Proof Principles Example. Reader and Writer Conclusions

Discrete Transition Systems (Recall)

The Language of Temporal Logi

Proof Principles Example. Reader and Writer Conclusions

Given the following transition system S

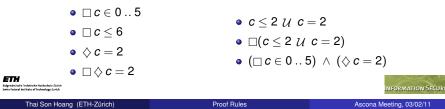
	Given the following transition system S
Proof Rules for Invariance and Liveness Properties	system Svariables $\overline{v} \in \overline{T}$ initially init(\overline{v})
Thai Son Hoang	$\begin{array}{rcl} \text{events} \\ \text{evt}_i & \cong & G_i(\overline{\nu}) \longrightarrow \overline{\nu} := \overline{f}_i(\overline{\nu}) \end{array}$
Chair of Information Security, Department of Computer Science Swiss Federal Institute of Technology Zürich (ETH Zürich)	 v denotes the vector of variables v₁,, v_n. <i>init</i>(v) is the initialisation.
3rd February 2011, Ascona Meeting	• $G_i(\overline{v})$ is the guard of event evt _i .
	• evt_i is said to be enabled in some state <i>s</i> if $G_i(\overline{v})$ holds in <i>s</i> .
Exercise Bigenetic related in 2004 Seto Your Letter of Hodewig Grids	$\underset{\text{We need to the original states of the section of event evt}_{i}.$
The Language of Temporal Logic Proof Principles Example. Reader and Writer Conclusions Executions and Traces (of States)	The Language of Temporal Logic Proof Principles Example. Reader and Writer Conclusions
Executions $\alpha = s_0 \xrightarrow{a_0} s_1 \xrightarrow{a_1} s_2 \xrightarrow{a_2} s_3 \xrightarrow{a_3} \dots$ Traces $\sigma = s_0, s_1, s_2, s_3, \dots$ $\mathcal{T}(S)$ denotes the set of all traces of system S.	The Language of Temporal Logic
Example	Proof Principles
system Countereventsvariables $c \in \mathbb{Z}$ inc $\hat{=}$ $c \neq 5 \longrightarrow c := c + 1$ initially $c = 0$ dec $\hat{=}$ $c > 3 \longrightarrow c := c - 1$	Example. Reader and Writer
$\sigma_{Counter}: \langle 0 \rangle, \langle 1 \rangle, \langle 2 \rangle, \langle 3 \rangle, \langle 4 \rangle, \langle 5 \rangle, \langle 4 \rangle, \langle 3 \rangle, \langle 4 \rangle, \langle 5 \rangle, \dots$	Conclusions
ETT: Represented Sectors Instantion Reals Network and Instantian of Instanting Sector	ETH Bigenerate Advantes schedule schedule Sched Main Hand Haltes of Holdings (2x4)

The Language of Temporal Logic Proof Principles Example. Reader and Writer

Temporal Formulas

Temporal formulas to be interpreted over traces.

- A (basic) state formula Q(v) is any first-order logic formula, e.g. 0 ≤ c, ¬(c = 0) ∧ c < 2, ∀m ⋅ m ≠ 0 ⇒ m ≤ c.
- The basic formulas can be extended by combining the Boolean operators (¬, ∧, ∨, ⇒) with temporal operators:
 - I: always
 - \diamond : eventually
 - U: until
- Example of extended formulas:



The Language of Temporal Logic Proof Principles Example. Reader and Writer Conclusions

 $\sigma \vDash \phi$ means that a trace σ satisfies formula ϕ

- For state formula ϕ , $\sigma \vDash \phi$ if all only if s_0 satisfies ϕ .
- Boolean operators are interpreted in the natural way, e.g.
 - $\sigma \vDash \phi_1 \land \phi_2$ if and only if $\sigma \vDash \phi_1$ and $\sigma \vDash \phi_2$.
- Temporal operators are interpreted as follows.

$\sigma\vDash \Box \phi$	if and only if	$orall m{k} \cdot m{0} \leq m{k} < m{l}(\sigma), \sigma^{(k)} \vDash \phi$
$\sigma\vDash \diamondsuit \phi$	if and only if	$\exists \mathbf{k} \cdot 0 \leq \mathbf{k} < \mathbf{l}(\sigma), \sigma^{(\mathbf{k})} \vDash \phi$
$\sigma \vDash \phi_1 \mathcal{U} \phi_2$	if and only if	$\exists \mathbf{k} \cdot 0 \leq \mathbf{k} < \mathbf{l}(\sigma)$ such that
		$\sigma^{(k)} \vDash \phi_2$ and
		$\forall i \cdot 0 \leq i < k, \sigma^{(i)} \vDash \phi_1$

Proof Rules

The Language of Temporal Logic Proof Principles Example. Reader and Writer

Length and Suffixes of Traces

Let $\sigma : s_o, s_1, \ldots$ be any non-empty trace.

- The length of σ denoted by $I(\sigma)$.
 - Finite trace $\sigma : s_0, \ldots, s_k$: $l(\sigma) = k + 1$.
 - Infinite trace: $l(\sigma) = \infty$.
- For $0 \le k < l(\sigma)$, *k*-suffix of σ is defined as

 $\sigma^{(k)} = \mathbf{s}_k, \mathbf{s}_{k+1}, \dots$

Eigeraiseuche Stedarsche Hackschule Zurich Swiss Federal Weitzeter of Federalogy Zurich		Information Secu	JRITY
Thai Son Hoang (ETH-Zürich)	Proof Rules	Ascona Meeting, 03/02/11	6 / 29



For the simple cases, when ϕ , ϕ_1 , ϕ_2 are state predicates.

σ satisfies $\Box \phi$	if and only if	all states in σ satisfy ϕ
σ satisfies $\diamondsuit \phi$	if and only if	some states in σ satisfy ϕ
σ satisfies $\phi_1 \mathcal{U} \phi_2$	if and only if	some state s_k satisfies ϕ_2 and all the states until s_k (excluding s_k) satisfy ϕ_2

 $\sigma_{\textit{Counter}}: \langle \mathbf{0} \rangle, \langle \mathbf{1} \rangle, \langle \mathbf{2} \rangle, \langle \mathbf{3} \rangle, \langle \mathbf{4} \rangle, \langle \mathbf{5} \rangle, \langle \mathbf{4} \rangle, \langle \mathbf{3} \rangle, \langle \mathbf{4} \rangle, \langle \mathbf{5} \rangle, \dots$

• $\sigma_{Counter} \models \Box c \in 05$ • $\sigma_{Counter} \models \Box c \leq 6$ • $\sigma_{Counter} \models \diamondsuit c = 2$ • $\sigma_{Counter} \not\models \Box \diamondsuit c = 2$	• $\sigma_{Counter} \models c \leq 2 \ \mathcal{U} \ c = 2$ • $\sigma_{Counter} \not\models \Box (c \leq 2 \ \mathcal{U} \ c = 2)$ • $\sigma_{Counter} \models (\Box \ c \in 05) \land (\diamondsuit \ c = 2)$
desize Robednic Zirch	Information Se

Proof Rules

ETH

ETH Bidgerdssluche lech

The Language of Temporal Logic Proof Principles Example. Reader and Writer

Safety v.s. Liveness

- Safety properties: *something (bad) will never happen*.
 - Example: invariance properties.
 - Typically expressed by a temporal formula: $\Box \phi$ or $\phi_1 \Rightarrow \Box \phi_2$.
- Liveness properties: *something (good) will happen*.
 - Example: termination, responsiveness.
 - Typically expressed by a temporal formula: $\Diamond \phi$ or $\Box(\phi_1 \Rightarrow \Diamond \phi_2)$.
 - Extended: $\phi_1 \mathcal{U} \phi_2$ or $\Box(\phi_1 \Rightarrow \phi_2 \mathcal{U} \phi_3)$.

System Properties

- A system *S* satisfying property ϕ if all its traces satisfy ϕ . $S \models \phi$ if and only if $\sigma \models \phi$, for all $\sigma \in \mathcal{T}(S)$.
- $S \vdash \phi$ states that $S \models \phi$ is provable.

ar in traducture of tradematics 2000m	INFORMATION SECURITY	EEFFF Bigginghang standarding gravity banks variable attention of standarding stands.	Information Security
Thai Son Hoang (ETH-Zürich) Proof Rules	Ascona Meeting, 03/02/11 9 / 29	Thai Son Hoang (ETH-Zürich) Proof Rules	Ascona Meeting, 03/02/11 10 / :
The Language of Temporal Logic Proof Principles Example. Reader and Writer Conclusions		The Language of Temporal Logic Proof Principles Example. Reader and Writer Conclusions	
Proof Tools (1 of 3) μ_{1} to ϕ_{2}		Invariance Rules (1/2)	
Event leads from ϕ_1 to ϕ_2 • Let evt be an event of the form $G(\overline{v}) \longrightarrow \overline{v} := \overline{f}(v)$ • Let $\phi_1(\overline{v})$ and $\phi_2(\overline{v})$ be two state formulas.	(\overline{v})	$ \begin{array}{c} \vdash \textit{init}(\overline{v}) \Rightarrow \phi \\ \vdash S \text{ leads from } \phi \text{ to } \phi \end{array} \\ \hline S \vdash \Box \phi \end{array} $. INV _{induct}
• Event evt leads from $\phi_1(\overline{\nu})$ to $\phi_2(\overline{\nu})$ if		Counter $\vdash \Box c \in 05$	
$\phi_1(\overline{v}) \wedge G(\overline{v}) \Rightarrow \phi_2(\overline{f}(\overline{v}))$		system Countereventsvariables $c \in \mathbb{Z}$ inc $\widehat{=}$	\neq 5 \rightarrow c := c + 1
System leads from ϕ_1 to ϕ_2		initially $c = 0$ dec $\hat{=}$ c	$> 3 \longrightarrow c := c - 1$
A system <i>S</i> leads from ϕ_1 to ϕ_2 if every event evt in <i>S</i> leads from ϕ_1 to		• Initialisation: $\vdash c = 0 \Rightarrow c \in 05$ • inc: $c \in 05 \land c \neq 5 \Rightarrow c+1 \in 0$	5
• When S leads from ϕ_1 to ϕ_2 is provable, we writ	te	• dec: $c \in 0$ $5 \land c > 3 \Rightarrow c - 1 \in$	
		ETH	

Proof Principles Example, Reader and Writer

Invariance Rules (2/2)

$\vdash \phi_2 \Rightarrow \phi_1$	
$S \vdash \Box \phi_2$	INV theorem
$S \vdash \Box \phi_1$	ineorem

Counter $\vdash \Box c \leq 6$	
Choose ϕ_2 to be $c \in 0 5$.	
$\bullet \vdash c \in 05 \Rightarrow c \leq 6$	
● <i>Counter</i> ⊢ □ <i>c</i> ∈ 0 5	

EFFH Biggeräschelt Fichenleche Hickeichuld Zürich Swiha Felerini Institute of Fichenology Gräch		INFORMATION SEC	
Thai Son Hoang (ETH-Zürich)	Proof Rules	Ascona Meeting, 03/02/11	13 / 29



- Let ϕ be a state formula.
- System S is deadlock-free when ϕ holds if there exists an enabled event of S when ϕ holds.
- When the above fact is provable, we denote it as
 - \vdash *S* is deadlock-free when ϕ holds
- This is guaranteed by proving the following.

$$\phi(\overline{\mathbf{v}}) \Rightarrow G_1(\overline{\mathbf{v}}) \lor \ldots \lor G_n(\overline{\mathbf{v}})$$

Proof Rules

Proof Principles

Proof Tools (2 of 3) Convergence

Thai Son Hoang (ETH-Zürich)

• Let ϕ be a state formula.

- A trace is said to be convergent when ϕ holds if it does not end with an infinite sequences of states satisfying ϕ .
- System S is said to be convergent when ϕ holds if all its traces are convergent when ϕ holds.
- When the above fact is provable, we denote it as

$\vdash \mathbf{S} \downarrow \phi$

	Technique			
	• For system S wit	th events $\operatorname{evt}_{i} \widehat{=} \operatorname{G}_{\operatorname{i}}(\overline{v}) \longrightarrow \overline{v}$	$\overline{r} := \overline{f}_i(\overline{v})$	
	Give a integer value	ariant $V(\overline{v})$		
	S converges whe	en ϕ holds if for all events e	vt _i of <i>S</i>	
ЕТН	• $\phi(\overline{v}) \wedge G_i(\overline{v})$			
Fidgerösslache Swiss Federal In	• $\phi(\mathbf{v}) \wedge \mathbf{G}_i(\mathbf{v})$	$\Rightarrow V(\overline{f}_i(\overline{v})) < V(\overline{v})$		NTY
	Thai Son Hoang (ETH-Zürich)	Proof Rules	Ascona Meeting, 03/02/11	14/29

Proof Principles Liveness Rules (1/3) Always Eventually

> $\vdash S \downarrow \neg \phi$ $\vdash S$ is deadlock-free when $\neg \phi$ holds $LIVE_{\Box}$ \diamond $S \vdash \Box \diamondsuit \phi$

Proof Rules

ETH

15/29

Proof Principles Example. Reader and Writer

Liveness Rules (2/3)

$\vdash S \text{ leads from } \phi_1 \land \neg \phi_2 \text{ to } \phi_1 \lor \phi_2$ $S \vdash \Box \diamondsuit (\neg \phi_1 \lor \phi_2)$	
$S \vdash \Box(\phi_1 \Rightarrow \phi_1 \mathcal{U} \phi_2)$	$LIVE_{\mathcal{U}}$

Counter $\vdash \Box (c < 2 \Rightarrow (c < 2 \mathcal{U} c = 2))$
• Counter leads from $c < 2 \land \neg c = 2$ to $c < 2 \lor c = 2$, equivalently Counter leads from $c < 2$ to $c \le 2$
• inc: $c < 2 \land c \neq 5 \Rightarrow c+1 \leq 2$ • dec: $c < 2 \land c > 3 \Rightarrow c-1 \leq 2$
• Eventually: $\Box \diamondsuit (\neg c < 2 \lor c = 2)$, equivalent to $\Box \diamondsuit c \ge 2$

EFFFFF Edgeratorische Technische Hischschule Zürich Swiss Technische Mittalter of Februaring Zärich		Information Sec	URITY
Thai Son Hoang (ETH-Zürich)	Proof Rules	Ascona Meeting, 03/02/11	17/29

The Language of Temporal Logic Proof Principles Example. Reader and Writer

Example. Reader and Writer

system RdWr	events
variables $r, w \in \mathbb{Z}, \mathbb{Z}$	read $\widehat{=}$ $r \neq w \longrightarrow r := r + 1$
initially $r = 0 \land w = 0$	write $\hat{=} w < r + 3 \longrightarrow w := w + 1$

An execution

$\langle 0, 0 \rangle \xrightarrow{\text{write}}$	$\langle 0, 1 \rangle \xrightarrow{\text{write}}$	$\leftrightarrow \langle 0, 2 \rangle \xrightarrow{\text{write}}$	→ ⟨0, <mark>3</mark> ⟩ <u>rea</u>	\xrightarrow{ad} $\langle 1, 3 \rangle \xrightarrow{reac}$	$\xrightarrow{d} \langle 2, 3 \rangle \xrightarrow{read}$	$\langle 3, 3 \rangle$
write	$\langle 3, 4 \rangle \xrightarrow{\text{write}}$	$(3, 5) \xrightarrow{read}$	$\langle 4,5\rangle \xrightarrow{\text{write}}$	$\xrightarrow{\text{te}}$ $\langle 4, 6 \rangle \xrightarrow{\text{read}}$	$\rightarrow \langle 5, 6 \rangle \xrightarrow{\text{write}}$	$\langle 5, 7 \rangle \dots$

Proof Rules

he Language of Temporal Logic Proof Principles Example. Reader and Writer

Liveness Rules (3/3) Response

$egin{aligned} & Sdash \square(\phi_1\Rightarrow\phi_3)\ & Sdash \square(\phi_3\Rightarrow(\phi_3\mathcal{U}\phi_2)) \end{aligned}$	LIVEresponse
$\qquad \qquad $	Li v Liesponse

Counter $\vdash \Box (c = 0 \Rightarrow \Diamond c = 2)$	
Choose $\phi_3 \stackrel{_\frown}{_=} c < 2$	
• $\Box(c=0 \Rightarrow c<2)$	
• \Box (c < 2 \Rightarrow (c < 2 \mathcal{U} c = 2))	

EFFFF Bilgreinsche Technische Hickschule Zürich Swiss Federal Institzte ef Technology Zirich		INFORMATION SEC	URITY
Thai Son Hoang (ETH-Zürich)	Proof Rules	Ascona Meeting, 03/02/11	18 / 29

he Language of Temporal Logic Proof Principles Example. Reader and Writer

A Progress Properties

Reader's progress

The Reader will eventually read the data that the Writer wrote.

Formalisation. First attempt

- Can we prove $RdWr \models \Box \Diamond r = w$?
 - No, the Reader might always be behind the Writer (despite progressing).

Formalisation. Second attempt

$$RdWr \models \Box(w = K \Rightarrow \Diamond r = K)?$$

Proof Rules

ETH Bidgenässische Technische Hachschule Zürich

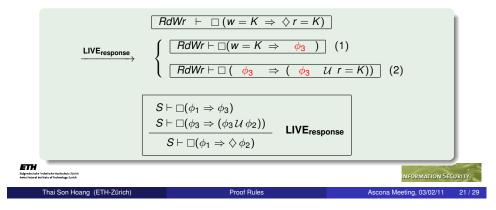
Hochschule Zürleh

ETH

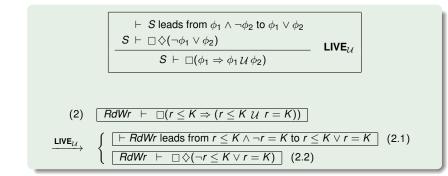
INFORMATION SECURITY Ascona Meeting, 03/02/11 19 / 29 Example. Reader and Writer

A Proof (1/6)

system RdWr	events
variables $r, w \in \mathbb{Z}, \mathbb{Z}$	read $\hat{=}$ $r \neq w \longrightarrow r := r + 1$
initially $r = 0 \land w = 0$	write $\hat{=} w < r + 3 \longrightarrow w := w + 1$



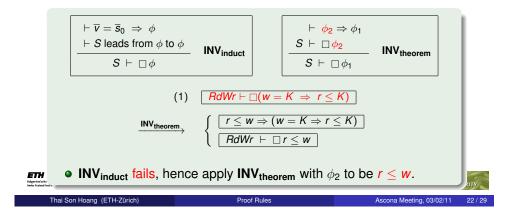
The Language of Temporal Proof Print Example. Reader and Conclu	iples Vriter
A Proof (3/6)	



Proof Principles Example. Reader and Writer

A Proof (2/6)

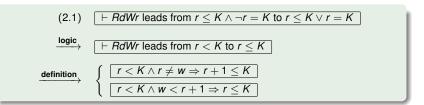
system RdWr	events
variables $r, w \in \mathbb{Z}, \mathbb{Z}$	read $\hat{=} r \neq w \longrightarrow r := r + 1$
initially $r = 0 \land w = 0$	write $\hat{=} w < r + 3 \longrightarrow w := w + 1$



Proof Principles Example. Reader and Writer

A Proof (4/6)

system RdWr	events
variables $r, w \in \mathbb{Z}, \mathbb{Z}$	read $\widehat{=}$ $r \neq w \longrightarrow r := r + 1$
initially $r = 0 \land w = 0$	write $\hat{=} w < r + 3 \longrightarrow w := w + 1$



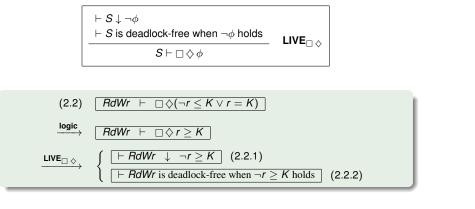


Eigeränschaft Trichenber Hischachnie Zarich Smith Federal huffteter of Federalogy Zarich		Information S
Thai Son Hoang (ETH-Zürich)	Proof Rules	Ascona Meeting, 03/02/11

24 / 29

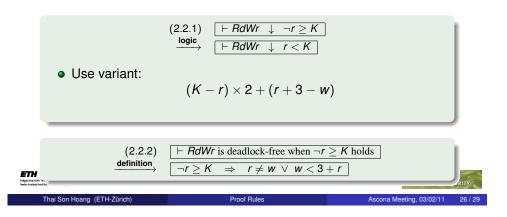
he Language of Temporal Logic Proof Principles Example. Reader and Writer

A Proof (5/6)



A Proof (6/6)

system RdWr	events
variables $r, w \in \mathbb{Z}, \mathbb{Z}$	read $\widehat{=}$ $r \neq w \longrightarrow r := r + 1$
initially $r = 0 \land w = 0$	write $\hat{=} w < r + 3 \longrightarrow w := w + 1$



EXAM Bilgrinnhar Indexider Holsebuls Zifch Seiss Feine Indexider Holsebulg Zifch		Information Secu	URITY
Thai Son Hoang (ETH-Zürich)	Proof Rules	Ascona Meeting, 03/02/11	25 / 29

The Language of Temporal Logic Proof Principles Example. Reader and Writer Conclusions	

Summary

• Proof rules for certain classes of invariance and liveness properties.

Proof Rules

- The proof rules based on the reasoning about:
 - the system leads from ϕ_1 to ϕ_2
 - the system is convergence when ϕ holds
 - the system is deadlock-free when ϕ holds.

The Language of Temporal Logic	
Proof Principles	
Example. Reader and Writer	
Conclusions	

Further Directions

- Proofs become tedious when the system becomes large.
- Refinement helps to reduce the complexity.
 - Invariance properties are maintained.
 - How about liveness?
- Concurrent systems: fairness assumptions.
 - Expect some weaker rules.
 - Interaction with refinement?



27 / 29



Appendix For Further Reading

For Further Reading I

- Zohar Manna and Amir Pnueli. Adequate Proof Principles for Invariance and Liveness Properties of Concurrent Programs. *Science of Computer Programming* 4:259-289, 1984.
- Zohar Manna and Amir Pnueli. Completing the Temporal Picture. *Theoretical Computer Science* 81(1):97-130, 1991.

EFFH Bigzeissche Trehnliche Kochsbulz Zürch Swisz Feinrik Institute of Schwedigg Zurich		Information Security	18.1
Thai Son Hoang (ETH-Zürich)	Proof Rules	Ascona Meeting, 03/02/11 29 / 29	9