Accuracy and Efficiency of Photoionisation Algorithms

Jonathan Mackey
Argelander Institute for Astronomy, Bonn, Germany

Paper: Mackey (2012), A&A, 539, A147. arXiv:1201.5651

- Why Photoionisation is difficult
- Description of Algorithms
- Accuracy Comparison
- Accuracy vs. Runtime
- Parallel Scaling
- Conclusions
- Photoionisation movie

Argelander-Institut für Astronomie

Unterstützt von / Supported by
Alexander von Humboldt Stiftung/Foundation

DFG Priority Program 1573
ISM-SPP
Interstellar Medium

Photoionisation: Numerical Difficulties

- Velocity of ionisation fronts (IFs) limited only by c.
- Equations can be stiff - Photoionisation rate, A(PI), can be orders of magnitude larger than recombination rate.
- Internal energy, E, and ion fraction, y, change by orders of magnitude as grid-cell is ionised, and so does optical depth.
- Whalen & Norman (2006) limit dt by $t_{\rm chem} = 0.1 \frac{n_e}{\dot{n}_e}$. Need 24 raytracings to change y by 10x.
- Column densities N(H) along rays cannot be fully parallelised unless rays are parallel to grid axes.

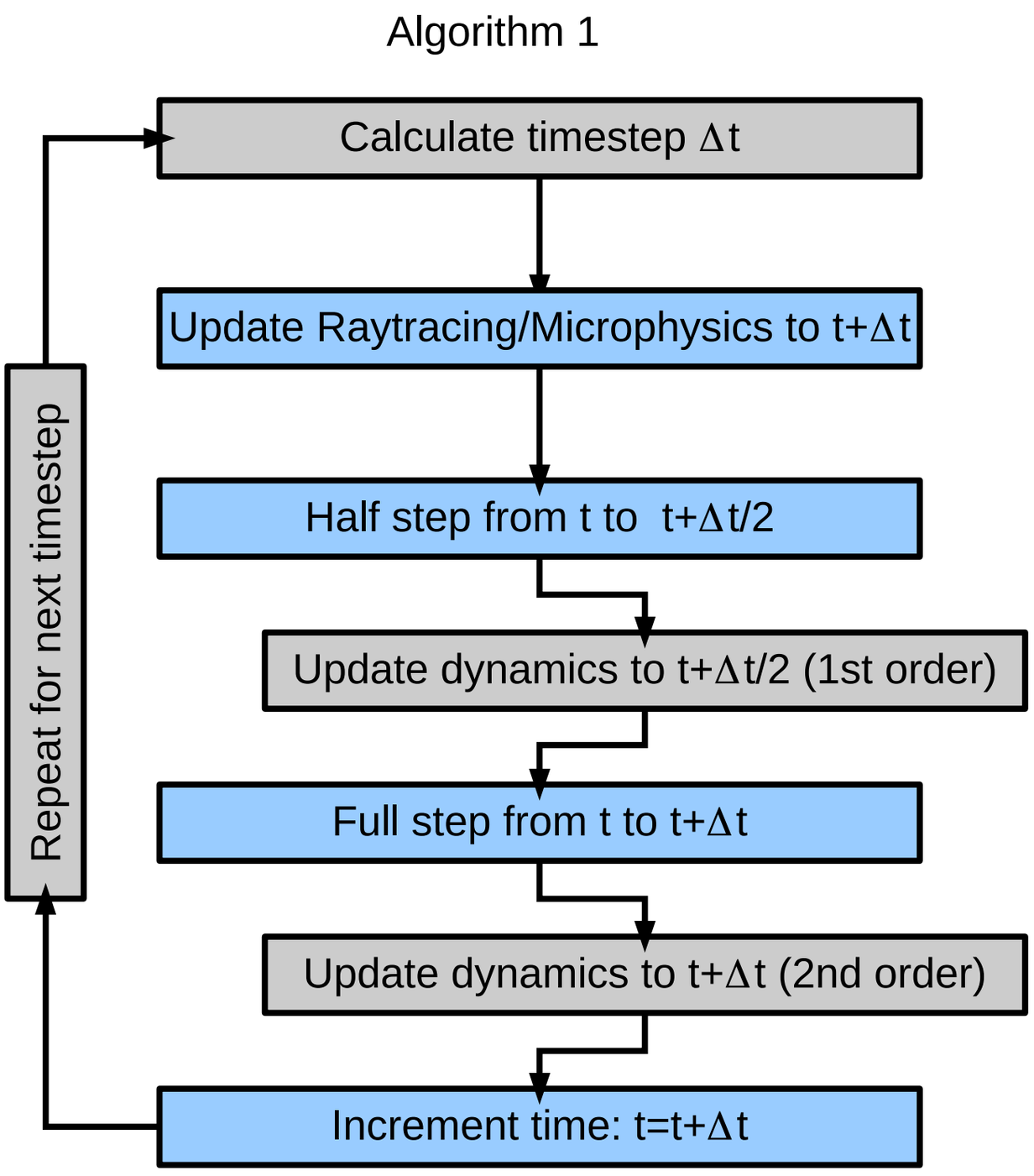
$$\int_{\nu_{\rm th}}^{\infty} \frac{L_\nu {\rm e}^{-\tau_\nu}}{h\nu} \frac{1-{\rm e}^{-\Delta\tau_\nu}}{n_{\rm HI} V_{\rm shell}} {\rm d}\nu.$$

Photon-conserving PI rate (Mellema+06)

$$\dot{y} = A_{\rm pi}(\rho, y, N_{\rm H0})[1-y] + A_{\rm ci}(T) n_{\rm H} y [1-y] - \alpha^{\rm B}_{\rm rr}(T) n_{\rm H} y^2$$

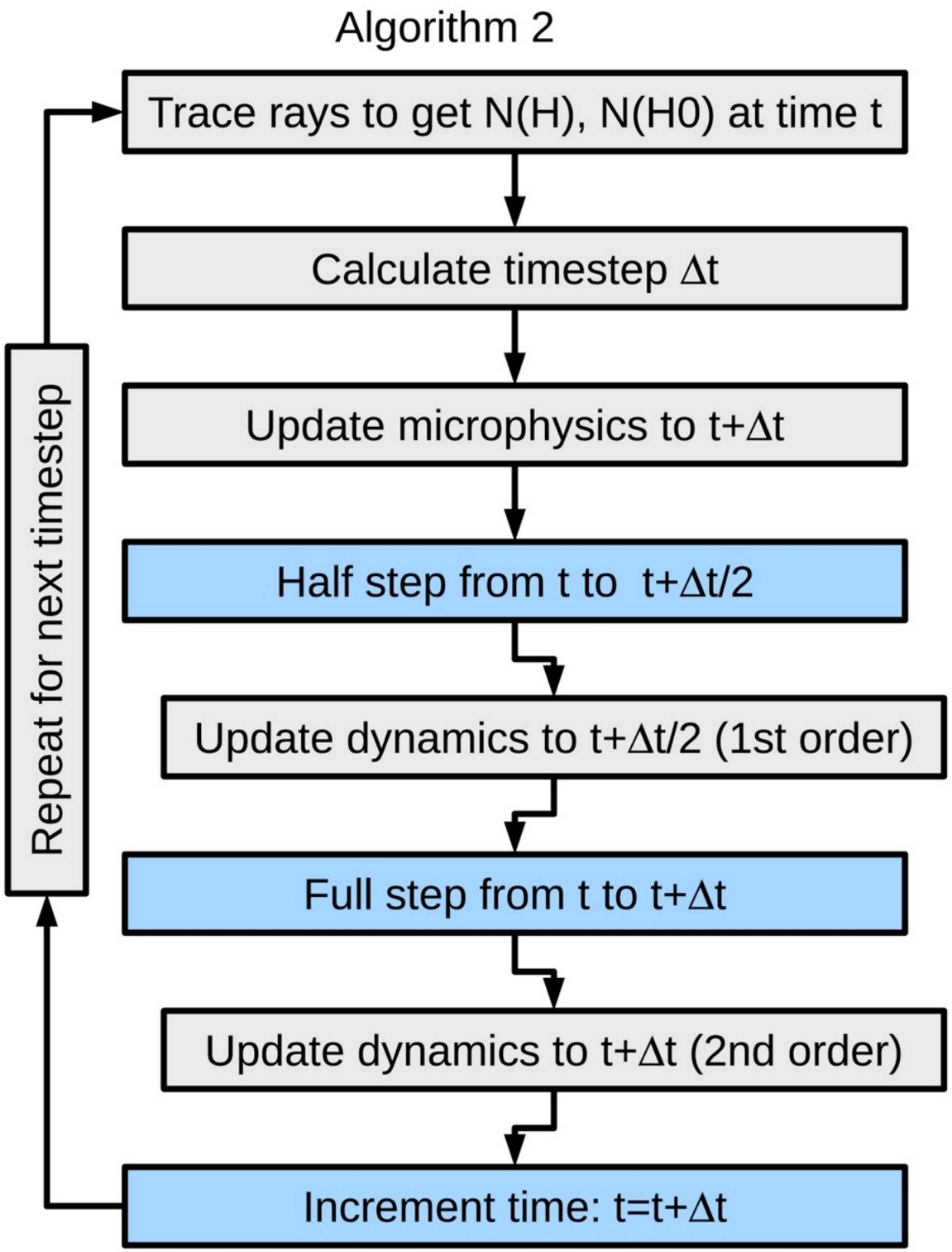
$$\dot{E}_{\rm int} = \Gamma(\rho, y, N_{\rm H}, N_{\rm H0}) - \Lambda(\rho, y, T). \qquad (2)$$

Implicit Algorithm (A1)



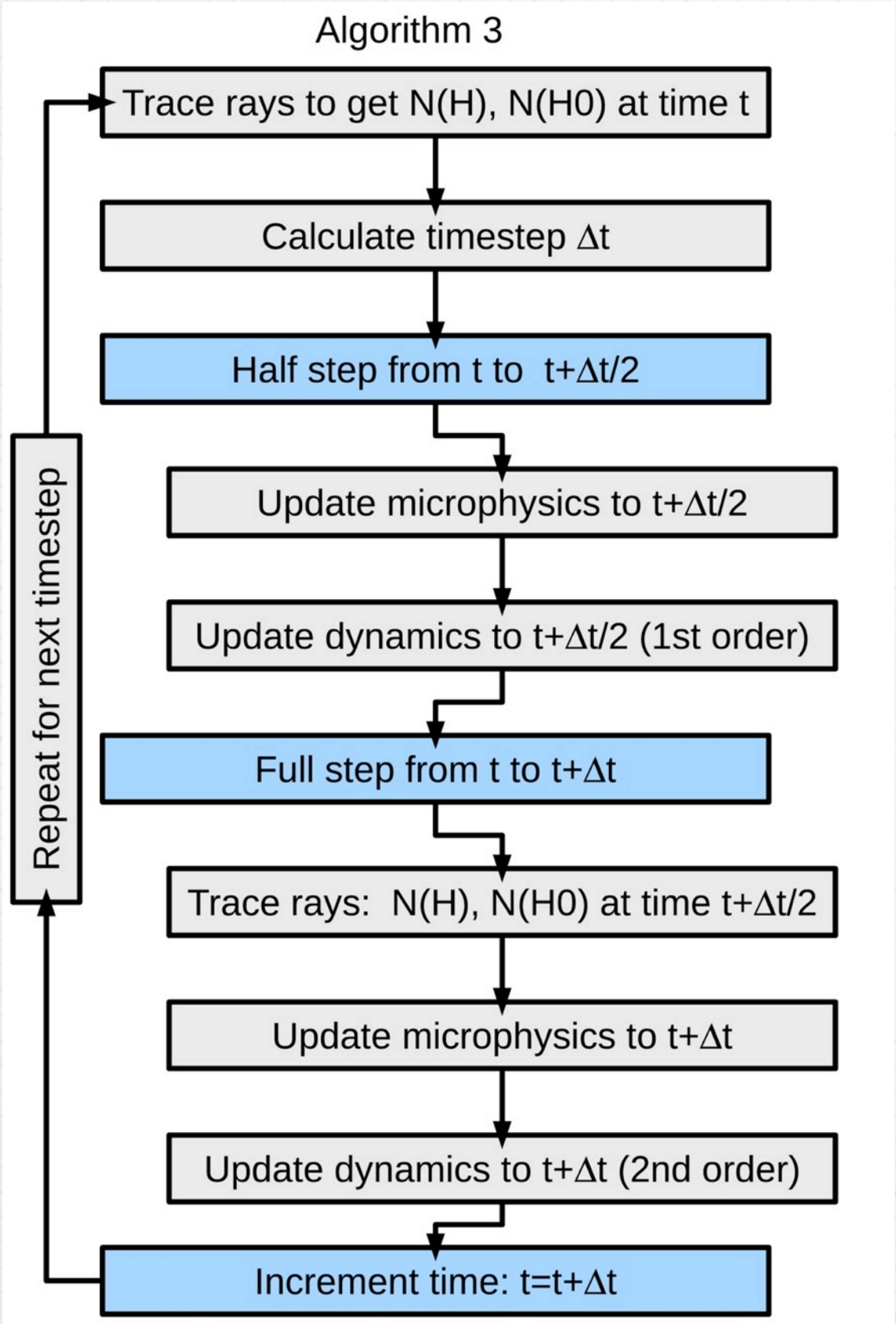
* Introduced by Mellema et al. (2006) - C2-ray algorithm.
* Similar scheme used by Mackey & Lim (2010,2011).
* Microphysics updated as rays are traced.
* Allowed time-averaged column densities from source to cell.
* I-front can cross many cells per step, maintaining accurate velocity.
* see also Friedrich+(2012) for newer+better scheme.

First Order Explicit Algorithm (A2)



* Frank & Mellema (1994); Whalen & Norman (2006)
* N(H) used to calculate timestep and integrate microphysics.
* Implicit integrator used for ion frac., internal energy.
* 1st order Euler integration nevertheless, in terms of column densities and photon conservation.
* 24 raytracings needed per 1dex increase in electron/ion fraction.

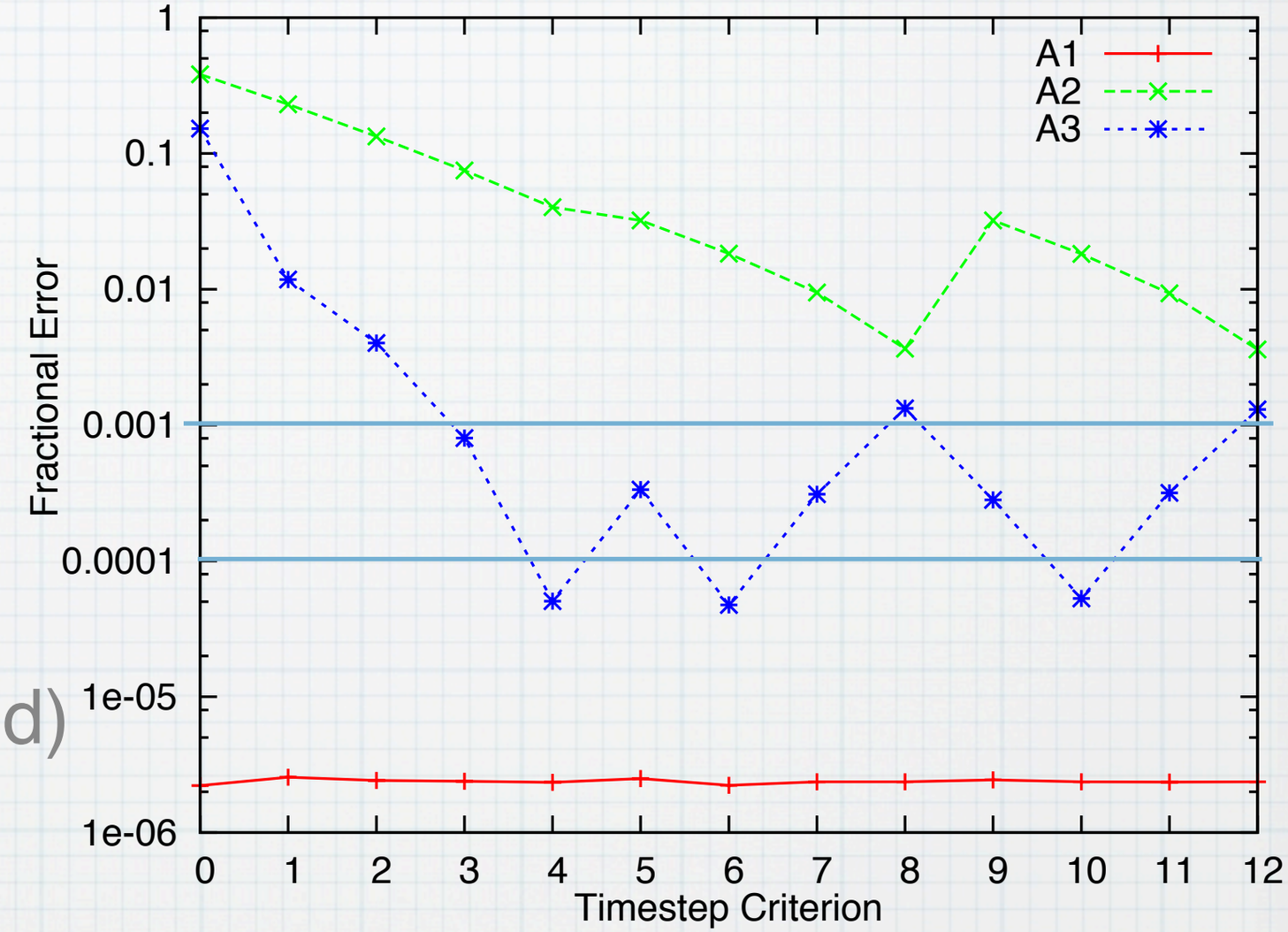
Second Order Explicit Algorithm



* 2 raytracings per step.
* Time-centred column densities mean photon conservation is 2nd order.
* Still explicit scheme.
* Fits in well with 2nd order dynamics update.
* Allows full ionisation of cell in 4 timesteps (8 raytracings).
* Still needs 4 steps for I-front to cross cell.

Planar, constant velocity, I-fronts

* Monochromatic radiation
* No recombinations
* I-front has constant velocity v=F/n(H0)
* 13 timestep criteria:
  0-4: dt=K.(1/ydot)
  5-8: dt=K.(y/ydot)
  9-12: dt=K.min(y/yd,E/Ed)
* Implicit A1 v. good by construction.
* A3 converges much faster than A2, error <1% very quickly.



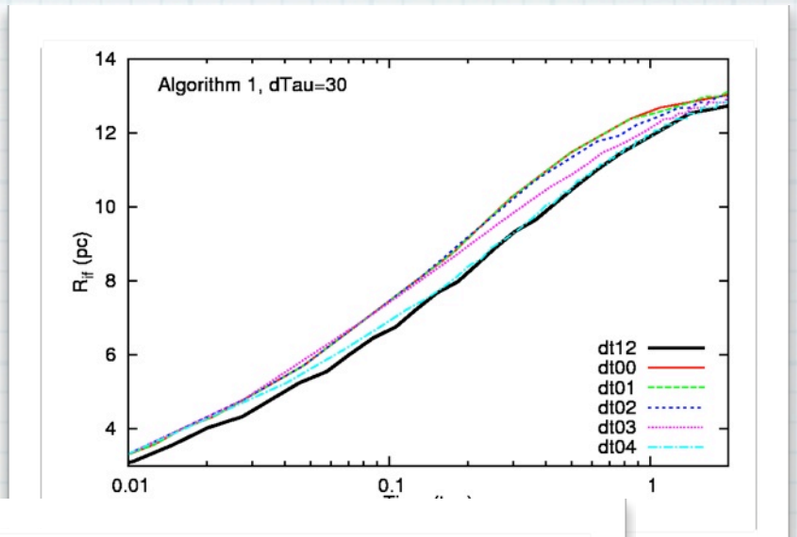
--> More restrictive dt -->

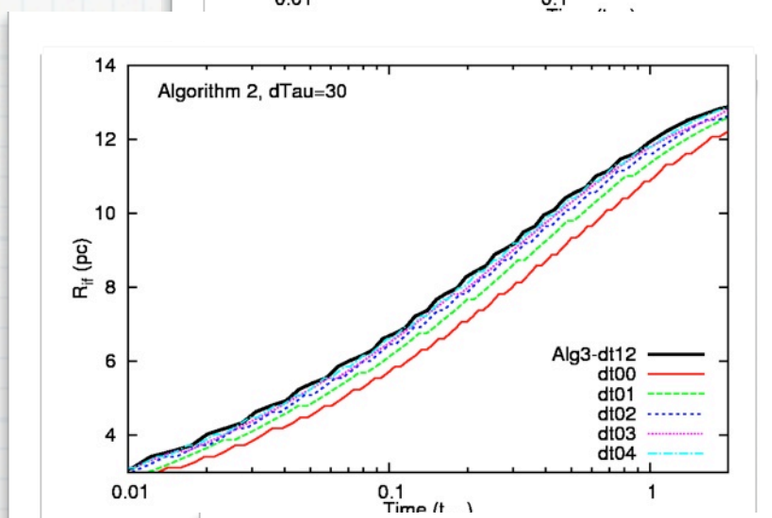
1D Stromgren Sphere Calculation

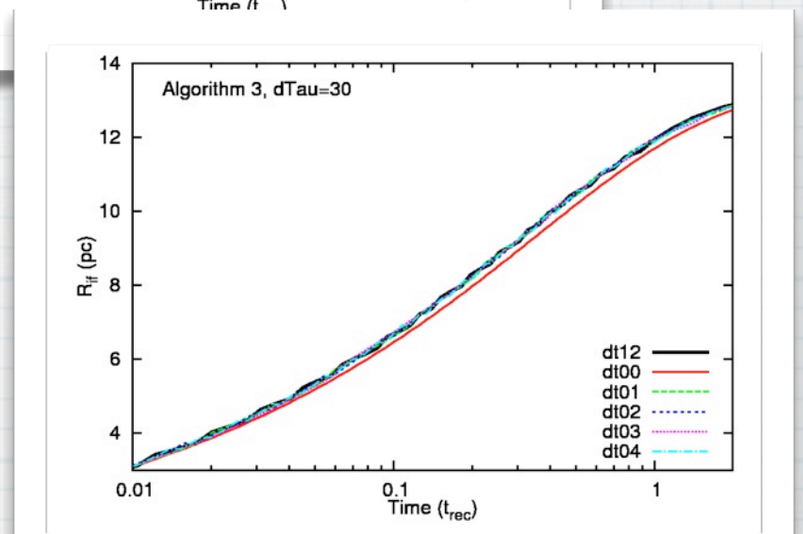
$$r_{\rm I} = r_{\rm S}[1-\exp(-t/t_{\rm rec})]^{1/3}.$$

$$v_{\rm I} = \frac{r_{\rm S}}{3t_{\rm rec}} \frac{\exp(-t/t_{\rm rec})}{[1-\exp(-t/t_{\rm rec})]^{2/3}}$$

* Multi-frequency radiation
* No dynamics, just IF expansion.
* Cell Tau=30 (at 13.6eV):
* Implicit A1 better for optically thin cells, A2/A3 more accurate for tau>>1.
* Implicit A1 propagates I-front too rapidly.
* A3 essentially converged for all timestep criteria.
* A2 has errors comparable to A1.

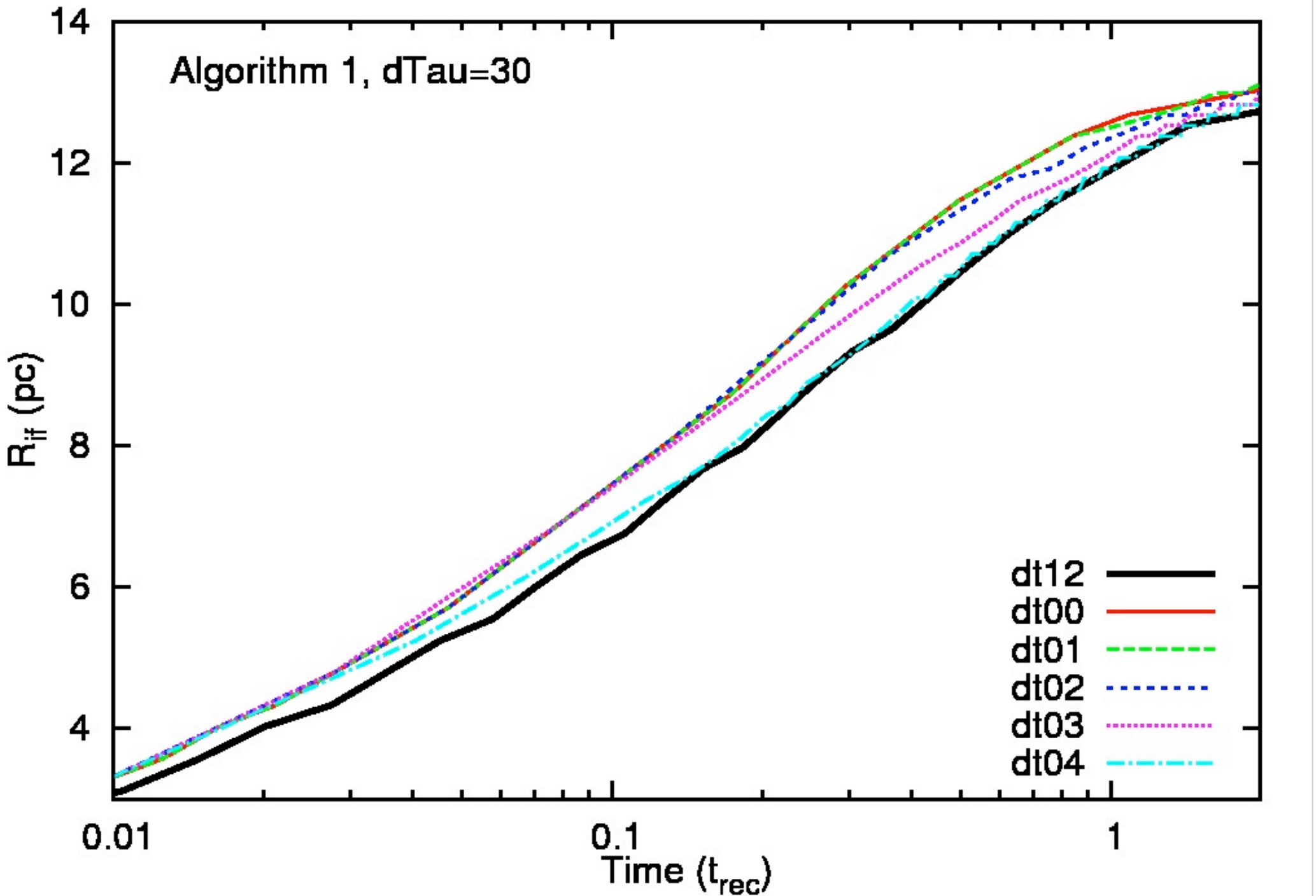






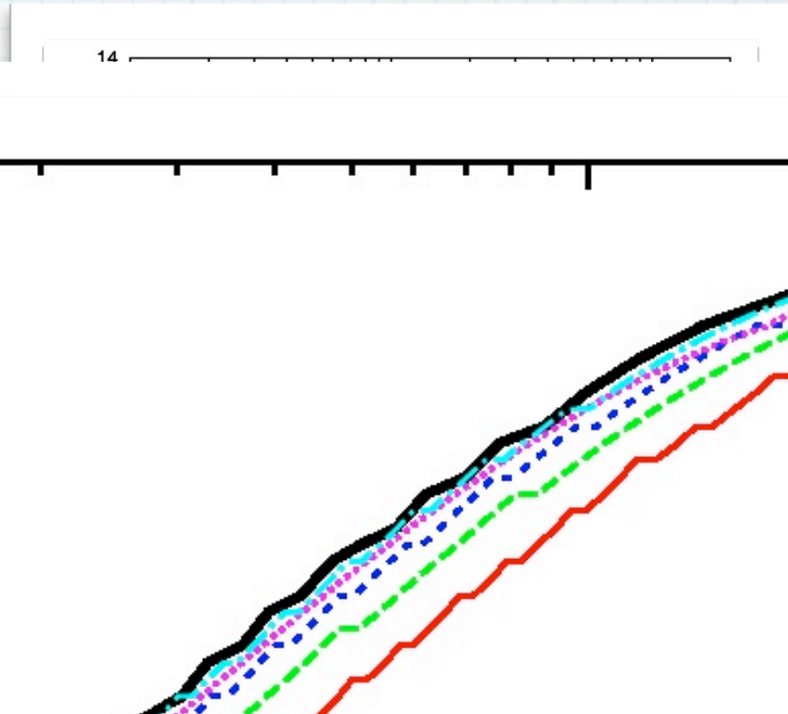
1D Stromgren Sphere Calculation

$$\left[1 - \exp(-t/t_{rec})\right]^{1/3}$$



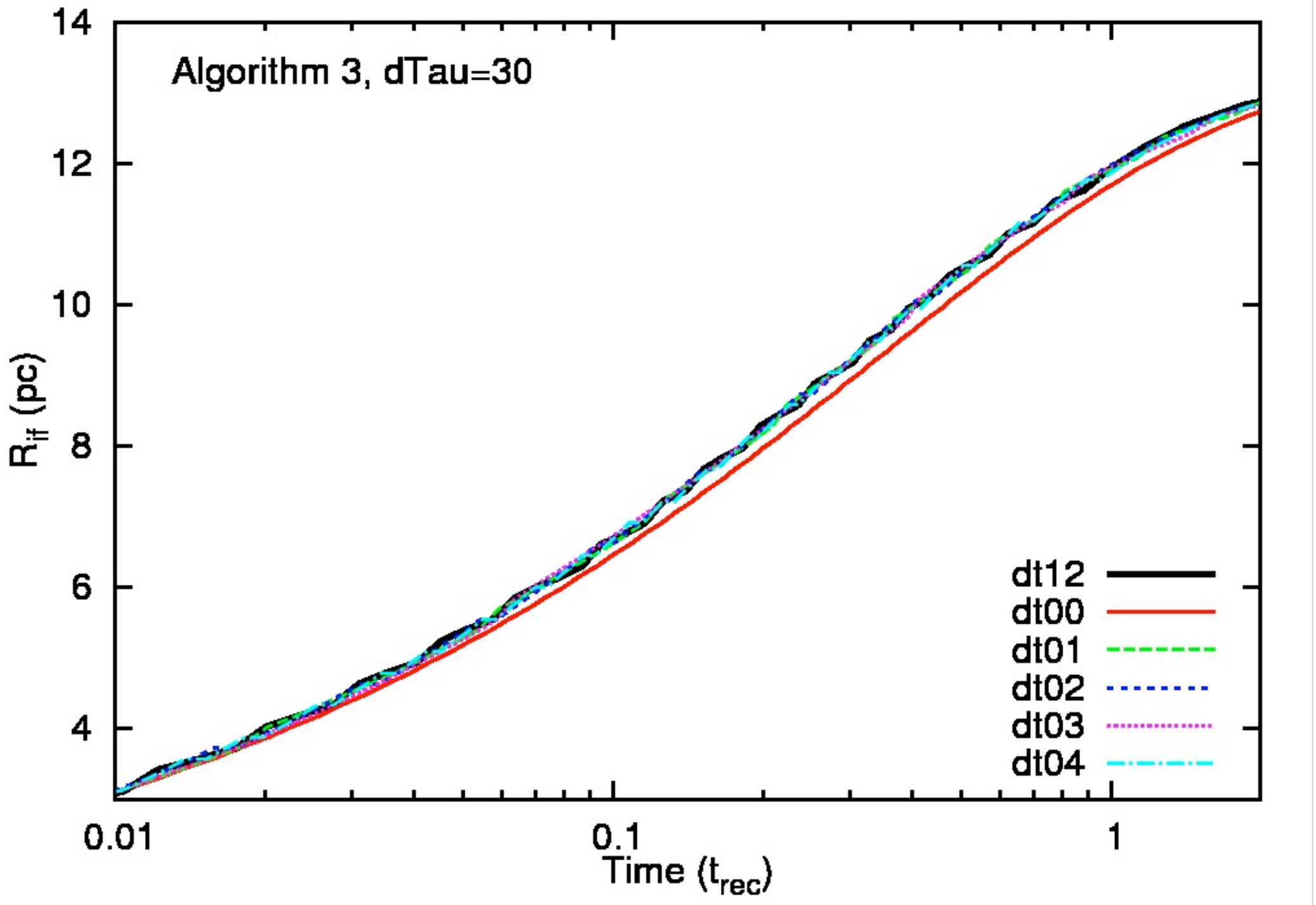
1D Stromgren Sphere Calculation

$$\left[1 \qquad \left(\quad t/t_{rec}\right)\right]^{1/3}$$



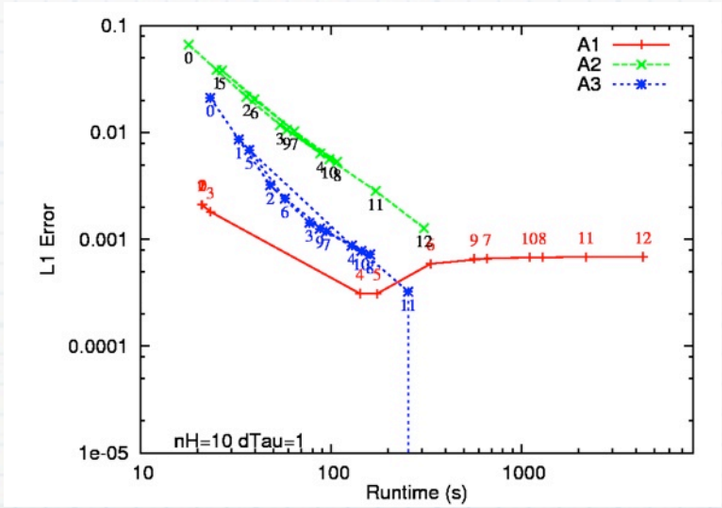
1D Stromgren Sphere Calculation

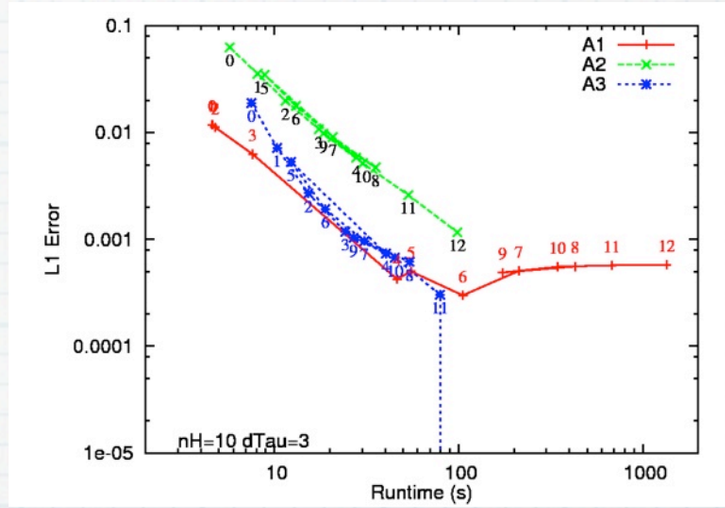
$$\left[1 - \exp(-t/t_{rec})\right]^{1/3}$$



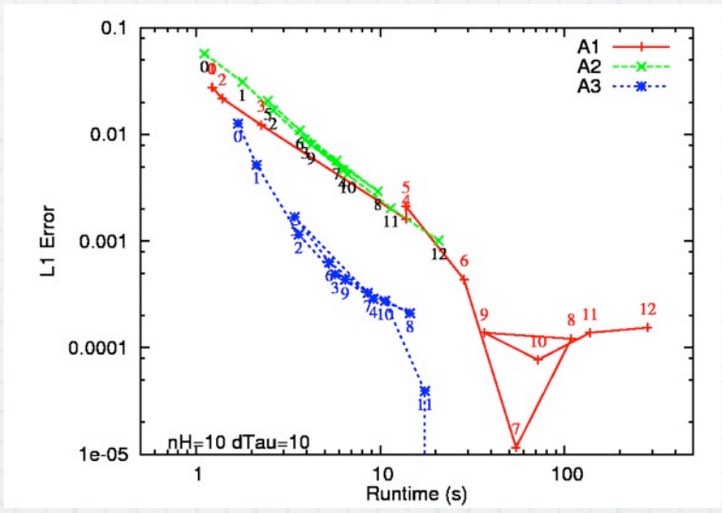
Accuracy vs. Runtime

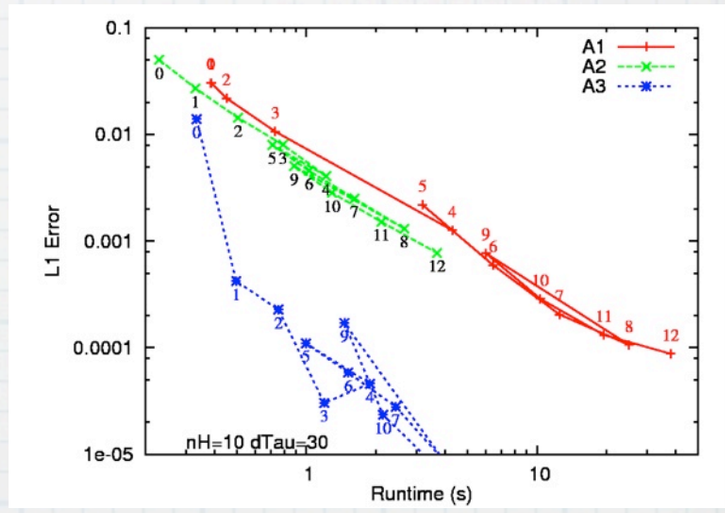
* Multi-frequency radiation, no dynamics, 1D expansion of Stromgren sphere.





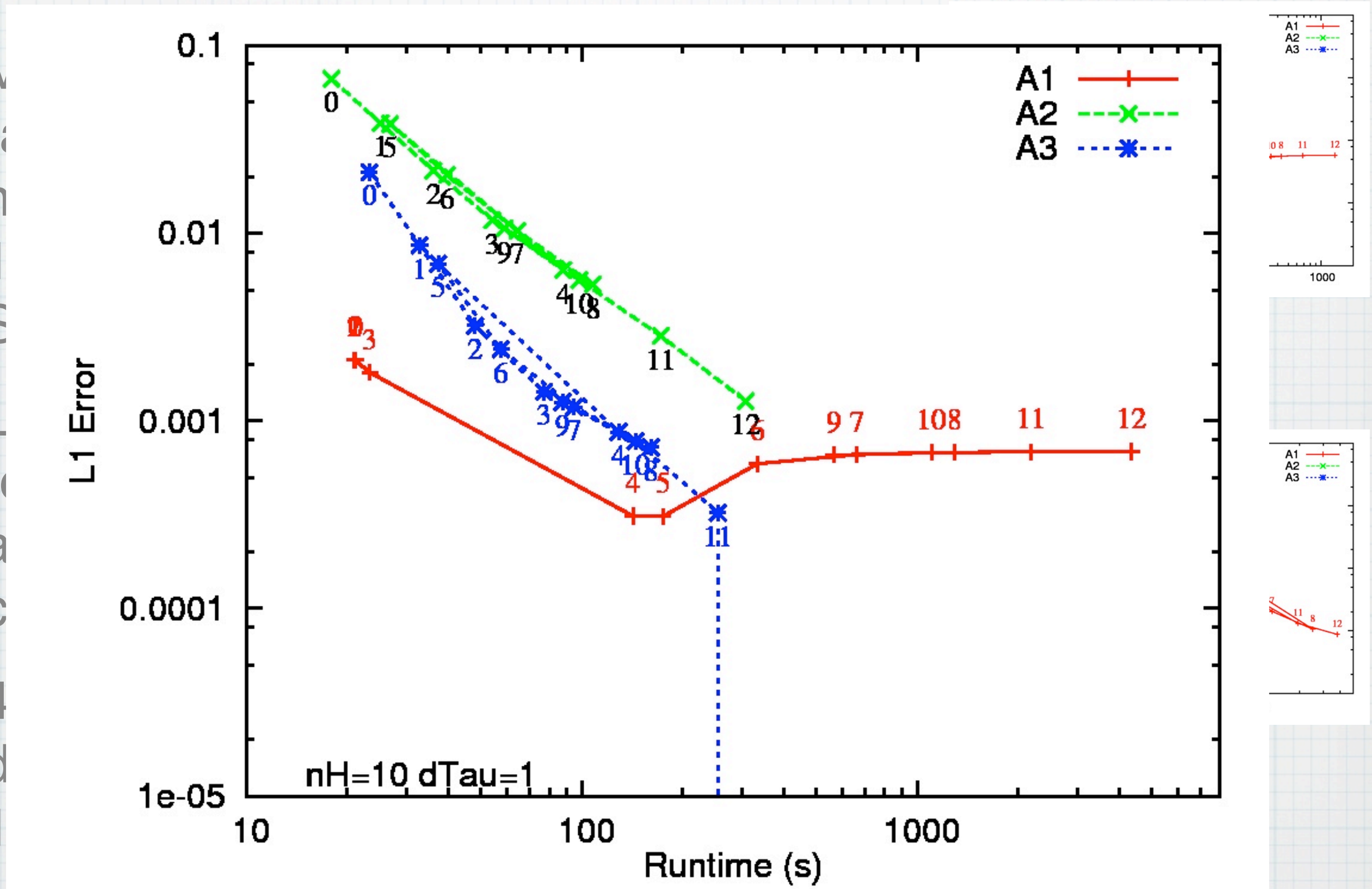
* L1 error after one recombination time, as function of calculation time.
* 4 different cell optical depths: dTau=1, 3, 10, and 30.





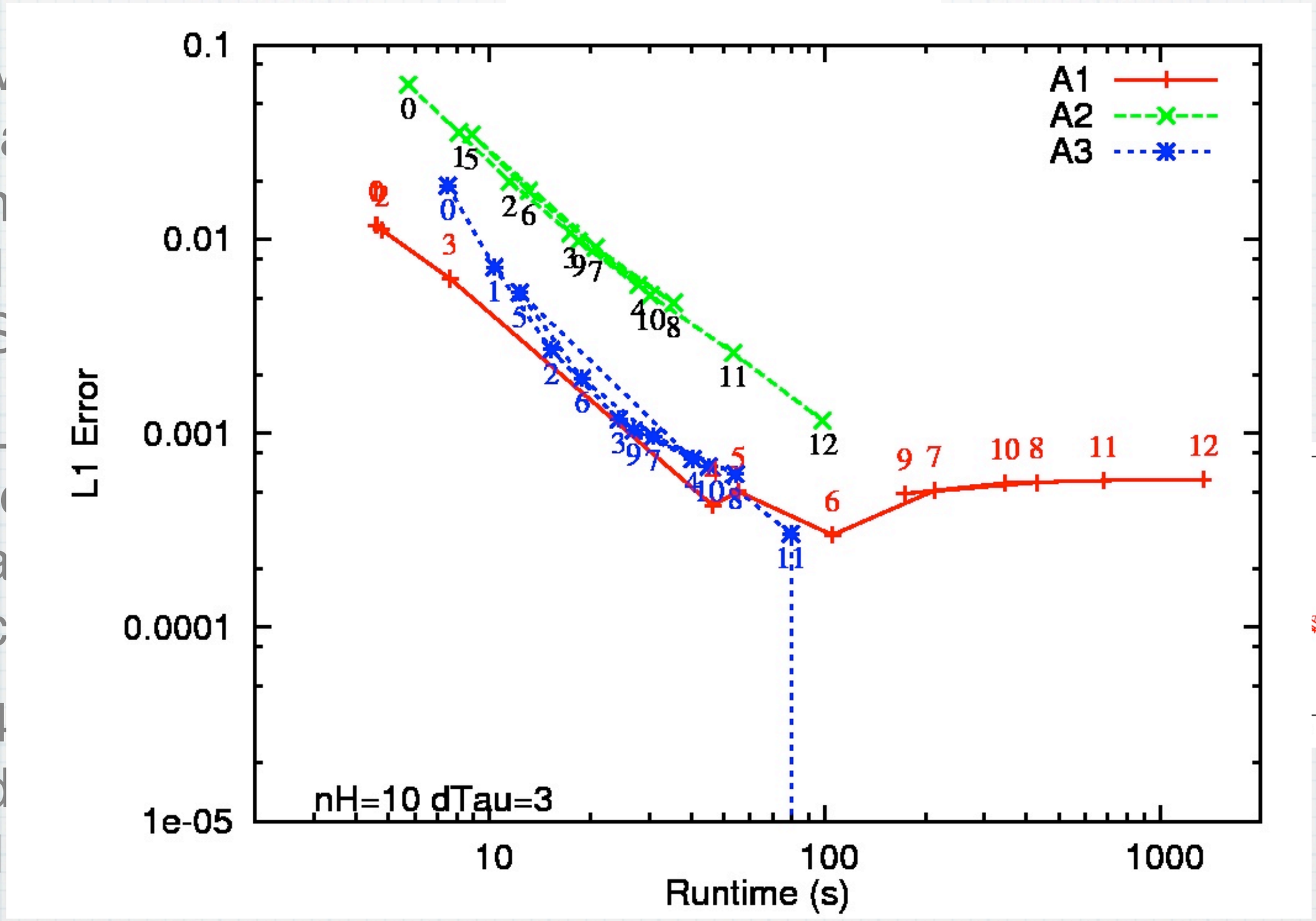
Accuracy vs. Runtime

* M
  r
  n
  1
  S
* L
  r
  a
  c
* 4
  d
  1



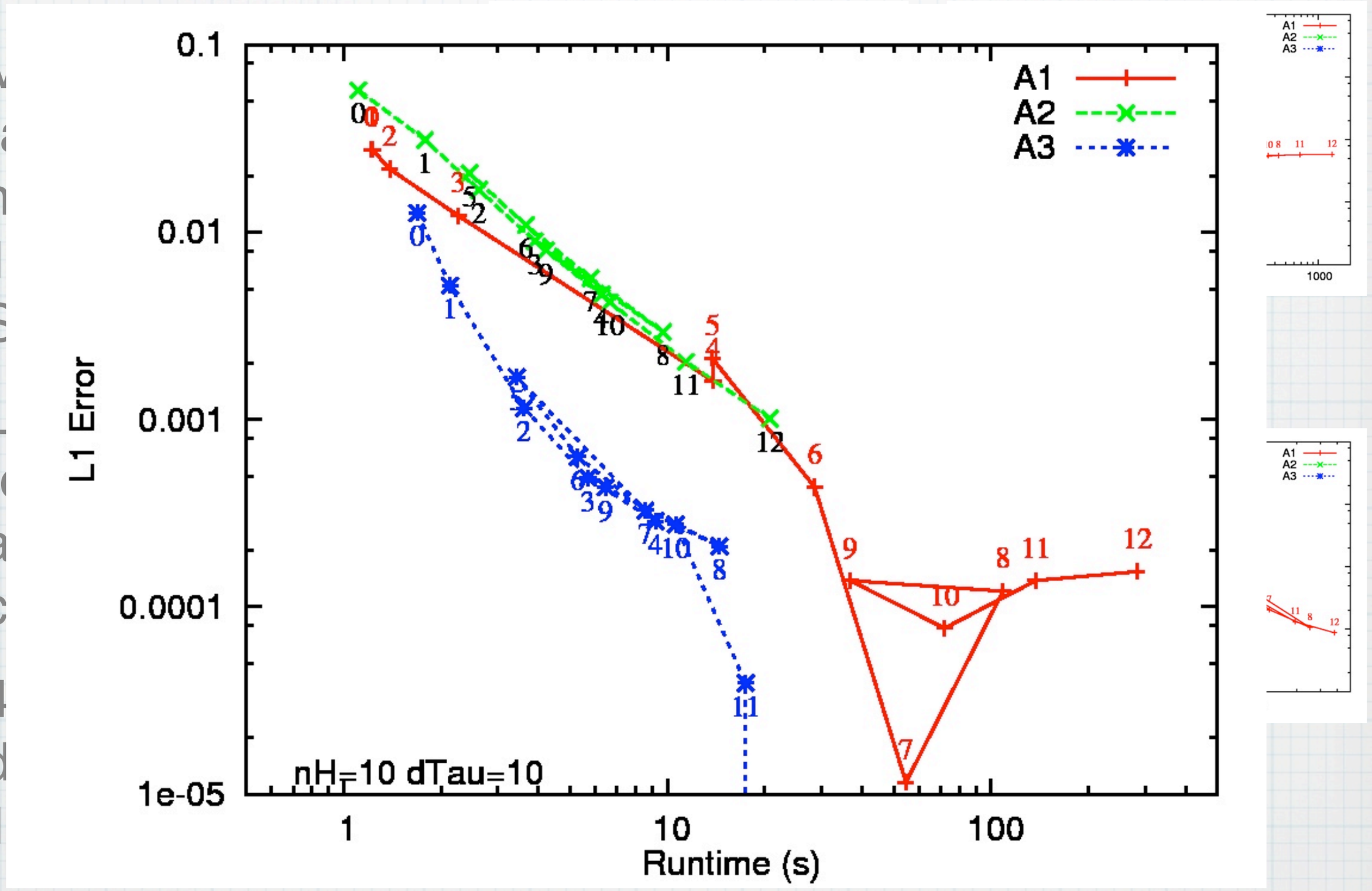
Accuracy vs. Runtime

* M
  r
  n
  1
  S
* L
  r
  a
  c
* 4
  d
  1



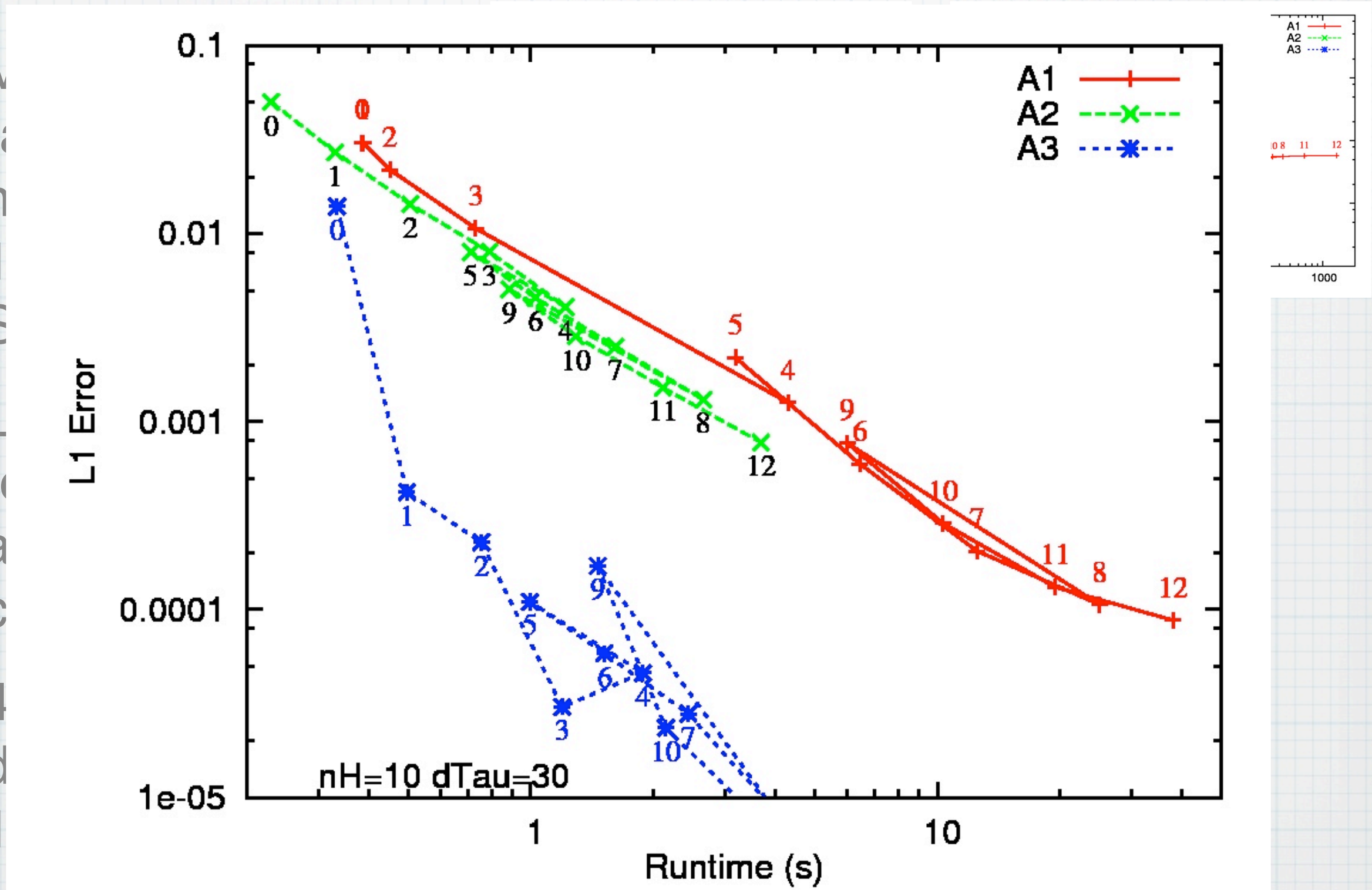
Accuracy vs. Runtime

* M
  r
  n
  1
  S
* L
  r
  a
  c
* 4
  d
  1



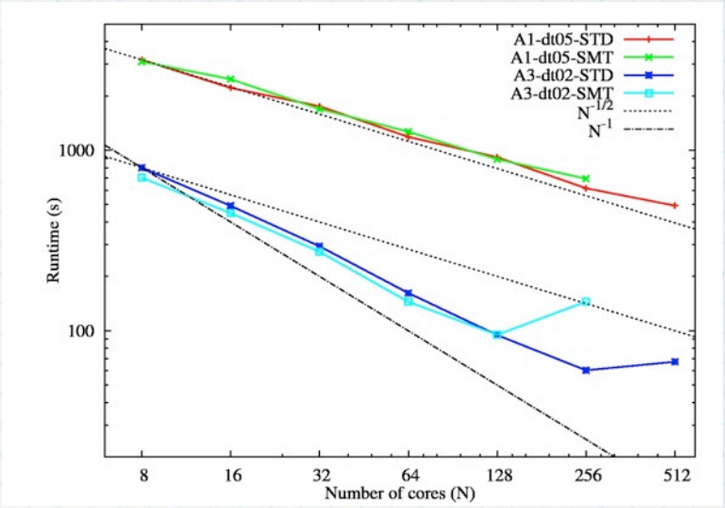
Accuracy vs. Runtime

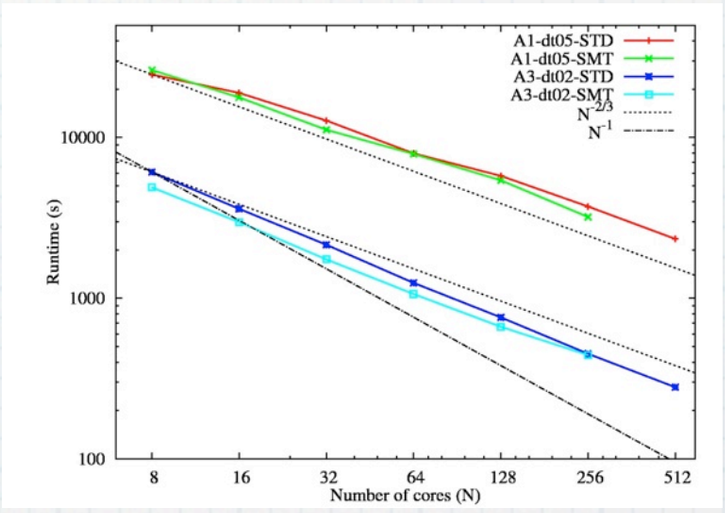
* M
  r
  n
  1
  S
* L
  r
  a
  c
* 4
  d
  1

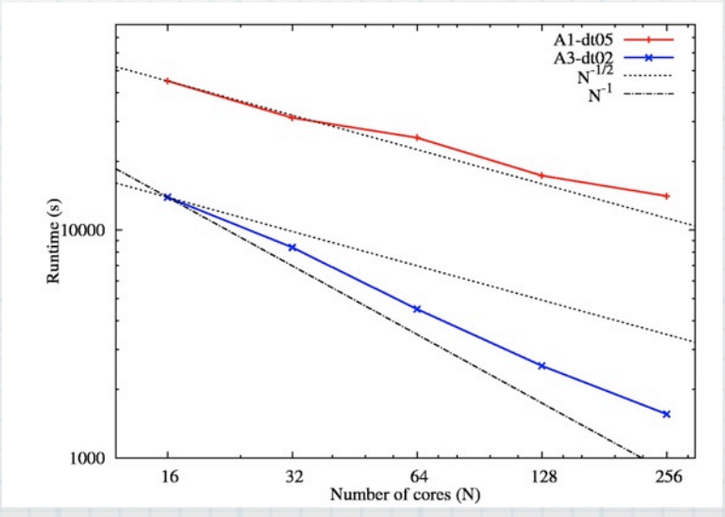


Parallel Scaling A1 vs. A3

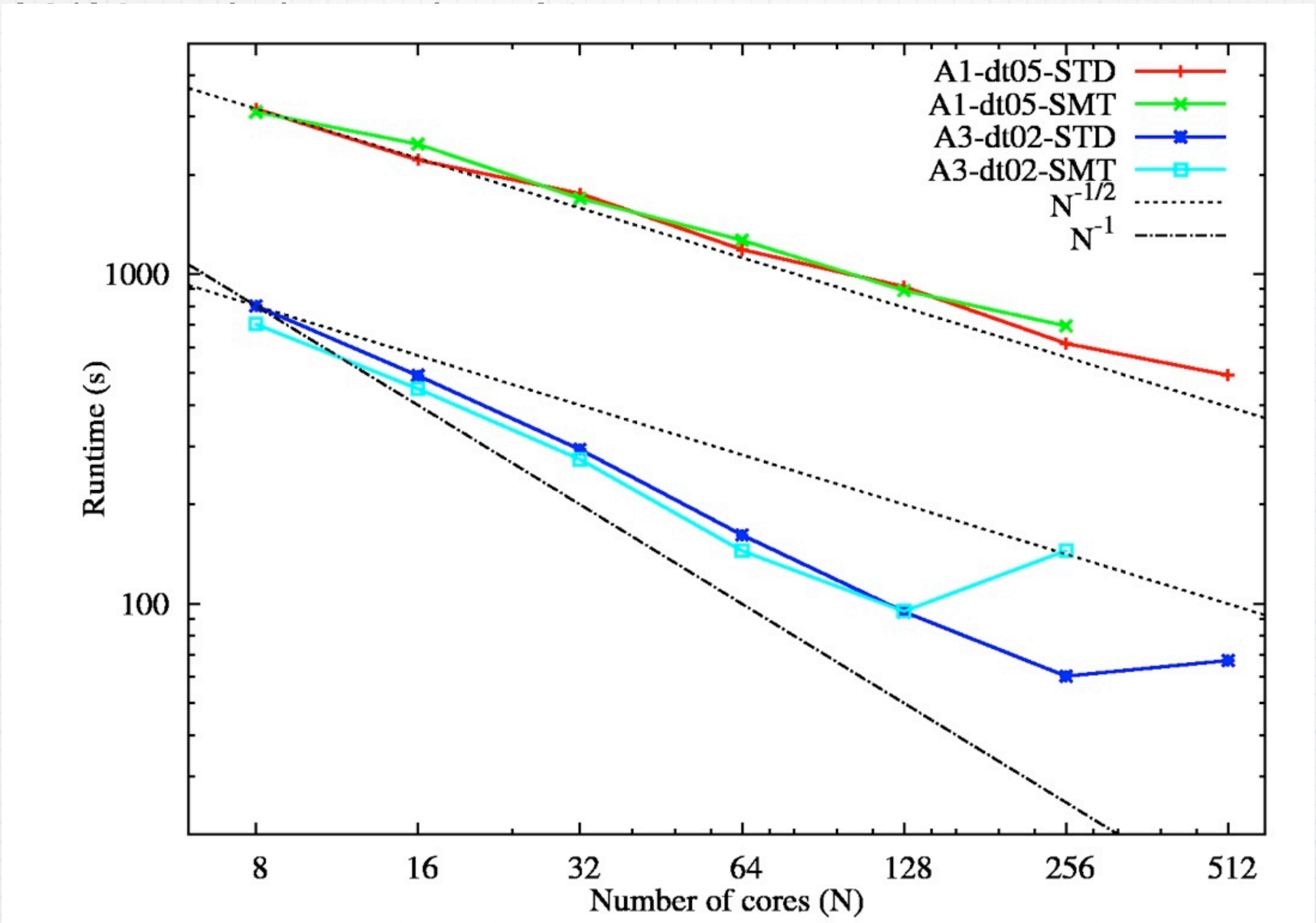
- A2/A3 scale better than A1 because microphysics integration is not in the raytracing step.
- Scaling limited by causal raytracing.
- Runtime plotted vs. number of cores, N, using JUROPA at Juelich.
- Tests w/ SMT have 2 MPI processes per core.
- Ideal scaling t=c/N (c a constant)
- 2D RT has t=c.N^(-1/2)
3D RT has t=c.N^(-2/3)



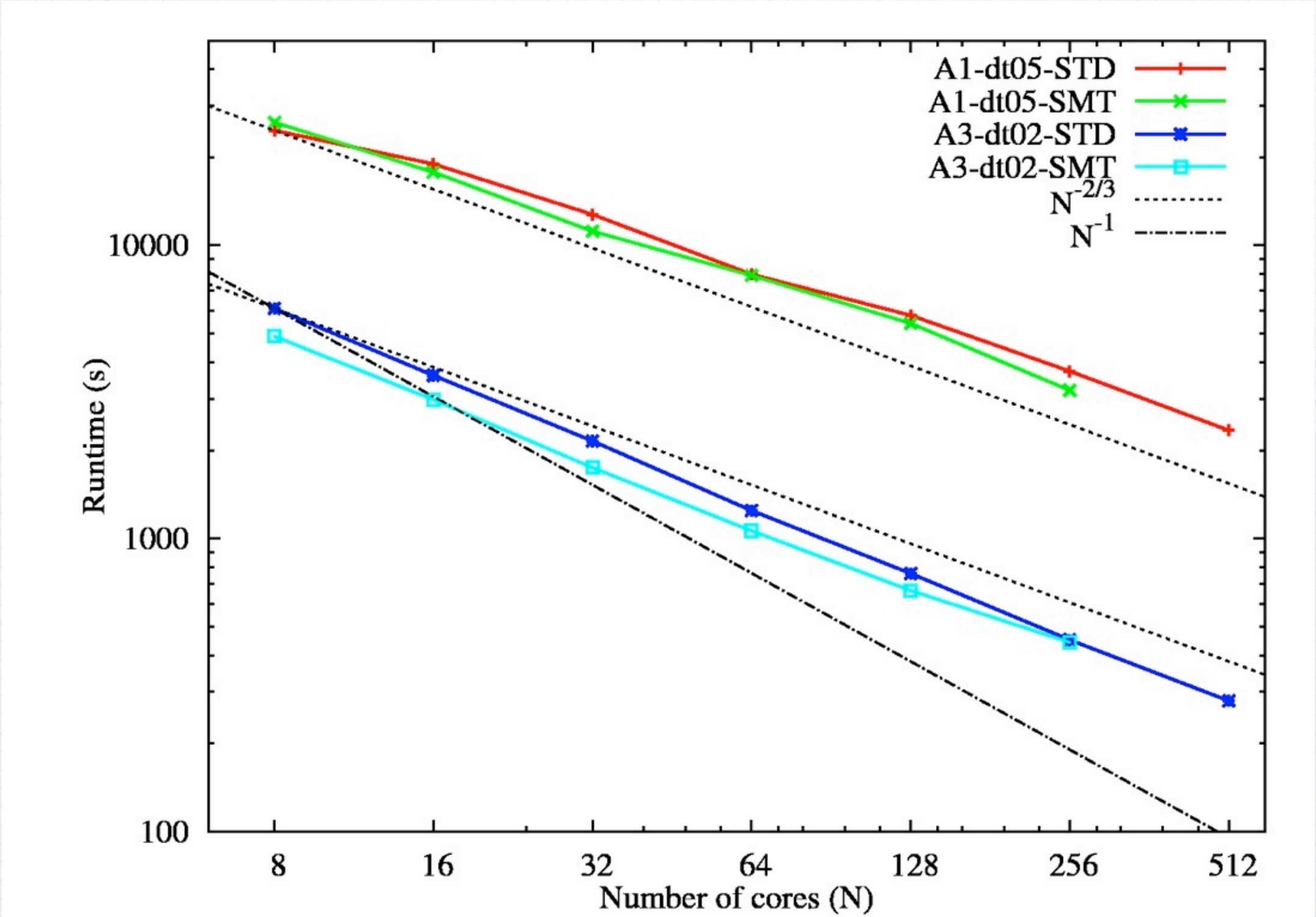




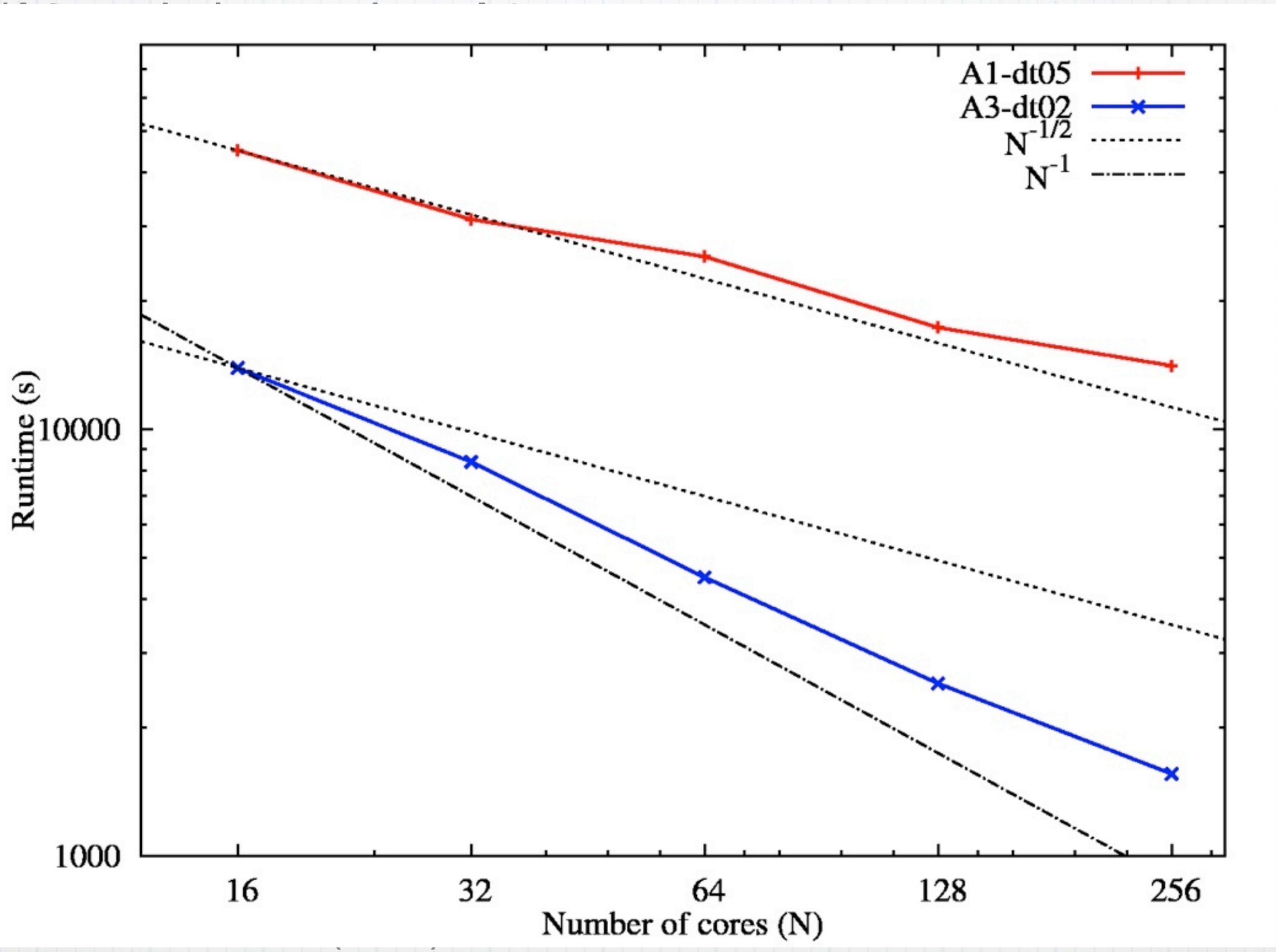
Parallel Scaling - 2D Static



Parallel Scaling - 3D Static

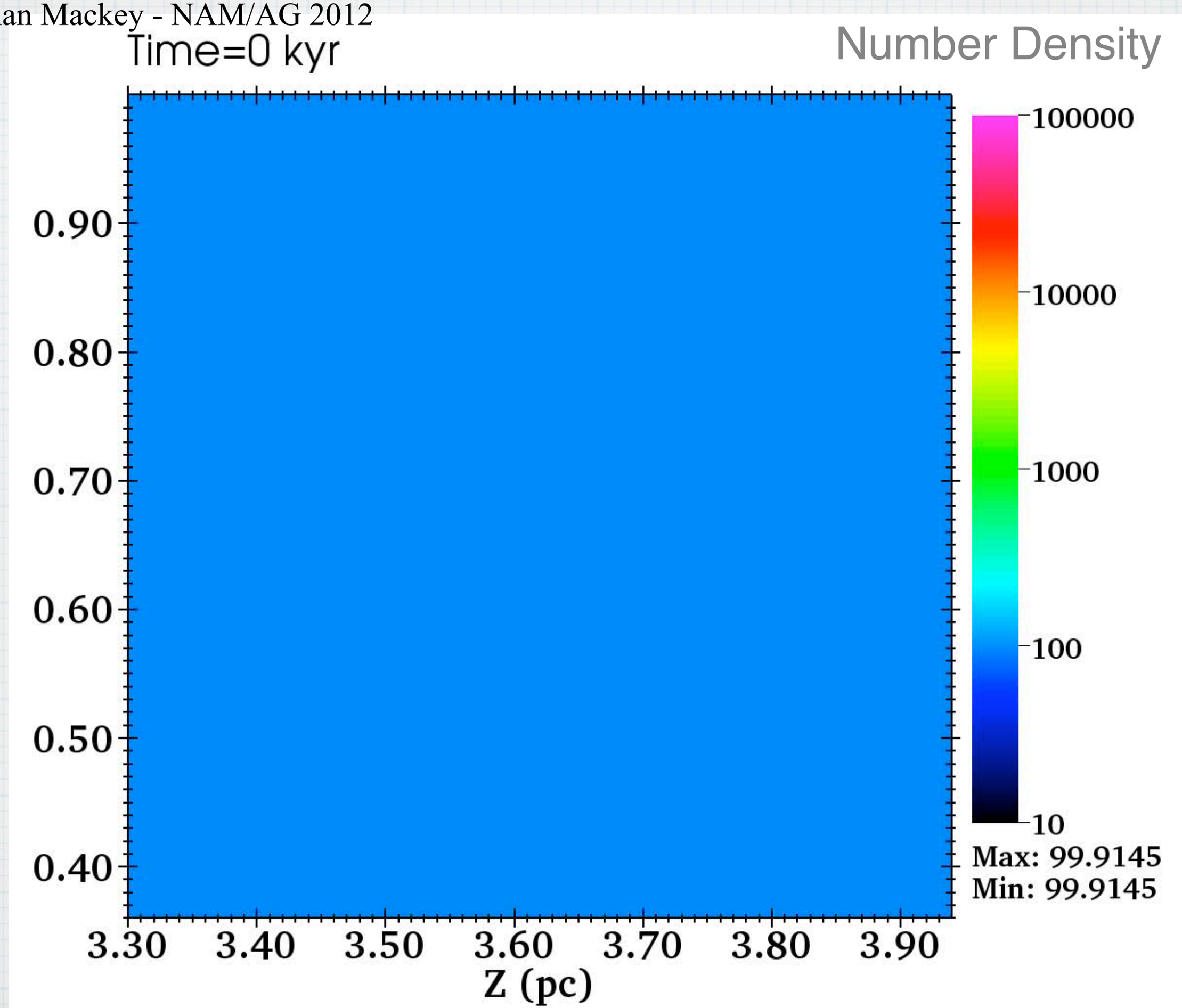


Parallel Scaling - 2D Dynamic



Conclusions

* 2nd order explicit algorithm (A3) is both more accurate and efficient than 1st order scheme (A2) commonly used.
* A3 is also more efficient than implicit method for this implementation,
  (but see Friedrich+(2012) for updated C2-ray algorithm).
* A3 allows full ionisation of grid-cell with 8 raytracings, with error <2% for all cases tested.
* This is a factor of 5-7x better than 1st order scheme.
* Upgrade from A2 to A3 should be straightforward, regardless of grid structure (also for diffuse radiation?).
* Parallel scaling is good - 50% efficiency on 256 cores, and continued speed-up to 1024 cores (for uniform grid).


Time=0 kyr
Number Density
100000
10000
1000
100
10
Max: 99.9145
Min: 99.9145
0.90
0.80
0.70
0.60
0.50
0.40
3.30
3.40
3.50
3.60
3.70
3.80
3.90
Z (pc)