

Fault? Eh...keeping going!

Michael Carbin

MIT CSAIL

Approximate Computing: Examples

- **Approximate Hardware**

PCMOS, Palem et al. 2005; Narayanan et al., DATE '10; Liu et al. ASPLOS '11; Sampson et al, PLDI '11; Esmaeilzadeh et al. , ASPLOS '12, MICRO' 12

- **Function Substitution**

Hoffman et al., APLOS '11; Ansel et al., CGO '11; Zhu et al., POPL '12

- **Approximate Memoization**

Alvarez et al., IEEE TOC '05; Chaudhuri et al., FSE '12; Samadi et al., ASPLOS '14

- **Relaxed Synchronization (Lock Elision)**

Renganarayana et al., RACES '12; Rinard, HotPar '13; Misailovic, et al., RACES '12

- **Code Perforation**

Rinard, ICS '06; Baek et al., PLDI 10; Misailovic et al., ICSE '10; Sidiroglou et al., FSE '11; Misailovic et al., SAS '11; Zhu et al., POPL '12; Carbin et al. PEPM '13;

Loop Perforation

```
for (uint i = 0; i < n; ++i) {...}
```



```
for (uint i = 0; i < n/2; ++i) {...}
```

Loop Perforation Results

(ICSE '10, ASPLOS '11, FSE '11, PEPM '13)

Applications (Parsec) Framework

Media Processing

- Developer specifies maximum acceptable error using metric

Computer Vision

- Automatically identifies loops perforations with acceptable error

Machine Learning

Performance improvement

Search

- Typically over a factor of two
- Up to a factor of seven

Finance

Quality Impact

- < 10% change in output

Community Goals

- Exploit Uncertainty
 - Performance (e.g., less work, increased parallelism)
 - Energy (e.g., less work, efficient approximate hardware)
 - Resilience (e.g., elide/approximate failed tasks)
- Provide Guarantees
 - Safety (e.g., memory safety) (PEPM '13)
 - Worst-case error bounds (PLDI '12, OOPSLA '13)
 - Probabilistic error bounds (SAS '11, POPL '12)

Panel Questions

- 1) What other levels of the stack should resilience depend on?
All of them. Uncertainty will be a first class concern in future systems.
- 2) How can resilience schemes best exploit application, runtime, or programming model semantics?
Developers expose application-specific flexibility via programming model. Runtime has different capabilities. Resilience schemes coordinate.
- 3) What are the biggest missing pieces needed from the various layers to make resilience schemes succeed?
Coordinated understanding of uncertainty across stack (UQ?). We've explored uncertainty/approximation only in limited scopes.
- 4) What is the impact on resilience of the wide range of expected operating scenarios?
Not obvious. State-of-the art: explore a variety reasoning approaches and mechanisms. Open problem: balancing complexity and benefit