Lazy Ring + Directed Shortcut: Co-located Fig. 1 and Sensitivity Analysis

Abstract

This report is written for a broad audience and focuses on a single co-located figure (now in stacked panels) and the associated sensitivity analyses. The default example uses $N = 100$, $K \in \{2, 4\}$, $q = 2/3$, $\beta = 0.01$, $\rho = 1.0$, and the lazy_selfloop mode. Figure 1 stacks two panels (top $K = 2$, bottom $K = 4$); each panel co-locates the first-passage-time distribution $f(t)$ with three time windows (peak1/valley/peak2) and the four trajectory classes on the same axis. We summarize sensitivity to detection thresholds, window placement/width, and Monte Carlo (MC) uncertainty.

Quick Symbols

- N: number of nodes on the ring; $K = 2/4$ means steps to $\pm r$ with $r = \lfloor K/2 \rfloor$, plus lazy waiting and a directed shortcut.
- jump-over: the trajectory passes the target but does not absorb; counted independently from shortcut usage.
- Four classes: `C0J0`, `C1pJ0`, `C0J1p`, `C1pJ1p`, meaning no shortcut/no jump-over, shortcut/no jump-over, no shortcut/jump-over, shortcut/jump-over.
- $C1p/J1p$ means "at least once shortcut/jump-over," $C0/J0$ means "never."

Parameters and Window Definition

Default parameters: $N = 100$, $K \in \{2, 4\}$, $q = 2/3$, $\beta = 0.01$, $\rho = 1.0$, lazy_selfloop. Here q is the lazy waiting probability, β controls shortcut strength, ρ is the absorption rate, and t is the discrete time step.

Window width follows the same fixed rule as the existing workflow: given two early peaks t_1, t_2 and the valley t_v,

$$\Delta = \max\{1, \lfloor \delta_{\text{frac}}(t_2 - t_1) \rfloor\}, \quad \delta_{\text{frac}} = 0.05,$$

windows are defined as $[t_1 - \Delta, t_1 + \Delta]$, $[t_v - \Delta, t_v + \Delta]$, $[t_2 - \Delta, t_2 + \Delta]$. The centers (t_1, t_v, t_2) are detected from exact $f(t)$ via a peak/valley rule, so they are not random; implementation details are recorded in the analysis code. We detect (t_1, t_v, t_2) separately for each K from its exact $f(t)$, and compute Δ by the same fixed rule. Each panel uses its own windows so the bars align with that panel's peak/valley structure.

The window choice is justified both qualitatively and quantitatively: qualitatively, a bimodal distribution corresponds to fast and slow channels; peak1/peak2 probe the two channels and the valley captures their separation, so $P(\text{class} \mid T \in \text{window})$ is most interpretable there. Quantitatively, the centers come from peak/valley detection, the width scales with peak separation, and the window-shift/width sensitivity shows only 10^{-3}-level changes for $\Delta = \pm 1, 2, 5$, so conclusions do not depend on precise boundaries. Table 1 lists window centers, ranges, widths, and sample counts.

Table 1: Window parameters and sample counts ($\beta = 0.01, N = 100$; centers from $K = 4$ peak/valley detection).

window	t_{center}	t_L	t_R	Δ	width	$f_{K=2}(t_c)$	$f_{K=4}(t_c)$	$n_{K=2}$	$n_{K=4}$
peak1	32	9	55	23	46	1.22e-04	1.10e-04	2337	924
valley	84	61	107	23	46	1.26e-04	9.52e-05	1981	940
peak2	495	472	518	23	46	1.62e-04	6.23e-04	5796	5741

Reproducibility (Shortest Commands)

Run the following in this report directory:

```
make fig2
make sensitivity
make pdf
```

Outputs: Figure 1 in `figures/`, sensitivity results in `outputs/sensitivity/`, LaTeX build files in `build/`.

Data Provenance

Figure/Table	Script	Inputs	Output/Command
Fig. 1 (stacked panels)	`code/plot_fig2_overlap_binbars.py`	`data/fig2_bins_bars_beta0.01.json`, `data/ft_beta0.01.csv`	`figures/fig2_overlap_binbars_beta0.01_x1350.pdf`; `make fig2`
Threshold sensitivity	`code/sensitivity_thresholds.py`	`data/ft_beta0.01.csv`	`outputs/sensitivity/threshold_sweep_heatmap.pdf`
Window shift/width	`code/sensitivity_bin_intervals.py`	`data/fig2_bins_bars_beta0.01.json` + `cond_by_t.csv`	`outputs/sensitivity/bin_shift_plot.pdf`
MC uncertainty	`code/sensitivity_mc_uncertainty.py`	`data/fig2_bins_bars_beta0.01.json` + `cond_by_t.csv`	`outputs/sensitivity/mc_uncertainty_plot.pdf`

1 Co-located Curves and Stacked Window Bars

Figure 1 stacks two panels: the top shows $K = 2$ and the bottom shows $K = 4$. Each panel co-locates the first-passage-time distribution $f(t)$ and the windowed class proportions on one axis. The $K = 2$ curve is dark blue/black and the $K = 4$ curve is red, making the peak/valley contrast explicit. The horizontal axis is discrete time t (from `data/ft_beta0.01.csv`; here $t = 1 \dots 1350$ for readability). Three stacked bars correspond to peak1/valley/peak2 windows, which are determined from that panel's (t_1, t_v, t_2) and the same fixed Δ rule. Each stacked bar spans $[t_L, t_R]$ and encodes $P(\text{class} \mid T \in \text{window})$ for the four trajectory classes within the panel's K. If a class proportion is zero, its color does not appear. If time-by-time class probabilities are missing, the plot falls back to window-level aggregates and records this in the JSON meta.

What the windowed bars say about the bimodality mechanism

Phenomenon. The two panels of Figure 1 show that $f(t)$ has an early hump near t_1, a valley near t_v, and a late hump near t_2, with $t_2 \gg t_1$ (here $t_2/t_1 \approx 21$ for $K = 2$ and ≈ 15 for $K = 4$). Bimodality is not a numerical artefact: the windowed class composition above each bar tells us *which kind of trajectory contributes to each peak.*

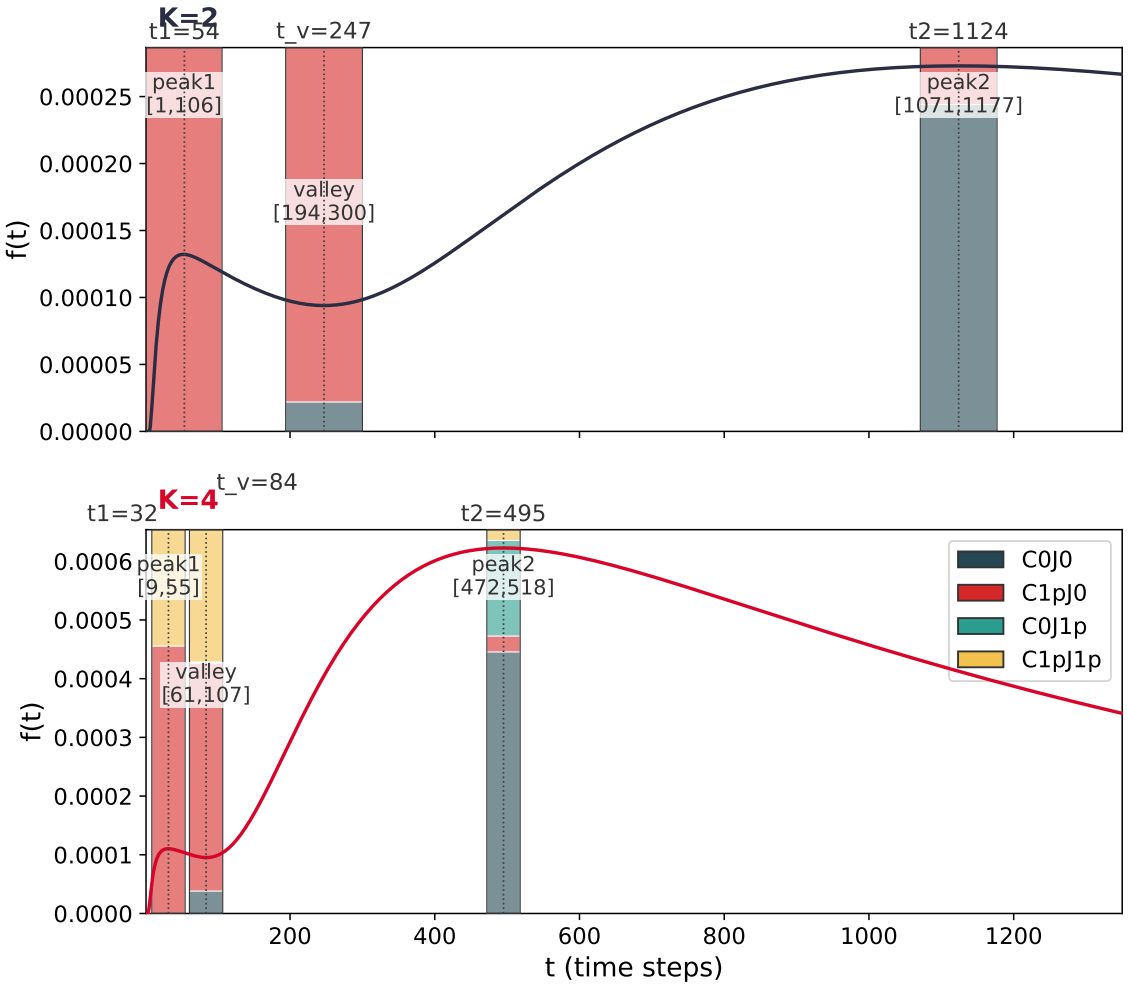


Figure 1: Two-panel view of co-located $f(t)$ and windowed class proportions. Top: $K = 2$ (dark blue/black). Bottom: $K = 4$ (red). t is discrete time (x-axis shown as $t = 1 \ldots 1350$ for readability); the labels show each panel's t_1, t_v, t_2 and window ranges. Colors inside each bar represent $P(\text{class} \mid T \in \text{window})$ for that K.

Evidence. Reading the bars from peak1 to peak2 in either panel, the class composition flips. In the $K = 2$ panel, peak1 is essentially a pure `C1pJ0` population (walker used the shortcut at least once, no jump-over); peak2 is essentially a pure `C0J0` population (walker never used the shortcut and never jumped over the target). The valley window sits in the transition between the two compositions. The $K = 4$ panel shows the same handover but with a second axis: the peak1 window picks up a sizeable `C1pJ1p` (shortcut + jump-over) share, and the peak2 window picks up a sizeable `C0J1p` (no shortcut + jump-over) share. The MC uncertainty bars in Figure 4 confirm that these compositional differences are well outside the 95% CIs, so the handover is statistical fact rather than sampling fluctuation.

Mechanism. The early peak is a *fast channel*: walkers that encounter the directed shortcut soon after starting take a near-direct path to the target and absorb at small t. The late peak is a *slow channel*: walkers that never trigger the shortcut have to traverse the ring the long way around, and their first-passage time is governed by the diffusion timescale on the unmodified ring. The valley is the time interval where neither channel deposits much probability mass, because the fast channel has already finished and the slow channel has not yet begun to arrive. For $K = 4$, the wider step $r = \lfloor K/2 \rfloor = 2$ adds a second separation axis: a walker can step *over* the target without absorbing (jump-over), which delays absorption and inflates the slow channel further. This explains why $K = 4$ is more reliably bimodal than $K = 2$: it has two independent timescale-separating mechanisms (shortcut and jump-over), whereas $K = 2$ has only one.

Conclusion. The bimodality of $f(t)$ in the lazy ring + shortcut model is a superposition of two trajectory populations with different first-passage timescales — it is structural, not stochastic. The peak1/peak2 windows are not just "where the curve is high"; they are *the parameter regions where each population dominates*, and the windowed class proportions are the direct evidence for that interpretation.

2 Sensitivity Analyses

2.1 Threshold Sensitivity

We scan $h_{\min} \in \{10^{-8}, 10^{-7}, 10^{-6}\}$, second_frac$\in \{0.005, 0.01, 0.02\}$, $t_2/t_1 \in \{8, 10, 12\}$, and hv_over_max$\in \{0.10, 0.20, 0.30\}$. Figure 2 summarizes the pass rates by two heatmaps: left for hv_over_max vs second_frac (averaged over $h_{\min}, t_2/t_1$), right for t_2/t_1 vs $h_{\min}$ (averaged over hv_over_max and second_frac). Aggregating $N = 100$ and $\beta \in [0.001, 0.03]$ (five values), $K = 2$ passes about 0.33 and $K = 4$ about 0.60; hv_over_max has the strongest effect ($K = 2$: 0.20/0.40/0.40; $K = 4$: 0.40/0.60/0.80), while $h_{\min}$, second_frac, and t_2/t_1 are weaker. Thus the "$K = 4$ is more likely bimodal than $K = 2$" conclusion is robust, with highest sensitivity to hv_over_max.

2.2 Window Position and Width

For each window $[t_L, t_R]$, we test shifts and width changes with $\Delta \in \{-5, -2, -1, 0, 1, 2, 5\}$ time steps. Shift uses $[t_L + \Delta, t_R + \Delta]$ and width uses $[t_L - \Delta, t_R + \Delta]$. Figure 3 shows both modes. If per-time class probabilities are missing, we impute from window-level baselines. When class probabilities are approximately monotone in t near a window, Δprop can look like triangular or piecewise-linear shapes; this is an expected effect of window sliding. Results: $K = 2$ has max change $\max|\Delta p| \le 4.49 \times 10^{-3}$ (shift) and $\le 4.26 \times 10^{-3}$ (width); $K = 4$ has $\le 5.78 \times 10^{-3}$ (shift) and $\le 5.13 \times 10^{-3}$ (width). Changes remain below 1%, and the main composition conclusions stay the same.

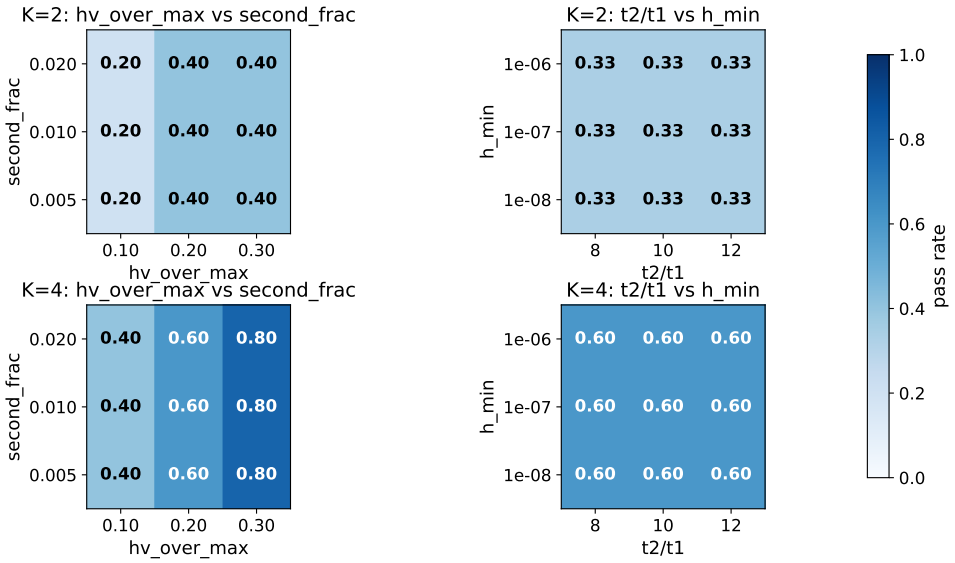


Figure 2: Threshold sensitivity heatmaps (pass rate).

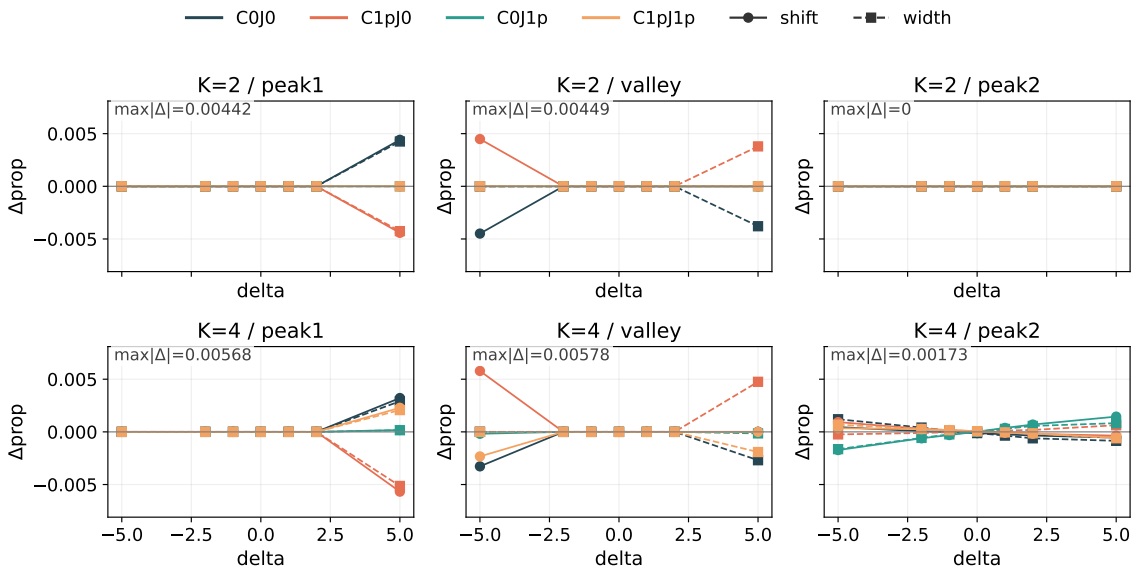


Figure 3: Window position and width sensitivity (shift/width).

2.3 MC Uncertainty

We report 95% confidence intervals using a normal approximation based on window sample counts. If cond_by_t does not cover a window, we use window-level counts (summary/JSON) as a fallback. Figure 4 shows that $K = 2$ has mean CI width ≈ 0.011 (max ≈ 0.036), and $K = 4$ has mean CI width ≈ 0.029 (max ≈ 0.063). These intervals are still smaller than class differences and do not change the main interpretation.

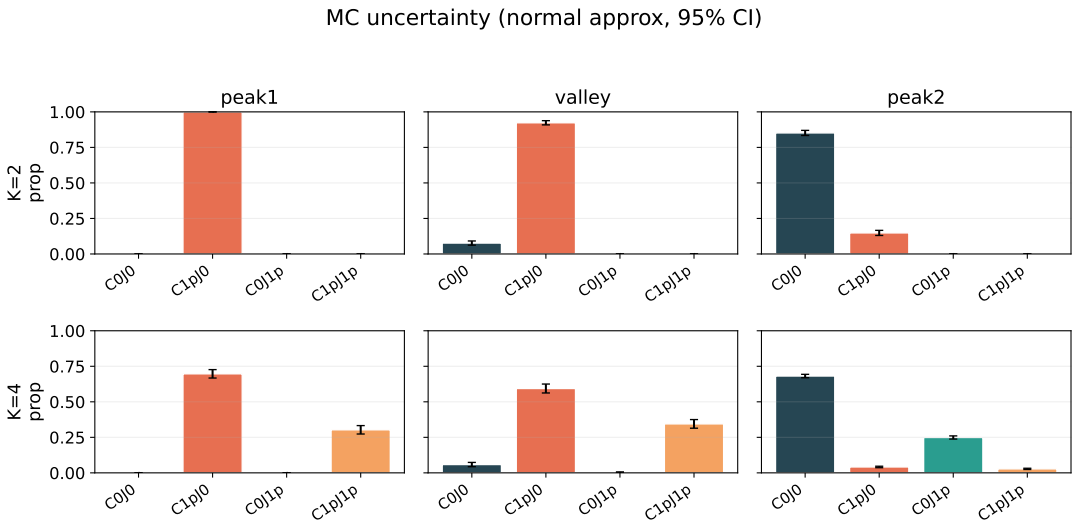


Figure 4: MC uncertainty (error bars/CI).

Conclusion

The bimodality of $f(t)$ in the lazy ring + directed shortcut model is the signature of two distinct trajectory populations: a *fast channel* (`C1p`$\cdot$) of walkers that take the shortcut, contributing to peak1 near t_1, and a *slow channel* (`C0`$\cdot$) of walkers that traverse the ring the long way, contributing to peak2 near t_2. The valley between the two peaks is the temporal gap where neither channel contributes substantial mass. Figure 1 makes this mechanism directly visible: the peak1/peak2 bars are dominated by `C1p`/`C0` respectively in both $K = 2$ and $K = 4$, and the $K = 4$ panel reveals a second separation axis (`J1p` jump-over) which explains its stronger bimodality.

This interpretation is robust along three axes: (i) threshold sensitivity keeps the "$K = 4$ more bimodal than $K = 2$" verdict over a wide grid of detection parameters, with only `hv_over_max` producing non-negligible variation; (ii) window placement and width perturbations ($\Delta = \pm 1, 2, 5$) change the windowed class proportions by less than 1%, so the channel-handover narrative does not depend on the precise window edges; (iii) MC confidence intervals are smaller than the between-class differences in every window, so the dominant class in each peak is statistically well-separated from the others.

For model derivations, the full β sweep, and broader K, N, q ranges, see `reports/lazy_jump_ext/`. The companion report `ring_lazy_flux/` gives an alternative diagnostic via the shortcut flux, and `ring_valley/` treats the valley structure without the shortcut.