Encoding Planning Problems as Propositional Logic Satisfiability

Sertac Karaman
16.410-13
October 18th, 2010

Assignment

• Remember:
  • Problem Set #6 Propositional Logic, due next Wednesday, October 27th.
  • 16:413 Project Part 1: Sat-based Activity Planner, due Wednesday, November 3rd.

• Reading
  – Today: [AIMA] Chapter 10, re-read sections on SatPlan.
Planning problem

• Recall the planning problem:
  – Objects
    • robot1, robot2, load1, load2, room1
  – Predicates describing properties of objects
    • (IN ?robot ?room), (HAS ?robot ?load)
  – Actions as means to change these properties
    • Navigate (?robot, ?room_from, ?room_to)
  – Initial condition
  – Goal statement

Propositional logic SAT problem

• Recall the SAT problem:
  – Given a set of clauses, find an assignment to all propositions to satisfy all the clauses.
    \[ p_1 \lor \neg p_2 \lor p_3 \]
    \[ \neg p_1 \lor \neg p_2 \lor p_4 \]
    \[ \neg p_3 \lor p_4 \lor p_5 \]

• SAT solvers are very powerful.
  • Can process problems with tens of thousands of variables
Encoding planning as SAT

- Idea:
  - Define propositions for predicates and decisions
  - Encode problem description in propositional logic

\[ \text{initial state} \land \text{all possible action descriptions} \land \text{goal} \]

Encoding planning as SAT

- **Initial condition**
- Encode the truth of predicates:
  \[ (\text{IN robot1 bedroom})^0 \land (\text{IN robot2 kitchen})^0 \]

- Remember to include those that are false:
  \[ \neg(\text{IN robot1 kitchen})^0 \land \neg(\text{IN robot2 bedroom})^0 \]
Encoding planning as SAT

• **Actions**

  • **Straightforward approach:**
    - One proposition for each action:
      \[
      \text{Navigate(robot1, bedroom, kitchen)}^0
      \]
    - True if robot navigates from *bedroom* to *kitchen* at time 0

\[
\begin{align*}
\text{(IN robot1 kitchen)}^1 & \iff \\
& ((\text{IN robot1 kitchen})^0 \land \neg (\text{Navigate(robot1, kitchen, bedroom})^0 \land (\text{IN robot1 kitchen})^0)) \\
& \lor (\text{Navigate(robot1, bedroom, kitchen})^0 \land (\text{IN robot1 bedroom})^0))
\end{align*}
\]

Robot was in the kitchen at time 0 and did not leave the kitchen at time 0.
Robot was in the bedroom at time 0 and left the bedroom to go to kitchen at time 0.

---

Encoding planning as SAT

• **Actions**

  • **What may go wrong?**

    \[
    \text{Navigate(robot1, kitchen, bedroom)}^0
    \]

    -- However, robot1 is not in the kitchen at time 0!

  • **Precondition axioms:**

    \[
    \text{Navigate(robot1, kitchen, bedroom)}^0 \Rightarrow (\text{IN robot1 kitchen})^0
    \]
Encoding planning as SAT

• **Actions**

• What else may go wrong?
  
  Navigate(robot1, kitchen, bedroom)^0
  Navigate(robot1, bedroom, livingroom)^0

• Ensure that one action can be taken at a time:
  
  ¬((Navigate(robot1, kitchen, bedroom)^0 ∧ Navigate(robot1, bedroom, livingroom)^0))

---

Encoding planning as SAT

• **Outline of the algorithm:**
  
  – Check satisfiability for increasing number of steps

  \[ i = 1 \]

  If satisfiable for \( i \) steps then
  
  construct the solution

  Else
  
  \[ i = i + 1 \]