(1,T) \in \otimes_{t=1}^{T} Z_{\mathrm{TX}}} \frac{\delta^2 \ln P(\boldsymbol{z}_{\mathrm{TX}}(1,T)|\boldsymbol{u}_{\mathrm{TX}}(1,T))}{\delta[\boldsymbol{u}_{\mathrm{TX}}(1,T)]^i \delta[\boldsymbol{u}_{\mathrm{TX}}(1,T)]^j} P(\boldsymbol{z}_{\mathrm{TX}}(1,T)|\boldsymbol{u}_{\mathrm{TX}}(1,T)), \tag{41}$$

where $[\boldsymbol{u}_{\mathrm{TX}}(1,T)]^i$ is the ith component in the sequence of driver variables, and $\mathcal{M}_{\mathrm{TX}}$ refers to the real-world process. The finite differences must be evaluated at the particular driver sequence of interest. Relating to drivers, δ denotes discrete differences but under the assumption that all values in the relevant component in $\boldsymbol{u}_{\mathrm{TX}}(1,T)$ are uniformly spaced. Since the true images are unknown,

$$J_{ij}(\boldsymbol{u}_{\mathrm{TX}}(1,T); \mathcal{M}_{\mathrm{TX}}) \quad \approx \quad \sum_{\tilde{\boldsymbol{z}}(1,T) \in \otimes_{t=1}^{T} Z} \frac{\delta^2 \ln P(\tilde{\boldsymbol{z}}(1,T)|\boldsymbol{u}_{\mathrm{TX}}(1,T))}{\delta[\boldsymbol{u}_{\mathrm{TX}}(1,T)]^i \delta[\boldsymbol{u}_{\mathrm{TX}}(1,T)]^j} P(\tilde{\boldsymbol{z}}(1,T)|\boldsymbol{u}_{\mathrm{TX}}(1,T)). \tag{42}$$

Unfortunately, the transmitter distributions are also unknown, so the Fisher information must be approximated at the receiver,

$$J_{ij}(\boldsymbol{u}_{\mathrm{TX}}(1,T); \mathcal{M}_{\mathrm{TX}}) \quad \approx \quad J_{ij}(\boldsymbol{u}_{\mathrm{RX}q}(1,T); \mathcal{M}_{\mathrm{RX}q})$$

$$= \sum_{\tilde{\boldsymbol{z}}(1,T) \in \otimes_{t=1}^{T} Z} \frac{\delta^2 \ln P_{\mathrm{RX}q}(\tilde{\boldsymbol{z}}(1,T)|\boldsymbol{x}_{\mathrm{RX}q}(1,T))}{\delta[\boldsymbol{u}_{\mathrm{RX}q}(1,T)]^i \delta[\boldsymbol{u}_{\mathrm{RX}q}(1,T)]^j} P_{\mathrm{RX}q}(\tilde{\boldsymbol{z}}(1,T)|\boldsymbol{x}_{\mathrm{RX}q}(1,T)). \tag{43}$$

The Fisher information defined upon the receiver is subject to the receiver's statistical assumptions. Since the Fisher information is defined on $\mathcal{M}_{\mathrm{RX}q}$, it does not evaluate the sensitivity of the decoded solution $\hat{\boldsymbol{x}}_{\mathrm{RX}q}(1,T)$ or $\hat{\boldsymbol{u}}_{\mathrm{RX}q}(1,T)$. In terms of the objective function, and assuming $P_{\mathrm{RX}q}(\boldsymbol{x}_{\mathrm{RX}q}(1,T))$ is uniform,

$$J_{ij}(\boldsymbol{u}_{\mathrm{RX}q}(1,T);\mathcal{M}_{\mathrm{RX}q}) = - \sum_{\tilde{\boldsymbol{z}}(1,T) \in \otimes_{t=1}^{T} Z} \frac{\delta^2 f_{\mathrm{RX}q}(\tilde{\boldsymbol{z}}(1,T),\boldsymbol{x}_{\mathrm{RX}q}(1,T))}{\delta[\boldsymbol{u}_{\mathrm{RX}q}(1,T)]^i \delta[\boldsymbol{u}_{\mathrm{RX}q}(1,T)]^j} P_{\mathrm{RX}q}(\tilde{\boldsymbol{z}}(1,T)|\boldsymbol{x}_{\mathrm{RX}q}(1,T)). \tag{44}$$

Then given a single sample $\tilde{\boldsymbol{z}}_l(1,T)$ and a level 1 receiver,

$$J_{ij}(\boldsymbol{u}_{\mathrm{RX1}}(1,T);\mathcal{M}_{\mathrm{RX1}}) \approx -\frac{\delta^2 f_{\mathrm{RX1}}(\tilde{\boldsymbol{z}}_l(1,T),\boldsymbol{u}_{\mathrm{RX1}}(1,T))}{\delta[\boldsymbol{u}_{\mathrm{RX1}}(1,T)]^i \delta[\boldsymbol{u}_{\mathrm{RX1}}(1,T)]^j}, \tag{45}$$

noting $\boldsymbol{x}_{\mathrm{RX1}}(1,T) = \boldsymbol{u}_{\mathrm{RX1}}(1,T)$. In this case the Fisher information may be 'estimated' using the negative Hessian of the objective function. However such an 'estimate' may be misleading since it is 'tuned' to one particular image sequence only.

6 Discussion

The selection of driver and image variables for $U_{\mathrm{RX}q}$ and Z respectively is critical. If the selection is not sufficiently descriptive, then the noise distributions implied in the transmitter may be very broad due to the effect of latent variables. Too much useful information may then be lost in the encoding process. Unfortunately, the choice is often restricted by the particular receiver codebook, e.g. empirical or mathematical model, used.

In designing the receiver, the codebook is intuitively most important and effort should first be directed at improving this. The choice of noise model and state transition model should be data-dependent, since they should be learnt from data. This influences the choice of objective function and level of receiver. The simpler receivers should be more robust if there is a lack of good quality data. However it is important to understand the assumptions implicit in the simpler receivers, and that the objective function may in effect 'penalise the same mismatch more than once'. Of course, the level 3 and level 2 receivers reduce to the level 1 receiver if the level 1 assumptions hold in the transmitter. The choice of h_s, h_c and h may be driven by limitations in data rather than scientific knowledge. If the receiver codebook mapping is bijective, then the inverse problem of decoding state variables $\hat{\boldsymbol{x}}_{\mathrm{RX}q}(1,T)$ for a noisy image sequence $\tilde{\boldsymbol{z}}(1,T)$ has a unique solution, providing the objective function has a single global minimum. If the receiver is modelled as a HMM, then a full Viterbi decoder [15] would be very computationally expensive unless the codebook mapping was precomputed and implemented in lookup tables. One of the fundamental limitations of the channel model proposed is its inability to model noise sources which are not additive, for example convolutional noise.

The receiver codebook has been defined using a single deterministic model. However multiple models may be used in parallel where the data fusion occurs at the level of $\boldsymbol{z}_{\mathrm{RX}q}(1,T)$ or $\boldsymbol{x}_{\mathrm{RX}q}(1,T)$. For ionospheric modelling, the fusion may additionally use geographic information where alternative ionospheric models are weighted differently at different global locations, for example according to their ability to model low latitude or polar/auroral processes.

Similar 'receivers' have been developed in nonlinear time series analysis, for example the nonlinear autoreggresive (NLAR) model [3], nonlinear moving average (NLMA) model [18] and state-dependent model (SDM) [14]; these however assume the state is not hidden but directly observed. Variable order Markov models (VMMs) [2] also regard states as observed and hence only encode source, rather than both channel and source, memory. The ARMA-filtered hidden Markov model [12] assumes various linear dependencies, for example that the observation is linearly dependent on the mean vectors of previous states and an innovation which is also dependent on previous states. The ARMA-filtered HMM may possibly be regarded as a constrained special case of the level 2 receiver, at least in concept if not mathematically. There are similarities between the view of the ionosphere in [7] and the 'transmitter' described above, but without development in terms of a discrete communication channel.

Image-model coupling is simply an attempt to recognise or classify an image in terms of its driver variables, or regress an image onto its drivers. The application of further techniques from machine learning and decision theory [5] may be useful. The framework described should be applicable to other tasks in image-model coupling, or indeed classification and regression.

Besides the application to the ionosphere, the approach assuming channel memory may be useful for other geophysical systems which include integrator-like processes or exhibit 'sluggish responses'. Possibilities include long-term climate modelling, the gradual build-up of stresses along fault lines prior to an earthquake, the accumulation of water vapour resulting in a hurricane, or sustained rainfall producing flooding. As stated, a challenge for such systems is the selection of driver and image variables. For example, drivers for the ionosphere should include measures of solar and geomagnetic activity since it is known the sun and geomagnetic state of the Earth influence the distribution and movement of ionospheric plasma [8]. However if a 3-dimensional map of electron density measurements is used as the image, but the resolution is too coarse, the ability of the receiver to 'recognise' small-scale structures is inhibited. A simple ionospheric application, ignoring channel memory and with a very simple noise model, is detailed in [17].

7 Conclusions

Presented above is an information theoretic framework describing image-model coupling when the true-world system has memory. Examples of such systems include the ionosphere and other geophysical systems with a 'sluggish response'. A discrete channel model is used to help quantify the match between images and model output, and analyse the coupling. The approach is statistical in nature. It should be possible to harness any increased availability of data to derive objective functions which better reflect the spatial and temporal statistical relationships in the true underlying process, and thereby improve coupling accuracy. However for complex systems such as the ionosphere it is probably more beneficial to first direct effort at improving the accuracy of the proposed model (i.e. the codebook). It is hoped that the framework described above may encourage the further use of statistical and information-based 'tools' in image-model coupling and its analysis. In general, image-model coupling may be used to help us better understand the underlying processes which produce the effects being imaged, but also better understand the limitations of the models themselves. A practical application of some of these concepts, using a very simple objective function, is presented for ionospheric modelling in [17].

A Distributions implicit in the transmitter

The noisy image $\tilde{\boldsymbol{z}}(t)$ may be regarded as sampled from a distribution, the functional form of which varies with scale, i.e. the number of conditional variables. An expression for the fully marginalised distribution may be derived under the following assumptions, consistent with the transmitter illustrated in Figure 1.

- Each driver variable in $U_{\mathrm{TX}} \otimes U'_{\mathrm{TX}}$ is linearly independent of all other driver variables.
- Each description variable in $Z \otimes Z'$ is linearly independent of all other description variables.
- Measurement noise is stationary temporally, and spatially with regard to $\boldsymbol{z}_{\mathrm{TX}}(t)$. Hence $P_{\mathrm{TX}}(\boldsymbol{n}_{\mathrm{TX}}(t)|\boldsymbol{z}_{\mathrm{TX}}(t)) = P_{\mathrm{TX}}(\boldsymbol{n}_{\mathrm{TX}}) \forall \boldsymbol{z}_{\mathrm{TX}}(t)$.

- The current noisy image $\tilde{\boldsymbol{z}}(t)$ is fully and uniquely determined by an initialisation h_c timesteps previous where $h_c \in \mathbb{N}_0$, the history of driver variables since then, and the current measurement noise, i.e. the following mapping is injective,

$$(\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t), \boldsymbol{u}'_{\mathrm{TX}}(t-h_c,t), \boldsymbol{z}_{\mathrm{TX}}(t-h_c), \boldsymbol{z}'_{\mathrm{TX}}(t-h_c), \boldsymbol{n}_{\mathrm{TX}}(t)) \mapsto \tilde{\boldsymbol{z}}(t). \tag{46}$$

  If this condition does not exist, then a value of h_c is chosen such that the history of driver variables prior to timestep $(t-h_c)$ has no significant effect on the current description.

- Temporal causality is assumed so only past events influence the current description.

So,

$$P_{\mathrm{TX}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t), \boldsymbol{u}'_{\mathrm{TX}}(t-h_c,t), \boldsymbol{z}_{\mathrm{TX}}(t-h_c), \boldsymbol{z}'_{\mathrm{TX}}(t-h_c)) \quad = \quad P_{\mathrm{TX}}(\boldsymbol{n}_{\mathrm{TX}})\delta(\boldsymbol{n}_{\mathrm{TX}}, \tilde{\boldsymbol{z}}(t) - \boldsymbol{z}_{\mathrm{TX}}(t)), \tag{47}$$

where $\delta(\cdot,\cdot)$ is the Kronecker delta. Introducing redundant variables into the list of conditional variables,

$$\begin{aligned} &P_{\mathrm{TX}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t), \boldsymbol{u}'_{\mathrm{TX}}(t-h_c,t), \boldsymbol{z}_{\mathrm{TX}}(t-h_c,t), \boldsymbol{z}'_{\mathrm{TX}}(t-h_c,t)) \\ &\qquad = P_{\mathrm{TX}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t), \boldsymbol{u}'_{\mathrm{TX}}(t-h_c,t), \boldsymbol{z}_{\mathrm{TX}}(t-h_c), \boldsymbol{z}'_{\mathrm{TX}}(t-h_c)). \end{aligned} \tag{48}$$

Marginalising over the unknown description and driver variables, and substituting from above,

$$\begin{aligned} &P_{\mathrm{TX}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t), \boldsymbol{z}_{\mathrm{TX}}(t-h_c,t)) \\ = &\sum_{\boldsymbol{u}'_{\mathrm{TX}}(t-h_c,t)\in\otimes_{i=1}^{h_c+1}U'_{\mathrm{TX}}} \; \sum_{\boldsymbol{z}'_{\mathrm{TX}}(t-h_c,t)\in\otimes_{i=1}^{h_c+1}Z'_{\mathrm{TX}}} P_{\mathrm{TX}}(\boldsymbol{u}'_{\mathrm{TX}}(t-h_c,t), \boldsymbol{z}'_{\mathrm{TX}}(t-h_c,t)) \\ &P_{\mathrm{TX}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t), \boldsymbol{u}'_{\mathrm{TX}}(t-h_c,t), \boldsymbol{z}_{\mathrm{TX}}(t-h_c,t), \boldsymbol{z}'_{\mathrm{TX}}(t-h_c,t)) \\ = &\sum_{\boldsymbol{u}'_{\mathrm{TX}}(t-h_c,t)\in\otimes_{i=1}^{h_c+1}U'_{\mathrm{TX}}} \; \sum_{\boldsymbol{z}'_{\mathrm{TX}}(t-h_c,t)\in\otimes_{i=1}^{h_c+1}Z'_{\mathrm{TX}}} P_{\mathrm{TX}}(\boldsymbol{n}_{\mathrm{TX}})\delta(\boldsymbol{n}_{\mathrm{TX}}, \tilde{\boldsymbol{z}}(t) - \boldsymbol{z}_{\mathrm{TX}}(t)) \\ &[\prod_{a=0}^{h_c-1} P_{\mathrm{TX}}(\boldsymbol{z}'_{\mathrm{TX}}(t-a)|\boldsymbol{z}'_{\mathrm{TX}}(t-h_c,t-a-1), \boldsymbol{u}'_{\mathrm{TX}}(t-h_c,t-a))]P_{\mathrm{TX}}(\boldsymbol{z}'_{\mathrm{TX}}(t-h_c)|\boldsymbol{u}'_{\mathrm{TX}}(t-h_c)) \\ &[\prod_{a=0}^{h_c-1} P_{\mathrm{TX}}(\boldsymbol{u}'_{\mathrm{TX}}(t-a)|\boldsymbol{u}'_{\mathrm{TX}}(t-h_c,t-a-1))]P_{\mathrm{TX}}(\boldsymbol{u}'_{\mathrm{TX}}(t-h_c)). \end{aligned} \tag{49}$$

Then at a still coarser scale,

$$\begin{aligned} &P_{\mathrm{TX}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{TX}}(t)) \\ = &\sum_{\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t-1)\in\otimes_{i=1}^{h_c}U_{\mathrm{TX}}} \; \sum_{\boldsymbol{z}_{\mathrm{TX}}(t-h_c,t)\in\otimes_{i=1}^{h_c+1}Z_{\mathrm{TX}}} P_{\mathrm{TX}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t), \boldsymbol{z}_{\mathrm{TX}}(t-h_c,t)) \\ &P_{\mathrm{TX}}(\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t-1), \boldsymbol{z}_{\mathrm{TX}}(t-h_c,t)) \\ = &\sum_{\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t-1)\in\otimes_{i=1}^{h_c}U_{\mathrm{TX}}} \; \sum_{\boldsymbol{z}_{\mathrm{TX}}(t-h_c,t)\in\otimes_{i=1}^{h_c+1}Z_{\mathrm{TX}}} P_{\mathrm{TX}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t), \boldsymbol{z}_{\mathrm{TX}}(t-h_c,t)) \\ &[\prod_{a=1}^{h_c-1} P_{\mathrm{TX}}(\boldsymbol{z}_{\mathrm{TX}}(t-a)|\boldsymbol{z}_{\mathrm{TX}}(t-h_c,t-a-1), \boldsymbol{u}_{\mathrm{TX}}(t-h_c,t-a))] \\ &P_{\mathrm{TX}}(\boldsymbol{z}_{\mathrm{TX}}(t)|\boldsymbol{z}_{\mathrm{TX}}(t-h_c,t-1), \boldsymbol{u}_{\mathrm{TX}}(t-h_c,t-1))P_{\mathrm{TX}}(\boldsymbol{z}_{\mathrm{TX}}(t-h_c)|\boldsymbol{u}_{\mathrm{TX}}(t-h_c)) \\ &[\prod_{a=1}^{h_c-1} P_{\mathrm{TX}}(\boldsymbol{u}_{\mathrm{TX}}(t-a)|\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t-a-1))]P_{\mathrm{TX}}(\boldsymbol{u}_{\mathrm{TX}}(t-h_c)), \end{aligned} \tag{50}$$

where $P_{\mathrm{TX}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{TX}}(t-h_c,t), \boldsymbol{z}_{\mathrm{TX}}(t-h_c,t)$ is as given in Equation 49 above. This expression gives an indication of the complexity implied in the distribution $P_{\mathrm{TX}}(\tilde{\boldsymbol{z}}(t)|\boldsymbol{u}_{\mathrm{TX}}(t))$. Much of the complexity derives from the conditional probability terms introduced in marginalising over, or 'averaging out', all possible histories. Of course the expression is simplified if the deeper dependencies do not exist, for example if h_c is small, or if the source memory length is much shorter than the channel memory length, i.e. $h_s << h_c$.

Acknowledgments

Approximately and in brief, N.D. Smith was primarily responsible for the development and writing; C.N. Mitchell gave the initial context and motivation for the work; all helped in proofreading and giving suggestions. N.D. Smith and C.J. Budd are grateful for being supported in BICS by EPSRC grant GR/S86525/01. C.N. Mitchell thanks the EPSRC for financial support. N.D. Smith would like to thank individuals for useful comments or discussion, in particular Simon Harris for help with expanding states over time to ensure Markov properties, Dimitry Pokhotelov regarding ionospheric science, Merrilee Hurn, Jan Van lent, Gundula Behrens and Gary Bust.

References

[1] N. Abramson. *Information Theory and Coding.* McGraw-Hill Electronic Sciences Series. McGraw-Hill Book Company, Inc., 1963.

[2] R. Begleiter, R. El-Yaniv, and G. Yona. On Prediction Using Variable Order Markov Models. *Journal of Artificial Intelligence Research*, 22:385–421, 2004.

[3] C. Chatfield. *The Analysis of Time Series: An Introduction.* Texts in Statistical Science. Chapman & Hall/CRC, CRC Press LLC, Sixth edition, 2004.

[4] R. Daley. *Atmospheric Data Analysis.* Cambridge atmospheric and space science series. Cambridge University Press, 1999.

[5] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification.* A Wiley-Interscience Publication. John Wiley & Sons,Inc., Second edition, 2001.

[6] P.D. Grünwald, I.J. Myung, and M.A. Pitt, editors. *Advances in Minimum Description Length: Theory and Applications.* Neural Information Processing Series. The MIT Press, 2005.

[7] J.-S. Guo, S.-P. Shang, J. Shi, M. Zhang, X. Luo, and H. Zheng. Optimal assimilation for ionospheric weather - Theoretical aspect. *Space Science Reviews*, 107(1-2):229–250, 2003.

[8] J.K. Hargreaves. *The solar-terrestrial environment.* Cambridge atmospheric and space science series. Cambridge University Press, 2003.

[9] A.I. Khinchin. *Mathematical Foundations of Information Theory.* Dover Publications, Inc., 1957. (translated by R.A. Silverman and M.D. Friedman).

[10] G.A. Korn and T.M. Korn. *Mathematical Handbook for Scientists and Engineers: Definitions, Theorems, and Formulas for Reference and Review.* McGraw-Hill,Inc., second, enlarged and revised edition, 1968.

[11] D.J.C. MacKay. *Information Theory, Inference and Learning Algorithms.* Cambridge University Press, 2004.

[12] S. Michalek, M. Wagner, and J. Timmer. A New Approximate Likelihood Estimator for ARMA-Filtered Hidden Markov Models. *IEEE Transactions on Signal Processing*, 48(6):1537–1547, 2000.

[13] M.J.D. Powell. A view of algorithms for optimization without derivatives. *Mathematics TODAY*, 43(5):170–174, 2007.

[14] M.B. Priestley. *Non-linear and Non-stationary Time Series Analysis.* Academic Press Limited, 1988.

[15] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition.* Prentice Hall Signal Processing Series. PTR Prentice-Hall, Inc., 1993.

[16] P. Salamon, P. Sibani, and R. Frost. *Facts, Conjectures, and Improvements for Simulated Annealing.* SIAM Monographs on Mathematical Modeling and Computation. Society for Industrial and Applied Mathematics, 2002.

[17] N.D. Smith, D. Pokhotelov, C.N. Mitchell, and C.J. Budd. Image-model coupling: application to an ionospheric storm, 2008. BICS preprint available at: http://www.bath.ac.uk/math-sci/bics/preprints/index.html.

[18] H. Tong. *Non-linear Time Series: A Dynamical System Approach.* Oxford Statistical Science Series. Oxford University Press, 1990.

[19] Wikipedia, access: October 2008. http://www.wikipedia.org.