Automated Analysis of Weak Memory Models

Artem Yushkovskiy[1,2]

MSc Candidate

Supervisors: **Assoc. Prof. Keijo Heljanko**[1]

Docent Igor I. Komarov[2]

[1]Department of Computer Science,
School of Science,
Aalto University (Espoo, Finland)

[2]Faculty of Information Security
and Computer Technologies,
ITMO University (Saint Petersburg, Russia)

Espoo, Saint Petersburg, 2018

Outline

Outline

Verification of concurrent software

Example: Write-write reordering (by compiler)

`{ x=0; y=0; }`	
P	Q
p_0 : $x \leftarrow 1$	q_0 : $y \leftarrow 1$
p_1 : $r_p \leftarrow y$	q_1 : $r_q \leftarrow x$

Verification of concurrent software

Example: Write-write reordering (by compiler)

`{ x=0; y=0; }`	
P	Q
p_0 : $x \leftarrow 1$	q_0 : $y \leftarrow 1$
p_1 : $r_p \leftarrow y$	q_1 : $r_q \leftarrow x$

p_0, p_1, q_0, q_1 $(0; 1)$
q_0, q_1, p_0, p_1 $(1; 0)$

Verification of concurrent software

Example: Write-write reordering (by compiler)

`{ x=0; y=0; }`	
P	Q
$p_0 : x \leftarrow 1$	$q_0 : y \leftarrow 1$
$p_1 : r_p \leftarrow y$	$q_1 : r_q \leftarrow x$

p_0, p_1, q_0, q_1	$(0; 1)$
q_0, q_1, p_0, p_1	$(1; 0)$
p_0, q_0, p_1, q_1	$(1; 1)$
p_0, q_0, q_1, p_1	$(1; 1)$
q_0, p_0, p_1, q_1	$(1; 1)$
q_0, p_0, q_1, p_1	$(1; 1)$

Verification of concurrent software

Example: Write-write reordering (by compiler)

`{ x=0; y=0; }`	
P	Q
p_0 : $x \leftarrow 1$	q_0 : $y \leftarrow 1$
p_1 : $r_p \leftarrow y$	q_1 : $r_q \leftarrow x$

p_0, p_1, q_0, q_1 $(0;1)$	$\underline{p_1}, \underline{p_0}, q_0, q_1$ $(0;1)$	$p_0, p_1, \underline{q_1}, \underline{q_0}$ $(0;1)$	$\underline{p_1}, \underline{p_0}, \underline{q_1}, \underline{q_0}$ $(0;1)$
q_0, q_1, p_0, p_1 $(1;0)$	$q_0, q_1, \underline{p_1}, \underline{p_0}$ $(1;0)$	$\underline{q_1}, \underline{q_0}, p_0, p_1$ $(1;0)$	$\underline{q_1}, \underline{q_0}, \underline{p_1}, \underline{p_0}$ $(1;0)$
p_0, q_0, p_1, q_1 $(1;1)$	$\underline{p_1}, q_0, \underline{p_0}, q_1$ $(0;1)$	$p_0, \underline{q_1}, p_1, \underline{q_0}$ $(0;1)$	$\underline{p_1}, \underline{q_1}, \underline{p_0}, \underline{q_0}$ $(0;0)$
p_0, q_0, q_1, p_1 $(1;1)$	$\underline{p_1}, q_0, q_1, \underline{p_0}$ $(0;0)$	$p_0, \underline{q_1}, \underline{q_0}, p_1$ $(1;1)$	$\underline{p_1}, \underline{q_1}, \underline{q_0}, \underline{p_0}$ $(0;0)$
q_0, p_0, p_1, q_1 $(1;1)$	$q_0, \underline{p_1}, \underline{p_0}, q_1$ $(1;1)$	$\underline{q_1}, p_0, p_1, \underline{q_0}$ $(0;0)$	$\underline{q_1}, \underline{p_1}, \underline{p_0}, \underline{q_0}$ $(0;0)$
q_0, p_0, q_1, p_1 $(1;1)$	$q_0, \underline{p_1}, q_1, \underline{p_0}$ $(1;0)$	$\underline{q_1}, p_0, \underline{q_0}, p_1$ $(1;0)$	$\underline{q_1}, \underline{p_1}, \underline{q_0}, \underline{p_0}$ $(0;0)$

Verification of concurrent software

Example: Store buffering (by hardware)

{ x=0; y=0; }	
P	Q
p_0 : $x \leftarrow 1$	q_0 : $y \leftarrow 1$
p_1 : $r_p \leftarrow y$	q_1 : $r_q \leftarrow x$

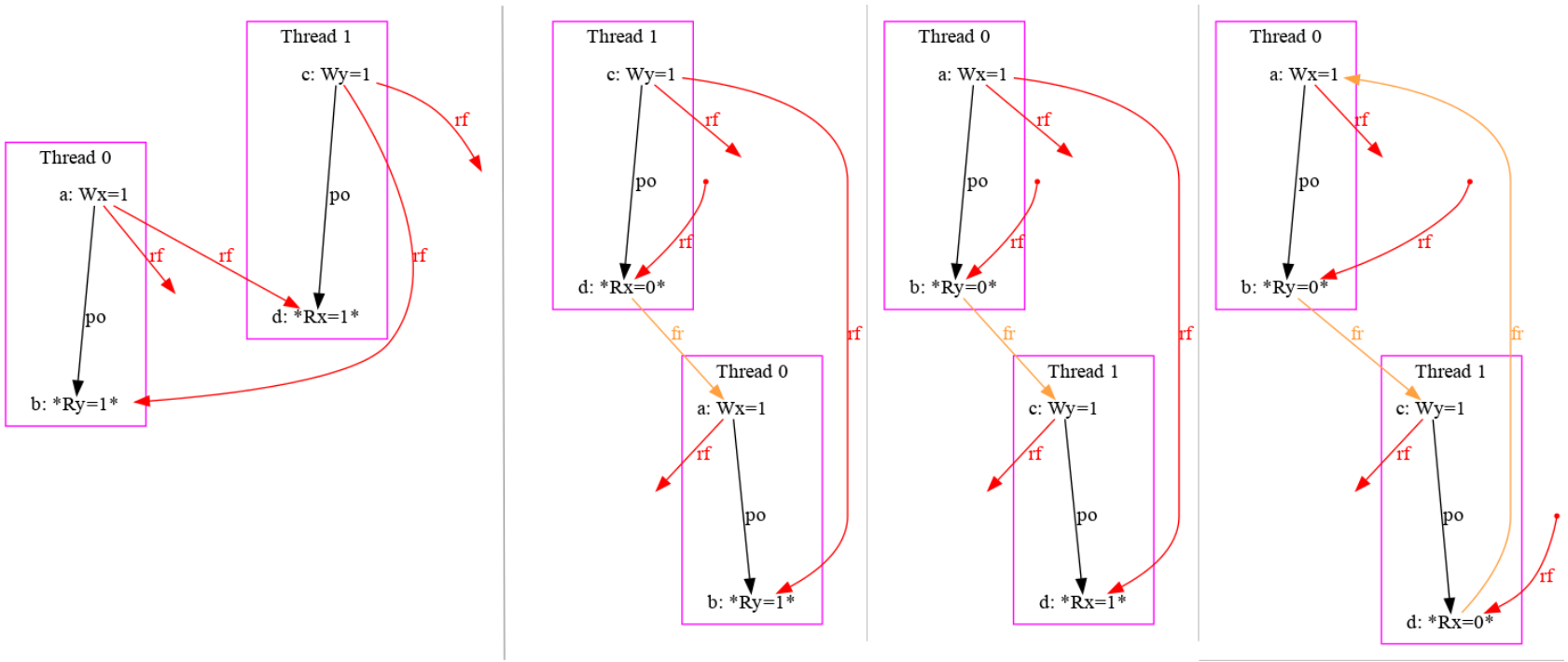

Figure: The four candidate executions

Verification of concurrent software

A weak memory model: Operational semantics

{ x=0; y=0; }	
P	Q
p_0 : $x \leftarrow 1$	q_0 : $y \leftarrow 1$
p_1 : $r_p \leftarrow y$	q_1 : $r_q \leftarrow x$

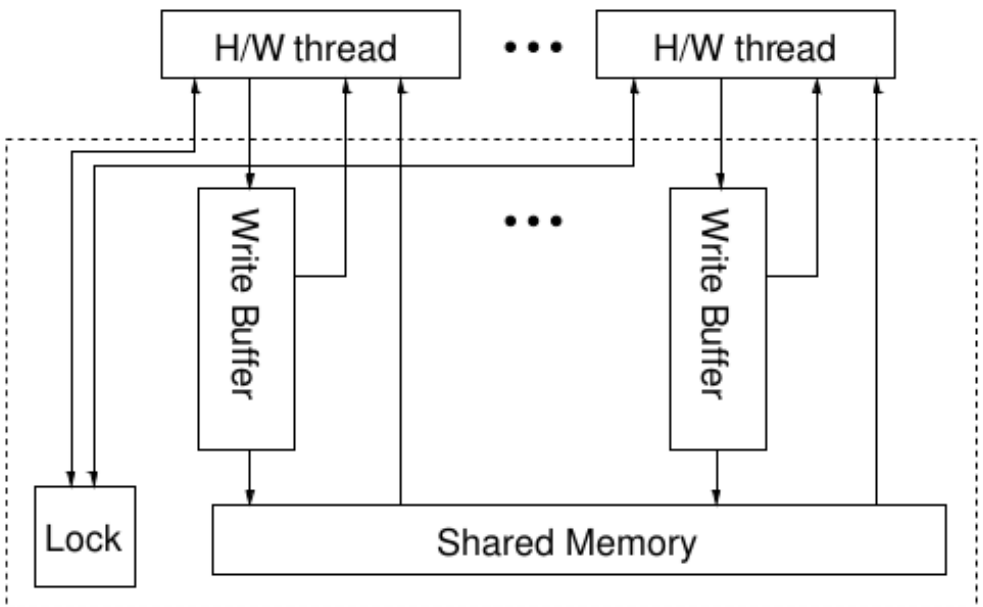

Figure: An x86-TSO abstract machine [**sewell2010x86**]

Verification of concurrent software

A weak memory model: Axiomatic semantics

-

Outline

Outline

1) 2) 3)

Outline

First Slide Title

Optional Subtitle

- My first point.
- My second point.

Outline

Second Slide Title

- First item.

Second Slide Title

- First item.
- Second item.

Second Slide Title

- First item.
- Second item.
- Third item.

Second Slide Title

- First item.
- Second item.
- Third item.
- Fourth item.

Second Slide Title

- First item.
- Second item.
- Third item.
- Fourth item.
- Fifth item.

Second Slide Title

- First item.
- Second item.
- Third item.
- Fourth item.
- Fifth item. Extra text in the fifth item.

Outline

Blocks

Block Title

You can also highlight sections of your presentation in a block, with it's own title

Theorem

There are separate environments for theorems, examples, definitions and proofs.

Example

Here is an example of an example block.

Summary

- The first main message of your talk in one or two lines.
- The second main message of your talk in one or two lines.
- Perhaps a third message, but not more than that.
- Outlook
  - Something you haven't solved.
  - Something else you haven't solved.

Bibliography I