Synchronous Parallel Kinetic Monte Carlo Application to Critical, billion-atom, 3D Ising Systems

Enrique Martinez1, Paul R. Monasterio2, Mal Kalos3, Jaime Marian3

1Los Alamos National Laboratory
2Massachusetts Institute of Technology
3Lawrence Livermore National Laboratory

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.
Lawrence Livermore National Laboratory, P.O. Box 808, Livermore, CA 94551

Motivation: Method

- Kinetic Monte Carlo is widely used in many scientific disciplines.
- Several methods to (more or less) efficiently parameterize kMC models.
- Not so much work on speeding up kMC itself.
- Parallelization is commonplace in MD simulations and as a way to 'slave' computationally-demanding calculations of saddle points, attempt frequencies, etc, in kMC methods.
- What about parallelization of kMC?

Motivation: Application

- Ising model can be used to map discrete-lattice systems: lattice gas, binary alloys, magnetism, etc.
- Ising system is interesting for studying second-order phase transformations.
- Belongs to a universality class for systems with long-range correlated disorder.
- Very large systems must be considered to capture the kinetic behavior, particularly in 3D.
- No analytical solution for critical kinetics in 3D, only slow converging numerical solutions.
- Many methods employed over the years.

Ising system kinetics

Master equation:

$$\frac{\partial p(\sigma,t)}{\partial t} = \sum_i \{\mathcal{W}_i(\sigma)p(\sigma,t) - \mathcal{W}_i(\sigma')p(\sigma',t)\}$$

Transition rates for Glauber dynamics:

$$\mathcal{W}_i(\sigma) = \frac{\lambda}{2}\left[1 - \sigma_i \tanh(2\beta\Delta E_i)\right]$$

ΔEi follows from the Ising Hamiltonian

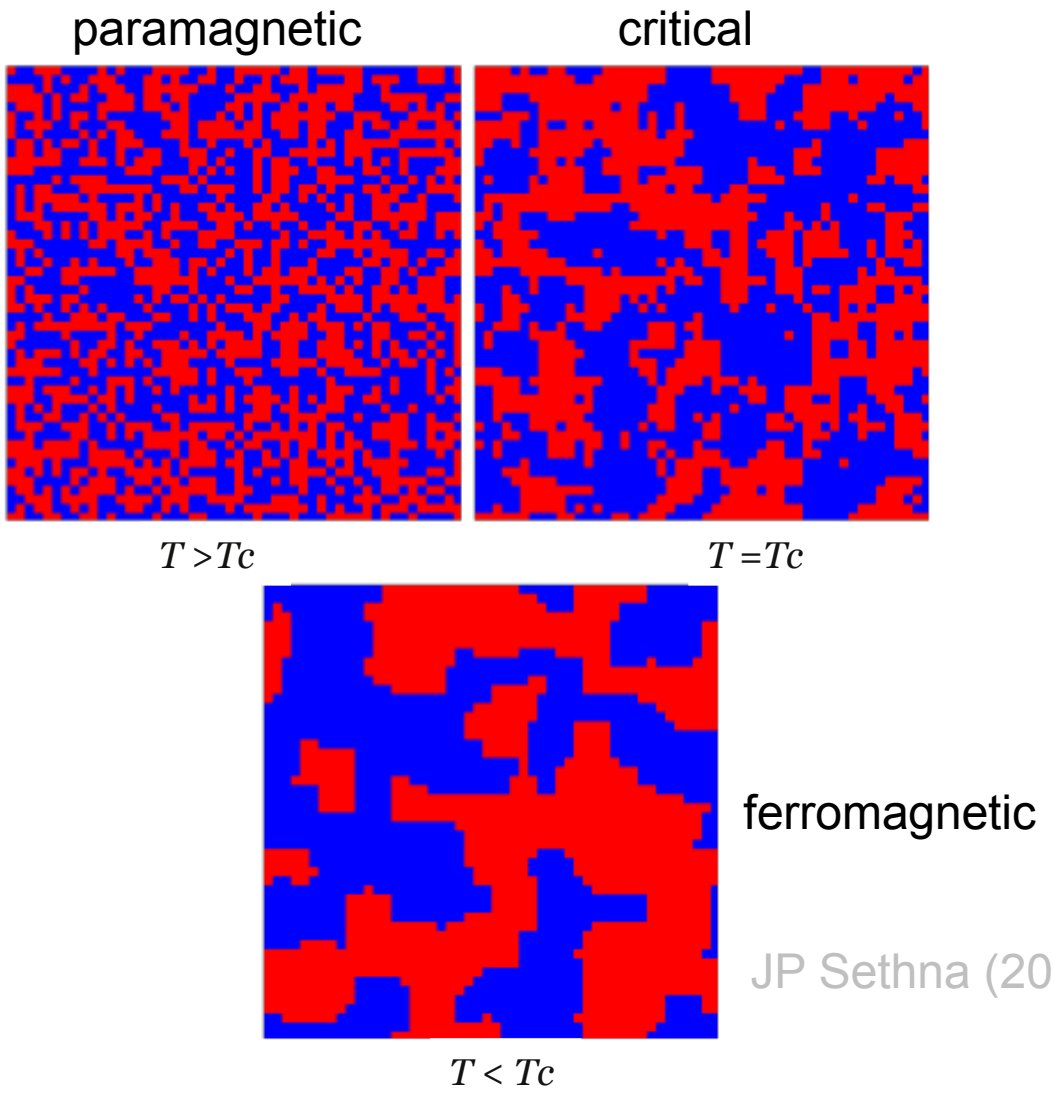
$$\mathcal{H}(\sigma) = -J\sum_{\langle i,j\rangle}\sigma_i\sigma_j \rightarrow \Delta E_i = -J\sum_{\langle i,j\rangle}\sigma_j$$

Temperature behavior of Ising systems

- At the critical temperature Tc, domains of aligned spins are created.
- These domains are defined by a correlation length ξ:

$$\xi \propto |T - T_c|^{\nu}$$

- v is the 'scale' critical exponent.
- Critical exponents not converged for 3D.



JP Sethna (2009)

The net magnetization is the order parameter of the Ising system

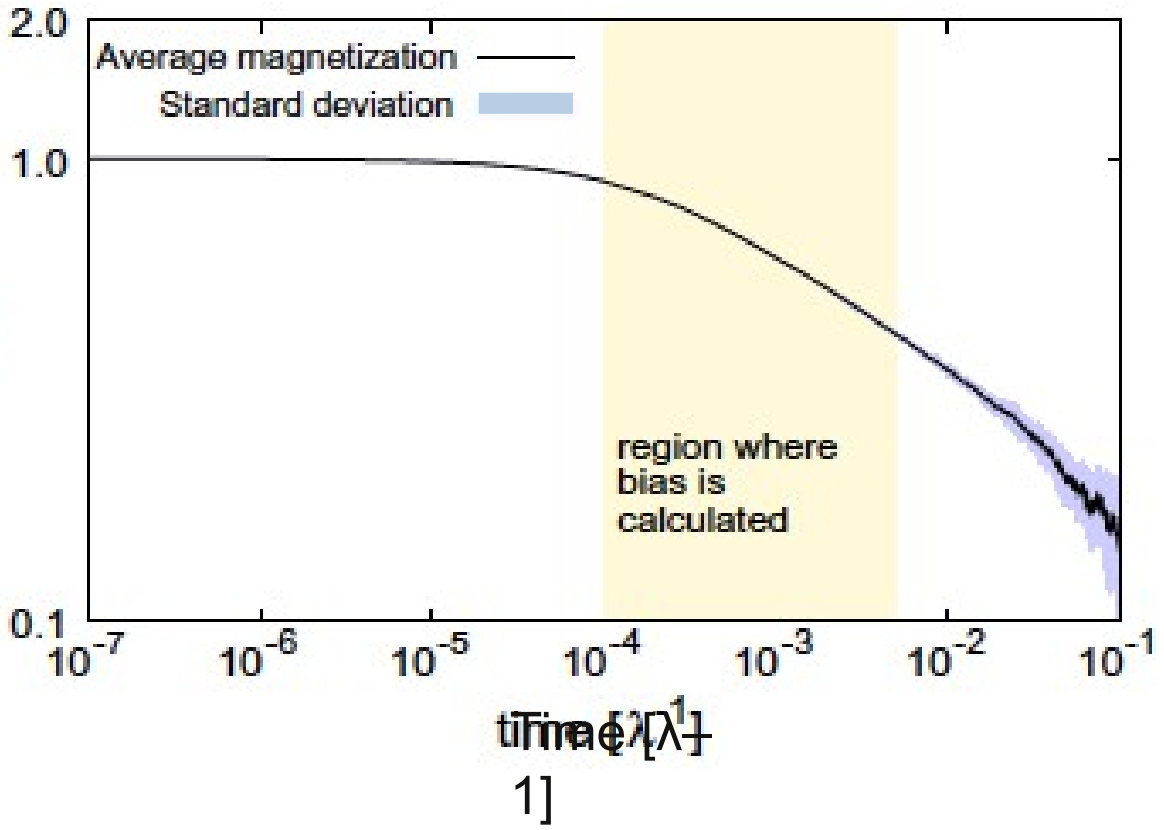
$$m(\sigma) = \frac{1}{N}\sum_i \sigma_i$$

$$m(t) \propto t^{-\kappa/z\nu}$$

The value of the critical exponents in 1D and 2D is analytically known and can be converged for 4 and higher dimensions.

In 3D: no converged numerical solution.

$$z = -\frac{\kappa}{\nu}\left[\frac{d(\log m)}{d(\log t)}\right]^{-1}$$



Serial kMC not sufficient

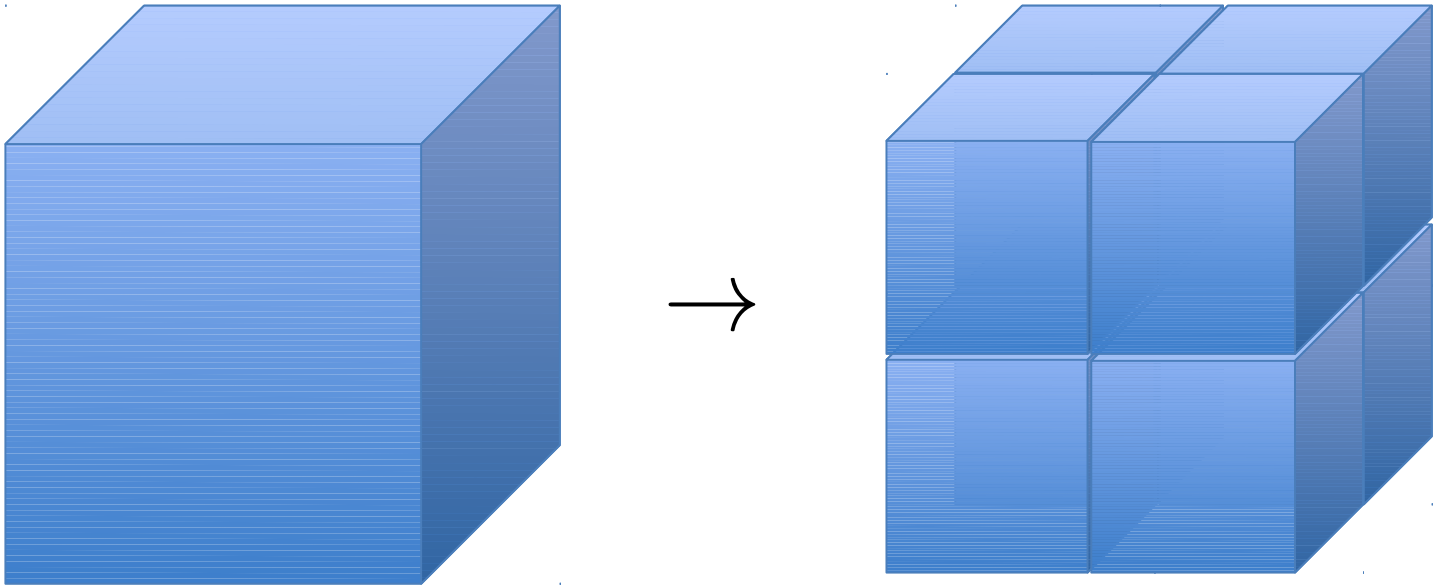
Parallel kMC algorithms to study large kinetic systems

- Discrete event kinetics are inherently difficult to parallelize.
- Traditional parallelization approaches based on **asynchronous** kinetics (Lubachevsky 1988, Jefferson 1985).
- **Causality errors** arise with these approaches: mutually affecting events occurring in different domains.
- This requires '**roll-back**' techniques to reconcile the time evolution of different processors.
- This leads to implementation complexity and regions of low efficiency.
- Rigorous and semi-rigorous algorithms have been proposed (Amar and Shim 2003, Shim and Amar 2005,

We use a novel synchronous parallel kMC algorithm to study very large systems

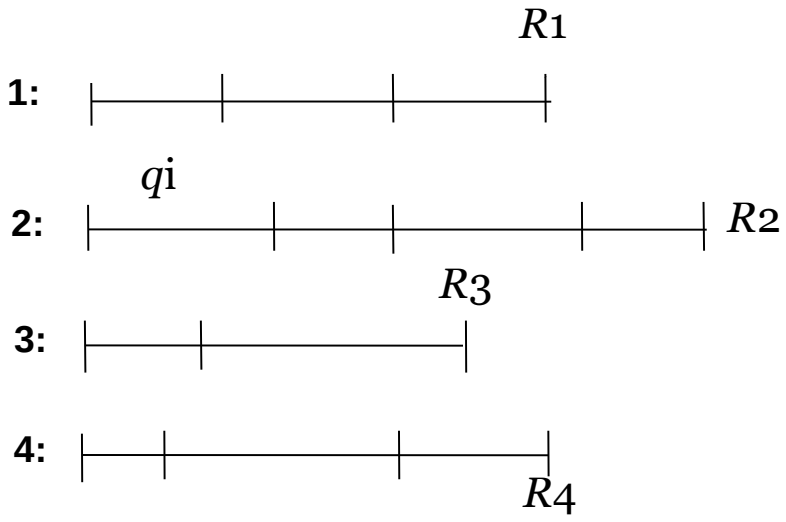
Assume a spatial domain containing *N* walkers:

- Each walker defined by a rate q_i, $R_{\text{tot}} = \sum_i^N q_i$
- Perform spatial domain decomposition:



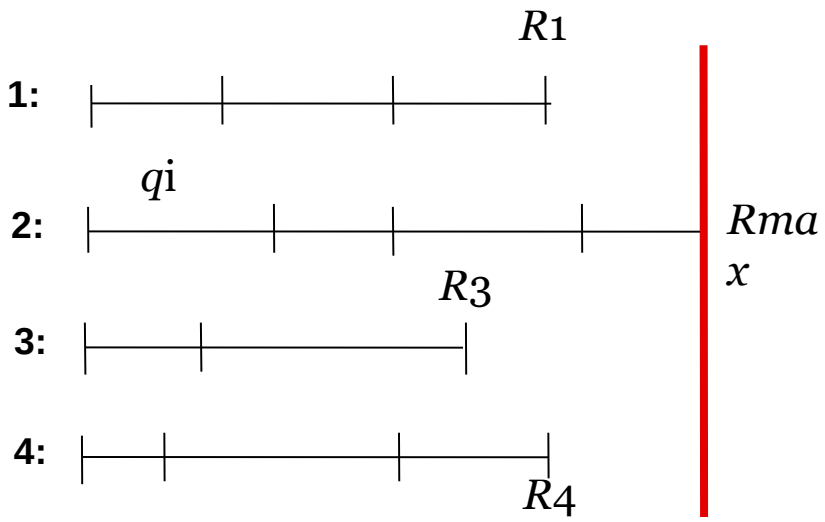
We use a novel synchronous parallel kMC algorithm to study very large systems

Now, for parallel kMC, perform K (4) domain partitions and construct frequency lines:



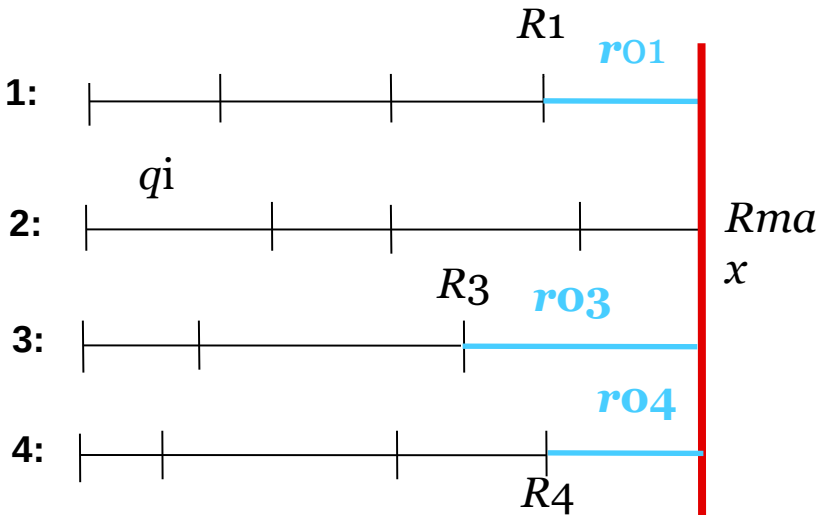
We use a novel synchronous parallel kMC algorithm to study very large systems

Now, for parallel kMC, perform K (4) domain partitions and construct frequency lines:



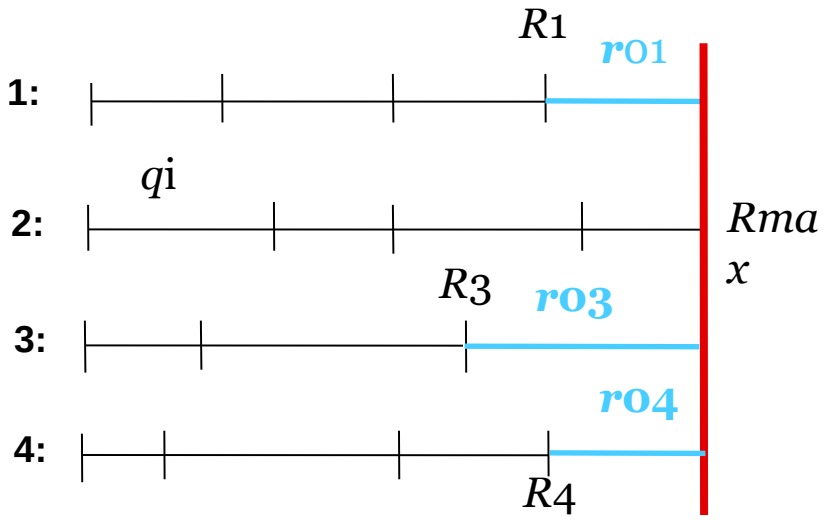
We use a novel synchronous parallel kMC algorithm to study very large systems

Now, for parallel kMC, perform K (4) domain partitions and construct frequency lines:



We use a novel synchronous parallel kMC algorithm to study very large systems

Now, for parallel kMC, perform K (4) domain partitions and construct frequency lines:



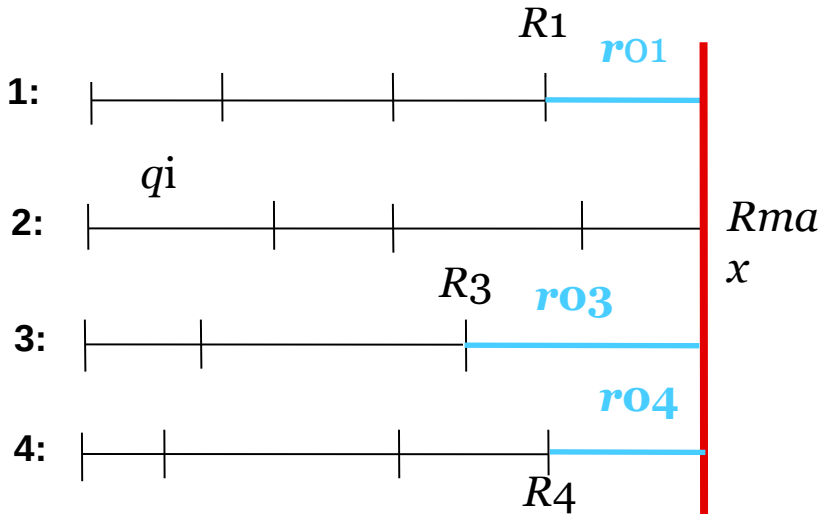
The $r0k$ are the 'dummy' rates (no event) that ensure synchronicity:

$$R_{max} = r_k + r_{0k}$$

$$R_{tot} = \sum_k r_k \leq K R_{max}$$

We use a novel synchronous parallel kMC algorithm to study very large systems

Now, for parallel kMC, perform K (4) domain partitions and construct frequency lines:



The $r0k$ are the 'dummy' rates (no event) that ensure synchronicity:

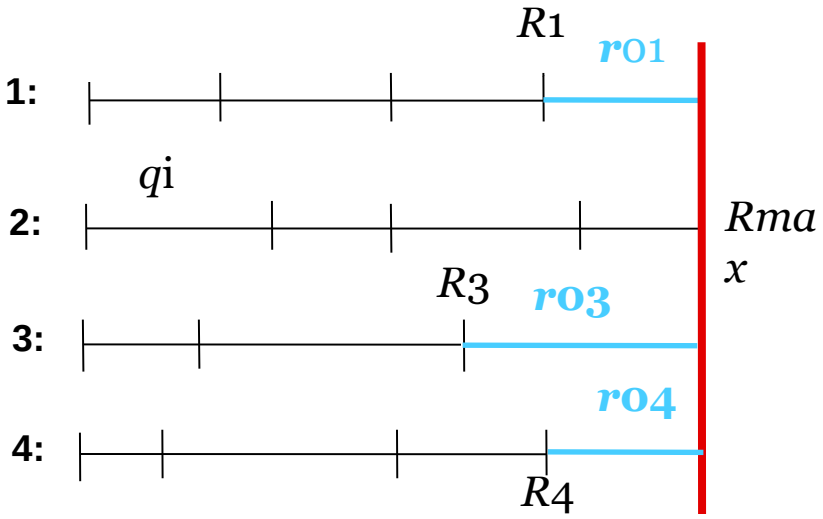
$$R_{max} = r_k + r_{0k}$$

$$R_{tot} = \sum_k r_k \leq K R_{max}$$

$$\left.\begin{aligned} \delta t_p &\approx \frac{1}{R_{max}} \\ \delta t_s &\approx \frac{1}{R_{tot}} \end{aligned}\right\} \delta t_p \leq K \delta t_s$$

We use a novel synchronous parallel kMC algorithm to study very large systems

Now, for parallel kMC, perform K (4) domain partitions and construct frequency lines:



The $r0k$ are the 'dummy' rates (no event) that ensure synchronicity:

$$R_{max} = r_k + r_{0k}$$

$$R_{tot} = \sum_k r_k \leq K R_{max}$$

$$\left.\begin{aligned} \delta t_p &\approx \frac{1}{R_{max}} \\ \delta t_s &\approx \frac{1}{R_{tot}} \end{aligned}\right\} \delta t_p \leq K \delta t_s$$

For optimum scalability, perform domain decomposition subject to the following constraint:

$$\min \left[\sum_k r_{0k} \right]$$

Parallel kMC algorithm

1. Perform spatial decomposition into K domains.
2. Define partial aggregate rates in each Ω_k: $r_k = \sum_i^{n_k} q_{ik}$ $R_{\text{tot}} = \sum_k^{K} r_k$
3. Choose the maximum partial rate as: $R_{\text{max}} = \max_k \{r_k\}$
4. Assign 'null' rates to each Ω_k such that: $r_{k0} = R_{\text{max}} - r_k$
5. Sample event from each subdomain with probability q_{ik}/R_{max} $\delta t_p = -\frac{\ln \xi}{R_{\text{max}}}$
6. Execute event and advance time by

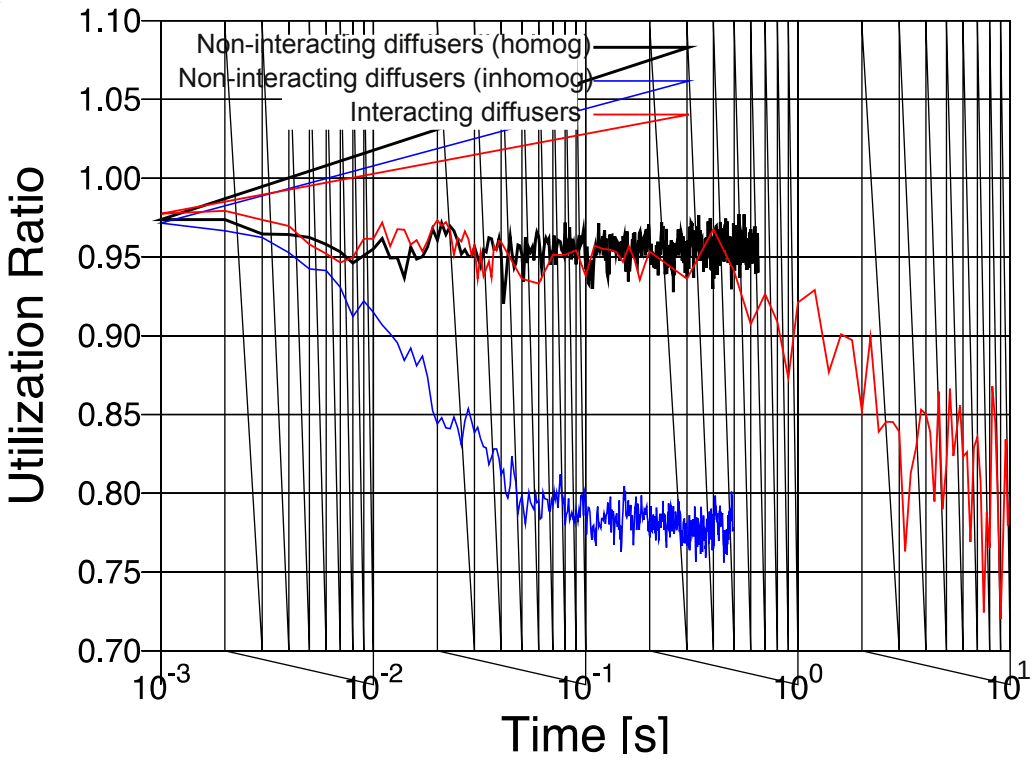
Parallel boost

Utilization Ratio (UR):
probability of not having a null event in a given kMC cycle

$$UR = 1 - \frac{\sum_k r_{k,0}}{KR_{max}}$$

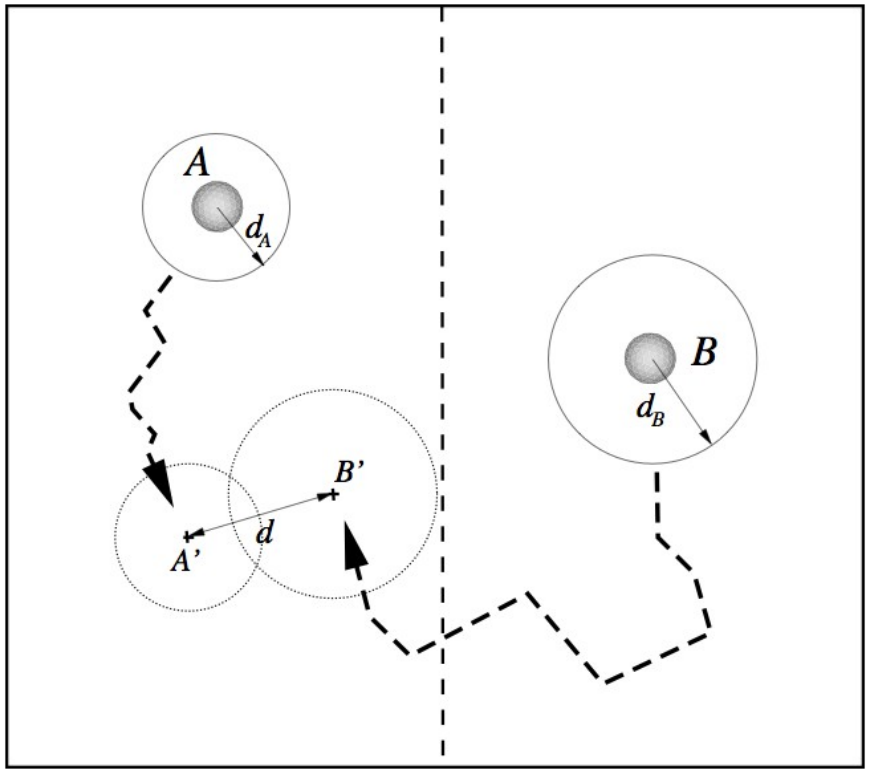
parallel timestep:
proportional to the number of processors *K* and the *UR*

$$dt_p = K \cdot UR \cdot dt_s$$

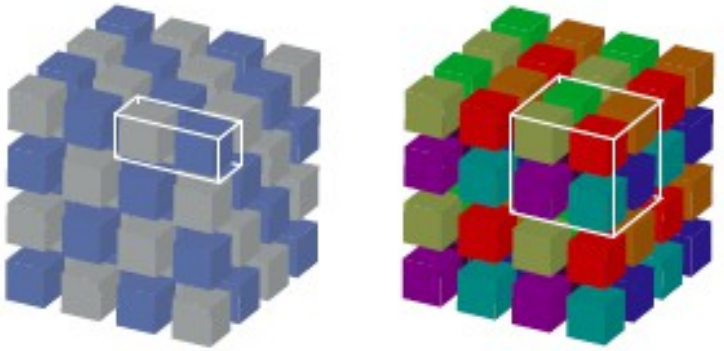


Solution of boundary conflicts via sublattice decomposition

Boundary conflicts appear when mutually-influencing events occur simultaneously on different domains



A simple solution is to use a sublattice decomposition (chess method in 2D)



Amar *et al*. (2004, 2005)

Co-occurring events only on identically-colored subcells

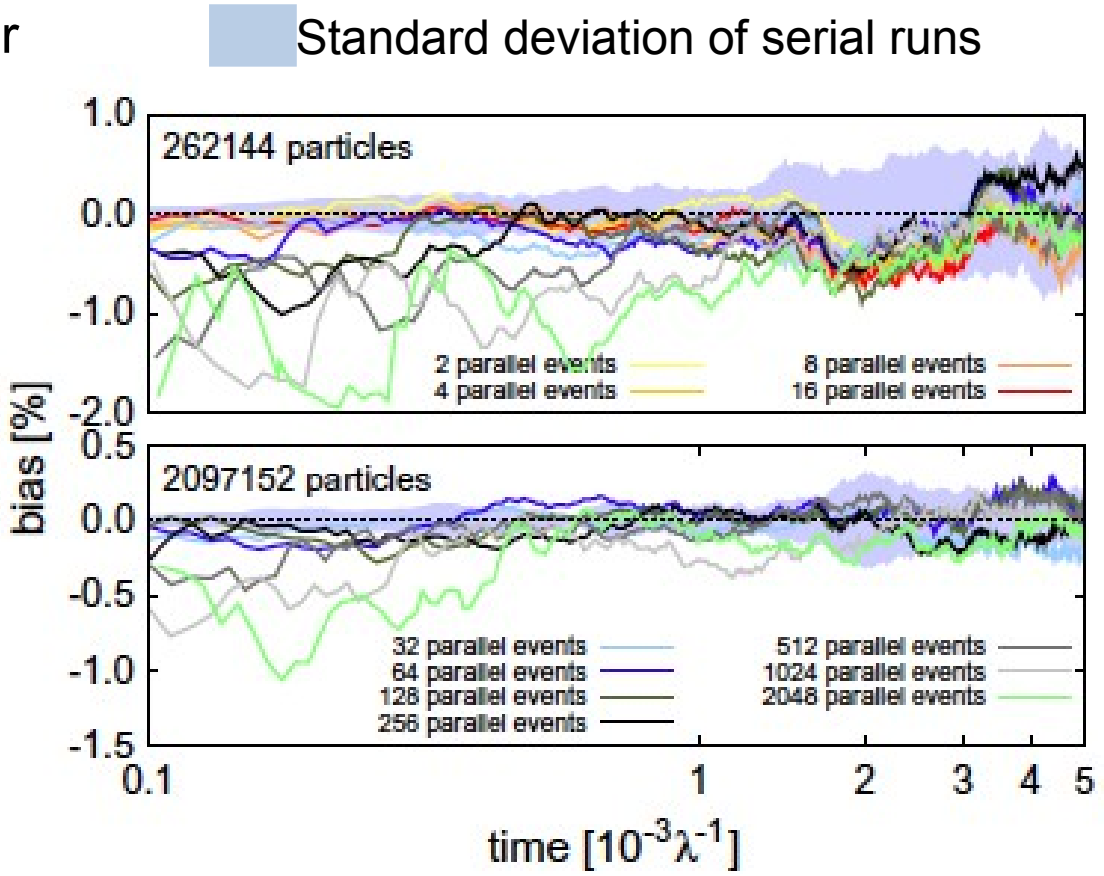
Sublattice decomposition introduces a bias

- Reduced sampling space introduces a systematic error (bias).

$$\text{bias} = \langle m(\sigma_c)\rangle_p - \langle m(\sigma_c)\rangle_s$$

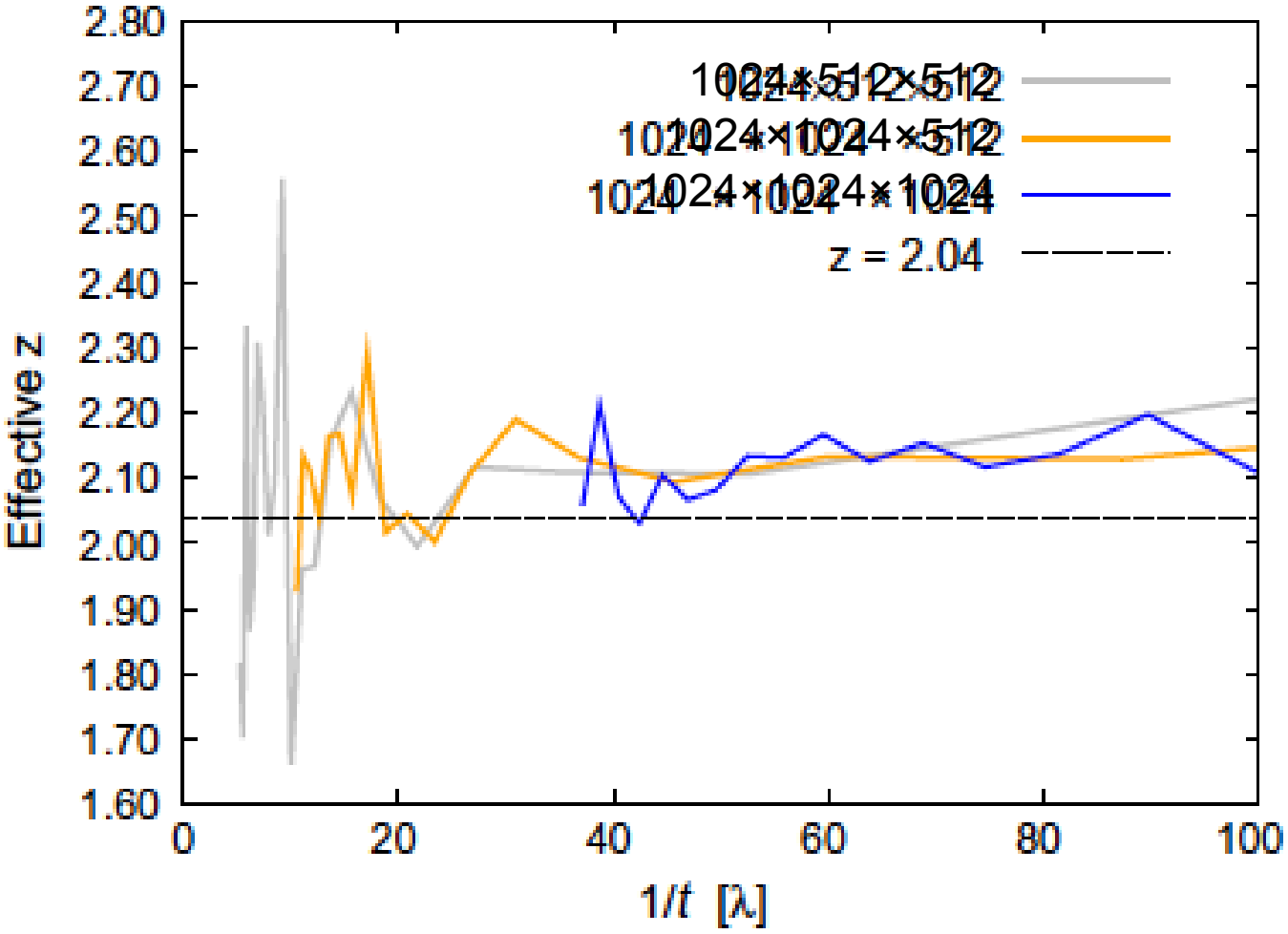
- Bias can be controlled with system size and numbers of processors.

- One must also include intrinsic statistical fluctuations of the parallel runs: $\sigma_b = \sqrt{\sigma_p^2 + \sigma_s^2}$



- We find that σb is always less than the standard deviation of the serial calculations

Calculation of critical exponent z

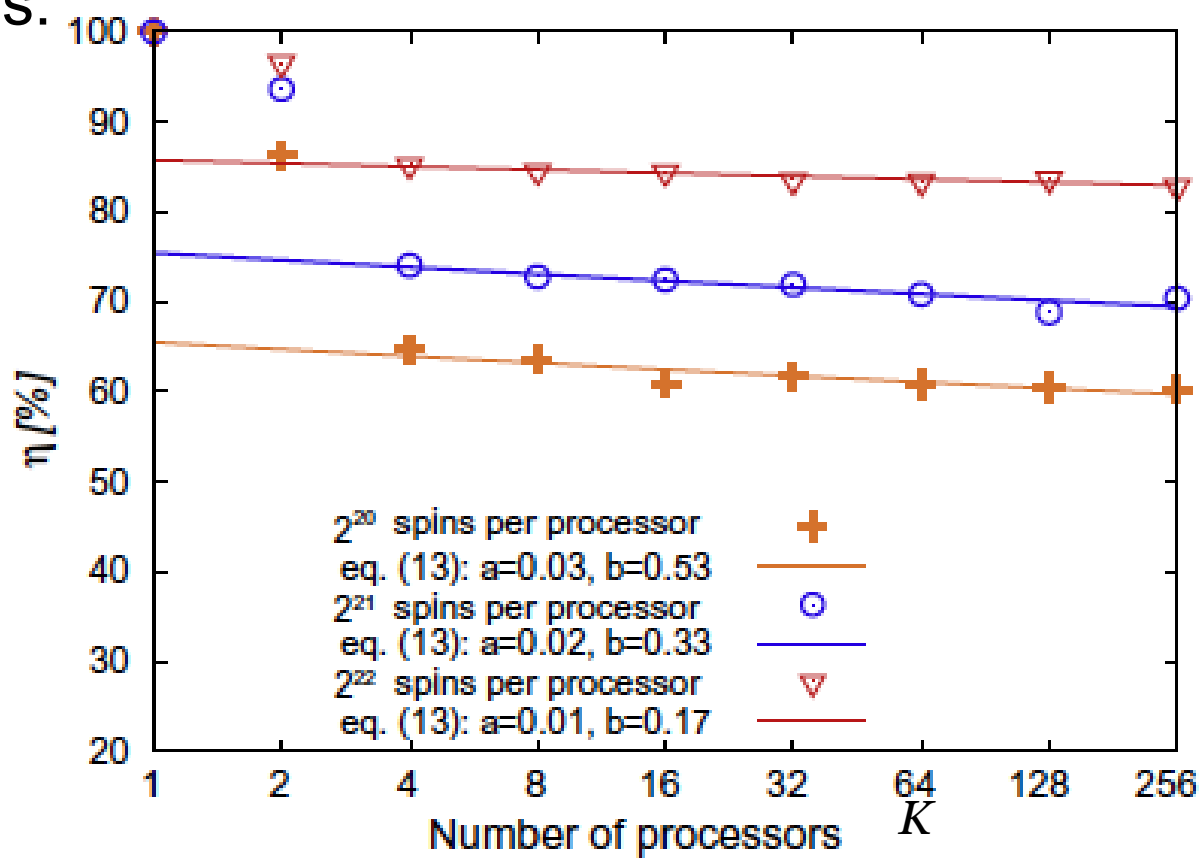


Martinez, Monasterio, Marian, JCP (2010)

Weak scaling of parallel kMC algorithm is good.

Parallel efficiency governed by local MPI calls:

$$\eta = \frac{1}{1 + a \log K + b}$$



Conclusions

· Parallel synchronous kMC algorithm suitable for large systems.

- Resolution of boundary conflicts
- Good scalability
- Controlled sampling errors

· Critical behavior of Ising systems is well reproduced and converged to the state of the art.

· Current and future applications of the method include solid solution precipitation, segregation, and in general situations where large systems are required.