



Semantics and Verification of Software

Summer Semester 2015

Lecture 18: Axiomatic Semantics of WHILE VI
(Proving Timed Correctness)

Thomas Noll

Software Modeling and Verification Group

RWTH Aachen University

<http://moves.rwth-aachen.de/teaching/ss-15/sv-sw/>

Online Registration for Seminars and Practical Courses (Praktika) in Winter Term 2015/16

Who?

- Students of:
- Master Courses
 - Bachelor Informatik (~~Pro~~Seminar!)

Where?

www.graphics.rwth-aachen.de/apse

When?

25.06.2015 - 08.07.2015

Schedule

- Lectures:

- Tue 30 June, Tue 7 July
- *not* Thu 2 July, Thu 9 July

- Exams:

- Thu 23 July
- Wed 26 August
- Thu 24 September

Recap: Correctness Properties for Execution Time

Outline of Lecture 18

Recap: Correctness Properties for Execution Time

Proving Timed Correctness

Recap: Correctness Properties for Execution Time

Timed Evaluation of Arithmetic Expressions

Definition (Timed Evaluation of arithmetic expressions (extends Definition 2.2))

Expression a **evaluates to** $z \in \mathbb{Z}$ in state σ in $\tau \in \mathbb{N}$ steps (notation: $\langle a, \sigma \rangle \xrightarrow{\tau} z$) if this relationship is derivable by means of the following rules:

Axioms:

$$\frac{}{\langle z, \sigma \rangle \xrightarrow{1} z} \quad \frac{}{\langle x, \sigma \rangle \xrightarrow{1} \sigma(x)}$$

Rules:

$$\frac{\langle a_1, \sigma \rangle \xrightarrow{\tau_1} z_1 \quad \langle a_2, \sigma \rangle \xrightarrow{\tau_2} z_2}{\langle a_1 + a_2, \sigma \rangle \xrightarrow{\tau_1 + \tau_2 + 1} z} \quad \text{where } z := z_1 + z_2$$

$$\frac{\langle a_1, \sigma \rangle \xrightarrow{\tau_1} z_1 \quad \langle a_2, \sigma \rangle \xrightarrow{\tau_2} z_2}{\langle a_1 - a_2, \sigma \rangle \xrightarrow{\tau_1 + \tau_2 + 1} z} \quad \text{where } z := z_1 - z_2$$

$$\frac{\langle a_1, \sigma \rangle \xrightarrow{\tau_1} z_1 \quad \langle a_2, \sigma \rangle \xrightarrow{\tau_2} z_2}{\langle a_1 * a_2, \sigma \rangle \xrightarrow{\tau_1 + \tau_2 + 1} z} \quad \text{where } z := z_1 \cdot z_2$$

Recap: Correctness Properties for Execution Time

Timed Evaluation of Boolean Expressions

Definition (Timed Evaluation of Boolean expressions (extends Definition 2.7))

For $b \in BExp$, $\sigma \in \Sigma$, $\tau \in \mathbb{N}$, and $t \in \mathbb{B}$, the **timed evaluation relation** $\langle b, \sigma \rangle \xrightarrow{\tau} t$ is defined by:

$$\begin{array}{c}
 \frac{\langle a_1, \sigma \rangle \xrightarrow{\tau_1} z \quad \langle a_2, \sigma \rangle \xrightarrow{\tau_2} z}{\langle a_1 = a_2, \sigma \rangle \xrightarrow{\tau_1 + \tau_2 + 1} \text{true}} \quad \frac{\langle a_1, \sigma \rangle \xrightarrow{\tau_1} z_1 \quad \langle a_2, \sigma \rangle \xrightarrow{\tau_2} z_2}{\langle a_1 = a_2, \sigma \rangle \xrightarrow{\tau_1 + \tau_2 + 1} \text{false}} \text{ if } z_1 \neq z_2 \\
 \frac{\langle a_1, \sigma \rangle \xrightarrow{\tau_1} z_1 \quad \langle a_2, \sigma \rangle \xrightarrow{\tau_2} z_2}{\langle a_1 > a_2, \sigma \rangle \xrightarrow{\tau_1 + \tau_2 + 1} \text{true}} \text{ if } z_1 > z_2 \quad \frac{\langle a_1, \sigma \rangle \xrightarrow{\tau_1} z_1 \quad \langle a_2, \sigma \rangle \xrightarrow{\tau_2} z_2}{\langle a_1 > a_2, \sigma \rangle \xrightarrow{\tau_1 + \tau_2 + 1} \text{false}} \text{ if } z_1 \leq z_2 \\
 \frac{\langle b, \sigma \rangle \xrightarrow{\tau} \text{false}}{\langle \neg b, \sigma \rangle \xrightarrow{\tau + 1} \text{true}} \quad \frac{\langle b, \sigma \rangle \xrightarrow{\tau} \text{true}}{\langle \neg b, \sigma \rangle \xrightarrow{\tau + 1} \text{false}} \\
 \frac{\langle b_1, \sigma \rangle \xrightarrow{\tau_1} \text{true} \quad \langle b_2, \sigma \rangle \xrightarrow{\tau_2} \text{true}}{\langle b_1 \wedge b_2, \sigma \rangle \xrightarrow{\tau_1 + \tau_2 + 1} \text{true}} \quad \frac{\langle b_1, \sigma \rangle \xrightarrow{\tau_1} \text{true} \quad \langle b_2, \sigma \rangle \xrightarrow{\tau_2} \text{false}}{\langle b_1 \wedge b_2, \sigma \rangle \xrightarrow{\tau_1 + \tau_2 + 1} \text{false}} \\
 \frac{\langle b_1, \sigma \rangle \xrightarrow{\tau_1} \text{false} \quad \langle b_2, \sigma \rangle \xrightarrow{\tau_2} \text{true}}{\langle b_1 \wedge b_2, \sigma \rangle \xrightarrow{\tau_1 + \tau_2 + 1} \text{false}} \quad \frac{\langle b_1, \sigma \rangle \xrightarrow{\tau_1} \text{false} \quad \langle b_2, \sigma \rangle \xrightarrow{\tau_2} \text{false}}{\langle b_1 \wedge b_2, \sigma \rangle \xrightarrow{\tau_1 + \tau_2 + 1} \text{false}}
 \end{array}$$

(\vee analogously)

Recap: Correctness Properties for Execution Time

Timed Execution of Statements

Definition (Timed execution relation for statements (extends Definition 3.2))

For $c \in \text{Cmd}$, $\sigma, \sigma' \in \Sigma$, and $\tau \in \mathbb{N}$, the **timed execution relation** $\langle c, \sigma \rangle \xrightarrow{\tau} \sigma'$ is defined by:

$$\begin{array}{c} \text{(skip)} \frac{}{\langle \text{skip}, \sigma \rangle \xrightarrow{1} \sigma} \qquad \text{(asgn)} \frac{\langle a, \sigma \rangle \xrightarrow{\tau} z}{\langle x := a, \sigma \rangle \xrightarrow{\tau+1} \sigma[x \mapsto z]} \\ \text{(seq)} \frac{\langle c_1, \sigma \rangle \xrightarrow{\tau_1} \sigma' \quad \langle c_2, \sigma' \rangle \xrightarrow{\tau_2} \sigma''}{\langle c_1 ; c_2, \sigma \rangle \xrightarrow{\tau_1 + \tau_2} \sigma''} \qquad \text{(if-t)} \frac{\langle b, \sigma \rangle \xrightarrow{\tau} \text{true} \quad \langle c_1, \sigma \rangle \xrightarrow{\tau_1} \sigma'}{\langle \text{if } b \text{ then } c_1 \text{ else } c_2 \text{ end}, \sigma \rangle \xrightarrow{\tau + \tau_1 + 2} \sigma'} \\ \text{(wh-f)} \frac{\langle b, \sigma \rangle \xrightarrow{\tau} \text{false}}{\langle \text{while } b \text{ do } c \text{ end}, \sigma \rangle \xrightarrow{\tau+1} \sigma} \qquad \text{(if-f)} \frac{\langle b, \sigma \rangle \xrightarrow{\tau} \text{false} \quad \langle c_2, \sigma \rangle \xrightarrow{\tau_2} \sigma'}{\langle \text{if } b \text{ then } c_1 \text{ else } c_2 \text{ end}, \sigma \rangle \xrightarrow{\tau + \tau_2 + 1} \sigma'} \\ \text{(wh-t)} \frac{\langle b, \sigma \rangle \xrightarrow{\tau} \text{true} \quad \langle c, \sigma \rangle \xrightarrow{\tau_1} \sigma' \quad \langle \text{while } b \text{ do } c \text{ end}, \sigma' \rangle \xrightarrow{\tau_2} \sigma''}{\langle \text{while } b \text{ do } c \text{ end}, \sigma \rangle \xrightarrow{\tau + \tau_1 + \tau_2 + 2} \sigma''} \end{array}$$

Recap: Correctness Properties for Execution Time

Timed Correctness Properties

Now: **timed correctness properties** of the form

$$\{A\} c \{e \Downarrow B\}$$

where $c \in \text{Cmd}$, $A, B \in \text{Assn}$, and $e \in \text{AExp}$

Validity of property $\{A\} c \{e \Downarrow B\}$

For all states $\sigma \in \Sigma$ which satisfy A : the execution of c in σ terminates in a state satisfying B , and the required **execution time** is in $\mathcal{O}(e)$

Example

1. $\{x = 3\} y := 1; \text{ while } \neg(x=1) \text{ do } y := y * x; x := x - 1 \text{ end } \{1 \Downarrow \text{true}\}$ expresses that for constant input 3, the execution time of the factorial program is bounded by a constant
2. $\{x > 0\} y := 1; \text{ while } \neg(x=1) \text{ do } y := y * x; x := x - 1 \text{ end } \{x \Downarrow \text{true}\}$ expresses that for positive inputs, the execution time of the factorial program is linear in that value

Recap: Correctness Properties for Execution Time

Semantics of Timed Correctness Properties

Definition (Semantics of timed correctness properties (extends Definition 11.1))

Let $A, B \in Assn$, $c \in Cmd$, and $e \in AExp$. Then $\{A\} c \{e \Downarrow B\}$ is called **valid** (notation: $\models \{A\} c \{e \Downarrow B\}$) if there exists $k \in \mathbb{N}$ such that for each $l \in Int$ and each $\sigma \models^l A$, there exist $\sigma' \in \Sigma$ and $\tau \leq k \cdot \mathcal{A}[[e]]\sigma$ such that $\langle c, \sigma \rangle \xrightarrow{\tau} \sigma'$ and $\sigma' \models^l B$

Note: e is evaluated in initial (rather than final) state

Recap: Correctness Properties for Execution Time

Proving Timed Correctness I

Definition (Hoare Logic for timed correctness (extends Definition 11.3))

The **Hoare rules for timed correctness** are given by (where $i, u \in LVar$)

$$\frac{}{(\text{skip}) \{A\} \text{ skip } \{1 \Downarrow A\}}$$

$$\frac{}{(\text{asgn}) \{A[x \mapsto a]\} x := a \{1 \Downarrow A\}}$$

$$\frac{\{A \wedge e'_2 = u\} c_1 \{e_1 \Downarrow C \wedge e_2 \leq u\} \{C\} c_2 \{e_2 \Downarrow B\}}{(\text{seq}) \{A\} c_1 ; c_2 \{e_1 + e'_2 \Downarrow B\}}$$

$$\frac{\{A \wedge b\} c_1 \{e \Downarrow B\} \{A \wedge \neg b\} c_2 \{e \Downarrow B\}}{(\text{if}) \{A\} \text{ if } b \text{ then } c_1 \text{ else } c_2 \text{ end } \{e \Downarrow B\}}$$

$$\frac{\{i \geq 0 \wedge A(i+1) \wedge e' = u\} c \{e_0 \Downarrow A(i) \wedge e \leq u\}}{(\text{while}) \{\exists i. i \geq 0 \wedge A(i)\} \text{ while } b \text{ do } c \text{ end } \{e \Downarrow A(0)\}}$$

where $\models (i \geq 0 \wedge A(i+1)) \Rightarrow (b \wedge e \geq e_0 + e')$ and $\models A(0) \Rightarrow (\neg b \wedge e \geq 1)$

$$\frac{\models (A \Rightarrow (A' \wedge \exists k \in \mathbb{N}. e' \leq k \cdot e)) \{A'\} c \{e' \Downarrow B'\} \models (B' \Rightarrow B)}{(\text{cons}) \{A\} c \{\Downarrow e\} B}$$

Proving Timed Correctness

Outline of Lecture 18

Recap: Correctness Properties for Execution Time

Proving Timed Correctness

Proving Timed Correctness

Examples of Proving Timed Correctness

Example 18.1

1. Prove that

$$\vdash \{x > 0\} y:=1; \text{ while } \neg(x=1) \text{ do } y:=y*x; x:=x-1 \text{ end } \{x \Downarrow \text{true}\}$$

(on the board)

Proving Timed Correctness

Examples of Proving Timed Correctness

Example 18.1

1. Prove that

$$\vdash \{x > 0\} y:=1; \text{ while } \neg(x=1) \text{ do } y:=y*x; x:=x-1 \text{ end } \{x \Downarrow \text{true}\}$$

(on the board)

2. Determine expression e_{fac} such that

$$\vdash \{x > 0\} y:=1; \text{ while } \neg(x=1) \text{ do } y:=y*x; x:=x-1 \text{ end } \{e_{fac} \Downarrow \text{true}\}$$

(on the board)