POMDPs

Hal Daumé III

Computer Science
University of Maryland

me@hal3.name

CS 421: Introduction to Artificial Intelligence

13 Mar 2012

Many slides courtesy
Of Vincent Conitzer,
Minqing Hu and
Mayaan Roth

Announcements

- Next HW is up – mostly to help with midterm!
- Previous midterms are up
- P2 results will be out by Thursday

Hal Daumé III (me@hal3.name)   CS421: Intro to AI

Background on Solving POMDPs

- MDPs policy: to find a mapping from states to actions

- POMDPs policy: to find a mapping from probability distributions (over states) to actions.
    - belief state: a probability distribution over states
    - belief space: the entire probability space, infinite

3

Partially observable Markov decision processes

- Markov process + partial observability = HMM
- Markov process + actions = MDP
- Markov process + partial observability + actions = HMM + actions = MDP + partial observability = **POMDP**

	full observability	*partial observability*
no actions	**Markov process**	**HMM**
actions	**MDP**	**POMDP**

Policies in MDP

- *k*-horizon Value function:

$$V_t^{\delta_t}(s_i) = q_i^{\delta_t(s_i)} + \beta \sum_j p_{ij}^{\delta_t(s_i)} V_{t-1}^{\delta_{t-1}}(s_j)$$

- Optimal policy δ^*, is the one where, for all states, s_i and all other policies,

$$V^{\delta^*}(s_i) \geq V^{\delta}(s_i)$$

Hal Daumé III (me@hal3.name)   CS421: Intro to AI

Finite k-horizon POMDP

- POMDP: <S, A, P, Z, R, W>
- transition probability: p_{ij}^a
- probability of observing *z* after taking action *a* and ending in state s_j: w_{ijz}^a
- immediate rewards: r_{jz}^a
- Immediate reward of performing action *a* in state s_i:

$$q_i^a = \sum_{j,z} p_{ij}^z r_{jz}^a w_{ijz}^a$$

- Object: to find an optimal policy for finite k-horizon POMDP

$$\delta^* = (\delta_1, \delta_2, \ldots, \delta_k)$$

A two state POMDP

➢ represent the belief state with a single number *p*.

➢ the entire space of belief states can be represented as a line segment.

belief space for a 2 state POMDP

belief state updating

- finite number of possible next belief states, given a belief state
  - a finite number of actions
  - a finite number of observations
- b' = T(b| a, z). Given a and z, b' is fully determined.
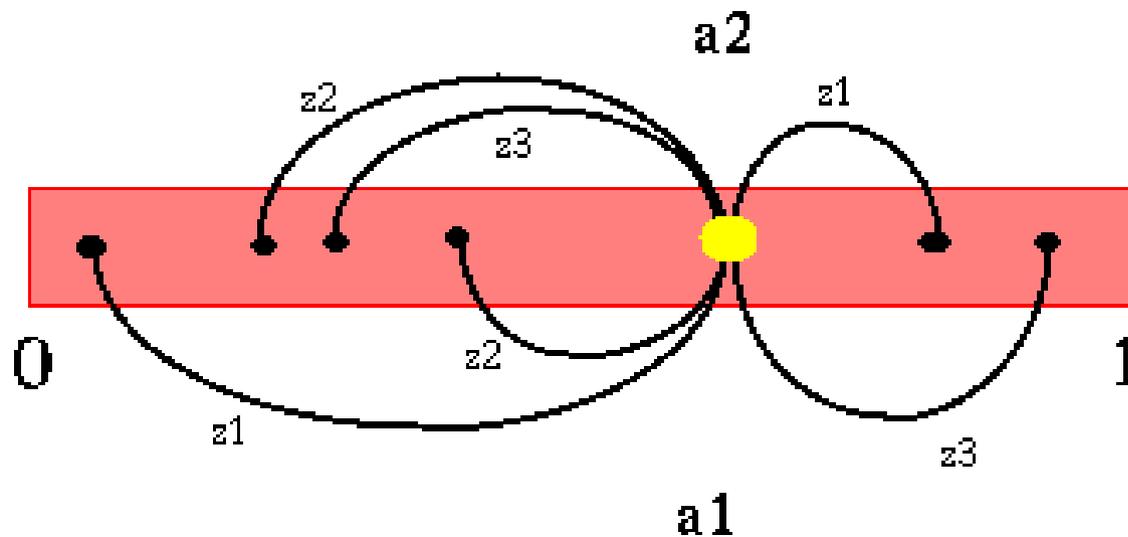
➢ the process of maintaining the belief state is Markovian: the next belief state depends only on the current belief state (and the current action and observation)

➢ we are now back to solving a MDP policy problem with some adaptations

Hal Daumé III (me@hal3.name)   CS421: Intro to AI

➢ continuous space: value function is some arbitrary function
  ➢ b: belief space
  ➢ V(b): value function

➢ Problem: how we can easily represent this value function?
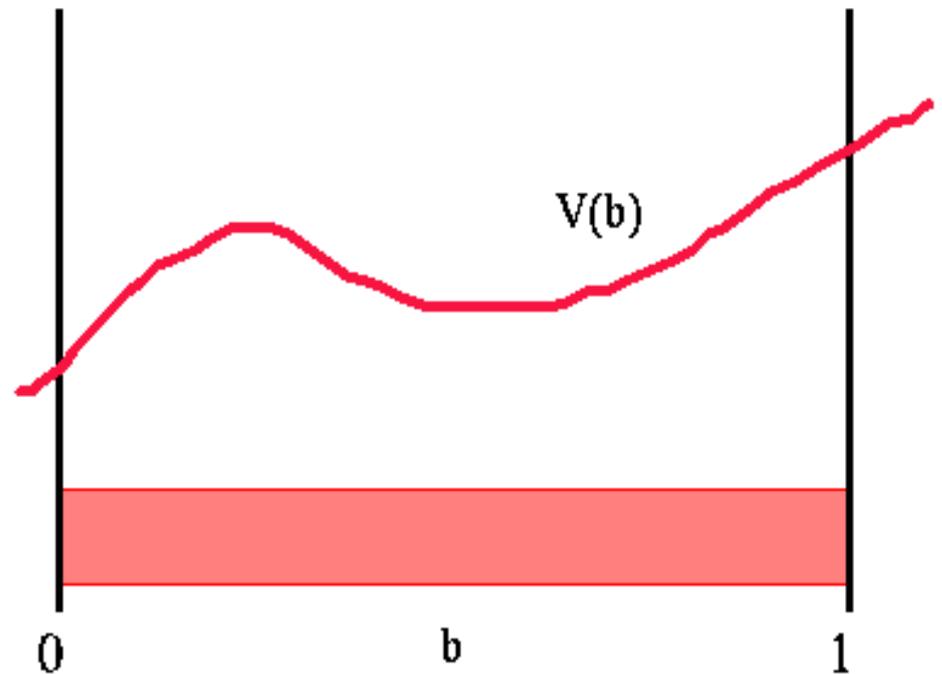
Value function over belief space

Fortunately, the finite horizon value function is piecewise linear and convex (PWLC) for every horizon length.
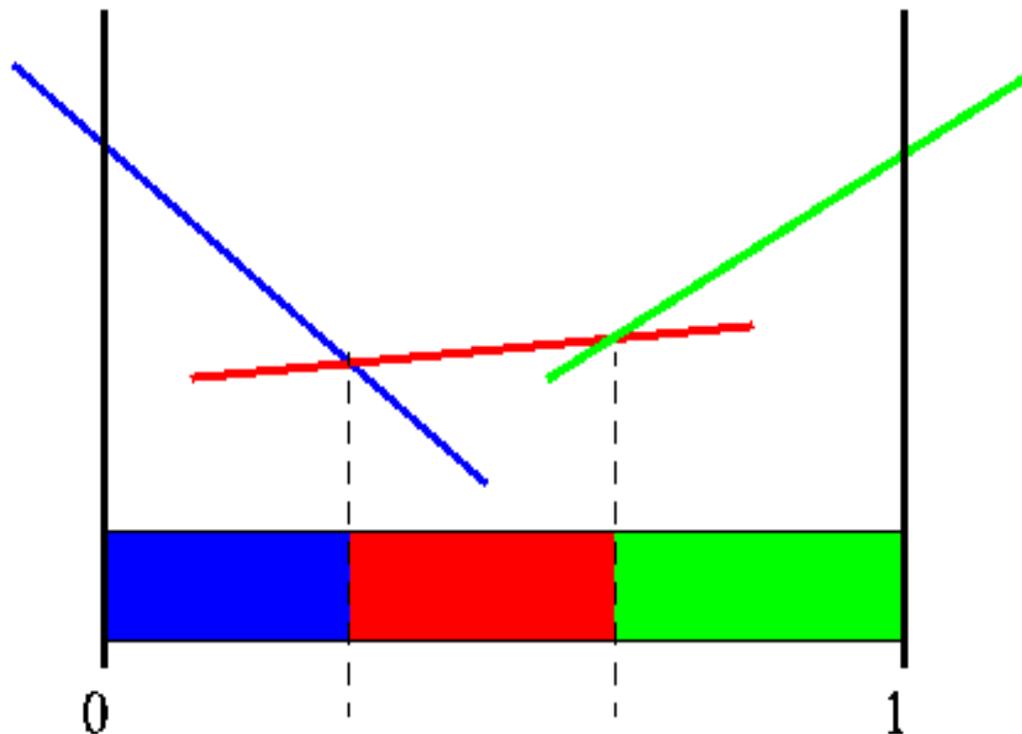
Sample PWLC function

- A Piecewise Linear function consists of linear, or hyper-plane segments
  - Linear function:

  $$\sum \alpha_i x_i = \alpha_0 x_0 + \alpha_1 x_1 + ... + \alpha_N x_N$$

  - Kth linear segment:

  $$\sum_{i=0}^{N} \alpha_i^k x_i$$

  - the α -vector

  $$\alpha^k = [\alpha_0^k, \alpha_1^k, ..., \alpha_N^k]$$

  - each liner or hyper-plane could be represented with

  $$\alpha^k(t)$$

12

➤ Value function:

$$V_t^*(b) = \max_k \sum_i b_i \alpha_i^k(t)$$

➤ a convex function

- 1-horizon POMDP problem
  - Single action *a* to execute
  - Starting out belief state *b*
  - Ending belief state *b'*
    - *b' = T(b | a, z)*
  - Immediate rewards q_i^a
  - Terminating rewards q_i^0 for state s_i

   Expected terminating reward in *b'*

$$V_0(b') = \sum_i b_i' q_i^0$$

14

Geometric interpretation of value fn

➢ $|S| = 2$
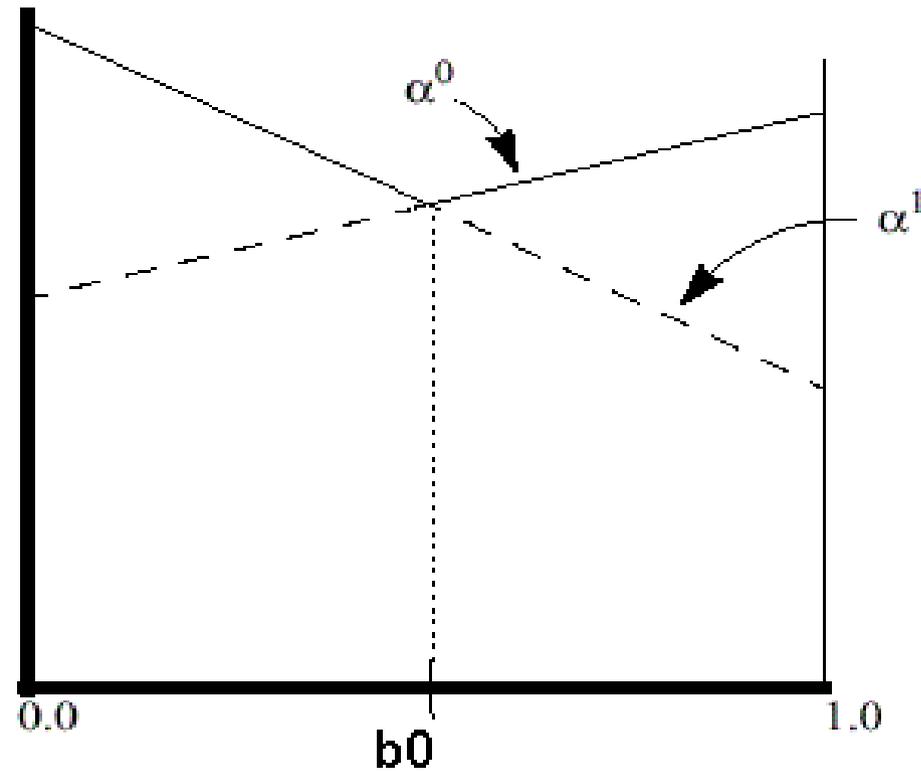
Sample value function for |S| = 2
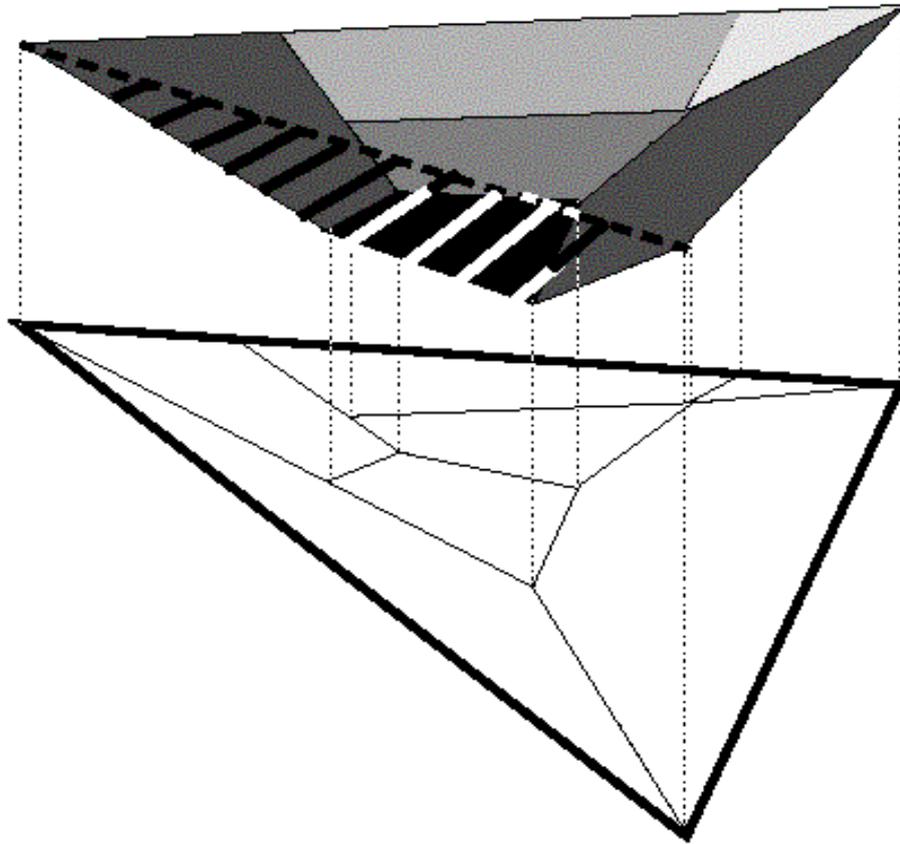
- ➢ |S| = 3
- ➢ Hyper-planes
- ➢ Finite number of regions over the simplex

Sample value function for |S| = 3

Multi-Agent Coordination

- Teams:
  - Agent work together to achieve a common goal
  - No individual motivations

- Objective:
  - Generate policies (individually or globally) to yield best team performance

Observability

- "Observability" - degree to which agents can, either individually or as a team, identify current world state
  - Individual observability
    - MMDP [Boutilier, 1996]
  - Collective observability
    - $O_1 + O_2 + \ldots + O_n$ uniquely identify the state
    - e.g. [Xuan, Lesser, Zilberstein, 2001]
  - Collective partial observability
    - DEC-POMDP [Bernstein et al., 2000]
    - POIPSG [Peskin, Kim, Meuleau, Kaelbling, 2000]
  - Non-observability

Communication

- "Communication" - explicit message-passing from one agent to another
  - Free Communication
    - No cost to send messages
    - Transforms MMDP to MDP, DEC-POMDP to POMDP
  - General Communication
    - Communication is available but has cost or is limited
  - No Communication
    - No explicit message-passing

Taxonomy of Cooperative MAS

Hal Daumé III (me@hal3.name)   CS421: Intro to AI
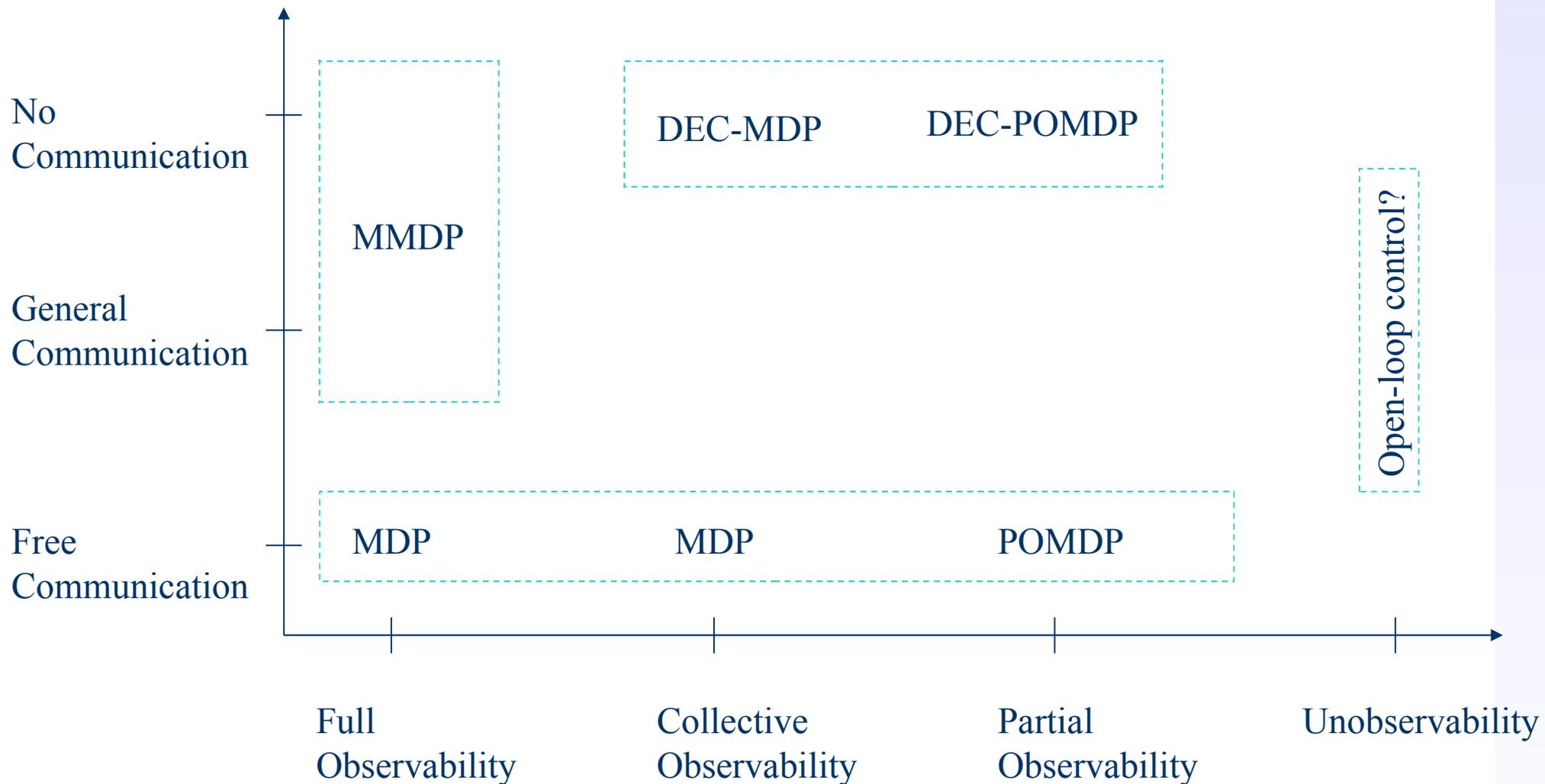
Complexity

Polynomial time
Ex: circuit evaluation

Nondet. Exponential Time
= NTIME($2^{p(n)}$)
Worse than NP

	Individually Observable	Collectively Observable	Collectively Partially Observable
No Comm.	P-complete	NEXP-complete	NEXP-complete
General Comm.	P-complete	NEXP-complete	NEXP-complete
Free Comm.	P-complete	P-complete	PSPACE-complete

Polynomial space
Ex: Quant SAT
Contained in EXP

Multi-Agent MDP (MMDP)

➢ Also called IPSG (identical payoff stochastic games)

➢ $M = <S, \{A_i\}_{i \in m}, T, R>$

  ➢ **S** is set of possible world states
  ➢ $\{A_i\}_{i \in m}$ is set of joint actions, $<a_1, \ldots, a_m>$ where $a_i \in A_i$
  ➢ **T** defines transition probabilities over joint actions
  ➢ **R** is team reward function

➢ State is fully observable by each agent

➢ P-complete

Multi-Agent POMDP

> **DEC-POMDP** [Bernstein et al., 2000], **MTDP** [Pynadath et al., 2002], **POIPSG** [Peshkin et al., 2000]

> $M = <S, \{A_i\}_{i \in m}, T, \{\Omega_i\}_{i \in m}, O, R>$

>> **S** is set of possible world states

>> $\{A_i\}_{i \in m}$ is set of joint actions, $<a_1, \ldots, a_m>$ where $a_i \in A_i$

>> **T** defines transition probabilities over joint actions

>> $\{\Omega_i\}_{i \in m}$ is set of joint observations, $<\omega_1, \ldots, \omega_m>$ where $\omega_i \in \Omega_i$

>> **O** defines observation probabilities over joint actions and joint observations

>> **R** is team reward function

Tiger Domain: (States, Actions)

➤ Two-agent tiger problem [Nair et al., 2003]:
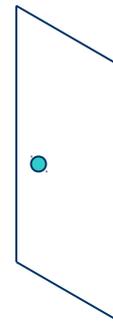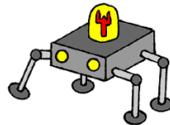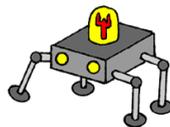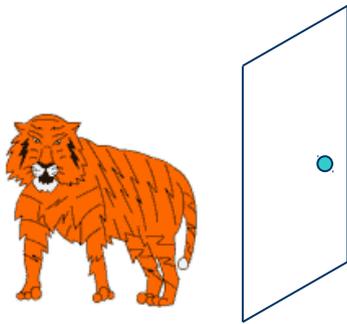
Individual Actions:

$a_i \in$ {OpenL, OpenR, Listen}

Robot can open left door, open right door, or listen

S: {SL, SR}

Tiger is either behind left door or behind right door

Tiger Domain: (Observations)
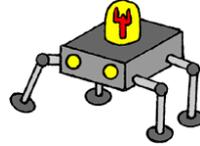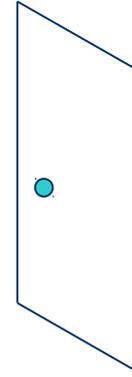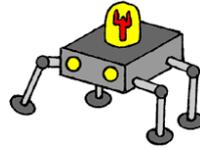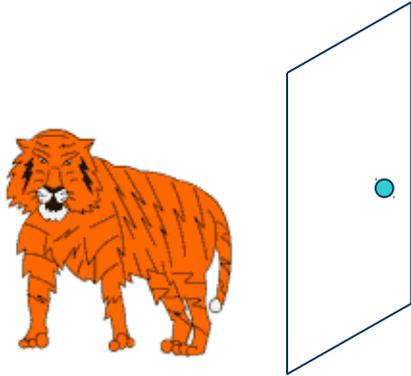
Individual Observations:

$\omega_I \in \{HL, HR\}$

Robot can hear tiger behind left door or hear tiger behind right door

Observations are noisy and independent.

Hal Daumé III (me@hal3.name)   CS421: Intro to AI

Tiger Domain: (Reward)

➢ Coordination problem – agents must act together for maximum reward

Listen has small cost (-1 per agent)

Both agents opening door with tiger leads to medium negative reward (-50)

Maximum reward (+20) when both agents open door with treasure

Minimum reward (-100) when only one agent opens door with tiger

Joint Beliefs

➢ Joint belief (b^t) – distribution over world states

➢ Why compute possible joint beliefs?

  ➢ action coordination

  ➢ transition and observation functions depend on joint action

  ➢ agent can't accurately estimate belief if joint action is unknown

➢ To ensure action coordination, agents can only reason over information known by all teammates

Hal Daumé III (me@hal3.name)

Possible Joint Beliefs

HL

HL

L^0

b: P(SL) = 0.5

p: p(b) = 1.0

ω = {}

a = <Listen, Listen>

b: P(SL) = 0.8	b: P(SL) = 0.5	b: P(SL) = 0.5	b: P(SL) = 0.2
p: p(b) = 0.29	p: p(b) = 0.21	p: p(b) = 0.21	p: p(b) = 0.29
ω = {HL,HL}	ω = {HL,HR}	ω = {HR,HL}	ω = {HR,HR}

L^1
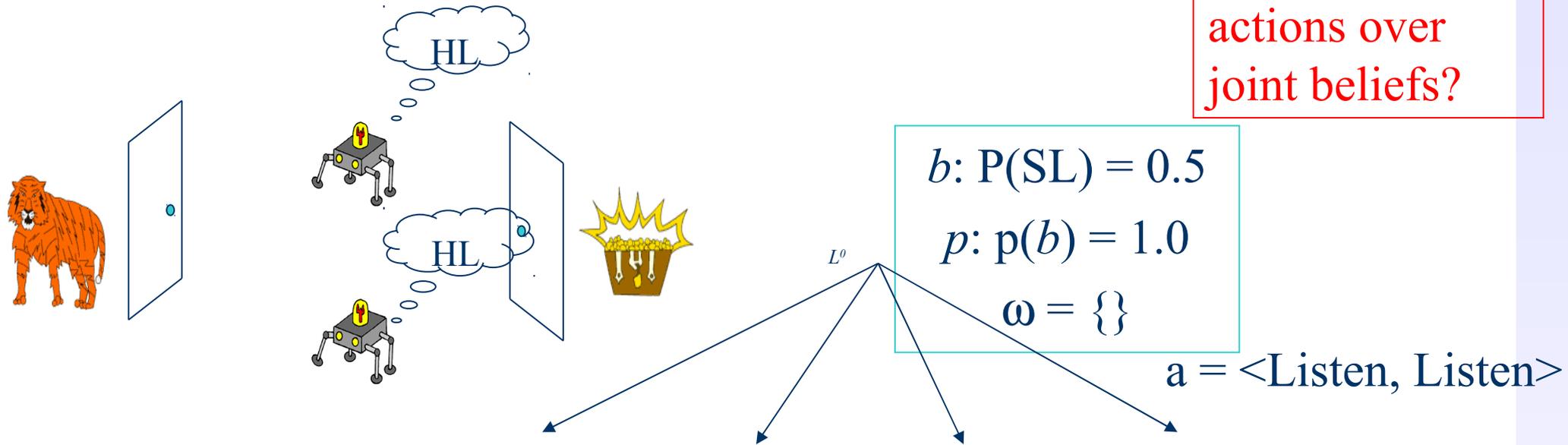
Q-POMDP Heuristic

$$Q_{POMDP}(L^t) = \arg\max_{a} \sum_{L_i^t \in L^t} p(L_i^t) \times Q(b(L_i^t), a)$$

Choose joint action by computing expected reward over all leaves
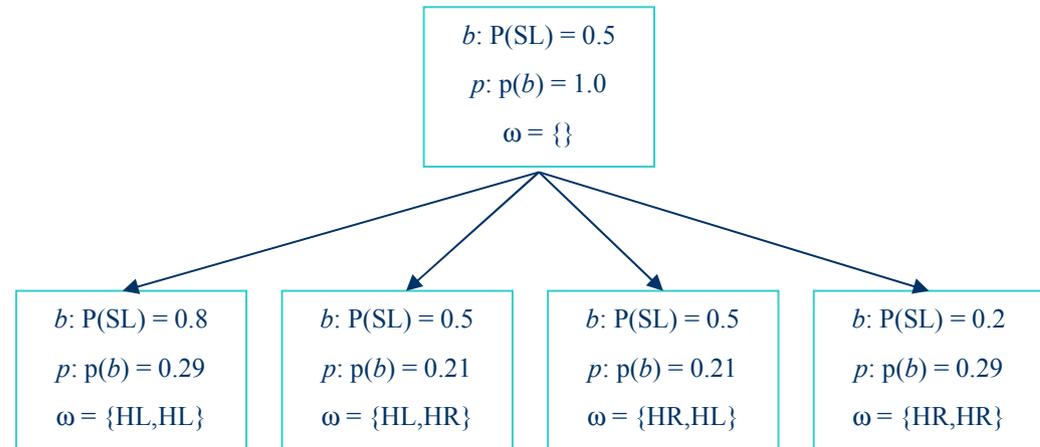
Agents will independently select same joint action…

but action choice is very conservative (always <Listen,Listen>)

DEC-COMM: Use communication to add local observations to joint belief

Dec-Comm Example
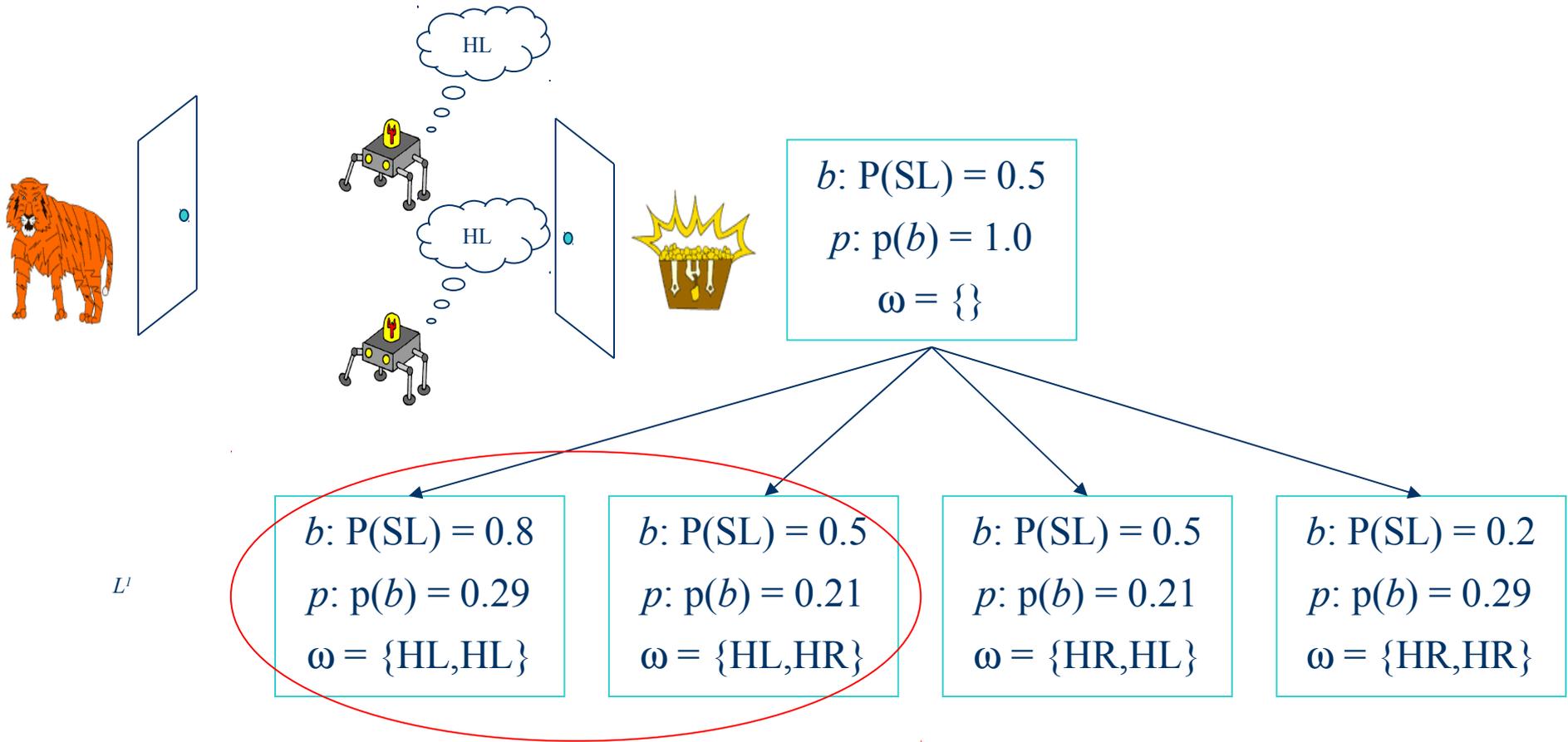
b: P(SL) = 0.5
p: p(b) = 1.0
ω = {}

b: P(SL) = 0.8
p: p(b) = 0.29
ω = {HL,HL}

b: P(SL) = 0.5
p: p(b) = 0.21
ω = {HL,HR}

b: P(SL) = 0.5
p: p(b) = 0.21
ω = {HR,HL}

b: P(SL) = 0.2
p: p(b) = 0.29
ω = {HR,HR}

L^1

a_{NC} = Q-POMDP(L^1) = <Listen,Listen>

$L*$ = circled nodes

a_C = Q-POMDP(L^*) = <Listen,Listen>

Don't communicate

Dec-Comm Example (cont'd)

b: P(SL) = 0.5
p: p(b) = 1.0
ω = {}

b: P(SL) = 0.85
p: p(b) = 0.29
ω = {HL,HL}

b: P(SL) = 0.5
p: p(b) = 0.21
ω = {HL,HR}

b: P(SL) = 0.5
p: p(b) = 0.21
ω = {HR,HL}

b: P(SL) = 0.15
p: p(b) = 0.29
ω = {HR,HR}

a = <Listen, Listen>

b: P(SL) = 0.97
p: p(b) = 0.12
ω:
{<HL,HL>,
<HL,HL>}

b: P(SL) = 0.85
p: p(b) = 0.06
ω:
{<HL,HL>,
<HL,HR>}

b: P(SL) = 0.85
p: p(b) = 0.06
ω:
{<HL,HL>,
<HR,HL>}

b: P(SL) = 0.5
p: p(b) = 0.04
ω:
{<HL,HL>,
<HR,HR>}

b: P(SL) = 0.85
p: p(b) = 0.06
ω:
{<HL,HR>,
<HL,HL>}

b: P(SL) = 0.5
p: p(b) = 0.04
ω:
{<HL,HR>,
<HL,HR>}

b: P(SL) = 0.5
p: p(b) = 0.04
ω:
{<HL,HR>,
<HR,HL>}

b: P(SL) = 0.16
p: p(b) = 0.06
ω:
{<HL,HR>,
<HR,HR>}
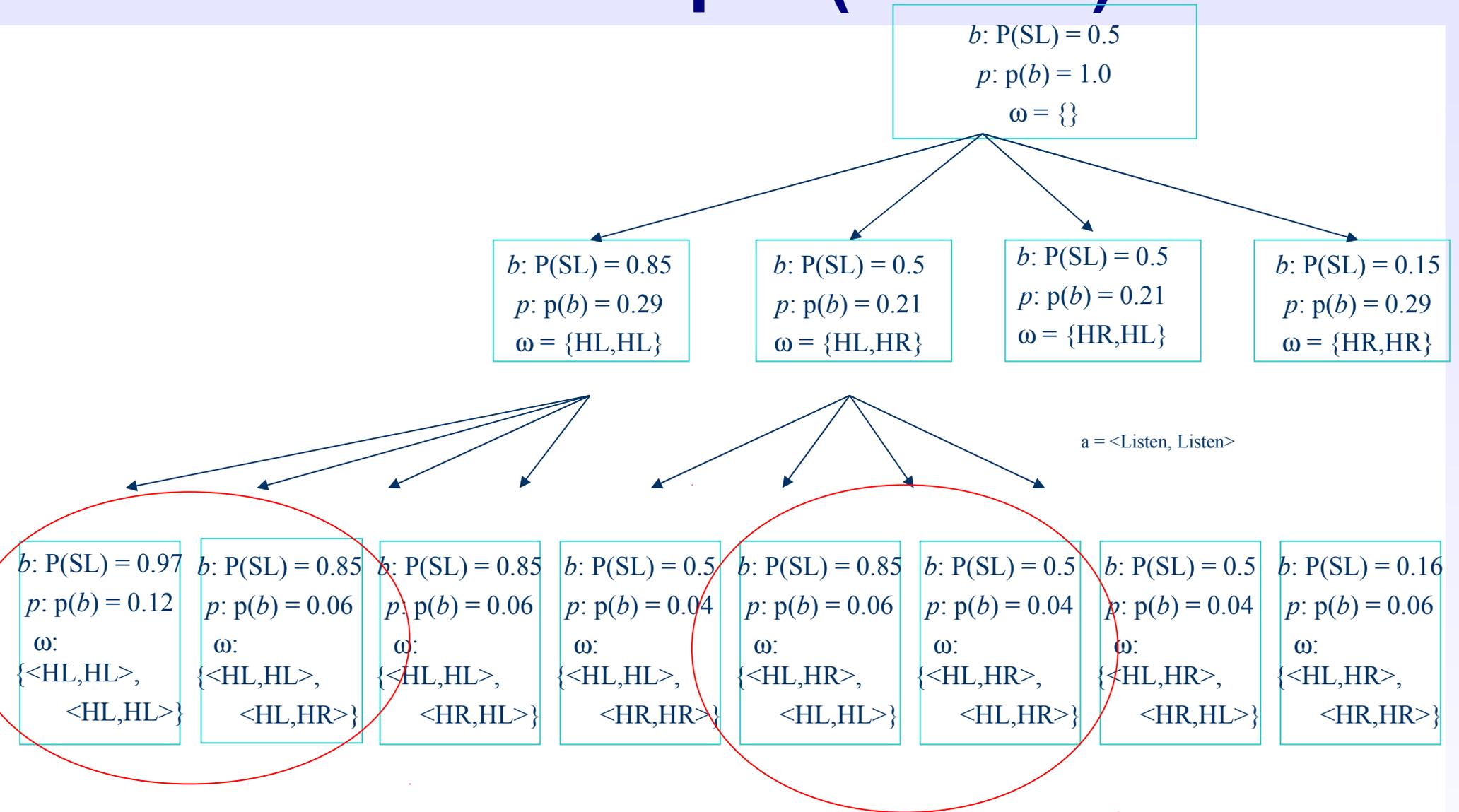
a_{NC} = Q-POMDP(L^2) = <Listen, Listen>
L^* = circled nodes
a_C = Q-POMDP(L^*) = <OpenR,OpenR>
$V(a_C)$ - $V(a_{NC})$ > ε

Agent 1 communicates

Dec-Comm Example (cont'd)

b: P(SL) = 0.5
p: p(b) = 1.0
ω = {}

b: P(SL) = 0.85
p: p(b) = 0.29
ω = {HL,HL}

b: P(SL) = 0.5
p: p(b) = 0.21
ω = {HL,HR}

b: P(SL) = 0.5
p: p(b) = 0.21
ω = {HR,HL}

b: P(SL) = 0.15
p: p(b) = 0.29
ω = {HR,HR}

L^1

a = <Listen, Listen>

b: P(SL) = 0.97
p: p(b) = 0.12
ω: {<HL,HL>,
<HL,HL>}

b: P(SL) = 0.85
p: p(b) = 0.06
ω: {<HL,HL>,
<HL,HR>}

b: P(SL) = 0.85
p: p(b) = 0.06
ω: {<HL,HL>,
<HR,HL>}

b: P(SL) = 0.5
p: p(b) = 0.04
ω: {<HL,HL>,
<HR,HR>}

b: P(SL) = 0.85
p: p(b) = 0.06
ω: {<HL,HR>,
<HL,HL>}

b: P(SL) = 0.5
p: p(b) = 0.04
ω: {<HL,HR>,
<HL,HR>}

b: P(SL) = 0.5
p: p(b) = 0.04
ω: {<HL,HR>,
<HR,HL>}

b: P(SL) = 0.16
p: p(b) = 0.06
ω: {<HL,HR>,
<HR,HR>}

L^2

...

<HL,HL>

<HL,HL>
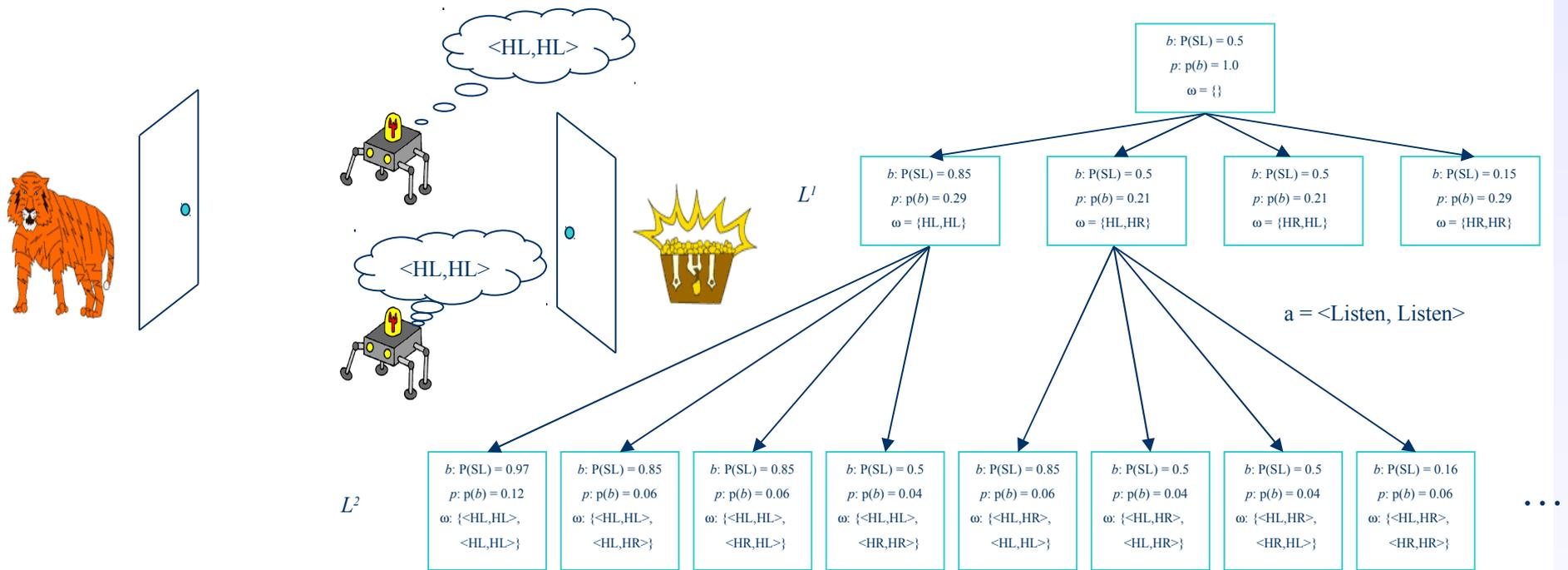
Agent 1 communicates <HL,HL>

Agent 2 communicates <HL,HL>

Q-POMDP(L^2) = <OpenR, OpenR>

Agents open right door!

References

➢ Becker, R., Zilberstein, S., Lesser, V., Goldman, C. V. Transition-independent decentralized Markov decision processes. In *International Joint Conference on Autonomous Agents and Multi-Agent Systems*, 2003.

➢ Bernstein, D., Zilberstein, S., Immerman, N. The complexity of decentralized control of Markov decision processes. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, 2000.

➢ Boutilier, C. Sequential optimality and coordination in multiagent systems. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, 1999.

➢ Hansen, E. A., Bernstein, D. S., Zilberstein, S. Dynamic programming for partially observable stochastic games. In *National Conference on Artificial Intelligence*, 2004.

➢ Nair, R., Roth, M., Yokoo, M., Tambe, M. Communication for improving policy computation in distributed POMDPs. To appear in *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems, 2004.*

➢ Nair, R., Pynadath, D., Yokoo, M., Tambe, M., Marsella, S. Taming decentralized POMDPs: Towards efficient policy computation for multiagent settings. In *Proceedings of International Joint Conference on Artificial Intelligence, 2003.*

➢ Peshkin, L., Kim, K.-E., Meuleau, N., Kaelbling, L. P. Learning to cooperate via policy search. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2000.

➢ Pynadath, D. and Tambe, M. The communicative multiagent team decision problem: Analyzing teamwork theories and models. In *Journal of Artificial Intelligence Research*, 2002.

➢ Roth, M., Simmons, R., Veloso, M. Reasoning about joint beliefs for execution-time communication decisions. To appear in *International Joint Conference on Autonomous Agents and Multi-Agent Systems*, 2005.

➢ Xuan, P., Lesser, V., Zilberstein, S. Communication decisions in multi-agent cooperation: Model and experiments. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2001.