Characterization of Error Tolerant Applications When Protecting Control Data

Darshan D. Thaker
John Oliver
Tzvetan S. Metodi

UC Davis

Derek Lockhart
Diana Franklin

CalPoly
San Luis Obispo

Susmit Biswas
Frederic T. Chong

UC Santa Barbara

Motivation

With decrease in feature size, soft errors on the rise.

Most research primarily at architectural level or at circuit level.

Understanding degradation of fidelity in an application, in presence of errors, is critical.

Use static analysis to protect instructions affecting control flow.

Outline

Overview of recent work.

Static analysis mechanism.

Fidelity measure of an application.

Experiments and results.

Concluding remarks.

Soft Errors in μ-Processors

Studies of architectural vulnerability factors.
(Biswas 05, Weaver 04, Mukherjee 03)

Circuit level mitigation techniques.
(Mitra 05, Iyer 05, Wang 04)

Concurrent thread execution.
(Smolens 05, Gomaa 03, Reinhardt 00)

Analyzed behavior of algorithms from stereo vision and speech recognition. (Vong 06)

Mpeg Example

Perceptual applications can tolerate soft errors.

Extreme example to make errors obvious.

Fidelity measure of an application can quantify degraded output.

Example: Mpeg 2 decoder.

Mpeg Original

Mpeg with Errors

Outline

Overview of recent work.

Static analysis mechanism.

Fidelity measure of an application.

Experiments and results.

Concluding remarks.

Methodology

Programmer identifies functions tolerant to errors.

Static Analysis: start at a branch and use Def-Use chain to identify instructions that effect the outcome of that branch.

Tag instructions that do not effect the branch.
    these instructions can be exposed to errors.

All other instructions are assumed to be protected.

Static Analysis Example

B.Blk 0:

```
.
.
I0:  $2  = $4 + 1
I1:  LD  $3, addr
I2:  $2  = $3 + 2
I3:  $3  = $3 + 8
I4:  $10 = $8 - $4
.
.
```

B.Blk 1:

```
I5:  $10 = $3 << $2
I6:  $4  = $3 + $6
I7:  $3  = $3 + 1
I8:  BNE  $3, $10, label
```

[$3, $10] ∈ CVAR

Static Analysis Example

B.Blk 0:

```
    .
    .
    I0:  $2  = $4 + 1
    I1:  LD  $3, addr
    I2:  $2  = $3 + 2
    I3:  $3  = $3 + 8
    I4:  $10 = $8 - $4
    .
    .
```

B.Blk 1:

```
    I5:  $10 = $3 << $2
    I6:  $4  = $3 + $6
    I7:  $3  = $3 + 1
    I8:  BNE  $3, $10, label
```

[$3, $10] ∈ CVAR

Static Analysis Example

B.Blk 0:

```
.
.
I0:  $2  = $4 + 1
I1:  LD  $3, addr
I2:  $2  = $3 + 2
I3:  $3  = $3 + 8
I4:  $10 = $8 - $4
.
.
```

B.Blk 1:

```
I5:  $10 = $3 << $2
I6:  $4  = $3 + $6
I7:  $3  = $3 + 1
I8:  BNE  $3, $10, label
```

[$3, $10] ∈ CVAR

Static Analysis Example

B.Blk 0:

```
.
.
I0:  $2  = $4 + 1
I1:  LD  $3, addr
I2:  $2  = $3 + 2
I3:  $3  = $3 + 8
I4:  $10 = $8 - $4
.
.
```

[$3, $2] ∈ CVAR

B.Blk 1:

```
I5:  $10 = $3 << $2
I6:  $4  = $3 + $6
I7:  $3  = $3 + 1
I8:  BNE  $3, $10, label
```

Static Analysis Example

B.Blk 0:

```
    .
    .
    I0:   $2  = $4 + 1
    I1:   LD  $3, addr
    I2:   $2  = $3 + 2
    I3:   $3  = $3 + 8
    I4:   $10 = $8 - $4
    .
    .
```

B.Blk 1:

```
    I5:   $10 = $3 << $2
    I6:   $4  = $3 + $6
    I7:   $3  = $3 + 1
    I8:   BNE  $3, $10, label
```

[ ] ∈ CVAR

Outline

Overview of recent work.

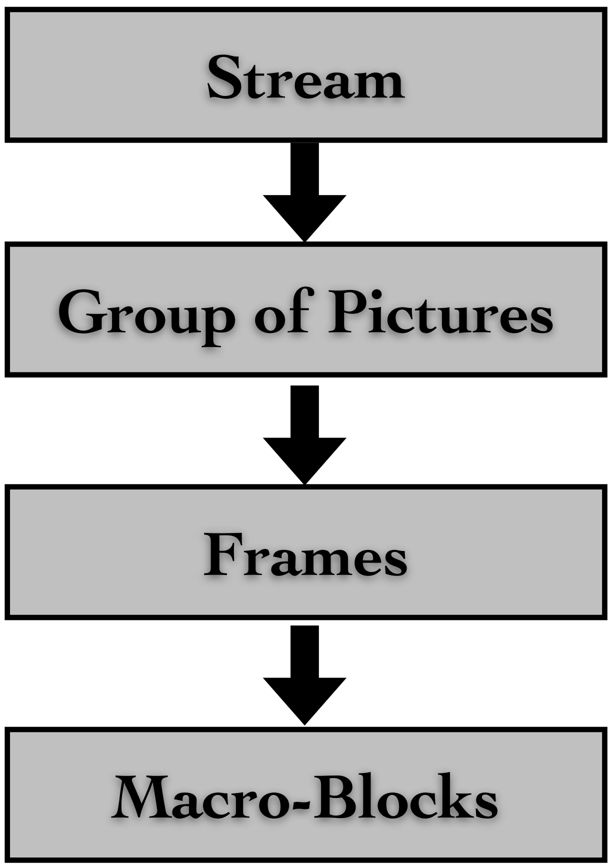
Static analysis mechanism.

Fidelity measure of an application.

Experiments and results.

Concluding remarks.

Fidelity: Mpeg



Compare SNR of every frame.

Difference > 2dB, implies bad frame.

Fidelity measure: number of bad frames.

Fidelity: Susan and MCF

Susan (MiBench):
Edge detection. Compare SNR of outputs.

MCF (SPEC 2000int):
Generates optimal schedules.
Fidelity measure: Distance from optimal schedule.

Art (SPEC 2000int):
Neural net to identify objects in an image.
Fidelity measure: % objects identified correctly.

Fidelity: Blowfish and GSM

Blowfish (MiBench):
Encryption - Decryption algorithm.
Fidelity measure: % matching bytes post-decryption.

GSM (MiBench):
Audio compression scheme.
Fidelity measure: SNR of output.

Outline

Overview of recent work.

Static analysis mechanism.

Fidelity measure of an application.

Experiments and results.

Concluding remarks.

Error Insertion

Errors randomly inserted as single bit-flips.

Every error alters the result of an ALU operation.

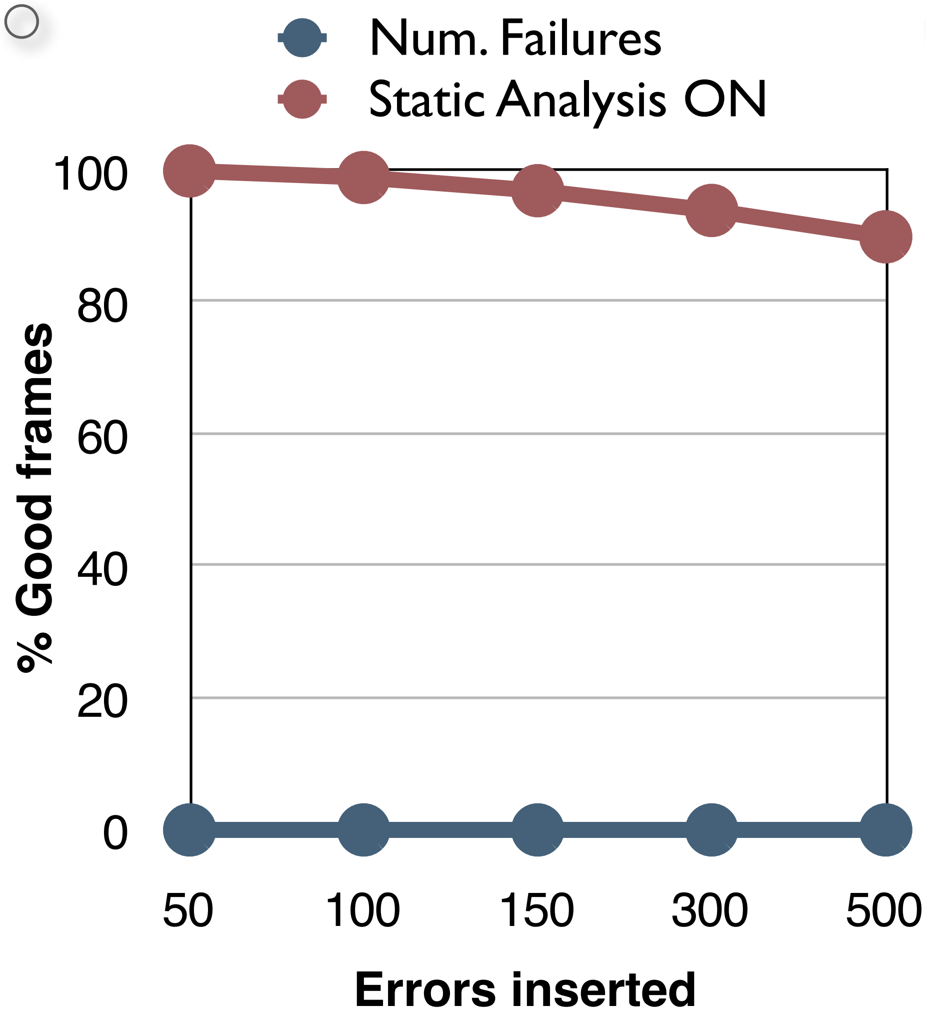
Errors inserted only in tagged instructions.

Used SimpleScalar for our simulations and for error insertions.

Benefits of Static Analysis

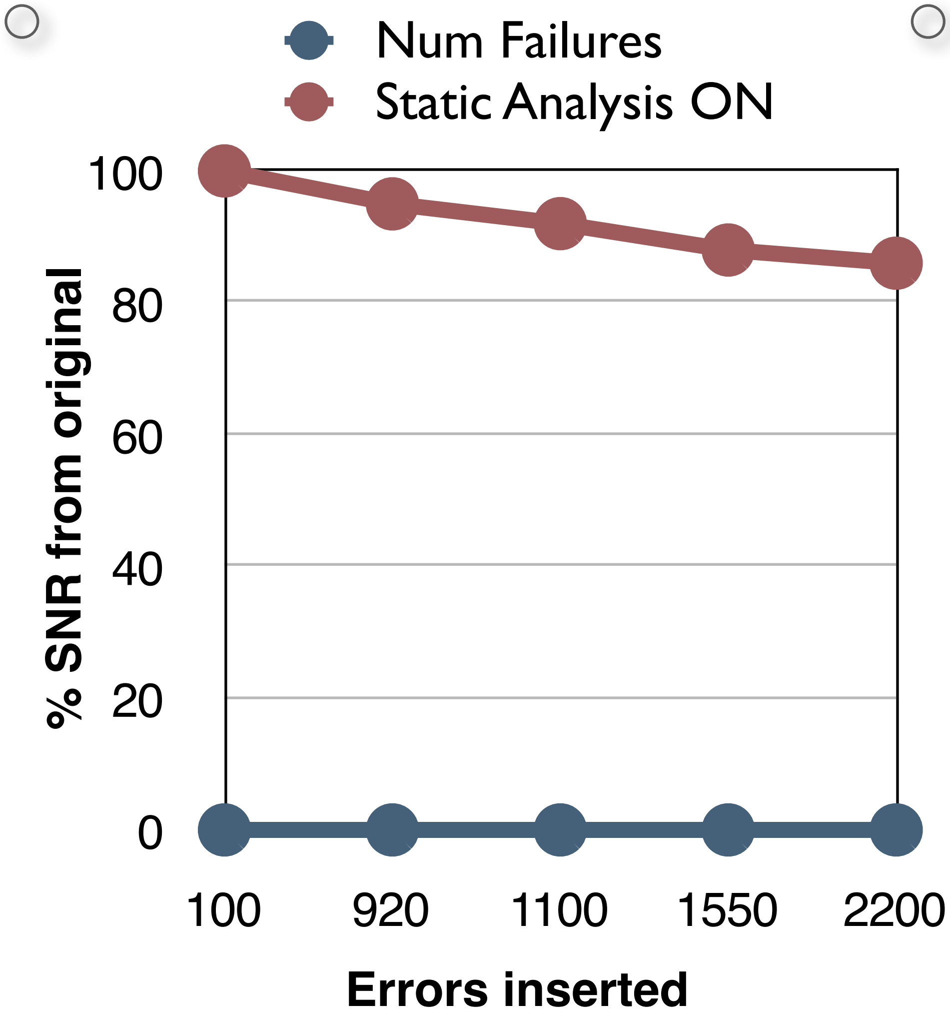
Application	Errors inserted	Total instructions (millions)	Failures with static analysis	Failures without static analysis
Susan	2200	144	0 %	10 %
Mcf	320	201	6 %	100 %
Mpeg	120	2740	0 %	100 %
GSM	40	892	0 %	100 %
Blowfish	20	507	19 %	48 %
Art	4	42770	0 %	0 %

Results: Mpeg and Susan



Mpeg

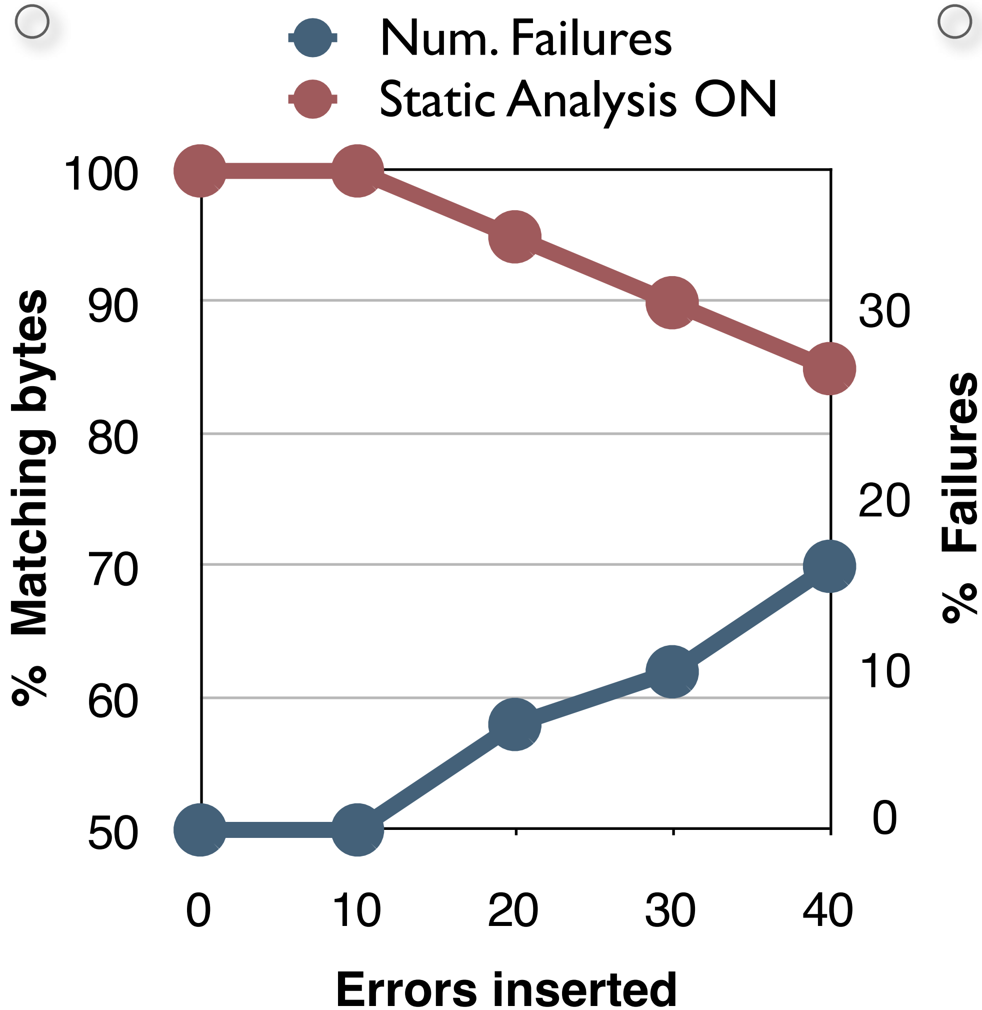
(0.80 billion cycles)



Susan

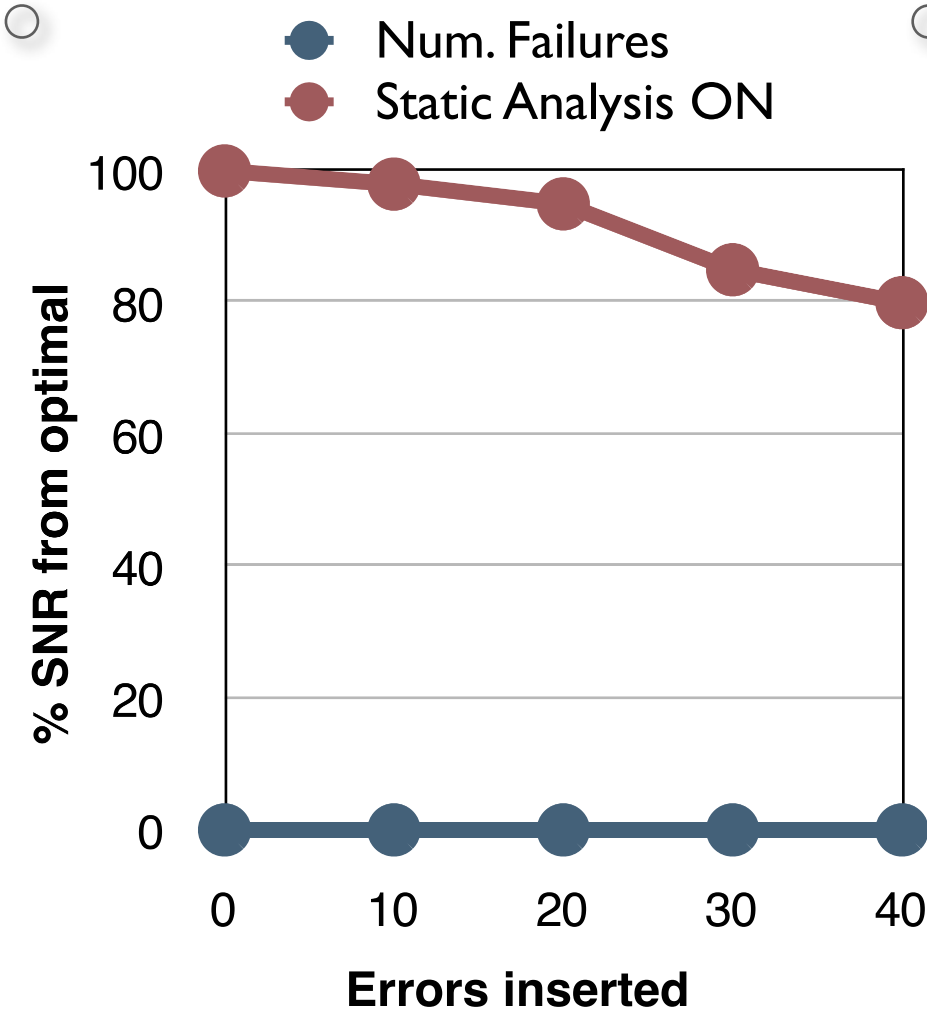
(0.51 billion cycles)

Results: Blowfish and GSM



Blowfish

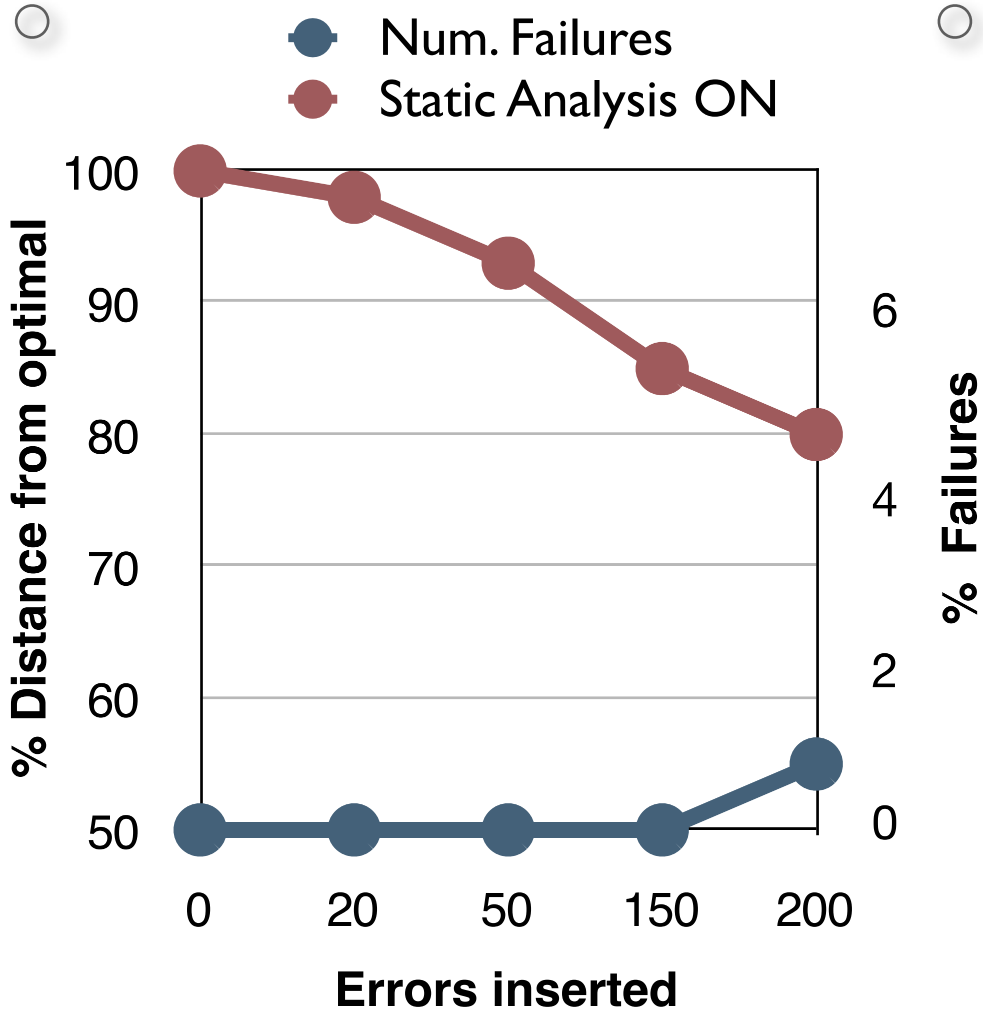
(0.18 billion cycles)



GSM

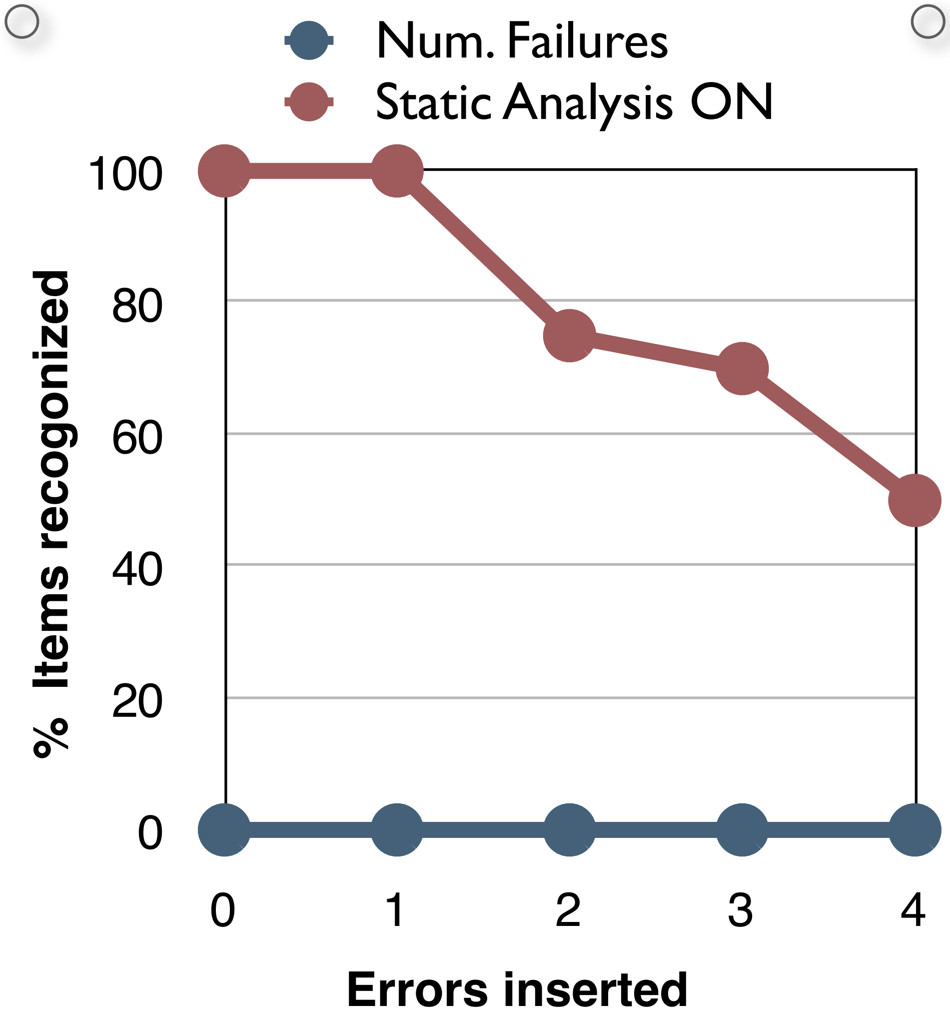
(0.38 billion cycles)

Results: Mcf and Art



Mcf

(0.3 billion cycles)



Art

(15.27 billion cycles)

Outline

Overview of recent work.

Static analysis mechanism.

Fidelity measure of an application.

Experiments and results.

Concluding remarks.

The Bottom Line

Certain applications have an inherent ability to tolerate soft errors.

Simple static analysis protects code especially susceptible to errors.

Loss of fidelity, when errors inserted, within acceptable limits.

Instructions Tagged

Application	Errors inserted	Total instructions (millions)	Tagged instructions	Failures with static analysis	Failures without static analysis
Susan	2200	144	91.3%	0 %	10 %
Mcf	320	201	8.9%	6 %	100 %
Mpeg	120	2740	50.3%	0 %	100 %
GSM	40	892	19.6%	0 %	100 %
Blowfish	20	507	62.4%	19 %	48 %
Art	4	4277	70.8%	0 %	0 %

Future Work

Differentiate between data, pointer and stack variables during static analysis.

Feedback-directed optimizations driven by programmer-defined fidelity measures.

Architectural mechanisms to protect non-tagged instructions.

¿ Questions ?

Thank you!

Your questions...

Error Rate

Application	Errors inserted	Total instructions (millions)	Tagged instructions	Error Rate
Susan	2200	144	91.3%	1.6 e-5
Mcf	320	201	8.9%	1.7 e-5
GSM	40	892	19.6%	2.28 e-7
Blowfish	20	507	62.4%	6.32 e-8
Mpeg	120	2740	50.3%	8.7 e-8
Art	4	4277	70.8%	1.3 e-9