The Super Model forecasting method applied to the imperfect model scenario

Carsten Grabow
Potsdam Institute for Climate Impact Research, Potsdam, Germany

Thomas Stemler
School of Mathematics and Statistics, The University of Western Australia, Crawley, Australia

Jürgen Kurths
Potsdam Institute for Climate Impact Research, Potsdam, Germany
Department of Physics, Humboldt University Berlin, Berlin, Germany and
Institute of Complex Systems and Mathematical Biology,
University of Aberdeen, Aberdeen, United Kingdom

Recently a new forecast method that goes beyond the traditional ensemble forecasting was proposed. While traditional ensemble forecasts try to determine the uncertainty of the dynamics by initializing the model with many different states, the so called super-model (SUMO) ensemble method tries to reduce the uncertainty of the forecast by coupling different models and let them exchange information during run-time. As we have seen so far SUMO was evaluated in what we would call the perfect model scenario (PMS), that is the models and the system have structurally the same equations. Therefore the only mismatch between the model and the system tested so far is that the ensemble of models and the system have different coefficients or control parameters. Here [2] we report on the much more realistic imperfect model scenario (IPMS). We assume that in the modeling process there occurred an error and the flow equations governing the models and the systems are therefore different. It is far from obvious that techniques developed for the PMS work in the IPMS as well.

Forecasting or predicting the future dynamics of a complex system is one of the most challenging endeavors we are currently facing as scientists. Whether it is the path of a hurricane, the future climate or the opinion and therefore the behavior of societies, we have to deal with several problems that can be characterized as follows: determining a suitable model of the complex system, estimation of the current state of the system given measurement data and finally using a forecast method that minimizes the uncertainty of the predicted dynamics.

Given that these problems arise almost every time we want to know the future dynamics of a complex system it is not surprising that over the last couple of decades more and more advanced methods to perform these tasks have been suggested and tested. While in most cases it is sufficient to deduce the model from first principles, time series driven models [5], statistical models, and network models become more frequent when dealing with complex systems. Similar in the area of data assimilation, that is the state estimation of the model's state based on observations, we have come a long way since the Wiener filter. Nowadays methods like 4Dvar, Kalman filters and shadowing filters are the state of the art.

Recently a new forecast method that goes beyond the traditional ensemble forecasting was proposed. While traditional ensemble forecasts try to determine the uncertainty of the dynamics by initializing the model with many different states, the so called super-model (SUMO) ensemble method tries to reduce the uncertainty of the forecast by coupling different models and let them exchange information during run-time [4, 6]. The idea behind this approach is that each model hopefully captures some aspects of the true dynamics of the system and therefore the overall quality of the forecast increases if the models exchange this information. Using a training data set the coupling between the different models can be optimized such that only the wanted information is exchanged.

The far aim is obviously to combine the SUMO method with the traditional ensemble method and consequently to do ensemble runs with many initial conditions of the SUMO. But in the moment we are at the state where we try to understand and evaluate the performance of the SUMO method.

Here [2] we deal with a different problem. As we have seen so far SUMO was evaluated in what we would call the perfect model scenario (PMS), that is the models and the system have structurally the same equations. Therefore the only mismatch between the model and the system tested so far is that the ensemble of models and the system have different coefficients or control parameters. We report on the much more realistic imperfect model scenario (IPMS). We assume that in the modeling process there occurred an error and the flow equations governing the models and the systems are therefore different. It is far from obvious that techniques developed for the PMS work in the IPMS as well. In the following we define the problem of the IPMS and introduce the SUMO framework. In addition we will define the quantities we are using to evaluate the performance of our forecasting method. Furthermore, we report on two specific examples based on the Lorenz and Rössler model. In both examples we use the standard equations as well as a more complex form of the equations. We are using the later as our systems and forecast its dynamics by the SUMO method. The ensemble of models is built from the standard, simpler models and therefore is structurally different from the system. We compare the performance of SUMO in the IPMS to the PMS as well as to a simple traditional forecast.

In the following we define the fundamental differences between the PMS and IPMS. In addition we are going to describe the different classes of imperfect models. Assume that the system's dynamics $y(t)$ and model's dynamics $x(t)$ are given as differentiable ordinary differential equations:

$$\dot{y} = f(y, \mu) \quad \text{with} \begin{cases} y(t) & \in \mathcal{R}^c \\ \mu & \in \mathcal{R}^n \end{cases}, \tag{1}$$

$$\dot{x} = g(x, \eta) \quad \text{with} \begin{cases} x(t) & \in \mathcal{R}^d \\ \eta & \in \mathcal{R}^m \end{cases}, \tag{2}$$

where μ and η are a set of model parameters.

We define the PMS as the special, somehow artificial case of $g(\xi, \mu) = f(\xi, \eta)$ for all $\xi \in \mathcal{R}^n$. Obviously this implies that $\dim(y) = c = \dim(x) = d$, $\mu_i = \eta_i$ for $i = 1, \dots, n$ and $n = m$. Note that the SUMO investigations in [Berge] considered this special case but allow for variation in the models parameters. That is for the N different models in the ensemble $\dot{x}_j = g(x, \eta_j)$ $\eta_{i,j} \neq \mu_i$ for $j = 1, \dots, N$. As it turns out they implied an additional constraint: $\min(\eta_{i,j}) < \mu_i < \max(\eta_{i,j})$, where the minimization (maximization) for each i is done over the ensemble.

In IPMS the previous equations do not hold and therefore $g(\hat{\xi}, \mu) \neq f(\xi, \eta)$ for almost all points in the state space. Since in general in the IPMS $n \neq m$, $\hat{\xi}$ (ξ) is a projection of ξ ($\hat{\xi}$) onto the lower dimensional space $\mathcal{R}^c$ ($\mathcal{R}^d$) if $c < d$ ($d < c$). A simple classification of the typical modeling errors leading to the IPMS can be done like this:

1. $\dim(y) = c = \dim(x) = c$ and the general form of g and f are the same, but $\eta_i \neq \mu_i$ for some or all $i = 1, \dots, n$.

2. $\dim(y) = c \neq \dim(x) = d$, which often occurs when the system has different time scales and a slow or fast subspace is ignored by the modeler.

3. $\dim(y) = c = \dim(x) = c$ but higher order terms in g are ignored when modeling.

As mentioned above the trivial case 1 of a mismatch between the model's and system's parameters can be easily be dealt with and L. A. van Berge *et al.* have used this example to introduce the SUMO method.. Case 2 and 3 are much more settle and define the true IPMS in our understanding. There has been much work on case 2, since A. Einstein's seminal work on stochastic modeling. If the ignored subspace is fast, its action on the dynamics can usually be modeled by means of a suitable noise term [1, 3]. If the ignored subspace is slow forecasting $y(t)$ should be possible by adding a constant term to the model $f(x, \eta)$. Ignoring such an offset can lead to very characteristic shortcomings in the forecasts, that allow identification of this specific modeling error.

In the following we focus on case 3. We assume that the model's and system's dynamics result from different flow equations. Obviously we want the difference between the system and model to be small but big enough to be non-negligible. In particular we are going to investigate two examples. First the Lorenz equations:

$$\begin{aligned} \dot{x} &= \sigma(y - x), \\ \dot{y} &= x(\rho - z) - y, \\ \dot{z} &= xy - \beta z, \end{aligned} \tag{3}$$

where we use the standard parameters $\sigma = 28$, $\rho = 10$, and $\beta = 8/3$.
Second the Rössler equations:

$$\begin{aligned} \dot{x} &= y - z, \\ \dot{y} &= x + ay, \\ \dot{z} &= b + z(x - c), \end{aligned} \tag{4}$$

where $a = 0.2$, $b = 0.2$, and $c = 5.7$.

We use eqs. (4) and (5) as our *model* equations and assume that a third order term was missed when modeling the dynamics. That is, the true systems' dynamics is governed by eqs. (4) or (5) when the state variables are replaced by:

$$\begin{aligned} x &\to x + \epsilon x^3, \\ y &\to y + \epsilon y^3, \\ z &\to z + \epsilon z^3. \end{aligned} \tag{5}$$

Note that we use an odd term, since this is the order in which the variables enter in our model–equations (4) and (5).

I. SUMO AND FORECASTING MEASURES

The general SUMO method is described somewhere else [6], therefore we are only going to report on the general setup and how to optimize the coupling between the different models in the ensemble. In addition we are presenting the measures we used to evaluate the performance of the forecast as well as the different type of experiments done.

Assume that we have j models:

$$\dot{x_j} = g(x_j, \eta_j), \tag{6}$$

where $x_j = (x_{j,1}, \dots, x_{j,d})^T$ and the parameters $\eta_j = (\eta_{j,1}, \dots, \eta_{j,n})$. We want to use linear coupling of the d–dimensional models to give us a $j \times d$–dimensional SUMO:

$$\dot{x_k} = g(x_k, \eta_k) + \sum_{m \neq k}^{j} \sum_{p=1}^{d} C_{mk}^{x_p}(x_{m,p} - x_{k,m}), \tag{7}$$

where the first summation is over the ensemble of the models and the second summation over the d components of the state vector. The coupling constant $C_{mk}^{x_p}$ determines how much model k is influenced by the dynamics of the other models. To determine the coupling constant we follow the idea presented in [Berge]. They suppose to minimize the following cost function given the true dynamics $y(t)$:

$$\min_C(F(C)) = \min_C \left(\frac{1}{K\Delta} \int_{t_i}^{t_i+\Delta} |x(C,t) - y(t)|^2 \gamma^t \mathrm{d}t \right), \tag{8}$$

where the minimization is done over the vector of possible coupling constants $C \in \mathcal{R}^d$ and γ^t is some scalar error–weighting factor between the model and the system. Note that the factor γ^t in the minimization means that errors for small t contribute more than errors at larger t. Following the conclusions of L. A. van Berge *et al.* [6] we choose $\gamma = 1/2$.

In our investigation we used eqs. (4) and (5) as our model and $y(t)$ results from integrating these equations after using the substitutions (6). While in the ensemble forecasting field there exists one superior measure [7], often it is not necessary to go to these extremes. Instead to determine the performance of our algorithm we use the *separation time* T_m, that is, the largest *lead-time* for which the *forecast error* remains less than some threshold. Define:

$$T_m = \max\left\{T\colon \|y(t_0+t) - x(t_0+t)\| \leq 2\sigma,\ \forall\ 0 \leq t \leq T\right\}, \tag{9}$$

where $y(t_0) = x(t_0)$ is the initial condition of the system and the SUMO. We have chosen 2σ as some threshold, that depends on the system investigated.

For both of our experiments we determine the separation time T_m for three different forecast methods: (1) forecast $y(t)$ with one one standard Lorenz or Rössler model, that is $j = 1$ and the model parameters are the standard parameters given in eqs. (4) or (5); (2) forecast $y(t)$, given that the system as well as the SUMO are given by eqs. (4) or (5), that is $\epsilon = 0$ in eq. (6); (3) forecast $y(t)$ (given by $\epsilon \neq 0$) with an ensemble of models given by eqs. (4) or (5).

Note that these experiments allow us to compare (1) the most simple forecast scheme in the IPMS with (2) the SUMO method given by [6] and (3) the SUMO method applied to a true IPMS. In all three cases we used an ensemble different initial conditions $y(t_0)$, so that our results do not just give one special case, but represent the overall performance of the method.

[1] Wolfram Just, Holger Kantz, Christian Rödenbeck and Mario Helm J. Phys. A: Math. Gen. **34** (2001) 3199–3213.

[2] Carsten Grabow, Thomas Stemler, Jürgen Kurths in preparation (2015)

[3] Adi R. Buisara, Katja Lindenberg and K. E. Shuler Journal of Statistical Physics, **27** (1982) 787–808

[4] Gregory S. Duane, Dongchuan Yu and Ljupco Kocarev Physics Letters A **371** (2007) 416–420.

[5] K. Judd, Journal of Nonlinear Science, **18** (2008) 57–74.
D. Ridout and K. Judd. Physica D, **165** (2002) 27–48.
T. Stemler and K. Judd, Physica D, **238** (2009) 1260–1273.
K. Judd and L.A. Smith. Physica D, **196** (2004) 224–242.
K. Judd, Physica D, **237** (2008) 216–232.

[6] L. A. van den Berge, F. M. Selten, W. Wiegerinck and G. S. Duane Earth Syst. Dynam., **2** (2011) 161–177.

[7] J. Bröcker and L. A. Smith, Tellus A, 60(4):663-678 (2007).