1. The Tikhonov-Regularized (TR) formulation

The TR problem can be described as following:

\[ u(\lambda) = \text{argmin}_u \{ \|Hu - v\|^2_R + \lambda \|u - u^M\|^2_B \} \]

where \( \| \cdot \|_R \) and \( \| \cdot \|_B \) denote the weighted norms with respect to the error covariance matrices \( B \) and \( R \) and \( \lambda \) is the regularization parameter.

2. The decomposed TR formulation

Let \( \Omega \in \mathbb{R}^3 \) be the domain decomposed into a sequence of overlapping sub-domains \( \Omega_i \in \mathbb{R}^3 \), such that:

\[ \Omega = \bigcup_{i=1}^{p} \Omega_i \]

The decomposed TR formulation:

\[ u_i^{DA} = \text{argmin}_u \{ J(\Omega_i) + O(\Omega_i) \} \]

3. Performance Analysis – Scaleup factor

If \( p \in N \) and \( p > 1 \), the algorithm associated to the decomposition given is:

\[ A(\Omega, p) := (A(\Omega_1), A(\Omega_2), \ldots, A(\Omega_p)) \]

Let \( p_1, p_2 \in N \) and \( p_1 < p_2 \). Let \( T(A(\omega, p_i)) \) denote the time complexity of \( A(\omega, p_i) \), \( i = 1, 2, \forall i \neq j \) we define the (relative) scale-up factor of \( A(\omega, p_2) \), in going from \( p_1 \) to \( p_2 \), the following ratio:

\[ S_{p_2, p_1}(N) = \frac{T(A(\Omega, p_2))}{T(A(\Omega, p_1))(p_2/p_1)T(A(\Omega, p_2))} \]

4. Case Study: Data Assimilation problem

Let \( t \in [0,T] \) denote the time variable. Let \( u^{true}(t, x) \) be the evolution state of a predictive system governed by the mathematical model \( M \) with \( u^{true}(t_0, x), t_0 = 0 \) as initial condition. Here we consider a 3D shallow water model. Let \( v(t, x) = H(u^{true}(t, x)) \) denote the observations mapping, where \( H \) is a given nonlinear operator which includes transformations and grid interpolations.

5. Results

We consider two hybrid architectures: HA1 is a 288 CPU-multicores, HA2 is a GPU+CPU architecture.

\[
\begin{array}{c|c|c|c|c|c}
\text{N} & \text{p} & \text{pproc} & \text{T}_{\text{flip}}(N) & \text{T}_{\text{nproc}}(N) & \text{S}_{\text{nproc}} \\
\hline
10^6 & 1 & 2 & 1 & 1 & 1 \\
10^6 & 2 & 4 & 0.044 & 0.019 & 2.3 \\
10^6 & 8 & 4 & 0.025 & 0.023 & 1.09 \\
10^6 & 4 & 8 & 0.025 & 0.025 & 1.09 \\
10^6 & 16 & 8 & 0.025 & 0.025 & 1.09 \\
\hline
\end{array}
\]

Values of execution time of algorithm running on GPU for a problem size \( O(10^6) \), \( \lambda = 1 \), \( \alpha = 1 \), thus the above analysis validates the experimental results.

6. Discussion

We now discuss scalability results shown in the tables. To this end, we introduce

\[
S_{\text{loc}}^{\text{nproc}} = \frac{T_{\text{flip}}(N/p) + \alpha T_{\text{nproc}}(N/p)}{T_{\text{nproc}}(N/p)}
\]

which denotes the speed up of the (local) algorithm \( A(D_N(\Omega), N/p) \) for solving the local problem on subdomain \( D_N(\Omega) \). Let us express the measured scale up factor in terms of \( S_{\text{loc}}^{\text{nproc}} \). We have:

\[
S_{\text{loc}}^{\text{nproc}} = \alpha S_{\text{loc}}^{\text{nproc}} + \frac{s_{\text{loc}}}{s_{\text{nproc}}} + \frac{\alpha}{1 + \alpha} \]

Finally, it is worth noting that in our experiments, on HA1, local DA problems are sequentially solved, then \( S_{\text{nproc}} = 1 \) and \( \alpha > 1 \), thus the above analysis validates the experimental results.