Approximate accelerated stochastic simulation of chemically reacting systems

Colin Gillespie

October 16, 2013

Overview

- Stochastic kinetic models
- Issues with the Direct method
- τ-leap method (Gillespie, 2001)
  - Leap conditions
  - Mid-point estimation
- Examples

Source code (R and LaTeX of these slides):

https://github.com/csgillespie/talks/

Stochastic kinetic models

A biochemical network is represented as a set of pseudo-biochemical reactions:

u species & v reactions

$$R_i: \quad p_{11}\mathcal{X}_1 + p_{12}\mathcal{X}_2 + \cdots + p_{1u}\mathcal{X}_u \xrightarrow{c_i} q_{11}\mathcal{X}_1 + q_{12}\mathcal{X}_2 + \cdots + q_{1u}\mathcal{X}_u$$

Stochastic rate constant c_i.

Hazard/instantaneous rate: $h_i(X_t, c_i)$ where $X_t = (X_{1,t}, \ldots, X_{u,t})$ is the current state of the system.

Under mass-action stochastic kinetics, the hazard function is proportional to a product of binomial coefficients, with

$$h_i(X_t, c_i) = c_i \prod_{j=1}^{u} \binom{X_{j,t}}{p_{ij}}.$$

Stochastic kinetic model

- Describe the SKM by a Markov jump process (MJP)
- The effect of reaction R_k is to change the value of each species X_i by $q_{ji} - p_{ji}$
- The stoichiometry matrix S has elements $s_{ij} = q_{ji} - p_{ji}$
- It can be shown that the time to the next reaction is

$$t \sim Exp(h_0(X_t, c)) \quad \text{where} \quad h_0(X_t, c) = \sum_{i=1}^{v} h_i(X_i, c_i)$$

  and the reaction is of type i with probability $h_i(X_t, c_i)/h_0(X_t, c)$
- The process is easily simulated using the Direct method (Gillespie algorithm)

Example: Lotka-Volterra system

- R_1: Prey reproduction

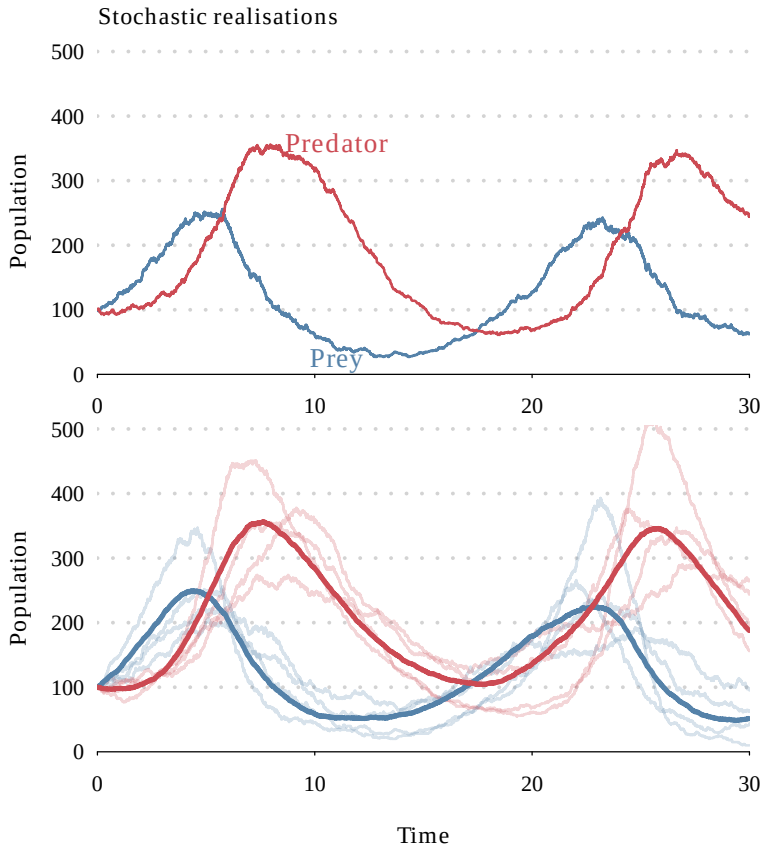
$$\mathcal{X}_1 \xrightarrow{c_1} 2\mathcal{X}_1$$

- R_2: Prey death, predator reproduction

$$\mathcal{X}_1 + \mathcal{X}_2 \xrightarrow{c_2} 2\mathcal{X}_2$$

- R_3: Predator death

$$\mathcal{X}_2 \xrightarrow{c_3} \emptyset$$



$\mathbf{X}(\mathbf{0}) = (100, 100)$

$\mathbf{c} = (0.5, 0.0025, 0.3)$

The direct method

1. **Initialisation:** initial conditions, reactions constants, and random number generators
2. **Propensities update:** Update each of the v hazard functions, $h_i(x)$
3. **Propensities total:** Calculate the total hazard $h_0 = \sum_{i=1}^{v} h_i(x)$
4. **Reaction time:** $\tau = -ln[U(0,1)]/h_0$ and $t = t + \tau$
5. **Reaction selection:** A reaction is chosen proportional to it's hazard
6. **Reaction execution:** Update species
7. **Iteration:** If the simulation time is exceeded stop, otherwise go back to step 2

Typically there are a large number of iterates

The direct method

1. **Initialisation:** initial conditions, reactions constants, and random number generators
2. **Propensities update:** Update each of the v hazard functions, $h_i(x)$
3. **Propensities total:** Calculate the total hazard $h_0 = \sum_{i=1}^{v} h_i(x)$
4. **Reaction time:** $\tau = -ln[U(0,1)]/h_0$ and $t = t + \tau$
5. **Reaction selection:** A reaction is chosen proportional to it's hazard
6. **Reaction execution:** Update species
7. **Iteration:** If the simulation time is exceeded stop, otherwise go back to step 2

Typically there are a large number of iterates

The direct method

1. **Initialisation:** initial conditions, reactions constants, and random number generators
2. **Propensities update:** Update each of the v hazard functions, $h_i(x)$
3. **Propensities total:** Calculate the total hazard $h_0 = \sum_{i=1}^{v} h_i(x)$
4. **Reaction time:** $\tau = -ln[U(0,1)]/h_0$ and $t = t + \tau$
5. **Reaction selection:** A reaction is chosen proportional to it's hazard
6. **Reaction execution:** Update species
7. **Iteration:** If the simulation time is exceeded stop, otherwise go back to step 2

Typically there are a large number of iterates

The direct method

1. **Initialisation:** initial conditions, reactions constants, and random number generators
2. **Propensities update:** Update each of the v hazard functions, $h_i(x)$
3. **Propensities total:** Calculate the total hazard $h_0 = \sum_{i=1}^{v} h_i(x)$
4. **Reaction time:** $\tau = -ln[U(0,1)]/h_0$ and $t = t + \tau$
5. **Reaction selection:** A reaction is chosen proportional to it's hazard
6. **Reaction execution:** Update species
7. **Iteration:** If the simulation time is exceeded stop, otherwise go back to step 2

Typically there are a large number of iterates

Approximations

Relax some assumptions (e.g. discreteness and stochasticity) in order to make simulation faster and more scalable

- Diffusion approximation / chemical Langevin equation (CLE)
- Linear noise approximation (LNA) / moment closure (2MA)
- ODE
- Hybrid discrete-continuous models

Poisson leap

- If all reactions are zeroth-order, then the model is a homogeneous Poisson process
- Hence the number of reactions in (t_0, t_1) follows a Poisson distribution
- For a more general model, if we consider a *small* time interval, $(t, t + \Delta t)$, then:
  - the hazard rates should be approximately constant
  - the number of reactions (of a given type) can be sampled from a Poisson distribution
- A balance between speed and accuracy

Poisson leap method

1. Set $t = 0$. Initialise the rate constants and the initial molecule numbers x
2. Calculate $h_i(x, c_i)$, for $i = 1, \ldots, v$, and simulate the v-dimensional reaction vector r, with i^{th} entry a $Po(h_i(x, c_i)\Delta t)$ random quantity
3. Update the state according
4. Update $t := t + \Delta t$
5. Output t and x. If $t < T_{max}$ return to step 2

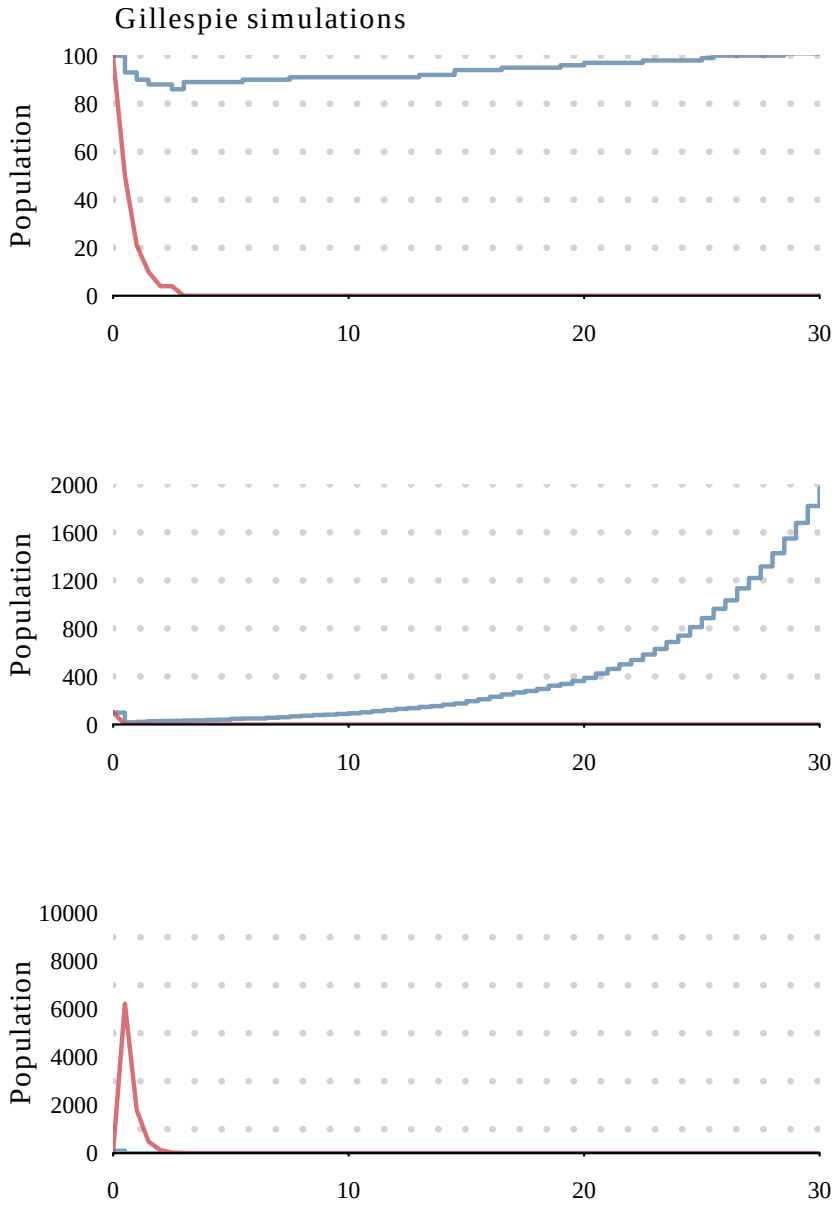
Poisson leap method

1. Set $t = 0$. Initialise the rate constants and the initial molecule numbers x
2. Calculate $h_i(x, c_i)$, for $i = 1, \ldots, v$, and simulate the v-dimensional reaction vector r, with i^{th} entry a $Po(h_i(x, c_i)\Delta t)$ random quantity
3. Update the state according
4. Update $t := t + \Delta t$
5. Output t and x. If $t < T_{max}$ return to step 2

How do you chose Δt?

Example: Lotka-Volterra

- Suppose we are interested in parameter inference
- A possible prior could be independent Uniform priors over $U(-8, 8)$ for each $\log(c_i)$
- Three samples from this prior yield very different realisations
  - Probability of extinction by time $t = 30$, is around 0.86
- Each simulation would require a very different Δt



Basic τ-leap method

- Suppose a temporal leap τ will result in a state change λ
- Choose a value of τ that satisfies the *leap condition*. For each reaction, R_j, we want

$$|h_j(\mathbf{x} + \lambda) - h_j(\mathbf{x})|$$

  to be *small*
- Sample $k_j \sim P(h_j(\mathbf{x})\tau)$
- Compute λ
- Set $t := t + \tau$ and $\mathbf{x} := \mathbf{x} + \lambda$

Choosing τ

- If the reactions don't depend on $\mathbf{x}$, the leap condition will be satisfied exactly for any τ (and so exact)
- If population numbers are large, then it would take a large number of reactions to "noticeably" change the hazard functions
- If satisfying the leap condition requires a very small value $\tau << 1/h_0(\mathbf{x})$, then we may as well use the exact SSA

A procedure for selecting τ (Gillespie 2001)

- The expected net change in $(t, t+\tau)$ will be:

$$\bar{\lambda} = \sum_{j=1}^{v} [h_j(\mathbf{x})\tau]\mathbf{s}_j = \tau\xi(\mathbf{x})$$

- So we require that the *expected* changes in the propensity functions in time τ, are bounded by some fraction of all propensity functions, i.e.

$$|h_j(\mathbf{x}+\lambda) - h_j(\mathbf{x})| < \epsilon h_0(\mathbf{x}) \quad \text{for } j = 1, \ldots, v.$$

- Estimate the difference using a Taylor expansion:

$$h_j(\mathbf{x}+\lambda) - h_j(\mathbf{x}) \simeq \sum_{i=1}^{u} \tau\xi_i(\mathbf{x}) \frac{\partial}{\partial x_i} h_j(\mathbf{x})$$

- Defining

$$b_{ji}(\mathbf{x}) \equiv \frac{\partial h_j(\mathbf{x})}{\partial x_i} \quad (j = 1, \ldots, v; i = 1, \ldots, u)$$

A procedure for selecting τ

- The requirement becomes

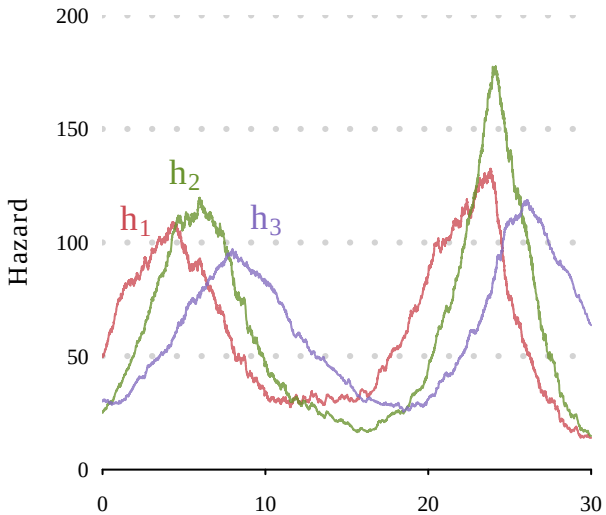
$$\tau \left| \sum_{i=1}^{u} \xi_i(\mathbf{x}) b_{ji}(\mathbf{x}) \right| \leq \epsilon h_0(\mathbf{x}) \quad (j = 1, \ldots, v)$$

- The largest value of τ consistent with this condition (and hence optimal) is

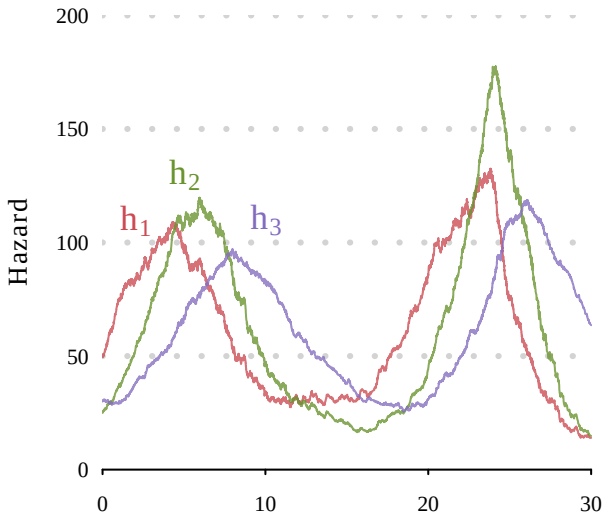
$$\tau = \min_{j \in [1,v]} \left\{ \frac{\epsilon h_0(\mathbf{x})}{\left| \sum_{i=1}^{u} \xi_i(\mathbf{x}) b_{ji}(\mathbf{x}) \right|} \right\}$$

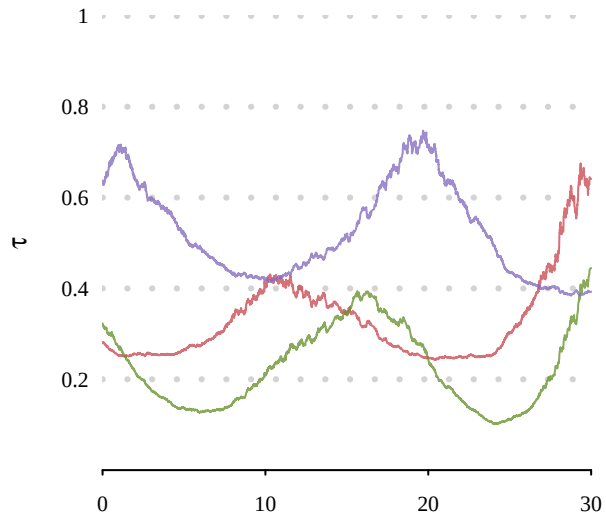
Typical values of ϵ are around 0.05

Example: Lotka-Volterra

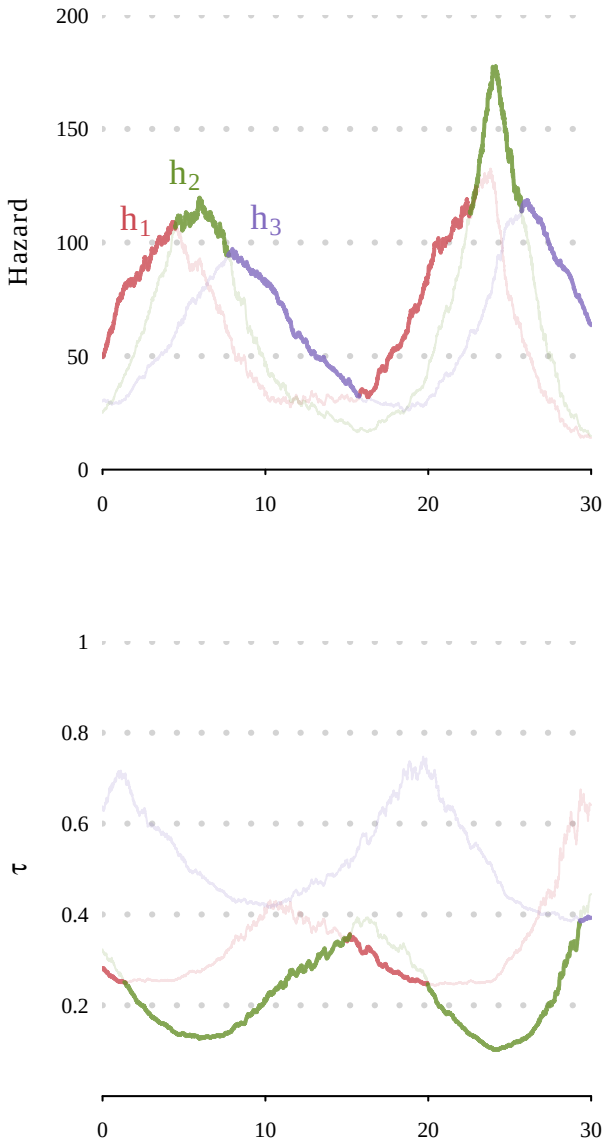


Example: Lotka-Volterra

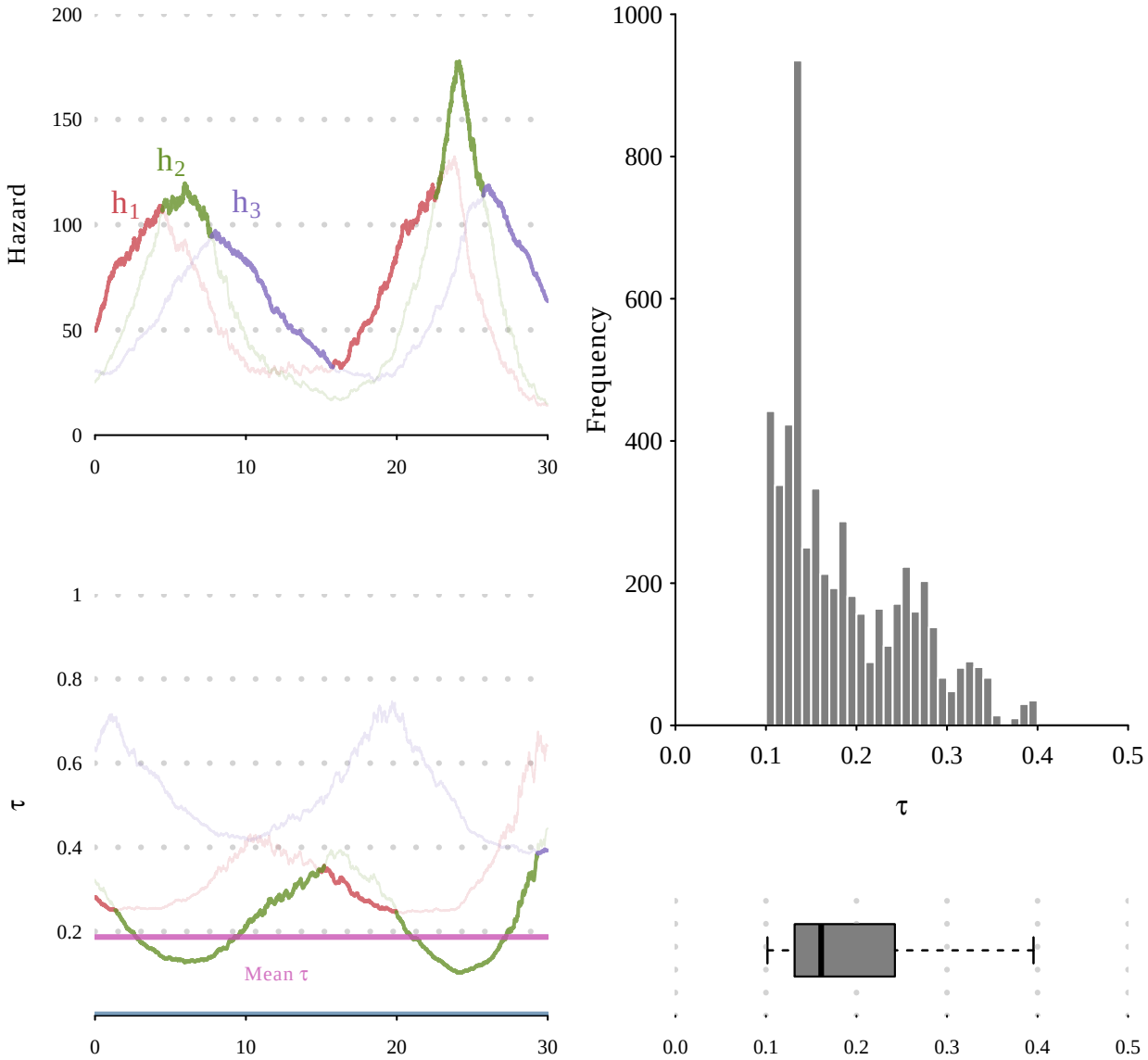




Example: Lotka-Volterra



Example: Lotka-Volterra



The estimated-midpoint technique

- The leap condition requires that the hazards functions do not "appreciably" change in the course of a leap
- But we want to take large leaps, so we will inevitably get computational errors
- This is similar to solving the ODE

$$\frac{dX(t)}{dt} = f(X)$$

  using an Euler scheme
- A standard technique is to use a *second-order Runge-Kutta* or *modified Euler* method

$$X(t + \Delta t) = X(t) + f[X(t) + 0.5f(X(t))\Delta t]\Delta t$$

  i.e. we use an Euler method to estimate the midpoint during $[t, t + \Delta t]$, then calculate the increment in X by evaluating the slope function f at that estimated midpoint

Example: Death model

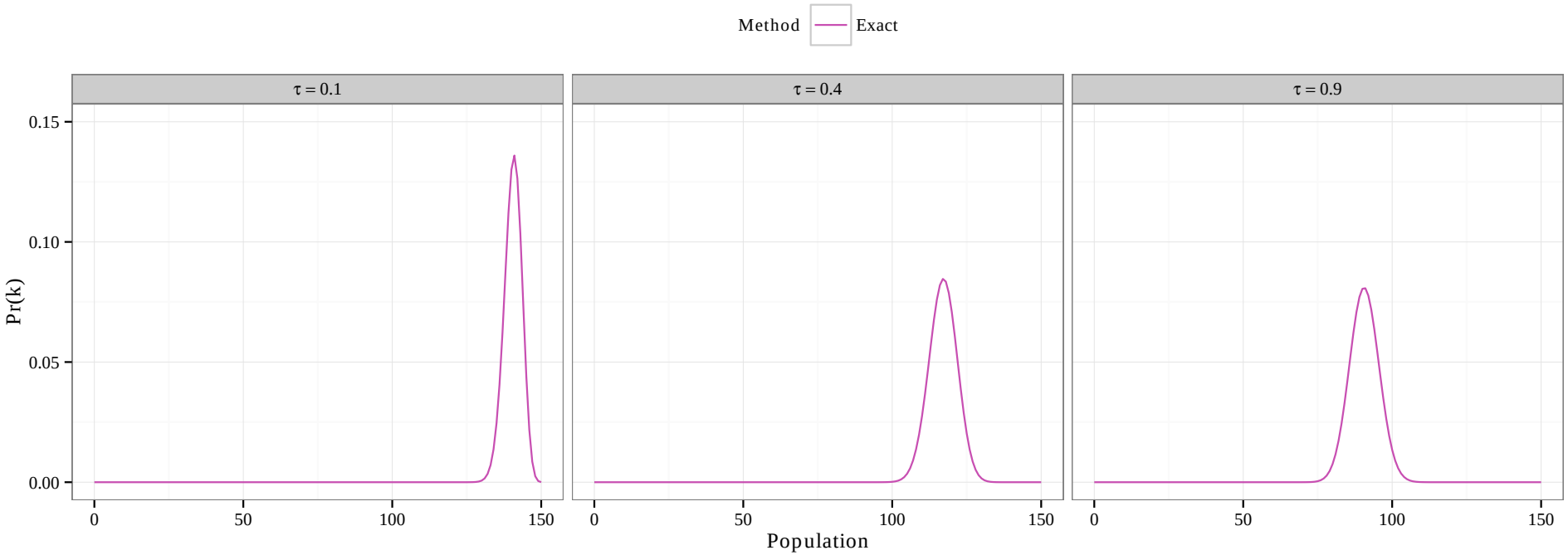
- The death model contains a single reaction

$$\mathcal{X} \xrightarrow{\mu} \emptyset$$

  and has hazard function $h_1(x, \mu) = \mu x$ and state change vector $s = -1$.
- The solution to the CME is:

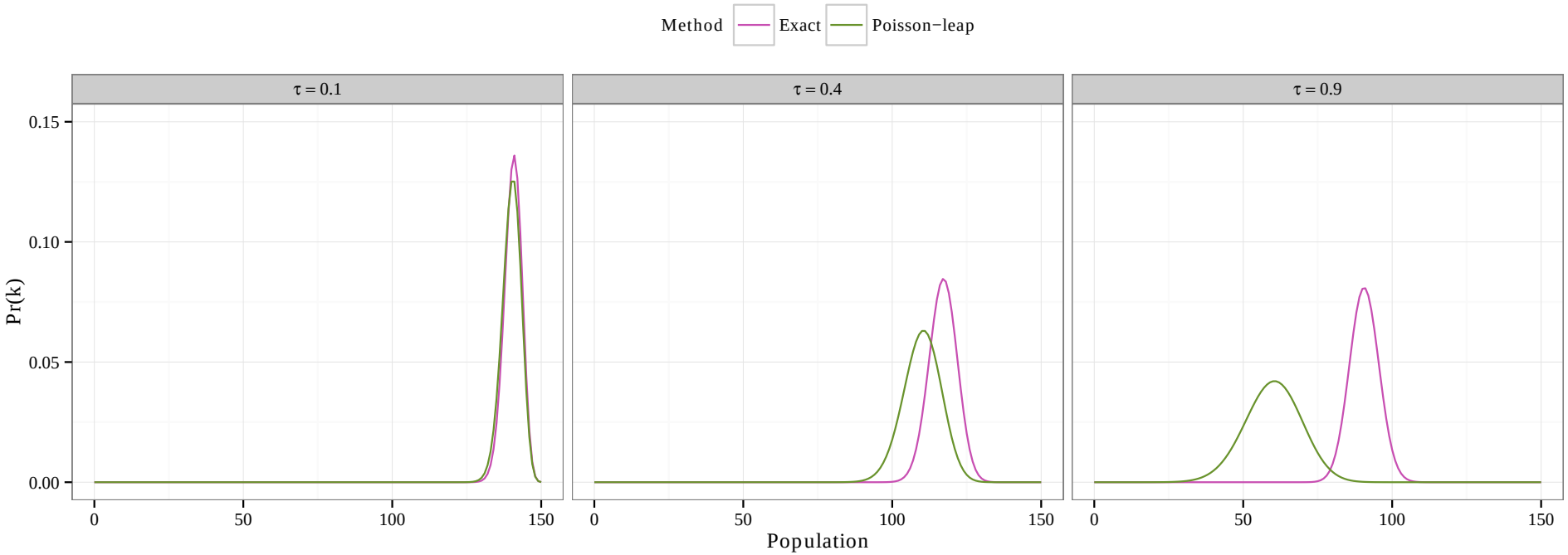
$$\Pr(X = x; t) = \binom{x_0}{x} e^{-\mu\tau(x_0 - x)}(1 - e^{-\mu\tau})^x$$

Death model



$$\Pr(X = x; \tau) = \binom{x_0}{x} e^{-\mu\tau(x_0 - x)} (1 - e^{-\mu\tau})^x$$

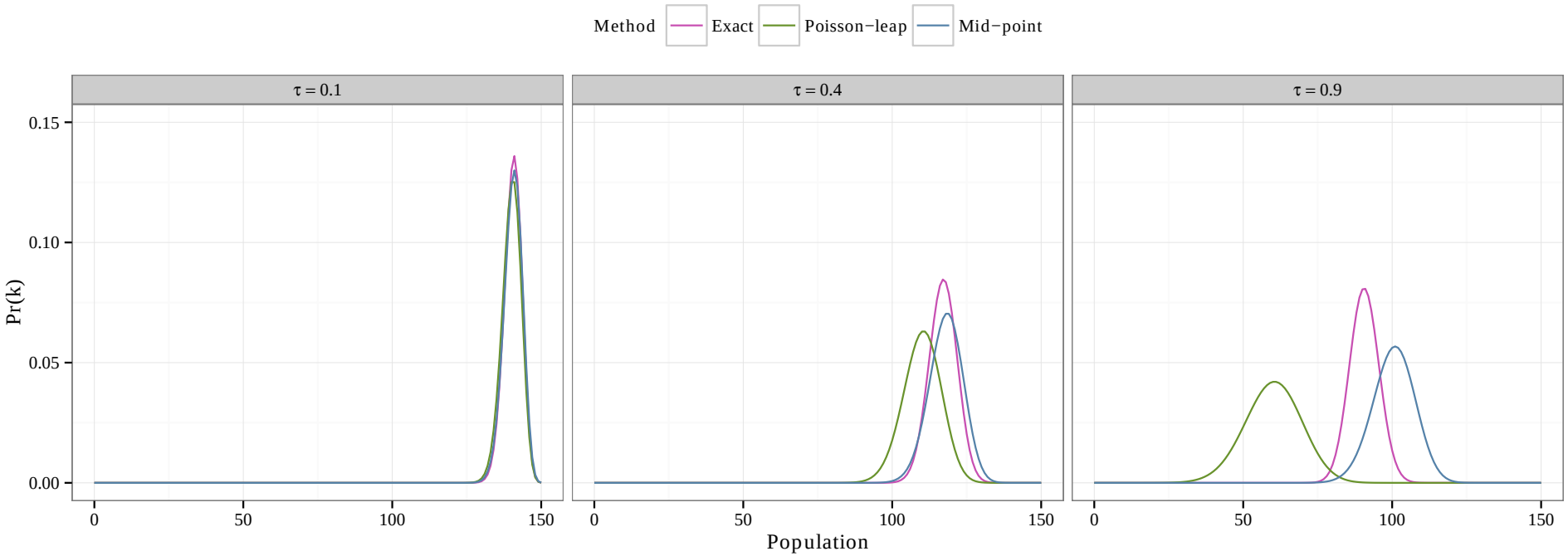
Death model: p-leap



If we perform a single leap of length τ, the number of executed reactions are

$$P_p(k; \mu x_0, \tau) = \frac{e^{\mu x_0 \tau} (\mu x_0 \tau)^k}{k!} \quad \text{for } k = 0, \ldots$$

Death model: τ-leap + mid-point



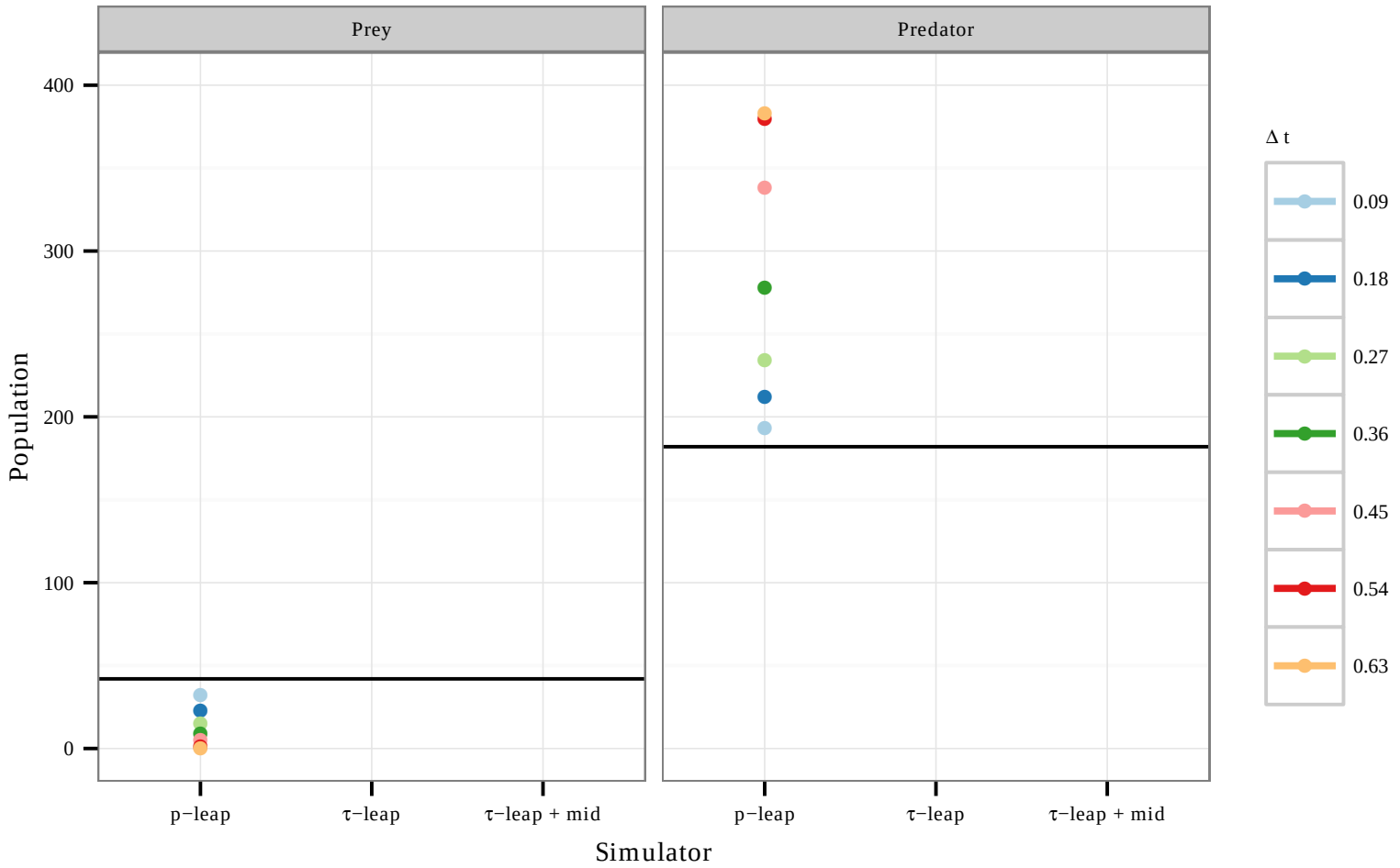
We estimate the mid-point to be:

$$x' \equiv x_0 - \lfloor 0.5\tau\mu x_0 \rfloor$$

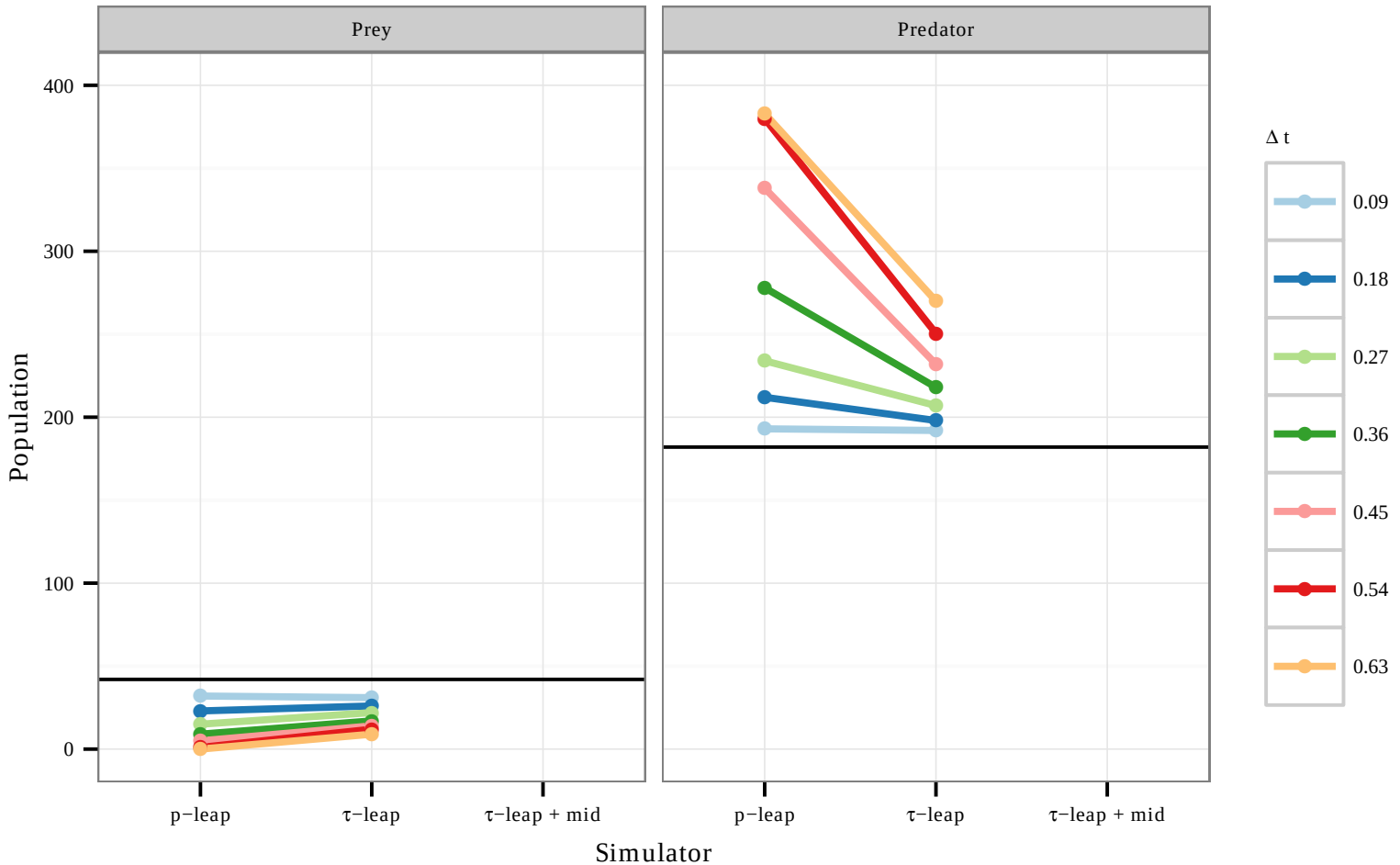
so the number of reactions executed is

$$P_p(k; \mu x_0, \tau) = \frac{e^{\mu x'\tau}(\mu x'\tau)^k}{k!} \quad \text{for } k = 0, \ldots$$

Example: Lotka-Volterra



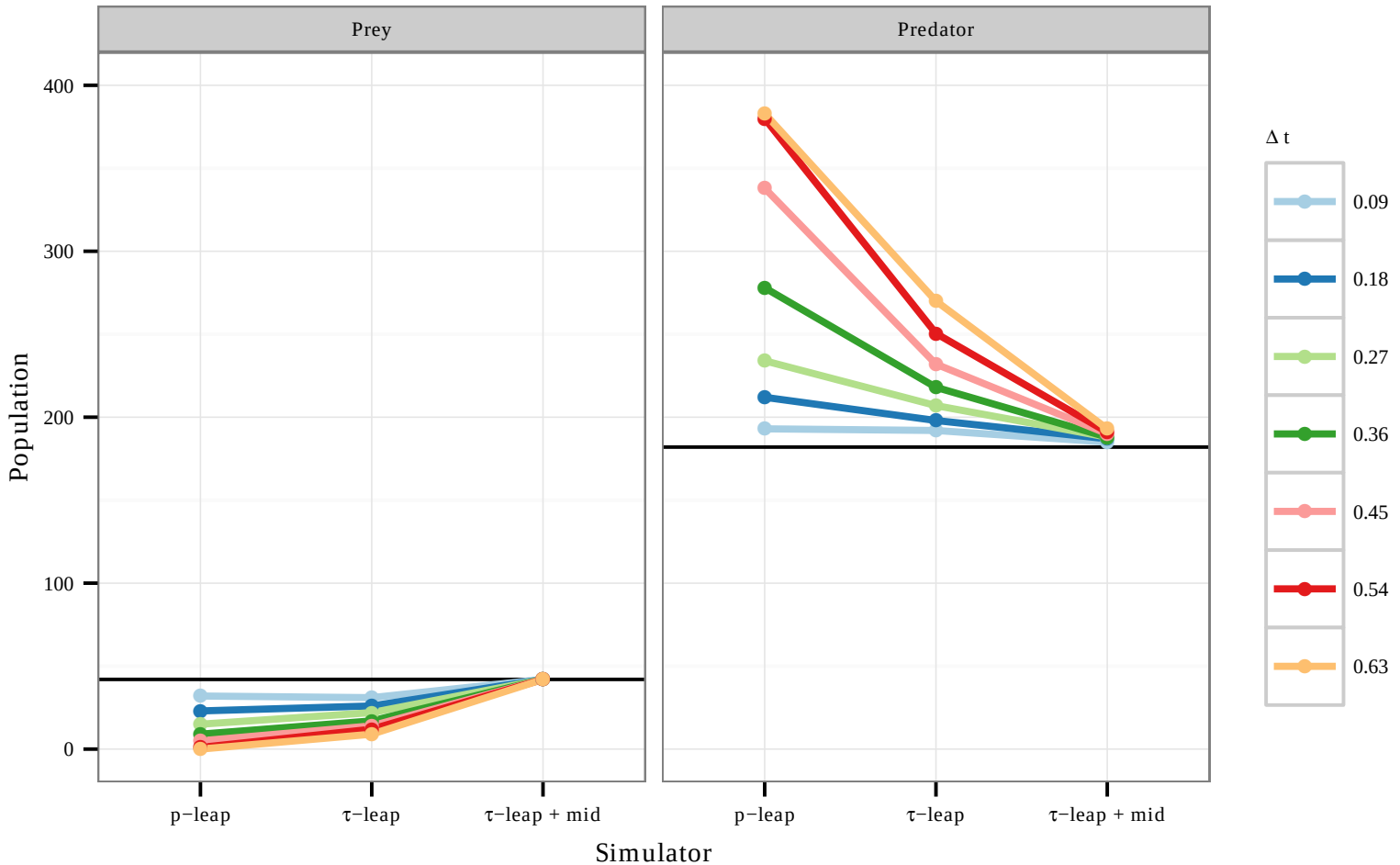
Example: Lotka-Volterra



For these parameter values and this model, we have the approximate relationship (obtained via simulation)

$$\epsilon \simeq 0.556 \times \Delta t$$

Example: Lotka-Volterra



For these parameter values and this model, we have the approximate relationship (obtained via simulation)

$$\epsilon \simeq 0.556 \times \Delta t$$

Summary

- Similar issues arise when solving SDEs with the Euler-Maruyama scheme
  - In fact, if we substitute Poisson with Gaussian random numbers in the p-leap scheme, we get the Euler-Maruyama algorithm
- Choosing a fixed Δt for a wide range of parameter combinations doesn't make sense
- Is it possible to these ideas when constructing bridges for SDEs?

References

- Gillespie DT. *Exact stochastic simulation of coupled chemical reactions*. The Journal of Physical Chemistry, 1977, **81**: 2340 – 2361. Direct method
- Gillespie, DT. *Approximate accelerated stochastic simulation of chemically reacting systems.* The Journal of Chemical Physics, 2001, **115**: 1716. τ-leap method
- Sandmann, W. *Streamlined formulation of adaptive explicit-implicit tau-leaping with automatic tau selection.* Simulation Conference (WSC), Proceedings of the 2009 Winter. IEEE, 2009. Nice overview of the various τ-leap flavours.
- Golightly, A & Gillespie, CS. *Simulation of Stochastic Kinetic Models.* In Silico Systems Biology. Humana Press, 2013, 169 – 187. Overview of various SSA.
  - `https://github.com/csgillespie/In-silico-Systems-Biology`