Spectrum-based Diagnosis for Run-time Systems

Nuno Cardoso

nunopcardoso@gmail.com

Supervisor : Professor Rui Abreu

Co-Supervisor : Professor David Garlan

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

HASLab INESCTEC

27/04/2016

Introduction

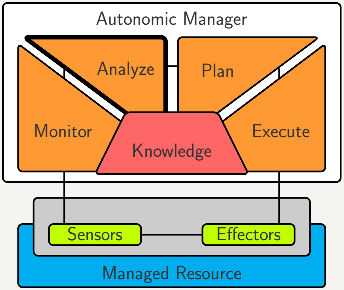
- Context
- Spectrum-based Fault Localization
- Candidate Generation
- Candidate Ranking

Context

- Despite the advances made in the Computer Science domain, it remains practically impossible to create faultless systems

Context

- Research in Self-healing Systems aims at:
  1. Improving the systems' dependability
  2. Reducing human intervention



Context

- A diagnostic problem occurs whenever the behavior of a particular system deviates from the expected behavior
- The challenge is to find the true root causes of the abnormal behavior

Spectrum-based Fault Localization

Component – An indivisible[1] element of the system

Transaction – A set of component activations that:

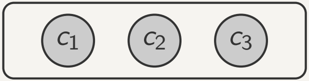
1. Share a common goal
2. The correctness of the output can be verified

[1]From a diagnosis point-of-view

Spectrum-based Fault Localization

Spectrum – A pair (A, e) encoding:

- A_{ij} – Activity of component j in transaction i
- e_i – Error state of transaction i (pass/fail)



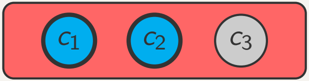
i	A_{ij}			e_i
	c_1	c_2	c_3	
1	1	1	0	1
2	1	0	1	1
3	1	0	0	0

Spectrum-based Fault Localization

Spectrum – A pair (A, e) encoding:

A_{ij} – Activity of component j in transaction i

e_i – Error state of transaction i (pass/fail)

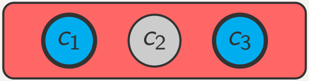


i	A_{ij}			e_i
	c_1	c_2	c_3	
1	1	1	0	1
2	1	0	1	1
3	1	0	0	0

Spectrum-based Fault Localization

Spectrum – A pair (A, e) encoding:

- A_{ij} – Activity of component j in transaction i
- e_i – Error state of transaction i (pass/fail)

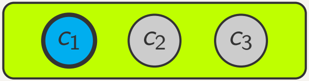


i	A_{ij}			e_i
	c_1	c_2	c_3	
1	1	1	0	1
2	1	0	1	1
3	1	0	0	0

Spectrum-based Fault Localization

Spectrum – A pair (A, e) encoding:

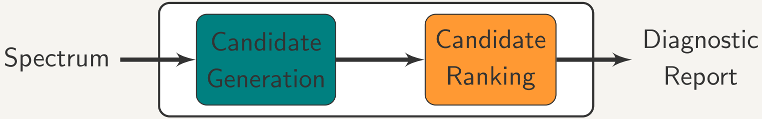
- A_{ij} – Activity of component j in transaction i
- e_i – Error state of transaction i (pass/fail)



i	A_{ij}			e_i
	c_1	c_2	c_3	
1	1	1	0	1
2	1	0	1	1
3	1	0	0	0

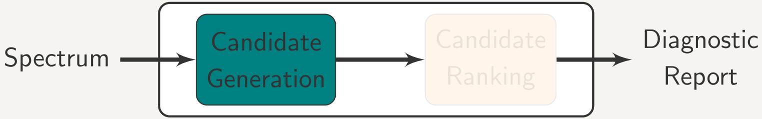
Spectrum-based Fault Localization

Workflow



1. Generate possible explanations for the erroneous behavior
2. Minimize the number of unnecessarily inspected components

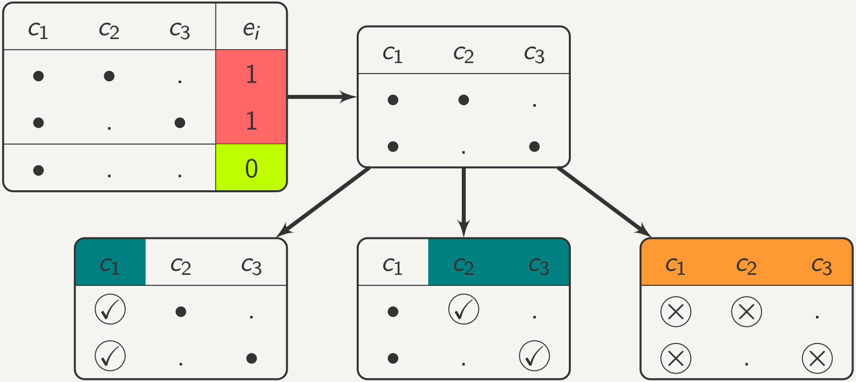
Candidate Generation



Candidate Every failed transaction activates at least one faulty component

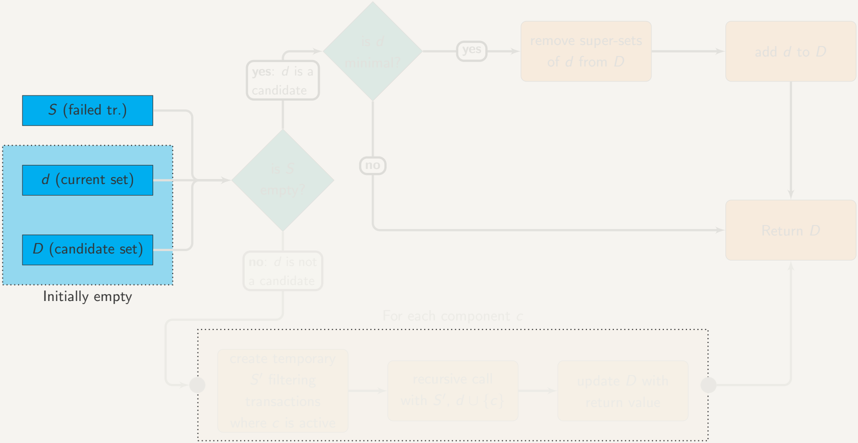
Minimal No smaller subset of components meets the above criterion

Candidate Generation



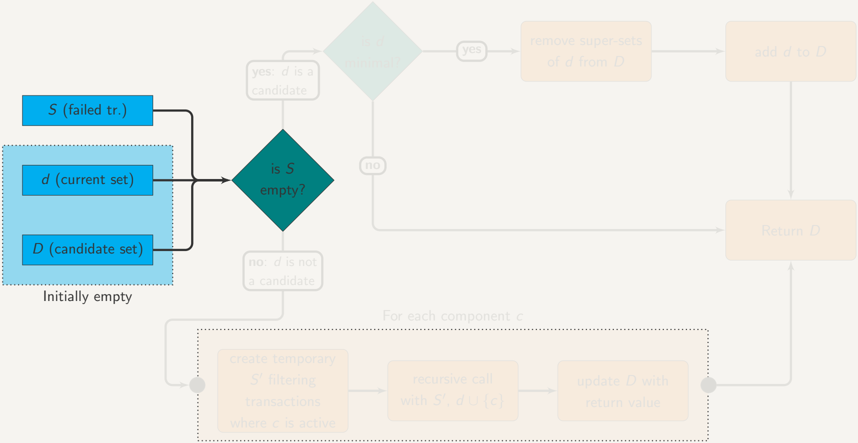
Candidate Generation

Algorithm



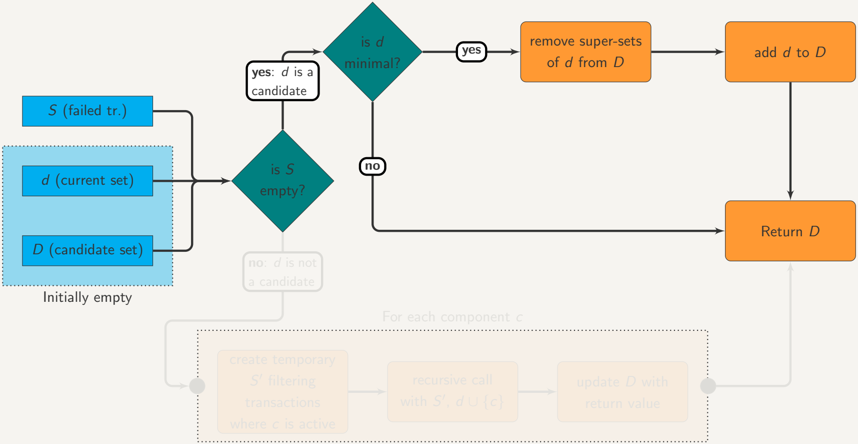
Candidate Generation

Algorithm



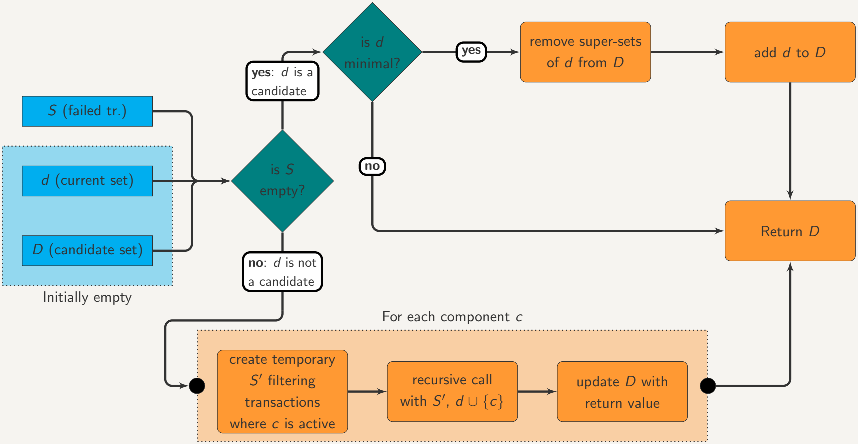
Candidate Generation

Algorithm



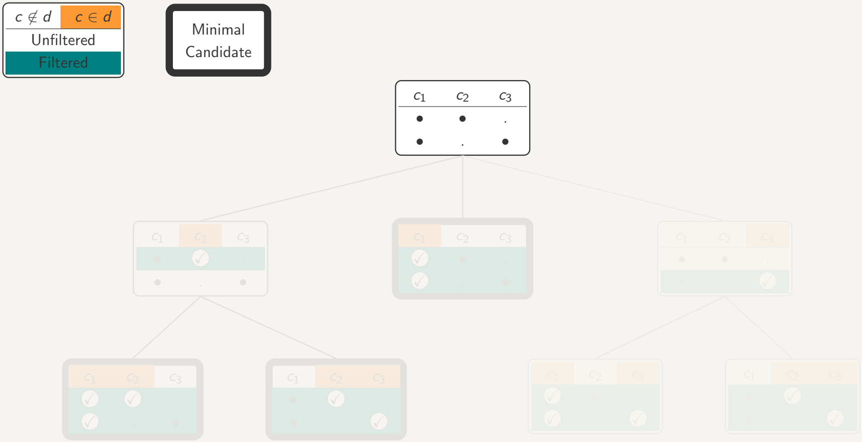
Candidate Generation

Algorithm



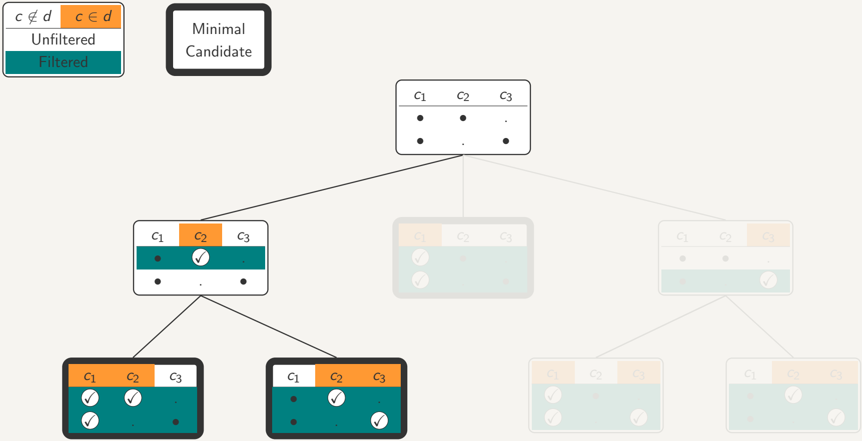
Candidate Generation

Example



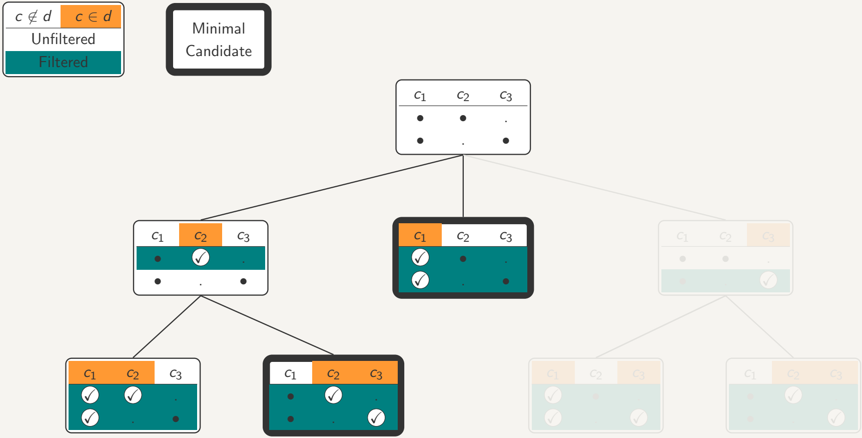
Candidate Generation

Example



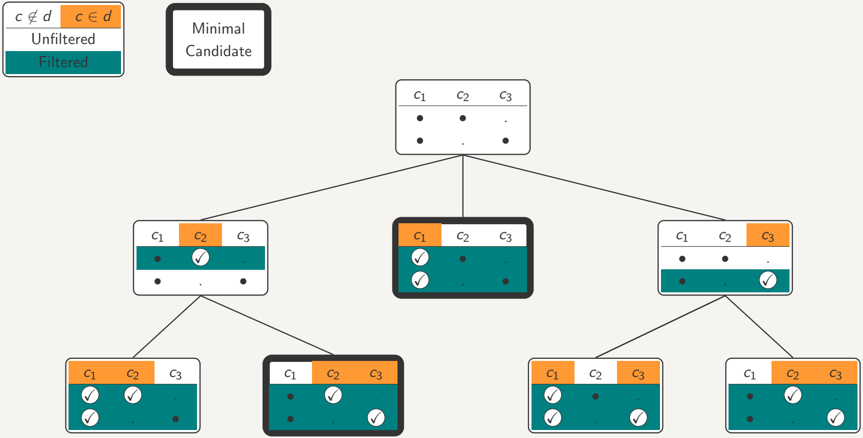
Candidate Generation

Example

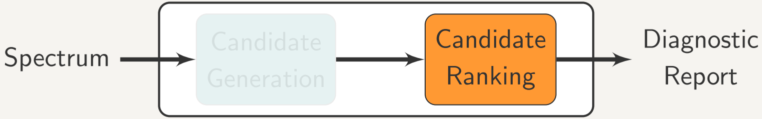


Candidate Generation

Example



Candidate Ranking

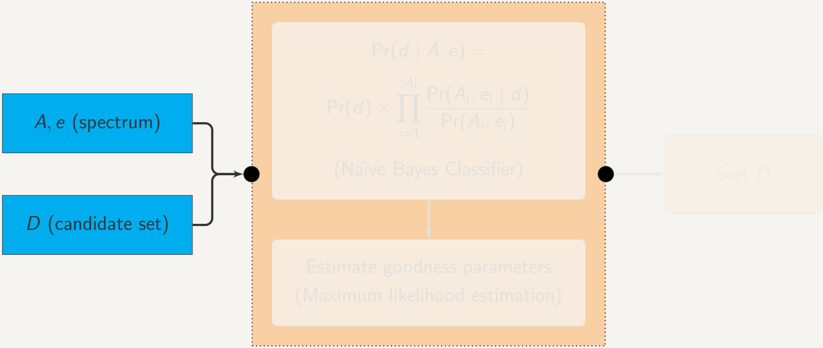


Intermittent Fault A fault that is not consistently triggered when activated

Candidate Ranking

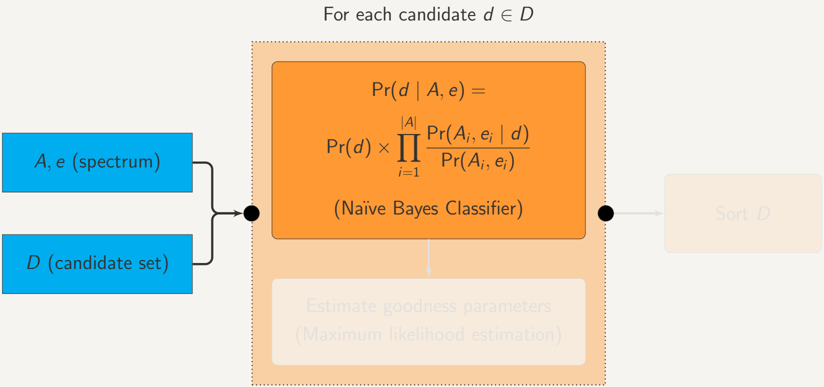
Algorithm

For each candidate $d \in D$



Candidate Ranking

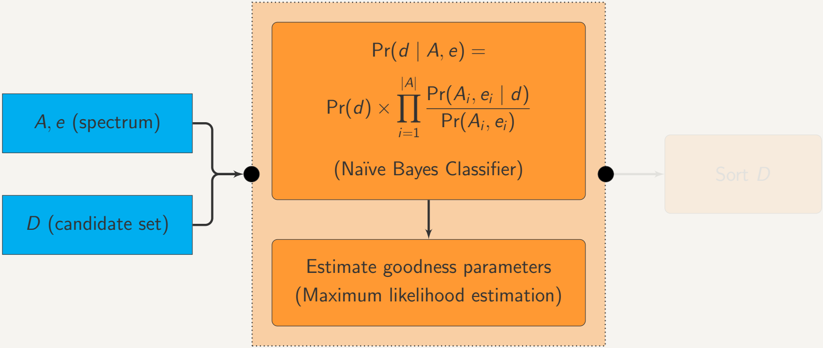
Algorithm



Candidate Ranking

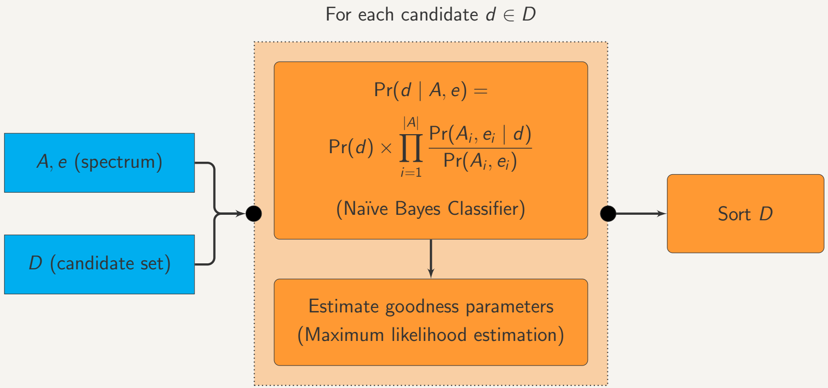
Algorithm

For each candidate $d \in D$



Candidate Ranking

Algorithm



Candidate Ranking

Example

i	A_{ij}			e_i
	c_1	c_2	c_3	
1	1	1	0	1
2	1	0	1	1
3	1	0	0	0

$$D = \Big\{ \{c_1\}, \{c_2, c_3\} \Big\}$$

$$\Pr(\{c_1\} \mid A, e) = \overbrace{\frac{1}{1000} \cdot \frac{999}{1000} \cdot \frac{999}{1000}}^{\Pr(d)} \times \overbrace{\underbrace{(1 - g_1)}_{t_1} \cdot \underbrace{(1 - g_1)}_{t_2} \cdot \underbrace{g_1}_{t_3}}^{\Pr(A, e \mid d)}$$

$$\Pr(\{c_2, c_3\} \mid A, e) = \cdots$$

$$\cdots$$

$$\Pr(\{c_1\} \mid A, e) = 0.993$$
$$\Pr(\{c_2, c_3\} \mid A, e) = 0.007$$

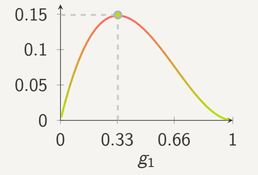
Candidate Ranking

Example

i	A_{ij}			e_i
	c_1	c_2	c_3	
1	1	1	0	1
2	1	0	1	1
3	1	0	0	0

$$D = \left\{ \{c_1\}, \{c_2, c_3\} \right\}$$

$$\Pr(\{c_1\} \mid A, e) = \overbrace{\frac{1}{1000} \cdot \frac{999}{1000} \cdot \frac{999}{1000}}^{\Pr(d)} \times \overbrace{\underbrace{(1 - g_1)}_{t_1} \cdot \underbrace{(1 - g_1)}_{t_2} \cdot \underbrace{g_1}_{t_3}}^{\Pr(A,e|d)}$$



$\Pr(\{c_2, c_3\} \mid A, e) = \cdots$

$\cdots$

$\Pr(\{c_1\} \mid A, e) = 0.993$

$\Pr(\{c_2, c_3\} \mid A, e) = 0.007$

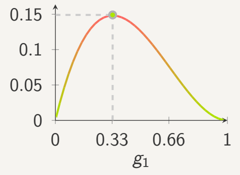
Candidate Ranking

Example

i	A_{ij}			e_i
	c_1	c_2	c_3	
1	1	1	0	1
2	1	0	1	1
3	1	0	0	0

$$D = \left\{ \{c_1\}, \{c_2, c_3\} \right\}$$

$$\Pr(\{c_1\} \mid A, e) = \overbrace{\frac{1}{1000} \cdot \frac{999}{1000} \cdot \frac{999}{1000}}^{\Pr(d)} \times \overbrace{\underbrace{(1-g_1)}_{t_1} \cdot \underbrace{(1-g_1)}_{t_2} \cdot \underbrace{g_1}_{t_3}}^{\Pr(A,e|d)}$$



$$\Pr(\{c_2, c_3\} \mid A, e) = \cdots$$

$\cdots$

$$\Pr(\{c_1\} \mid A, e) = 0.993$$
$$\Pr(\{c_2, c_3\} \mid A, e) = 0.007$$

MHS^2

- Motivation
- Optimizations
- Parallelization

Introduction

- At run-time, diagnosis is a time-critical task
- The candidate generation problem is NP-Hard

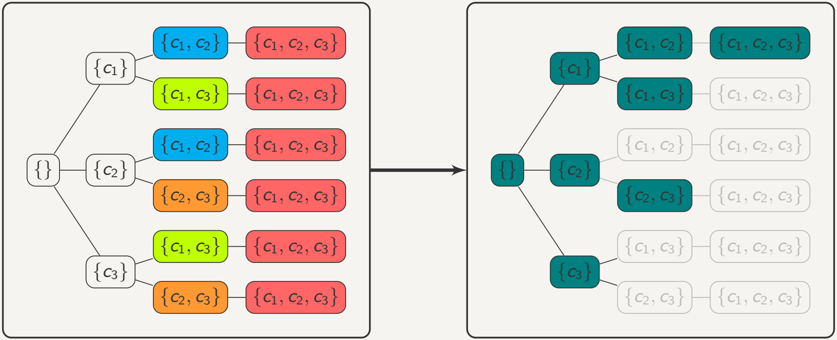
Introduction

Goals

- Compute the same # of candidates in a shorter period of time:
  1. Improve the algorithm's efficiency
  2. Enable parallel processing

Optimizations

Optimization 1



Optimizations

Optimization 2

c_1	c_2	c_3	c_4	e_i
•	•	.	.	1
•	.	•	.	1
•	.	.	•	0

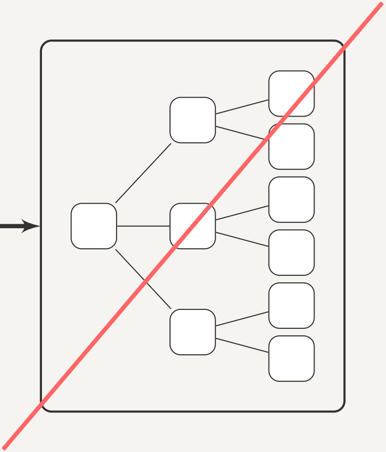
$\longrightarrow$

c_1	c_2	c_3	e_i
•	•	.	1
•	.	•	1
•	.	.	0

Optimizations

Optimization 3

c_1	c_2	c_3	e_i
●	●	.	1
●	.	●	1
.	.	.	1
●	.	.	0



Optimizations – Benchmark

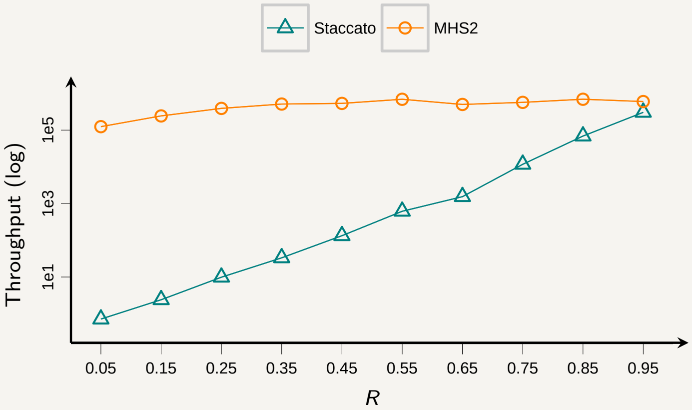
- A generated by a Bernoulli process with parameters M, N, and R:

	M	e_i
		1
N	$\Pr(A_{ij} = 1) = R$	1
		1

Throughput Number of candidates generated per second

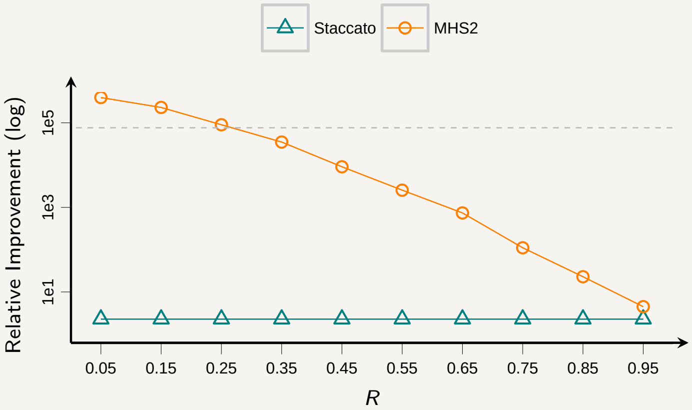
Optimizations – Benchmark

$M, N = 10$



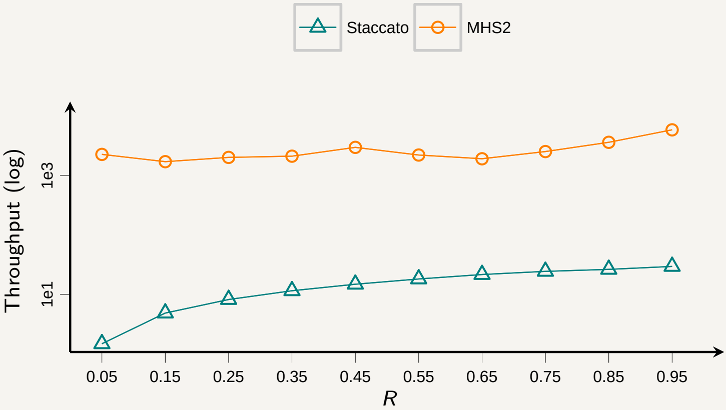
Optimizations – Benchmark

$M, N = 10$



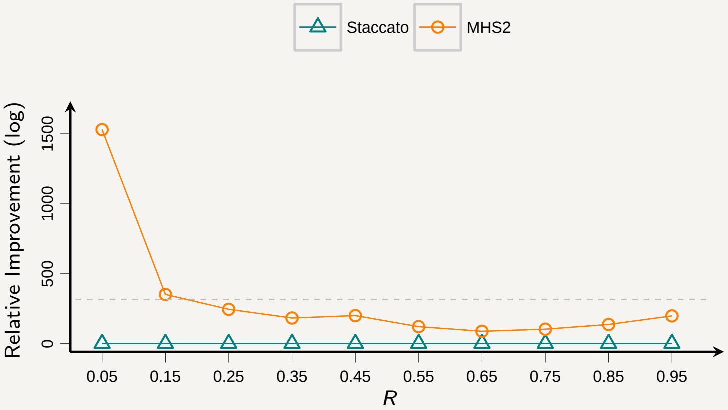
Optimizations – Benchmark

$M, N = 10000$



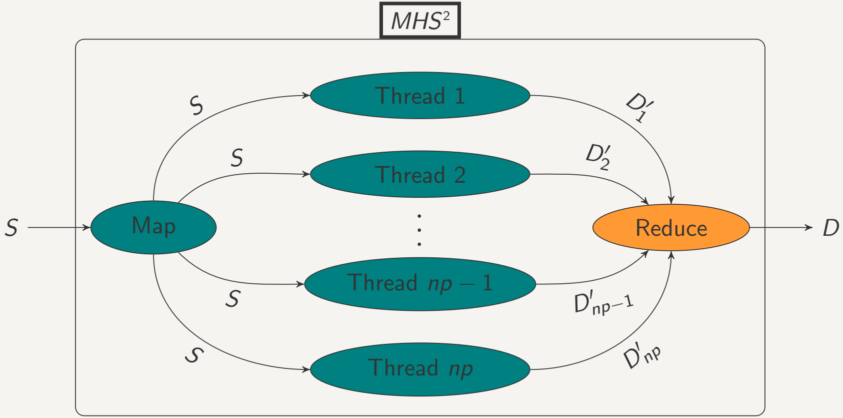
Optimizations – Benchmark

$M, N = 10000$



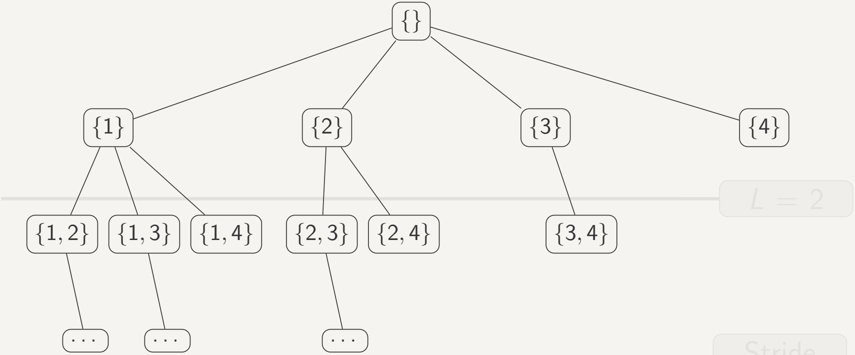
Parallelization

Map-Reduce



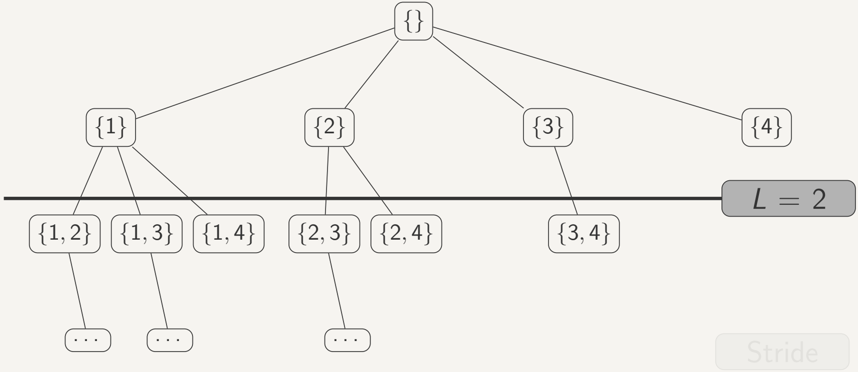
Parallelization

Load Division



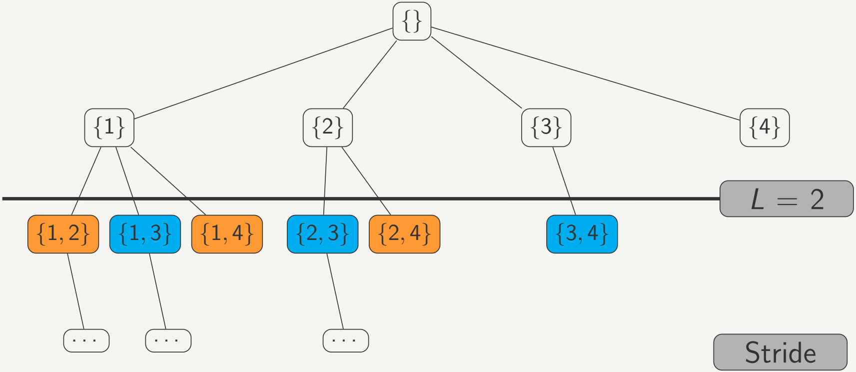
Parallelization

Load Division



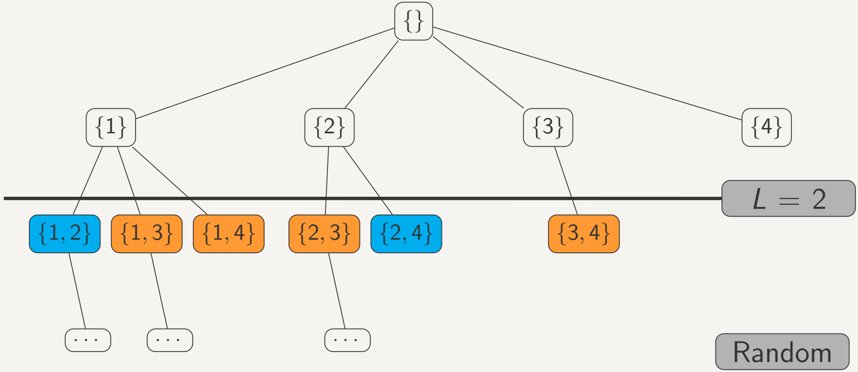
Parallelization

Load Division



Parallelization

Load Division



Parallelization – Benchmark

- A generated by a Bernoulli process with parameters M, N, and R:

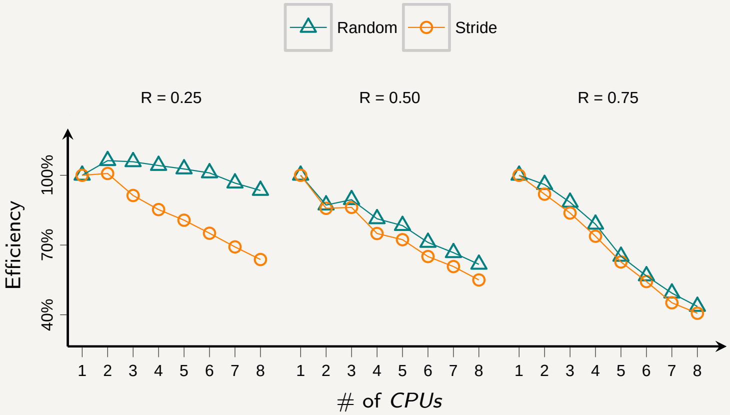
	M	e_i
N	$\Pr(A_{ij} = 1) = R$	1 1 1

Efficiency

$$E_{np} = \frac{T_1}{T_{np} \times np}$$

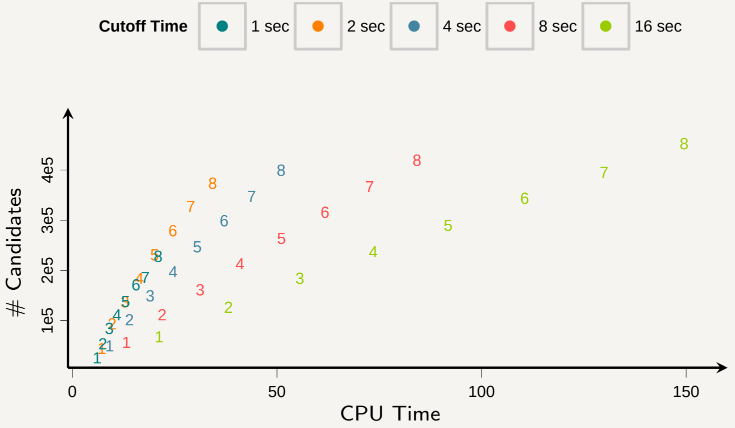
Parallelization – Benchmark

$M, N = 40$



Parallelization – Benchmark

$M, N = 10000$



Fuzzinel

- Motivation
- Approach
- Benchmark

Introduction

The distinction between "healthy" and "broken" is often indistinct and fuzzy, and often there is a gradual transition between these two states; a system often does not break down recognizably but deteriorates over time.

Ghosh, D., Sharman, R., Rao, H. R., and Upadhyaya, S. (2007).
Self-healing systems - survey and synthesis.

Introduction

- ✓ The classic pass/fail model correctly encodes functional errors
- × The pass/fail model cannot encode the symptoms associated with performance errors (e.g., slowness)

Approach

Error Encoding

$$e = \begin{cases} 1 & \longrightarrow \text{Fail} \\ 0 & \longrightarrow \text{Pass} \end{cases}$$

$$e = \begin{cases} 1 & \longrightarrow \text{Fail} \\ 0 & \longrightarrow \text{Pass} \\ ]0;1[& \longrightarrow \text{Sub-optimal performance} \end{cases}$$

Approach

Error Encoding – Example

✓ The throughput of a data storage system (tp) must be greater 1 GB/s

✗ Between 1GB/s and 2GB/s the performance is sub-optimal

$$e_{crisp}(tp) = \begin{cases} 1 & \text{if } tp < 1 \\ 0 & \text{if } tp \geq 1 \end{cases}$$

$$e_{fuzzy}(tp) = \begin{cases} 1 & \text{if } tp < 1 \\ 0 & \text{if } tp \geq 2 \\ 2 - (tp - 1) & \text{if } 1 \leq tp < 2 \end{cases}$$

Approach

Error Encoding – Example

✓ The throughput of a data storage system (tp) must be greater 1 GB/s

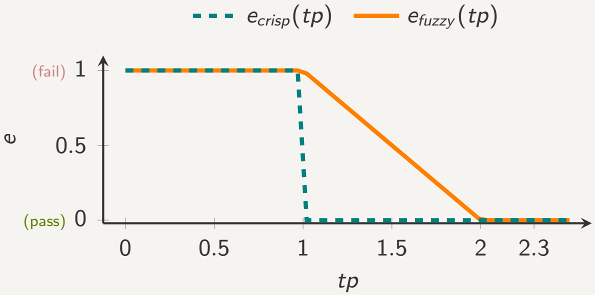
✓ Between 1GB/s and 2GB/s the performance is sub-optimal

$$e_{crisp}(tp) = \begin{cases} 1 & \text{if } tp < 1 \\ 0 & \text{if } tp \geq 1 \end{cases}$$

$$e_{fuzzy}(tp) = \begin{cases} 1 & \text{if } tp < 1 \\ 0 & \text{if } tp \geq 2 \\ 2 - (tp - 1) & \text{if } 1 \leq tp < 2 \end{cases}$$

Approach

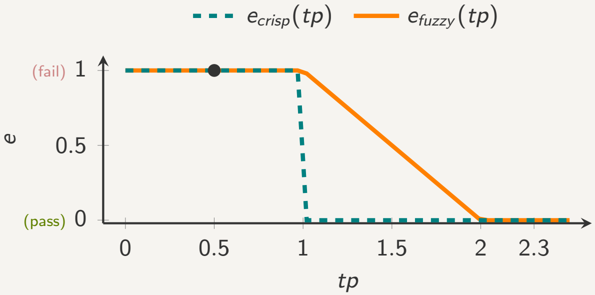
Error Encoding – Example



i	tp	e_{crisp}	e_{fuzzy}
1	0.5	1	1
2	2.3	0	0
3	1.5	0	0.5

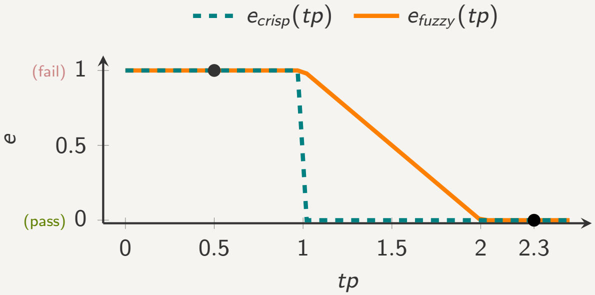
Approach

Error Encoding – Example



i	tp	e_{crisp}	e_{fuzzy}
1	0.5	1	1
2	2.3	0	0
3	1.5	0	0.5

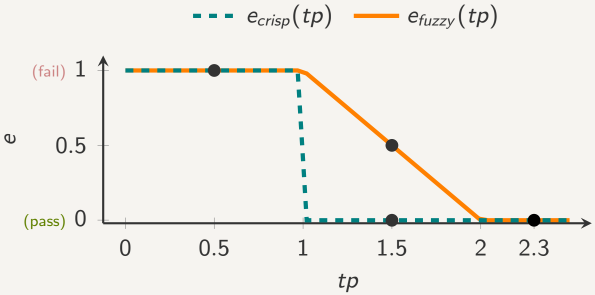
Approach



i	tp	e_{crisp}	e_{fuzzy}
1	0.5	1	1
2	2.3	0	0
3	1.5	0	0.5

Approach

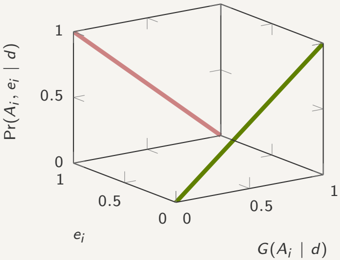
Error Encoding – Example



i	tp	e_{crisp}	e_{fuzzy}
1	0.5	1	1
2	2.3	0	0
3	1.5	0	0.5

Approach

$$\Pr(A_i, e_i \mid d) = \begin{cases} 1 - G(A_i \mid d) & \text{if } e_i = 1 \\ G(A_i \mid d) & \text{if } e_i = 0 \end{cases}$$

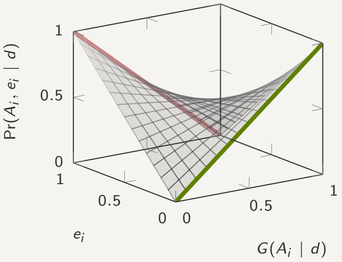


$$\Pr(A_i, e_i \mid d) = (1 - G(A_i \mid d)) \times e_i + G(A_i \mid d) \times (1 - e_i)$$

Approach

Error Diagnostic

$$\Pr(A_i, e_i \mid d) = \begin{cases} 1 - G(A_i \mid d) & \text{if } e_i = 1 \\ G(A_i \mid d) & \text{if } e_i = 0 \end{cases}$$

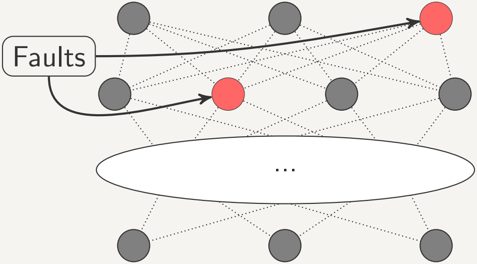


$$\Pr(A_i, e_i \mid d) = (1 - G(A_i \mid d)) \times e_i + G(A_i \mid d) \times (1 - e_i)$$

Benchmark

Setup

- (A, e) generated using a simulator[1]

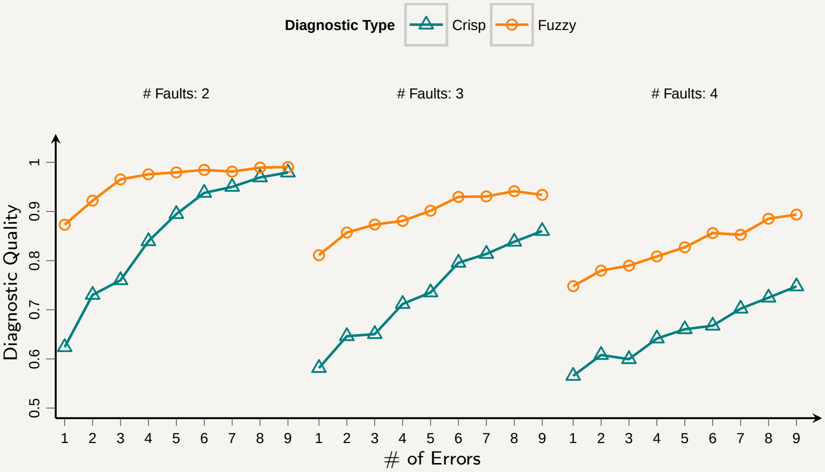


Diagnostic Quality Percentage of components that did not need to be inspected for the fault to be found

[1] `https://github.com/SERG-Delft/sfl-simulator`

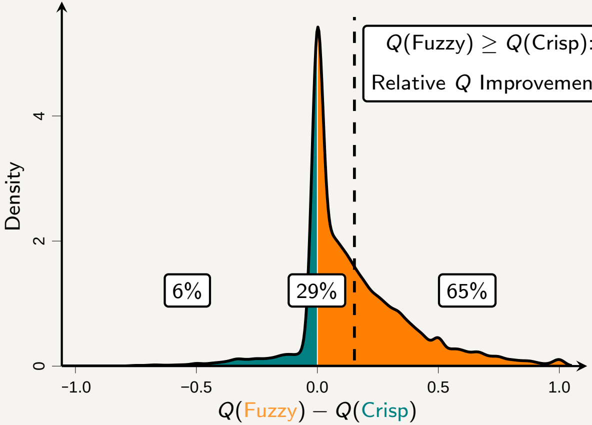
Benchmark

Average Results



Benchmark

Paired Comparison



NFGE

- Motivation
- Approach
- Benchmark

Introduction

Both, autonomic computing and self-adaptive system research agree, only the knowledge of internal state (self aware) and external situation (context aware) allows for proper adaptation.

Psaier, H. and Dustdar, S. (2011).
A survey on self-healing systems: Approaches and systems.

NFGE

Goodness Modeling

1. Select system variables
2. Pass/Fail feedback lists
3. Kernel Density Estimation $\rightarrow$ Probability Density Function

$$\breve{\mathbf{g}}_{\mathbf{j}}(st) = \frac{\mathbf{pdf}_{\mathbf{pass}}(st)}{\mathbf{pdf}_{\mathbf{pass}}(st) + \mathbf{pdf}_{\mathbf{fail}}(st)}$$

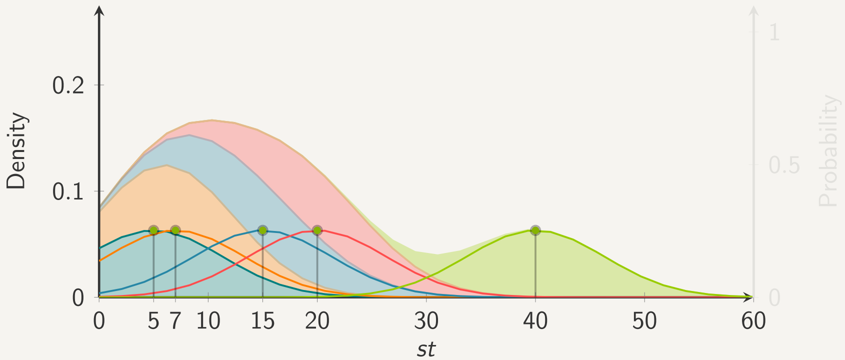
Approach

Example

$FB_{pass} = \{5, 7, 15, 20, 40\}$

$FB_{fail} = \{40, 44, 60\}$

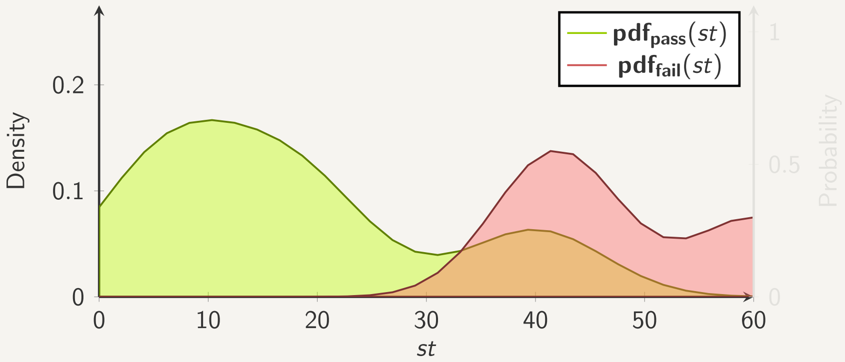
$$\breve{\mathbf{g}}_{\mathbf{j}}(st) = \frac{\mathbf{pdf}_{\mathbf{pass}}(st)}{\mathbf{pdf}_{\mathbf{pass}}(st) + \mathbf{pdf}_{\mathbf{fail}}(st)}$$



Approach

$$FB_{pass} = \{5, 7, 15, 20, 40\}$$
$$FB_{fail} = \{40, 44, 60\}$$

$$\breve{g}_j(st) = \frac{\mathbf{pdf}_{\mathbf{pass}}(st)}{\mathbf{pdf}_{\mathbf{pass}}(st) + \mathbf{pdf}_{\mathbf{fail}}(st)}$$

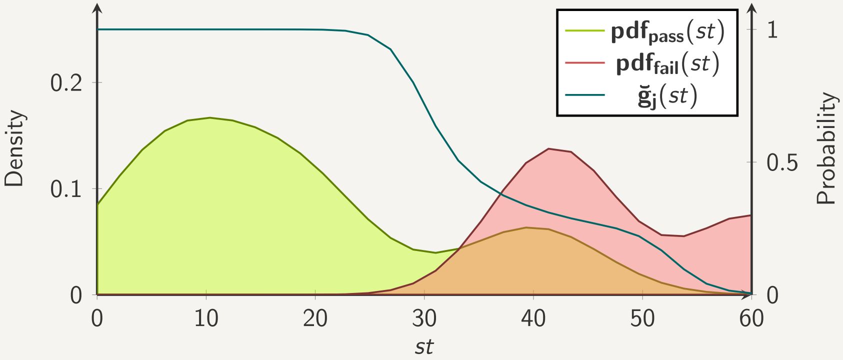


Approach

Example

$$FB_{pass} = \{5, 7, 15, 20, 40\}$$
$$FB_{fail} = \{40, 44, 60\}$$

$$\breve{\mathbf{g}}_{\mathbf{j}}(st) = \frac{\mathbf{pdf}_{\mathbf{pass}}(st)}{\mathbf{pdf}_{\mathbf{pass}}(st) + \mathbf{pdf}_{\mathbf{fail}}(st)}$$



NFGE

State Spectrum

$$A_{ij} = \begin{cases} 1, & c_j \text{ involved in transaction } i \\ 0, & \text{otherwise} \end{cases}$$

$$A_{ij} = \begin{cases} \langle st_1, \cdots, st_k \rangle, & c_j \text{ involved in transaction } i \\ \emptyset, & \text{otherwise} \end{cases}$$

Benchmark

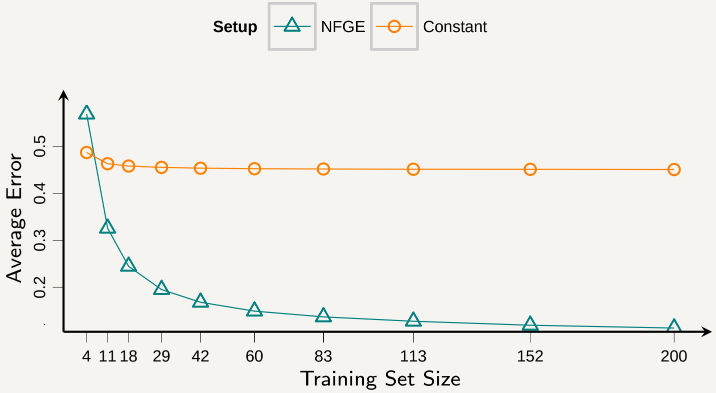
1. Random reference goodness models
2. Feedback generated from reference models using a Bernoulli process
3. Goodness estimations generated for different # of feedback observations

Estimation Error Absolute difference of the reference and estimated goodness values

Constant estimator $\breve{\mathbf{g}}_{\mathbf{j}}(st) = \frac{|FB_{pass}|}{|FB_{pass}|+|FB_{fail}|}$

Benchmark

Goodness Modeling



Conclusions

- Hypothesis
- Contributions
- Future Work

Hypothesis

SFL algorithms can be improved to better cope with the accuracy and latency requirements of run-time environments.

Publications

Published

- *MHS*^{2}
  - **Cardoso, N. and Abreu, R. 2014. An efficient distributed algorithm for computing minimal hitting sets. In Proceedings of the 25th International Workshop on Principles of Diagnosis, DX'14. (Best Paper Award)**
  - Cardoso, N. and Abreu, R. 2013. *MHS*^{2}: A map-reduce heuristic-driven minimal hitting set search algorithm. In Proceedings of the 2013 International Conference on Multicore Software Engineering, Performance, and Tools, MUSEPAT'13, pages 25–36.
  - Cardoso, N. and Abreu, R. 2013. A distributed approach to diagnosis candidate generation. In Proceedings of the 16th Portuguese Conference on Artificial Intelligence, EPIA'13, pages 175–186.
- Fuzzinel
  - **Cardoso, N. and Abreu, R. 2014. Enhancing reasoning approaches to diagnose functional and non-functional errors. In Proceedings of the 25th International Workshop on Principles of Diagnosis, DX'14. (Best Paper Award Nominee)**
- NFGE
  - **Cardoso, N. and Abreu, R. 2013. A kernel density estimate-based approach to component goodness modeling. In Proceedings of the 27th AAAI Conference on Artificial Intelligence, AAAI'13.**

Publications

Under Review

- *MHS*^{2}
  - Artificial Intelligence Journal
- Fuzzinel
  - European Conference on Artificial Intelligence

Software

- *MHS*^{2}: `https://github.com/npcardoso/MHS2`
- Crowbar

Future Work

- Technological challenges
  - Abstration specification
  - Distributed spectra collection
- Continuous system operation
  - Observation relevance
  - Model ageing
- Error Detection
  - Automatic oracle generation
  - Oracle confidence
- NFGE
  - Relevant state variables' detection
  - Automatic feedback generation

Spectrum-based Diagnosis for Run-time Systems

Nuno Cardoso

nunopcardoso@gmail.com

Supervisor : Professor Rui Abreu

Co-Supervisor : Professor David Garlan

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

HASLab INESCTEC

27/04/2016