The Instability of all Backoff Protocols

Leslie Ann Goldberg, University of Oxford

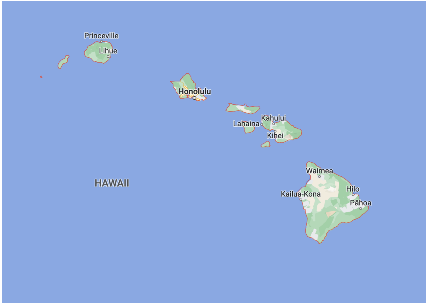
IFM 20th Anniversary, Sorbonne, Paris, March 2026

joint work with

John Lapinskas, Bristol

ArXiv 2602.21315

Warmup: ALOHA network, University of Hawaii, 1971



- radio transmitters on islands
- base station
- transmitters send messages
- If exactly 1 message sends at a time, base station sends acknowledgement
- Otherwise, sender waits a random time and tries again

Formal model: slotted ALOHA

- discrete time
- message length: 1 time step

- message: with probability p send, else wait
- n messages

probability of successful transmission: $f(p) := np\,(1-p)^{n-1}$

Formal model: slotted ALOHA

- discrete time
- message length: 1 time step

- message: with probability p send, else wait
- n messages

probability of successful transmission: $f(p) := np\,(1-p)^{n-1}$

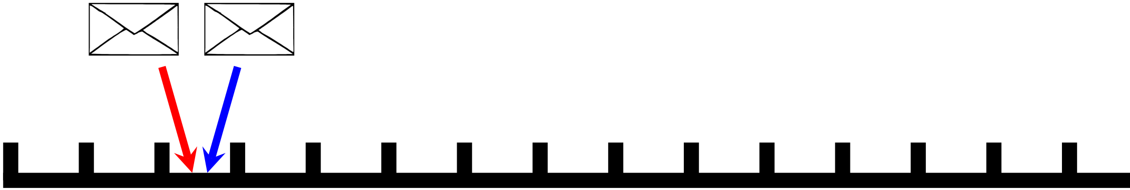
$f(p)$ maximised at $p = 1/n$

$\lim_{n\to\infty} f(1/n) = 1/e$.

The transmitters don't know n (and it may change over time)!

Motivation: coordinating requests to shared resource

- users send messages to request resource
- users can't communicate



- If a single message is sent in a time step, it succeeds (and leaves the system)
- If two (or more) messages send simultaneously, they collide (and need to be re-transmitted later)
- users arrive (and leave) over time — there is no n

Backoff Protocol

Randomised algorithm for determining how long to wait

Waiting time of message depends on how many times it has collided.

Send sequence $\overline{p} = (p_0, p_1, p_2, \ldots)$

If message collided k times, sends with probability p_k, silent with probability $1 - p_k$

Slotted ALOHA: $p_k = p$

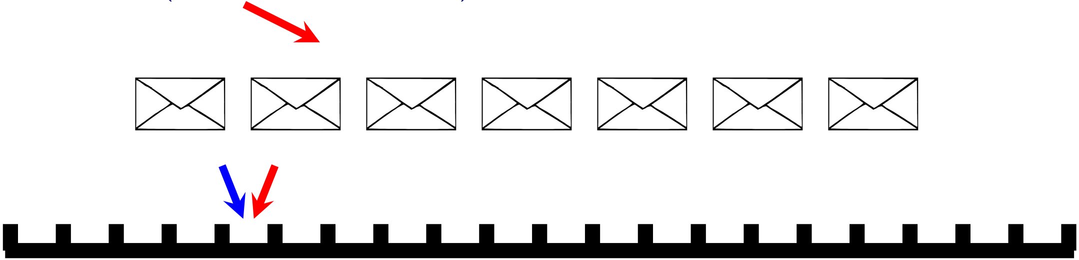
Binary exponential backoff: $p_k = 2^{-k}$ (basis of Ethernet, TCP/IP)

“backing off” to find the best value of p...

Studying Backoff Protocols

Frank Kelly, 1985

Poisson(λ) $\left(\text{prob of } k \text{ births is } \frac{\lambda^k}{e^\lambda k!}\right)$



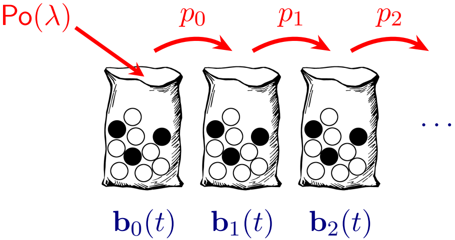
Is the resulting “backoff process” stable (positive recurrent)?

A backoff process is a Markov chain

send sequence $\overline{p} = (p_0, p_1, p_2, \ldots)$ birth rate $\lambda > 0$

State at time t: $\overline{\mathbf{b}}(t) = (\mathbf{b}_0(t), \mathbf{b}_1(t), \ldots)$

$\mathbf{b}_k(t)$: messages (balls) that have already collided k times in steps $1, \ldots, t$



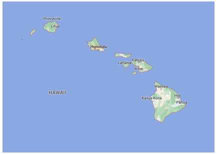
ball (message) escapes
if only one to send

Stability

- State of the system at time t: $\overline{\mathbf{b}}(t) = (\mathbf{b}_0(t), \mathbf{b}_1(t), \ldots)$
- Stable: expected time to return to empty state is finite
  "empty state is positive recurrent"
- Stable ⇒
  - converges to unique stationary distribution
    with finite expected number of messages waiting
- Not Stable ⇒
  - Messages build up!
  - Expected return time from any state x is infinite
  - As $t \to \infty$, probability of being in any given state at time t tends to 0.

obviously not stable for $\lambda > 1$. What about smaller λ?...

bad news for slotted ALOHA



send sequence
$\overline{p} = (p, p, \ldots)$
birth rate $\lambda > 0$

Kelly and MacPhee (1987)

characterised $\overline{p}$ such that, with probability 1, backoff only has finitely many successful sends.

polynomial backoff $p_k = k^{-c}$ also unstable in this very strong sense

Landmark result of Aldous (1987)

Theorem: Binary exponential backoff is unstable for any $\lambda > 0$

Why is it surprising?

- practical: Ethernet is built on binary exponential backoff!
- mathematical: His proof shows that binary exponential backoff doesn't back off enough!

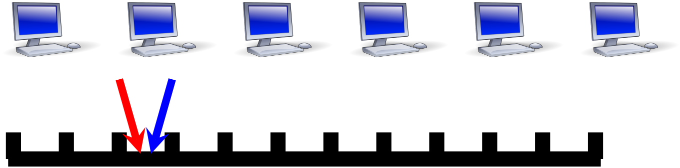
A brief digression into the practical...

Why is the Ethernet still useful (despite Aldous's result)?

- dropping messages that collided $\geq K$ times is stabilising
  (makes Markov chain positive recurrent).
- supplementing backoff with queues so that $\leq n$ messages can send at a time
  sometimes stabilising
  (but doesn't fit application — coordinating shared resource on cloud!).
- original Ethernet: $K = 16$, $n = 1024$.

queue-assisted model

n processors, each with a queue of messages



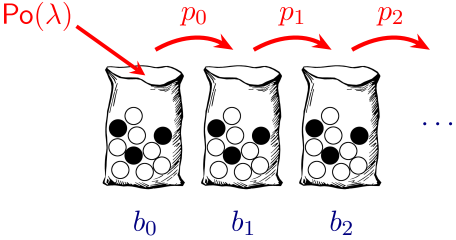
- Håstad, Leighton, Rogoff (1995): binary exponential backoff unstable for $\lambda > 1/2$ but polynomial backoff stable for any $\lambda < 1$
- Al-Ammal, G, MacKenzie (2001): binary exponential backoff stable $\lambda = O(n^{-0.75})$.

Other interesting models...

- Another algorithm-friendly model modification (that doesn't fit the application)
  - allow messages to continuously "listen" to channel (even when they don't send)
  - all messages at each step can distinguish between some or all of these outcomes: silence, a successful send, a collision
- Cool stuff:
  - Shah, Shin, Tetali (FOCS 2011)
  - Bender, Kopelowitz, Kuszmaul, Pettie (STOC 2020)
  - Chen, Jiang, Zheng (PODC 2021)
  - Bender, Fineman, Gilbert, Kuszmaul, Young (SICOMP 2025)
  - Xie, Zheng (SPAA 2025)
  - Cai, Chen, Du, Kopelowitz, Pettie, Plosk (STOC 2026)
  - ...

back to our model...

send sequence $\overline{p} = (p_0, p_1, p_2, \ldots)$ birth rate $\lambda > 0$



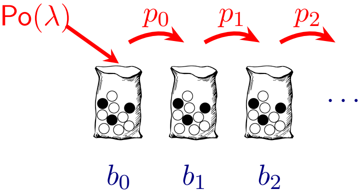
ball escapes
if only one to send

Bin b_k holds $b_k(t)$ balls at step t.

State $\overline{b}(t) = (b_0(t), b_1(t), \ldots)$

Aldous: Binary exponential backoff ($p_k = 2^{-k}$) unstable for any $\lambda > 0$

Aldous's analysis



- $\overline{b}(t) = (b_0(t), b_1(t), \ldots)$
- expected noise at t: $\mathcal{N}(t) = \sum_k p_k b_k(t-1)$
- $\mathcal{N}(t)$ grows without bound
- with positive prob, every t has $\mathcal{N}(t) = \Omega(\log t)$
- "summable failure probability" over t (union bound)
- Even though BEB backs off a lot, $\mathcal{N}(t)$ grows!

- conjectured the same for every $\overline{p}$
- believed argument could be modified to show instability of every backoff protocol for every positive λ

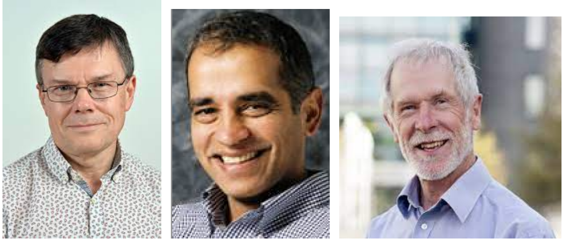
Aldous's Conjecture. *No backoff process is stable for any birth rate* $\lambda > 0$.

Aldous's Conjecture. *No backoff process is stable for any birth rate $\lambda > 0$.*

Our result: finally proved this.

Story more interesting than that, because the truth is kind of subtle...

- interleave slotted ALOHA and BEB to get $\overline{p}$ which goes completely silent infinitely often
- expected sends at time t — $\Omega(\log\log\log t)$ (conditioned on current state, goes to 0 infinitely often).



G, Jerrum, Kannan, Paterson (2004)

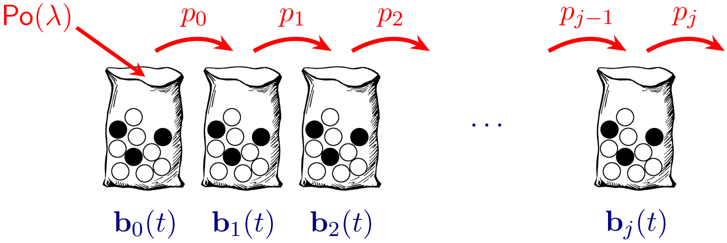
Theorem: *Every backoff protocol is unstable for any* $\lambda \geq 0.42$.

$1/e \sim 0.368$

Some progress since then...

useful tool: j-jammed process

State after step t: $\overline{\mathbf{b}}_{[j]}(t) = (\mathbf{b}_0(t), \mathbf{b}_1(t), \ldots, \mathbf{b}_j(t))$



no escapes from bins $0, \ldots, j-1$. All senders escape from bin j

Stationary distribution: $\mathrm{dist}(\bar{b}_{[j]}(t)) = \mathsf{Po}(\lambda W_0, \ldots, \lambda W_j)$ (Note notation!)
$W_k = 1/p_k$
Upper bound for backoff: bin k is "full" when $b_k(t) = \lambda W_k$.

Start from empty: $\mathrm{dist}(\bar{b}_{[j]}(t)) = \mathsf{Po}(f_1(t), \ldots, f_j(t))$.

$\forall k \in [j], t \geq 1, \quad f_k(t) = (1 - p_k) f_k(t-1) + p_{k-1} f_{k-1}(t-1)$.

SODA 2023 paper (with John Lapinskas)

- Aldous's proof relies on concentration for $b_j(t)$ as j gets large (each bin remains full)
- At time t, the expected noise comes from bins with indices between $\log(t)/10$ and $\log(t)$
- In BEB, $W_j = 2^j$.
- Concentration would fail for low-weight bins (small W_j) — these would become empty often
- In general, can't do the union bounds, can't escape the dependence !!
- Showed instability for all $\lambda > 0$ for "a bunch of" send sequences

Theorem: If $\overline{p} = (p_0, p_1, \ldots)$ is monotonically non-increasing then unstable.

Theorem: Let $m_{\overline{p}}(n)$ be the median of $p_0, \ldots, p_n$. If $m_{\overline{p}}(n) = o(1)$ then unstable.

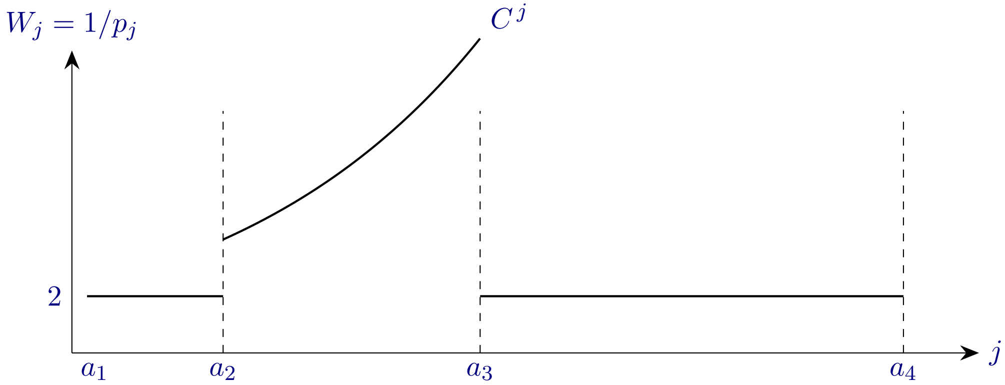
still needs expected noise to remain high whp

Something interesting - this doesn't always happen

- slotted ALOHA: send sequence $\overline{p} = (p, p, \ldots)$. Ever-growing expected noise
- exponential backoff: $p_k = C^{-k}$. Ever-growing expected noise.
- interleave them: silent infinitely often.

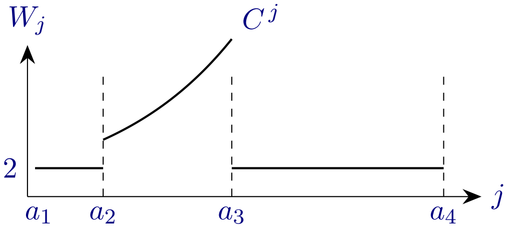
A backoff protocol that goes silent infinitely often ...

C large. $\overline{a} = (a_0, a_1, \ldots)$ fast-growing



$p_j = 1/2$ (slotted ALOHA) between a_k and a_{k+1} for k odd.
$p_j = C^{-j}$ (exponential backoff) between a_k and a_{k+1} for k even.

$p_j = 1/2$ between a_k and a_{k+1} for k odd. $p_j = C^{-j}$ between a_k and a_{k+1} for k even.



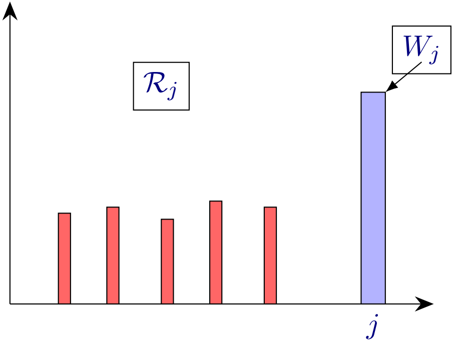
a_k for any sufficiently large even k is a “strongly exposed bin”

- likely behaviour - processs gets to a state where
    - all balls are in bin a_k
    - number of balls in bin a_k is $\ll C^{a_k}$
    - expected noise is near 0
- protocol alternates between “quiet periods” and “noisy periods” where expected noise grows rapidly and stays high for a very long time

Types of bins

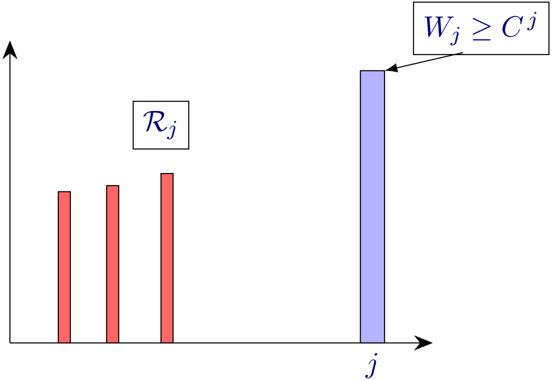
vocabulary: low weight (lw) $W_j < j^2$. high weight (hw) $W_j \geq j^2$.
very high weight (vhw) $W_j \geq j^{1000}$.

j is covered (all bins of BEB are covered)



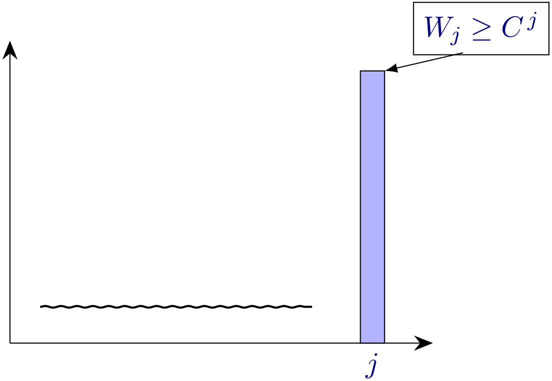
- $|\mathcal{R}_j| \geq \log(j)$
- bins in $\mathcal{R}_j$ are large enough that, once full, expect them to stay full for $\Omega(W_j)$ steps (enabling bin j to fill)
- technical def: Some faff.
  - Maybe $j-1$ bins of size 1 (if W_j isn't too big).
  - Maybe $\log(j)$ hw bins...

Types of bins: weakly exposed



- not covered
- $|\mathcal{R}_j| \geq \log\log(j)$
- bins in $\mathcal{R}_j$ are vhw
- bins in $[j-1]$ may go mostly empty while j is filling (to point where balls likely escape before reaching j) but vhw bins in $\mathcal{R}_j$ make enough noise for these to fill back up
- noise from $\mathcal{R}_j$ ensures new balls arrive faster than existing balls escape

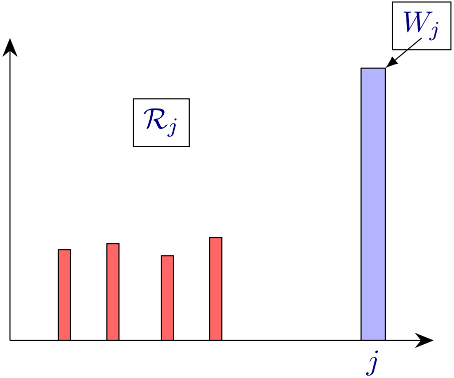
Types of bins: strongly exposed



- Not covered
- only $O(\log\log j)$ vhw bins in $[j-1]$
- bins in $[j-1]$ may go completely empty while j is filling

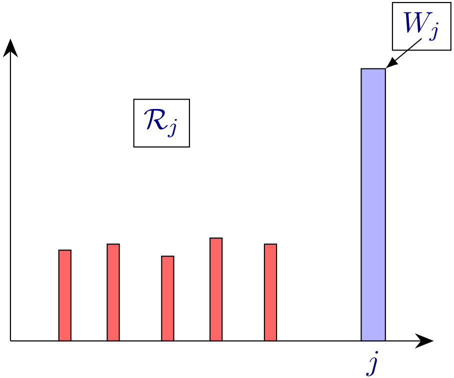
Two different proofs, depending on whether there are infinitely-many SE bins

Only finitely-many strongly exposed



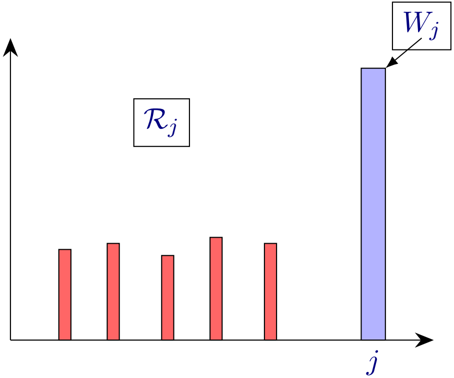
- all SE in $[j_0]$
- τ_0, depending on $\overline{p}$, j_0
- possible, but unlikely, event $\mathcal{E}_{\text{init}}$
- Conditioned on $\mathcal{E}_{\text{init}}$,
  $\text{dist}(\bar{b}_{[j_0]}) \gtrsim \mathsf{Po}(\varepsilon\lambda W_1, \ldots, \varepsilon\lambda W_{j_0})$ (notation!)
- Condition on $\mathcal{E}_{\text{init}}$ for rest of proof
- Consider bins j in order, from $j = j_0 + 1$
- Bin j is covered or weakly exposed, so it has some $\mathcal{R}_j$
- Bin j fills in time $W_j \cdot \text{poly}(j)$ (except with failure probability that can be summed over j).

Filling bin j when j is covered



- like Aldous's proof
- $|\mathcal{R}_j| \geq \log(j)$
- bins in $\mathcal{R}_j$ are large enough that, once full, we expect them to stay full for $\Omega(W_j)$ steps
- escape probability at t $\lesssim \exp(-\mathcal{N}(t))$
- newborn only sends j times on its way to bin j so unlikely to escape
- bin j fills with summable fp

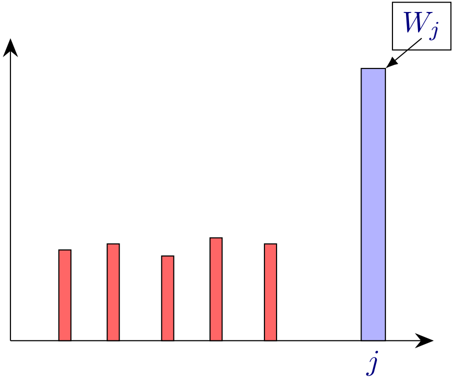
Filling bin j when j is weakly exposed



- Start with good situation:
  $\mathrm{dist}(\bar{b}_{[j-1]}(t)) \gtrsim \mathsf{Po}(\varepsilon\lambda W_1, \ldots, \varepsilon\lambda W_{j-1})$
- when good situation fails
  (it is going to fail!)
  balls become likely to escape before reaching j
  vhw bins in $\mathcal{R}_j$ maintain
  $\log\log(j)$ noise for $\geq \mathrm{poly}(j)$ time
- noise enables it to re-establish original good situation (ensuring more balls arrive than escape)

Big problem (for proof): How to fall back to lower noise level without losing Poisson domination

Filling weakly exposed bin j - Key Insight



- Start with $\mathrm{dist}(\bar{b}_{[j-1]}(t)) \gtrsim \mathsf{Po}(\varepsilon\lambda W_1, \ldots, \varepsilon\lambda W_{j-1})$
- What could cause this to fail before bin j fills (before domination extends to j)?
- "Send problem": many balls with unusual send patterns, leading to (for example) long stretch of empty low-weight bins
- "Escape problem": more balls escaped than expected

• events influencing escape problem are heavily dependent, but unlikely enough to union bound away

• send problems fairly likely, but the events they depend on (individual ball trajectories) truly independent

The coupling

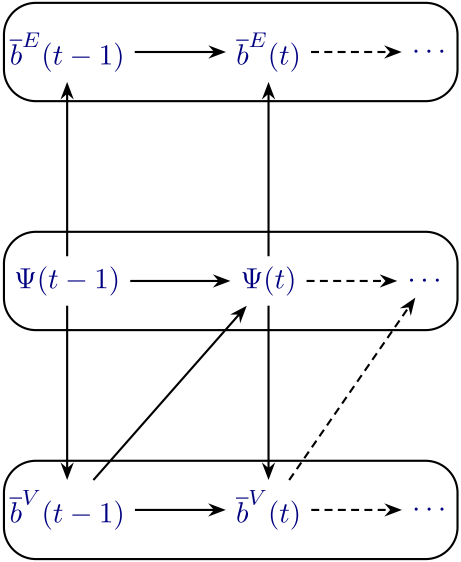
Couple the backoff process X with

- volume process V – tracks send patterns
- escape process E – tracks escapes
- tightly-controlled limited interface to make E sufficiently independent

Coupling Invariant: $\mathbf{b}_i^V(t) \setminus \mathbf{b}_i^E(t) \subseteq \mathbf{b}_i^X(t)$

- Proof property: With positive probability, $\forall\, t \, \exists\, i$ s.t. $b_i^V(t) \setminus b_i^E(t)$ non-empty
- while j is filling, V and E act as j-jammed processes,
  but E has no births –
  has (independent) Bernoulli arrivals in each bin at end of each step,
  rates depending on interface need to be large enough for coupling invariant and small enough for proof property

The basic idea: high-level state (hls) algorithm



$$\begin{tikzcd}
\bar{b}^E(t-1) \arrow[r] & \bar{b}^E(t) \arrow[r, dashed] & \cdots \\
\Psi(t-1) \arrow[r] \arrow[u] \arrow[d] & \Psi(t) \arrow[r, dashed] \arrow[u] \arrow[d] & \cdots \\
\bar{b}^V(t-1) \arrow[r] \arrow[ur] & \bar{b}^V(t) \arrow[r, dashed] \arrow[ur, dashed] & \cdots
\end{tikzcd}$$

- “high-level state” $\Psi(t)$
- high-level states evolve with V
- $\Psi(t)$ controls arrival rates in E at time t.
- Evolution of E does not depend on V, except through high-level states.

Future work

- Is every backoff protocol transient for every positive λ?

  Would have to improve proof for infinitely-many SE bins, and upper bound on bin weights in finite proof.

- Is every acknowledgement-based protocol unstable for every positive λ?

  Asked by Kelly (1985). MacPhee conjectured no! Kelly: stability impossible for $\lambda > 0.567$. Improved by [GJKP] to $\lambda > 0.531$.

- What about meta-stability?