Preliminary Results of GRAPES' Helmholtz solver using GCR and PETSc tools

Xiangjun Wu[1] , Lilun Zhang[2] , Junqiang Song[2] and Dehui Chen[1]

(1) Center for Numerical Weather Prediction, CMA

(2) School of Computer Science, National University of Defense Technology

wuxj@cma.gov.cn

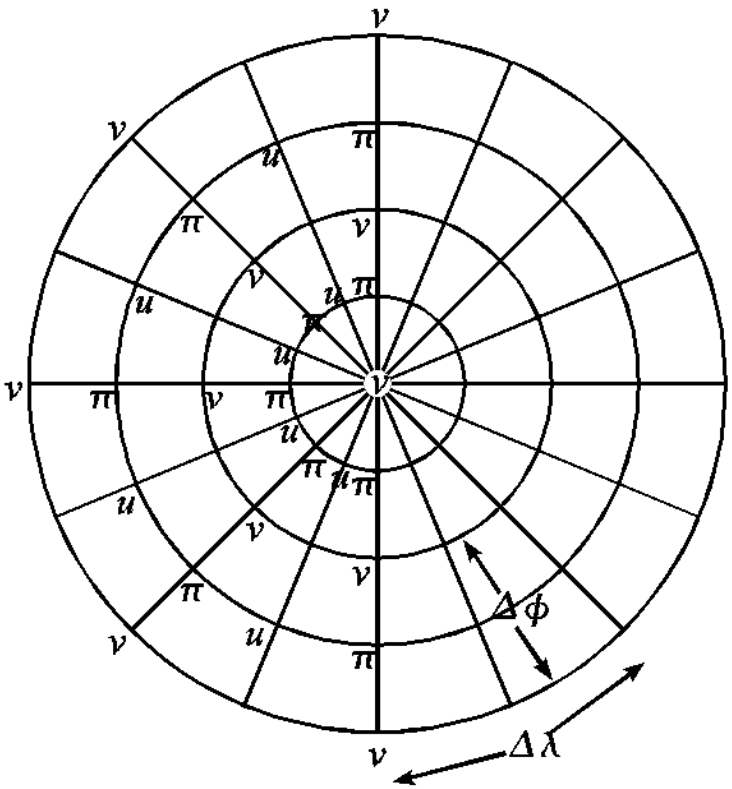
Outline:

- **Some features of GRAPES-global model**
- **Using GCR-CL scheme for solving Helmholtz equation**
- **Using PETSc tools for solving Helmholtz equation**
- **Some tests**
- **Summary and future work**

Main features of the GRAPES Global model

- Latitude-longitude grid point, spherical coordinate
- Non-hydrostatic/Hydrostatic optionally
- Height terrain-following coordinate
- Charney-Phillips variable staggering in vertical
- 2 time-level SISL scheme
- 3D preconditioned GCR solver for Helmholtz equation
- Arakawa-C grid horizontally: v at poles
- Polar filtering

Arakawa-C grid for horizontal arrangement of variables for GRAPES model:



Charney-Phillips vertical arrangement of variables:

	Charney-Phillips	Vertical level	Lorenz
Top level	$z, \hat{z}, \theta, w, \hat{w}$	$k = N$	$z, \hat{z}, w, \hat{w}$
	$z, \hat{z}, \rho, P, u, v$	$k = N - 0.5$	$z, \hat{z}, \theta, \rho, P, u, v$
	$z, \hat{z}, \theta, w, \hat{w}$	$k = N - 1.0$	$z, \hat{z}, w, \hat{w}$
	$z, \hat{z}, \rho, P, u, v$	$k = N - 1.5$	$z, \hat{z}, \theta, \rho, P, u, v$
	•	•	•
	•	•	•
	•	•	•
	$z, \hat{z}, \theta, w, \hat{w}$	$k = 2.0$	$z, \hat{z}, w, \hat{w}$
	$z, \hat{z}, \rho, P, u, v$	$k = 1.5$	$z, \hat{z}, \theta, \rho, P, u, v$
	$z, \hat{z}, \theta, w, \hat{w}$	$k = 1.0$	$z, \hat{z}, w, \hat{w}$
	$z, \hat{z}, \rho, P, u, v$	$k = 0.5$	$z, \hat{z}, \theta, \rho, P, u, v$
Earth's surface	$z, \hat{z}, \theta, w, \hat{w}$	$k = 0$	$z, \hat{z}, w, \hat{w}$

Prediction equation for GRAPES model:

$$u^{n+1} = [\xi_{u_1} \frac{1}{a\cos\varphi}\frac{\partial}{\partial\lambda} + \xi_{u_2}\frac{1}{a}\frac{\partial}{\partial\varphi} + \xi_{u_3}\frac{\partial}{\partial\hat{z}}](\Pi')^{n+1} + \xi_{u_0} \quad (1)$$

$$v^{n+1} = [\xi_{v_1} \frac{1}{a\cos\varphi}\frac{\partial}{\partial\lambda} + \xi_{v_2}\frac{1}{a}\frac{\partial}{\partial\varphi} + \xi_{v_3}\frac{\partial}{\partial\hat{z}}](\Pi')^{n+1} + \xi_{v_0} \quad (2)$$

$$\hat{w}^{n+1} = [\xi_{w_1} \frac{1}{a\cos\varphi}\frac{\partial}{\partial\lambda} + \xi_{w_2}\frac{1}{a}\frac{\partial}{\partial\varphi} + \xi_{w_3}\frac{\partial}{\partial\hat{z}}](\Pi')^{n+1} + \xi_{w_0} \quad (3)$$

$$(\theta')^{n+1} = \xi_{\theta_3}\frac{\partial}{\partial\hat{z}}(\Pi')^{n+1} + \xi_{\theta_0} \quad (4)$$

$$(\Pi')^{n+1} = \xi_{\Pi_1}u^{n+1} + \xi_{\Pi_2}v^{n+1} + \xi_{\Pi_3}\hat{w}^{n+1} + \xi_{\Pi_4}(D_3)_{\hat{z}} + A_{\Pi} \quad (5)$$

Equation (5) is Helmholtz equation . Solution of Helmholtz equation is key for GRAPES mode computing. Due to variables u、v、w、θ represented by function of π ,It's possible to solve other variables only after its solution.

Let the Helmholtz equation is: $A\vec{x} = \vec{b}_0$

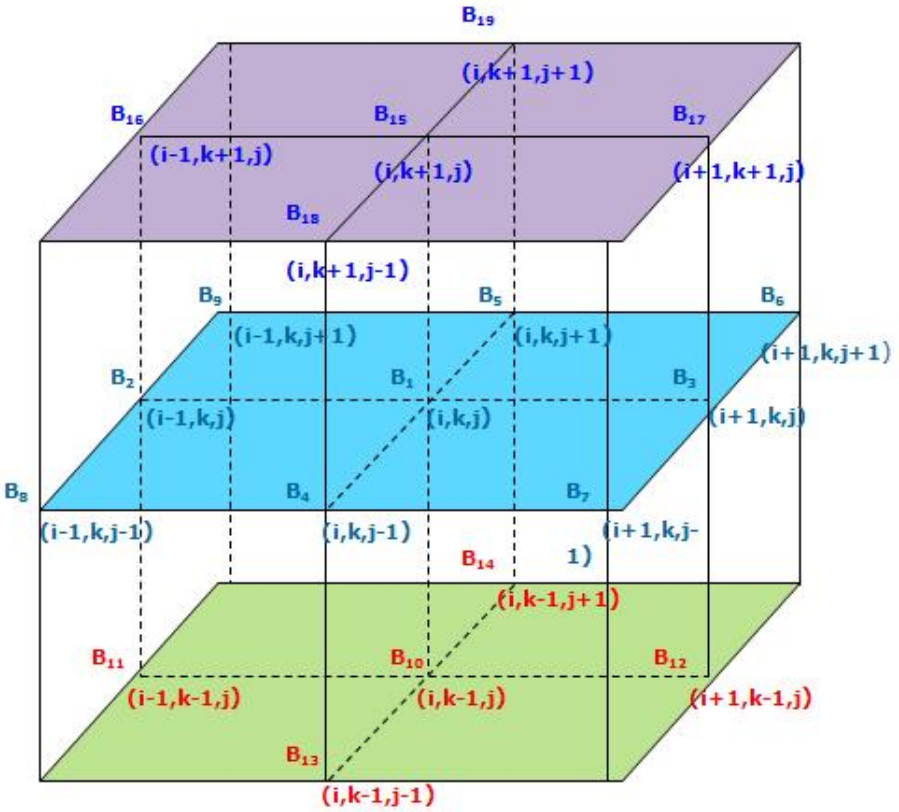
The approximate solution is: $\vec{x}_0$

The residual is: $\vec{r} = \vec{b}_0 - A\vec{x}_0$

The preconditioner is: M

So the Helmholtz equation with precondition is: $(M^{-1}A)\vec{x} = M^{-1}\vec{b}$

Arrangement of 19 coefficients for Helmholtz equation of GRAPES:



arrangement of 19 coefficients for Helmholts equation of Grapes

- ✓The coefficient matrix of helmholtz equation is a larger sparse matrix
- ✓The values of B_1,B_{10},B_{15},B_2,B_3 are more larger than others

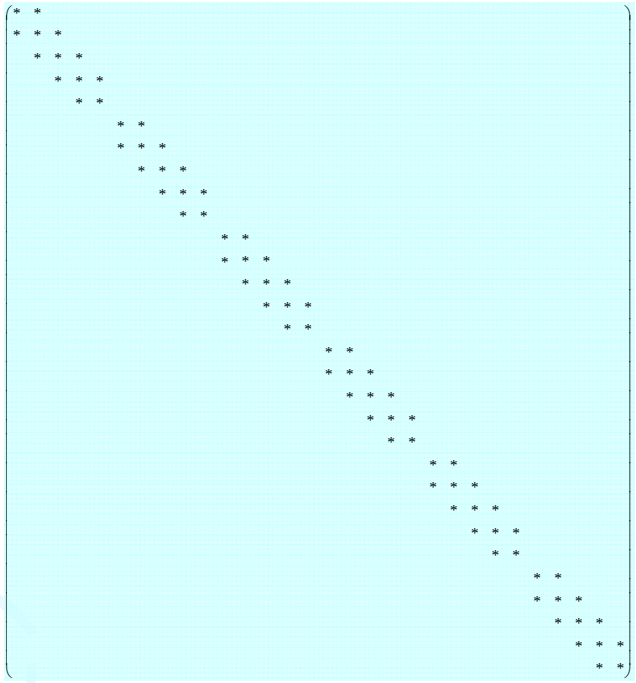
Precondition:

- a good preconditioner requires fewer iterations to converge for a given sparse linear system.

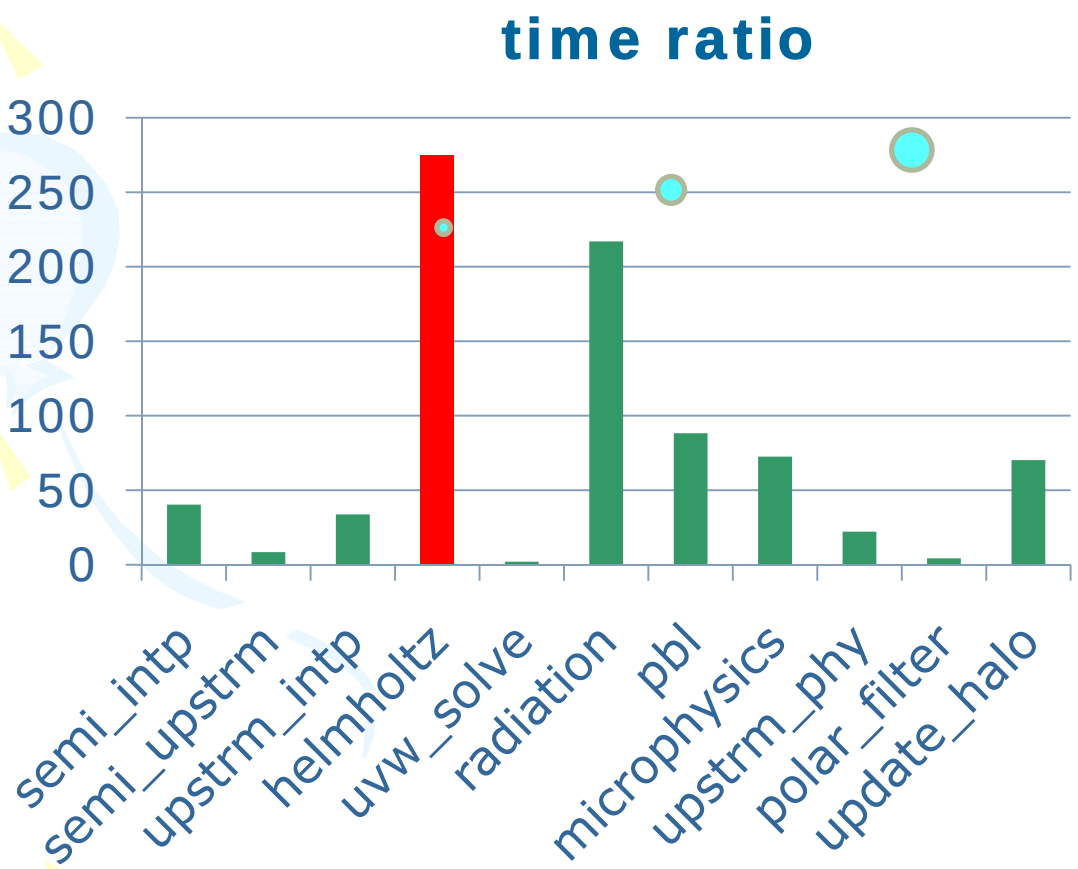
- Roughly speaking, a preconditioner is any form of implicit or explicit modification of the original coefficient matrix.

- For GRAPES model, choosing the main elements to build the preconditioner maybe a good ideal

For 19 coefficients, choose B1,B10,B15,B2,B3 to build the precondition matrix.



The escaped time ratio for main subroutines of 0.5º GRAPES Global model(1day forecast on 64)



The higher the model resolution, the higher the time ratio

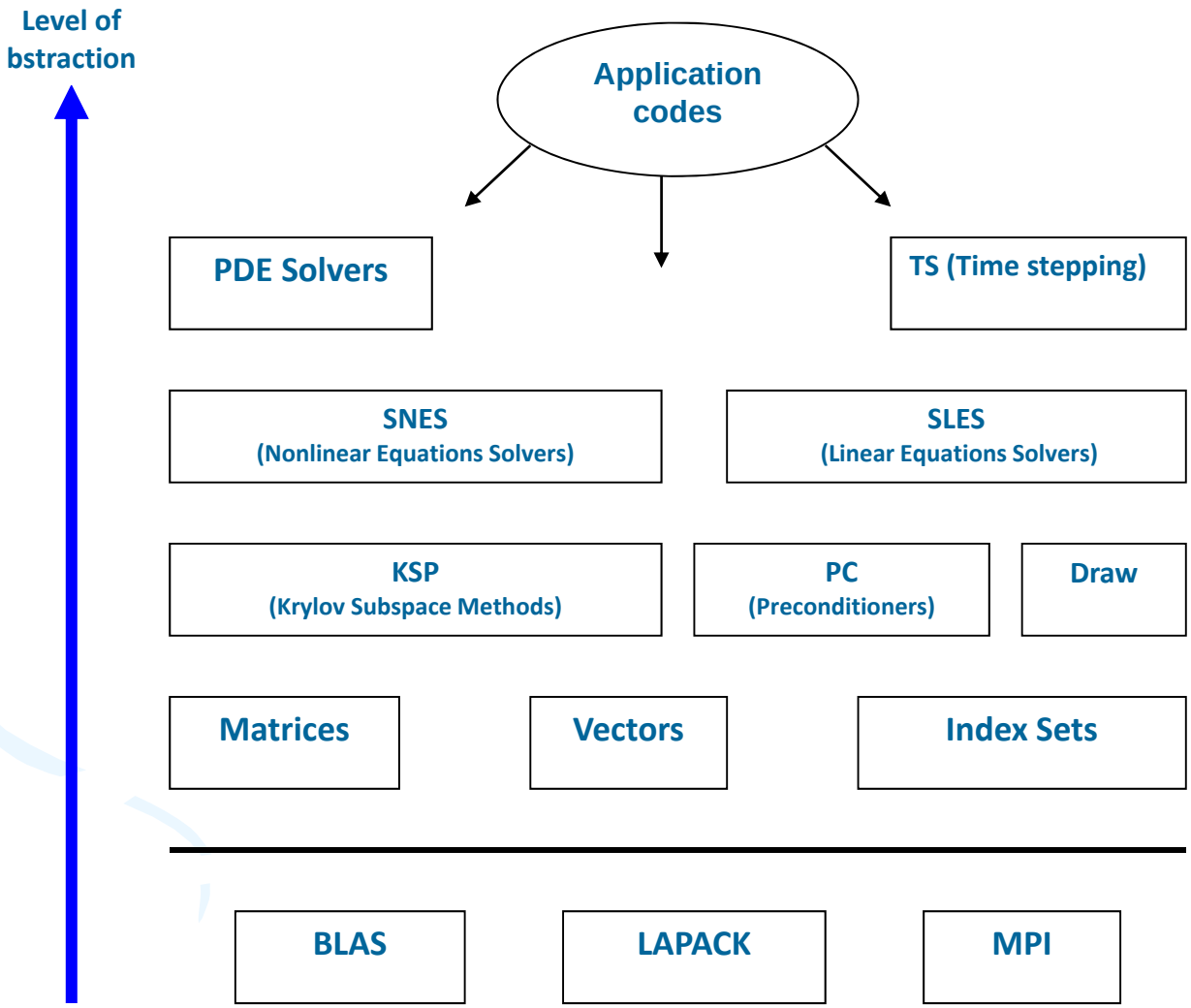
The Solver for Helmholtz equation of GRAPES model:

1. Using GCR(Generalised Conjugate Residual) with preconditioner algorithm (called it as gcr-cl now used in GRAPES);

2. Using PETSc ksp solver + hypre preconditioner。

What's PETSc? (http://www.mcs.anl.gov/petsc/petsc-as/) the Portable, Extensible Toolkit for Scientific Computation

PETSc is a suite of data structures and routines for the scalable (parallel) solution of scientific applications modeled by partial differential equations. It consists of a variety of libraries which manipulates a particular family of objects(such as vectors, matrix, distributed arrays, Krylov subspace method, preconditioners, et).



Organization of the PETSc Libraries

Parallel Numerical Components of PETSC

Nonlinear Solvers		
Newton-based Methods		other
Line Search	Trust Region	

Time Steppers			
Euler	Backward Euler	Pseudo-Time Stepping	other

Krylov Subspace Method							
GMRES	CG	CGS	Bi-CG-Stab	TFQMR	Richardson	Chebyshev	other

Precondition						
Additive Schwarz	Block Jacobi	Jacobi	ILU	ICC	LU (sequential only)	other

Matrices				
Compressed Sparse Row (AIJ)	Block Compressed Sparse Row (BAIJ)	Block Diagonal (BDiag)	Dense	other

Vectors

Index Sets			
Indices	Block Indices	Stride	other

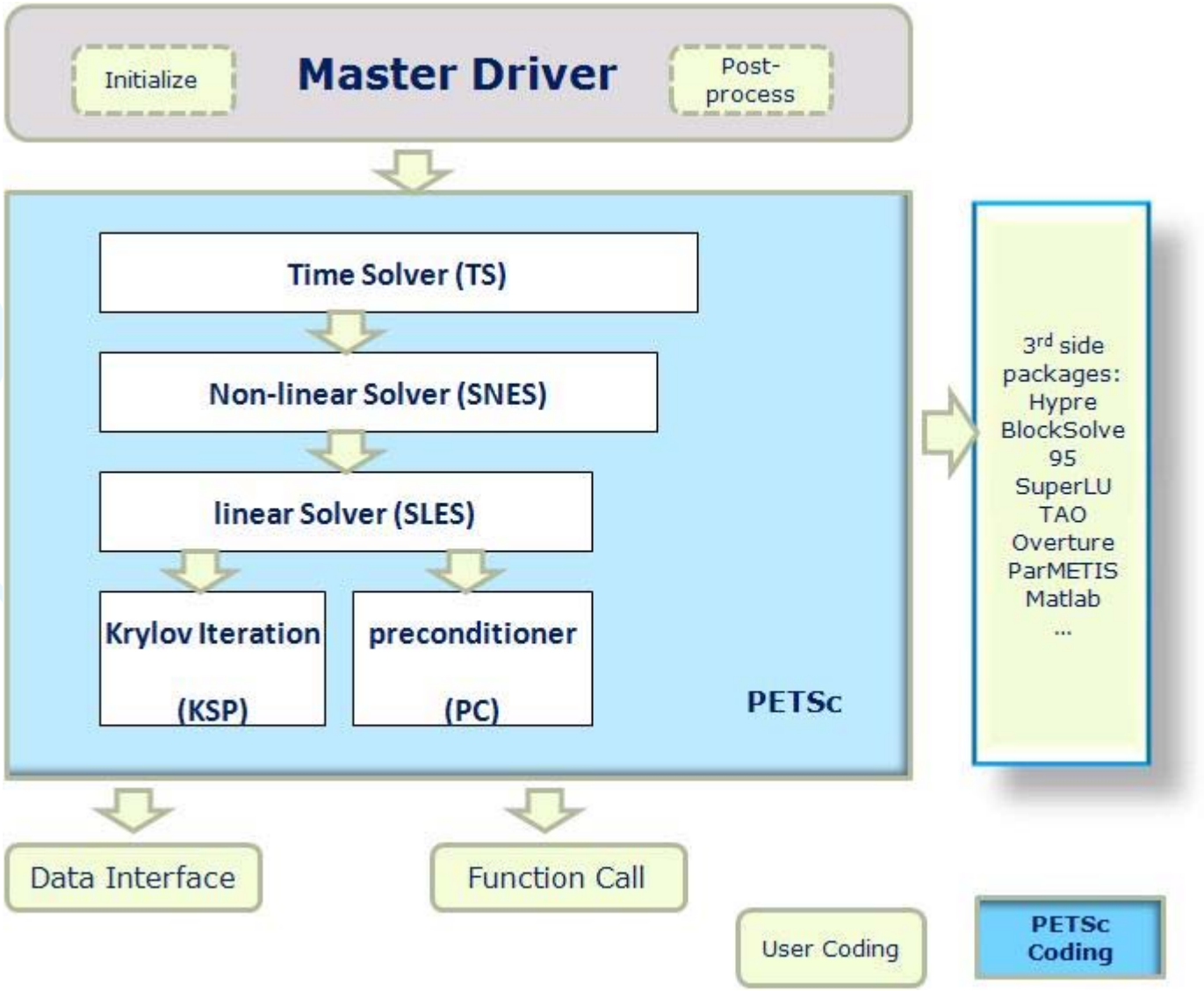
Numerical Libraries of PETSc

What's Hypre? (http://acts.nersc.gov/hypre/)

high performance preconditions

Hypre is a library for solving large, sparse linear systems of equations on massively parallel computers. Hypre contains several families of preconditioner algorithms focused on the scalable solution of very large sparse linear systems. These algorithms include structured multigrid and element-based algebraic multigrid .

User View of PETSc

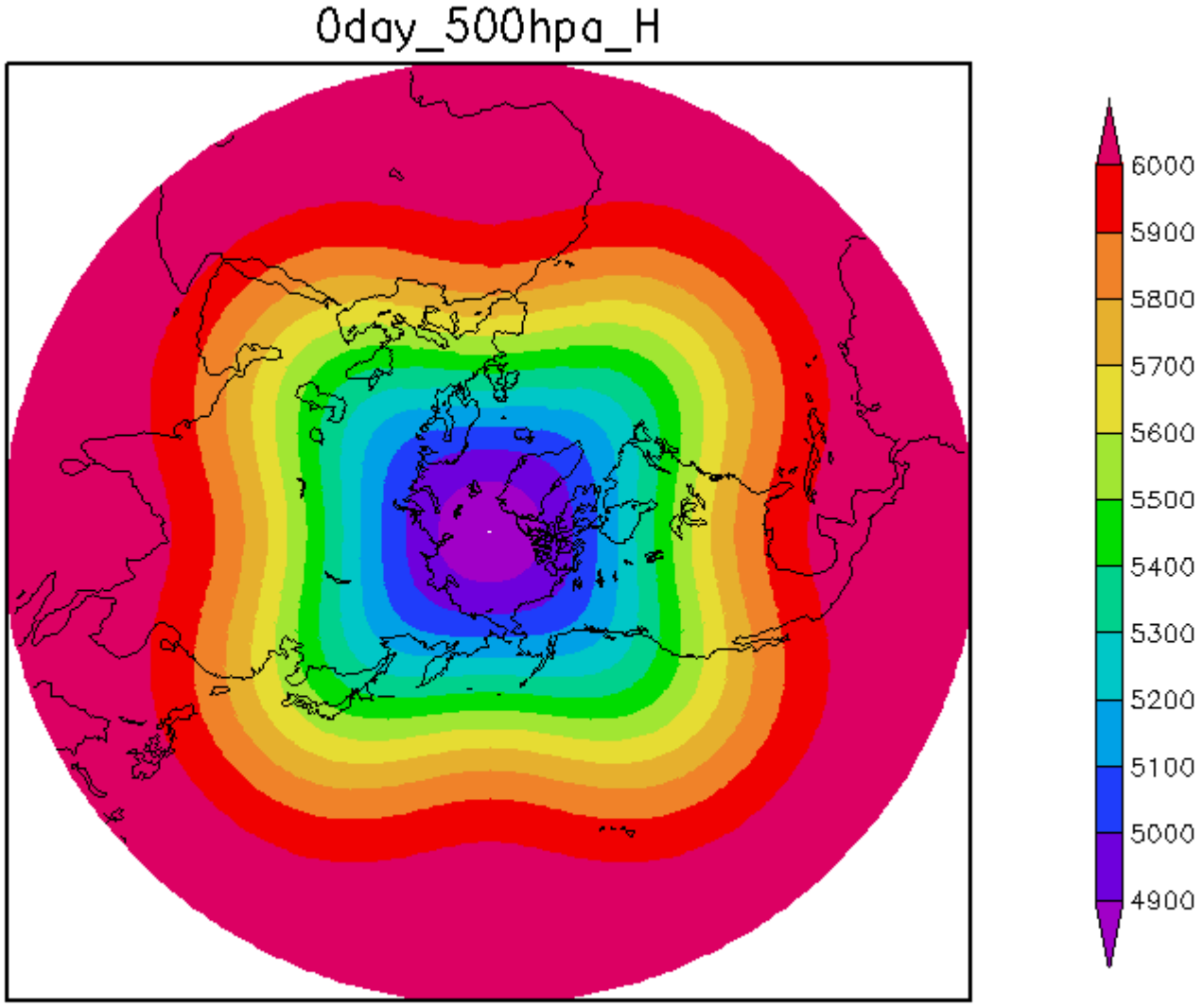


A example subroutine using PETSc functions:

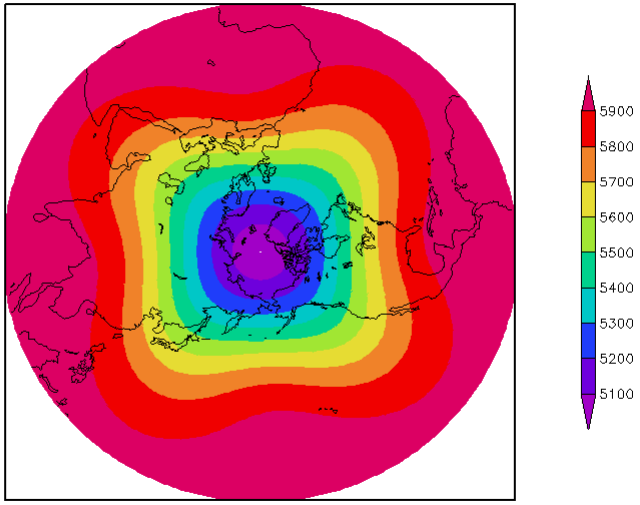
```
…
If(nstep==1)
    Call PetscInitialize()
    …
    Call MatCreatMPIAIJ()
    …
    Call set_A-petsc()
   …
    Call KspCreate()
   …
Endif
…
Call set_b_petsc()
…
Call kspsolver()
…
Call PetscFinalize()
stop
```

Test for Rossby-Haurwitz wave

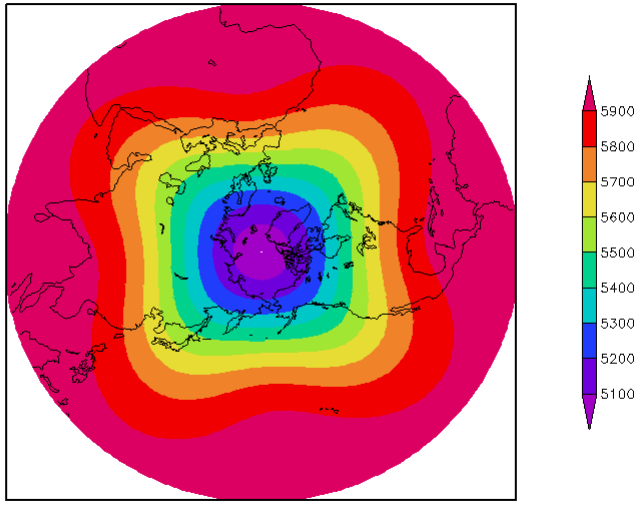
- **1.0º resolution for global model (grid size:360x180x36)**
- **The residual l_2-norm $<10^{-9}$ for convergence**
- **Main variables are double precision**
- **Run on IBM-cluster**
- **Use 64 MPI tasks**
- **80days forecast**
- **For PETSc scheme, use GMRES+** BoomerAMG(hypre)


0day_500hpa_H
6000
5900
5800
5700
5600
5500
5400
5300
5200
5100
5000
4900

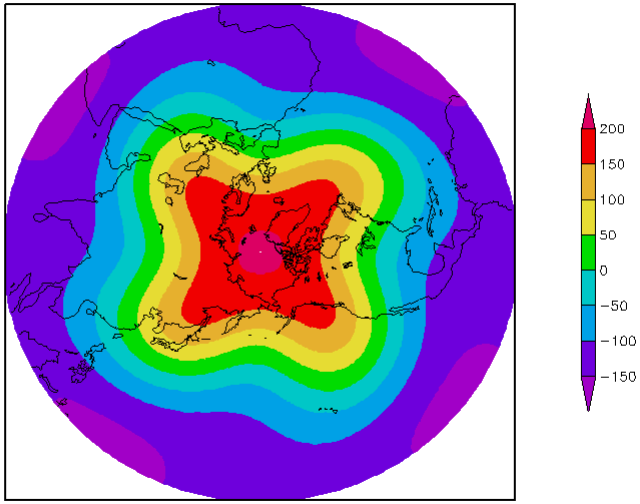
30day_gcr_500hpa_H



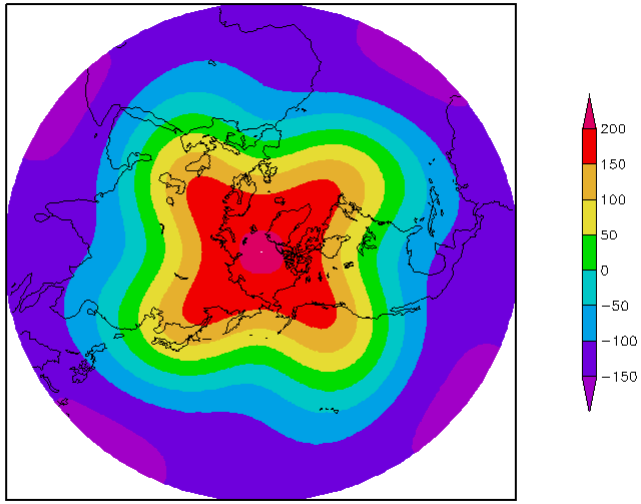
30day_gmres_500hpa_H



30day-0day_gcr_500hpa_H

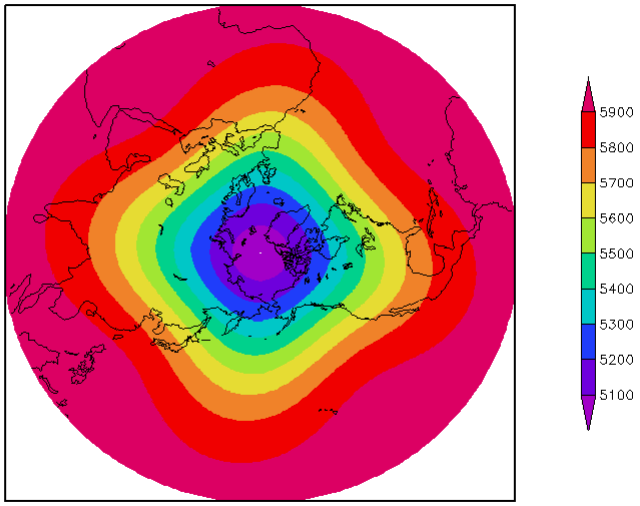


30day-0day_gmres_500hpa_H

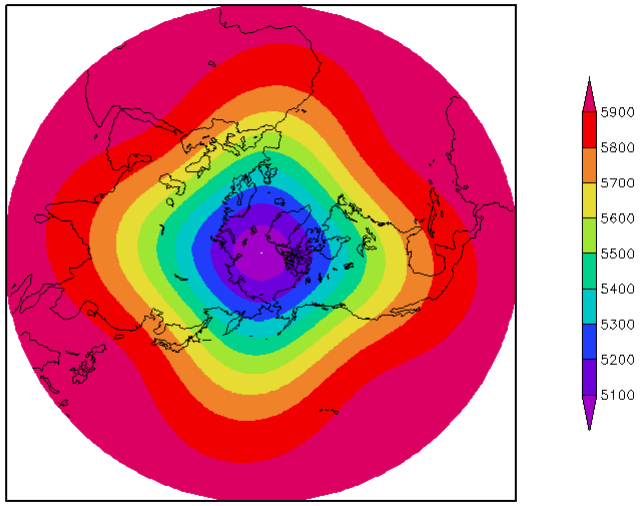


Global and Regional Assimilation and PrEdiction System

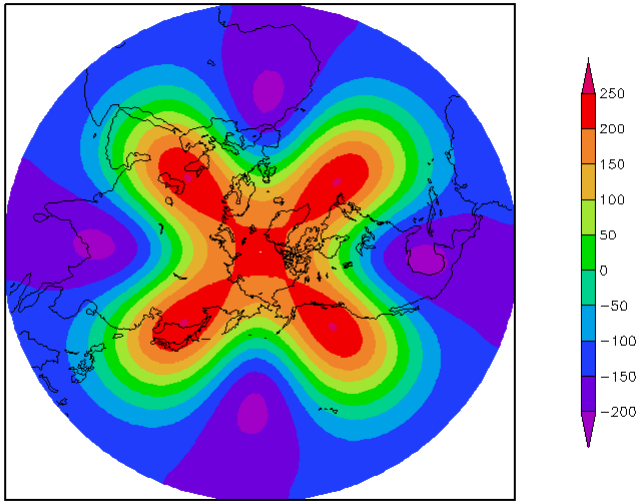
50day_gcr_500hpa_H



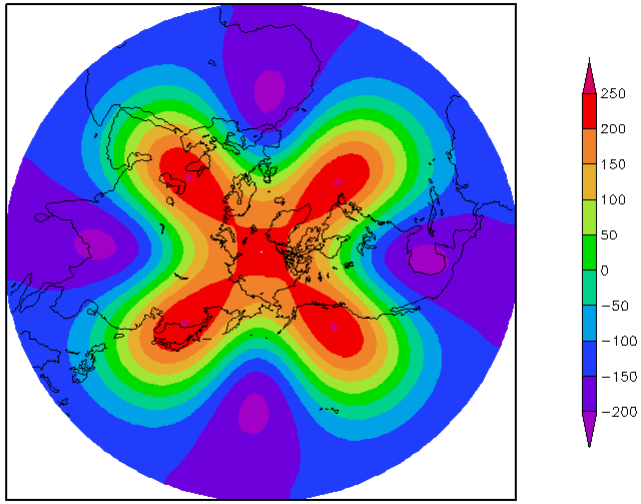
50day_gmres_500hpa_H



50day-0day_gcr_500hpa_H

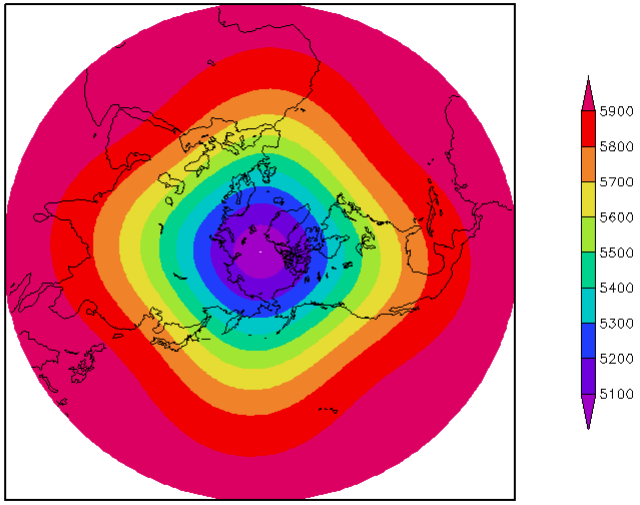


50day-0day_gmres_500hpa_H

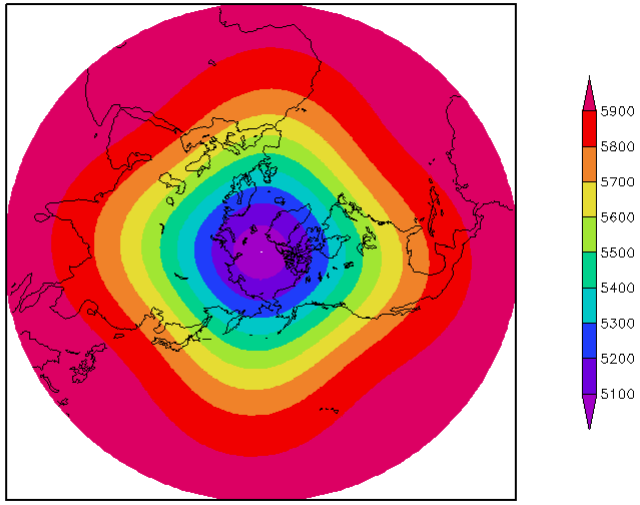


Global and Regional Assimilation and PrEdiction System

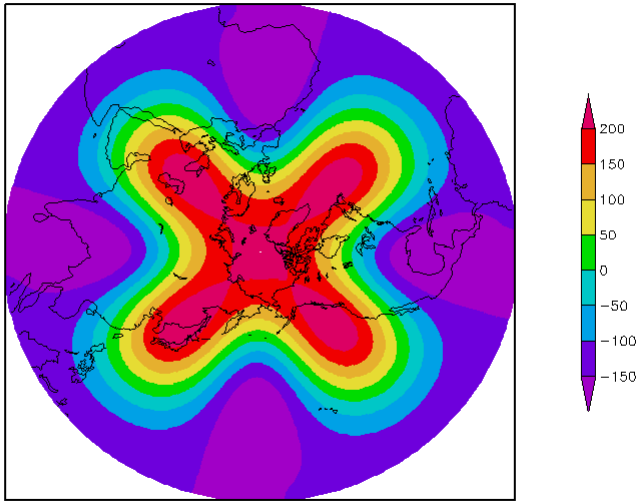
80day_gcr_500hpa_H



80day_gmres_500hpa_H



80day-0day_gcr_500hpa_H



80day-0day_gmres_500hpa_H

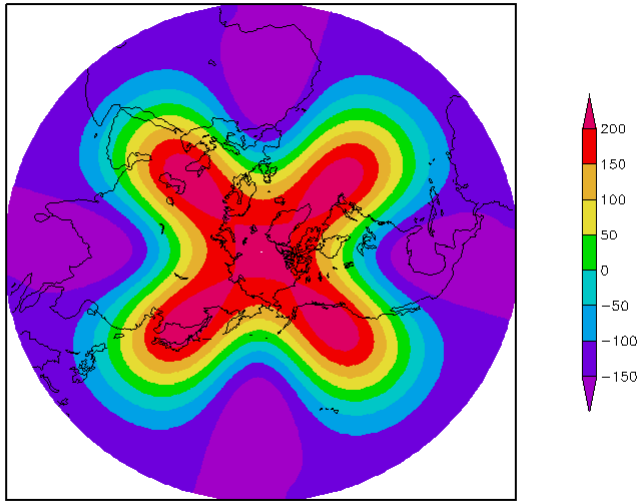


Table1：Escaped Time for 80day forecast of Rossby-Haurwitz wave with 19 coefficients on IBM-cluster1600

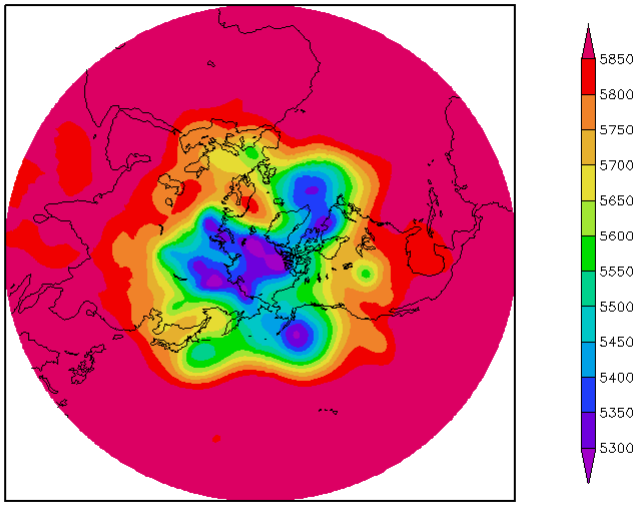
scheme	gcr-cl		Petsc	
	Time helm/total	Number of iterations	Time helm/total	Number of iterations
Step 1	5.11/7.71	275	9.43/17.24	12
Step 520	3.70/4.66	196	2.36/3.32	12
Step 5319	2.27/3.23	147	2.38/3.34	12
Wall clock	5h37m11.40s		5h21m30s	

Test for real data

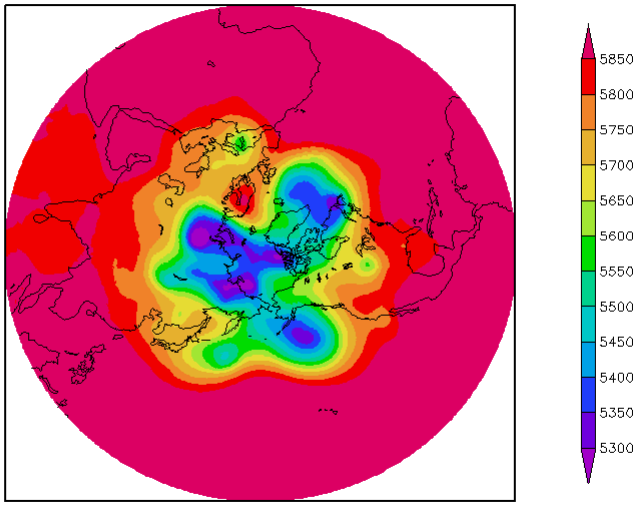
- 0.5° resolution for global model (grid size:720x361x36)
- Main variables are double precision
- On IBM-cluster 1600
- Use 64 processors
- Initial data: 31 may 2007
- 60 min for radiation
- 1days forecast， 144 steps, DT=600s
- For petsc scheme, use GMRES+ BoomerAMG(hypre)
- Test1 for convergence using residual l_2-norm $<10^{-7}$
- Test2 for convergence using residual l_2-norm $<10^{-12}$

Global and Regional Assimilation and PrEdiction System

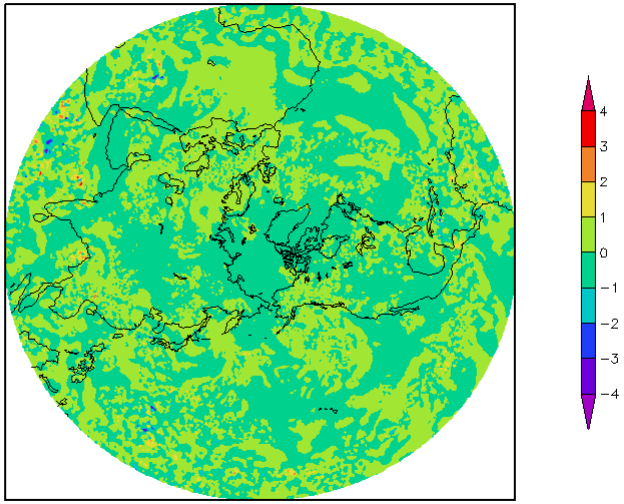
0day_500hpa_H



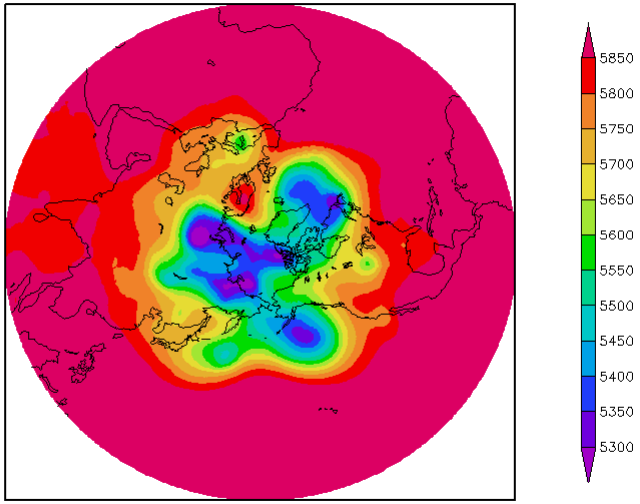
1day_500hpa_H-org



Diff_1day_500hpa_H(org-petsc)



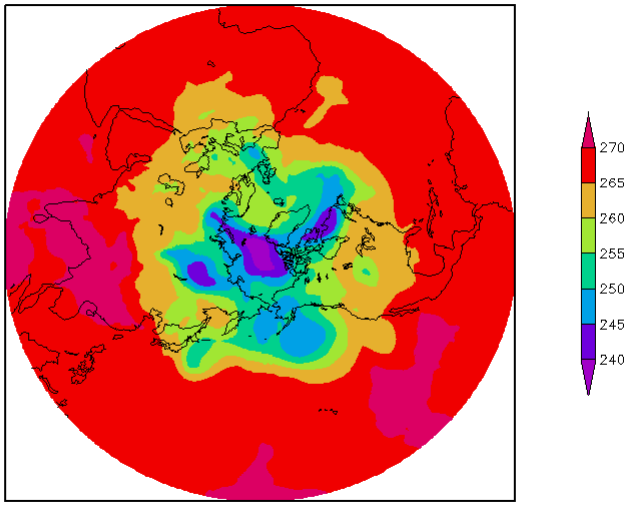
1day_500hpa_H-petsc



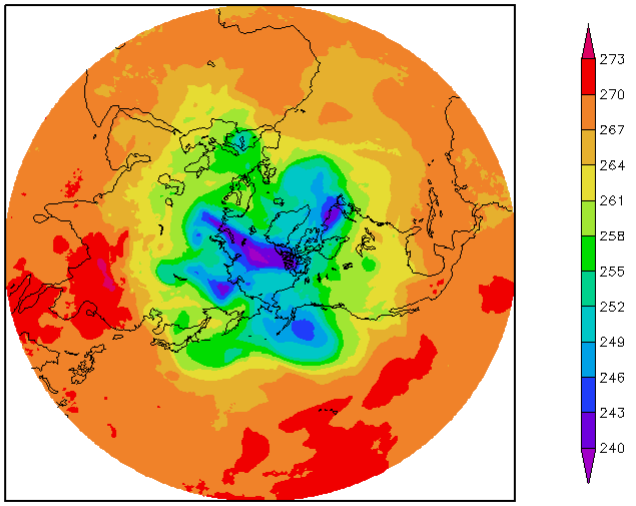
test1

Global and Regional Assimilation and PrEdiction System

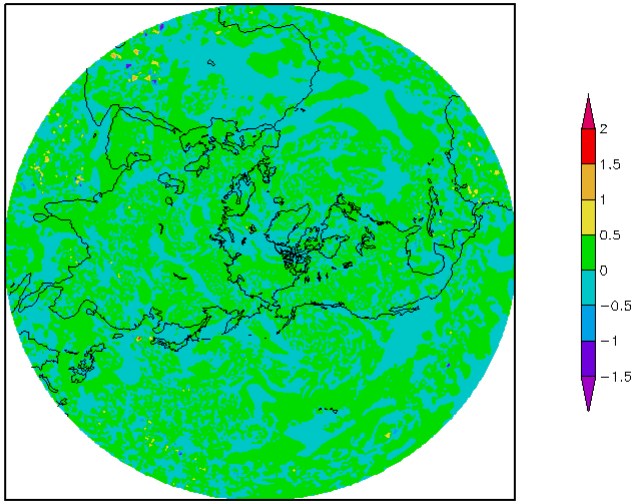
0day_500hpa_T



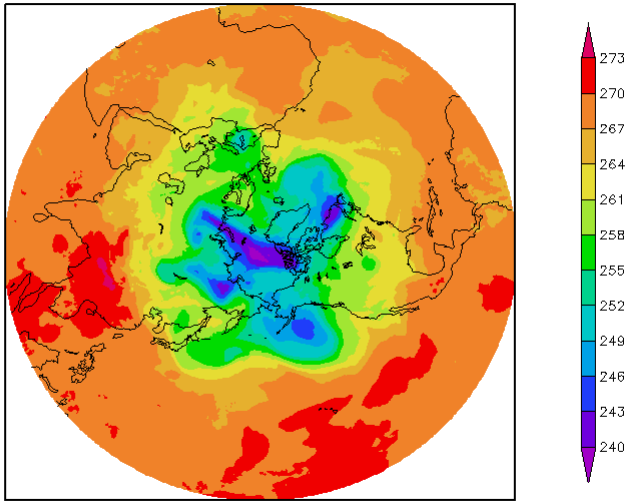
1day_500hpa_T-org



Diff_1day_500hpa_T(org-petsc)

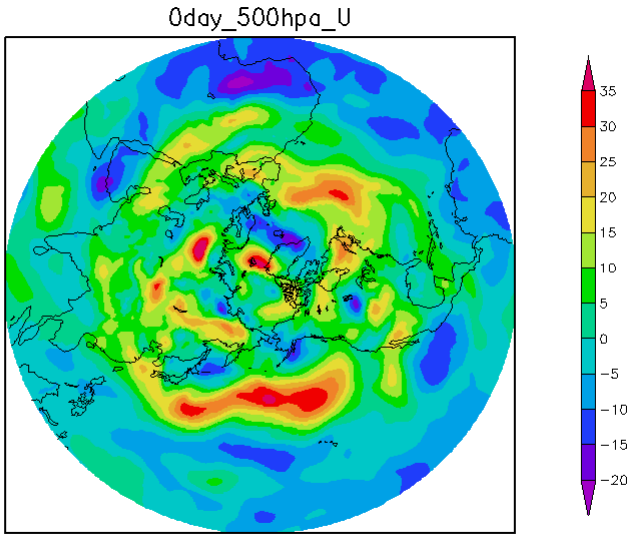


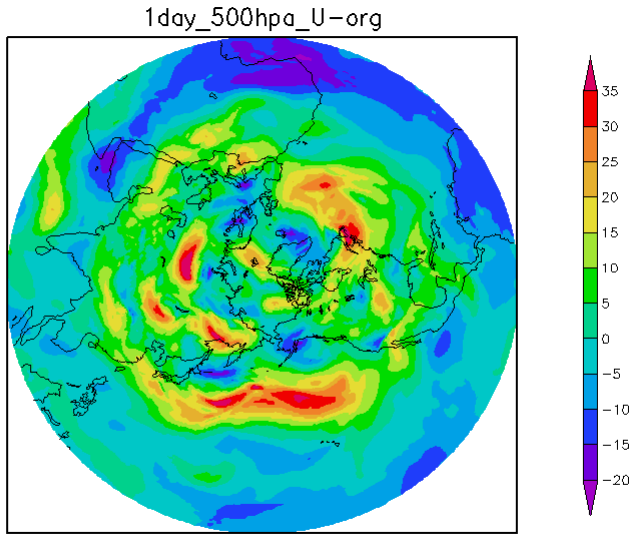
1day_500hpa_T-petsc

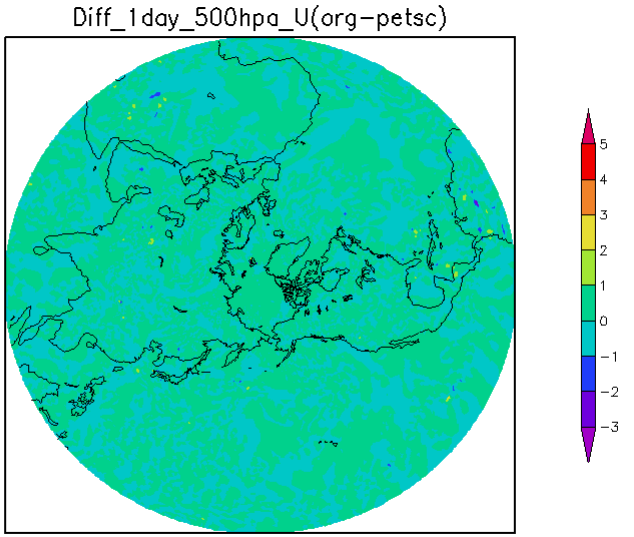


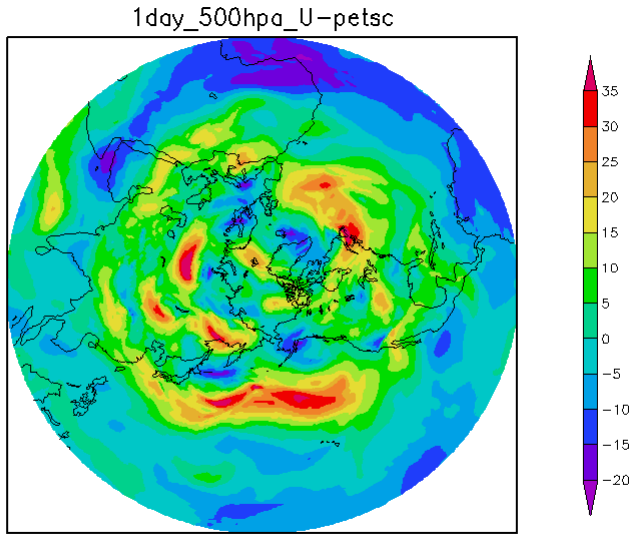
test1

Global and Regional Assimilation and PrEdiction System



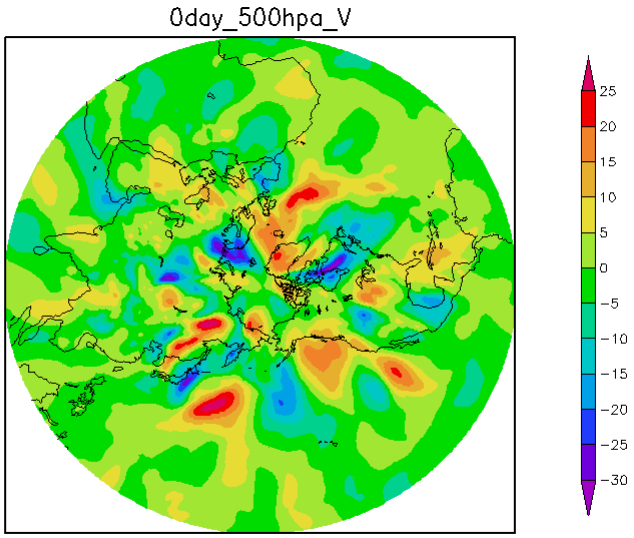


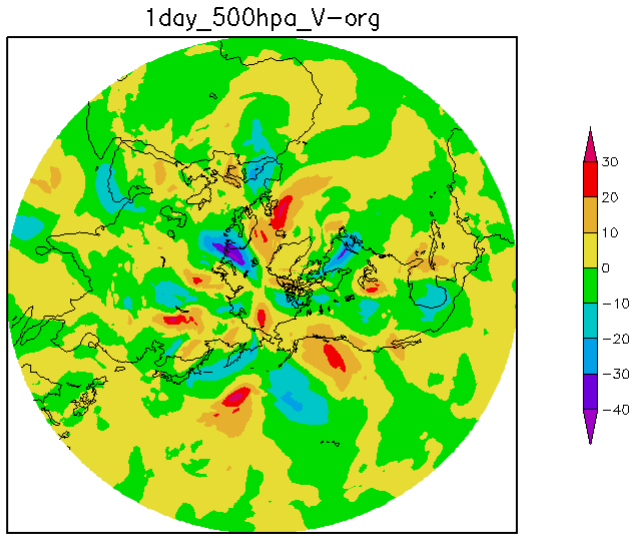


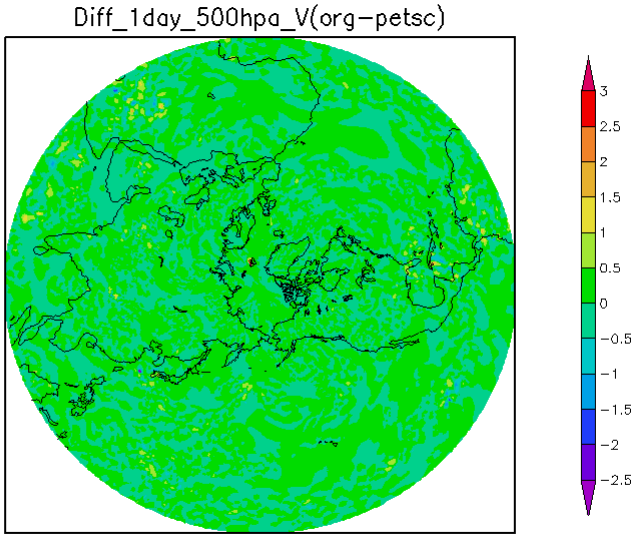


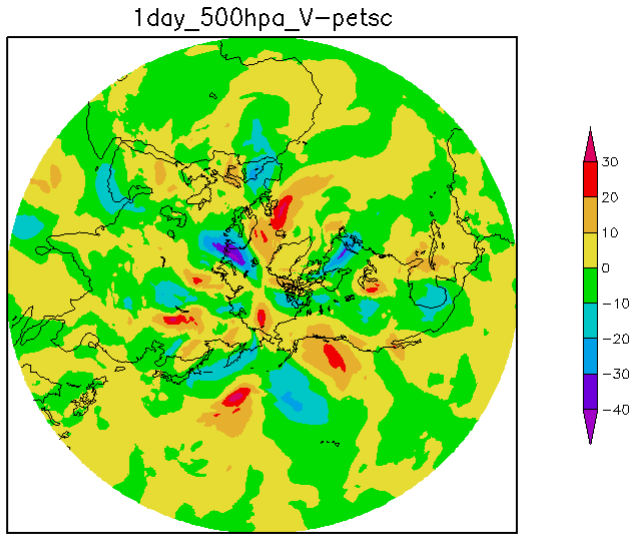
test1

Global and Regional Assimilation and PrEdiction System

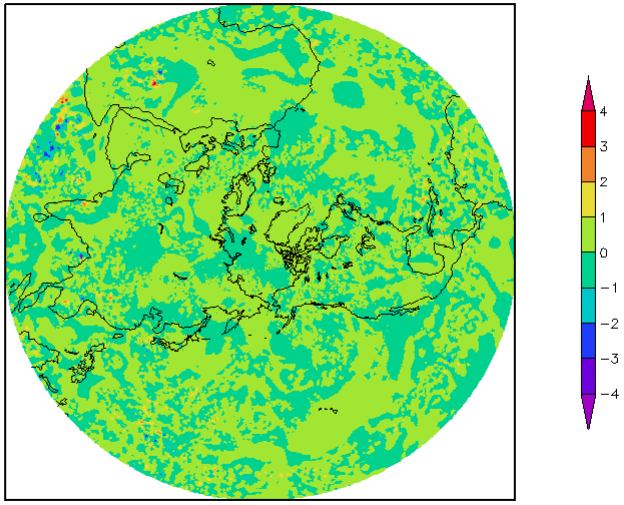




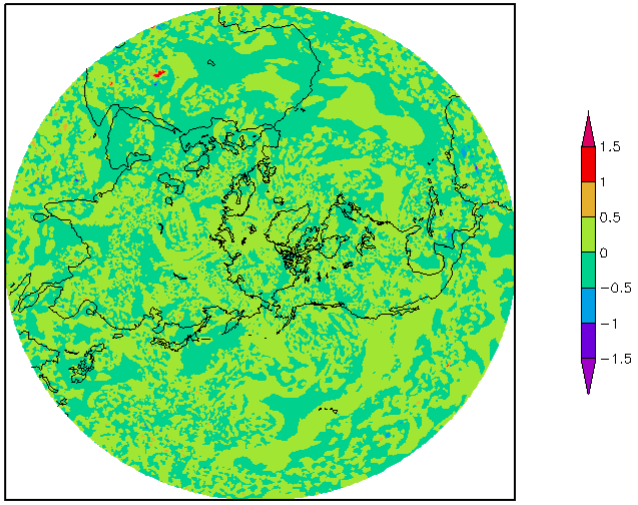




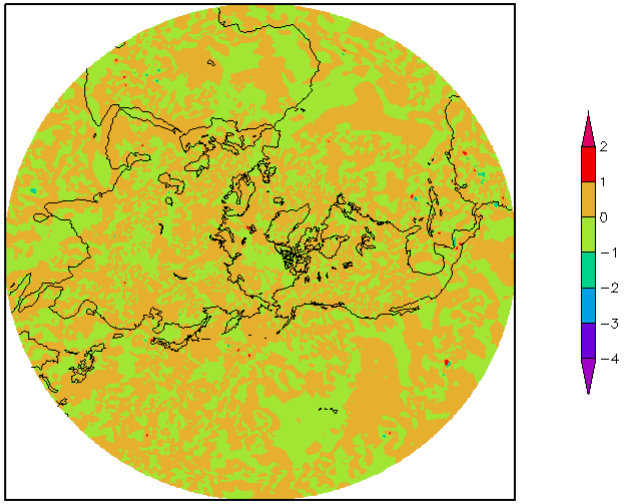
Diff_1day_500hpa_H(org-petsc)



Diff_1day_500hpa_T(org-petsc)



Diff_1day_500hpa_U(org-petsc)



Diff_1day_500hpa_V(org-petsc)

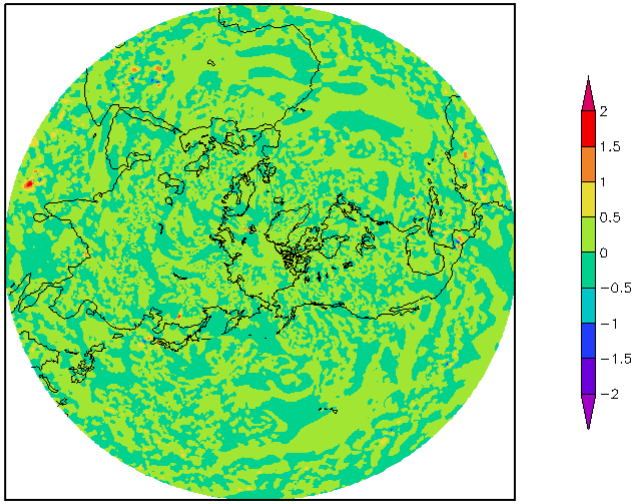


Table2(test1)：Escaped Time for test of real data on IBM-cluster 1600(residual l^2-norm $<10^{-7}$)

convergence	E07-d-org		E07-petsc	
	Time helm/total	Number of iterations	Time helm/total	Number of iterations
Step 1	19.01/80.56	184	62.56/122.9	26
Step 42	15.32/46.58	147	26.64/57.73	29
Step 137	13.45/23.37	147	26.63/36.58	28
Wall clock	1h13m53.93s		1h44m12.95s	

Table3(test2)：Escaped Time for test of real data on IBM-cluster1600 (residual l^2-norm $<10^{-12}$)

convergence	E12-d-org		E12-petsc	
	Time helm/total	Number of iterations	Time helm/total	Number of iterations
Step 1	45.61/107.76	623	84.80/145.99	50
Step 42	38.16/69.33	519	47.99/79.13	53
Step 137	37.29/47.22	511	47.09/57.04	52
Wall clock	2h12m22.18s		2h33m34.26s	

Fig: One step time of helmholtz solver for different number of processor (residual l^2-norm $<10^{-12}$)

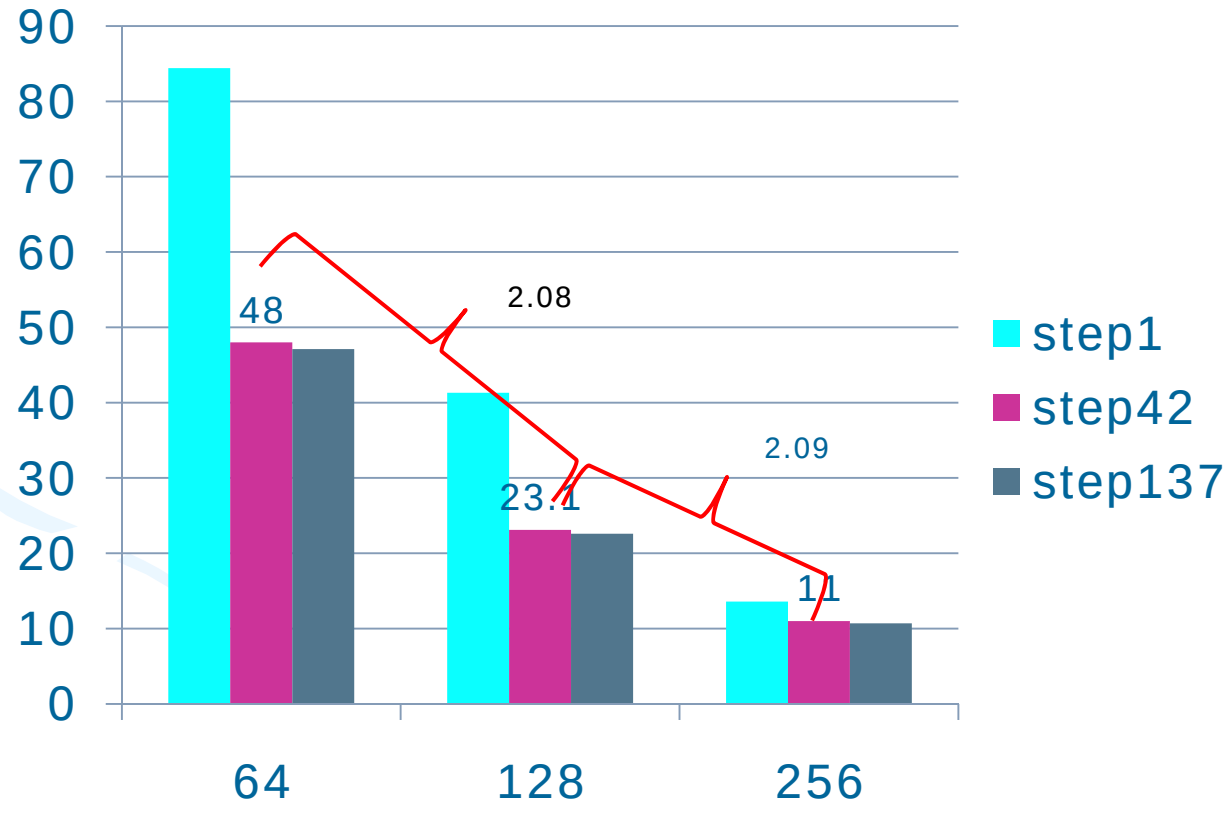


Fig: wall clock of 1day forecast and the efficiency for different number of processor (residual l^2-norm $<10^{-12}$)

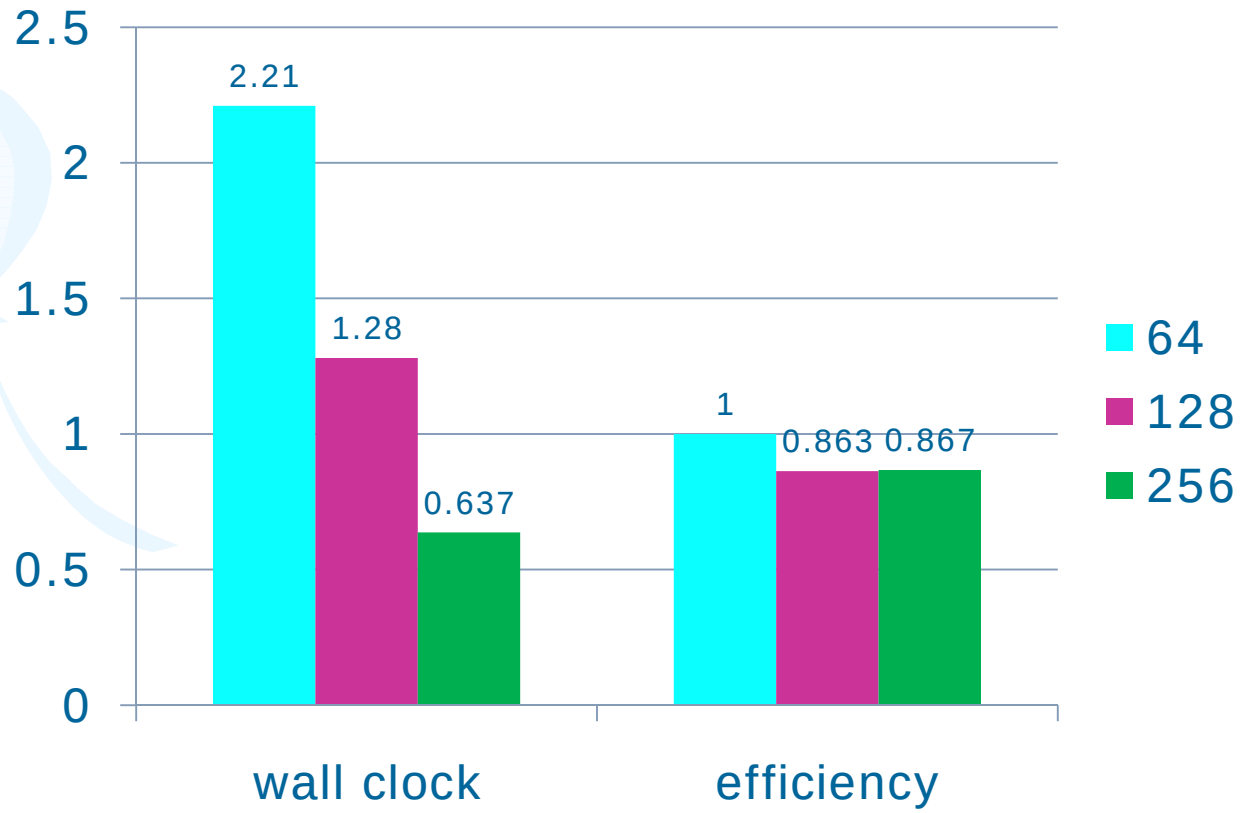


Table4: Escaped Time for real data test on Intel based machine (residual l2-norm $<10^{-12}$)

Number processor	Original scheme	PETSc tools
	Time helm for one step	Time helm for one step
1		
4	171.885	
8	155.88	53.18
16	83.04	25.16
32	42.64	13.93
64	25.47	8.12
128	21.14	3.97
256	11.24	2.16

summary:

- *Good Solution for Helmholtz equation using* PETSc tools
- Same convergence precision, fewer iterations *using* PETSC for Solution of *Helmholtz equation*
- *The solution for Helmholtz equation using* PETSc has more well scalability
- The running time of two scheme is comparable, and at some platform using PETSc tools were more better.

Future work:

- Testing more cases and give the profile.
- Testing different combination of hypre preconditioner plus GRMES
- Testing with large number processors (more than 256 etc.) under high-resolution model.
- Optimize PETSc solution for Helmholtz equation.
- Comparable research on parallel algorithm library of preconditioner except hypre, such as Trillions and SuperLU-dist, etc.

Thank you!