An Event-Driven Kinetic Monte Carlo Algorithm for Reaction-Diffusion Systems

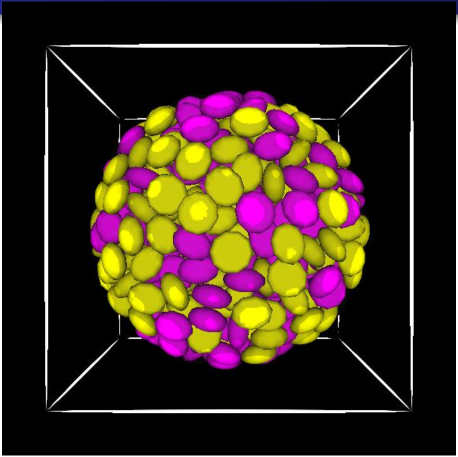
Aleksandar Donev, Courant Institute[1]
Vasily Bulatov, Lawrence Livermore National Laboratory

[1]Work done while a Lawrence Postdoctoral Fellow at LLNL under the auspices of the U.S. Department of Energy under Contract DE-AC52-07NA27344 (LLNL-PRES-409441).

CIMS Graduate Student Seminar
Dec. 2010

Outline

Packing of M&Ms



(MNG)(MPEG)

Improving the Density of Jammed Disordered Packings using Ellipsoids, **Science, 2004** [1]
A. Donev, I. Cisse, D. Sachs, E. A. Variano, F. H. Stillinger, R. Connelly, S. Torquato and P. M. Chaikin

Time-Driven (TD) Molecular Dynamics

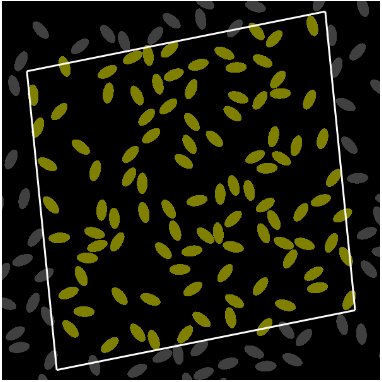
Time-Driven Molecular Dynamics (TDMD) for soft particles (ODE integrators):

1. All of the particles are displaced *synchronously* in small *time steps* Δt, calculating positions and forces on each particle at every time step.
2. It is *not* rigorous (there is an error $\sim \Delta t$), but it is very well-understood and widely implemented.
3. Discontinuous changes of the state, aka events, occur *a posteriori*, in the middle of time steps (e.g., chemical reactions).

Asynchronous Event-Driven (AED) Algorithms

- *Event-Driven Molecular Dynamics* (EDMD) for hard particles:
  - Time is advanced from one *event* to the next event.
  - *Asynchronous*: Each particle is at the point in time when the last event involving it happened.
  - Given infinite numerical precision, this kind of approach can *rigorously* follows the dynamics of the system.
- Note: There also exist **synchronous** event-driven algorithms, for example, dynamic Markov chain Monte Carlo algorithms.
- Asynchronous event-driven algorithms naturally handle **variable time-scales**.

Molecular Dynamics (MD) Algorithm



(MNG)(MPEG)

- Event-driven MD (**EDMD**) packing algorithm *ala* **Lubachevsky-Stillinger**
- The very first MD simulations (Alder & Wainwright) were event-driven hard-disk MD!
- Non-spherical particles are much more difficult to handle than spheres (collision prediction).

Neighbor List Collision-Driven Molecular Dynamics Simulation for Nonspherical Particles [2]
A. Donev, F. H. Stillinger, and S. Torquato**, J. Comp. Phys, 2005**

Basic Algorithm

- Each particle has its own *current time* t predicts its *impending event* (t_e, p_e).
- Types of events: *binary collision*, boundary events, internal events, geometrical events, etc.
- Each particle i predics events with particles and objects in its *neighbourhood* $\mathcal{N}(i)$.
- *Collision predictions* must be kept symmetric, that is, if i predicts an event with j, it changes j's prediction as well.
- *Event schedule* consists of a *priority queue* of time-ordered impending events, one for each particle.

Event Loop

1. Delete (pop) the top of the event queue (heap) to find the next particle i to have an event with $p_e(i)$ at $t_e(i)$.
2. Advance the global simulation time $t \leftarrow t_e(i)$.
3. Move i to time t, $\mathbf{r}(i) \leftarrow \mathbf{r}(i) + [t - t(i)]\,\mathbf{v}_i$, and set $t(i) \leftarrow t$, if necessary.
4. If $p_e(i) \equiv \mathcal{N}(i)$, then update $\mathcal{N}(i)$.
5. If event is a wall collision, process the collision (update the momentum of i).
6. If event is a binary collision, then:
    1. Move particle $j = p_e(i)$ to time t and set $t(j) \leftarrow t$ and mark j's event as an update.
    2. Process the binary collision between i and j.
7. *Critical*: Predict the next event for particle i, checking for collisions with walls and particles in $\mathcal{N}(i)$.
8. Insert particle i back into the event heap with key $t_e(i)$.

Neighbour Search

The **Linked-List Cell** method:

1. Partition the simulation domain into cells, and bin the particles into the bins based on *centroid* position.
2. Cell partitioning is *independent* of the particle motion.
3. *Transfer* events monitor centroids.

The **Near-Neighbor List** (NNL) method:

1. Each particle has its *bounding neighborhood*: region of space where interacting particles may be present.
2. Each particle has a *list of neighborhoods* its bounding neighborhood overlaps with.
3. Use the cell partitioning when building the NNLs.

Reaction-Diffusion Particle Models

- Systems of diffusing particles that react with other particles upon collision are a common model in computational materials science: **reaction-diffusion models**.
- Examples include: *diffusion-limited chemical reactions, signal transduction in cells, radiation damage in metals, dopant implantation in semiconductors, epitaxial deposition and growth of thin films, population dynamics*, etc.
- Continuum models are often unable to correctly capture some key property, notably the **strong heterogeneity** in space/time (e.g., *clustering*), and intrinsic **fluctuations** (e.g., *nucleation*)
- **Continuous-Time Markov Chain** models are an attractive but *expensive* alternative:
  A collection of Brownian hard spheres that diffuse through a homogeneous continuum and react upon collision with other particles or surfaces.

Example: Chemotaxis in *E. Coli*.

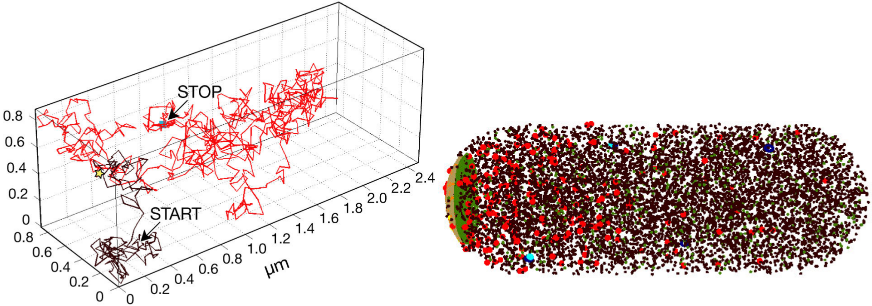


Figure: **Bacterial chemotaxis** as studied using *Smoldyn* by Karen Lipkow and Steven Andrews [*J. Bacteriol. 187(1):45-53, (2005)*]

Example: Radiation Damage

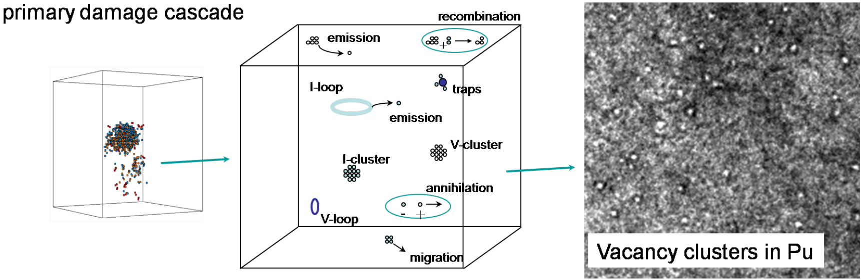
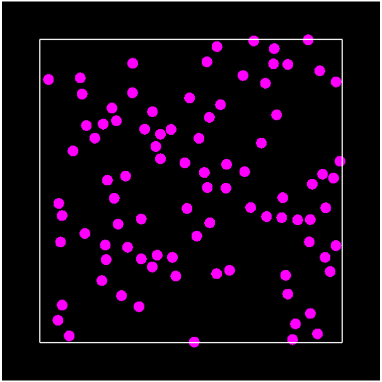


Figure: Defect creation and clustering in metals during irradiation.

Diffusion Kinetic Monte Carlo

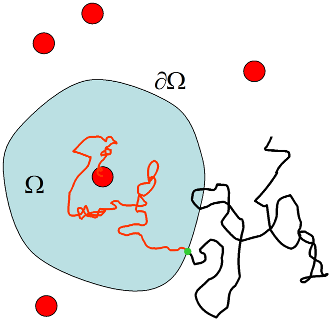


(MNG)

- Some or all unit events are diffusion hops: a set of N hard objects walking randomly on a lattice or in continuum space.
- Upon collision particles *react* (**collision events**).
- Example: Diffusion-controlled annihilation $A + A \rightarrow 0$.
- Great many diffusion hops necessary to bring particles to collisions at low density.

Traditional *synchronous* $n-$fold event-driven algorithm (BKL). Other types of Poisson events (birth, decay, boundary, etc.) are easy to handle.

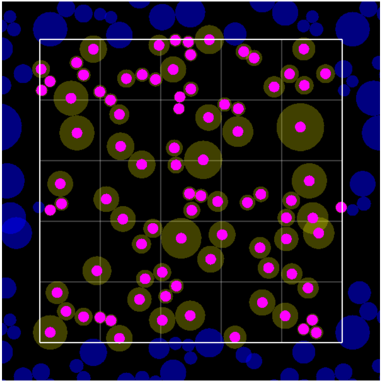
Green's Function Diffusion Theory



- Given a region of space Ω, one can determine the probability distributions for when and where (on ∂Ω) a particle will first leave that region (*first-event prediction*).
- Given that a particle has not yet left that region, one can determine the probability of finding the particle at some point inside the region at a given time (*no-event propagation*).

For pairs of particles, reduce to two *independent* **center-of-mass** and **difference walkers**.

First Passage Kinetic Monte Carlo (FPKMC)



(MNG)

- Construct disjoint protective regions (cubes, spheres) at $t = 0$.
- Main events are **(super)hops** to $\partial\Omega$. For each walker (particle or pair) **randomly** draw first passage time from the appropriate PDF.
- Find the earliest time in the queue, propagate the particle/pair to boundary/collision, construct a new protective region, insert back into queue with a new event time, repeat [3, 4]!

Advantages of the Algorithm

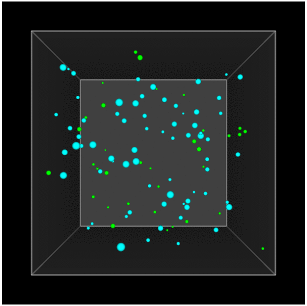
- The FPKMC algorithm is **exact** for continuous diffusion problems because it breaks the hard N-body problem into tractable one- and two-body problems.
- It is the first use we know of of **time-dependent Green's functions**.
- The algorithm automatically adjusts to variable timescales: **multiscale**.
- We have a code that implements different types of reactions (annihilation, coalescence, chemical reactions, decay/emission, hard-sphere repulsion).

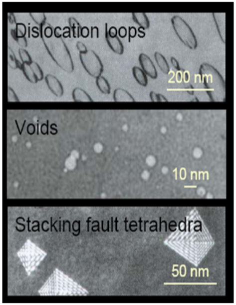
Disadvantages of the Algorithm

- The method is significantly more **complicated** to implement than BKL KMC and it requires analytical solutions (1-body and 2-body problems).
- Multi-particle reactions cause complications or slowdown (ex., nearly triple collisions).
- One can combine the asynchronous super-hops with local synchronous small hops in a **mixed time-driven/event-driven approach** [5].
- FPKMC can be viewed as a **general-purpose accelerator** that brings particle within interaction range quickly, after which application-specific handling should take over.

FPKMC for Radiation Damage

- Diffusion-reaction model for **radiation damage in metals**: diffusing and reacting *vacancies* and *interstitials* and their clusters





- A Kinetic Monte Carlo (KMC) simulation faithfully follows every **atomistic event**: *cascade insertion*, *diffusion hop*, *annihilation*, *recombination*, *clustering*, *dissociation*, *trapping*, *escape*, etc [6].

Radiation Damage KMC Model

- Very simple additive hard-sphere model for **testing** purposes, based on work by Barbu *et al*.
- **Species**:
  - *monomers*, including highly-mobile interstitials (I) and less-mobile vacancies (V), with diffusion coefficient

$$D_1 = D_0 e^{-E_m/kT}$$

  - mobile *cluster species*, including dimers (I_2 and V_2) and trimers (I_3 and V_3), with radius

$$R_c \sim R_0 + (R_1 - R_0) c^{1/3}$$

  - *immobile* species representing clusters larger than any of the mobile species (I_c and V_c)
  - *Frenkel pairs* (IV), inserted randomly with some rate

Model contd.

- **Reactions**:
  - *Coalescence*: $I + I \to I_2$ or $V + V_3 \to V_{c=4}$
  - Partial *annihilation*: $I_2 + V_{c=4} \to V_2$
  - Decay or *emission*: $V_{c=5} \to V_{c=4} + V$, or $I_2 \to I + I$, with rate

$$\Gamma_c = \Gamma_0 D_1 a^{-2} c^{2/3} e^{-E_b(c)/kT},$$

$$E_b(c) = E_f + \left[E_b(2) - E_f\right] \frac{c^{2/3} - (c-1)^{2/3}}{2^{2/3} - 1}.$$

Validation

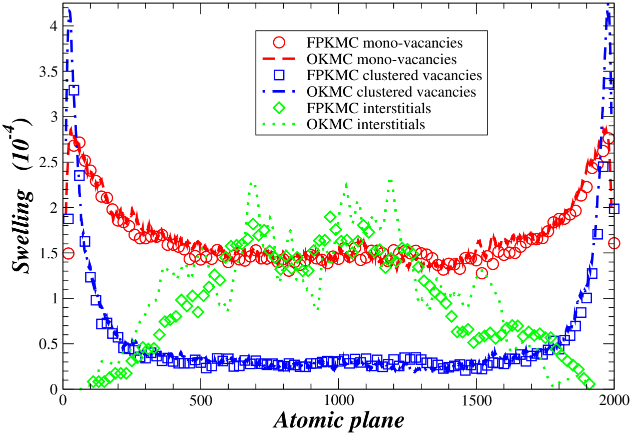
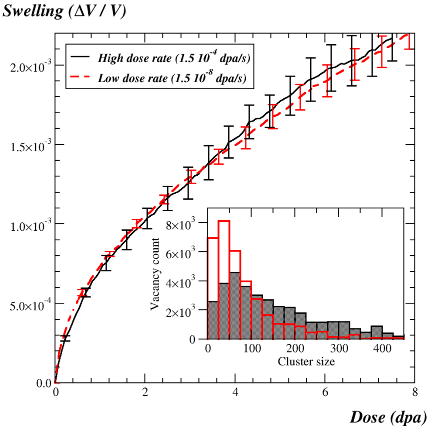


Figure: Comparison of the density profile between FPKMC (symbols) and CEA OKMC code from LAKIMOCA [7] (lines) simulations of a 0.287μm-thick film of α-iron subjected to 120 seconds of electron radiation at a temperature $T = 200^o C$.

Computational Extrapolation

- We can develop FPKMC models that match *accelerator* (e.g., JANNUS) experiments (**high dose rate**) then use the same models to simulate material behavior over a *nuclear reactor* lifetime (**low dose rate**): **computational extrapolation**
- *Approximate* scaling of results from the accelerated testing experiments is achieved when the temperature is raised sufficiently so as to keep the ratio of vacancy diffusion coefficient to irradiation rate constant.
- FPKMC allows us to quantify the mismatch with **atomistic fidelity** up to *realistic doses* (10dpa)!
- But the model is **too simple** to capture realistic physics! Work is underway at CEA to develop more detailed and accurate models.

Lab vs. Reactor



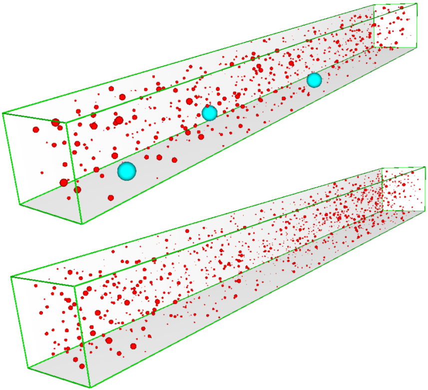


Figure: Swelling due to 10dpa electron irradiation in a pure *Fe* film

Conclusions

- Event-driven algorithms are a *very efficient alternative* to traditional time-driven simulations in situations where the evolution of a system is dominated by **discontinuous state changes** (events).
- ED algorithms are significantly **more complicated** than TD ones and work best for simple models such as hard-particle systems.
- Time-driven and event-driven handling can be *combined* together: each piece does what it is best for!
- Unfortunately, event-driven algorithms are *not* widely used and most computational scientists are not familiar with them.
- **Parallelization** of event-driven algorithms remains an important challenge.

Future Directions for FPKMC

- **Apply** FPKMC to a wider range of problems, notably, **systems biology** (compare to *Smoldyn* library).
- Extend first-passage to *lattice models* (discrete space), notably, defect diffusion through **dense alloys**.
- Approximately handle *continuous interaction potentials* (ala Brownian Dynamics), and most importantly, **long-ranged interactions** (e.g., electrostatic or elastic).
- When very fast species (e.g., interstitials) are present they slow down even event-driven algorithms: multiscale methods to utilize this **separation of time-scales**.
- The particle model can be coupled to a more efficient *continuum* model in a **hybrid method**.

References/Questions?

A. Donev, I. Cisse, D. Sachs, E. A. Variano, F. H. Stillinger, R. Connelly, S. Torquato, and P. M. Chaikin.
Improving the Density of Jammed Disordered Packings using Ellipsoids.
Science, 303:990–993, 2004.

A. Donev, S. Torquato, and F. H. Stillinger.
Neighbor List Collision-Driven Molecular Dynamics Simulation for Nonspherical Particles: I. Algorithmic Details II. Applications to Ellipses and Ellipsoids.
J. Comp. Phys., 202(2):737–764, 765–793, 2005.

T. Oppelstrup, V. V. Bulatov, G. H. Gilmer, M. H. Kalos, and B. Sadigh.
First-Passage Monte Carlo Algorithm: Diffusion without All the Hops.
Phys. Rev. Lett., 97(23):230602, 2006.

T. Oppelstrup, V. V. Bulatov, A. Donev, M. H. Kalos, G. H. Gilmer, and B. Sadigh.
First-Passage Kinetic Monte Carlo Method.
Phys. Rev. E, 80(6):066701, 2009.
arXiv:0905.3575.

A. Donev.
Asynchronous event-driven particle algorithms.
SIMULATION: Transactions of The Society for Modeling and Simulation International, 85(4):229–242, 2008.

A. Donev, V. V. Bulatov, T. Oppelstrup, G. H. Gilmer, B. Sadigh, and M. H. Kalos.
A First-Passage Kinetic Monte Carlo Algorithm for Complex Diffusion-Reaction Systems.
J. Comp. Phys., 229(9):3214–3236, 2010.
arXiv:0905.3576.

C. Domain, C. S. Becquart, and L. Malerba.
Simulation of radiation damage in Fe alloys: an object kinetic Monte Carlo approach.
Journal of Nuclear Materials, 335(1):121–145, 2004.