

PerfFuzz: Automatically Generating Pathological Inputs

Caroline Lemieux, Rohan Padhye, Koushik Sen, Dawn Song University of California, Berkeley

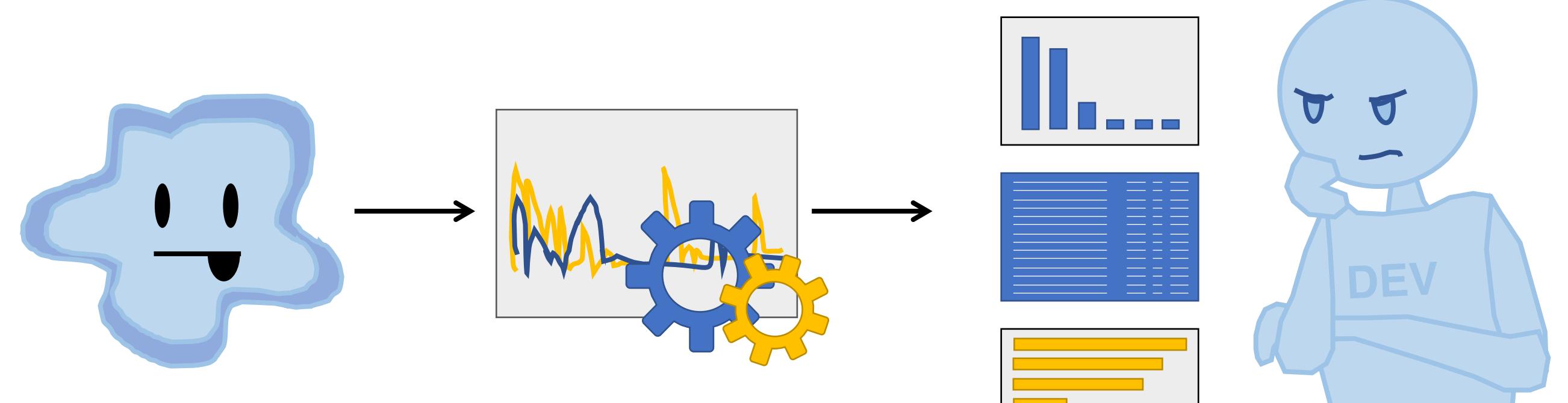
<https://github.com/carolemieux/perffuzz>

Performance problems arise unexpectedly



when programs are run on *pathological* inputs.

Profiling tools help devs diagnose the problem



...if they have a pathological input to start with!

PerfFuzz generates pathological inputs using
feedback-directed mutational fuzzing.



PerfFuzz maximizes **control-flow-graph edge hits**:
a less noisy signal for pathological behavior.

A sequence of tokens: t ?t xt at\$ #a))t Qwaa

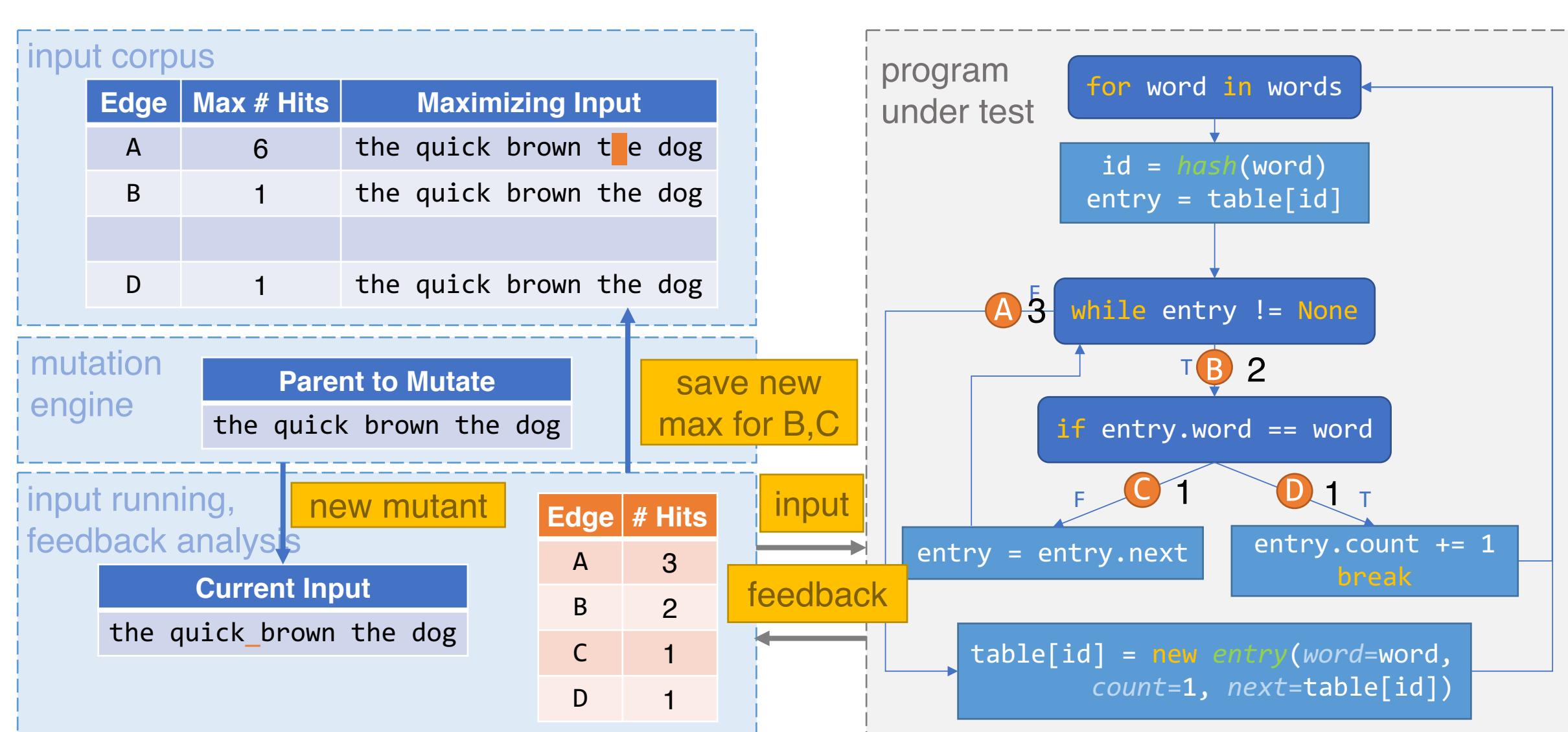
Edge	# Hits
A	7
B	21
C	21
D	0

A blue double-headed arrow connects the sequence of tokens to the table.

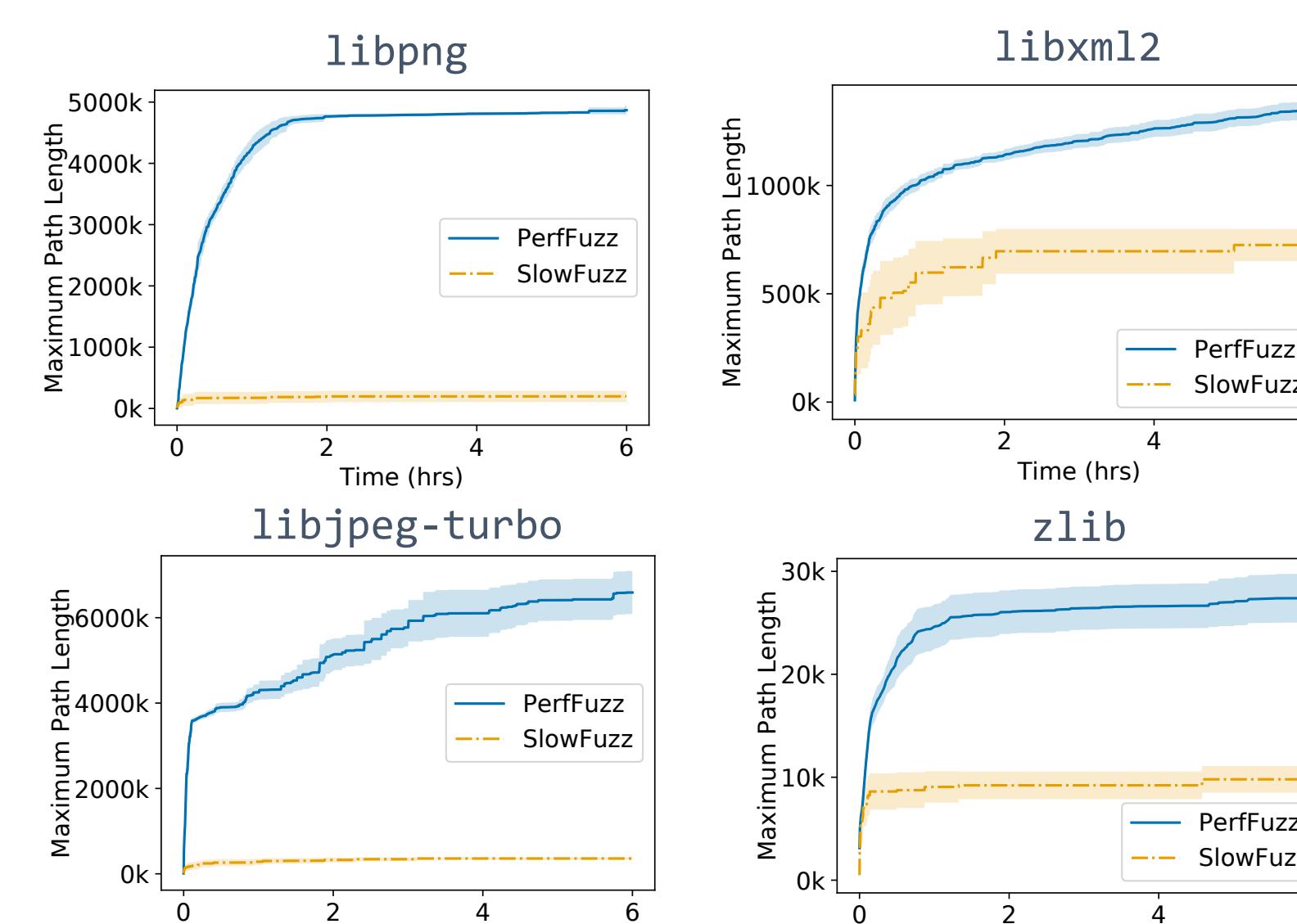
Hash collisions

Linked list traversal edges

Detailed architecture.



Macro-Benchmarks



A diagram illustrating a table of edge hits and its relationship to path length and a hot spot. The table has two columns: "Edge" and "# Hits". The "# Hits" column contains values 7, 21, 21, and 0 for edges A, B, C, and D respectively. Edge C is highlighted with a blue border. Arrows point from the table to two boxes: "Path length: 49" and "Hot spot: 49".

Edge	# Hits
A	7
B	21
C	21
D	0

Path length: 49

Hot spot: 49

PerfFuzz worst case? A loop in xml strncpy...

```
parser error : Double hyphen within comment: <!--3
<a>>>0>>>#>G<!--3--6-----4-----  
^  
parser error : Double hyphen within comment: <!--3--6
<a>>>0>>>#>G<!--3--6-----4-----  
^  
parser error : Double hyphen within comment: <!--3--6
<a>>>0>>>#>G<!--3--6-----4-----  
^  
parser error : Double hyphen within comment: <!--3--6
<a>>>0>>>#>G<!--3--6-----4-----  
^  
parser error : Double hyphen within comment: <!--3--6
<a>>>0>>>#>G<!--3--6-----4-----  
^  
parser error : Double hyphen within comment: <!--3--6
<a>>>0>>>#>G<!--3--6-----4-----  
^  
parser error : Double hyphen within comment: <!--3--6
<a>>>0>>>#>G<!--3--6-----4-----  
^  
parser error : Double hyphen within comment: <!--3--6
<a>>>0>>>#>G<!--3--6-----4-----  
^
```

quadratic complexity

Micro-Benchmarks

