Doeblin Trees

James Murphy

Joint work with:
François Baccelli and Mir-Omid Haji-Mirsadeghi

February 13, 2019

Sampling Stationary Distributions

Interested in a Markov chain:

- Finite state space S.
- Transition matrix P: irreducible, aperiodic, positive recurrent.
- Stationary distribution π.

Sampling Stationary Distributions

Interested in a Markov chain:

- Finite state space S.
- Transition matrix P: irreducible, aperiodic, positive recurrent.
- Stationary distribution π.

How to get a sample from π?

Naive Method

How to get a sample from π?

- Run MC for a very long time, return the result.

Naive Method

How to get a sample from π?

- Run MC for a very long time, return the result.

Problems:

- How long is "very long"?
  1 hundred steps?
  1 million steps?
  1 billion steps?
- How good is sample?

Naive Method

How to get a sample from π?

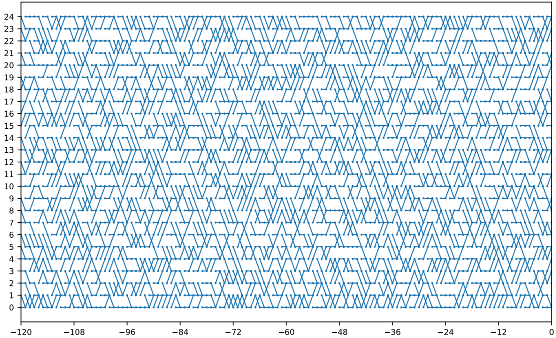
- Run MC for a very long time, return the result.

Problems:

- How long is "very long"?
  1 hundred steps?
  1 million steps?
  1 billion steps?
- How good is sample?

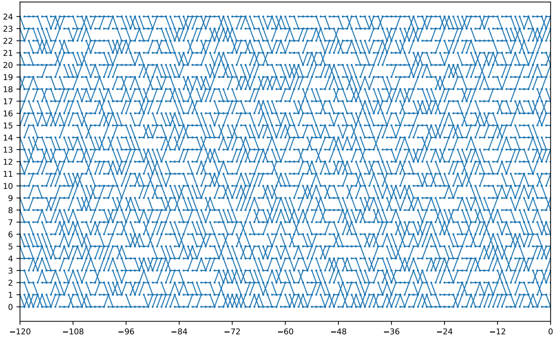
Can answer if mixing time is known, but often it is not.
Can we do better?

Coupling From the Past



- Imagine starting a copy of the MC from every possible state at every possible time.
  When two paths meet, they merge.
- Model this with a random graph on $\mathbb{Z} \times S$.

Coupling From the Past



- Every time t gets gets an independent source of randomness ξ_t to determine transitions to the next time.
- Fully independent case: ξ_t has a product distribution, i.e. $\xi_t = (\xi_{t,x})_x$ so every vertex (t, x) gets an independent source of randomness $\xi_{t,x}$.

Coupling From the Past

Specifically, transitions are determined by

$$(t, x) \mapsto (t + 1, h(x, \xi_t))$$

where h is chosen to satisfy

$$\mathbb{P}(h(x, \xi_0) = y) = P(x, y), \qquad x, y \in S.$$

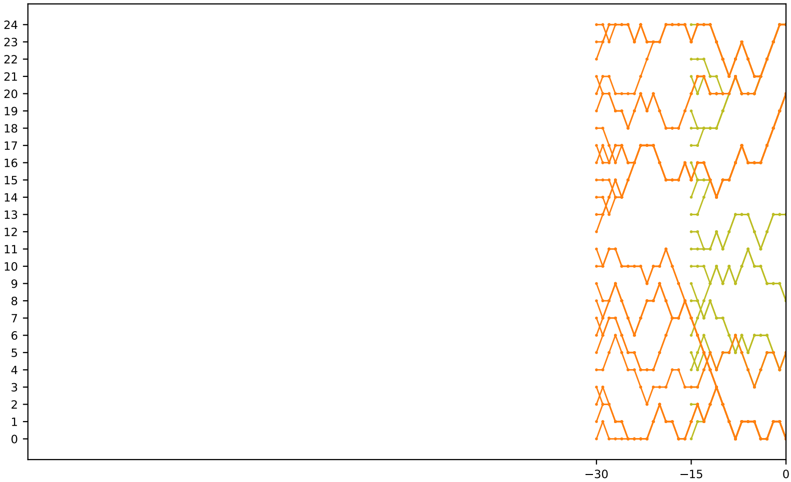
Coupling From the Past

Algorithm:

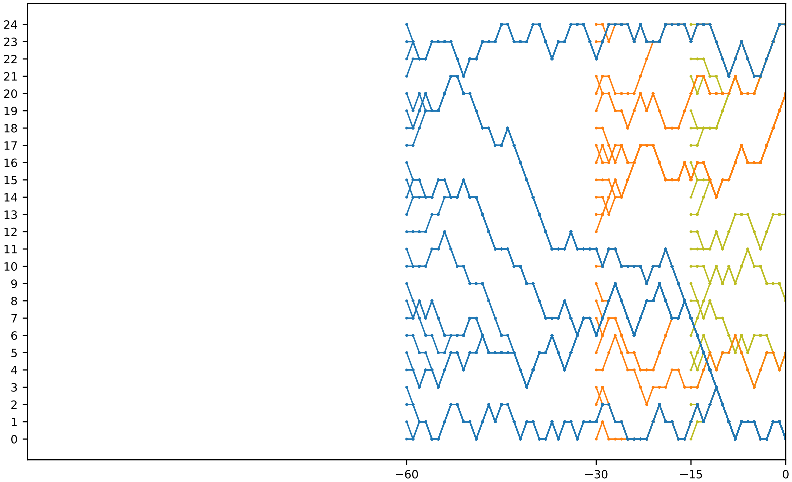
1. Simulate MC starting from all states at a given time $t < 0$.
2. Check if, at time 0, all states end up in the same place.
3. If so, return this value.
4. If not, start again twice as far back in time.

Coupling From the Past

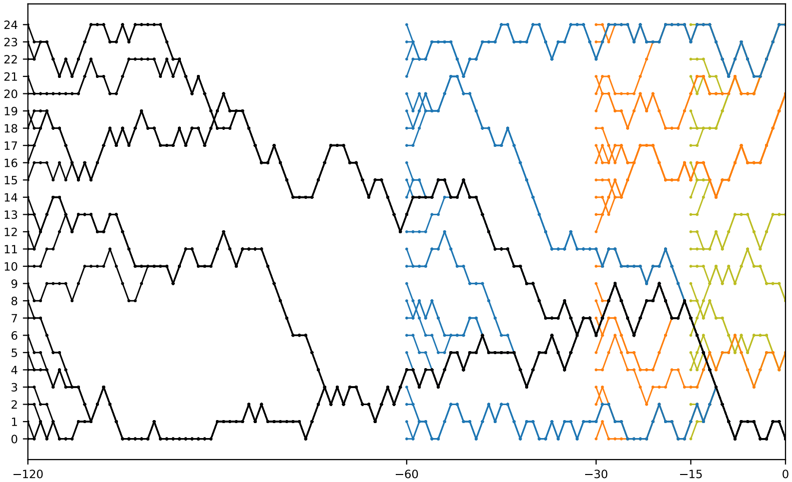
Coupling From the Past



Coupling From the Past



Coupling From the Past



Coupling From the Past

- CFTP returns a value $\beta_0 \sim \pi$.
  If the sim finished, it is a perfect sample.
  Could still take a long time.

Coupling From the Past

- CFTP returns a value $\beta_0 \sim \pi$.
  If the sim finished, it is a perfect sample.
  Could still take a long time.
- As described, requires space at least as big as number of states, but (anti)monotonicity can be used for some models to reduce space usage significantly.

Coupling From the Past

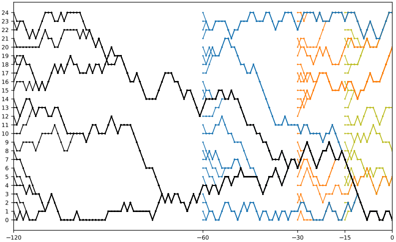
- CFTP returns a value $\beta_0 \sim \pi$.
  If the sim finished, it is a perfect sample.
  Could still take a long time.
- As described, requires space at least as big as number of states, but (anti)monotonicity can be used for some models to reduce space usage significantly.
- Huge effort spent to specialize CFTP to specific chains makes it good choice in many cases.

Taking a step back

But why is this β_0 a sample from π?

Taking a step back

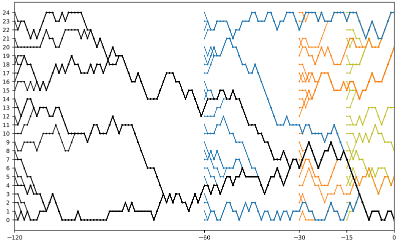
From time -120 to 0, all states collapsed to a single state.



IID structure in time implies this kind of collapse will happen infinitely often backwards in time.

Taking a step back

From time -120 to 0, all states collapsed to a single state.



IID structure in time implies this kind of collapse will happen infinitely often backwards in time.

Gives bi-infinite path $(\beta_t)_{t\in\mathbb{Z}}$ which is stationary version of the chain. CFTP returns β_0.

Doeblin Graphs

- Keep the same picture as CFTP, but study the graph itself.
- State space S allowed to be countably infinite.
- $(\xi_t)_t$ allowed to be stationary ergodic instead of IID, but for this talk assume IID.
- P still irreducible, aperiodic, positive recurrent.

Doeblin Graphs

- Keep the same picture as CFTP, but study the graph itself.
- State space S allowed to be countably infinite.
- $(\xi_t)_t$ allowed to be stationary ergodic instead of IID, but for this talk assume IID.
- P still irreducible, aperiodic, positive recurrent.

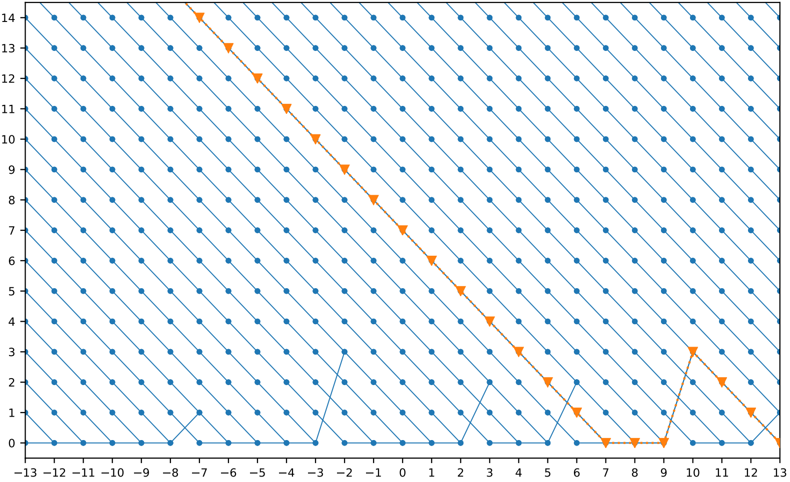
Main question:
What properties does this Doeblin Graph have?

Doeblin Graphs

Unique bi-infinite path?

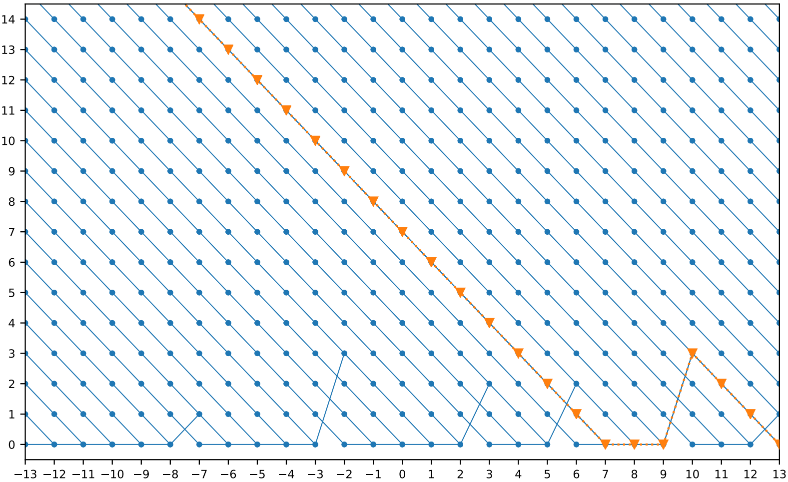
Doeblin Graphs

Unique bi-infinite path? No. Fall to 0 then Geom(1/2) jumps.



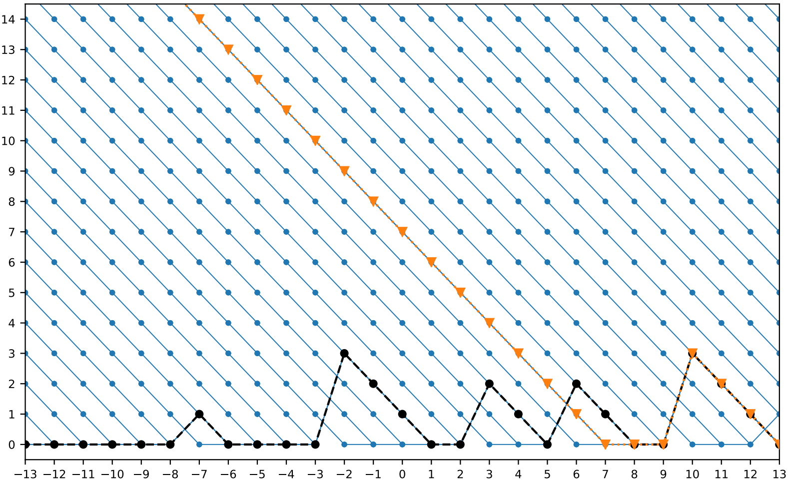
Doeblin Graphs

However, one bi-infinite path here is not like the others.



Doeblin Graphs

However, one bi-infinite path here is not like the others.



Bi-recurrence

Definition

A sequence $(x_t)_t$ in S is called **bi-recurrent** if $\{t : x_t = x\}$ is unbounded above and below for all $x \in S$. A random process $(X_t)_{t \in \mathbb{Z}}$ in S is called **bi-recurrent** if a.s. its trajectory is bi-recurrent.

Bi-recurrence

Definition

A sequence $(x_t)_t$ in S is called **bi-recurrent** if $\{t : x_t = x\}$ is unbounded above and below for all $x \in S$. A random process $(X_t)_{t\in\mathbb{Z}}$ in S is called **bi-recurrent** if a.s. its trajectory is bi-recurrent.

In the previous example, we saw the picture had a unique bi-recurrent path $(\beta_t)_t$.

Bi-recurrence

Existence of a unique bi-infinite path in a tree is a consequence of a familiar theorem of random networks, the cardinality classification theorem of vertex-shifts in unimodular networks.

Random Networks

Network $\Gamma \approx$ nonempty, locally finite, connected graph with vertices and edges marked.

Random Networks

Network $\Gamma \approx$ nonempty, locally finite, connected graph with vertices and edges marked.
Random network $[\boldsymbol{\Gamma}, \boldsymbol{o}] \approx$ random element in space of rooted networks up to isomorphism.

Random Networks

Network $\Gamma \approx$ nonempty, locally finite, connected graph with vertices and edges marked.
Random network $[\mathbf{\Gamma}, \boldsymbol{o}] \approx$ random element in space of rooted networks up to isomorphism.
Unimodular network $\approx$ random network with root picked "uniformly" (satisfies a mass transport principle).

Random Networks

Network $\Gamma \approx$ nonempty, locally finite, connected graph with vertices and edges marked.

Random network $[\boldsymbol{\Gamma}, \boldsymbol{o}] \approx$ random element in space of rooted networks up to isomorphism.

Unimodular network $\approx$ random network with root picked "uniformly" (satisfies a mass transport principle).

I.e. for all measurable $g \geq 0$

$$\mathbb{E} \sum_{v \in V(\boldsymbol{\Gamma})} g[\boldsymbol{\Gamma}, \boldsymbol{o}, v] = \mathbb{E} \sum_{v \in V(\boldsymbol{\Gamma})} g[\boldsymbol{\Gamma}, v, \boldsymbol{o}].$$

Random Networks

Network $\Gamma \approx$ nonempty, locally finite, connected graph with vertices and edges marked.
Random network $[\boldsymbol{\Gamma}, \boldsymbol{o}] \approx$ random element in space of rooted networks up to isomorphism.
Unimodular network $\approx$ random network with root picked "uniformly" (satisfies a mass transport principle).
I.e. for all measurable $g \geq 0$

$$\mathbb{E} \sum_{v \in V(\boldsymbol{\Gamma})} g[\boldsymbol{\Gamma}, \boldsymbol{o}, v] = \mathbb{E} \sum_{v \in V(\boldsymbol{\Gamma})} g[\boldsymbol{\Gamma}, v, \boldsymbol{o}].$$

Vertex-shift $\Phi \approx$ for each network Γ, Φ_Γ is a map from vertices to vertices. Must commute with isomorphisms. Not a random object, it is defined for all networks.

Doebin Graph as a Unimodular Network?

Finite case: OK, independent of the graph, choose root uniformly from states at time 0.
Unsatisfying and doesn't extend to infinite state space.

Doebin Graph as a Unimodular Network?

Finite case: OK, independent of the graph, choose root uniformly from states at time 0.
Unsatisfying and doesn't extend to infinite state space.

Some hurdles in infinite case:

Doebin Graph as a Unimodular Network?

Finite case: OK, independent of the graph, choose root uniformly from states at time 0.
Unsatisfying and doesn't extend to infinite state space.

Some hurdles in infinite case:

1. Local finiteness?

Doebin Graph as a Unimodular Network?

Finite case: OK, independent of the graph, choose root uniformly from states at time 0.
Unsatisfying and doesn't extend to infinite state space.

Some hurdles in infinite case:

1. Local finiteness? Can't fix.

Doebin Graph as a Unimodular Network?

Finite case: OK, independent of the graph, choose root uniformly from states at time 0.
Unsatisfying and doesn't extend to infinite state space.

Some hurdles in infinite case:

1. Local finiteness? Can't fix.
2. More than one bi-infinite path?

Doebin Graph as a Unimodular Network?

Finite case: OK, independent of the graph, choose root uniformly from states at time 0.
Unsatisfying and doesn't extend to infinite state space.

Some hurdles in infinite case:

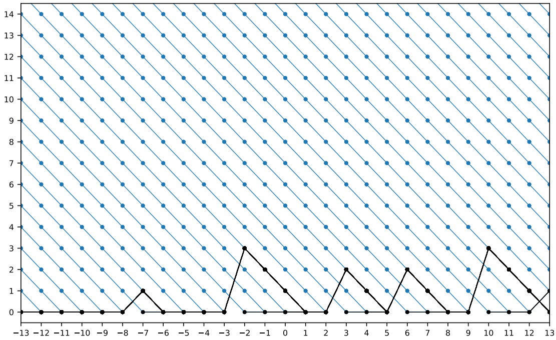
1. Local finiteness? Can't fix.
2. More than one bi-infinite path? Can't fix.

Bridge Graph

Fix state x^* (=0 in the picture).

Definition

The **bridge graph B for state** x^* is the subgraph induced by all paths started in state x^*.



Bridge Graph

Let $\mathbf{B}_t$ denote the vertical slice of $\mathbf{B}$ sitting at time t.
Let $M < \infty$ be the mean return time of a path started in state x^* to return to x^*.

Proposition

For all $t \in \mathbb{Z}$, $\mathbb{E}[\#\mathbf{B}_t] \leq M$. *In particular, the bridge graph is locally finite even if the Doeblin graph is not.*

Bridge Graph

Consider vertices (t, y) in bridge graph marked by (y, ξ_t).
Assume **B** is connected for the rest of the talk.

Theorem

Any random network with distribution

$$\mathbb{P}^{\square}(A) := \frac{1}{\mathbb{E}[\#\mathbf{B}_0]}\mathbb{E}\left[\sum_{(0,y)\in V(\mathbf{B})} 1\left\{[\mathbf{B},(0,y)]\in A\right\}\right]$$

is unimodular.

Note: $\mathbb{P}^{\square}$ is a size-biased version of original network.

Bridge Graph

A subset A of rooted networks is **root-invariant** if for all $[\Gamma, o] \in A$ one has $[\Gamma, v] \in A$ for all $v \in V(\Gamma)$.

Lemma

$\mathbb{P}^{\square}$ and $\mathbb{P}([\mathbf{B}, (0, x^*)] \in \cdot)$ have the same root-invariant sets of measure 0 or 1.

Consequences of Unimodularity

- "Follow the arrows" defines a vertex-shift, and applied to $\mathbf{B}$ this vertex-shift draws $\mathbf{B}$.
- Can now reap benefits of unimodularity of $\mathbb{P}^{\square}$.

Consequences of Unimodularity

- "Follow the arrows" defines a vertex-shift, and applied to **B** this vertex-shift draws **B**.
- Can now reap benefits of unimodularity of $\mathbb{P}^\square$.

Proposition

B *has an a.s. unique bi-infinite path* $(\beta_t)_t$. *This path is bi-recurrent. This is the only bi-recurrent path in all of the larger Doeblin graph. The path is shift-covariant, stationary, and for each* t, β_t *is measurable with respect to* $\sigma(\xi_s : s < t)$. *Moreover,* $(\beta_t)_t$ *is a Markov chain with transition matrix* P.

Consequences of Unimodularity

- "Follow the arrows" defines a vertex-shift, and applied to **B** this vertex-shift draws **B**.
- Can now reap benefits of unimodularity of $\mathbb{P}^{\square}$.

Proposition

B *has an a.s. unique bi-infinite path* $(\beta_t)_t$*. This path is bi-recurrent. This is the only bi-recurrent path in all of the larger Doeblin graph. The path is shift-covariant, stationary, and for each* t*,* β_t *is measurable with respect to* $\sigma(\xi_s : s < t)$*. Moreover,* $(\beta_t)_t$ *is a Markov chain with transition matrix* P*.*

Note that since $(\beta_t)_t$ is the only bi-recurrent path in the whole Doeblin graph, it does not depend on the state x^* used to generate **B**. Using another x^* would give a different bridge graph that has the same bi-recurrent path.

Consequences of Unimodularity

Recall that P is assumed irreducible, aperiodic, and positive recurrent.
The following is a consequence of the fact that any Markov chain can be embedded as a path in a Doeblin graph.

Theorem

Suppose $(X_t)_{t\in\mathbb{Z}}$ is a Markov chain with transition matrix P. Then $(X_t)_t$ is stationary if and only if it is bi-recurrent. Note that time index set is $\mathbb{Z}$, not $\mathbb{N}$.

Other bi-infinite paths

We saw that in some cases (finite state space) there appears to be a unique bi-infinite path, not just bi-recurrent path, but in other cases there were infinitely many bi-infinite paths.

Definition

Call a bi-infinite path that is not bi-recurrent **spurious**.

Other bi-infinite paths

We saw that in some cases (finite state space) there appears to be a unique bi-infinite path, not just bi-recurrent path, but in other cases there were infinitely many bi-infinite paths.

Definition

Call a bi-infinite path that is not bi-recurrent **spurious**.

Definition

Say $P^n \to \pi$ **uniformly** if $\sup_{x \in S} ||P^n(x, \cdot) - \pi||_{TV} \to 0$ as $n \to \infty$. (Some sources say P is **uniformly ergodic**.)

Other bi-infinite paths

Proposition

In fully IID case, if $P^n \to \pi$ uniformly, then the Doeblin graph contains no spurious bi-infinite paths.

Other bi-infinite paths

Proposition

In fully IID case, if $P^n \to \pi$ uniformly, then the Doeblin graph contains no spurious bi-infinite paths.

Proposition

Suppose $P^n \to \pi$ uniformly, then every Markov chain $(X_t)_{t\in\mathbb{Z}}$ with transition matrix P is stationary and bi-recurrent. The subtle assumption again is that the chain is indexed by $\mathbb{Z}$, not $\mathbb{N}$.

The previous result is a partial converse to the well known result that stationary sequence indexed on $\mathbb{N}$ can be extended to $\mathbb{Z}$.

Other bridge graphs

What about bridge graphs generated using a different x^*?.

Proposition

In the fully IID case, if S is infinite, and if the Doeblin graph is locally finite and contains no spurious bi-infinite paths, then the intersection of all bridge graphs using different values of x^* is exactly the bi-recurrent path.

On the other hand, if S is finite and has ≥ 2 states, then there is a.s. more.

Things left out

In the arxiv paper “Doeblin Trees”, you may also find results on:

- Local weak convergence to bridge graph by finite networks.
- Vertical slices of bridge graph are themselves a Markov chain with recurrence known to compute its transition matrix.
- Many mass transport relationships amongst objects discussed.
- Non Markov case and no need for connectedness (irreducible, aperiodic) assumption.

References

[1] Baccelli, F., Haji-Mirsadeghi, M.-O., and Murphy. J.: Doeblin Trees (2018). arXiv preprint.

[2] Propp, J. and Wilson D.: Exact sampling with coupled Markov chains and applications to statistical mechanics (1996).

[3] Stochastically recursive sequences and their generalizations (1992).

[4] Foss, S. and Tweedie R.: Perfect simulation and backwards coupling (1998).