CS256/Spring 2008 — Lecture #6

Zohar Manna

Chapter 1

Invariance: Proof Methods

For assertion q
and SPL program P

show $\quad P \vDash \square \, q$
$\quad\quad$ (i.e., q is P-invariant)

Proving Invariances

Definitions

Recall:

- the <u>variables of assertion</u>:
    - free (flexible) system variables

$$V = Y \cup \{\pi\}$$

    where Y are the program variables and π is the control variable

    - quantified (rigid) specification variables

- q' is the <u>primed version</u> of q, obtained by replacing each free occurrence of a system variable $y \in V$ by its primed version y'.

- ρ_τ is the <u>transition relation</u> of τ, expressing the relation holding between a state s and any of its τ-successors $s' \in \tau(s)$.

Verification Conditions
(proof obligations)

standard verification condition

For assertions φ, ψ and transition τ,

$\{\varphi\} \, \tau \, \{\psi\}$ ("Hoare triple") stands for the state formula

$$\boxed{\rho_\tau \; \wedge \; \varphi \; \rightarrow \; \psi'}$$

"Verification condition (VC) of φ and ψ
relative to transition τ"

$$
\begin{array}{ccc}
\varphi & \tau & \psi \\
\rule{0pt}{1ex} & \rule{8em}{0.4pt} & \\
j & & j+1
\end{array}
$$

Verification Conditions (Con't)

Example:

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

φ: $y = 3$ $\qquad$ 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 (Con't)

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

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

  "τ <u>leads from</u> φ to ψ in P"

- for $\mathcal{T}$ in P

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

  "$\mathcal{T}$ <u>leads from</u> φ to ψ in P"

`Claim` (Verification Condition)

If $\{\varphi\}\tau\{\psi\}$ is P-state valid,

  then every τ-successor of a φ-state is a ψ-state.

Verification Conditions (Con't)

Special Cases

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

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

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

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

- idle

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

always valid, since

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

so $\varphi' = \varphi$.

Verification Conditions (Con't)

<u>Substituted Form of Verification Condition</u>

Transition relation can be written as
$$\rho_\tau: \quad C_\tau \;\wedge\; (\overline{V}' = \overline{E})$$

where
C_τ: enabling condition
$\overline{V}'$:   primed variable list
$\overline{E}$:   expression list

- The substituted form of
  verification condition $\{\varphi\}\tau\{\psi\}$:

$$\boxed{C_\tau \;\wedge\; \varphi \;\rightarrow\; \psi[\overline{E}/\overline{V}]}$$

   where
$\psi[\overline{E}/\overline{V}]$:   replace each variable $v \in \overline{V}$
       in ψ by the corresponding $e \in \overline{E}$
Note: No primed variables!

> The substituted form of a verification condition is P-state valid iff the standard form is

Verification Conditions (Con't)

Example:

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

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

Standard

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

$$\rightarrow \; \underbrace{y' = x' + 3}_{\psi'}$$

Substituted

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

Verification Conditions (Con't)

Example:

$$\varphi: \ x = y \qquad \psi: \ x = y + 1$$

$$\rho_\tau: \ \underbrace{x \ \geq \ 0}_{C_\tau} \ \wedge \ \underbrace{(x', y')}_{\overline{V'}} = \underbrace{(x + 1, y)}_{\overline{E}}$$

The substituted form of $\{\varphi\}\tau\{\psi\}$ is

$$\underbrace{x \geq 0}_{C_\tau} \ \wedge \ \underbrace{x = y}_{\varphi} \ \rightarrow$$

$$\underbrace{(x = y + 1)[(x + 1, y)/(x, y)]}_{\psi[\overline{E}/\overline{V}]}$$

or equivalently

$$x \geq 0 \ \wedge \ x = y \ \rightarrow \ x + 1 = y + 1$$

Simplifying Control Expressions

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

e.g., for $L_1 = \{\ell_1\}, L_2 = \{\ell_2\}$
$move(\ell_1, \ell_2): \quad \ell_1 \in \pi \ \wedge \ \pi' = (\pi - \{\ell_1\}) \cup \{\ell_2\}$

Consequences implied by $move(L_1, L_2)$:

- for every $[\ell] \in L_1$
  $at\_\ell = \text{T}$ (i.e., $[\ell] \in \pi$)

- for every $[\ell] \in L_2$
  $at'\_\ell = \text{T}$ (i.e., $[\ell] \in \pi'$)

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

Proving invariance properties: $P \vDash \Box \, q$

We <u>want to show</u> that for every
computation of P

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

assertion q holds in every state s_j, $j \geq 0$,
i.e., $s_j \Vvdash q$.

Recall:

A sequence $\sigma : s_0, s_1, s_2, \ldots$ is a <u>computation</u>
if the following hold (from Chapter 0):

1. Initiality: $s_0 \Vvdash \Theta$

2. Consecution: For each $j \geq 0$,
   s_{j+1} is a τ-successor of s_j for some $\tau \in \mathcal{T}$
   $(s_{j+1} \in \tau(s_j))$

3, 4. Fairness conditions are respected.

Note: Truth of *safety* properties over programs
does not depend on fairness conditions.

Proving invariance properties (Con't)

This definition suggests a way to prove
invariance properties $\Box\, q$:

1. Base case:
   Prove that q holds initially

   $$\Theta \rightarrow q$$

   i.e., q holds at s_0.

2. Inductive step:
   prove that q is preserved by all transitions

   $$\underbrace{q \wedge \rho_\tau \rightarrow q'}_{\{q\}\tau\{q\}} \quad \text{for all } \tau \in \mathcal{T}$$

   i.e., if q holds at s_j, then it holds at every τ-successor s_{j+1}.

Rule B-INV (basic invariance)

Show $P \vDash \square\, q$ (i.e. q is <u>P-invariant</u>)

For assertion q,

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

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

$$P \vDash \square\, q$$

where B2 stands for

$$\boxed{P \Vdash \{q\}\, \tau\, \{q\} \;\; \text{for every } \tau \in \mathcal{T}}$$

- The rule states that if we can prove the
  P-state validity of $\Theta \rightarrow q$ and $\{q\}\mathcal{T}\{q\}$
  then we can conclude that $\square\, q$ is P-valid.

- Thus the proof of a temporal property
  is reduced to the proof of $1 + |\mathcal{T}|$
  first-order verification conditions.

Example 1: REQUEST-RELEASE

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

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

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

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

Prove

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

using B-INV.

Example 1: request-release (Con't)

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

  holds since $x = 1 \rightarrow x \geq 0$

B2:

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

  holds since $x > 0 \rightarrow x - 1 \geq 0$

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

  holds since $x \geq 0 \rightarrow x \geq 0$

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

  holds since $x \geq 0 \rightarrow x + 1 \geq 0$

Example 1: request-release (Con't)

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

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

We proved

$$P \vDash \square \, x \geq 0$$

using B-INV.

Now we want to prove

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

Example 1: request-release (Con't)

Attempted proof:

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

BUT

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

Cannot prove: $\boxed{\text{not state-valid}}$

What is the problem?

We need a stronger rule.

Strategies for invariance proofs

Rule B-INV (basic invariance)

For assertion q,

$$\text{B1.} \qquad P \Vvdash \Theta \rightarrow q$$

$$\text{B2.} \qquad P \Vvdash \{q\} \, \mathcal{T} \, \{q\}$$

$$\overline{\qquad\qquad\qquad\qquad\qquad}$$

$$P \vDash \square \, q$$

- q is <u>inductive</u> if B1 and B2 are (state) valid

- By rule B-INV,
  <u>every inductive assertion q is P-invariant</u>

- <u>The converse is not true</u>

Example: In REQUEST-RELEASE

$$at\_\ell_1 \rightarrow x = 0$$

is P-invariant, but not inductive

Rule B-INV(Con't)

The problem is:

"The invariant is not inductive"

i.e., it is not strong enough to be preserved by all transitions.

Another way to look at it is to observe that

$$\{q\}\ \tau_{\ell_0}\ \{q\}$$

is not state valid, but it is P-state valid,
i.e., it is true in all P-accessible states,
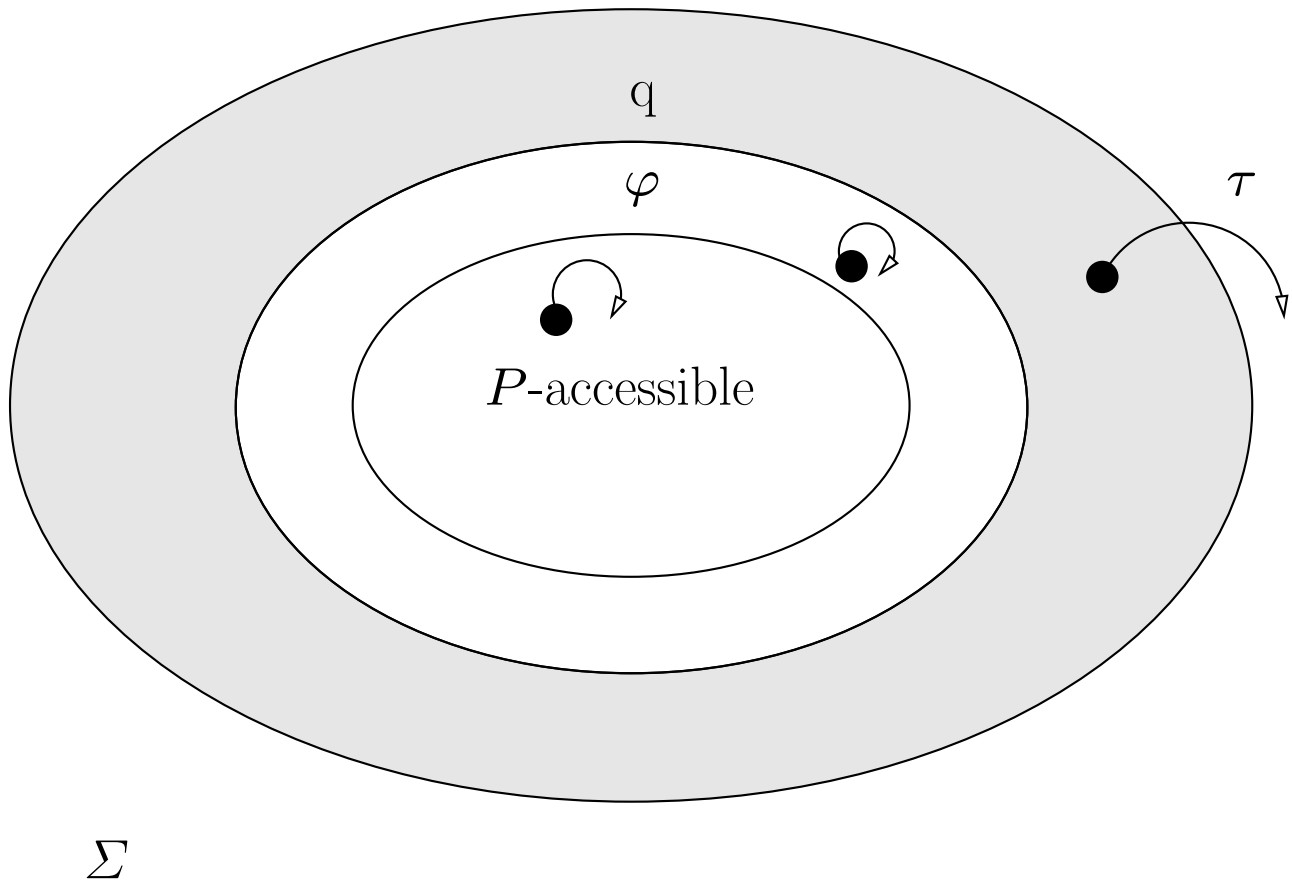since in all P-accessible states

$$x = 1 \text{ when at location } \ell_0.$$

This suggests two strategies to overcome this problem:

- strengthening
- incremental proof

Strategy 1: Strengthening

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

In Chapter 2 it will be shown that there always exists
such an assertion φ.

Strategy 1: Strengthening (Con't)

Example:

To show

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

strengthen q to

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

and show

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

by rule B-INV.

Strategy 1: Strengthening (Con't)

The strengthening strategy relies on the following rule, MON-I, which, combined with B-INV leads to the general invariance rule INV.

Rule MON-I (Monotonicity)

For assertions q_1, q_2,

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

Strategy 1: Strengthening (Con't)

<u>Rule INV</u> (general invariance)

For assertions q, φ

$$\text{I1.} \qquad P \Vdash \varphi \;\rightarrow\; q$$

$$\text{I2.} \qquad P \Vdash \Theta \;\rightarrow\; \varphi$$

$$\text{I3.} \qquad P \Vdash \{\varphi\}\; \mathcal{T} \;\{\varphi\}$$

$$\rule{6cm}{0.4pt}$$

$$P \vDash \square\, q$$

Soundness: If we manage to prove $\square\, q$ using the INV rule for some program P, is q really an invariant for the program?

We can prove that this is indeed the case. So INV rule is *sound*.

Completeness: What if q is an invariant for a program P but there is **no** way of proving it under the INV rule?

We can prove that this never happens. There always exists an appropriate φ. In other words INV rule is *complete*.

Strategy 1: Strengthening (Con't)

Motivation:

$$P \vDash \Box \, \varphi \qquad \text{(by I2 and I3)}$$

$$P \Vdash \varphi \rightarrow q \quad \text{(by I1)}$$

Therefore,

$$P \vDash \Box \, q \qquad \text{(by MON-I)}$$

i.e., this rule requires that $\Box \, \varphi$ holds and φ implies q, then $\Box \, q$ can be concluded to hold by monotonicity.

Control Invariants

Some control invariants that can always be used (without mentioning them)

- 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:
  $$\Box \bigvee_{\ell \in \mathcal{L}_i} at\_\ell$$

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

Control Invariants (Con't)

- 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:

Using the invariant CONFLICT,

$move(\ell_2, \ell_3)$ implies $l_0 \notin \pi,\ l_1 \notin \pi,\ l_3 \notin \pi$
$l_0 \notin \pi',\ l_1 \notin \pi',\ l_2 \notin \pi'$

Strategy 1: Strengthening (Con't)

`Example:`

We proposed the strengthened invariant

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

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 $\ell_0 \in \pi, \ell_1 \notin \pi, \ell_1 \in \pi', \ell_0 \notin \pi'$

Therefore

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

$$\rightarrow (\text{F} \rightarrow \ldots) \wedge (\text{T} \rightarrow x' = 0)$$

holds.

Strategy 1: Strengthening (Con't)

Example (Con't):

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

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

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

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

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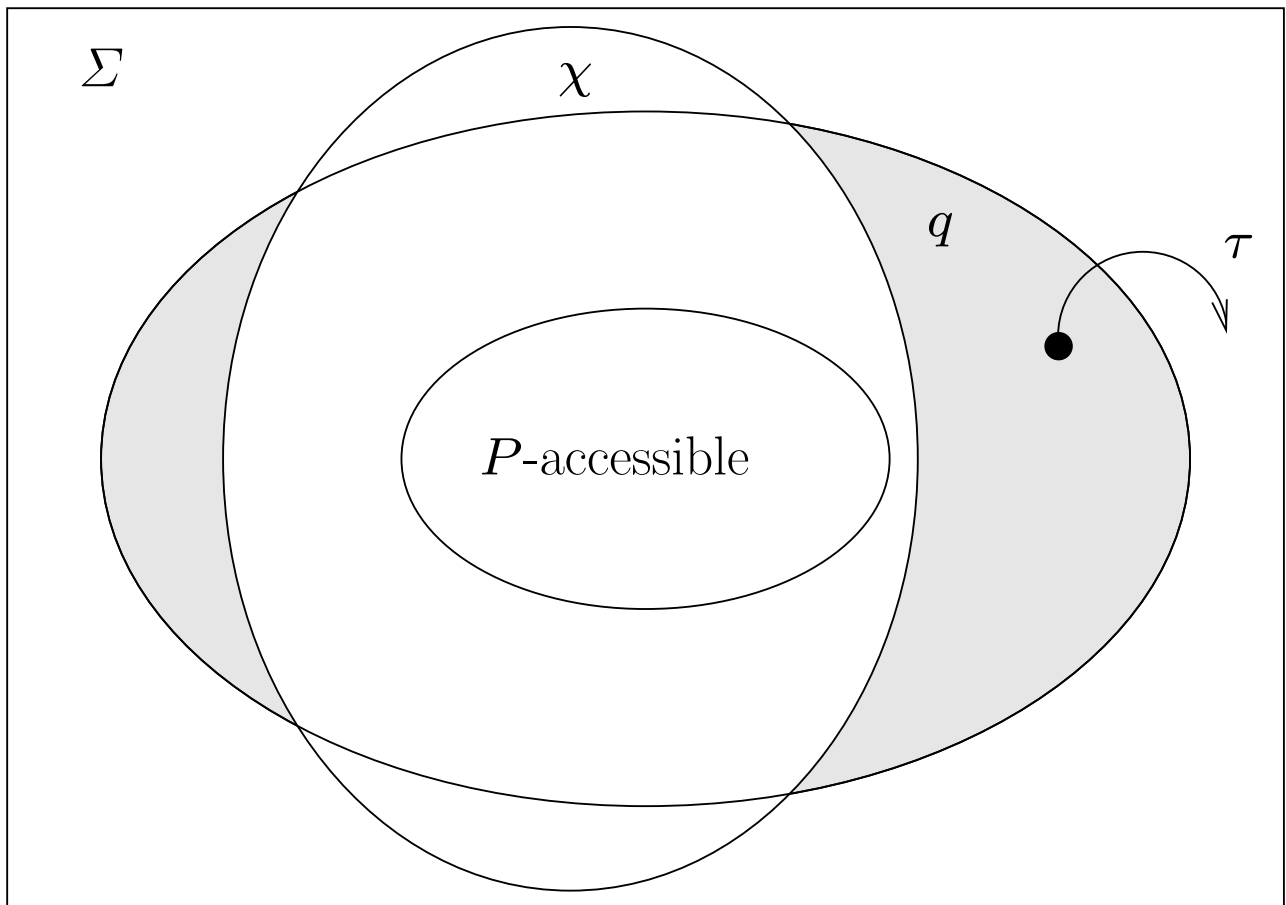and by CONFLICT invariants $\ell_0, \ell_1 \notin \pi'$.

Therefore

$$\ldots \wedge \ldots \to (\text{F} \to x' = 1) \wedge (\text{F} \to x' = 0)$$

holds.

$\{\varphi\} \tau_{\ell_2} \{\varphi\}$ is not state-valid,
but it is P-state valid. Why?

Strategy 2: Incremental proof

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

Strategy 2: Incremental proof (Con't)

Example:

To show

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

prove first (separately) by rule B-INV

$$\Box \underbrace{(at\_\ell_0 \;\rightarrow\; x = 1)}_{\chi},$$

then show

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

by rule B-INV, but add the conjunct

$$at\_\ell_0 \;\rightarrow\; x = 1$$

to the antecedent of all verification conditions.

(Example continues...)

Strategy 2: Incremental proof (Con't)

Example: (cont'd)

e.g., to show $\{\chi \land q\}\tau_{\ell_0}\{q\}$, prove

$$\underbrace{at\_\ell_0 \to x = 1}_{\chi} \land \underbrace{at\_\ell_1 \to x = 0}_{q} \land$$

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

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

Strategy 2: Incremental proof (Con't)

In an incremental proof we use previously proven properties to eliminate parts of the state space (non P-accessible states) from consideration, relying on the following rules:

Rule SV-PSV: from state validities to
P-state validities

For assertions q_1, q_2 and χ,

$$P \vDash \Box\, \chi$$

$$P \Vdash \chi \wedge q_1 \;\rightarrow\; q_2$$

$$P \vDash \Box(q_1 \rightarrow q_2)$$

Rule I-CON: Conjunction

For assertions q_1 and q_2,

$$P \vDash \Box\, q_1$$

$$P \vDash \Box\, q_2$$

$$P \vDash \Box(q_1 \wedge q_2)$$

Strategy 2: Incremental proof (Con't)

`Example:` Program MUX-SEM

                 (mutual exclusion by semaphores)

local y: integer where $y = 1$

$$P_1 :: \begin{bmatrix} \ell_0\text{: loop forever do} \\ \begin{bmatrix} \ell_1 : \text{noncritical} \\ \ell_2 : \text{request } y \\ \ell_3 : \text{critical} \\ \ell_4 : \text{release } y \end{bmatrix} \end{bmatrix} \parallel P_2 :: \begin{bmatrix} m_0\text{: loop forever do} \\ \begin{bmatrix} m_1\text{: noncritical} \\ m_2\text{: request } y \\ m_3\text{: critical} \\ m_4\text{: release } y \end{bmatrix} \end{bmatrix}$$

Prove mutual exclusion

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

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

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

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

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

Let π_ℓ: $\quad \pi \cap \{\ell_0, \ldots, \ell_4\}$
$\quad \pi_m$: $\quad \pi \cap \{m_0, \ldots, m_4\}$

By control invariants (CONFLICT, SOMEWHERE and PARALLEL)

$$|\pi_\ell| = |\pi_m| = 1$$

Program MUX-SEM (Con't)

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

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")

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

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

holds since $y > 0 \;\rightarrow\; y{-}1 \geq 0$

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

holds since $y \geq 0 \;\rightarrow\; y{+}1 \geq 0$.

Similarly for m_2, m_4.

Step 2:

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

by rule B-INV

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

$$\underbrace{\underbrace{at\_\ell_{3,4}}_{0} + \underbrace{at\_m_{3,4}}_{0} + \underbrace{y}_{1} = 1}_{\varphi_2}$$

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

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

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

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

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

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

$$\text{Program MUX-SEM (Con't)}$$

Step 3: Show $P \vDash \square \underbrace{\neg(at_-\ell_3 \ \wedge \ at_-m_3)}_{q}$

- By I-CON

$$\frac{P \vDash \square \ \varphi_1, \ P \vDash \square \ \varphi_2}{P \vDash \square(\varphi_1 \ \wedge \ \varphi_2)}$$

- By MON-I

$P \vDash \square(\varphi_1 \ \wedge \ \varphi_2)$

$$P \vDash \underbrace{y \geq 0}_{\varphi_1} \ \wedge \ \underbrace{at_-\ell_{3,4} + at_-m_{3,4} + y = 1}_{\varphi_2}$$
$$\rightarrow \ \underbrace{\neg(at_-\ell_3 \ \wedge \ at_-m_3)}_{q}$$

$$\overline{P \vDash \square \underbrace{\neg(at_-\ell_3 \ \wedge \ at_-m_3)}_{q}}$$