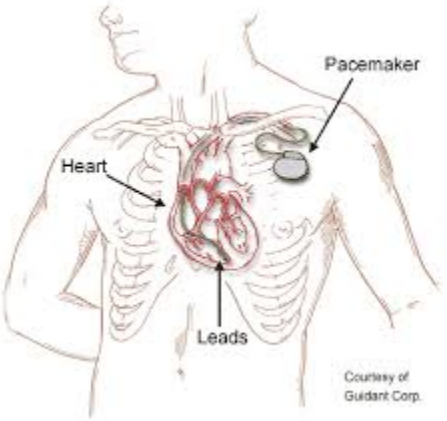
A Flexible Intrusion Detection System for Memory-Constrained Embedded Systems

Farid Molazem Tabrizi
Karthik Pattabiraman

University of British Columbia (UBC)
http://blogs.ubc.ca/karthik/

Embedded Systems: IoT



IoT Example: Smart Meter

- **Smart meter Attacks**
  - No need for physical presence
  - Hard to detect by inspection or testing
  - Attacks can be large-scale

Analog Meter

Smart Meter

Our Goal

09 APR 12

FBI: Smart Meter Hacks Likely to Spread

A series of hacks perpetrated against so-called "smart meter" installations over the past several years may have cost a single U.S. electric utility hundreds of millions of dollars annually, the FBI said in a cyber intelligence bulletin obtained by KrebsOnSecurity. The law enforcement agency said this is the first known report of criminals compromising the hi-tech meters, and that it expects this type of fraud to spread across the country as more utilities deploy smart grid technology.

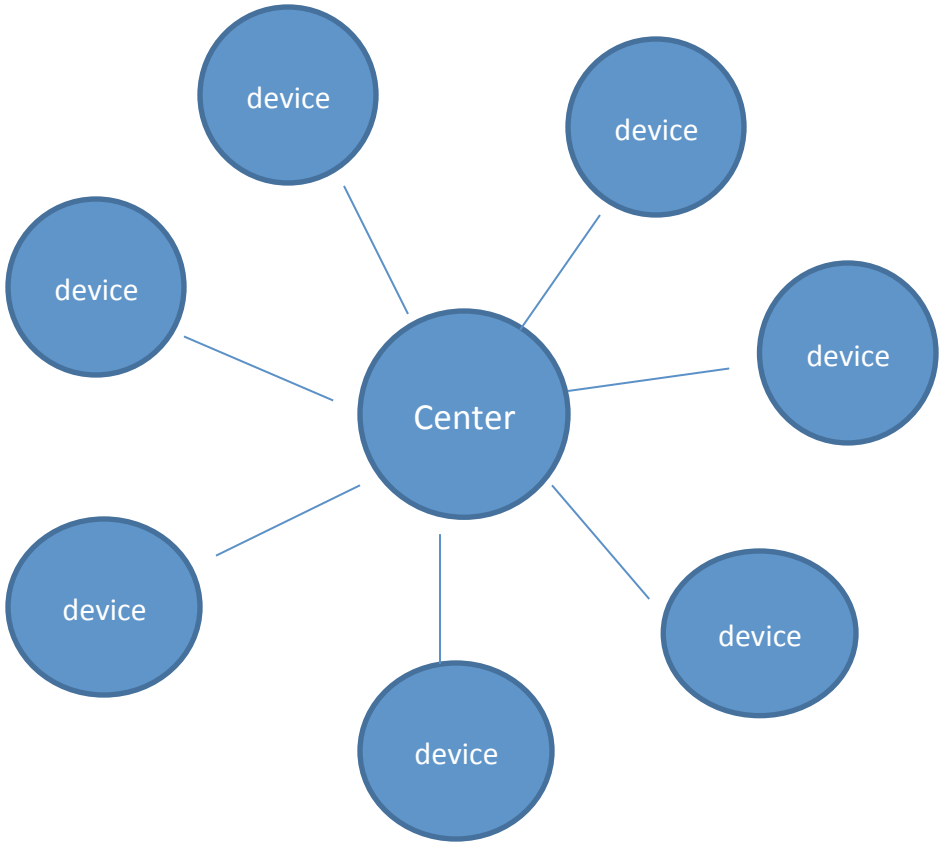
Smart meters are intended to improve efficiency, reliability, and allow the electric utility to charge different rates for

FEDERAL BUREAU OF INVESTIGATION
INTELLIGENCE BULLETIN
Cyber Intelligence Section

27 May 2010

Build an Intrusion Detection System

Challenge: False Positives

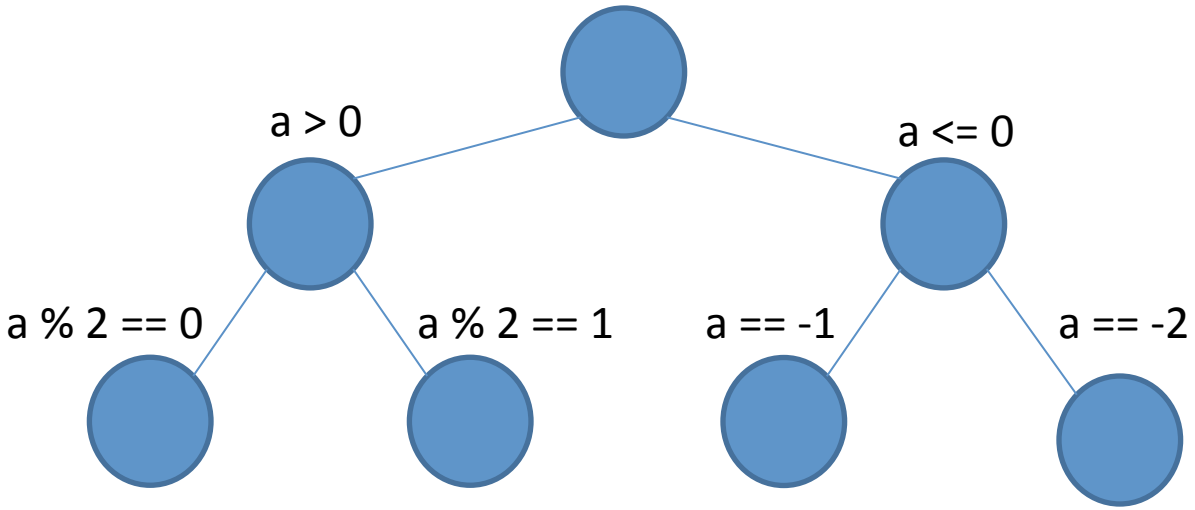


Challenge: Memory Constraints

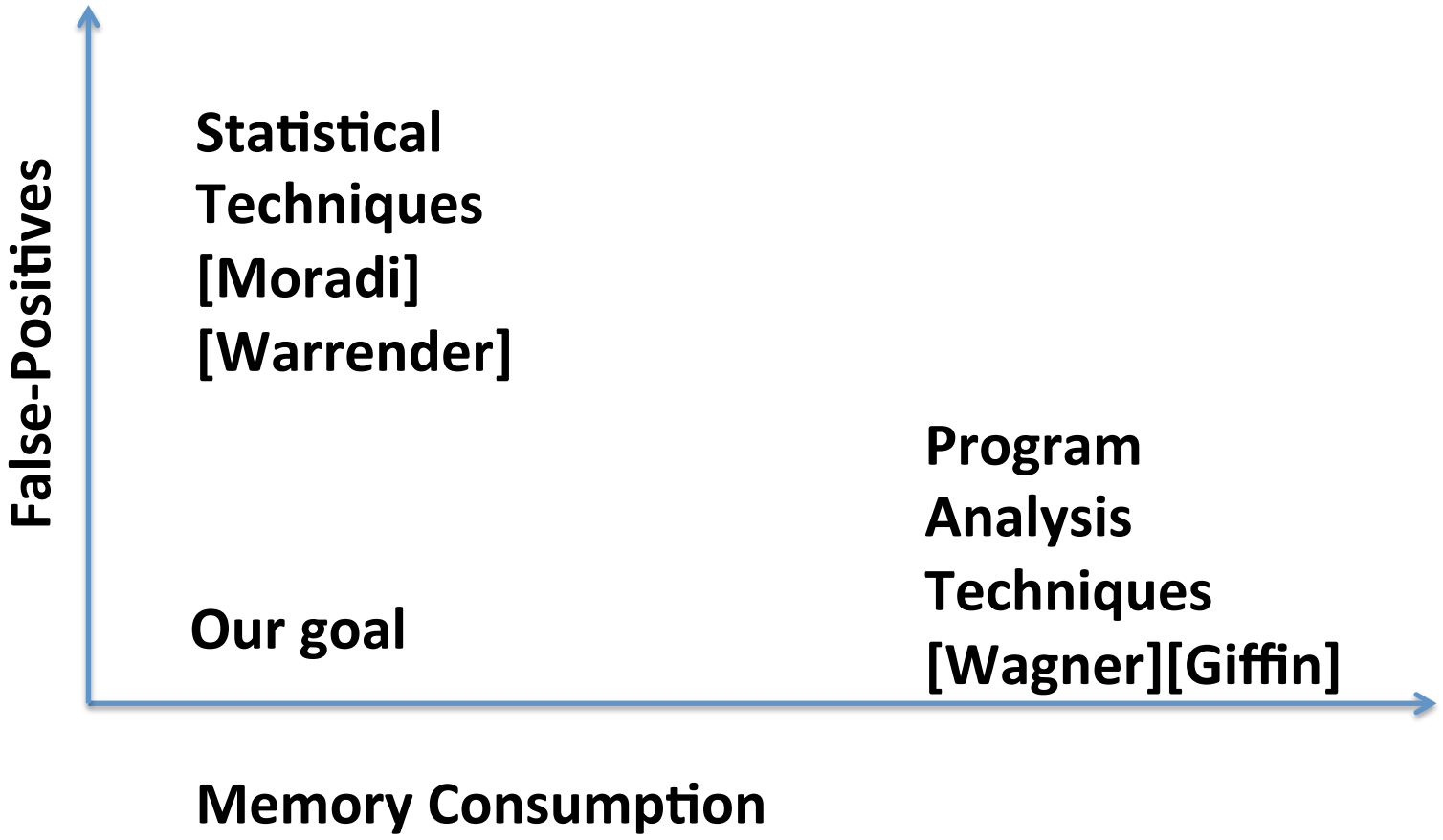
```
{
  a = receive();
  if (a > 0)
    foo(a);
  else
    bar(a);
}
```

```
void foo(int a) {
  if (a % 2 == 0)
    even(a);
  else
    odd(a);
}
```

```
void bar(int a) {
  if (a == -1)
    error1();
  else if (a == -2)
    error2();
}
```



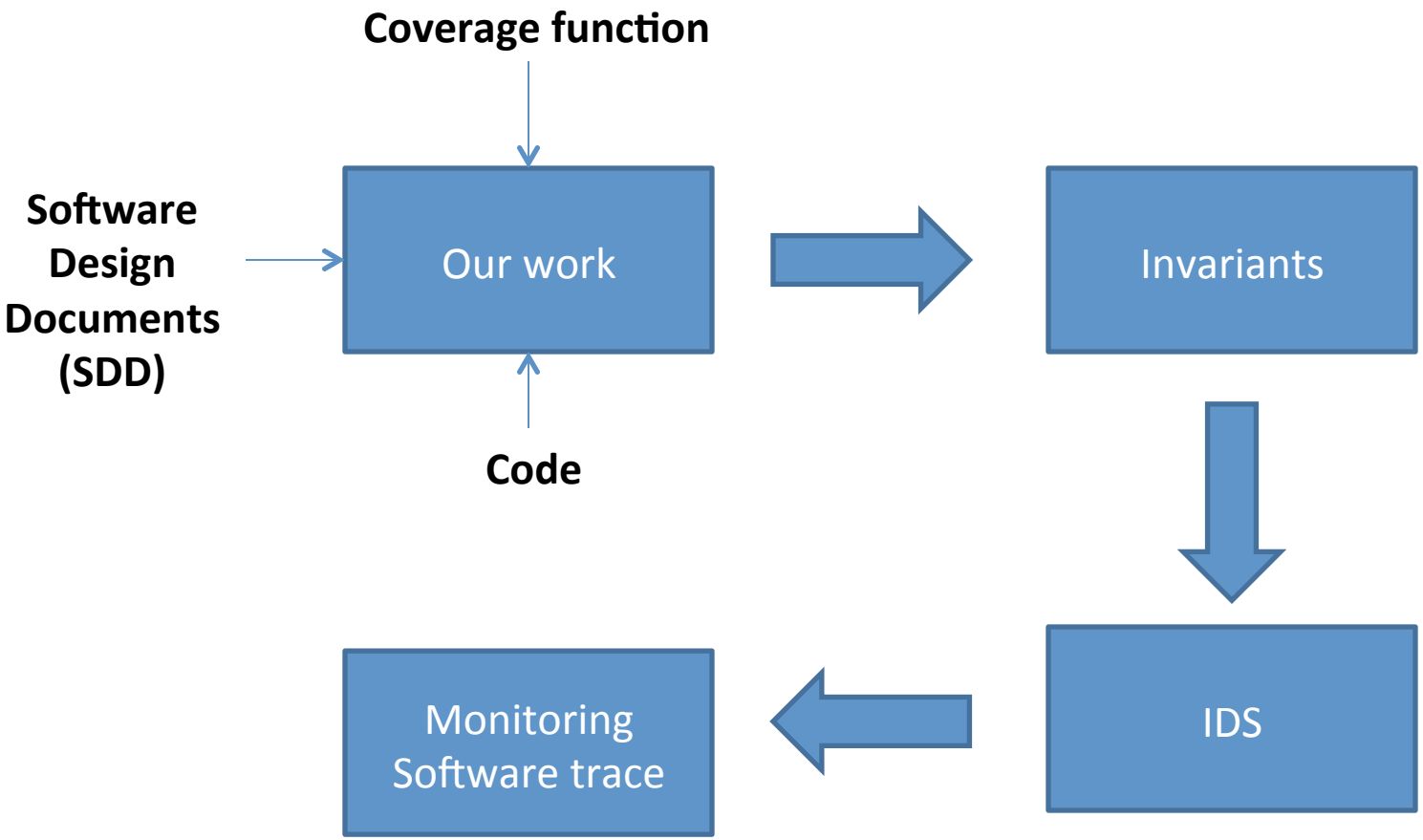
Existing Solutions



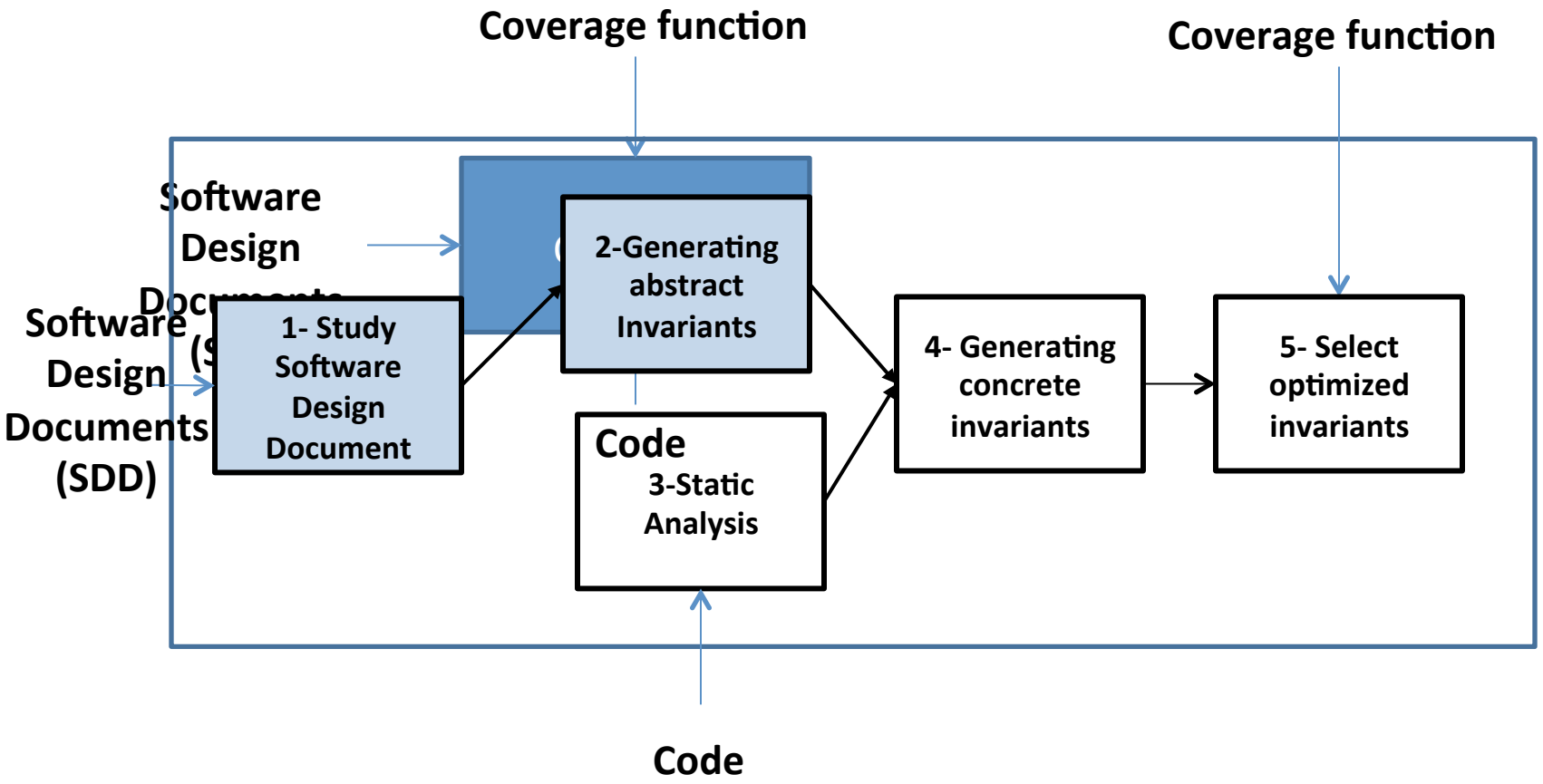
Idea

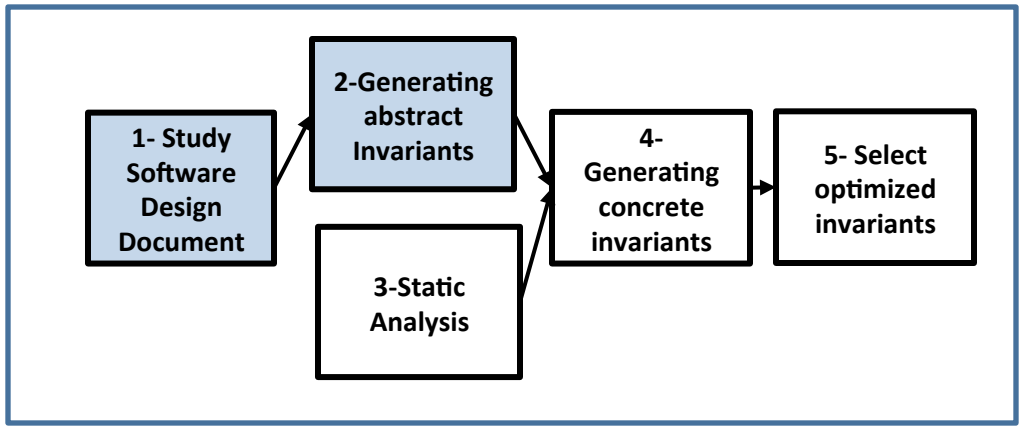
- Quantify security to detect only the most critical attacks, subject to memory constraints

Approach: Overview



Approach: Details

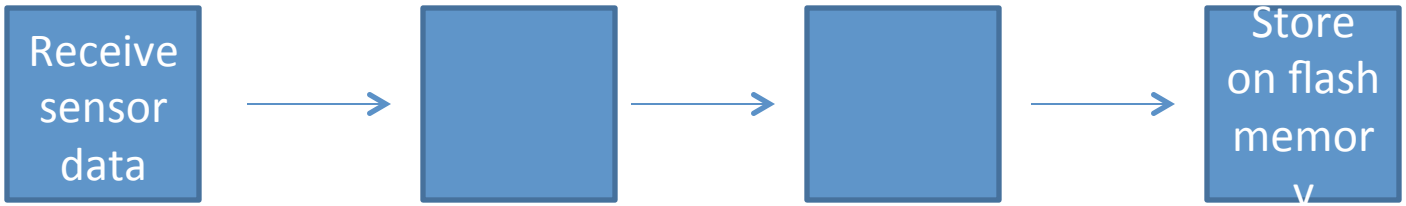




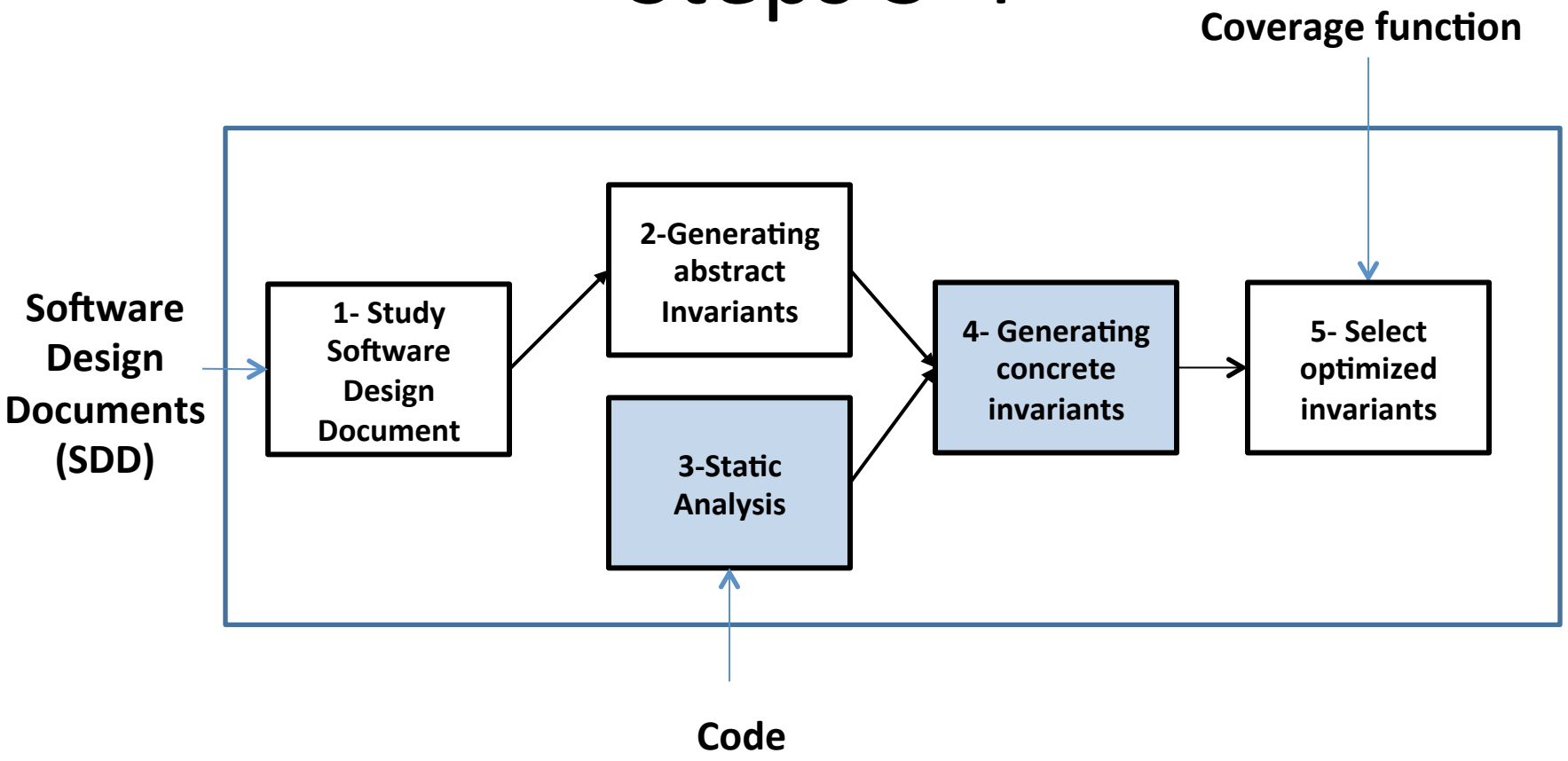
- Storage/Retrieval integrity

Sensor data must eventually be stored on flash memory

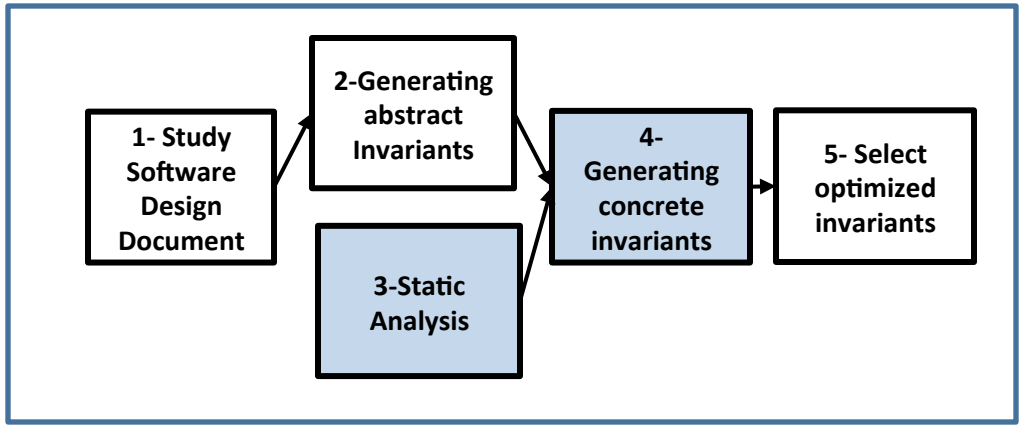
$$\Box(getting\ sensorData \Longrightarrow (\Diamond store\ on\ flash))$$



Steps 3-4



Abstract invariants → Concrete invariants (contain system calls)




```
{
    ....
    data = socket.receive();
    ....
}
```

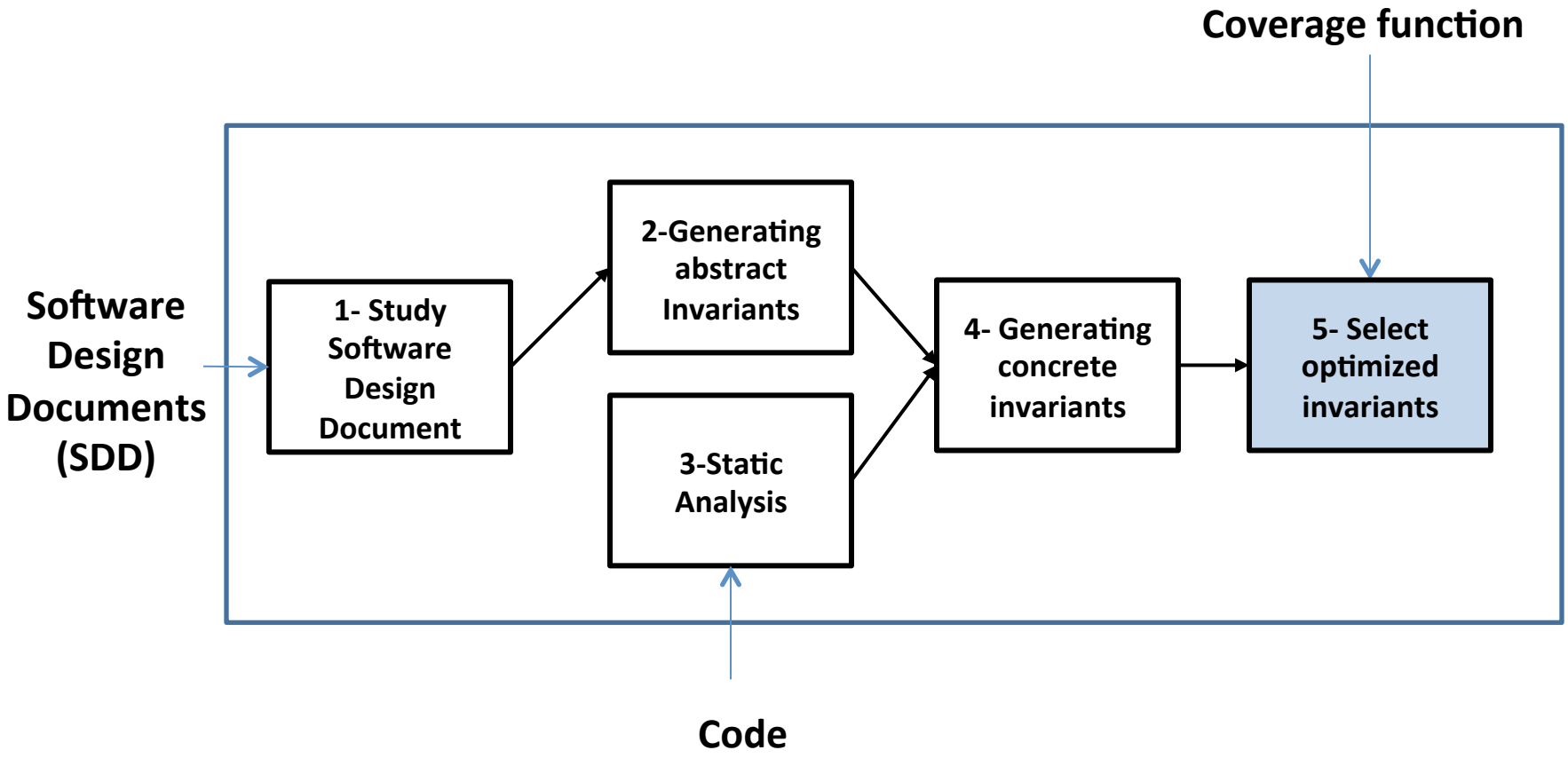
```
{
    ....
    write(f, data);
    ....
}
```

```
...
recv(4, 0x47cf68, 8192, 0)
...
write(1, 0x47cf68, 4) = 4
...
```

$$\square(getting\ sensorData(data) \Longrightarrow (\Diamond store\ on\ flash(data)))$$

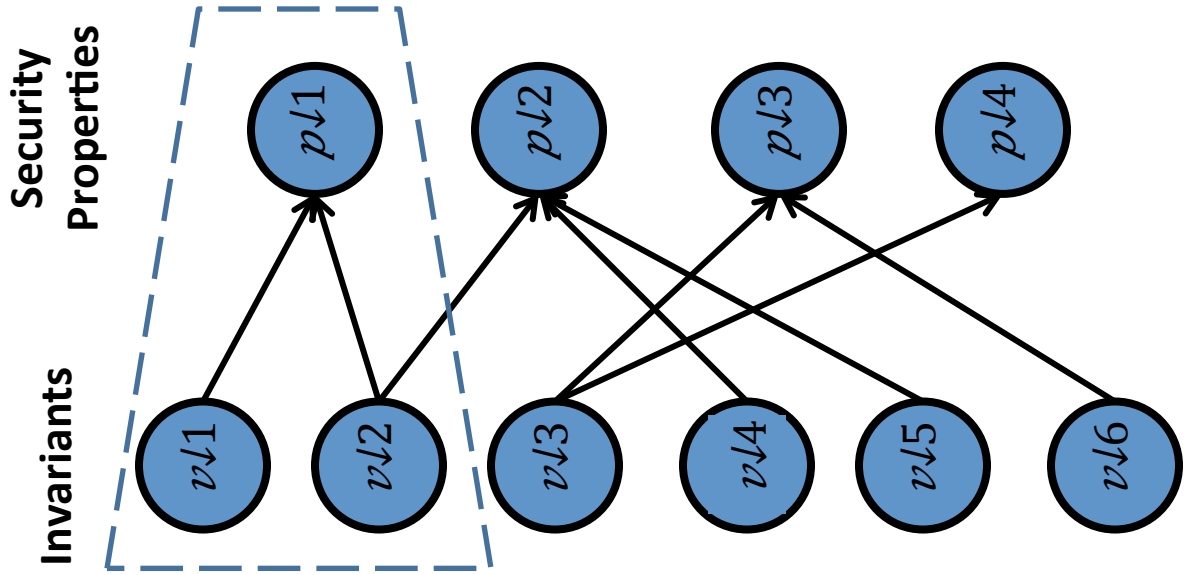
$$\square(receive(d) \Longrightarrow (\Diamond write(d)))$$

Step 5


Coverage function
Software Design Documents (SDD)
1- Study Software Design Document
2-Generating abstract Invariants
3-Static Analysis
4- Generating concrete invariants
5- Select optimized invariants
Code

Approach: Building the IDS

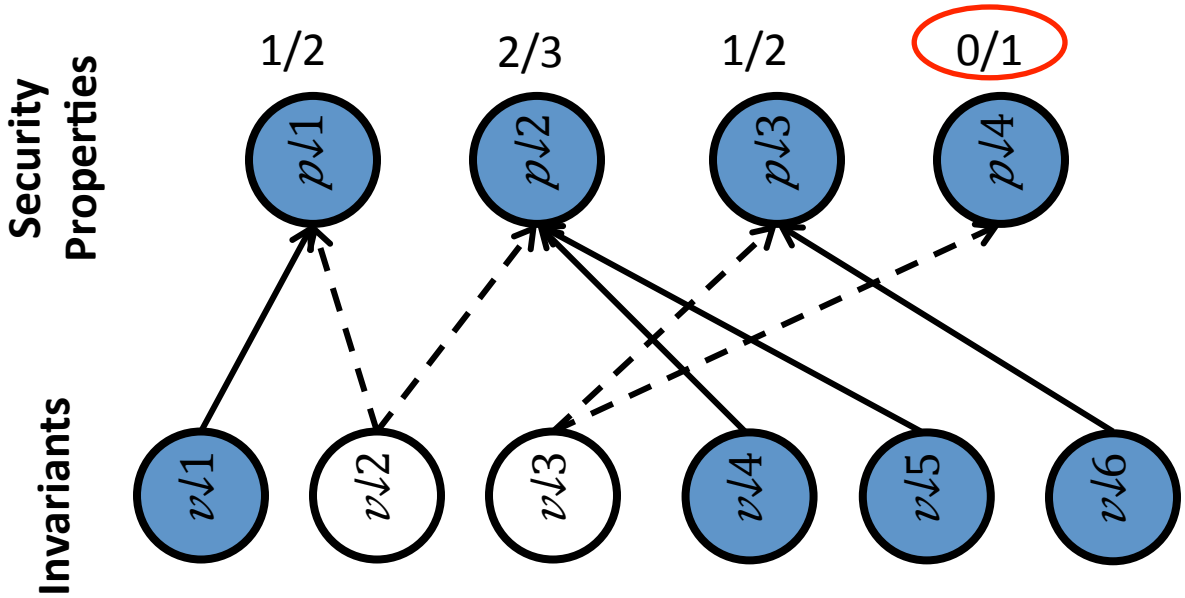
Define a coverage function on the graph and maximize it



Coverage Function: Example

MaxMin Coverage IDS:

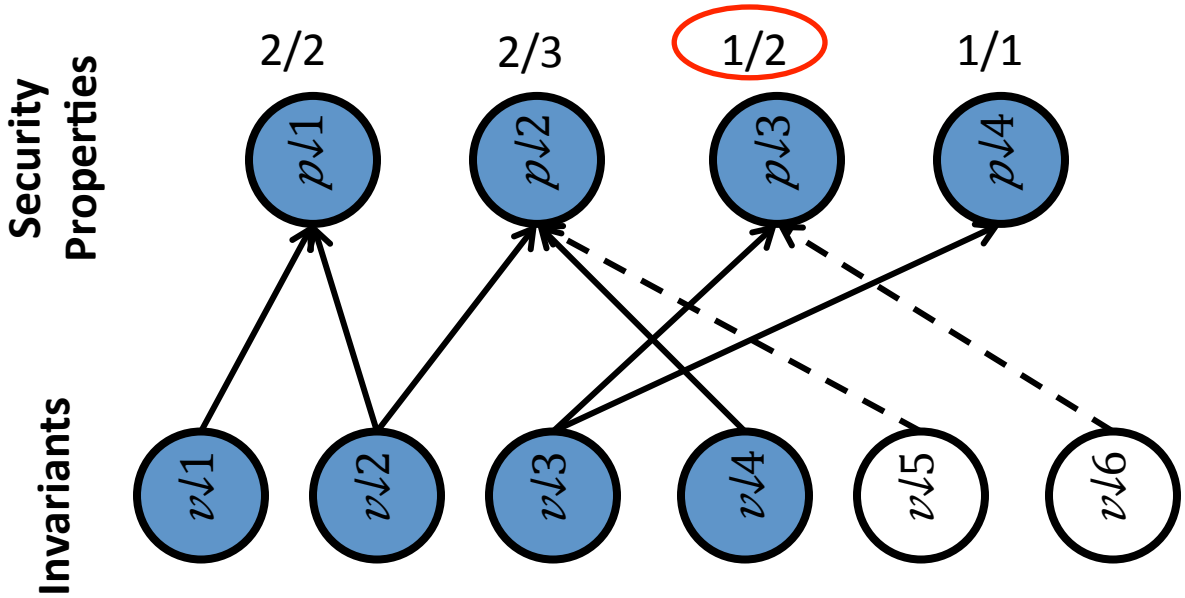
Intuition: Make the weakest coverage as strong as possible



Coverage Function: Example

MaxMin Coverage IDS:

Intuition: Make the weakest coverage as strong as possible



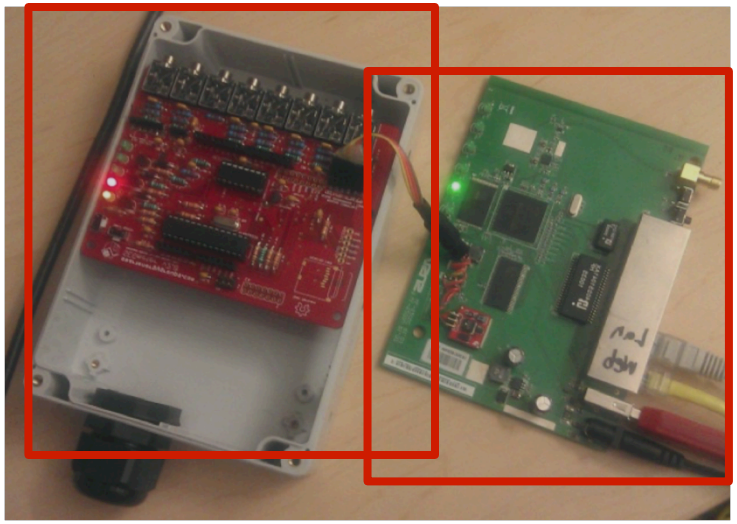
Building the IDS

Select the invariants from the graph

Automatically convert it to Buchi Automaton

Evaluation: Experimental Setup

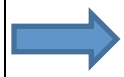
- Testbed: SEGMeter, open source smart meter from Australia
- Meter:
  - Arduino board
    - ATMEGA 32x series microcontroller
    - Sensors
  - Gateway board
    - Broadcom BCM 3302 240MHz CPU
    - 16 MB RAM
    - **4 MB available for IDS**
    - OpenWRT Linux
  - IDS runs on the Gateway board



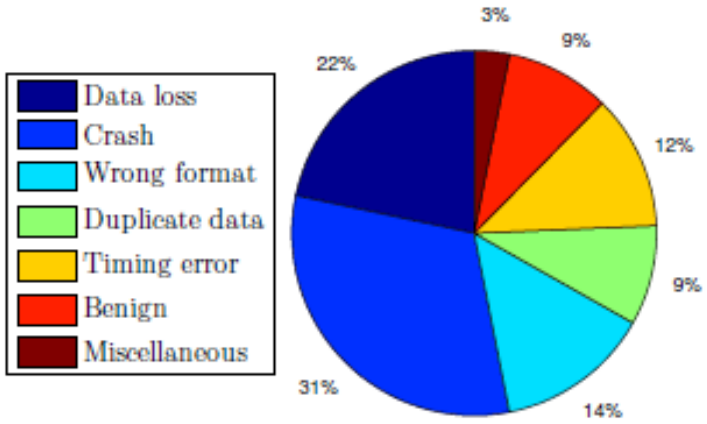
Evaluation: Fault injection

- Flipping branches

```
if (data_file ~= nil) then
   big_string = data_file:read("*all")
   …
end
```

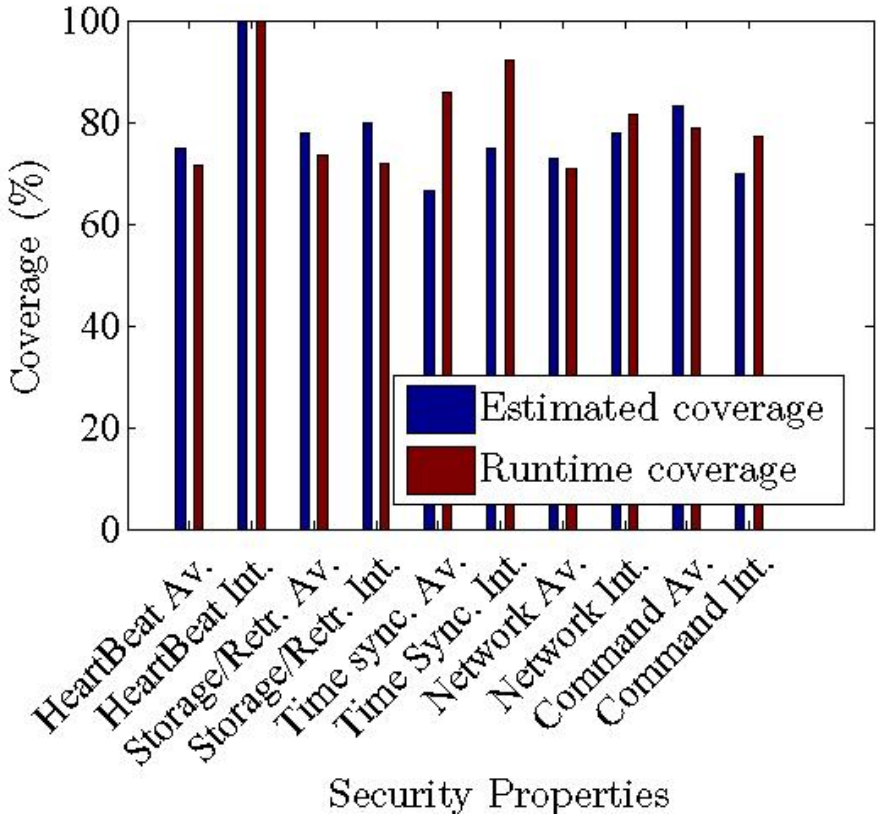


```
if (data_file == nil) then
   big_string = data_file:read("*all")
   …
end
```



Results (MaxMin IDS)

- How good is the coverage of the IDS?
- How good the graph-based optimization is reflected at run-time ?



Conclusions

- **We can quantify security** through coverage functions
- The security coverage formulated at design time is a good reflection of the security coverage at run-time
- IDSes are optimized for memory constrained embedded devices

For more info, please read:

Our EDCC'15 paper titled "Flexible Intrusion Detection Systems for Memory-Constrained Embedded Systems"
- Winner of distinguished paper award