Planning Representations:
Symbolic Representation for Task Planning

Maxim Likhachev
Robotics Institute
Carnegie Mellon University
Planning to Construct a Birdcage

- Robot takes in a 3D model of a birdcage it needs to build
Planning to Construct a Birdcage

- Robot takes in a 3D model of a birdcage it needs to build.

Planning the order in which to assemble pieces is an example of Task Planning.
Famous “Blocksworld” Example

• Planning to re-order the blocks

start state

goal state
Famous “Blocksworld” Example

• Planning to re-order the blocks

Assuming the arm can reach/move all the top blocks, 
the problem is in figuring out the order

start state  

goal state
Famous “Blocksworld” Example

- Planning to re-order the blocks

**Actions:**

\[ \text{Move}(b,x,y) \] moves block \( b \) from \( x \) to \( y \)

\[ \text{MoveToTable}(b,x) \] moves block \( b \) from \( x \) to table \( y \)

\[ \begin{array}{c}
A \\
B \\
C
\end{array} \quad \begin{array}{c}
B \\
C \\
A
\end{array} \]

\textit{start state} \quad \textit{goal state}
Famous “Blocksworld” Example

• Planning to re-order the blocks

**Actions:**

\[ \text{Move}(b,x,y) \text{ – moves block } b \text{ from } x \text{ to } y \]

\[ \text{MoveToTable}(b,x) \text{ – moves block } b \text{ from } x \text{ to table} \]

What is a plan that achieves the goal?
Defining it as a Graph Search (State-space Search)

- Planning to re-order the blocks

**Actions:**

\[ \text{Move}(b,x,y) \] – moves block \( b \) from \( x \) to \( y \)

\[ \text{MoveToTable}(b,x) \] – moves block \( b \) from \( x \) to table

Any ideas for how to represent a state in a graph?
Defining it as a Graph Search (State-space Search)

- Planning to re-order the blocks

**Actions:**

- $\text{Move}(b,x,y)$ – moves block $b$ from $x$ to $y$
- $\text{MoveToTable}(b,x)$ – moves block $b$ from $x$ to table

**start state**

\[
\begin{array}{c}
A \\
B \\
C
\end{array}
\]

**goal state**

\[
\begin{array}{c}
B \\
C \\
A
\end{array}
\]

- $A=\text{on } B$
- $B=\text{on table}$
- $C=\text{on table}$
Defining it as a Graph Search (State-space Search)

- Planning to re-order the blocks

**Actions:**

Move\((b,x,y)\) – moves block \(b\) from \(x\) to \(y\)

MoveToTable\((b,x)\) – moves block \(b\) from \(x\) to table

**start state**

| A | B | C |

**goal state**

| B | C | A |
Defining it as a Graph Search (State-space Search)

- Planning to re-order the blocks

**Actions:**

\[ \text{Move}(b,x,y) \text{ – moves block } b \text{ from } x \text{ to } y \]
\[ \text{MoveToTable}(b,x) \text{ – moves block } b \text{ from } x \text{ to table} \]

\[
\begin{array}{c}
A \\
B \\
C
\end{array}
\]

**Start state**

\[
\begin{array}{c}
A=\text{on } B \\
B=\text{on table} \\
C=\text{on table}
\end{array}
\]

\[
\begin{array}{c}
A,B,C
\end{array}
\]

\[
\begin{array}{c}
A=\text{on } C \\
B=\text{on table} \\
C=\text{on table}
\end{array}
\]

...  

\[
\begin{array}{c}
B \\
C \\
A
\end{array}
\]

**Goal state**

Cost of each edge is often set to 1 (minimization of the total # of actions)
Defining it as a Graph Search (State-space Search)

- Planning to re-order the blocks

**Actions:**

- Move\((b,x,y)\) – moves block \(b\) from \(x\) to \(y\)
- MoveToTable\((b,x)\) – moves block \(b\) from \(x\) to table

![Diagram showing the transition from start state to goal state](image)

Any ideas for heuristics?
We would like to be able to represent ANY planning problem with a single representational language that allows for the definition of: STATES, ACTIONS, GOAL
Generic Representation of Symbolic Planning Problems

- STRIPS (=Stanford Research Institute Problem Solver)

  State Representation:

  Goal Representation:

  Action Representation:
Generic Representation of Symbolic Planning Problems

- STRIPS (=Stanford Research Institute Problem Solver)

**State Representation:**

- conjunction of positive(true) literals
  
  \[ \text{On}(A, B) \land \text{On}(B, \text{Table}) \land \text{On}(C, \text{Table}) \land \text{Block}(A) \land \text{Block}(B) \land \text{Block}(C) \land \text{Clear}(A) \land \text{Clear}(C) \]

**Goal Representation:**

**Action Representation:**
Generic Representation of Symbolic Planning Problems

- STRIPS (=Stanford Research Institute Problem Solver)

**State Representation:**

*conjunction of positive(true) literals*

(e.g., $\text{On}(A, B)^\land \text{On}(B, \text{Table})^\land \text{On}(C, \text{Table})^\land \text{Block}(A)^\land \text{Block}(B)^\land \text{Block}(C)^\land \text{Clear}(A)^\land \text{Clear}(C)$)

**Goal Representation:**

*Closed-world assumption:*

any conditions not mentioned in the state are assumed to be false

**Action Representation:**
Generic Representation of Symbolic Planning Problems

- STRIPS (=Stanford Research Institute Problem Solver)

**State Representation:**

\[
\text{conjunction of positive(true) literals}
\]

(e.g, \(\text{On}(A,B)^\land \text{On}(B,\text{Table})^\land \text{On}(C,\text{Table})^\land \text{Block}(A)^\land \text{Block}(B)^\land \text{Block}(C)^\land \text{Clear}(A)^\land \text{Clear}(C)\))

**Goal Representation:**

\[
\text{desired conjunction of positive(true) literals}
\]

**Action Representation:**
Generic Representation of Symbolic Planning Problems

- STRIPS (=Stanford Research Institute Problem Solver)

**State Representation:**
- conjunction of positive(true) literals
  -(e.g. On(A,B)\(^\land\)On(B,Table)\(^\land\)On(C,Table)\(^\land\)Block(A)\(^\land\)Block(B)\(^\land\)Block(C)\(^\land\)Clear(A)\(^\land\)Clear(C))

**Goal Representation:**
- desired conjunction of positive(true) literals

**Action Representation:**
- What is it for this goal?
Generic Representation of Symbolic Planning Problems

- STRIPS (=Stanford Research Institute Problem Solver)

**State Representation:**

conjunction of positive(true) literals

(e.g., On(A,B)\(^\wedge\)On(B,Table)\(^\wedge\)On(C,Table)\(^\wedge\)Block(A)\(^\wedge\)Block(B)\(^\wedge\)Block(C)\(^\wedge\)Clear(A)\(^\wedge\)Clear(C))

**Goal Representation:**

desired conjunction of positive(true) literals

**Action Representation:**

Goal: any state where A is directly on the table
Generic Representation of Symbolic Planning Problems

- STRIPS (=Stanford Research Institute Problem Solver)

**State Representation:**

conjunction of positive(true) literals

(e.g, \(\text{On}(A,B)^\land\text{On}(B,\text{Table})^\land\text{On}(C,\text{Table})^\land\text{Block}(A)^\land\text{Block}(B)^\land\text{Block}(C)^\land\text{Clear}(A)^\land\text{Clear}(C)\))

**Goal Representation:**

desired conjunction of positive(true) literals

Could be partially-specified

**Action Representation:**

Goal: any state where \(A\) is directly on the table

What is it for this goal?
Generic Representation of Symbolic Planning Problems

• STRIPS (=Stanford Research Institute Problem Solver)

**State Representation:**

\[
\text{conjunction of positive(true) literals} \\
(\text{e.g., On}(A,B)^{\text{On}}(B,\text{Table})^\text{On}(C,\text{Table})^\text{Block}(A)^\text{Block}(B)^\text{Block}(C)^\text{Clear}(A)^\text{Clear}(C))
\]

**Goal Representation:**

\[
\text{desired conjunction of positive(true) literals}
\]

**Action Representation:**

**Preconditions:** conjunction of positive(true) literals that must be held true in order for the action to be applicable

**Effect:** conjunction of positive(true) literals showing how the state will change (what should be deleted and added)
STRIPS (Stanford Research Institute Problem Solver)

State Representation:

\[\text{conjunction of positive(true) literals}\]

\[\text{(e.g., } On(A,B) \land On(B,Table) \land On(C,Table) \land Block(A) \land Block(B) \land Block(C) \land Clear(A) \land Clear(C))\]

Goal Representation:

\[\text{desired conjunction of positive(true) literals}\]

What are preconditions & effect for MoveToTable(b,x) action?

Action Representation:

**Preconditions:** conjunction of positive(true) literals that must be held true in order for the action to be applicable

**Effect:** conjunction of positive(true) literals showing how the state will change (what should be deleted and added)
Generic Representation of Symbolic Planning Problems

- STRIPS (=Stanford Research Institute Problem Solver)

**State Representation:**

\[
\text{conjunction of positive(true) literals}
\]

(e.g., $\text{On}(A,B)^{\land} \text{On}(B,\text{Table})^{\land} \text{On}(C,\text{Table})^{\land} \text{Block}(A)^{\land} \text{Block}(B)^{\land} \text{Block}(C)^{\land} \text{Clear}(A)^{\land} \text{Clear}(C)$)

**Goal Representation:**

**What are preconditions & effect for MoveToTable(b,x) action?**

**Action Representation:**

- **Precondition:** conjunction of positive(true) literals that must be held true in order for the action to be applicable
- **Effect:** conjunction of positive(true) literals showing how the state will change (what will be deleted and added)

\[
\text{MoveToTable}(b,x)
\]

**Precond:** \(\text{On}(b,x)^{\land} \text{Clear}(b)^{\land} \text{Block}(b)^{\land} \text{Block}(x)\)

**Effect:** \(\text{On}(b,\text{Table})^{\land} \text{Clear}(x)^{\land} \sim \text{On}(b,x)\)
Generic Representation of Symbolic Planning Problems

• STRIPS (=Stanford Research Institute Problem Solver)

**State Representation:**

\[ \text{conjunction of positive(true) literals} \]
\[ (e.g., \text{On(A,B)} \land \text{On(B,Table)} \land \text{On(C,Table)} \land \text{Block(A)} \land \text{Block(B)} \land \text{Block(C)} \land \text{Clear(A)} \land \text{Clear(C)}) \]

**Goal Representation:**

\[ \text{desired conjunction of positive(true) literals} \]

**Action Representation:**

**Preconditions:** conjunction of positive(true) literals that must be held true in order for the action to be applicable

**Effect:** conjunction of positive(true) literals showing how the state will change (what should be deleted and added)

What are preconditions & effect for for Move(b,x,y) action?
• Representing it with STRIPS

Start state:
\( \text{On}(A,B)^\land \text{On}(B,\text{Table})^\land \text{On}(C,\text{Table})^\land \text{Block}(A)^\land \text{Block}(B)^\land \text{Block}(C)^\land \text{Clear}(A)^\land \text{Clear}(C) \)

Goal state:
\( \text{On}(B,C)^\land \text{On}(C,A)^\land \text{On}(A,\text{Table}) \)

Actions:

MoveToTable\((b,x)\)
Precond: \( \text{On}(b,x)^\land \text{Clear}(b)^\land \text{Block}(b)^\land \text{Block}(x) \)
Effect: \( \text{On}(b,\text{Table})^\land \text{Clear}(x)^\land \sim \text{On}(b,x) \)

Move\((b,x,y)\)
Precond: \( \text{On}(b,x)^\land \text{Clear}(b)^\land \text{Clear}(y)^\land \text{Block}(b)^\land \text{Block}(y)^\land (b \sim = y) \)
Effect: \( \text{On}(b,y)^\land \text{Clear}(x)^\land \sim \text{On}(b,x)^\land \sim \text{Clear}(y) \)
Representing it with STRIPS

**Start state:**
\[\text{On}(A,B) \land \text{On}(B,\text{Table}) \land \text{On}(C,\text{Table}) \land \text{Block}(A) \land \text{Block}(B) \land \text{Block}(C) \land \text{Clear}(A) \land \text{Clear}(C)\]

**Goal state:**
\[\text{On}(B,C) \land \text{On}(C,A) \land \text{On}(A,\text{Table})\]

**Actions:**

- **MoveToTable\((b,x)\)**
  - **Precond:** \(\text{On}(b,x) \land \text{Clear}(b) \land \text{Block}(b) \land \text{Block}(x)\)
  - **Effect:** \(\text{On}(b,\text{Table}) \land \text{Clear}(x) \land \neg \text{On}(b,x)\)

- **Move\((b,x,y)\)**
  - **Precond:** \(\text{On}(b,x) \land \text{Clear}(b) \land \text{Clear}(y) \land \text{Block}(b) \land \text{Block}(y) \land (b \neq y)\)
  - **Effect:** \(\text{On}(b,y) \land \text{Clear}(x) \land \neg \text{On}(b,x) \land \neg \text{Clear}(y)\)
Back to the Example

- Representing it with STRIPS

We can now write a (domain-independent) program that takes in such specifications and automatically provides a function GetSuccessors(state S, action A) required for implicit graph construction.

**Start state:**
On(A,B)\(^\land\)On(B,Table)\(^\land\)On(C,Table)\(^\land\)Block(A)\(^\land\)Block(B)\(^\land\)Block(C)\(^\land\)Clear(A)\(^\land\)Clear(C)

**Goal state:**
On(B,C)\(^\land\)On(C,A)\(^\land\)On(A,Table)

**Actions:**
- **MoveToTable(b,x)**
  Precond: On(b,x)\(^\land\)Clear(b)\(^\land\)Block(b)\(^\land\)Block(x)
  Effect: On(b,Table)\(^\land\)Clear(x)\(^\land\)\neg On(b,x)

- **Move(b,x,y)**
  Precond: On(b,x)\(^\land\)Clear(b)\(^\land\)Clear(y)\(^\land\)Block(b)\(^\land\)Block(y)\(^\land\)(b\(\sim\)y)
  Effect: On(b,y)\(^\land\)Clear(x)\(^\land\)\neg On(b,x)\(^\land\)\neg Clear(y)
• Representing it with STRIPS

We can now write a (domain-independent) program that takes in such specifications and automatically provides a function GetSuccessors(state S, action A) required for implicit graph construction.

Start state:
On(A,B)\land On(B,Table)\land On(C,Table) \land Block(A) \land Block(B) \land Block(C) \land Clear(A) \land Clear(C)

Goal state:
On(B,C) \land On(C,A) \land On(A,Table)

Actions:
MoveToTable(b,x)
Precond: On(b,x) \land Clear(b) \land Block(b) \land Block(x)
Effect: On(b,Table) \land Clear(x) \land \neg On(b,x)

Move(b,x,y)
Precond: On(b,x) \land Clear(b) \land Clear(y) \land Block(b) \land Block(y) \land (b \neq y)
Effect: On(b,y) \land Clear(x) \land \neg On(b,x) \land \neg Clear(y)

This graph can be searched with A* or any other search.

This is often referred to as domain-independent planning.
• Representing it with STRIPS

**Start state:**
\[
\text{On}(A,B)^\text{On}(B,\text{Table})^\text{On}(C,\text{Table})^\text{Block}(A)^\text{Block}(B)^\text{Block}(C)^\text{Clear}(A)^\text{Clear}(C)
\]

**Goal state:**
\[
\text{On}(B,C)^\text{On}(C,A)^\text{On}(A,\text{Table})
\]

**Actions:**

- **MoveToTable** \((b,x)\)
  
  **Precond:** \(\text{On}(b,x)^\text{Clear}(b)^\text{Block}(b)^\text{Block}(x)\)
  
  **Effect:** \(\text{On}(b,\text{Table})^\text{Clear}(x)^{\sim}\text{On}(b,x)\)

- **Move** \((b,x,y)\)
  
  **Precond:** \(\text{On}(b,x)^\text{Clear}(b)^\text{Clear}(y)^\text{Block}(b)^\text{Block}(y)^{(b\sim=y)}\)
  
  **Effect:** \(\text{On}(b,y)^\text{Clear}(x)^{\sim}\text{On}(b,x)^{\sim}\text{Clear}(y)\)

Any ideas for domain-independent heuristics?
What You Should Know…

• How to represent a particular planning problem using STRIPS language and how this translates into a graph

• The motivation behind creating domain-independent planning representations such as STRIPS