Optimizing lattice Monte Carlo algorithms for diffusion

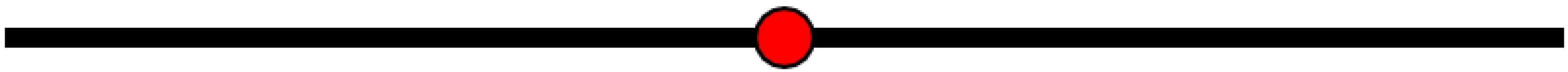
Mykyta V. Chubynsky, Hendrick W. de Haan, and Gary W. Slater

Department of Physics, University of Ottawa, Canada

We've had a few talks mentioning rejection-free MC algorithms. Situations where for standard moves the rejection rate is high and people do something more complicated to avoid that. Makes perfect sense.

What I am going to do is exactly the opposite: for a very simple problem where the rejection-free algorithm is entirely obvious, I will introduce a nonzero rejection probability artificially and show that doing this can make sense – at least for this problem.

Consider a diffusing particle (a random walker). Start with the 1D case.

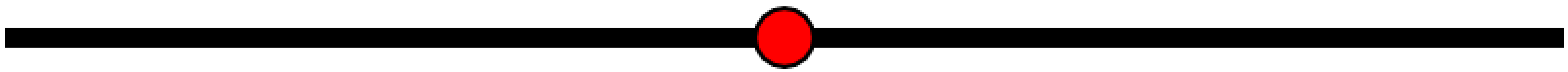


Characterized by a single parameter D (the diffusion constant, or diffusivity).

The mean-square displacement (MSD) is $\langle x^2 \rangle = 2Dt$ and the displacement distribution (DisD) is

$$P(x,t) = \frac{1}{2\sqrt{\pi D t}} \exp\left(-\frac{x^2}{4Dt}\right)$$

Consider a diffusing particle (a random walker). Start with the 1D case.



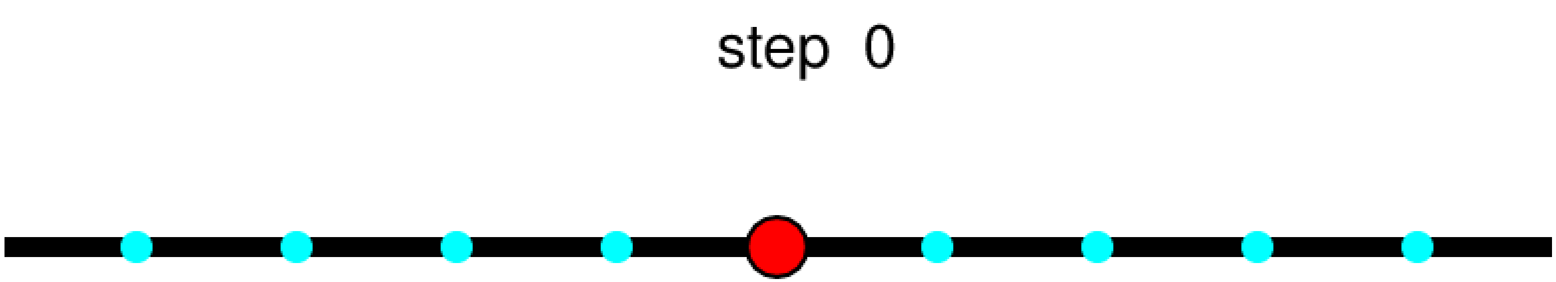
Characterized by a single parameter D (the diffusion constant, or diffusivity).

The mean-square displacement (MSD) is $\langle x^2 \rangle = 2Dt$ and the displacement distribution (DisD) is

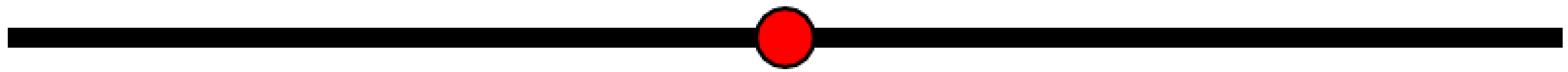
$$P(x,t) = \frac{1}{2\sqrt{\pi D t}} \exp\left(-\frac{x^2}{4Dt}\right)$$

Suppose we want to approximate this by a lattice walk.

Simplest choice: move left or right to a neighbouring lattice site at every step.



Consider a diffusing particle (a random walker). Start with the 1D case.



Characterized by a single parameter D (the diffusion constant, or diffusivity).

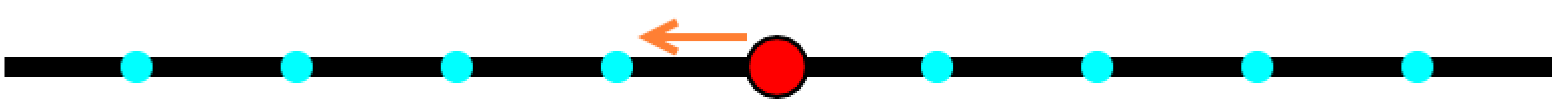
The mean-square displacement (MSD) is $\langle x^2 \rangle = 2Dt$ and the displacement distribution (DisD) is

$$P(x,t) = \frac{1}{2\sqrt{\pi D t}} \exp\left(-\frac{x^2}{4Dt}\right)$$

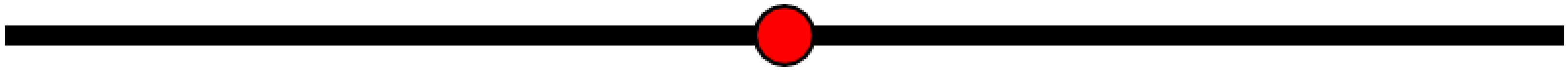
Suppose we want to approximate this by a lattice walk.

Simplest choice: move left or right to a neighbouring lattice site at every step.

step 0



Consider a diffusing particle (a random walker). Start with the 1D case.



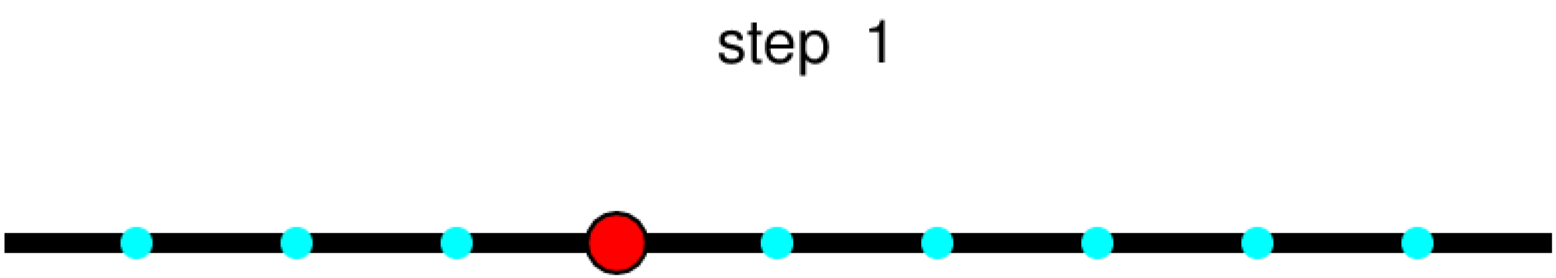
Characterized by a single parameter D (the diffusion constant, or diffusivity).

The mean-square displacement (MSD) is $\langle x^2 \rangle = 2Dt$ and the displacement distribution (DisD) is

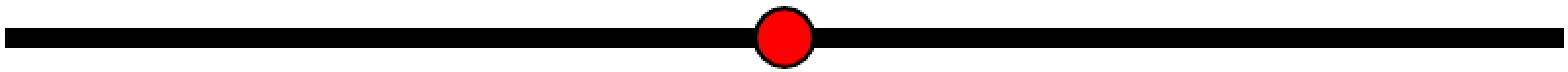
$$P(x,t) = \frac{1}{2\sqrt{\pi D t}} \exp\left(-\frac{x^2}{4Dt}\right)$$

Suppose we want to approximate this by a lattice walk.

Simplest choice: move left or right to a neighbouring lattice site at every step.



Consider a diffusing particle (a random walker). Start with the 1D case.



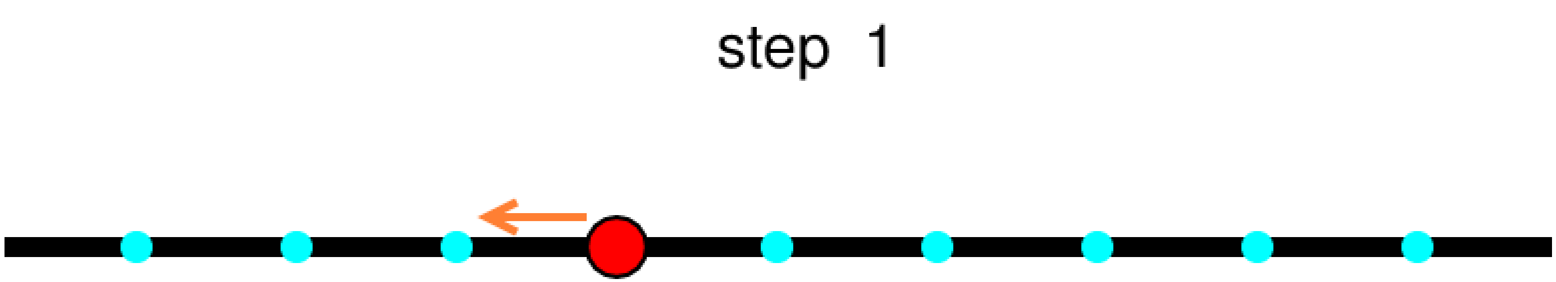
Characterized by a single parameter D (the diffusion constant, or diffusivity).

The mean-square displacement (MSD) is $\langle x^2 \rangle = 2\,D\,t$ and the displacement distribution (DisD) is

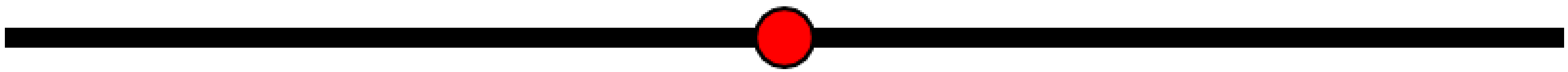
$$P(x,t) = \frac{1}{2\sqrt{\pi D t}} \exp\left(-\frac{x^2}{4Dt}\right)$$

Suppose we want to approximate this by a lattice walk.

Simplest choice: move left or right to a neighbouring lattice site at every step.



Consider a diffusing particle (a random walker). Start with the 1D case.



Characterized by a single parameter D (the diffusion constant, or diffusivity).

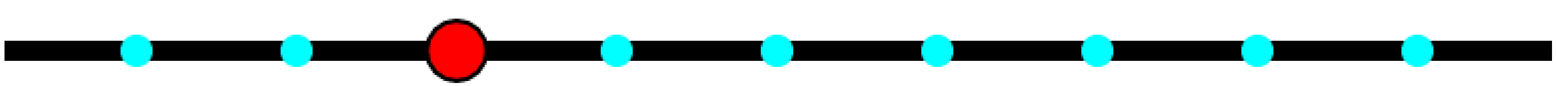
The mean-square displacement (MSD) is $\langle x^2 \rangle = 2Dt$ and the displacement distribution (DisD) is

$$P(x,t) = \frac{1}{2\sqrt{\pi D t}} \exp\left(-\frac{x^2}{4Dt}\right)$$

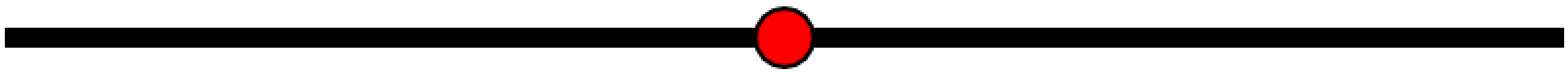
Suppose we want to approximate this by a lattice walk.

Simplest choice: move left or right to a neighbouring lattice site at every step.

step 2



Consider a diffusing particle (a random walker). Start with the 1D case.



Characterized by a single parameter D (the diffusion constant, or diffusivity).

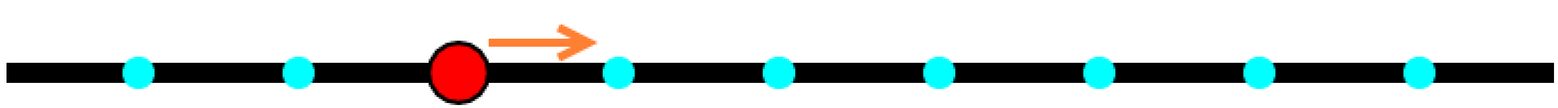
The mean-square displacement (MSD) is $\langle x^2 \rangle = 2Dt$ and the displacement distribution (DisD) is

$$P(x,t) = \frac{1}{2\sqrt{\pi D t}} \exp\left(-\frac{x^2}{4Dt}\right)$$

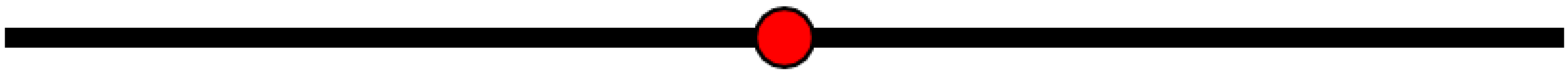
Suppose we want to approximate this by a lattice walk.

Simplest choice: move left or right to a neighbouring lattice site at every step.

step 2



Consider a diffusing particle (a random walker). Start with the 1D case.



Characterized by a single parameter D (the diffusion constant, or diffusivity).

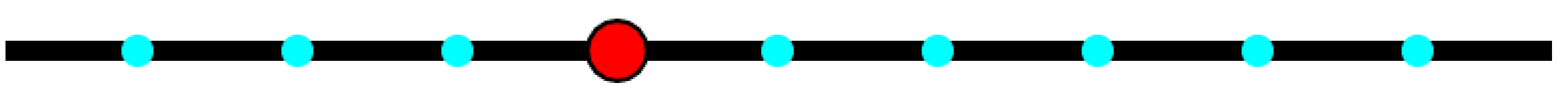
The mean-square displacement (MSD) is $\langle x^2 \rangle = 2Dt$ and the displacement distribution (DisD) is

$$P(x,t) = \frac{1}{2\sqrt{\pi D t}} \exp\left(-\frac{x^2}{4Dt}\right)$$

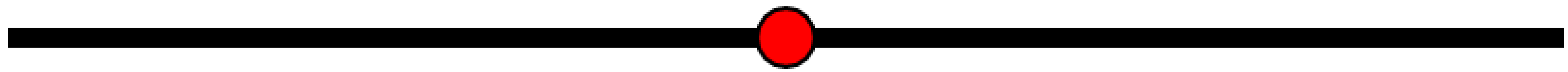
Suppose we want to approximate this by a lattice walk.

Simplest choice: move left or right to a neighbouring lattice site at every step.

step 3



Consider a diffusing particle (a random walker). Start with the 1D case.



Characterized by a single parameter D (the diffusion constant, or diffusivity).

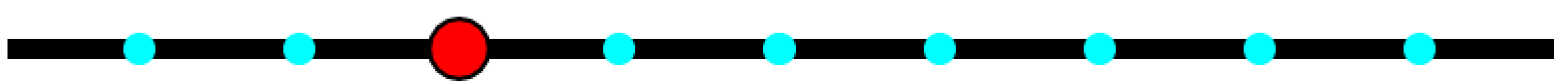
The mean-square displacement (MSD) is $\langle x^2 \rangle = 2Dt$ and the displacement distribution (DisD) is

$$P(x,t) = \frac{1}{2\sqrt{\pi D t}} \exp\left(-\frac{x^2}{4Dt} \right)$$

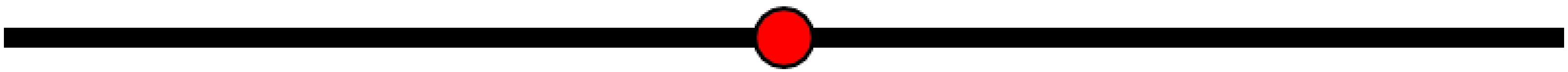
Suppose we want to approximate this by a lattice walk.

Simplest choice: move left or right to a neighbouring lattice site at every step.

step 4



Consider a diffusing particle (a random walker). Start with the 1D case.



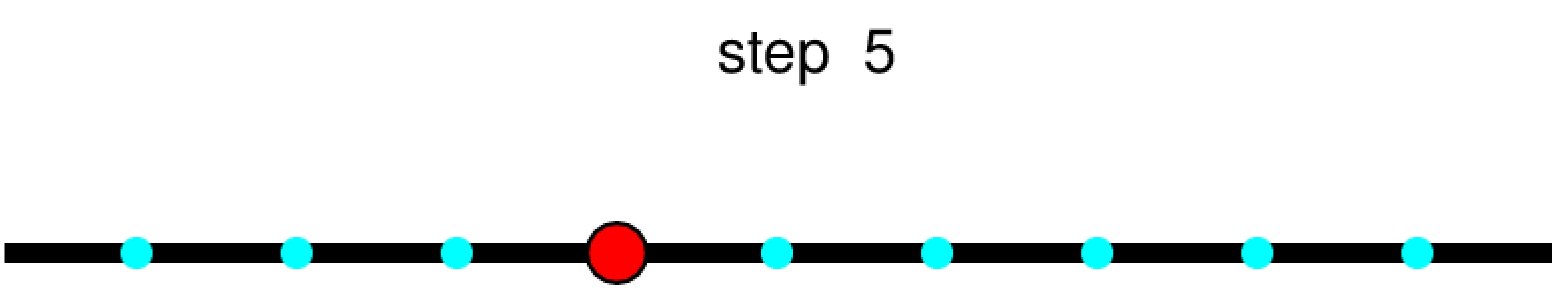
Characterized by a single parameter D (the diffusion constant, or diffusivity).

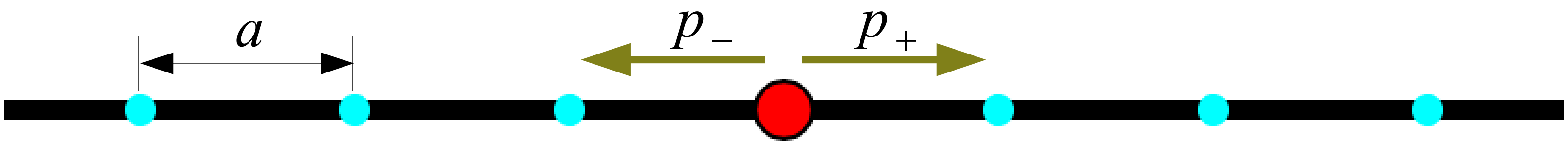
The mean-square displacement (MSD) is $\langle x^2 \rangle = 2Dt$ and the displacement distribution (DisD) is

$$P(x,t) = \frac{1}{2\sqrt{\pi D t}} \exp\left(-\frac{x^2}{4Dt}\right)$$

Suppose we want to approximate this by a lattice walk.

Simplest choice: move left or right to a neighbouring lattice site at every step.

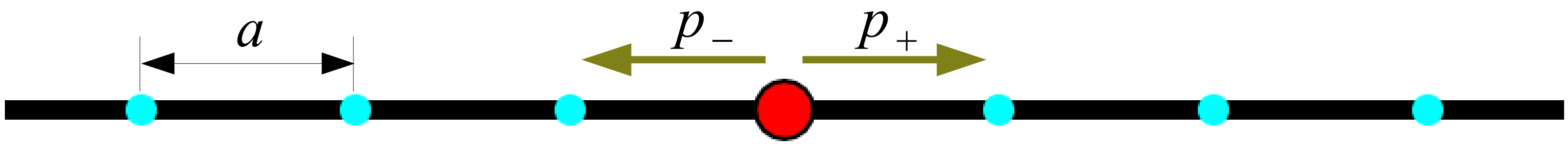




$$\langle x^2 \rangle = \sum_{i=1}^{N} \langle \Delta x_i^2 \rangle = \sum_{i=1}^{N} a^2 (p_- + p_+) = Na^2 = \frac{a^2}{\tau} t$$

$N = \frac{t}{\tau}$ $p_- = p_+ = 1/2$ Need $\langle x^2 \rangle = 2Dt \Rightarrow$ time step $\tau = \frac{a^2}{2D}$

Gives the correct MSD at all times.

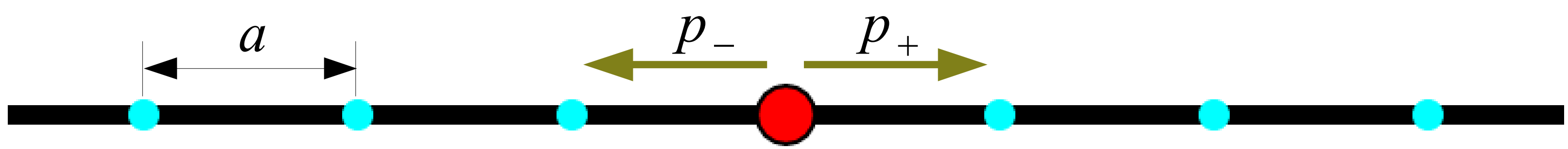


$$\langle x^2 \rangle = \sum_{i=1}^{N} \langle \Delta x_i^2 \rangle = \sum_{i=1}^{N} a^2 (p_- + p_+) = Na^2 = \frac{a^2}{\tau} t$$

$N = \frac{t}{\tau}$ $\quad p_- = p_+ = 1/2 \quad$ Need $\langle x^2 \rangle = 2Dt \Rightarrow$ time step $\tau = \frac{a^2}{2D}$

Gives the correct MSD at all times.

What if the particle “skips” some steps and stays put?



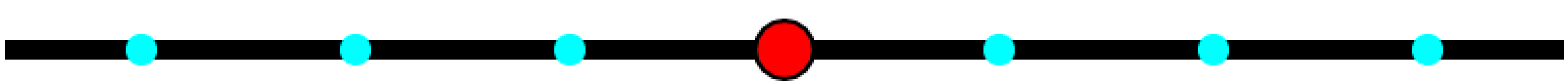
$$\langle x^2\rangle = \sum_{i=1}^{N} \langle \Delta x_i^2\rangle = \sum_{i=1}^{N} a^2(p_- + p_+) = Na^2 = \frac{a^2}{\tau} t$$

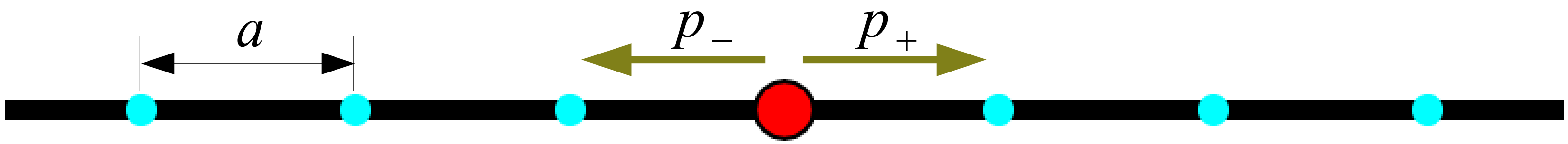
$N = \dfrac{t}{\tau}$ $\quad p_- = p_+ = 1/2 \quad$ Need $\langle x^2\rangle = 2Dt \Rightarrow$ time step $\tau = \dfrac{a^2}{2D}$

Gives the correct MSD at all times.

What if the particle “skips” some steps and stays put?

step 0



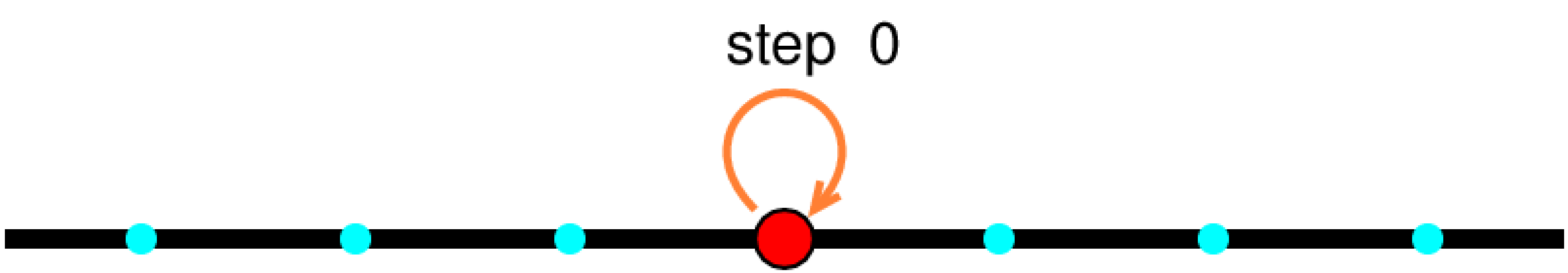


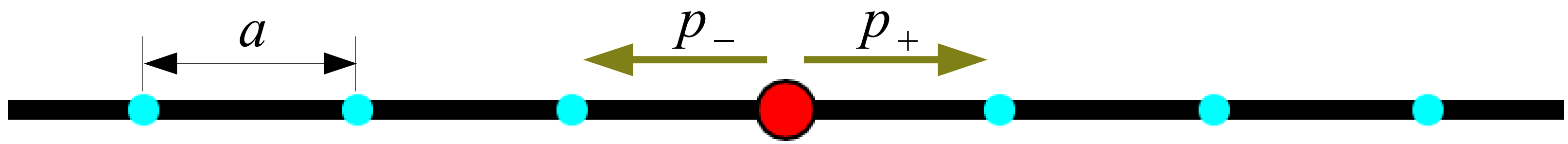
$$\langle x^2\rangle = \sum_{i=1}^{N} \langle \Delta x_i^2\rangle = \sum_{i=1}^{N} a^2 (p_- + p_+) = Na^2 = \frac{a^2}{\tau} t$$

$N = \frac{t}{\tau}$ $\quad p_- = p_+ = 1/2 \quad$ Need $\langle x^2\rangle = 2Dt \Rightarrow$ time step $\tau = \frac{a^2}{2D}$

Gives the correct MSD at all times.

What if the particle "skips" some steps and stays put?



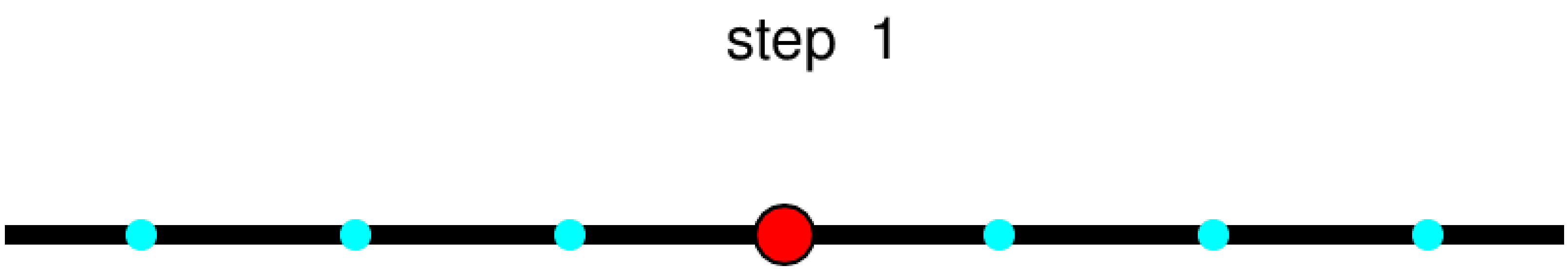


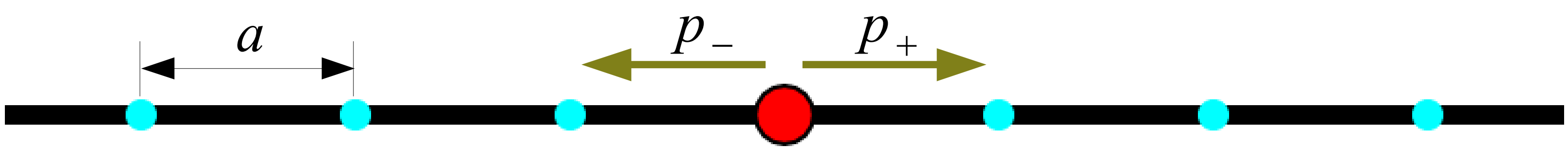
$$\langle x^2 \rangle = \sum_{i=1}^{N} \langle \Delta x_i^2 \rangle = \sum_{i=1}^{N} a^2 (p_- + p_+) = N a^2 = \frac{a^2}{\tau} t$$

$N = \frac{t}{\tau}$ $\quad p_- = p_+ = 1/2 \quad$ Need $\langle x^2 \rangle = 2Dt \Rightarrow$ time step $\tau = \frac{a^2}{2D}$

Gives the correct MSD at all times.

What if the particle “skips” some steps and stays put?





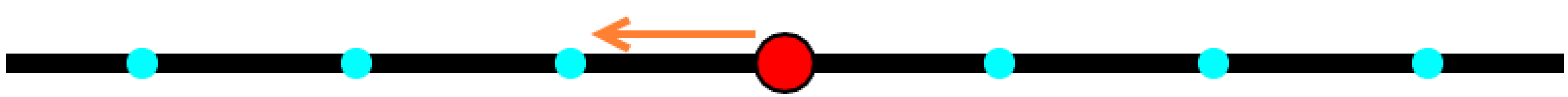
$$\langle x^2 \rangle = \sum_{i=1}^{N} \langle \Delta x_i^2 \rangle = \sum_{i=1}^{N} a^2 (p_- + p_+) = N a^2 = \frac{a^2}{\tau} t$$

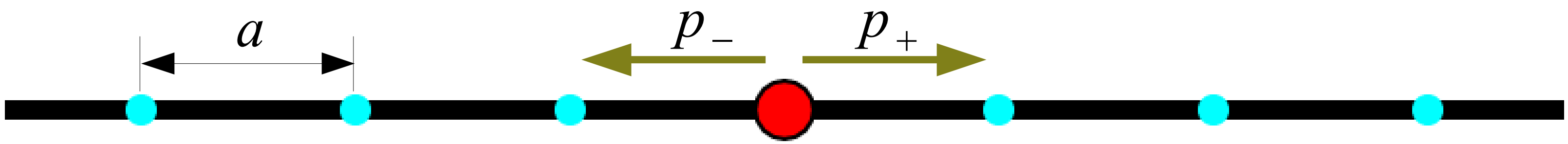
$N = \frac{t}{\tau}$ $\quad p_- = p_+ = 1/2 \quad$ Need $\langle x^2 \rangle = 2Dt \Rightarrow$ time step $\tau = \frac{a^2}{2D}$

Gives the correct MSD at all times.

What if the particle “skips” some steps and stays put?

step 1





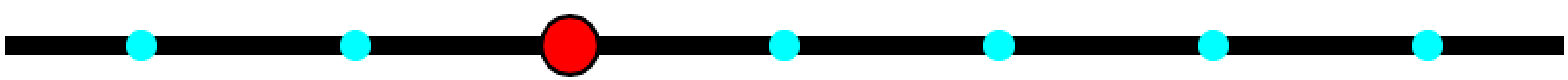
$$\langle x^2 \rangle = \sum_{i=1}^{N} \langle \Delta x_i^2 \rangle = \sum_{i=1}^{N} a^2 (p_- + p_+) = N a^2 = \frac{a^2}{\tau} t$$

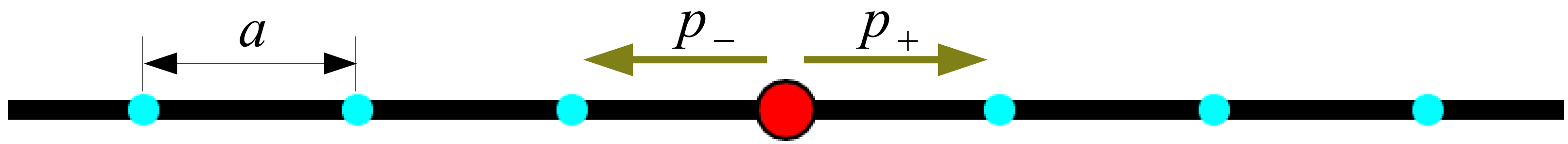
$N = \frac{t}{\tau}$ $\quad p_- = p_+ = 1/2 \quad$ Need $\langle x^2 \rangle = 2Dt \Rightarrow$ time step $\tau = \frac{a^2}{2D}$

Gives the correct MSD at all times.

What if the particle “skips” some steps and stays put?

step 2



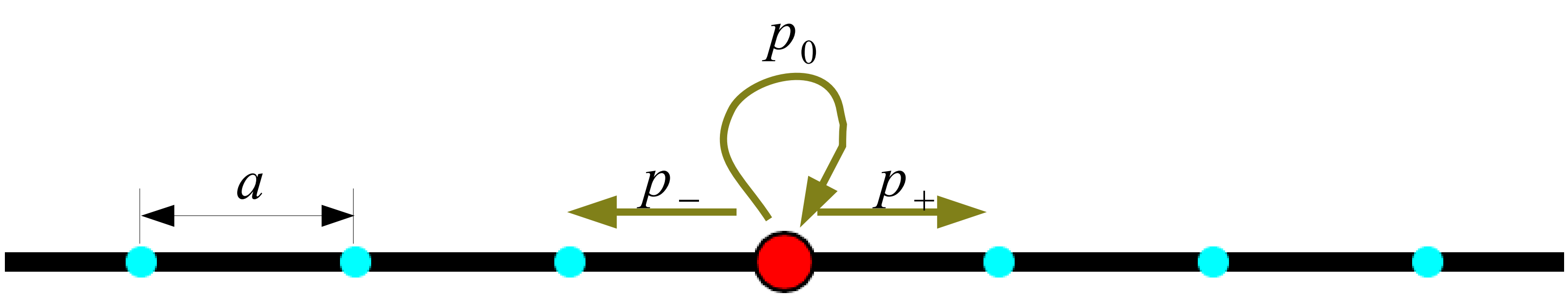


$$\langle x^2\rangle = \sum_{i=1}^{N} \langle \Delta x_i^2\rangle = \sum_{i=1}^{N} a^2 (p_- + p_+) = Na^2 = \frac{a^2}{\tau} t$$

$N = \frac{t}{\tau}$ $\quad p_- = p_+ = 1/2 \quad$ Need $\langle x^2\rangle = 2Dt \Rightarrow$ time step $\tau = \frac{a^2}{2D}$

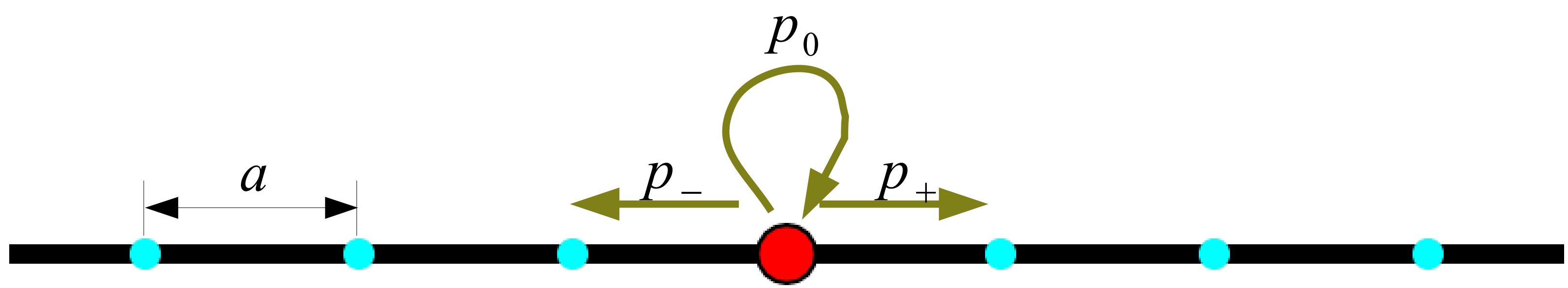
Gives the correct MSD at all times.

What if the particle “skips” some steps and stays put?



$$\langle x^2\rangle = \sum_{i=1}^{N} \langle \Delta x_i^2\rangle = \sum_{i=1}^{N} a^2 (p_- + p_+) + 0 \times p_0 = Na^2 (p_+ + p_-) = \frac{a^2}{\tau}(1 - p_0) t$$

$$\tau = \frac{a^2}{2D}(1 - p_0)$$



$$\tau = \frac{a^2}{2D}(1 - p_0)$$

Why do this? Wasteful at first glance. The larger τ, the more efficient $\Rightarrow p_0 = 0$.

Can this make the simulation more accurate?

For instance, look at the 4th moment of the displacement distribution. After 1 step,

$\langle x^{2m} \rangle = a^{2m}(p_- + p_+) = a^{2m}(1 - p_0)$. So, regardless of the value of p_0, $\frac{\langle x^4 \rangle}{\langle x^2 \rangle} = a^2$.

On the other hand, in the continuum case,

$$P(x,t) = \frac{1}{2\sqrt{\pi D t}} \exp\left(-\frac{x^2}{4Dt}\right) \Rightarrow \langle x^4 \rangle = 12 D^2 t^2 \Rightarrow \frac{\langle x^4 \rangle}{\langle x^2 \rangle} = 6Dt.$$

$$6D\tau = a^2 \Rightarrow \tau = \frac{a^2}{6D} \Rightarrow p_0 = \frac{2}{3}.$$

The only way to ensure both 2nd and 4th moments are correct after 1 step is to move every 3rd step on average.

After N steps,

$$\langle x^4\rangle = a^4[(3N^2-3N)(1-p_0)^2+N(1-p_0)] = \frac{3a^4}{\tau^2}(1-p_0)^2 t^2 + \frac{a^4}{\tau}(1-p_0)[1-3(1-p_0)]t$$

Using $\tau = \frac{a^2}{2D}(1-p_0)$, $\langle x^4\rangle = 12D^2t^2 + 2Da^2[1-3(1-p_0)]t$.

Correct at large t for any p_0, but at any t only for $p_0 = 2/3$.

Sixth moment. Continuum: $\langle x^6\rangle = 120D^3t^3$. Lattice MC:

$$\langle x^6\rangle = 120D^3t^3 + 60D^2a^2[1-3(1-p_0)]t^2 + 2Da^4[1-15(1-p_0)+30(1-p_0)^2]t.$$

Choosing $p_0 = 2/3$, $\langle x^6\rangle = 120D^3t^3 - (2/3)Da^4t$. Error $O(t)$, instead of $O(t^2)$.

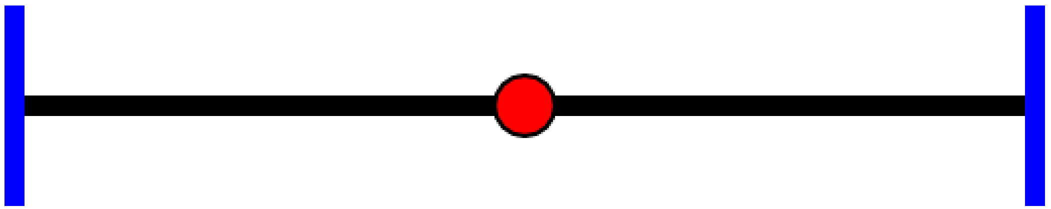
In general, continuum $\langle x^{2m}\rangle = (2m-1)!!\,(2Dt)^m$. Lattice MC:

$$\langle x^{2m}\rangle = (2m-1)!!\,(2Dt)^m + \frac{(2m-1)!!\; m(m-1)a^2}{6}[1-3(1-p_0)](2Dt)^{m-1} + O(t^{m-2}).$$

Choosing $p_0 = 2/3$ makes the 4th moment of the displacement distribution correct at all times and minimizes the error in higher moments.

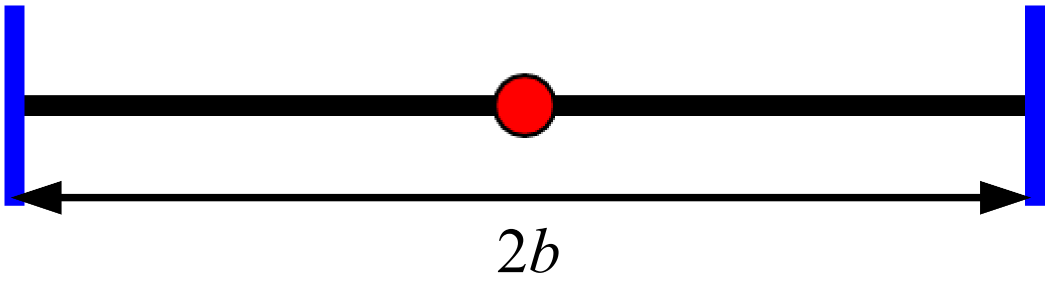
The first-passage-time problem

Start in the middle. How long does it take to reach one of the boundaries?



The first-passage-time problem

Start in the middle. How long does it take to reach one of the boundaries?



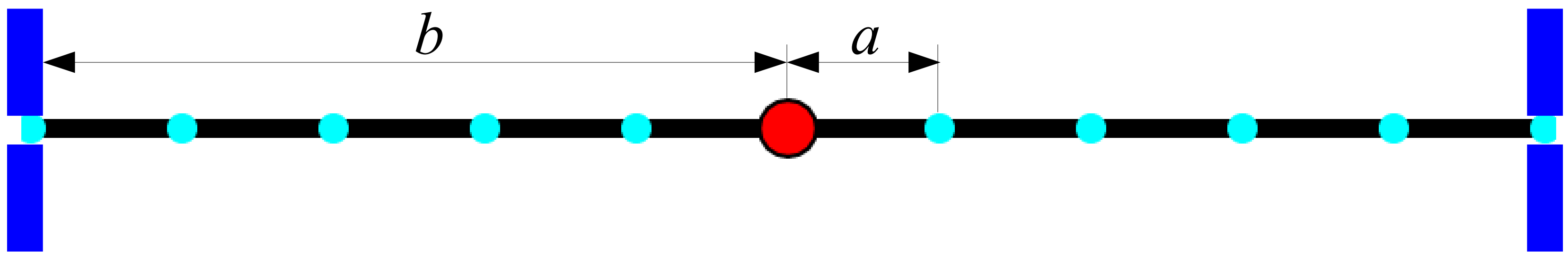
Moments of the first-passage-time distribution:

mean first-passage time (MFPT) $\langle t_1 \rangle = \dfrac{b^2}{2D}$;

mean-square first-passage time (MSFPT) $\langle t_1^2 \rangle = \dfrac{5b^4}{12D^2}$;

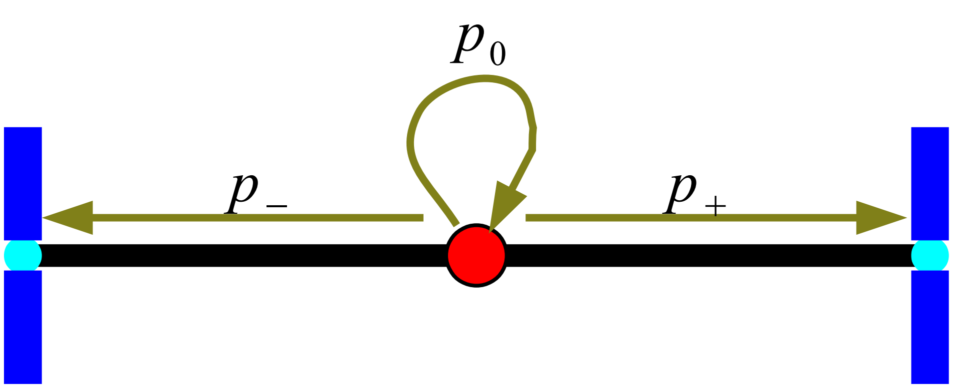
variance $\langle t_1^2 \rangle - \langle t_1 \rangle^2 = \dfrac{b^4}{6D^2} > 0$

The lattice analogue:



Walls coincide with lattice sites. $b = Ma$; $M = 5$ in the figure.

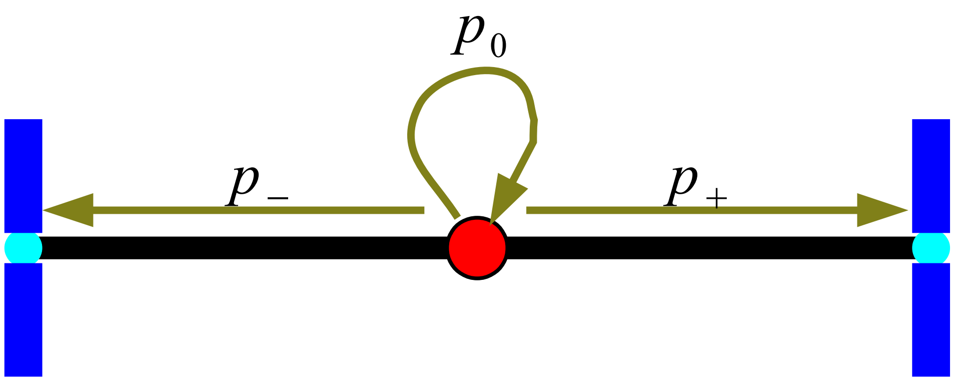
Consider $M = 1$ $(a = b)$:



When $p_0 = 0$, always reaches in one step – deterministic FPT = $\tau = a^2/2D$.

Thus the mean $\langle t_1 \rangle = a^2/2D$ is correct. But the variance is obviously zero.

$$\langle t_1^2 \rangle = \langle t_1 \rangle^2 = a^4/4D^2 \text{ instead of } 5a^4/12D^2 .$$



$$\langle \tau \rangle = \frac{a^2}{2D}(1-p_0)$$

Continuum

$$\langle t_1 \rangle = \frac{a^2}{2D}$$

$$\langle t_1^2 \rangle = \frac{5a^4}{12D^2}$$

When $p_0 \neq 0$, no longer deterministic.

Reaches in 1 step with probability $1-p_0$, in 2 steps with prob. $p_0(1-p_0)$,

in 3 steps with prob. $p_0^2(1-p_0)$, etc.

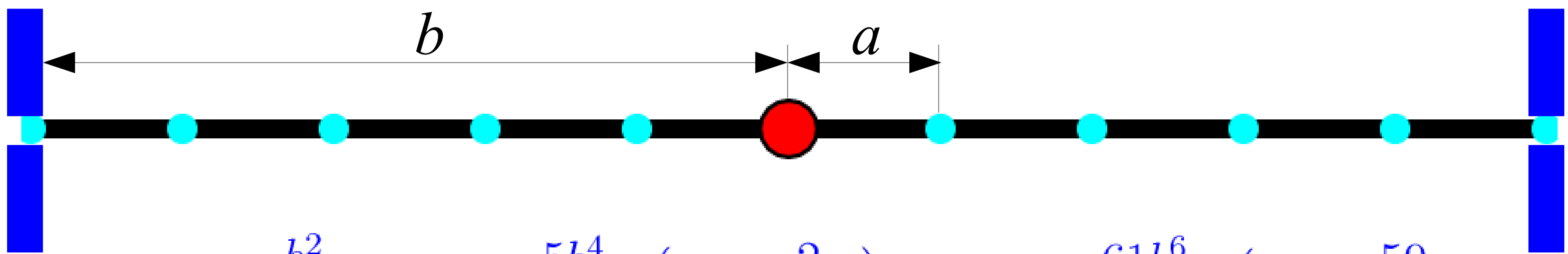
The mean number of steps is $\langle N \rangle = \dfrac{1}{1-p_0} \Rightarrow \langle t_1 \rangle = \langle N \rangle \tau = \dfrac{a^2}{2D}$.

The mean-square number of steps is $\langle N^2 \rangle = \dfrac{1+p_0}{(1-p_0)^2} \Rightarrow \langle t_1^2 \rangle = \langle N^2 \rangle \tau^2 = \dfrac{a^4}{4D^2}(1+p_0)$.

MFPT is always correct. MSFPT is only correct when $p_0 = 2/3$.

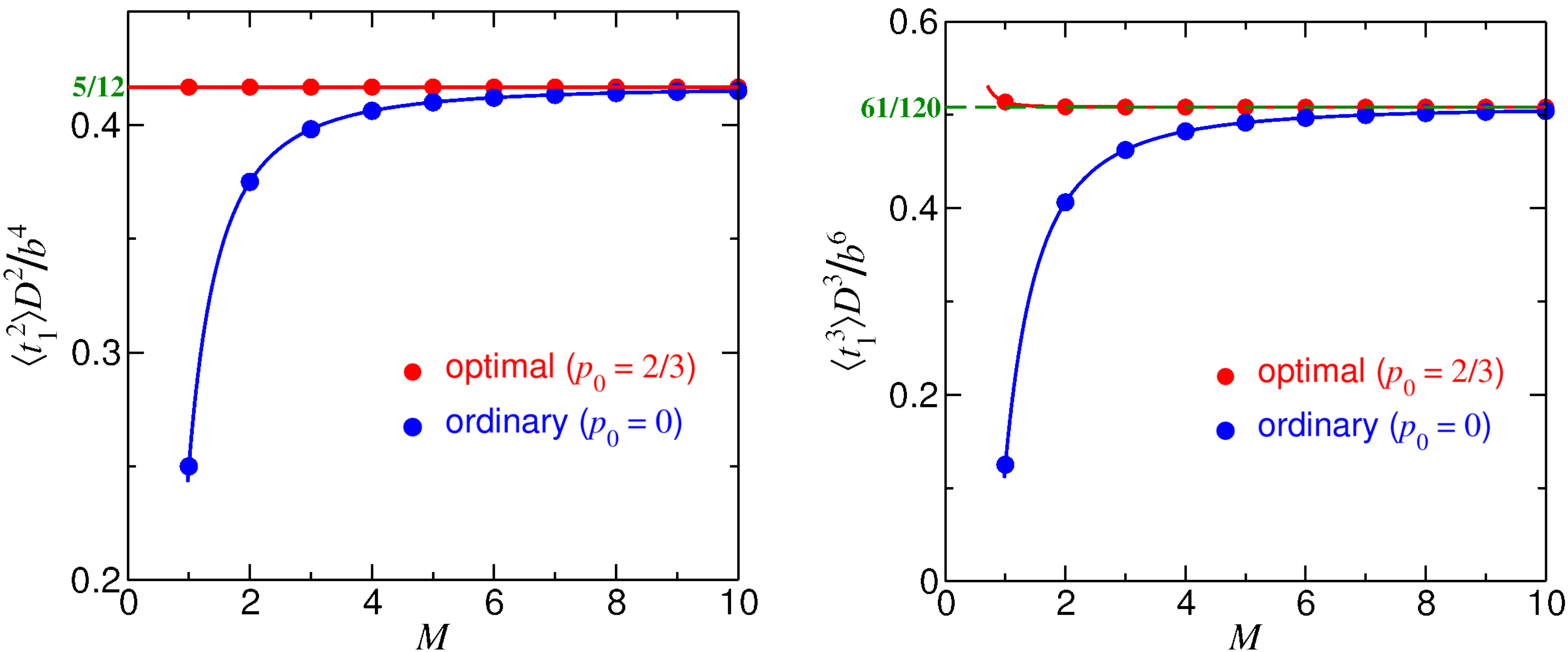
Same as the optimal value for the moments of the displacement distribution.

For arbitrary $M = b/a$:



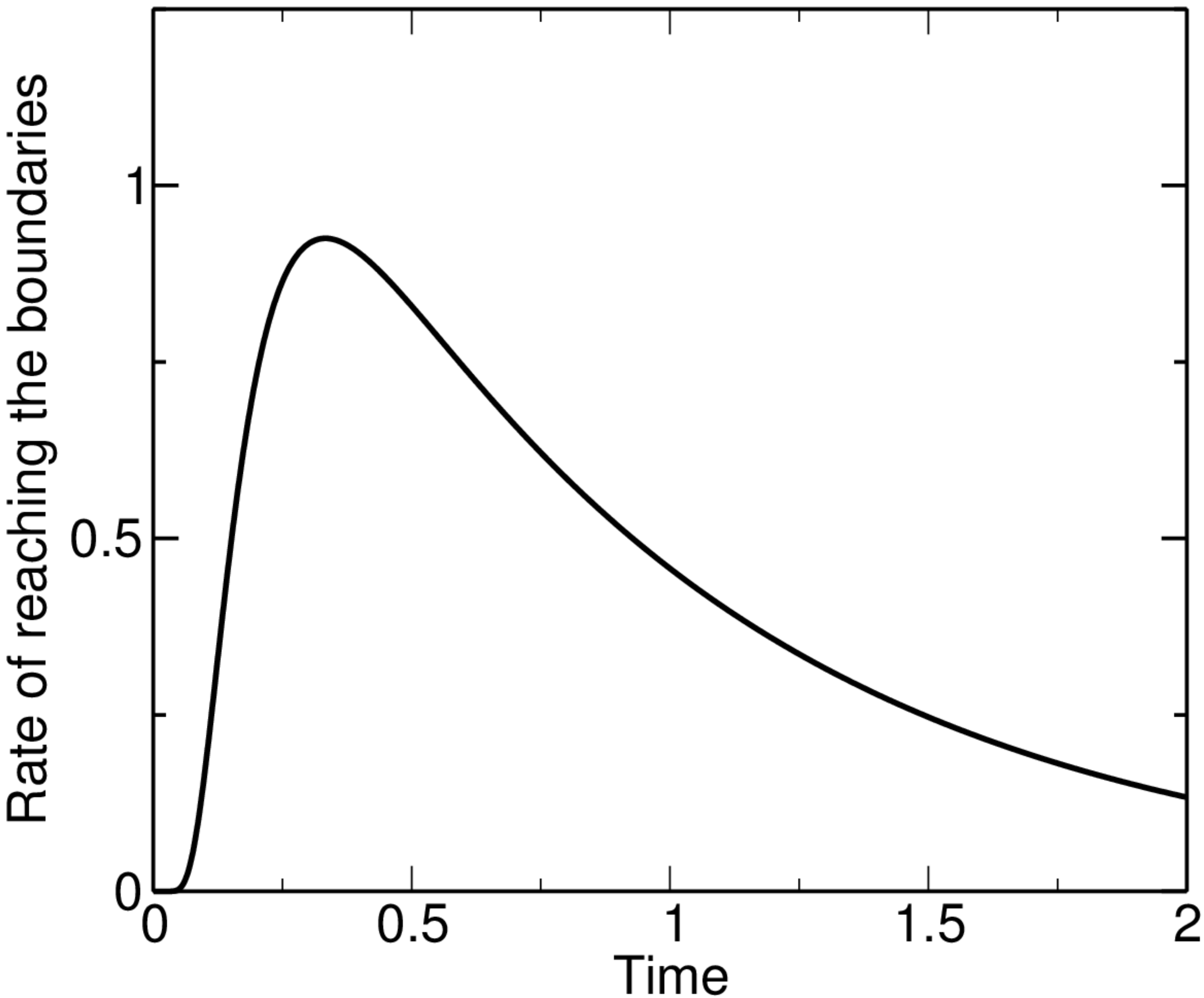
$$p_0 = 0: \quad \langle t_1 \rangle = \frac{b^2}{2D}; \; \langle t_1^2 \rangle = \frac{5b^4}{12D^2}\left(1 - \frac{2}{5M^2}\right); \; \langle t_1^3 \rangle = \frac{61b^6}{120D^3}\left(1 - \frac{50}{61M^2} + \frac{4}{61M^4}\right)$$

$$p_0 = 2/3: \quad \langle t_1 \rangle = \frac{b^2}{2D}; \; \langle t_1^2 \rangle = \frac{5b^4}{12D^2}; \qquad \langle t_1^3 \rangle = \frac{61b^6}{120D^3}\left(1 + \frac{2}{183M^4}\right)$$



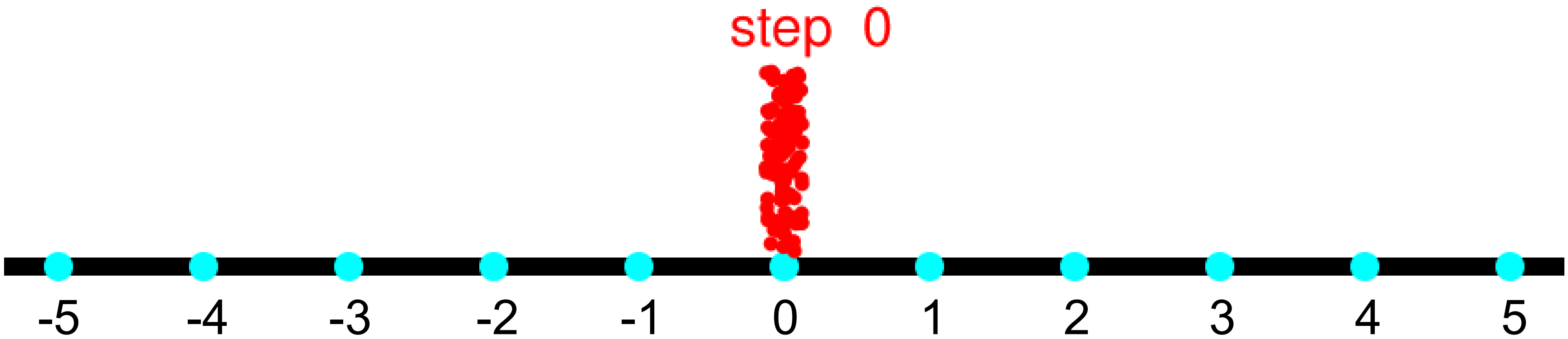
Full first-passage-time distribution

Continuum; $b = 1$, $D = 1/2$.



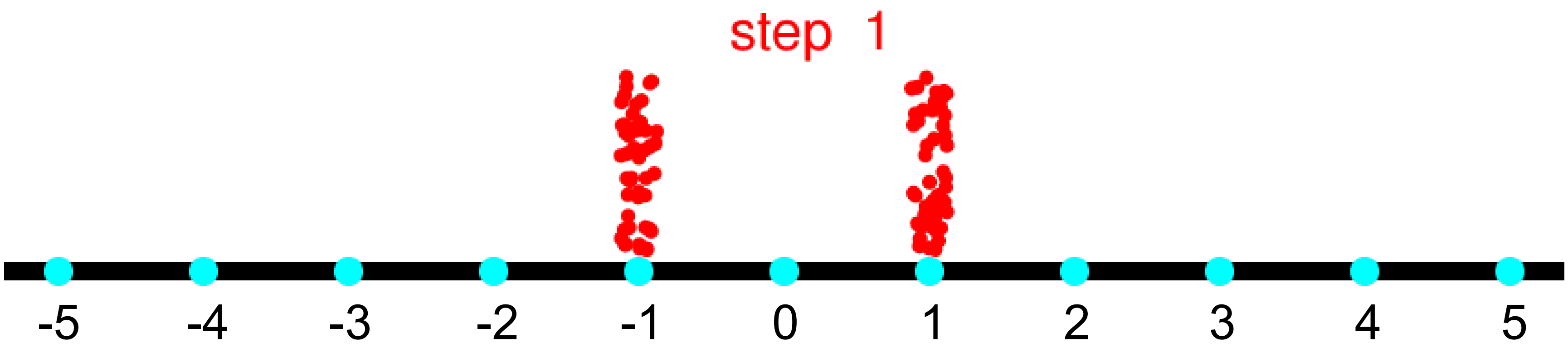
Full first-passage-time distribution

For the ordinary algorithm ($p_0 = 0$), even displacements at even steps and odd displacements at odd steps:



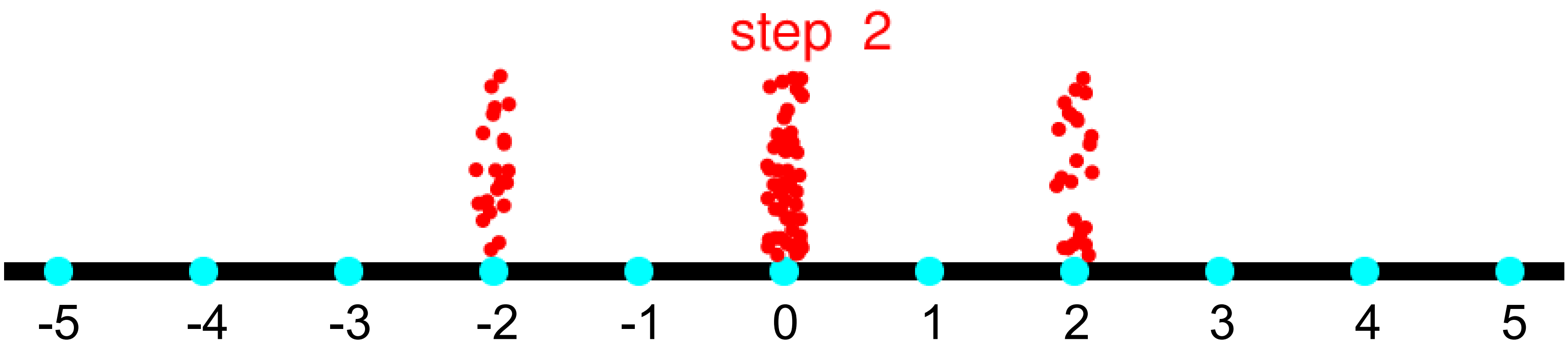
Full first-passage-time distribution

For the ordinary algorithm ($p_0 = 0$), even displacements at even steps and odd displacements at odd steps:



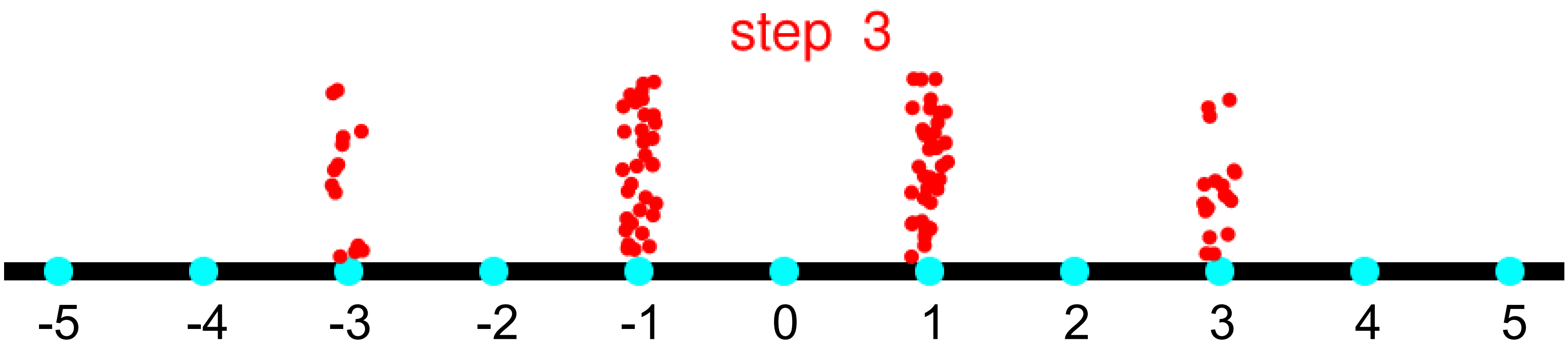
Full first-passage-time distribution

For the ordinary algorithm ($p_0 = 0$), even displacements at even steps and odd displacements at odd steps:



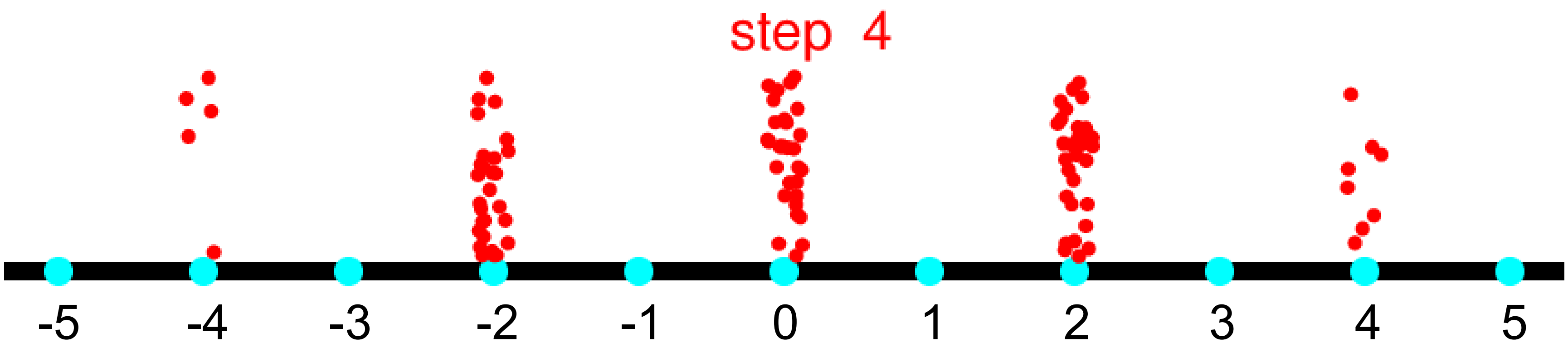
Full first-passage-time distribution

For the ordinary algorithm ($p_0 = 0$), even displacements at even steps and odd displacements at odd steps:



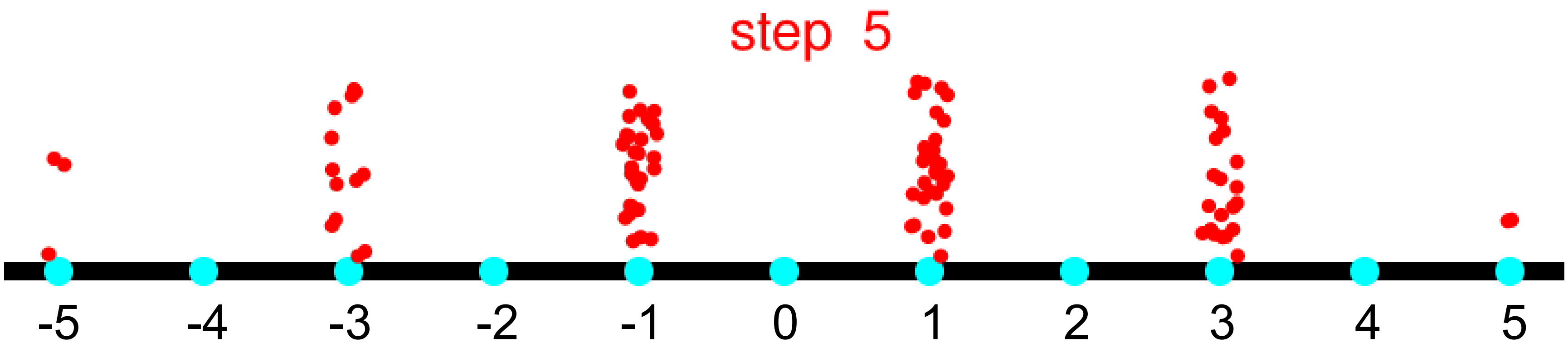
Full first-passage-time distribution

For the ordinary algorithm ($p_0 = 0$), even displacements at even steps and odd displacements at odd steps:



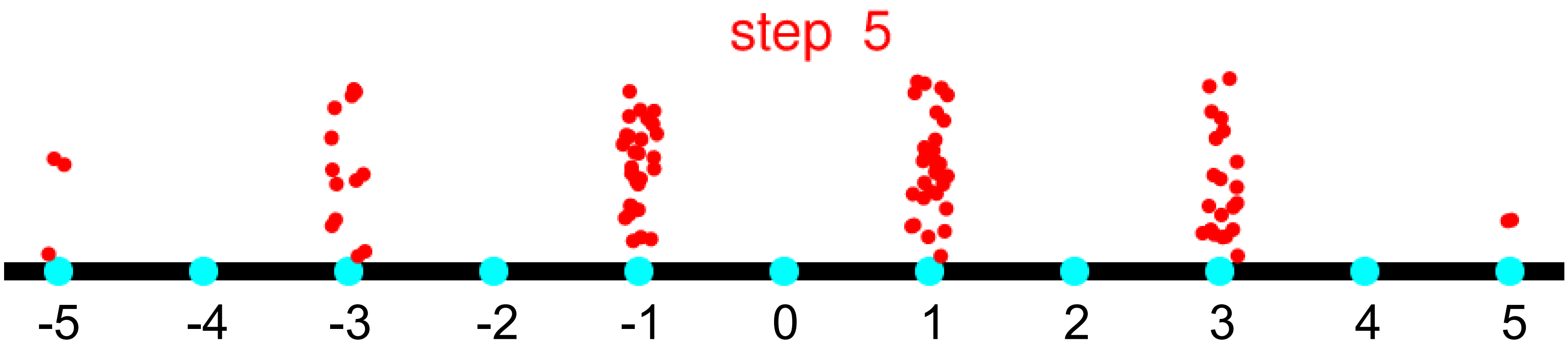
Full first-passage-time distribution

For the ordinary algorithm ($p_0 = 0$), even displacements at even steps and odd displacements at odd steps:



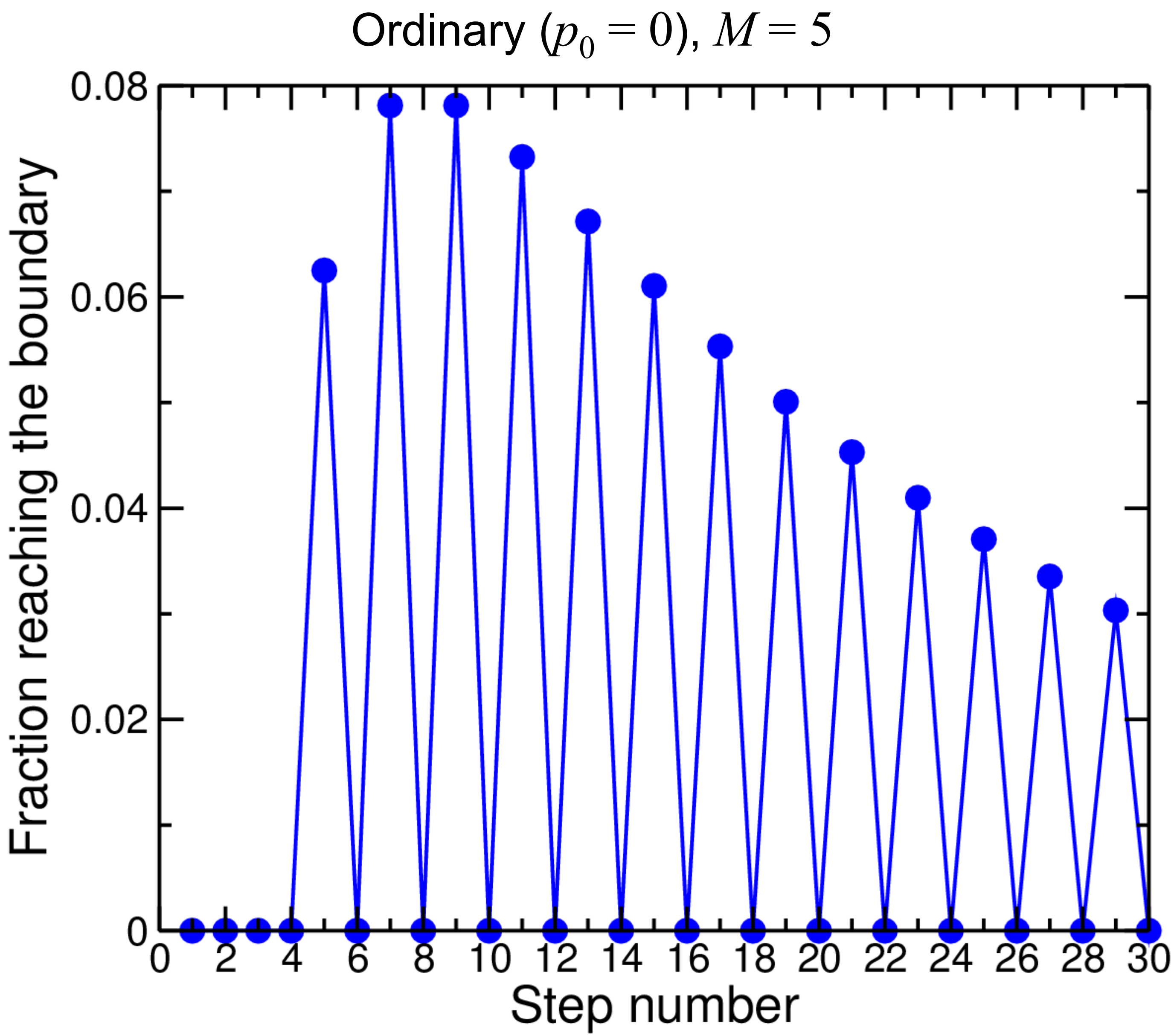
Full first-passage-time distribution

For the ordinary algorithm ($p_0 = 0$), even displacements at even steps and odd displacements at odd steps:



Thus, for the first-passage problem, when the distance M between the initial position and the wall is even, particles can only reach the wall at even steps, and vice versa.

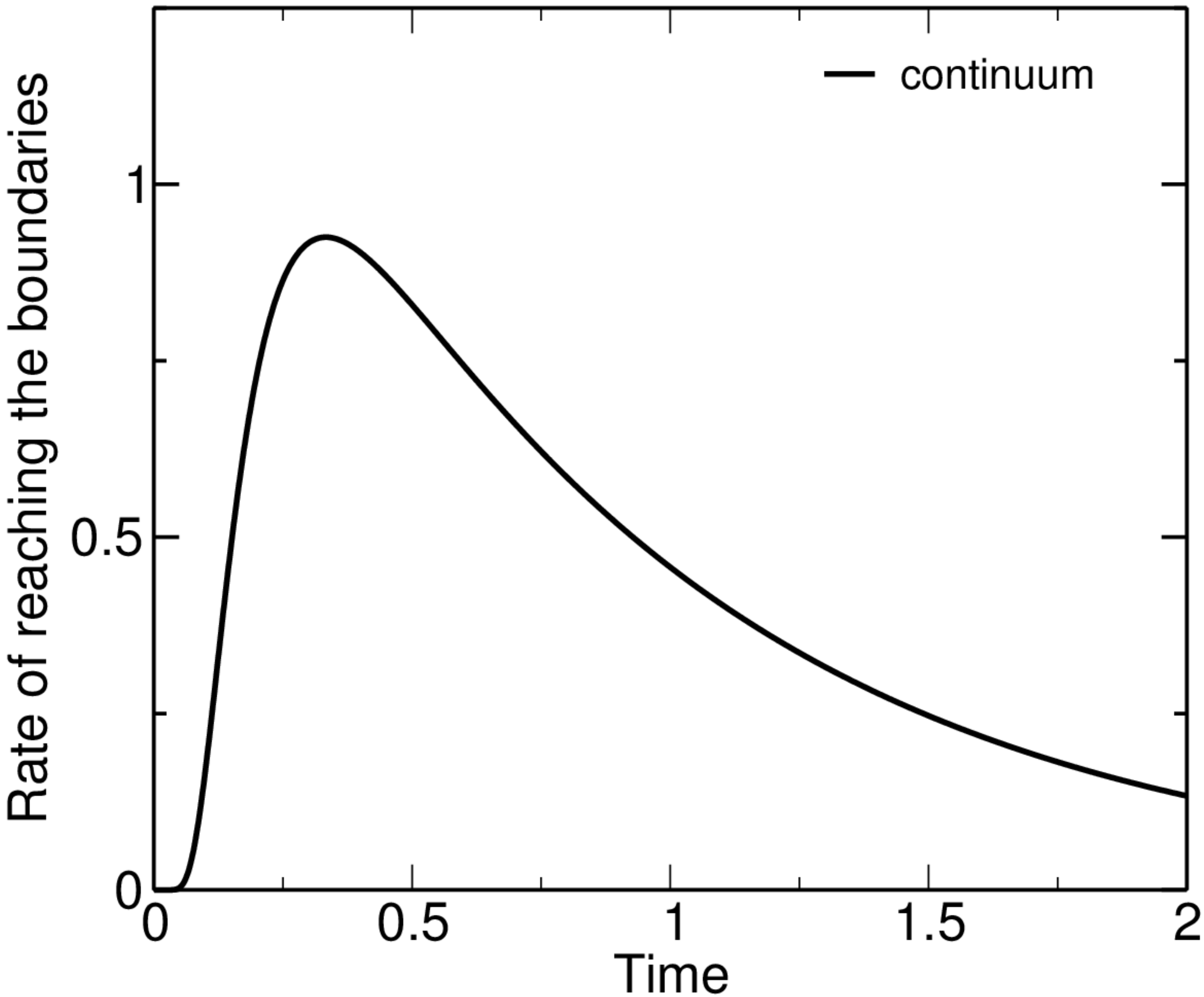
Full first-passage-time distribution



Not the case when $p_0 \neq 0$. For comparison, plot only nonzero values, but divide by 2.

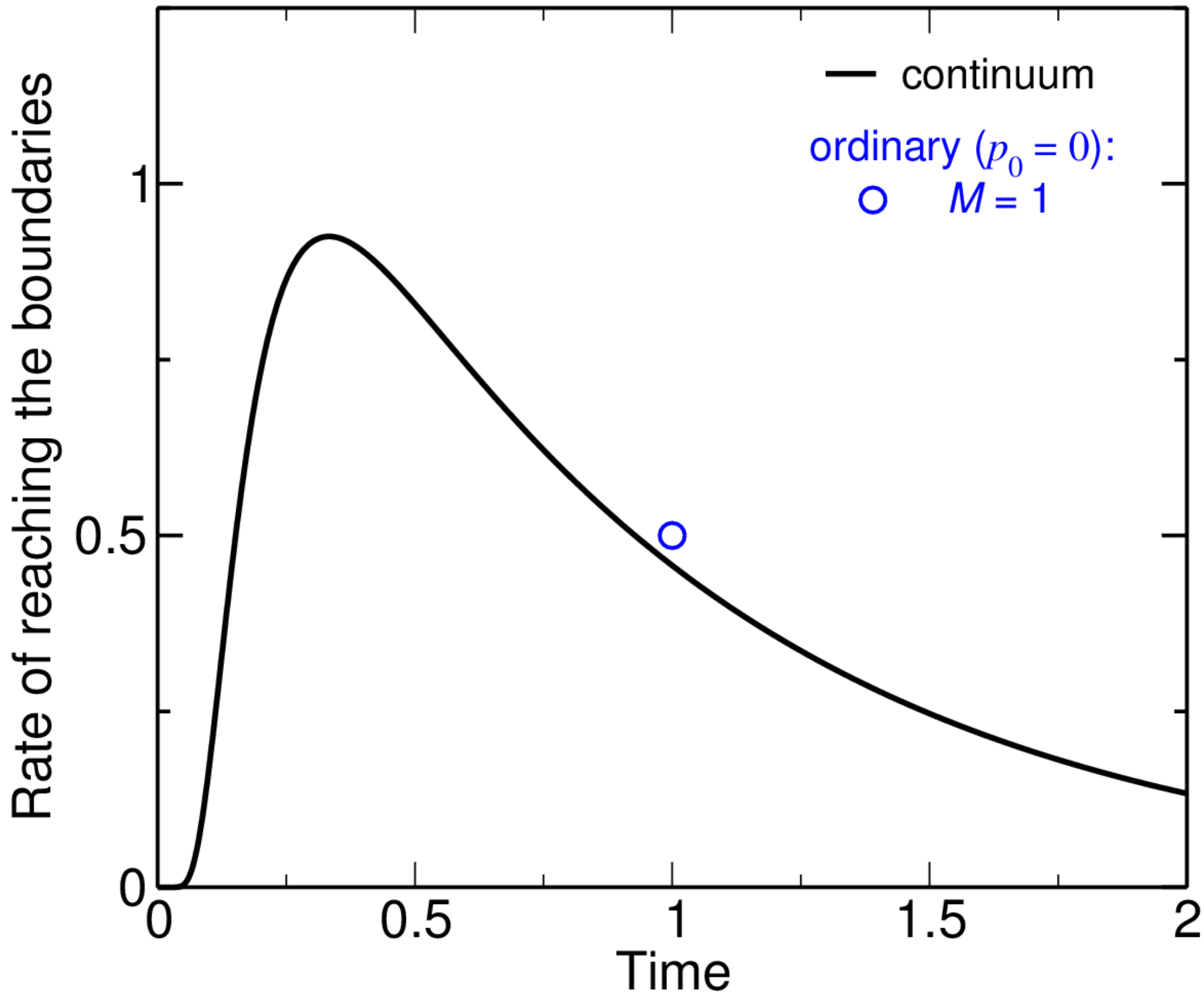
Full first-passage-time distribution

$b = 1, D = 1/2$



Full first-passage-time distribution

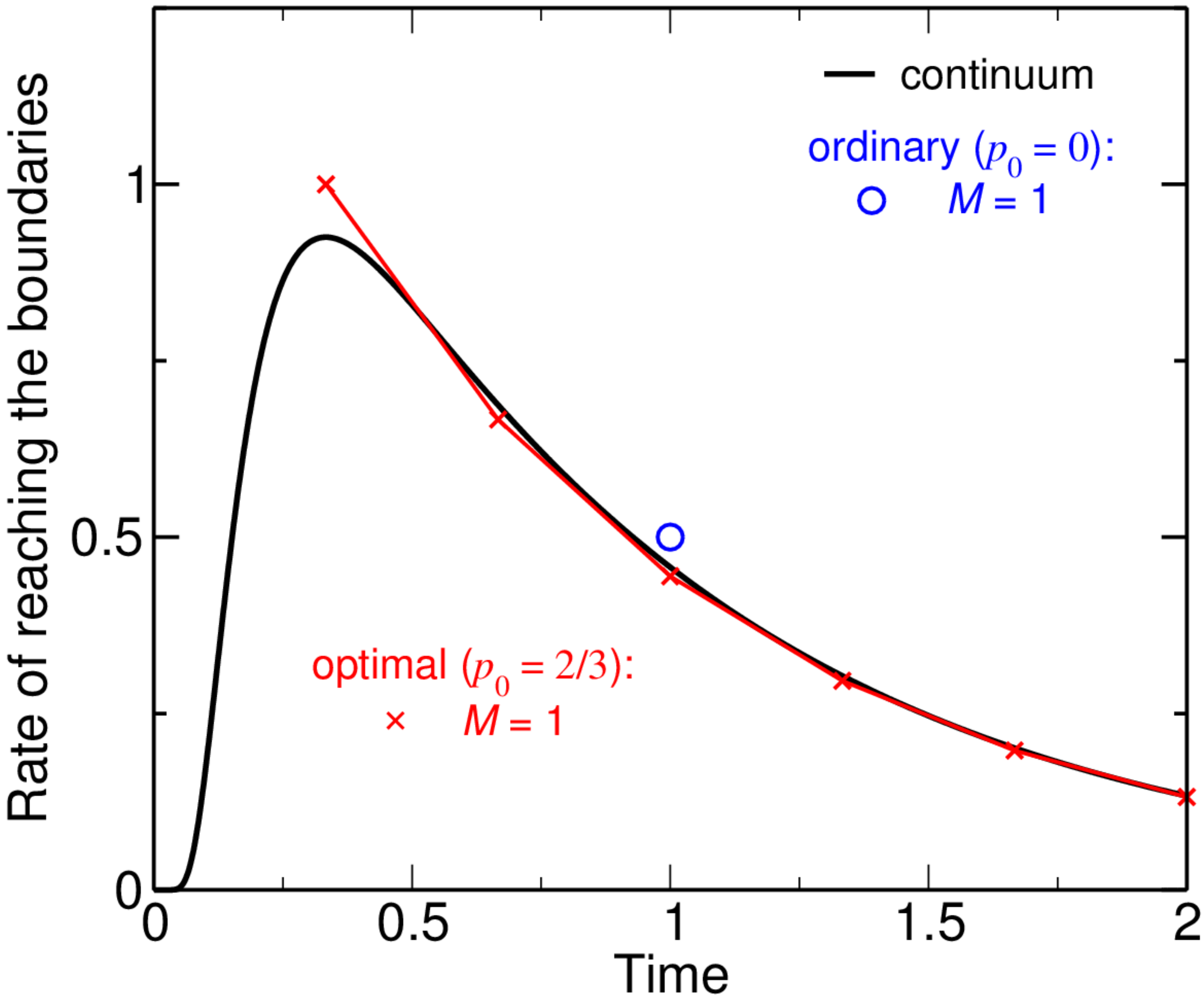
$b = 1, D = 1/2$



Ordinary algorithm, $M = 1$: deterministic FPT. Only one nonzero value.

Full first-passage-time distribution

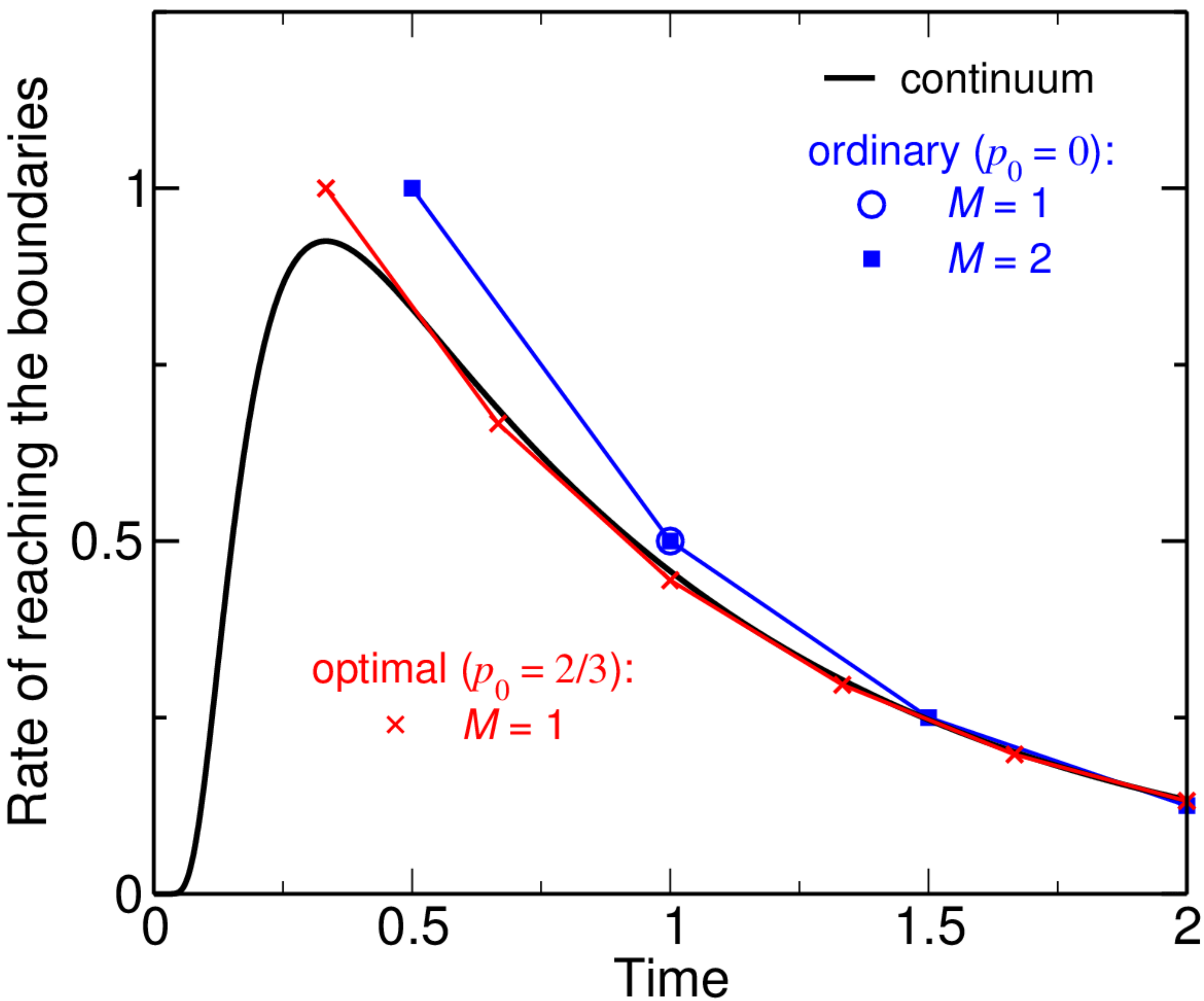
$b = 1, D = 1/2$



Optimal algorithm, $M = 1$: exponential decay. Rate very close to continuum one.

Full first-passage-time distribution

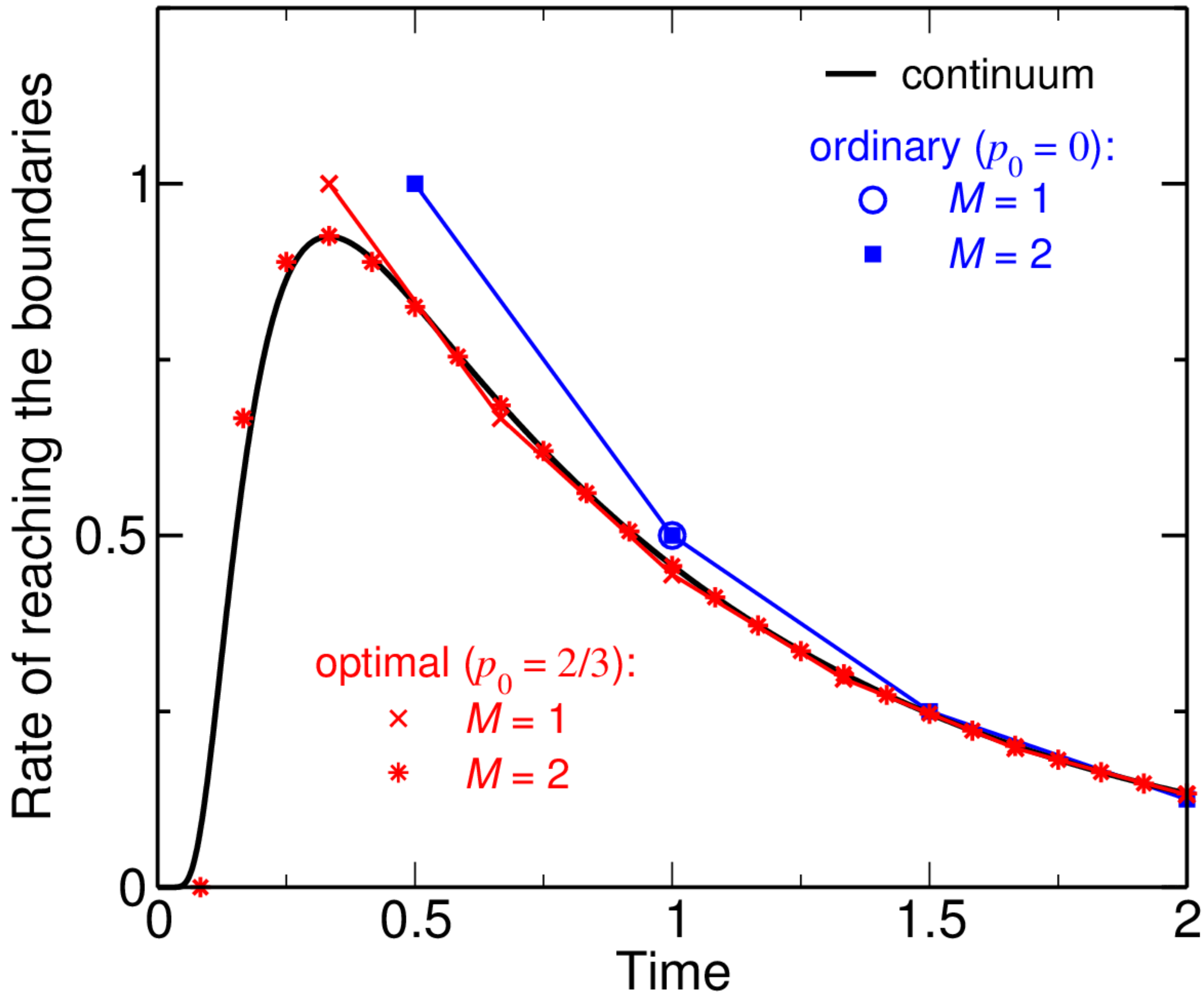
$b = 1, D = 1/2$



Ordinary algorithm, $M = 2$: similar to optimal with $M = 1$, but actually less accurate.

Full first-passage-time distribution

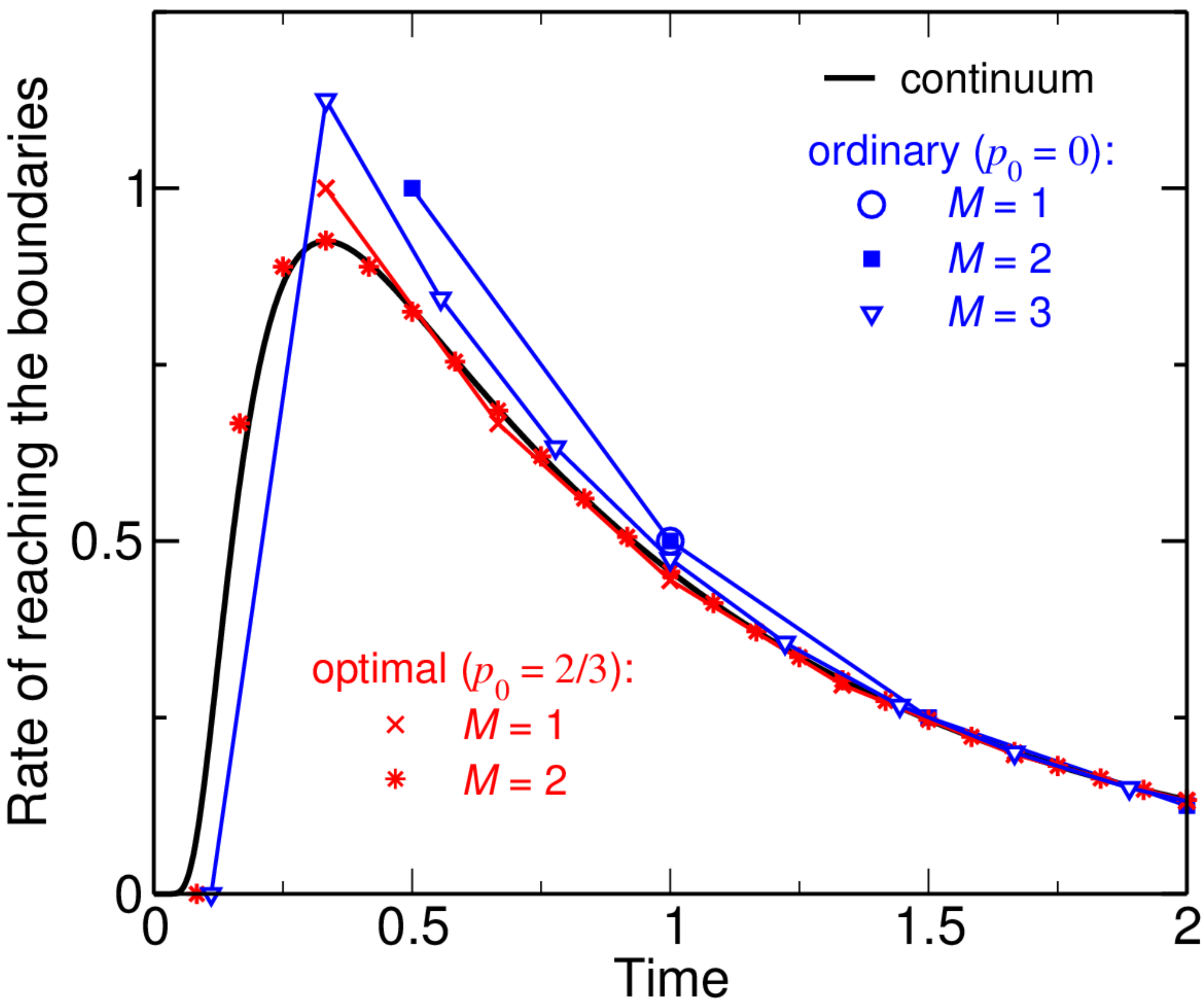
$b = 1, D = 1/2$



Optimal algorithm, $M = 2$: nearly perfect.

Full first-passage-time distribution

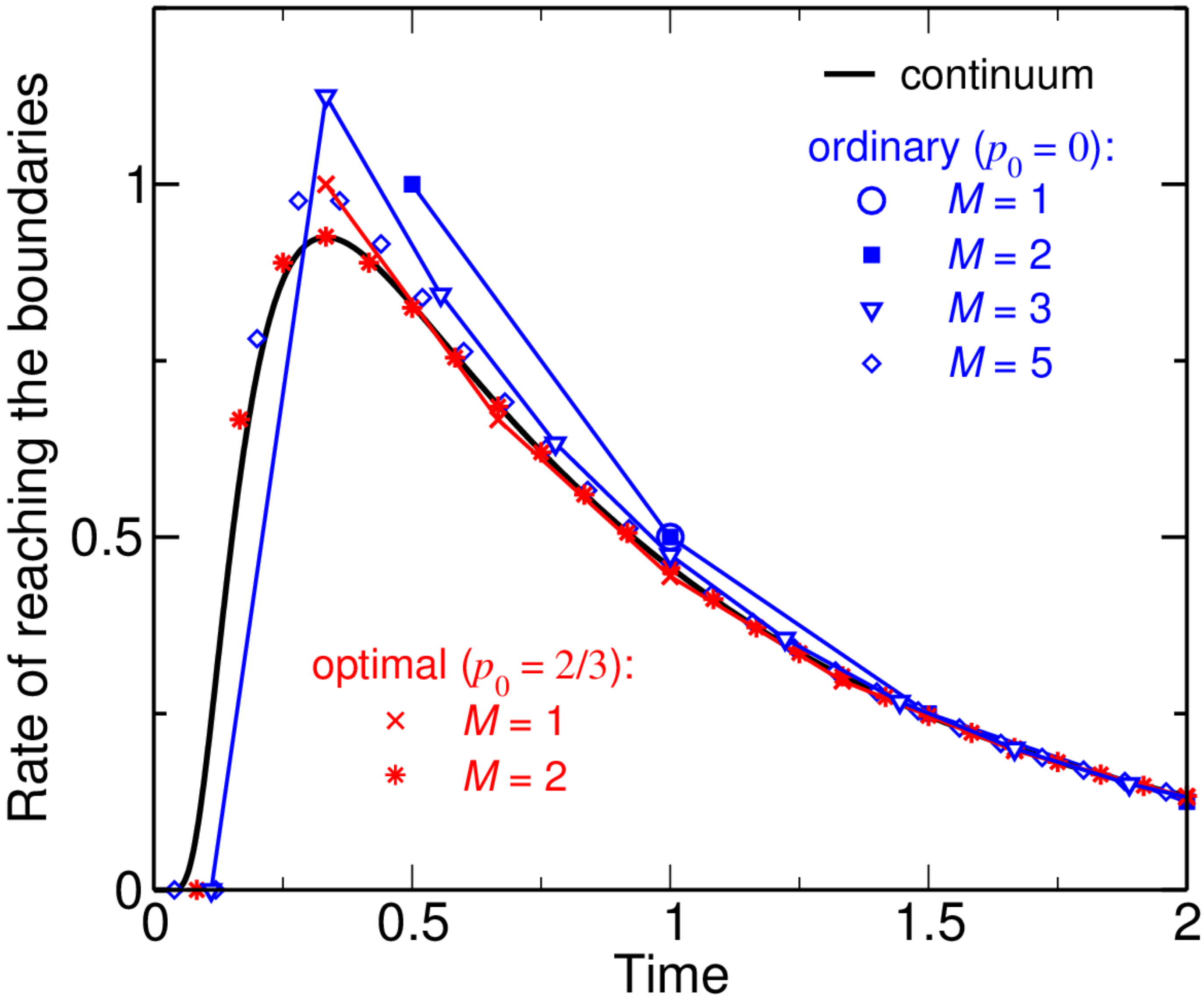
$b = 1, D = 1/2$



For the ordinary algorithm, as M increases, the distribution approaches the continuum one, but rather slowly.

Full first-passage-time distribution

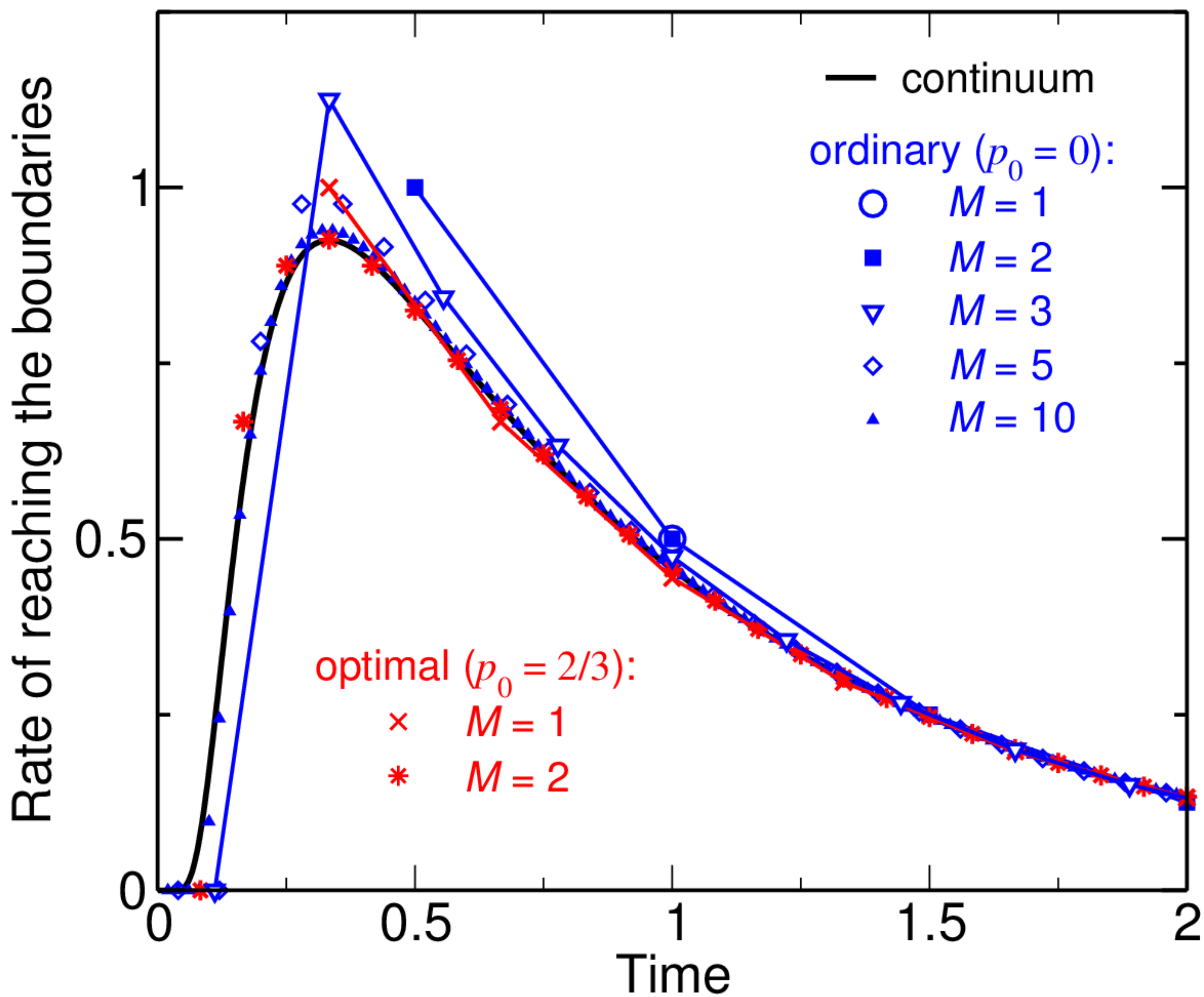
$b = 1, D = 1/2$



For the ordinary algorithm, as M increases, the distribution approaches the continuum one, but rather slowly.

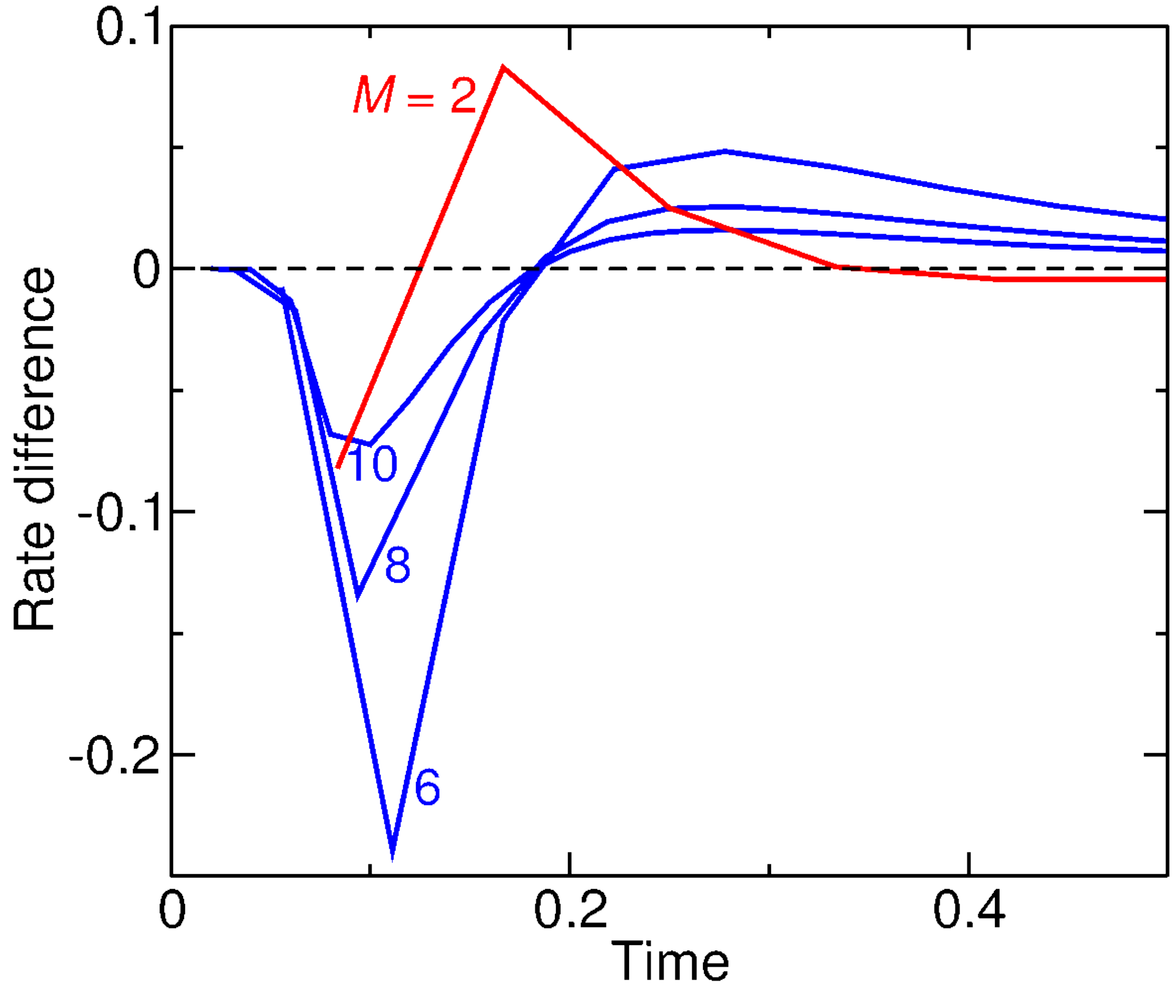
Full first-passage-time distribution

$b = 1, D = 1/2$



For the ordinary algorithm, as M increases, the distribution approaches the continuum one, but rather slowly.

The differences between the LMC and continuum first-passage rates



$b = 1, D = 1/2$

For the ordinary algorithm ($p_0 = 0$) it takes M^2 steps on average to reach the boundary.

For the optimal algorithm ($p_0 = 2/3$) it takes $3M^2$ steps.

But the optimal algorithm with $M = 2$ is about as accurate as the ordinary algorithm with $M = 8 - 10$.

$3 \times 2^2 = 12$ steps vs. $8^2 = 64$ steps. Speedup ~ a factor of 5 due to the possibility of using a coarser mesh. Of course, a highly idealized situation ...

Evolution of the particle concentration

Continuum: diffusion equation $$\frac{\partial n(x,t)}{\partial t} = D\frac{\partial^2 n(x,t)}{\partial x^2}.$$

LMC: master equation for the evolution of the mean particle number in site i:

$$n_j(t+\tau) = p_+ n_{j-1}(t) + p_0 n_j(t) + p_- n_{j+1}(t)$$

This is actually a discrete approximation of the diffusion equation (forward time centred space finite difference scheme):

$$\frac{\partial n(x_j,t)}{\partial t} \approx \frac{n(x_j,t+\tau)-n(x_j,t)}{\tau} \to \frac{n_j(t+\tau)-n_j(t)}{\tau}$$

$$\left.\frac{\partial^2 n(x,t)}{\partial x^2}\right|_{x=x_j} \approx \frac{n(x_{j+1},t)-2n(x_j,t)+n(x_{j-1},t)}{a^2} \to \frac{n_{j+1}(t)-2n_j(t)+n_{j-1}(t)}{a^2}$$

$$n_j(t+\tau) = \frac{D\tau}{a^2}n_{j-1}(t) + \left(1-\frac{D\tau}{a^2}\right)n_j(t) + \frac{D\tau}{a^2}n_{j+1}(t)$$

Coincides with the master equation above, when $p_0 = 1-\frac{D\tau}{a^2}, \quad p_+ = p_- = \frac{1-p_0}{2}.$

For what τ (or p_0) is this the best approximation?

$$\frac{\partial n(x,t)}{\partial t} = D\frac{\partial^2 n(x,t)}{\partial x^2}.$$ General solution $$n(x,t) = \int_{-\infty}^{\infty} C(k)\exp[ikx - \alpha_c(k)t]dk.$$

Dispersion relation

$\alpha_c(k) = Dk^2$ – rate of decay of the mode with wave vector k.

$$n_j(t+\tau) = p_+ n_{j-1}(t) + p_0 n_j(t) + p_- n_{j+1}(t) \Rightarrow n(x,t) = \int_{-\pi/a}^{\pi/a} C(k)\exp[ikx - \alpha_d(k)t]dk.$$

$$\alpha_d(k) = -\frac{1}{\tau}\ln[p_0 + p_+\exp(-ika) + p_-\exp(ika)] = A_0 k^0 + A_1 k^1 + A_2 k^2 + A_3 k^3 + A_4 k^4 + \ldots$$

Match as many coefficients to $\alpha_c(k)$ as possible.

$A_0 = 0 \;\Rightarrow\; p_0 + p_+ + p_- = 1.$

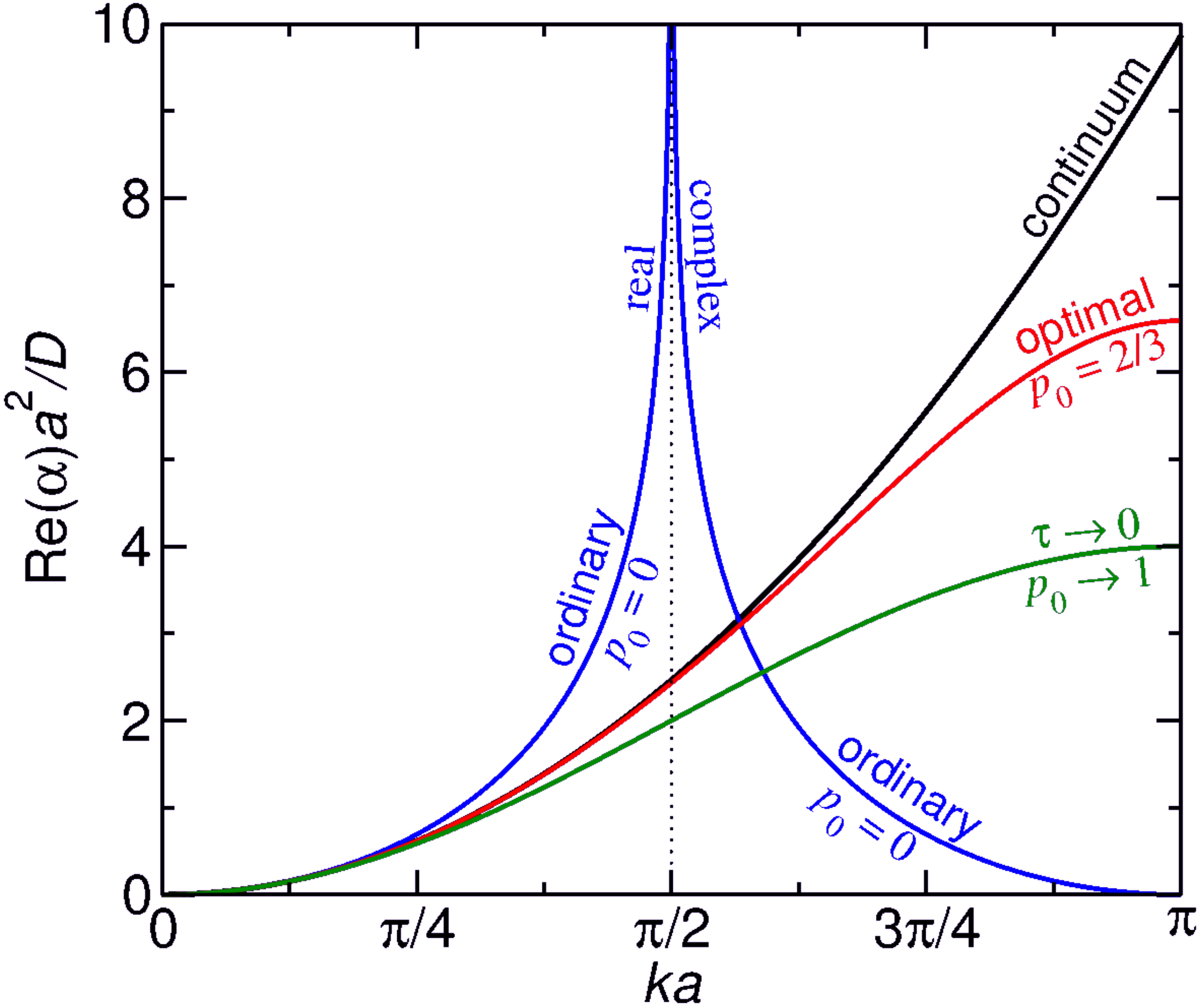
$A_1 = 0 \;\Rightarrow\; p_+ = p_-.$ Also makes all other odd coefficients zero.

$A_2 = \frac{a^2(1-p_0)}{2\tau} = D.$ The usual relation between p_0 and τ giving the correct MSD.

$A_4 = 0 \Rightarrow$ $p_0 = 2/3$. Same optimal algorithm. $\alpha_d(k) = Dk^2 - \frac{Da^4}{540}k^6 + \ldots$

For $p_0 = 0$, $\alpha_d(k) = Dk^2 + \frac{Da^2}{6}k^4 + \frac{2Da^4}{45}k^6 + \ldots$

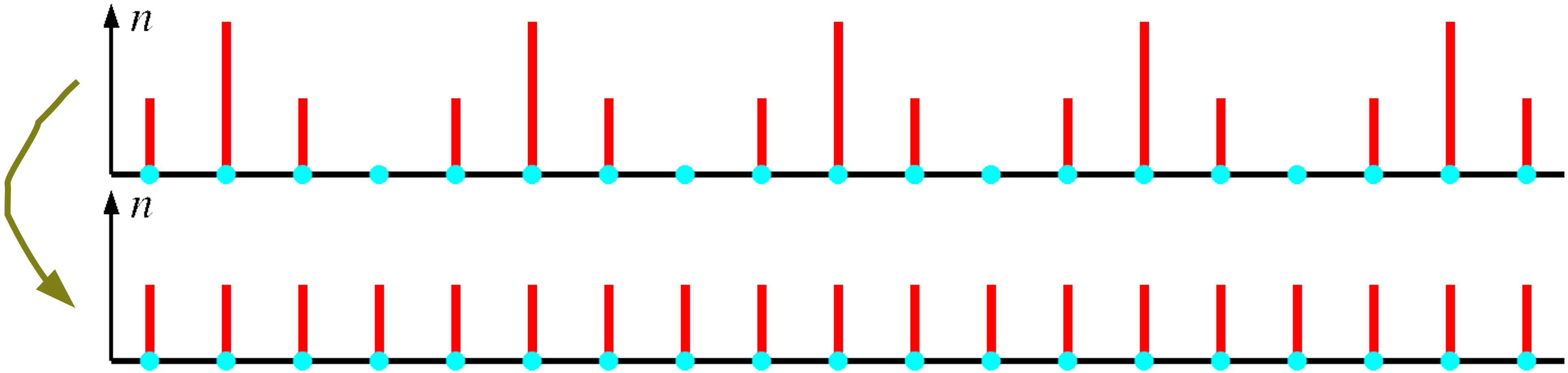
Comparison of the dispersion curves



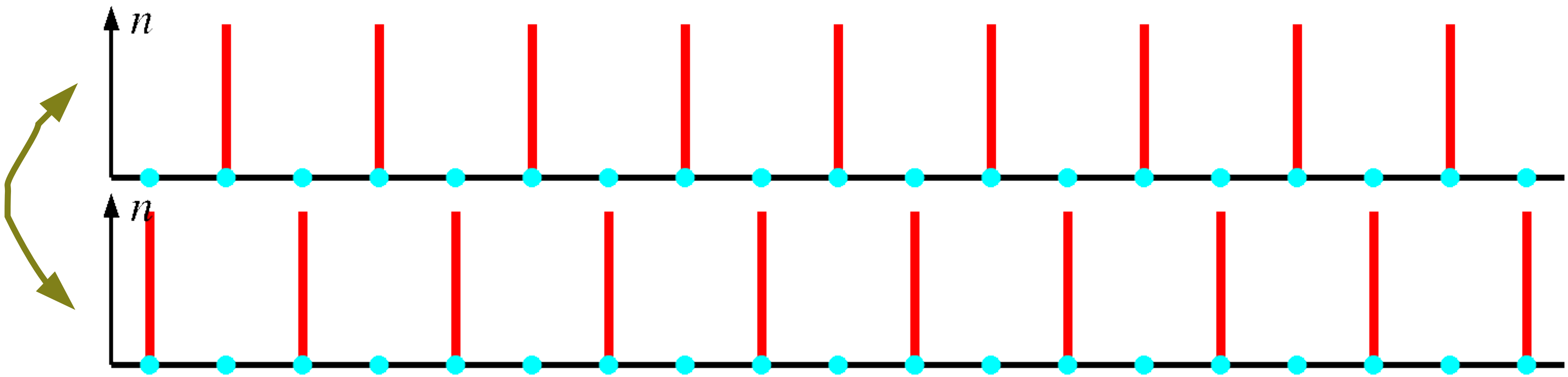
The optimal algorithm is more accurate than both the ordinary one and the one with time step → zero. For the ordinary algorithm, 3 peculiarities: 1) α diverges at $k = \pi a/2$; 2) α complex for $\pi a/2 < k < \pi a$ (decaying oscillations); 3) α purely imaginary at $k = \pi a$.

Artifacts of the ordinary algorithm

1) Decay rate $\alpha = \infty$ at $k = \pi a/2$. Decays to uniform in one step.



2) Decay rate $\alpha = i\pi/\tau$ at $k = \pi a$. Oscillates indefinitely without decaying.



Not a problem for the optimal algorithm – gradual decay in both cases.

Two dimensions

Continuum: $$\frac{\partial n(x,y,t)}{\partial t}=D\left(\frac{\partial^2 n(x,y,t)}{\partial x^2}+\frac{\partial^2 n(x,y,t)}{\partial y^2}\right).$$ Solution:

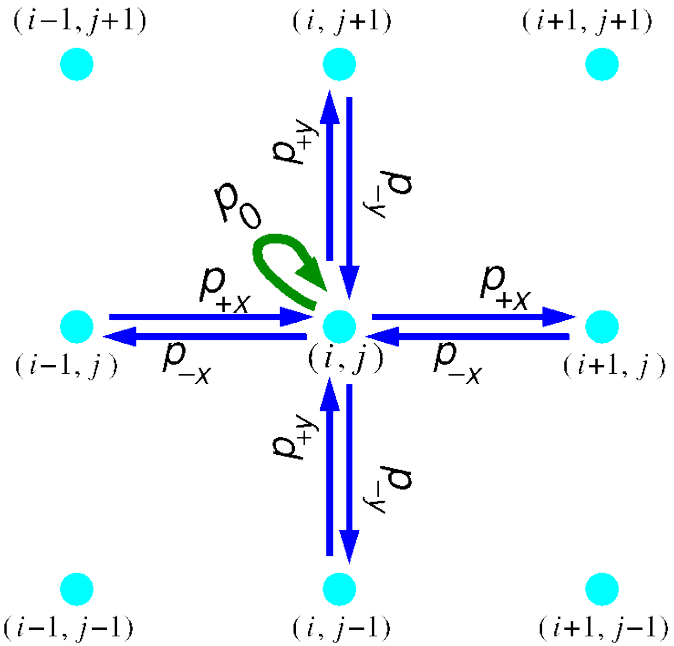
$$n(x,y,t)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} C(k_x,k_y)\exp[i(k_x x+k_y y)-\alpha_c(k_x,k_y)t]dk_x dk_y, \quad \alpha_c(k)=Dk^2.$$

LMC: the simplest choice is the square lattice with moves along the *x* and *y* axes.

$$n_{j,k}(t+\tau)=p_0 n_{j,k}(t)+p_{+x}n_{j-1,k}(t)+p_{-x}n_{j+1,k}(t)+p_{+y}n_{j,k-1}(t)+p_{-y}n_{j,k+1}(t)$$

$$n_{j,k}(t)=\int_{-\pi/a}^{\pi/a}\int_{-\pi/a}^{\pi/a} C(k_x,k_y)\exp[i(k_x x_j+k_y y_k)-\alpha_d(k_x,k_y)t]dk_x dk_y.$$

In particular, the ordinary algorithm is $p_0=0,\ \ p_{\pm x}=p_{\pm y}=1/4,\ \ \tau=a^2/4D.$



In general, from symmetry considerations

$$p_{\pm x}=p_{\pm y}=(1-p_0)/4.$$

$$\alpha_d(k_x,k_y)=-\frac{1}{\tau}\ln\{p_0+(1-p_0)[\cos(k_x a)+\cos(k_y a)]/2\}$$
$$=-\frac{1}{\tau}\ln\{1+(1-p_0)[f(k_x^2)+f(k_y^2)]\}$$

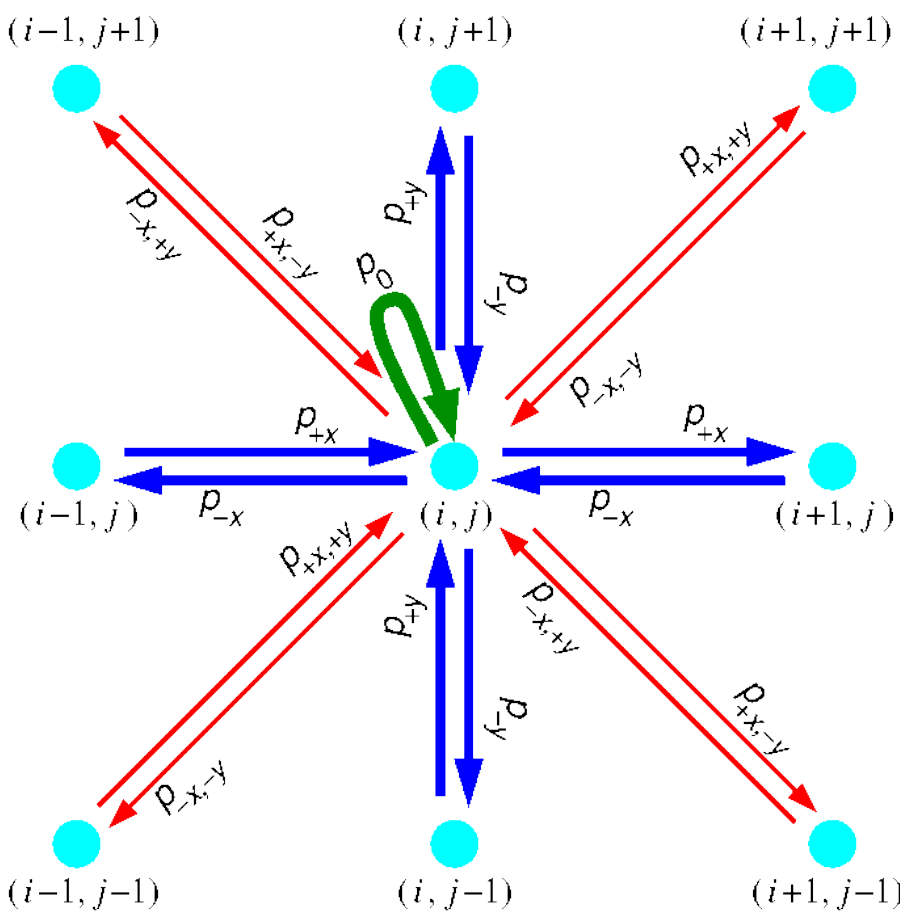
When expanded, there will inevitably be cross-terms, in particular, $k_x^2 k_y^2$ – can't match $\alpha_c(k)$.

Two dimensions

Need to include the possibility to move simultaneously in the *x* and *y* directions.

Probabilities $p_0,\ \ p_{\pm x} = p_{\pm y},\ \ p_{\pm x, \pm y}.$

$$\alpha_d(k_x, k_y) = -\frac{1}{\tau}\ln\{p_0 + 2p_{+x}[\cos(k_x a) + \cos(k_y a)] + 4p_{+x,+y}\cos(k_x a)\cos(k_y a)\}$$



Can choose the probabilities so that the cross-terms vanish and

$$\alpha_d(k_x, k_y) = D(k_x^2 + k_y^2) + O(k^6)$$

$$p_0 = 4/9;\ p_{\pm x} = p_{\pm y} = 1/9;\ p_{\pm x, \pm y} = 1/36.$$

These probabilities are products of the probabilities of the 1D moves that the 2D move "consists of":

$$p_0^{(2D)} = p_0^{(1D)} \times p_0^{(1D)} = (2/3)\times(2/3) = 4/9;$$
$$p_{+x}^{(2D)} = p_+^{(1D)} \times p_0^{(1D)} = (1/6)\times(2/3) = 1/9;$$
$$p_{-x,+y}^{(2D)} = p_-^{(1D)} \times p_+^{(1D)} = (1/6)\times(1/6) = 1/36,$$

etc. The time step τ is still $a^2/6D$.

Two dimensions (continued)

The matrix of 2D optimal probabilities can be represented as the direct product of the vectors of the 1D optimal probabilities:

$$\begin{pmatrix} p^{(2D)}_{-x,-y} & p^{(2D)}_{-x} & p^{(2D)}_{-x,+y} \\ p^{(2D)}_{-y} & p^{(2D)}_{0} & p^{(2D)}_{+y} \\ p^{(2D)}_{+x,-y} & p^{(2D)}_{+x} & p^{(2D)}_{+x,+y} \end{pmatrix} = \begin{pmatrix} p^{(1D)}_{-} \\ p^{(1D)}_{0} \\ p^{(1D)}_{+} \end{pmatrix} \otimes \begin{pmatrix} p^{(1D)}_{-} & p^{(1D)}_{0} & p^{(1D)}_{+} \end{pmatrix}$$

Three dimensions

The "direct product" algorithm $p^{(3D)} = p^{(1D)} \otimes p^{(1D)} \otimes p^{(1D)}$ likewise has the same 4th-order accuracy. Has moves in 1, 2 and all 3 directions, as well as staying put.

$$p_0 = 8/27;\ p_{\pm x} = p_{\pm y} = p_{\pm z} = 2/27;\ p_{\pm x,\pm y} = p_{\pm x,\pm z} = p_{\pm y,\pm z} = 1/54;$$
$$p_{\pm x,\pm y,\pm z} = 1/216;\ \tau = a^2/6D.$$

But it is also possible to have the same order of accuracy with moves in 1 and 2 directions only.

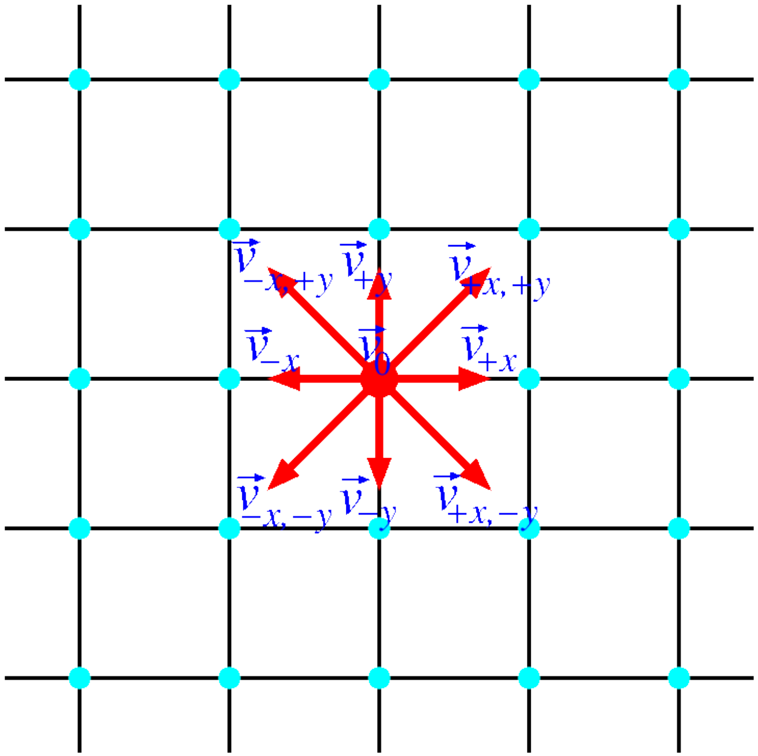
$$p_0 = 1/3;\ p_{\pm x} = p_{\pm y} = p_{\pm z} = 1/18;\ p_{\pm x,\pm y} = p_{\pm x,\pm z} = p_{\pm y,\pm z} = 1/36;\ \tau = a^2/6D.$$

Relation to Lattice Boltzmann algorithms

A method for fluid simulations. Boltzmann's kinetic equation for the particle distribution function in the Bhatnagar-Gross-Krook approximation:

$$\frac{\partial f(\vec{r},\vec{v},t)}{\partial t} = -\vec{v}\cdot\frac{\partial f(\vec{r},\vec{v},t)}{\partial \vec{r}} - \omega\left[f(\vec{r},\vec{v},t) - f_0(\vec{v};\vec{V}(\vec{r}))\right], \text{ where } f_0(\vec{v};\vec{V}(\vec{r}))$$

is the Maxwell distribution shifted by the average velocity at $\vec{r}$, $\vec{V}(\vec{r})$.



This is solved on a lattice, and the possible set of velocities is discrete, such that in 1 step of the algorithm a particle would move between lattice sites.

Notation: D*m*Q*n*, where *m* is the dimensionality and *n* is the number of velocities. E.g., D2Q9 in the figure.

Need to find the best discretization of the Maxwell distribution: $f_0(\vec{v};0) \to f^{(0)}_{\vec{v}_i}$

Turns out for D2Q9 $f^{(0)}_0 = p^{(2D)}_0$; $f^{(0)}_{+x} = p^{(2D)}_{+x}$; $f^{(0)}_{+x,+y} = p^{(2D)}_{+x,+y}$, etc.

Relation to Lattice Boltzmann algorithms

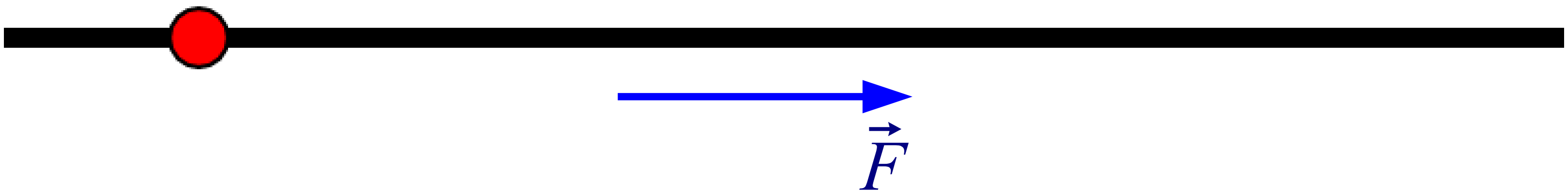
Likewise, for the D3Q19 LB algorithm, the approximations to the Maxwell distribution are the **same** as the probabilities of the moves of the 3D optimal algorithm with moves in 1 and 2 directions, and for the D3Q27 algorithm, these approximations are the **same** as the probabilities of the moves of the 3D "direct product" optimal algorithm with moves in 1, 2 and 3 directions.

This is not a coincidence: the probabilities of the optimal LMC are chosen so that the displacement distribution after 1 step is as close to the continuum displacement distribution as possible (gives as many correct moments as possible). Of course, this distribution is the same as the Maxwell distribution:

$$P(x,t)=\frac{1}{2\sqrt{\pi D t}}\exp\left(-\frac{x^2}{4Dt}\right) \qquad P(v_x,t)=\sqrt{\frac{m}{2\pi k_B T}}\exp\left(-\frac{m v_x^2}{2 k_B T}\right)$$

Moreover, there are also LB methods for simulating diffusion [e.g., Zhang et al., Water Resour. Res. 38 (2002) 1167]. In certain particular cases, they are equivalent to solving the master equation for the optimal LMC.

Diffusion with drift due to an external force



A LMC method should satisfy at least the following 3 conditions:

1) detailed balance $p_+/p_- = \exp(Fa/k_B T)$; 2) correct diffusivity D;

3) correct drift velocity $v = (D/k_B T)F$.

The algorithm satisfying all 3 criteria has [M.G. Gauthier and G.W. Slater, Phys. Rev. E **70** (2004) 015103]

$$p_0 = \frac{\coth\epsilon}{\epsilon} - \operatorname{csch}^2\epsilon;\ p_\pm = \frac{1-p_0}{1+e^{\mp 2\epsilon}};\ \tau = \frac{a^2}{2D}\frac{\tanh\epsilon}{\epsilon}(1-p_0);\ \epsilon = \frac{Fa}{2k_B T}.$$

Interestingly, in the limit $F \to 0$ these become $p_0 = 2/3,\ p_\pm = 1/6,\ \tau = \dfrac{a^2}{6D},$

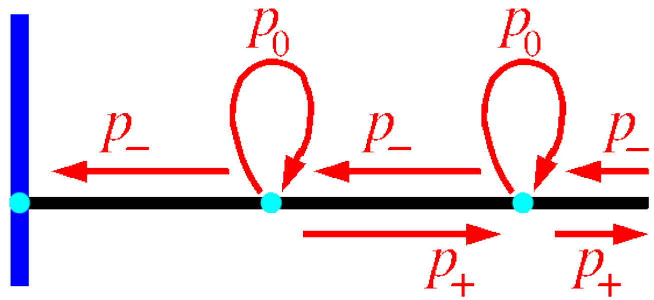
i.e., the optimal algorithm for unbiased diffusion. Not obvious *a priori*, because even the ordinary algorithm satisfies all 3 conditions.

The biased diffusion algorithm gives the correct MFPT and MSFPT, but the 4th moment of the displacement distribution is only correct in the limit $F \to 0$.

Boundaries

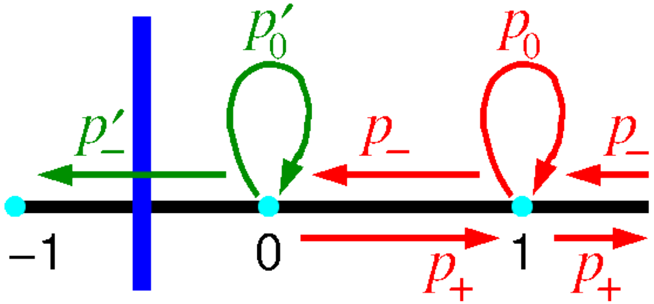
Of course, diffusion in free space is not interesting by itself – we know the solution!

Already considered absorbing boundaries when looking at the first-passage problem



The boundary coincides with a lattice site.

All probabilities are the same as in free space, but particles reaching the boundary site disappear.

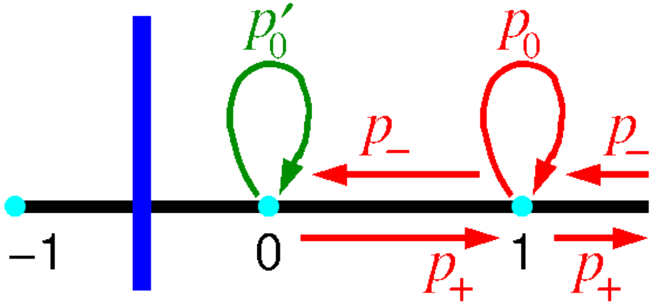


Also possible to achieve the same accuracy for the case when the boundary is midway between sites.

$$p'_0 = 1/2; \quad p'_- = 1/3$$

Usual master equation(ignoring the boundary)

$$n_0(t+\tau) = \frac{1}{6} n_{-1}(t) + \frac{2}{3} n_0(t) + \frac{1}{6} n_1(t). \quad \text{Fix} \quad n_{-1} = -n_0 \;\Rightarrow\; n_0(t+\tau) = \frac{1}{2} n_0(t) + \frac{1}{6} n_1(t).$$

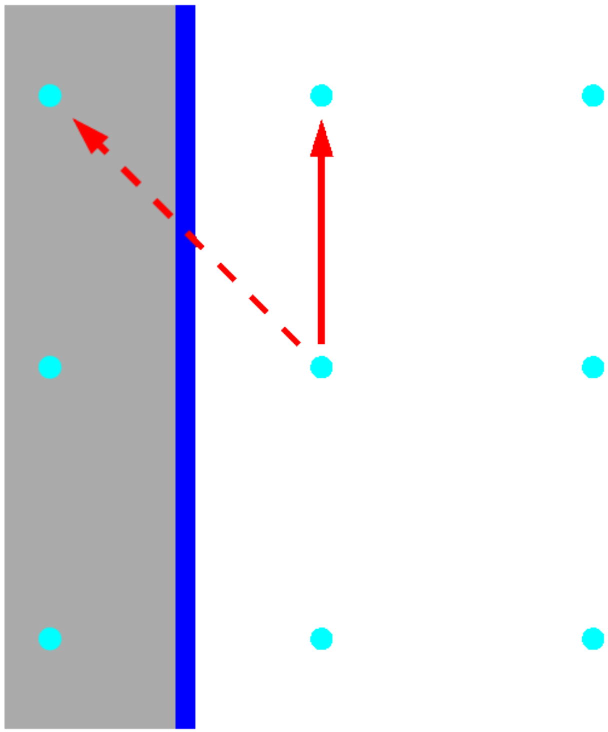


For impenetrable reflecting boundaries, only placing the boundary halfway preserves the accuracy.

$p'_0 = 5/6.$ Attempts to cross the boundary rejected

Same master equation, but $n_{-1} = n_0 \;\Rightarrow\; n_0(t+\tau) = \dfrac{5}{6} n_0(t) + \dfrac{1}{6} n_1(t).$

Boundaries



Mostly straightforward generalization to higher dimensions for boundaries parallel to the axes.

In the reflecting boundary case, diagonal moves attempting to cross the boundary are projected onto it, rather than rejected.

The same as the “collisions” we've heard about.

While it is impossible to preserve the same accuracy for other distances between the wall and the adjacent sites, the optimal way to treat that situation is still a relevant and important question, especially in view of generalizations for curved boundaries in higher dimensions.

Boundaries between regions with different diffusivities.

More details:

M.V. Chubynsky and G.W. Slater, Phys. Rev. E **85**, 016709 (2012).

Application to a 1D model of polymer translocation through a nanopore:

H.W. de Haan, M.G. Gauthier, M.V. Chubynsky, and G.W. Slater, Comput. Phys. Commun. **182**, 29 (2011).

What we considered are just “toy models”. Does this approach still make sense in more complicated situations (curved boundaries, space- and time-dependent external forces and diffusivities)?

So far, considered an ideal situation assuming averaging over ∞ realizations.

In reality, MC simulations are done by averaging over a finite number of runs, so actual averages will deviate from the ideal ones. Using the optimal algorithm only makes sense if the improvement on the ordinary algorithm is larger than the standard deviation.

Consider the MSFPT. The deviation of the ordinary result from the exact optimal one is $b^4/6D^2M^2$. On the other hand, the standard deviation of the optimal result is

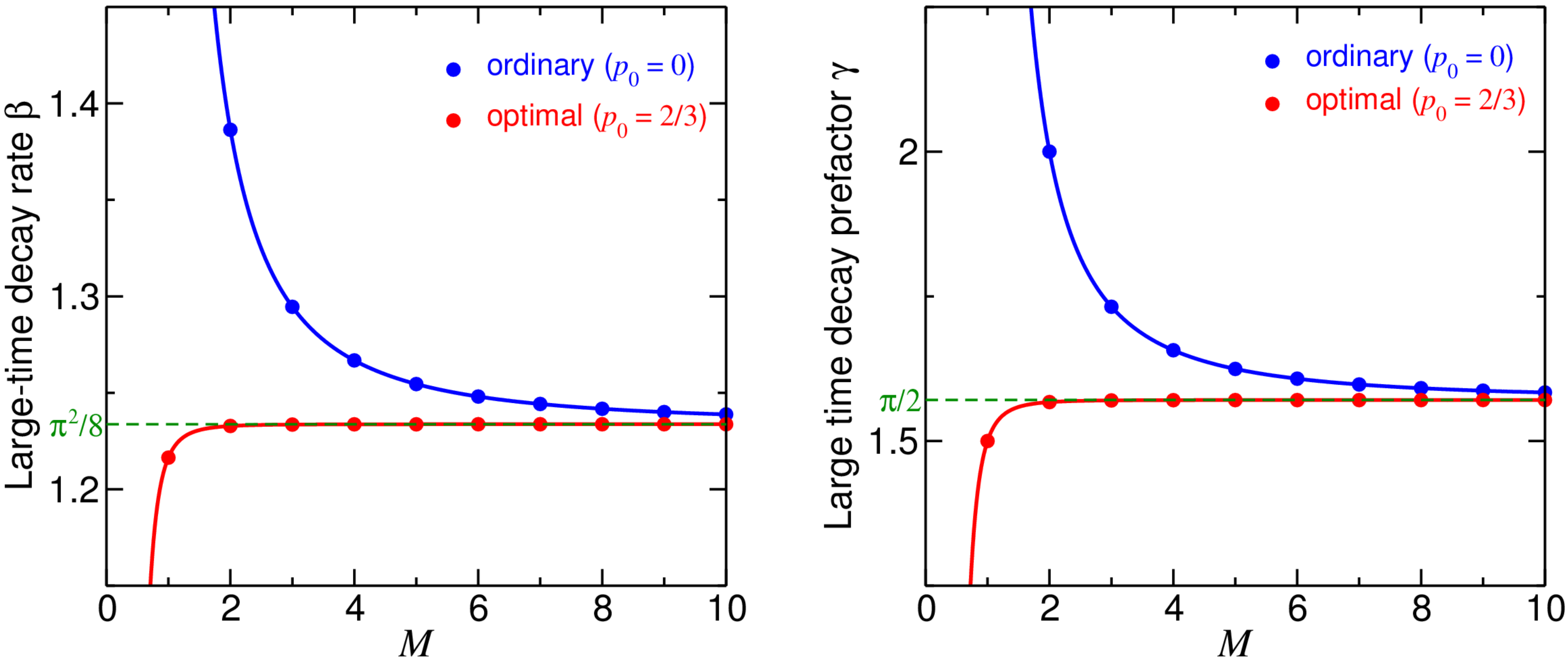
$$\sqrt{\frac{1}{N}[\langle t_1^4\rangle - \langle t_1^2\rangle^2]} \approx \sqrt{\frac{41}{63N}}\frac{b^4}{D^2}.$$

Thus, to benefit from the optimal algorithm, one needs to do $N \sim \frac{1476}{63}M^4 \approx 23.4\,M^4$ realizations. Grows rapidly with M, but quite small for small M, when the optimal algorithm is already rather accurate. Makes sense for coarse discretizations.

It is also possible to solve master equations directly – numerically exact methods [e.g., Mercier, Slater, Guo, JCP 110, 6050 (1999)], in which case this issue does not arise.

First-passage-time distribution. Large-time behaviour.

$r \simeq \gamma \exp(-\beta t)$ Continuum: $\beta = \pi^2/8$; $\gamma = \pi/2$



Undefined for ordinary, $M = 1$, because the FPT is deterministic. For optimal, quite close already for $M = 1$ and essentially exact for $M = 2$ (better than ordinary with $M = 10$).