Perfect Sampling
1. Coupling from the past

Duncan Murdoch
University of Western Ontario

Outline

1. Background: MCMC
2. Coupling from the past
   (a) The basic algorithm
   (b) How long does it take?
   (c) How to use CFTP?
3. Tricks of the trade
   (a) Monotonicity
   (b) Unbounded state spaces: compactification, mixing with an independence sampler, dominated CFTP
   (c) Continuous state spaces: gamma coupling, shift coupling

1. Background: MCMC

- We want to study the distribution of $X \sim \pi(\cdot)$, $X \in \chi$.
- If we could simulate i.i.d. values $X_i \sim \pi(\cdot)$, $i = 1, \ldots, n$, then we could approximate quantities like $E(f(X))$ by $(1/n)\sum_{i=1}^{n} f(X_i)$. The law of large numbers (LLN) gets us convergence; the central limit theorem (CLT) gives us a distribution.
- We don't know how to simulate from $\pi(\cdot)$ directly, but we do know how to sample from a Markov chain X_t, whose steady-state distribution is $\pi(\cdot)$. See (Gilks, Richardson and Spiegelhalter, 1996) for lots of details and good advice.

- Creating the Markov chain is surprisingly easy: there are a large number of recipes: Metropolis-Hastings, Gibbs sampler, slice sampler, hit and run algorithm, etc.
- If we have *irreducibility* (the chain can't get permanently caught in a subset of the state space) and *positive [or Harris] recurrence* (the chain is guaranteed to return to [a neighbourhood of] a state in finite expected time), then a LLN works: averages converge. If the chain is also *aperiodic* (return times to a state are not necessarily at multiples of some cycle length), then the distribution of X_t converges to $\pi(\cdot)$, and averages converge to expectations with respect to $\pi(\cdot)$. We call such chains *ergodic*.

- We will be dealing with ergodic chains. These are the basis of a large amount of recent effort in Bayesian statistics (where MCMC is often the easiest way to approximate properties of posterior distributions), and other areas such as statistical physics (where MCMC is used to do difficult high-dimensional integrals).
- Problem: The distribution of the X_t values converges to $\pi(\cdot)$, and averages converge to expectation w.r.t. $\pi(\cdot)$, but how quickly? If we haven't converged, we have an *initialization bias*.
- Theory exists to address this, but it is very difficult to apply in practice.
- Solution: use perfect sampling to generate a sample from the limiting distribution of the chain.

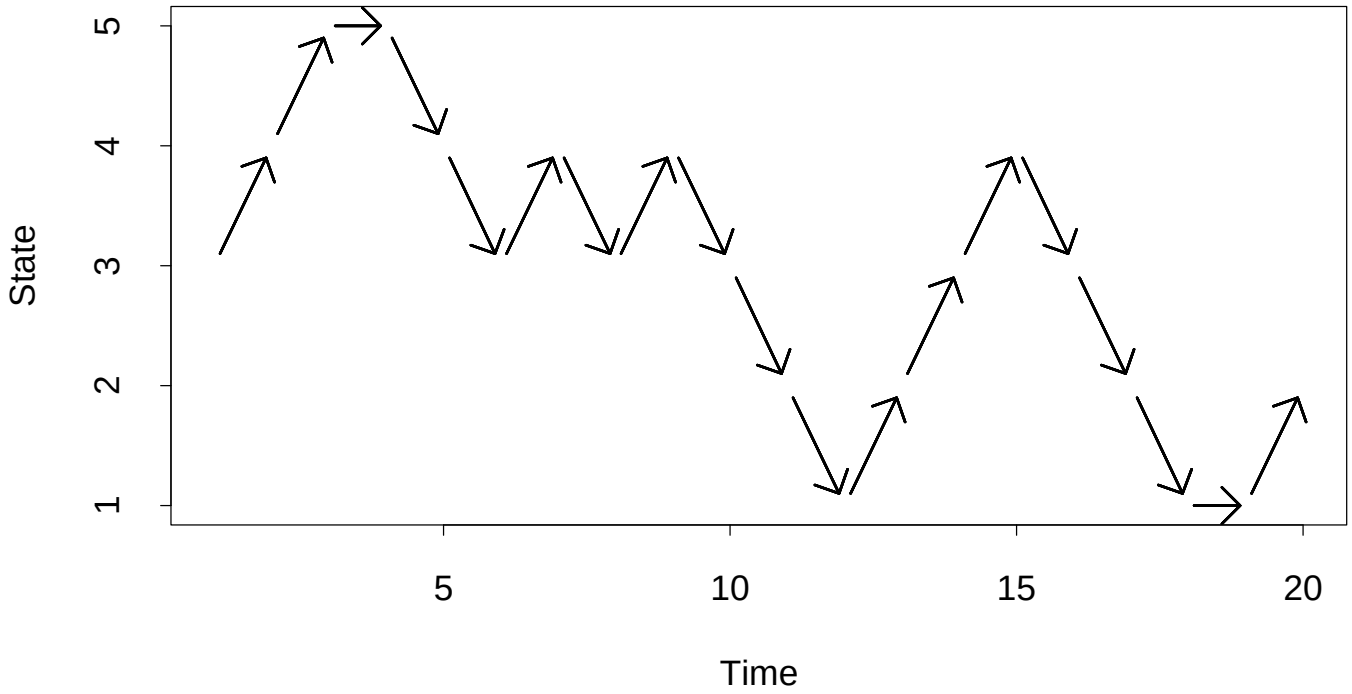
2. Coupling from the past (Propp and Wilson, 1996)

- Idea: Compute the result of an infinitely long run from the past by coupling all possible tails of shorter runs. If they all give the same answer, it must be in steady-state!
- We describe CFTP in terms of stochastic recursive sequences. Write our Markov chain as $X_{t+1} = \phi(X_t, U_{t+1})$, where U_t is an i.i.d. sequence from some distribution, and $\phi(\cdot, \cdot)$ is a fixed function. U_t may be a single random number, or a complex random structure, depending on the particular Markov chain being described.

- I find it useful to think of $\phi(\cdot, \cdot)$ as the computer program used to write a simulation of the Markov chain. U_t then represents all of the output of the computer's pseudo-random number generator necessary to update the state.

Example: Random walk

$$X_{t+1} = \begin{cases} \min(X_t + 1, 5) & \text{w.p. } 1/2 \\ \max(X_t - 1, 1) & \text{w.p. } 1/2 \end{cases}$$



Stochastic recursive sequence

Write $X_{t+1} = \phi(X_t, U_{t+1})$, where

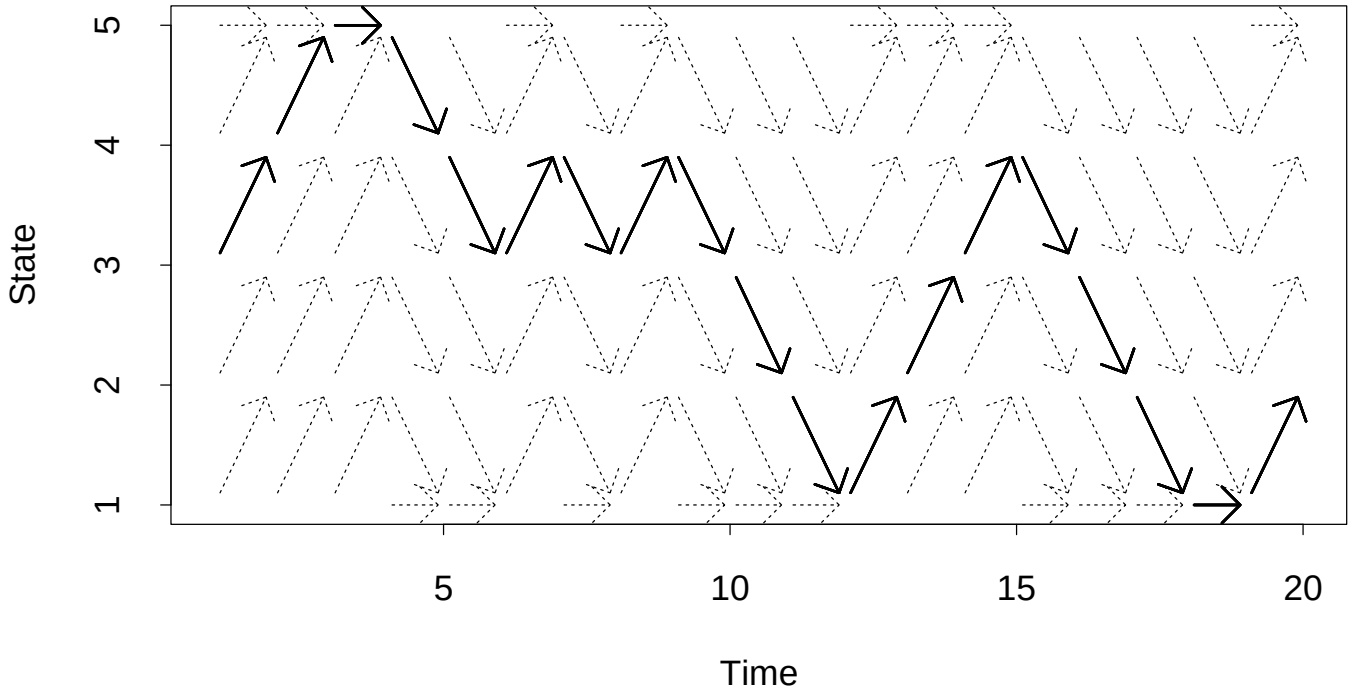
$$\begin{aligned} \phi(x, u) &= \min[\max(x + u, 5), 1] \\ U_t &= \pm 1 \text{ (with equal probability)} \end{aligned}$$

Then if the U_t values are i.i.d., X_t follows the random walk Markov chain.

This is a random walk Metropolis sampler whose steady-state distribution $\pi(\cdot)$ is the uniform distribution on the set $\{1, 2, 3, 4, 5\}$.

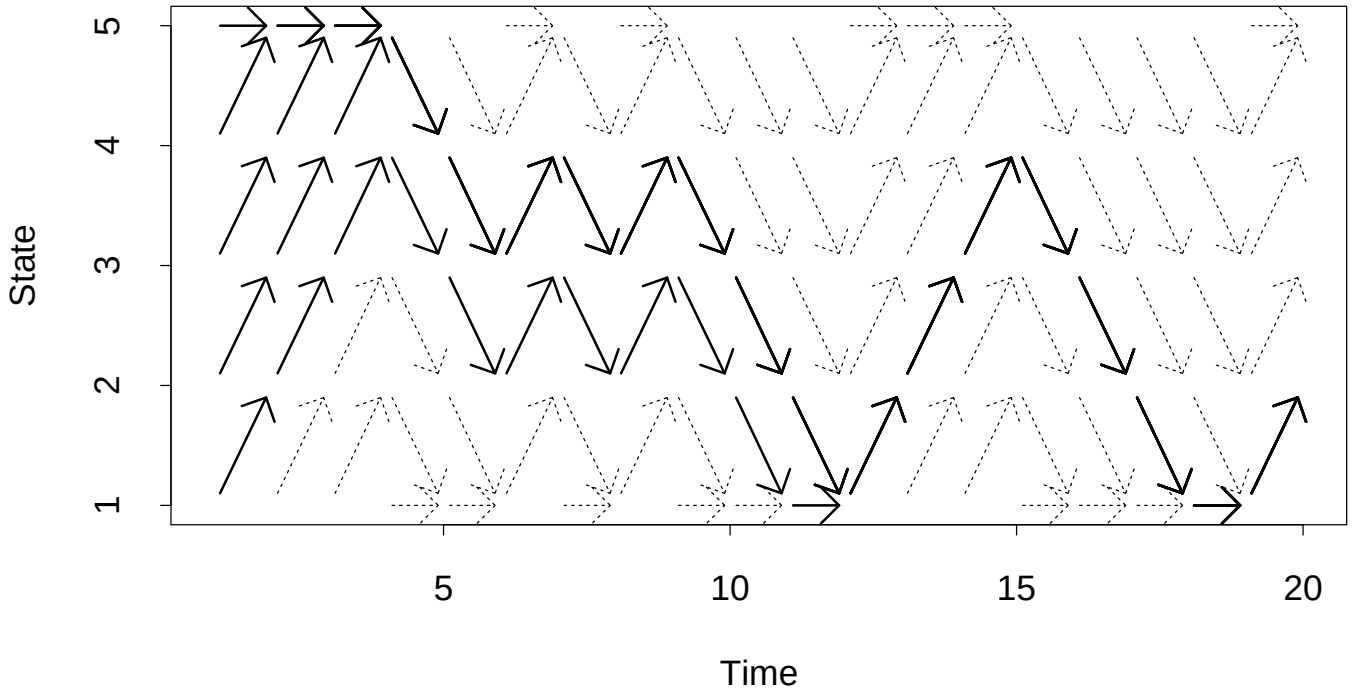
Coupling

This representation allows us to imagine paths that were not sampled. We can imagine that the U_t sequence is fixed, and $\phi(x, U_{t+1})$ is a function that may be applied to any value x. We get a number of "virtual chains" starting from different starting points.



Coupling continued ...

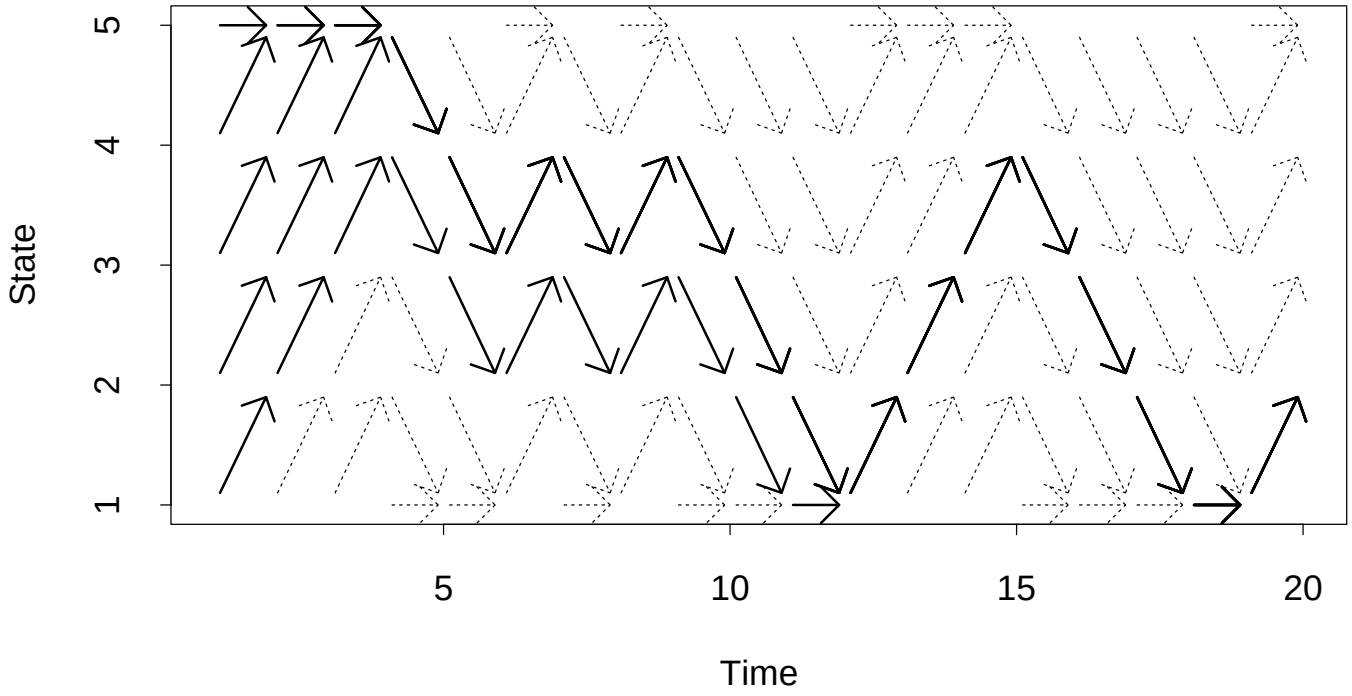
With many update functions, paths *coalesce*: regardless of the initial state, the value of X_t is the same for large enough t. The past is forgotten; no initialization bias remains.



There may still be a *coupling time bias*.

Coupling time bias

The *coupling time* is the time when all virtual chains coalesce into a single chain. A draw at this time is generally not a draw from $\pi(\cdot)$. In the random walk example, $\pi(\cdot)$ is uniform on all states, but X_t is always 1 or 5 at the coupling time.

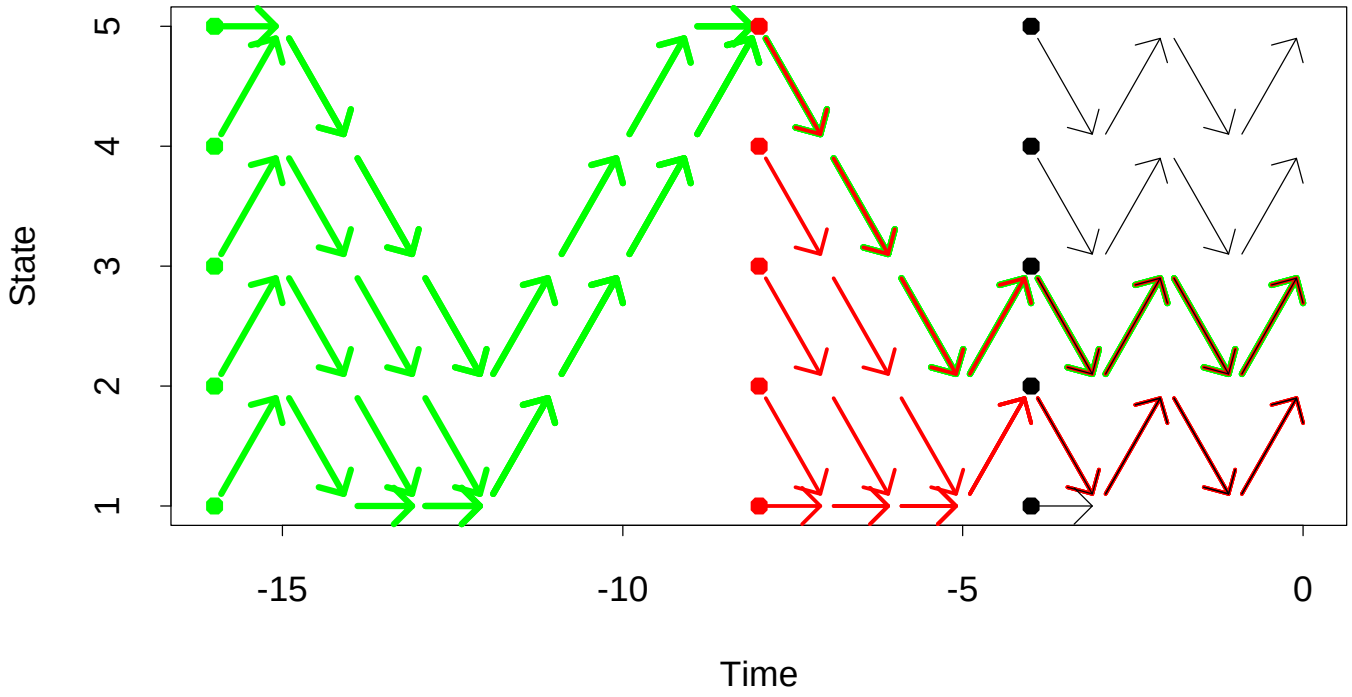


Coupling from the past (CFTP)

To avoid the coupling time bias, choose the observation time *before* testing for coalescence: compute the result of an infinitely long run from the past by coupling all possible tails of shorter runs.

Specifically: To draw X_0, start all virtual chains at time $t = -1$. If they coalesced, then starting at any more distant point would pass through one of those states and give the same final answer. If they did not, then try again from $t = -2, -3, -4, \ldots$. (In practice, it is somewhat more efficient to double the distance each time, i.e. use $t = -1, -2, -4, \ldots$ to search for coalescence).

When doing the search, it is essential that U_t values be treated as fixed once they are generated.



If U_t values are regenerated at each iteration, the coupling time bias would come back: results would be biased in favour of values resulting from shorter coupling times.

CFTP(T):

$t \leftarrow -T$	*Try from* $-T$
$B_t \leftarrow \chi$	*All states possible*
while $t < 0$	*Run to* $t = 0$
$\quad B_{t+1} \leftarrow \phi(B_t, U_{t+1})$	*Update* B_t
$\quad t \leftarrow t + 1$	*Update* t
if $\#B_0 = 1$ then	*Coalesced?*
$\quad$ return(B_0)	*Yes, we're done!*
else	
$\quad$ CFTP($2T$)	*No, try from* $-2T$

How long does it take?

Let T^* be the minimum value of T for which CFTP terminates for a given U_t sequence. Then Propp and Wilson (1996) show that $E(T^*)$ can not be much smaller than the "mixing time" of the chain. It could be much larger if a poor coupling is chosen. (The best couplings are monotone ones; see below.)

If $P(T^* \leq t) \geq p$, then $E(T^*) \leq t/p$ (because each block of t steps can be treated as an independent opportunity for coalescence).

By the same argument, the distribution of T^* has at worst a geometrically decaying upper tail.

How to use CFTP (Murdoch and Rosenthal, 1999)

- Each generated value takes a **lot** of computation.
- For efficiency, only generate a small number (e.g. 25) using CFTP; use standard MCMC with these as starting values.
- Run your MCMC chains as long as necessary to reduce variance.
- The 25 independent chain averages are usually enough to get good confidence intervals for $E(f(X))$.
- If $f(\cdot)$ is multi-dimensional, we'll probably need more chains.

3. What is involved in doing CFTP?

- Either $\pi(\cdot)$ or the Markov chain is given.
- Whether or not the Markov chain is given, we have some flexibility in its specification.
- The coupling is up to us.
- Detecting coalescence is up to us.

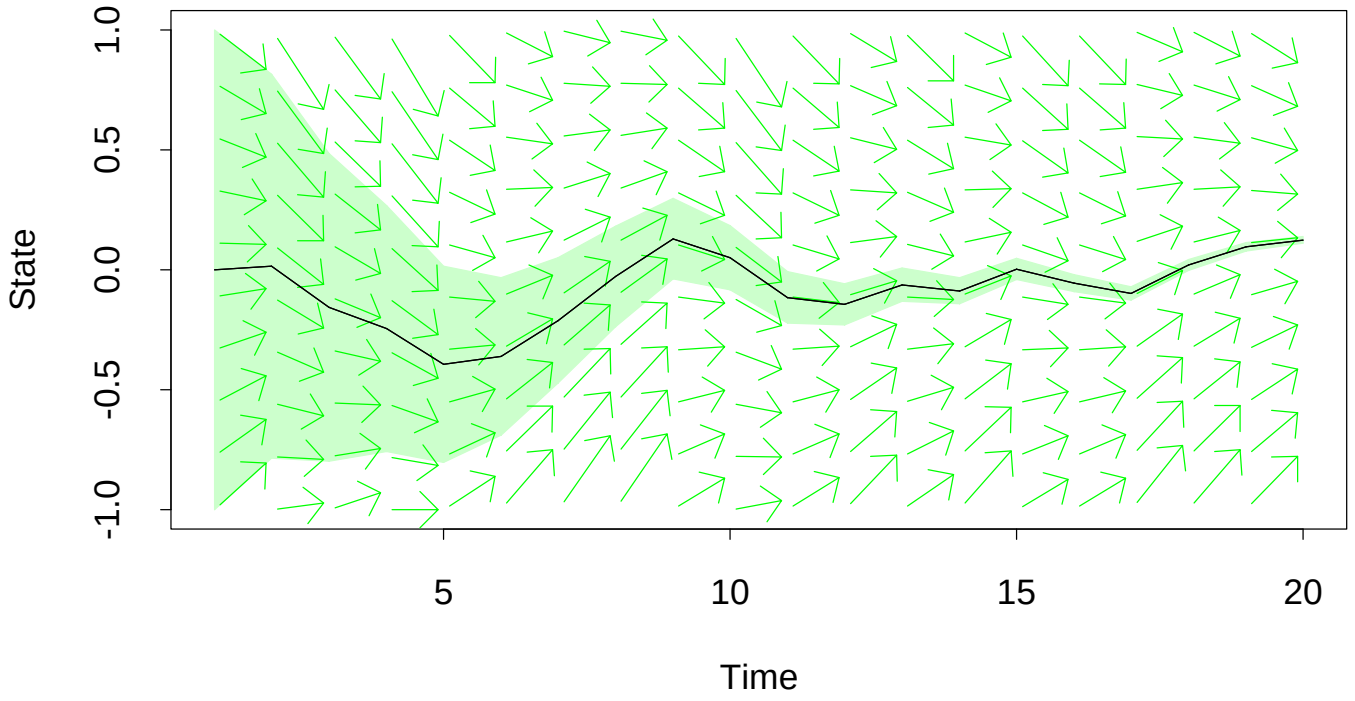
Choosing a coupling

- Need to write $X_{t+1} = \phi(X_t, U_{t+1})$, with U_t i.i.d.
- Want coalescence. This can be tricky when the state space χ is continuous.
- Want easy coalescence detection.

Monotonicity

- It's generally too time consuming to follow every path.
- *Monotonicity* allows sandwiching:

$$X_t \leq Y_t \Rightarrow \phi(X_t, U_{t+1}) \leq \phi(Y_t, U_{t+1})$$



In general, we don't need a linear order. A partial order with

$$X_t \preceq Y_t \Rightarrow \phi(X_t, U_{t+1}) \preceq \phi(Y_t, U_{t+1})$$

allows coalescence detection if we follow paths started at all minimal and maximal elements.

E.g. χ contains p-tuples $(x_1, \ldots, x_p)$, where $x_i \in [0, 1]$. The componentwise partial order is $(x_1, \ldots, x_p) \preceq (y_1, \ldots, y_p)$ when $x_i \leq y_i$ for all $i = 1, \ldots, p$.

Then $(0, \ldots, 0)$ is a minimal element and $(1, \ldots, 1)$ is a maximal element. If we have a monotone coupling and follow paths started from those two points, all other points will be sandwiched between.

Other characteristics can substitute for monotonicity.

Anti-monotone chains satisfy

$$X_t \preceq Y_t \Rightarrow \phi(X_t, U_{t+1}) \succeq \phi(Y_t, U_{t+1})$$

Anything that allows us to update B_t (or reasonably tight bounds on B_t) is good enough.

Compactification

In an unbounded state space χ, CFTP may never terminate, and monotone versions may have no minimal and maximal elements.

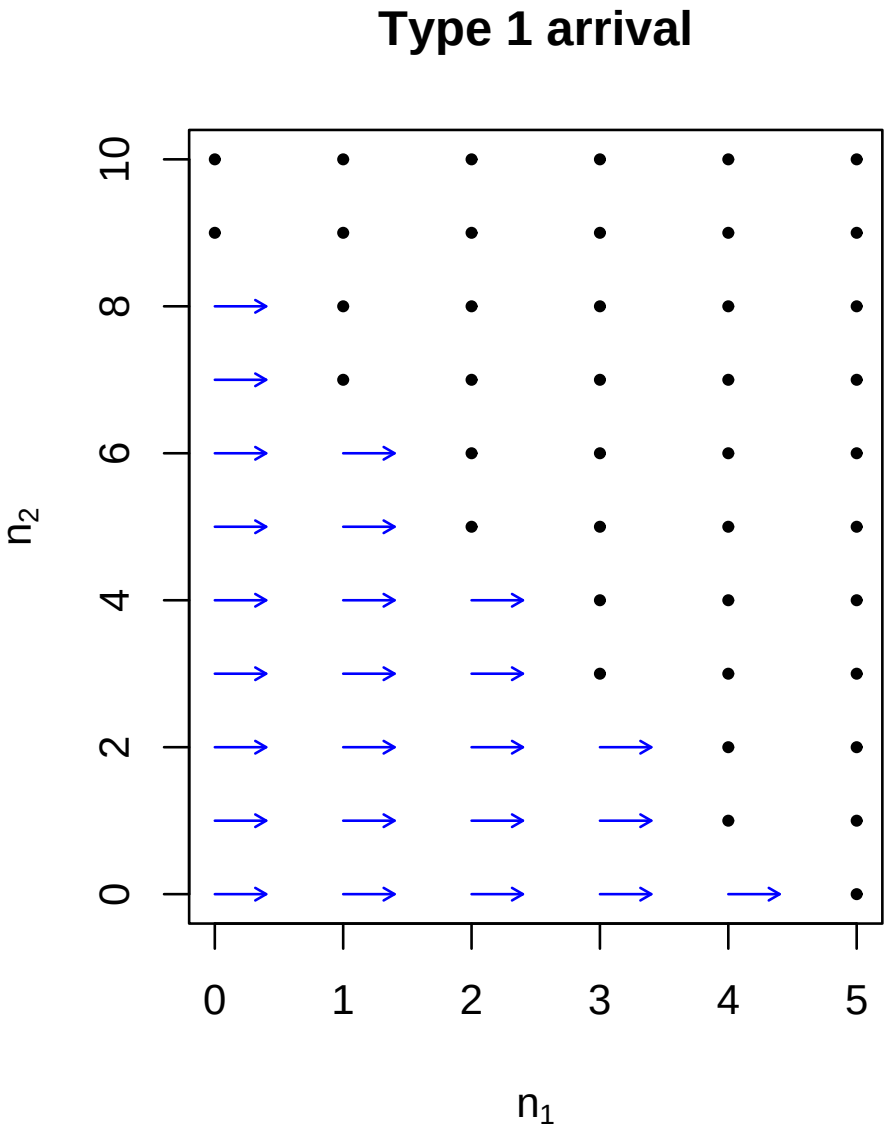
One solution is to artificial "compactify" the state space:

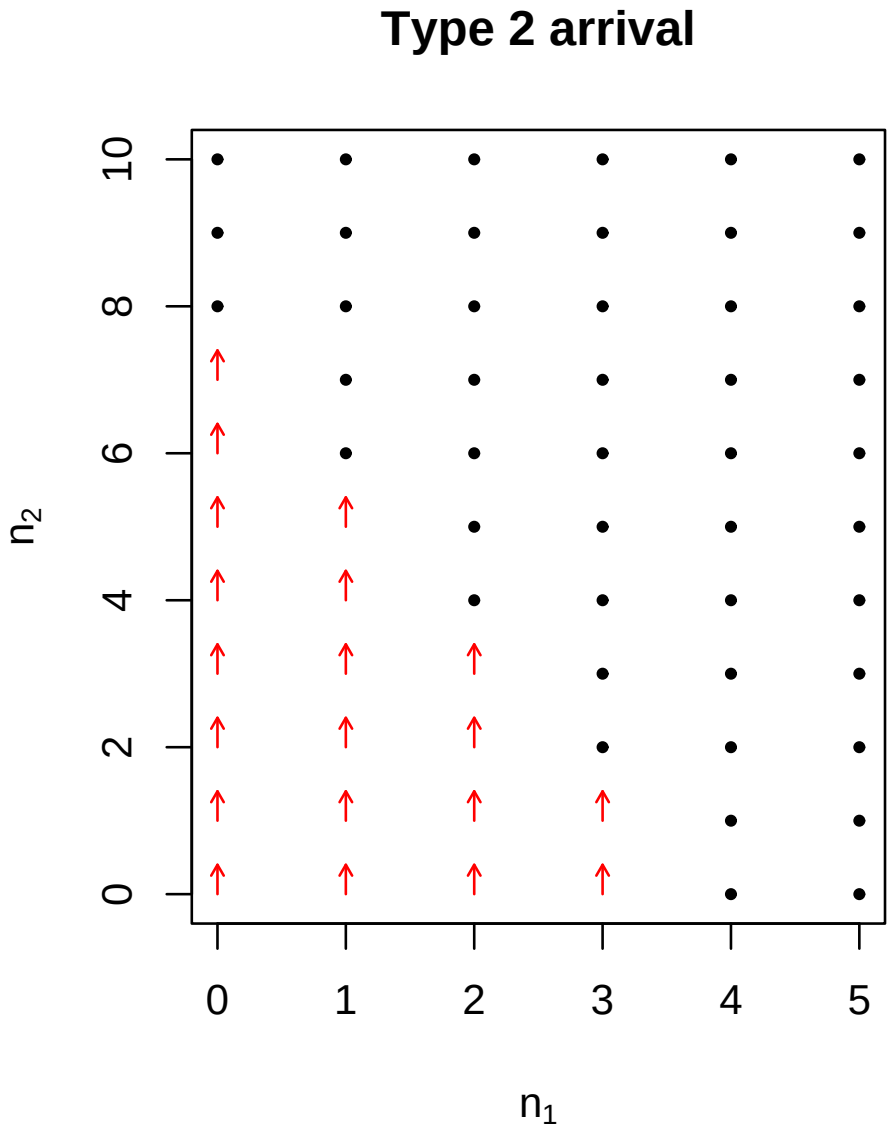
1. Add artificial points $\pm\infty$ to χ.
2. Define the transitions from these points into χ in such a way that we retain monotonicity.

E.g. In a queue, let $(n_1, \ldots, n_p)$ be the counts of customers of various types, with an overall bound of $\sum b_i n_i + r_i < C$, where $b_1, \ldots, b_p$ and $r_1, \ldots, r_p$ are constants.

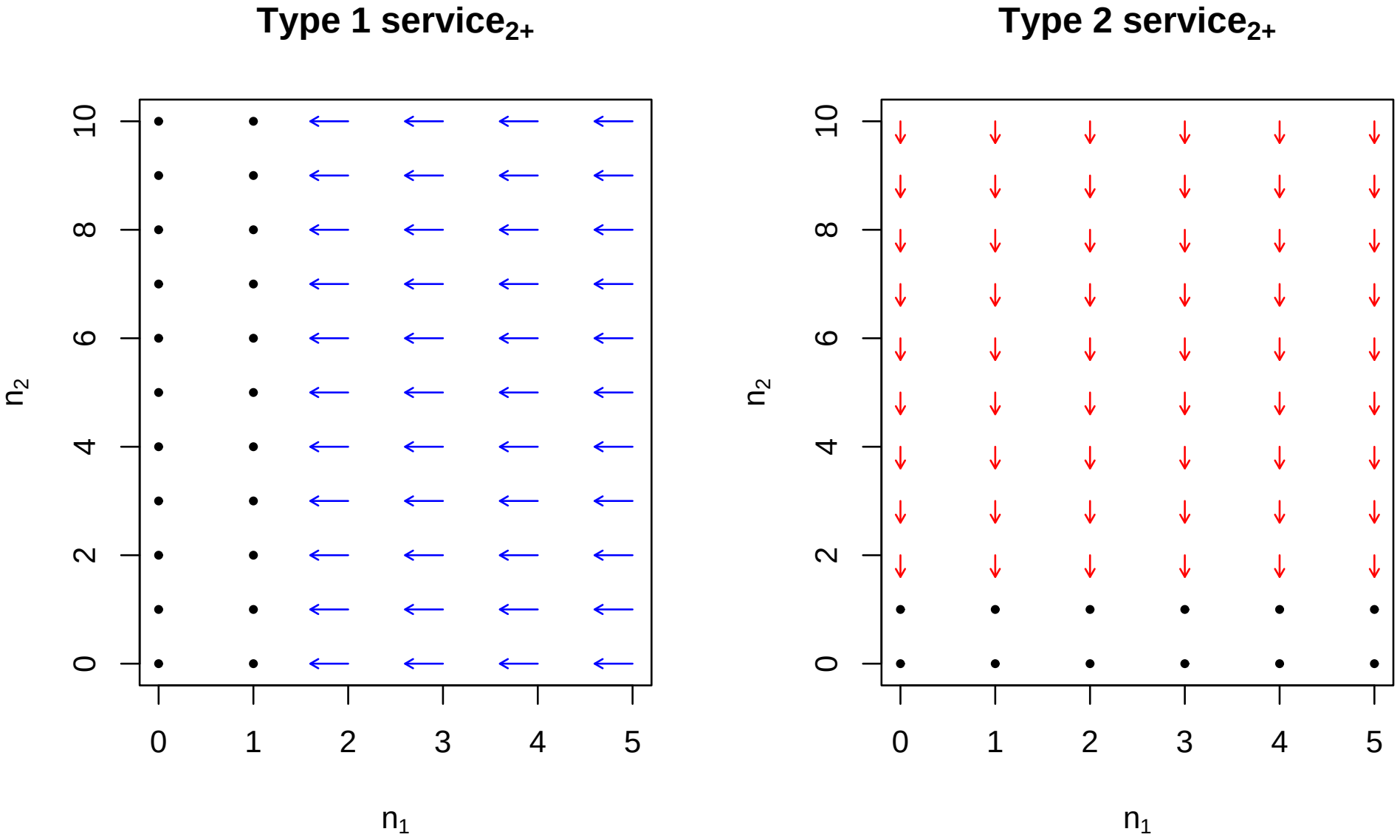
The point $(0, \ldots, 0)$ is minimal in the componentwise partial order. There is no single maximal element. There are only a finite number of elements, but if p and C are large it is inconvenient to follow all maximal elements.

However, filling the rectangle out to $+\infty = (B, \ldots, B)$ with transient states may allow a simple simulation.

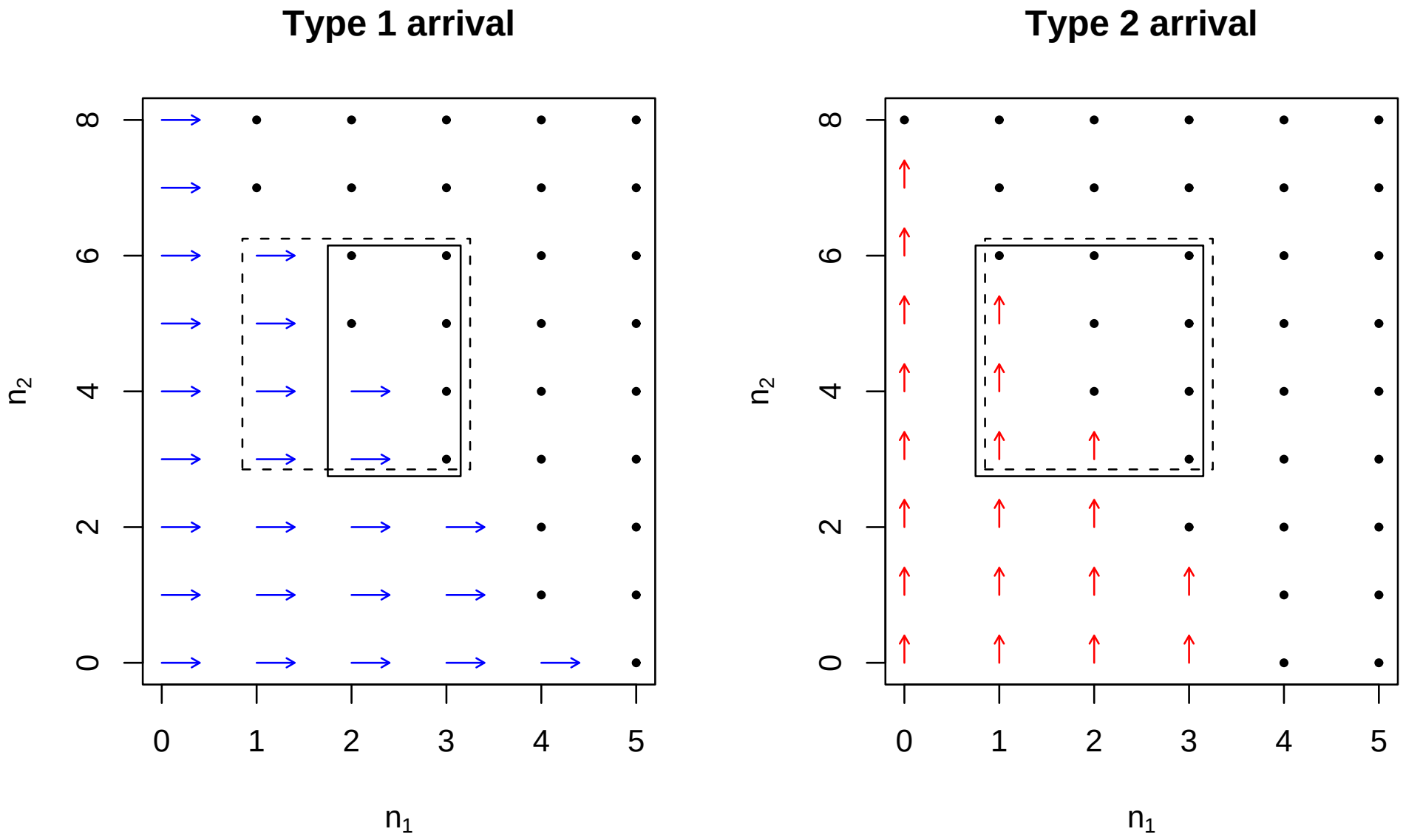

Type 1 arrival
n2
n1
0
2
4
6
8
10
0
1
2
3
4
5


Type 2 arrival
n2
n1
0
2
4
6
8
10
0
1
2
3
4
5

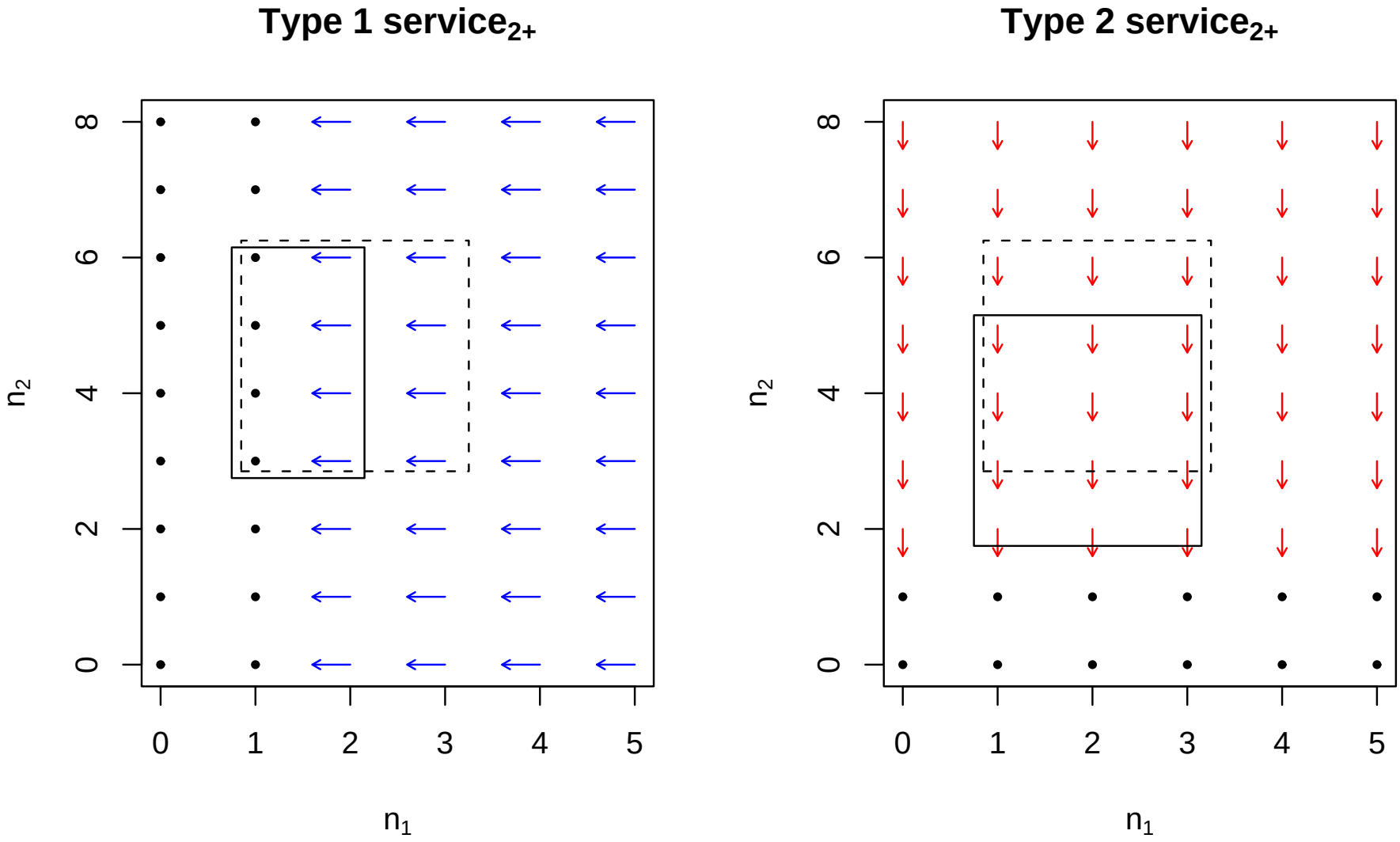
$$C = 10, b = (2, 1), r = (0, 2)$$



$$C = 10, b = (2, 1), r = (0, 2)$$



$$C = 10, b = (2, 1), r = (0, 2)$$



$$C = 10, b = (2, 1), r = (0, 2)$$

Random walk Metropolis

Consider using the random walk Metropolis algorithm on the whole real line. Given X_t we calculate a proposal $Y = X_t + Z$, where Z is a draw from a symmetric distribution (e.g. $N(0,1)$). We also draw $U \sim \text{Unif}(0,1)$. Then

$$X_{t+1} = \begin{cases} Y & \text{if } U < \pi(Y)/\pi(X_t) \\ X_t & \text{otherwise} \end{cases}$$

If B_t is the whole real line, then this sampler won't shrink it. CFTP will fail every time.

Mixing with an "independence sampler"

An independence sampler is like random walk Metropolis, but Y is drawn from a proposal distribution $p(\cdot)$, independent of X_t.

We accept Y when

$$U < \frac{\pi(Y)/p(Y)}{\pi(X_t)/p(X_t)}$$

If we choose $p(\cdot)$ with heavier tails than $\pi(\cdot)$, the ratio is large when X_t is sufficiently far out in the tails: B_t is reduced to a compact interval of values.

Independence samplers are usually not very good for MCMC, but we get their benefit by using them in combination with a better sampler.

Dominated CFTP

Another approach due to Kendall (1998) for unbounded chains is as follows. Couple X_t to a "dominating process" D_t that has the property $X_t \preceq D_t \Rightarrow X_{t+1} \preceq D_{t+1}$. We also need to know that in the $t \to \infty$ limit, $X_t \preceq D_t$ will eventually be true.

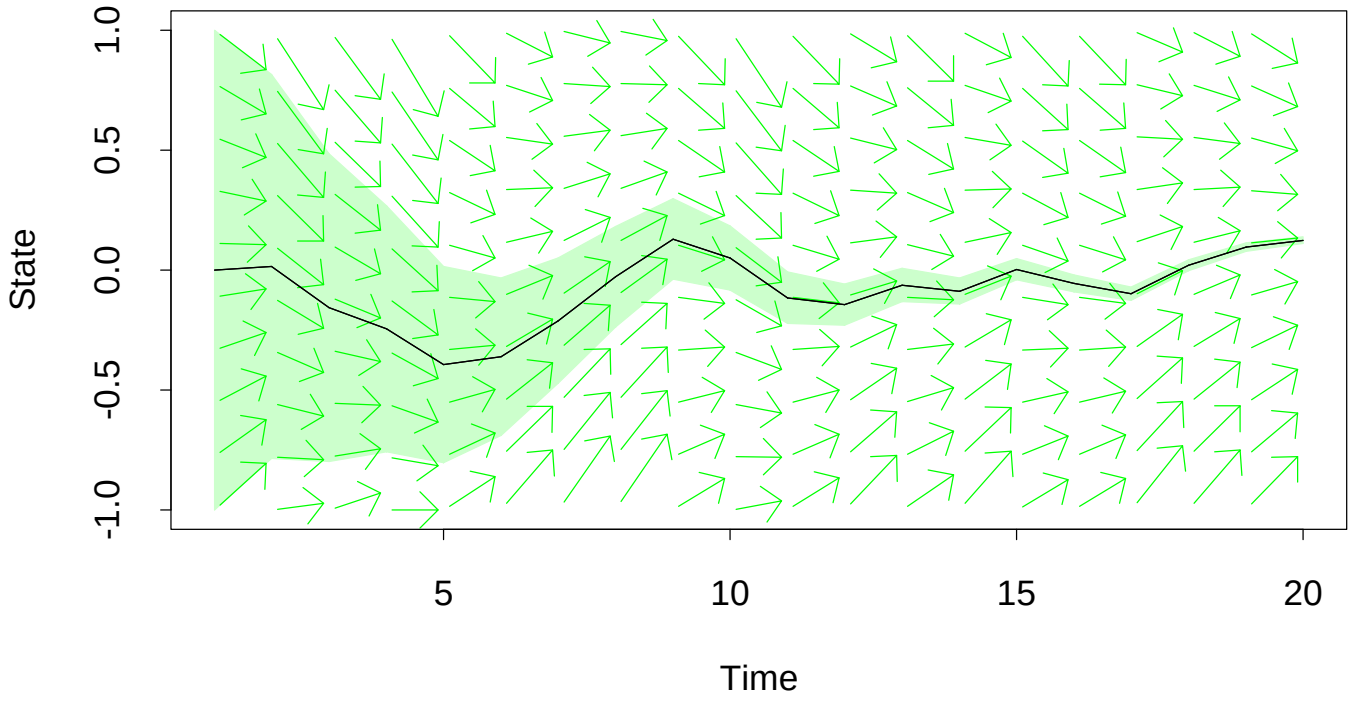
Then we can use D_t as a random upper bound on X_t as follows. Simulate D_0 at steady-state, and $D_{-1}, D_{-2}, \ldots, D_{-T}$ by running D_t in reverse time. Then we can start CFTP from time $-T$ using $B_{-T} = \{x : x \preceq D_{-T}\}$, and run forwards, coupled to the already-simulated D_t values.

More on reverse time simulation in the next lecture.

Continuous state spaces

When χ is continuous, we have special problems. Simple couplers will never coalesce. E.g.

$$\begin{aligned} X_{t+1} &= 0.8X_t + U_{t+1} \\ U_{t+1} &\sim \text{Unif}(-0.2, 0.2) \end{aligned}$$

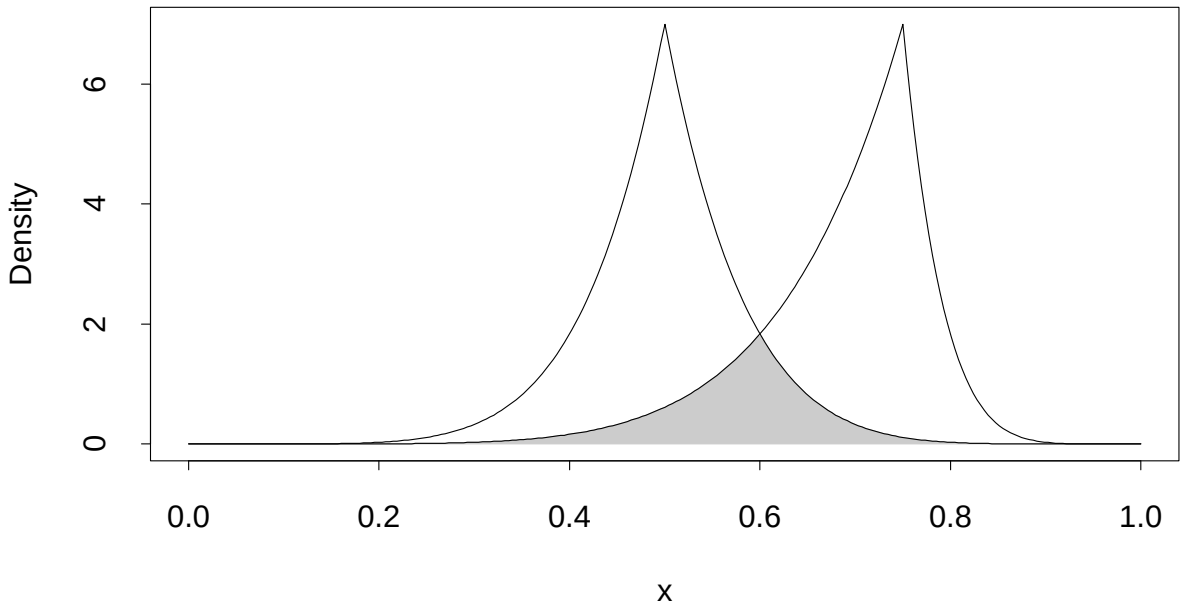


Gamma Coupling

Suppose $X_{t+1} \sim f(\cdot|X_t)$, and $f(\cdot|X_t)$ is either $f_0(\cdot)$ or $f_1(\cdot)$; set $r(y) = \min(f_0(y), f_1(y))$, $\rho = \int r(y)dy$.

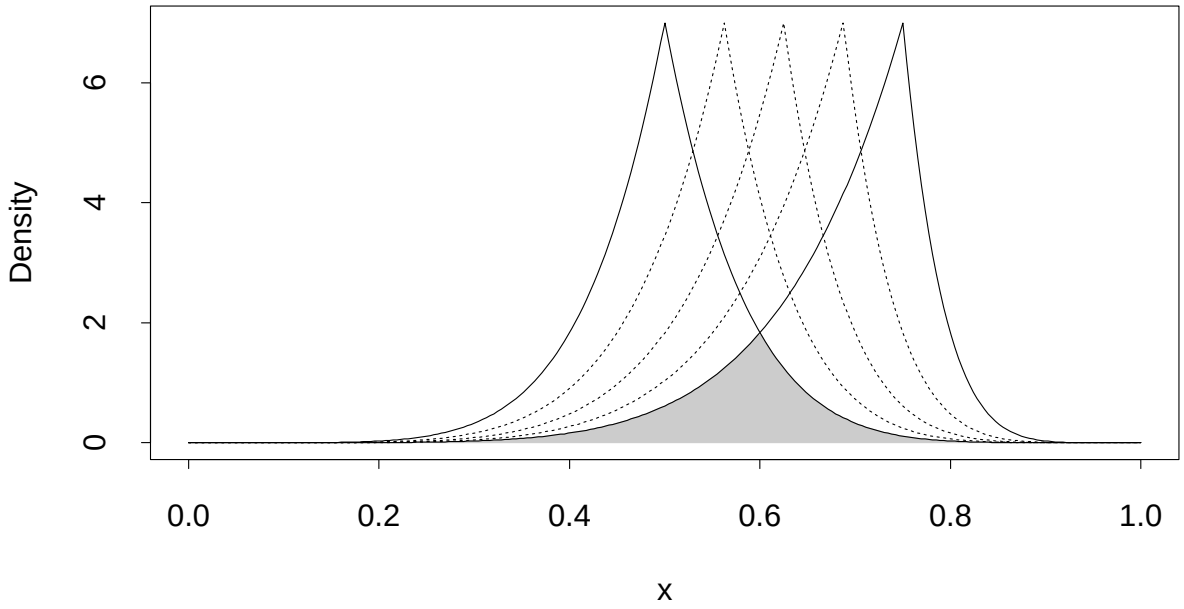
1. Draw Y from $f_0(\cdot)$, U from Unif$(0, 1)$.
2. If $U f_0(Y) < r(Y)$, set X_{t+1} to Y.
3. Otherwise, draw X_{t+1} from $[f(\cdot|X_t) - r(\cdot)]/(1 - \rho)$.

E.g. $f(y|x) \propto \min[y/x, (1 - y)/(1 - x)]^6$



The Multigamma Coupler (Murdoch and Green, 1998)

If $f(y|x) \geq r(y)$ for all x, y, we can do gamma coupling over a whole continuous range of possible prior states X_t simultaneously. With probability ρ, they all coalesce to the same update X_{t+1}.

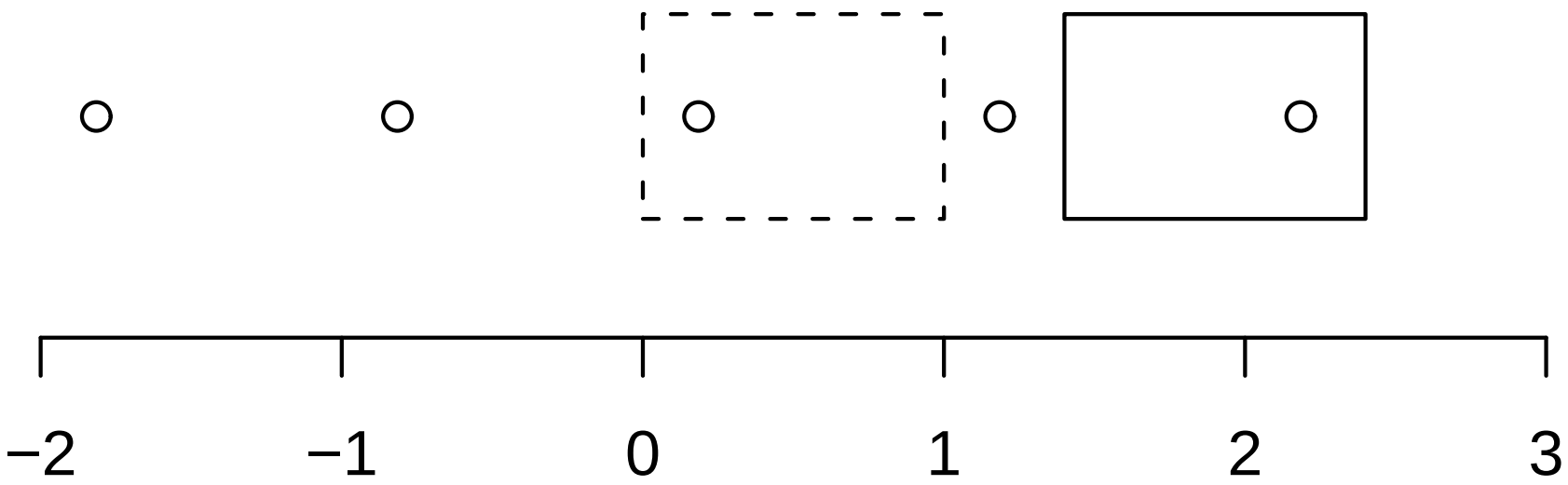


Wilson's Multishift Coupler

Wilson (2000) invented a very clever coupler for the situation where the conditional densities of $f(\cdot|X_t)$ form a location family.

E.g. To get $\phi(a, \cdot) \sim \text{Unif}(a, a+1)$ for all a:

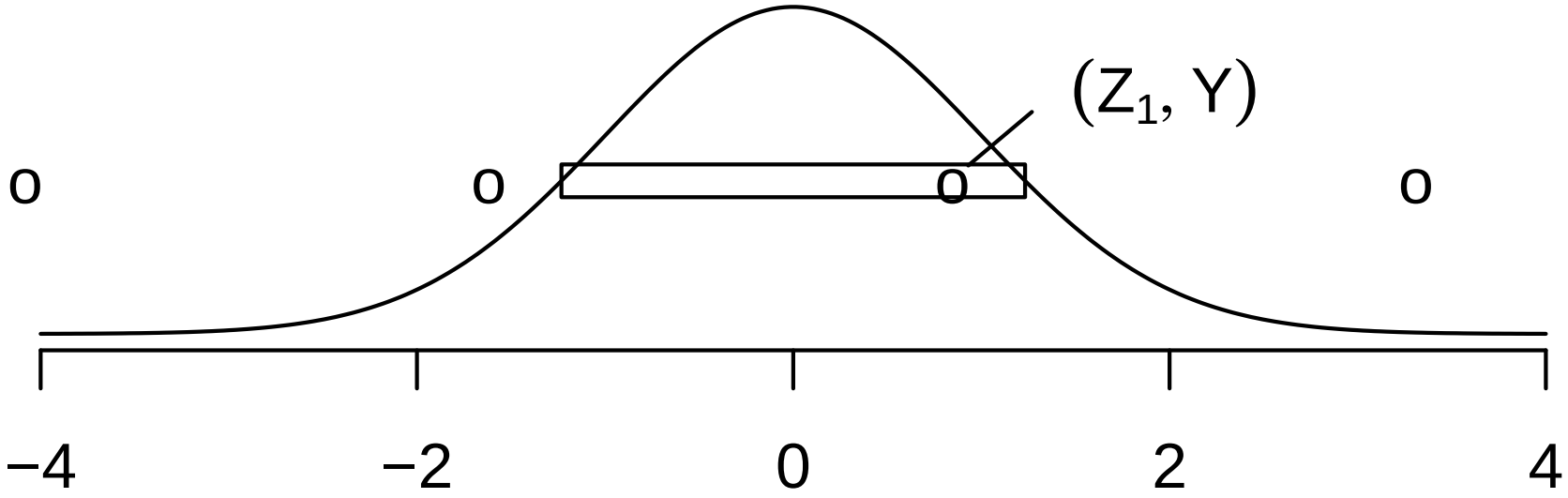
Sample $U \sim \text{Unif}(0, 1)$, then set $\phi(a, U) = U + k$, where k is an integer chosen so that $U + k \in (a, a+1]$.



Extension to location families

To extend the multishift coupler to general location families, construct them as a continuous mixture of uniforms.

E.g. Choose $Z_1 \sim N(0,1)$, $Y \sim \text{Unif}(0, \phi(Z_1))$, then condition on Y and proceed as before.



References

Gilks, W. R., Richardson, S., and Spiegelhalter, D. J., editors (1996). *Markov Chain Monte Carlo in Practice*. Chapman and Hall.

Kendall, W. (1998). Perfect simulation for the area-interaction point process. In Accardi, L. and Heyde, C. C., editors, *Probability Towards 2000*, pages 218–234. Springer, New York.

Murdoch, D. J. and Green, P. J. (1998). Exact sampling from a continuous state space. *Scandinavian Journal of Statistics*, 25:483–502.

Murdoch, D. J. and Rosenthal, J. S. (1999). Efficient use of exact samples. *Statistics and Computing*, 10:237–243.

Propp, J. G. and Wilson, D. B. (1996). Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures and Algorithms*, 9:223–252.

Wilson, D. (2000). Layered multishift coupling for use in perfect sampling algorithms (with a primer on CFTP). In Madras, N., editor, *Monte Carlo Methods—Fields Institute Communications Vol. 26*. AMS.

Also see David Wilson's web site
`http://dbwilson.com/exact/`.