Lecture 4 State Machines. 27/09/2025

State machines consist of Set Q of States
- Start with some State $q_0 \in Q$
- Set T of allowed transitions each of the form $q \mapsto r$ for some $q, r \in Q$.

Example: Counter

$$\to \textcircled{0} \to \textcircled{1} \to \textcircled{2} \to \textcircled{3} \to \dots$$

Start State

$Q = \{0, 1, 2, 3, 4 \dots\} = \mathbb{N}$

Set of states i.e ~~Aa~~ $\mathbb{N}$

One state can have many transitions out of it, can do loops & all sorts of crazy things

Counter - $q_0 = 0$

$$T = \{ i \to i+1 \; | \; i \in \mathbb{N} \}$$

for every.

Execution of state machines is an (in)finite sequence a sequence of states you can reach by a sequence of transitions

$$q_0 \to q_1 \to q_2 \to \dots (\to q_n)$$

Where each $q_i \to q_{i+1} \in T$

- Writing Algorithm, dont want forever infinite and so distinguish between infinite & finite executions is important.

Reachability: State $r \in Q$ is reachable if $\exists$ execution $q_0 \to \dots \to r$.

State predictate,

8puzzle

4	6	
5	1	3
7	8	2

$\longrightarrow$

1	2	3
4	5	6
7	8	

Two good examples

1 3 x
4 2 5
7 8 6

$\Rightarrow$ ✓ possible ~~✓~~ NOT

1 2 3
4 6 5
7 8 x

8puzzle state machine

Q = { all possible arrangements of 1, 2, 3 . . 8, blank in 3x3 }

↑ everything about the puzzle so all different arrangements.

$q_0 = ?$ ↑ Start State.

q_f = sorted by reading order ↑ final State

T = { $q \to r$ | can get from state q → state r in one slide

≤4 T from any state.

Claim cannot get to state $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 6 & 5 \\ 7 & 8 & x \end{pmatrix}$ to State $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & x \end{pmatrix}$

So build technology to do this.

State Predictate

predicate $P(q)$ on states $q \in Q$

(function from Q to {true, false}).

preserved: if $\forall q \to r \in T$: $P(q) \to P(r)$

wherever theres a transition from $q \to r$ P of q implies P of r. So P remains true

Invariance: If $P(q)$ is true for all reachable states q.

If $P(q)$ is invariant, & $P(q)$ is false then r is unreachable.

How to prove state predicate is invariable.

<u>Invariant Principle:</u> ← type of induction

for state predicate $p(q)$. If $P(q_0)$ is true and
$P(q)$ is preserved
then $P(q)$ is invariant

Proof of Invariant principle using induction.

$IH\ P(n) = \forall$ length-n executions $q_0 \to \dots \to q_{(n)}$:
$P(q_n)$ is true

Base case: $IH(0)$. Executions q_0, $P(q_0)$ by ①

Step. Assume ~~IH(n)~~ $IH(n-1)$. Prove $IH(n)$

Consider any length n execution $q(0) \to q_n \to \dots \to q_{n-1} \to q_n$

WTS $P(q_n)$.

By Induction hypothesis $n-1$ $IH(n-1)$, $P(q_{n-1})$ is true

By ② $P(q_{n-1})$ imp $\to$ $P(q_n)$.

So. $P(q_n)$

By induction, $\forall n \in \mathbb{N}: IH(n) \Rightarrow P(q)$ $\forall$ reachable states q.

Template

Thm: $P(q)$ is invariant.

Proof by invariant principle

① $P(q_0)$ is true b/c $\boxed{?}$

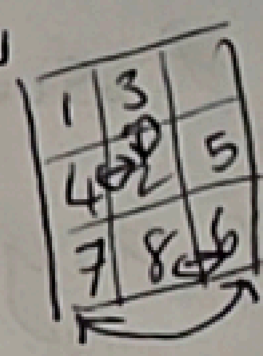
② $P(q)$ preserved

~~ConState~~ consider any $q \to r \in T$ with $P(q)$

Then $P(r)$ is true b/c $\boxed{?}$

State predicate for 8-puzzle

- for $1 \le i < j \le 8$, call $\{i,j\}$ an inverted pair. an if j appears before i in reading order (in state q)
- $P(q)$ = number of inverted pairs in state q is odd.

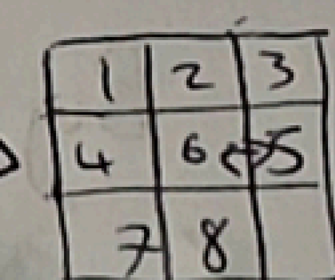


4 inverted pairs in this example

To prove invariance prove preserved first.

no natural start point for 8 puzzle, but preserved is a general property about system so dont need to ~~prove any~~ assume anything for it.

1 i.p.



Thm: $P(q)$ is preserved

Proof. Consider any $q \to r \in T$ with $P(q)$

WTS $P(r)$ true. By cases because only 2 moves horizontal swaps & vertical swaps.

Case ① horizontal move $q \to r$

$ix \to xi$. (we care about reading order)

No changes in reading order or in inverted pairs so $P(r)$.

Case ② Vertical move: $q \to r$.

$$\begin{array}{ccc|ccc|ccc} i & j & k & & i & j & & & i \\ x & & & k & x & & j & k & x \end{array}$$

Swapping $i \Leftrightarrow x$
equivalent to $i \Leftrightarrow j$ then $i \Leftrightarrow k$

- Swapping 2 adjacent items, i, j in reading order.
Changes # inverted pairs by ± 1

If $P(q)$ is invariant and $P(r)$ is false then r is unreachable.

2 adjacent swaps changes by $\underbrace{(\pm 1)}_{\text{odd}} + \underbrace{(\pm 1)}_{\text{odd}} = \text{even}$

$\Rightarrow P(r)$ true.

50:00 time.