INFORMATICS INSTITUTE OF TECHNOLOGY

CM 2602 - Artificial Intelligence

PLANNING

Introduction to Planning

- Planning is an important topic in AI.
- Planning is required for every task.
- Example: Reaching a particular destination requires planning.

What is Planning?

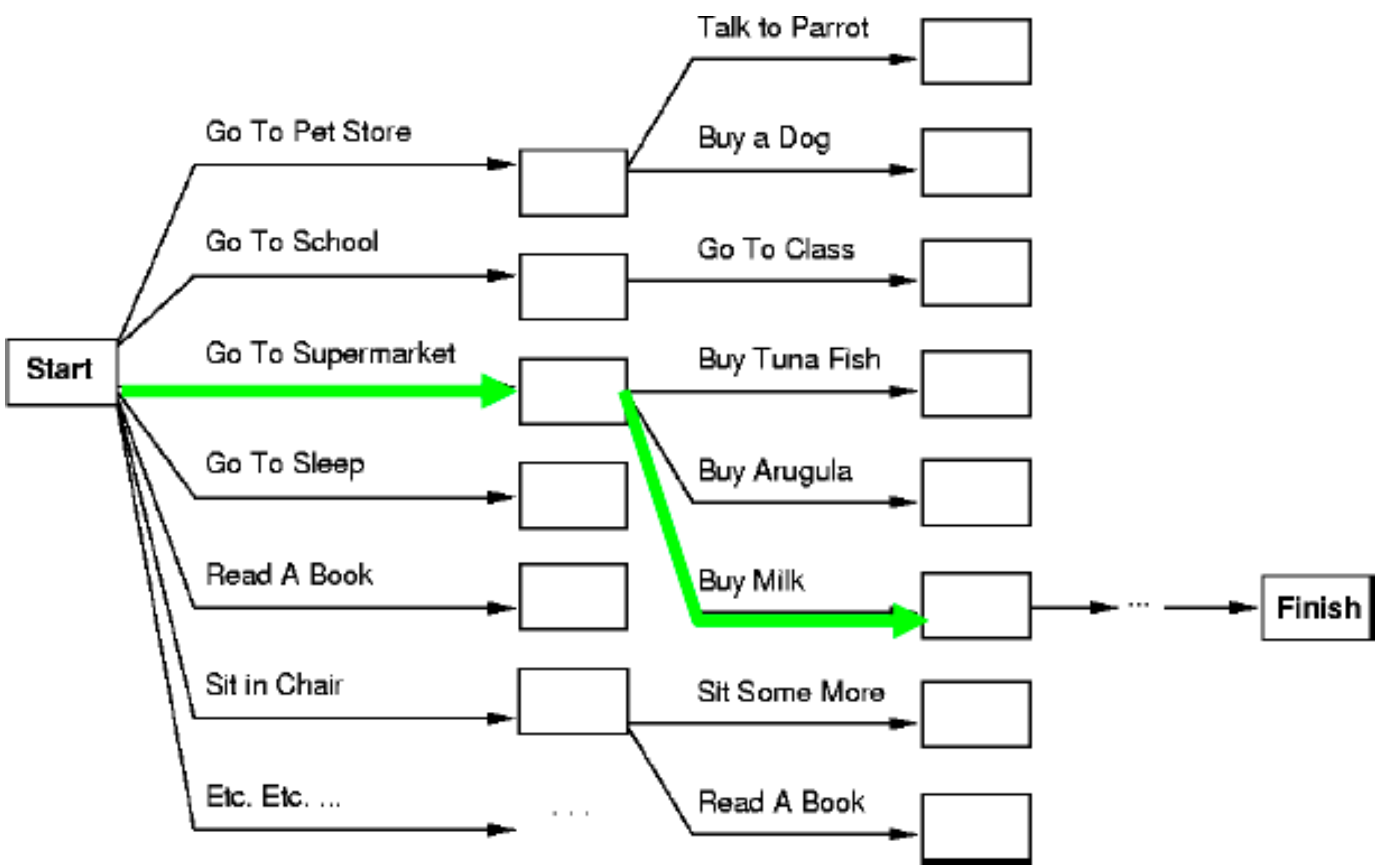
- **Planning** is the process of computing several steps of a problem-solving procedure before executing any of them.
- Find **sequence of actions** that achieves a given **goal** when executed from a given **initial world state**.

Problems with Standard Search

- Overwhelmed by irrelevant actions
- Finding a good heuristic function is difficult
- Cannot take advantage of problem decomposition

Example:

- Consider the task: get milk, bananas, and a cordless drill
  - Standard search algorithms seem to fail miserably
  - Why? Huge branching factor & heuristics



Planning vs. Problem Solving

- Planning agent is different from problem solving agent in:
    - Representation of goals, states, actions
    - Use of explicit, logical representations
    - Way it searches for solutions

- Planning is more powerful because of the representation and methods used

Planning vs. Problem Solving

	Problem Solving	Planning
States	Data Structures	Logical Sentences
Actions	Code	Preconditions / Outcomes
Actions	Code	Logical Sentences
Plan	Sequence from S_0	Constraints on Actions

Defining a Planning System:

- It requires;
  - **domain description,**
  - **action specification,**
  - **goal description.**

- A plan is assumed to be a sequence of actions and each action has its own set of preconditions to be satisfied before performing the action and also some effects which can be positive or negative.

Two Types of Planning

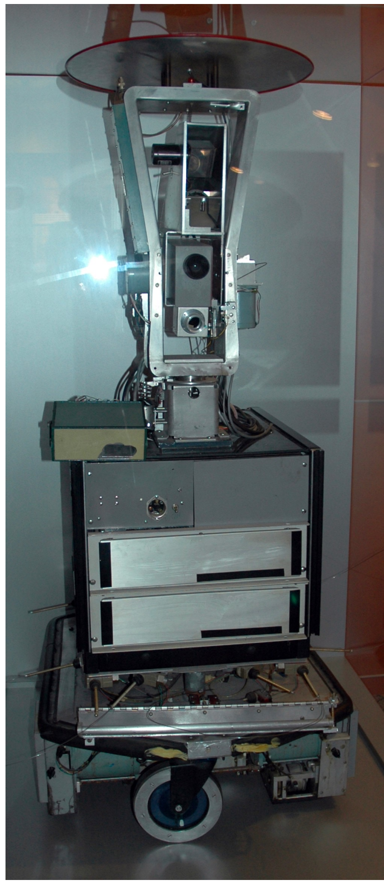
1. **Classical Planning**:
    - Fully Observable
    - Deterministic
    - Static

2. **Non- Classical Planning**:
    - Partially Observable
    - Stochastic

Many AI Planners in History

- Well-known AI Planners:
  - STRIPS (Fikes and Nilsson, 1971): theorem-proving system
  - ABSTRIPS (Sacerdoti, 1974): added hierarchy of abstractions
  - HACKER (Sussman, 1975): use library of procedures to plan
  - NOAH (Sacerdoti, 1975): problem decomposition and plan reordering

STRIPS (Stanford Research Institute Problem Solver)



- Use first-order logic and theorem proving to plan strategies from start to goal

- STRIPS language:
  - “Classical” approach that most planners use
  - Lends itself to efficient planning algorithms

- STRIPS is a restrictive way to express states, actions and goals, but leads to more efficiency.

STRIPS Example:

- States: conjunctions of ground, function-free, and positive literals, such as At(Home) ^ Have(Banana)
  - Closed-world assumption is used
- Goals: conjunctions of literals, may contain variables (existential), hence goal may represent more than one state
  - E.g. At(Home) ^ ¬ Have(Bananas)
  - E.g. At(x) ^ Sells(x, Bananas)
- Actions: preconditions that must hold before execution and the effects after execution

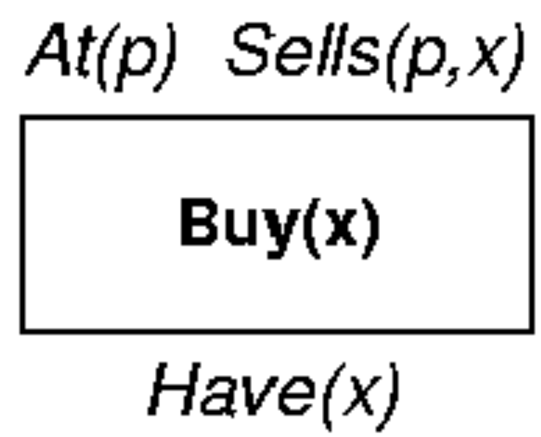
STRIPS Action Schema:

- An action schema includes:
  - action name & parameter list (variables)
  - precondition: a conjunction of function-free positive literals. Any variables in it must also appear in parameter list
  - effect: a conjunction of function-free literals (positive or negative)
    - add-list: positive literals
    - delete-list: negative literals
- Example:

Action: Buy (x)

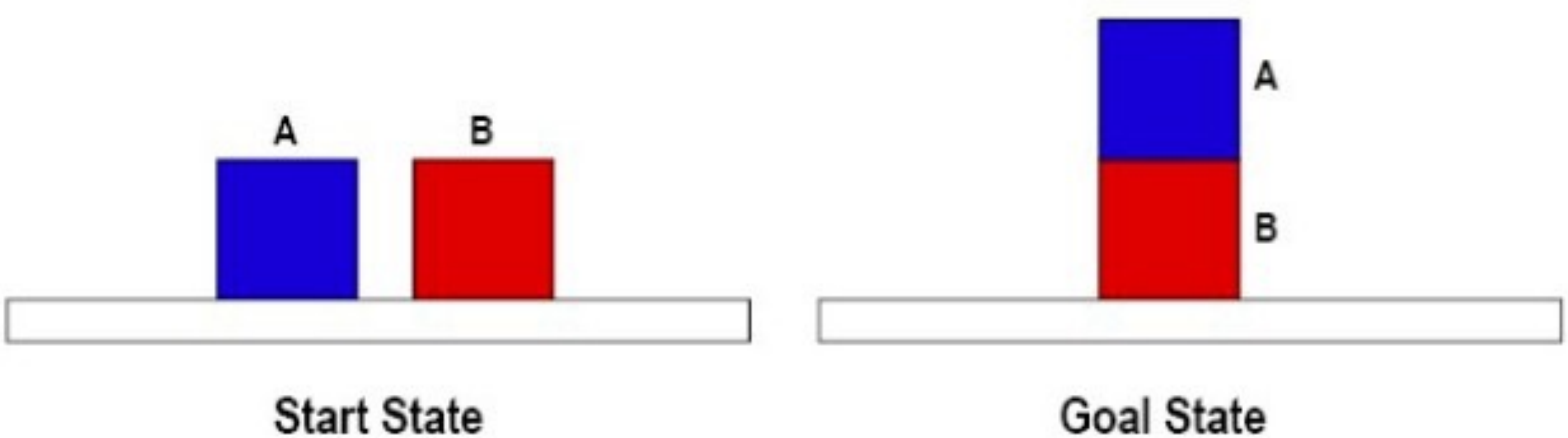
Precondition: At (p), Sells (p, x)

Effect: Have(x)



STRIPS Example:

- Goal state: ON(A, B)
- Start state: ON(A, Table); ON(B, Table); EMPTYTOP(A); EMPTYTOP(B)
- Operators:
  - Move(x, y)
    - Preconditions: ?
    - Add-list: ?
    - Delete-list: ?



- Preconditions: ON(x, Table); EMPTYTOP(y)
- Add-list: ON(x, y)
- Delete-list: EMPTYTOP(y); ON(x, Table)

Planning Algorithms:

1. Forward State Space Planning (FSSP)
   - Progression

2. Backward State Space Planning (BSSP)
   - Regression

Progression

- A plan is a sequence of STRIPS operators
- From initial state, search forward by selecting operators whose preconditions can be unified with literals in the state
- New state includes positive literals of effect; the negated literals of effect are deleted
- Search forward until goal unifies with resulting state
- This is state-space search using STRIPS operators

Regression

- A plan is a sequence of STRIPS operators
- The goal state must unify with at least one of the positive literals in the operator's effect
- Its preconditions must hold in the previous situation, and these become subgoals which might be satisfied by the initial conditions
- Perform backward chaining from goal
- Again, this is just state-space search using STRIPS operators

Partial Order Planning

- Idea:
  - works on several subgoals independently
  - solves them with subplans
  - combines the subplans
  - flexibility in ordering the subplans
  - least commitment strategy:
    - delaying a choice during search
    - Example, leave actions unordered, unless they must be sequential