



UNIVERSITÄT ZU LÜBECK
INSTITUTE FOR IT SECURITY

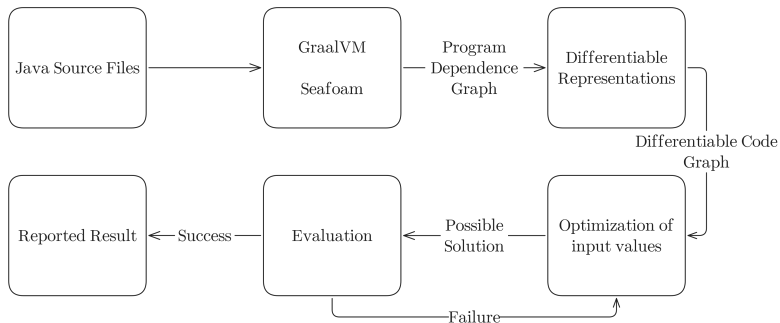
DASA: Differentiable Abstract Symbolic Analyzer

SV-Comp 2026

Felix Mächtle, **Jan-Niclas Serr**, Nils Loose, Thomas Eisenbarth

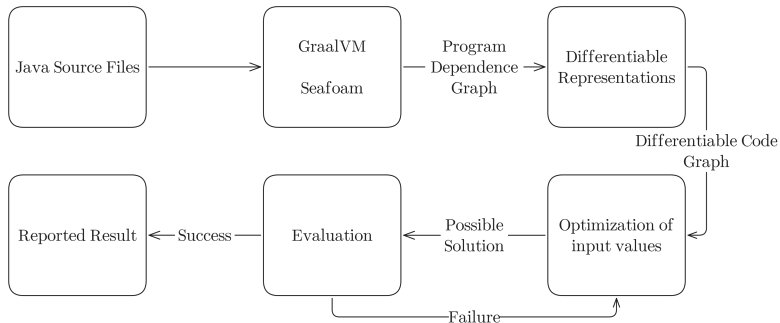
April 13, 2026

Institute for IT Security,
University of Lübeck,
Germany



- Idea: Make the entire System under Test differentiable
- Use gradient-based optimizer to find suitable input values
- Similar to static Symbolic Execution

Differentiable Representations

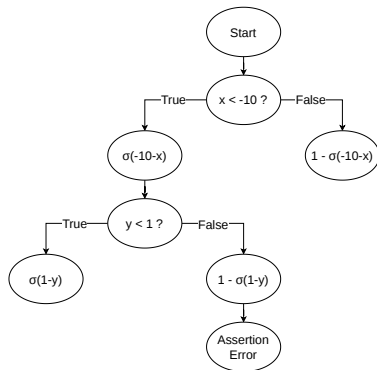


- Internally, everything is represented as float
- Every node has a value that indicates if it is reached with current inputs
- Use stretched sigmoid curves for decision functions

Example

```
public static void main(String[] args) {  
    int x = Verifier.nondetInt();  
    int y = Verifier.nondetInt();  
    if (x < -10 && y > 1) {  
        assert false;  
    }  
}
```

$$\Rightarrow \sigma(-10 - x) * (1 - \sigma(1 - y))$$



Try DASA!

Any Questions?



f.maechtle@uni-luebeck.de



[https://github.com/UzL-ITS/
dasa-program-analysis](https://github.com/UzL-ITS/dasa-program-analysis)



Institute for IT Security, Uni Lübeck



GitHub