NOAA Technical Memorandum ERL PMEL-65

# SIMULATION DATA SETS FOR TESTING MOS (MODEL OUTPUT STATISTICS) PREDICTION METHODS

Rudolph W. Preisendorfer

Pacific Marine Environmental Laboratory
Seattle, Washington
October 1985

**UNITED STATES DEPARTMENT OF COMMERCE**

**Malcolm Baldrige, Secretary**

NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION

Anthony J. Calio, Administrator

Environmental Research Laboratories

Vernon E. Derr, Director

## NOTICE

Mention of a commercial company or product does not constitute an endorsement by NOAA/ERL. Use of information from this publication concerning proprietary products or the tests of such products for publicity or advertising purposes is not authorized.

Contribution No. 811 from Pacific Marine Environmental Laboratory

CONTENTS

# SIMULATION DATA SETS FOR TESTING MOS (MODEL OUTPUT STATISTICS) PREDICTION METHODS

Rudolph W. Preisendorfer

ABSTRACT. It is shown how to construct a set of time series with prescribed autocorrelations and cross correlations which will serve as simulators of real fields drawn from Nature for use in testing methods of prediction. These constructions are applied in particular to the problem of evaluating the skill of prediction methods in the context of a model output statistics (MOS) framework. In an MOS framework there are three main ingredients: (i) a model (e.g. an atmospheric general circulation model) that produces a set of predictor time series called the *model predictors*, (ii) an observer (a human or an instrument) that produces the *observed predictand*, and (iii) a prediction method (usually a set of statistical algorithms) that forecasts the predictand given the predictors. The distinguishing feature of an MOS framework is that the observed predictands are not in the list of outputs of the model. It therefore falls upon the prediction method to link the model predictors and the observed predictand during a training period for the method. Then, when fresh realizations of the predictors are produced by the model, the prediction method will produce its forecasts of the predictand. The skill of the prediction method is determined by comparing its forecasts with subsequent estimates of the predictand by the observer. The model and observer in (i) and (ii) above are generally imperfect and these imperfections find their way into the skill scores of the prediction method--itself an imperfect instrument in practice. To help sort out the various contributions of these three types of imperfection to the final skill scores of the prediction method, we use the time series construction techniques mentioned above to produce controlled simulators of the real predictor and real predictand fields, and we also develop controlled simulators of the fields produced by the model and observer. Hence in the simulation of an MOS prediction setting there are basically these four fields to generate and interrelate. In such a controlled experimental setting a prediction method's inherent and apparent skills and its robustness to changing prediction conditions may be measured and studied.

## 1. INTRODUCTION

### A. Purpose

In this note we outline a procedure to generate data sets for the purpose of determining the skills of various MOS (model output statistics) prediction methods. These data sets have controllable parameters so that the prediction methods are set to their tasks under known and controllable conditions of predictability and reproducibility of results. The distractions of incomplete and uncertain real-data sets are momentarily set aside so that we may discern the inherent and apparent skills (to be defined below) of each prediction method, along with the stability of these skills and the robustness of the prediction methods. In the present note we report only on the theoretical basis of the simulation of the data sets.

## B. Background

The present work is based on some research for the Naval Prediction Research Facility (NEPRF) at Monterey, California. It was required to set up a procedure whereby various MOS prediction methods were to be evaluated in a systematic way prior to full operational use in conjunction with the Naval Operational Global Atmospheric Prediction System (NOGAPS). The latter provided the 'model' in the Model Output Statistics program. Research support was provided in part by NEPRF through Prof. Robert Renard, Chairman of the Meteorology Department of the Naval Postgraduate School, Monterey, CA.* The work was begun during the author's tenure at the School as the 1983 holder of the Haltiner Research Chair in Meteorology.

A history of the MOS applications to weather prediction may be reconstructed from the references found in the bibliography by Carter and Dallavalle (1985).

* The MOS aspect of this work was funded by the Naval Environmental Prediction Research Facility, Monterey, California, under Program Element 63207N, project 7W0513: 'Model Output Statistics'. The development of the simulation model of Appendixes A, B and C was done under the auspices of a TOGA (Tropical Ocean, Global Atmosphere) Grant from the U.S. TOGA Project Office, NOAA headquarters, Rockville, Maryland.

Here at PMEL, word processor assistance was provided by Ryan Whitney and the diagrams were made by Gini Curl.

## 2. THE SIMULATION PROBLEM IN THE MOS SETTING

The problem of designing simulation (or artificial) data sets for MOS prediction methods is a relatively interesting one, for it requires a mix of two simulated natural fields, a modeled field, and a field estimated by an observer. In particular it requires the simulation of four main data sets: (i) primary real fields (e.g., winds, temperatures, pressures), (ii) the modeled versions of these primary real fields, (iii) secondary real fields (e.g., visibility, fog, cloud height), and finally, (iv) an estimate of the secondary real fields by an observer (a human or an instrument). The secondary real fields are, by definition, those not in the output menu of the MOS model. Once these four fields are in place we allow a prediction method (e.g., conditional probability method, discriminant method, or linear regression) to enter the scene and to be trained on part of the set of modeled primary fields (the predictors) and estimated secondary fields (the predictands). Then the prediction method is tested on another portion of the set of simulated predictors and predictands, saved just for that purpose, and prediction skill scores are accumulated and interpreted.

In the present simulation setting the model of the primary real field takes the place of an actual general circulation model such as the Naval Operational Global Atmospheric Prediction System (NOGAPS) at NPS, Monterey, CA, while the observer in this study simulates, for example, someone on the deck of a ship estimating a secondary field (say) atmospheric visibility. In setting up an actual MOS prediction method, i.e., to train it to forecast some predictand (say, visibility), the observations of the predictand must be coupled with a set of predictors supplied by the NOGAPS model (say, winds, temperatures, or pressures). When the NOGAPS model is subsequently used, it will produce a fresh set of predictors. The prediction method is applied to these predictors to produce its forecast. In attempting to emulate this activity in an artificial data set we must construct a simulation of the NOGAPS model fields as well as construct a simulation of the primary real fields it is trying to model. Moreover, we must simulate an observer estimating some aspect (visibility, say) of a secondary real field, as well as simulate the secondary field itself. In the construction of the artificial data set we therefore have the opportunity not only to build in the inherent correlations between the real predictors--the fields coming from Nature--but also to build in the errors of the observer viewing the effects of these fields (e.g., visibility) and also the errors of the NOGAPS in its attempts to model the real predictors.

To see this in more detail, consider Fig. 1 which summarizes the MOS concept diagrammatically. The real atmosphere and oceans are the ones given to us by Nature and are denoted by the top box of the left column. In the real atmosphere and oceans, all fields (primary or secondary) and all their interconnections are known--if only to the Creator. The real primary fields are the ones traditionally we have come to incorporate in the laws of motion, such as the Newtonian equations of motion of the fluids comprising the oceans and atmosphere, including the first and second laws of thermodynamics. The real secondary fields are largely those that interest navigators of the sea or the air above it. A representative but not exhaustive list of such real secondary fields is displayed in the left hand column of Fig. 1. While many of the real secondary atmospheric fields are in principle obtainable from the real primary fields by special physical theories, they usually are not in the

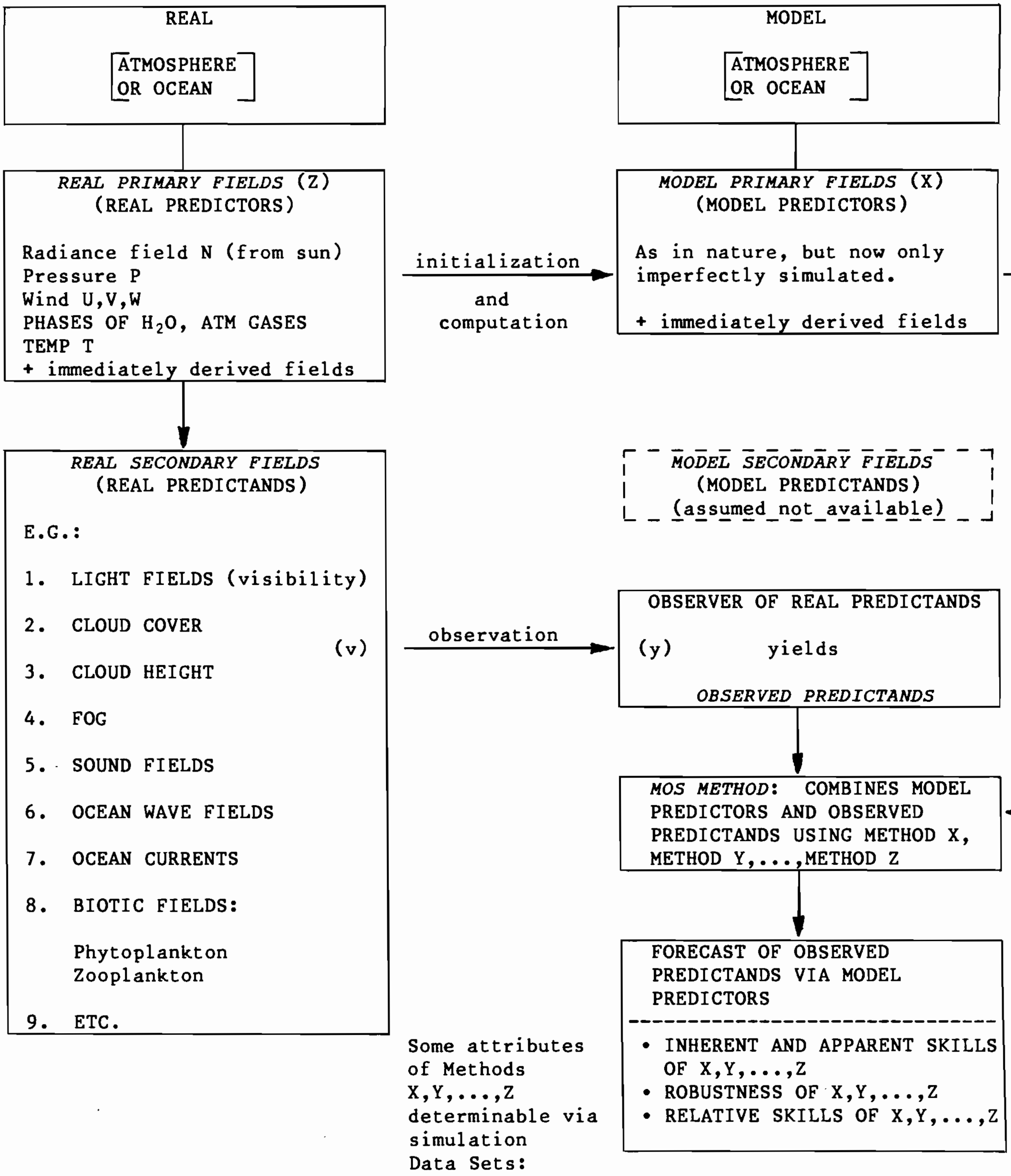

REAL
ATMOSPHERE OR OCEAN
MODEL
ATMOSPHERE OR OCEAN
*REAL PRIMARY FIELDS* (Z)
(REAL PREDICTORS)
Radiance field N (from sun)
Pressure P
Wind U,V,W
PHASES OF $H_2O$, ATM GASES
TEMP T
+ immediately derived fields
initialization
and
computation
*MODEL PRIMARY FIELDS* (X)
(MODEL PREDICTORS)
As in nature, but now only imperfectly simulated.
+ immediately derived fields
*REAL SECONDARY FIELDS*
(REAL PREDICTANDS)
E.G.:
1. LIGHT FIELDS (visibility)
2. CLOUD COVER
(v)
3. CLOUD HEIGHT
4. FOG
5. SOUND FIELDS
6. OCEAN WAVE FIELDS
7. OCEAN CURRENTS
8. BIOTIC FIELDS:
Phytoplankton
Zooplankton
9. ETC.
*MODEL SECONDARY FIELDS*
(MODEL PREDICTANDS)
(assumed not available)
observation
OBSERVER OF REAL PREDICTANDS
(y) yields
*OBSERVED PREDICTANDS*
*MOS METHOD*: COMBINES MODEL PREDICTORS AND OBSERVED PREDICTANDS USING METHOD X, METHOD Y,...,METHOD Z
FORECAST OF OBSERVED PREDICTANDS VIA MODEL PREDICTORS
• INHERENT AND APPARENT SKILLS OF X,Y,...,Z
• ROBUSTNESS OF X,Y,...,Z
• RELATIVE SKILLS OF X,Y,...,Z
Some attributes of Methods X,Y,...,Z determinable via simulation Data Sets:

Figure 1.--The general concept of the MOS (Model Output Statistics) approach to forecasting.

derived products generated by NOGAPS or other present-day theoretical atmospheric systems. These latter systems are therefore *models*, not only because many of the needed secondary fields are missing from their outputs, but also because the primary fields and the derived products they do generate are usually more or less in error.

## 3. THE GENERAL MOS PREDICTION CONCEPT AND SOME SPECIAL CASES

We describe first the so-called *standard prog* form of the general MOS concept. Then we shall make an overview of the general MOS concept and illustrate it with several of its other special cases.

### A. Standard Prog

By definition, in the present study, an MOS prediction method is concerned only with forecasting those real secondary fields that are not in the output menu of the model. The general approach of MOS methods to the atmospheric (or oceanographic) forecasting problem is depicted in the right-hand column of Fig. 1. An observer records some estimate of a real secondary field of interest (say marine atmosphere visibility). It is assumed that this real secondary field (the real predictand) is not available as a model secondary field. Hence we settle for an estimated version of the real secondary field of interest over some given region. This estimate is the one to be predicted; it is called the *observed predictand*. It is usual to assume that the real secondary field is constant over this region. The region is taken small enough so that this is a reasonable assumption. It is also assumed that the predicted model primary fields (of NOGAPS, say) have been recorded at various locations of the given region and at the same times when the real secondary field was observed. The model primary fields are called the *model predictors* for this region. Such model-predictor/observed-predictand pairings of fields is done over the geographic region of interest at selected times t+τ with a chosen lag τ between the present (t) and the future (t+τ). This produces a *Basic Data Set* of predictor/predictand pairs over the region for lag τ measured from time t. The lag times allow pairings of the recorded model predictors at times t+τ, i.e., τ hours (say) in the future of t with the values of the observed predictands also at times t+τ. This particular uniting of predictor and predictand pairs defines the *standard prog* form of the general MOS prediction concept. (We will place the standard prog form into perspective, below, when making the overview of the general MOS prediction concept.)

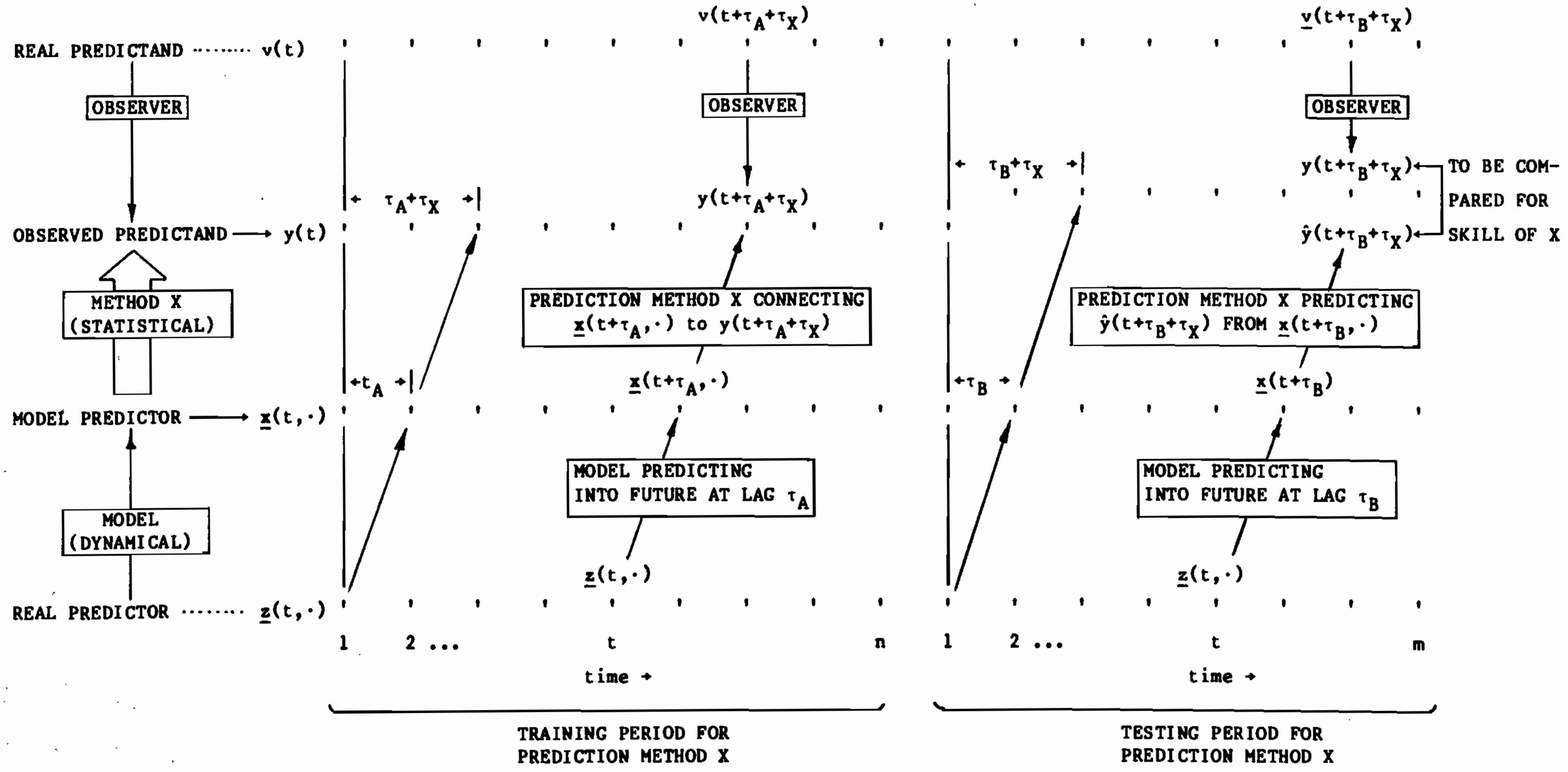
An MOS prediction method (X,Y,..., or Z) is then trained on part of the Basic Data Set to decide on a value of the predictand field at time t+τ (say, visibility) given a set of one or more values of the model predictor fields predicted by the model for t+τ from analyzed real predictors obtained at time t (the present). This method is tested on the remaining part of the predictor/predictand Basic Data Set as follows: A set of model predictor values is chosen from the testing-part of the Basic Data Set, using the predictors determined during the training stage. The method produces a forecast of visibility (say) from this set. The forecast is compared with the observed predictand (visibility) that was paired in the Basic Data Set with this set of predictor values. As a result of the comparison, a skill score is assigned to the forecast of the method. This is done for all the prediction methods (X,Y,...,Z) under study. The various methods (X,Y,...,Z) are intercompared using these scores, and some assessment of their relative skills is made.

The assumption underlying this testing activity is that these relative standings of the prediction methods (X,Y,...,Z) under simulated conditions will be indicative of their relative standings when the model and the associated prediction methods (X,Y,...,Z) are put to work in the real world (i.e., when using non-simulated data sets). This assumption will be well-founded provided that: (i) the simulation data set is representative (in a statistical sense) of the real primary fields and also (ii) representative of the real secondary fields encountered in Nature as well; (iii) the errors of the model (e.g., NOGAPS) have been well-simulated; and (iv) the errors of the observer have been well-simulated also. We shall consider these matters below, in turn. For the moment we pause to make an overview of the general MOS prediction concept.

### B. General Prog and Some Special Cases

In figure 2 we indicate the *general MOS concept* leading to the *general prog*. On the left margin of the diagram we see the four basic data sets displayed: the sets of real predictors, model predictors, real predictands and observed predictands. The model and observer, as explained above, relate the real fields to the model-predictor/observed-predictand pairs, and the latter pairs are coupled statistically by the prediction method (say X). Along the bottom axis of the diagram is displayed the time domain, partitioned into a training period and a testing (or application) period. During the training period the p model predictor fields $\underline{x}(t+\tau_A,\cdot) = [x(t+\tau_A,1), \cdots, x(t+\tau_A,p)]^T$, produced by the model for a lag $\tau_A$ into the future beyond the present time t, are paired with the observed predictand values $y(t+\tau_A+\tau_x)$. Here '$\tau_X$' denotes an extra lag into the future beyond $t+\tau_A$ produced with the aid of prediction method X. In general, the model and the method each can project its given field into the future: the model projects $\underline{z}(t,\cdot) = [z(t,1),...,z(t,p)]^T$ into $\underline{x}(t+\tau_A,\cdot)$ and the method links $\underline{x}(t+\tau_A,\cdot)$ with $y(t+\tau_A+\tau_X)$ during the training period. During the testing period the model, having some flexibility as a general circulation model, can project

THE USE OF TIME LAGS $\tau_A$, $\tau_X$, $\tau_B$ TO DEFINE THE GENERAL MODEL OUTPUT STATISTICS APPROACH TO PREDICTION, AND SOME SPECIAL CASES


REAL PREDICTAND ........ v(t)
OBSERVER
OBSERVED PREDICTAND → y(t)
METHOD X (STATISTICAL)
MODEL PREDICTOR → x(t,·)
MODEL (DYNAMICAL)
REAL PREDICTOR ........ z(t,·)
τA+τX
τA
v(t+τA+τX)
OBSERVER
y(t+τA+τX)
PREDICTION METHOD X CONNECTING x(t+τA,·) to y(t+τA+τX)
x(t+τA,·)
MODEL PREDICTING INTO FUTURE AT LAG τA
z(t,·)
1
2 ...
t
n
time →
TRAINING PERIOD FOR PREDICTION METHOD X
τB+τX
τB
v(t+τB+τX)
OBSERVER
y(t+τB+τX)
ŷ(t+τB+τX)
TO BE COMPARED FOR SKILL OF X
PREDICTION METHOD X PREDICTING ŷ(t+τB+τX) FROM x(t+τB,·)
x(t+τB)
MODEL PREDICTING INTO FUTURE AT LAG τB
z(t,·)
1
2 ...
t
m
time →
TESTING PERIOD FOR PREDICTION METHOD X

DYNAMICAL PROG: $\tau_A > 0$, $\tau_X = 0$, $\tau_B > 0$

STATISTICAL PROG: $\tau_A = 0$, $\tau_X > 0$, $\tau_B = 0$

STATISTICAL/DYNAMICAL PROG: $\tau_A > 0$, $\tau_X > 0$, $\tau_B > 0$

STANDARD PROG: $\tau_A = \tau$, $\tau_X = 0$, $\tau_B = \tau > 0$

PERFECT PROG: $\tau_A = 0$, $\tau_X = 0$, $\tau_B > 0$

(General Prog)

Figure 2.

$\underline{z}(t,\cdot)$ to $\underline{x}(t+\tau_B,\cdot)$, where $\tau_B$ is generally different from $\tau_A$. However, the statistical method is considered here as less flexible, and during the testing period we must retain the same lag $\tau_X$ used in the training period. Therefore the prediction method predicts $\hat{y}(t+\tau_B+\tau_X)$ as an estimate of the observed predictand $y(t+\tau_B+\tau_X)$. Then $y(t+\tau_B+\tau_X)$ and $\hat{y}(t+\tau_B+\tau_X)$ are compared, and skill scores are recorded and interpreted, as usual.

Several special cases of general prog are of interest. First of all, the *standard prog*, as described in par. A above, is the case where $\tau_A = \tau_B = \tau > 0$ and $\tau_X = 0$. In standard prog, therefore, the model's lag into the future is fixed and positive during both the training and testing periods. The prediction method's lag $\tau_X$ in standard prog is set for zero. Thus the prediction method's role in standard prog is to produce a link between $\underline{x}(t+\tau,\cdot)$ and $y(t+\tau)$, i.e., between two contemporaneous fields. In this sense the method X serves more as a means of *specifying* $y(t+\tau)$ as a (statistical) function of $\underline{x}(t+\tau,\cdot)$ rather than as a true *prediction* of $\underline{x}(t+\tau,\cdot)$.

Another common variant of general prog is the so-called *perfect prog* wherein $\tau_A = 0$, $\tau_X = 0$ and $\tau_B > 0$. In this case, during the training period, we use the method X to link $\underline{x}(t,\cdot)$ with $y(t)$. The field $\underline{x}(t,\cdot)$, with time t thought of as the present, is sometimes called the *initialized field* of the model, or alternately, the *analyzed field* version of the primary field $\underline{z}(t,\cdot)$. The latter is the raw field coming in from Nature observed at time t; $\underline{x}(t,\cdot)$ is $\underline{z}(t,\cdot)$'s smoothed, gap-filled, and generally objectively analyzed version; and $\underline{x}(t+\tau_B,\cdot)$, used during the testing period, is computed by the model with $\underline{x}(t,\cdot)$ as an initial value.

Both standard prog and perfect prog are special cases of the prediction scheme called *dynamical prog*, where $\tau_A > 0$, $\tau_X = 0$, $\tau_B > 0$. Once again the burden of prognostication is carried by the dynamical model rather than the prediction method X. This is the antithesis of *statistical prog* for which

$\tau_A = 0$, $\tau_X > 0$, $\tau_B = 0$. Now method X carries the model's analyzed field $\underline{x}(t,\cdot)$ into the future to match up with and attempt to predict $y(t+\tau_X)$.

All of the above schemes are special cases of the general prog form of the MOS idea, which is also called the *statistical/dynamical prog*. For this we generally have all three lags positive: $\tau_A > 0$, $\tau_X > 0$, $\tau_B > 0$.

## 4. SIMULATING THE REAL PRIMARY FIELDS (REAL PREDICTORS)

The theory of simulation of a real primary field $z(t,\xi)$ over a region is given in Appendix A. The main formula is (A4.6). The developments in the Appendix show how to find the various main pieces of $z(t,\xi)$, the $\xi$th field's value in the region at time t, such as the principal component $a_j(t)$ and eigenvector component $e_j(\xi)$. Some physical interpretations of $z(t,\xi)$ are listed in the 'Real Primary Fields' box of Fig. 1. The $z(t,\xi)$ are potentially of great generality and can be used to simulate arbitrary finite sets of fields at each of an arbitrary finite set of points of some region. This is possible by suitably defining the meaning of $\xi$. For example suppose

$$z(t,(\eta,1)),\ z(t,(\eta,2)),\ z(t,(\eta,3))$$

are respectively three atmospheric fields such as air temperature, north-south wind component, and relative humidity at a point $\eta$ in space near the sea surface. $\eta$ itself is an integer over the range $\eta = 1,\ldots,m$ which codifies a latitude and longitude and a distance from mean sea level. The $\xi$ values in the general notation $z(t,\xi)$ are then defined for each pair $(\eta,k)$ via:

$$\xi = 3(\eta-1) + k \quad ; \quad k = 1,2,3 \quad ; \quad \eta = 1,\ldots,m.$$

Therefore, we may include for study all three atmospheric fields at each of the m points in space. In what follows below, for brevity, we will simply write '$\xi$' for $(\eta,k)$, and understand that generally $\eta$ and k can vary over the ranges $\eta = 1,\ldots,m$ and $k = 1,\ldots,\ell$, respectively. Hence as $\xi$ runs from 1 to $p = \ell m$ it can cover all possible $\ell$ real primary fields that we may wish to simulate, at each of a given set of m points in space.

## 5. SIMULATING THE REAL SECONDARY FIELDS (REAL PREDICTANDS)

The real secondary fields of this study are those that are of interest to the navigators of the sea or the air above it. Some examples are depicted in Fig. 1. The variety of these fields is virtually unlimited; they arise under all conditions of visual or auditory search below and above sea level, and for many submarine travels under, sailing conditions on, or flying conditions over the sea. The main characteristic of these real secondary fields is that they are not immediately available through algorithms applied to either the real or modeled primary fields. It may be that such algorithms exist in various subdisciplines of oceanography or meteorology (e.g., radiative transfer theory for visibility; marine acoustic theory for sound fields; surface-wave spectrum theory for sea state) but that the special relations governing such algorithms simply have not yet been incorporated in the derived products of the model primary field; or perhaps it is simply desired to predict the real secondary fields in a new way, say by some novel statistical MOS prediction method. Hence in either case the MOS prediction method must statistically combine the observed predictand field with the model's predictors in some effective way. In order to simulate the real secondary fields (real predictands) we must invent reasonably realistic functional relations between the real predictand $v(t)$ and the real predictors $z(t,\xi)$, $\xi = 1,\ldots,p$ where 't' denotes time and '$\xi$' denotes a predictor index. To fix ideas, suppose that at some point $\eta$ just above the ocean surface, the atmospheric state is given as follows:

v(t) is visibility

z(t,1) is air temperature

z(t,2) is north-south wind component

z(t,3) is relative humidity.

All of these $z(t,\xi)$ values are readily derived from the real primary fields. Then some possibilities for simulating the connections of these three fields $z(t,\xi)$ with v(t) are:

(a) $v(t) = a\, v(t-1) + b|z(t,1)| + c|z(t,2)| + d|z(t,3)|$

(b) $v(t) = a|z(t,1)|^b\, |z(t,2)|^c\, |z(t,3)|^d$

(c) $v(t) = a \exp[-bz(t,1)]/[c+d|z(t,2)| + e|z(t,3)|]$

(d) $w(t) = E\{v(t)\}$

(e) $v(t) = z(t,p+1)$ (see comment, below)

We omit the $\xi$ index from the notation for v(t) because, by assumption, the predictors $z(t,\xi)$ , $\xi = 1,\ldots,p$, are associated with a region of space over which v(t) is independent of location. For example, in (a), on setting a = 1 and b = c = d = 0, we have a simple persistence relation for v(t) over the region of interest. Otherwise, for a = 0, and b,c,d arbitrary, we have a simple linear regression formula. In (b) there is a multiplicative power law between the $z(t,\xi)$ and v(t), all suitably nondimensionalized. This law could be linearized by taking logarithms of each side. In (c) we have a typically nonlinear relation to confound the prediction methods during the simulation studies. Finally, in (d), v(t) could be any of the v(t) in (a)-(c). Here w(t) is the ensemble average of v(t) so that w(t) is the climatology of v(t). In (e), leaving the present example of p = 3, we generalize, and we simply single out the (p+1)st of p+1 real primary fields to act as the real secondary field of interest. The remaining p fields $z(t,\xi)$, $\xi = 1,\ldots,p$, take up the role of the real primary fields (see §6, Appendix B).

The preceding examples merely give the flavor of the simulations of the real secondary fields that could be made. Further examples are likely to be tried as the simulation study progresses.

## 6. SIMULATING THE MODEL PRIMARY FIELDS (MODEL PREDICTORS)

From Fig. 1 we see that the model primary fields attempt to represent their real counterparts. Such attempts are rarely perfect and the result is a somewhat distorted and noisy version of the real primary fields coming out of Nature's mill. Moreover, when it comes to predicting the real primary field's future states, the model reveals still further imperfections. To quantify this aspect of the model, let $x(t,\xi)$ be the model's version of the real primary field $z(t,\xi)$, for $\xi = 1,\ldots,p$, and time t. Then for a given time lag $\tau$ we simulate the model's $\xi$th predicted field value $x(t+\tau,\xi)$ in terms of the $\xi'$th real primary field's future value $z(t+\tau,\xi')$ at $t+\tau$ by:

$$x(t+\tau,\xi) = \sum_{\xi'=1}^{p} s_\tau(\xi,\xi')\, z(t+\tau,\xi') + n(t+\tau,\xi) \qquad (6.1)$$

$$t = 1,\ldots,n \quad ; \quad \xi = 1,\ldots,p$$

We may write this in vector form:

$$\underline{x}(t+\tau) = \underline{S}_\tau\, \underline{z}(t+\tau) + \underline{n}(t+\tau) \qquad (6.2)$$

Here $s_\tau(\xi,\xi')$, $\xi,\xi' = 1,\ldots,p$ are chosen real constants of a linear transformation from the $z(t+\tau,\xi)$'s to the $x(t+\tau,\xi)$'s. Moreover, $n(t+\tau,\xi)$ is the noise which further distorts the attempt to precisely represent the $z(t+\tau,\xi)$. Ideally (i.e., if the model were perfect) we should have

$s_\tau(\xi,\xi') = 0$ for $\xi \neq \xi'$ and 1 for $\xi = \xi'$; and $n(t+\tau,\xi) = 0$ for all t, $\tau$ and $\xi$. In practice a good model would have $s_\tau(\xi,\xi')$ and $n(t+\tau,\xi)$ values close to these ideals. In general, it is natural to expect $s_\tau(\xi,\xi')$ to decrease and $n(t+\tau,\xi)$ to increase with increasing $\tau$, for fixed $\xi,\xi'$. This simulates the decreasing ability of the model to predict the future values of $z(t,\xi)$. One must decide on these matters before undertaking the simulation study (cf. step 4, §8).

## 7. SIMULATING THE OBSERVED REAL SECONDARY FIELDS (OBSERVED PREDICTANDS)

In Fig. 1 the observer of the real secondary fields makes estimates of their magnitudes, and these estimates are combined with the model predictors by the MOS methods X,Y,...,Z to be studied. Here then arises another possible source of error that must be simulated, that produced by an imperfect observer. This error is particularly evident in the case of human observers estimating atmospheric visibility at sea where there may be few visual benchmarks to guide such an estimate. When observers make estimates of visibility, the estimates are usually in terms of a finite set of discrete categories. Suppose, e.g., there are three categories 1, 2, 3 (representing low, medium and high visibility) to be estimated.* When it is the case that category 1 indeed prevails in Nature at some locale and time, the observer may correctly estimate category 1 visibility. However, the observer may also at that locale and time, with some non zero probability, estimate category 2 or category 3. The 3×3 table below summarizes the relative frequency $e_{ij}$ of the observer estimating category i when actually category j exists in Nature.

* Three categories are chosen here to be specific. In general, the theory below can handle any finite number of categories, formable in any of several ways.

Observer-error Table

$e_{ij}$

Estimated category i

$$\begin{array}{|c|c|c|c} \multicolumn{1}{c}{1} & \multicolumn{1}{c}{2} & \multicolumn{1}{c}{3} & \\ \hline e_{11} & e_{12} & e_{13} & 1 \\ \hline e_{21} & e_{22} & e_{23} & 2 \\ \hline e_{31} & e_{32} & e_{33} & 3 \\ \hline \end{array} \qquad (7.1)$$

Actual category j

Here the $e_{ij}$ have been normalized so that each column of (7.1) sums to 1:

$$\sum_{i=1}^{3} e_{ij} = 1 \quad \text{for} \quad j = 1,2,3 \qquad (7.2)$$

The perfect observer would have $e_{ij} = 0$ if $i \neq j$ and $e_{ii} = 1$ for $i = 1,2,3$. A bad observer would have relatively large off-diagonal $e_{ij}$ values.

One uses the observer's error table in the simulation procedure as follows. Suppose that a realization of the real primary field $\{z(t,\xi): t = 1,\ldots,n; \xi = 1,\ldots,p\}$ has been generated over some given region of space. From this, by means of one of the algorithms in (a)-(d) of §5, we generate values of the real predictand $v(t)$, $t = 1,\ldots,n$ that are assumed representative of the real predictand over the given region. Arrange these n values $v(t)$ in ascending order so that, after relabeling, we have

$$v_1 \leq v_2 \leq \cdots \leq v_n \ . \qquad (7.3)$$

Next, group these values into three equally populous category sets $C_1, C_2$, and $C_3$ (or nearly so if n is not divisible by 3)*. $C_1, C_2$, and $C_3$ is then respectively the lowest, middle, and highest visibility class.† Now these $v_j$ values (which are of the form v(t) for some t) are the values of visibility as they actually occur in Nature; i.e., in the context of the present simulation exercise, they come to us by a known algorithm (given by (a)-(d) of §5, e.g.) applied to the simulated primary field. When the observer is confronted with say $v_k$, which happens to actually be in the set $C_j$, the probabilities are $e_{1j}, e_{2j}, e_{3j}$ that the observer will estimate $v_k$ to be in class sets $C_1, C_2, C_3$, respectively. To simulate the observer's estimate of the category to which $v_k$ belongs we randomly choose a uniformly distributed number v on the interval $I = [0,1]$ which has been partitioned into three subintervals

$$\begin{aligned} A_{1j} &= \{u: \quad 0 \le u \le e_{1j}\} \\ A_{2j} &= \{u: \quad e_{1j} < u \le e_{1j} + e_{2j}\} \qquad (7.4) \\ A_{3j} &= \{u: \quad e_{1j} + e_{2j} < u \le 1\} \end{aligned}$$

so that $$I = A_{1j} + A_{2j} + A_{3j} \quad , \quad j = 1,2,3 \qquad (7.5)$$

If the random number v falls, e.g., in $A_{2j}$ then $v_k$ is assigned to category 2 by the observer simulator. In the long run, then, $v_k$, when actually in set

* For example if n is 5, then $[v_1\ v_2], [v_3\ v_4], [v_5] = C_1, C_2, C_3$ is one grouping. If n is 7, then $C_1, C_2, C_3 = [v_1\ v_2], [v_3\ v_4], [v_5\ v_6\ v_7]$ is the associated grouping. In general, the third group $C_3$ takes up the remainder so as to have one more or one less member than its two predecessors. For n on the order of 100, this convention should not materially affect the subsequent results. Otherwise, the odd category $C_i$, that takes up the odd $v_1$ value, is chosen randomly.

† It should be noted that there are other ways of defining categories $C_1, C_2$, and $C_3$. For instance, instead of having them contain the same number of points, they could be constructed to have the same variance about their centroids; or the range of the v(t)'s could be partitioned into categories of equal length.

$C_j$, will be placed with probability $e_{ij}$ in $C_i$, and hence with probability $e_{ij}$ the associated visibility $v(t)$ will be placed in category $i$, $i = 1,2,3$.

## 8. PUTTING IT ALL TOGETHER

When the prediction methods X,Y,...,Z have been selected for evaluation, we construct the Basic Data Set on which they will be tested by going through the procedure outlined below. All prediction methods will then be trained on the same part of this Basic Data Set and tested on the same remaining part of the set. At the outset of the task of constructing the Basic Data Set, several broad decisions should be made. Thus decide whether:

a) the field-to-field correlations $m(\xi,\xi')$ should be high or low (cf. (A2.1) and (B2.1)).

b) the temporally lagged field-to-field correlations $k(\xi,\xi')$ should be high or low (cf. (A2.2) and (B3.1)).

These decisions will simulate the level of determinism (high correlation) or randomness (low correlation) of the real primary field, as encountered by the prediction methods.

Moveover, choose:

c) the algorithm for determining the real secondary field from the real primary field ((a)-(d) §5; cf. §6, Appendix B).

d) the prediction-error level $s_\tau(\xi,\xi')$, $n(t+\tau,\xi)$ in (6.1) of the model in its forecasts of the model primary field (the model predictors) with lag $\tau$ (cf. (B4.1)).

e) the error level (via table (7.1)) of the observer in his estimates of the real secondary field (the real predictand) (cf. §5, Appendix B).

These latter three decisions will fix the inherent difficulty of forecasting the real predictand (choice c), the quality of the model (choice d), and the quality of the observer (choice e). Appendix B has been written to facilitate the five decisions a)-e).

Once these broad decisions have been made, the procedures for constructing the Basic Data Set and for applying and evaluating the prediction methods reduce to the following:

Step 1: Initial Choices

a) Choose the number $p$ of real primary fields $z(t,\xi)$, $\xi = 1,\ldots,p$ (e.g., $p = 9$).

b) Choose the number $n$ of samples of $z(t,\xi)$, $t = 1,\ldots,n$ (e.g., $n = 100$).

c) Choose the field-to-field correlations $m(\xi,\xi')$, $\xi,\xi' = 1,\ldots,p$ with $m(\xi,\xi) = 1$ for $\xi = 1,\ldots,p$. Observe the constraints (A3.4) which hold for $m(\xi,\xi')$, i.e., $|m(\xi,\xi')| \leq 1$. See Appendix B for suggestions on parameterizing $m(\xi,\xi')$. See Appendix C for a package construction of the data set.

d) Determine the field-to-field lagged correlations $k(\xi,\xi')$, $\xi,\xi' = 1,\ldots,p$. There are two options.

Option 1: Choose the temporally lagged field-to-field correlations $k(\xi,\xi')$, $\xi,\xi' = 1,\ldots,p$ with $0 \leq k(\xi,\xi) < 1$ for $\xi = 1,\ldots,p$. Observe the constraint (A3.6), i.e., $|k(\xi,\xi')| < 1$, $\xi,\xi' = 1,\ldots,p$. See Appendix B for suggestions on parameterizing $k(\xi,\xi')$. See Appendix C for a package construction of the data set.

Option 2: Choose autoregressive correlations $\rho_j$ such that $|\rho_j| < 1$, $j = 1,\ldots,p$. If the package construction of Appendix C is chosen, then the $\rho_j$ are automatically determined.

If Option 1 in d) is chosen, then condition $|\rho_j| < 1$, $j = 1,\ldots,p$ should be checked at step 2b below. If Option 2 in d) is chosen, then condition (A3.6) should be checked at step 2b below.

Step 2: Construct the Real Primary Field

a) Starting with the p×p matrix $\underline{M}$ chosen in Step 1, find $\underline{E}$, and $\underline{\Lambda}$ via (A2.6). Check that all $\lambda_j$ are positive.

b) If Option 1 in Step 1 above was used, then find $\rho_j$ via (A2.20). Check that $|\rho_j| < 1$. If check doesn't hold, then redefine the $k(\xi,\xi')$ (generally make them smaller in magnitude).

Generate realizations of $a_j(t)$, $j = 1,\ldots,p$; $t = 1,\ldots,n$ using (A3.1). $\varepsilon_j(t)$ for each j and t is randomly drawn from the normal distribution $N(0,1-\rho_j^2)$ (i.e., of zero mean and variance $1-\rho_j^2$).

If Option 2 in Step 1 above was used, then check that (A3.6) holds using (A2.4) on the scalar level:

$$k(\xi,\xi') = \sum_{j=1}^{p} \rho_j n_j(\xi)\, n_j(\xi') \tag{8.1}$$

$\xi,\xi' = 1,\ldots,p$

c) From $\rho_j$, $a_j(t)$, $e_j(\xi)$ for $\xi,j = 1,\ldots,p$; $t = 1,\ldots,n$, find $z(t,\xi)$ via (A4.6). These $z(t,\xi)$ values form the *real primary fields*.

Step 3: Construct Real Secondary Field and its Category Classes

a) Choose the functional representation of v(t) from among (a)-(d) of §5. Find the *real predictand* values v(t), $t = 1,\ldots,n$. See §6 of Appendix B for the alternative adoption of the In-House Real Secondary Field.

b) Arrange these n values v(t) in ascending order, as in (7.3), and determine the *category classes* $C_1,C_2,C_3$ as described in §7.

Step 4: Construct Model Predictors

a) Choose the $s_\tau(\xi,\xi')$, $\xi,\xi' = 1,\ldots,p$ in (6.1), along with $n(t+\tau,\xi)$. See (B4.1) for a parameterization of $s_\tau(\xi,\xi')$. $n(t+\tau,\xi)$ is a normally distributed variate of zero mean and variance $\sigma_n^2$ which we choose to be independent of $t,\tau$, and $\xi$, for simplicity. The relative sizes of $s_\tau(\xi,\xi')$ and $\sigma_n$ determine the signal to noise ratio. A good model will have small noise parameter $\sigma_n$ relative to $s_\tau(\xi,\xi)$, $\xi = 1,\ldots,p$. The result of the construction will be the set $\{x(t+\tau,\xi)$: $t = 1,\ldots,n$; $\xi = 1,\ldots,p\}$ of *model predictor* values fashioned for lag $\tau$.

Step 5: Construct Observed Predictands

a) Go systematically through v(t), t = 1,...,n of Step 3 and classify each v(t), i.e., determine whether it falls into set $C_1,C_2$ or $C_3$, (cf. (7.3)).

b) Simulate the observer's erroneous classification of the v(t) values by choosing and using the Observer-error Table (7.1). Thus go through the classified v(t) values and make the following simulations: If v(t) is in $C_j$ as determined in a) above, then use a random number generator, as explained in §7, to simulate the observer's estimate of the category to which v(t) belongs. Let y(t) (= 1, 2, or 3) be the resultant category index of the predictand v(t), t = 1,...,n. Then the set of integers $\{y(t)$: $t = 1,\ldots,n\}$ is the set of *observed predictands*. See §5 of Appendix B for a parametric version of the observer-error table.

### Step 6: Construct Basic Data Set for the MOS Methods

a) For lag $\tau$ and each t, pair the observed predictands $y(t+\tau)$ with the model predictors $x(t+\tau,\xi)$, $\xi = 1,\ldots,p$ to form the *Basic Data Set* $D(\tau) = \{[x(t+\tau,1),\ldots,x(t+\tau,p)],\ y(t+\tau)]:\ \ t = 1,\ldots,n\}$.

b) Randomly partition $D(\tau)$ into two subsets $D_{trn}(\tau)$, $D_{tst}(\tau)$ the *training* and *testing* sets, respectively (the number of elements in $D_{trn}$ should, for starters, be about twice that in $D_{tst}$). Let the number of elements of $D_{trn}(\tau)$ be $n_{trn}$ and the number in $D_{tst}(\tau)$ be $n_{tst}$, so that $n = n_{trn} = n_{tst}$.

### Step 7: Application of the Prediction Methods X,Y,...,Z. Skill Scores

a) Train* prediction methods X,Y,...,Z on the set $D_{trn}(\tau)$ of Step 6.

b) Test** prediction methods X,Y,...,Z on the set $D_{tst}(\tau)$ of Step 6. The verification of a forecast is made as follows. Suppose it is time t (= 1,...,n) and that a forecast $\hat{y}(t+\tau)$ has been made by method X. $\hat{y}(t+\tau)$ is an integer, namely 1, 2, or 3. Suppose in actuality the observed predictand value at time $t+\tau$ is $y(t+\tau)$, as found in Step 5, above. The forecast is of *j-class error*, $j = 0,1,2$ if and only if $|\hat{y}(t+\tau) - y(t+\tau)| \equiv j \equiv j_X(t)$. Go through all times t and find $j_X(t)$ for $t = 1,\ldots,n$. Make a tally of the number $n_0$ of 0-class errors, the number $n_1$ of 1-class errors, and the number $n_2$

* By '*train*' is meant the procedure whereby a prediction method is fitted to the data set $D_{trn}(\tau)$. This procedure of course depends on the method.

** By '*test*' is meant the quantitative intercomparison of the forecast and observed predictand values. The *forecast procedure* consists of taking a member $[[x(t+\tau,1),\ldots,x(t+\tau,p)],\ y(t+\tau)]$ from $D_{tst}(\tau)$, and applying the method X to the predictor set $[x(t+\tau,1),\ldots,x(t+\tau,p)]$ to produce the *forecast* $\hat{y}(t+\tau)$, the method's estimate of $y(t+\tau)$. The skill scores defined below will then be a measure of the distance between $\hat{y}(t+\tau)$ and $y(t+\tau)$. The smaller this distance, the better the forecast.

of 2-class errors for method X. Do the same for methods Y,...,Z. The integers $n_0, n_1, n_2$ for each method constitute the (integral) *skill scores* for that method. Clearly $n_0 + n_1 + n_2 = n$. An alternate (fractional) *skill score* is $a_j = n_j/n$, $j = 0,1,2$, so that $a_0 + a_1 + a_2 = 1$.

## 9. INHERENT AND APPARENT SKILLS OF METHODS X,Y,...,Z, AND ROBUSTNESS

a) To place in perspective the scores obtained in Step 7 of §8--i.e., to assess the effects of imperfect models and imperfect observers on the skills of each of the prediction methods X,Y,...,Z--one should perform the following experiments on each method with the same real primary fields generated in Step 2 of §8. Return to Step 4, §8 and construct model predictors that are error-free. That is, for the given $\tau$, set $s_\tau(\xi,\xi') = 1$ for $\xi' = \xi$ and 0 for $\xi' \neq \xi$, along with $n(t+\tau,\xi) = 0$ for all t and $\xi$. The result will be a set of perfectly predicted values $\{x(t+\tau,\xi): t = 1,...,n; \xi = 1,...,p\}$ for lag $\tau$. Next, return to Step 5, §8 and set $e_{ij} = 1$ for $i = j$ and 0 for $i \neq j$. The result will be a set $\{y(t): t = 1,...,n\}$ of observed predictands (in integer form) which agree exactly with the classifications of the v(t) done in Step 3, §8. Then go on to Step 6, §8 and construct the error-free Basic Data Set D(t). When the methods X,Y,...,Z are in turn applied to this Basic Data Set, as in Step 7, §8, we will accumulate skill scores that show the *inherent skills* of each of the prediction methods. These skill scores act as benchmarks for the performance of the prediction methods when less ideal models and observers are encountered.

b) The skills found in the error-distorted experiments of Step 1 to Step 7 of §8 are called *apparent skills*. The differences between the inherent and apparent skills will then be traceable to the effects, of model and observer errors, on the skill scores of each prediction method. In this way we will discover that some prediction methods are more *robust* (skill scores less affected by model and observer errors) than others. Moreover, when comparing the inherent skills of methods X,Y,...,Z, we will gain a view into how these methods cope with the linearities or nonlinearities of the connection between the real predictands and real predictors. At this level we may be led to design a prediction method tailored to handle a specific analytic form of connection, e.g., one of those listed in (a)-(d) of §5. We can then go on to test the method's robustness for MOS forecasting by gradually and systematically introducing model and observer errors.

## 10. RELATIVE SKILLS OF METHODS X,Y,...,Z

The training and testing of the prediction methods, as outlined in Step 7, §8, and in §9, above are all initially to be done for a fixed set of n samples and p fields. It is of interest to study the behavior of the fractional skills $a_0$ and $a_1$ (defined in Step 7, §8) of each method as a function of n and p, holding all other parameters fixed. Since we split n into a training part $n_{trn}$ and a testing part $n_{tst}$ (so that $n = n_{trn} + n_{tst}$) we should also consider, for fixed n, the effect of increasing or decreasing $n_{trn}$ relative to $n_{tst}$. Too small a training set (relative to a fixed n) would likely not adequately set up the prediction method's structure, so that its skills, as brought out on $D_{tst}(\tau)$, may not be high or may be erratic (high scatter on the $a_0a_1$ skill diagram*). Too large a training set (relative to a fixed n) would perhaps leave too small a test set on which to see how well the method works. For each prediction method, the present simulation procedure could be used to find the optimal size of n and the optimal split of n into $n_{trn}$ and $n_{tst}$. Further, for the optimal ratio of $n_{trn}/n_{tst}$, as n increases,

* As examples of the $a_0a_1$ skill diagram, see the figures in Preisendorfer and Mobley (1984). There u and v are the integer correspondents to $a_0$ and $a_1$.

what are the method's limiting average values of $a_0$, and $a_1$? Even more desirable: what are the method's population distributions of $a_0$ and $a_1$ for large n? This last question is of importance in deciding whether or not two methods are significantly different, and particularly in deciding which of the two prediction methods is the better. This is one of the ultimate goals in the problem of designing and intercomparing prediction methods.

## 11. REFERENCES

Carter, G.M., J.P. Dallavalle, 1985: *Applications of the MOS Technique: a Bibliography--No. 4*. Techniques Development Laboratory Office Note 85-12, Techniques Development Laboratory, Office of Systems Development, National Weather Service, National Oceanic and Atmospheric Administration, October 1985.

Preisendorfer, R.W., C.D. Mobley, 1984: Climate Forecast Verifications, United States Mainland, 1984-83, *Mon. Weather Rev., 112*, 809-825.

# APPENDIX A

## Construction of the Simulator of the Real Primary Field

### 1. INTRODUCTION

The present simulator of the real primary field (cf. Fig. 1) is basically statistical in character. *It is built around the concept of a principal eigenvector decomposition of a data set that has principal component time series generated by first order autoregressive processes.* This basic idea can be elaborated by generating principal component time series using higher order autoregressive processes. Moreover, simulated data sets can be generated by using realistic dynamical partial differential equations reduced to coupled systems of ordinary differential equations governing the principal components by means of empirical orthogonal function decompositions of the data sets. In the final stages of this exercise, one can use the real data and the model's output directly. In the present exposition of the basic idea we will keep the discussion to its simplest (and thereby probably its most useful) level, for simulation purposes, by using first order processes.

### 2. SIMULTANEOUS DIAGONALIZATION OF A PAIR OF CORRELATION MATRICES

Given two p×p matrices $\underline{M}$ and $\underline{K}$, with $\underline{M}$ positive definite, which respectively describe the field-to-field correlations and the time lagged field-to-field correlations of a data set $\underline{Z} = \{z(t,\xi): \ t = 1,\ldots,n; \ \xi = 1,\ldots,p\}$. Here 't' denotes time, and '$\xi$' denotes a field. Thus, let $z(t,\xi)$, be a standardized (zero mean unit variance) variate, such that the following sets of numbers are specified at the outset:

$$\mathbb{E}\{z(t,\xi)\; z(t,\xi')\} = m(\xi,\xi') \qquad \text{(entries of } \underline{M}) \tag{A2.1}$$

$$\mathbb{E}\{z(t,\xi)\; z(t-1,\xi') = k(\xi,\xi') \qquad \text{(entries of } \underline{K}) \tag{A2.2}$$

for $t = 1,\ldots,n$; $\xi,\xi' = 1,\ldots,p$. Here $\mathbb{E}$ is the ensemble average operator applied to the random variate $z(\omega;t,\xi)$, where the realization index $\omega$ is suppressed in (A2.1), (A2.2) for brevity. Clearly $\underline{M}$ and $\underline{K}$ are symmetric ($\underline{M}^T = \underline{M}$, $\underline{K}^T = \underline{K}$, where "T" denotes transpose). Positive definiteness of $\underline{M}$ means that its eigenvalues $\lambda_j$ (found below) are all positive. Note that the $m(\xi,\xi')$ are to be independent of $t$, along with the $k(\xi,\xi')$. In practice the entries $m(\xi,\xi')$, $k(\xi,\xi')$ of $\underline{M}$ and $\underline{K}$, $\xi,\xi' = 1,\ldots,p$, are dimensionless and are specified at the outset of the simulation study (step 1, §8). Also see (A3.4), (A3.6), below, for basic constraints on $\underline{M}$ and $\underline{K}$.

We wish to simultaneously diagonalize $\underline{M}$ and $\underline{K}$. This entails finding a common set of orthogonal vectors $\underline{N} \equiv [\underline{n}_1 \cdots \underline{n}_p]$, with $\underline{n}_j \equiv [n_j(1),\ldots,n_j(p)]^T$, $j = 1,\ldots,p$, and a diagonal matrix $\underline{R}$ such that

$$\underline{M} = \sum_{j=1}^{p} \underline{n}_j\underline{n}_j^T = \underline{N}\;\underline{N}^T \qquad (p\times p) \tag{A2.3}$$

$$\underline{K} = \sum_{j=1}^{p} \rho_j\underline{n}_j\underline{n}_j^T = \underline{N}\;\underline{R}\;\underline{N}^T \qquad (p\times p) \tag{A2.4}$$

where

$$\underline{R} = \operatorname{diag}[\rho_1,\ldots,\rho_p] \tag{A2.5}$$

The construction of $\underline{N}$ begins with a diagonalization of $\underline{M}$ by solving the eigenvector problem:

$$\underline{M}\ \underline{E} = \underline{E}\ \underline{\Lambda} \tag{A2.6}$$

where

$$\underline{E} = [\underline{e}_1 \cdots \underline{e}_p] \quad , \qquad (p\times p) \tag{A2.7}$$

$$\underline{e}_j = [e_j(1),\ldots,e_j(p)]^T \quad , \qquad j = 1,\ldots,p$$

and where

$$\underline{\Lambda} = \mathrm{diag}[\lambda_1,\ldots,\lambda_p] \quad , \qquad \lambda_j > 0,\ j = 1,\ldots,p \tag{A2.8}$$

The $\underline{e}_j$ are orthonormal:

$$\underline{E}^T\underline{E} = \underline{E}\ \underline{E}^T = \underline{I}_p \tag{A2.9}$$

where $\underline{I}_p$ is the p×p identity matrix. Thus on the vector level we have:

$$\underline{e}_j^T\underline{e}_k = \delta_{jk} \quad , \qquad j,k = 1,\ldots,p \tag{A2.10}$$

and on the scalar level (A2.9) states:

$$\sum_{\xi=1}^{p} e_j(\xi)\ e_k(\xi) = \delta_{jk} \quad , \qquad j,k = 1,\ldots,p \tag{A2.11}$$

$$\sum_{j=1}^{p} e_j(\xi)\ e_j(\xi') = \delta_{\xi\xi'} \quad , \qquad \xi,\xi' = 1,\ldots,p \tag{A2.12}$$

where $\delta_{jk},\delta_{\xi\xi'}$ are Kronecker deltas. The $\underline{e}_j$ and $\lambda_j$ can be found by standard subroutines. Arrange the $\lambda_j$ in descending order: $\lambda_1 \geq \cdots \geq \lambda_p$. This automatically orders the $\underline{e}_j$. Avoid ambiguity of $\underline{e}_j$ by multiplying $\underline{e}_j$ by -1, if necessary, to make the first non zero component of $\underline{e}_j$ positive, for all $j = 1,\ldots,p$.

Using the property (A2.9) we can rearrange (A2.6) to the form:

$$\underline{M} = \underline{E}\;\underline{\Lambda}\;\underline{E}^T \tag{A2.13}$$

We attain the representation (A2.3) by introducing $\underline{N}$ via:

$$\underline{N} \equiv \underline{E}\;\underline{\Lambda}^{\frac{1}{2}} \tag{A2.14a}$$

from which:

$$\underline{n}_j = \lambda_j^{\frac{1}{2}}\underline{e}_j \tag{A2.14b}$$

where, as before,

$$\underline{n}_j \equiv [n_j(1),\ldots,n_j(p)]^T \quad , \quad j = 1,\ldots,p$$

From this and (A2.13) we find (A2.3):

$$\begin{aligned}\underline{M} = \underline{E}\;\underline{\Lambda}\;\underline{E}^T &= \underline{E}\;\underline{\Lambda}^{\frac{1}{2}}\underline{\Lambda}^{\frac{1}{2}}\underline{E}^T \\ &= \underline{N}\;\underline{N}^T\end{aligned} \tag{A2.15}$$

Clearly the $\underline{n}_j$ are orthogonal by virtue of (A2.10) and (A2.14b). To find $\underline{R}$ in (A2.4) we use the form of $\underline{N}$ as given in (A2.14a) to obtain:

$$\underline{K} = (\underline{E}\;\underline{\Lambda}^{\frac{1}{2}})\;\underline{R}(\underline{E}\;\underline{\Lambda}^{\frac{1}{2}})^T \tag{A2.16}$$

whence, on solving for $\underline{R}$:

$$\underline{R} = \underline{\Lambda}^{-\frac{1}{2}}\underline{E}^{T}\underline{K}\ \underline{E}\ \underline{\Lambda}^{-\frac{1}{2}} \tag{A2.17}$$

Setting

$$\underline{C} \equiv \underline{E}\ \underline{\Lambda}^{-\frac{1}{2}} \tag{A2.18}$$

we find

$$\underline{R} = \underline{C}^{T}\underline{K}\ \underline{C} \tag{A2.19}$$

as the required formula for $\underline{R}$. A convenient form of (A2.17) or (A2.19) for numerical work is:

$$\rho_j = \lambda_j^{-1} \sum_{\xi=1}^{p} \sum_{\xi'=1}^{p} e_j(\xi)\ k(\xi,\xi')\ e_j(\xi') \tag{A2.20}$$

$$j = 1,\dots,p$$

Thus we have found the required orthogonal set $\underline{N} = [\underline{n}_1 \cdots \underline{n}_p]$ of vectors and the diagonal matrix $\underline{R} = \mathrm{diag}[\rho_1,\dots,\rho_n]$ which yield the simultaneous diagonalization of $\underline{M}$ and $\underline{K}$.

## 3. CONSTRUCTING THE AUTOREGRESSIVE PROCESS

We next use the diagonal elements $\rho_j$ of $\underline{R}$ to construct p autoregressive processes $\{a_j(t)\colon\ t\varepsilon J\}$, $j = 1,\dots,p$, where J is the set of integers. The $a_j(t)$ are generated by:

$$a_j(t) = \rho_j a_j(t-1) + \varepsilon_j(t) \quad , \quad |\rho_j| < 1 \tag{A3.1}$$
$$j = 1,\ldots,p \quad ; \quad t\varepsilon J$$

Here $\varepsilon_j(t)$, for each j and t, is by construction a normally distributed random variable of zero mean and, for the moment, with general variance $\sigma_j^2$. In particular the $\varepsilon_j(t)$ are of zero mean and are independent variates in the sense that

$$\mathbf{E}\{\varepsilon_j(t)\} = 0$$
$$\mathbf{E}\{\varepsilon_j(t)\ \varepsilon_k(t')\} = \sigma_j^2\ \delta_{jk}\ \delta_{tt'} \quad , \quad \sigma_j^2 > 0 \tag{A3.2}$$
$$j,k = 1,\ldots,p \quad ; \quad t,t'\varepsilon J$$

The condition $|\rho_j| < 1$, $j = 1,\ldots,p$ is needed for physically realizable autoregressive processes $\{a_j(t)\text{: } t\varepsilon J\}$. In view of (A2.20), we can always choose the $|k(\xi,\xi')|$ small enough so that this condition holds. In this connection, observe that the diagonal elements $m(\xi,\xi)$ of $\underline{M}$ are by construction unity:

$$m(\xi,\xi) = 1 \quad , \quad \xi = 1,\ldots,p \tag{A3.3}$$

This places a constraint on the off-diagonal elements of $\underline{M}$ by means of Schwarz's inequality:

$$|m(\xi,\xi')| = |\mathbf{E}\{z(t,\xi)\ z(t,\xi')\}| \le [\mathbf{E}\{z^2(t,\xi)\}\ \mathbf{E}\{z^2(t,\xi')\}]^{\frac{1}{2}}$$
$$= [m(\xi,\xi)\ m(\xi',\xi')]^{\frac{1}{2}} = 1 \tag{A3.4}$$

for $\xi,\xi' = 1,\ldots,p$.

As for the diagonal elements $k(\xi,\xi)$ of $\underline{K}$, in view of (A2.2), it is physically likely that strict inequality holds:

$$k(\xi,\xi) < 1 \quad , \quad \xi = 1,\ldots,p \tag{A3.5}$$

Then the off diagonal elements $k(\xi,\xi')$ of $\underline{K}$ are constrained by the condition:

$$\begin{aligned} |k(\xi,\xi')| = |E\{z(t,\xi)\ z(t-1,\xi')\}| &\le [E\{z^2(t,\xi)\}\ E\{z^2(t-1,\xi')\}]^{\frac{1}{2}} \\ &= [k(\xi,\xi)\ k(\xi',\xi')]^{\frac{1}{2}} < 1 \end{aligned} \tag{A3.6}$$

for $\xi,\xi' = 1,\ldots,p$.

Under the conditions (A3.2) it follows from (A3.1) that for $j,k=1,\ldots,p$ and all $t\varepsilon J$,

$$E\{a_j(t)\ a_k(t)\} = \frac{\sigma_j^2}{1-\rho_j^2} \cdot \delta_{jk} \tag{A3.7}$$

$$E\{a_j(t)\ a_k(t-1)\} = \rho_j \frac{\sigma_j^2}{1-\rho_j^2} \cdot \delta_{jk} \tag{A3.8}$$

The result (A3.7), e.g., may be arrived at as follows. Rearrange (A3.1) by writing

$$a_j(t) = \rho_j a_j(t-1) + \varepsilon_j(t) = \rho_j[\rho_j a_j(t-2) + \varepsilon_j(t-1)] + \varepsilon_j(t)$$

$$= \rho_j^2 a_j(t-2) + [\rho_j \varepsilon_j(t-1) + \varepsilon_j(t)]$$

$$= \rho_j^3 a_j(t-3) + [\rho_j^2 \varepsilon_j(t-2) + \rho_j \varepsilon_j(t-1) + \varepsilon_j(t)] \tag{A3.9}$$

Continuing this way we arrive at the infinite series representation of $a_j(t)$:

$$a_j(t) = \sum_{\tau=0}^{\infty} \rho_j^{\tau} \, \varepsilon_j(t-\tau) \tag{A3.10}$$

$$j = 1,\ldots,p \quad ; \quad t\varepsilon J$$

where the leading term in (A3.9) converges to zero as $\tau\to\infty$, by virtue of the condition $|\rho_j| < 1$ on $\rho_j$, $j = 1,\ldots,p$. Using (A3.10) we can compute:

$$E\{a_j(t)a_k(t)\} = \sum_{\tau=0}^{\infty} \sum_{\tau'=0}^{\infty} \rho_j^{\tau}\rho_k^{\tau'} \, E\{\varepsilon_j(t-\tau) \, \varepsilon_k(t-\tau')\}$$

$$= \sum_{\tau=0}^{\infty} \sum_{\tau'=0}^{\infty} \rho_j^{\tau}\rho_j^{\tau'} \cdot \sigma_j^2 \, \delta_{jk} \, \delta_{(t-\tau),(t-\tau')}$$

$$= \delta_{jk}\sigma_j^2 \sum_{\tau=0}^{\infty} (\rho_j^2)^{\tau} = \frac{\sigma_j^2}{1-\rho_j^2} \cdot \delta_{jk} \tag{A3.11}$$

as was to be shown. The relation (A3.8) follows in a similar way.

## 4. CONSTRUCTING THE SIMULATOR OF THE REAL PRIMARY FIELD

The properties (A3.7) and (A3.8) of $a_j(t)$ and the representation (A2.3) of $\underline{M}$ suggest the following form of the $\xi$th variable's value $z(t,\xi)$ at time $t\varepsilon J$:

$$z(t,\xi) = \sum_{j=1}^{p} \sigma_j^{-1}[1-\rho_j^2]^{\frac{1}{2}} \, a_j(t) \, n_j(\xi) \tag{A4.1}$$

$$t\varepsilon J \quad ; \quad \xi = 1,\ldots,p$$

The values $z(t,\xi)$ in (A4.1) comprise one version of the simulation of the real primary field. Here $a_j(t)$ is defined in (A3.1) and $n_j(\xi)$ is the $\xi$th component of $\underline{n}_j$ in (A2.14b). We have explicitly included $\sigma_j$ in (A3.2) and (A4.1) to show that, without loss of generality, we may set $\sigma_j^2 = 1-\rho_j^2$, $j = 1,\ldots,p$. This will be seen as a matter of course in establishing (A4.2), below: $m(\xi,\xi')$ and $k(\xi,\xi')$ emerge, regardless of the value of $\sigma_j$.

The resultant data set $\underline{Z} = \{z(t,\xi)\colon\ t\varepsilon J;\ \xi = 1,\ldots,p\}$ has the required correlation properties. Thus, for $\xi,\xi' = 1,\ldots,p$ and for $t\varepsilon J$; using (A3.7) and (A2.3):

$$E\{z(t,\xi)z(t,\xi')\} = E\{[\sum_{j=1}^{p}\sigma_j^{-1}[1-\rho_j^2]^{\frac{1}{2}}a_j(t)\ n_j(\xi)][\sum_{k=1}^{p}\sigma_k^{-1}[1-\rho_k^2]^{\frac{1}{2}}a_k(t)n_k(\xi')]\}$$

$$= \sum_{j=1}^{p}\sum_{k=1}^{p}\sigma_j^{-1}\sigma_k^{-1}[1-\rho_j^2]^{\frac{1}{2}}[1-\rho_k^2]^{\frac{1}{2}}\ E\{a_j(t)a_k(t)\}\ n_j(\xi)\ n_k(\xi')$$

$$= \sum_{j=1}^{p} n_j(\xi)\ n_j(\xi') = m(\xi,\xi') \qquad \text{(A4.2)}$$

Thus the data set $\underline{Z}$ has the required property (A2.1).

In a similar manner, using (A3.8) and (A2.4):

$$E\{z(t,\xi)z(t-1,\xi')\} = E\{[\sum_{j=1}^{p}\sigma_j^{-1}[1-\rho_j^2]^{\frac{1}{2}}a_j(t)n_j(\xi)][\sum_{k=1}^{p}\sigma_k^{-1}[1-\rho_k^2]^{\frac{1}{2}}\ a_k(t-1)n_k(\xi')]\}$$

$$= \sum_{j=1}^{p}\sum_{k=1}^{p}\sigma_j^{-1}\sigma_k^{-1}[1-\rho_j^2]^{\frac{1}{2}}[1-\rho_k^2]^{\frac{1}{2}}\ E\{a_j(t)a_k(t-1)\}\ n_j(\xi)\ n_k(\xi')$$

$$= \sum_{j=1}^{p}\rho_j n_j(\xi)\ n_j(\xi') = k(\xi,\xi') \qquad \text{(A4.3)}$$

which establishes the required property (A2.2).

The reader may have noted that (A4.1) is very close to the general form of the principal component analysis of the data set $\underline{Z} = \{z(t,\xi):\ t = J;\ \xi = 1,\ldots,p\}$, with $\underline{n}_j$, $j = 1,\ldots,p$, the unnormalized eigenvectors of the covariance matrix $\underline{M}$ of the data set, and $\sigma_j^{-1}[1-\rho_j^2]^{\frac{1}{2}}a_j(t)$, the rescaled principal components of the set. In the present development we start with the covariance matrix $\underline{M}$ and construct $\underline{Z}$. This is just the reverse of the usual PCA activity of starting with $\underline{Z}$ and constructing $\underline{M}$. During this reversed procedure, of course, we are constantly guided by the desire to end up with the PCA-type of form (A4.1) of the representation of $\underline{Z}$.

In view of the preceding observations, we may reformulate the construction of the time series $a_j(t)$, $t = 1,\ldots,n$, so that (A4.1) takes on exactly the form of the singular value decomposition of $z(t,\xi)$. We return to (A3.1) and, for each $t \varepsilon J$, draw the $\varepsilon_j(t)$ from $N(0,1-\rho_j^2)$, i.e., a normal population with zero mean and variance $\sigma_j^2 = 1-\rho_j^2$, $j = 1,\ldots,p$. Then the discussion leading to (A3.7), (A3.8) may be repeated to yield

$$E\{a_j(t)\ a_k(t)\} = \delta_{jk} \tag{A4.4}$$

$$E\{a_j(t)\ a_k(t-1)\} = \rho_j\delta_{jk} \tag{A4.5}$$

for $j,k = 1,\ldots,p$, $t\varepsilon J$, and (A4.1) becomes:

$$\boxed{\begin{array}{l} z(t,\xi) = \sum\limits_{j=1}^{p} \lambda_j^{\frac{1}{2}}\ a_j(t)\ e_j(\xi) \\ t\varepsilon J\quad ;\quad \xi = 1,\ldots,p \end{array}} \tag{A4.6}$$

with

$$\sum_{\xi=1}^{p} e_j(\xi)\, e_k(\xi) = \delta_{jk} \quad , \quad j,k = 1,\ldots,p$$

$$\sum_{j=1}^{p} e_j(\xi)\, e_j(\xi') = \delta_{\xi\xi'} \quad , \quad \xi,\xi' = 1,\ldots,p \ . \tag{A4.7}$$

as in (A2.11), (A2.12).

We make one final observation that considerably extends the scope of the preceding theory. By construction $z(t,\xi)$, for each $t$ and $\xi$, is a normally distributed variate of zero mean and unit variance, with the required covariance properties (A4.2), (A4.3). If desired, by rescaling the $e_j(\xi)$, the variances at the $\xi$ points can be made of arbitrary size. This of course will modify $\underline{M}$ and $\underline{K}$ accordingly. Taking this simple idea one step further, the data set $\underline{Z}$ can be transformed so as to change the probability density functions of $z(t,\xi)$ at each $t$ and $\xi$ to other desired continuous functions. By reversing this transformation, we apparently can generate simulation data sets of great versatility. Thus suppose we require a data set $\underline{W} = \{w(t,\xi):\ t = 1,\ldots,n;\ \xi = 1,\ldots,p\}$ to have certain correlation matrices $\underline{M}'$ and $\underline{K}'$, and also require $w(t,\xi)$ at $t$ and $\xi$ to be distributed according to some continuous function $\phi(w)$ on the real line $-\infty < w < \infty$. We then determine, for fixed $t$ and $\xi$, the mapping $z = g(w)$, of the real line into itself, that reshapes $\phi$ into the gaussian function; this can be done with sufficient accuracy by numerical means in all cases. As a result, the $w(t,\xi)$ go over into the normally distributed $z(t,\xi)$ variates under $g$; the associated correlation matrices $\underline{M}$ and $\underline{K}$ as a matter of course can then be computed from the resultant $z(t,\xi)$ variates. We are then at the starting point (A2.1), (A2.2) of our constructions above. In this way, after having found $z(t,\xi)$ in (A4.6), we can then use $w = g^{-1}(z)$ on the $z(t,\xi)$ to generate the desired data set $\underline{W}$.

# APPENDIX B

## Analytic Aids in Parameter Decisions

### 1. INTRODUCTION

To help in the decisions needed to start any simulation exercise, some suggestions are given below for the structures of the $\underline{M}$ and $\underline{K}$ matrices, along with that of the matrix $\underline{S}_\tau$, and the observer-error table. The suggested structures reduce the many decisions to a relatively few, namely those concerning the parameters in the suggested analytic aids. In the case of the p×p matrix $\underline{M}$, e.g., instead of having to decide on the $\frac{1}{2}[p^2-p]+p = \frac{1}{2}p(p+1)$ possibly distinct entries of $\underline{M}$, only one parameter $\mu$ in (B2.1) below need be chosen. Similar savings occur for the $\underline{K}$ and $\underline{S}_\tau$ matrices. It should be emphasized that these analytic aids are not definitive, nor exhaustive. They are merely suggestions that may be followed in some initial explorations of the present simulation strategy. Considerations of other possibilities are encouraged.

### 2. PARAMETRIC FORM OF $\underline{M}$

The matrix $\underline{M}$ in (A2.1) may have its entries $m(\xi,\xi')$ parameterized via:

$$m(\xi,\xi') = \mathrm{sgn}(\xi,\xi')\exp[-\mu|\xi-\xi'|] \quad , \quad \mu > 0 \qquad \text{(B2.1)}$$

$$\xi,\xi' = 1,\ldots,p$$

$$\mathrm{sgn}(\xi,\xi') = \mathrm{sgn}(\xi',\xi) \quad \text{(symmetry constraint)}$$

Here one can set $\mathrm{sgn}(\xi,\xi') = \pm 1$ at the choice of the investigator, except for the above symmetry constraint and when $\xi = \xi'$, for which we set $\mathrm{sgn}(\xi,\xi) = 1$. Large $\mu$ will reduce sharply the correlations between distinct fields, small $\mu$ will keep the correlations larger.

## 3. PARAMETRIC FORM OF $\underline{K}$

The matrix $\underline{K}$ in (A2.2) may have its entries $k(\xi,\xi')$ parameterized via:

$$k(\xi,\xi') = \mathrm{sgn}(\xi,\xi')\cdot\kappa_a \cdot \exp[-\kappa_b|\xi-\xi'|] \qquad \text{(B3.1)}$$

$$\xi,\xi' = 1,\ldots,p \quad ; \quad \kappa_a,\kappa_b > 0 \quad , \quad \kappa_a < 1$$

The parameter $\kappa_a$ controls the general level of the lagged correlations, while $\kappa_b$ governs the inter-field correlation fall-off.

## 4. PARAMETRIC FORM OF $\underline{S}_\tau$ and $\underline{n}$

The matrix $\underline{S}_\tau$ in (6.2) may have its entries $s_\tau(\xi,\xi')$ parameterized by:

$$s_\tau(\xi,\xi') = \exp[-(\alpha\tau + \beta|\xi-\xi'| + \gamma)] \qquad \text{(B4.1)}$$

$$\xi,\xi' = 1,\ldots,p \quad ; \quad \alpha,\beta,\gamma > 0$$

The parameter $\alpha$ controls the temporal decay of predictability of the model predictors. A good model will keep $\alpha$ small. Moreover, parameter $\beta$ controls the cross-talk between model predictors. A good model will keep $\beta$ large. The parameter $\gamma$ gives the level of unpredictability of the model predictors for the case $\tau = 0$, $\xi = \xi'$. A good model will keep $\gamma$ small.

The noise term $n(t+\tau,\xi)$ in (6.2) is distributed as $N(0,\sigma_n^2)$, the variance $\sigma_n^2$ being the adjustable parameter in this case. Small $\sigma_n$ are associated with good models.

## 5. PARAMETRIC FORM OF THE OBSERVER-ERROR TABLE

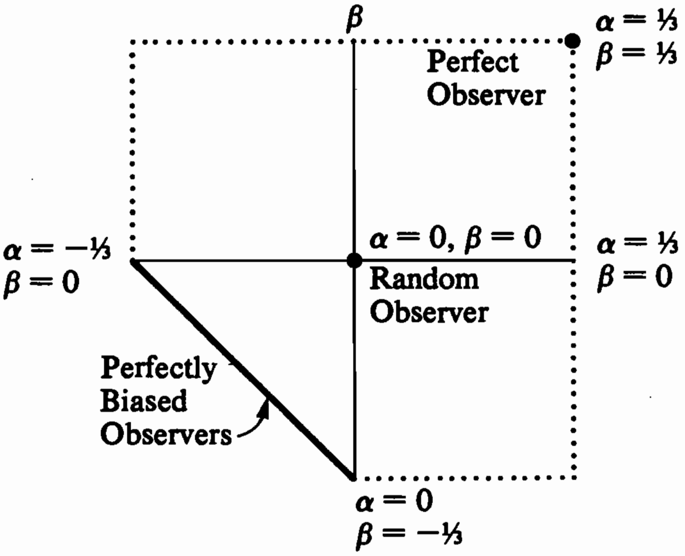
The observer-error table in (7.1) generally has nine entries of which six may vary independently, because of the sum-constraint on the elements of its three columns. Below is suggested a two parameter form of the table, thereby giving a three-fold savings in the number of decisions to be made in constructing it:

| 1 | 2 | 3 | |
|---|---|---|---|
| $\frac{1}{3}+\alpha+\beta$ | $\frac{1}{3}-\alpha$ | $\frac{1}{3}-\alpha$ | 1 |
| $\frac{1}{3}-\alpha$ | $\frac{1}{3}+\alpha+\beta$ | $\frac{1}{3}-\beta$ | 2 |
| $\frac{1}{3}-\beta$ | $\frac{1}{3}-\beta$ | $\frac{1}{3}+\alpha+\beta$ | 3 |

estimated category

actual category

When $\alpha = \beta = \frac{1}{3}$, the observer is perfect: each category is correctly estimated. When $\alpha = \beta = 0$, the observer behaves randomly. The permitted range of $\alpha,\beta$ falls within the truncated square in the $\alpha\beta$-plane, below:



## 6. THE CASE OF THE IN-HOUSE REAL SECONDARY FIELD

In §5 of the main text one of the suggestions made for the real secondary field was to set aside one of the real primary fields to assume the role of a real secondary field. This suggestion has the merit of not having the investigator construct a special functional relation of the forms displayed in §5. If one decides to have such an 'in-house' real secondary field, i.e., one drawn from the family of real primary fields, then it is suggested that p be raised by 1 when going through the initial setups in §7 of the main text. Then choose *any one of the p+1* fields to be the real secondary field, relabel it as 'v(t)', and re-index the remaining fields so that $\xi$ runs from 1 to p. In choosing the candidate for the real secondary field note that it can be fixed to be in the 'middle' of the set of fields or on the 'end' of the set of fields, in the sense of the $\underline{M}$ matrix sketched below (case of p = 5):

$$\underline{M} = \begin{array}{cc} \begin{array}{ccccc} \xi = 1 & 2 & 3 & 4 & 5 \end{array} & \\ \left[\begin{array}{ccccc} 1 & \cdot & \cdot & \cdot & \cdot \\ & 1 & \cdot & \cdot & \cdot \\ & & 1 & \cdot & \cdot \\ & & & 1 & \cdot \\ & & & & 1 \end{array}\right] & \begin{array}{l} 1 = \xi' \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \end{array}$$

For example, z(t,3) is in the middle and so has two closely correlated neighbors z(t,2), z(t,4) (using (B2.1)). On the other hand z(t,1) is on the end and has only one closely correlated neighbor (using (B2.1)). Hence one can control to some extent in this way the relatedness (predictability) of the real secondary field relative to the real primary fields.

# APPENDIX C

## Stationary Real Primary Fields

### 1. INTRODUCTION

Savings in computation effort are possible when adopting stationary real primary fields for the MOS simulation problem. A real primary field is stationary when $\underline{M}$ and $\underline{K}$ have essentially the forms given in (B2.1), (B3.1), i.e., the covariances depend only on the differences $\xi-\xi'$ of the arguments in $m(\xi,\xi')$ and $k(\xi,\xi')$. When such is the case, the eigenvalues and eigenvectors of $\underline{M}$ and $\underline{K}$ are expressible in simple algebraic forms; the need for computer routines to find these quantities is thereby eliminated; and one has only to program simple algebraic formulas for these variables. In the following section the basic relations, ready for programming, are listed, resulting in the present stationary counterpart to (A4.6), the main generator of the simulator of the real primary field. In section 5 below, derivations of the formulas are given.

### 2. THE STATIONARY CONTEXT

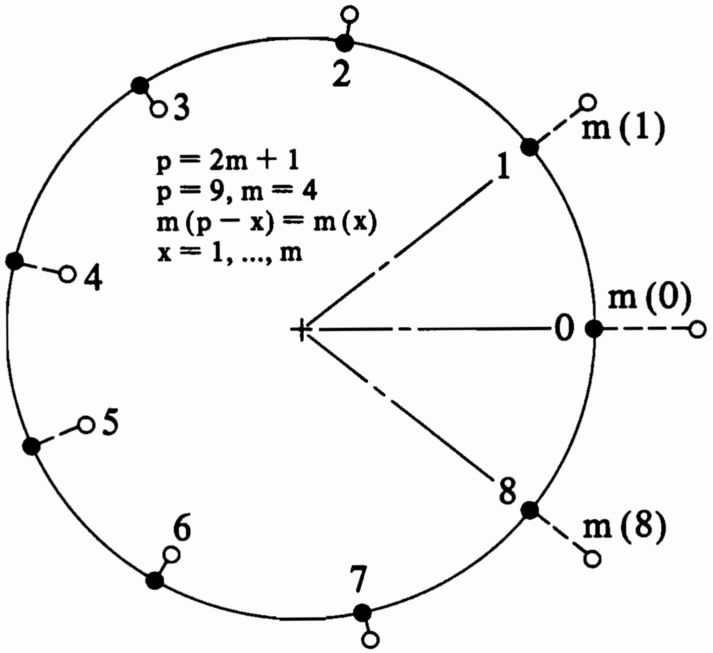
As in the main procedure of Appendix A above, one begins with a choice of the covariance matrices $\underline{M}$ and $\underline{K}$. Thus choose parameters $\mu$, $\kappa_a$, $\kappa_b$ in:

$$m(\xi,\xi') = \mathrm{sgn}(\xi,\xi') \exp[-\mu|\xi-\xi'|] \quad , \quad 0 < \mu < \infty \qquad \text{(C2.1)}$$

$$k(\xi,\xi') = \mathrm{sgn}(\xi,\xi') \kappa_a \exp[-\kappa_b|\xi-\xi'|] \quad , \quad \kappa_a,\kappa_b > 0 \qquad \text{(C2.2)}$$

$$\kappa_a < 1$$

where $\xi,\xi' = 0,\ldots,p-1$ and $\mathrm{sgn}(\xi,\xi')$ is defined as in (B2.1). This range of variables $\xi,\xi'$ is specially tailored for the stationary context. In fact in that context, we must do arithmetic modulo p on ξ,ξ'. Thus if ξ-ξ' is not in the set {ξ: 0,...,p-1}, then ξ-ξ' must be reduced modulo p to place it back in the set. The diagram below shows that this procedure may be visualized as taking place on a circle around which 0,...,p-1 have been plotted. We consider the case of p = 2m+1, where m = 4, so p = 9.



To reduce ξ-ξ' = 3-4 modulo 9, we have 3-4 ≡ 8 modulo 9. (Start at 3 and go clockwise 4 units.) Again: ξ-ξ' = 2-4 ≡ 7 modulo 9. This convention has the effect of extending {ξ: 0,...,p-1} to the set J of all integers: J = {ξ: ...,-1,0,1,...}, which, by the modular arithmetic, is viewable as being wrapped around the circle, modulo p.

By construction, the values of m(ξ,ξ') depend only on ξ-ξ', and indeed, m(ξ,ξ') = m(ξ-ξ',0) ('modulo p'--this qualification will be omitted as understood, henceforth). To simplify notation we will write 'm(x)' for m(x,0); similarly we will write 'k(x)' for k(x,0), x = 0,...,p-1. From now on p = 2m+1, in the discussions of this Appendix.

By adopting the stationary setting we in effect define $m(\xi,\xi')$ for all pairs of integers $\xi,\xi'$. This, coupled with the symmetry $m(\xi,\xi') = m(\xi',\xi)$ of $\underline{M}$, implies an important symmetry of $m(x)$ and $k(x)$ on the finite set $\{x: 0,\ldots,p-1\} = \{x: 0,\ldots,2m\}$. It is easy to see that

$$\begin{aligned} & m(p-x) = m(x) \\ \text{and} \quad & k(p-x) = k(x) \qquad \text{(C2.3)} \\ \text{for} \quad & x = 1,\ldots,m \end{aligned}$$

A diagram of the nine values of $m(x)$ is shown on the circle above. Thus, by arithmetic modulo 9, $m(8) = m(1)$, $m(7) = m(2)$, etc. For example, $m(8,7) = m(1)$, while $m(7,8) = m(-1) = m(8)$. Since $m(7,8) = m(8,7)$ (by symmetry of construction), we have $m(8) = m(1)$.

## 3. LIST OF BASIC FORMULAS

We now list the basic formulas, culminating in the present counterpart to (A4.6). These are listed in order of occurrence in a computation procedure. In what follows, $p = 2m+1$.

The eigenvalues of the p×p covariance matrix $\underline{M}$, are given by:

$$\lambda_j = 1 + 2\sum_{x=1}^{m} m(x)\cos\kappa_j x \qquad \text{(C3.1)}$$
$$j = 0,\ldots,m$$

with

$$\lambda_{p-j} \equiv \lambda_j \quad , \qquad \kappa_j = 2\pi j/p \qquad \text{(C3.2)}$$
$$j = 1,\ldots,m \qquad\qquad j = 0,\ldots,p-1\ (= 2m)$$

For $m(\xi,\xi')$ ($= m(\xi-\xi')$) as in (C2.1), it can be shown that $\lambda_j > 0$ for $j = 0,\ldots,m$.

The autoregressive correlations $\rho_j$ (cf. (A2.20)) are now given by:

$$\rho_j = \lambda_j^{-1}[k(0) + 2\sum_{x=1}^{m} k(x)\cos\kappa_j x] \qquad j = 0,\ldots,m \tag{C3.3}$$

with

$$\rho_{p-j} \equiv \rho_j \qquad j = 1,\ldots,m \tag{C3.4}$$

The variances for the random forcing terms of the autoregressive equations are given by:

$$\sigma^2_{\beta j} = 2(1-\rho_j^2)(1+\delta_{0j}) \tag{C3.5}$$

$$j = 0,\ldots,m$$

$$\sigma^2_{\gamma j} = 2(1-\rho_j^2)(1-\delta_{0j}) \tag{C3.6}$$

where $\delta_{0j}$ is a special case of Kronecker's delta $\delta_{ij}$ with $i = 0$.

The autoregressive equations for the time dependent coefficients $b_j(t)$, $c_j(t)$ are:

$$b_j(t) = \rho_j b_j(t-1) + \beta_j(t) \tag{C3.7}$$

$$c_j(t) = \rho_j c_j(t-1) + \gamma_j(t) \tag{C3.8}$$

$$j = 0,\ldots,m$$

with

$$\beta_j(t) \sim N(0,\sigma^2_{\beta j}) \tag{C3.9}$$

$$\gamma_j(t) \sim N(0,\sigma^2_{\gamma j}) \tag{C3.10}$$

Observe that $\sigma^2_{\beta j} = \sigma^2_{\gamma j}$ for $j = 1,\ldots,m$, and that $\sigma^2_{\beta 0} = 4(1-\rho_0^2)$, while $\sigma^2_{\gamma 0} = 0$ (so $c_0(t) = 0$ for all $t$). Program accordingly.

The main formula is then:

$$\boxed{z(t,\xi) = \tfrac{1}{2}\nu_0 b_0(t) + \sum_{j=1}^{m} \nu_j[b_j(t) \cos\kappa_j\xi + c_j(t) \sin\kappa_j\xi]} \qquad (C3.11)$$

where

$t\varepsilon J$, $\xi = 0,\ldots,p-1$, $\nu_j = [\lambda_j/p]^{\frac{1}{2}}$, $\kappa_j = 2\pi j/p$, $p = 2m+1$,
for $j = 0,\ldots,m$.

## 4. COMPUTATION NOTES

a) The main purpose in adopting the stationary setting is to be able to compute the $\lambda_j$ in (C3.1) directly without using an eigenvector, eigenvalue routine. Thus (C3.1) yields up the $\lambda_j$ with minimum effort. Even this formula may be condensed by algebraically summing the indicated products $m(x) \cos\kappa_j x$ using the adopted form (C2.1) and the symmetry (C3.2). However, this will not be done here.

b) The part of the present computation that needs some extra care is that which generates the random coefficients $b_j(t)$, $c_j(t)$. Towards this end decide on the actual values of $p$ and the number $n$ of times $t$ to be used: $t = 1,\ldots,n$. In this way one can construct a single realization of the data set $\underline{Z} = \{z(t,\xi)\colon\ t = 1,\ldots,n\colon\ \xi = 0,\ldots,p-1\}$. The $\omega$th realization of the set $\underline{Z}$ may be made by randomly constructing the $\omega$th samples of $b_j(t)$ and $c_j(t)$ in (C3.7), (C3.8) for a fixed $t$:

$$b_0(\omega,t),\ b_1(\omega,t),\ldots,b_m(\omega,t)$$

$$c_1(\omega,t),\ldots,c_m(\omega,t)$$

For this t, we then construct

$$z(\omega,t,0),\ldots,z(\omega,t,p-1)$$

using (C3.11), $t = 1,\ldots,n$.

The end result, having done this n times, is the $\omega$th data matrix:

$$\underline{Z}(\omega) = \begin{bmatrix} z(\omega,1,0) & \cdots & z(\omega,1,p-1) \\ z(\omega,2,0) & \cdots & z(\omega,2,p-1) \\ \vdots & & \vdots \\ z(\omega,n,0) & \cdots & z(\omega,n,p-1) \end{bmatrix} \quad (n\times p) \qquad \text{(C4.1)}$$

When finding $b_j(\omega,t)$ and $c_j(\omega,t)$ in (C3.11), one must find $\beta_j(\omega,t)$, $\gamma_j(\omega,t)$ for use in (C3.7), (C3.8). Thus for each $\omega$ and t (= $1,\ldots,n$) the $2m+1$ numbers

$$\beta_0(\omega,t),\ \beta_1(\omega,t),\ldots,\ \beta_m(\omega,t)$$

$$\gamma_1(\omega,t),\ldots,\ \gamma_m(\omega,t)$$

are to be pairwise independent, i.e., freshly chosen from their respective populations in each individual case. So it may be well to select these numbers prior to use by forming, for a given $\omega$, all $(m+1)\cdot n$ of the $\beta_j(\omega,t)$'s and all $m\cdot n$ of the $\gamma_j(\omega,t)$'s. Thus there are $m\cdot n + (m+1)\cdot n = (2m+1)\cdot n = p\cdot n$ random numbers in all needed to form $\underline{Z}(\omega)$ in (C4.6). This is also the same number needed to form the $n\times p$ data set $\underline{Z}$ from (A4.6).

c) It may be advisable to test the statistical properties of the data set $\underline{Z}$ by making up a batch of, say, r realizations $\underline{Z}(\omega)$, $\omega=1,\ldots,r$. The set has been designed so that we have, for each $t=1,\ldots,n$,

$$\underset{\omega}{\mathbb{E}}\{z(\omega,t,\xi)\ z(\omega,t,\xi')\} = m(\xi,\xi') = m(\xi-\xi')$$

Here, for the purpose of the test, we use

$$\frac{1}{r}\sum_{\omega=1}^{r} z(\omega,t,\xi)\ z(\omega,t,\xi') \tag{C4.2}$$

to approximate the ensemble average operation $\underset{\omega}{\mathbb{E}}$. Compare (C4.2) with $m(\xi-\xi')$ (as usual reduce the difference $\xi-\xi'$, modulo p). It may be that r will have to be on the order of 100 to get reasonable agreement.

The number generator should also be tested by seeing if the approximate equality holds:

$$\frac{1}{r}\sum_{\omega=1}^{r} [b_j^2(\omega,t) + c_j^2(\omega,t)] \simeq 4$$

for $j = 0,\ldots,m$, for each t.

## 5. DERIVATION OF MAIN RESULT

We will now derive the basic data set representation (C3.11). Our approach is to represent the random variable $z(t,\xi)$ at each t and $\xi$ in the form of a principal component analysis, as in (C5.5), below. However, our approach is reversed from the usual direction of PC analysis: we begin with the covariance matrix $\underline{M}$ and construct $\underline{Z}$.

a) Having postulated stationary $\underline{M}$ and $\underline{K}$ matrices of order p×p and having adopted the periodic extension of their entry arguments $\xi,\xi'$ to all of J (the set of integers), some results from linear algebra now may be used to show that the jth eigenvector $\underline{e}_j$ of $\underline{M}$, namely

$$\underline{e}_j = [e_j(0),\ldots,e_j(p-1)]^T \quad , \quad j = 0,\ldots,p-1 \tag{C5.1}$$

has components of the form:

$$e_j(\xi) = p^{-\frac{1}{2}} \exp[2\pi i j\xi/p] \tag{C5.2}$$
$$\xi = 0,\ldots,p-1$$

We take this result as given in this study, as it would constitute an unnecessary digression from the main line of the present argument to formulate a proof. The basic eigenstructure equation for $\underline{M}$ is

$$\sum_{\xi'=0}^{p-1} m(\xi,\xi')\, e_j(\xi') = \lambda_j e_j(\xi) \tag{C5.3}$$

Using (C5.2) in this, we can solve for $\lambda_j$ to find

$$\lambda_j = \sum_{x=0}^{p-1} m(x) \exp[2\pi i jx/p] \tag{C5.4}$$

$$x = 0,\ldots,p-1 \quad ; \quad j = 0,\ldots,p-1$$

Using the symmetry (C2.3), this reduces to (C3.1), (C3.2). By linear algebra we know $\lambda_j > 0$, $j = 0,\ldots,p-1$.

b) Since the main purpose of this study is to generate data values $z(\omega,t,\xi)$ with certain covariance properties, we start with the general form for $z(\omega,t,\xi)$ suggested by a principal component synthesis:

$$z(\omega,t,\xi) = \sum_{j=0}^{p-1} \lambda_j^{\frac{1}{2}} \alpha_j(\omega,t)\ e_j(\xi) \tag{C5.5}$$

where $\lambda_j, e_j(\xi)$ are as defined in (C5.2), (C5.4). The desired covariance properties of $z(\omega,t,\xi)$ are obtained by suitably tailoring the random variables $\alpha_j(\omega,t)$. Since $e_j(\xi)$ is complex and $\lambda_j$, $z(\omega,t,\xi)$ are real, the $\alpha_j(\omega,t)$ must be complex. To see the condition to be imposed on the $\alpha_j(\omega,t)$, reform the covariance matrix entries of $\underline{M}$ starting now with (C5.5) ("*" denotes complex conjugation):

$$m(\xi,\xi') \equiv \underset{\omega}{\mathbb{E}}\{z(\omega,t,\xi)\ z^*(\omega,t,\xi')\} \tag{C5.6}$$

$$= \sum_{j=0}^{p-1} \sum_{k=0}^{p-1} \lambda_j^{\frac{1}{2}}\lambda_k^{\frac{1}{2}}\ e_j(\xi)e_k^*(\xi')\ \underset{\omega}{\mathbb{E}}\{\alpha_j(\omega,t)\ \alpha_k^*(\omega,t)\} \tag{C5.7}$$

$$= \sum_{j=0}^{p-1} \lambda_j e_j(\xi)e_j^*(\xi') \tag{C5.8}$$

Observe that this is consistent with (A2.3) and (A2.13). In order to go from (C5.7) to the representation of the matrix entry in (C5.8) we need the property

$$\underset{\omega}{\mathbb{E}}\{\alpha_j(\omega,t)\ \alpha_k^*(\omega,t)\} = \delta_{jk} \tag{C5.9}$$

$$j,k = 0,\dots,p-1$$

In a similar way the requirement (cf. (A2.4)):

$$k(\xi,\xi') = \underset{\omega}{\mathbf{E}}\{z(\omega,t,\xi)\ z^*(\omega,t-1,\xi')\} \tag{C5.10}$$

$$= \sum_{j=0}^{p-1} \sum_{k=0}^{p-1} \lambda_j^{\frac{1}{2}}\lambda_k^{\frac{1}{2}}\ e_j(\xi)\ e_k^*(\xi')\ \underset{\omega}{\mathbf{E}}\{\alpha_j(\omega;t)\ \alpha_k^*(\omega,t-1)\} \tag{C5.11}$$

$$= \sum_{j=0}^{p-1} \rho_j\lambda_j e_j(\xi)e_j^*(\xi') \tag{C5.12}$$

states that we must have for $t\varepsilon J$ and all $\omega$:

$$\underset{\omega}{\mathbf{E}}\{\alpha_j(\omega,t)\alpha_k^*(\omega,t-1)\} = \rho_j\delta_{jk} \tag{C5.13}$$
$$j,k = 0,\ldots,p-1$$

c) The two properties (C5.9), (C5.13) of $\alpha_j(\omega,t)$ can be secured, e.g., by letting $\alpha_j(\omega,t)$ be generated by a first order autoregressive process:

$$\alpha_j(\omega,t) = \rho_j\alpha_j(\omega,t-1) + \varepsilon_j(\omega,t) \tag{C5.14}$$
$$j = 0,\ldots,p-1$$

for $t\varepsilon J$ and all realization indexes $\omega$. We split $\alpha_j(\omega,t)$ into its real and imaginary parts $\frac{1}{2}b_j(\omega,t)$ and $-\frac{1}{2}c_j(\omega,t)$:

$$\alpha_j(\omega,t) = \tfrac{1}{2}[b_j(\omega,t) - ic_j(\omega,t)] \tag{C5.15}$$

and then, following (C5.14), generate $b_j(\omega,t)$, $c_j(\omega,t)$ (on dropping the realization index $\omega$) by means of:

$$b_j(t) = \rho_j b_j(t-1) + \beta_j(t) \tag{C5.16a}$$

$$c_j(t) = \rho_j c_j(t-1) + \gamma_j(t) \tag{C5.16b}$$

with

$$\beta_j(t) \sim N(0,\sigma^2_{\beta j}) \quad , \quad t\varepsilon J \tag{C5.17a}$$

$$\gamma_j(t) \sim N(0,\sigma^2_{\gamma j}) \quad , \quad j = 0,\ldots,p-1 \tag{C5.17b}$$

and where

$$\varepsilon_j(t) = \tfrac{1}{2}[\beta_j(t) - i\gamma_j(t)] \tag{C5.17c}$$

The real valued random variables $\beta_j(t)$, $\gamma_k(t')$ are to be independent when $j \neq k$ or when $t \neq t'$. (The symbol '~' means 'distributed as', and '$N(0,\sigma^2)$' represents a normal distribution of zero mean and variance $\sigma^2$).

To see the requirements on $\sigma^2_{\beta j}$ and $\sigma^2_{\gamma j}$ imposed by (C5.9), we first observe that (C5.14) may be written in the equivalent form

$$\alpha_j(\omega,t) = \sum_{\tau=0}^{\infty} \rho_j^{\tau} \; \varepsilon(\omega,t-\tau) \tag{C5.18}$$

Then we compute

$$\underset{\omega}{E}\{\alpha_j(\omega,t) \; \alpha_k^*(\omega,t)\} = \sum_{\tau=0}^{\infty} \; \sum_{\tau'=0}^{\infty} \rho_j^{\tau}\rho_k^{\tau'} \; \underset{\omega}{E}\{\varepsilon_j(\omega,t-\tau) \; \varepsilon_k^*(\omega,t-\tau')\} \tag{C5.19}$$

By construction of the $\beta_j(\omega,t)$ and $\gamma_j(\omega,t)$ (in particular their independence) (C5.19) reduces to

$$\underset{\omega}{E}\{\alpha_j(\omega,t)\alpha_k^*(\omega,t)\} = \frac{(\sigma^2_{\beta j} + \sigma^2_{\gamma j})}{4(1-\rho_j^2)} \; \delta_{jk} \tag{C5.20}$$

Since the left side must be $\delta_{jk}$, we then arrive at the conditions (C3.5), (C3.6) on the variances. The factors $(1 \pm \delta_{0j})$ come about because we want to end up with (C5.5) looking just like the usual Fourier synthesis formula for a stationary process. Indeed, we construct the first m+1 terms $b_j(t)$, $c_j(t)$ in (C3.7), (C3.8) as shown. Then the remaining m are given by symmetry:

$$\left.\begin{aligned} b_{p-j}(t) &\equiv b_j(t) \\ c_{p-j}(t) &\equiv -c_j(t) \end{aligned}\right\} \quad j = 1,\ldots,m \tag{C5.21}$$

It is easy to check, by following the procedure in (C5.18)-(C5.20), that (C5.13) holds, too.

d) We now may reduce a typical term in (C5.5):

$$\begin{aligned} \lambda_j^{\frac{1}{2}}\alpha_j(t)\ e_j(\xi) &= \tfrac{1}{2}\nu_j[b_j(t) - ic_j(t)][\cos\kappa_j\xi + i\ \sin\kappa_j\xi] \\ &= \tfrac{1}{2}\nu_j[(b_j(t)\ \cos\kappa_j\xi + c_j(t)\ \sin\kappa_j\xi) \\ &\quad + i(b_j(t)\ \sin\kappa_j\xi - c_j(t)\ \cos\kappa_j\xi)] \end{aligned} \tag{C5.22}$$

where $\nu_j = [\lambda_j/p]^{\frac{1}{2}}$, and $\kappa_j = 2\pi j/p$. Observe that the real term in (C5.22) has even functions of j as one goes round the unit circle depicted in §2, above, while the imaginary term has odd functions of j, and so they cancel. Thus we arrive at the basic representation (C3.11) of $z(t,\xi)$, on reducing (C5.5).