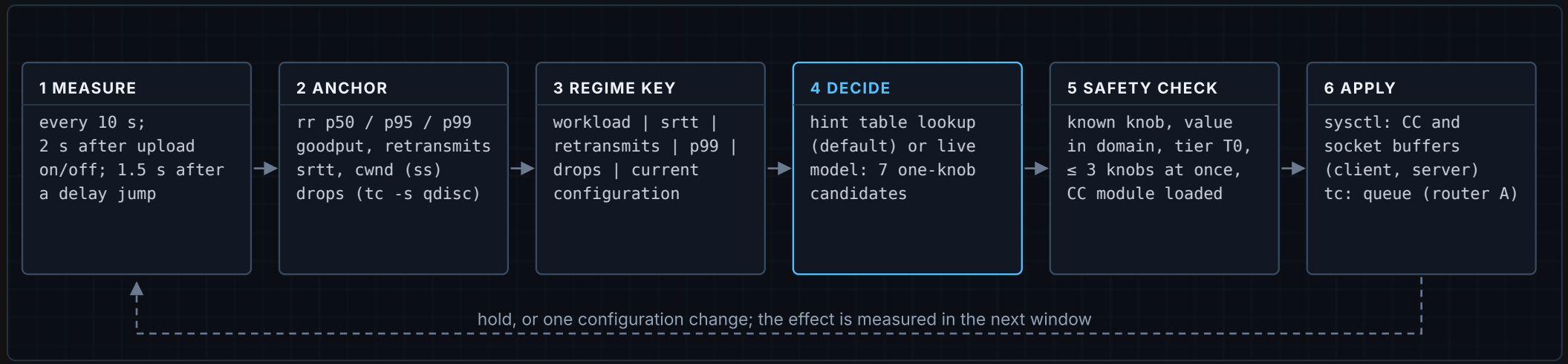
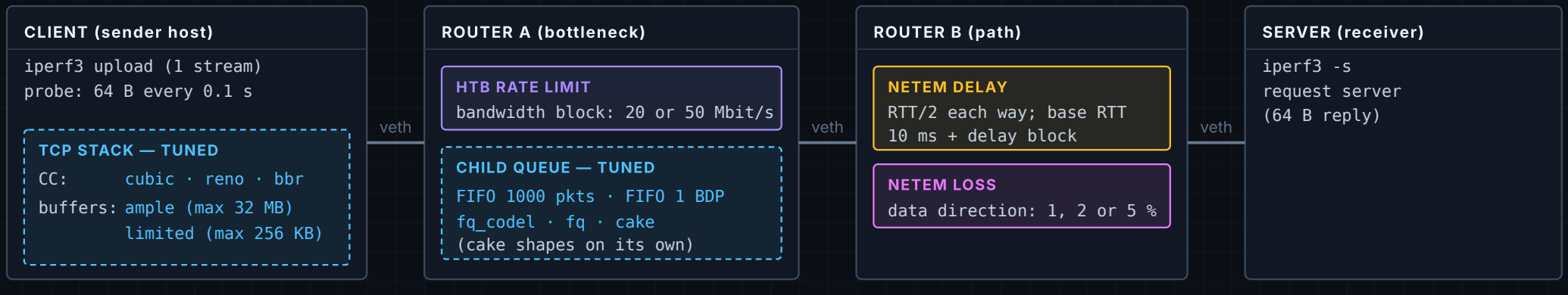
NetOracle: how it works

Companion to "Toward a World Model for the Linux Kernel Network Stack" (Luyao Bai and Xiaoguang Wang, HotNets '26). The control loop, where each knob lives in the Linux kernel, the model, the compiled hint table, and the configuration space.

Control loop



Where each knob lives in the kernel



CC	`$ sysctl -w net.ipv4.tcp_congestion_control=reno`	client; applies to new connections, so the upload reconnects
buffers	`$ sysctl -w net.ipv4.tcp_wmem='4096 65536 262144'`	client and server, with tcp_rmem; limited preset shown
queue	`$ tc qdisc add dev <A→B> parent 1:1 handle 10: fq_codel`	router A egress, under the HTB class
scenario	`$ tc qdisc replace dev <B→server> root netem delay 30ms loss 2%`	router B; set by the blocks, never by a method

The model

INPUT (prompt, abridged)

```
Testbed: namespaces dumbbell; routerA: HTB
rate limit 20Mbit + qdisc; routerB: netem.
A measured run on this testbed:
workload=mixed cc=cubic qdisc=pfifo_deep
rtt=10ms loss=0.0% bw=20Mbit buf=ample
-> rr_p99_ms=..., goodput_mbps=..., srtt...
Now exactly one knob changes: qdisc goes
from pfifo_deep to fq_codel. Predict rr_p99_ms.
```

→

MODEL

```
Qwen3.5-0.8B
+ LoRA r 32, alpha 64
12.8 M trainable params
726 measured cells,
~3,800 training examples
```

→

OUTPUT (strict JSON)

```
{"direction": "decrease",
"ratio": <new / old>}
within ±10 % counts as "same"
```

LIVE GATE (--netoracle model)

for each of the 7 one-knob candidates: 5 samples at temperature 0.8 → change only if ≥ 4 agree on a gain ≥ 15 %; the largest gain wins, otherwise hold

From model to hint table

REGIME GROUPS
measured cells grouped by regime key (observables only)

→

MEMBERS
up to 4 cells per group, spread over the group's p99 or goodput range

→

EFFECT RATIOS
per one-knob neighbor: measured if that cell was run, else model: 15 samples, ≥ 12 agree

→

NO-REGRESSION
drop it if any member loses > 10 %; keep it if the geometric-mean gain is ≥ 15 %

→

TABLE ENTRY
best surviving change, or hold; every number keeps its source

table_v4	300 entries: 120 prescribe a change, 180 hold. Estimates behind them: 1,067 measured, 2,650 from the model.
lookup	exact key, else the nearest key with the same workload, configuration and drop bucket (bucket distance).
key	mixed | s0 <30 ms · s1 <80 · s2 ≥80 | r0 none · r1 ≤20 · r2 >20 | p0 <30 ms · p1 <100 · p2 ≥100 | d0 · d1 | cubic/pfifo_deep/ample

Configuration space and knob tiers



T0 tuned here	congestion control, queue, socket buffers
T1 opt-in	initcwnd, rto_min, ECN, pacing, AQM parameters (not used in this demo)
T2 host-wide	NIC coalescing, global buffer limits, txqueuelen (never allowed here)
EX correctness	SACK, timestamps, window scaling (never actuated)

What differs between the methods

Method	Step 4 (decide)	Uses the anchor?	Typical changes per run
Kernel default	never changes	no	0
Bayesian optimization	Gaussian process + expected improvement over the 30 configurations; tries a new configuration at each decision until its budget of 8 is spent, then returns to the best one seen	only the metric it is optimizing	7–9
NetOracle	hint-table lookup by regime key (default), or the live model gate; changes only when it predicts a clear gain	yes, all signals	0–1

All three methods share the same testbed, measurements, schedule and safety check; only the decision step differs. Numbers on this page come from the code and the released model card.