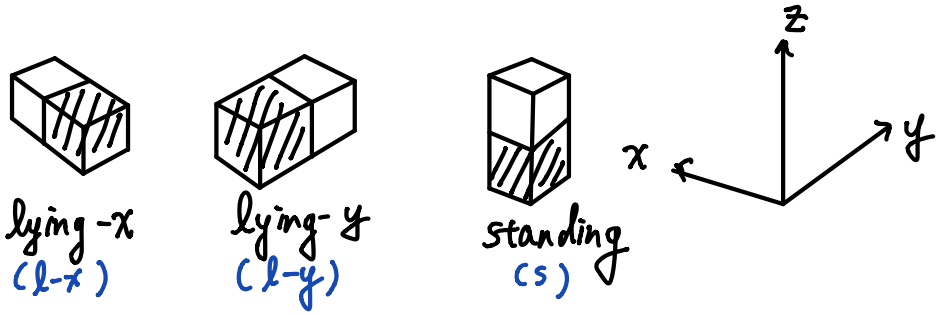
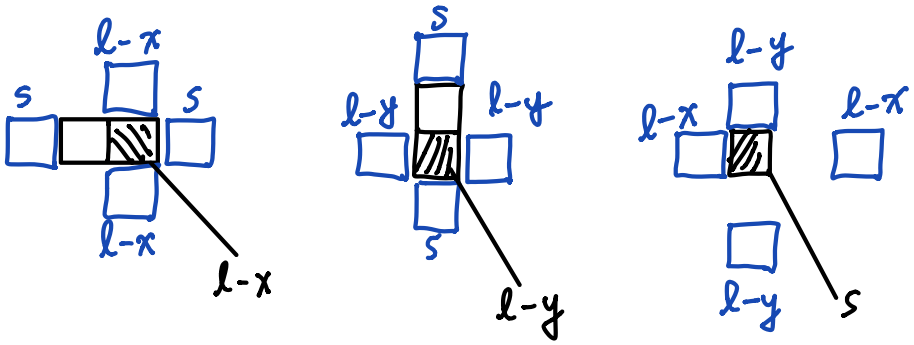
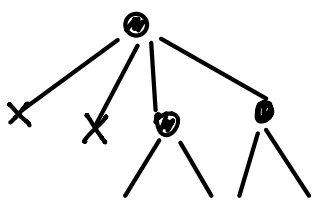


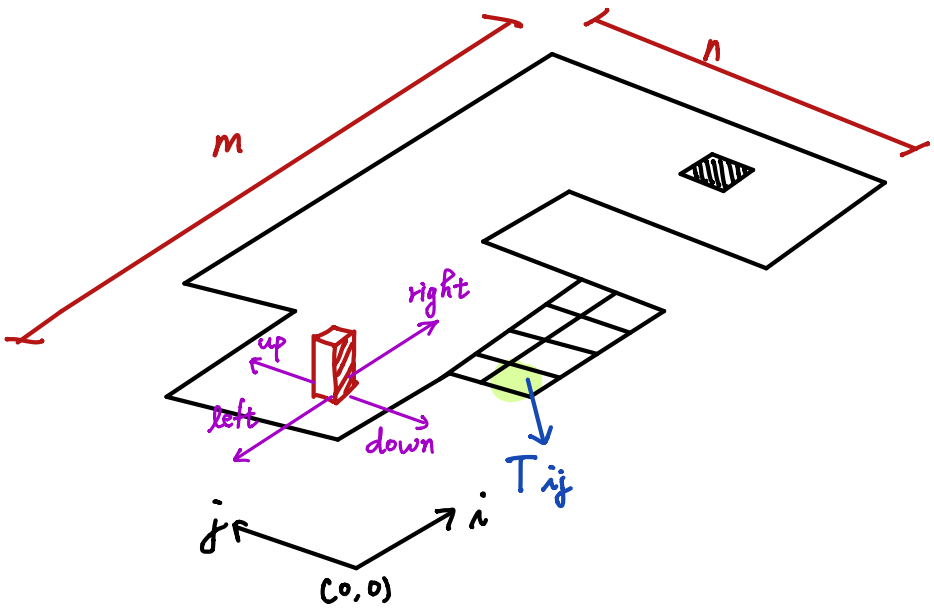


排除 Illegal Move

固定 size $\mathbb{B}$, an upper bound of move $\mathbb{B}$ ⇒
4 choices for each decision



Total 步數從 1 步開始, SAT solver can learn invalid pattern gradually



1. Set path length range :

Distance ≤ Path length P_n ≤ Size of map

2. For each P_n, determine variables

$\mathbb{A} = \{L_x, L_y, S\}$

Each tile $T_{ij} \Rightarrow v_{ija \in \mathbb{A}}$ (constant) $= \begin{cases} 1, & \text{state A on tile } T_{ij} \text{ is valid} \\ 0, & \text{not valid} \end{cases}$

略

Block 的 State (变数) $b_{ij_{a \in \mathbb{A}}} = \begin{cases} 1, & \text{block at the } T_{ij} \text{ with state a} \\ 0, & \text{otherwise} \end{cases}$

: $3 \times |\mathbb{B}|$

Move, for each node (变数)

: $3 \times |\mathbb{B}| \times 4$

$\mathbb{M} = \{\text{left}, \text{right}, \text{down}, \text{up}\}$

$move_{ija_{m \in \mathbb{M}}} = \begin{cases} 1, & \text{Given block's state } b_{ija}, \text{ the player next take action } m \in \mathbb{M} \\ 0, & \text{otherwise} \end{cases}$

Rules

- Matching in V_{ija} and d_{ija}

$$\bigwedge_{i\in[m] j\in[n]} \bigwedge_{a\in\mathbb{A}} (b_{ija} \rightarrow V_{ija})$$ 略

B_{ij} on the map

- b_{ijL_x}, b_{ijL_y}, b_{ijS} choose one among them

$$\bigwedge_{i\in[m],j\in[n]} \left(b_{ijL_x} \rightarrow \bigwedge_{\substack{a\in\mathbb{A} \\ a\neq L_x}} (\overline{b_{ija}}) \bigwedge_{a\in\mathbb{A}} (\overline{b_{i+1ja}}) \right)$$

$$\bigwedge_{i\in[m],j\in[n]} \left(b_{ijL_y} \rightarrow \bigwedge_{\substack{a\in\mathbb{A} \\ a\neq L_y}} (\overline{b_{ija}}) \bigwedge_{a\in\mathbb{A}} (\overline{b_{ij+1a}}) \right)$$

$$\bigwedge_{i\in[m],j\in[n]} \left(b_{ijS} \rightarrow \bigwedge_{\substack{a\in\mathbb{A} \\ a\neq S}} (\overline{b_{ija}}) \right)$$

$$b_{i_{start} j_{start} S} = 1, \quad b_{i_{end} j_{end} S} = 1$$

- Choosen b_{ija} can form a single path

Given a node v_{ija} and its outgoing edge ε_k, incoming edge δ_k

- P1 A node on the path puts its "one" outgoing edge on the path (expect goal)

Covered by Successor function

$$\bigwedge_{i\in[m]/i_{end},\, j\in[n]/j_{end}, a\in\mathbb{A}} \{ b_{ija} \leftrightarrow \bigvee_{m\in M} (move_{ijam}) \}$$

$$\bigwedge_{m\in M} (\overline{move_{i_{end} j_{end} S m}})$$

$$\bigwedge_{i\in[m],j\in[n],a\in\mathbb{A}} \left(\bigwedge_{m_1,m_2\in M, m_1\neq m_2} (\overline{move_{ija m_1}} \vee \overline{move_{ija m_2}}) \right)$$

start → intermediate → goal

$\neg b_{ija} \vee (U_{mom} \dots)$

$(\wedge_{mom} (\overline{move_{ijam}})) \vee b_{ija}$

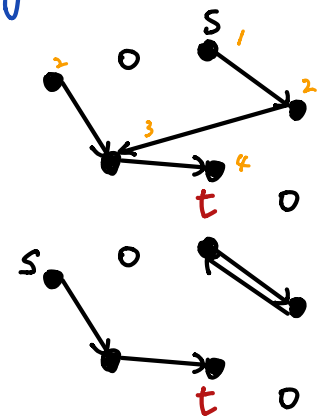
⇒ Guarantee that there are k nodes and k-1 edges

Start → V_2 → V_3 → .... → V_k → end

$p(V_2) = S(start)$

$p(V_i) = S(p(V_{i-1}))$ for $i \in 3 \dots k$

$p(V_k) = S^{k-1}(start)$



○ Successor Function (Unary Encoding)

$U_{ijap} = 1$ iff v_{ija}'s position is p, $p \in 1 \dots$ mapsize

△ 1-1 mapping between b_{ija} and U_{ijap} (# of b_{ija} = # of U_{ijap})

$\wedge_{i \in [n],\ j \in [n].\ ij \neq End,\ a \in A} \left(b_{ija} \leftrightarrow \vee_{p=1}^{mapsize} (U_{ijap}) \right)$

⇒ → $\wedge_{ija} \left(\overline{b_{ija}} \vee \left(\vee_{p=1}^{mapsize} (U_{ijap}) \right) \right)$

unique → $\wedge_{ija} \left(\wedge_{p_1 \neq p_2} \left(\overline{U_{ijap_1}} \vee \overline{U_{ijap_2}} \right) \right)$

⇐ → $\wedge_{ija} \wedge_{p=1}^{mapsize} \left(\overline{U_{ijap}} \vee b_{ija} \right)$

△ P's Continuality

• Each P maps to at most one node

$$\wedge_{p=1}^{mapsize} \left(\wedge_{i_1 \neq i_2,\ j_1 \neq j_2,\ a_1 \neq a_2} \left(\overline{U_{i_1 j_1 a_1 p}} \vee \overline{U_{i_2 j_2 a_2 p}} \right) \right)$$

⇒ Replace with that each state has at

- Recursion (State + action → Next State)

$$U_{i_{start}\, j_{start}\, S\, 0} = 1$$

$$U_{i_1 j_1 a_1 p} \wedge move_{i_1 j_1 a_1 m} \longrightarrow U_{i_2 j_2 a_2\, p+1}$$

$(i_1, j_1) \neq End$, $m \in M$

(i, j) & End

$$(U_{ij\, S\, p}) \wedge \begin{cases} move_{ij\, S\, up} \longrightarrow U_{ij+1\, L_y\, p+1} \\ move_{ij\, S\, down} \longrightarrow U_{ij-2\, L_y\, p+1} \\ move_{ij\, S\, right} \longrightarrow U_{i+1\, j\, L_x\, p+1} \\ move_{ij\, S\, left} \longrightarrow U_{i-2\, j\, L_x\, p+1} \end{cases}$$

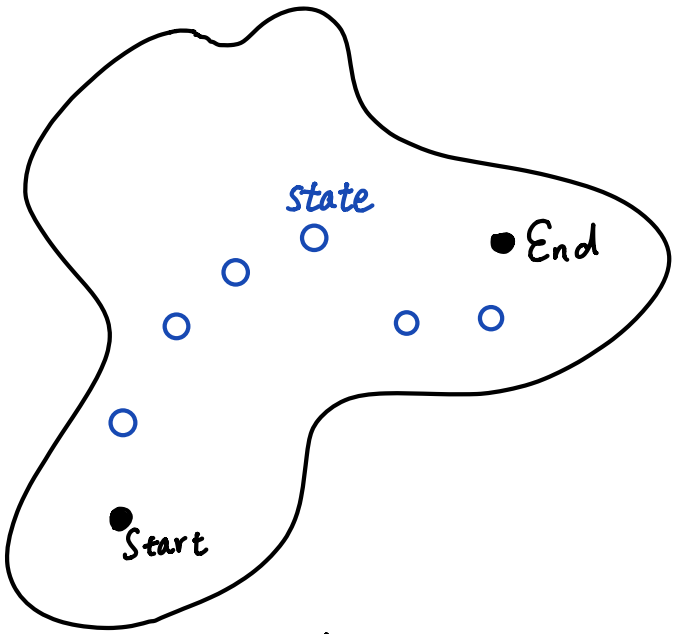
stand

$$(U_{ij\, L_x\, p}) \wedge \begin{cases} move_{ij\, L_x\, up} \longrightarrow U_{ij+1\, L_x\, p+1} \\ move_{ij\, L_x\, down} \longrightarrow U_{ij-1\, L_x\, p+1} \\ move_{ij\, L_x\, right} \longrightarrow U_{i+2\, j\, S\, p+1} \\ move_{ij\, L_x\, left} \longrightarrow U_{i-1\, j\, S\, p+1} \end{cases}$$

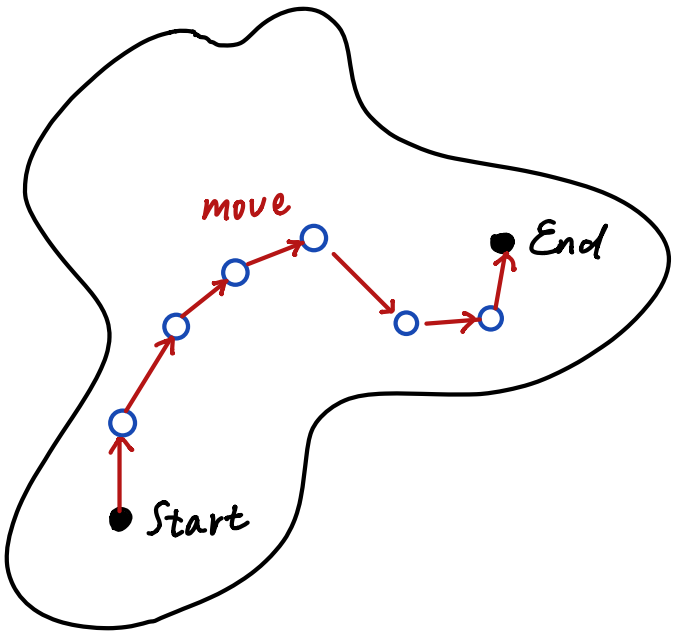
$(U_{ijLyp}) \wedge$

- $move_{ijLy\ up} \longrightarrow U_{i\ j+2\ S\ p+1}$
- $move_{ij\ Ly\ down} \longrightarrow U_{i\ j-1\ S\ p+1}$
- $move_{ij\ Ly\ right} \longrightarrow U_{i+1\ j\ Ly\ p+1}$
- $move_{ij\ Ly\ left} \longrightarrow U_{i-1\ j\ Ly\ p+1}$

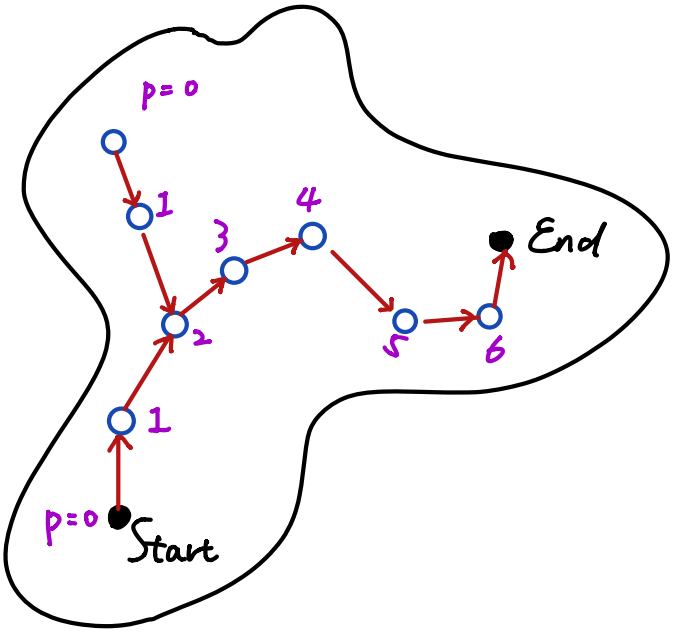
Some moves are INVALID!!
depend on i, j, a



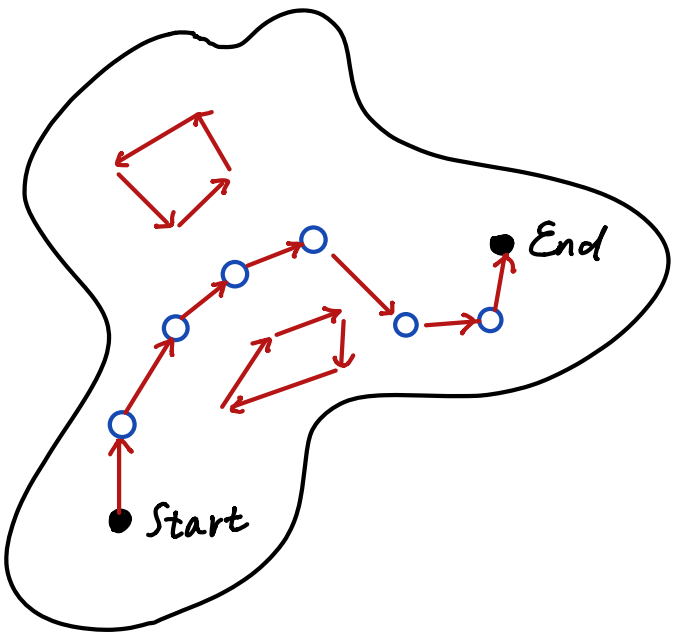
1.



2.



Branch



Sub-cycle