Decision Making in Robots and Autonomous Agents

Nudging and User Modelling Case Study

Subramanian Ramamoorthy
School of Informatics

3 April, 2015
Diversity in Modes of Thinking

• Automatic System (Type 1)
  – Fast, unconscious
  – Parallel, associative
  – Low energy
  – “Doer”

• Reflective System (Type 2)
  – Slow, conscious
  – Serial
  – Expensive
  – “Planner”
Training To Go Between Modes
Example: Board Memory in Chess

https://www.youtube.com/watch?v=rWuJqCwfjjc
Utilising Type I Thinking

• from behavioral science
• argues that positive reinforcement and indirect suggestions to try to achieve non-forced compliance can influence motives, incentives and decision making
• at least as effectively – if not more effectively - than direct instruction, legislation, or enforcement
Application Example:
Persuasive Technology (Beeminder)
Application Example: Changing Driving Behaviour - Lottery Incentive

- Enter people in a lottery if they come to campus off-peak
- Observe that this small probability of earnings actually causes non-essential travellers to alter their behaviour
Example from Games Industry: Dynamic Difficulty Adjustment

- Alter attributes of game to keep players in the sweet spot
- Can’t observe their ‘flow’, but can use proxy variables such as ‘inventory level’ of variables such as resources
- *Architect choices* in order to achieve optimal level of play in the game

[Hunicke ACE ‘05]
Case Study:

Problem: Learn to Interact
Technical Problem

• Many configuration parameters
• Performance = f(configuration parameters, user parameters)
• Diverse User base
  – Varied performance for any given configuration setting

• Issue: Mismatch between users’ skill and settings
  – Sub-optimal performance (e.g., if joystick too sensitive or not enough)
  – Perceptual: how precisely is the user aware of computational context?

Goal of this paper:
Model and algorithm for choosing action set(tings) adaptively, online
Interaction Shaping as Action Set Selection

- User $\leftrightarrow$ MDP
  - Invokes optimal policy immediately when given an action set
  - i.e., we assume introspection via planning
- User skill level or type determines transition function
- Learning agent gets to change action set, $\alpha$, as interaction proceeds, including within episode
Latent variable MDP Model

• Subject is modelled as a parameterised MDP:

\[ M \triangleq \{(S, A, R, T(\cdot; \tau), R, \gamma) \mid \tau \in Sk\} \]

• They differ in the transition function, \( T(s'|s, a; \tau) \)

• For instance:
  – Poorly skilled user only makes coarse moves (irrespective of joystick sensitivity)
  – Highly skilled user, if given sensitive joystick, executes sharp turns
Latent Variable Model of Users - Action

• Action set is large but partitioned, $AS$ being the set of all cells
• Any two cells, $\alpha, \alpha' \in AS$, are disjoint and

$$\bigcup_{\alpha \in AS} \alpha = A.$$  

• The action set determines what the user could possibly do,

• What they actually do depends on their latent skill level
Latent Variable Model of Users

• The value function of a policy $\pi$ depends on type $\tau$, $V_\tau^{\pi}$
• We assume an arbitrary but fixed start state, for presentation

• For a given action set, we then write

\[ V_\tau^\alpha \triangleq \max_\pi V_\tau^{\pi} \]

• And define a corresponding $\alpha$-specific optimal policy as

\[ \pi_\alpha^* \triangleq \arg \max_\pi V_\tau^{\pi} \]
Algorithm Flow

- Agent solves *sequence* of MDPs from M, in collaboration with learner who maximizes future expected discounted reward

At the beginning of a problem phase,
- User arrives with type $\tau^*$, as per a distribution $K_1(\tau)$
- Learner chooses an action set, $\alpha$, restricting agent’s actions

- Learner can’t see actual $\tau^*$, but knows $T$, $R$ for any $\tau$
- Agent knows $T$, $R$, $\tau^*$; once given $\alpha$, acts according to

$$\pi^*_\alpha \triangleq \arg\max_\pi V^\pi_{\tau^*}$$
Learner’s Objective

• An optimal action set

\[ \alpha^* \triangleq \arg \max_{\alpha} V_{\tau^*}^\alpha \]

• A priori optimal action set

\[ \alpha_* \triangleq \arg \max_{\alpha} \mathbb{E}_{K_1(\tau)} [V_{\tau}^\alpha] \]

• A posteriori optimal action set ...
Determining Agent Type

• Assume we know user types, \( S_k = \{\tau_1, \tau_2, \cdots, \tau_n\} \)
  – In experiments, we get this separately by off-line learning from a corpus of observed user data

• At time \( t \), we have observed the sequence,

\[
sa_{0:t} \triangleq s_0a_0s_1a_1\cdots s_t \text{ (with } sa_{0:0} \triangleq s_0)\]
Bayes Optimal Action Set

- We start with the likelihood of a type given history,

\[ L(\tau_i|s_{0:t}) = \prod_{i=0}^{t-1} T(s_{i+1}|s_i, a_i; \tau_i) \]

and a posterior probability over types,

\[ Pr(\tau_i|sa_{0:t}) = L(\tau_i|sa_{0:t})W(\tau_i) \quad [W: \text{prior}] \]

- We can then define an action set maximising expected value,

\[ \alpha_{BO}(sa_{0:t}) \triangleq \arg \max_{\alpha} \sum_{i} Pr(\tau_i|sa_{0:t})V_{\tau_i}^\alpha \]
Experiments: Setup

An experimental game domain

- User must move ball from a start to a goal location
- Ball moves with constant speed \( z \), user picks direction
- Loss when reaching goal = normalised time + normalised num. collisions

- **Two skill issues:**
  - Perceptual (ball size)
  - Motor (noisy ball motion)
Some Experiment Details

- Different ‘speed’ for actions: $z \in \{30, 40, 50, 60, 70\}$
- Constant action noise for ball: ball moves requested amount + Gaussian noise

- Ball size determines type: $b \in \{2, 5, 10, 20, 40, 60\}$
- Type dependent noise level: $n_b = b/2\%$

- Type determines transition probability:
  - By noise levels
  - Ball size determines where collisions happen (e.g., fat ball needs more clearance)
Performance Profiles of Users

Simulated Users

Human Users

3/4/2015
Convergence

Simulated Users

Human Users
Population Diversity Experiment

• Compare against two baselines: \( EXP-3 \) and \( \alpha_\ast \) (pop-optimal)

• We perform comparative experiments as follows:
  – Define different levels of heterogeneity in user population by defining different mixtures of types (ball sizes, etc.)
  – Population diversity corresponds to all possible, \( 5C_x \), combinations of types

• All results involve averaged performance over several runs
Population Diversity Experiment: Results

**Simulated Users**

- Blue: $\alpha_3$
- Green: BOA
- Red: EXP-3

**Human Users**

- Blue: $\alpha_3$
- Green: BOA
Points to Ponder

• What did we assume of the user model?
  – What does MDP user model imply?
• What would a more general user model look like?
  – How would it alter our methodology?
• What about “behavioural effects”?
  – What might you change?