

Ajitha Rajan
School of Informatics

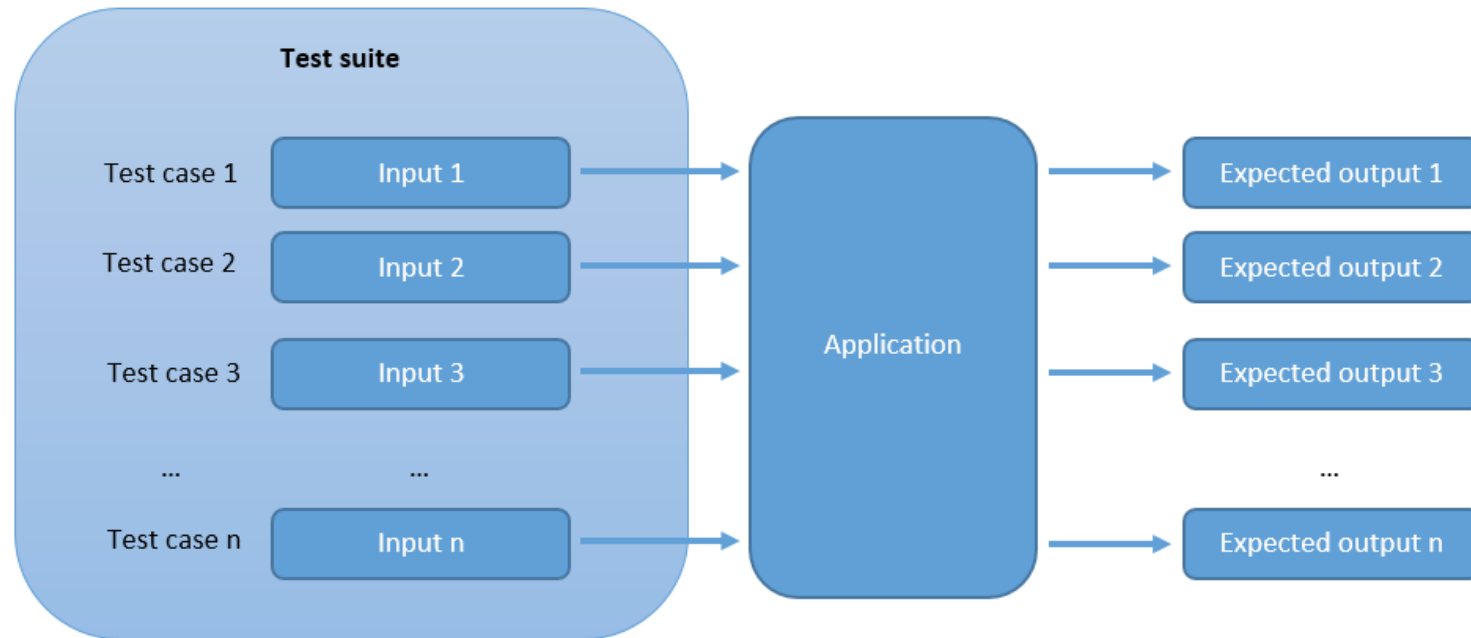
Research Interests – Software Testing,
Metrics for Quality Assurance, Optimising
Software Energy Consumption

arajan@staffmail.ed.ac.uk
<http://homepages.inf.ed.ac.uk/arajan/>

The Problem

Software testing is time consuming

Functional testing



The test suite of a non-trivial system:

- could have **thousands** of test cases
- could take **hours, days** or even **weeks** to execute



Software Testing

- Widespread and applied to software of any size
- Major problems with the efficiency and effectiveness of testing in its current form in practice. (Annual costs in the US run up to \$59.5 billion [NIST])
- Software is getting larger and more complex



12 Million Lines of Code



61 Million LOC



14 Million LOC

- The global cost of software bugs is estimated at **312 billions of dollars annually.**

Why is Software Testing Hard

Input/state space

```
public Line(int x0, int y0, int x1, int y1)
```

- Input per *int*: 2^{32} different values:
- $2^{32} \times 2^{32} \times 2^{32} \times 2^{32}$ different values

1000 lines per second: 10^{28} years

Execution Sequences

```
for (int i=0, i < n; i++) {  
    if (a.get(i) == b.get(i)) {  
        x[i] += 100;  
    } else {  
        x[i] /=2;  
    }  
}
```

2^n paths with n iterations

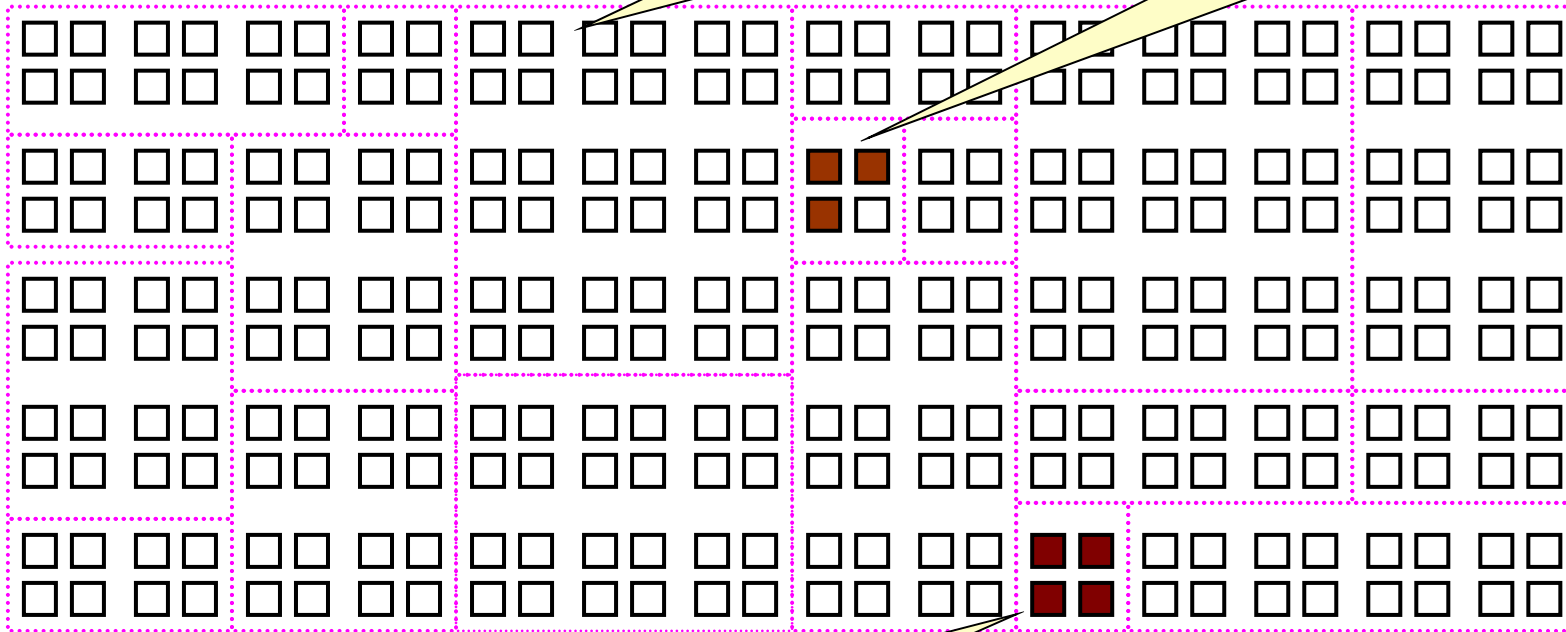
Systematic Partition Testing

- Failure (valuable test case)
- No failure

Failures are sparse in the space of possible inputs ...

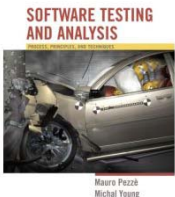
... but dense in some parts of the space

The space of possible input values
(the haystack)



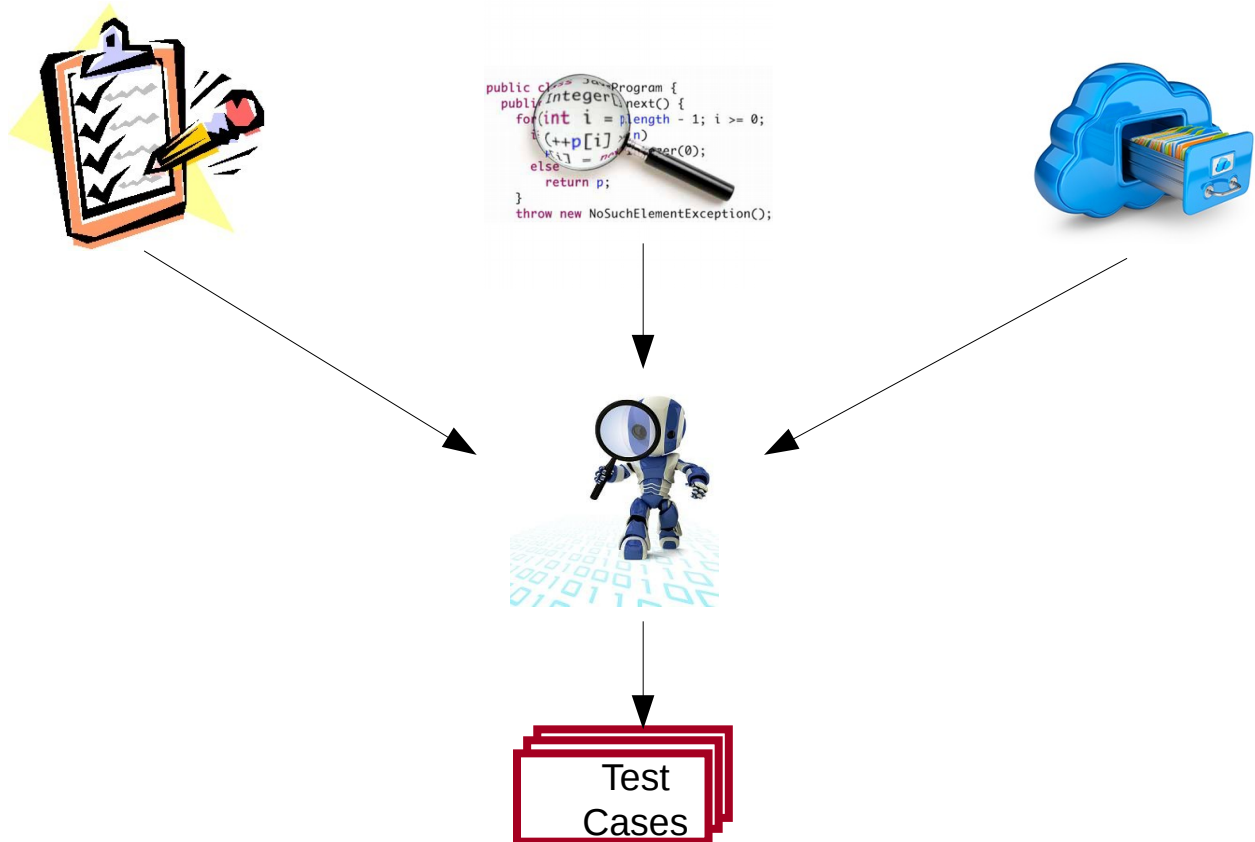
If we systematically test some cases from each part, we will include the dense parts

Functional testing is one way of drawing pink lines to isolate regions with likely failures

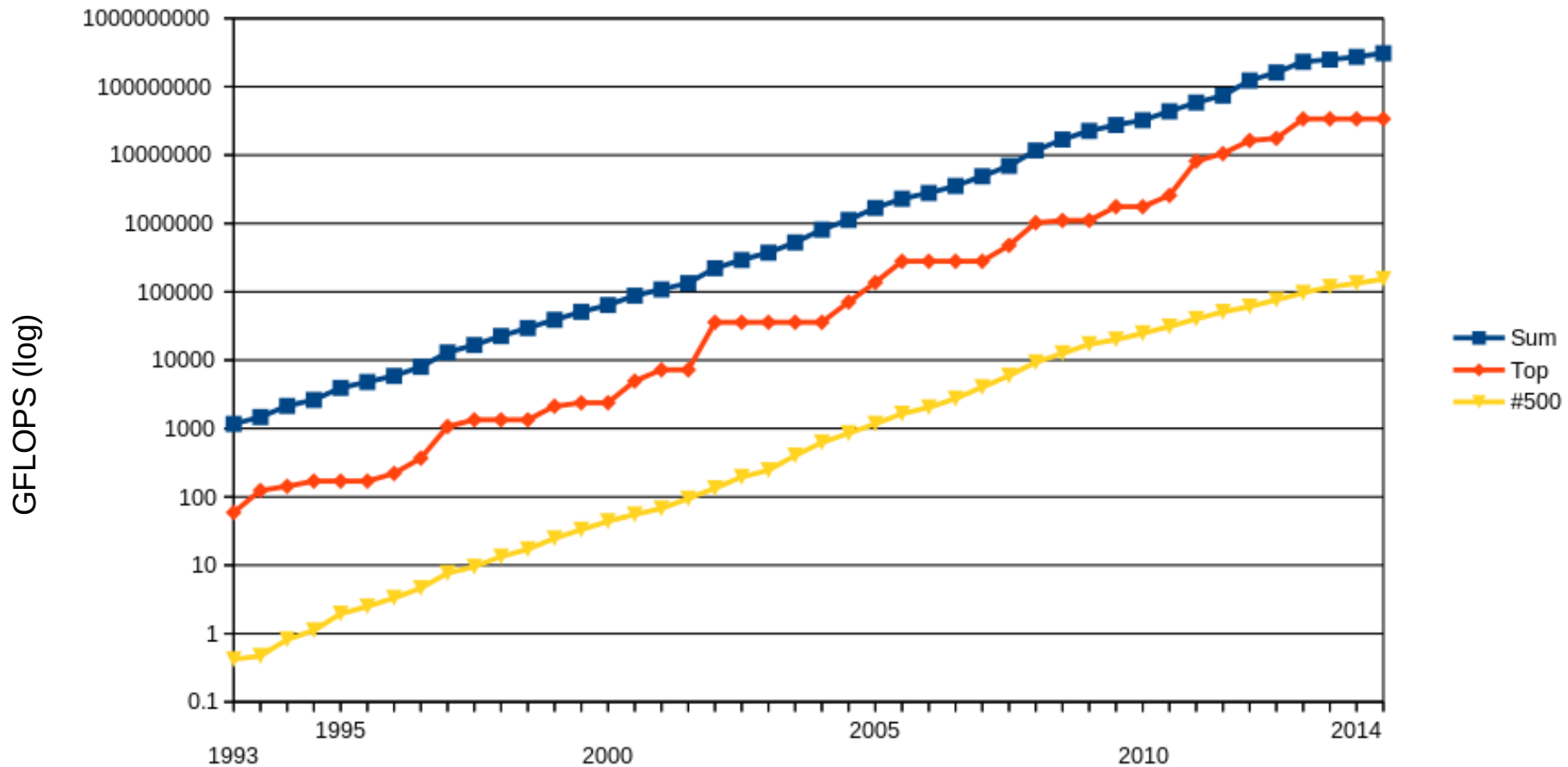


Research Problem

Generating software test cases using a combination of machine learning and program analysis techniques.



Ever increasing computing power

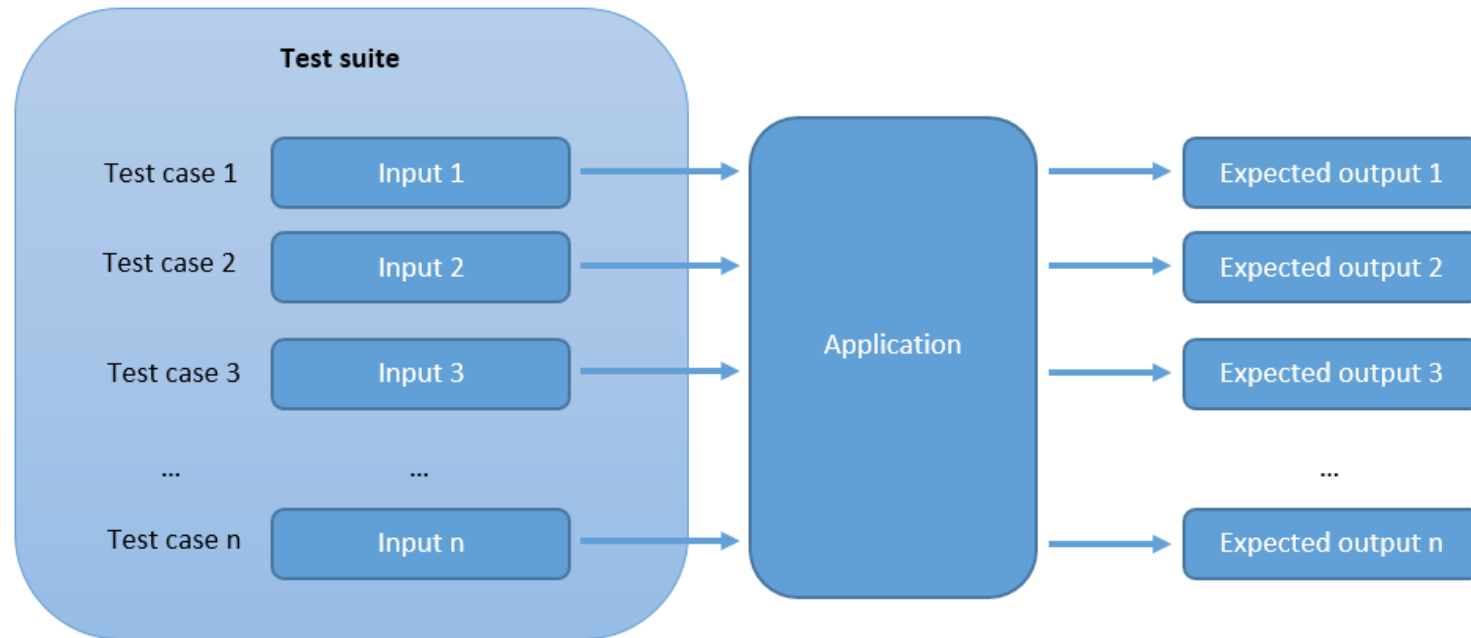


Only about 20% of the available power resources is getting used!

The Problem

Software testing is time consuming

Functional testing



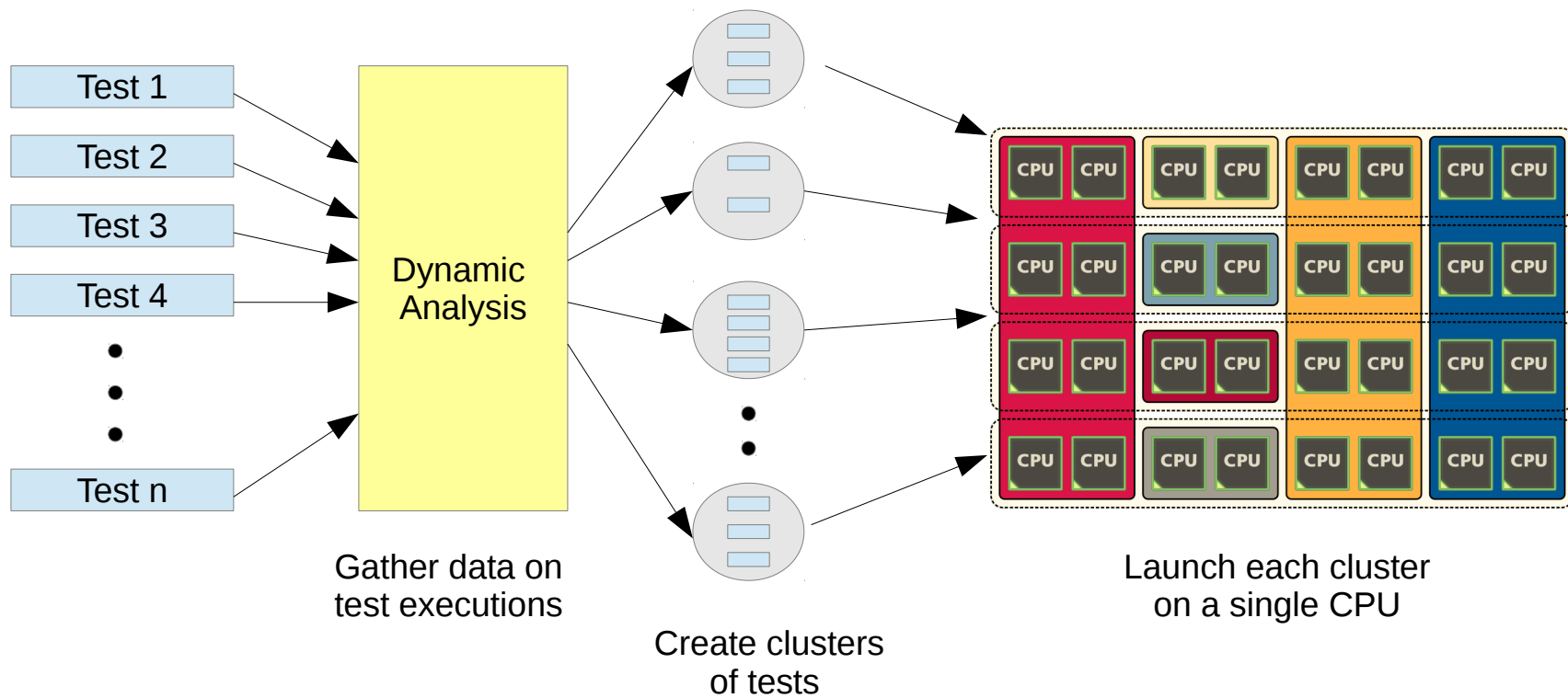
Characteristics:

1. Executions are independent
2. Executions are data parallel

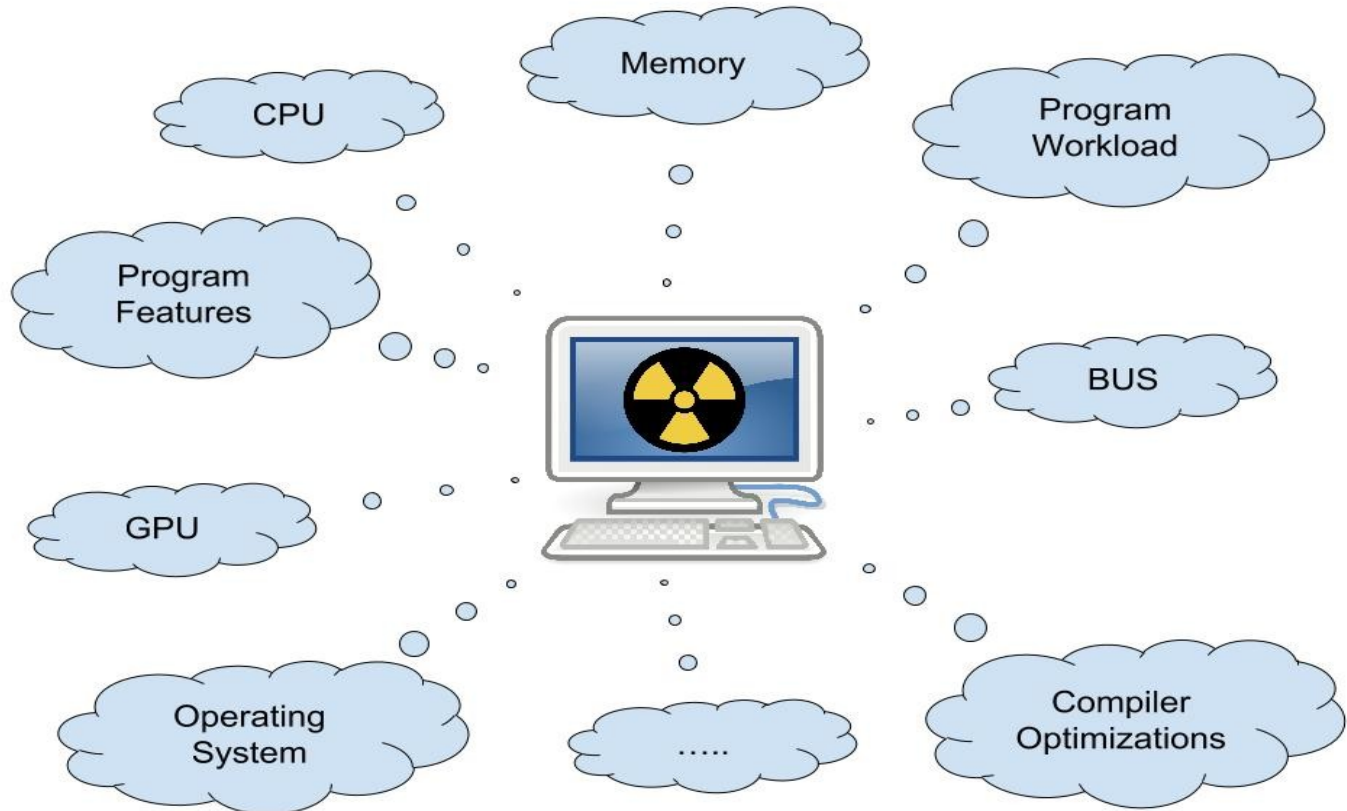


Research problem

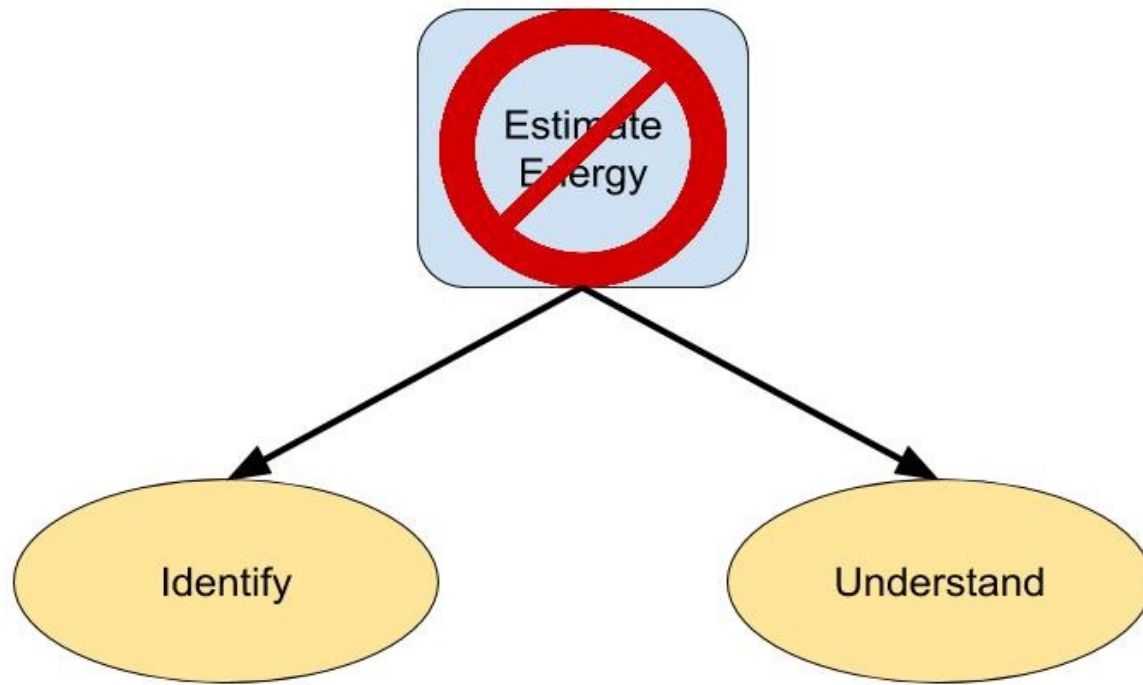
How do you cluster tests for parallel execution on multiple cores?



Factors Affecting Energy in Software



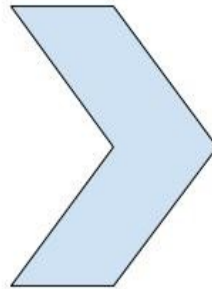
Our Empirical Study



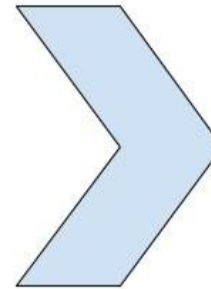
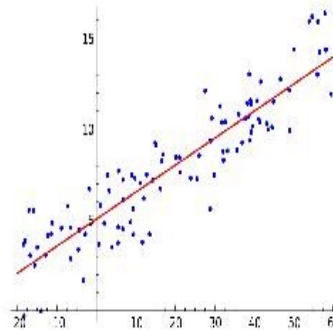
Both Software and Hardware factors.

Approach

Data Collection



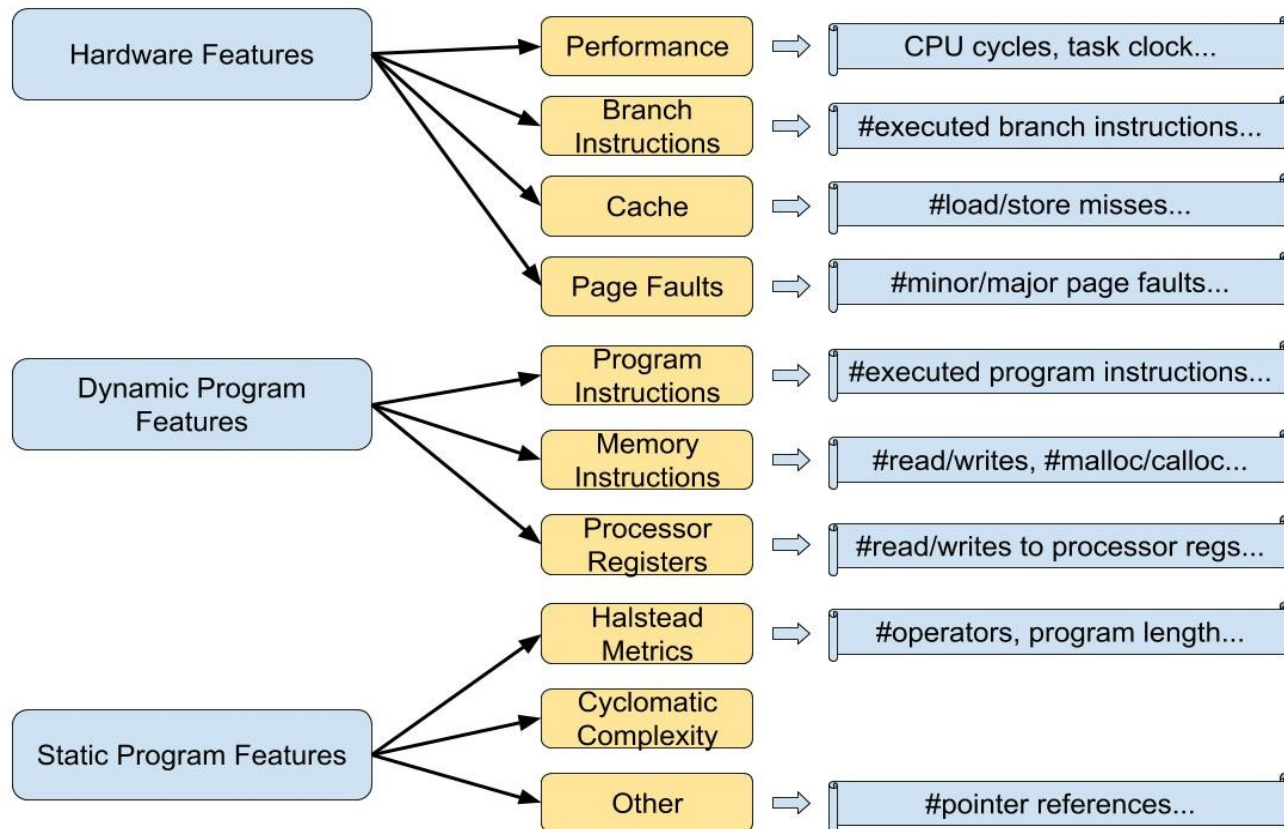
Feature Selection



Validation



Data Collection – Metrics





Feature Selection

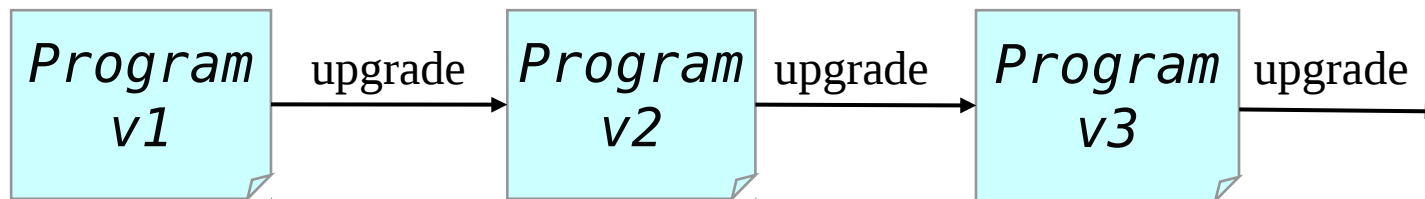
- A machine learning technique.
- Selects a subset of statistically relevant features for use in model construction.



Evolving Software

Large software systems are usually built incrementally:

- **Maintenance** - fixing errors and flaws, hardware changes
- **Enhancements** - new functionality, improved efficiency, extension, new regulations



Research Interests

1. What is the effort/cost of upgrades?
2. Test adequacy criteria for regression testing
3. What is the regression test effort/cost?

Cost Models

- Highly inaccurate and research in building these models is from a decade ago
- DATA DATA DATA – Machine learning techniques that considers program change sizes, organisational factors, developer experience, past project data.
- Combine them with precise static analysis