# Computational statistical physics: The example of hard spheres

Lectures at the 16th Chris Engelbrecht Summer School in Theoretical Physics,
Alpine Heath Resort, Drakensberg, KwaZulu-Natal, South Africa
First part: Introduction, Molecular Dynamics

Werner Krauth

Laboratoire de Physique Statistique
Ecole Normale Supérieure, Paris, France

24 January 2005

- W. Krauth "**Introduction to Monte Carlo Algorithms**" in "Advances in Computer Simulation", Lecture Notes in Physics; J. Kertesz and I. Kondor, eds, (Springer Verlag, 1998) (cond-mat/9612186)
(. . . introduction)
- W. Krauth "**Statistical Mechanics: Algorithms and Computations**" (Oxford University Press, fall 2005)
(. . . everything)

# Paris, France, July 14th, 2004

# Goals of these lectures

- study computational physics, esp. Monte Carlo method.
- study close relation betweeen physics and computing.
- don't use codes, understand them, write our own.
- use algorithms, understand them, come up with our own.

…(have to) concentrate on simple, fundamental models …

**keywords**

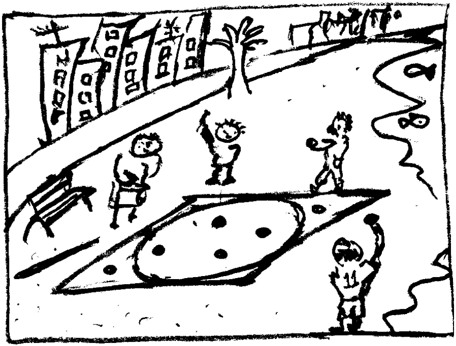
sampling (direct, Markov chain), Detailed balance, ergodicity, A priori probabilities, entropic phase transitions, cluster algorithms

# Fundamental models in statistical physics

- Ising model
- dimers
- random walks, random manifolds (surfaces)
- free bosons
- hard spheres
- quantum spins, lattice fermions

Many connections (theory, methods)

# Direct Sampling Monte Carlo



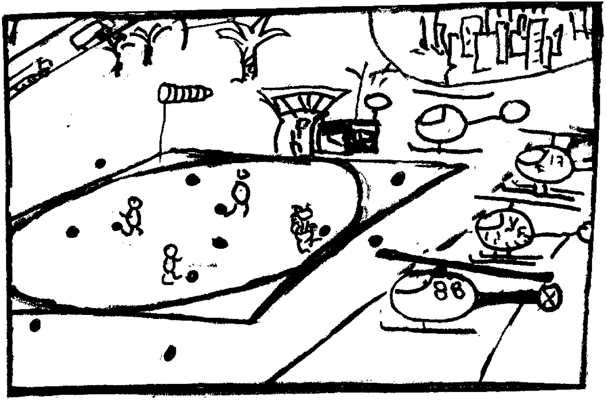
# Direct sampling Monte Carlo (algorithm)

**procedure** `simple-pi`
hits ← 0 (initialize)
**for** $i = 1, \ldots, N_{\text{throw}}$ **do**
$$\left\{ \begin{array}{l} x \leftarrow \texttt{ran}[-1, 1] \\ y \leftarrow \texttt{ran}[-1, 1] \\ \textbf{if } (x^2 + y^2 < 1) \text{ hits} \leftarrow \text{hits} + 1 \end{array} \right.$$
**output** hits

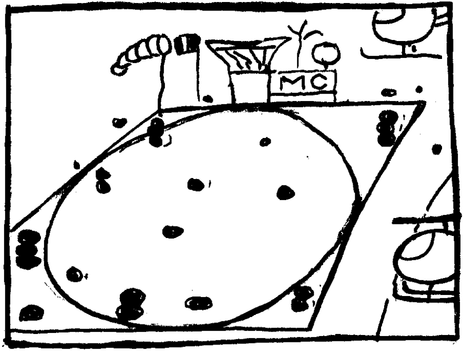
Five trials with $N_{\text{throw}} = 4000$

| run | hits | estimation |
|---|---|---|
| 1 | 3156 | 3.156 |
| 2 | 3129 | 3.129 |
| 3 | 3154 | 3.154 |
| 4 | 3134 | 3.134 |
| 5 | 3148 | 3.148 |

# Markov-chain sampling (adults' game)



# After the game




- Metropolis et al. algorithm (1953).
- Crucial role of rejections.

# Markov-chain sampling (algorithm)

**procedure** `markov-pi`

hits $\leftarrow 0$; $x \leftarrow 1$; $y \leftarrow -1$

**for** $i = 1, \ldots, N_{\text{throw}}$ **do**

$$\begin{cases} \delta x \leftarrow \texttt{ran}[-\delta, \delta] \\ \delta y \leftarrow \texttt{ran}[-\delta, \delta] \\ \textbf{if } (|x + \delta x| < 1 \text{ and } |y + \delta y| < 1) \textbf{ then} \\ \quad \begin{cases} x \leftarrow x + \delta x \\ y \leftarrow y + \delta y \end{cases} \\ \textbf{if } (x^2 + y^2 < 1) \text{ hits} \leftarrow \text{hits} + 1 \end{cases}$$

**output** hits

Alg.`markov-pi` is (far) less precise than Alg.`direct-pi`

| run | hits | estimation |
|---|---|---|
| 1 | 3123 | 3.123 |
| 2 | 3118 | 3.118 |
| 3 | 3040 | 3.040 |
| 4 | 3066 | 3.066 |
| 5 | 3263 | 3.263 |

# Throwing range $\delta$

- parameter $\delta$: 'throwing range'.

- small $\delta$: small steps, small rejection rate: low precision
- large $\delta$: large steps, large rejection rate: low precision

Rule (of thumb): use $\delta$ such that rejection rate $\simeq \frac{1}{2}$

- rejections are bad.

# Detailed balance

$$\underbrace{p(a \to a)}_{\text{probability to go from } a \text{ to } a} + p(a \to b) + p(a \to c) = 1$$

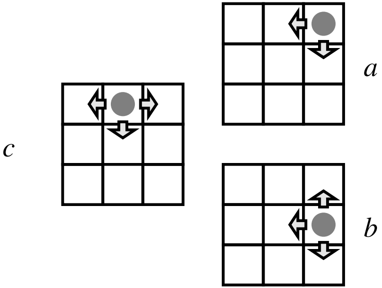
$$\underbrace{\pi(a)}_{\text{probability to be at } a} = \pi(a)p(a \to a) + \pi(b)p(b \to a) + \pi(c)p(c \to a)$$

$$\pi(a)p(a \to c) + \pi(a)p(a \to b) = \pi(b)p(b \to a) + \pi(c)p(c \to a)$$

**detailed balance condition**

$$\pi(a)p(a \to b) = \pi(b)p(b \to a) \quad \text{etc}$$

# Detailed balance (discrete heliport)




$$p(a \to b) \stackrel{!}{=} p(b \to a)$$
$$p(a \to c) \stackrel{!}{=} p(c \to a)$$
$$p(b \to d) \stackrel{!}{=} p(d \to b)$$
$$\vdots \quad \stackrel{!}{=} \quad \vdots$$

- Only simple solution $p(\ldots \to \ldots) = \frac{1}{4}$
- implies $p(a \to a) = \frac{1}{2};\ p(b \to b) = \frac{1}{4}$

# Generalized detailed balance ($\pi(x) \neq$ constant)

$$\pi(a)p(a \to b) = \pi(b)p(b \to a)$$

is solved by the Metropolis Algorithm (1953):

$$p(a \to b) = \min(1, \frac{\pi(b)}{\pi(a)})$$

We can understand this by a bureaucratic procedure:

| case | $\pi(a) > \pi(b)$ | $\pi(b) > \pi(a)$ | |
|---|---|---|---|
| $p(a \to b)$ | | | 1 |
| $\pi(a)p(a \to b)$ | | | 2 |
| $p(b \to a)$ | | | 3 |
| $\pi(b)p(b \to a)$ | | | 4 |

Important concept:
The probability to go from *a* to *b* is composite:

$$\mathcal{P}(a \to b) = \underbrace{\mathcal{A}(a \to b)}_{\text{consider } a\to b} \times \underbrace{p(a \to b)}_{\text{accept } a\to b}$$

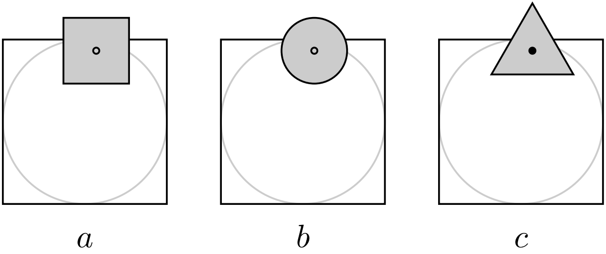
Generalized Metropolis algorithm

$$\frac{p(a \to b)}{p(b \to a)} = \frac{\pi(b)}{\mathcal{A}(a \to b)} \frac{\mathcal{A}(b \to a)}{\pi(a)}$$

$$p(a \to b) = \min\left[1, \frac{\pi(b)}{\mathcal{A}(a \to b)} \frac{\mathcal{A}(b \to a)}{\pi(a)}\right]$$

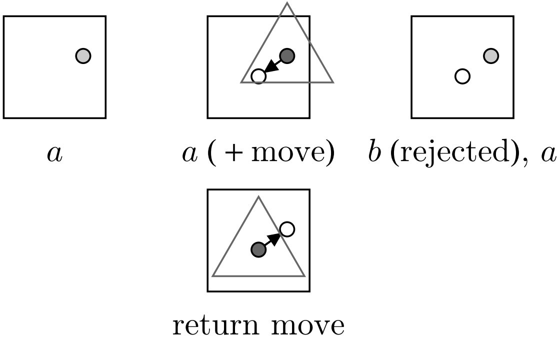
# "A priori" probabilities—Triangle algorithm I

- Variants of Alg. `markov-pi`



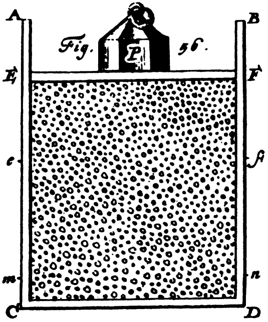

- The triangle algorithm ($c$) needs special attention.

# “A priori” probabilities—Triangle algorithm II



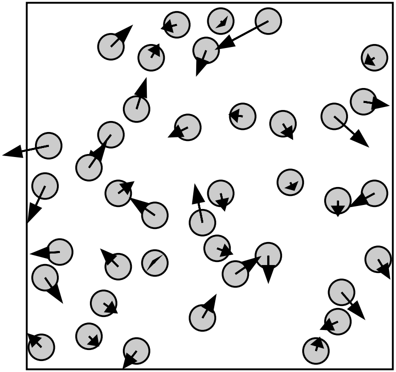

- remember: $p(a \to b) = \min\left[1, \frac{\pi(b)}{\mathcal{A}(a\to b)} \frac{\mathcal{A}(b\to a)}{\pi(a)}\right]$

# Hard spheres I




- Daniel Bernoulli (1738)
  - For $N \rightarrow \infty$: position $E - F$ stable
  - For velocities $v \rightarrow 2v$: $P \rightarrow 4P$ including atmospheric pressure
  - experiments (at constant volume).

- French King Louis XIV (1638-1715) playing billiards (Engrav. A. Trouvain)

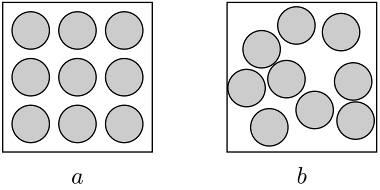


$t = 0$

- All of classical liquid-solid physics ...
  ... except distinction liquid-gas

# Hard spheres III

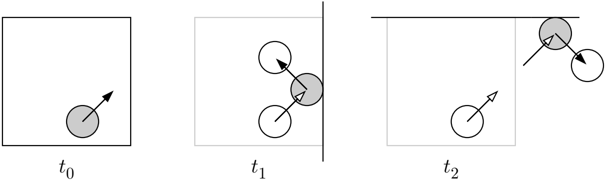
- pebbles on beach: problem in statistics
- hard spheres: problem in physics (masses, positions, equations of motion, velocities, energies)



- $\pi(a) = \pi(b)$
- Test-bed of algorithms

# collision-time with box

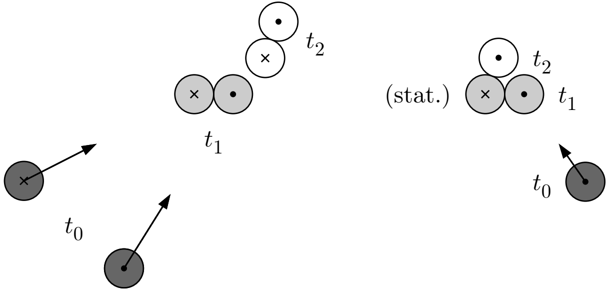
- collision of a particle with box




- trivial

# pair-collision time

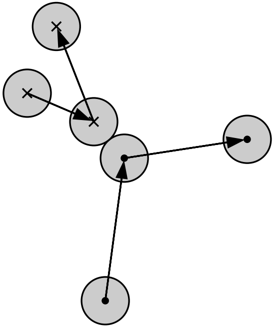
- collision of a pair of particles



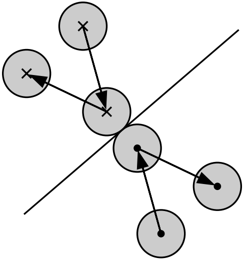
# ‘Events’ II: pair-collision time (algorithm)

**procedure** `pair-time`
**input** $\mathbf{x}_\Delta$ $(\equiv \mathbf{x}_k(t_0) - \mathbf{x}_l(t_0))$
**input** $\mathbf{v}_\Delta$ $(\equiv \mathbf{v}_k - \mathbf{v}_l \neq 0)$
$\Upsilon \leftarrow (\mathbf{x}_\Delta \cdot \mathbf{v}_\Delta)^2 - |\mathbf{v}_\Delta|^2 \cdot (|\mathbf{x}_\Delta|^2 - 4\sigma^2)$
**if** $(\Upsilon > 0$ and $(\mathbf{x}_\Delta \cdot \mathbf{v}_\Delta) < 0)$ **then**
$\left\{ t_{\text{pair}} \leftarrow t_0 - \left[(\mathbf{x}_\Delta \cdot \mathbf{v}_\Delta) + \sqrt{\Upsilon}\right] / \mathbf{v}_\Delta^2 \right.$
**else**
$\left\{ t_{\text{pair}} \leftarrow \infty \right.$
**output** $t_{\text{pair}}$

# Hard sphere collision (Newton)



lab frame



center of mass frame

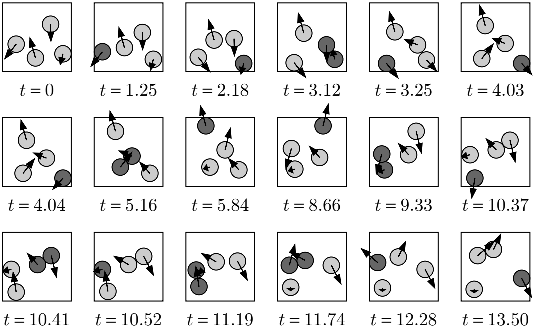
# Hard sphere collision (Newton) (algorithm)

**procedure** `pair-collision`
**input** $\mathbf{x}_k, \mathbf{x}_l$ (particles in contact: $|\mathbf{x} - \mathbf{x}'| = 2\sigma$)
**input** $\mathbf{v}_k, \mathbf{v}_l$
$\mathbf{x}_\Delta \leftarrow \mathbf{x}_k - \mathbf{x}_l$
$\hat{\mathbf{e}}_\perp \leftarrow \mathbf{x}_\Delta / |\mathbf{x}_\Delta|$
$\mathbf{v}_\Delta \leftarrow \mathbf{v}_k - \mathbf{v}_l$
$\mathbf{v}_k \leftarrow \mathbf{v}_k - \hat{\mathbf{e}}_\perp (\mathbf{v}_\Delta \cdot \hat{\mathbf{e}}_\perp)$
$\mathbf{v}_l \leftarrow \mathbf{v}_l + \hat{\mathbf{e}}_\perp (\mathbf{v}_\Delta \cdot \hat{\mathbf{e}}_\perp)$
**output** $\mathbf{v}_k, \mathbf{v}_l$

- works in all reference frames

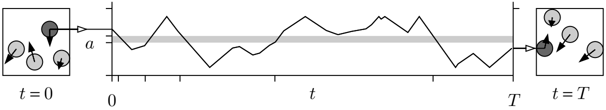
**procedure** `event-disks`
**input** $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}, \{\mathbf{v}_1, \ldots, \mathbf{v}_N\}, t$
$(t_{\text{box}}, j) \leftarrow$ next box collision
$(t_{\text{pair}}, k, l) \leftarrow$ next pair collision
$t_{\text{next}} \leftarrow \texttt{min}[t_{\text{box}}, t_{\text{pair}}]$
**for** $m = 1, \ldots, N$ **do**
  $\{\ \mathbf{x}_m \leftarrow \mathbf{x}_m + (t_{\text{next}} - t) \cdot \mathbf{v}_m$
**if** $(t_{\text{box}} < t_{\text{pair}})$ **then**
  $\{$ **call** `box-collision`$[j]$
**else**
  $\{$ **call** `pair-collision`$[k, l]$
**output** $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}, \{\mathbf{v}_1, \ldots, \mathbf{v}_N\}, t_{\text{next}}$

# “Event-driven” molecular dynamics



# Observables I (explicit time average)

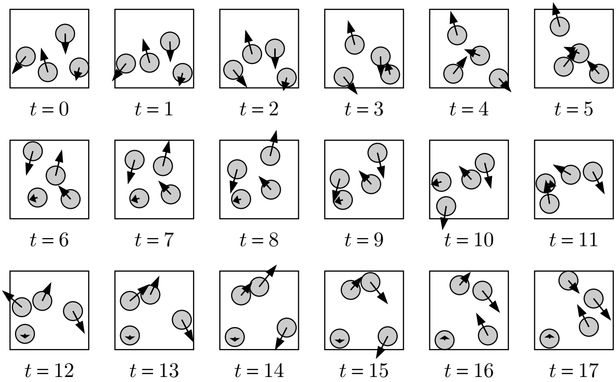
Time averages:




$$\left\{ \begin{matrix} \text{projected density} \\ \text{at } y = a \end{matrix} \right\} = \eta_y(a) = \frac{1}{T} \sum_{\substack{\text{intersections} \\ \text{with gray strip}}} \frac{1}{|v_y(i)|}$$

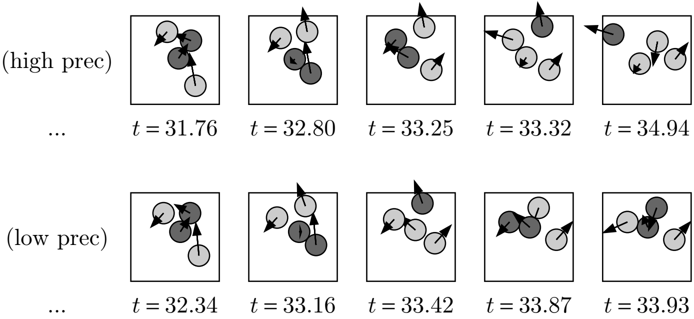
- Best of all worlds: Simulation and data acquisition are perfect

# Observables II (equal-interval frames)

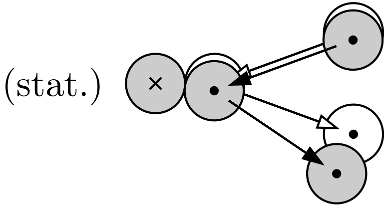



$\{$any observable $\mathcal{O}\} \simeq \{$Average over equal-time frames$\}$

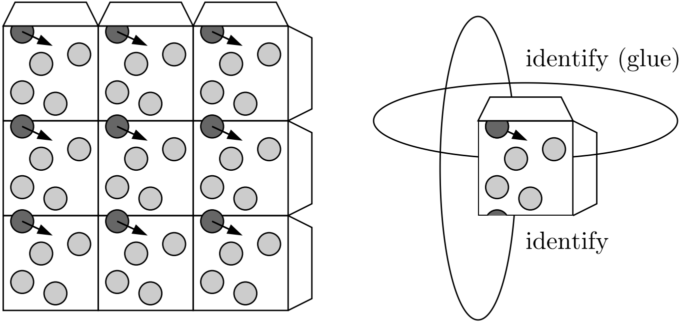
# Chaos


(high prec)
...
$t = 31.76$
$t = 32.80$
$t = 33.25$
$t = 33.32$
$t = 34.94$
(low prec)
...
$t = 32.34$
$t = 33.16$
$t = 33.42$
$t = 33.87$
$t = 33.93$

- Explanation for appearance of chaos:



- useful, improve thermodynamic limit



# Periodic boundary conditions (algorithm)

**procedure** `box-it`
**input x**
$x \leftarrow \texttt{mod}[x, L_x]$
**if** $(x < 0)$ $x \leftarrow x + L_x$
$y \leftarrow \texttt{mod}[y, L_y]$
**if** $(y < 0)$ $y \leftarrow y + L_y$
**output x**

**procedure** `diff-vec`
**input** $\mathbf{x}, \mathbf{x}'$
$\mathbf{x}_\Delta \leftarrow \mathbf{x}' - \mathbf{x}$ $(\mathbf{x}_\Delta \equiv (x_\Delta, y_\Delta))$
**call** `box-it`$[\mathbf{x}_\Delta]$
**if** $(x_\Delta > L_x/2)$ $x_\Delta \leftarrow x_\Delta - L_x$
**if** $(y_\Delta > L_y/2)$ $y_\Delta \leftarrow y_\Delta - L_y$
**output** $\mathbf{x}_\Delta$

# Sinai's hard-sphere system (1970, 1973)