

Free Ride for Resilience: from the Perspective of Software Engineering

Bohan Zhang (Iowa), Zhu Zhu (GMU), **Lishan Yang (GMU)**,
Hui Xu (Fudan U), Guanpeng Li (Iowa)

The logo for Iowa University of Science and Technology, featuring the word "IOWA" in bold, black, sans-serif capital letters on a yellow rectangular background.

IOWA

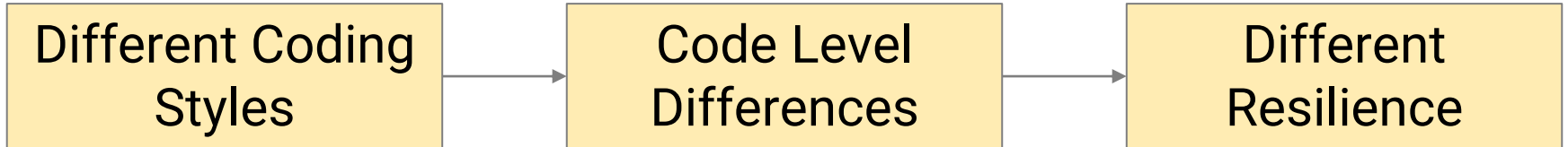
The logo for George Mason University, featuring a stylized green and yellow leaf above the text "GEORGE MASON UNIVERSITY" in green, sans-serif capital letters.

GEORGE
MASON
UNIVERSITY

The logo for Fudan University, featuring a circular seal with the text "FUDAN UNIVERSITY" and "1905" around the perimeter, and the Chinese characters "復旦大學" in blue calligraphy to the right, with "FUDAN UNIVERSITY" in blue, sans-serif capital letters below.

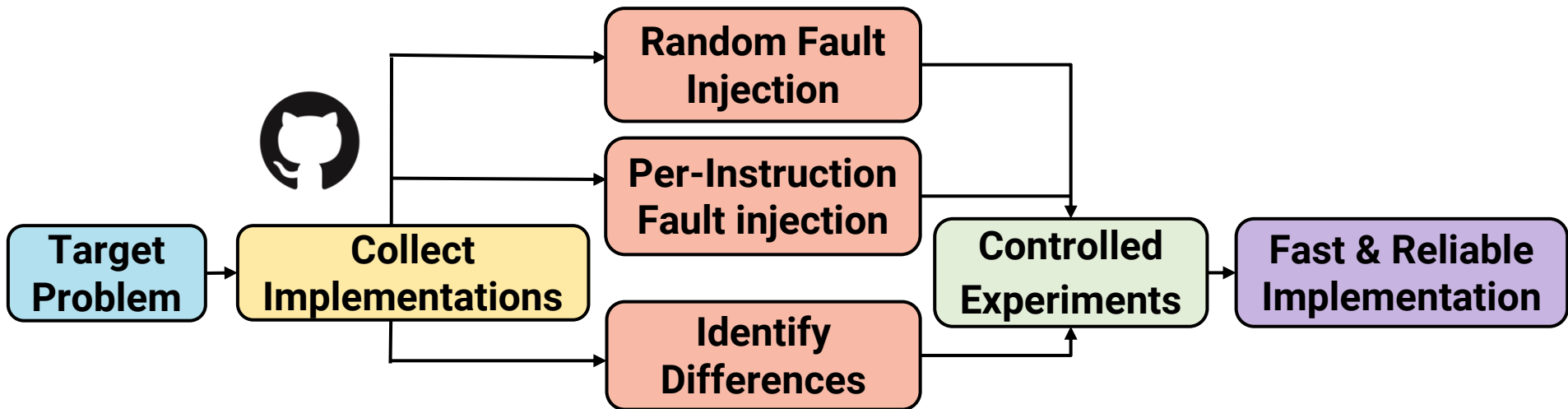
FUDAN UNIVERSITY
1905
復旦大學
FUDAN UNIVERSITY

Idea

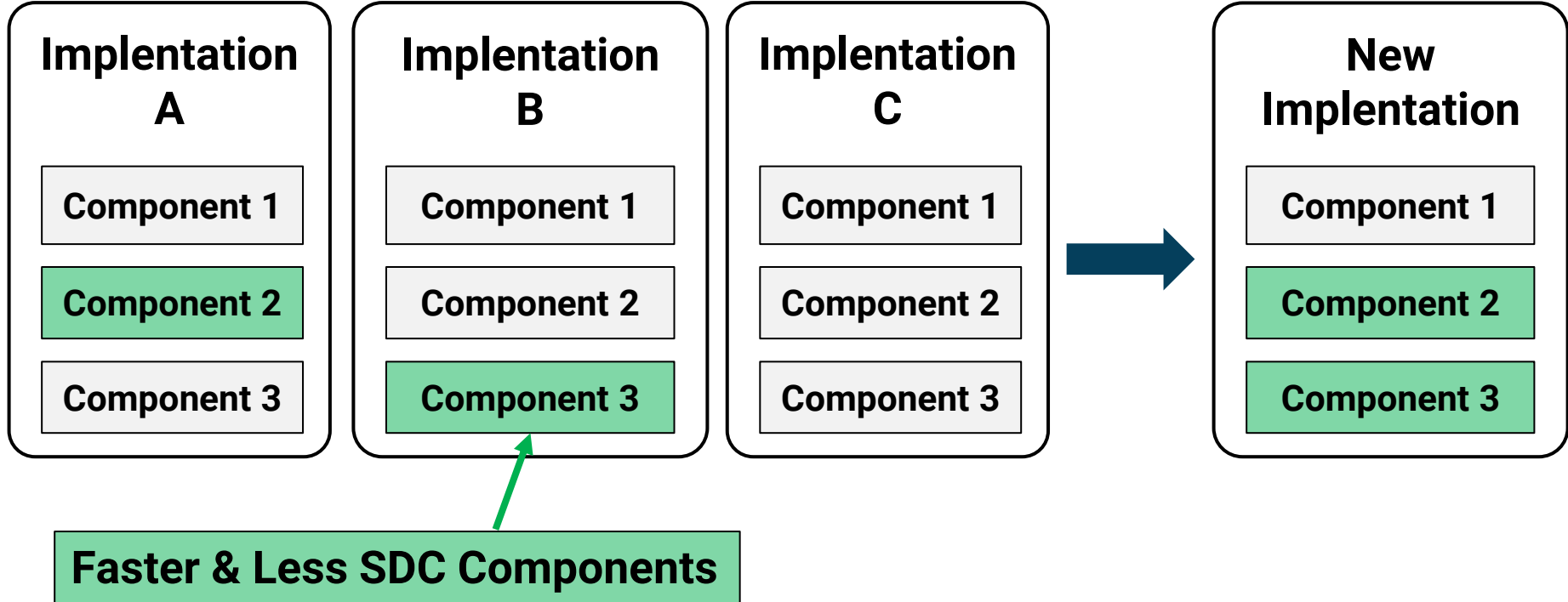


Free Ride: Low SDC Probability & High Performance?

Approach



Approach



Proof-of-Concept

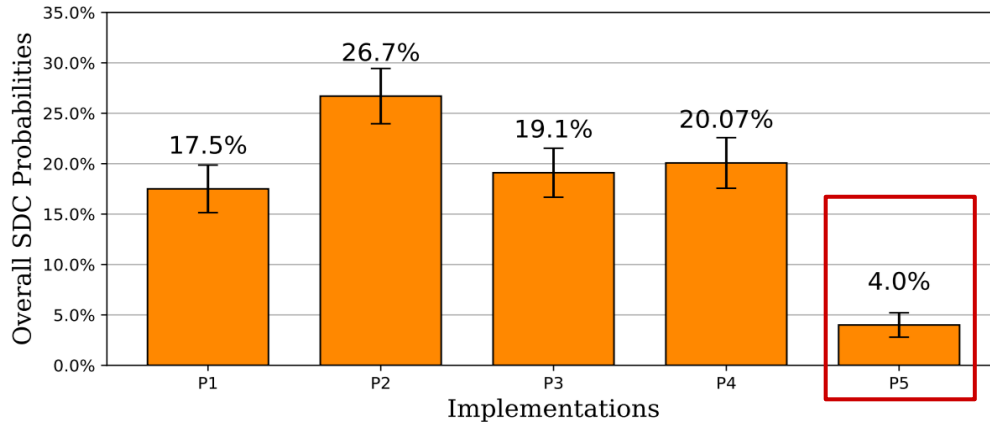
- A Debian server with two 20-core CPU
- Fault Model
 - Single bit-flip injections - accurate
 - Errors in computation units/data path
 - One fault per program execution
 - Use LLFI for fault injection
- Application : BubbleSort
 - **Outer loop**
 - **Inner loop**
 - **Swaption**

Algorithm 1 Bubble Sort

```
1: procedure BUBBLESORT( $A$ )
2:    $n \leftarrow |A|$ 
3:   for  $i \leftarrow 1$  to  $n - 1$  do                                ▷ Outer loop
4:     for  $j \leftarrow 0$  to  $n - i - 1$  do                          ▷ Inner loop
5:       if  $A_j > A_{j+1}$  then                                       ▷ Swaption
6:         swap  $A_j$  and  $A_{j+1}$ 
7:       end if
8:     end for
9:   end for
10: end procedure
```

Proof-of-Concept

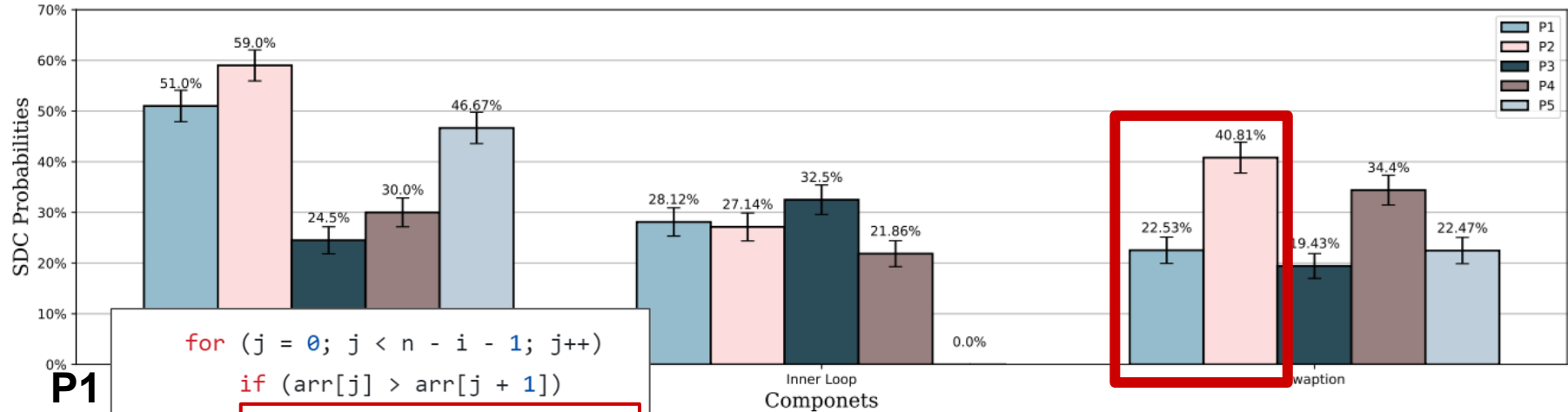
- **Overall SDC probabilities accros 5 programs**
 - The SDC probabilities range: 4% ~ 26.7%
 - P5's DI count: much higher



NUMBER OF DYNAMIC INSTRUCTIONS OF 5 PROGRAMS.

Implementations	Number of Dynamic Instructions
P1	5,889,058,005
P2	6,087,856,752
P3	5,685,594,374
P4	5,489,077,998
P5	9,088,698,022

Proof-of-Concept



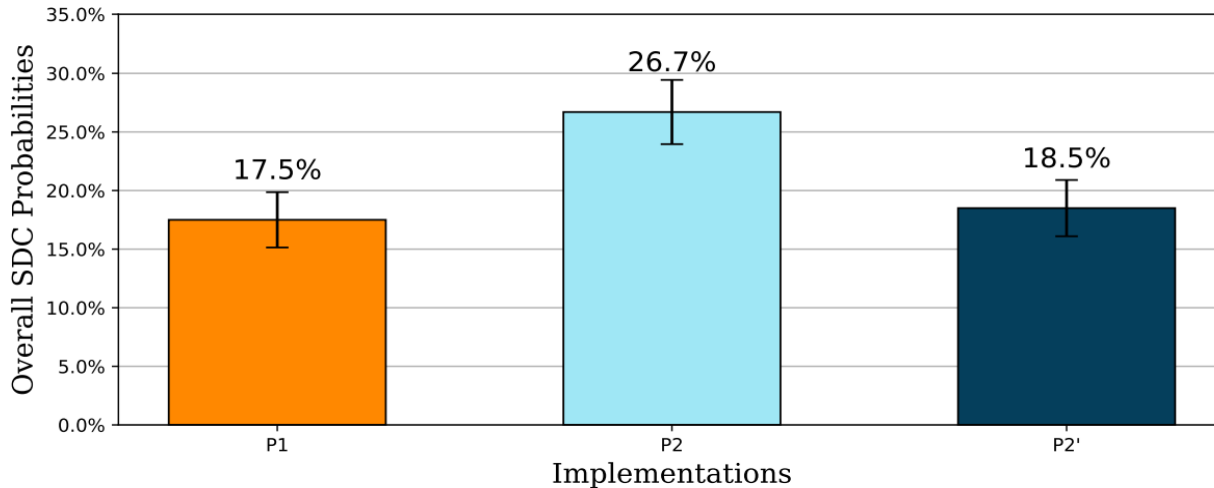
```
for (j = 0; j < n - i - 1; j++)  
    if (arr[j] > arr[j + 1])  
        swap(arr[j], arr[j + 1]);
```

```
for (int i = 0; i < size - step; ++i) {  
    if (array[i] > array[i + 1]) {  
        int temp = array[i];  
        array[i] = array[i + 1];  
        array[i + 1] = temp;
```

Difference: The swaption in P1 is through a function call whereas it is directly inlined in P2.

Result

- **Controlled experiment of Hypothesis**
 - P1 swaption replaced with P1 implementation to generate P2'
 - The SDC probability of P2' is only **18.5%**



Future Works

- Future Work
 - Generalize methodology
 - Different performance measures and tradeoffs
 - Create comprehensive guidance for reliable coding

 - Github & LeetCode problems
 - Course Projects
 - Human studies