CSE 591: Human-aware Robotics

Instructor: Dr. Yu ("Tony") Zhang

Location & Times: CAVC 359, Tue/Thu, 9:00--10:15 AM
Office Hours: BYENG 558, Tue/Thu, 10:30--11:30AM

Sep 22, 2016

Slides adapted from Henry Kautz and Oregon State University

This set of slides borrow from various online sources; it is used for educational purposes only.
Plan/Goal Recognition

Action plan:
1. Insert message in bottle....

Bruce Woodcock

PLAN, ACTIVITY, AND INTENT RECOGNITION
THEORY AND PRACTICE
GITA SUKTHANKAR
ROBERT P. GOLDMAN
CHRISTOPHER GEIB
DAVID PYNADATH
HUNG BUI

Ira A. Fulton
Schools of Engineering
ARIZONA STATE UNIVERSITY
Approaches to goal/plan recognition

- **Consistency-based**
  - Hypothesize & revise
  - Closed-world reasoning
  - Version spaces

- **Probabilistic**
  - Stochastic grammars
  - Pending sets
  - Layered hidden Markov models
  - Policy recognition
  - Hierarchical hidden semi-Markov models
  - Dynamic probabilistic relational models
  - Dynamic Bayes networks
  - Example application: Assisted Cognition

Can be complementary..
First pick the consistent plans, and check which of them is most likely (tricky if the agent can make errors)
Hypothesize & Revise

Based on psychological theories of human narrative understanding

Mention of objects suggest hypothesis

Pursue single hypothesis until matching fails

The Plan Recognition Problem C. Schmidt, 1978
Closed-world reasoning

- Infers the minimum set(s) of independent plans that entail the observations
- Observations may be incomplete
- Infallible agent
- Complete plan library

Version Space Algebra

- Recognizes novel plans
- Complete observations
- Sensitive to noise
- Start with all goals
- Remove goals when actions are not consistent

A sound and fast goal recognizer Lesh & Etzioni

ASU Ira A. Fulton Schools of Engineering Arizona State University
Approaches to goal/plan recognition

• Consistency-based
  – Hypothesize & revise
  – Closed-world reasoning
  – Version spaces

• Probabilistic
  – Stochastic grammars
  – Pending sets
  – Layered hidden Markov models
  – Policy recognition
  – Hierarchical hidden semi-Markov models
  – Dynamic probabilistic relational models
  – Dynamic Bayes networks
  – Example application: Assisted Cognition

Can be complementary...
First pick the consistent plans, and check which of them is most likely (tricky if the agent can make errors)
## Stochastic grammars

CF grammar w/ probabilistic rules 

Successful for highly structured tasks (e.g. playing cards) 

Problems: errors, context

<table>
<thead>
<tr>
<th>Production Rules</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>S → AB [1.0]</td>
<td>Blackjack → “play game” “determine winner”</td>
</tr>
<tr>
<td>A → CD [1.0]</td>
<td>play game → “setup game” “implement strategy”</td>
</tr>
<tr>
<td>B → EP [1.0]</td>
<td>determine winner → “eval. strategy” “cleanup”</td>
</tr>
<tr>
<td>C → HL [1.0]</td>
<td>setup game → “place bets” “deal card pairs”</td>
</tr>
<tr>
<td>D → GK [1.0]</td>
<td>implement strategy → “player strategy”</td>
</tr>
<tr>
<td>E → LKM [0.6]</td>
<td>eval. strategy → “dealer down-card” “dealer hits” “player down-card”</td>
</tr>
<tr>
<td>K → LM [0.4]</td>
<td>eval. strategy → “dealer down-card” “player down-card”</td>
</tr>
<tr>
<td>F → NO [0.5]</td>
<td>cleanup → “settle bet” “recover card”</td>
</tr>
<tr>
<td>G → ON [0.5]</td>
<td>→ “recover card” “settle bet”</td>
</tr>
<tr>
<td>J → E [0.8]</td>
<td>player strategy → “Basic Strategy”</td>
</tr>
<tr>
<td>H → HJ [0.1]</td>
<td>→ “Splitting Pair”</td>
</tr>
<tr>
<td>H → f [0.5]</td>
<td>→ “Doubling Down”</td>
</tr>
<tr>
<td>I → fffI [0.5]</td>
<td>place bets</td>
</tr>
<tr>
<td>J → fffI [0.5]</td>
<td>deal card pairs</td>
</tr>
<tr>
<td>J → ee [0.5]</td>
<td></td>
</tr>
<tr>
<td>J → f [0.8]</td>
<td>Basic strategy</td>
</tr>
<tr>
<td>J → fJ [0.2]</td>
<td></td>
</tr>
<tr>
<td>K → e [0.6]</td>
<td>house hits</td>
</tr>
<tr>
<td>K → eK [0.4]</td>
<td></td>
</tr>
<tr>
<td>L → ae [1.0]</td>
<td>Dealer downcard</td>
</tr>
<tr>
<td>M → dh [1.0]</td>
<td>Player downcard</td>
</tr>
<tr>
<td>N → jn [1.6]</td>
<td>settle bet</td>
</tr>
<tr>
<td>N → kN [1.6]</td>
<td></td>
</tr>
<tr>
<td>N → j [0.16]</td>
<td></td>
</tr>
<tr>
<td>N → jn [1.6]</td>
<td></td>
</tr>
<tr>
<td>N → k [0.18]</td>
<td></td>
</tr>
<tr>
<td>O → k [0.25]</td>
<td>recover card</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Domain-Specific Events (Terminals)</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>dealer removed card from house</td>
</tr>
<tr>
<td>b</td>
<td>dealer removed card from player</td>
</tr>
<tr>
<td>c</td>
<td>player removed card from house</td>
</tr>
<tr>
<td>d</td>
<td>player removed card from player</td>
</tr>
<tr>
<td>e</td>
<td>dealer added card to house</td>
</tr>
<tr>
<td>f</td>
<td>dealer dealt card to player</td>
</tr>
<tr>
<td>g</td>
<td>player added card to house</td>
</tr>
<tr>
<td>h</td>
<td>player added card to player</td>
</tr>
<tr>
<td>i</td>
<td>dealer removed chip</td>
</tr>
<tr>
<td>j</td>
<td>player removed chip</td>
</tr>
<tr>
<td>k</td>
<td>dealer pays player chip</td>
</tr>
<tr>
<td>l</td>
<td>player bets chip</td>
</tr>
</tbody>
</table>

Pending sets

Explicitly models the agent’s “plan agenda” using Poole’s “probabilistic Horn abduction” rules

Handles multiple concurrent interleaved plans & negative evidence

Number of different possible pending sets can grow exponentially

Context problematic?
Metric time?

- **A new model of plan recognition.** Goldman, Geib, and Miller
- **Probabilistic plan recognition for hostile agents.** Geib, Goldman
Layered hidden Markov models

Cascade of HMM’s, operating at different temporal granularities

Inferential output at layer K is “evidence” for layer K +1

- N. Oliver, E. Horvitz, and A. Garg.
Policy Recognition

Model agent using hierarchy of abstract policies (e.g. abstract by spatial decomposition)

Compute the conditional probability of top-level policy given observations

Compiled into DBN

- **Tracking and Surveillance in Wide-Area Spatial Environments Using the Hidden Markov Model.** Hung H. Bui, Svetla Venkatesh and West.

Hierarchical hidden semi-Markov models

Combine hierarchy (function call semantics) with metric time
Compile to DBN
Time nodes represent a distribution over the time of the next state “switch”
“Linear time” smoothing
– Research issues – parametric time nodes, varying granularity

- **Hidden semi-Markov models (segment models)**
Dynamic probabilistic relational models


PRM - reasons about classes of objects and relations

Lattice of classes can capture plan abstraction

DPRM – efficient approximate inference by Rao-Blackwellized particle filtering

Open: approximate smoothing?
Dynamic Bayesian Network

Time and Change in Probabilistic Reasoning
Markov processes (Markov chains)

Construct a Bayes net from these variables: parents?

**Markov assumption:** $X_t$ depends on **bounded** subset of $X_{0:t-1}$

**First-order Markov process:** $P(X_t | X_{0:t-1}) = P(X_t | X_{t-1})$

**Second-order Markov process:** $P(X_t | X_{0:t-1}) = P(X_t | X_{t-2}, X_{t-1})$

**Sensor Markov assumption:** $P(E_t | X_{0:t}, E_{0:t-1}) = P(E_t | X_t)$

**Stationary process:** transition model $P(X_t | X_{t-1})$ and sensor model $P(E_t | X_t)$ fixed for all $t$
Time and uncertainty

The world changes; we need to track and predict it

Diabetes management vs vehicle diagnosis

Basic idea: copy state and evidence variables for each time step

\[ X_t = \text{set of unobservable state variables at time } t \]

\[ \text{e.g., } BloodSugar_t, StomachContents_t, \text{ etc.} \]

\[ E_t = \text{set of observable evidence variables at time } t \]

\[ \text{e.g., } MeasuredBloodSugar_t, PulseRate_t, FoodEaten_t \]

This assumes \textbf{discrete time}; step size depends on problem

Notation: \[ X_{a:b} = X_a, X_{a+1}, \ldots, X_{b-1}, X_b \]
Example

Issues: 1-order Markov; huge CPT table

Possible fixes:
1. **Increase order** of Markov process
2. **Augment state**, e.g., add $Temp_t$, $Pressure_t$

Example: robot motion.
Augment position and velocity with $Battery_t$
Dynamic Bayesian networks are “templates” for specifying the relation between the values of a random variable across time-slices. e.g. How is Rain at time t related to Rain at time t+1? We call them templates because they need to be expanded (unfolded) to the required number of time steps to reason about the connection between variables at different time points.

\[ X_t, E_t \] contain arbitrarily many variables in a replicated Bayes net.
Special Cases of DBNs are well known in the literature

- **Restrict number of variables per state**
  - Markov Chain: DBN with one variable that is fully observable
  - Hidden Markov Model: DBN with only one state variable that is hidden and can be estimated through evidence variable(s)

- **Restrict the type of CPD**
  - Kalman Filters: DBN where the system transition function as well as the observation variable are *linear gaussian*
  - The advantage of Gaussians is that the posterior distribution remains Gaussian
Kalman filters

Modelling systems described by a set of continuous variables, e.g., tracking a bird flying—$X_t = X, Y, Z, \dot{X}, \dot{Y}, \dot{Z}$.

Airplanes, robots, ecosystems, economies, chemical plants, planets, ...

Gaussian prior, linear Gaussian transition model and sensor model
DBNs vs Kalman filters

Every Kalman filter model is a DBN, but few DBNs are KFs; real world requires non-Gaussian posteriors

E.g., where are bin Laden and my keys? What’s the battery charge?
Plan Recognition Approaches based on setting up DBNs
Dynamic Bayes nets

Models relationship between user’s recent actions and goals (help needs)

Probabilistic goal persistence


Towards a Bayesian model for keyhole plan recognition in large domains Albrecht, Zukermann, Nicholson, Bud
Dynamic Bayesian Nets

Learning and Inferring Transportation Routines Lin Liao, Dieter Fox, and Henry Kautz, Nineteenth National Conference on Artificial Intelligence, San Jose, CA, 2004.
Applications
Assisted cognition

Computer systems that improve the independence and safety of people suffering from cognitive limitations by...

- **Understanding** human behavior from low-level sensory data
  - Using commonsense knowledge
  - Learning individual user models
- **Actively** offering prompts and other forms of help as needed
- **Alerting** human caregivers when necessary

Activity Compass

• Zero-configuration personal guidance system
  – Learns model of user’s travel on foot, by public transit, by bike, by car
  – Predicts user’s next destination, offers proactive help if lost or late

• Integrates user data with external constraints
  – Maps, bus schedules, calendars, ...
  – EM approach to clustering & segmenting data

The Activity Compass  Don Patterson, Oren Etzioni, and Henry Kautz (2003)
Activity of daily living monitor & prompter

Recognizing unexpected events using online model selection

- User errors, abnormal behavior
- Select model that maximizes likelihood of data:
  - Generic model
  - User-specific model
  - Corrupt (impaired) user model
- Neurologically-plausible corruptions
  - Repetition
  - Substitution
  - Stalling

Fox, Kautz, & Shastri