Extrapolating Coverage Rate in Greybox Fuzzing

Danushka Liyanage*
University of Sydney

Seongmin Lee*
MPI-SP

Chakkrit Tantithamthavorn
Monash University

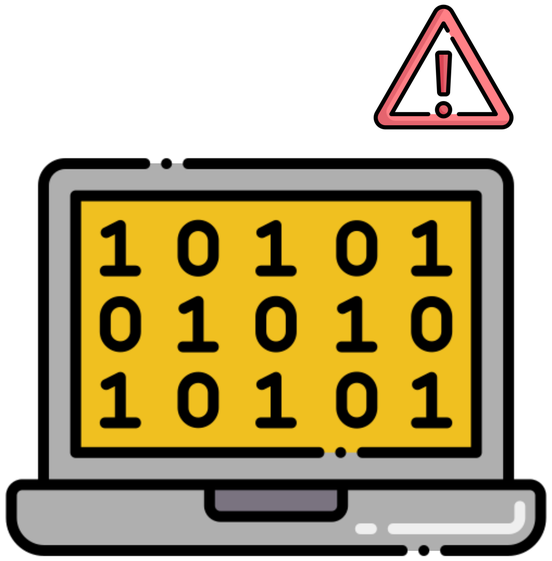
Marcel Böhme
MPI-SP

ICSE 2024

* Co-first Authors

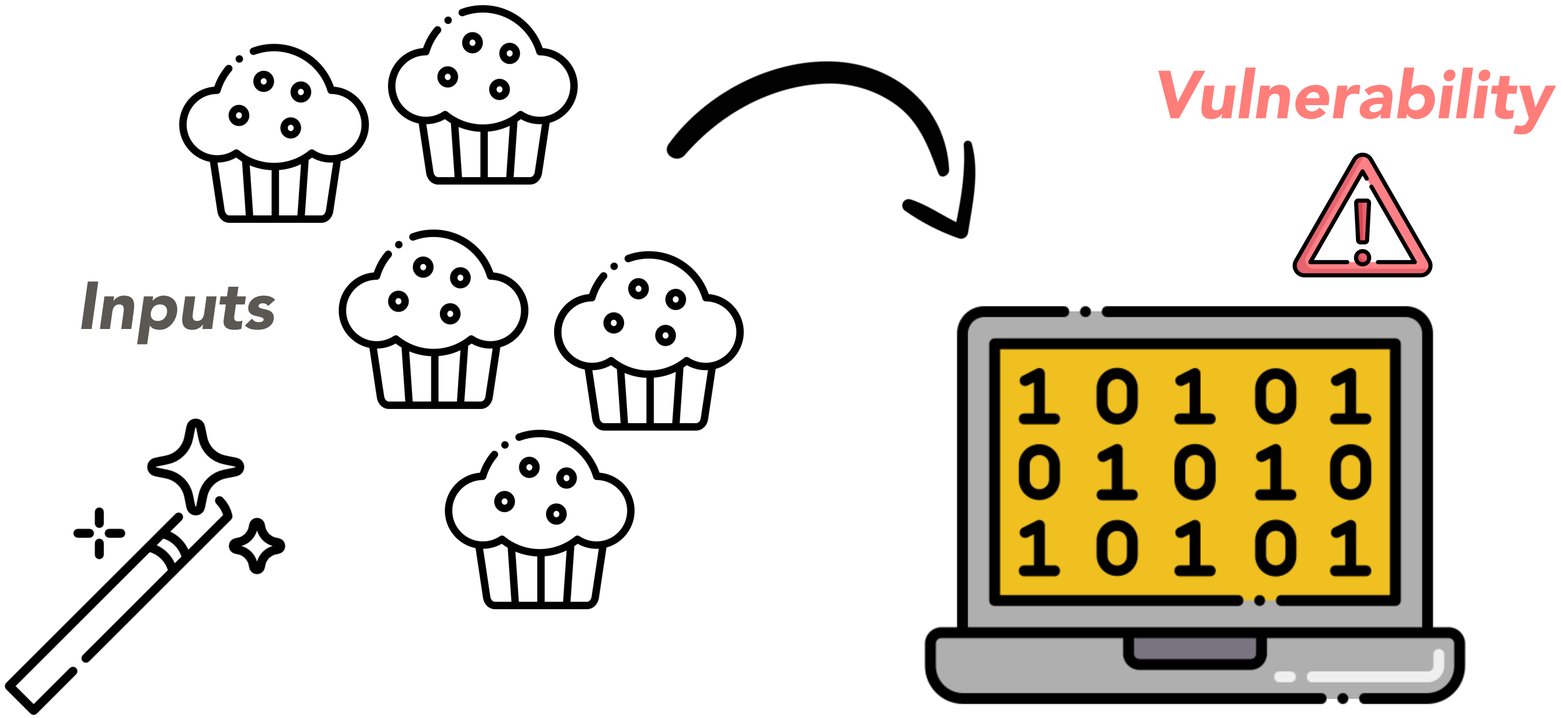
Fuzzing

Vulnerability

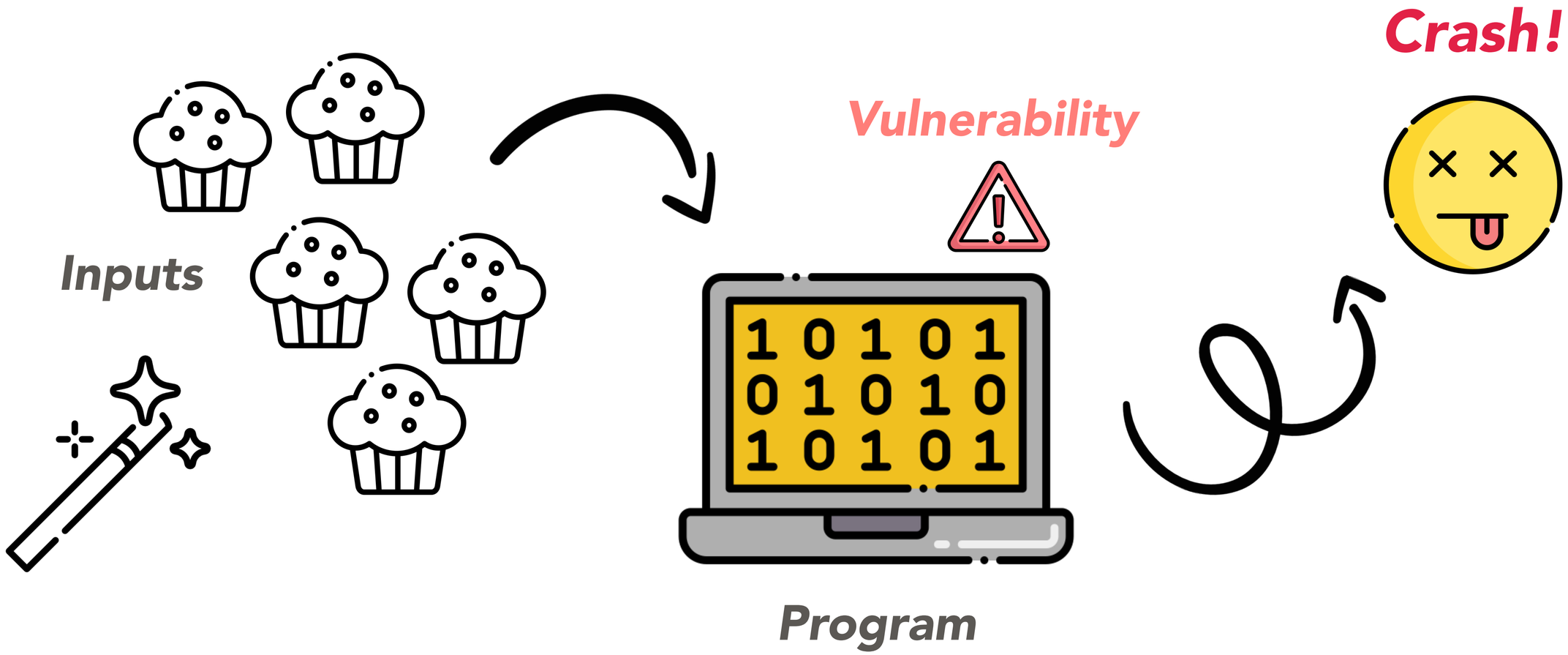


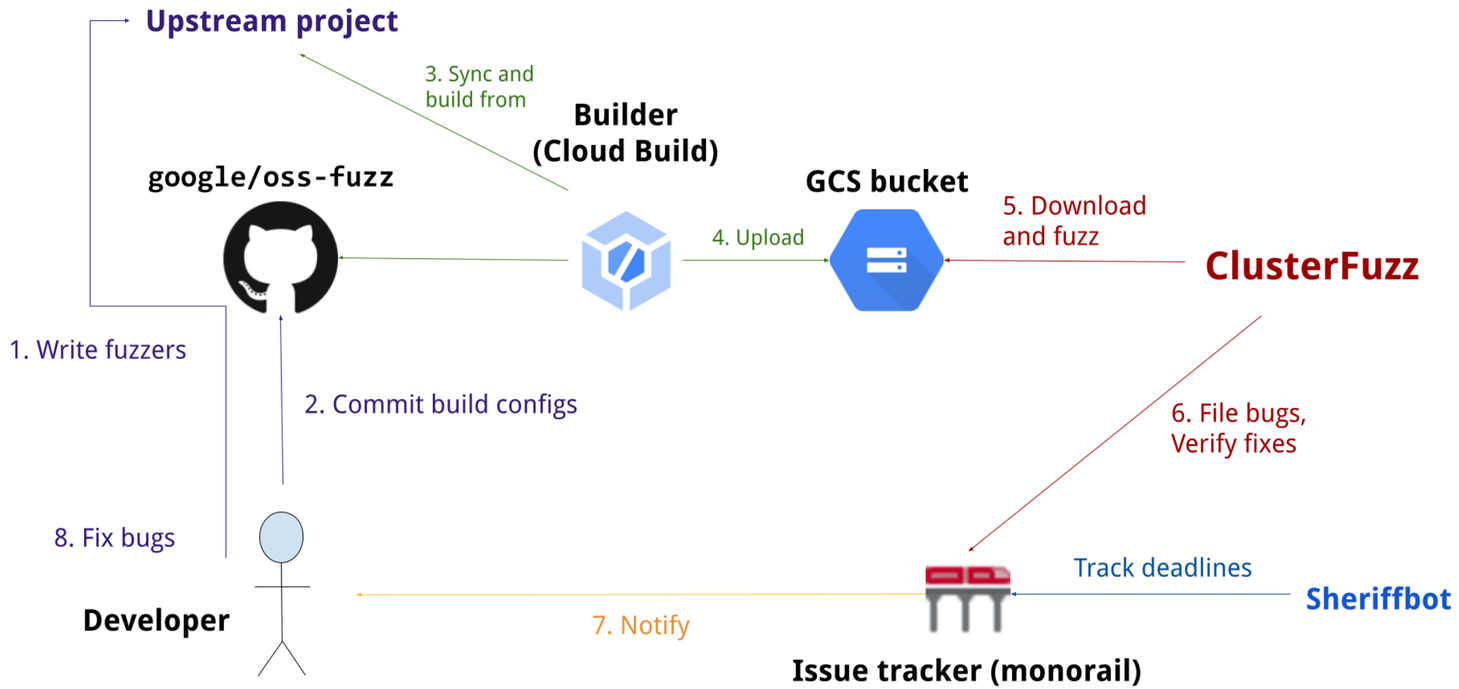
Program

Fuzzing



Fuzzing





OSS-Fuzz

- 10,000 vulnerabilities and 36,000 bugs across 1,000 open-source projects. (~August 2023)

OSS-FUZZ: Continuous Fuzzing for Open Source Software

★ FuzzSlice: Pruning False Positives in Static Analysis Warnings through Function-Level Fuzzing

Manual confirmation of static analysis reports is
of false positives among them. Fuzzing techniqu
that fuzzing the whole project to reach all static a
time to increase code coverage linearly. Therefor
positives among static analysis warnings. Unlike
warnings, which inevitably requires end-to-end f
in static analysis reports. The key insight that we
function level in a given time budget is most likel
slices at th
FuzzSlice
function le
openssh-p
FuzzSlice,
independe
developer
repositorie

★ FuzzInMem: Fuzzing Programs via In-memory Structures

In recent years, coverage-based greybox fuzzing has proven to be an effective technique for discovering software vulnerabilities. The availability of American Fuzzy Loop (AFL) has facilitated numerous advances in overcoming challenges in fuzzing. However, the issue of mutating complex file formats, such as PDF, remains unresolved due to strict constraints. Existing fuzzers often produce mutants that fail to parse by applications, limited by bit/byte mutations performed on input files. Our observation is that most in-memory representations of file formats are simple, and well-designed applications have built-in printer functions to emit these structures as files. Thus, we propose a new technique that mutates the in-memory structures of inputs and utilizes printer functions to regenerate mutated

que leverages the printer function
as well as other state-of-the-art
ble and substantially outperforms
n to real-world applications, we

★ Are We There Yet? Unraveling the State-of-the-Art Smart Contract Fuzzers

Given the growing importance of smart contracts in various applications, ensuring their security and reliability is critical. Fuzzing, an effective vulnerability detection technique, has recently been widely applied to smart contracts. Despite numerous studies, a systematic investigation of smart contract fuzzing techniques remains lacking. In this experience paper, we fill this gap by: 1) providing a comprehensive review of current research in contract fuzzing, and 2) conducting an in-depth empirical study to evaluate state-of-the-art contract fuzzers' usability. To guarantee a fair evaluation, we employ a carefully-labeled benchmark and introduce a set of pragmatic performance metrics, evaluating fuzzers from five complementary perspectives. Based on our findings, we provide direction for the future research and development of contract fuzzers.

do so

ACM SIGSOFT Distinguished Paper Award

vulnerability of 2019. However, there are few automated tools—either in research or industry—to effectively find and remediate such issues. This is unsurprising as the problem lacks an explicit test oracle: the vulnerability does not manifest through explicit abnormal behaviours (e.g., program crashes or memory access violations).

that we call EDEFuzz—to
nspection and ad-hoc text-

otential leaks—illustrating our
s in Australia, EDEFuzz
nd efficiency.

vulnerabilities. The
However, the issue of
roduce mutants that
n-memory
ese structures as files.
to regenerate mutated
s the printer function
other state-of-the-art
hat FuzzInMem is scalable and substantially outperforms
we

★ Crossover in Parametric Fuzzing

Parametric fuzzing combines evolutionary and generator-based fuzzing to create structured test inputs that exercise unique execution behaviors. Parametric fuzzers internally represent inputs as bit strings referred to as "parameter sequences". Interesting parameter sequences are saved by the fuzzer and perturbed to create new inputs without the need for type-specific operators. However, existing work on parametric fuzzing only uses mutation operators, which modify a single input; it does not incorporate crossover, an evolutionary operator that blends multiple inputs together. Crossover operators aim to combine advantageous traits from multiple inputs. However, the nature of parametric fuzzing limits the effectiveness of traditional crossover operators. In this paper, we propose linked crossover, an approach for using dynamic execution information to identify and exchange analogous portions of parameter sequences. We created an implementation of linked crossover for Java and evaluated linked crossover's ability to preserve advantageous traits. We also evaluated linked crossover's impact on fuzzer performance on seven real-world Java projects and found that linked crossover consistently performed as well as or better than three state-of-the-art parametric fuzzers and two other forms of crossover on both long and short fuzzing campaigns.

ems

f configuration parameters. This leads to a
n space. Existing configuration testing
ctly inject them into the program under test to
find configuration-induced bugs. However, they do not fully consider the complexity of large-scale systems, resulting in low testing effectiveness. In this paper, we propose ECFuzz, an effective configuration fuzzer for large-scale systems. Our core approach consists of (i) Multi-dimensional configuration generation strategy. ECFuzz first designs different mutation strategies according to different dependencies and selects multiple configuration parameters from the candidate configuration parameters to effectively generate configuration parameters; (ii) Unit-testing-oriented configuration validation strategy. ECFuzz introduces unit testing into configuration testing techniques to filter out configuration parameters that are unlikely to yield errors before executing system testing, and effectively validate generated configuration parameters. We have conducted extensive experiments in real-world large-scale systems including HCommon, HDFS, HBase, ZooKeeper and Alluxio. Our evaluation shows that ECFuzz is effective in finding configuration-induced crash bugs. Compared with the state-of-the-art configuration testing tools including ConfTest, ConfErr and ConfDiagDetector, ECFuzz finds 60.3–67 more unexpected failures when the same 1000 testcases are injected into the system with an increase of 1.87x–2.63x. Moreover, ECFuzz has exposed 14 previously unknown bugs, and 5 of them have been confirmed.

Fuzzing has achieved tremendous s
(SUTs) that take in programming or
with accessible APIs, are especially
fuzzers for such systems often targe
of the same language. Moreover, th
thus can hardly reveal bugs related
sense that it can target many differe
to leverage large language models (
and realistic inputs for any practicall
creates LLM prompts that are well-s
create new fuzzing inputs. We evalu
and Python) as inputs. The evaluation shows, across all six languages, that universal fuzzing achieves higher coverage than existing, language-specific fuzzers. Furthermore, Fuzz4All has identified 76 bugs in widely used systems, such as GCC, Clang, Z3, CVC5, OpenJDK, and the Qiskit quantum computing platform, with 47 bugs already confirmed by developers as previously unknown.

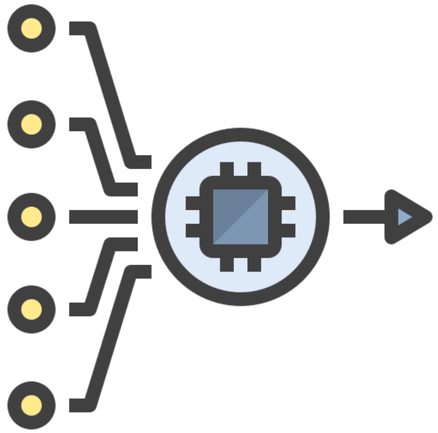
Fuzzing workshop 2024

- **How** can we make the fuzzer **smarter** at generating useful inputs?

- **How** can we make the fuzzer **smarter** at generating useful inputs?
- **What/Where** else can we apply the fuzzing?

Q. When should we stop the fuzzing campaign?

Every possible Input?



Well, it's simply infeasible...

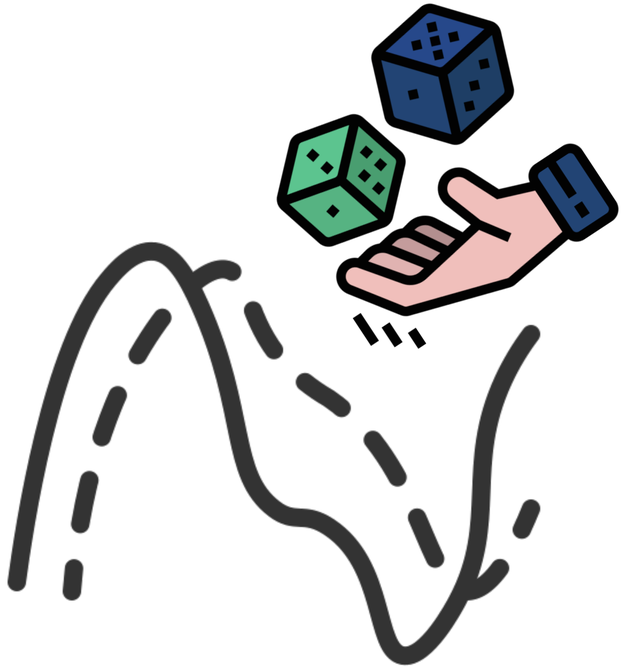
100% Coverage?

Which level? Line? Method? Class?
What about unreachable elements?

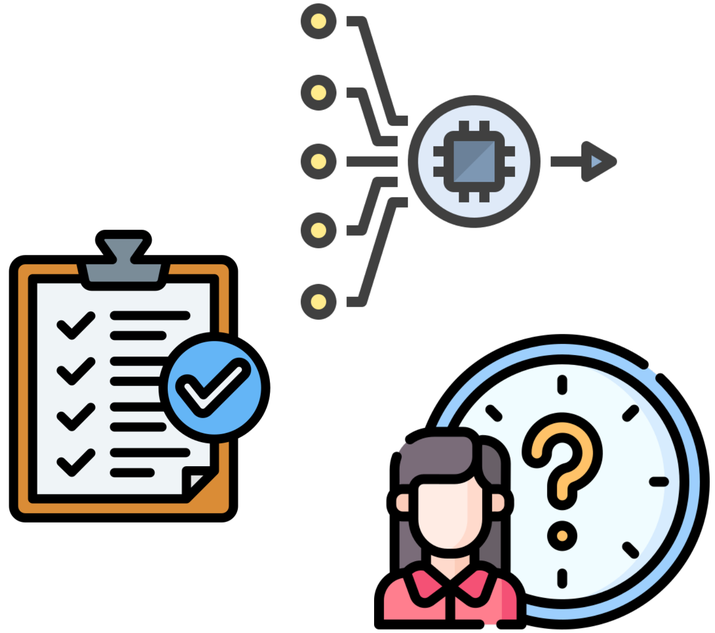
More importantly,

Even if we have some criteria,
how much of a resource do we need to use more?

Q. When should we stop the fuzzing campaign?



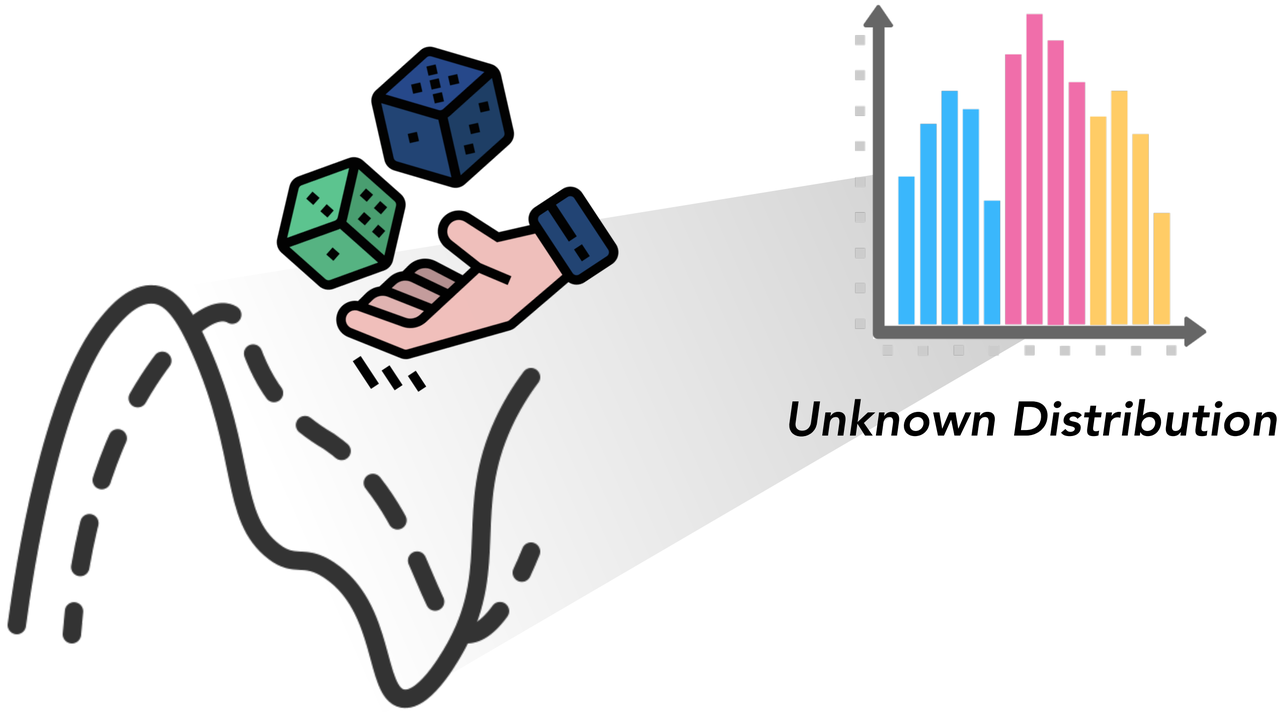
Stochastic Process



Even if we have some criteria,

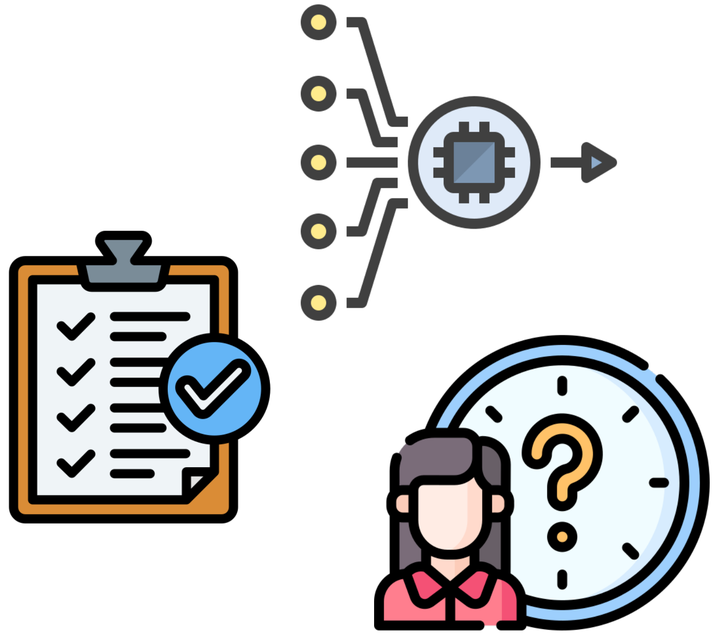
how much of a resource do we need to use more?

Q. When should we stop the fuzzing campaign?



Unknown Distribution

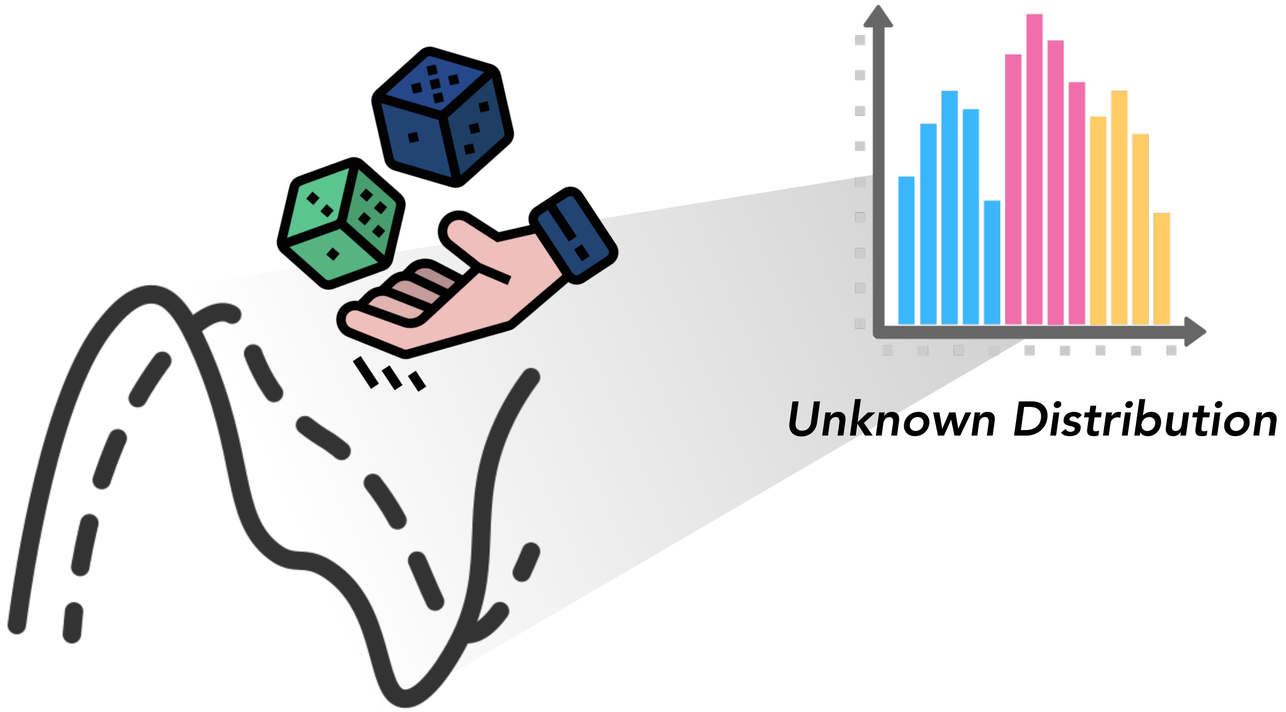
Stochastic Process



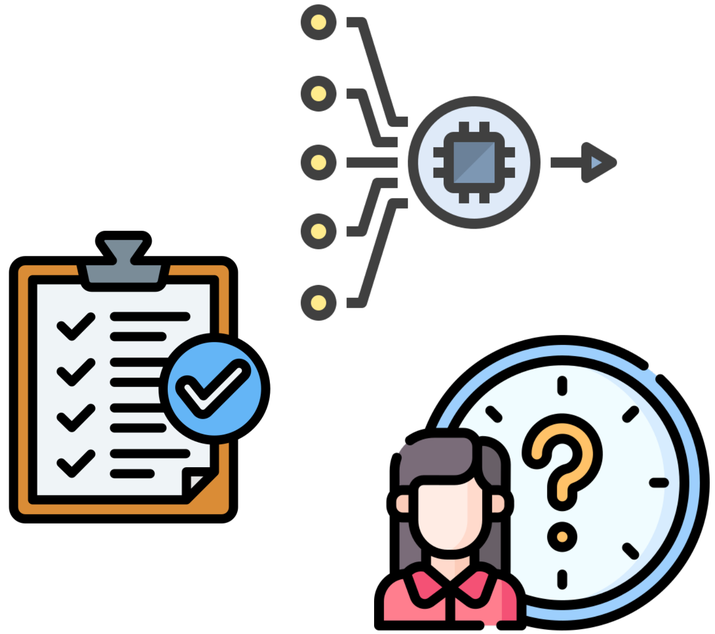
Even if we have some criteria,

how much of a resource do we need to use more?

Q. When should we stop the fuzzing campaign?



Stochastic Process



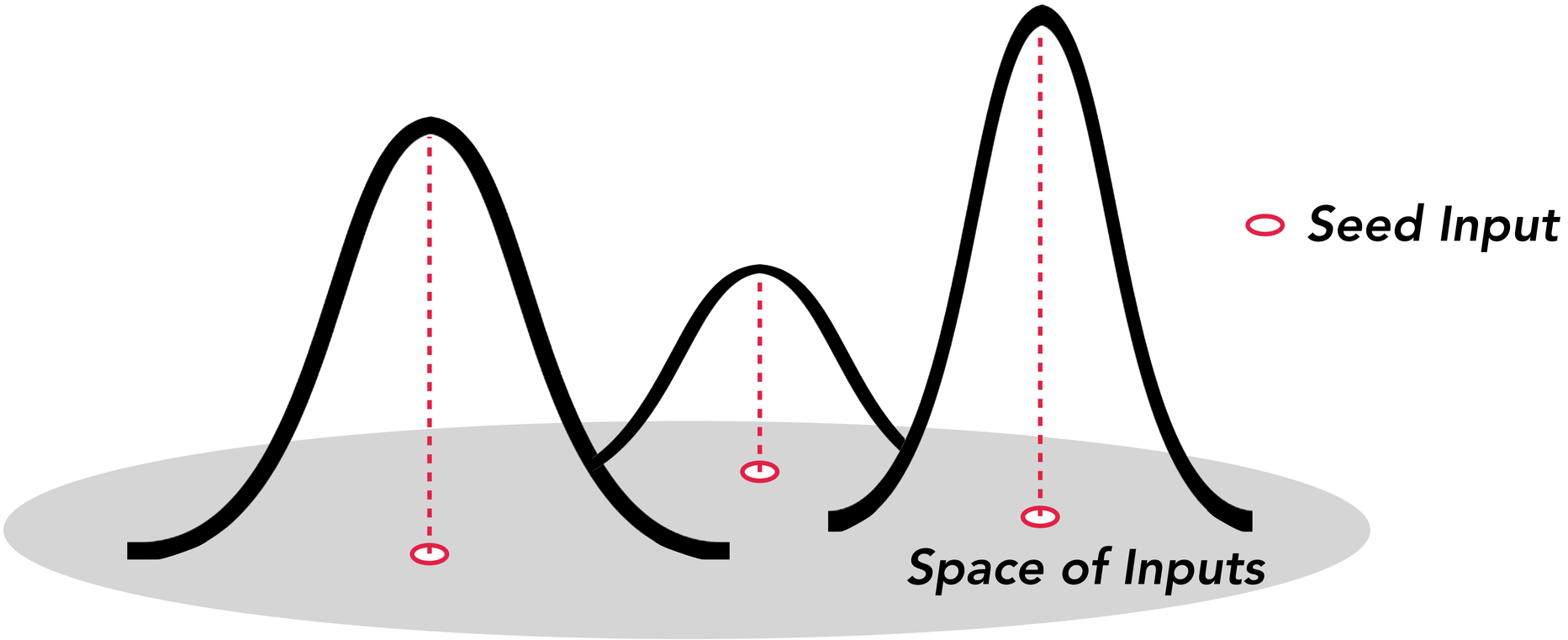
Even if we have some criteria,

how much of a resource do we need to use more?

Unsolvable with a formal/analytic method

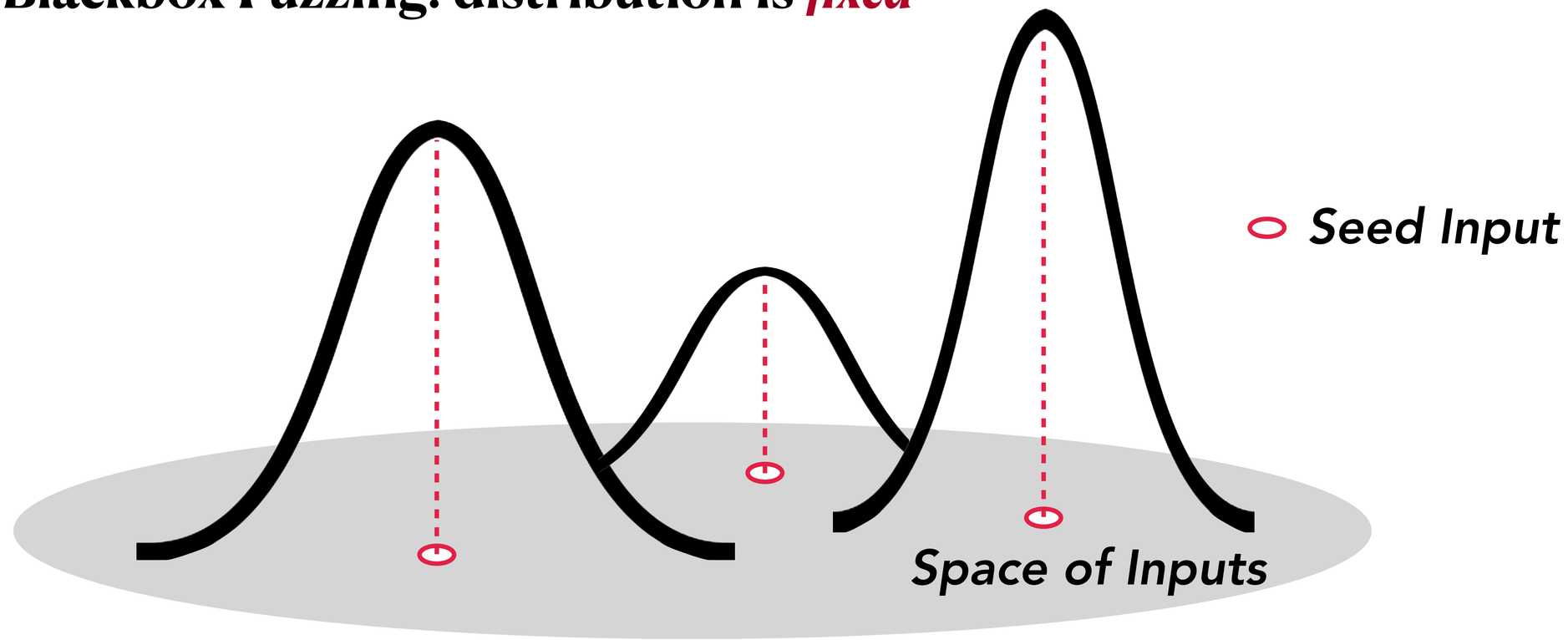
Fuzzing in terms of Statistics

Fuzzing in terms of Statistics



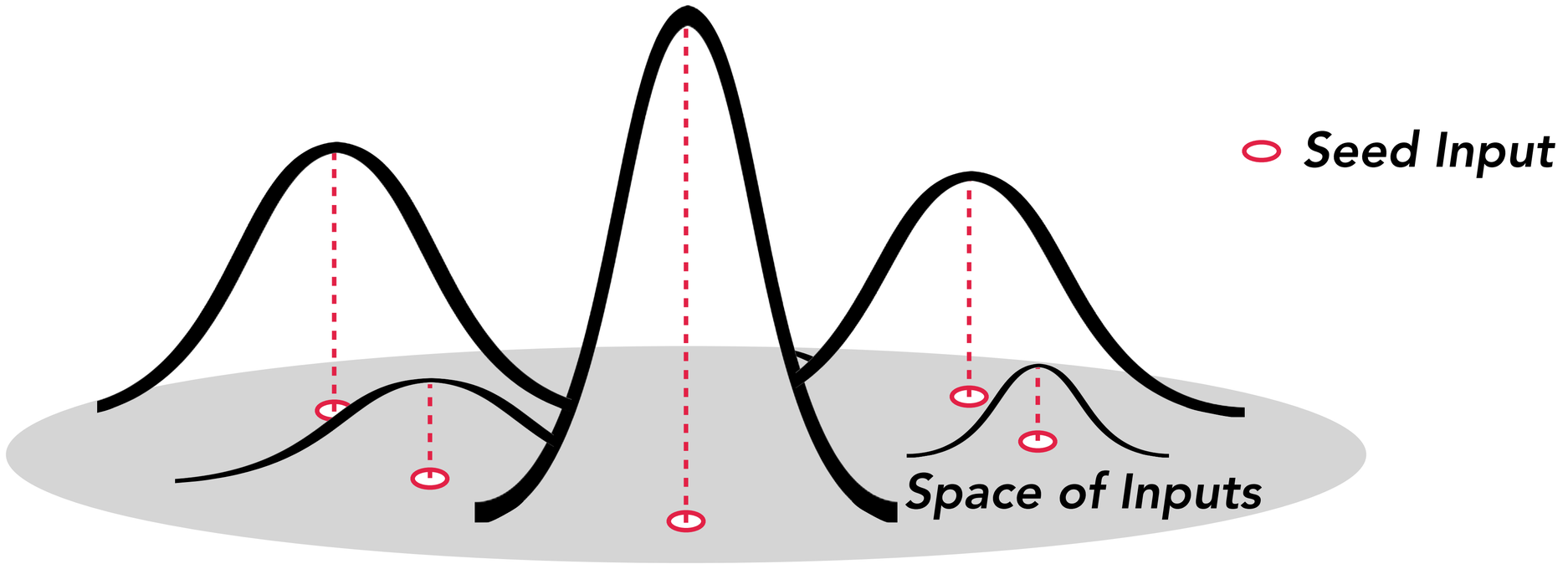
Fuzzing in terms of Statistics

Blackbox Fuzzing: distribution is *fixed*



Fuzzing in terms of Statistics

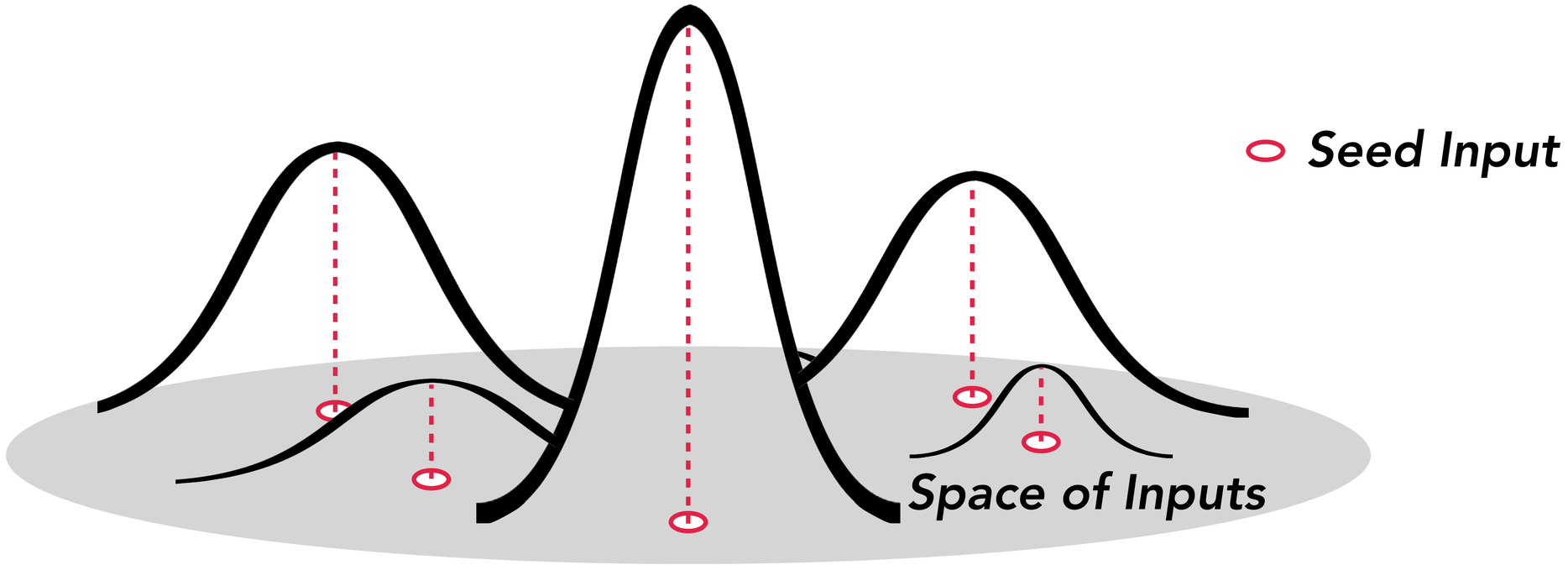
Greybox Fuzzing: distribution *changes* as time goes on



Fuzzing in terms of Statistics

Adaptive bias

Greybox Fuzzing: distribution *changes* as time goes on



Q. When should we stop the fuzzing campaign?

Statistically Extrapolating the Blackbox Fuzzing

- Marcel Böhme, “STADS: Software testing as species discovery.” TOSEM, 2018.

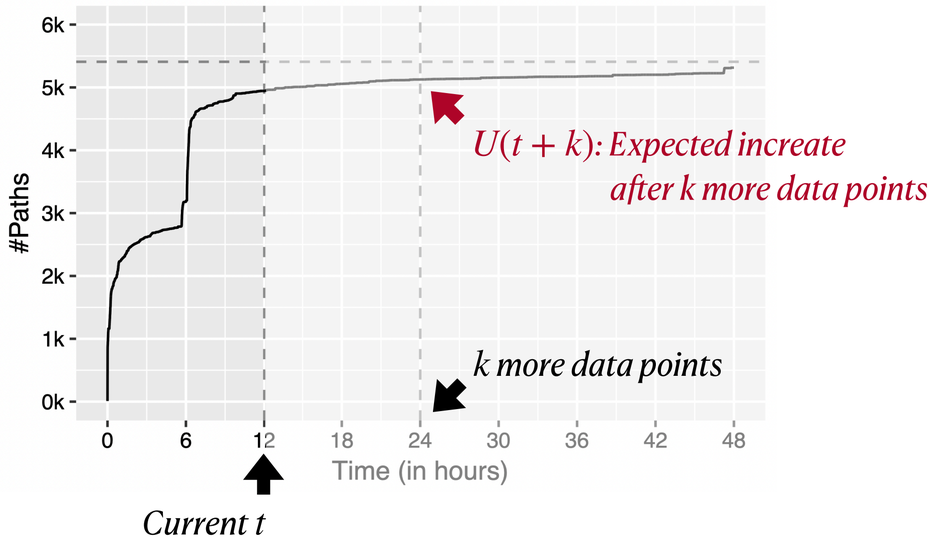
  *“A random sampling process of a **fixed distribution**, as a **Blackbox Fuzzing**, can be extrapolated statistically.”*

Anne Chao and Lou Jost. 2012. “Coverage-based rarefaction and extrapolation: standardizing samples by completeness rather than size.” Ecology 93, 12 (2012)

Statistically Extrapolating the Blackbox Fuzzing

- Provides statistical estimators for a **coverage rate**
  - Coverage rate $U(t)$: the expected number of newly covered elements in the $(t+1)$-th data point.
  - Extrapolation of **coverage rate** $U(t+k)$

$$\hat{U}(t+k) = \hat{\Phi}_0 \left[1 - \left(1 - \frac{\Phi_1}{t\hat{\Phi}_0 + \Phi_1} \right)^{k+1} \right] \quad \textbf{, where } \hat{\Phi}_0 = \frac{t-1}{t} \frac{\Phi_1^2}{2\Phi_2}$$



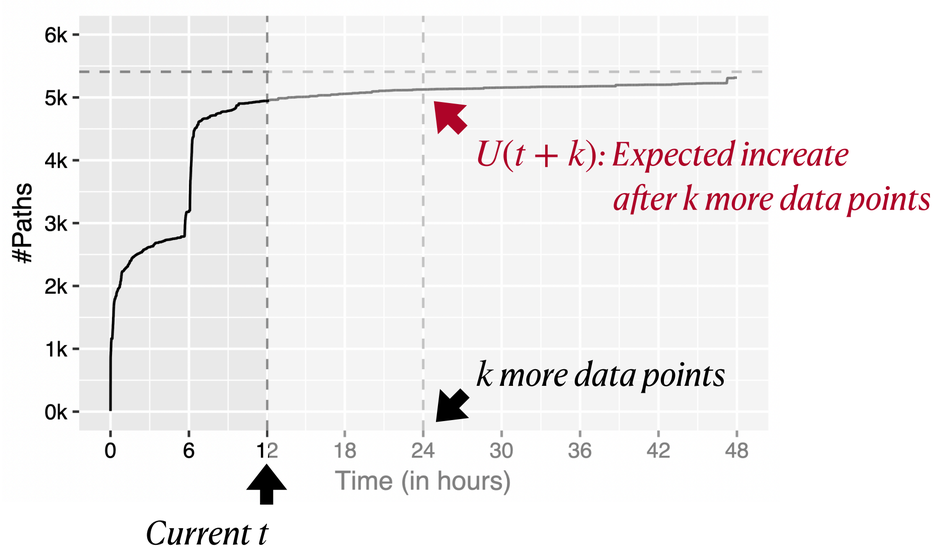
Anne Chao and Lou Jost. 2012. "Coverage-based rarefaction and extrapolation: standardizing samples by completeness rather than size." Ecology 93, 12 (2012)

Statistically Extrapolating the Blackbox Fuzzing

- Provides statistical estimators for a **coverage rate**
  - Coverage rate $U(t)$: the expected number of newly covered elements in the $(t+1)$-th data point.
  - Extrapolation of **coverage rate** $U(t+k)$

$$\hat{U}(t+k) = \hat{\Phi}_0 \left[1 - \left(1 - \frac{\Phi_1}{t\hat{\Phi}_0 + \Phi_1} \right)^{k+1} \right] \quad \textbf{, where } \hat{\Phi}_0 = \frac{t-1}{t} \frac{\Phi_1^2}{2\Phi_2}$$

  - Φ_k : number of program elements covered k times in t data points (*frequencies of frequency*)
    - Φ_1 = number of singleton elements,
      Φ_2 = number of doubleton elements



Anne Chao and Lou Jost. 2012. "Coverage-based rarefaction and extrapolation: standardizing samples by completeness rather than size." Ecology 93, 12 (2012)

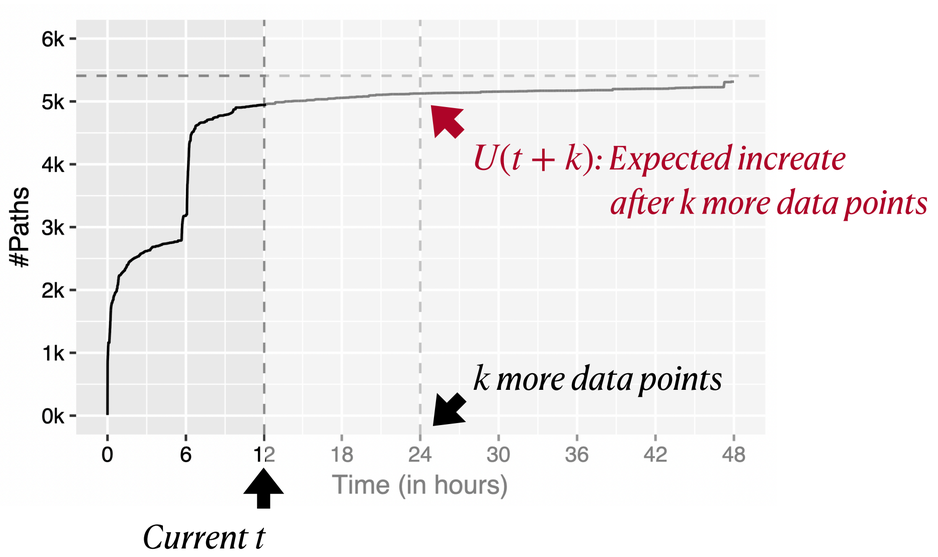
Statistically Extrapolating the Blackbox Fuzzing

- Provides statistical estimators for a **coverage rate**
  - Coverage rate $U(t)$: the expected number of newly covered elements in the $(t+1)$-th data point.

- Extrapolation of **coverage rate** $U(t+k)$

$$\hat{U}(t+k) = \hat{\Phi}_0 \left[1 - \left(1 - \frac{\Phi_1}{t\hat{\Phi}_0 + \Phi_1} \right)^{k+1} \right] \quad \textbf{, where } \hat{\Phi}_0 = \frac{t-1}{t} \frac{\Phi_1^2}{2\Phi_2}$$

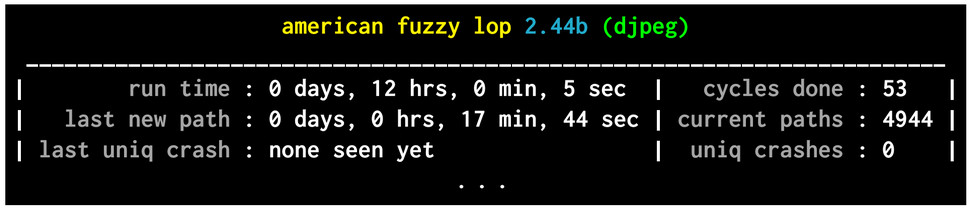
- Φ_k : number of program elements covered k times in t data points (*frequencies of frequency*)
  - Φ_1 = number of singleton elements,
    Φ_2 = number of doubleton elements



Anne Chao and Lou Jost. 2012. "Coverage-based rarefaction and extrapolation: standardizing samples by completeness rather than size." Ecology 93, 12 (2012)

Statistically Extrapolating the Blackbox Fuzzing

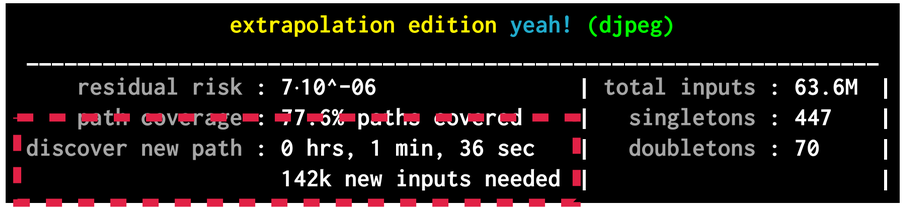
Without Extrapolation



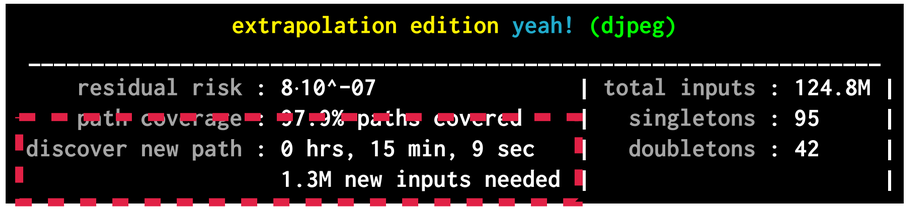
12 hours of running, the last new path was 17 minutes ago.
... should I stop this fuzzing?

VS

With Extrapolation



A new path will come in 2 minutes? Let's keep going!

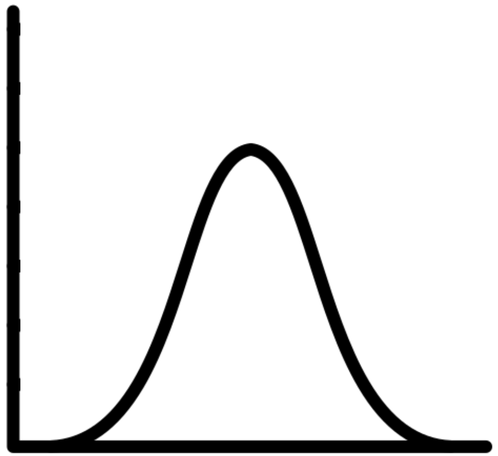


1/4 hour is needed for the next path? Let's stop!

*Extrapolation gives richer information for the **stopping criteria** for the fuzzing campaign*

How about the Greybox fuzzing?

Blackbox

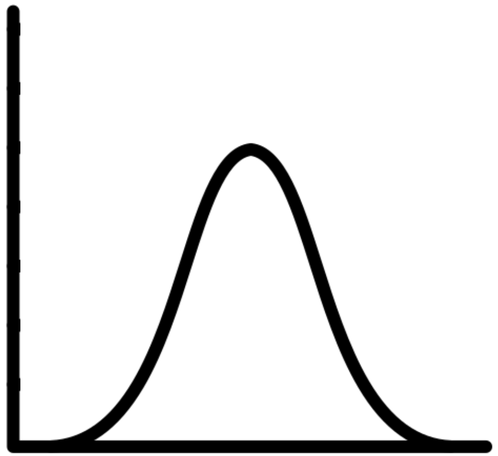


Sampling distribution is consistent

Böhme et al., “Estimating Residual Risk in Greybox Fuzzing.” ESEC/FSE’21.

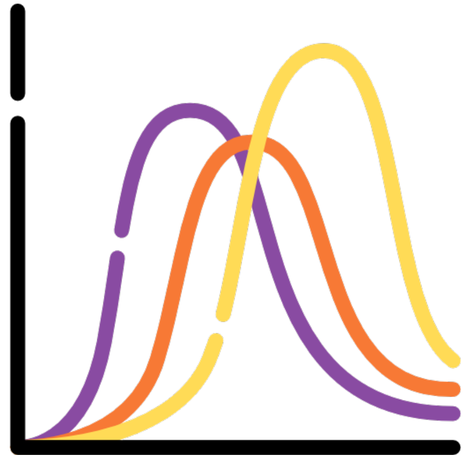
How about the Greybox fuzzing?

Blackbox



Sampling distribution is consistent

Greybox



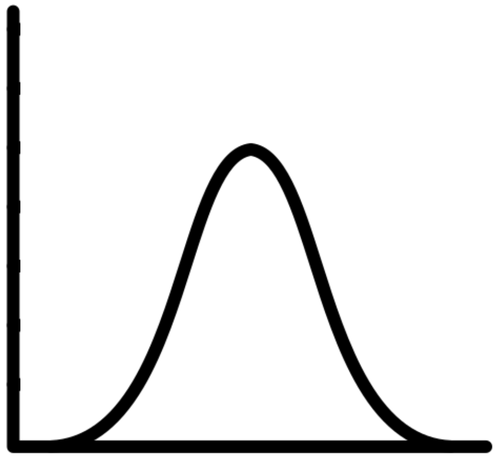
Sampling distribution keeps change

Adaptive bias

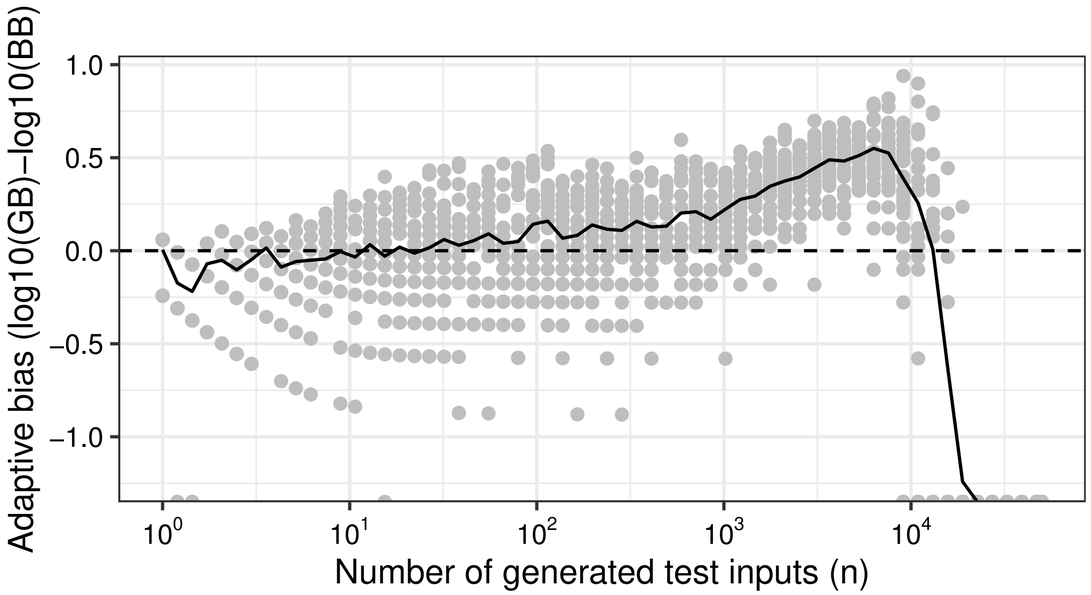
Böhme et al., "Estimating Residual Risk in Greybox Fuzzing." ESEC/FSE'21.

How about the Greybox fuzzing?

Blackbox

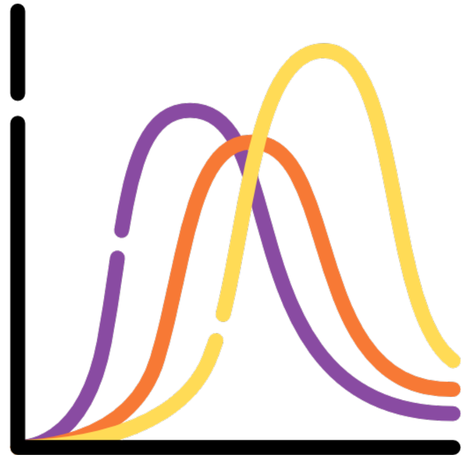


Sampling distribution is consistent



"Due to adaptive bias, the discovery probability is consistently higher in the greybox campaign than the blackbox campaign."

Greybox



Sampling distribution keeps change

Adaptive bias

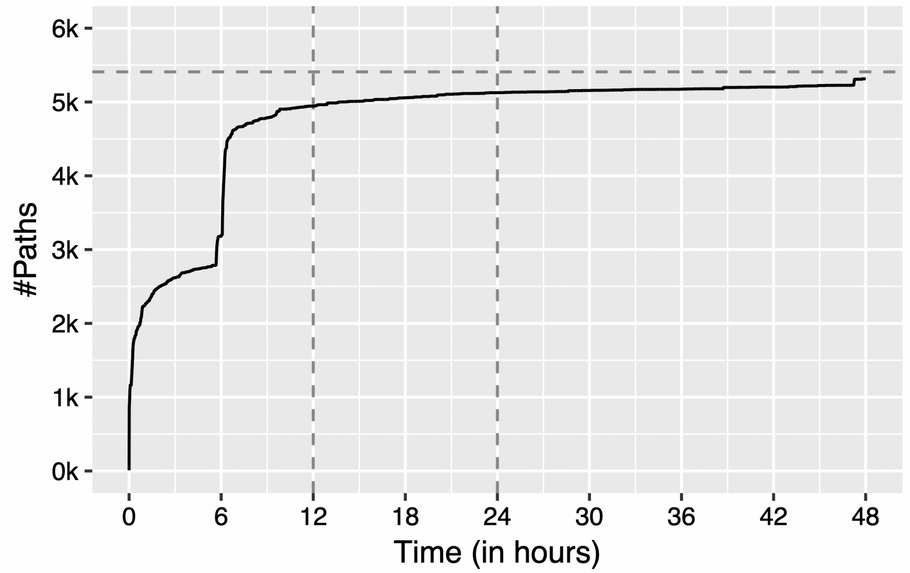
Böhme et al., "Estimating Residual Risk in Greybox Fuzzing." ESEC/FSE'21.

Extrapolating the Greybox Fuzzing Campaign

- Aim: Predict the future coverage rate of the greybox fuzzing campaign

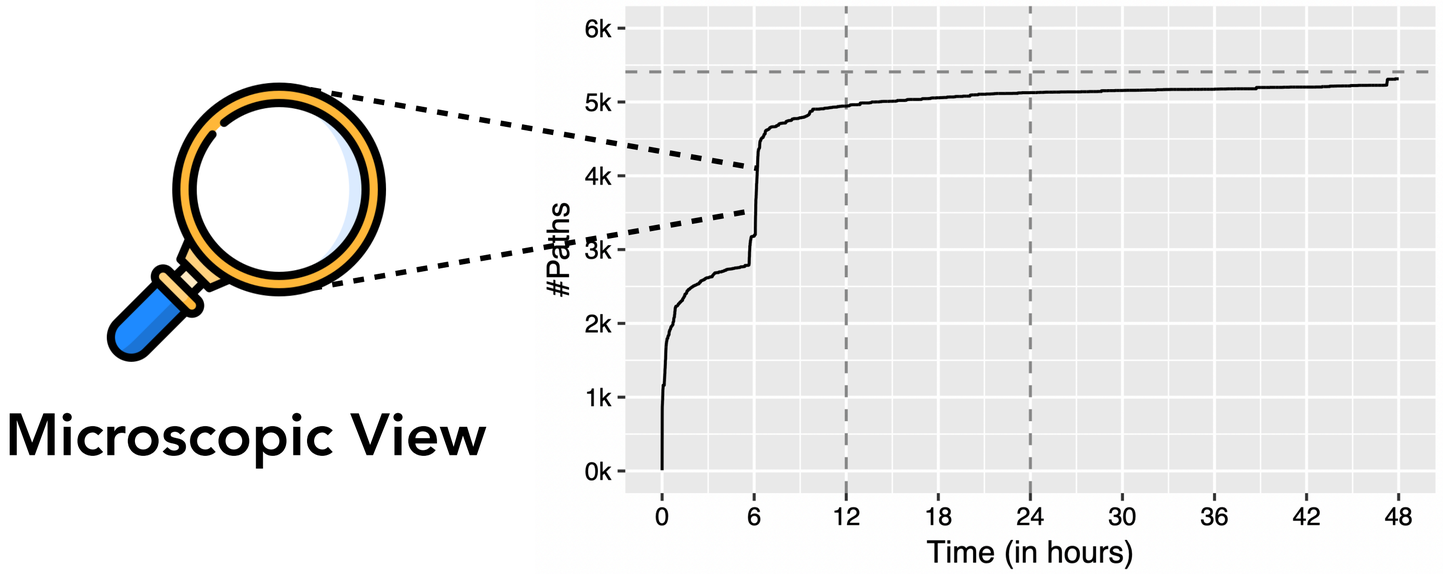
Extrapolating the Greybox Fuzzing Campaign

- Aim: Predict the future coverage rate of the greybox fuzzing campaign
- In other words, how can we solve the ***adaptive bias*** problem?



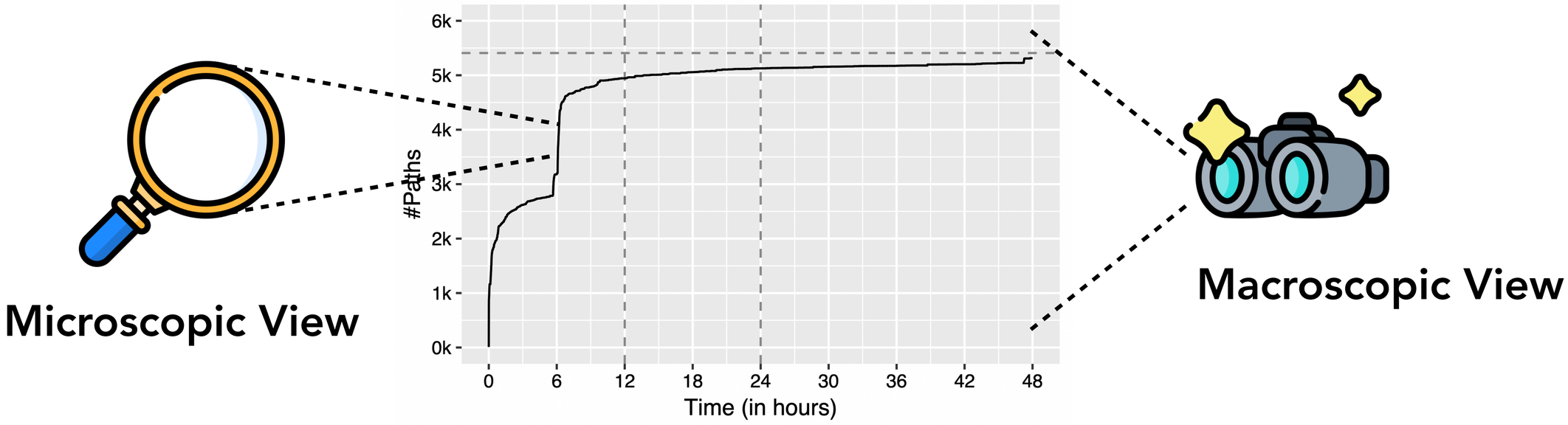
Extrapolating the Greybox Fuzzing Campaign

- Aim: Predict the future coverage rate of the greybox fuzzing campaign
- In other words, how can we solve the ***adaptive bias*** problem?



Extrapolating the Greybox Fuzzing Campaign

- Aim: Predict the future coverage rate of the greybox fuzzing campaign
- In other words, how can we solve the ***adaptive bias*** problem?

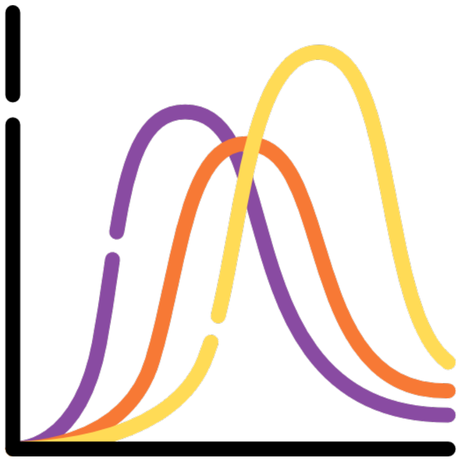


Extrapolating the Greybox Fuzzing Campaign

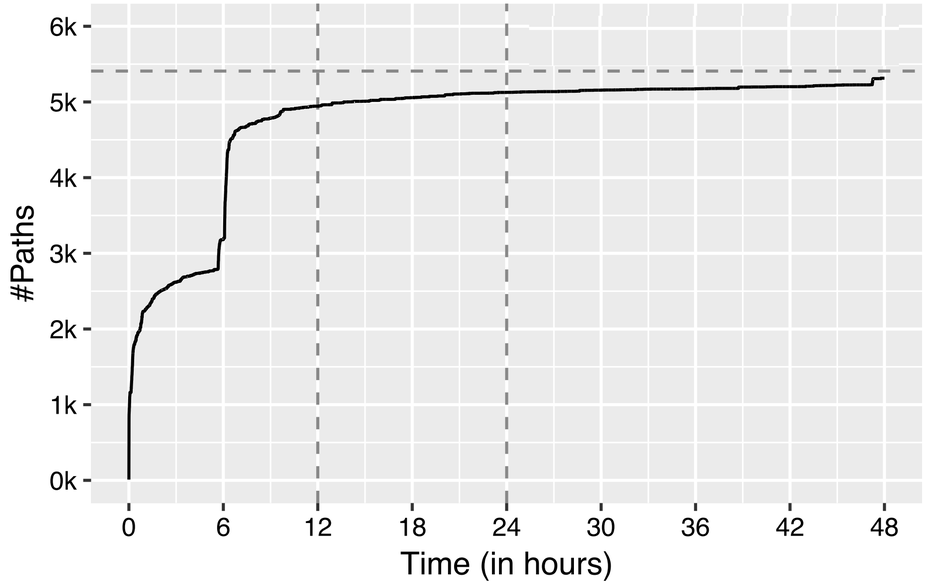
- First key insight — *Microscopic view*

Extrapolating the Greybox Fuzzing Campaign

- First key insight — *Microscopic view*

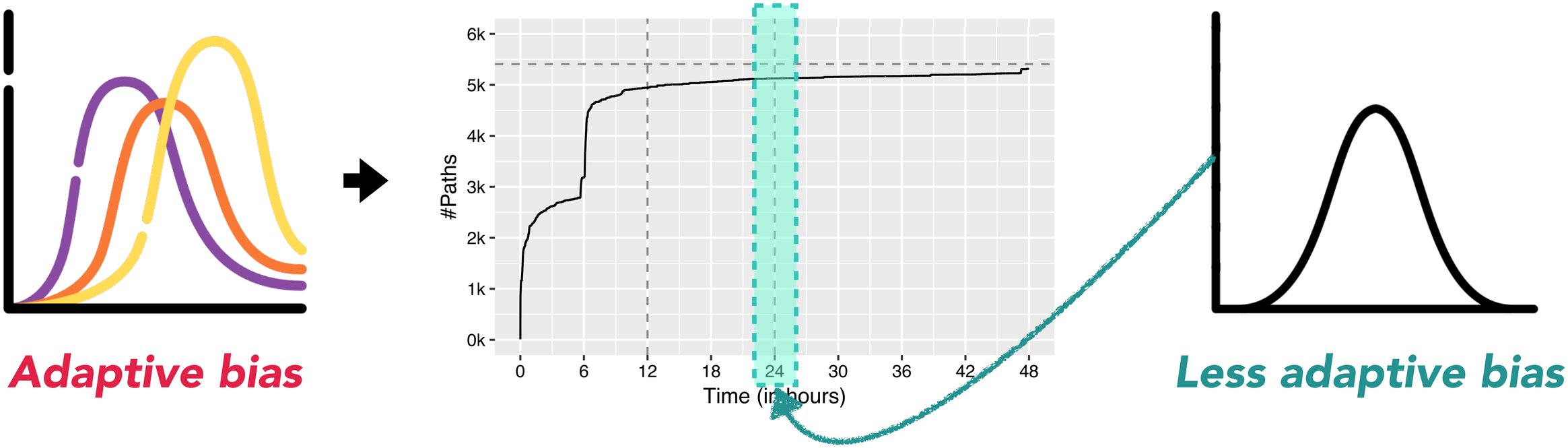


Adaptive bias



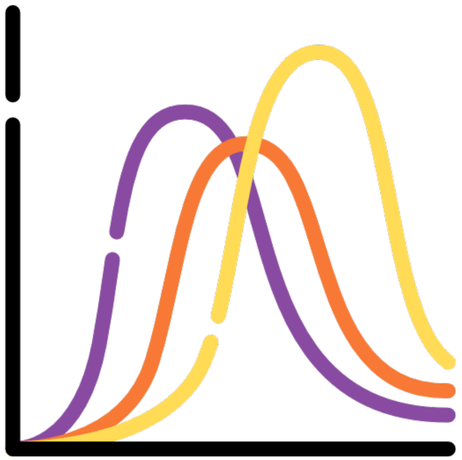
Extrapolating the Greybox Fuzzing Campaign

- First key insight — *Microscopic view*

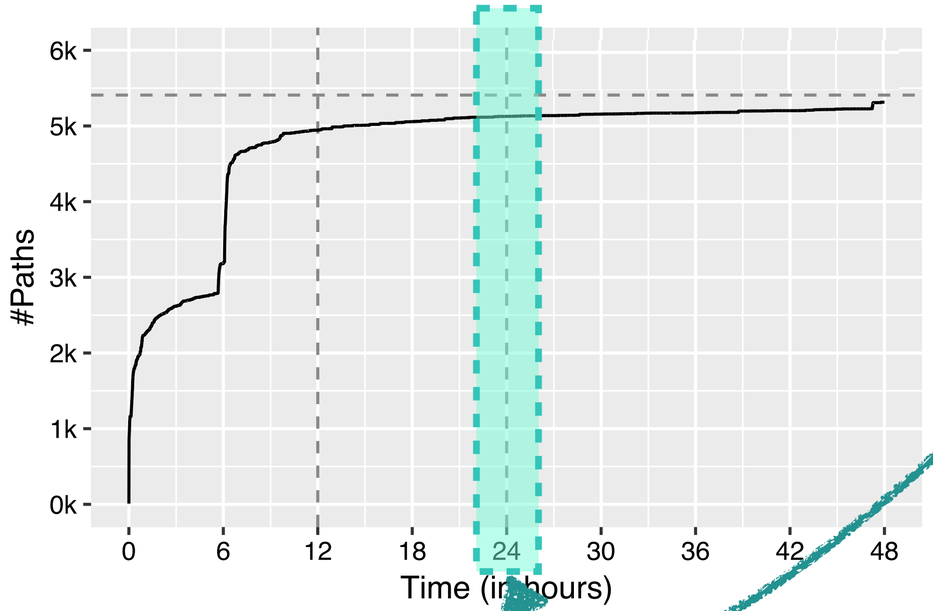


Extrapolating the Greybox Fuzzing Campaign

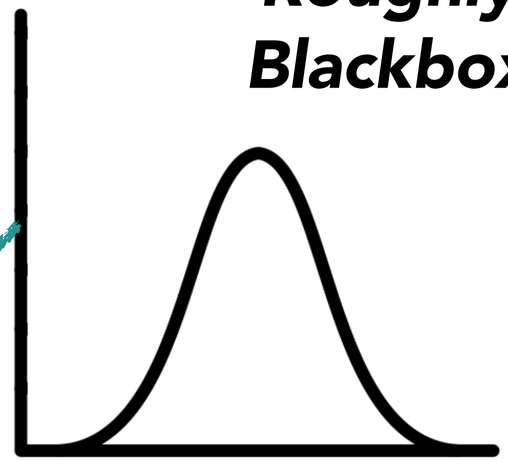
- First key insight — *Microscopic view*



Adaptive bias



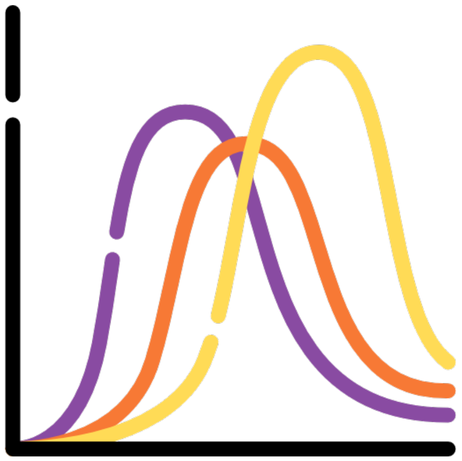
Roughly Blackbox



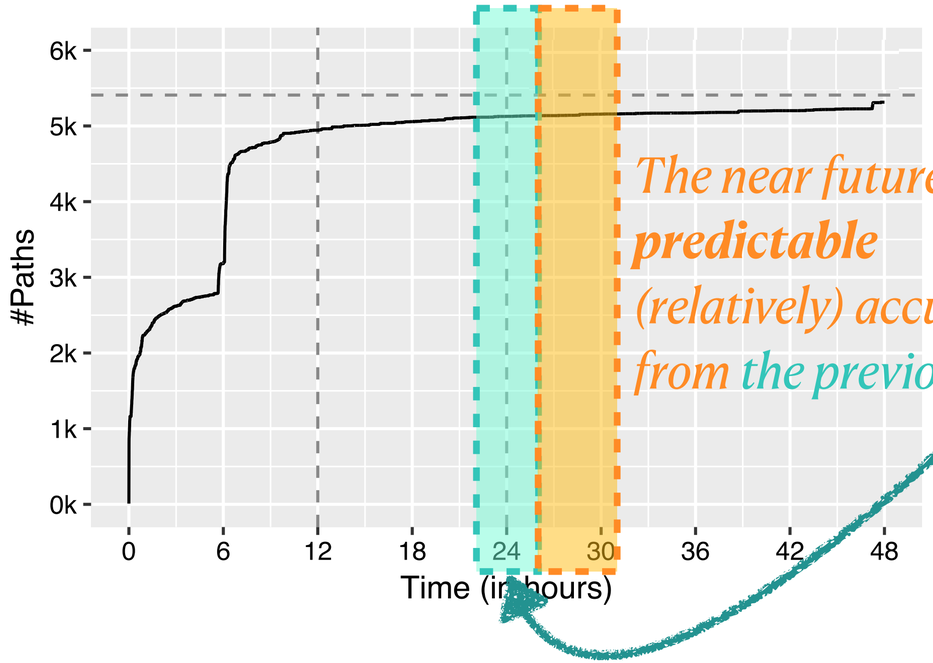
Less adaptive bias

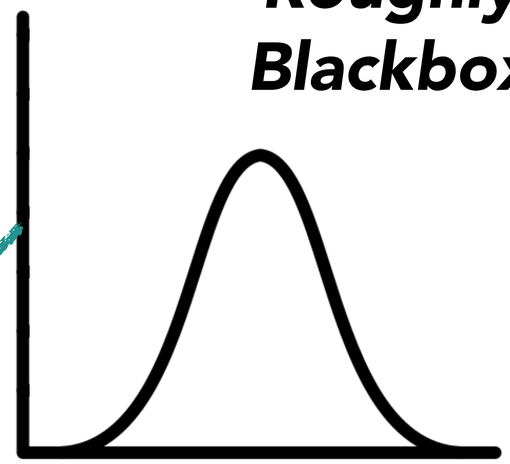
Extrapolating the Greybox Fuzzing Campaign

- First key insight — *Microscopic view*



Adaptive bias

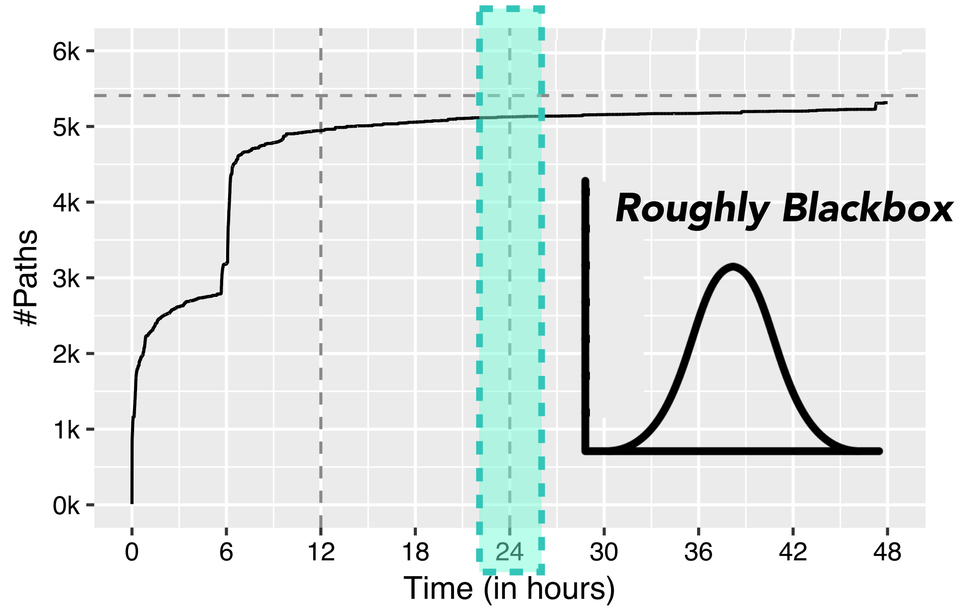




Less adaptive bias

Extrapolating the Greybox Fuzzing Campaign

- First key insight — *Microscopic view*
- Potential drawback: ***Small region has small data to use***

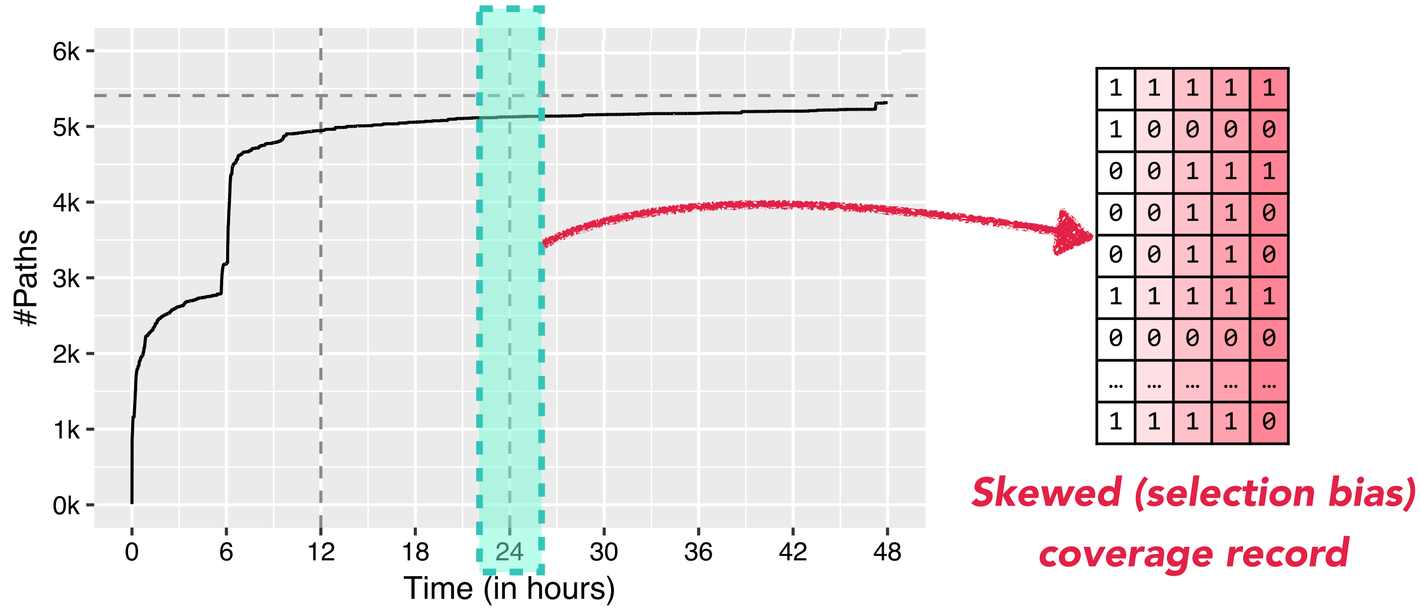


$$\hat{U}(t) = \frac{\Phi_1}{t}\left[\frac{(t-1)\Phi_1}{(t-1)\Phi_1 + 2\Phi_2}\right]$$

Coverage rate estimator for a blackbox fuzzing campaign

Extrapolating the Greybox Fuzzing Campaign

- First key insight — *Microscopic view*
- Potential drawback: ***Small region has small data to use***

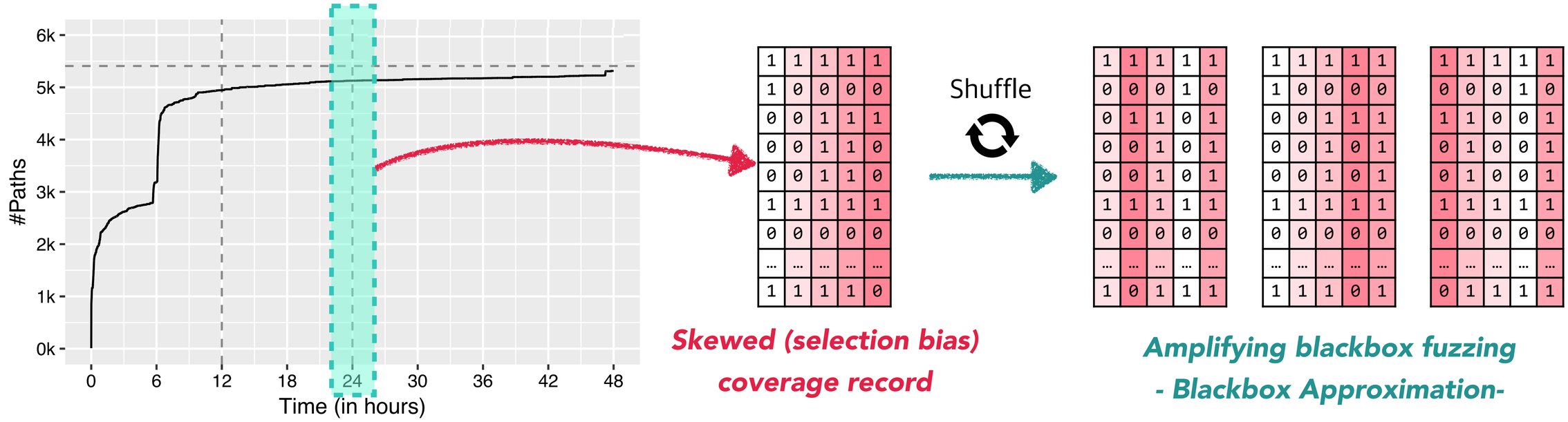


1	1	1	1	1
1	0	0	0	0
0	0	1	1	1
0	0	1	1	0
0	0	1	1	0
1	1	1	1	1
0	0	0	0	0
…	…	…	…	…
1	1	1	1	0

Skewed (selection bias) coverage record

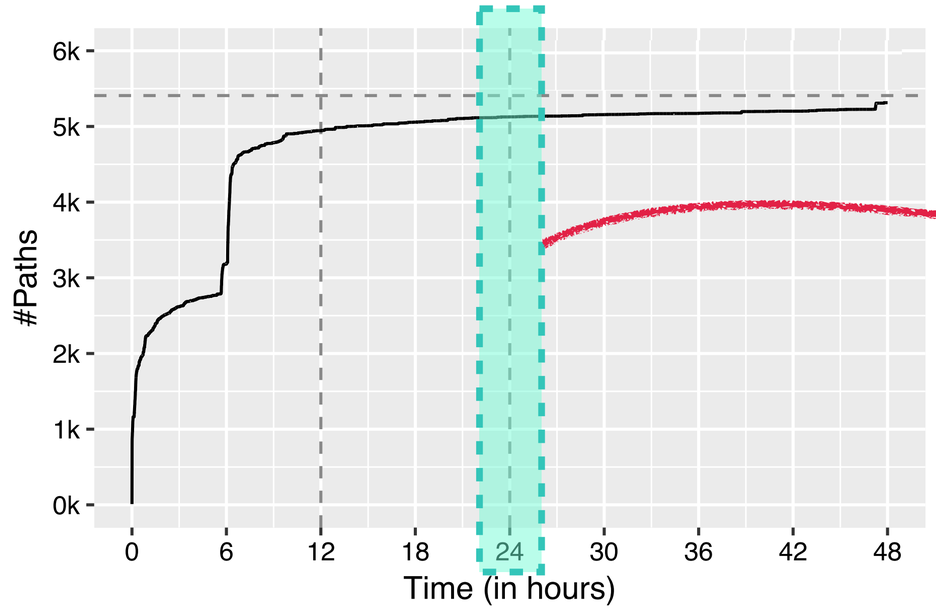
Extrapolating the Greybox Fuzzing Campaign

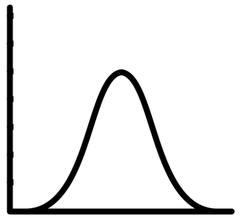
- First key insight — *Microscopic view*
- Solution: ***Shuffle to amplify***



Extrapolating the Greybox Fuzzing Campaign

- First key insight — *Microscopic view*
- Solution: ***Shuffle to amplify***





1	1	1	1	1
1	0	0	0	0
0	0	1	1	1
0	0	1	1	0
0	0	1	1	0
1	1	1	1	1
0	0	0	0	0
…	…	…	…	…
1	1	1	1	0

Shuffle

1	1	1	1	1
0	0	0	1	0
0	1	1	0	1
0	0	1	0	1
0	0	1	0	1
1	1	1	1	1
0	0	0	0	0
…	…	…	…	…
1	0	1	1	1

1	1	1	1	1
1	0	0	0	0
0	0	1	1	1
0	0	1	0	1
0	0	1	0	1
1	1	1	1	1
0	0	0	0	0
…	…	…	…	…
1	1	1	0	1

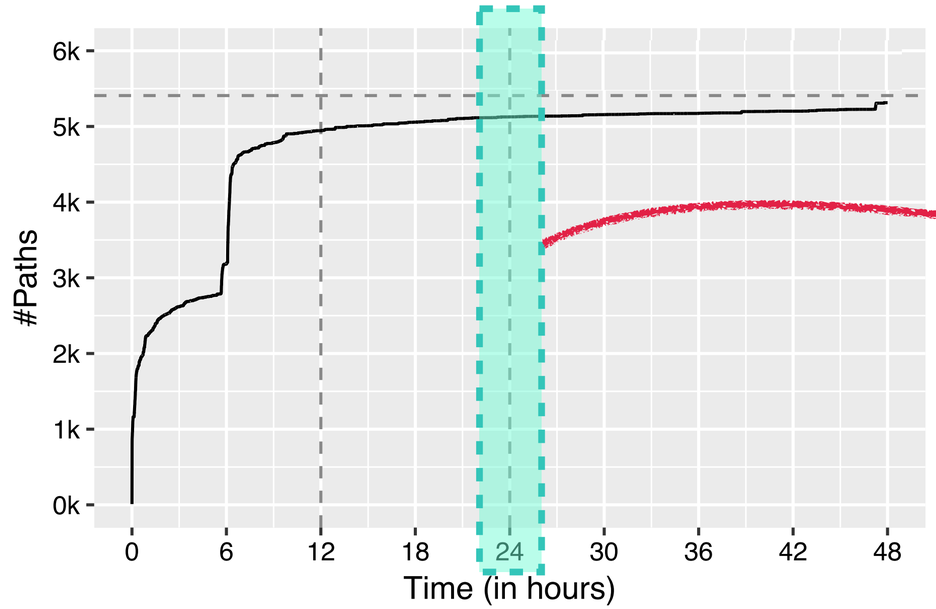
1	1	1	1	1
0	0	0	1	0
1	1	0	0	1
0	1	0	0	1
0	1	0	0	1
1	1	1	1	1
0	0	0	0	0
…	…	…	…	…
0	1	1	1	1

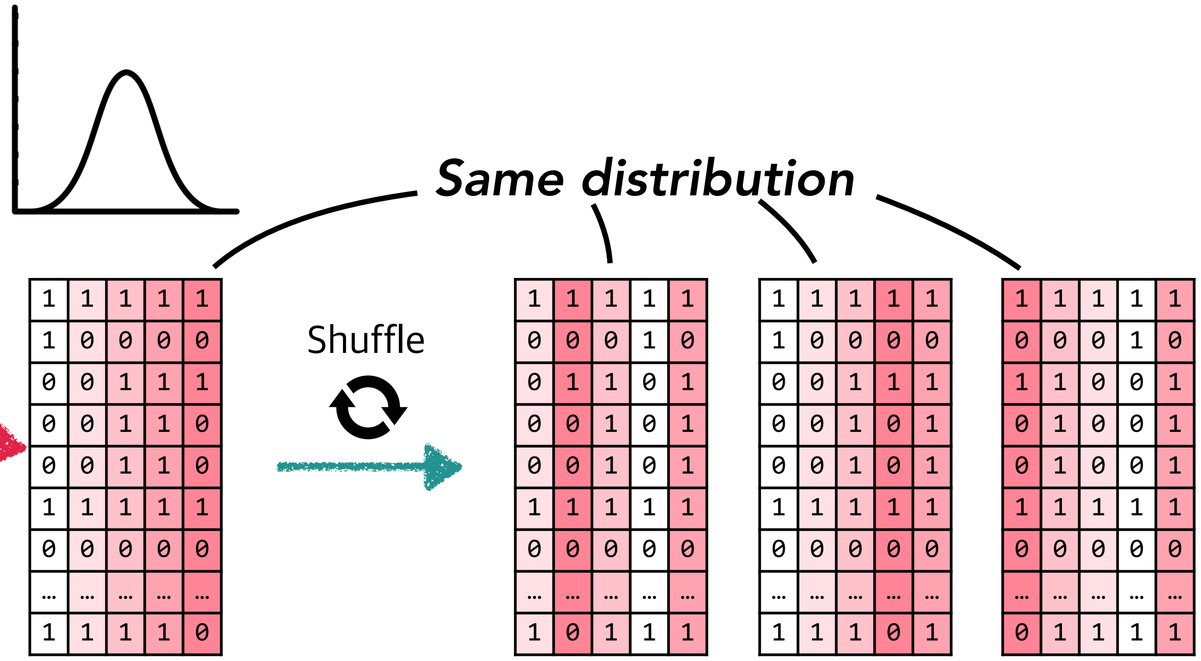
Skewed (selection bias) coverage record

Amplifying blackbox fuzzing - Blackbox Approximation-

Extrapolating the Greybox Fuzzing Campaign

- First key insight — *Microscopic view*
- Solution: ***Shuffle to amplify***





Skewed (selection bias) coverage record

Amplifying blackbox fuzzing - Blackbox Approximation-

Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

*"Greybox fuzzing's adaptive bias could be **predictable**."*

Extrapolating the Greybox Fuzzing Campaign

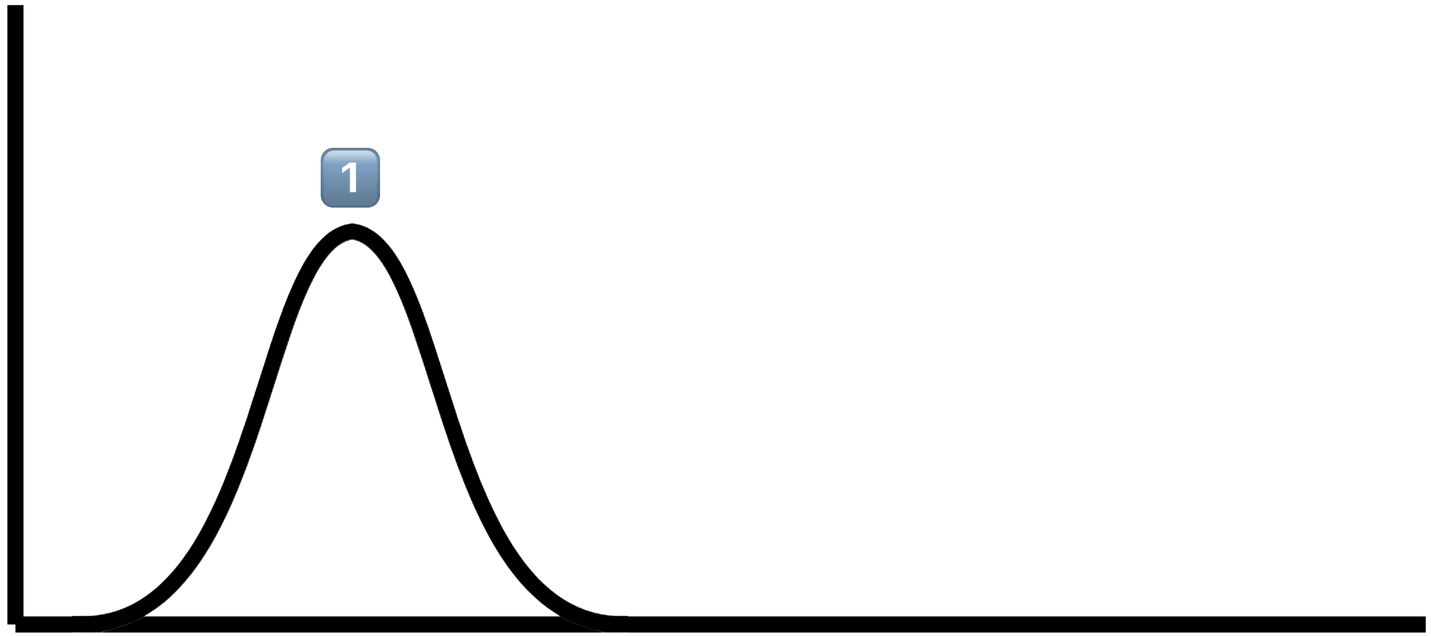
- Second key insight — *Macroscopic view*

*"Greybox fuzzing's adaptive bias could be **predictable**."*

Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

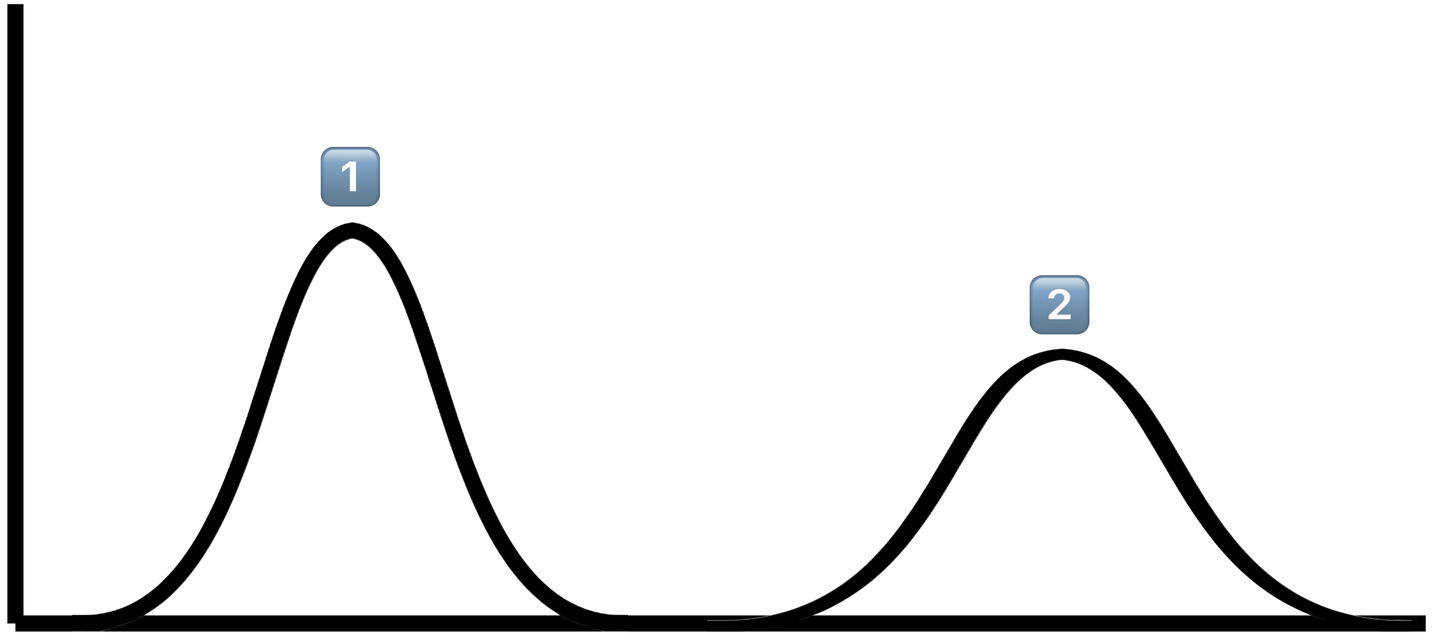
*"Greybox fuzzing's adaptive bias could be **predictable**."*



Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

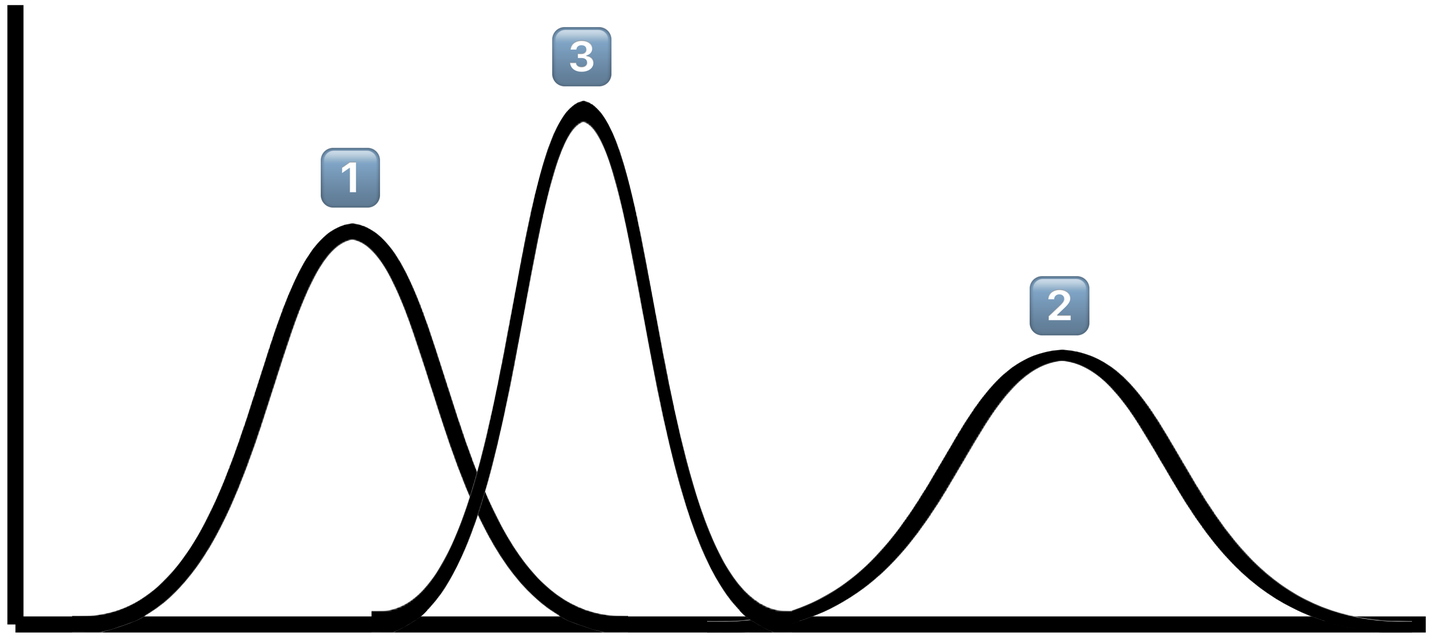
*"Greybox fuzzing's adaptive bias could be **predictable**."*



Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

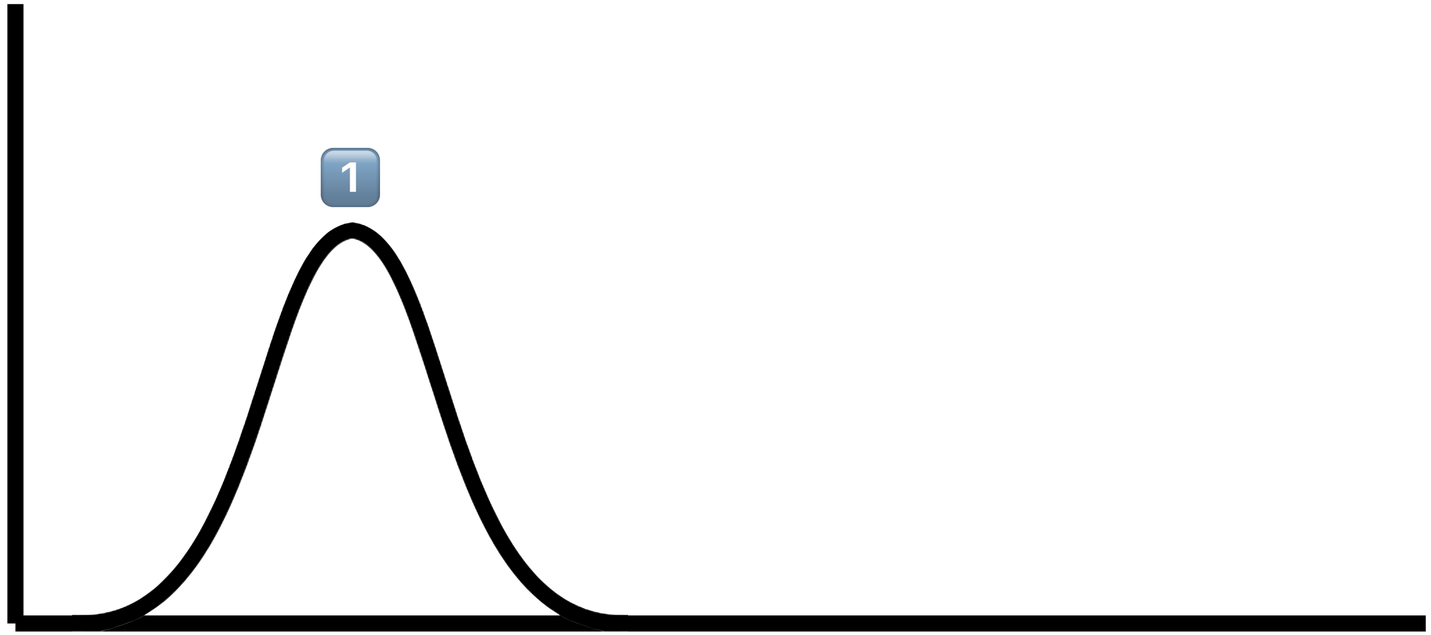
*"Greybox fuzzing's adaptive bias could be **predictable**."*



Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

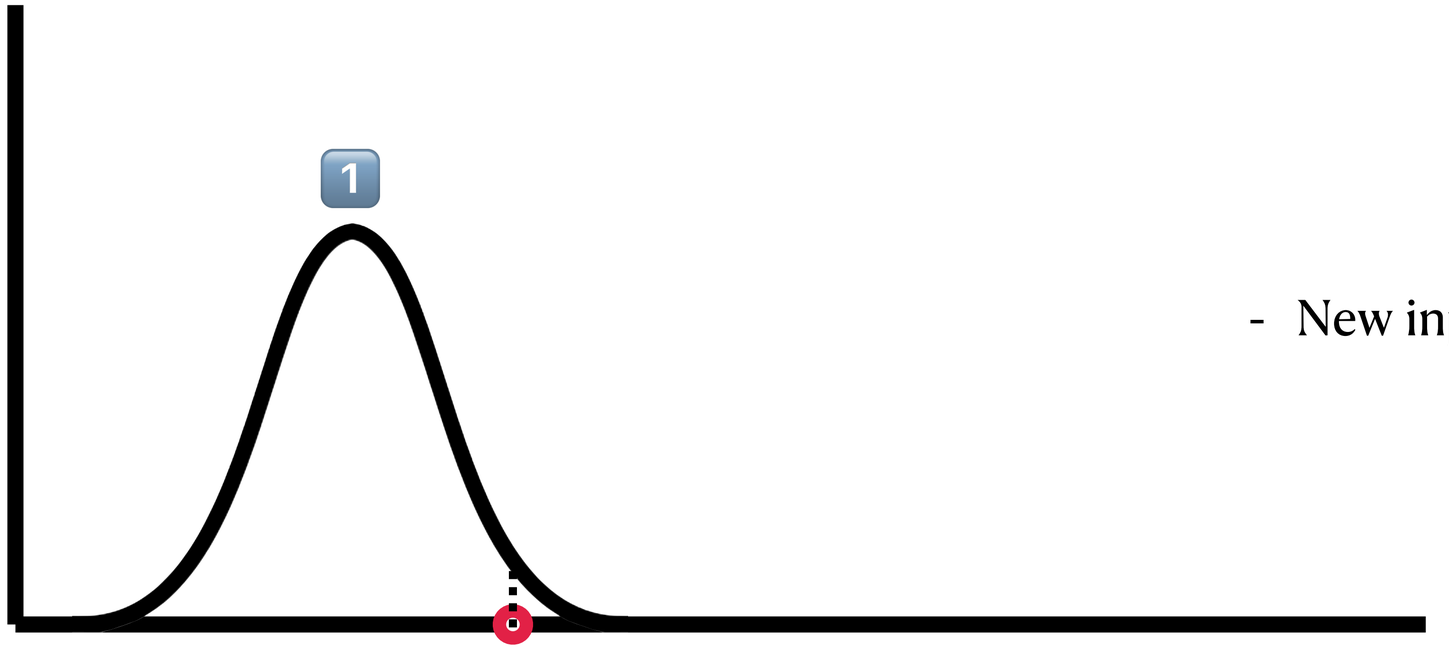
*"Greybox fuzzing's adaptive bias could be **predictable**."*



Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

*"Greybox fuzzing's adaptive bias could be **predictable**."*

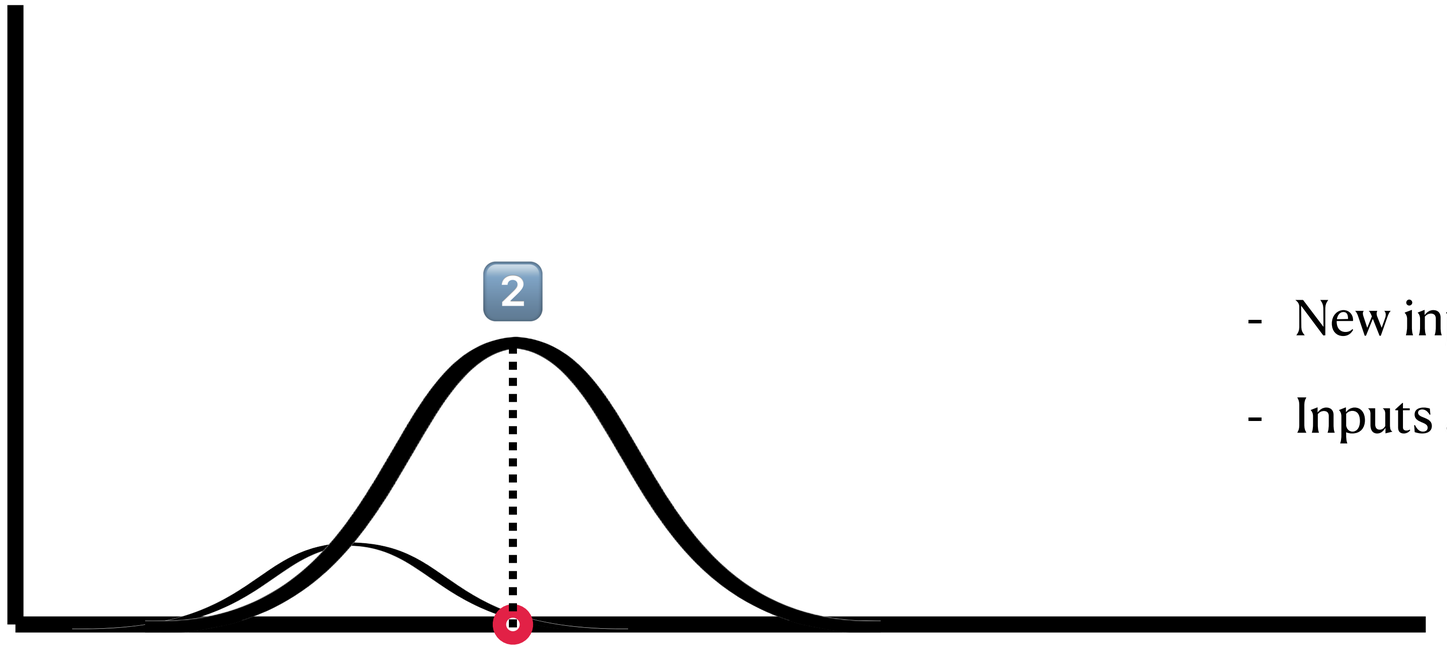


- New input that *increases coverage* is found

Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

*"Greybox fuzzing's adaptive bias could be **predictable**."*

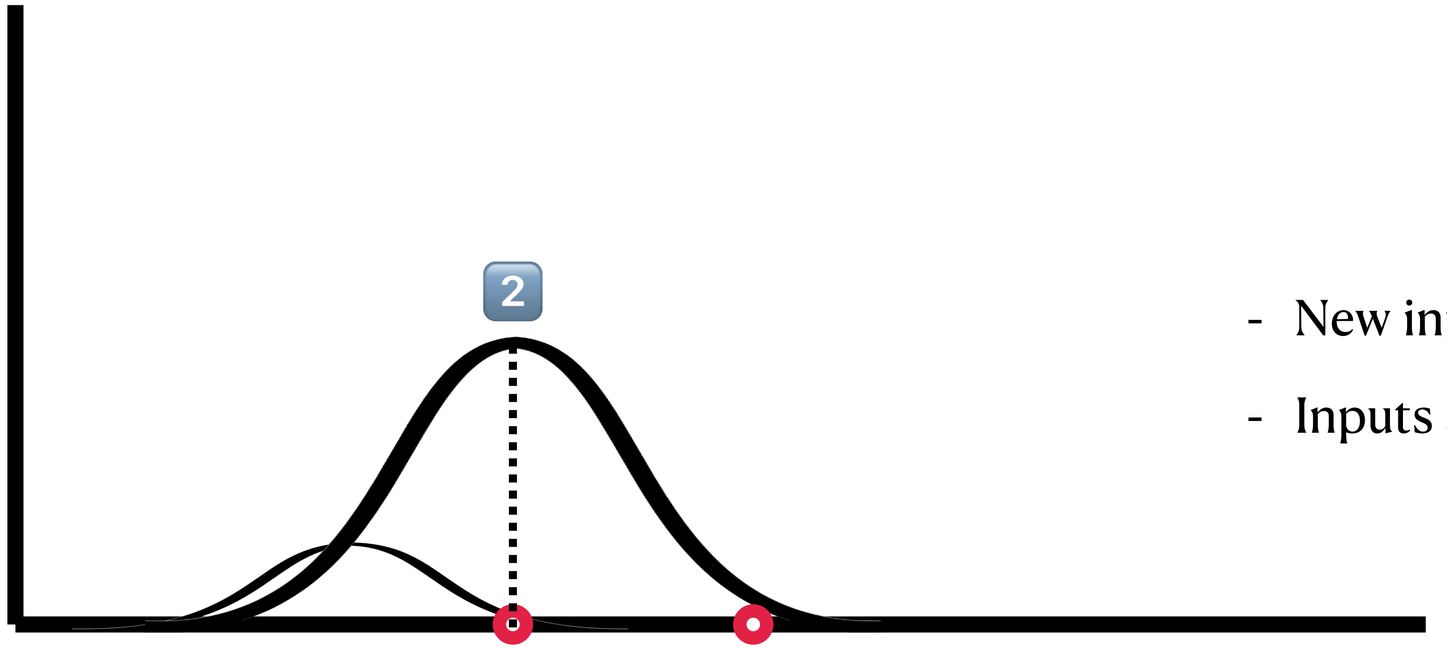


- New input that *increases coverage* is found
- Inputs around the new input are sampled

Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

*"Greybox fuzzing's adaptive bias could be **predictable**."*

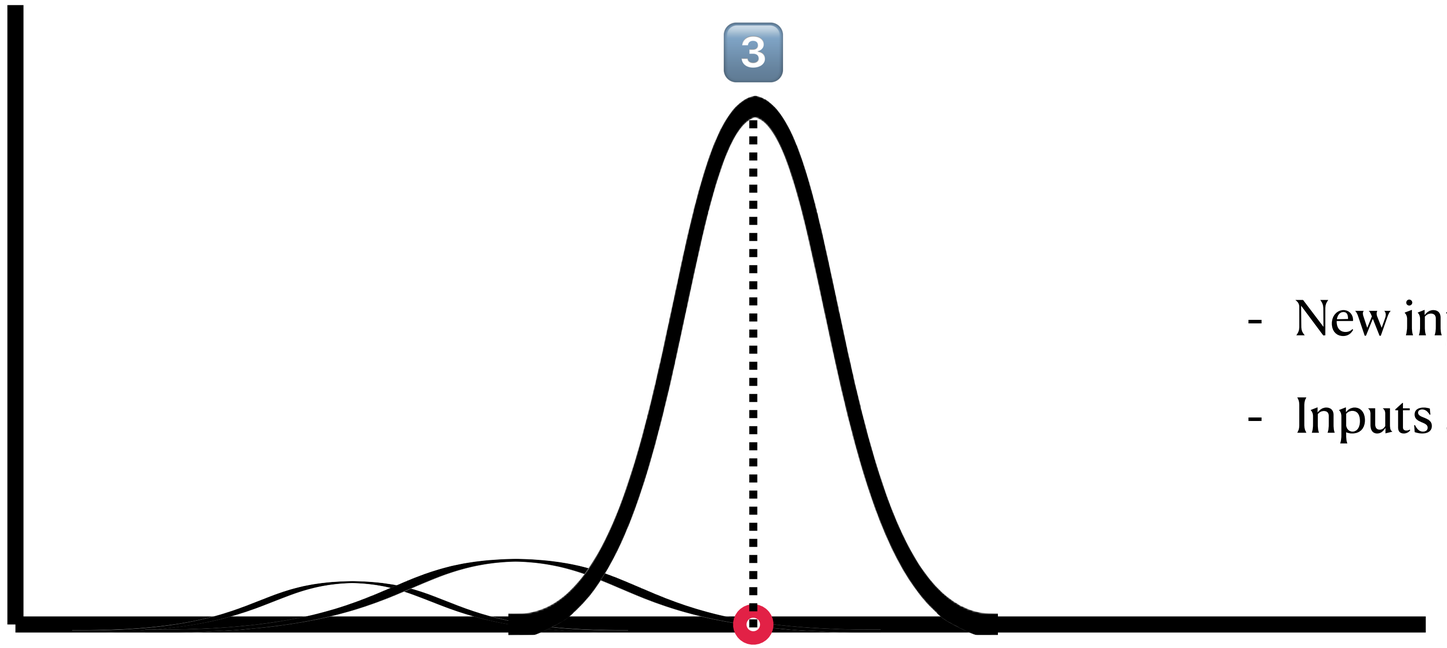


- New input that *increases coverage* is found
- Inputs around the new input are sampled

Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

*"Greybox fuzzing's adaptive bias could be **predictable**."*

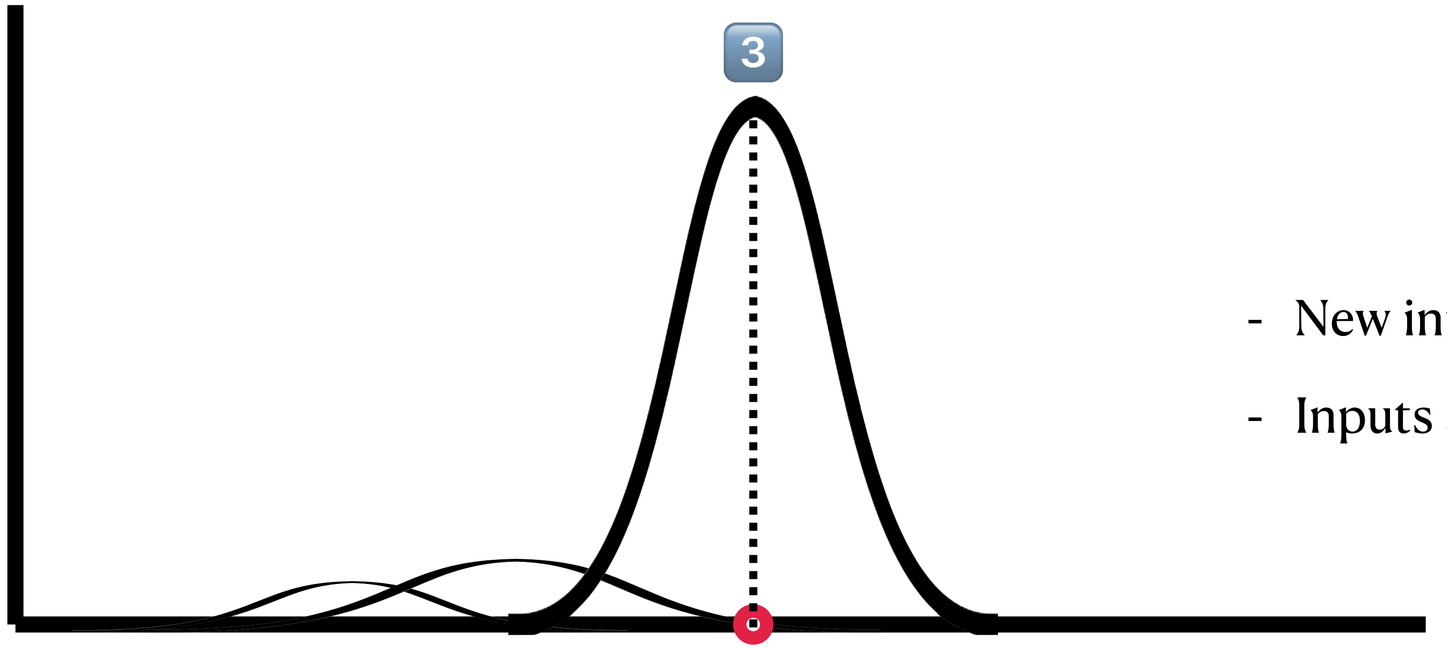


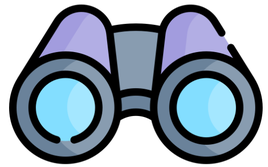
- New input that *increases coverage* is found
- Inputs around the new input are sampled

Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

*"Greybox fuzzing's adaptive bias could be **predictable**."*



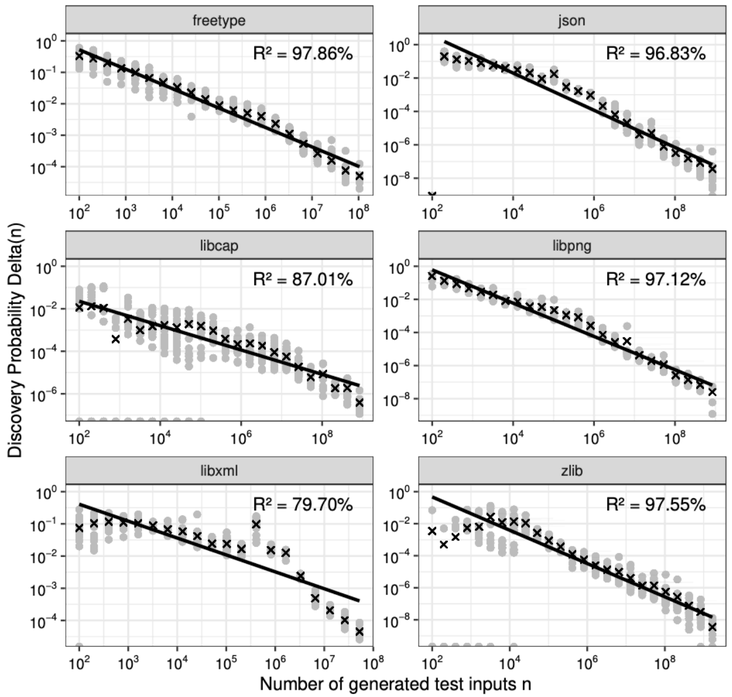


- New input that *increases coverage* is found
- Inputs around the new input are sampled

Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

*"Greybox fuzzing's adaptive bias could be **predictable**."*



For example,

Böhme et al. "Estimating Residual Risk in Greybox Fuzzing." ESEC/FSE'21

log(# of data points) has linear relation with
log(probability of discovering new program element)

(probability) *discovery probability* ≠ *coverage rate* (# of new element)

Methodology

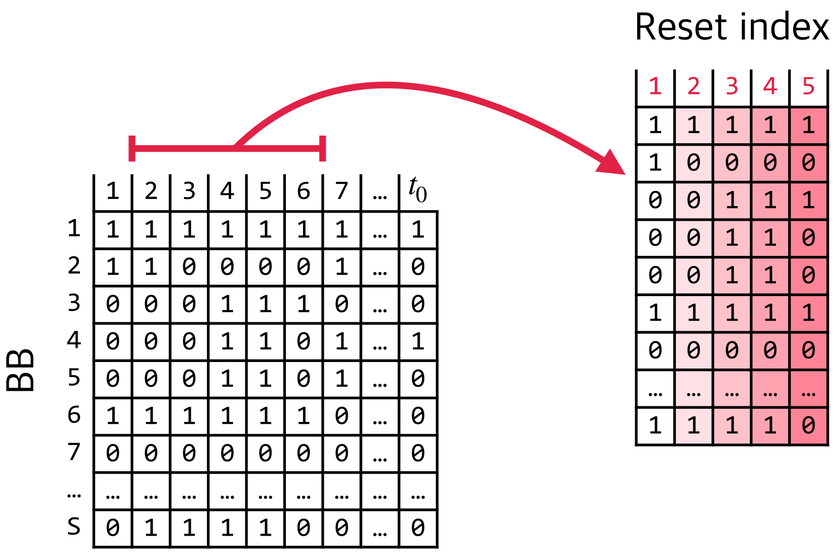
Methodology

BB

	1	2	3	4	5	6	7	…	t_0
1	1	1	1	1	1	1	1	…	1
2	1	1	0	0	0	0	1	…	0
3	0	0	0	1	1	1	0	…	0
4	0	0	0	1	1	0	1	…	1
5	0	0	0	1	1	0	1	…	0
6	1	1	1	1	1	1	0	…	0
7	0	0	0	0	0	0	0	…	0
…	…	…	…	…	…	…	…	…	…
S	0	1	1	1	1	0	0	…	0

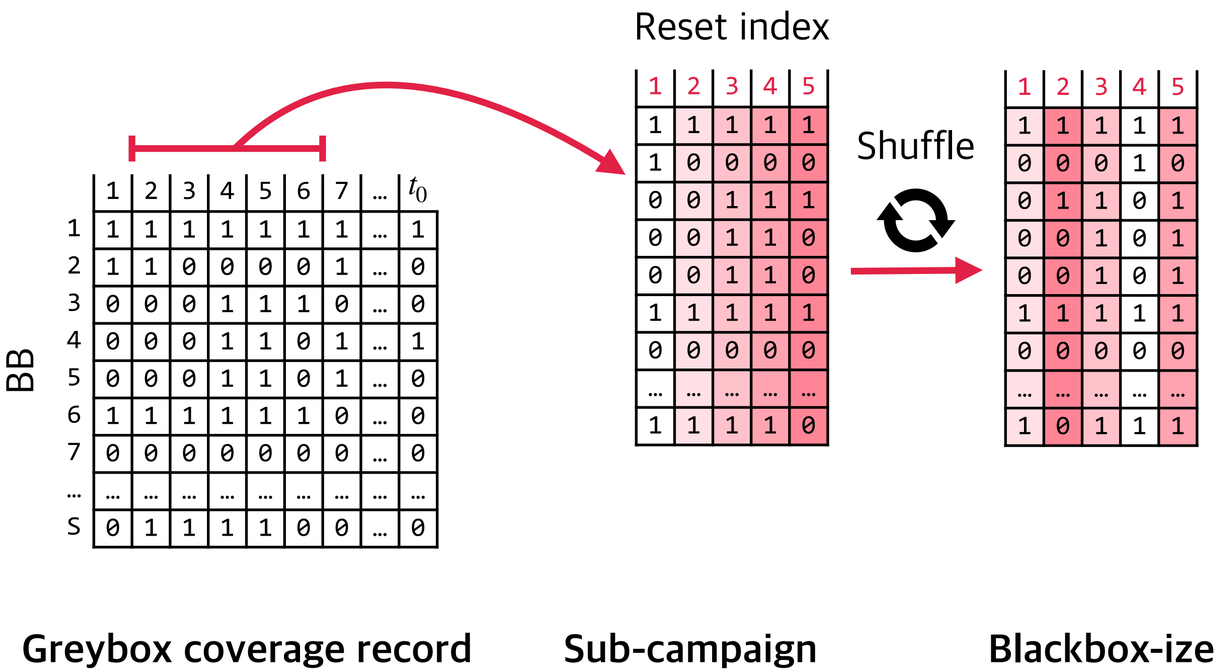
Greybox coverage record

Methodology



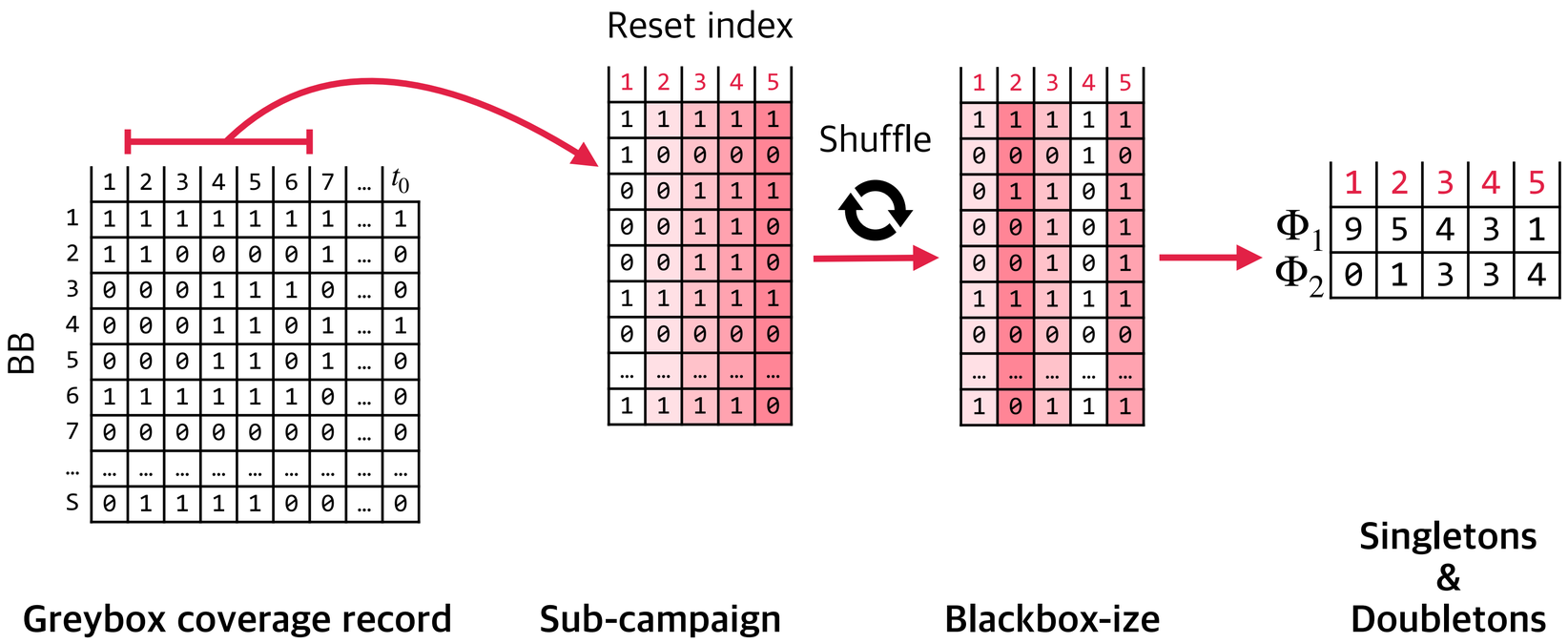
Greybox coverage record **Sub-campaign**

Methodology



Greybox coverage record **Sub-campaign** **Blackbox-ize**

Methodology



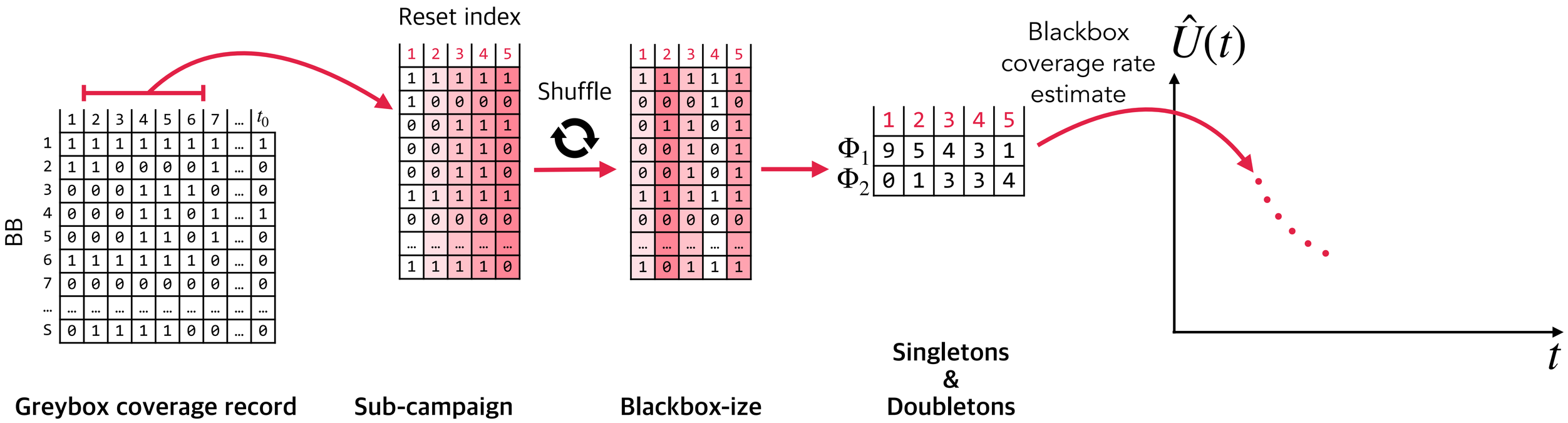
	1	2	3	4	5	6	7	…	t_0
1	1	1	1	1	1	1	1	…	1
2	1	1	0	0	0	0	1	…	0
3	0	0	0	1	1	1	0	…	0
4	0	0	0	1	1	0	1	…	1
5	0	0	0	1	1	0	1	…	0
6	1	1	1	1	1	1	0	…	0
7	0	0	0	0	0	0	0	…	0
…	…	…	…	…	…	…	…	…	…
S	0	1	1	1	1	0	0	…	0

1	2	3	4	5
1	1	1	1	1
1	0	0	0	0
0	0	1	1	1
0	0	1	1	0
0	0	1	1	0
1	1	1	1	1
0	0	0	0	0
…	…	…	…	…
1	1	1	1	0

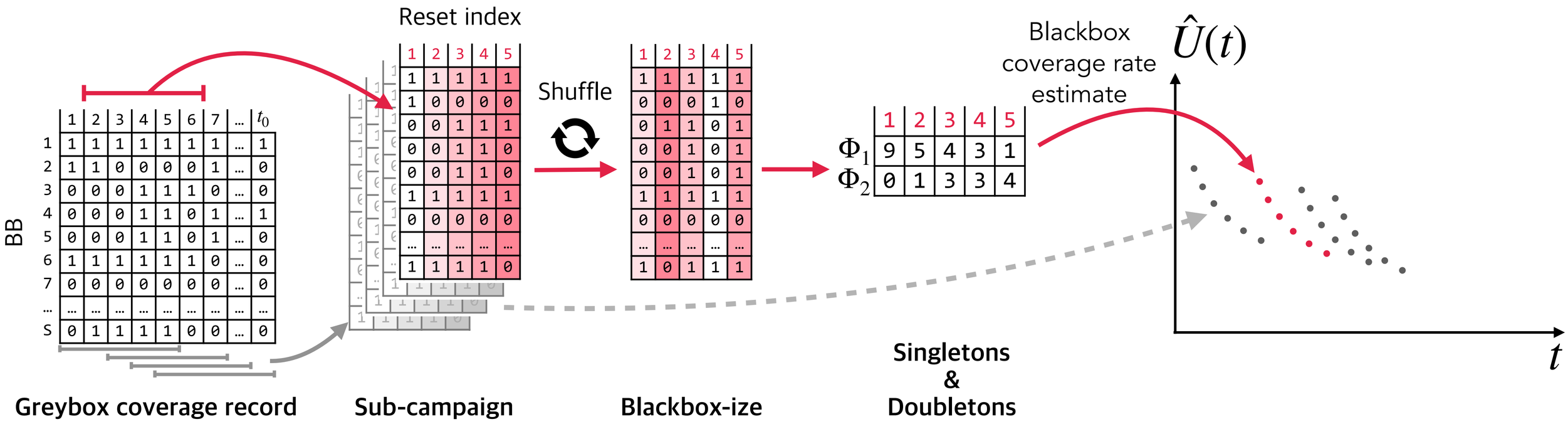
1	2	3	4	5
1	1	1	1	1
0	0	0	1	0
0	1	1	0	1
0	0	1	0	1
0	0	1	0	1
1	1	1	1	1
0	0	0	0	0
…	…	…	…	…
1	0	1	1	1

	1	2	3	4	5
Φ_1	9	5	4	3	1
Φ_2	0	1	3	3	4

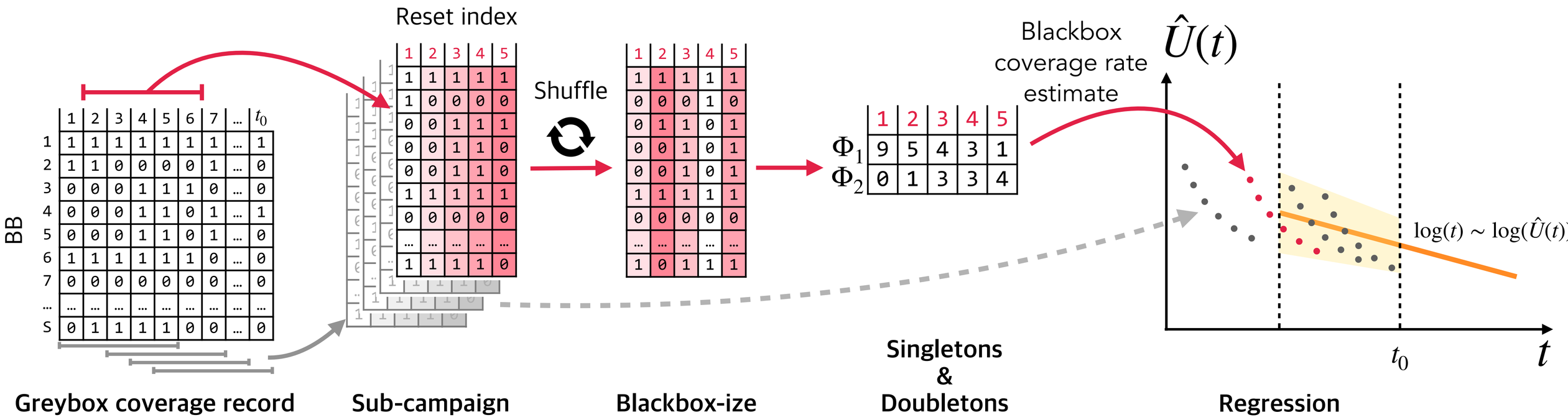
Methodology



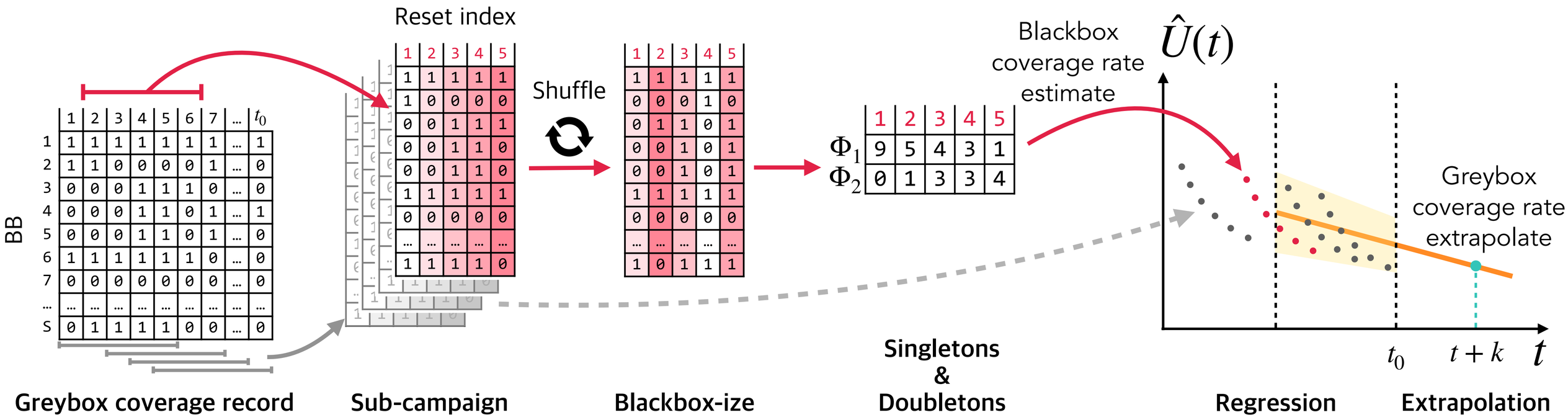
Methodology



Methodology



Methodology



RQ 1. Coverage Rate Prediction

Q. How accurate is our ***regression model considering the adaptive bias*** compared to the ***existing blackbox extrapolation model*** ?

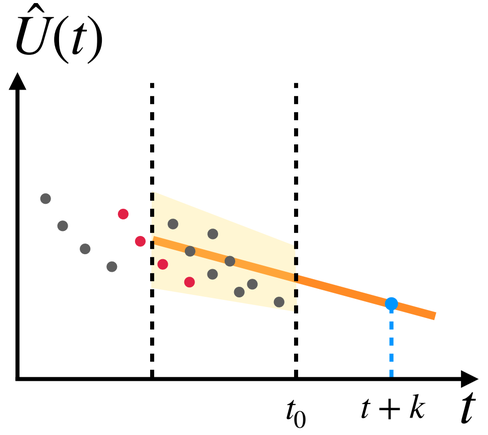
Existing $\hat{U}(t+k)$

Extrapolator

Ignores the adaptive bias

$$\hat{\Phi}_0 \left[1 - \left(1 - \frac{\Phi_1}{t\hat{\Phi}_0 + \Phi_1} \right)^k \right]$$

VS



Our $\hat{U}(t+k)$

Extrapolator

Consider the adaptive bias

RQ 1. Coverage Rate Prediction

Q. How accurate is our ***regression model considering the adaptive bias*** compared to the ***existing blackbox extrapolation model*** ?

- Subject program: five open-source C libraries
- Evaluation Scenario:

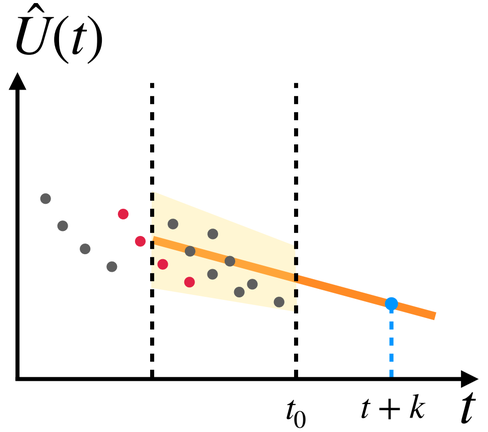
Existing $\hat{U}(t+k)$

Extrapolator

$$\hat{\Phi}_0 \left[1 - \left(1 - \frac{\Phi_1}{t\hat{\Phi}_0 + \Phi_1} \right)^k \right]$$

Ignores the adaptive bias

VS



Our $\hat{U}(t+k)$

Extrapolator

Consider the adaptive bias

RQ 1. Coverage Rate Prediction

Q. How accurate is our ***regression model considering the adaptive bias*** compared to the ***existing blackbox extrapolation model*** ?

- Subject program: five open-source C libraries
- Evaluation Scenario: 1) run the greybox fuzzer until having t data points
  2) apply each extrapolator to extrapolate $\hat{U}(t+k)$
  3) run the greybox fuzzer for k more data points to get $U(t+k)$.

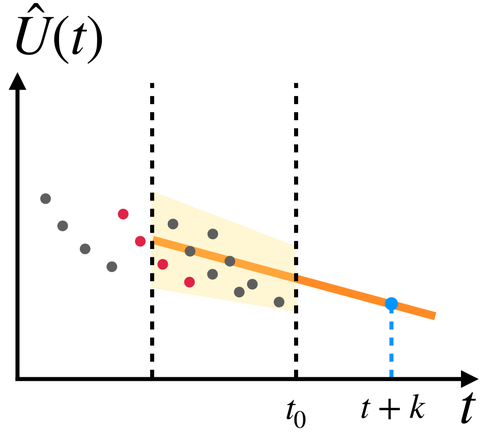
Existing $\hat{U}(t+k)$

Extrapolator

Ignores the adaptive bias

$$\hat{\Phi}_0 \left[1 - \left(1 - \frac{\Phi_1}{t\hat{\Phi}_0 + \Phi_1} \right)^k \right]$$

VS



Our $\hat{U}(t+k)$

Extrapolator

Consider the adaptive bias

RQ 1. Coverage Rate Prediction

Q. How accurate is our ***regression model considering the adaptive bias*** compared to the ***existing blackbox extrapolation model*** ?

- Subject program: five open-source C libraries
- Evaluation Scenario: 1) run the greybox fuzzer until having t data points
  2) apply each extrapolator to extrapolate $\hat{U}(t+k)$
  3) run the greybox fuzzer for k more data points to get $U(t+k)$.

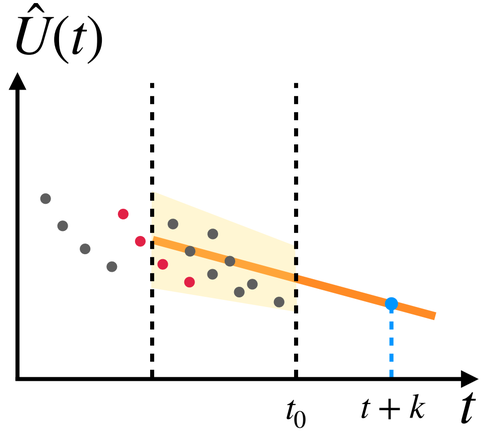
Compare

Existing $\hat{U}(t+k)$ ***Extrapolator***

$$\hat{\Phi}_0 \left[1 - \left(1 - \frac{\Phi_1}{t\hat{\Phi}_0 + \Phi_1} \right)^k \right]$$

Ignores the adaptive bias

VS

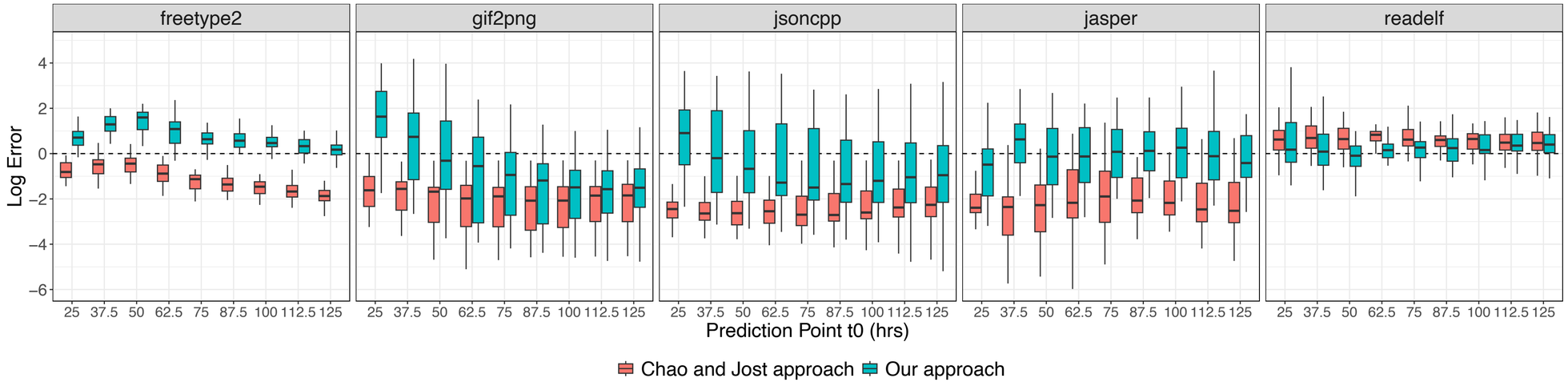


Our $\hat{U}(t+k)$ ***Extrapolator***

Consider the adaptive bias

Result 1: Coverage Rate Prediction

Difference between $\log(U(t+k))$ ***vs.*** $\log(\hat{U}(t+k))$



(close to 1 is better)

The average ratio $U(t+k)/\hat{U}(t+k)$:

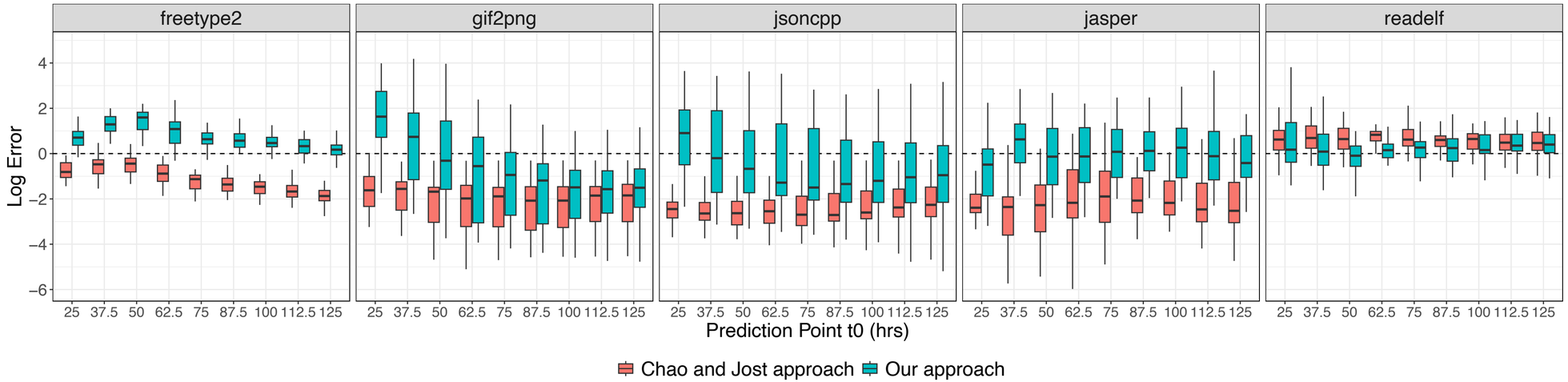
[Existing] **1.6 - 800** [Ours] **1.17 - 7**

across all subjects.

"Our extrapolator exhibits
at least one order of magnitude lower absolute bias
than the existing extrapolator for *4 out of 5 subjects*,
especially for long-term prediction."

Result 1: Coverage Rate Prediction

Difference between $\log(U(t+k))$ ***vs.*** $\log(\hat{U}(t+k))$



(close to 1 is better)

The average ratio $U(t+k)/\hat{U}(t+k)$:

[Existing] **1.6 - 800** [Ours] **1.17 - 7**

across all subjects.

"Our extrapolator exhibits
at least one order of magnitude lower absolute bias
than the existing extrapolator for *4 out of 5 subjects*,
especially for long-term prediction."

⇒ ***Well-handled the adaptive bias***

RQ 2. Stopping-Time Estimation

Q. Is our ***extrapolator*** useful for the stopping criteria?

Greybox Coverage Record

	1	2	3	4	5	6	7	…	t_0
1	1	1	1	1	1	1	1	…	1
2	1	1	0	0	0	0	1	…	0
3	0	0	0	1	1	1	0	…	0
4	0	0	0	1	1	0	1	…	1
5	0	0	0	1	1	0	1	…	0
6	1	1	1	1	1	1	0	…	0
7	0	0	0	0	0	0	0	…	0
…	…	…	…	…	…	…	…	…	…
S	0	1	1	1	1	0	0	…	0

&

U: Target coverage rate

RQ 2. Stopping-Time Estimation

Q. Is our ***extrapolator*** useful for the stopping criteria?

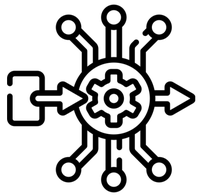
Greybox Coverage Record

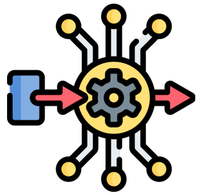
	1	2	3	4	5	6	7	…	t_0
1	1	1	1	1	1	1	1	…	1
2	1	1	0	0	0	0	1	…	0
3	0	0	0	1	1	1	0	…	0
4	0	0	0	1	1	0	1	…	1
5	0	0	0	1	1	0	1	…	0
6	1	1	1	1	1	1	0	…	0
7	0	0	0	0	0	0	0	…	0
…	…	…	…	…	…	…	…	…	…
S	0	1	1	1	1	0	0	…	0

&

U: Target coverage rate

Existing Extrapolator





Our Extrapolator

RQ 2. Stopping-Time Estimation

Q. Is our ***extrapolator*** useful for the stopping criteria?

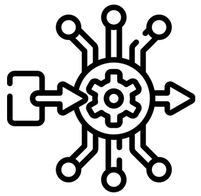
Greybox Coverage Record

	1	2	3	4	5	6	7	…	t_0
1	1	1	1	1	1	1	1	…	1
2	1	1	0	0	0	0	1	…	0
3	0	0	0	1	1	1	0	…	0
4	0	0	0	1	1	0	1	…	1
5	0	0	0	1	1	0	1	…	0
6	1	1	1	1	1	1	0	…	0
7	0	0	0	0	0	0	0	…	0
…	…	…	…	…	…	…	…	…	…
S	0	1	1	1	1	0	0	…	0

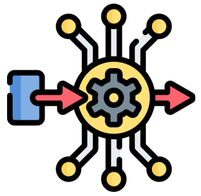
&

***U*: Target coverage rate**

Existing Extrapolator



t'



t''

Our Extrapolator

Estimated time

RQ 2. Stopping-Time Estimation

Q. Is our ***extrapolator*** useful for the stopping criteria?

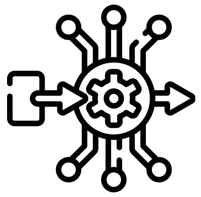
Greybox Coverage Record

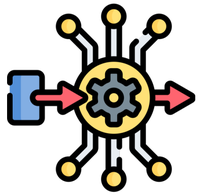
	1	2	3	4	5	6	7	…	t_0
1	1	1	1	1	1	1	1	…	1
2	1	1	0	0	0	0	1	…	0
3	0	0	0	1	1	1	0	…	0
4	0	0	0	1	1	0	1	…	1
5	0	0	0	1	1	0	1	…	0
6	1	1	1	1	1	1	0	…	0
7	0	0	0	0	0	0	0	…	0
…	…	…	…	…	…	…	…	…	…
S	0	1	1	1	1	0	0	…	0

&

U: Target coverage rate

Existing Extrapolator



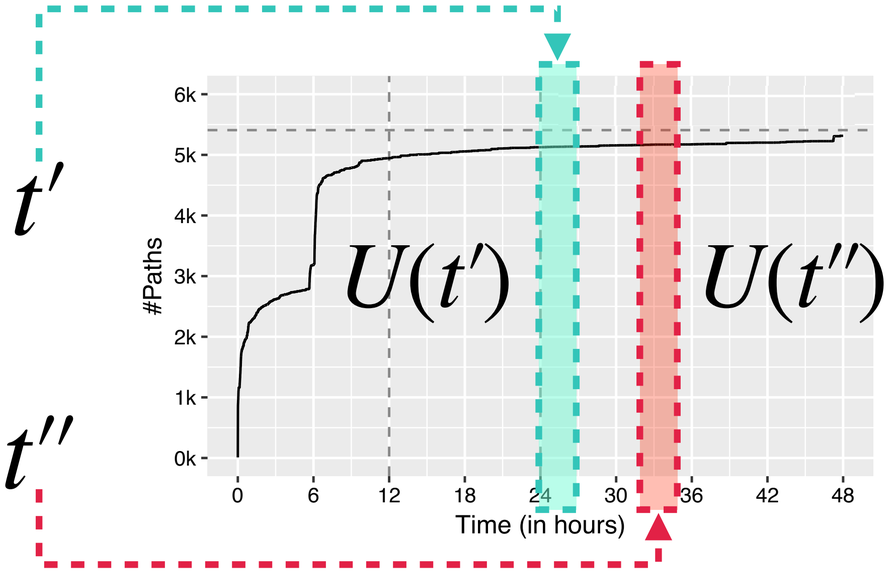


Our Extrapolator

t'

t''

Estimated time



Actual coverage rate

RQ 2. Stopping-Time Estimation

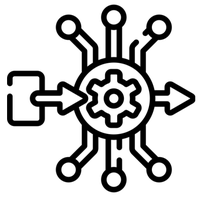
Q. Is our ***extrapolator*** useful for the stopping criteria?

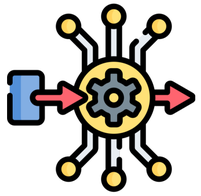
Greybox Coverage Record

	1	2	3	4	5	6	7	…	t_0
1	1	1	1	1	1	1	1	…	1
2	1	1	0	0	0	0	1	…	0
3	0	0	0	1	1	1	0	…	0
4	0	0	0	1	1	0	1	…	1
5	0	0	0	1	1	0	1	…	0
6	1	1	1	1	1	1	0	…	0
7	0	0	0	0	0	0	0	…	0
…	…	…	…	…	…	…	…	…	…
S	0	1	1	1	1	0	0	…	0

&

Existing Extrapolator



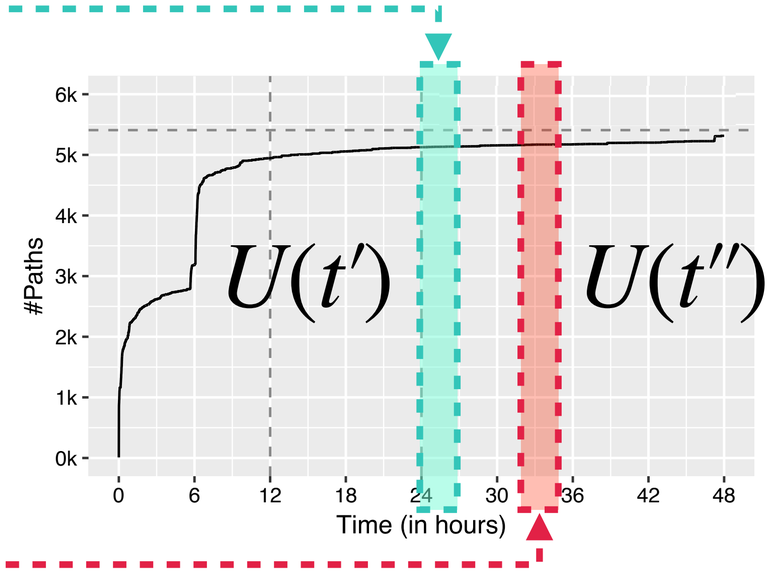


Our Extrapolator

t'

t''

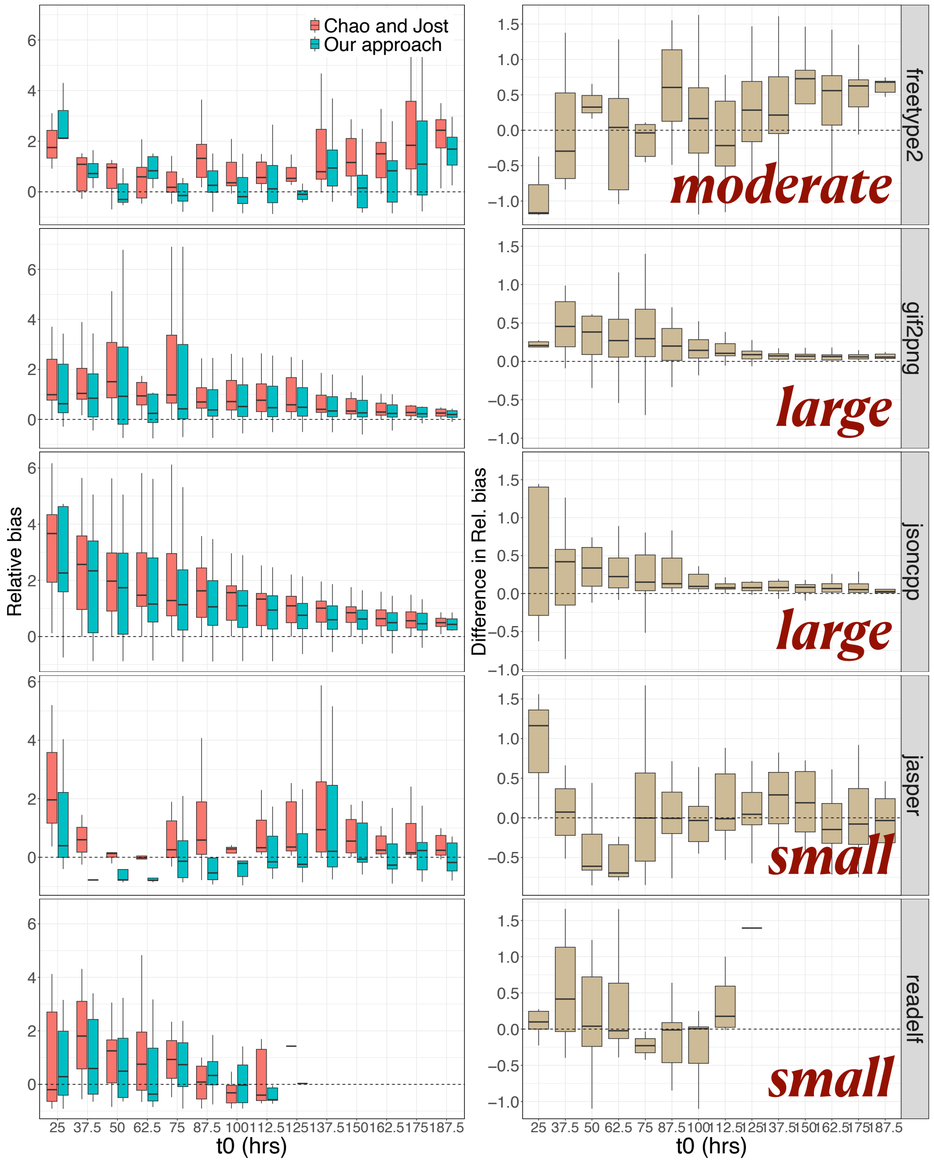
Estimated time



Actual coverage rate

U***: Target coverage rate*** ◀ ····· ▶ **v.s.** U

Average difference between $U(\hat{t})$ and target U



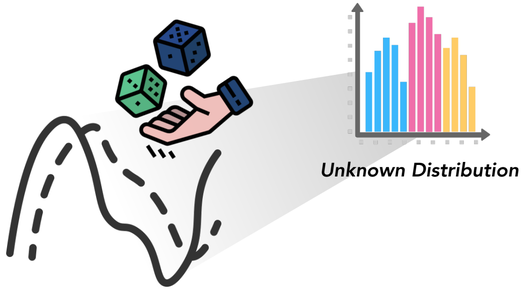
Result 2. Stopping-Time Estimation

(>0 is better)

Bias(Exist) — Bias(Our)

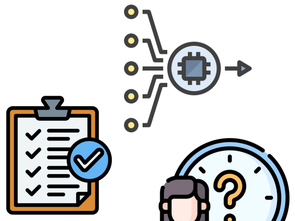
- Our extrapolator predicts the time significantly better (**large/moderate effect size**) than the existing extrapolator for **3 out of 5** subjects.
- For significant improvement, our extrapolator achieves **35-77% closer prediction** than the existing extrapolator.

Q. When should we stop the fuzzing campaign?



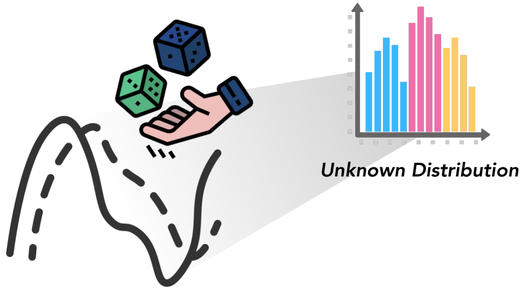
Stochastic Process

Unsolvable with an analytical methodology



Even if we have some criteria,
how much of a resource do we need to use more?

Q. When should we stop the fuzzing campaign?



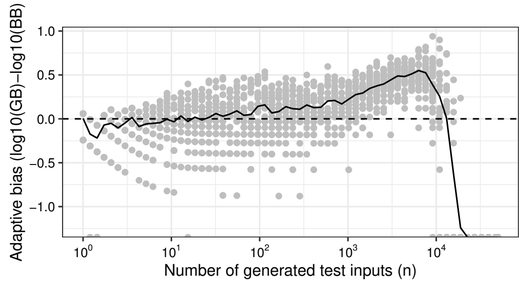
Stochastic Process

Unknown Distribution

Even if we have some criteria,
how much of a resource do we need to use more?

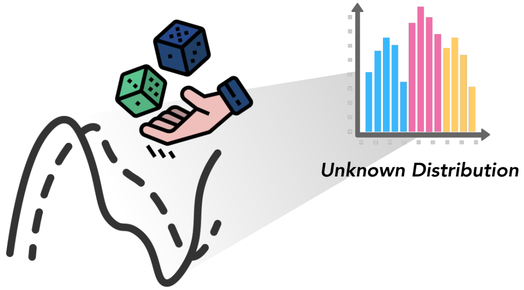
Unsolvable with an analytical methodology

Adaptive bias



"Due to adaptive bias, the discovery probability is consistently higher in the greybox campaign than the blackbox campaign."

Q. When should we stop the fuzzing campaign?



Stochastic Process

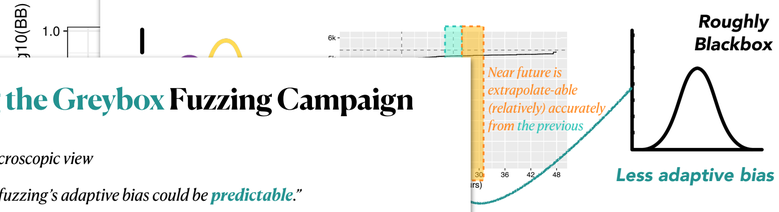
Unknown Distribution

Even if we have some criteria,
how much of a resource do we need to use more?

Unsolvable with an analytical methodology

Extrapolating the Greybox Fuzzing Campaign

- First key insight — *Microscopic view*

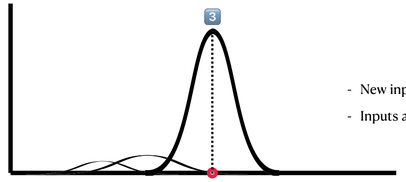


tently
aign."

Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

*"Greybox fuzzing's adaptive bias could be **predictable**."*



- New input that *increases coverage* is found
- Inputs around the new input are sampled

Q. When should we stop the fuzzing campaign?

Even if we have some criteria,
how much of a resource do we need to use more?

Unsolvable with an analytical methodology

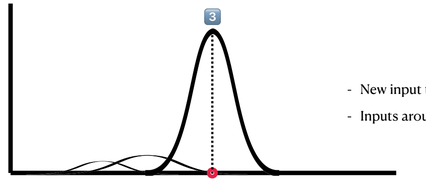
Extrapolating the Greybox Fuzzing Campaign

- First key insight — *Microscopic view*

Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

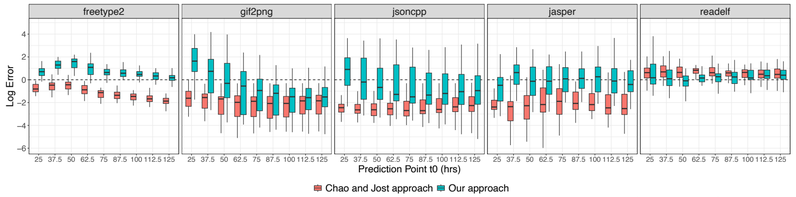
*"Greybox fuzzing's adaptive bias could be **predictable**."*



- New input that *increases coverage* is found
- Inputs around the new input are sampled

Result 1: Coverage Rate Prediction

Difference between $\log(U(t+k))$ ***vs.*** $\log(\hat{U}(t+k))$



(close to 1 is better)

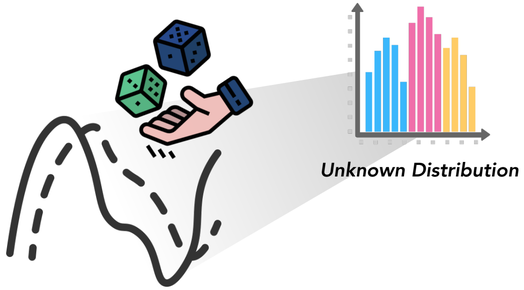
"Our extrapolator exhibits
at least one order of magnitude lower absolute bias
than the existing extrapolator for *4 out of 5 subjects*,
especially for long-term prediction."
⇒ ***Well-handled the adaptive bias***

The average ratio $U(t+k)/\hat{U}(t+k)$:

[Existing] **1.6 - 800** [Ours] **1.17 - 7**

across all subjects.

Q. When should we stop the fuzzing campaign?



Stochastic Process

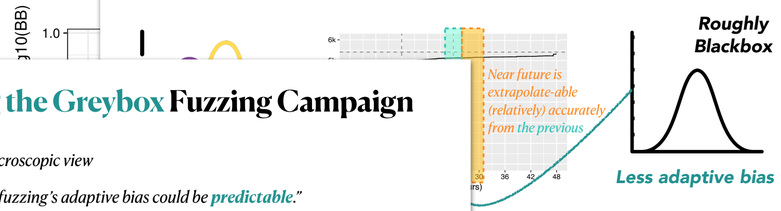
Unknown Distribution

Even if we have some criteria,
how much of a resource do we need to use more?

Unsolvable with an analytical methodology

Extrapolating the Greybox Fuzzing Campaign

- First key insight — *Microscopic view*

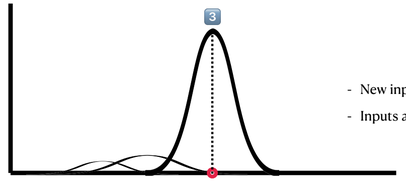


tently
aign."

Extrapolating the Greybox Fuzzing Campaign

- Second key insight — *Macroscopic view*

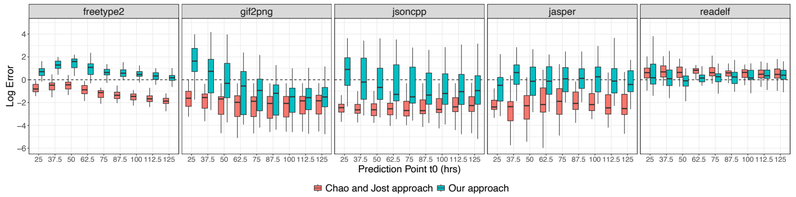
*"Greybox fuzzing's adaptive bias could be **predictable**."*



- New input that *increases coverage* is found
- Inputs around the new input are sampled

Result 1: Coverage Rate Prediction

Difference between $\log(U(t+k))$ ***vs.*** $\log(\hat{U}(t+k))$



(close to 1 is better)

"Our extrapolator exhibits
at least one order of magnitude lower absolute bias
than the existing extrapolator for *4 out of 5 subjects*,
especially for long-term prediction."

⇒ ***Well-handled the adaptive bias***

The average ratio $U(t+k)/\hat{U}(t+k)$:

[Existing] **1.6 - 800** [Ours] **1.17 - 7**

across all subjects.

Dr. Seongmin Lee

🏠 https://nimgnoeseel.github.io/