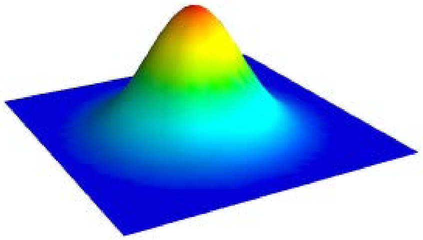
Generalization of Multigrid Beta Filter Scheme for Modeling Background Error Covariance



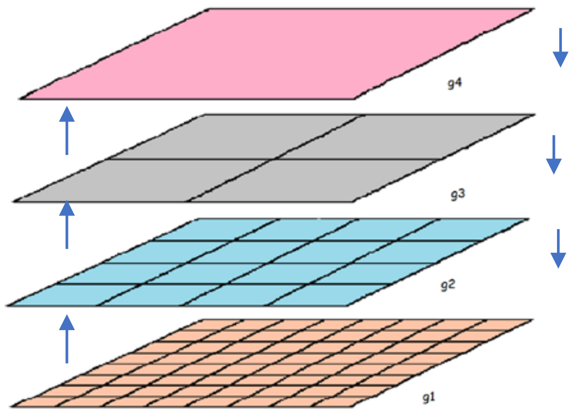
Miodrag Rancic

R, James Purser

Manuel dePondeca

Teppei Kinami

IMSG at EMC/NCEP/NOAA, College Park, MD, USA



3D RTMA Project and MG (Multigrid) Beta Filter

Real Time Mesoscale Analysis (RTMA) provides current conditions for transportation customers, verification of forecasts and is used as the reference for bias correction in the National Blend of Models.

The major development underway is a fully three-dimensional (3D) RTMA system which will provide analyses of a range of parameters at

- high horizontal resolutions (~2.5 km)
- frequent time intervals (~15 min)

The key prerequisite for the success of this enterprise is a vastly improved efficiency in producing those analyses.

The new approach to modeling of background error covariance (**B**), an estimation of the weight by which the background field participates in formulation of the cost function that is minimized within a data assimilation procedure, is one of the key components for the success of that effort.

Modeling of the Background Error Covariance

For calculation of **B** in data assimilation system at EMC so-far we were using recursive filters (e.g., Wu et al., 2002; de Pondeca et al., 2011).

Though recursive filters are a very good and efficient approximation of Gaussian (e.g., Purser et al. 2002, 2003) they have a series of shortcomings. The most serious one is that they are essentially sequential operators, very difficult to successfully parallelize.

Our solution to this problem within the UFS effort is the development of a new filter based on Beta distributions, incorporated within a parallel multigrid structure, which:

- Describes covariances across various scales
- Includes cross-correlations
- Provides negative sidelobes, which realistic covariances do posses
- Has a finite support and is more parallelizable, resulting in much better scaling

Description of Beta-filter

Our alternative to recursive filters is based on the Beta distribution filters. In 2D case, the radial Beta filter is defined as

$$\beta(x, y) = (1 - \rho)^p \qquad , \rho \leq 1$$

where p is a small positive integer and, in the isotropic case,

$$\rho = \frac{1}{s^2} \boldsymbol{r} \cdot \boldsymbol{r}^T$$

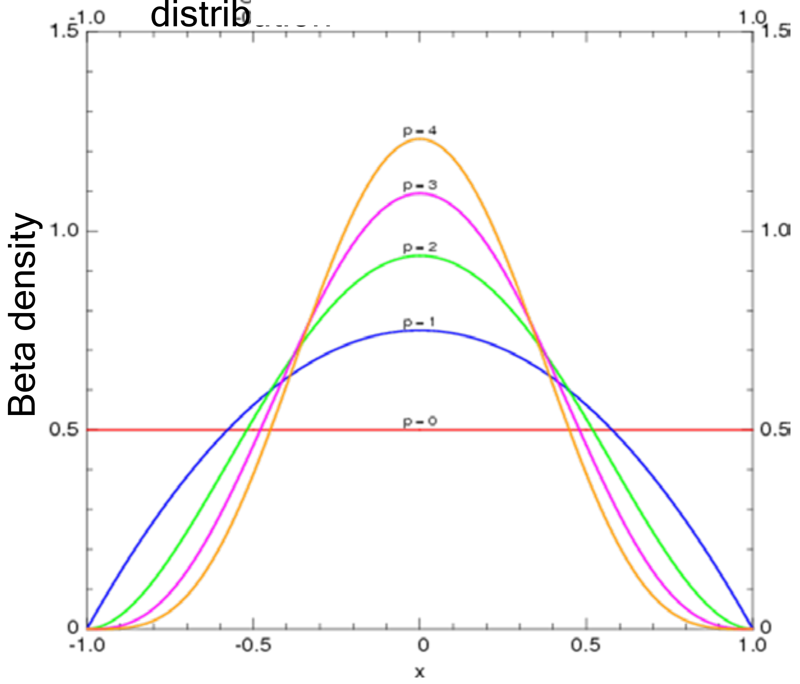
Here, s is a radial scale and $\boldsymbol{r}$ a position vector (x, y). Such a function also has a quasi-Gaussian shape, but with a finite support.

In an anisotropic generalization, s^2 is replaced by a 2×2 symmetric, positive definite "aspect tensor", used as a matrix inverse $\mathbf{A}^{-1}$, so that

$$\rho = \boldsymbol{r} \mathbf{A}^{-1} \boldsymbol{r}^T$$

A larger p implies a more Gaussian shape, but also a narrower one. The 3D radial Beta filter has a similar formulation.

Symmetric Beta density distrib



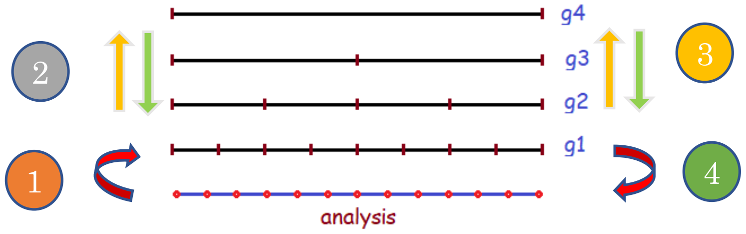
Basic MG Beta filter

Beta filter is further used at a hierarchy of different scales, combined into a parallel multigrid scheme in order to achieve a larger coverage and potentially a more versatile synthesis of anisotropic covariances, allowing a greater control over the shape.

There are four stages of this process:

1. Adjoint interpolate from the analysis grid to generation one (g1) of the filter grid
2. Adjoint filter (conservative) stage
3. Forward filter(smoothing) stage
4. Interpolate from g1 to the analysis grid

Generally, g1 has a lower resolution than analysis grid



Conservative Stage

2_1 Adjoint interpolate ("up-send") from g1 to g2. Then repeat procedure all the way to g4

2_2 Apply weights at all generations in parallel

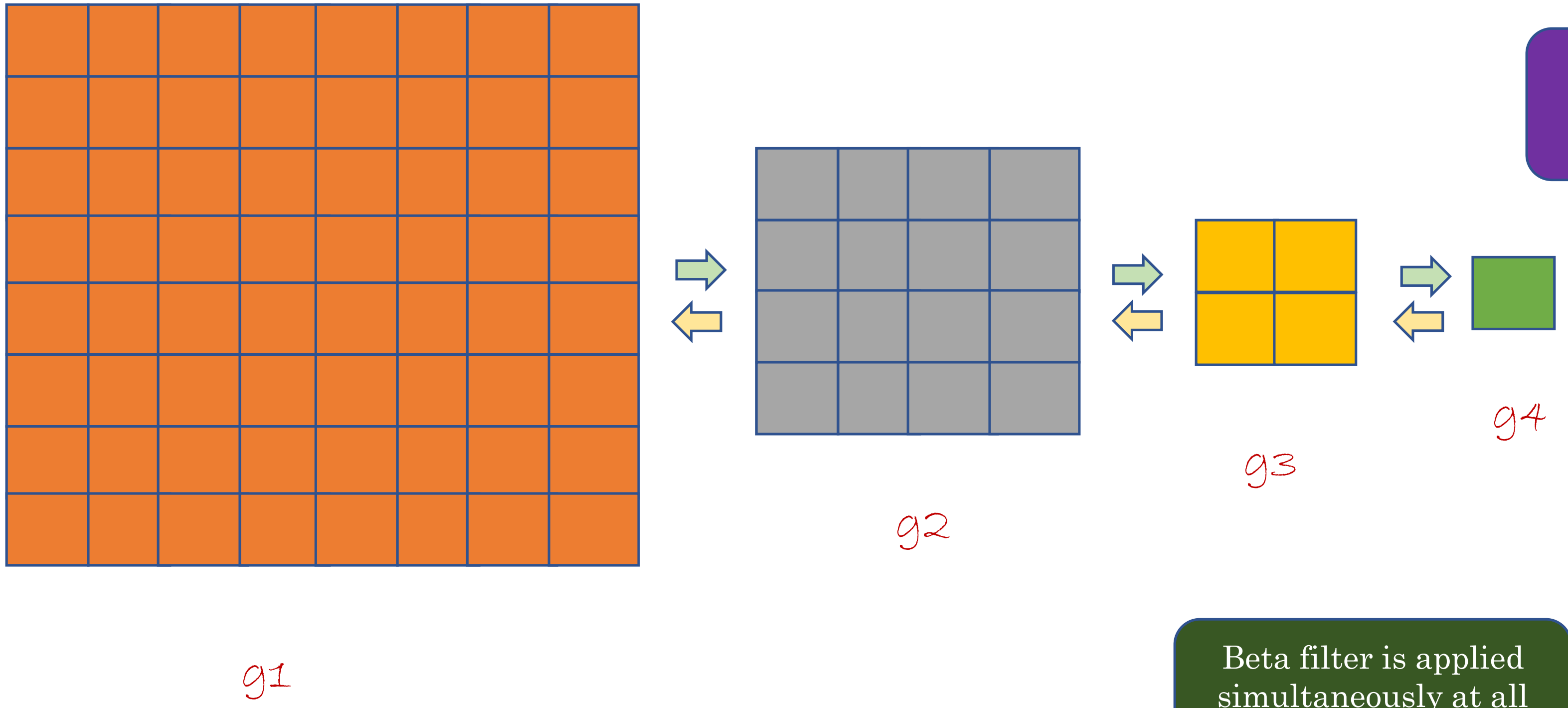
2_3 Apply adjoint of Beta filter at all generations in parallel

2_4 Interpolate ("down-send") result of adjoint filter from g4 to g3 and add it to adjoint at g3. Then repeat procedure all the way to g1

Smoothing Stage

Identical to the Conservative stage except that we apply Beta filter instead of it adjoint

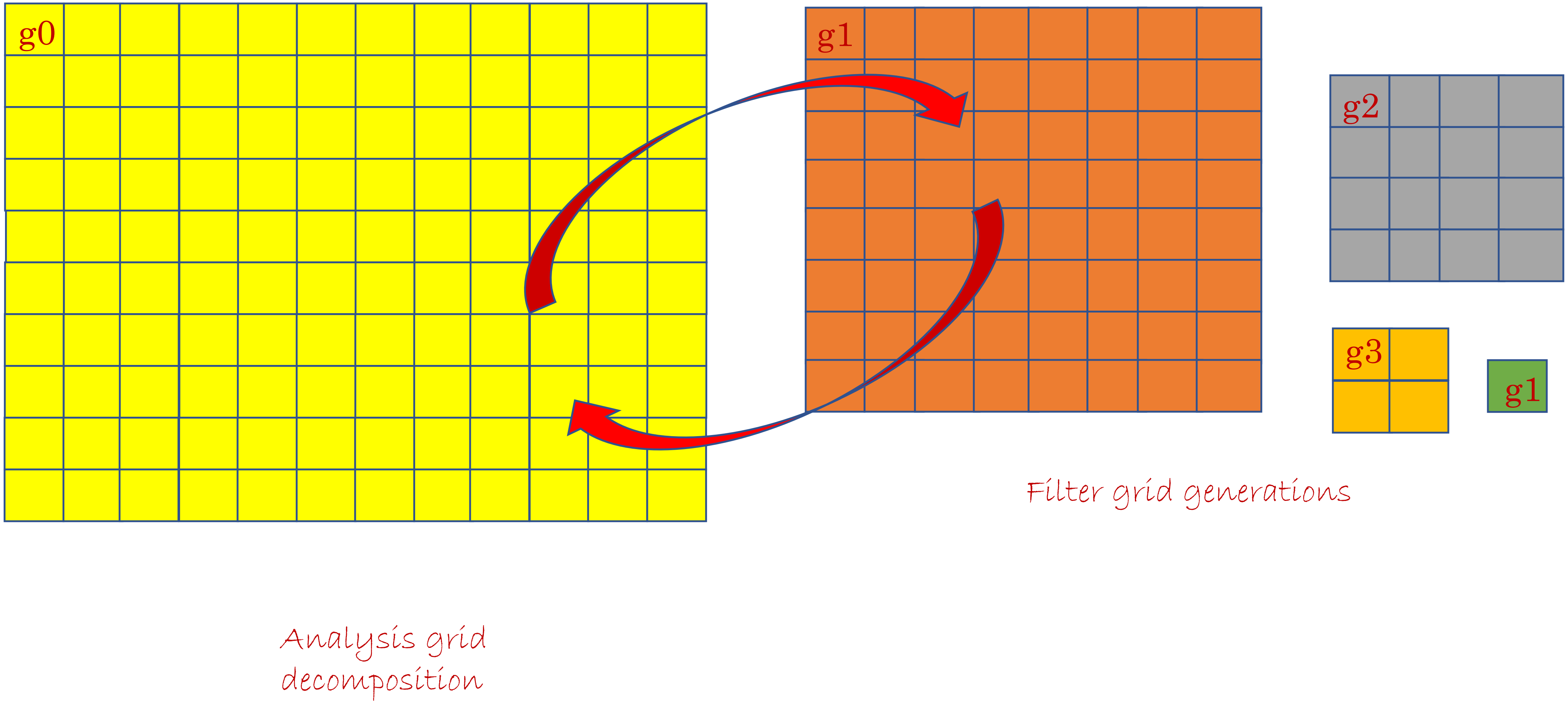
In the stage of up-sending we half the resolution in transition from one generation to another. Consistently, we half the number of processors in each direction. Thus, ideally, the number of processors arranged in each direction of generation g1 must be divisible by 2^{n-1} where n is the number of generations. The opposite happens in the stage of down-sending, when we double the resolution and double the number of processors in each direction.



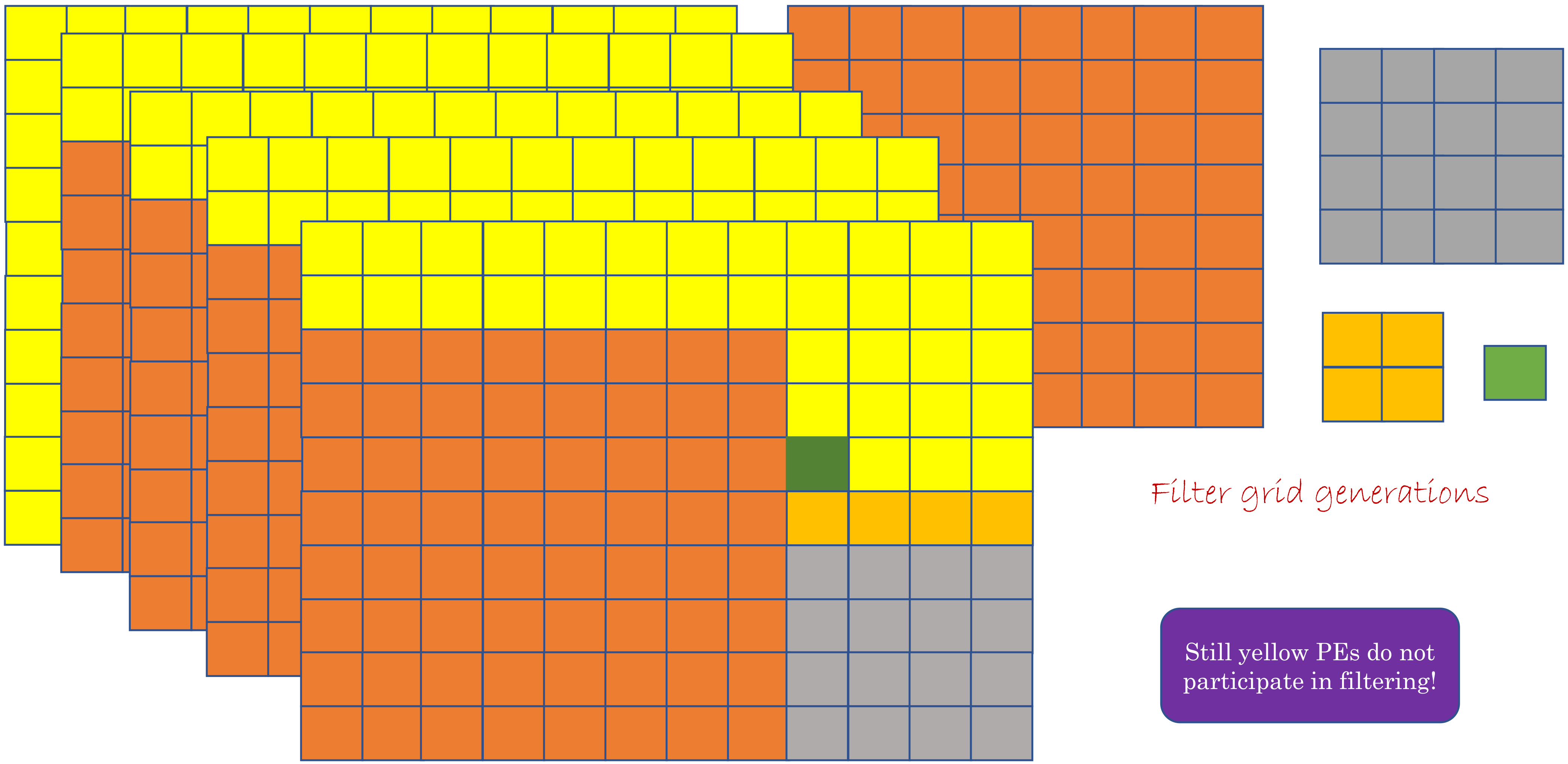
Each PE (small square) has the same resolution

Beta filter is applied simultaneously at all generations

In practice, the analysis grid has its own resolution and decomposition. Thus, at each inner iteration, we need to remap and re-decompose between the analysis grid and g1.



Filter grid generations need to be collocated with the analysis processors:



Still yellow PEs do not participate in filtering!

Generalization of MG

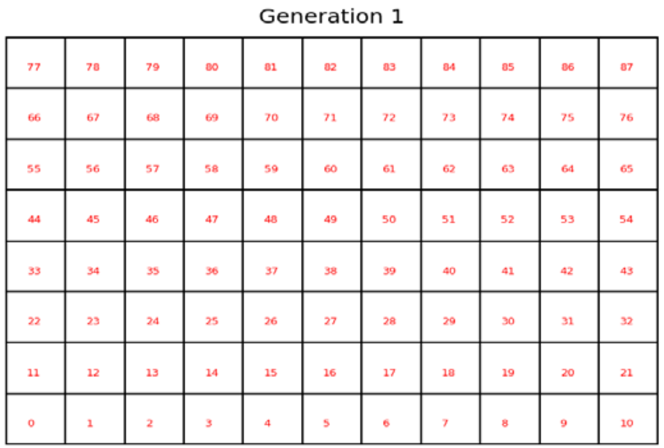
The described paradigm has two problems:

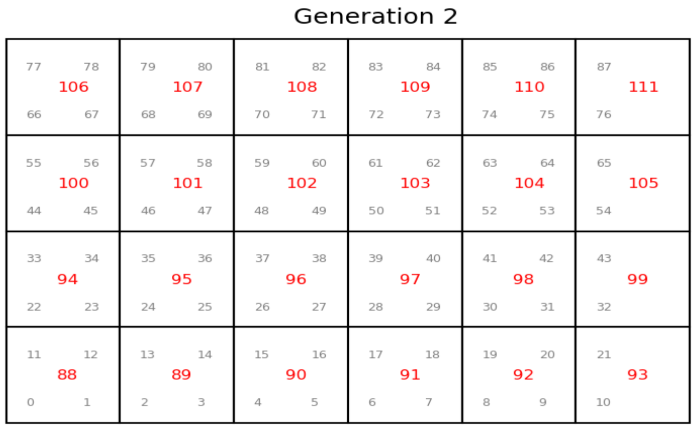
1. Re-decomposition between analysis and filter grid is a bottleneck which slows down filtering process
2. The whole procedure is very hard to generalize for various arrangements of processors

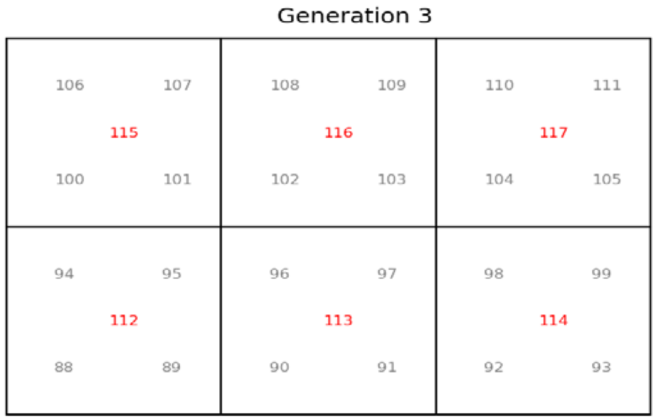
In Rancic et al. (2020) we considered a series of possible solutions, none of which able to fully overcome both issues. Here we present a new solution which eliminates the bottleneck and allow us to generalize MG Beta filter without degrading its performance

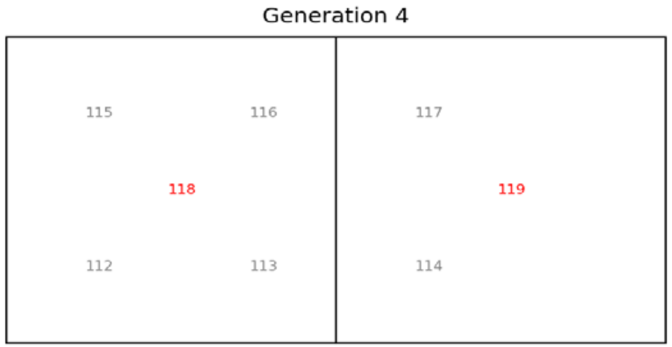
1. Keep the g1 at the same decomposition as the analysis grid
2. In construction of higher generations we allow inclusion of "empty space" (keeping the boundaries of the physical domain unchanged).

Example of new decomposition









Pros:

There is no more need for re-decomposition between analysis and filter grid

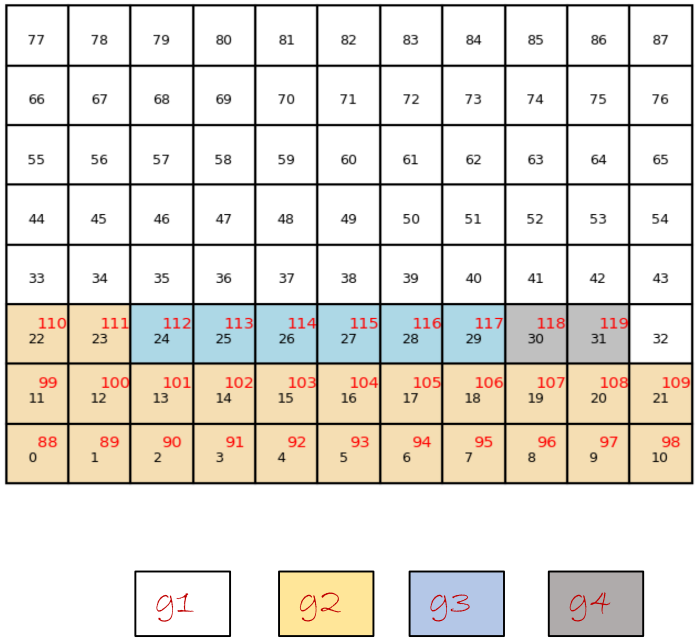
Filter grid is run on more PEs

The code is automatically adjustable to any decomposition

Cons:

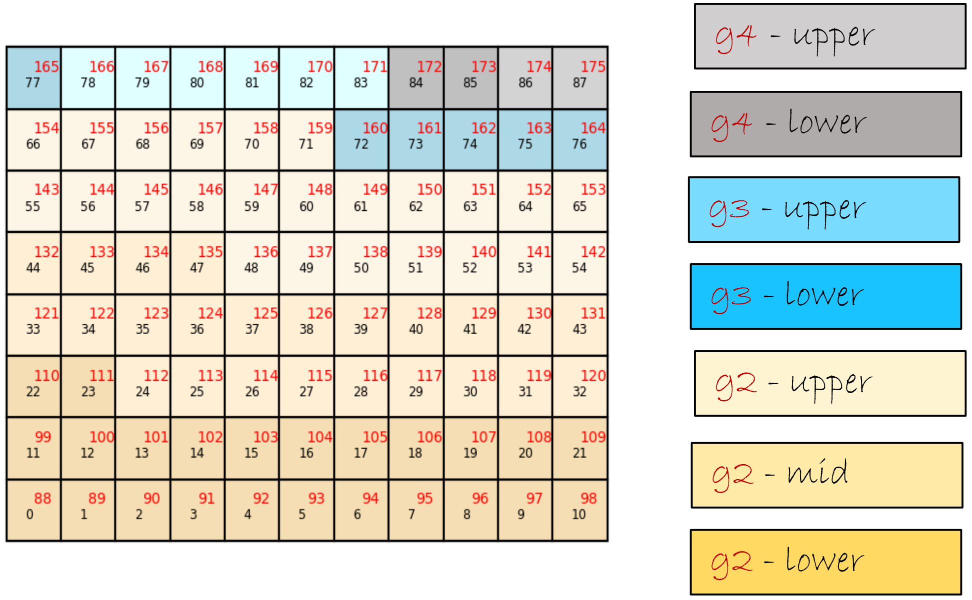
Higher generations are executed in parallel among themselves but sequentially with g1

Collocation of PEs in the new paradigm



In this original form the higher generations do not take the full advantage of the available processing capabilities.

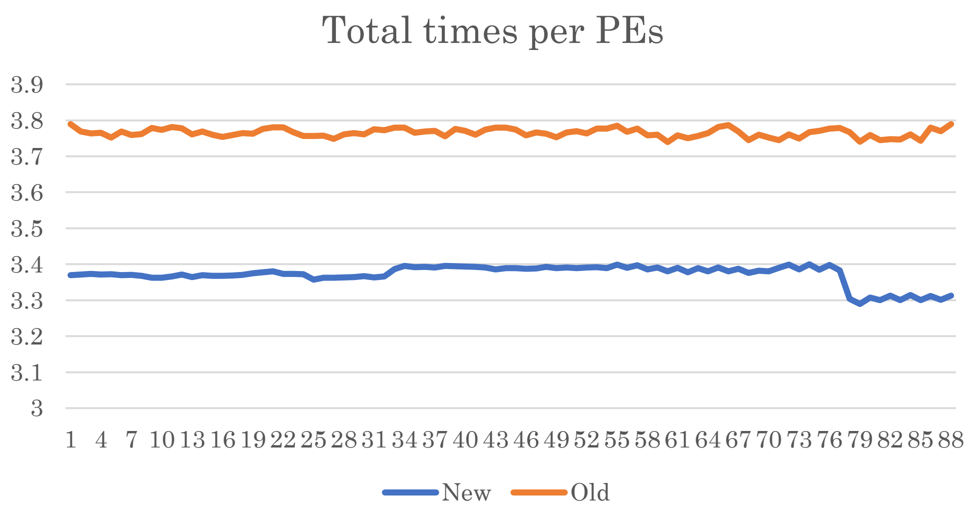
In principle, by using lower vertical resolution for 3D arrays of higher generations, and by judiciously vertically splitting and sharing their load among processors, it is possible to solve this issue



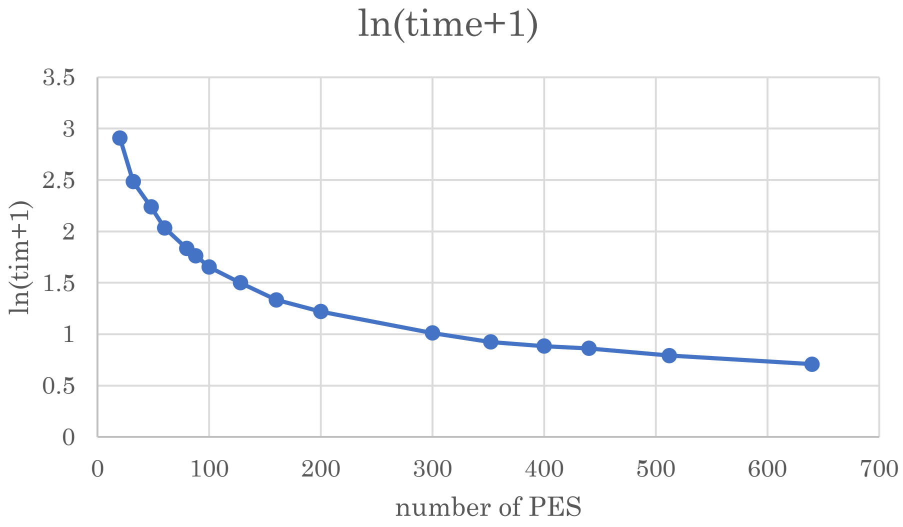
g1: 50 levels
g2: 45 levels → Split in 3 layers
g3: 30 levels → Split in 2 layers
g4: 30 levels → Split in 2 layers

→ ~ 30% faster than g1

Examples of performance



Timings per PEs derived in tests on 88 PEs with old and generalized scheme (new). There is a gain of about 10% in efficiency.



Timings in tests of generalized MG Beta filter code.

Future developments

- A new version of MG Beta filter is developed which requires only one onset of up- and down-sending. Through application of a differential Helmholtz operator it will allow inclusion of negative sidelobes of the covariances. In addition, we are working on a version that will allow inclusion of cross-covariances (for this first time to our knowledge)
- We are working on a series of novelties, such as, a new method for normalization of covariances; extension for global cubed sphere domain; application of AI for definition of scale weights, etc. Among them, perhaps the most important place takes the replacement of the radial filters with a sequence of line filters (Purser 2020).
- A Triad (3-components, in 2D) and Hexad (6-components in 3D) versions have been developed that will replace radial filter. A consistent extension of this approach lead us to a fully 4D extension (so-call Decad algorithm) giving us a tool that would enable future extension of the RTMA procedure into a fully 4D scheme
- We begin integrating the first version of MG Beta filter code in GSI and JEDI

References

Purser, R. J., 2005: A geometrical approach to the synthesis of smooth anisotropic covariance operators for data assimilation. NOAA/NCEP Office Note 447.

Purser, R. J., 2020: A formulation of the Hexad algorithm using the geometry of the Fano projective plane. Draft NOAA/NCEP Office Note (In preparation).

Purser, R. J., 2020: A formulation of the Decad algorithm using the symmetries of the Galois field GF(16). Draft NOAA/NCEP Office Note (In preparation).

Purser, R. J., W.-S. Wu, D. F. Parrish, and N. M. Roberts: 2003: Numerical aspects of the application of recursive filters to variational statistical analysis. Part II: Spatially inhomogeneous and anisotropic general covariances. *Mon. Wea. Rev.*, **131,** 1536—1548.

Rancic, M., M. Pondeca, R. J. Purser, and J. R. Carley, 2020: MPI re-decomposition and remapping algorithms used within a multigrid approach to modeling of the background error covariance for high-resolution data assimilation. J59.3. 100th Annual Meeting of American Meteorological Society, Boston, Massachusetts, 13-16 January 2020.

Wu, W.-S., R. J. Purser, and D. F. Parrish, 2002: Three-dimensional variational analysis with spatially inhomogeneous covariances. *Mon. We. Rev.*, **130,** 2905—2916.