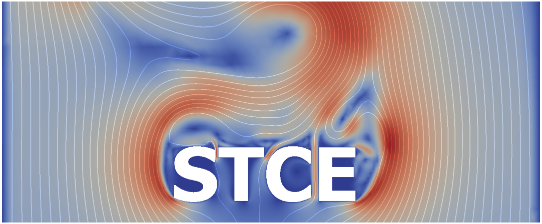


Interval-Adjoint Significance Analysis: A Case Study

J. Deussen J. Riehme U. Naumann
Software and Tools for Computational Engineering
RWTH Aachen University

2nd Workshop on Approximate Computing, Prague, January 2016

Outline

SCoRPiO

Interval-Adjoint Significance Analysis
- Overview
- Fundamentals of Interval-Adjoint Significance Analysis
- Extension of the Interval-Adjoint Significance Analysis

Case study: 1-D heat equation

Summary of Results and Future Work

SCoRPiO

SCoRPiO
Significance-Based Computing for Reliability and Power Optimization

SEVENTH FRAMEWORK PROGRAMME

Motivation

- Compute the significance of each arithmetic operation for a given source code
- Significance identifies less important operations with regards to the quality of the output
- Insignificant operations can be approximated

SCoRPiO

SCoRPiO
Significance-Based Computing for
Reliability and Power Optimization

SEVENTH FRAMEWORK PROGRAMME

Motivation

- Compute the significance of each arithmetic operation for a given source code
- Significance identifies less important operations with regards to the quality of the output
- Insignificant operations can be approximated

Approximation

Save energy by

- performing computations on low-power but unreliable hardware
- using lower precision on less power consuming hardware
- replacing computations with representative expressions or constants

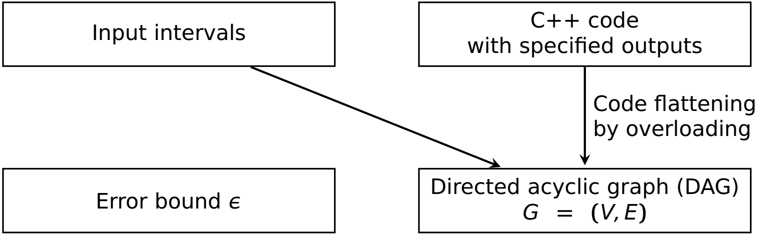
Interval-Adjoint Significance Analysis

Input intervals

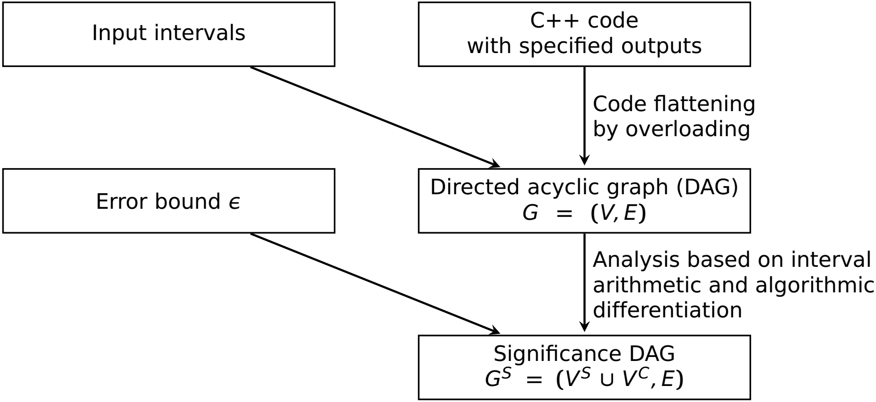
C++ code
with specified outputs

Error bound ϵ

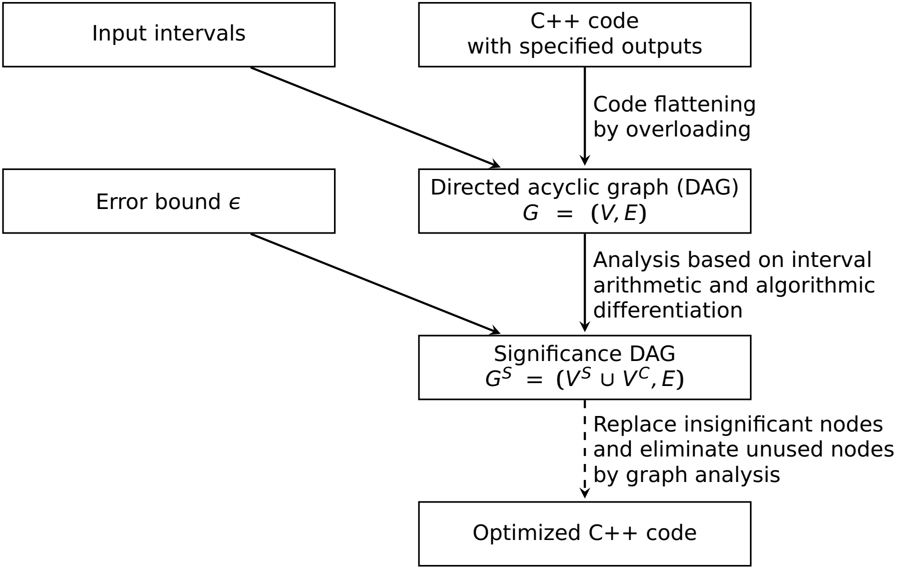
Interval-Adjoint Significance Analysis



Interval-Adjoint Significance Analysis



Interval-Adjoint Significance Analysis



Fundamentals of Interval-Adjoint Significance Analysis

- Interval Arithmetics (IA)
  - Forward propagation of intervals
  - Computes interval enclosures of all intermediates and outputs for given inputs
  - Issues: relational operators, overestimation (e.g. wrapping effect)

Fundamentals of Interval-Adjoint Significance Analysis

- Interval Arithmetics (IA)
  - Forward propagation of intervals
  - Computes interval enclosures of all intermediates and outputs for given inputs
  - Issues: relational operators, overestimation (e.g. wrapping effect)
- Adjoint Algorithmic Differentiation (AAD)
  - Backward propagation of adjoints
  - Computes derivatives of the outputs with respect to all inputs and intermediates
  - Issues: differentiability required, potential high memory consumption

Fundamentals of Interval-Adjoint Significance Analysis

- Interval Arithmetics (IA)
  - Forward propagation of intervals
  - Computes interval enclosures of all intermediates and outputs for given inputs
  - Issues: relational operators, overestimation (e.g. wrapping effect)
- Adjoint Algorithmic Differentiation (AAD)
  - Backward propagation of adjoints
  - Computes derivatives of the outputs with respect to all inputs and intermediates
  - Issues: differentiability required, potential high memory consumption
- Significance of v_i for output y defined as

$$S_y(v_i) = w[v_i] \cdot \max|\nabla_{[v_i]}[y]|$$

with

  - $w[v_i]$: influence of the input domain on variable v_i
  - $\max|\nabla_{[v_i]}[y]|$: influence of variable v_i on the output

Fundamentals of Interval-Adjoint Significance Analysis

- Interval Arithmetics (IA)
  - Forward propagation of intervals
  - Computes interval enclosures of all intermediates and outputs for given inputs
  - Issues: relational operators, overestimation (e.g. wrapping effect)
- Adjoint Algorithmic Differentiation (AAD)
  - Backward propagation of adjoints
  - Computes derivatives of the outputs with respect to all inputs and intermediates
  - Issues: differentiability required, potential high memory consumption
- Significance of v_i for output y defined as

$$S_y(v_i) = w[v_i] \cdot \max |\nabla_{[v_i]}[y]|$$

  with
  - $w[v_i]$: influence of the input domain on variable v_i
  - $\max |\nabla_{[v_i]}[y]|$: influence of variable v_i on the output
- Variable v_i is insignificant if

$$S_y(v_i) \leq \epsilon$$

Extension of the Interval-Adjoint Significance Analysis

Interval Splitting

- Control flow splitting:
  - Enables computation of piecewise differentiable functions

	$v < c$ with $v \in [\underline{v_i}, \overline{v_i}]$		
$\overline{v_i} < c$	true		
$\underline{v_i} \leq c \leq \overline{v_i}$	true	if	$v \in [\underline{v_i}, c)$
	false	if	$v \in [c, \overline{v_i}]$
$c < \underline{v_i}$	false		

Interval Splitting

- Control flow splitting:
  - Enables computation of piecewise differentiable functions

	$v < c$ with $v \in [\underline{v_i}, \overline{v_i}]$
$\overline{v_i} < c$	true
$\underline{v_i} \le c \le \overline{v_i}$	true if $v \in [\underline{v_i}, c)$ false if $v \in [\overline{c}, \overline{v_i}]$
$c < \underline{v_i}$	false

- Exploration mode:
  - black-box analysis
  - requires maximal allowed width of intervals
  - split variables with large value (or adjoint) interval

Interval Splitting

- Control flow splitting:
  - Enables computation of piecewise differentiable functions

	$v < c$ with $v \in [\underline{v_i}, \overline{v_i}]$		
$\overline{v_i} < c$	true		
$\underline{v_i} \leq c \leq \overline{v_i}$	true	if	$v \in [\underline{v_i}, c)$
	false	if	$v \in [c, \overline{v_i}]$
$c < \underline{v_i}$	false		

- Exploration mode:
  - black-box analysis
  - requires maximal allowed width of intervals
  - split variables with large value (or adjoint) interval
- Quantification/verification mode:
  - requires domain expert knowledge
  - split only user selected input variables

Interval Splitting

- Control flow splitting:
  - Enables computation of piecewise differentiable functions

	$v < c$ with $v \in [\underline{v_i}, \overline{v_i}]$		
$\overline{v_i} < c$	true		
$\underline{v_i} \leq c \leq \overline{v_i}$	true	if	$v \in [\underline{v_i}, c)$
	false	if	$v \in [c, \overline{v_i}]$
$c < \underline{v_i}$	false		

- Exploration mode:
  - black-box analysis
  - requires maximal allowed width of intervals
  - split variables with large value (or adjoint) interval
- Quantification/verification mode:
  - requires domain expert knowledge
  - split only user selected input variables
- Ranking of scenario k for example with number of insignificant variables $|V_k^C|$ and input domain size $\prod_{i=1}^{n} w(D_k^i)$ by

$$r_k = |V_k^C| \cdot \prod_{i=1}^{n} w(D_k^i)$$

Given C++ code for solving the 1-D heat equation

- Input:

rod length	L	$=$	2	
spatial discretization	n	$=$	41	
temporal discretization	m	$=$	800	
final time	t_f	$=$	32	
heat coefficient	c	$=$	0.01	
initial rod temperature	$T(0,x)$	$=$	$[280,300]$,	$0 \le x \le 2$
temperature at the boundary	$T(t,0)$	$=$	$[280,300]$,	$t \in [0,t_f]$
temperature at candle	$T(t,L)$	$=$	$[1650,1700]$,	$t \in [0,t_f]$

- Output:

temperature at the middle $T(t_f, L/2)$

Given C++ code for solving the 1-D heat equation

- Input:

rod length	L	$=$	2	
spatial discretization	n	$=$	41	
temporal discretization	m	$=$	800	
final time	t_f	$=$	32	
heat coefficient	c	$=$	0.01	
initial rod temperature	$T(0,x)$	$=$	$[280,300],$	$0 \leq x \leq 2$
temperature at the boundary	$T(t,0)$	$=$	$[280,300],$	$t \in [0,t_f]$
temperature at candle	$T(t,L)$	$=$	$[1650,1700],$	$t \in [0,t_f]$

- Output:
  temperature at the middle $T(t_f, L/2)$

Result of IASA with error bound $\epsilon = 0$

- 2,702,061 nodes
- 1,292,973 nodes (48%) marked as insignificant

$\longrightarrow$ IASA exposed automatically: code is inefficient

Model

Heat distribution:

$$\frac{\partial T}{\partial t} = c \cdot \frac{\partial^2 T}{\partial x^2}, \qquad T = T(t, x, c),$$

The initial condition:

$$T(0, x, c) = T_0 \ \forall x \in (0, 2)$$

Boundary conditions:

$$T(t, 0, c) = T_{x1}, T(t, 2, c) = T_{x2} \ \forall t \in [0, t_f]$$

Model

Heat distribution:

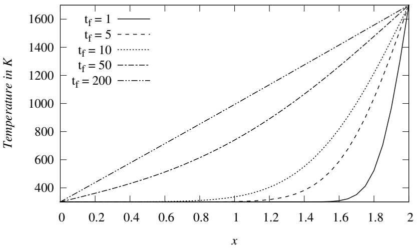
$$\frac{\partial T}{\partial t} = c \cdot \frac{\partial^2 T}{\partial x^2}, \qquad T = T(t, x, c),$$

The initial condition:

$$T(0, x, c) = T_0 \ \forall x \in (0, 2)$$

Boundary conditions:

$$T(t, 0, c) = T_{x1}, T(t, 2, c) = T_{x2} \ \forall t \in [0, t_f]$$



Case study: 1-D heat equation

Discretized System

- Number of spatial discretization points n
- Number of temporal discretization points m
- Linear system

$$\begin{pmatrix} 1 & {\color{blue}0} & 0 & \cdots & 0 \\ -a & 1+2a & -a & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -a & 1+2a & -a \\ 0 & \cdots & 0 & {\color{blue}0} & 1 \end{pmatrix} T^{k+1} = T^k, \quad k = 1, \ldots, m$$

- with

$$a = \frac{c\Delta t}{\Delta x^2}$$

Dense solver

- dense LU decomposition of system matrix A — $\mathcal{O}(n^3)$
- for t = 1, ..., m
  - dense forward substitution — $\mathcal{O}(n^2)$
  - dense backward substitution — $\mathcal{O}(n^2)$
- total complexity of the code with a dense solver: $\mathcal{O}(mn^2 + n^3)$

Dense solver

- dense LU decomposition of system matrix *A* — $\mathcal{O}(n^3)$
- for t = 1, ..., m
  - dense forward substitution — $\mathcal{O}(n^2)$
  - dense backward substitution — $\mathcal{O}(n^2)$
- total complexity of the code with a dense solver: $\mathcal{O}(mn^2 + n^3)$

Tridiagonal sparse solver

- sparse LU decomposition of system matrix *A* — $\mathcal{O}(n)$
- for t = 1, ..., m
  - sparse forward substitution — $\mathcal{O}(n)$
  - sparse backward substitution — $\mathcal{O}(n)$
- total complexity of the code with a sparse solver: $\mathcal{O}(mn)$

Case study: 1-D heat equation

Result of IASA with error bound $\epsilon = 0$ for the sparse code

- 161,018 nodes
- 1,655 nodes (1%) marked as insignificant

Case study: 1-D heat equation

Result of IASA with error bound $\epsilon = 0$ for the sparse code

- 161,018 nodes
- 1,655 nodes (1%) marked as insignificant

New Configuration

- Error bound $\epsilon = 1$
- Quantification mode:
  - Binary splitting
  - Only splitting of final time t_f

Case study: 1-D heat equation

Result of IASA with error bound $\epsilon = 0$ for the sparse code

- 161,018 nodes
- 1,655 nodes (1%) marked as insignificant

New Configuration

- Error bound $\epsilon = 1$
- Quantification mode:
  - Binary splitting
  - Only splitting of final time t_f

Interval Splitting Results and Ranking

k	t_f	$\lvert V_k^C \rvert$	$\frac{r_k}{1000}$	k	t_f	$\lvert V_k^C \rvert$	$\frac{r_k}{1000}$
1	[0, 1]	95583	95583	6	[2, 4]	16412	32824
2	[1, 2]	71007	71007	7	[4, 5]	31887	31887
3	[0, 2]	30832	61664	8	[5, 6]	27068	27068
4	[2, 3]	51363	51363	9	[0, 4]	5964	23856
5	[3, 4]	39132	39132	10	[6, 7]	23594	23594

Results of the case study

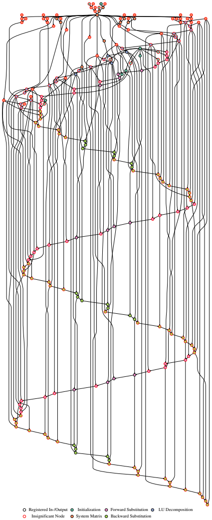
- Quantification mode yields that for small final simulation times most of the computations are insignificant
- 95,583 of 161,018 computations are marked as insignificant for $t_f = 1$

Results of the case study

- Quantification mode yields that for small final simulation times most of the computations are insignificant
- 95,583 of 161,018 computations are marked as insignificant for $t_f = 1$

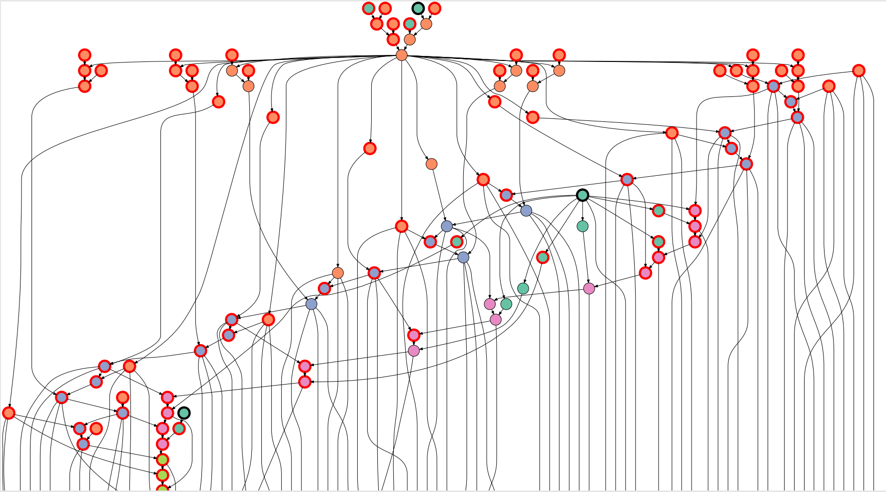
Visualization of the Significance Graph

- Number of time steps $m = 3$
- Number of space distributions $n = 9$
- Red framed nodes are insignificant
- Black framed nodes are Inputs or Outputs
- Different sections of the code have different colors



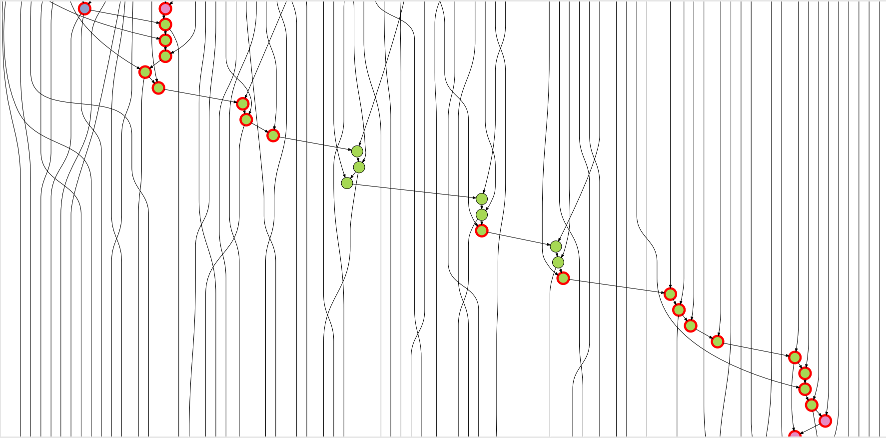
Case study: 1-D heat equation

Initialization, system matrix, first forward substitution



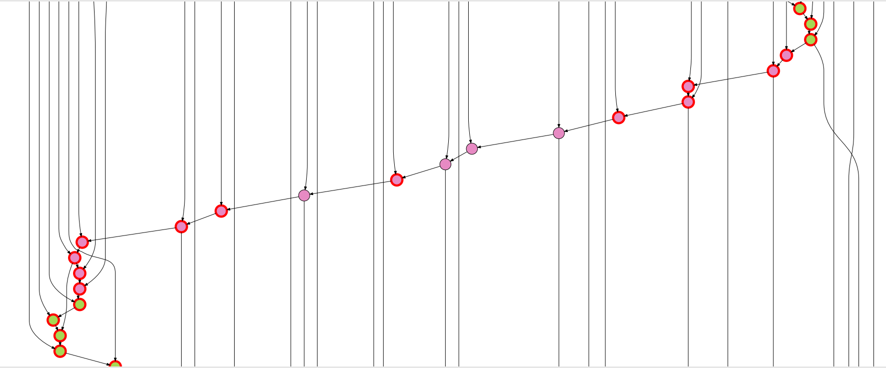
Case study: 1-D heat equation

Backward substitution



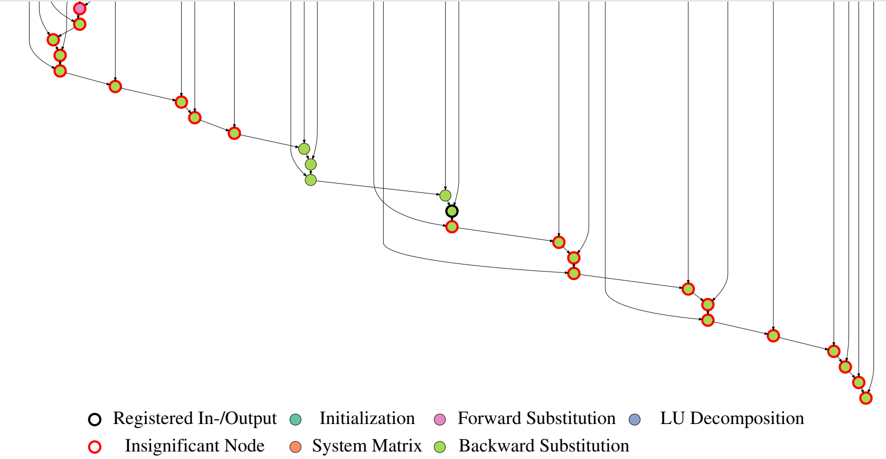
Case study: 1-D heat equation

Forward substitution



Case study: 1-D heat equation

Backward substitution



Summary of Results and Future Work

Summary of Results

- IASA is able to identify sparse systems
- Quantification mode was used to verify intuitive and well-known characteristics
- Results of the case study were used as a sanity check for IASA

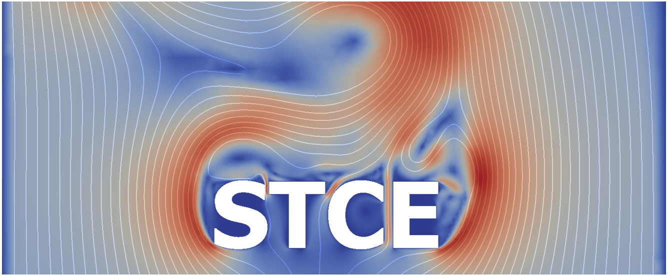
Summary of Results and Future Work

Summary of Results

- IASA is able to identify sparse systems
- Quantification mode was used to verify intuitive and well-known characteristics
- Results of the case study were used as a sanity check for IASA

Future Work

- Identify the quality of the optimized code for the corresponding input domain
- Implementation and testing of the exploration mode
- Calculate significances of larger applications by using IASA



Thank you very much!

www.stce.rwth-aachen.de
[deussen,riehme,naumann]@stce.rwth-aachen.de