UNIVERSITY OF CALIFORNIA
SANTA CRUZ

QUANTIFYING THE IMPACT OF CLIMATE CHANGE ON OCEANIC VARIABLES.

A dissertation submitted in partial satisfaction of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in

APPLIED MATHEMATICS AND STATISTICS

by

Francisco M Beltrán

March 2014

The Dissertation of Francisco M Beltrán
is approved:

Professor Bruno Sansó, Chair

Professor Raquel Prado

Professor Abel Rodríguez

Tyrus Miller
Vice Provost and Dean of Graduate Studies

Table of Contents

List of Figures

List of Tables

Abstract

Quantifying the impact of climate change on oceanic variables.

by

Francisco M Beltrán

The focus of this thesis is to develop a general methodology to obtain high-resolution spatial-temporal forecasts of Sea Surface Temperature (SST) using an ensembles of general circulation model (GCM) output and historical records as the major driving force. As a case study, we consider Sea Surface Temperature (SST) in the North Pacific Ocean. We use two ensembles of different GCM simulation output, made available in the 4th Assessment Report of the Intergovernmental Panel on Climate Change: one corresponds to 20th century forcing conditions and the other corresponds to the A1B emissions scenario for the 21st century. Given a representation of the SST spatio-temporal fields based on a common set of empirical orthogonal functions (EOFs), we use a hierarchical Bayesian model for the EOF coefficients to estimate a baseline and a set of model discrepancies. These components are all time-varying. The model enables us to extract relevant temporal patterns of variability from both the observations and simulations and obtain common patterns from all eighteen series. This is used to obtain unified 21st century forecasts of relevant oceanic indexes as well as whole fields of forecast North Pacific SST. The unified forecast captures large longterm oceanic behavior, however the coarse resolution prevents us from capturing coastal behaviors. We use the unified forecast to model high resolution SST by establishing a link between large and small scale

variability using statistical downscaling techniques. Using a combination of a discrete process convolution and a dynamic linear model, we obtain a smooth high-resolution forecast of SST fields off the coast of California. To model the high resolution data faster and efficiently, we developed and implement a parallel version of the forward filtering backwards sampling algorithm. We finish the work with remarks on the model results and address future avenues this work can take.

Acknowledgments

This dissertation would not have been possible without the support of many people.

I would like to thank my advisor and friend, Professor Bruno Sansó for the tremendous support and guidance he gave me throughout my entire graduate studies at UC Santa Cruz. His enthusiasm, endless support, and knowledge allowed me to succeed as a student and improve as a person. It has been an honor and a privilege, both from a personal and academic stand point, to have worked with him. I am forever grateful.

My profound thanks go out to Ricardo Lemos. His willingness to help and intuition set the foundation for many of the paths that I took. I would also like to thank Roy Mendelssohn for the opportunities, valuable advice, and suggestions on my research. My appreciation for their help and efforts is beyond words.

I would also like to thank my colleagues Marian, Kassie, and Tracy who made graduate school a fun place to be and made the tough days seems bearable. I am very fortunate to have shared my graduate experience with them.

I would like to thank my wife Tiffany. Without her support, love, and patience I would not be where I am today. She gave me the strength I needed to accomplish my goals. My parents, my brothers and sisters, for their unconditional love and support throughout my entire life. The sacrifices made by my wife and family are a treasure that I will be eternally grateful for.

Finally I would like to thank Juan Carlos, Paul, and Kayden for being the best friends a person could ask for. Their friendship is something I will have for the rest of

my life. Their support, encouragement, advice, and cooking are experiences that I will never be able to repay. Thank you fellas.

Chapter 1

Introduction

Average global sea surface temperature is used as one of the baselines for gauging and depicting climate change. These changes are projected to affect weather patterns resulting in more occurrences of severe storms and changes in seasonal agricultural growing patterns, thus leading to an adverse effect on food production and marine ecosystems. Quantifying and assessing these changes is an important and complicated problem. Sea Surface Temperature (SST) measurements have been recorded as early as 1700s. This was accomplished by passerbys on large ships sailing across the Atlantic, writing down measurements they obtained with mercury thermometers. Since then, scientists have used large cargo ships to collect measurements at automated times during their passage as well as setting up buoys and weather stations. This has given us an incredible amount information about our past climate. When coupled with our current technology of high resolution satellite data, it allows us to create useful models in predicting future climate. As interest in climate change continues to grow, so does the

need for higher resolution climate models. Massive space-time datasets are produced using computer model simulations to study natural phenomena such as climate, oceanic behavior, and weather. These multivariate spatio-temporal outputs are simulated over many decades and large areas to capture long term behavior over the regions.

In its fourth assessment report (AR4), the Intergovernmental Panel on Climate Change (IPCC) made data available for 24 Global Climate Models (GCMs) under different emission scenarios. If we inspect the different datasets, we see substantial disagreement in both hindcasts and future predictions, a fact that hampers the decision making process. The question that arises is how to blend the information from ensembles of climate model simulations in such a way that we can obtain a better depiction of future climate. Knutti et al. (2010) summarize the challenges involved in such a task. The simplest approach is to weigh all models equally and take an ensemble average. This clearly disregards the fact that some models may be more accurate than others, or accuracy may differ at different time-scales. Alternatively, we can produce a weighted average by assigning weights to models depending on their agreement with observational records. This is known in the climate science literature as Reliability Ensemble Average (REA; Giorgi and Mearns, 2002). Both approaches are usually applied to very large regions and large periods of time, on the premise that a clear signal can not be found at finer resolutions. Interest in the statistical community for this problem has been growing the last few years, following the early work in Tebaldi et al. (2005).

A problem that remains with this framework is the following: Is it fair to compare climate model simulations for, say, a given year to the corresponding observational

records? A simulation indexed by a given year is not meant to reproduce that year's observations. It is just a sample from the climate that is typical of that year, as estimated by the climate model. Averaging over large areas and time spans to analyze ensembles of climate model simulations is a way to compensate for the fact that such models are not meant to reproduce specific weather conditions that affect individual observations. The ability to produce forecasts for coastal areas at a seasonal level is key to assess the impact of climate on marine ecosystems. Similarly, the population dynamics of many species can be affected by changes in the phase and amplitude of the seasonal cycles. These are examples of the need to have climate forecasts with high temporal and spatial resolution.

The availability of high resolution spatial measurements make assessing local regions possible. These models have only been available for short periods of time so capturing long term behavior from these observations is not possible. So how can we establish a relationship between the models that provide longterm behavior with the high resolution short-term observations? How do we handle and model these measurements when the information being studied is too large to fit on standard computational devices? These are the questions that arise when we try to implement statistical models to large datasets. Inference for models that use simple spatial structures are impractical with serial programing techniques, and thus parallel implementation is necessary.

1.1 Outline

In chapter 2 we discuss how to extract important indices from spatio-temporal data using a specific set of basis functions. The resulting temporal variation is then modeled using a Bayesian hierarchical model to blend information from different GCMs. In chapter 3, we present a method to establish a link between large and small scale variability using statistical downscaling techniques. We use the methods in chapter 2 to obtain longterm oceanic trends and use that large scale information to obtain a smooth spatio-temporal field. The computational problems that arise from the large datasets are described in chapter 4. The implementation of parallel computing are discussed and the algorithms are presented in full detail.

Chapter 2

Blended oceanic indexes projections.

In this chapter, we develop a general methodology to obtain joint projections of climate indexes using a multi-model ensemble of GCMs. Our approach begins with extracting global spatial features of both observations and simulations. We then model their temporal variability at a seasonal level in a way that allows for smooth temporal changes. We explore three levels of temporal resolution – decadal, seasonal and monthly – to gauge the possibility of blending multi-model ensembles at high temporal frequencies. We apply our methods to indexes that are related to the Pacific Decadal Oscillation (PDO) and the North Pacific Gyre Oscillation (NPGO), and then assume that indexes obtained from observational data correspond to noisy versions of the processes of interest. Simulations from the GCMs also produce indexes that are noisy versions of the processes, but with the addition of discrepancy terms. We propose a model that assumes a smooth evolution in time for such discrepancies and use a Bayesian hierarchical model to weigh the simulations and obtain 21st century projections of the underlying

process for each oceanic index. The resulting blended time-varying coefficients of the main modes of spatial variability are then used for the reconstruction of the temperature fields. Section 2 contains a description of the data, the simulations, and the methods to obtain the oceanic indexes. Section 3 has a description of the models proposed for the different time scales considered. Section 4 reports the results, and Section 5 discusses the conclusions of our analysis.

2.1 Introduction

It is clear that extracting large scale spatial features from a spatio-temporal field can be a useful approach to blend information from climate model simulations indexed in space and time. This can be conveniently achieved by representing the field using a set of basis functions. An example is the well known Karhunen-Loève (KL) expansion (see, for example, Yaglom, 1986). This provides a representation of a spatio-temporal random field that is determined by its covariance function. Consider the process $x_t(\boldsymbol{s})$, where t indicates time and $\boldsymbol{s}$ location. Suppose that the covariance function $v(\boldsymbol{s}, \boldsymbol{s}') = \text{cov}(x_t(\boldsymbol{s}), x_t(\boldsymbol{s}'))$ does not depend on t. Then $x_t(s) = \sum_{j=1}^{\infty} \sqrt{\lambda_j}\psi_j(s)\alpha_j(t)$ for a set of orthogonal functions $\psi_1, \psi_2, \ldots$, random variables $\alpha_j(t)$ with $\text{cov}(\alpha_j(t), \alpha_i(t)) = \delta_{ij}$, and non-negative λ_j that satisfy the integral equation $\int v(\boldsymbol{s}, \boldsymbol{s}')\psi_j(\boldsymbol{s}')d\boldsymbol{s}' = \lambda_j\psi_j(\boldsymbol{s})$. KL expansions are difficult to obtain in general. In a finite setting they are obtained from the principle component (PC) analysis of the covariance matrix corresponding to the space-time process. The resulting $\psi_j(s)$ are known as the Empirical Orthogonal

Functions (EOF) and are used to reduce a dataset's dimensionality by extracting the main modes of variability (Hannachi et al., 2007; Cressie and Wikle, 2011; Jolliffe, 2002).

EOFs are just one example of possible basis functions that are used to represent random fields. EOFs are very popular in environmental sciences as the scientific community has linked the indexes to physical and biological effects. Unfortunately, their estimation often ignores important trends as well as time-varying dependencies between observations at different locations. Additionally, EOF analysis suffers from a number of issues: it depends heavily on data availability; large eigenvalues are likely to be inflated; the absence or addition of a station may alter the eigenvalues, resulting in significant changes in the estimation of spatial and temporal variability; if the data are non-stationary, the estimation of the covariance matrix becomes problematic; EOFs identify spatial patterns but do not propagate these patterns in time. Finally, as for any finite representation on basis functions, truncation of the number of components limits the total variability explained by the expansion. In spite of all the drawbacks, the reasons for the popularity of EOFs are that they are simple to calculate and, being a discrete version of a KL expansion, they provide results that are appealing to the scientific community.

When comparing observational records with climate model simulations, using a finite basis function representation can allow us to focus on the major patterns of variability. Even when the observations and the different simulated fields are very dissimilar, it is possible that there are commonalities between the components used to represent them. As for EOFs, they have been used in the atmospheric and oceanic

sciences to produce environmental indexes. Recent examples of EOF based indexes are the Pacific Decadal Oscillation (PDO; Mantua and Hare, 2002), the North Pacific Gyre Oscillation (NPGO; Di Lorenzo et al., 2008), and the Arctic Oscillation (AO; Thompson and Wallace, 1998). These indexes have been used in studying climate effects on salmon production (Mantua et al., 1997); when describing physical and biological changes in the North Pacific (Di Lorenzo et al., 2009); they have been linked to variations in the Kuroshio-Oyashio Extension (Di Lorenzo et al., 2008, 2009); and they have been associated with fish abundance in the San Francisco Bay (Cloern et al., 2010). Providing unified forecasts of EOF coefficients is not only useful for climate model simulation assessment, it is an important issue in itself.

2.2 Data

2.2.1 Historical Records

We consider 5° gridded SST for the 20th century (1900-1999), in the North Pacific region 22.5°N-62.5°N, 112.5°E-247.5°E. This results in a total of 171 grid cells. Observational data stem from the UK Meteorological Office, Hadley Centre (Rayner et al., 2003). We aggregate (to 5° resolution) the available 1° gridded monthly means, to make them compatible with calculations used for the PDO, and we also use three levels of temporal aggregation: Monthly (M, no aggregation), Quarterly (Q), and Monthly Decadal (D). We use these three levels to see how much information we can extract from each resolution. For the quarterly data we take the means of December–February,

March–May, June–August, and September–November for every year. For decadal we average the ten months of January of each decade, then the ten months of February, and so on, for each month. Thus, in the monthly case, we have, for each grid cell, 18 time series with 2400 time steps. In the quarterly case we have 800 time steps and in the decadal case we have 240 time steps.

2.2.2 Global Climate Model data

We consider an ensemble of SST simulations from seventeen different GCMs (see Section A of the Appendix), used to obtain the results in the IPCC AR4. The GCM output, available from `https://esg.llnl.gov:8443/index.jsp`, stemmed from the World Climate Research Programme's (WCRPs) Coupled Model Intercomparison Project (CMIP3) multi-model data set (Meehl et al., 2007). We obtained model simulations under two types of forcing: Climate of the 20th Century (20C3M), for the years 1900–1999, and Emissions Scenario A1B (Nakicenovic and Swart, 2000), for the years 2000–2099. Under the 20C3M scenario, greenhouse gas forcing is increased as observed in the 20th century. Under A1B, rapid global population and economic growth peak in the mid 21st century, and then start to decline. This scenario assumes that the technological change in energy systems will be balanced between fossil intensive (A1FI) and non-fossil energy sources (A1T). A1B forcing encompasses volcanic aerosols and emissions of sulfur, methane and other greenhouse gases (Randall et al., 2007). Because the spatial resolution of our ensemble of GCMs varies, we aggregate all simulations to a common 5° resolution.

2.2.3 Data processing

Let $X_t^0(\boldsymbol{s})$ denote the SST observation at time t ($t = 1, \ldots, \frac{T}{2}$) and grid point $\boldsymbol{s}$ ($\boldsymbol{s}_{i,j} = (i, j)$, $i = 1, \ldots, 9$, $j = 1, \ldots, 28$). The value of T corresponds to the end of the 21st century and depends on which level of time aggregation is used. To obtain monthly anomalies in the M dataset case, we consider a given location and average over all years for a given month. This produces a 20th century monthly climatology. For each grid cell, we then subtract the corresponding climatology from the observations. For datasets quarterly and decadal we proceed in a similar fashion. We denote the observational anomalies (SSTa) as $\hat{\mathbf{X}}_t^0(\boldsymbol{s})$. Similarly, we obtain the anomalies for each of the climate model simulations.

To obtain the EOF of the observational anomalies we consider the matrix

$$\hat{\mathbf{X}}^0 = \begin{pmatrix} \hat{X}_1^0(1,1) & \hat{X}_2^0(1,1) & \ldots & \hat{X}_{T/2}^0(1,1) \\ \vdots & \vdots & \ddots & \vdots \\ \hat{X}_1^0(9,28) & \hat{X}_2^0(9,28) & \ldots & \hat{X}_{T/2}^0(9,28) \end{pmatrix}$$

and decompose it using the SVD algorithm (Hannachi et al., 2007). Thus

$$\hat{\boldsymbol{X}}^0 = UDV' = \sum_{l=1}^{k} \boldsymbol{u}_l(\boldsymbol{s})\boldsymbol{\psi}_l(t)$$

Letting $p = 9 \times 28$, $\boldsymbol{U}$ is a $p \times p$ orthogonal matrix of left singular vectors, $\boldsymbol{V}$ is a $\frac{T}{2} \times \frac{T}{2}$ orthogonal matrix of right singular vectors, and $\boldsymbol{D}$ is a $p \times T$ diagonal matrix with nonnegative singular values, sorted in decreasing order. $u_l(\boldsymbol{s})$ and $\psi_l(t)$ are $p \times 1$ and $1 \times T$ vectors and are the l-th EOF and the EOF coefficient respectively thus creating a discretized version of the Karhunen-Loéve representation of the spatial

surface. As $u_l(\boldsymbol{s})$ form an orthogonal basis, we can obtain a common representation of all climate model simulation SST anomalies by projecting on it. Denote $\hat{\boldsymbol{X}}^j$ as the matrix of anomalies for the j-th climate model simulations. Then

$$\hat{\boldsymbol{X}}^j = UU^T\hat{\boldsymbol{X}}^j = \sum_{l=1}^{k} \boldsymbol{u}_l(\boldsymbol{s})\boldsymbol{\varphi}_{l,j}(t)$$

where $\varphi_{l,j}(t)$ is a $1 \times \frac{T}{2}$ vector and the coefficient for the l-th observational EOF, corresponding to the j-th member of the ensemble. In this paper we focus on the first two largest modes of variability, corresponding to $l = 1$ and 2, for each of the three time indexes. Figure 2.1 shows the first and second EOFs of the observational anomalies, for time aggregations M, Q and D. The leading EOF for D is similar to the second EOF for M and Q, suggesting that at the Decadal time level the coastal process is driving the variability. We project the GCM anomalies onto the observational EOF, as opposed to calculating an individual EOF for each model. This is done because some of the GCM EOFs are completely different from the observational EOF. By projecting GCM anomalies onto the observational EOF, we keep the spatial pattern consistent throughout all models, thus giving us a way to compare the temporal disagreements between models.

As described in Mantua and Hare (2002), the PDO is the leading PC from an EOF analysis of North Pacific SST anomalies. To calculate the PDO index, SST anomalies poleward from 20°N are obtained by subtracting the long-term (1900-1993) mean from monthly observations, considering data from November to March only. The global mean anomaly is further subtracted, to remove the effects of "global warming". In our analysis the global mean was not removed. The PDO behaves much like the El Niño-

Southern Oscillation but on the time scale of 20-30 years, as opposed to 16-18 months (Mantua and Hare, 2002). Similar to the PDO, the North Pacific Gyre Oscillation (NPGO) corresponds to the second leading PC of Sea Surface Height anomalies (SSHa). The NPGO closely tracks the second PC of SSTs which is referred to as the Victoria Mode (Di Lorenzo et al., 2008).

Figure 2.2 and 2.3 show the first and second EOF coefficients for some ensemble members. We note that, while the overall trend of coefficients corresponding to different climate models is similar, there are substantial variabilities between them. For the second half of the 21st century we observe a clear quasi-cyclical pattern for many of the climate model simulations. This is an indication that the simulations are out of phase with the 20th century observational climatology used to produce the anomalies. Monthly decadal series display noticeable jumps, due to the discontinuity that exists, for some decades, between the average SST for January and the average December SST for the previous decade.

2.3 Models to blend GCM and observational time series

As mentioned in the introduction, our goal is to blend the information from the different GCM simulations using the observations as a reference. To this end we build a hierarchical model that uses the EOF coefficients corresponding to present day observational data to estimate the model discrepancies. These are then propagated into the future to obtain 21st century forecasts.

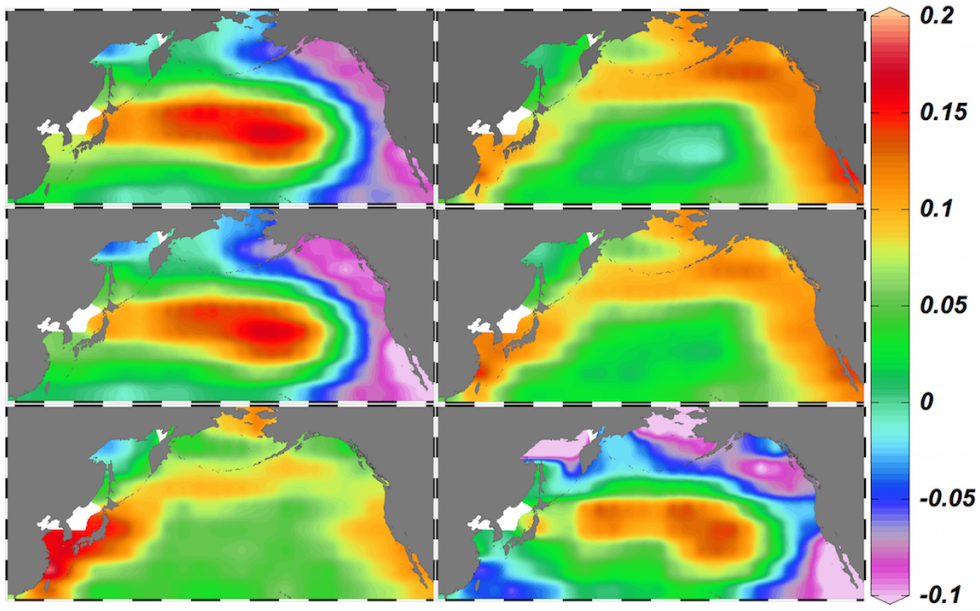

Figure 2.1: The figure shows the first (left) and second (right) observational EOF for the Monthly (top), Quarterly (middle), and Decadal Monthly (bottom) temporal scales.

To perform a comparison between models we use three methods including the Deviance Information Criterion (DIC; Spiegelhalter et al., 2002), Chi-Squared test (West and Harrison, 1997), and a Posterior Predictive P-value technique (PPP; De La Horra and Rodríguez-Bernal, 1999). Using these three tests we toggle different values of the discount factors as well as the seasonal components. We select the values for the discount factors for a specific EOF and time index based on the optimization of these three goodness of fit methods and a visual comparison.

2.3.1 Models for Monthly and Quarterly Averages

Consider the observational l-th EOF coefficient series $\psi_l(t)$. For simplicity we drop the index l and denote this as ψ_t. We assume that ψ_t follows an underlying level

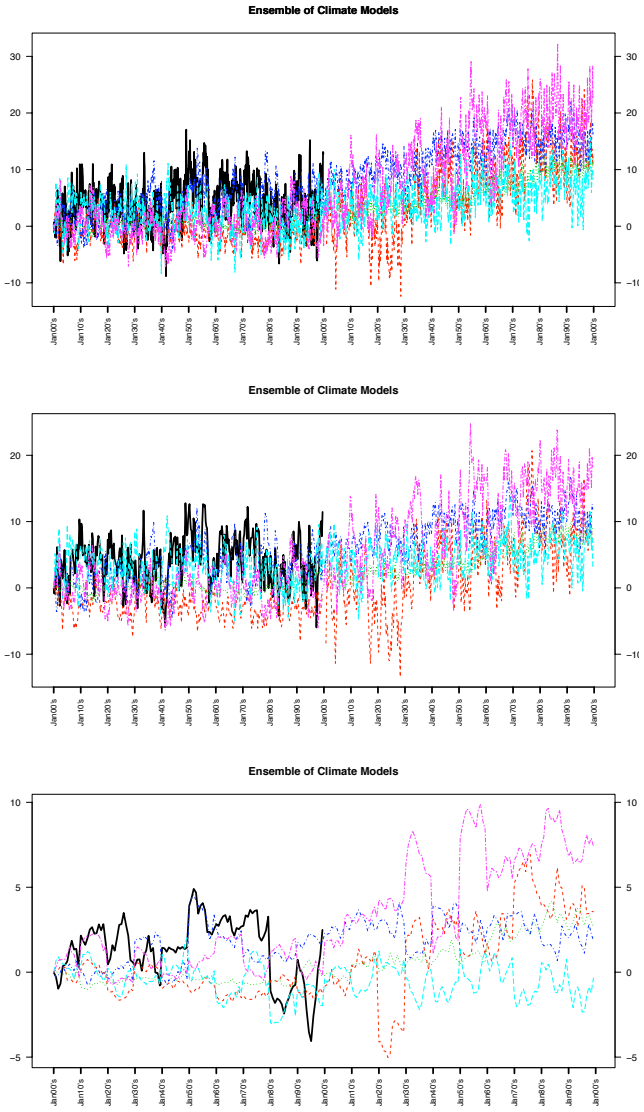

Figure 2.2: Monthly EOF1 coefficient (top), Quarterly EOF1 coefficient (middle), Decadal Monthly EOF2 coefficient (bottom) . SSTa (Solid Black) with the following model projections: GFDL-CM2.1, GISS-ER, CCSM3, NCAR-PCM, HadCM3.

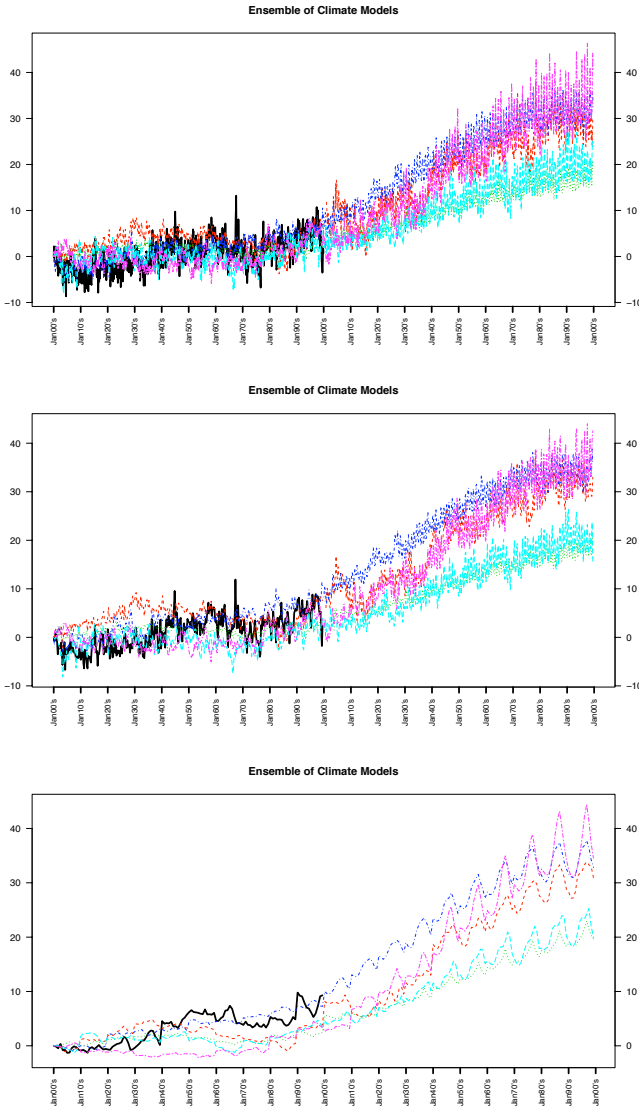

Figure 2.3: Monthly EOF2 coefficient (top), Quarterly EOF2 coefficient (middle), Decadal Monthly EOF1 coefficient (bottom) . SSTa (Solid Black) with the following model projections: GFDL-CM2.1, GISS-ER, CCSM3, NCAR-PCM, HadCM3.

θ_t:

$$\psi_t = \theta_t + \nu_t^0, \tag{2.1}$$

for some Gaussian errors ν_t^0 uncorrelated in time. Additionally, the j-th model l-th EOF coefficient $\varphi_{l,j}(t)$ has a bias δ_t^j, with respect to θ_t. Again, we drop the index l and simplify the notation to φ_t^j. Thus

$$\varphi_t^j = \theta_t + \delta_t^j + \alpha_{1,t}^1 + \alpha_{1,t}^2 + \alpha_{1,t}^3 + \nu_t^j, \;\; j = 1, \ldots, 17, \tag{2.2}$$

where $\alpha_{i,t}^1$ corresponds to a time varying seasonal component having annual ($i = 1$), semestral ($i = 2$) and quarterly ($i = 4$) periods. The Gaussian errors ν_t^j are uncorrelated in time. While we are using anomalies which should eliminate the seasonality, the climate models create an apparent artificial oscillation, which, if not accounted for, will affect the models effectiveness. To complete the model we specify the evolution of the parameters as

$$\begin{array}{rcl} \theta_t &=& \theta_{t-1} + \beta_{t-1} + \omega_t^\theta \\ \beta_t &=& \beta_{t-1} + \omega_t^\beta \\ \delta_t^j &=& \delta_{t-1}^j + \omega_t^{\delta^j} \end{array} \quad , \quad \begin{pmatrix} \alpha_{1,t}^i \\ \alpha_{2,t}^i \end{pmatrix} = \begin{pmatrix} \cos(2\pi\lambda_i) & \sin(2\pi\lambda_i) \\ -\sin(2\pi\lambda_i) & \cos(2\pi\lambda_i) \end{pmatrix} \begin{pmatrix} \alpha_{1,t-1}^i \\ \alpha_{2,t-1}^i \end{pmatrix}, \tag{2.3}$$

for $i = 1, 2, 4$ and $\lambda_i = i/12$. To specify the distribution of the error terms, denoted as $\boldsymbol{\nu}_t = (\nu_t^1, \ldots, \nu_t^{17})'$, we assume that $(\nu_t^0, \boldsymbol{\nu}_t')' \sim N_{18}(\mathbf{0}, \Sigma)$, for $t = 1, \ldots, \frac{T}{2}$, where $N_J(\cdot, \cdot)$ denotes a J-dimensional normal distribution. For the 21st century there are no observations and so no observational error term. We assume that $\boldsymbol{\nu}_t \sim N_{17}(\mathbf{0}, \Sigma_2)$, $\Sigma \sim W^{-1}(r_\Sigma, S_\Sigma)$, $\Sigma_2|\Sigma \sim W^{-1}(r_{\Sigma_2}, S_{\Sigma_2})$, where $W^{-1}(r, S)$ denotes an inverse-Wishart

distribution with r degrees of freedom and scale matrix S. Conditioning the distribution of Σ_2 on Σ establishes a link between the observational variability in the 20th and the 21st centuries. By partitioning the covariance matrix Σ as,

$$\Sigma = \begin{pmatrix} \Sigma_{1,1} & \Sigma_{1,2:J} \\ \Sigma_{2:J,1} & \Sigma_{2:J,2:J} \end{pmatrix}$$

we can also assume that $S_{\Sigma_2} = (r_{\Sigma_2} - 17)\Sigma_{2:J,2:J}$, where $\Sigma_{2:J,2:J}$ is the partition of the Σ minus the first row and column. This implies that $E(\Sigma_2|\Sigma) = \Sigma_{2:J,2:J}$. A large value of r_{Σ_2} produces a distribution with small variability around the mean value $\Sigma_{2:J,2:J}$. Let $\mathbf{\Theta}_t = (\theta_t, \beta_t, \alpha^1_{1,t}, \ldots, \alpha^3_{2,t}, \delta^1_t, \ldots, \delta^{17})'$. Then, conditional on (Σ, Σ_2), we can write Equations (2.1) – (2.3) as a Dynamic Linear Model (DLM; see, for example West and Harrison, 1997) with $\mathbf{\Theta}$ as the state-space parameters. Denote as W_t the variance of the evolution equation of $\mathbf{\Theta}$. For the 20th century we model W_t using discount factors (West and Harrison, 1997, Chapter 6). We use three blocks, one for θ_t, one for β_t and one for the seasonal components. Each block has a different discount factor $d_i, i = 1, 2, 3$. We set $(d_1, d_2, d_3) = (0.70, 0.95, 0.999)$ and $(d_1, d_2, d_3) = (0.70, 0.95, 0.98)$ for the first and second EOF respectively. For the 21st century we do not discount and we fix W_t to its last values in the 20th century. This is to reflect the loss of observational data.

The Q data set is modeled similarly to the M data set, with two modifications: i) we only include two seasonal components, the annual and the semestral; ii) we place block discount factors on W_t, such that $(d_1, d_2, d_3) = (0.70, 0.95, 0.98)$ and $(d_1, d_2, d_3) = (0.70, 0.80, 0.999)$ for the first and second EOF, respectively.

2.3.2 Model for Decadal Monthly Averages

Similar to the previous model, we consider the observational EOF coefficient series ψ_t. We assume that ψ_t follows an underlying level θ_t and decadal jump λ_d. The decadal jump parameter corrects for the time discontinuity between decades, an artificial feature caused by the level of averaging. Thus

$$\psi_t = \theta_t + \lambda_d + \nu_t^0 \tag{2.4}$$

for some Gaussian errors ν_t^0 that have intra-decadal correlation. Additionally the j-th model EOF has a bias δ_t^j with respect to θ_t and decadal jump bias λ_d^j with respect to λ_d. We denote time index $t = 12(d-1) + m = d, m$, where $d = 1, \ldots, \hat{D} = 20$ are decades, $m = 1, \ldots, M = 12$ are the months, so $t = 1, \ldots, T = 240$. Thus,

$$\varphi_t^j = \theta_t + \delta_t^j + \lambda_d + \lambda_d^j + \alpha_{1,t}^1 + \alpha_{1,t}^2 + \alpha_{1,t}^3 + \nu_t^j, \;\; j = 1, \ldots, 17 \tag{2.5}$$

where $\alpha_{i,t}^1$ has time varying components as in model (2.3). ν_t^j are Gaussian errors with time correlation extending only within a time span of a decade. We describe the evolution of the parameters as

$$\begin{aligned} \theta_t &= \theta_{t-1} + \beta_{t-1} + \omega_t^\theta \\ \beta_t &= \beta_{t-1} + \omega_t^\beta \\ \delta_t &= \delta_{t-1} + \omega_t^{\delta^j} \end{aligned} \quad , \quad \begin{pmatrix} \alpha_{1,t}^i \\ \alpha_{2,t}^i \end{pmatrix} = \begin{pmatrix} \cos(2\pi\lambda_i) & \sin(2\pi\lambda_i) \\ -\sin(2\pi\lambda_i) & \cos(2\pi\lambda_i) \end{pmatrix} \begin{pmatrix} \alpha_{1,t-1}^i \\ \alpha_{2,t-1}^i \end{pmatrix} \tag{2.6}$$

for $i = 1, 2, 4$, $\lambda_i = \frac{i}{12}$. To specify the distributions of the error terms we denote $\boldsymbol{\nu_t} = (\nu_t^1, \ldots, \nu_t^{17})$, $\nu_d^j = (\nu_{1,d}^j, \ldots, \nu_{12,d}^j)$ and $\boldsymbol{\nu_d} = (\nu_d^1, \ldots, \nu_d^{17})$, we then assume $(\nu_t^0, \boldsymbol{\nu_t'}) = N_{18}(\mathbf{0}, \tau^2\Sigma)$ and $(\nu_d^0, \boldsymbol{\nu_d}) = N_{18}(\mathbf{0}, \Sigma \otimes \tau^2 I_{12})$, for $t = 1, \ldots, T/2$, $d = 1, \ldots, \hat{D}/2$ where $T/2$ and $\hat{D}/2$ correspond to the end of the 20th century. For the 21st century we lose the observational error term so we assume $\boldsymbol{\nu_t'} = N_{17}(\mathbf{0}, \tau^2\Sigma_2)$ and $\boldsymbol{\nu_d} = N_{17}(\mathbf{0}, \Sigma_2 \otimes \tau^2 I_{12})$. We place prior distributions $\Sigma \sim W^{-1}(r_\Sigma, S_\Sigma)$, $\Sigma_2|\Sigma \sim W^{-1}(r_{\Sigma_2}, S_{\Sigma_2})$ where as stated in the M time index model, S_{Σ_2} is centered at the partition of Σ. Let $\Theta_t = (\theta_t, \beta_t, \alpha_{1,t}^1, \ldots, \alpha_{2,t}^3, \delta_t^1, \ldots, \delta^{17})'$ and $\Lambda_d = (\lambda_d, \lambda_d^1, \ldots, \lambda_d^{17})'$. Conditional on $(\Lambda, \Sigma, \Sigma_2, \tau^2)$ we can write Equations (2.4) – (2.6) as a DLM, with state-space parameter Θ. We place block discount factors on the evolution equation similar to the M model, such that $(d_1, d_2, d_3) = (0.50, 0.98, 0.999)$ and $(d_1, d_2, d_3) = (0.50, 0.90, 0.999)$ for the first and second EOF respectively. For the 21st century we fix W_t to its last values of the 20th century to reflect on the loss of observational data. We place prior distributions for $\Lambda_d \sim N(m_{1,d}, C_{1,d})$, $\tau^2 \sim \Gamma^{-1}(r_\tau, S_\tau)$, $m_{1,d} \propto 1$, and $C_{1,d} \sim W^{-1}(220, 50I)$. To explore the posterior distributions of the models above we use the Gibbs sampler, a special case of Markov Chain Monte Carlo methods.

2.4 Results

Figure 2.4 shows the first observational EOF and the projected underlying process for the ensemble of climate model projections for the three time indexes. M and Q behave very similarly, in that they capture the same general behavior in the 20th

century and have an increase in the 21st century, which ranges from 10 to 30 units and from 5 to 20 units, respectively. The decadal jump term in D corrects for the artificial jumps, and has a less pronounced increase into the 21st century, ranging from 0 to 5 units. The EOF scale is not a temperature scale, but once EOFs are projected onto their spatial counterparts, the outcome is in the original temperature units.

Figure 2.5 (top) shows the level process and the jump process, together with 95% probability intervals. We can see that the level process increases in the 21st century and that the model is able to describe significant jumps. The model discrepancy terms for time index D in Figure 2.6 show how the coefficients for some specific GCMs vary from the common level. We observe that two of the projections, NCAR-PCM and HadCM3, over and under estimate at a range of (-3.5,5). We can also see that jumps vary in intensity between models and become more apparent in the 21st century. In figure 2.6 we can see the model discrepancies for the three different time indexes for the first EOF. The discrepancies for M and Q are very similar in that the model bias follows the same structure, with more noise for M. For D, the weights are similar in the sense that NCAR-PCM and HadCM3 are given large positive and negative weights, suggesting that they are over and under approximating, while the remaining 3 are given approximately zero weight. If we look at the observation variance parameters, $\Sigma_{1,1}$ for the 3 time indexes we see a decrease, from 9.5 to 3.3 to 1.16, when comparing between M, Q, and D. This is to be expected since we smooth the noise for the monthly data by a third then a tenth for Q and D respectively. The variance parameters do not change for the climate model projections when comparing them over the 3 indexes.

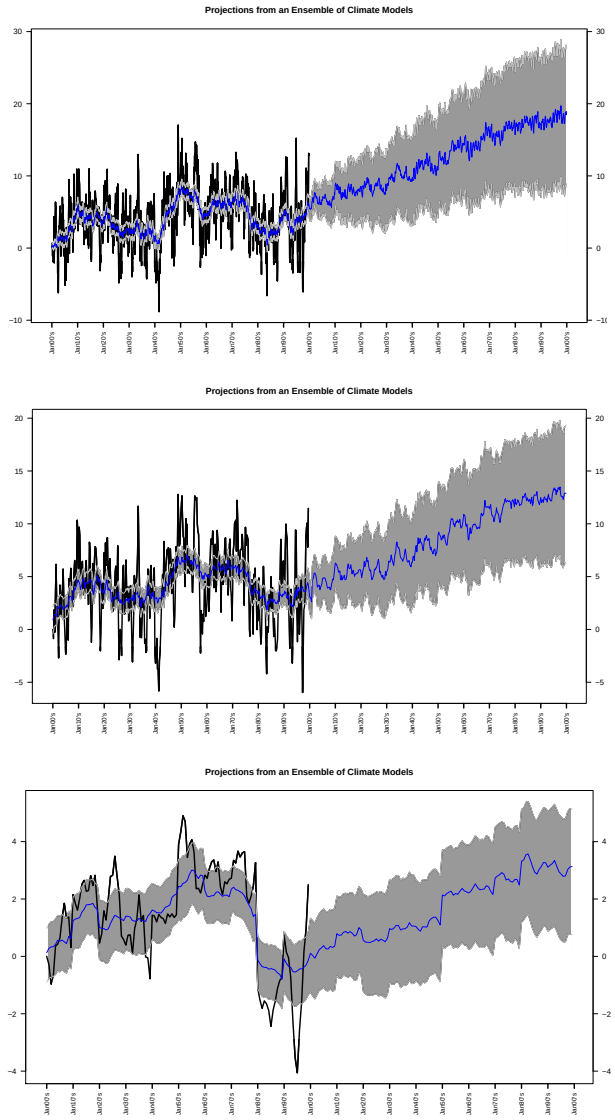

Figure 2.4: First EOF Projections from an Ensemble of Climate Models from top to bottom: Monthly, Quarterly, and Monthly Decadal.

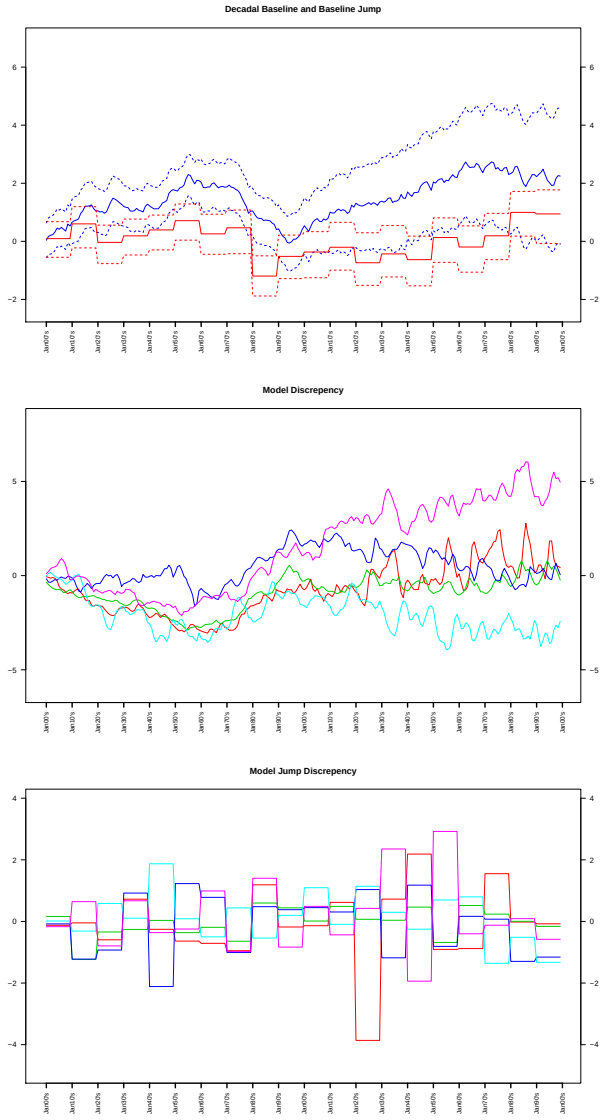

Figure 2.5: Underlying process and jump (top) with 95% credible intervals for the first EOF of time index D. Model bias terms (middle) and Model Jump bias terms (bottom) for the projections GFDL-CM2.1, GISS-ER, CCSM3, NCAR-PCM, HadCM3.

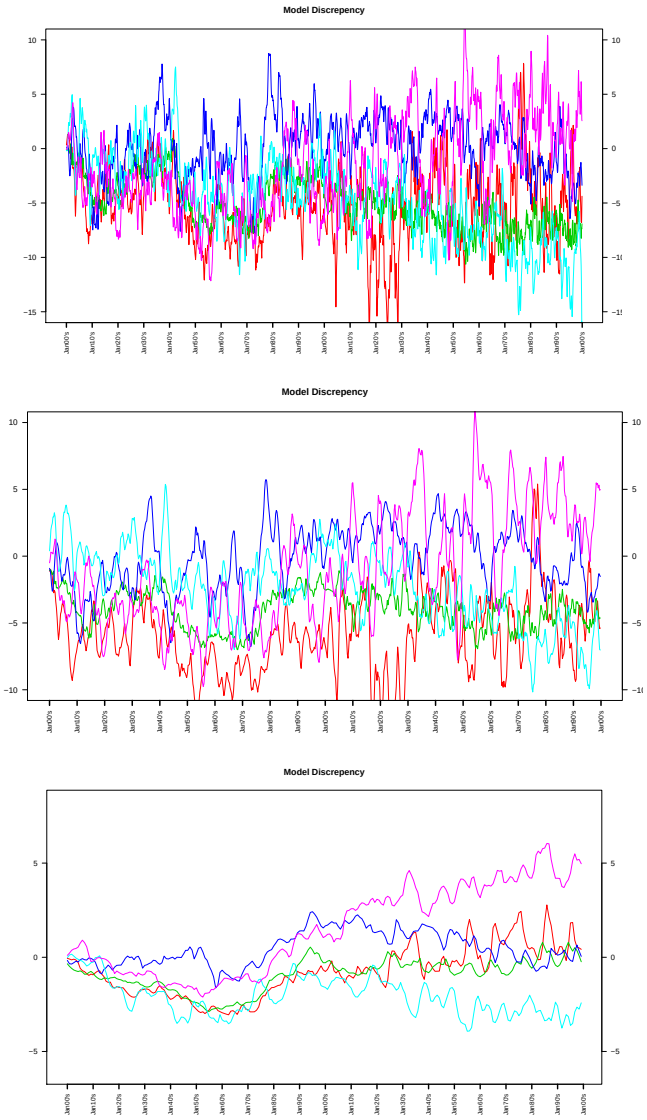

Figure 2.6: Model discrepancy terms for M (top), Q (middle) and D (bottom) for the projections GFDL-CM2.1, GISS-ER, CCSM3, NCAR-PCM, HadCM3.

From the ensemble projections, ψ_l, obtained for the EOFs, we perform a reconstruction of the sea surface temperature field. We define the reconstruction as

$$\tilde{X}^r(\mathbf{s},t) = \sum_{l=1}^{r} u_l(\mathbf{s})\psi_l(t) \tag{2.7}$$

where the climatology is then added to $\tilde{X}^r(\mathbf{s},t)$ to produce SST. We use this reconstruction for D and compare the results to the approximated observational SST using $r = 10$, which gives us most of the explained variability. The methods stated above were used in modeling the other eight indexes.

In Figure 2.7 we can see the differences between the present observational SST (January 1990's) and the reconstructed SST for the month of January, in the 1960s, 1990s and 2020s, for the model of time index D. Our model suggests that, compared to the present conditions, the past was cooler and the future will be warmer with differences ranging from (-0.45K,-0.10K) for the past and (0.12K,0.44K) for the future for the first and third quantile. The reconstruction for the current SST (middle) provides a visual examination of how well the model is performing, with differences ranging from (-0.24K,0.04K) . The difference is not large, suggesting that our model can describe the present day SST.

Figure 2.8 shows the time series of observational SST for the 20th century, as well as the SST reconstruction and GCM for the 20th and 21st century. From this figure we can see reconstructed SST closely follow observational SST, in both slope and amplitude, much closer than any single GCM. In the 21st century, it is important to note that the reconstruction is not just the average of the GCMs. While the amplitude

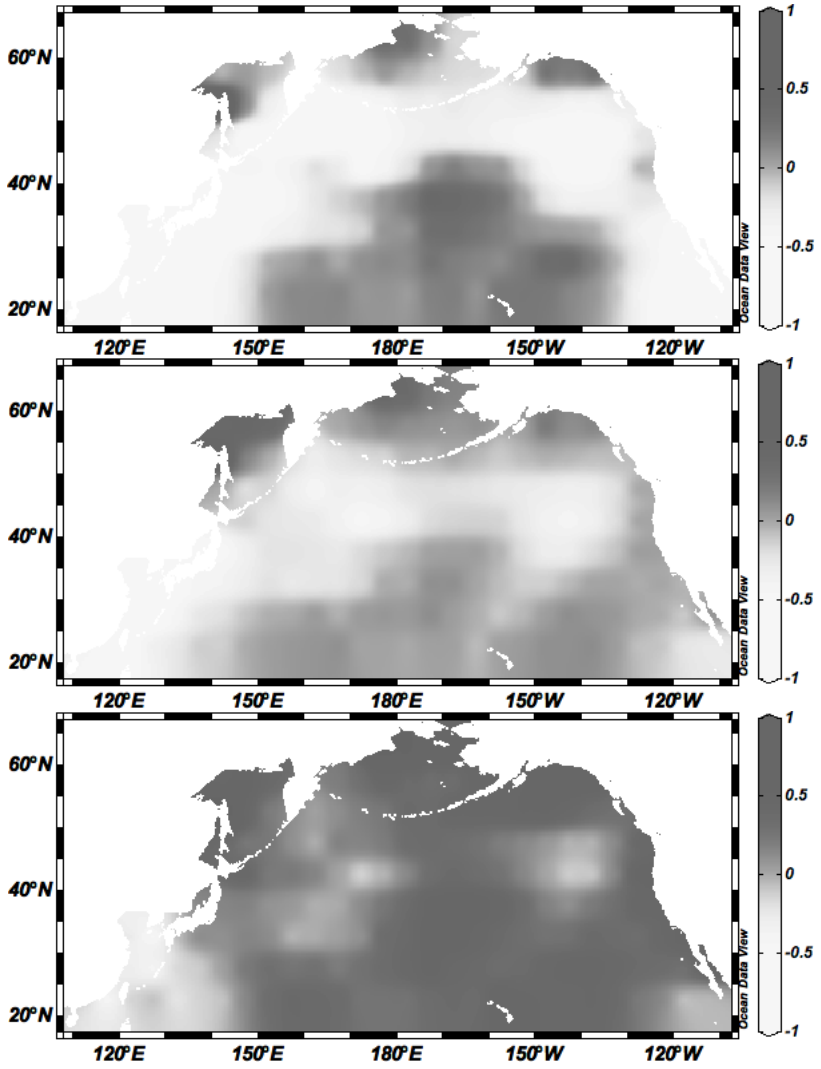

Figure 2.7: Differences between Monthly Decadal SST for January of 1990's and SST Reconstruction using ten EOFs for the decades of 1960s,1990s,2020s (top to bottom).

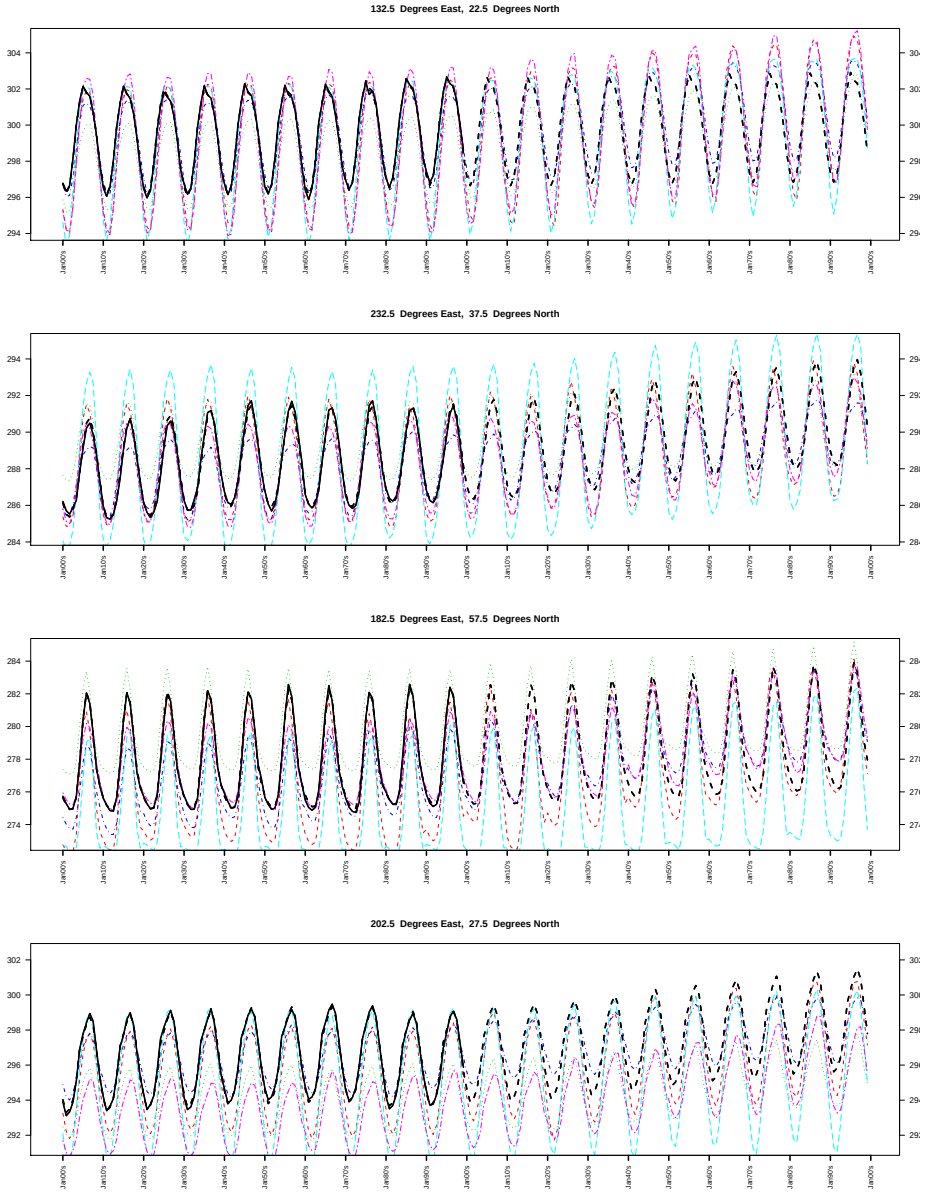

Figure 2.8: Monthly Decadal Observational SST (Solid Black), Global Climate Models (GFDL-CM2.1, GISS-ER, CCSM3, NCAR-PCM, HadCM3.), and SST Reconstruction (Doted Black) for station site (132.5,22.5), (232.5,37.5), (182.5,57.5), (202.5,27.5) where (Degrees East,Degrees North) (Top to Bottom).

is smaller in some cases, the time series do have a similar increasing slope. In Figure 2.9 we present the spatial reconstruction for monthly decadal SST, for the month of July in the years 2020, 2040, and 2060. We can see noticeable temperature increase in the far north and along the southeast coastline. This reconstruction suggests that the ocean is getting warmer in certain parts faster than others.

2.5 Conclusions

We have created a methodology to obtain a unified forecast of oceanic indexes using historical records and GCM simulations. In response to the desideratum of high frequency and high spatial resolution in the projections, we have developed a model that focuses on the main modes of spatial variability. The model blends the coefficients of those modes, using the different sources of information and accounting for seasonal cycles. The resulting index predictions have an interest of their own, as they can be associated with global oceanic dynamics and specific environmental changes. They can also be used to reconstruct the spatio-temporal fields and obtain high spatial resolution SST predictions. The problem of comparing month to month observational data and GCM simulations is tackled by assuming a dynamic model that produces smoothed estimates of the discrepancies.

Predictions show a degeneracy in the GCMs, around the middle of the 21st century. This could be attributed to the strong assumption in our model that the spatial EOF pattern does not change over time, and thus, the GCMs become out of synchrony

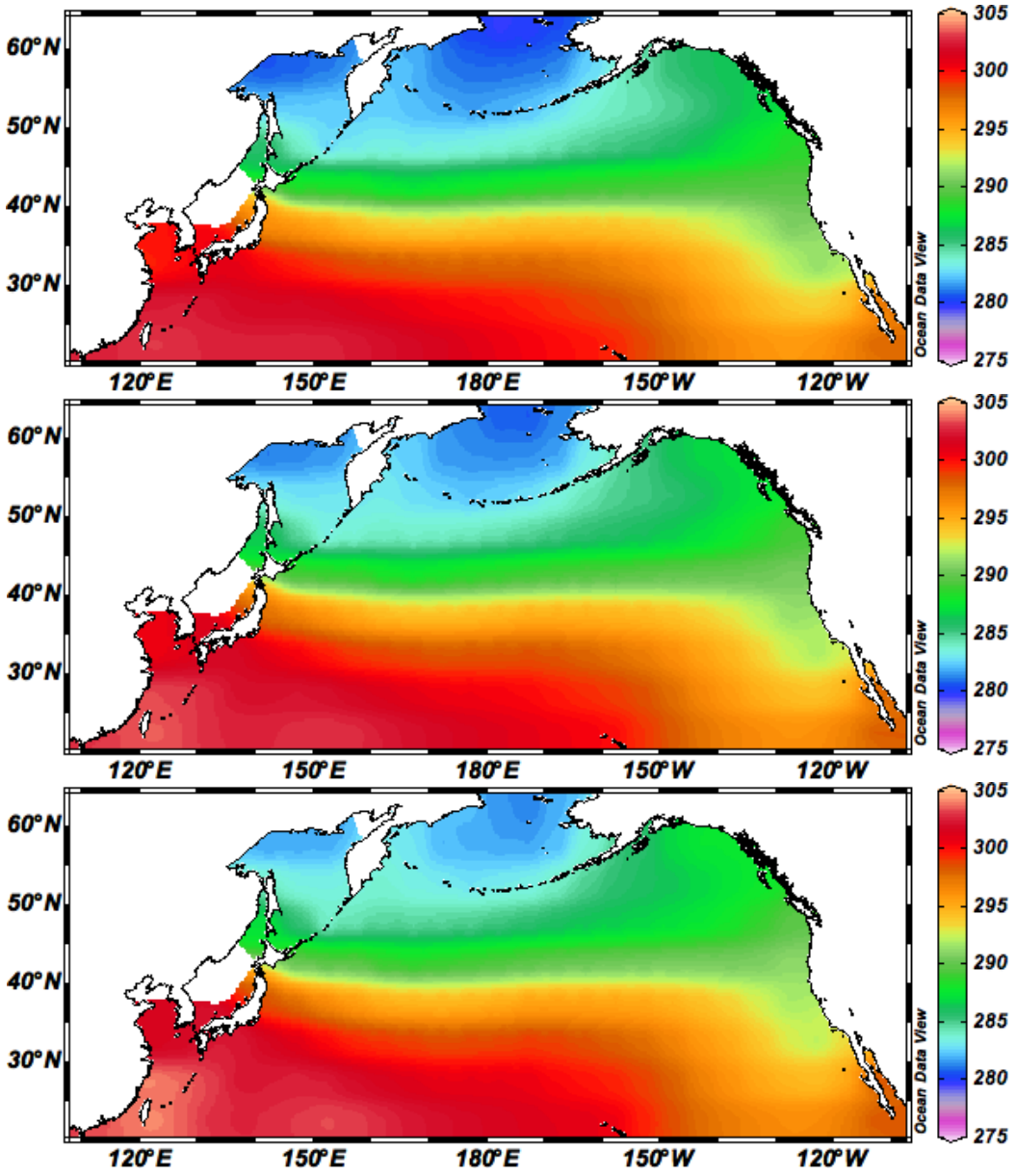

Figure 2.9: Monthly Decadal SST future reconstruction for July in the year 2020 (top), 2040 (middle), 2060 (bottom).

with the 20th century climatology. Alternatively, it could just be an artifact of the GCMs themselves, as they simulate climate into the future. Another possibility is that the climate models are predicting a dynamical shift in seasonal behavior, meaning winter may come earlier or later. Either way, by considering a joint analysis of high frequency data, as opposed to taking each season separately, our statistical model can tackle this issue. Our model is flexible enough to deal with time-varying cycles. The changing seasonality, though, remains an important GCM feature that needs to be considered carefully, and that highlights the difficulties of predicting the climate 100 years into the future.

We performed an analysis that considered three different levels of temporal aggregation. The results show that, at least for the first two EOF coefficients, all three analyses capture similar structural features. This is likely due to the smoothing induced by the dynamical model. We see, in particular, that the results from M are a noisier version of those from Q. We do observe some relevant differences when we compare D to M and Q. The level of increase in the 21st century for D is much smaller than for M and Q, and has less variability. Computationally, D is, of course, much faster to deal with that M, as it involves an order of magnitude less data. D corresponds to the level of time aggregation used most frequently in the literature. By taking decadal averages we create an artificial feature not present in the original data set. This feature is a jump caused by the time discontinuity created at the turn of each decade, which we account for in our modeling structure. The visible jumps in the 21st century projections suggest that the model does not disregard this feature after the 20th

century, but incorporates the jumps in the future predictions. The traditional approach to combining multi-model GCM ensembles considers decadal seasonal data separately for each season. In comparison with this approach, our model for D has the advantage of providing continuity between decades and months. This allows a detailed description of the dynamics of future seasonal patterns. We have focused on representation of the spatio-temporal fields in terms of EOFs.

The methodology proposed in this chapter can be used for expansions on any set of bases, provided the same set is used for all the GCMs and the observations. The resolution of the reconstructed predictions obtained in this chapter is high relative to the large regions commonly used in the literature, but it is still too coarse for most practical ecological studies. In the chapters 3 and 4 we explain how to obtain projections on a much finer spatial resolution using a change of support methodology.

Chapter 3

Downscaling

A spatial resolution of 5° is sufficient to observe large scale oceanic behaviors but too coarse to capture local and regional oceanic variability. Coastal behaviors such as upwelling, a wind driven phenomenon that brings nutrient rich water to the ocean surface impacting biological ecosystems such as phytoplankton sardine, and anchovy production (Ward et al., 2006), are completely smoothed over, making these patterns invisible at large resolutions. In Figure 3.1, we see how the coastal features in the coarser resolution are nonexistent and the pattern for upwelling is not distinct at larger resolutions. GCMs are essential in gathering information over large areas and time scales and until the last few decades, high-resolution datasets were not available. However, as the resolutions become finer, it has allowed us to model smaller regions with greater precision. The use of statistical downscaling is a computationally effective way to form a connection between large and small scale variability.

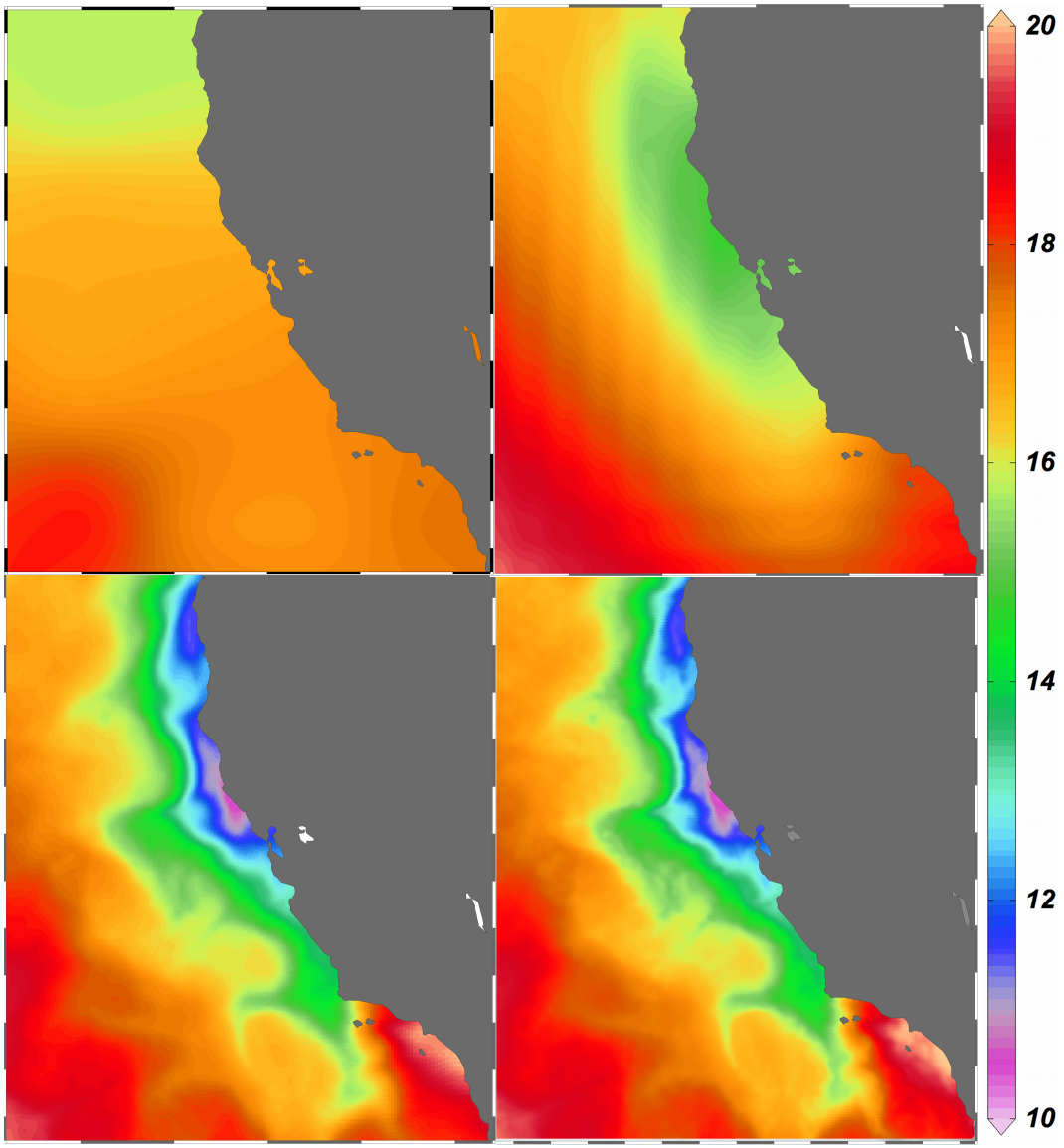

Figure 3.1: This figure shows the surface plot of California Coast for July 2003 at different spatial resolutions: (top, left to right) 5°, 1°; (bottom, left to right) 0.1°, 0.01°.

3.1 Introduction

We model spatial-temporal data by capturing temporal evolutions while smoothing the spatial surface. Modeling the spatial behavior through a covariance matrix using a covariogram as a function of Euclidean distance is the conventional way to model space-time data. This type of modeling is known as kriging models (for further explanation see (Cressie and Wikle, 2011)). Kriging models produce inference that is good for spatial prediction, but are impractical when dealing with a large number of spatial locations. Creating and decomposing a covariance matrix of ten thousand locations results in storing over fifty million unique numerical entries making these models computationally infeasible.

An alternative approach for space-time modeling is to use a more flexible model by specifying the covariance matrix though a smoothing kernel. A Gaussian process is developed by convoluting a white noise process over a fixed grid using a smoothing kernel; this is know as a discrete process convolution (DPC). In this chapter, we will focus on a DPC to model space-time data. In this next section, we use a discrete process convolution as a statistical downscaling technique to establish a link between the high resolution temperature output and the large scale variation of the GCM ensembles.

3.1.1 Process Convolution

If we consider the temperatures as a Gaussian process $y_t(s)$, over time t and space s, as the sum of process $z_t(s)$ and error $\epsilon_t(s)$ such that

$$y_t(s) = z_t(s) + \epsilon_t(s)$$
$$z_t(s) = \sum_{u=1}^{M} k(s, s_u) x_t(s_u).$$

The process $z_t(s)$ is a discrete process convolution of a Gaussian process defined by a smoothing kernel $k(s, s_u)$ which depends on $d = |s - s_u|$, the Euclidian distance between the two points. $x_t(s_u)$ is a Gaussian process that captures the temporal patterns over the much smaller grid s_u, the details of which can be found in Higdon (1998, 2002); Cressie and Wikle (2011). The smoothness of $z_t(s)$ depends on the convolution kernel $k(s, s_u)$ and conditional on process x_t, the spatial smoothness is independent over time. This also gives us the flexibility of making the covariance matrix a function of the kernel. For computational reasons we will use a Bezier kernel for the remainder of this chapter. Information about the other various types of smoothing kernels can be found in Kern (2000), however they will not be discussed in this thesis.

We define the kernel centered at the point s_u as

$$k[s - s_u; \omega] = \begin{cases} (1 - ||s - s_u||_\phi)^\omega & ||s - s_u||_{\phi_u} < 1 \\ 0 & o.w. \end{cases}$$

where $||s - s_u||_\phi = 1/\phi_u \sqrt{(x_s - x_u)^2 + (y_s - y_u)^2}$ is a simplified version of the kernel defined in Lemos and Sansó (2009) and ω is the smoothness parameter. This gives us a convolution kernel that is isotropic with circular support around s_u, with radius ϕ_u,

and allows for local spatial control of the process. For certain values of ω the kernel can give us Gaussian style smoothness with tapering occurring outside of the radius of ϕ_u allowing the kernel to be sparse. Having a sparse kernel allows us to lessen the computational load since most of kernel matrix K will be zero's.

PC models keep the appeal of kriging model, are similar to using an EOF representation of the spatio-temporal field (Storch and Cambridge, 1999), and the number of locations of the underlying process is not dependent on the location of the observations. This modeling approach allows us to reduce the dimension of our spatial process while still producing similar results to that of a covariogram structured model. We can extend this model by adding a time series structure using an auto-regressive component to capture the temporal evolution. In this chapter we model the temporal evolution via DLM techniques similar to the work in Huerta et al. (2004). Downscaling models have also been used recently when comparing observational data to numerical simulation data: Berrocal et al. (2012) use Regional Climate Models as covariates to predict SST on the seasonal scale; Berrocal et al. (2010) use a spatio-temporal model to Ozone data; and Sansó and Guenni (2004) model rainfall data using deterministic simulations.

3.2 Spatio-Temporal Model

Let $Z_t(\mathbf{s})$ denote the 0.1° SST at time t over the spatial grid $\mathbf{s} = (s_1, \ldots, s_I)'$. We construct the likelihood such that

$$Z_t(\mathbf{s}) = \mu_t(\mathbf{s}) + \sum_{u=1}^{U} k[\mathbf{s} - \mathbf{s_u}; \phi_u] \left(\Theta_t(\mathbf{s_u}) + \eta_t(\mathbf{s_u})\right) + \epsilon_t \tag{3.1}$$

where $\epsilon_t \sim N(0, \tau^2 I_I)$, $\mu_t(\mathbf{s})$ is the observational climatology at time t, and $k[\mathbf{s} - \mathbf{s_u}; \phi_u]$ is a Bezier kernel centered around $\mathbf{s_u}$ with range parameter ϕ_u. $\Theta_t(\mathbf{s_u})$ is the 1° resolution SST anomalies over the North Pacific obtained using the model blending techniques in Chapter 2. The analysis of $\Theta_t(\mathbf{s_u})$ are presented in a later section. The vector $\eta_t(\mathbf{s_u})$ is the residual signal not captured by the large scale signal Θ_t. For completion, the evolution equations for η_t are written as,

$$\eta_t(\mathbf{s_u}) = G(\rho)\eta_{t-1}(\mathbf{s_u}) + w_t(\mathbf{s_u}) \tag{3.2}$$

where $w_t \sim N(0, W_t)$ and $G(\rho) = \rho I_u$. The auto-regressive coefficient ρ is given a prior $U(0, 1)$, where $U(\cdot, \cdot)$ denotes the uniform distribution. We parameterize the equations as follows,

$$\begin{aligned} Y_t(\mathbf{s}) &= Z_t(\mathbf{s}) - \mu_t(\mathbf{s}) \\ \lambda_t(\mathbf{s_u}) &= \Theta_t(\mathbf{s_u}) + \eta_t(\mathbf{s_u}) \\ L_{t-1}(\mathbf{s_u}) &= \Theta_t(\mathbf{s_u}) - G(\rho)\Theta_{t-1}(\mathbf{s_u}). \end{aligned}$$

For simplicity we will drop the spatial notation $(\mathbf{s}, \mathbf{s_u})$ when describing the likelihood. This makes the state-space equations

$$\begin{aligned} Y_t &= K(\phi)\lambda_t + \epsilon_t, & \epsilon_t \sim N(0, \tau^2 I_I) \\ \lambda_t &= L_{t-1} + G(\rho)\lambda_{t-1} + \omega_t, & \omega_t \sim N(0, W_t). \end{aligned} \tag{3.3}$$

where the observational covariance has an uninformative prior $, p(\tau^2) \propto, (\tau^2)^{-1}$ and the evolution matrix W_t will be modeled using a discount factor approach. Let D_t denote all the information up to time t and for consistency we let $F_t^{'} = K(\phi)$. Using the FFBS algorithm decribed in A.2.1 we can write the one-step ahead forecast and posterior distributions as,

$$
\begin{aligned}
&p(\lambda_{t-1}|D_{t-1}) \sim N(m_{t-1}, C_{t-1}) && , R_t = G_t C_{t-1} G_t^{'} + W_t \\
&p(\lambda_t|D_{t-1}) \sim N(L_{t-1} + G(\rho)m_{t-1}, R_t) && , Q_t = F_t^{'} R_t F_t + V_t \\
&p(Y_t|D_{t-1}) \sim N(F_t^{'}(L_{t-1} + G(\rho)m_t), Q_t) && , C_t = R_t - A_t Q_t A_t^{'} \\
&p(\lambda_t|D_t) \sim N(m_t, C_t) && , m_t = C_t \left(F_t V_t^{-1} Y_t + R_t^{-1}(L_{t-1} + G(\rho)m_t)\right)
\end{aligned}
$$

The discount factor approach described in (West and Harrison, 1997) defines $W_t = \frac{1}{\delta} G_t C_{t-1} G_t^{'} - G_t C_{t-1} G_t^{'}$, where the value of $\delta \in (0, 1)$. We will use the discounting approach described in (Lemos, 2010) where we select W_t such that the discounting occurs on the prior variance $V[\lambda_{t-1}|D_{t-1}] = C_{t-1}$ making $W_t = \frac{1}{\delta} C_{t-1} - G_t C_{t-1} G_t^{'}$, which seems a more natural approach. Using this type of discounting technique simplifies the FFBS equation parameters to,

$$
\begin{aligned}
R_t &= \delta^{-1} C_{t-1} \\
Q_t &= \delta^{-1} F^{'} C_{t-1} F + V \\
C_t^{-1} &= \sum_{k=0}^{t-1} \delta^k F V^{-1} F^{'} \\
m_t &= C_t F V^{-1} Y_t + \delta L_{t-1} + \delta G m_{t-1}.
\end{aligned}
\tag{3.4}
$$

Since the discount value $0 < \delta < 1$, the summation of C_t^{-1} is a geometric series and quickly converges when $\delta < 0.95$. In figure 3.2 we can see the time of convergence is roughly 10 time steps. In this analysis we will use a discount factor of $\delta = 0.7$ and so the approximation of $C_t^{-1} = \dfrac{1}{1-\delta} F V^{-1} F^{'}$ will be used. This approximation simplifies

the backwards sampling equations to sample λ_t to be

$$\begin{aligned} p(\lambda_t|\lambda_{t+1}, D_T) &= N(h_t, H_t) \\ H_t &= C - \delta CG'C^{-1}GC \\ h_t &= m_t + \delta CG'C^{-1}(\lambda_{t+1} - (L_t + Gm_t)). \end{aligned} \tag{3.5}$$

We sample the parameters (ϕ, ρ, τ^2) from the marginal posterior distribution

$$p(\phi, \rho, \tau^2|D_T) \propto \prod_{t=1}^{T} N(f_t, Q_t)p(\phi)p(\rho)p(\tau^2).$$

The order in which we sample the parameters in done in such a way that we sample the more sensitive parameters first. We construct a sampling hierarchical with τ being at the top, since it's a Gibb step, followed by ρ and then ϕ. The parameters ρ and ϕ are sampled in a block such that if ρ is accepted, we sample ϕ, otherwise both get rejected.

The samples of τ^2 are obtained using a Gibbs step, while an MCMC algorithm is used to explore the posterior distributions for ρ and ϕ. In the next section we ran the Metropolis-Hastings algorithm for 500000 iterations with an additional 100000 burn in. The chain was then thinned by taking every one hundredth iterations leaving us a sample of 5000. To test for convergence we analyze only the two parameters ϕ^2 and ρ. We tested two chains with initial values for ρ at $(0.5, 0.95)$ and initial values for ϕ to be at the lower boundary and upper boundary of the kernel support. Using a Gelman and Rubin convergence diagnostic (Gelman A., 1992) we obtained the potential scale reduction factor to be an average of 1.6 and 1.5 for ϕ^2 and ρ respectively. Once convergence of the three parameters have been established, we can then move into the sampling algorithm for the parameter η. The algorithm on how the posterior samples

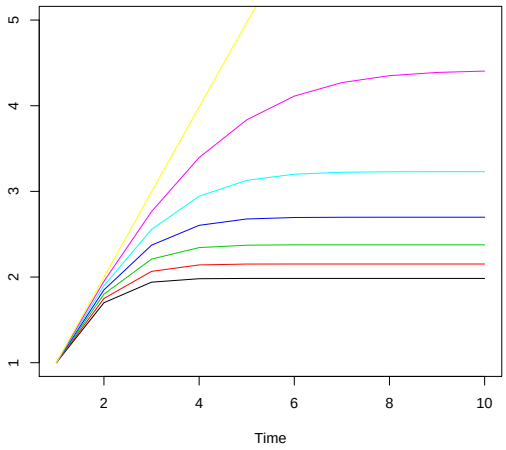

Figure 3.2: The number of time steps needed for the covariance of $p(\lambda_t|D_t)$ to converge for the different values of the discount factor δ: 0.70 (black), 0.75 (red), 0.80 (green), 0.85 (blue), 0.90 (cyan), 0.95 (purple), 0.999 (yellow).

are obtained is described in Chapter 4.

3.3 Data

We consider a monthly 0.1° resolution gridded SST for the years 2003 to 2012 in the North Pacific region 31.5°N-43.5°N, 230.5°E-243.5°E. The region encompasses the coast of California. The observational data were made available by the Group for High Resolution Sea Surface Temperature (GHRSST) (Donlon et al., 2007) and can be accessed at (podaac-ftp.jpl.nasa.gov/allData/ghrsst/). The GHRSST product used in the analysis for this chapter is L4, which uses a combinations of satellite, ship readings, and bouy's to produce a daily 0.01° gridded surface over the globe. We aggregate the data to a monthly temporal resolution and since the coastal behaviors are not

diminished at the 0.1° resolution, as pointed out in figure 3.1, we aggregate to a 0.1° spatial resolution resulting in a total of 9609 active grid cells.

The data we consider for the large scale variability is the reconstructed SST anomalies obtained using the ensemble blending analysis performed in Chapter 2. The historical SST data we use as observations stem from the UK Meteorological Office, Hadley Centre (Rayner et al., 2003). The dataset has monthly temporal resolution, 1° spatial resolution over the North Pacific region, 22.5°N-62.5°N, 112.5°E-247.5°E. This spatial resolution gives us 3926 active grid cells.

The GCMs used to perform the reconstruction analysis in section 2.3.1 were obtained from the CMIP3 multi-model database. We filtered through these GCMs by eliminating the models that did not have a spatial resolution of $\leq 1°$. We also eliminated the GCMs that did not have a high count of active grid cells along the coastal regions. This left us with three different GCMs: Bjerknes Centre for Climate Research, Norway, (BCCR-BCM2.0) 2005, National Oceanic and Atmospheric Administration/Geophysical Fluid Dynamics Laboratory, USA, (GFDL-CM2.1) 2005, and National Center for Atmospheric Research, USA, (PCM) 1998. We then aggregate the GCMs to a 1° spatial resolution. For the years 1900-1999 we use the SRES scenario 20C3M and for the years 2000-2099 we use the scenario A1B.

3.3.1 Large Scale Reconstruction

We perform an EOF analysis for the 1° observational anomalies and project the climate model anomalies onto the observational EOFs. In Figures 3.3 and 3.4 we

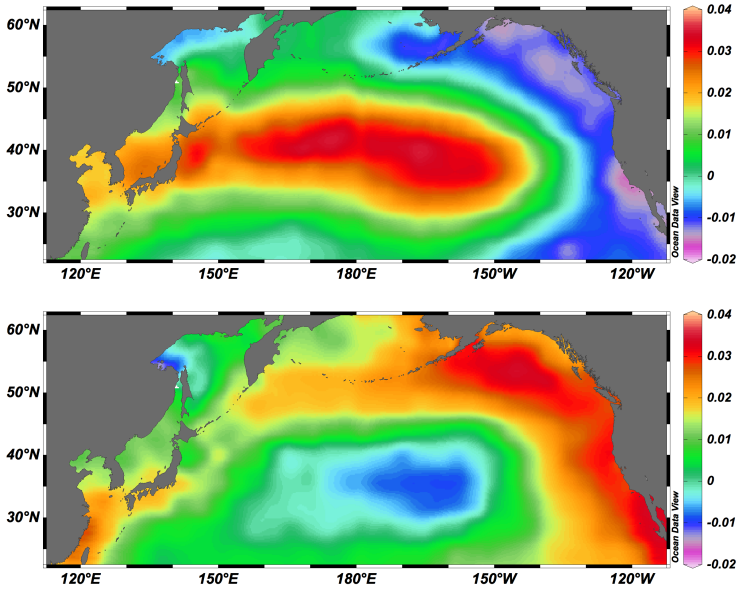

Figure 3.3: First (top) and second (bottom) observational EOF for monthly SST at a 1° resolution.

see the first and second observational EOF for the SST anomalies and corresponding EOF coefficient and model projections for the years 1900-2099. We use the equations 2.1 through 2.3 to create a unified forecast of model ensembles. The observational data for the years 2000 to 2010 were not included in performing this analysis. The eleven years of data will be used to gauge the projection accuracy of our model as well as to test the accuracy of our reconstruction.

While we are only presenting the first few EOFs in this analysis, we used a total of 80 EOFs to capture 68% of the variability. The discount factors for the first five EOFs are optimized using the same techniques described in Chapter 2. After the 5th EOF, the indexes become very similar and the spatial weights become localized. Due to this fact, we set the discount factors for the remanding 75 EOFs equal to the

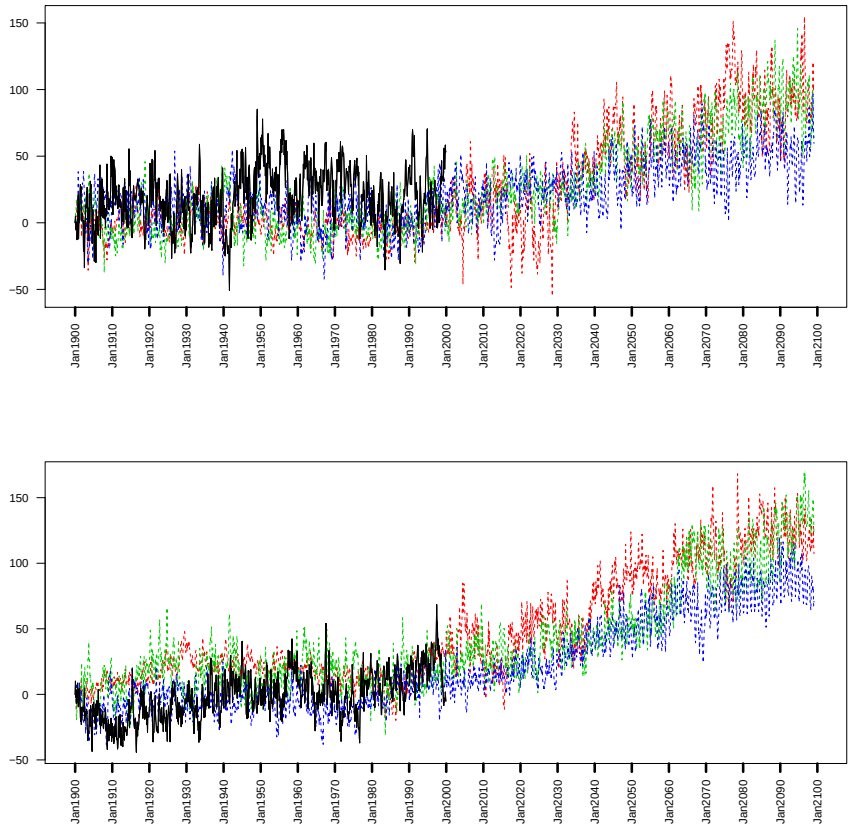

Figure 3.4: First (top) and second (bottom) observational EOF coefficient (black) and bcm2.0 (red), cm2.1 (green), and pcm (blue) model projections for monthly SST at a 1° resolution.

ones optimized in the 5th EOF. The posterior samples were obtained using Gibbs steps and the FFBS algorithm described in A.2.1. We obtained a posterior sample of 3000 iterations after thinning and burn in for each of the 80 EOFs.

3.3.2 Results

In figure 3.5 we present the results for the first four EOFs. The model does a good job in projecting the next eleven years. The ability in accurately project over these eleven years is vital since this reconstruction will be the driving force of our down-scaling projections. While we are only showing the first four EOF results, the accurate projections are constant for the first 40 EOFs. Using the reconstruction equation 2.7, we reconstruct the SST field and compare it to the observations and GCMs.

In figure 3.6 we see the fit of the posterior mean of the reconstruction in comparison to the GCMs. The plots correspond to a grid cell off the coast of San Francisco and a gris cell off the coast of Alaska for the years 1990 to 1999. These two locations illustrate the differences in slope, amplitude, and the degree of changing behavior of the SST field for different regions. We can see that reconstruction performs better in both slope and amplitude than any single GCM. In figure 3.7 we see the reconstructed projections posterior mean and 95% probability bands compared to the observations for the years 2000-2010. The amount of variability changes depending on the spatial location. When comparing our projection to the GCM mean, figure 3.8, that our model shows good ability to accurately project SST for the eleven years over just taking the GCM average.

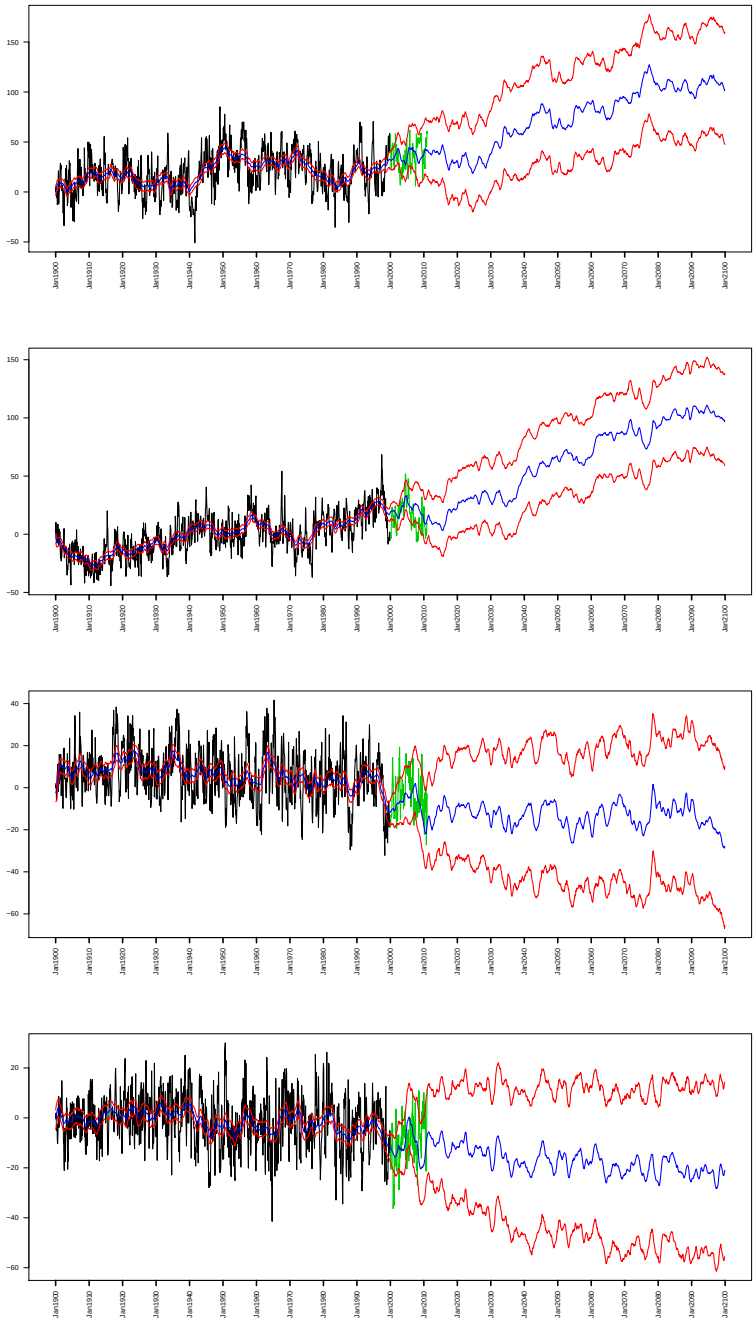

Figure 3.5: First four (top to bottom) observational EOF coefficient (black) with smooth posterior mean (blue) and 95% probability intervals (dotted red). The observations for the years 2000-2010 (green) are used to projection validation.

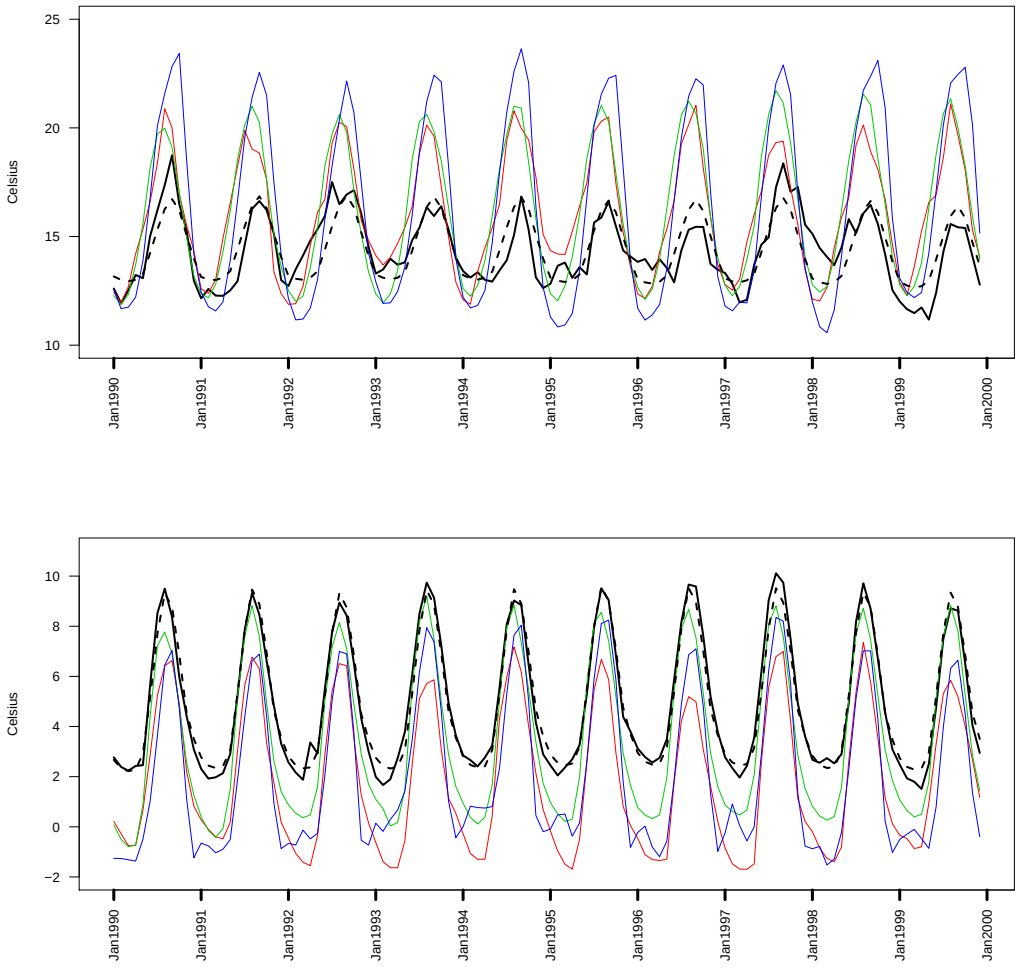

Figure 3.6: SST in degrees Celsius for two locations, [237.5,36.5] (top) and [180.5,57.5] (bottom). The observations (black), GCMs: bcm2.0 (red), cm2.1 (green), and pcm (blue), and posterior mean of the reconstruction (dotted black).

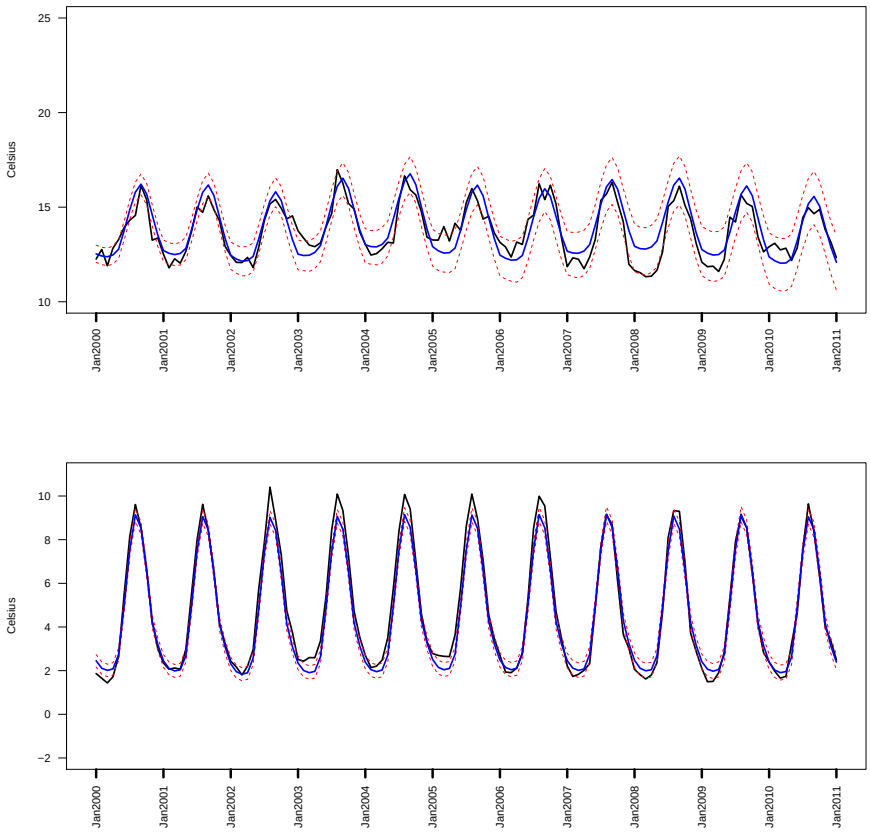

Figure 3.7: SST in degrees Celsius for two locations, [237.5,36.5] (top) is off the coast of San Francisco and [180.5,57.5] (bottom) is off the coast of Alaska. The observations (black), posterior mean projection of the reconstruction (blue) with 95% probability bands for the years 2000 to 2010.

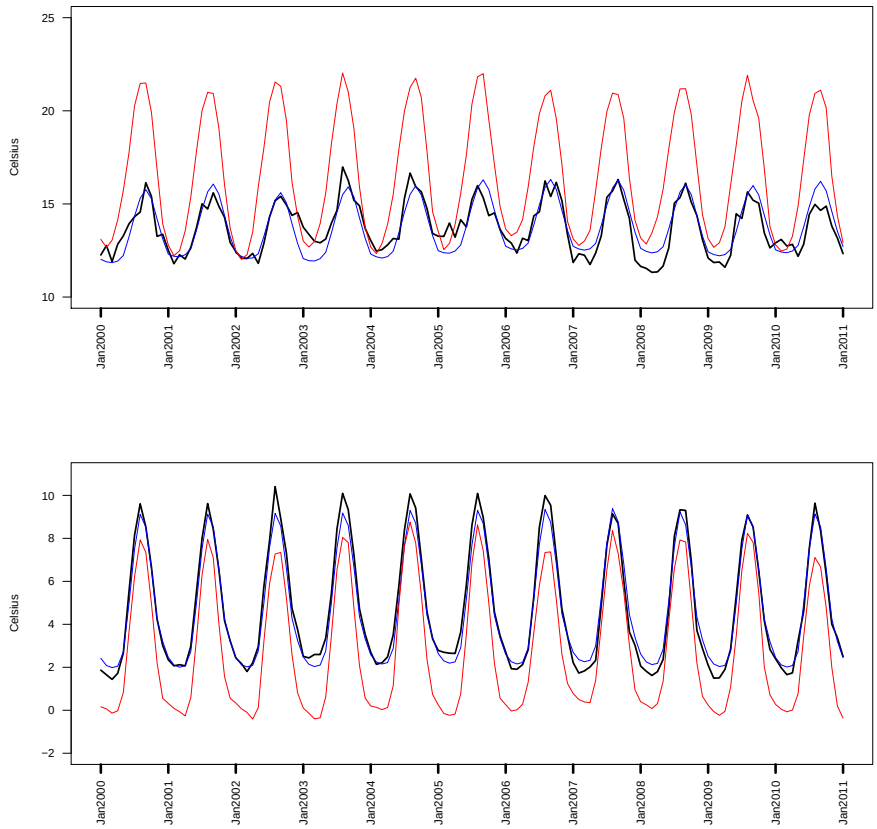

Figure 3.8: SST in degrees Celsius for two locations, [237.5,36.5] (top) is off the coast of San Francisco and [180.5,57.5] (bottom) is off the coast of Alaska. The observations (black), posterior mean projection of the reconstruction (blue), and GCM average (red) for the years 2000 to 2010.

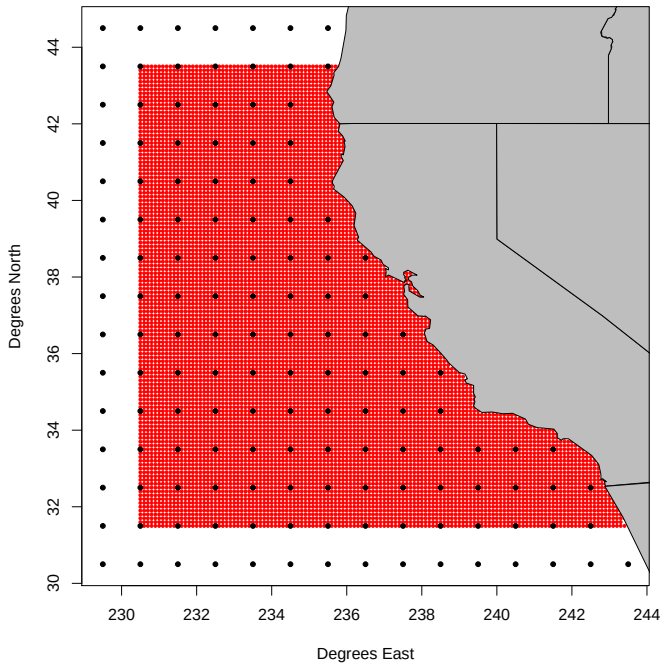

Figure 3.9: The figure shows the region 30.5°N-44.5°N, 229.5°E-243.5°E off the coast of California. The 0.1° high resolution dataset has evenly spaced grid cells in red. The 1° large scale parameter $\Theta_t(\mathbf{s_u})$ has grid cells encompassing the high resolution dataset in black.

The level of variability is small up to the year 2011. Do to this fact we will disregard this variability and take the large scale parameter $\Theta_t(\mathbf{s_u})$ in equation 3.1 to be the posterior mean of the reconstruction. The area of interest for our statistical downscaling model is the coast of California, we partition the reconstruction to the same domain of our observational high resolution data. This partition results in a total of 140 active grid cells as depicted in figure 3.9.

3.4 Results

In figure 3.12 we show the posterior sample means for the kernel range parameters ϕ. The outer boarder of the region was set equal to twice the resolution and the coastal parameters had a lower support boundary equal to the minimum distance to compass the observations. In the middle of the region we see the range parameters want to become localized with in increase in radii as we move outward to the boundaries.

In figure 3.13 we present the spatial field for the projected SST for the months of January and July for the years 2013 to 2016. For the month of July we see an increased cooling in the upwelling feature. We also see other small cooling in the southwest region, however the overall behavior does not seem to change. For the month of July we see a warming in the southwest region but mostly cooling everywhere else for the 2013 to 2014. For the years 2015 to 2016 we see a warming phase except for the area corresponding to the Monterey bay. This area of the coast appears to be unchanged for the three years.

To see the regions where the temperature change happens fastest, we use our posterior sample to compare the years 2012 and 2016 by calculating the probability that the change is greater then 1 degree Celsius for the month of January and 1.5 degrees Celsius for the month of July. In figure 3.10 we see the southern and northern most regions have an increase of over a degree in a span of four years.

Figure 3.14 we see the mean posterior surface projections plot for the month of January and July for the years 2020, 2040, and 2060. The projections at this level in time are no longer using the η correction parameter. Implementing a model with

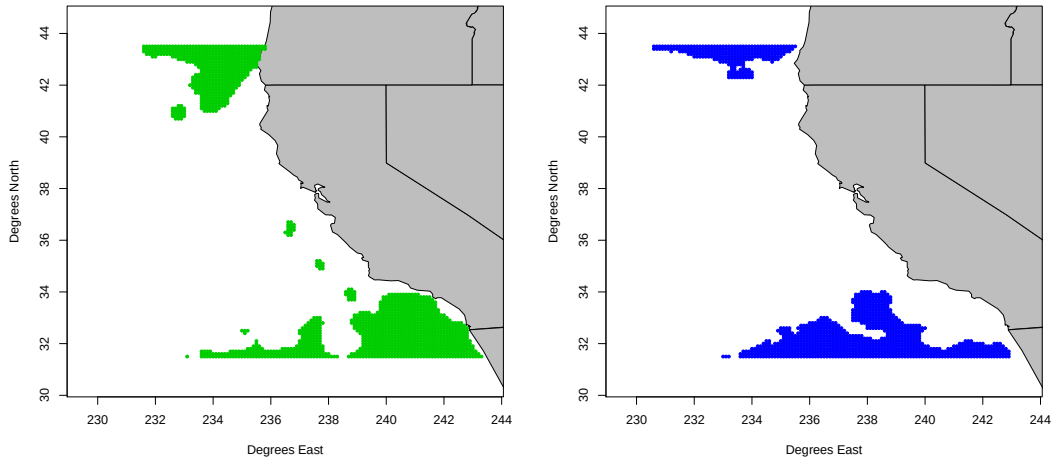

Figure 3.10: The figure shows the regions where the temperature change between January and July of 2012 and 2016 are greater than 1 degree and 1.5 degrees respectively for 95% of the posterior samples.

an auto-regressive component less than one will ultimately deteriorate the value of that process when projecting. We can see the large increase in temperature as we approach the mid 21st century.

Looking at the individual locations in figure 3.15 we can see the how the model adjusts to correct the the GCM ensemble. In some of the locations, mostly near the coast, the ensemble does not capture the localized variation and the model corrects for this. In figure 3.16 we can see how the varying degrees of adjustment the model makes.

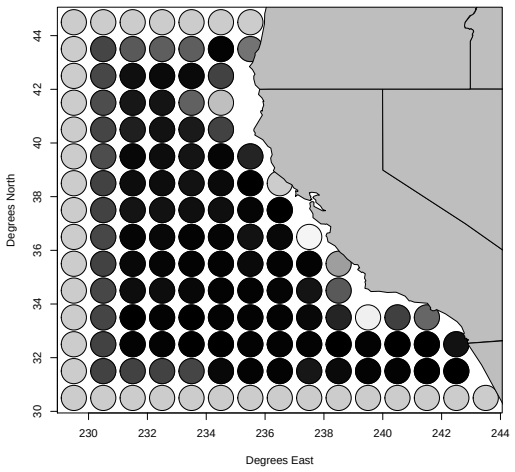

Figure 3.11: The figure shows the posterior sample means of the parameter ϕ with $\phi = 1$ (black), $\phi = 2$ (grey), and $\phi = 3$ (white).

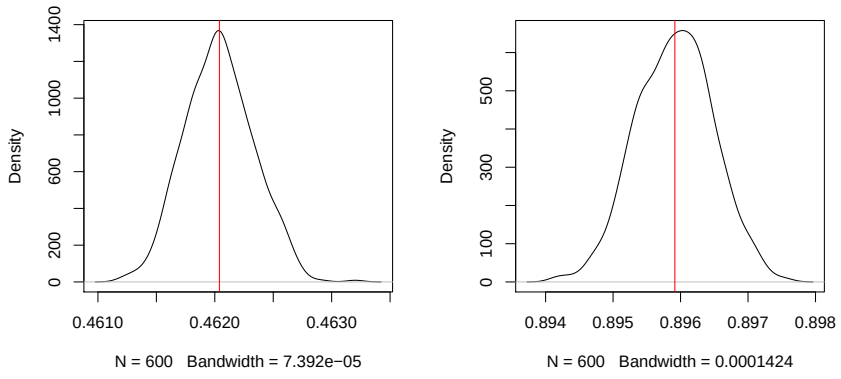

Figure 3.12: Posterior density for the parameters τ^2 (left) and ρ (right) with posterior mean (red).

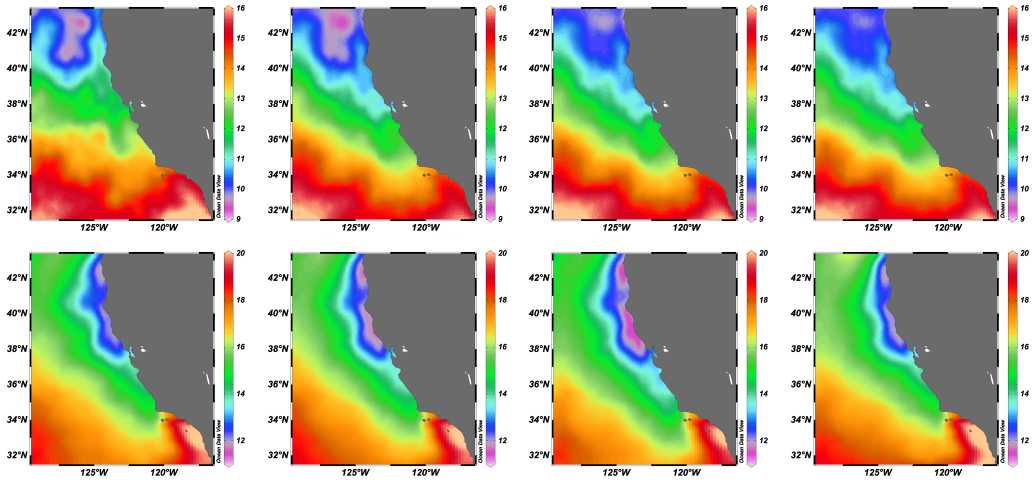

Figure 3.13: The figure shows projected SST fields for the month of January (top) and July (bottom) for the years 2013,2014,2015, and 2016 (left to right).

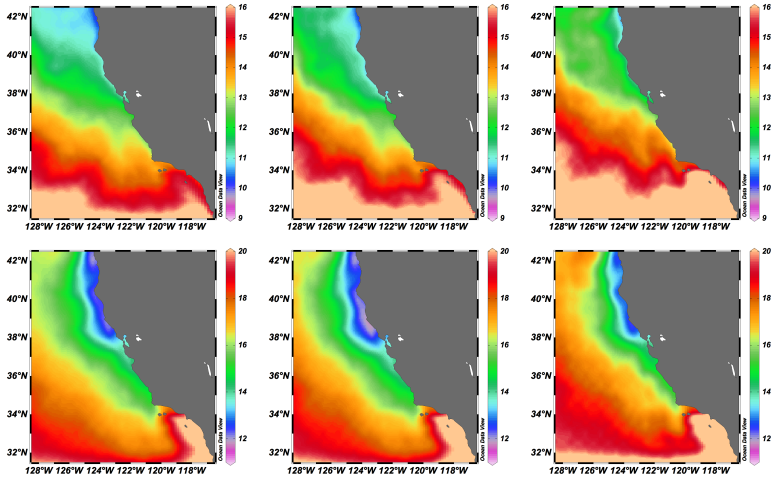

Figure 3.14: The figure shows projected SST fields for the month of January (top) and July (bottom) for the years 2020,2040,2015, and 2060 (left to right).

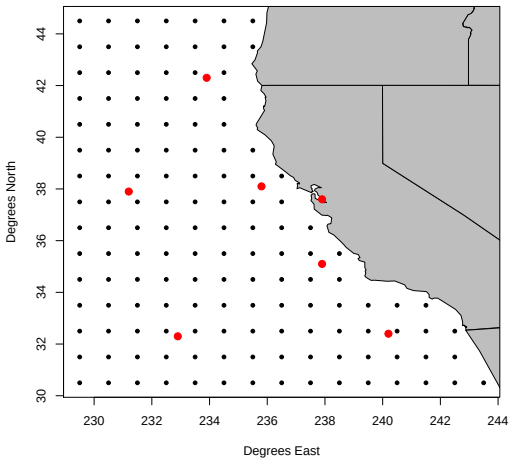

Figure 3.15: The figure shows the locations of six stations that are inspected in figure 3.16.

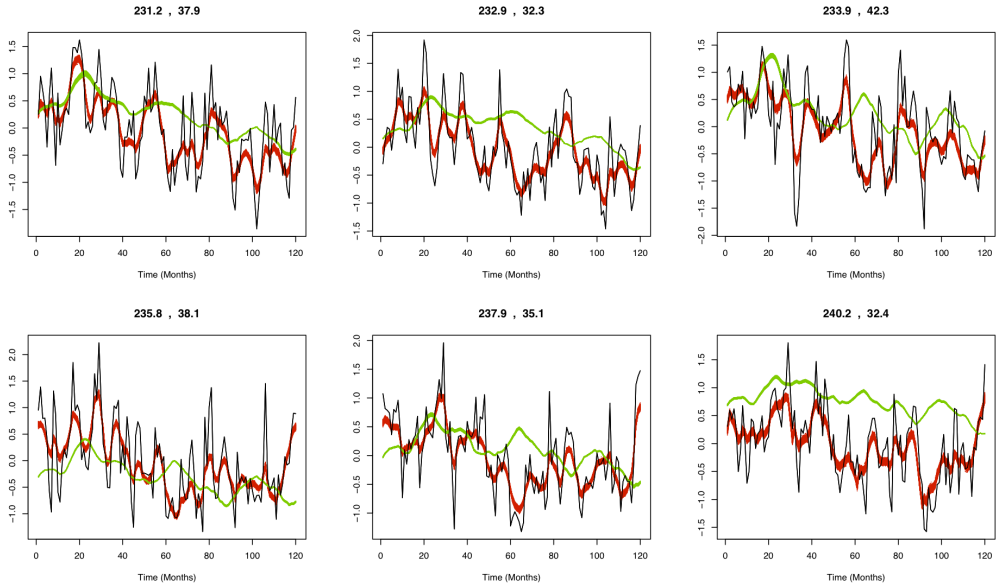

Figure 3.16: The figure shows the SST anomalies (green) and posterior 95% probability bands for the fitted anomalies (red) for six individual locations for the years 2003 to 2012.

3.4.1 Model Comparison

To validate the performance of our model we will compare our results to the GCMs used in the reconstuction. We perform a simple downscaling to the GCMs using a bilinear interpolation (BLI). The BLI is an extension of the linear interpolator over a 2-dimensional grid. We calculate the BLI for each of the three individual GCMs and then take the average. This gives us four different models on which to compare our projections.

We perform the statistical downscaling model for the time period 2003 to 2011 and see how well our model projects the observations for the 12 month period. We calculate the mean squared error (MSE) and mean absolute error (MAE) as

$$MSE = \frac{1}{IT}\sum_{t=1}^{12}\sum_{i=1}^{I}(Y_t(\mathbf{s}_i) - \tilde{Y}_t(\mathbf{s}_i))^2$$

$$MAE = \frac{1}{IT}\sum_{t=1}^{12}\sum_{i=1}^{I}|Y_t(\mathbf{s}_i) - \tilde{Y}_t(\mathbf{s}_i)|$$

where Y represents the observations and $\tilde{Y}$ represents the models. In table 3.1 we have the results of the MSE and MAE and we can see that our model performs better in predicting the 12 month period. The difference between the GCMs and our model is on the order of 3. Looking at the individual months in figure 3.17 we see that for the Summer and Fall months, our model does perform better compared to the GCMs downscaled model. This is a good indication that our statistical downscaling model is correcting the projections at the higher resolution.

	$\tilde{Y}$	$\bar{B}$	BCM	CM2	PCM
$\sqrt{MSE}$	0.639	1.926	1.819	1.999	2.861
MAE	0.490	1.364	1.411	1.467	2.202

Table 3.1: The table shows the square root of the Mean Square Errors and Mean Absolute Error for our model forecast $\tilde{Y}$, bilinear interpolation of the GCM average $\bar{B}$, and the bilinear interpolation for the three individual GCMs

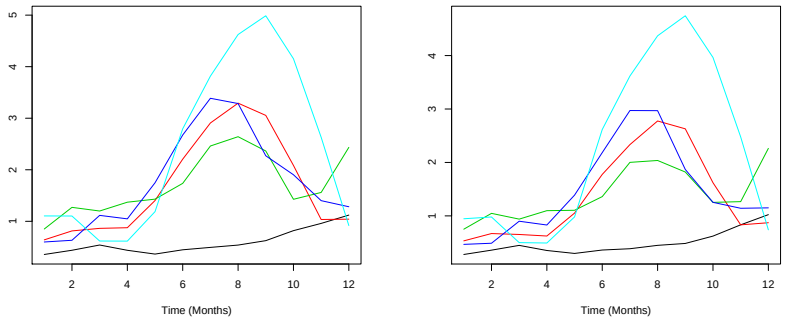

Figure 3.17: The MSE (left) and MAE (right) for the predicted year 2012. Our model (black), BLI GCM average (red), BCM-BL (green), CM2 (blue), and PCM (cyan) over the 12 months of prediction.

3.5 Concluding Remarks

The model considered in this chapter has a number of important characteristic that enable fast implementation of the estimation algorithms. One of those was to have a kernel with compact support so we could control the level of sparseness in the kernel matrix. This allows us to take advantage of the fact that most of the kernel matrix would be zero and manipulating this matrix would be computationally appealing. Another advantage of this model is the discounting technique implemented to the covariance matrix. The specification lets us to inflate the prior covariance as well as to simplify the Forward Filtering Backwards Sampling updating equations. Working with the simplified equation is the ground work for the parallel implementation in the next chapter.

There are a few avenues we can take to improve on these methods. One path would be to modify the kernel from circular support to allow ellipses. We could also explore other convolution kernel such as a tapered Matérn or by modeling the convolution kernel as a Gaussian Process.

Chapter 4

Parallel Computation

4.1 Introduction

The size of the problem increases drastically when looking at finer resolutions. Satellite data can now create a product with spatial resolution of $1/100°$, as seen in figure 3.1, which gives a reading every 0.75 miles and daily readings on the temporal scale. The amount of data that is available has surpassed the computational abilities we currently have; thus averaging on the spatial and temporal scale is necessary. At full spatial resolution the minimum size of the covariance matrix Q_t is on the scale of 1200 gigabytes per time point. This amount of information is unfeasible on most computing devices, as standard devices are on the level of 16 gigabytes per processor. Fitting a model to this data using serial programming would not be practical.

In this chapter, we will discuss the use of a Message Passing Interface (MPI), a library interface that allows you to run on distributed memory over multiple processors.

MPI is not itself a language, but is a library of commands that allows communication among the processors for the desired programming language. For the illustration in the next section, we will use Fortran as our programming language.

4.2 Data

In this chapter we will consider two sets of data. The first dataset that we focus on is the data described in Chapter 3. The spatial locations are of size 9609 and 120 on the temporal scale. The second data set we will consider is a simulated data set. We simulate the dataset for the purpose of benchmarking the algorithm. The spatial size remains the same, however we increase the temporal scale to 2400, which would represent 200 years of monthly data. In both cases we keep the large scale variability size to 140 locations.

4.3 Model

The model we use in this section is the same model we applied in Chapter 3. Recall the following model,

$$\begin{aligned} Y_t &= K(\phi)\lambda_t + \epsilon_t, & \epsilon_t \sim N(0, \tau^2 I_I) \\ \lambda_t &= L_{t-1} + G(\rho)\lambda_{t-1} + \omega_t, & \omega_t \sim N(0, W_t). \end{aligned} \tag{4.1}$$

where $p(\tau^2) \propto (\tau^2)^{-1}$ and W_t follow the discount approach described in chapter 3. This model has unknown parameters $\Omega = (\tau^2, \phi, \rho, \lambda)$. The one-step ahead and posterior distribution of $(Y_t|D_{t-1})$ and $(\lambda_t|D_t)$, assuming that the covariance matrices have reached

their limiting values, are as follows,

$$\begin{aligned} p(Y_t|D_{t-1}) &\sim N(F^{'}L_{t-1} + F^{'}Gm_{t-1}, Q) \\ p(\lambda_t|D_t) &\sim N(m_t, C) \\ Q &= \tau^2 \left(I_I + \tfrac{1-\delta}{\delta} F^{'}(FF^{'})^{-1}F\right) \\ C &= (1-\delta)\tau^2(FF^{'})^{-1} \\ m_t &= \tfrac{1}{\tau^2}CFY_t + \delta L_{t-1} + \delta G m_{t-1}. \end{aligned} \tag{4.2}$$

where Q is an $I \times I$ covariance matrix. In this model Q is the largest matrix to calculate. By increasing the resolution from 0.1° to 0.01° the covariance matrix Q cannot be calculated directly. Storing and decomposing Q is one of the challenges we simplify in the next section.

4.3.1 Posterior Distributions

When calculating the posterior distribution for (ϕ, ρ, τ^2) we do so using the marginal likelihood at time t. The marginal posterior is given by,

$$p(\phi, \rho, \tau^2|D_T) = \prod_{t=1}^{T} p(Y_t|D_{t-1}, \phi, \rho, \tau^2)p(\phi)p(\rho)p(\tau^2) \tag{4.3}$$

Focusing on the marginal likelihood $p(Y_t|D_{t-1}, \phi, \rho, \tau^2)$ and by letting $e_t = Y_t - F^{'}L_{t-1} - F^{'}Gm_{t-1}$, allows us to write the marginal likelihood as

$$\begin{aligned} p(Y_t|D_{t-1}, \phi, \rho, \tau^2) &\sim N(F^{'}L_{t-1} + F^{'}Gm_{t-1}, Q) \\ &\propto |Q|^{-\frac{1}{2}} exp(-\tfrac{1}{2}e_t^{'}Q^{-1}e_t). \end{aligned} \tag{4.4}$$

The covariance matrix Q in this likelihood is of the order 9609×9609 which makes it very difficult to work with as it will slow computations down significantly. One way

to deal with the Q is by not constructing the covariance matrix directly, but instead finding small manipulations to reduce the size of matrix needed to be store. We focus on simplifying the determinant of Q using Sylvester's determinant theorem (Mühlbach and Gasca, 1985),

$$\begin{aligned} |Q| &= |(\tau^2)(I_I + \tfrac{1-\delta}{\delta} F^{'}(FF^{'})^{-1}F)| \\ &= (\tau^2)^I |I_I + \tfrac{1-\delta}{\delta} F^{'}(FF^{'})^{-1}F| \\ &= (\tau^2)^I |I_u + \tfrac{1-\delta}{\delta} FF^{'}(FF^{'})^{-1}| \\ &= (\tau^2)^I |I_u + \tfrac{1-\delta}{\delta} I_u| \\ &= (\tau^2)^I \delta^{-u}. \end{aligned} \tag{4.5}$$

This simplification makes $|Q|$ a function of only τ^2 and the discount factor δ. By introducing a covariance matrix V to the observational equation, we have that the determinant is $|Q| = |V|\delta^{-m}$. If the matrix V is sparse this calculation can still be performed in a short time. This is one of the advantages of approximating the posterior covariance $C_t \approx C$.

The next step is to focus on calculations inside the exponential. Recall the definition of e_t, then

$$\begin{aligned} log(p(Y_t|D_{t-1})) &\propto e_t^{'} Q^{-1} e_t \\ &\propto {e_t}^{'} \tfrac{1}{\tau^2} \left(I_I - (1-\delta)F^{'}(FF^{'})^{-1}F \right) e_t \\ &\propto \tfrac{1}{\tau^2}({e_t}^{'} e_t - (1-\delta){e_t}^{'} F^{'}(FF^{'})^{-1}Fe_t). \end{aligned} \tag{4.6}$$

By defining the vector $n_t = Fe_t$, we can eliminate the need to create a matrix of size $I \times I$. Since the matrix $(FF^{'})^{-1}$ is $u \times u$ and n_t is $u \times 1$ we can effectively avoid creating a matrix larger than the smoothing kernel $F^{'} = K(\phi)$.

The posterior distribution for $p(\tau^2|D_T,\phi,\rho) \sim \Gamma^{-1}(\alpha_\tau,\beta_\tau)$, where α_τ and β_τ are the shape and scale parameters respectively of an inverse gamma distribution such that,

$$\begin{aligned} \alpha_\tau &= TI/2 \\ \beta_\tau &= \tfrac{1}{2}\sum_{t=1}^{T}(e_t^{'}e_t - (1-\delta)n_t^{'}(FF^{'})^{-1}n_t) \end{aligned} \tag{4.7}$$

The samples of τ^2 are obtained using Gibb steps conditional on the values of (ϕ,ρ). The posterior distribution for (ϕ,ρ) is

$$\begin{aligned} p(\phi,\rho|D_t,\tau^2) &\propto exp(-\tfrac{1}{\tau^2}\tfrac{1}{2}\sum_{t=1}^{T}(e_t^{'}e_t - (1-\delta)n_t^{'}(FF^{'})^{-1}n_t)) \\ &\times p(\phi)p(\rho). \end{aligned} \tag{4.8}$$

The order in which we sample the parameters is τ^2, ρ and then ϕ using MCMC algorithms and a Gibbs step for τ^2. In figure 4.1 we illustrate a simple diagram of the sampling algorithm.

4.4 Outline of the implementation on multiple processors.

Running a program on a distributed memory array gives you the advantage of running several independent tasks simultaneously. Moreover, by distributing the data over the different processors in the array, one can handle very large datasets. Also, spreading the workload evenly decreases computation time. Since each processor only carries a fraction of the dataset we must devise a way for the processors to communicate. Increasing runtime speedup can be achieved if communication between the processors is minimal.

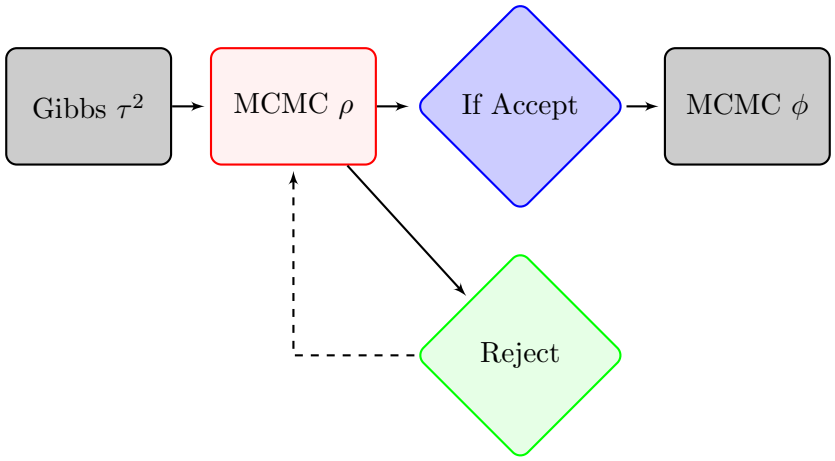

Figure 4.1: The order in which the parameters (ρ, ϕ, τ^2) are sampled. The color of the square indicates the number of times the FF algorithm has to be executed with grey representing one time and red representing more than once.

Let the time it takes a processor to complete a task be defined as o_j^k, where $j = 1, \ldots, J$ is the j-th task and $k = 0, \ldots, (K-1)$ is the $k-th$ processor. In this context a task is defined as a job to be completed by a processor k. Communication between the processors (k_u, k_v) can vary depending on the logistics of the hardware. Since the hardware is not something we can control, we will assume all processors take the same amount of time to communicate and define the communication time as $com(k) = max_k(com(k_u, k_v))$ where

$$\begin{cases} com(k_u, k_v) = 0 \quad if \quad u = v \\ com(k_u, k_v) > 0 \quad if \quad u \neq v. \end{cases}$$

We define R_k^d as the number of times the processors communicated for a work cycle d

where $d = 1, \ldots, D$. We denote the total time needed to perform a work cycle as

$$\psi_d^K = R_k^d com(k) + \sum_j max_k(o_j^k) \tag{4.9}$$

There are a few properties we must consider when looking at computation time. As K increases then o_j^k decreases and the number of communications R^d increases. Therefore there exists a equilibrium point K_0 such that for all $K_0 < K_j$ the cycle time $\psi_d^{K_0} > \psi_d^{K_j}$. While the value of K_0 gives you the maximum number of processors for equilibrium, it is not the most desired value for K as the benefit of increasing K will start to give diminishing returns.

For the purpose of this illustration, let us assume we have access to $K \in [1, K_0]$ many processors. Minimizing the computation time of a cycle can be done in multiple ways: increasing the number of processors K (we will show later that there is an upper limit for efficiency), optimizing task time o_j^k, and limiting the number of communications R_k^d.

4.4.1 Optimizing task time

The optimal way to assign work is in such a way that you assign unique work to each processor k and limit the communications. Global communication occurs when one processor must communicate to the entire group, this is also known as a broadcast. A broadcast should only be called when it is absolutely necessary as the processor who establishes the broadcast is not released until all other processors have established communication creating a communication bottle neck.

The total time it takes for a cycle of work to complete depends on communication time, $com(k)$. We will not be addressing how to optimize $com(k)$ since improving its performance is not determined by a modeling scheme, but by outside factors. We fix $com(k)$ as the slowest communication time between any two processors by limiting the sharing of information to a block format. There are two main types of communication. Blocking communication is when a processor sends a message to another processor and cannot be freed until the corresponding processor receives this message. Buffering communication allows for a processor to free itself by dumping the information on a memory buffer and proceeding. The receiving processor then collects the information on the buffer and continues. The latter communication is faster in cases where the communication speed between processors is not equal. Since our benchmark will require us to use all the nodes in our cluster the communication speed will not be equal.

In figure 4.2 we see a general outline of how the algorithm will run. We start by initializing communication among the processors. This opens communication with the group of processors and labels them from $k = 0, \ldots, (K-1)$. The number of processors to be used must be defined at the start of the program and these processors cannot be released until the end when we finalize and free them. We will define processor $k = 0$ as our MVP (most valuable processor) since all the information that will be outputted to the hard-drive will be gathered on MVP. In the serial step we initialize all the processors with the information we will need such as reading in data, creating storage arrays, and any other computations we need to prepare for the parallelization step. We also use the end of this step to broadcast any information calculated in MVP that needs to be

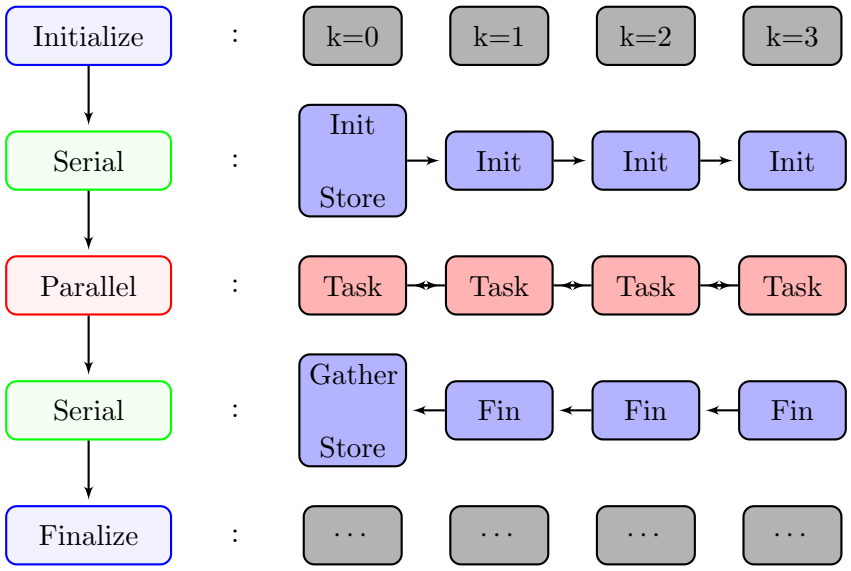

Figure 4.2: The general layout for a code that runs in a parallel architecture. At the initializing step you open communication to all the processors. In the serial step you perform all the operations needed prior to begin the parallel computation. The largest part of the computations should be performed during this parallel step. The second serial step you gather all the information and finally you close communication and release the processors.

shared. The parallel step is where the majority of the heavy computation should be taking place. Once the parallel step is complete, we gather all the information in our MVP and free the processors by finalizing the communication.

4.5 Parallelizing over time

The serial approach to a time dependent task begins it's calculations at $t = 1$ and end at $t = T$. The natural approach to parallelize is to break up time into K many intervals where the starting point $t_1^k = Tk/K + 1$ and end points $t_T^k = T(k+1)/K$ for processor k. Each processor will then have a time interval of size $n_k = T/K$ such

that $\sum_{k=0}^{K-1} n_k = T$. In the case where $\frac{T}{K}$ is not an integer, we distribute the remaining time as evenly as possible. Making the size of the intervals n^k as close to equal for all processors is important. If the time intervals n_k are too uneven, then the processors with the smaller intervals will compete faster and will sit idling. A processor is idle if it is not performing any calculations as a result of having to wait for other processors to finish. Below we discuss how we limit our idle time when running our algorithm.

4.5.1 Metropolis-Hastings algorithm

Assume that we have a random process θ with observations y. In the Metropolis-Hastings algorithm (MH) we calculate an acceptance ratio to determine whether we transition from the previous state to a new proposed state ($\theta^{(i-1)} \rightarrow \theta^*$). The acceptance rate α_{MH} is defined as

$$\alpha_{MH} = \frac{p(\theta^*|y)/J(\theta^*|\theta^{(i-1)})}{p(\theta^{(i-1)}|y)/J(\theta^{(i-1)}|\theta^*)} \tag{4.10}$$

where the function $J(\cdot,\cdot)$ is a proposal distribution. We accept the proposed value of θ^* with probability $min(1, \alpha_{MH})$. If we reject θ^*, then we let $\theta^{(i)} = \theta^{(i-1)}$ and restart the algorithm until all iterations have been completed.

To sample from the posterior distribution for (ρ, ϕ), we will use the MH algorithm. The proposal distribution we use for both parameters is a truncated normal distribution with lower and upper bounds $(0, 1)$ and $(1, 3)$ for ρ and ϕ respectively. We will focus on the ratio of the posterior distributions and how we will evaluate this ratio in the task environment. For the moment let us ignore the proposal distribution and

focus on the marginal posterior of ρ conditioning on ϕ. We can write α_{MH} as

$$\begin{aligned} \alpha_{MH} &= \frac{p(\rho^*|D_T,\ldots)}{p(\rho^{(i-1)}|D_T,\ldots)} \\ &= \prod_{k=0}^{K-1}\prod_{t_k=1}^{T}\frac{p(Y_{t_k}|D_{t_k-1},\rho^*,\ldots)}{p(Y_{t_k}|D_{t_k-1},\rho^{(i-1)},\ldots)} \\ &= \prod_{k=0}^{K-1}\alpha_{MH}^k \end{aligned} \tag{4.11}$$

so that the acceptance rate can be separated into parts over each processor k. This type of simplification allows each processor to calculate a portion of the acceptance ratio and return only the value α_{MH}^k. The MVP then collects all the values of α_{MH}^k and together with the proposal distribution the value is tested and the MH algorithm is completed for iteration i.

4.5.2 Posterior Distribution Simplification

By breaking up the full length of time into K many intervals of length n_k, we have done two things: reduced the work load of the processors by only needing n_k much of the data, and forced the processor to establish communication to collect the full time span. The only time dependent values in the posterior distribution in equation 4.6 are e_t, which in turn depends on m_{t-1}. This means that for all processors $k < (K-1)$, the value of m_{T_k} will be sent to processor $k+1$ and in turn, all processors $k > 0$ must receive that value. To calculate the marginal posterior distribution, we need to calculate e_{t_k} where

$$\begin{aligned} e_{t_k} &= Y_{t_k} - F^{'}L_{t_k-1} - F^{'}Gm_{t_k-1}. \\ m_{t_k} &= CFV^{-1}Y_{t_k} + \delta L_{t_k-1} + \delta G m_{t_k-1}. \end{aligned} \tag{4.12}$$

This will be defined as the FF task. The primary objective is to calculate the value of m_{T_k} as quickly as possible so that the next processor can begin. We can simplify the calculations by performing most of the matrix multiplications prior to performing the iterative time loop. We define the calculations

$$\begin{aligned} J_{t_k} &= CFV^{-1}Y_{t_k} + \delta L_{t_k-1} \\ N_{t_k} &= Y_{t_k} - F^{'}L_{t_k}. \\ P^{'} &= F^{'}G \end{aligned} \tag{4.13}$$

as The Loop Operations (TLO) step. This step allows us to cut back on computation time since we are only performing this step once instead of at each time step. TLO step simplifies our equation to

$$\begin{aligned} e_{t_k} &= N_{t_k} - P^{'}m_{t_k-1}. \\ m_{t_k} &= J_{t_k} + \delta G m_{t_k-1}. \end{aligned} \tag{4.14}$$

which limits the matrix multiplications to one per time step. The values of e_{t_k} will only be used locally, so calculating them will be performed after the communication has been established. Figure 4.3 shows a general outline of how communication is established for the FF task.

In this situation, having idle time is unavoidable. The processor $k+1$ cannot begin to work until processor k reaches the end of the loop. By creating the TLO step and calculating e_{t_k} after communication, we have decreased the idle time as much as possible. Every task in a work cycle will have a TLO step.

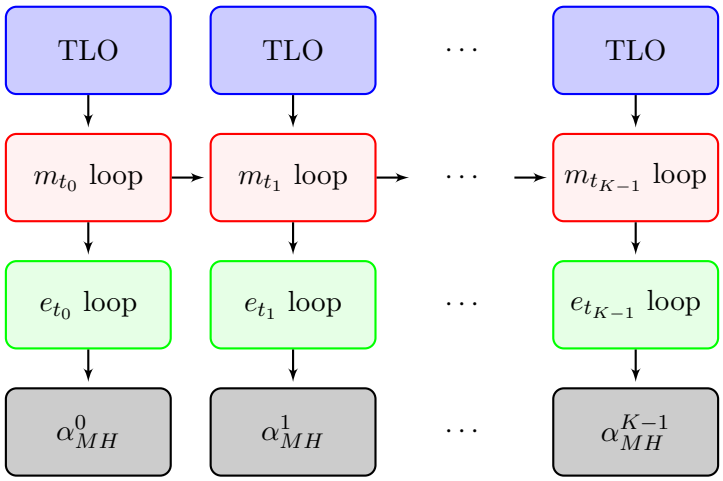

Figure 4.3: The diagram shows where communication is taking place during the FF task.

4.5.3 Task Identification

In figure 4.1, the order in which the parameters (ρ, ϕ, τ^2) would be sampled was established. The first task will be to implement a MH algorithm to sample ρ. Since we are blocking the algorithm to only continue if ρ is accepted, this task will be the most complicated of the three.

The Task$_\rho$ begins by sampling ρ^* from a normal proposal distribution truncated between 0 and 1,

$$J(\rho^*|0,1) \sim trN(\rho^{(i-1)}, 0.05^2). \tag{4.15}$$

The value of $(\rho^{(i-1)}, \rho^*)$ is then broadcasted from MVP to all the processors and the TLO step is ready to begin. For visual representation, we separate the Task$_\rho$ into two separate blocks as seen in figure 4.4. Block 1 and Block 2 represent the FF algorithm for the value of $\rho^{(i-1)}$ and ρ^* respectively. The values of $(m_{t_k}, m^*_{t_k})$ are calculated

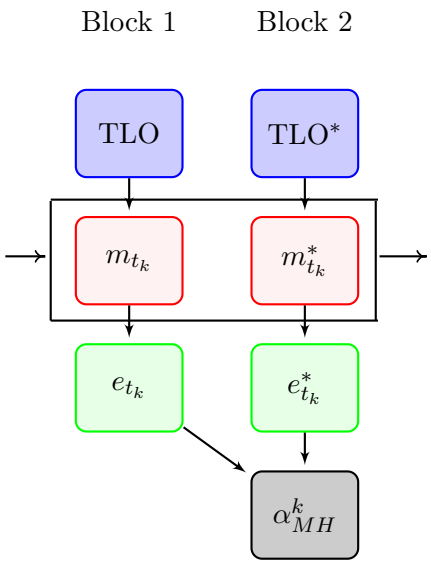

Figure 4.4: A diagram for how processor k distributes work for Task$_\rho$.

together and upon completion the final value is communicated to the next processor. Once communication is complete, the value for the acceptance rate is calculated and communicated back to MVP. If the value of ρ^* is rejected, a new value is sampled from the proposal distribution and only Block 2 is recalculated. Upon accepting the value ρ^*, we move on to the next task. We must note that we might encounter a situation in which we start to reject a large amount of samples before accepting. To avoid getting stuck, we have set up a limit to the number of consecutive rejections to thirty upon which the value of $\rho^{(i)} = \rho^{(i-1)}$ and we continue to the next task.

With the completion of Task$_\rho$ we move on to Task$_\phi$. We begin the task by sampling ϕ^* from the truncated normal proposal distribution

$$J(\phi^*_m|1,3) \sim N(\phi^{(i-1)}_m, 0.25^2) \tag{4.16}$$

with lower and upper truncation at $(1, 3)$. This task only has one block since the value of e_{t_k} has already been calculated in the previous task. Task_ϕ calculates Block 2 and returns the acceptance rate, thus completing the algorithm. If the sample is accepted or rejected we move on to Task_τ and use the value of e_{t_k} or $e^*_{t_k}$ accordingly. Task_τ does not require it's own FF algorithm since the value of e_{t_k} have already been calculated and so it's calculation is attached to the tail end of Task_ϕ. When we calculate the acceptance rate α^k_{MH} we also calculate the portion of β^k_τ in equation 4.7. Note that the correct value of $\phi^{(i)}$ is not known until the MH algorithm is completed by MVP. This is why we calculate both β^k_τ and $\beta^k_{\tau*}$ and use the appropriate values. This does not increase runtime since these values are a byproduct of calculating the acceptance rate.

A work cycle consists of the completion of these three step. We run these cycles iteratively until the posterior samples for (ρ, ϕ, τ^2) have converged. We remove the burn-in sample and denote the full posterior estimations for the converged chain to be $(\rho^l, \phi^l, \tau^{2,l})$ where $l = 1, \ldots, L$. With the posterior samples of (ρ, ϕ, τ^2) completed we move on to the next and final task of sampling the values of λ.

4.5.4 Forward Filtering Backwards Sampling in Parallel

In the previous section we explained how to perform the FF algorithm to obtain the marginal distribution for $p(Y_t|D_{t-1}, \ldots)$. We then use the marginal distribution to calculate the posterior distributions for (ρ, ϕ, τ^2) and using an MH algorithm and a Gibbs step to obtain our posterior samples. Having the full posterior samples for (ρ, ϕ, τ^2) allows us to perform multiple FF steps at once.

The Backwards Sampling equations for $p(\lambda_t|D_T, \lambda_{t+1})$,

$$\begin{aligned} H &= C - \delta CG'C^{-1}GC \\ h_t &= m_t + \delta CG'C^{-1}(\lambda_{t+1} - (L_t + Gm_t)) \end{aligned} \tag{4.17}$$

where $\lambda_t \sim N(h_t, H)$ for all $t = (T-1), \ldots, 1$ and for $t = T$ the distribution for $\lambda_T \sim N(m_T, C)$. Since the only unknown values in these equations are the values of λ_{t+1}, we set up our TLO step by calculating the following equations prior to communication,

$$\begin{aligned} P_H &= \delta CG'C^{-1} \\ u_{t_k} &= m_{t_k} - P_H(L_{t_k} + Gm_{t_k}) \\ H &= C - P_H GC \end{aligned} \tag{4.18}$$

and thus simplifying the mean vector h_{t_k} to

$$h_{t_k} = u_{t_k} + P_H \lambda_{t_k+1}. \tag{4.19}$$

The values of λ_{1_k} will be the only values communicated to processor $k-1$ where $k > 0$. The cycles in the previous section were memoryless. This was beneficial since the information calculated in one iterative step did not need to be stored for the remaining iterative steps. For Task$_\lambda$ each processor will have to remember $(K-k)$ many levels of information. Task$_\lambda$ consists of a series of FF block and BS block who's calculations and communication are shown in figure 4.5. It's important to note that the TLO step will always finish before communication is established by the previous processor. This means a processor that is receiving information will always be ready to get that information and thus the sending processor will immediately be freed to perform the following TLO step.

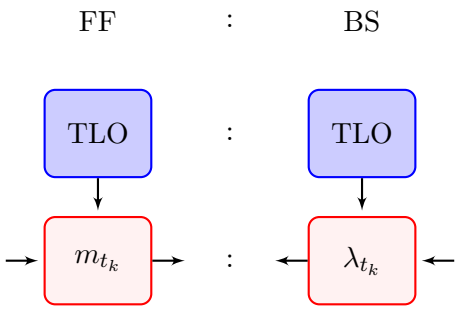

Figure 4.5: A diagram for how processor k distributes work for Task$_\lambda$.

We start Task$_\lambda$ by using the first posterior sample $(\rho^1, \phi^1, \tau^{2,1})$. We define the notation $FF^l_{n_{(k+1)}}$ and $BS^l_{n_{(k+1)}}$ for the l-th iteration and the k-th processor. In figure 4.6 we see an illustration of Task$_\lambda$ with $K = 4$ processors and an arbitrary number of iterations L. The algorithm can be separated into three different stages. The Warmup is the first stage and refers to the initial steps needed to be completed until all idle processors are eliminated. The Cool Down is almost a mirror image of the Warmup stage in that it is the final steps taken as the algorithm starts to finalize. The Pong stage refers to body of the algorithm. This stage is where all processors are running and communicating back and forth. We will identify the beginning and ending to these stages in terms of the MVP. The Warmup stage begins with MVP calculating the FF block for $l = 1$ and ends at $l = K$, the Cool Down stage starts when MVP is performing the BS for $l = L - K + 1$ and ends with $l = L$, the Pong stage takes place in between. The upper and lower triangular grey nodes in the Warmup and the Cool Down stages correspond to idle processors. The grey nodes below and above the FF and BS block

are semi-idle processors. We label them as semi-idle due to the fact that they are active but do not take a full block to run and are more for a visual representation. For any K many processors we will have $K(K-1)$ many idle processors in the Warmup and the Cool Down stages.

The colors represent the FF BS blocks for a given iteration with the red and green for i and orange and blue for $i+1$. The darker shades corresponds to the levels of information the processor must store throughout the algorithm where the darker the color, the less information stored. For example, processor $k = K-1$ has to store one level of information while MVP has to store K many levels of information. Once a processor had completed the BS block for an iteration i, all the information is discarded and replaced with the information for the next iteration.

The number of communications in this task is dependent on both the number of processors K and the number of iterations L. We can denote R_λ as the number of times communication is established for Task_λ and define it as

$$\begin{aligned} R_\lambda &= (K-1)(I+K) + K/2 \quad , \quad even \\ R_\lambda &= (K-1)(I+K) + (K-1)/2 \quad , \quad odd \end{aligned} \tag{4.20}$$

where $I > K$, making the minimum number of communications for an even number of processors to be $2K^2 - K/2 - 1$. We impose the condition that $I > K$ so that we may utilize the three stage approach. In the case that $I \leq K$, the MVP processor terminates work and becomes idle for many steps before it is given information and thus, the Pong stage is never formulated.

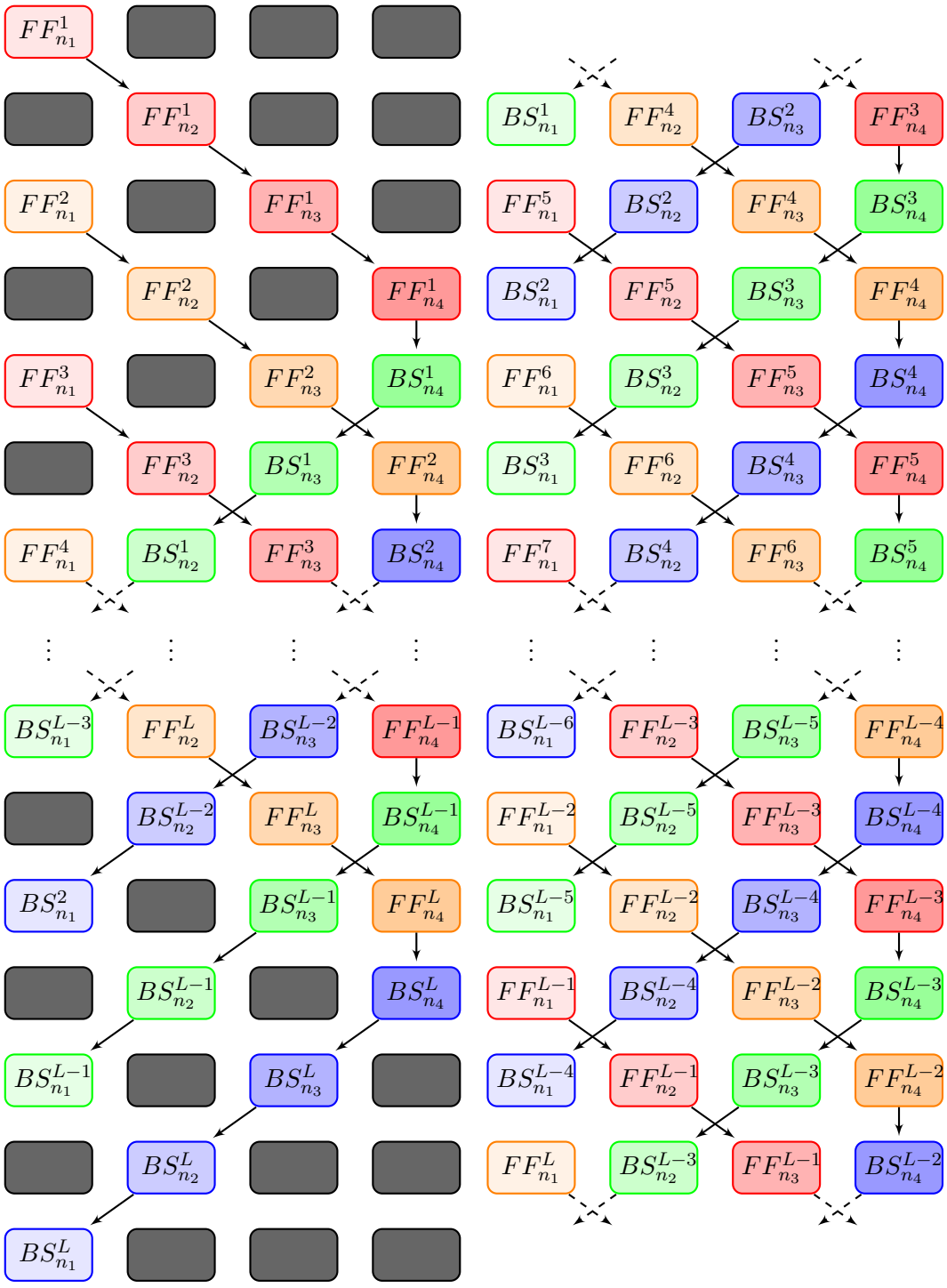

Figure 4.6: The FFBS Criss-Cross algorithm (FFBSCC) example with $K = 4$ processors for an arbitrary L many iterations. The FF is in red and orange with the BS in green and blue. The solid grey box indicated an "idle" state in which the processor is waiting.

There are a few things to keep note of when running this algorithm. To minimize runtime, the majority of the work must take place in the Pong stage since the Warmup and Cool Down stages have idle time. This allows you to become flexible with the number of processors used depending on the size of the posterior sample. In table 4.5.4 we see the total number of communications for a set of I and K. We see that as the number of iterations gets larger the number of communications starts to converge to $(K-1)I$ which is the ideal number of communications. For Task_λ, Table 4.5.4 is the

$I \backslash K$	2	4	8	16	32	64	128
100	103	314	760	1748	4108	10364	32640
1000	1003	3014	7060	15248	32008	67064	143320
10000	10003	30014	70060	150248	311008	634064	1286320
100000	100003	300014	700060	1500248	3101008	6304064	12716320

Table 4.1: Number of communications for I iterations and K processors.

minimum number of communication for the corresponding K. To minimize the total runtime of Task_λ we must find a balance between R_λ and the time it takes to perform the FF and BS block. In the section below we show the diagnostics for the performance of Task_λ.

4.6 Results

The Forward Filtering (FF) algorithm in 4.5.2 is used substantially when sampling the parameters from the posterior distribution. It is used to calculate the marginal likelihood for the parameters (ρ, ϕ, τ^2) and is necessary to begin the FFBSCC algorithm described in 4.5.4. The algorithm is embedded in the tasks: Task_ρ, Task_ϕ, and Task_τ. We define Task_α to be one cycle of performing Task_ρ, Task_ϕ, and Task_τ, this corresponds to one iterative cycle of sampling the parameters (ρ, ϕ, τ^2). We define Task_β as performing one iterative cycle in the FFBSCC algorithm. We compare the tasks for completing 100 iterations for the simulated data as well as the observational data corresponding to the model in Chapter 3. In order to run a balanced FFBSCC algorithm, we must increase the number of iterations when increasing the processors to allow the algorithm to run at full potential. We then scale the values down to compare with the other benchmarks at 100 iterations.

K	1	2	4	8	16
FF	208.23	108.11	68.11	48.84	51.29
BS	205.12	121.41	65.53	48.52	73.05

Table 4.2: The total time (seconds) it takes for Task_α and Task_β to complete completing 100 iterations for K processors using the observational dataset.

The left panels of figure 4.7 display the runtimes for Task_α and Task_β for the observational data. Both the tasks appear to bottom out at 8 processors which amounts to 15 time points per processor. The right two panels show the speedup and efficiency of

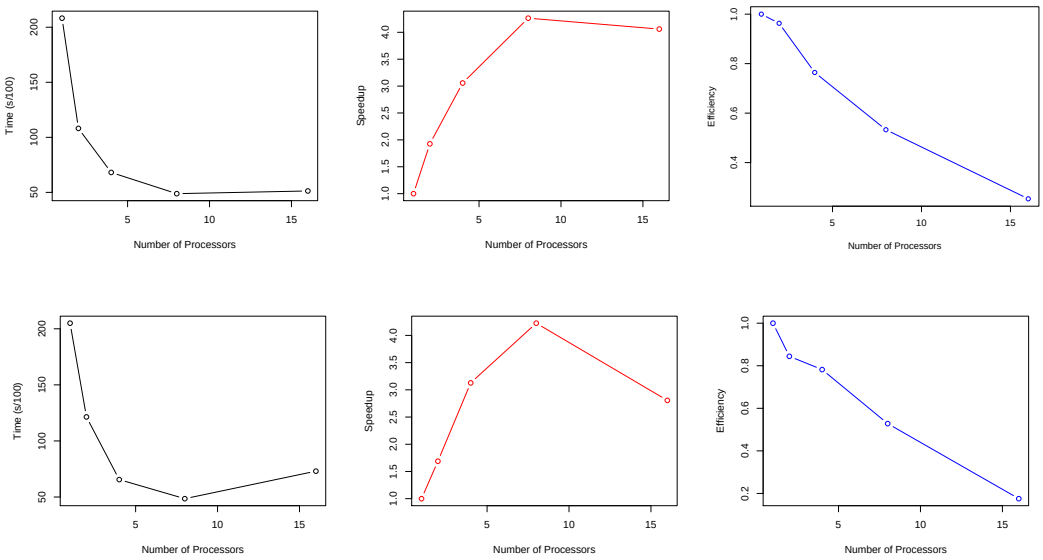

Figure 4.7: Runtime, speedup, and efficiency (left to right) of the Task_α (top) and Task_β (bottom) for the observational data corresponding to the model in Chapter 3.

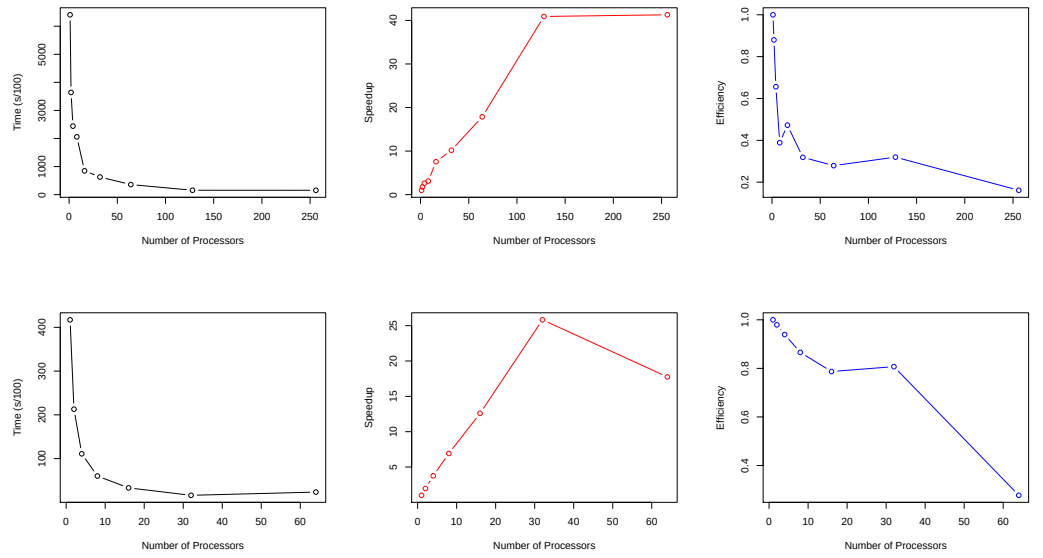

Figure 4.8: Runtime, speedup, and efficiency (left to right) of the Task_α (top) and Task_β (bottom) for the simulated data corresponding to 200 years of monthly SST.

K	1	2	4	8	16	32	64	128	256
FF	6406	3640	2440	2059	847	627	358	156	155
BS	417	213	111	60	33	16	23	NA	NA

Table 4.3: The total time (seconds) it takes for Task_α and Task_β to complete completing 100 iterations for K processors using the simulated dataset.

adding additional processors. For the observational dataset we see that at 8 processors we are able to run 4.2 times faster than on a single processor while losing only 48% efficiency.

Using the simulated dataset, in which the temporal size is increased by a factor of twenty, we see the algorithms full potential. In Figure 4.8 we present the runtime, speedup, and efficiency for Task_α and Task_β. Since the parallelism occurs in time we see a large increase in speedup time. Task_α tops out at $K = 128$ processors which amounts to 19 data points per processor. Task_β reaches its upper limit at a much smaller number of K=32 processors and begins to take longer with each additional processor. This limitation is most likely due to the limiting the communication into blocks. That being said, this gives us a speedup of 40 and 25 with a loss in efficiency of 68% and 20% for Task_α and Task_β respectively. These are encouraging results, however there is still areas for optimization.

4.6.1 Concluding Remarks

The algorithms in this chapter have been shown to be effective even at small temporal sizes. For the two example illustrated the algorithms reached an upper limit but for different reasons. The FF algorithm scales well when increasing the temporal size but will eventually max out due to the sequential portions of the code, this is known as Amdahl's law. The Forward Filtering algorithm is by construction sequential as is the Backwards Sampler algorithm. Testing the FF algorithms for different temporal sizes we discover the 'sweet spot' is to have 15-20 time points per processor. Any fewer than that and the algorithm takes longer to communicate than to perform the task assigned to it. The FFBSCC algorithm reaches a different type of limit. The method in which we establish communication heavily penalizes the algorithms speed for each additional processor. The FFBSCC algorithm does not scale as well as the FF algorithm, however the benefits of FFBSCC algorithm are still important as it allows us to implement the Forward Filtering Backwards Sampling techniques on datasets that are too large for one processor. There are still many ways these algorithm can be improved. One possibility is by extending the parallel implementation for space and thus allowing the regions to increase in size as well. Another extension would be to explore the load each processor must calculate as a decreasing function in time. For the time being, these approaches will have to wait.

Chapter 5

Conclusion

This work focuses on general methods to improve forecasting accuracy using information spanning from different outlets and different resolutions. As an illustration we focused on blending climate model simulations with observations to obtain more accurate SST fields. By using a set of basis functions to represent the spatial field of the observations, we provided an alternative approach on how to compare simulated output from various sources. This approach proves to be useful as it allows us to compare climate model indices to observational indices without directly comparing their values of SST. We implemented a Bayesian Hierarchical models to capture the underlying baseline while also modeling the climate models discrepancy to the observational signal. This resulted in a unified forecast obtained from the model ensembles.

The availability of satellite readings has increase the spatial resolution upwards to a 1/100th of a degree. This produces an abundance of spatial high resolution data for short periods of time. Capturing longterm behaviors is not possible using higher spatial

resolutions and coastal behaviors are not captured using coarse spatial resolution. Here we focused on creating a link between large and small scale variability. A combination of a Discrete Process Convolution and a Dynamic Linear Model proved suitable modeling tools in fitting and forecasting at the higher resolution. The choice of a flexible kernel allowed us to determine the sparseness which in turn diminishes some of the computational loads without compromising the models accuracy. The use of high resolution datasets makes computations on a single core computing devices time expensive. Sampling and tuning parameters in the model become increasing difficult when calculations begin to take days. When working on a single core is too time consuming, temporal parallelization is an effective way for speedup. The parallel algorithms developed in this work enable us to run MCMCs for large iterative chains and scale well in time.

Appendix A

Climate Model Details

A.1 Climate Models

1. Bjerknes Centre for Climate Research, Norway, (BCCR-BCM2.0) 2005
2. Canadian Centre for Climate Modelling and Analysis, Canada, (CGCM3.1.T47) 2005
3. Meteo-France/Centre National de Recherches Meteorologiques, France, (CNRM-CM3) 2004
4. Commonwealth Scientific and Industrial Research Organization Atmospheric Research, Australia, (CSIRO-MK3.0) 2001
5. National Key Laboratory of Numerical Modeling for Atmospheric Sciences and Geophysical Fluid Dynamics/Institue of Atmospheric Physics, China, (FGOALS-g1.0) 2004
6. National Oceanic and Atmospheric Administration/Geophysical Fluid Dynamics Laboratory, USA, (GFDL-CM2.1) 2005
7. National Aeronautics and Space Administration/Goddard Institute for Space Studies, USA, (GISS-ER) 2004
8. Institute for Numerical Mathematics, Russia, (INM-CM3.0) 2004
9. National Institute of Geophysics and Volcanology, Italy, (INGV-ECHAM4.6)

10. Institut Pierre Simon Laplace, France, (IPSL-CM4) 2005
11. Center for Climate System Research, National Institute for Environmental Studies, and Frontier Research Center for Global Change, Japan, (MIROC3.2 medres) 2004
12. Meterorological Institude of the University of Bonn, Meteorological Research Institude of the Korea Meteorological Administration and Model and Data Group, German/Korea, (Echo-G) 1999
13. Max Planck Institute for Meteorology, German, (ECHAM5/MPI-OM) 2005
14. Meteorological Research Institute, Japan, (MRI-CGCM2.3.2) 2003
15. National Center for Atmospheric Research, USA, (CCSM3) 2005
16. National Center for Atmospheric Research, USA, (PCM) 1998
17. Hadley Centre for Climate Prediction and Research/Met Office, UK, (UKMO-HadCM3) 1997

A.2 Dynamic Linear Models

A.2.1 Forward Filtering Backwards Sampling

The general multivariate DLM as described in West and Harrison (1997) defines the observational time series Y_t, a vector of size n, to have the likelihood and state space equations defined as,

$$\begin{aligned} Y_t &= F_t'\theta_t + \nu_t \\ \theta_t &= G_t\theta_{t-1} + \omega_t \end{aligned}$$

where $\nu_t \sim N(0, V_t)$ and $\omega_t \sim N(0, W_t)$. F'_t is the dynamic regression matrix, G_t is the state evolution equation, and θ_t is the state space vector of size r. The initial prior at

time $t = 0$ is defined as,

$$(\theta_0|D_0) \quad \sim \quad N(m_0, C_0)$$

where m_0 and C_0 are known mean and covariance matrix and $D_t = (Y_t, D_{t-1})$ represents all the information up to time t. The Forward Filtering equations are obtained as,

$$\begin{aligned}
p(\theta_t|D_{t-1}) &= \int p(\theta_t|\theta_{t-1})p(\theta_{t-1}|D_{t-1})d\theta_{t-1} &\sim N(G_t m_{t-1}, R_t) \\
p(Y_t|D_{t-1}) &= \int p(Y_t|\theta_t)p(\theta_t|D_{t-1})d\theta_t &\sim N(F'_t G_t m_{t-1}, Q_t) \\
p(\theta_t|D_t) &\propto p(Y_t|\theta_t)p(\theta_t|D_{t-1}) &\sim N(m_t, C_t)
\end{aligned}$$

for $t = 1, \ldots, T$. The Backwards Sampling equations are obtained as,

$$\begin{aligned}
p(\theta_T|D_T) &\sim N(m_T, C_T) \\
p(\theta_t|\theta_{t+1}, D_T) &\propto p(\theta_{t+1}|\theta_t)p(\theta_{t+1}|D_T) \sim N(h_t, H_t)
\end{aligned}$$

where the mean vector h_t depends on the previously sampled values of θ_{t+1}. This is the general scheme of the algorithm we implement for the models in Chapter 2 thru 4.

Bibliography

Berrocal, V., Gelfand, A., and Holland, D. (2010). A spatio-temporal downscaler for output from numerical models. *Journal of Agricultural, Biological, and Environmental Statistics*, 15(2):176–197.

Berrocal, V. J., Craigmile, P. F., and Guttorp, P. (2012). Regional climate model assessment using statistical upscaling and downscaling techniques. *Environmetrics*, 23(5):482–492.

Cloern, J., Hieb, K., Jacobson, T., Sansó, B., Di Lorenzo, E., Stacey, M., Largier, J., Meiring, W., Peterson, W., Powell, T., Winder, M., and Jassby, A. (2010). Biological communities in San Francisco Bay track a north Pacific climate shift. *Geophysical Research Letters*. To appear.

Cressie, N. and Wikle, C. K. (2011). *Statistics for Spatio-Temporal Data*. Wiley, Hoboken, NJ.

De La Horra, J. and Rodríguez-Bernal, M. (1999). The posterior predictive p-value for the problem of goodness of fit. *TEST*, 8:117–128. 10.1007/BF02595865.

Di Lorenzo, E., Fiechter, J., Schneider, N., Bracco, A., Miller, A. J., Franks, P. J. S., Bograd, S. J., Moore, A. M., Thomas, A. C., Crawford, W., Pena, A., and Hermann, A. J. (2009). Nutrient and salinity decadal variations in the central and eastern North Pacific. *Geophysical Research Letters*, 36(L14601). doi:10.1029/2009GL038261.

Di Lorenzo, E., Schneider, N., Cobb, K. M., Chhak, K., Franks, P. J. S., Miller, A. J., McWilliams, J. C., Bograd, S. J., Arango, H., Curchister, E., Powell, T. M., and Rivere, P. (2008). North Pacific Gyre Oscillation links ocean climate and ecosystem change. *Geophysical Research Letters*, 35(L08607). doi:10.1029/2007GL032838.

Donlon, C., Rayner, N., Robinson, I., Poulter, D. J. S., Casey, K. S., Vazquez-Cuervo, J., Armstrong, E., Bingham, A., Arino, O., Gentemann, C., May, D., LeBorgne, P., Piollé, J., Barton, I., Beggs, H., Merchant, C. J., Heinz, S., Harris, A., Wick, G., Emery, B., Minnett, P., Evans, R., Llewellyn-Jones, D., Mutlow, C., Reynolds, R. W., and Kawamura, H. (2007). The global ocean data assimilation experiment high-resolution sea surface temperature pilot project. *Bulletin of the American Meteorological Society*, 88(8):1197–1213.

Gelman A., Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences (with discussion). statistical science, 7, 457-472.

Giorgi, F. and Mearns, L. O. (2002). Calculation of average, uncertainty range, and reliability of regional climate changes from AOGCM simulations via the "reliability endemble averaging" (rea) method. *Journal of Climate*, 15:1141–1158.

Hannachi, A., Jolliffe, I. T., and Stephenson, D. B. (2007). Empirical orthogonal functions and related techniques in atmospheric science: A review. *Int J Climatolo*, 27:1119–1152.

Higdon, D. (1998). A process-convolution approach to modelling temperatures in the north atlantic ocean. *Environmental and Ecological Statistics*, 5(2):173–190.

Higdon, D. (2002). Space and space-time modeling using process convolutions. In Anderson, C., Barnett, V., Chatwin, P., and El-Shaarawi, A., editors, *Quantitative Methods for Current Environmental Issues*, pages 37–56. Springer London.

Huerta, G., Sansó, B., and Stroud, J. R. (2004). A spatiotemporal model for mexico city ozone levels. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 53(2):231–248.

Jolliffe, I. (2002). *Principal component analysis*. Springer series in statistics. Springer-Verlag.

Kern, J. (2000). *Bayesian Process-convolution Approaches to Specifying Spatial Dependence Structure*. Duke University.

Knutti, R., Furrer, R., Tebaldi, C., Cermak, J., and Meehl, G. A. (2010). Challenges in combining projections from multiple climate models. *Journal of Climate*, 23(10):2739–2758.

Lemos, R. (2010). *Hierarchical Bayesian Methods for the Marine Sciences: Analyses of Climate Variability and Fish Abundance*. PhD dissertation, Universidade de Lisboa.

Lemos, R. T. and Sansó, B. (2009). A spatio-temporal model for mean, anomaly, and trend fields of north atlantic sea surface temperature. *Journal of the American Statistical Association*, 104(485):5–18.

Mantua, N. and Hare, S. (2002). The Pacific Decadal Oscillation. *Journal of Oceanography*, 58(1):35–44.

Mantua, N., Hare, S., Zhang, Y., Wallace, J. M., and Francis, R. C. (1997). The Pacific interdecadal climateoscillation with impacts on salmon production. *American Meteorological Society,*, 78(6):1069–1079.

Meehl, G., Covey, C., Delworth, T., Latif, M., McAvaney, B., Mitchell, J. F. B., Stouffer, R. J., and Taylor, K. E. (2007). The WCRP CMIP3 multi-model dataset: A new era in climate change research. *Bulletin of the American Meteorological Society*, 88:1383–1394.

Mühlbach, G. and Gasca, M. (1985). A generalization of Sylvester's identity on determinants and some applications. *Linear Algebra and its Applications*, 66(0):221 – 234.

Nakicenovic, N. and Swart, R. (2000). *IPCC Summary for Policymakers: Emissions Scenarios*. Cambridge University Press, The Edinburgh Building Shaftesbury Road, Cambridge CB2 2RU ENGLAND.

Randall, D., Wood, R., Bony, S., Colman, R., Fichefet, T., Fyfe, J., Kattsov, V., Pitman, A., Shukla, J., Srinivasan, J., Stouffer, R., Sumi, A., and Taylor, K. (2007).

Cilmate Models and Their Evaluation. In: Climate Change 2007: The Physical Science Basis Contribution of Working Group I to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change [Solomon, S., D. Qin, M. Manning, Z. Chen, M. Marquis, K.B. Averyt, M.Tignor and H.L. Miller (eds.)]. Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA.

Rayner, N. A., Parker, D. E., Horton, E. B., Folland, C. K., Alexander, L. V., Rowell, D. P., Kent, E. C., and Kaplan, A. (2003). Global analyses of sea surface temperature, sea ice, and night marine air temperature since the late nineteenth century. *J. Geophys. Res.*, 108(D14). doi:10.1029/2002JD002670.

Sansó, B. and Guenni, L. (2004). A Bayesian approach to compare observed rainfall data to deterministic simulations. *Environmetrics*, 15:597–612.

Spiegelhalter, D., Best, N., Carlin, B., and van der Linde, A. (2002). Bayesian measures of model complexity and fit (with discussion). *Journal of the Royal Statistical Society, B*, **64**:583–639.

Storch, H. V. and Cambridge, F. W. Z. (1999). Statistical analysis in climate research.

Tebaldi, C., Smith, R., Nychka, D., and Mearns, L. (2005). Quantifying uncertainty in projections of regional climate change: A Bayesian approach to the analysis of multi-model ensembles. *Journal of Climate*, 18(10):1524–1540.

Thompson, D. W. J. and Wallace, J. M. (1998). The Arctic Oscillation signature in the

wintertime geopotential height and temperature fields. *Geophysical Research Letters*, 25(9):1297–1300.

Ward, T. M., McLeay, L. J., Dimmlich, W. F., Rogers, P. J., McClatchie, S., Matthews, R., Kämpf, J., and van Ruth, P. D. (2006). Pelagic ecology of a northern boundary current system: effects of upwelling on the production and distribution of sardine (Sardinops sagax), anchovy (Engraulis australis) and southern bluefin tuna (Thunnus maccoyii) in the Great Australian Bight. *Fisheries Oceanography*, 15(3):191–207.

West, M. and Harrison, P. (1997). *Bayesian Forecasting and Dynamic Models*. Springer-Verlag, New York, second edition.

Yaglom, A. (1986). *Correlation Theory of Stationary and Related Random Functions I: Basic Results*. Springer Series in Statistics. Springer-Verlag, New York.