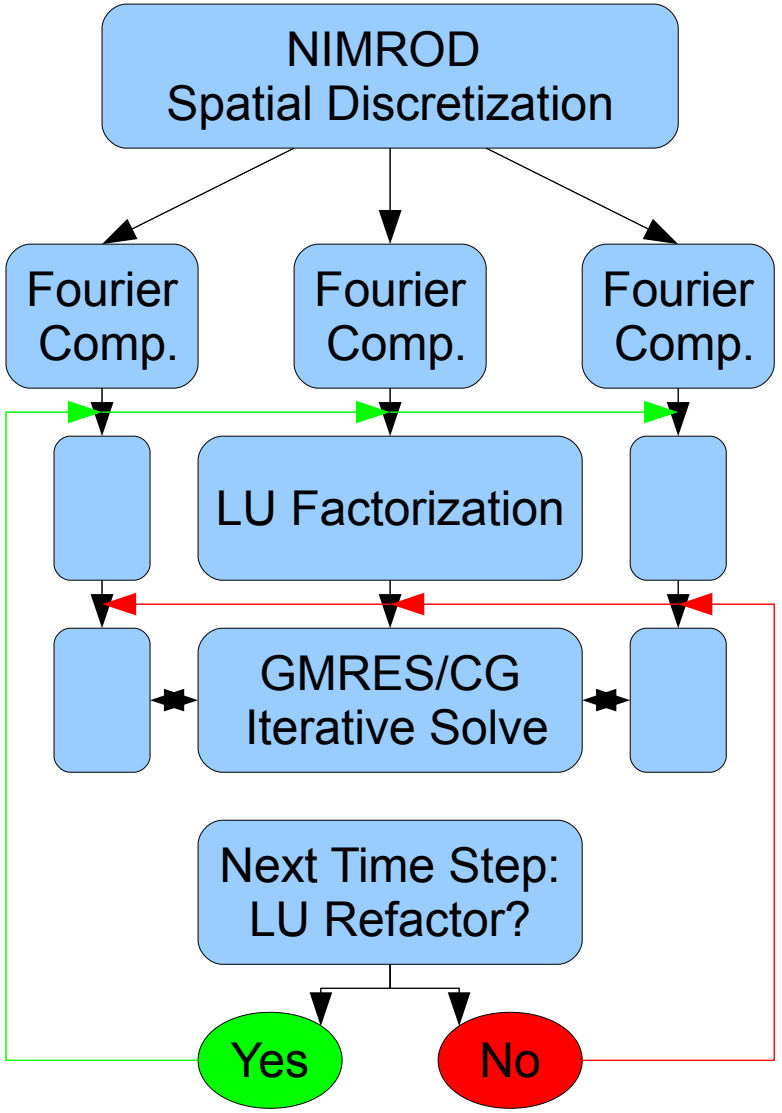
Exploring ILU Factorization with SuperLU 4.0

Jacob King
Carl Sovinec

NIMROD Team Meeting
7/29/09

NIMROD solves linear systems using the GMRES/CG algorithm with SuperLU as a preconditioner.



- In NIMROD RZΦ representation, the Φ direction is decomposed into Fourier components.
- A linear system of the form Mx=b is constructed for each (2D) Fourier component.
- x is the unknown solution vector that advances the time step.
- SuperLU finds the LU decomposition of M, to be used as a preconditioner for GMRES/CG.
- GMRES/CG then iteratively solves the full 3D system.

Sequential SuperLU 4.0 supports incomplete LU factorization.

- Currently NIMROD uses complete LU factorization.
  - ILU was implemented for linear elements in iter_cg_*.f but is not currently used.
- Incomplete factorization is now available in sequential SuperLU 4.0.
- Incomplete factorization drops LU matrix elements smaller than a drop tolerance, τ, times the maximum element in that row.
  - This reduces memory usage.
  - Leads to less arithmetic on matrix solves. (reduces run-time)
  - However, GMRES/CG may not converge as fast, and may require more iterations. (increases run-time)
  - With a distributed memory version it should scale better in parallel.
- A variety of other options are available in ILU SuperLU 4.0, including secondary dropping, as well as adaptive methods that set a limit on the size of LU.
- Parallelization is still possible, but only through layer decomposition. (SuperLU 4.0 is not available in a distributed memory form, yet)

The SuperLU bridge routines are modified to implement ILU decomposition.

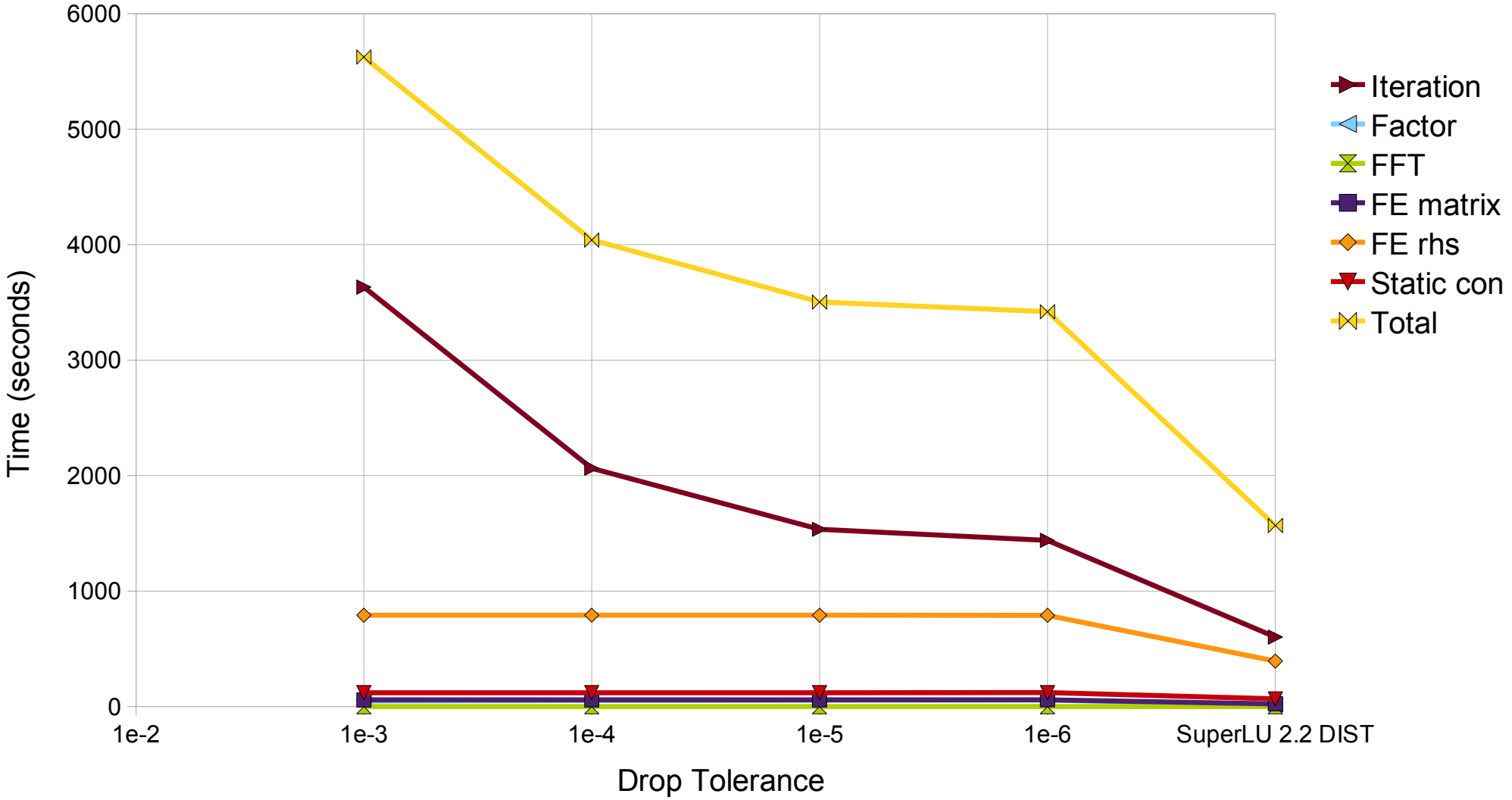
- The c function call name for factorization is modified, but the basic structure is similar.
  - externals/c_fortran_zgssv.c → externals/c_fortran_zgsisx.c
  - and **zgstrf()** → **zgsitrf()**
- As a starting point, the default options are set using **ilu_set_default_options()**
- Additional documentation on SuperLU and the ILU implementation is available at: http://crd.lbl.gov/~xiaoye/SuperLU/
- Bridge routines are not the standard ones included in SuperLU 3.0/4.0, an option to refactor without new memory allocation has been restored.

Test case 1: Linear two-fluid tearing mode

- S=5000
- P_m=0.1
- advect='V only'
- Ohms='2fl'
- nonlinear=.false.
- cont='full'
- p_model='isotropic'
- Θ=1.55
- 60x15 RZ mesh
- geom='tor'
- poly_degree=4
- lin_nmodes=1
- gridshape='rect'
- periodicity='y-dir'

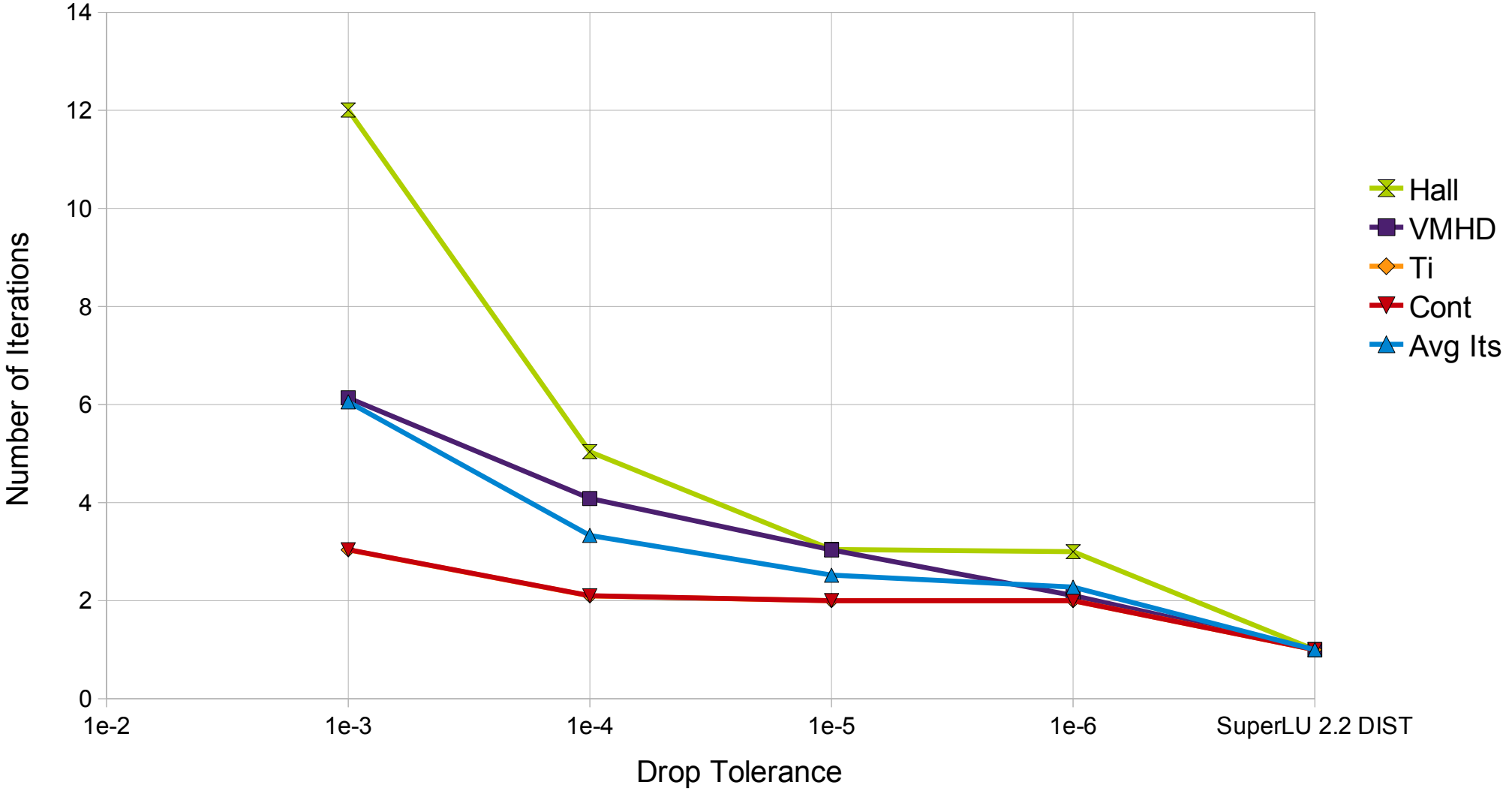
- Run with ILU decomposition for 2000 time steps
- τ, the drop tolerance is scanned for 10^{-2} to 10^{-6}.
- The growth rate is $\gamma\tau_a$=0.02495 for all cases excluding τ=10^{-2}.
- divbd=1 (elecd=2e-4) and split_divb=.false. are used, more on this later.
- All cases run on a 64-bit Intel Xeon 3.0GHz machine with 8Gb RAM.

As τ is decreased, the job takes longer to complete.



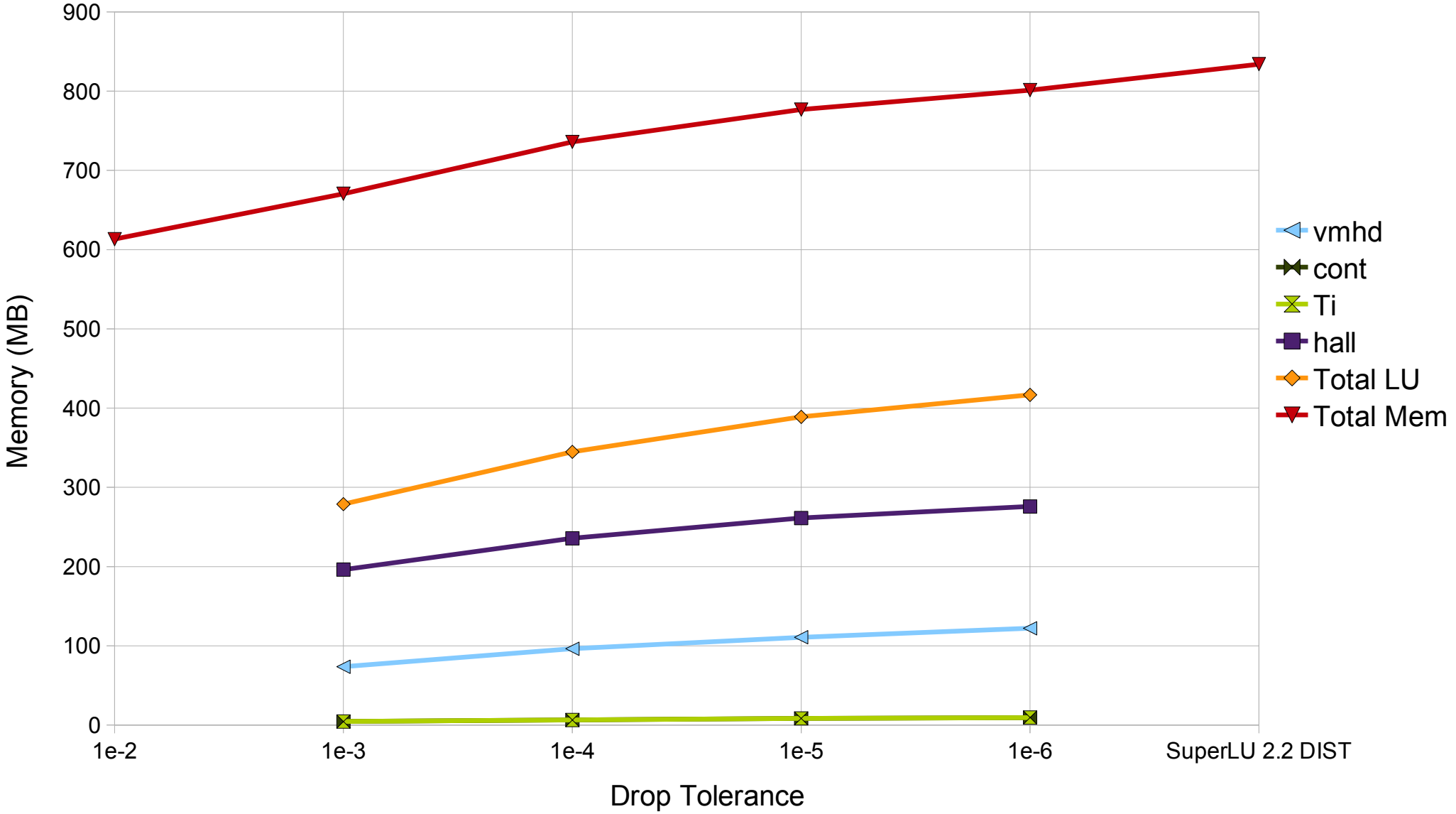
With $\tau=10^{-2}$, there is convergence is either slow, or problematic thus times are not meaningful for comparison. SuperLU 2.2 DIST case uses 2 cpu cores, whereas the others are using only 1.

The induction equation solve requires many more iterations at $\tau=10^{-3}$ and does not converge initially for $\tau=10^{-2}$.



With $\tau=10^{-2}$, there is convergence is either slow, or problematic thus times are not meaningful for comparison. SuperLU 2.2 DIST case uses 2 cpu cores, whereas the others are using only 1.

The ILU decomposition uses up to 26% less memory.



divbd must be set large enough to remove Div(B), but not far greater than elecd with the ILU decomposition.

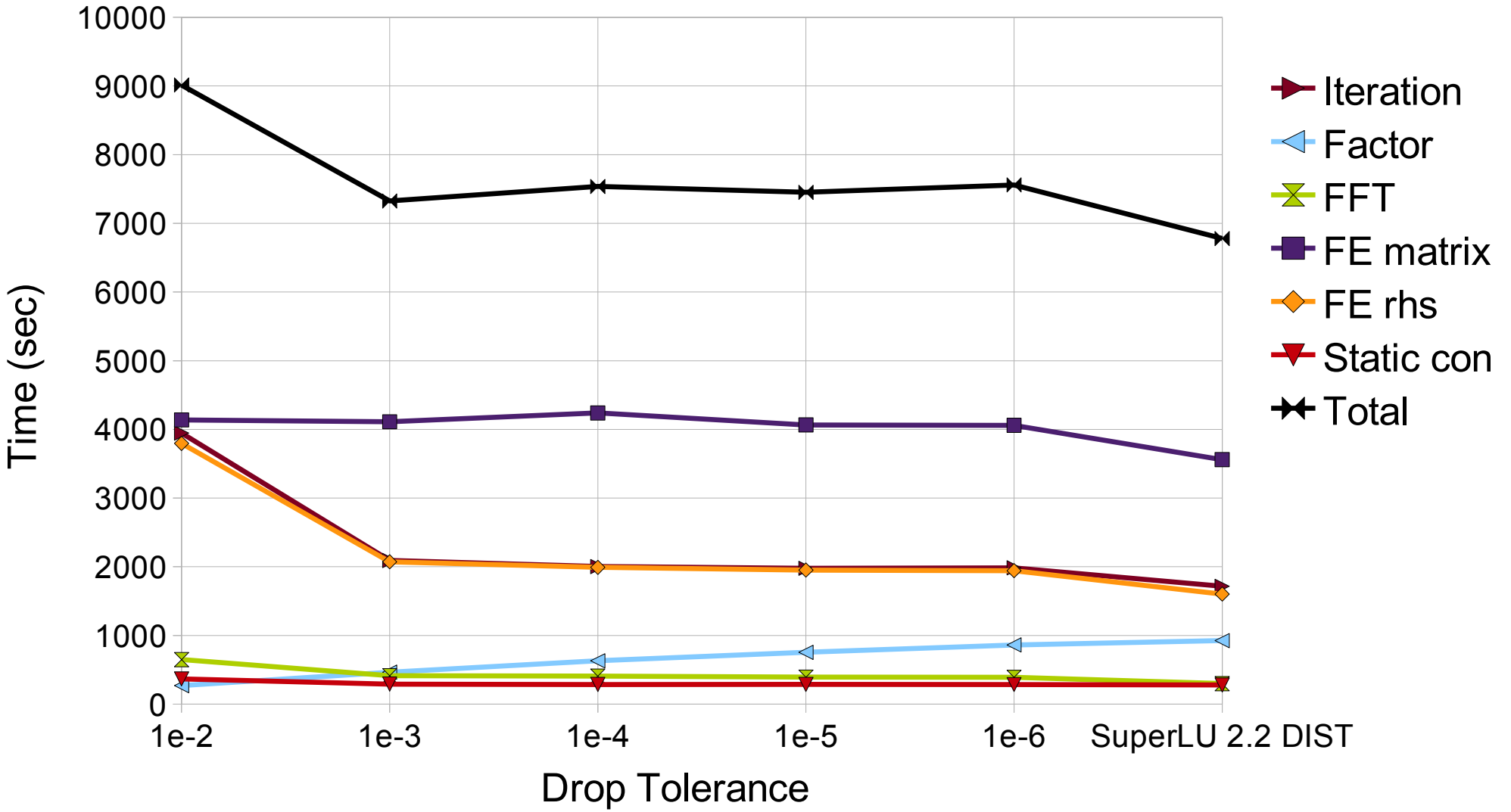
- If divbd is too large, it creates the dominant terms in the magnetic field advance.
- Since the ILU decomposition drops terms in a row of the matrix of a relative value τ smaller than maximum element of that row, important parts of the magnetic field advection and diffusion may be dropped from the preconditioner.

Test Case 2: Nonlinear visco-resisitive RFP, ILU τ scan.

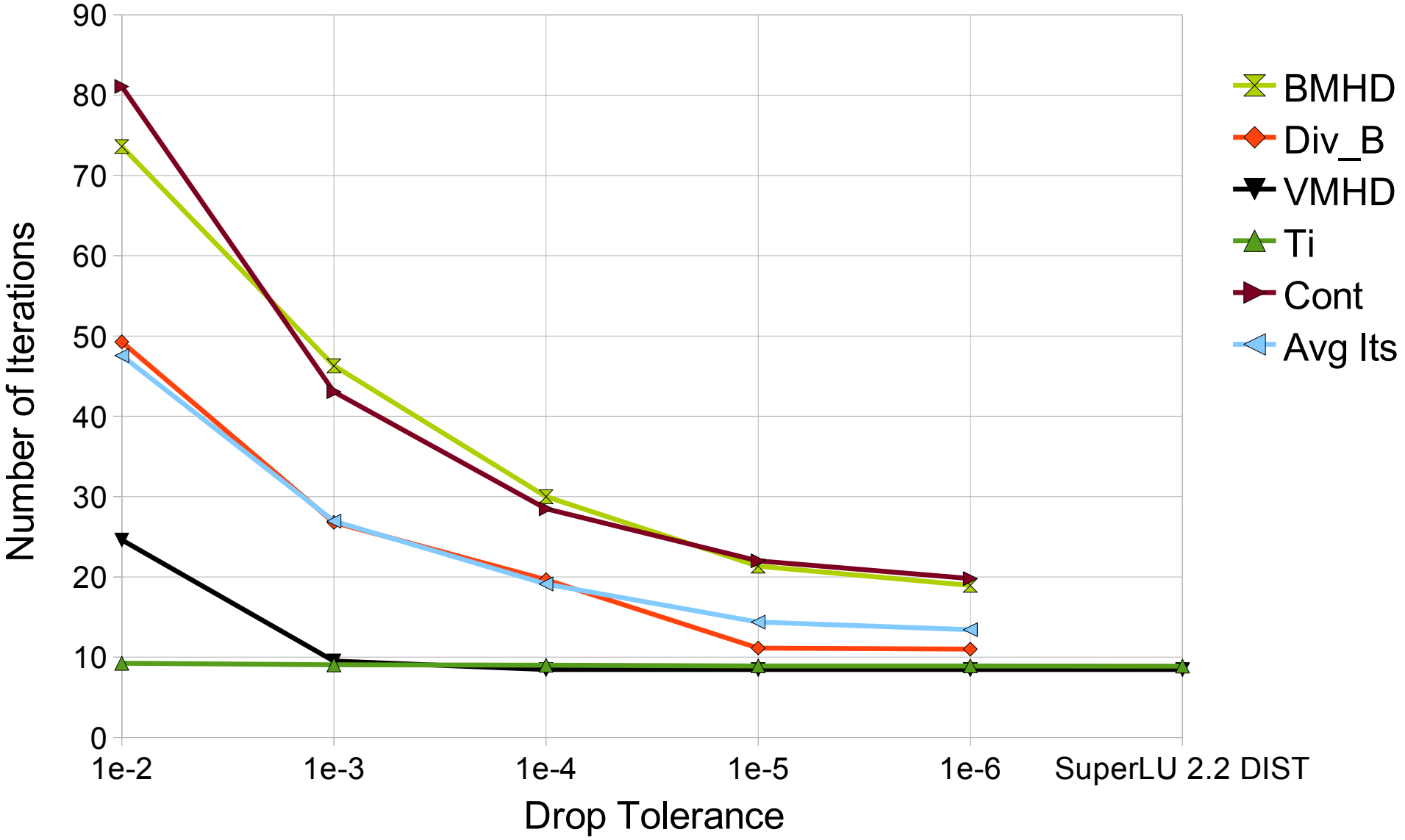
- S=5000
- P_m=1
- advect='V only'
- Ohms='mhd'
- nonlinear=.true.
- cont='full'
- p_model='isotropic'
- Θ = 1.55
- F ~ -0.05
- 30x30 RZ mesh
- geom='tor'
- poly_degree=4
- lphi=4 (6 modes)
- gridshape='rect'
- periodicity='y-dir'

- Initialized to a reversed state without density or temperature evolution. (using SuperLU 2.2 DIST)
- Run with ILU decomposition for 50 time steps with density and temperature evolution.
- τ, the drop tolerance is scanned for 10^{-2} to 10^{-6}.
- Physics evolution is approximately the same in all cases.
- divbd=1e-3 (elecd=2e-4) and split_divb=.true. are used.
- All cases run on 2 cores of a 64-bit Intel Xeon 3.0GHz machine with 8Gb RAM.

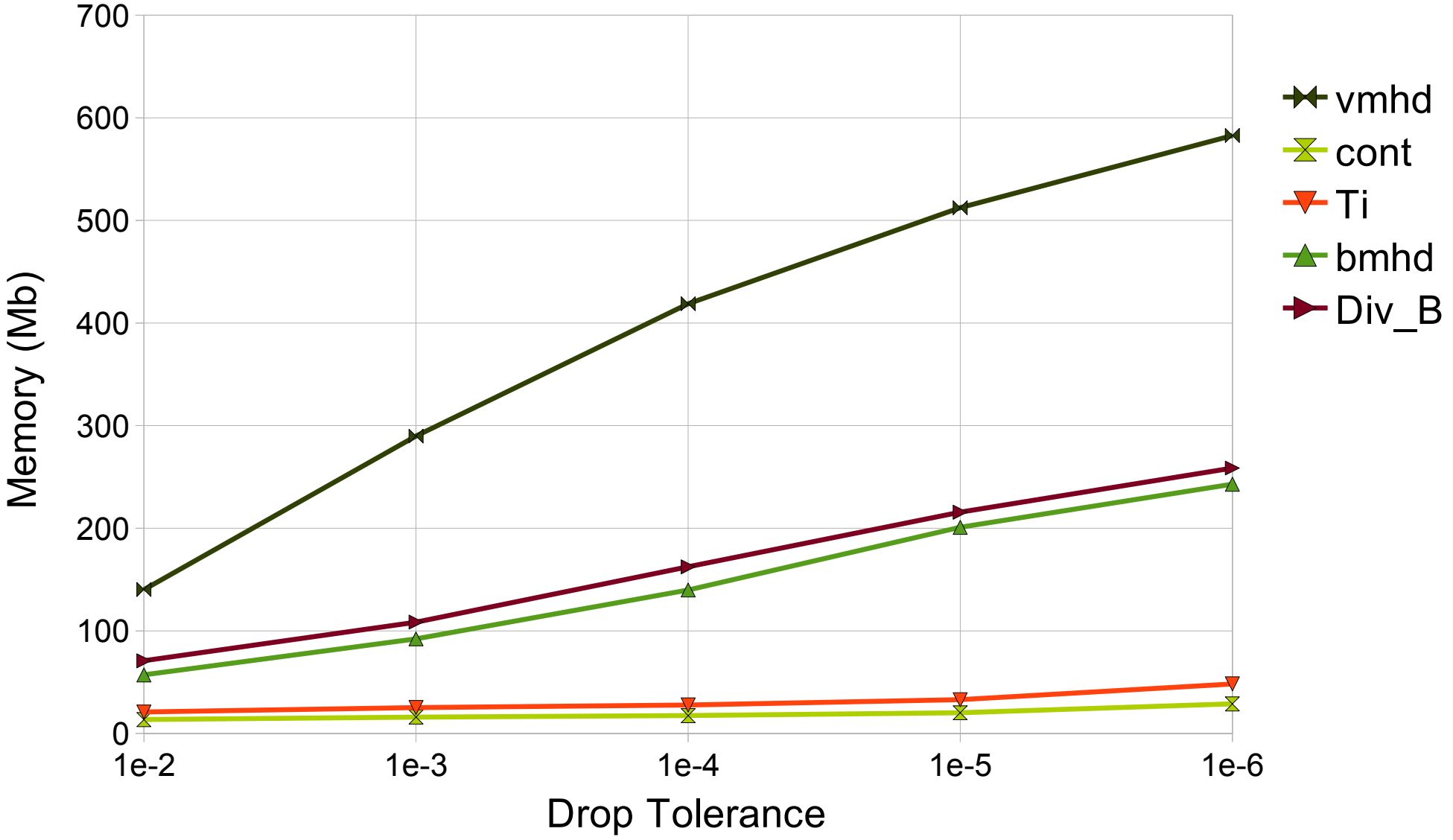
The total time is slightly greater with a large drop tolerance.



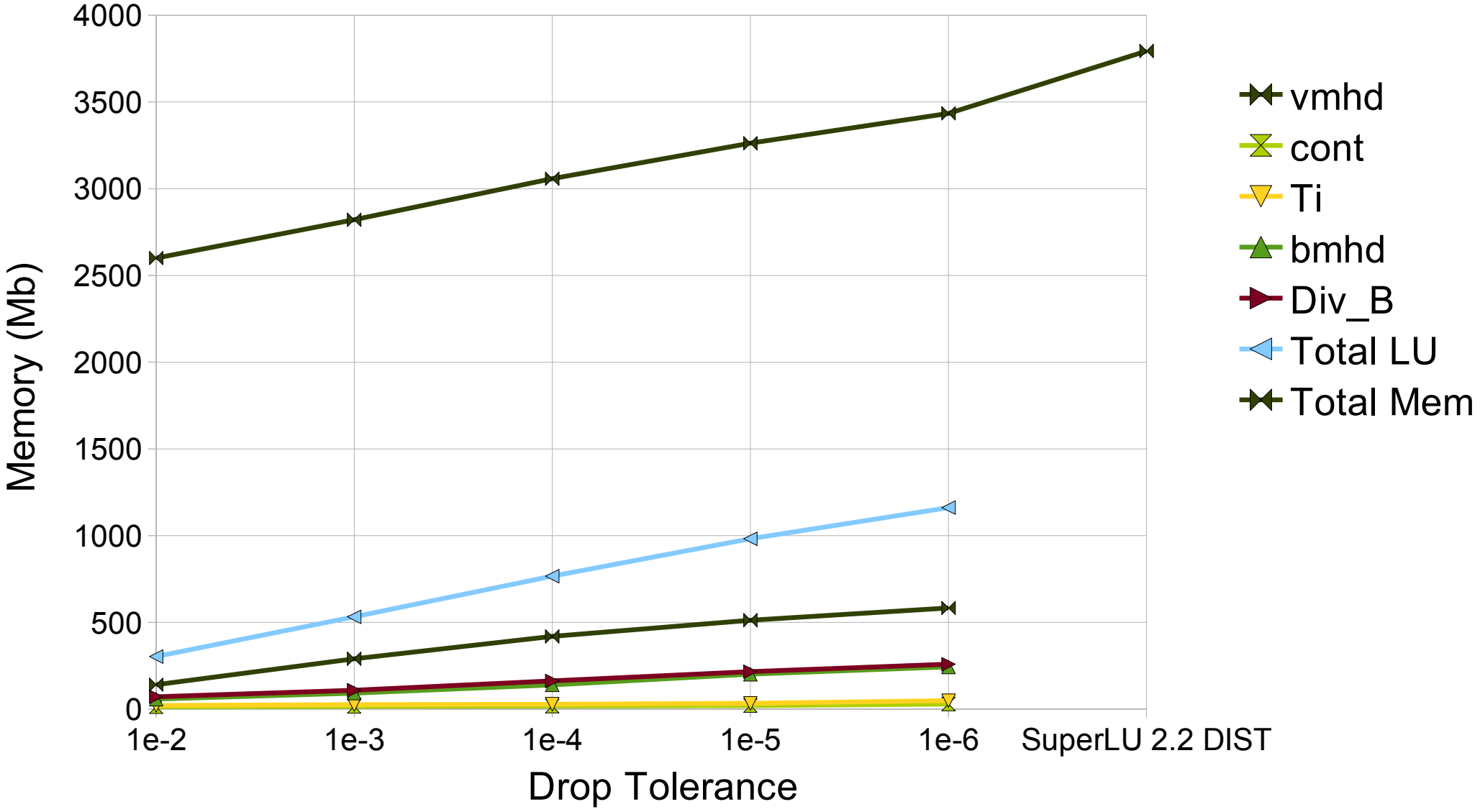
The number of iterations are increased by a factor of 4 at $\tau=10^{-2}$.



The memory used by the vmhd, bmhd and divb matricies can be greatly reduced.



The LU memory is reduced by 75%, and the total memory by 31% at $\tau=10^{-2}$.



Summary

- NIMROD memory usage may be reduced by up to 31% via the use of ILU factorization.
- ILU factorization may add additional complications to the magnetic field advance, if divbd is large. (as is usually the case)
- Further optimization may require different ILU input parameters for different matrix solves. (bmhd, vmhd, hall, cont, etc.)
- The total time used by NIMROD using either ILU or LU is comparable.
- Many of the ILU capabilities of SuperLU 4.0 are left unexplored.