# FTZ: A State-Inferring Fuzzer for the TCP/IP Stack of Zephyr

## Master's Thesis

Valentin Huber, 28.02.2025

# Background

Real-Time Operating System

Open Source

# Zephyr

Networking is Central

Widely Used

Repeated Execution of a Target
with Random(-ish) Inputs

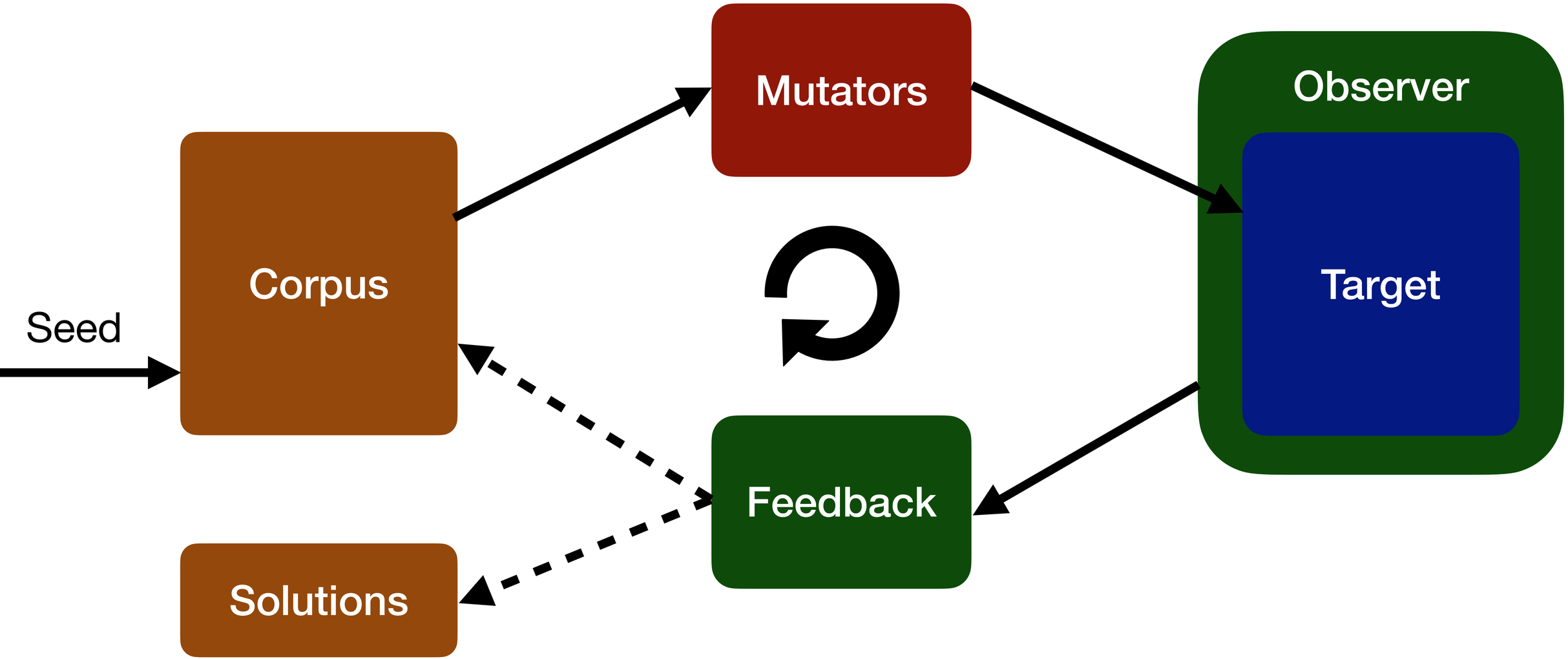
Significant Developments

# Fuzzing

Widely Used and Proven Effective

Looking for Illegal Program States

# Mutational Fuzzing



# Fuzzing OS Network Stacks Is Hard

# Fuzzing OS Network Stacks Is Hard

- Deep Integration with Operating System

# Fuzzing OS Network Stacks Is Hard

- Deep Integration with Operating System
- Network Packets Are Highly Structured

| Layer | Ethernet | | | | IPv4 | | | | | | | | | | | TCP | | | | | | | | | Payload | Ethernet |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Field** | Destination MAC | Source MAC | 802.1Q VLAN Tag | EtherType | Version | Internet Header Length | DSCP & ECN | Total Length | Identification | Flags & Fragment Offset | Time To Live | Protocol | Header Checksum | Source IP | Destination IP | Options | Source Port | Destination Port | Sequence Number | Acknowledgment Number | Data Offset & Reserved | Flags | Window | Checksum | Urgent Pointer | Options | Payload | Frame Check Sequence |
| **Size** | 6 | 6 | 4 | 2 | 1 | 1 | 1 | 2 | 2 | 2 | 1 | 1 | 2 | 4 | 4 | var | 2 | 2 | 4 | 4 | 1 | 1 | 2 | 2 | 2 | var | var | 4 |

# Fuzzing OS Network Stacks Is Hard

- Deep Integration with Operating System
- Network Packets Are Highly Structured
- TCP Stacks Have Internal State

Contributions: 27 PRs, 10,000 LoC

Fuzzing Library in Rust

# LibAFL

Advanced Implementations

Common Structures

Incompatible Improvements

# Implementation

# Implementation

# Implementation

- native_sim

# Implementation

- native_sim
- SanitizerCoverage

# Implementation

- native_sim
- SanitizerCoverage
- Custom Ethernet Driver

# Implementation

- native_sim
- SanitizerCoverage
- Custom Ethernet Driver
- Input Modeling and Mutation

# Input Modeling and Mutation

- Message Modeling

| Layer | Ethernet | | | | IPv4 | | | | | | | | | | | TCP | | | | | | | | | Payload | Ethernet |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Field** | Destination MAC | Source MAC | 802.1Q VLAN Tag | EtherType | Version | Internet Header Length | DSCP & ECN | Total Length | Identification | Flags & Fragment Offset | Time To Live | Protocol | Header Checksum | Source IP | Destination IP | Options | Source Port | Destination Port | Sequence Number | Acknowledgment Number | Data Offset & Reserved | Flags | Window | Checksum | Urgent Pointer | Options | Payload | Frame Check Sequence |
| **Size** | 6 | 6 | 4 | 2 | 1 | 1 | 1 | 2 | 2 | 2 | 1 | 1 | 2 | 4 | 4 | var | 2 | 2 | 4 | 4 | 1 | 1 | 2 | 2 | 2 | var | var | 4 |

# Input Modeling and Mutation

- Message Modeling
- Trace Modeling

# Input Modeling and Mutation

- Message Modeling
- Trace Modeling
- Appending Mutators

# Implementation

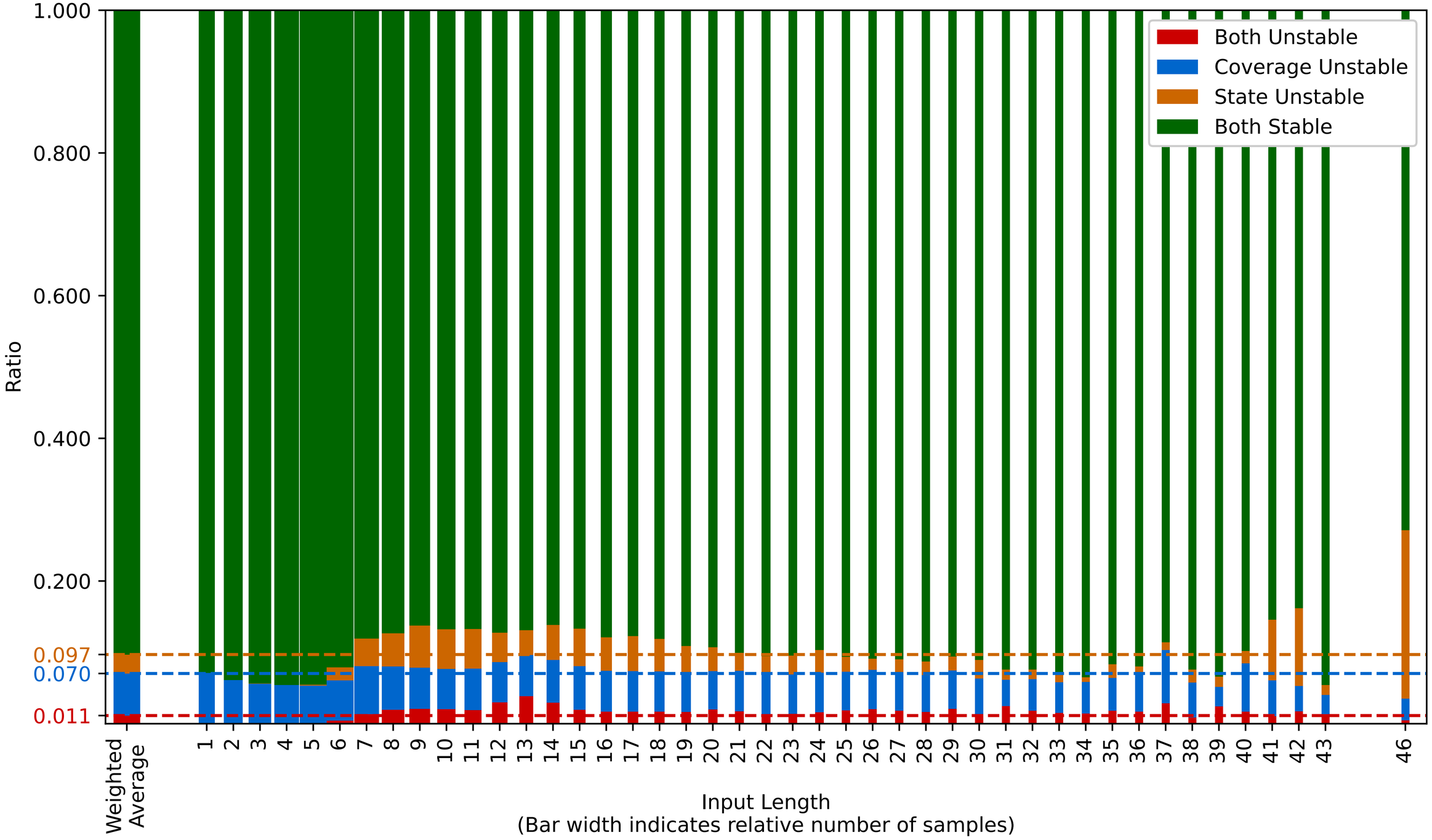
- native_sim
- SanitizerCoverage
- Custom Ethernet Driver
- Input Modeling and Mutation

# Implementation

- native_sim
- SanitizerCoverage
- Custom Ethernet Driver
- Input Modeling and Mutation
- State Inference

# Evaluation

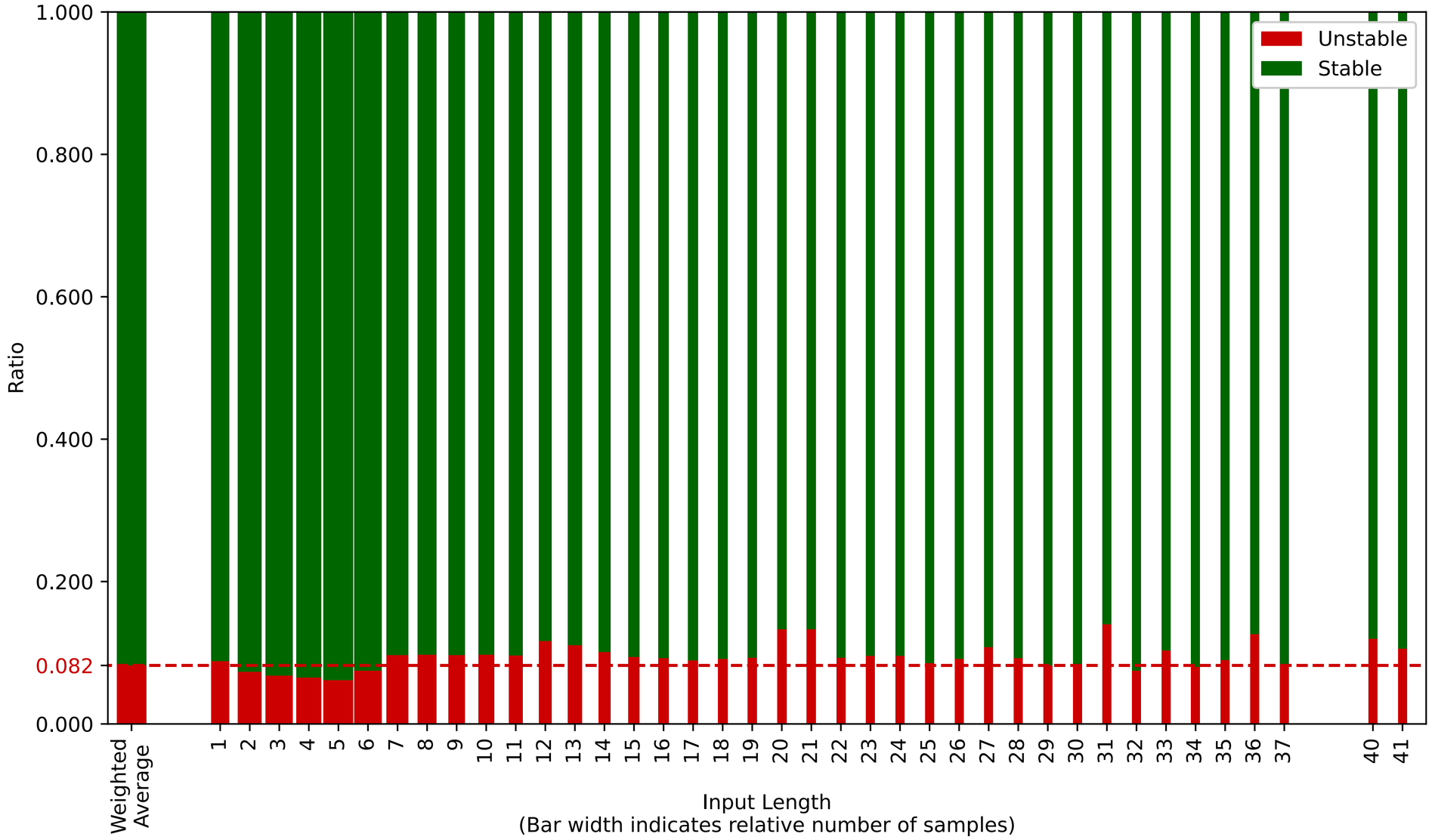
# Zephyr behaves inconsistently


Both Unstable
Coverage Unstable
State Unstable
Both Stable
Ratio
1.000
0.800
0.600
0.400
0.200
0.097
0.070
0.011
Weighted Average
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
46
Input Length
(Bar width indicates relative number of samples)

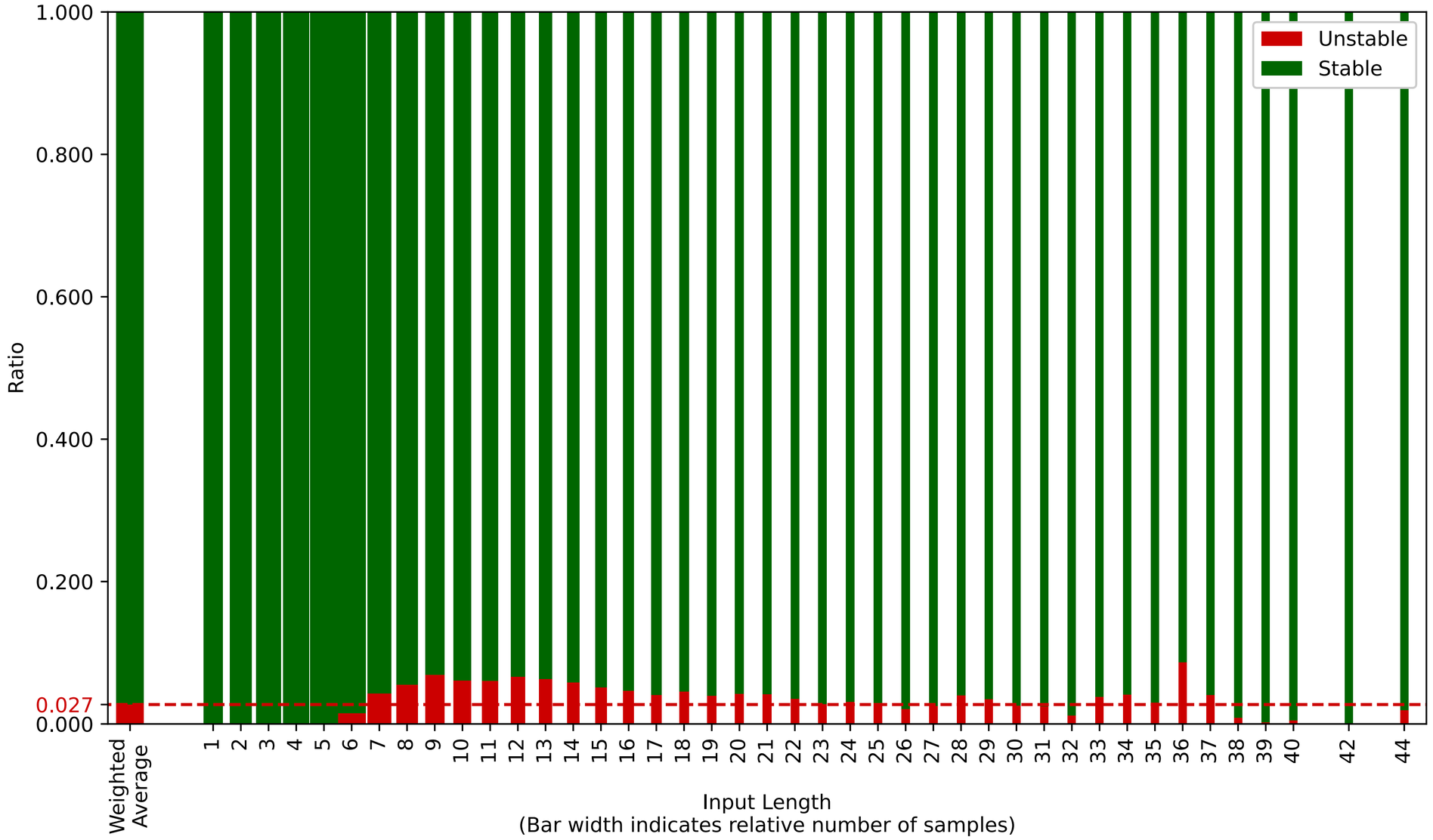
# Coverage Is Inconsistent

1. `k_work_init_delayable` from `kernel/work.c`
2. `net_ipv6_mld_init` from `net/ip/ipv6_mld.c`
3. `sys_slist_init` from `sys/slist.h`
4. `z_slist_tail_set` from `sys/slist.h`
5. `net_conn_init` from `net/ip/connection.c`

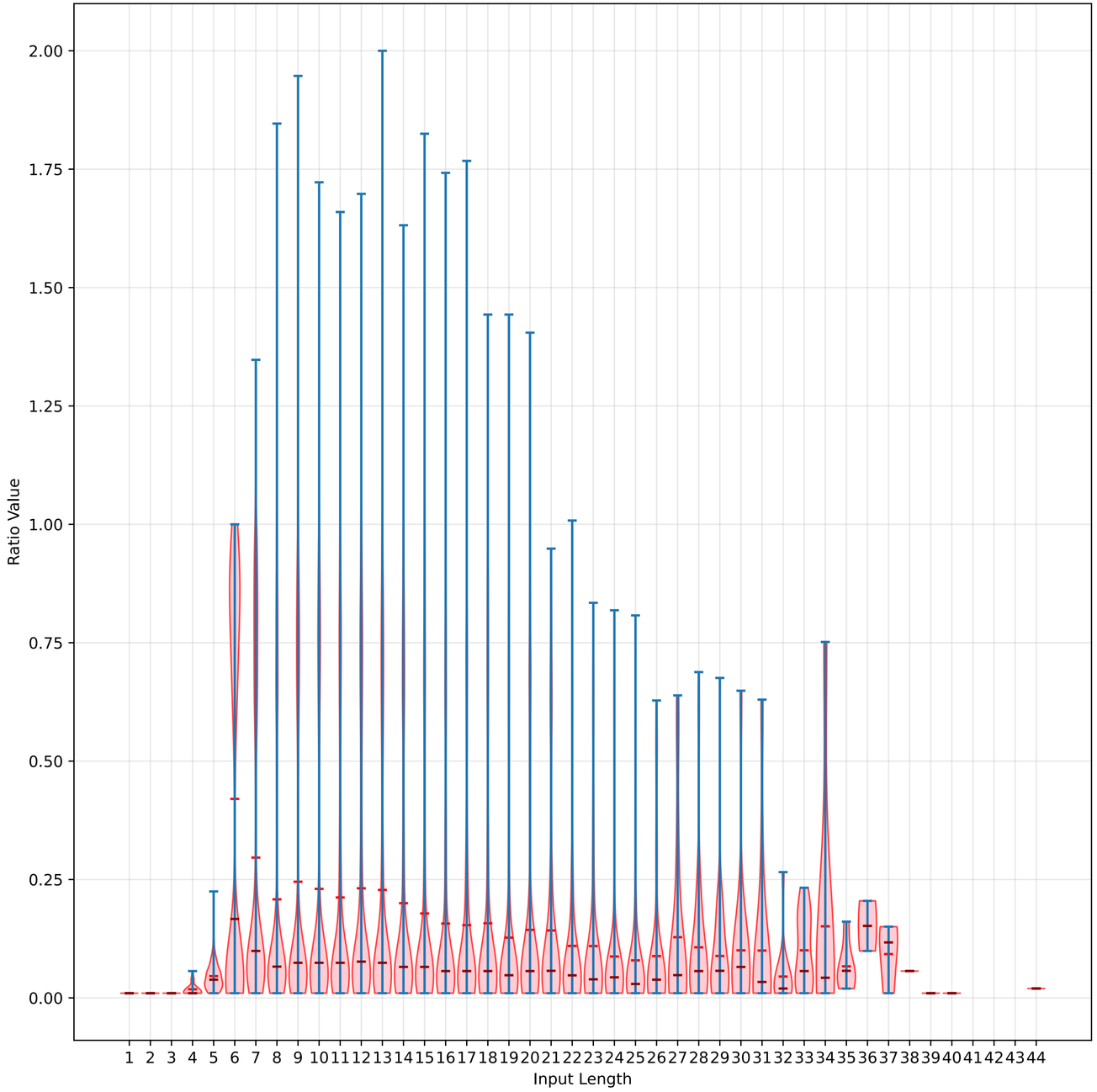
# Coverage Is Inconsistent



# Packet Responses Are Inconsistent

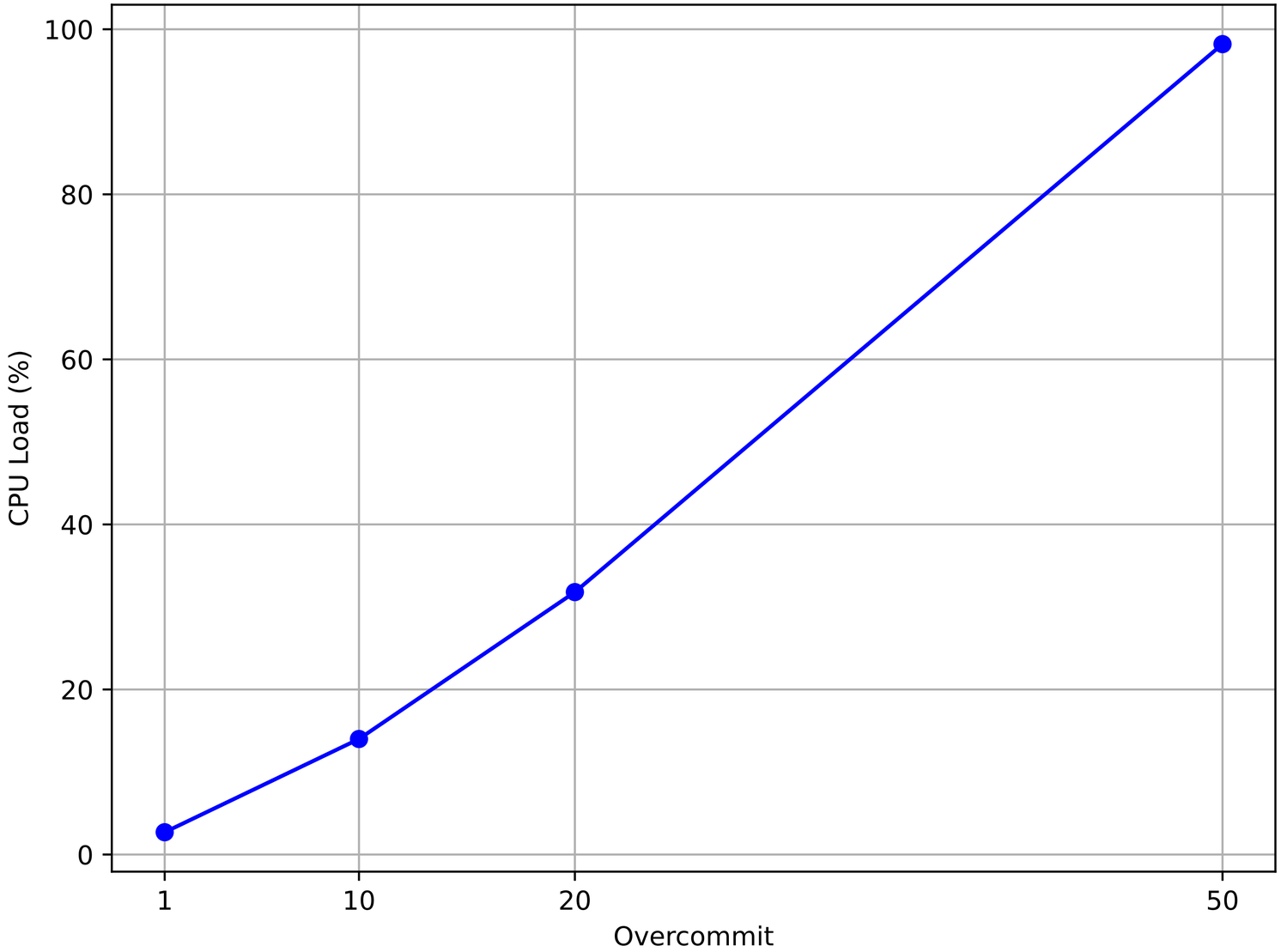


# State Is Inconsistent

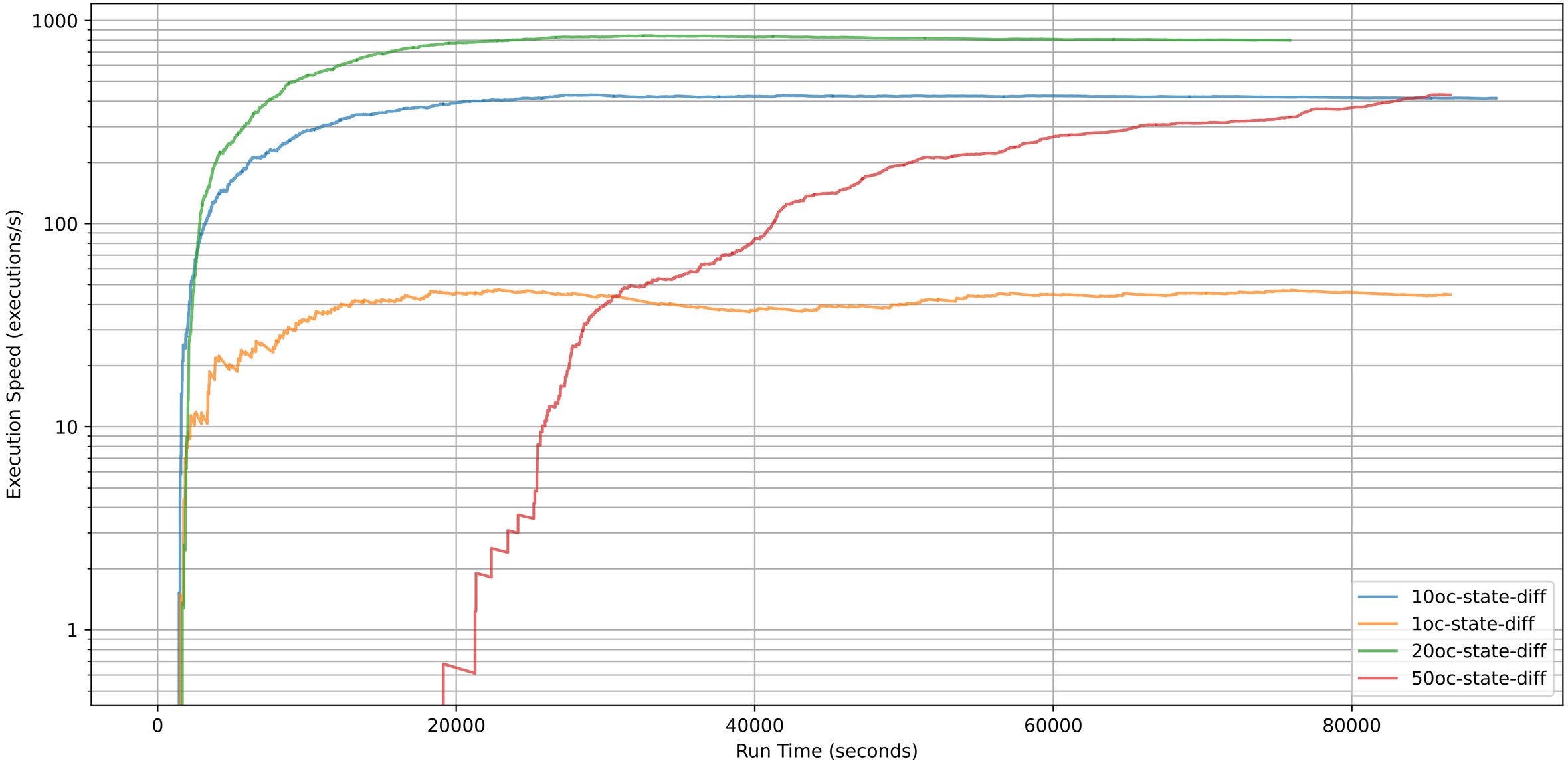


# Overcommit

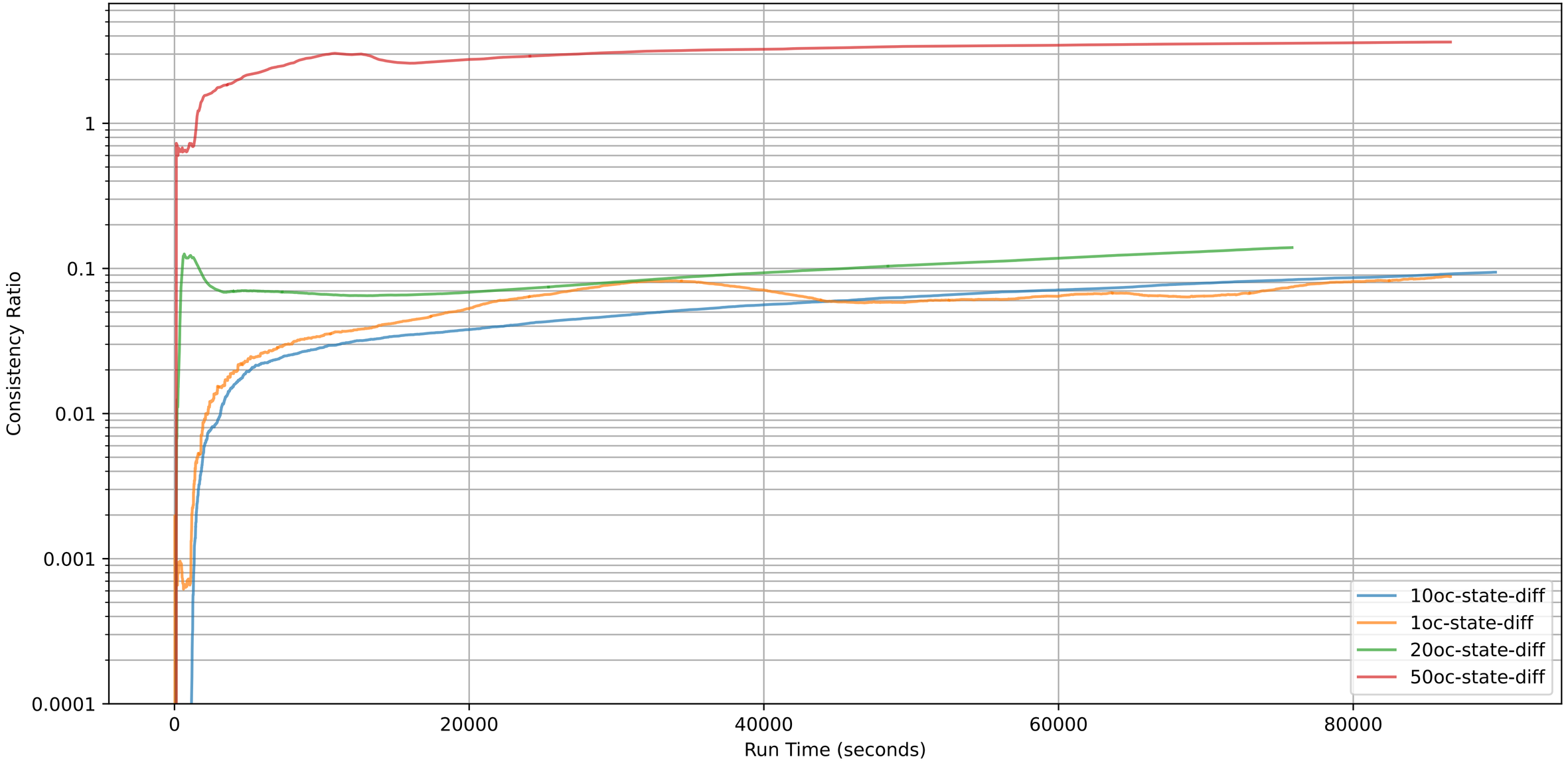
# Overcommit



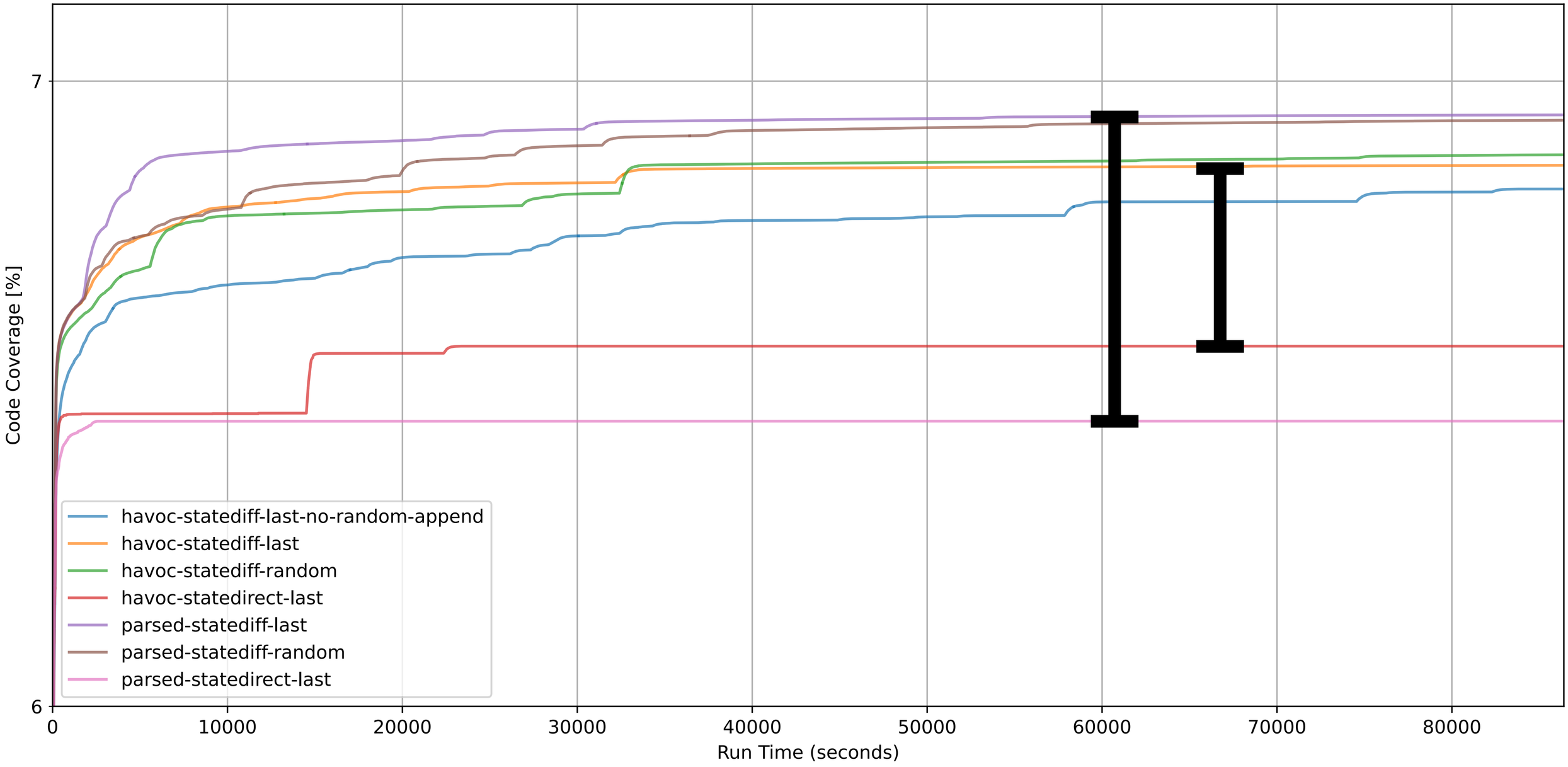
# Overcommit

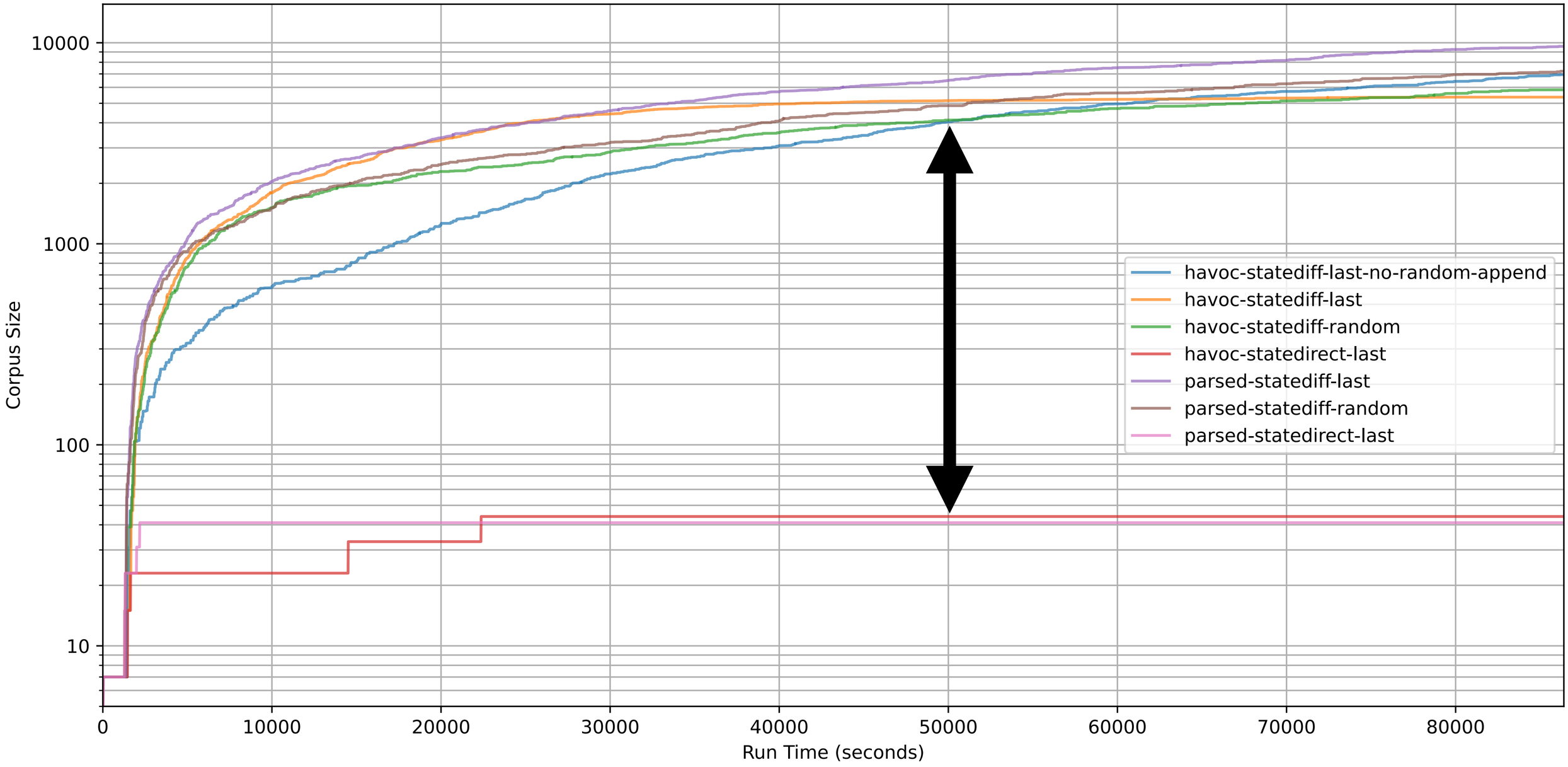

1000
100
10
1
Execution Speed (executions/s)
0
20000
40000
60000
80000
Run Time (seconds)
10oc-state-diff
1oc-state-diff
20oc-state-diff
50oc-state-diff

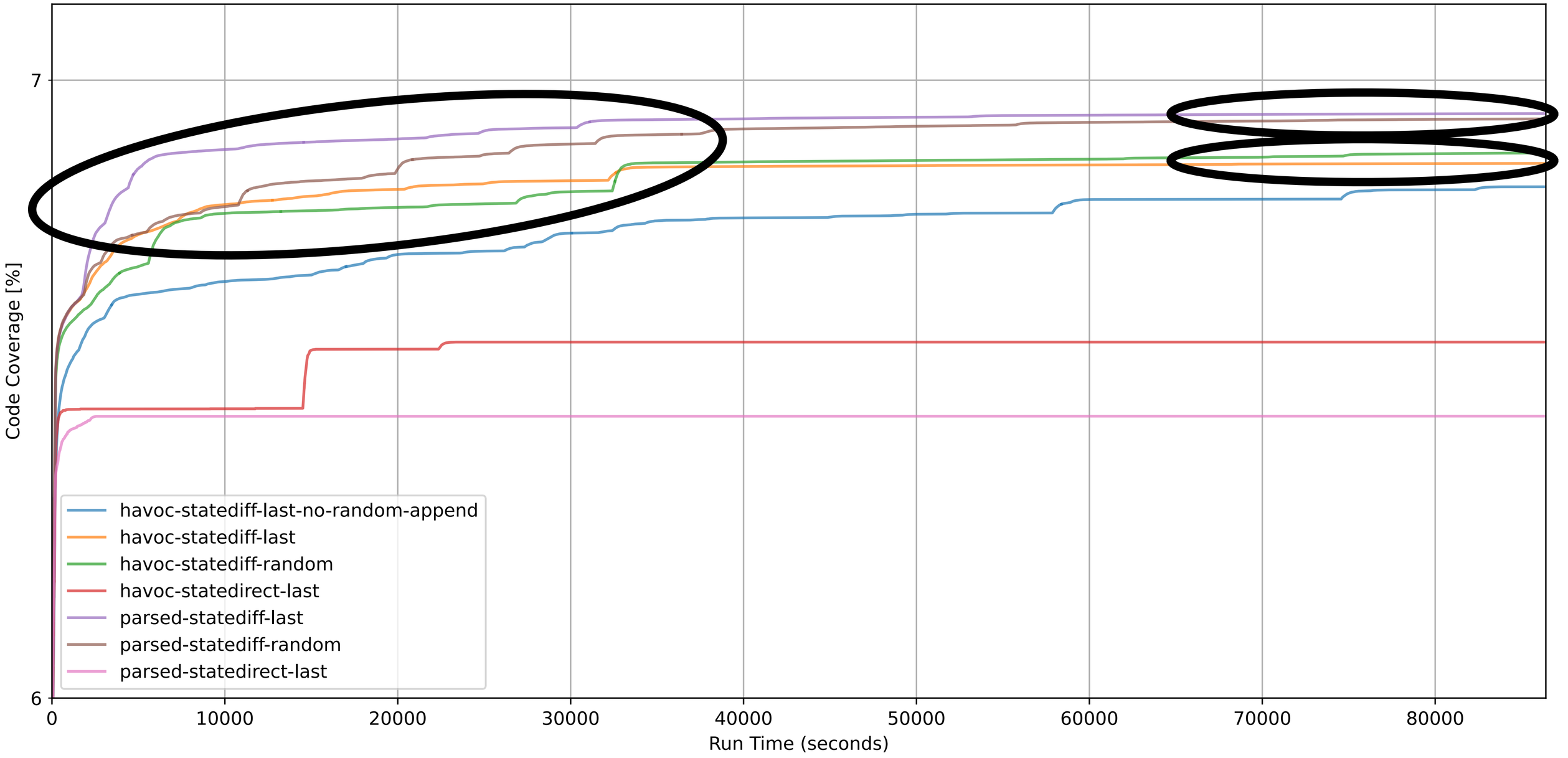
# Overcommit

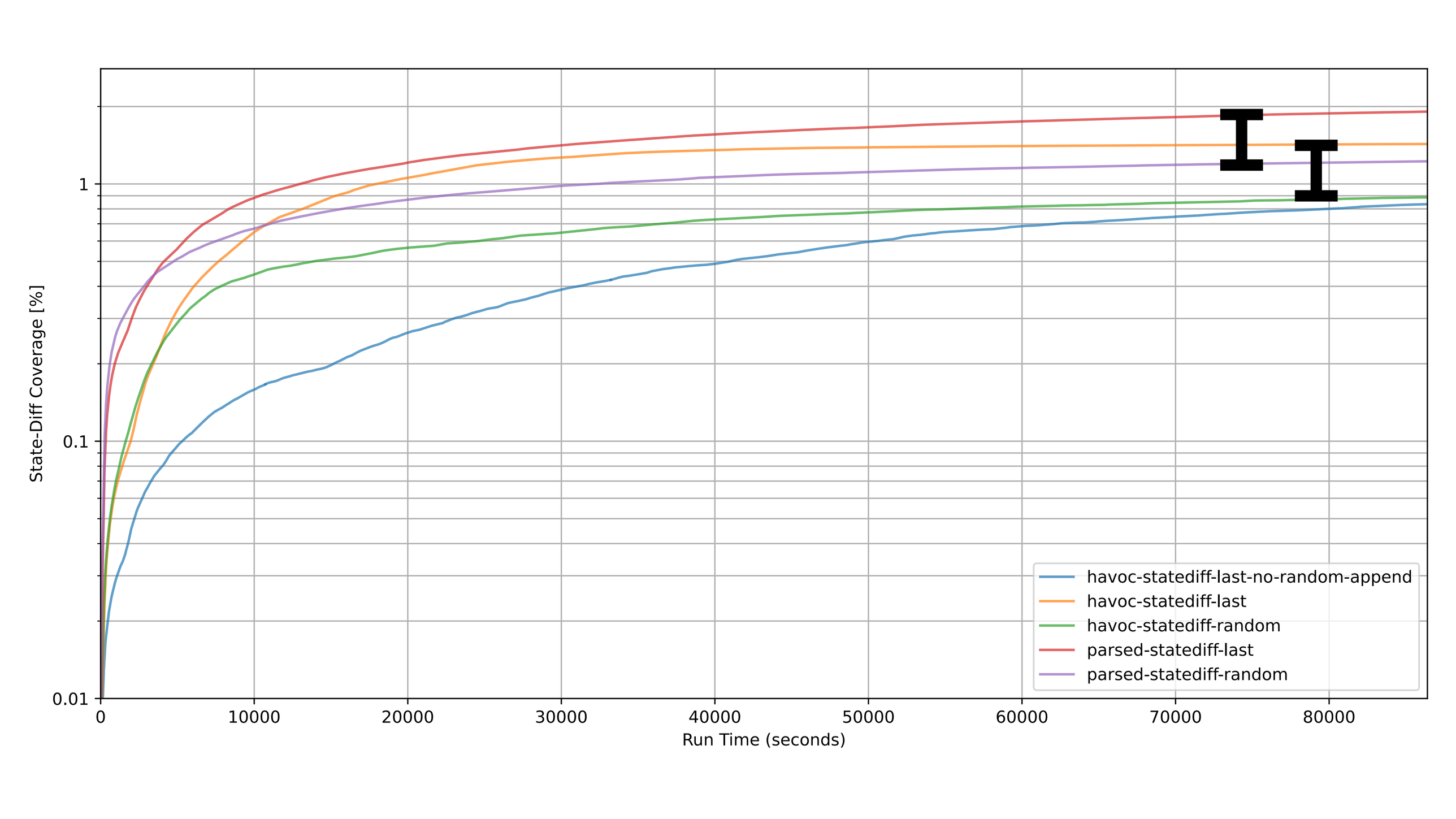

Consistency Ratio
1
0.1
0.01
0.001
0.0001
0
20000
40000
60000
80000
Run Time (seconds)
10oc-state-diff
1oc-state-diff
20oc-state-diff
50oc-state-diff

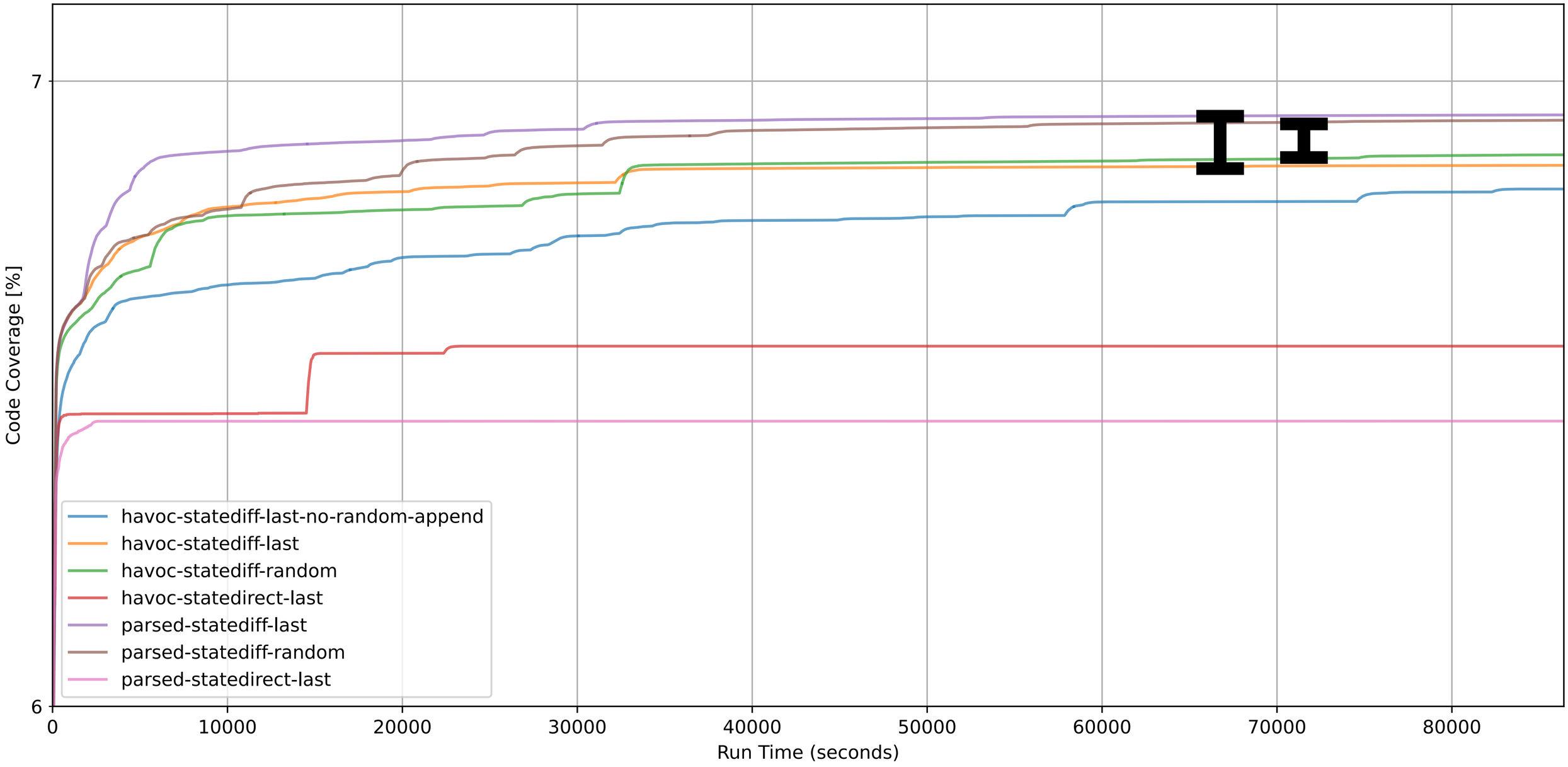
# Evaluating Input Modeling and State-Inference Feedback

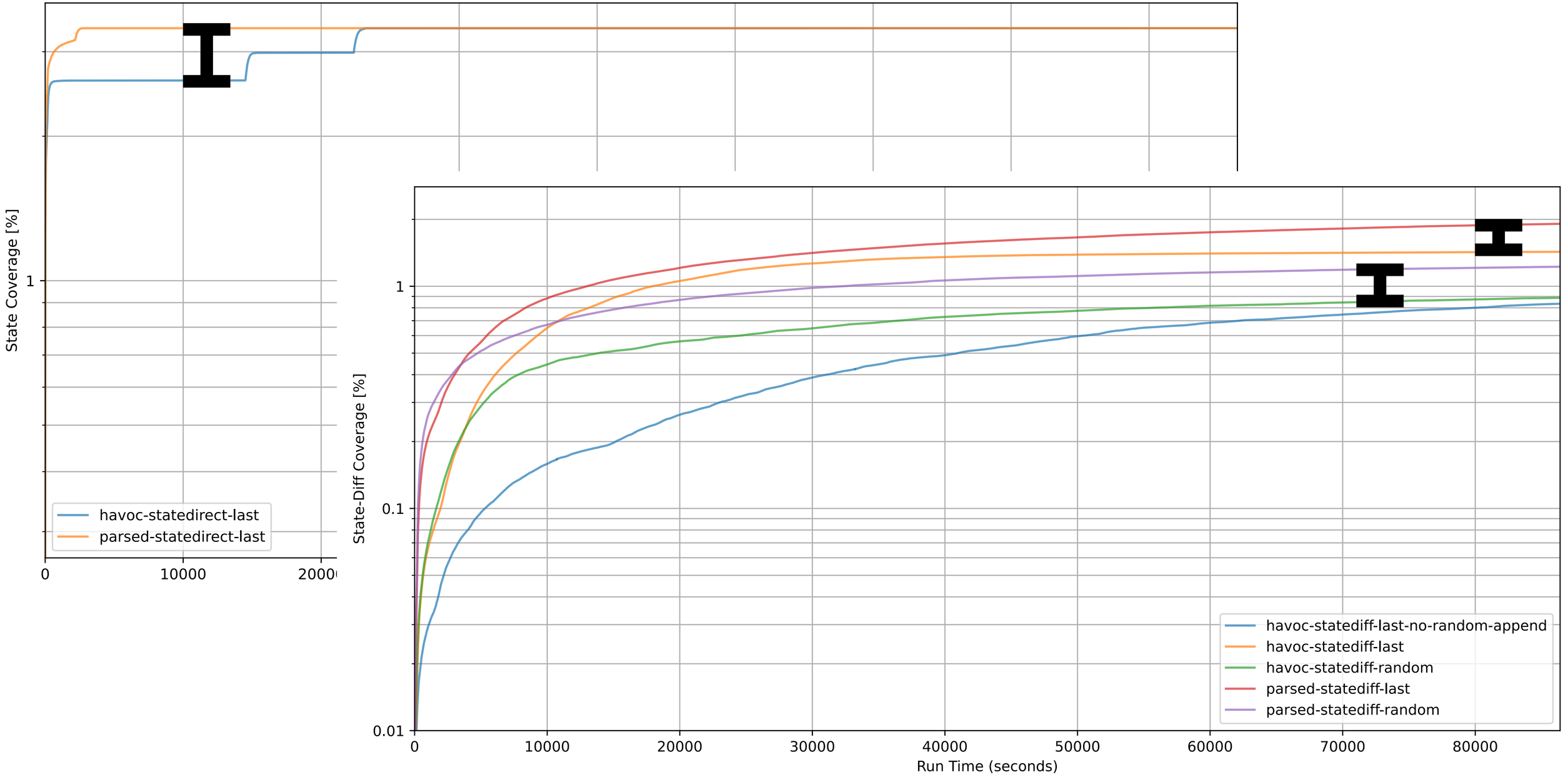

7
6
Code Coverage [%]
0
10000
20000
30000
40000
50000
60000
70000
80000
Run Time (seconds)
havoc-statediff-last-no-random-append
havoc-statediff-last
havoc-statediff-random
havoc-statedirect-last
parsed-statediff-last
parsed-statediff-random
parsed-statedirect-last

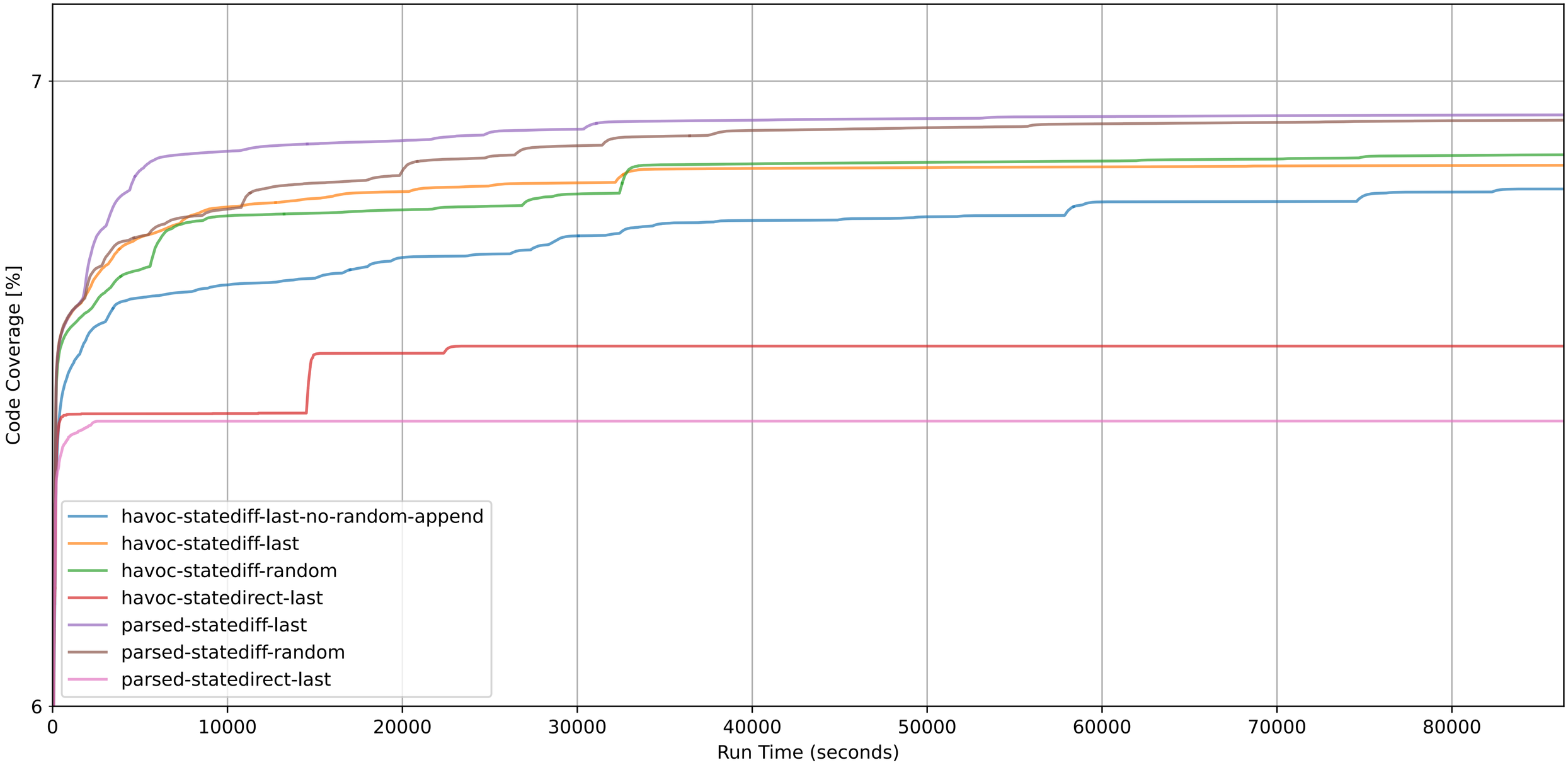

10000
1000
100
10
Corpus Size
0
10000
20000
30000
40000
50000
60000
70000
80000
Run Time (seconds)
havoc-statediff-last-no-random-append
havoc-statediff-last
havoc-statediff-random
havoc-statedirect-last
parsed-statediff-last
parsed-statediff-random
parsed-statedirect-last


7
6
Code Coverage [%]
0
10000
20000
30000
40000
50000
60000
70000
80000
Run Time (seconds)
havoc-statediff-last-no-random-append
havoc-statediff-last
havoc-statediff-random
havoc-statedirect-last
parsed-statediff-last
parsed-statediff-random
parsed-statedirect-last


State-Diff Coverage [%]
1
0.1
0.01
0
10000
20000
30000
40000
50000
60000
70000
80000
Run Time (seconds)
havoc-statediff-last-no-random-append
havoc-statediff-last
havoc-statediff-random
parsed-statediff-last
parsed-statediff-random


7
6
Code Coverage [%]
0
10000
20000
30000
40000
50000
60000
70000
80000
Run Time (seconds)
havoc-statediff-last-no-random-append
havoc-statediff-last
havoc-statediff-random
havoc-statedirect-last
parsed-statediff-last
parsed-statediff-random
parsed-statedirect-last


State Coverage [%]
1
0
10000
2000
havoc-statedirect-last
parsed-statedirect-last
State-Diff Coverage [%]
1
0.1
0.01
0
10000
20000
30000
40000
50000
60000
70000
80000
Run Time (seconds)
havoc-statediff-last-no-random-append
havoc-statediff-last
havoc-statediff-random
parsed-statediff-last
parsed-statediff-random


7
6
Code Coverage [%]
0
10000
20000
30000
40000
50000
60000
70000
80000
Run Time (seconds)
havoc-statediff-last-no-random-append
havoc-statediff-last
havoc-statediff-random
havoc-statedirect-last
parsed-statediff-last
parsed-statediff-random
parsed-statedirect-last

# Conclusion

# Future Work

github.com/riesentoaster/FTZ