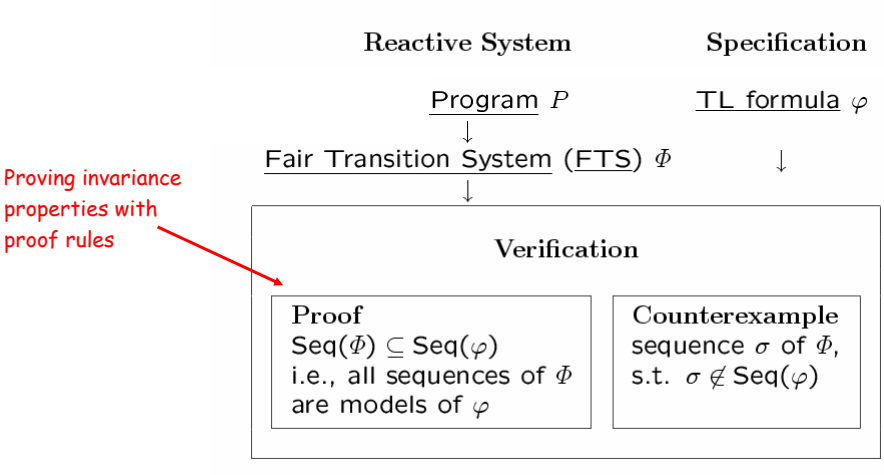
Verification – Lecture 3
Proving Invariance Properties

Bernd Finkbeiner – Sven Schewe
Rayna Dimitrova – Lars Kuhtz – Anne Proetzsch

Wintersemester 2007/2008

Goals for today



Today: For SPL program P and assertion q, show $P \models \Box q$

Review: Transition System

- A (finite) set of variables $V \subseteq \mathcal{V}$
  System variables: data variables + control variables
- Initial condition θ
  first-order assertion over the set of variables
  that characterizes all initial states
- A finite set of transitions, where each transition τ is represented by the transition relation $\rho(\tau)$
  (next-state relation)

  V values of variables in the current state

  V' values of variables in the next state

Verification Conditions

For assertions φ, ψ and transition τ,

$\{\varphi\}\ \tau\ \{\psi\}$ ("Hoare triple") stands for

$$\rho_\tau \ \wedge\ \varphi \ \rightarrow\ \psi'$$

Verification condition of φ and ψ relative to transition τ

Example:

ρ_τ: $x \geq 0 \ \wedge\ y' = x + y \ \wedge\ x' = x$

φ: $y = 3$ ψ: $y = x + 3$

Then $\{\varphi\}\ \tau\ \{\psi\}$:

$$\underbrace{x \geq 0 \ \wedge\ y' = x + y \ \wedge\ x' = x}_{\rho_\tau} \ \wedge\ \underbrace{y = 3}_{\varphi} \ \rightarrow\ \underbrace{y' = x' + 3}_{\psi'}$$

Verification Conditions

`Claim` (Verification Condition)

If $\{\varphi\}\tau\{\psi\}$ is P-state valid,
then every τ-successor of a P-accessible φ-state is a ψ-state

- for $\tau \in \mathcal{T}$ in P

  $\{\varphi\}\tau\{\psi\}$: $\quad \rho_\tau \wedge \varphi \rightarrow \psi'$

  "τ leads from φ to ψ in P"

- for $\mathcal{T}$ in P

  $\{\varphi\}\mathcal{T}\{\psi\}$: $\quad \{\varphi\}\tau\{\psi\}$ for every $\tau \in \mathcal{T}$

  "$\mathcal{T}$ leads from φ to ψ in P"

Special Cases

- while, conditional $\quad \rho_\tau$: $\rho_\tau^{\mathrm{T}} \vee \rho_\tau^{\mathrm{F}}$

$$\{\varphi\}\tau^{\mathrm{T}}\{\psi\}: \quad \rho_\tau^{\mathrm{T}} \wedge \varphi \rightarrow \psi'$$
$$\{\varphi\}\tau^{\mathrm{F}}\{\psi\}: \quad \rho_\tau^{\mathrm{F}} \wedge \varphi \rightarrow \psi'$$

$$\{\varphi\}\tau\{\psi\} \;:\; \{\varphi\}\tau^{\mathrm{T}}\{\psi\} \wedge \{\varphi\}\tau^{\mathrm{F}}\{\psi\}$$

- idle

$$\{\varphi\}\tau_I\{\varphi\}: \quad \rho_{\tau_I} \wedge \varphi \rightarrow \varphi'$$

always valid, since

$$\rho_{\tau_I} \rightarrow v' = v \qquad \text{for all } v \in V$$

Simplification by Partial Substitution

For given τ,

suppose

- $V = \overline{y} \cup \overline{v}$ where
  $\overline{y}$ are the τ-modifiable variables
  $\overline{v}$ are the τ-preserved variables

- The transition relation

  $$\rho_\tau: \quad C_\tau \ \wedge \ \overline{y}' = \overline{e} \ \wedge \ pres(\overline{v})$$

  where C_τ is the enabled condition of τ,
  $\overline{e}$ is an expression list.

Then

- simplified verification condition $\{\varphi\}\tau\{\psi\}$:

  $$C_\tau \ \wedge \ \overline{y}' = \overline{e} \ \wedge \ \varphi \ \rightarrow \ \psi[\overline{y}'/\overline{y}]$$

$\overline{y}'/\overline{y}$
Replace each variable in $\overline{y}$ with its primed version.

Example (Blackboard)

$$\rho_\tau: \ x \geq 0 \ \wedge \ y' = x + y \ \wedge \ x' = x$$

$$\varphi: \ y = 3 \qquad \psi: \ y = x + 3$$

Standard

$$\underbrace{x \geq 0 \ \wedge \ y' = x + y \ \wedge \ x' = x}_{\rho_\tau} \ \wedge \ \underbrace{y = 3}_{\varphi} \rightarrow \underbrace{y' = x' + 3}_{\psi'}$$

Simplified

$$\underbrace{x \geq 0}_{C_\tau} \ \wedge \ \underbrace{y' = x + y}_{\overline{y}' = \overline{e}} \ \wedge \ \underbrace{y = 3}_{\varphi} \rightarrow \underbrace{y' = x + 3}_{\psi[\overline{y}'/\overline{y}]}$$

Simplifying Control Expressions

$move(L_1, L_2)$: $\quad L_1 \subseteq \pi \ \wedge \ \pi' = (\pi - L_1) \cup L_2$

Consequences

- for every $[\ell] \in L_1$
  $at\_\ell = \mathrm{T}$ (i.e., $[\ell] \in \pi$)
- for every $[\ell] \in L_2$
  $at'\_\ell = \mathrm{T}$ (i.e., $[\ell] \in \pi'$)
- for every $[\ell] \in L_1 - L_2$
  $at'\_\ell = \mathrm{F}$ (i.e., $[\ell] \notin \pi'$)
- for every $\ell \notin L_1 \cup L_2$
  $at'\_\ell = at\_\ell$ (i.e., $[\ell] \in \pi, \pi'$ or $[\ell] \notin \pi, \pi'$)

Review: Runs

Infinite sequence of states

$$\sigma: s_0, s_1, s_2, \ldots$$

is a run of a transition system, if it satisfies the following:

- Initiality: s_0 satisfies θ
- Consecution: For each i= 0,1, …

there is a transition $\tau \in \mathcal{T}$ s.t. $s_{i+1} \in \tau(s_i)$

Proving Invariance

For assertion q,

$$\text{B1.} \quad P \Vdash \Theta \rightarrow q$$

$$\text{B2.} \quad P \Vdash \{q\}\ \mathcal{T}\ \{q\}$$

$$P \vDash \Box q \qquad \text{B-INV}$$

where B2 stands for

$P \Vdash \{q\}\ \tau\ \{q\}$ for every $\tau \in \mathcal{T}$

Example: REQUEST-RELEASE

$$\textbf{local}\ \ x\text{: integer where } x = 1$$

$$\begin{bmatrix} \ell_0: & \textbf{request } x \\ \ell_1: & \textbf{critical} \\ \ell_2: & \textbf{release } x \\ \ell_3: & \end{bmatrix}$$

$$P \vDash \Box \underbrace{x \geq 0}_{q}$$

Θ: $x = 1 \ \wedge \ \pi = \{\ell_0\}$

$\mathcal{T}$: $\{\tau_I, \tau_{\ell_0}, \tau_{\ell_1}, \tau_{\ell_2}\}$

Example:REQUEST-RELEASE (Blackboard)

B1: $\underbrace{x = 1 \wedge \pi = \{\ell_0\}}_{\Theta} \rightarrow \underbrace{x \geq 0}_{q}$

B2:

τ_I: $\underbrace{x \geq 0}_{q} \wedge \underbrace{x' = x \wedge \pi' = \pi}_{\rho_{\tau_I}} \rightarrow \underbrace{x' \geq 0}_{q'}$

τ_{ℓ_0}: $\underbrace{x \geq 0}_{q} \wedge \underbrace{move(\ell_0, \ell_1) \wedge x > 0 \wedge x' = x - 1}_{\rho_{\tau_{\ell_0}}} \rightarrow \underbrace{x' \geq 0}_{q'}$

τ_{ℓ_1}: $\underbrace{x \geq 0}_{q} \wedge \underbrace{move(\ell_1, \ell_2) \wedge x' = x}_{\rho_{\tau_{\ell_1}}} \rightarrow \underbrace{x' \geq 0}_{q'}$

τ_{ℓ_2}: $\underbrace{x \geq 0}_{q} \wedge \underbrace{move(\ell_2, \ell_3) \wedge x' = x + 1}_{\rho_{\tau_{\ell_2}}} \rightarrow \underbrace{x' \geq 0}_{q'}$

Example: REQUEST-RELEASE

local x: **integer** **where** $x = 1$

$$\left[\begin{array}{ll} \ell_0: & \textbf{request } x \\ \ell_1: & \textbf{critical} \\ \ell_2: & \textbf{release } x \\ \ell_3: & \end{array}\right]$$

$P \models \square \underbrace{x \geq 0}_{q}$ √

Let's try

$P \models \square \underbrace{(at\_\ell_1 \rightarrow x = 0)}_{q}$

Attempt #1 (Blackboard)

B1: $\underbrace{x = 1 \wedge \pi = \{\ell_0\}}_{\Theta} \rightarrow \underbrace{at\_\ell_1 \rightarrow x = 0}_{q}$

holds since $\pi = \{\ell_0\} \rightarrow at\_\ell_1 = \mathrm{F}$

B2: $\{q\}\tau_{\ell_0}\{q\}$

$$\underbrace{at\_\ell_1 \rightarrow x = 0}_{q} \wedge \underbrace{move(\ell_0, \ell_1) \wedge x > 0 \wedge x' = x - 1}_{\rho_{\tau_{\ell_0}}}$$

$$\rightarrow \underbrace{at'\_\ell_1 \rightarrow x' = 0}_{q'}$$

we have $move(\ell_0, \ell_1) \rightarrow at'\_\ell_1 = \mathrm{T}$

BUT

$(at\_\ell_1 \rightarrow x = 0) \wedge x > 0 \wedge x' = x - 1 \rightarrow x' = 0$

Cannot prove: not state-valid

Inductive Assertions

For assertion q,

$$\frac{\text{B1.}\quad P \models \Theta \rightarrow q \qquad \text{B2.}\quad P \models \{q\}\ \mathcal{T}\ \{q\}}{P \vdash \Box q}$$

B-INV

- q is inductive if B1 and B2 are (state) valid
- By rule B-INV,
  every inductive assertion q is P-invariant
- The converse is not true

Example: REQUEST-RELEASE

$$P \models \square \underbrace{(at\_\ell_1 \rightarrow x = 0)}_{q}$$

q is not inductive.

B2: $\{q\}\tau_{\ell_0}\{q\}$

$$\underbrace{at\_\ell_1 \rightarrow x = 0}_{q} \wedge \underbrace{move(\ell_0, \ell_1) \wedge x > 0 \wedge x' = x - 1}_{\rho_{\tau_{\ell_0}}}$$

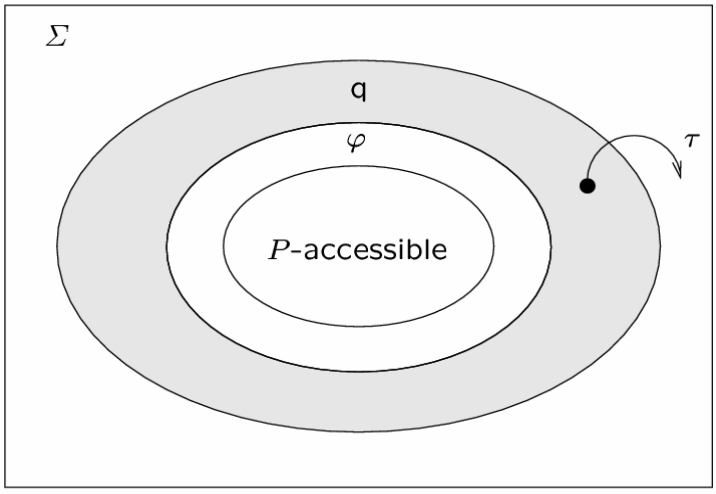
$$\rightarrow \underbrace{at'\_\ell_1 \rightarrow x' = 0}_{q'}$$

is not state valid, but it is P-state valid.

Two strategies:

1. Strengthening
2. Incremental Proofs

Strategy 1: Strengthening



Find a stronger assertion φ that is inductive and implies the assertion q we want to prove.

Rules

For assertions q_1, q_2,

$$\frac{P \models \Box q_1 \qquad P \Vdash q_1 \rightarrow q_2}{P \models \Box q_2}$$

MON-I

For assertions q, φ

I1. $P \Vdash \varphi \rightarrow q$

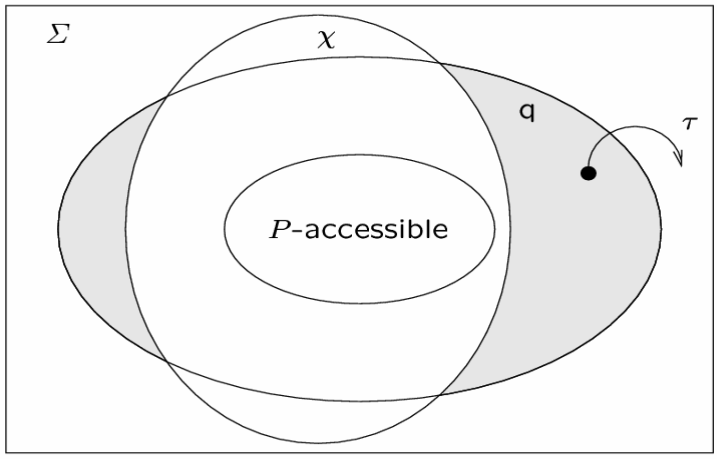
I2. $P \Vdash \Theta \rightarrow \varphi$

I3. $P \Vdash \{\varphi\}\ \mathcal{T}\ \{\varphi\}$

$$\frac{}{P \models \Box q}$$

INV

Strategy 2: Incremental Proofs



Use previously proven invariances χ to exclude parts of the state space from consideration.

Rules

For assertions q_1, q_2 and χ,

$$\frac{P \models \Box\chi \qquad \Vdash \chi \wedge q_1 \rightarrow q_2}{P \models \Box(q_1 \rightarrow q_2)}$$

SV-PSV

For assertions q_1 and q_2,

$$\frac{P \models \Box q_1 \qquad P \models \Box q_2}{P \models \Box(q_1 \wedge q_2)}$$

I-CON

Control Invariants

- CONFLICT:
  for labels ℓ_i, ℓ_j that are in conflict
  (i.e., not $\sim_L$, not parallel):
  $\Box \neg(at\_\ell_i \wedge at\_\ell_j)$

- SOMEWHERE:
  for the set of labels $\mathcal{L}_i$ in a
  top-level process:
  $\Box \bigvee_{\ell \in \mathcal{L}_i} at\_\ell$

Control Invariants

- EQUAL:
  for labels l, m, s.t. $l \sim_L m$:
  $\Box(at\_\ell \leftrightarrow at\_m)$

- PARALLEL:
  for substatement $[S_1||S_2]$:
  $\Box(in\_S_1 \leftrightarrow in\_S_2)$
  i.e, if control is in S_1 it must also be in S_2 and vice versa.

Example: REQUEST-RELEASE

local x: **integer** **where** $x = 1$

$$\left[\begin{array}{ll} \ell_0: & \textbf{request } x \\ \ell_1: & \textbf{critical} \\ \ell_2: & \textbf{release } x \\ \ell_3: & \end{array}\right]$$

$$P \models \Box \underbrace{(at\_\ell_1 \rightarrow x = 0)}_{q}$$

inductive assertion that implies

$$q: \; at\_\ell_1 \rightarrow x = 0$$

is

$$\varphi: \; (at\_\ell_1 \rightarrow x = 0) \wedge (at\_\ell_0 \rightarrow x = 1)$$

Example (cont'd) (Blackboard)

Consider $\{\varphi\}\ \tau_{\ell_0}\ \{\varphi\}$:

$$\underbrace{(at\_\ell_0 \rightarrow x = 1)\ \wedge\ (at\_\ell_1 \rightarrow x = 0)}_{\varphi}\ \wedge$$

$$\underbrace{move(\ell_0, \ell_1) \wedge x > 0 \wedge x' = x - 1}_{\rho_{\tau_{\ell_0}}}$$

$$\rightarrow\ \underbrace{(at'\_\ell_0 \rightarrow x' = 1)\ \wedge\ (at'\_\ell_1 \rightarrow x' = 0)}_{\varphi'}$$

$move(\ell_0, \ell_1)$ implies $\ell_0 \in \pi, \ell_0 \notin \pi', \ell_1 \in \pi'$

Therefore

$$(\mathrm{T} \rightarrow x = 1) \wedge \ldots \wedge x' = x - 1 \wedge x > 0$$
$$\rightarrow\ (\mathrm{F} \rightarrow \ldots) \wedge (\mathrm{T} \rightarrow x' = 0)$$

holds.

Example (cont'd) (Blackboard)

Consider $\{\varphi\}\ \tau_{\ell_2}\ \{\varphi\}$:

$$\underbrace{(at\_\ell_0 \rightarrow x = 1)\ \wedge\ (at\_\ell_1 \rightarrow x = 0)}_{\varphi}\ \wedge$$

$$\underbrace{move(\ell_2, \ell_3) \wedge \quad x' = x + 1}_{\rho_{\tau_{\ell_2}}}$$

$$\rightarrow\ \underbrace{(at'\_\ell_0 \rightarrow x' = 1)\ \wedge\ (at'\_\ell_1 \rightarrow x' = 0)}_{\varphi'}$$

$move(\ell_2, \ell_3)$ implies $\ell_2 \in \pi, \ell_2 \notin \pi', \ell_3 \in \pi'$
and by CONFLICT invariants $\ell_0, \ell_1 \notin \pi'$.

Therefore

$$\ldots \wedge \ldots\ \rightarrow\ (\mathrm{F} \rightarrow x' = 1) \wedge (\mathrm{F} \rightarrow x' = 0)$$

holds.

Trivial Verification Conditions

- Ignore $\{\varphi\}\ \tau_I\ \{\varphi\}$ – always true
- Ignore $\{\varphi\}\ \tau\ \{\varphi\}$
  if τ does not modify any variable in φ
- For $\{\varphi\}\ \tau\ \{\varphi\}$ where φ: $p \rightarrow q$

$$\rho_\tau \ \wedge \ \underbrace{p \rightarrow q}_{\varphi} \ \rightarrow \ \underbrace{p' \rightarrow q'}_{\varphi'}$$

Consider only τ's that
validate p or falsify q

Example: MUX-SEM

$$\textbf{local } y\textbf{: integer where } y = 1$$

$$P_1 :: \left[\begin{array}{l} \ell_0\textbf{: loop forever do} \\ \quad \left[\begin{array}{l} \ell_1 : \textbf{noncritical} \\ \ell_2 : \textbf{request } y \\ \ell_3 : \textbf{critical} \\ \ell_4 : \textbf{release } y \end{array}\right] \end{array}\right] \ || \ P_2 :: \left[\begin{array}{l} m_0\textbf{: loop forever do} \\ \quad \left[\begin{array}{l} m_1\textbf{: noncritical} \\ m_2\textbf{: request } y \\ m_3\textbf{: critical} \\ m_4\textbf{: release } y \end{array}\right] \end{array}\right]$$

Prove mutual exclusion

$$\Box \underbrace{\neg(at\_\ell_3 \ \wedge \ at\_m_3)}_{q}$$

Example (cont'd)

$$\text{local } y\text{: integer where } y = 1$$

$$P_1 :: \left[\begin{array}{l} \ell_0\text{: loop forever do} \\ \left[\begin{array}{l} \ell_1 : \text{noncritical} \\ \ell_2 : \text{request } y \\ \ell_3 : \text{critical} \\ \ell_4 : \text{release } y \end{array}\right] \end{array}\right] \;\|\; P_2 :: \left[\begin{array}{l} m_0\text{: loop forever do} \\ \left[\begin{array}{l} m_1\text{: noncritical} \\ m_2\text{: request } y \\ m_3\text{: critical} \\ m_4\text{: release } y \end{array}\right] \end{array}\right]$$

3 steps: $\square \underbrace{(y \geq 0)}_{\varphi_1}$

$\square \underbrace{(at\_\ell_{3,4} + at\_m_{3,4} + y = 1)}_{\varphi_2}$

$\square \underbrace{\neg(at\_\ell_3 \;\wedge\; at\_m_3)}_{p}$

where $\text{F} = 0, \text{T} = 1$.

Example (cont'd) (Blackboard)

$$\text{local } y\text{: integer where } y = 1$$

$$P_1 :: \left[\begin{array}{l} \ell_0\text{: loop forever do} \\ \left[\begin{array}{l} \ell_1 : \text{noncritical} \\ \ell_2 : \text{request } y \\ \ell_3 : \text{critical} \\ \ell_4 : \text{release } y \end{array}\right] \end{array}\right] \;\|\; P_2 :: \left[\begin{array}{l} m_0\text{: loop forever do} \\ \left[\begin{array}{l} m_1\text{: noncritical} \\ m_2\text{: request } y \\ m_3\text{: critical} \\ m_4\text{: release } y \end{array}\right] \end{array}\right]$$

Step 1: $\square \underbrace{(y \geq 0)}_{\varphi_1}$

by rule B-INV

B1. $\underbrace{\pi = \{\ell_0, m_0\} \;\wedge\; y = 1}_{\Theta} \;\rightarrow\; \underbrace{y \geq 0}_{\varphi_1}$

B2. $\rho_\tau \;\wedge\; y \geq 0 \;\rightarrow\; y' \geq 0$

check only ℓ_2, ℓ_4, m_2, m_4
("y-modifiable transitions")

Example (cont'd) (Blackboard)

$$\text{local } y\text{: integer where } y = 1$$

$$P_1 :: \left[\begin{array}{l} \ell_0\text{: loop forever do} \\ \left[\begin{array}{l} \ell_1 : \text{noncritical} \\ \ell_2 : \text{request } y \\ \ell_3 : \text{critical} \\ \ell_4 : \text{release } y \end{array}\right] \end{array}\right] \;\|\; P_2 :: \left[\begin{array}{l} m_0\text{: loop forever do} \\ \left[\begin{array}{l} m_1\text{: noncritical} \\ m_2\text{: request } y \\ m_3\text{: critical} \\ m_4\text{: release } y \end{array}\right] \end{array}\right]$$

ℓ_2: $\underbrace{move(\ell_2, \ell_3) \;\wedge\; y > 0 \;\wedge\; y' = y - 1}_{\rho_\tau} \;\wedge\; \underbrace{y \geq 0}_{\varphi}$

$$\rightarrow \underbrace{y' \geq 0}_{\varphi'}$$

ℓ_4: $\underbrace{move(\ell_4, \ell_0) \;\wedge\; y' = y + 1}_{\rho_\tau} \;\wedge\; \underbrace{y \geq 0}_{\varphi} \rightarrow \underbrace{y' \geq 0}_{\varphi'}$

Example (cont'd) (Blackboard)

$$\text{local } y\text{: integer where } y = 1$$

$$P_1 :: \left[\begin{array}{l} \ell_0\text{: loop forever do} \\ \left[\begin{array}{l} \ell_1 : \text{noncritical} \\ \ell_2 : \text{request } y \\ \ell_3 : \text{critical} \\ \ell_4 : \text{release } y \end{array}\right] \end{array}\right] \;\|\; P_2 :: \left[\begin{array}{l} m_0\text{: loop forever do} \\ \left[\begin{array}{l} m_1\text{: noncritical} \\ m_2\text{: request } y \\ m_3\text{: critical} \\ m_4\text{: release } y \end{array}\right] \end{array}\right]$$

Step 2:

$$\Box(\underbrace{at\_\ell_{3,4} + at\_m_{3,4} + y = 1}_{\varphi_2})$$

by rule B-INV

B1. $\underbrace{\pi = \{\ell_0, m_0\} \;\wedge\; y = 1}_{\Theta} \rightarrow$

$$\underbrace{at\_\ell_{3,4}}_{0} + \underbrace{at\_m_{3,4}}_{0} + \underbrace{y}_{1} = 1$$

Example (cont'd) (Blackboard)

B2. $\rho_\tau \wedge \varphi_2 \rightarrow \varphi_2'$

$$\rho_{\ell_0} \wedge 0 + at\_m_{3,4} + y = 1 \rightarrow 0 + at\_m_{3,4} + y = 1$$

$$\rho_{\ell_1} \wedge 0 + at\_m_{3,4} + y = 1 \rightarrow 0 + at\_m_{3,4} + y = 1$$

$$\rho_{\ell_2} \wedge 0 + at\_m_{3,4} + y = 1 \rightarrow 1 + at\_m_{3,4} + (y-1) = 1$$

$$\rho_{\ell_3} \wedge 1 + at\_m_{3,4} + y = 1 \rightarrow 1 + at\_m_{3,4} + y = 1$$

$$\rho_{\ell_4} \wedge 1 + at\_m_{3,4} + y = 1 \rightarrow 0 + at\_m_{3,4} + (y+1) = 1$$

Example (cont'd) (Blackboard)

Step 3: Show $\Box \underbrace{\neg(at\_\ell_3 \wedge at\_m_3)}_{p}$

- By I-CON

$$\frac{\Box\varphi_1,\ \Box\varphi_2}{\Box(\varphi_1 \wedge \varphi_2)}$$

- By MON-I

$$\frac{\Box(\varphi_1 \wedge \varphi_2) \qquad \underbrace{y \geq 0}_{\varphi_1} \wedge \underbrace{at\_\ell_{3,4} + at\_m_{3,4} + y = 1}_{\varphi_2} \rightarrow \underbrace{\neg(at\_\ell_3 \wedge at\_m_3)}_{p}}{\Box \underbrace{\neg(at\_\ell_3 \wedge at\_m_3)}_{p}}$$

Strengthening vs. Incremental Proof

We want to prove $\Box q$, but q is not inductive.

We have two options:

1. Strengthen it to $q \wedge \varphi$.
   Prove $\Box(q \wedge \varphi)$ and deduce $\Box q$.
2. First prove $\Box \varphi$ and then prove $\Box q$ relative to φ.

1

I1. $\Theta \rightarrow q \wedge \varphi$

I2. $\{q \wedge \varphi\}\ \mathcal{T}\ \{q \wedge \varphi\}$

2

I1. $\Theta \rightarrow \varphi$	I1. $\Theta \rightarrow q$
I2. $\{\varphi\}\ \mathcal{T}\ \{\varphi\}$	I2. $\{q \wedge \varphi\}\ \mathcal{T}\ \{q\}$
$\Box \varphi$	$\Box q$

Strengthening vs. Incremental Proofs

- 2 implies 1 since

$$\left[\begin{array}{l} \rho_\tau \wedge \varphi \rightarrow \varphi' \\ \rho_\tau \wedge q \wedge \varphi \rightarrow q' \end{array}\right] \rightarrow [\rho_\tau \wedge q \wedge \varphi \rightarrow q' \wedge \varphi']$$

- In practice, 2 is often more useful than 1
  - allows breaking down the proof in more manageable pieces
  - smaller verification conditions
  - more intuitive

Example

local x: **integer where** $x = 1$

ℓ_0: **loop forever do**

$\left[\ \ell_1 :\ x := x + 1\ \right]$

Show q_1: $at\_\ell_0 \rightarrow x > 0$

q_2: $at\_\ell_1 \rightarrow x > 0$

- both are P-valid
- neither of them is inductive
- but $q_1 \wedge q_2$ is inductive!

Combining the Strategies

For assertions $q, \varphi, \chi_1, \ldots, \chi_k$

I0. $P \models \Box\chi_1, \ldots, \Box\chi_k$

I1. $P \Vdash (\bigwedge_{i=1}^{k} \chi_i) \wedge \varphi \rightarrow q$

I2. $P \Vdash \Theta \rightarrow \varphi$

I3. $P \Vdash \{(\bigwedge_{i=1}^{k} \chi_i) \wedge \varphi\}\ \mathcal{T}\ \{\varphi\}$

$P \models \Box q$ INC-INV

Finding Inductive Invariants

Construction of inductive assertions by

1. Bottom-up methods:
   - Based on program text only
   - Algorithmic
   - Guaranteed to produce an inductive invariant
2. Top-down methods:
   - Guided by the property we want to prove
   - Heuristic
   - Not guaranteed to produce an inductive invariant

Transition-Validated Assertions

ℓ_1: [**while** c **do** S]; ℓ_2: $\boxed{at\_\ell_2 \rightarrow \neg c}$

if no statement parallel to ℓ_1 can modify variables in c

ℓ_1: $y := e$; ℓ_2: $\boxed{at\_\ell_2 \rightarrow y = e}$

if no statement parallel to ℓ_1 can modify y or variables occurring in e and if y does not occur in e.

Single-Variable Assertions

$$y = 0$$

$$\left[\begin{array}{l}\textbf{loop forever do} \\ \quad \left[\begin{array}{l}\cdots \\ \textbf{request } y \\ \cdots \\ \textbf{release } y\end{array}\right]\end{array}\right] \qquad \boxed{y \geq 0}$$

$$s = 1$$

$$\left[\begin{array}{l}\cdots \\ s := 1 \\ \cdots\end{array}\right] \;||\; \left[\begin{array}{l}\cdots \\ s := 2 \\ \cdots\end{array}\right] \qquad \boxed{s = 1 \;\vee\; s = 2}$$

where no other statement modifies s

Multi-Variable Assertions

Example: Program SQUARE-ROOT

	in	x:	**integer**	**where** $x \geq 0$
	local	u, w:	**integer**	**where** $u = 1, w = 1$
	out	z:	**integer**	**where** $z = 0$
ℓ_0 :	**while** $w \leq x$ **do**			
	ℓ_1 :	$(z,\ u,\ w)$	:=	$(z+1,\ u+2,\ w+u+2)$
ℓ_2 :				

$$at\_\ell_2 \;\rightarrow\; z^2 \leq x < (z+1)^2$$

Intuitive Argument

$$\boxed{at\_\ell_2 \;\rightarrow\; z^2 \leq x < (z+1)^2}$$

$$\begin{aligned} z &= 0, 1, \ldots, n \\ u &= 1, 3, \ldots, 2n+1 \\ w &= \underbrace{1+3+\ldots+(2n+1)}_{(n+1)^2} = (z+1)^2 \end{aligned}$$

first time $w > x$

$x < (z+1)^2$

last time $w \leq x$

$z^2 \leq x$

Thus at ℓ_2:

$z^2 \leq x < (z+1)^2$

Example (cont'd) (Blackboard)

We show ψ_2: $at\_\ell_2 \;\rightarrow\; x < (z+1)^2$

with bottom-up invariants.

$$\begin{cases} z_0 = 0 \\ z_n = z_{n-1} + 1 \quad \text{for } n > 0 \end{cases}$$

$$\begin{cases} u_0 = 1 \\ u_n = u_{n-1} + 2 \quad \text{for } n > 0 \end{cases}$$

$$\begin{cases} w_0 = 1 \\ w_n = w_{n-1} + u_{n-1} + 2 \quad \text{for } n > 0 \end{cases}$$

Example (cont'd) (Blackboard)

- Step 1

$$\left.\begin{array}{ll} z_n = n & \text{for } n \geq 0 \\ u_n = 2n+1 & \text{for } n \geq 0 \end{array}\right\} \Rightarrow \qquad \boxed{\varphi_1\colon\ u = 2z+1}$$

- Step 2

$$\begin{cases} w_0 = 1 \\ w_n = w_{n-1} + \overbrace{(2(n-1)+1)}^{u_{n-1}} + 2 \\ \quad\ = w_{n-1} + (2n+1) \qquad \text{for } n \geq 0 \end{cases}$$

$$w_n = \sum_{k=0}^{n} (2k+1) = (n+1)^2 \quad \text{for } n \geq 0$$

$$\boxed{\varphi_2\colon\ w = (z+1)^2}$$

$$w_n = (z_n+1)^2 \quad \text{for } n \geq 0$$

Example (cont'd) (Blackboard)

$$\boxed{\varphi_1\colon\ u = 2z+1}$$

$$\boxed{\varphi_2\colon\ w = (z+1)^2}$$

- Step3

$$\boxed{at\_\ell_2 \ \rightarrow\ x < w}$$

Therefore

$$\boxed{\psi_2\colon\ at\_\ell_2 \ \rightarrow\ x < (z+1)^2}$$

Top-Down Approach

previously proven $\Box\chi$

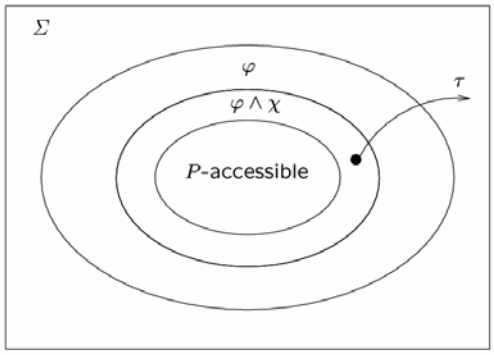
want to prove $\Box\varphi$

but

$$\{\chi \wedge \varphi\}\tau\{\varphi\}$$

not state-valid

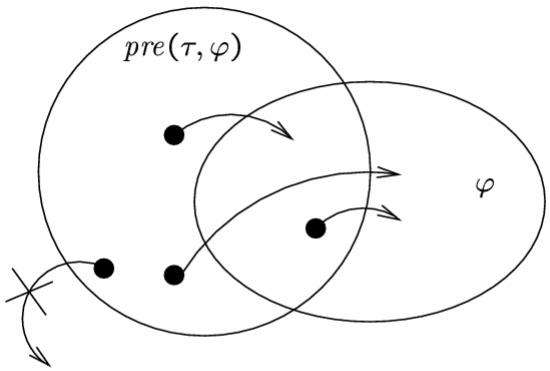
for some $\tau \in \mathcal{T}$.



Solution: Take the largest set of states that will result in a φ-state when τ is taken.

Precondition

$pre(\tau, \varphi)$: $\forall V' . \rho_\tau \rightarrow \varphi'$



a state s satisfies $pre(\tau, \varphi)$

iff

all τ-successors of s satisfy φ.

Example

$V : \{x\}$

$\rho_\tau : x > 0 \ \wedge \ x' = x - 1$

$\varphi : x \geq 2$

$pre(\tau, \varphi)$:

$$\forall x' . \ \underbrace{x > 0 \ \wedge \ x' = x - 1}_{\rho_\tau} \ \rightarrow \ \underbrace{x' \geq 2}_{\varphi'}$$

$$x > 0 \ \rightarrow \ x - 1 \geq 2$$

$$x \leq 0 \ \vee \ x \geq 3$$