Improving the LPJ-GUESS modelled carbon balance with a Marginal particle filter data assimilation technique

Andrew D. McRobert,[1] Unn Dahlen,[1] Marko Scholze,[1] Sarah Kemp,[2] and Ben Smith[1]

Lund University

[1]Department of Physical Geography, Lund University, Lund, Sweden

[2]Department of Earth Sciences, University of Bristol, Bristol, UK

Motivation

Global climate is dramatically warming as a result of increases in anthropogenic greenhouse gases. The global carbon cycle is the distribution and exchange of carbon across several pools or states. The Lund-Potsdam-Jena General Ecosystem Simulator (LPJ-GUESS) [2] model simulates the natural ecosystem and has been used in this study to quantify recent changes in the terrestrial carbon cycle. In all Dynamic Global Vegetation Models (DGVM) like LPJ-GUESS there are many assumptions, often with the value of process parameters (see Figure 1), which can result in large differences from observed carbon fluxes. This study provides a method to reduce the uncertainties of these process parameters. The method employed is a Marginal particle filter data assimilation technique that optimizes the most sensitive process parameters driving LPJ-GUESS. The particle filter will optimize 30 process parameters in LPJ-GUESS by minimizing the mismatch between model output Net Ecosystem Production (NEP) and CO_2 eddy flux observations.

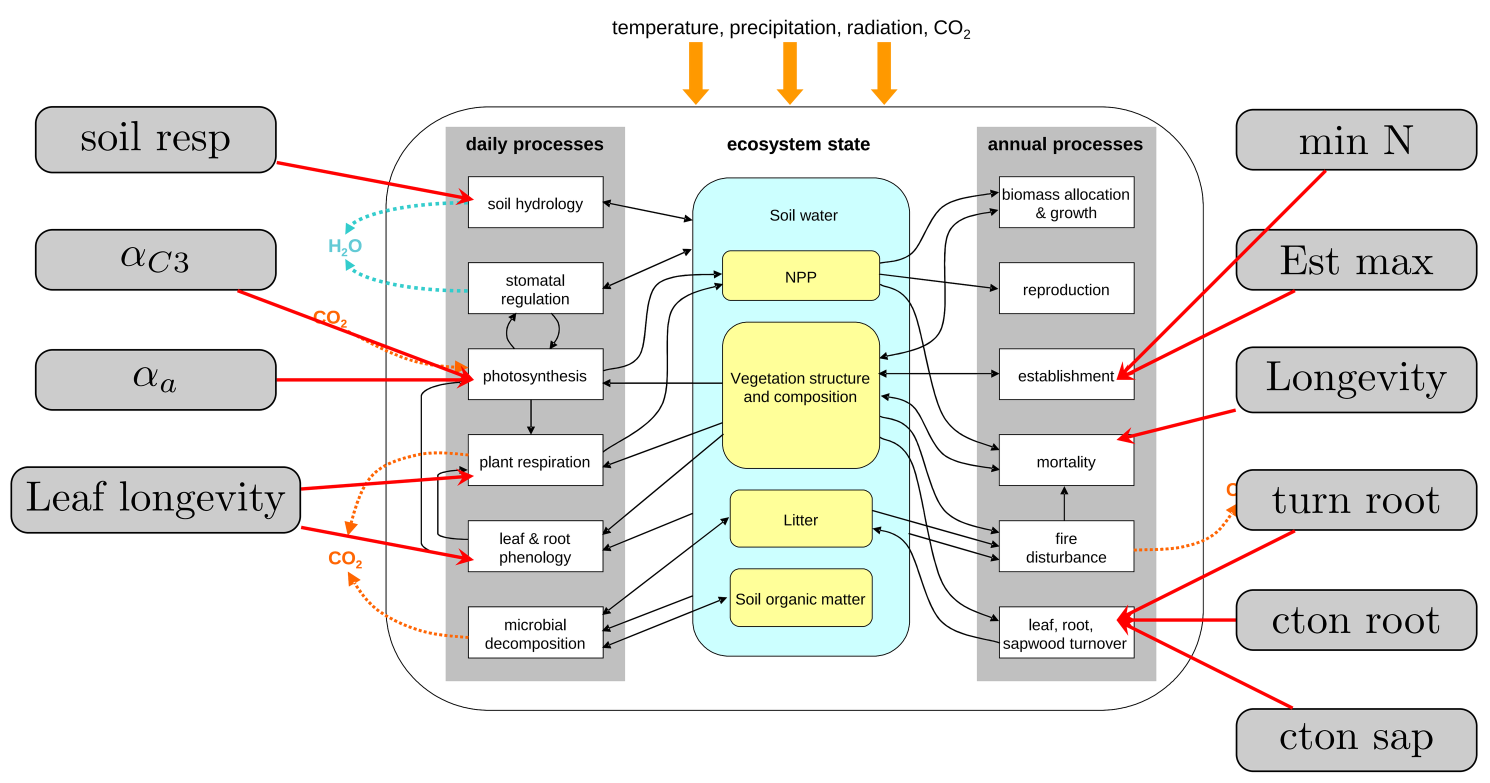

Figure 1: A flow chart where the input to LPJ-GUESS is shown, including some of the optimized process parameters. Along with a stylized example of the computational areas of the ecosystem model.

Methods

The traditional particle filter was used to optimize 30 process parameters of the LPJ-GUESS model. However, the traditional particle filter suffers from filter degeneracy and as the LPJ-GUESS model is very highly non-linear, the filter was consistently stuck in local minima.

To solve the problem of filter degeneracy we present an adapted version of the Marginal particle filter [1]. The marginal particle filter (MPF) works by filtering for the marginal distribution $p(x_t|y_{1:t})$ as a-posed to $p(x_{1:t}|y_{1:t})$ employed during the traditional particle filter. This means that the MPF optimises for the latest iteration t, while the traditional method optimises for all iterations up to the current. This simple switch greatly limits the degrees of freedom within the joint space.

To solve for the marginal distribution we need to solve the integral in Equation 1.

$$p(x_t|y_{1:t}) = p(y_t|x_t)\int p(x_t|x_{t-1})p(x_{t-1}|y_{1:t-1})dx_{t-1} \quad (1)$$

That is not possible analytically, however, we can use the particle approximation (Equation 2). We choose the solution in the form of Equation 2, which solves for the proposal density $q(x_t|y_{1:t})$.

$$q(x_t|y_{1:t}) = \sum_{j=1}^{N} w_{t-1}^{(j)} q(x_t|y_t, x_{t-1}^{(j)}) \quad (2)$$

In Equation 2, the w term represents the weights, which are calculated using Equation 3. The weights from Equation 3 are normalised before being used in Equation 2.

$$w_t = \frac{p(x_t|y_{1:t})}{q(x_t|y_{1:t})} \quad (3)$$

The weights in this form become a balance between how close the particle is to the observations and the likelyhood of the particle being chosen from the previous pdf. This means that particles that are from the tails of the pdf are given relatively higher weights compared to particles from the most likely centre of the pdf, if there cost functions are the same. In this way, the MPF will maintain a wider and more sensible pdf, which is significantly better at avoiding getting stuck in local minima, thus improving the robustness of this technique.

We go further to avoid the problems of filter degeneracy by adapting the marginal particle filter to also include a technique called tempering. This involves artificially broadening the probability density function at the resampling step, so that particles from the tails of the distribution become much more likely. This ensures that a much wider area of the parameter space is investigated by the filter (see Figure 2). We have adapted the formula for the weights in the MPF (Equation 3) to include the term b, as shown in Equation 4. The trick is to set a value of b such that we gain the benefit of searching a wider parameter space, while not going too far and adversely slowing the computation of the filter as the parameters struggle to converge. We set a value for b_t, such that $\sum_{t=1}^{M} b_t = 1$, where M is the number of iterations. The disadvantage of this tempering technique is that we need to specify the number of iterations in advance.

$$w_t = \frac{p(y|x_t)^{b_t}}{\sum_{j=1}^{N} q(y|x_t)^{b_t}} \quad (4)$$

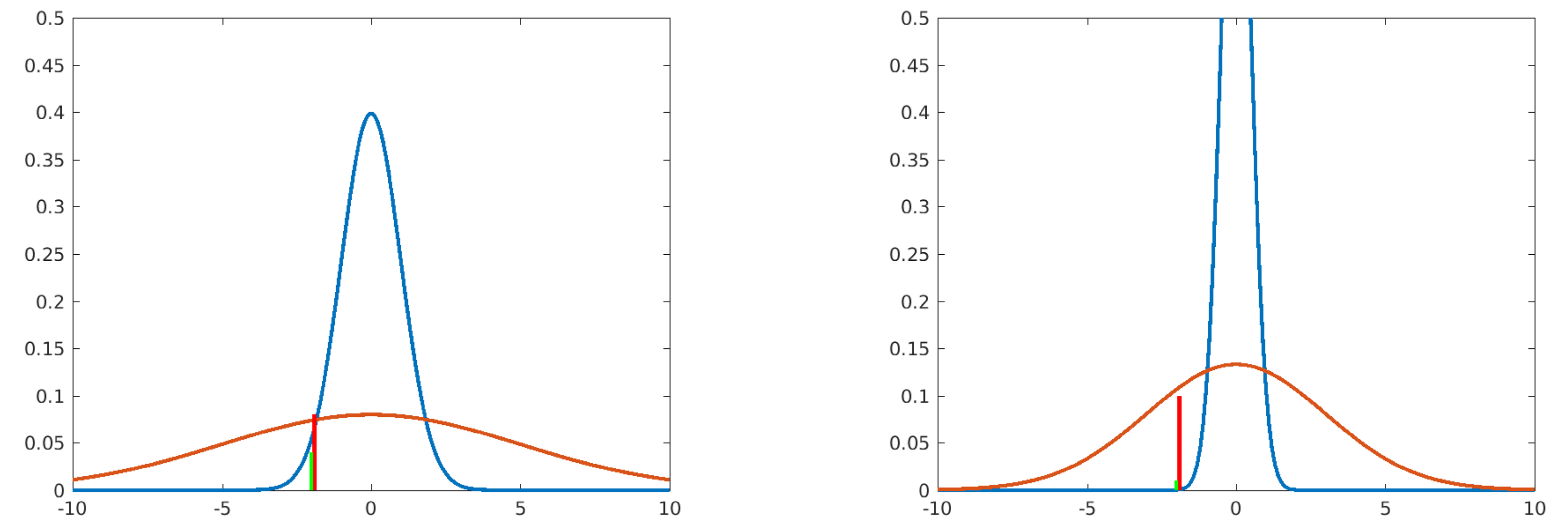

Figure 2: A example of the re-sampling employed by the Marginal particle filter. In blue is the pdf of the traditional particle filter, in red is the pdf of the Marginal particle filter at the re-sampling step. The green line is the likelihood of selecting this particle with the traditional filter, while the red line is the chance of the same particle being selected using the Marginal particle filter.

Results

Our first results are from an experiment called "The Twin", which is designed to test the Marginal particle filter setup. Initially, LPJ-GUESS is run using the prior parameters values to obtain a set of "observed" NEE values. The prior parameters values are perturbed before the full particle filter model is run using the NEE data from the first run as observations. In this way we are testing the particle filter in a controlled environment.

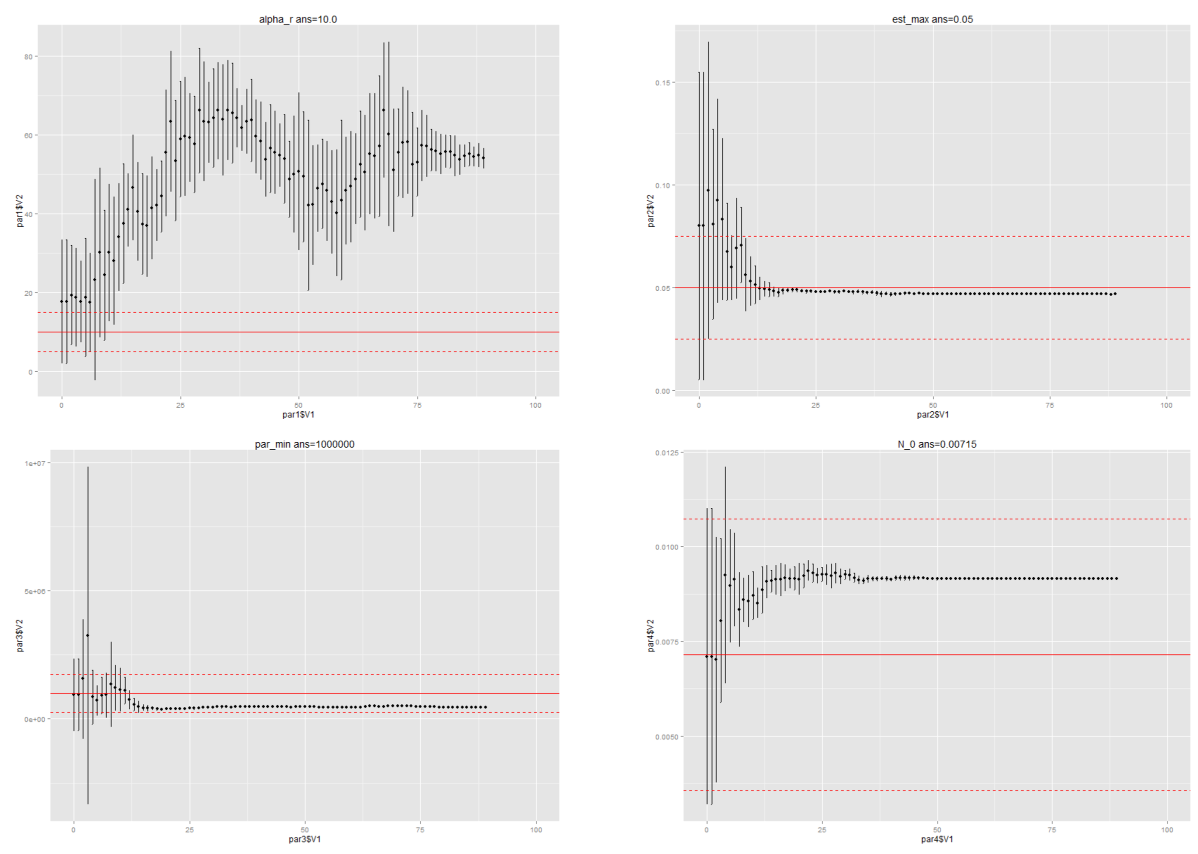

Figure 3: Each panel is the mean value of a parameter for each particle. The four parameters shown here are the shape parameter for juvenile growth rate, maximum sapling establishment rate, minimum photosynthetically active radiation and minimum nitrogen.

We have chosen parameters from LPJ-GUESS that have the most control over the NEP, which we optimize against using the cost function. Figure 3 shows the progression of four parameters during the optimization. The maximum sapling establishment rate (top right), minimum photosynthetically active radiation (bottom left) and the minimum nitrogen (bottom right) are very well constrained after the first 100 iterations, to values within the solution uncertainty but not to the exact solution. While, Alpha R (top left) is poorly optimized by the particle filter, which doesn't find a stable value during the optimization.

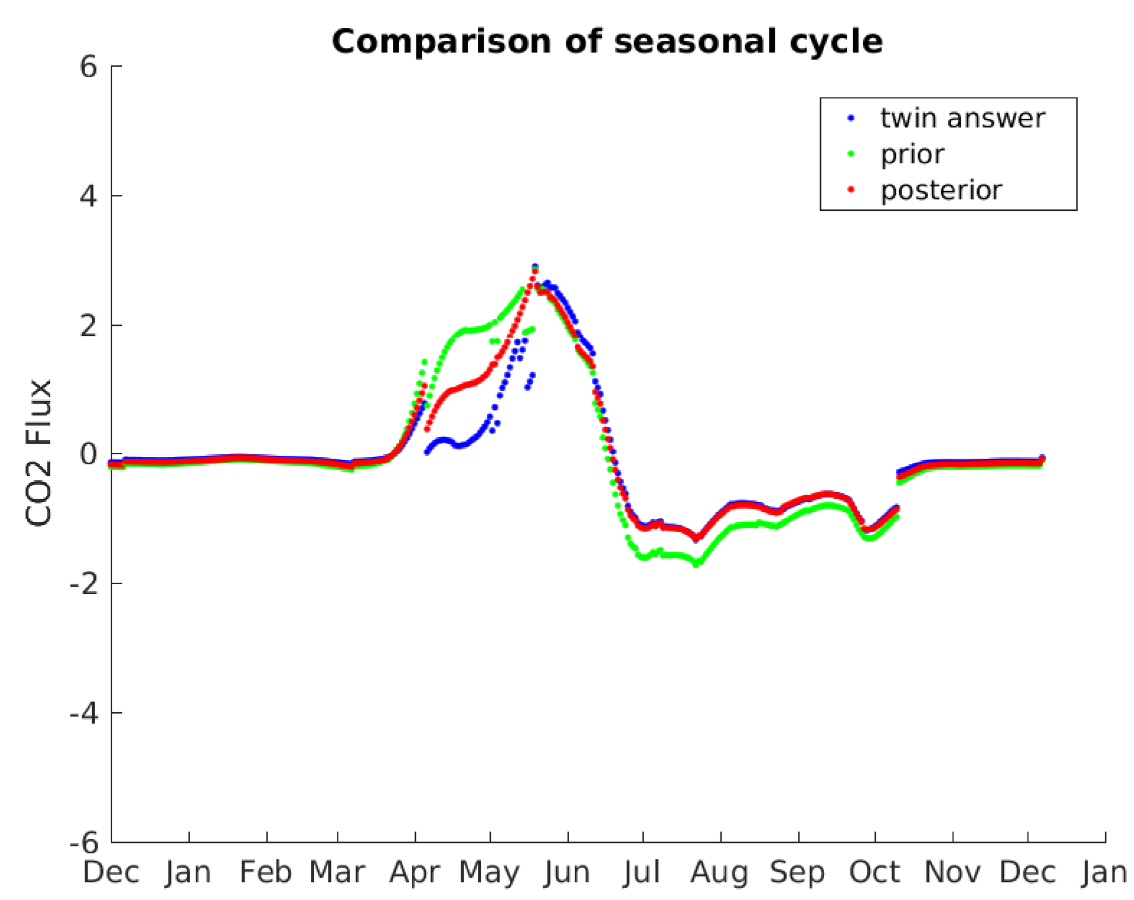

Figure 4: This figure shows the model output CO_2 flux for 1 year. In blue is the CO_2 flux results from the twin solution parameter set, while the green is the prior and red the posterior. In this case the posterior is the set of parameters obtained after 100 iterations.

We have compared the seasonal cycles in Figure 4 and find that in winter when there is no CO_2 flux the prior and posterior are both on the solution but that they diverge in the timing and magnitude of the summer uptake. The posterior struggles to match the onset of the summer uptake in particular.

References

[1] M. Klass, N de Freitas, and A. Doucet. Toward Practical N^2 Monte Carlo: the Marginal Particle Filter. *ARXIV*, (8), JULY 2012.

[2] Benjamin Smith, I. Colin Prentice, and Martin T. Sykes. Representation of vegetation dynamics in the modelling of terrestrial ecosystems: Comparing two contrasting approaches within European climate space. *Global Ecology and Biogeography*, 10(6):621–637, jun 2001.