Lecture 5

Monday, February 3, 2020 4:43 PM

CSC1699

Lecture 5.

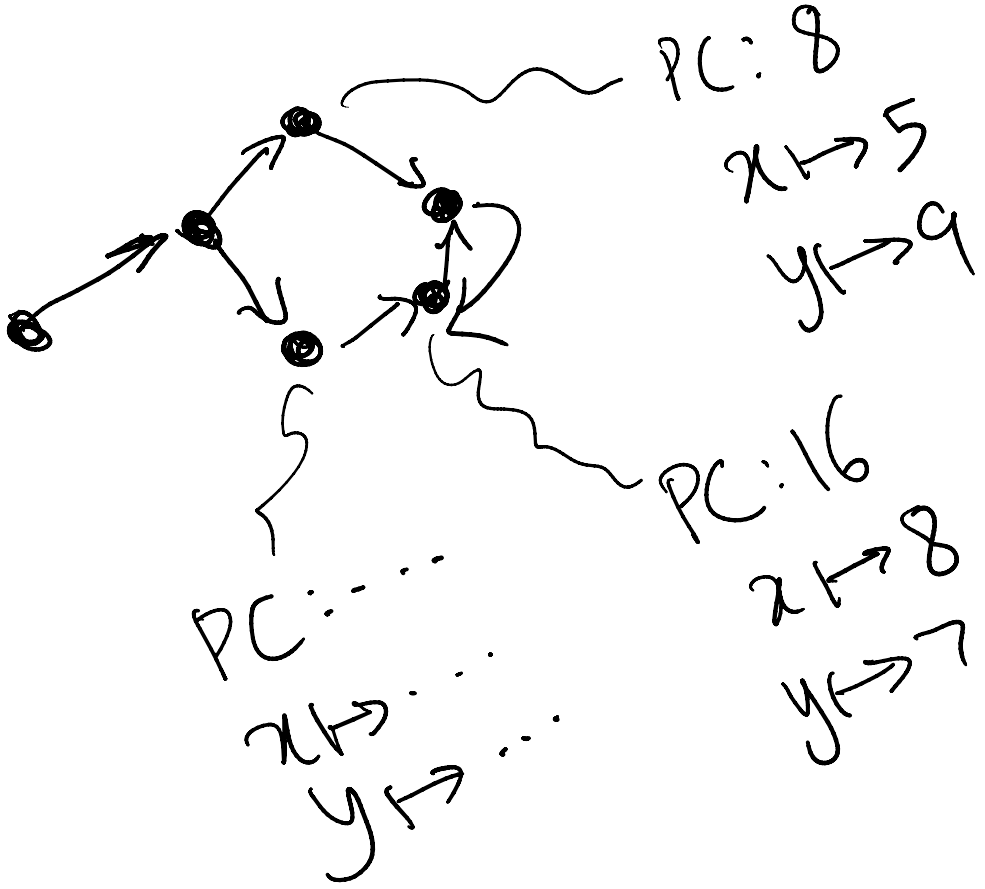
- Inductive invariant vs invariants
- Invariant generation
  - Michael Ernst et al. ICSE 1999

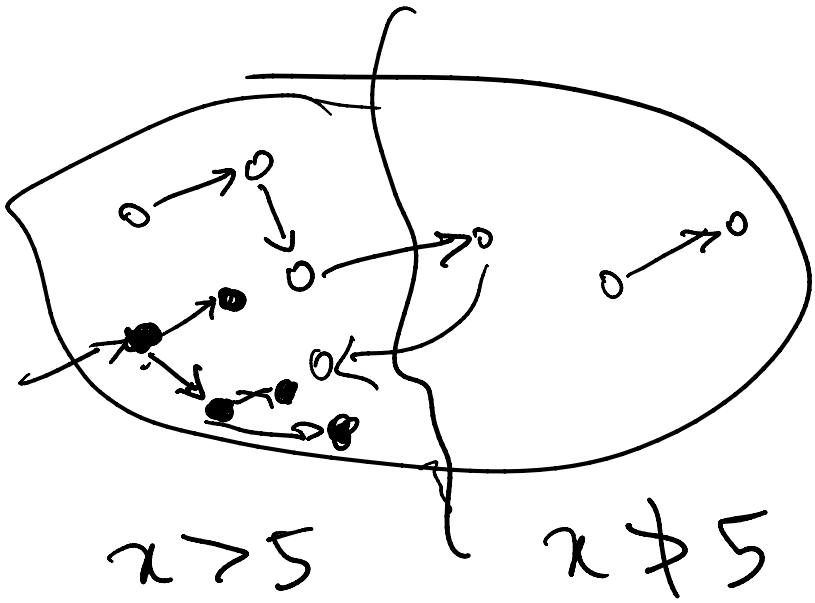
Dynamic invariant detector, Daikon.

- Houdini 2001
  Cormac Flanagan
  K Rustan M Leino

- Implication Counterexamples
  Madhusudan Parthasarathy
  Pranav Garg UIUC
  CAV 2014, -...

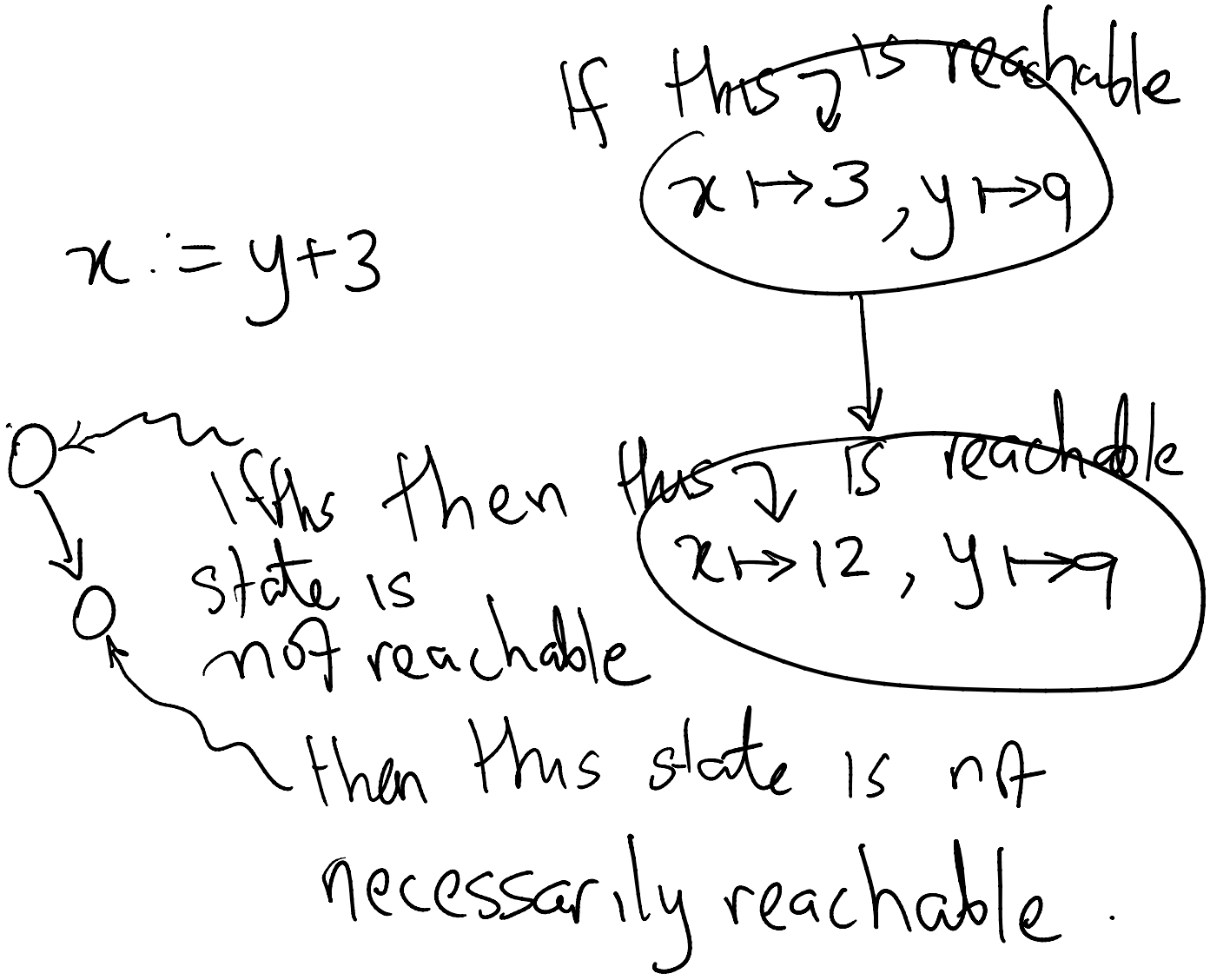
CAV 2014, ...

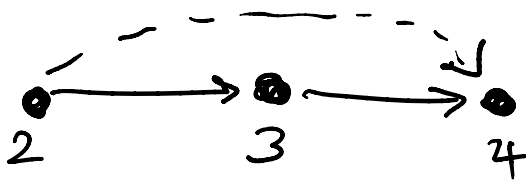




States vs. reachable states

- Initial state is reachable
- Transitive closure



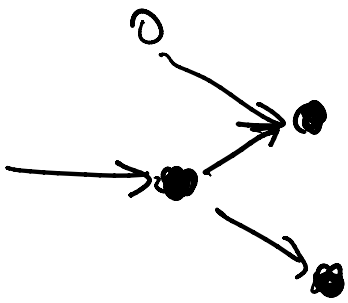


$$\frac{2<3 \quad 3<4}{2<4}$$

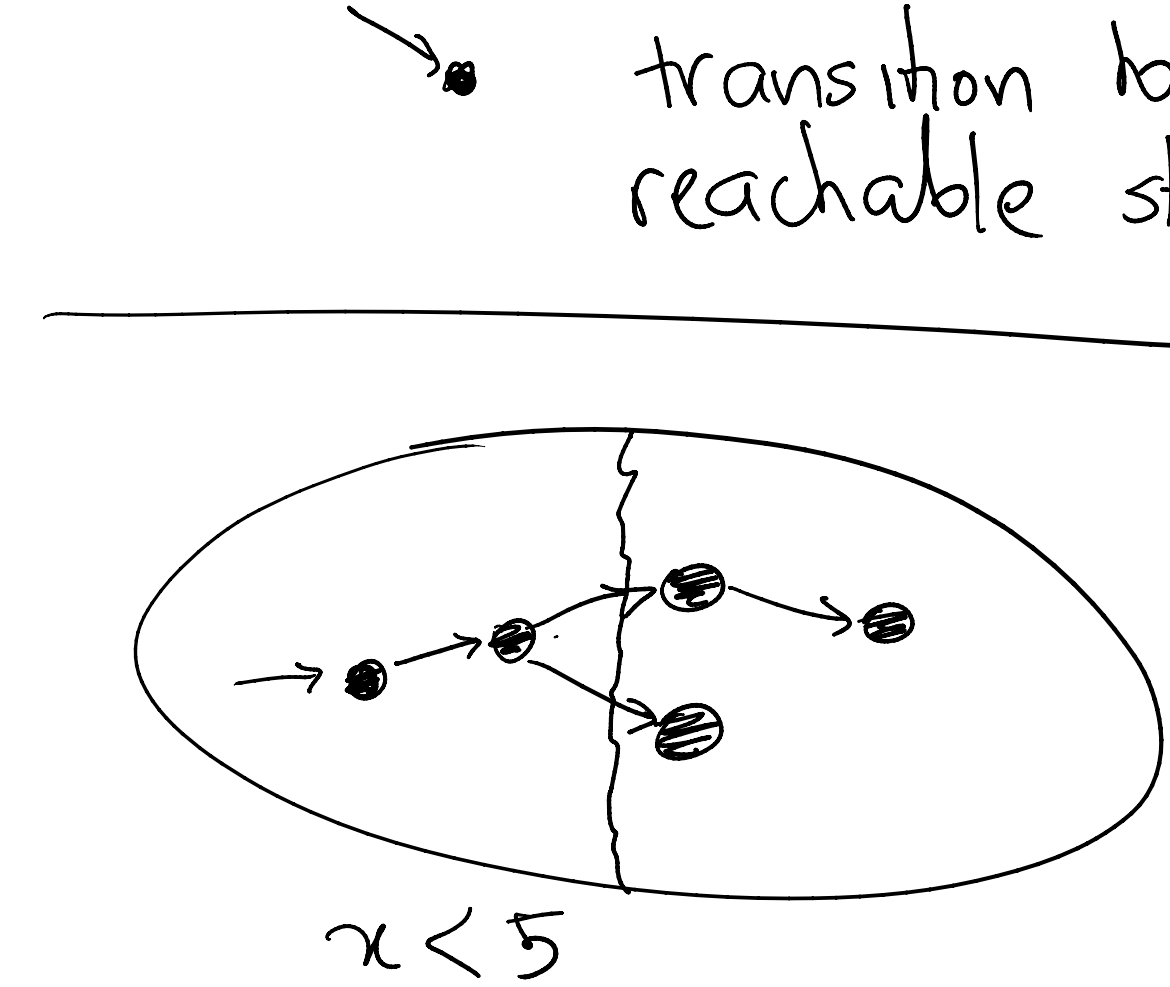
A relation $<$ is <u>transitive</u>

if whenever $a<b$ & $b<c$,

$a<c$



: Unreachable states can transition to



Can it be the case that

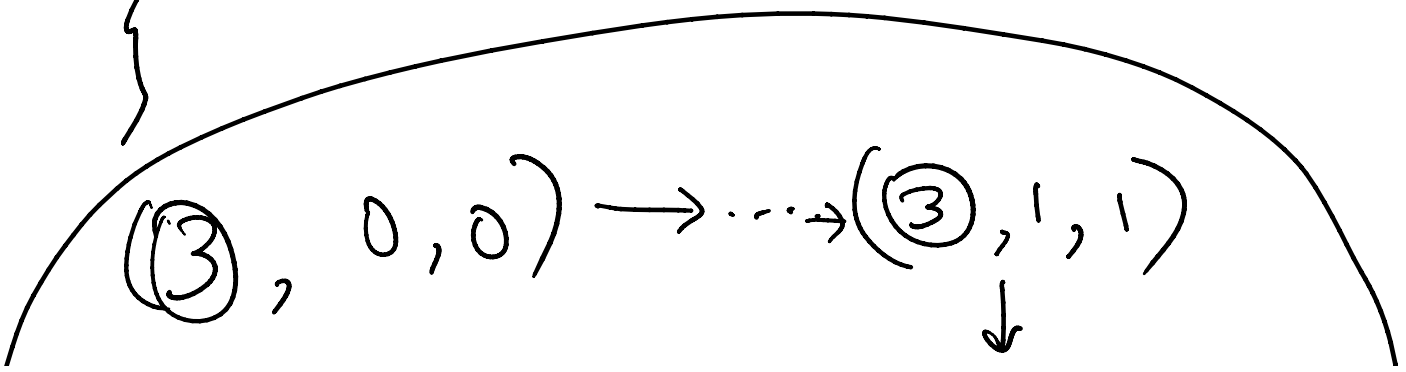
"Always $x < 5$" ?

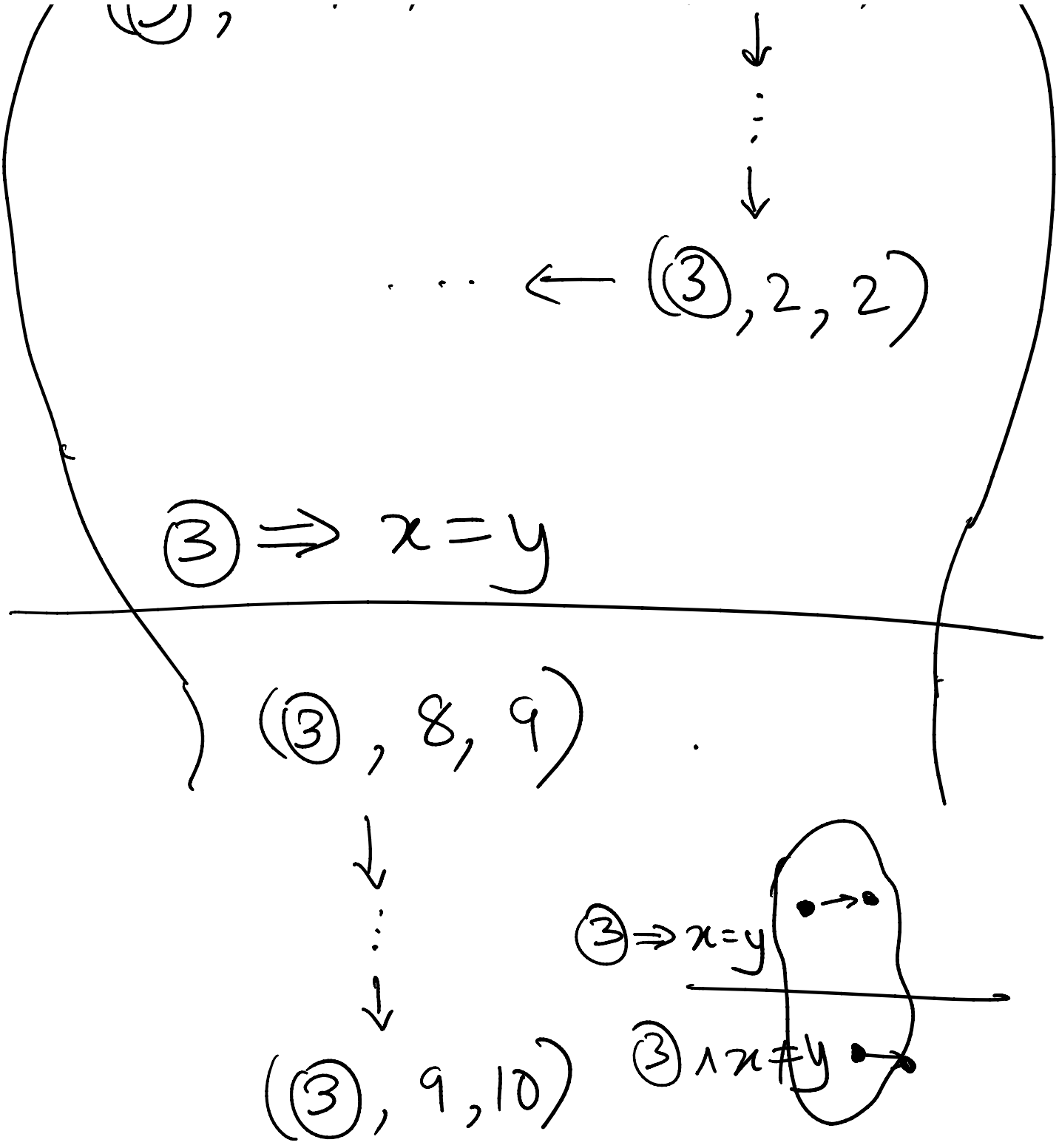
No. This proposition is false.

$x < 5$ cannot be an invariant.

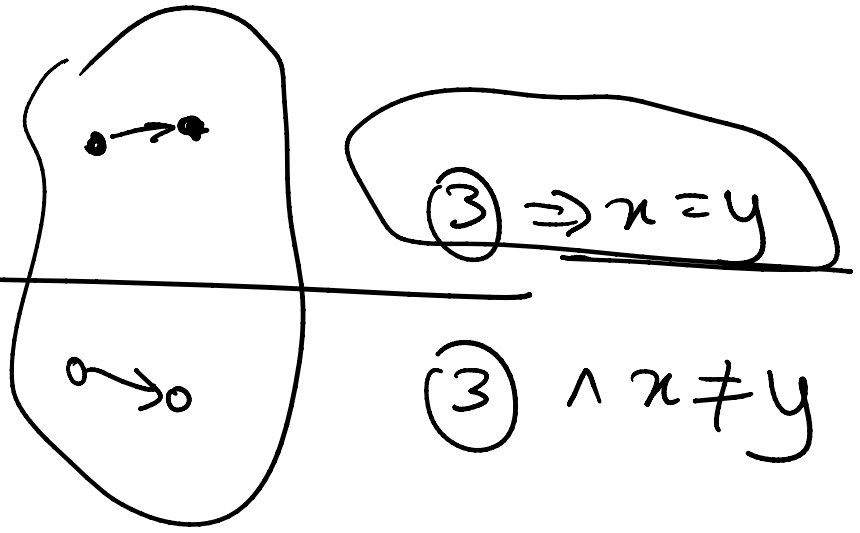
Defn: A property P is an invariant if for all reachable states, P holds.

```
int x := 0
int y := 0
while (③ y < 5) {
    x := x+1
    y := y+1
}
```




(3), 2, 2)
... ← (3)
(3) ⇒ x = y
(3, 8, 9)
(3, 9, 10)
(3) ⇒ x = y
(3) ∧ x ≠ y

But,
There are
no crossing
arrows.



① Inital state satisfies
③ ⇒ $x = y$

② There are no crossing
arrows.

Therefore, ∀ reachable states,
③ ⇒ $x = y$.

Inductive invariant P

① Initial state satisfies P

② ∀ states q if q satisfies P & $q \to q'$, then q' also satisfies P.

∴ ∀ reachable states q, q satisfies P.

$x := 20$

$y := 0$

while (③ $y < 100$) {

$x := 2x - 16$

10	20
4	24
-8	32
⋮	48
⋮	

$y := y+1$ | | 80

}

assert ($x > 0$) assert ($x \geq 20$)

③ $\Rightarrow x > 0$ ~~~ is an invariant
is not an inductive invariant.

Defn: A property P is an invariant if ∀ reachable states q, q satisfies P.

$x > 0 \Longrightarrow x > 20$

Invariant ≀ Inductive invariant

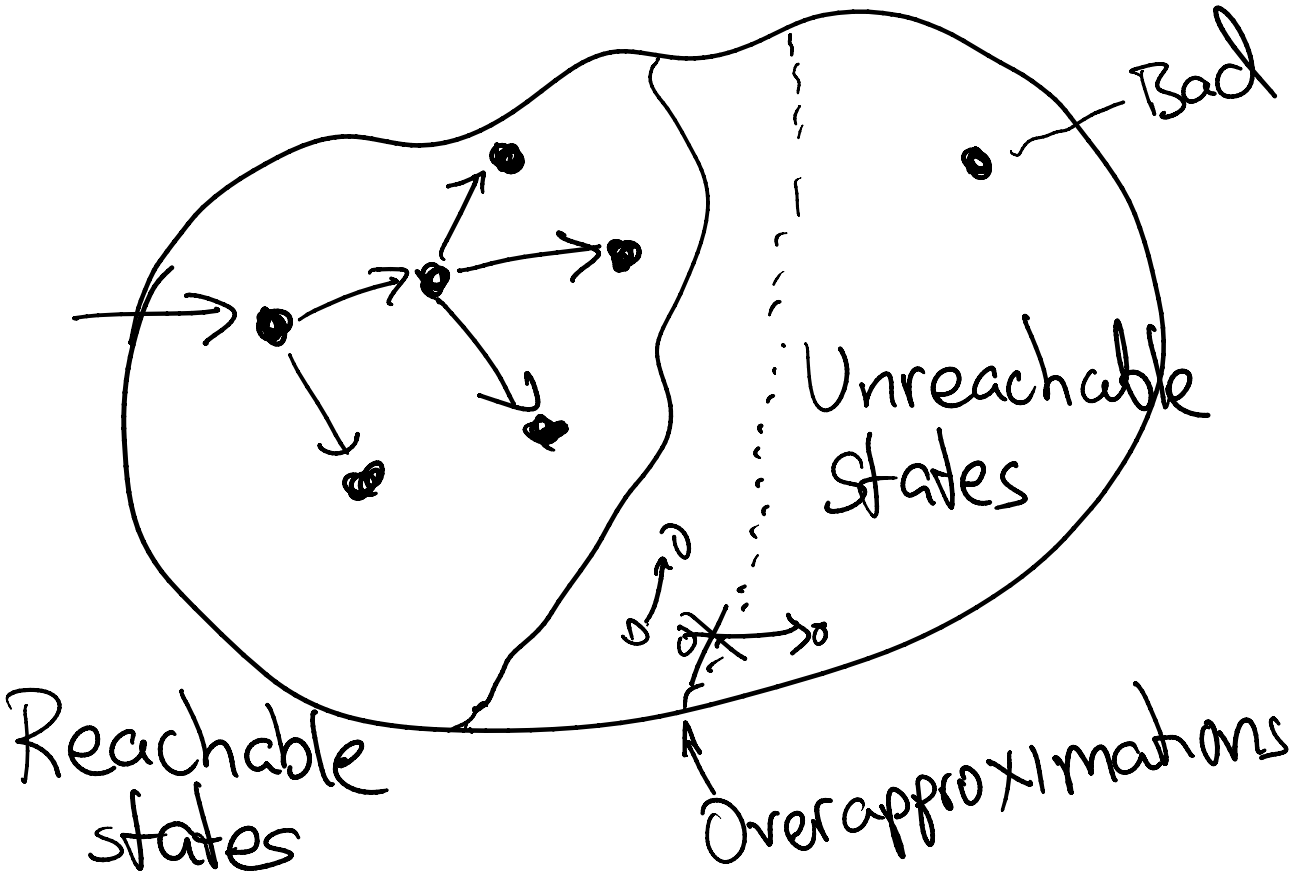
1. | |

Inductive strengthening.

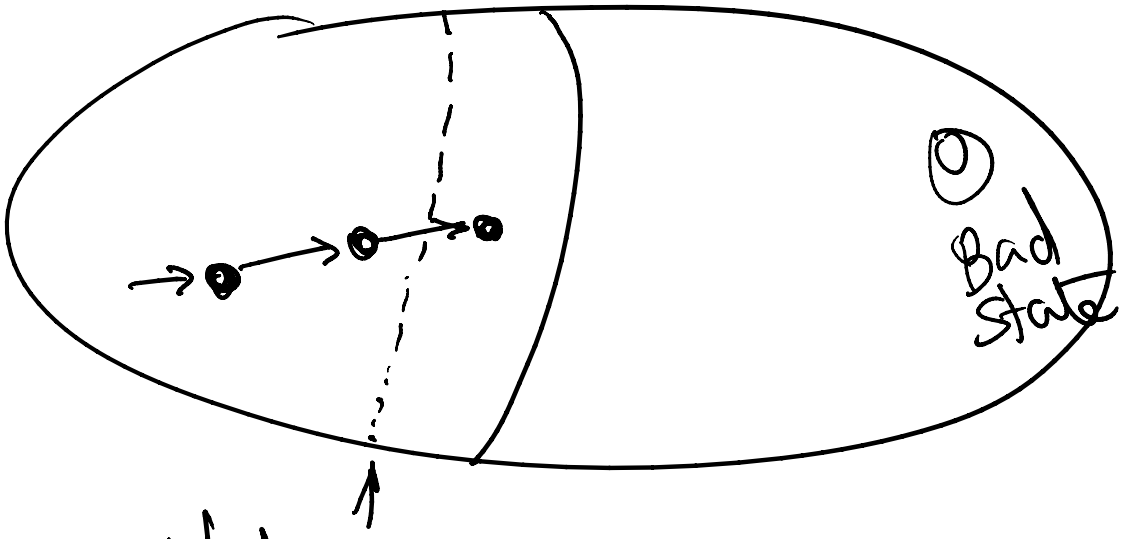
$$\frac{\{P \wedge b\}\ c\ \{P\} \qquad P \wedge \neg b \Rightarrow Q}{\{P\}\ \text{while } b \text{ do } c\ \{Q\}}$$

$$\frac{\{Inv \wedge b\}\ c\ \{Inv\}}{\{Inv\}\ \text{while } b \text{ do } c\ \{Inv \wedge \neg b\}}$$

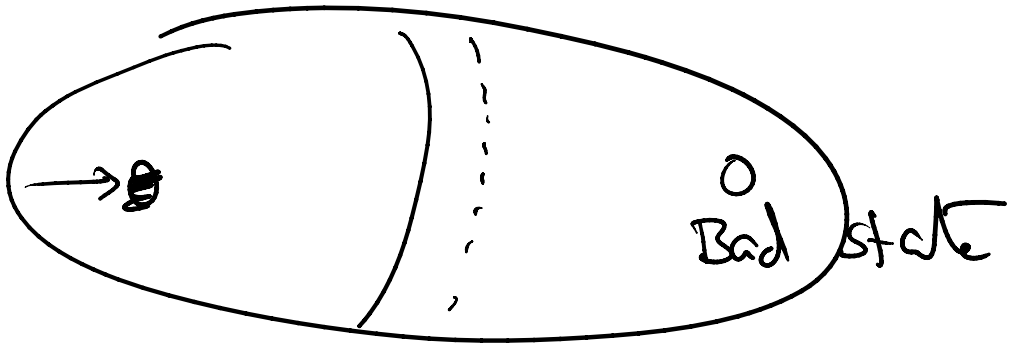
The only invariants powerful enough for use in the while rule are inductive invariants.



P(x, y, z) is true iff (x, y, z) is a reachable state.



Under-approximations will not work as a proof device



Over-approximations might work.