Bayesian learning for effective coordination in uncertain multi-agent systems

Mair Allen-Williams

Overview

- Uncertain, dynamic, multi-agent systems
- How to behave?
- Need to find out about the system while solving whatever the problem is
- ... and do both in a coordinated way

Uncertain systems: learning

- Handling uncertainty: learning
- Act, receive new state and reward
- Adjust beliefs about world
- (Markov assumption)

Definitions

- s: state, a: action, r: reward (single shared reward: co-operative systems)
- γ : “myopia” (how far into the future do we care?)
- $V(s)$: “value” of s over time
- $Q(s, a)$: “value” of s, if we take action a
- π : “policy”, for every s, a, $\pi(a, s) = P(a|s)$

So:

(1)
$$Q(s,a) = \sum_{s'} P(s'|s,a)V(s')$$

And

(2)
$$V^{\pi}(s) = \sum_{a} \pi(a,s)Q(s,a)$$

Bellman

When the world models (transition, reward) are completely known:

$$V_\pi(s) = \sum_a \pi(s,a) \sum_{s'} P(s'|s,a)\{E[r_{t+1}] + \gamma V_\pi(s')\}$$

$$V^*(s) = \max_\pi V_\pi(s) \text{ for all } s \in S$$

- "Bellman" equations
- π is the policy
- π^* is an optimal policy

- But the world models aren't usually known
- Have to use estimates
- Typically update: $Est \leftarrow Est + \alpha * Obs$ ($\alpha = 0.2, 0.1, 0.01...$)
- Either estimate $Q(s, a)$ directly ("model-free" learning)
- Or estimate $P(s'|s, a)$ and $P(r|s, a)$ and solve Bellman equations ("model-based" learning)

Model-based vs Model-free

Model-free:

- Straightforward
- No bias

Model-based:

- Bias may be what you want
- Re-usable
- Permit simulation alongside real-world steps

Bayesian learning

(Dearden et. al)

- Model based.
- Point estimates don't take uncertainty into account

Bayes' Rule:

$$P(Model|obs) \propto P(obs|Model)P(Model) \quad (3)$$

So instead of point estimates of a model (transitions, rewards), maintain probabilities over all models (parameterised).

- States are probability distributions over models
- Called belief states
- $$E[Q(s,a)] = \int_M Q(s,a|M)P(M) \quad (4)$$
- Act, update belief state, compute E[Q(s,a)] given current belief state, take optimal action ...

Multi-agent ... ?

- All very nice, but only for a single agent
- Extend into multi-player domain
- Agent’s action should be a “best response” to what it expects other agents to do
- Can continue to use single-agent methods
- Or can explicitly model the other agents

Multi-agent Bayesian

(Chalkiadakis...)

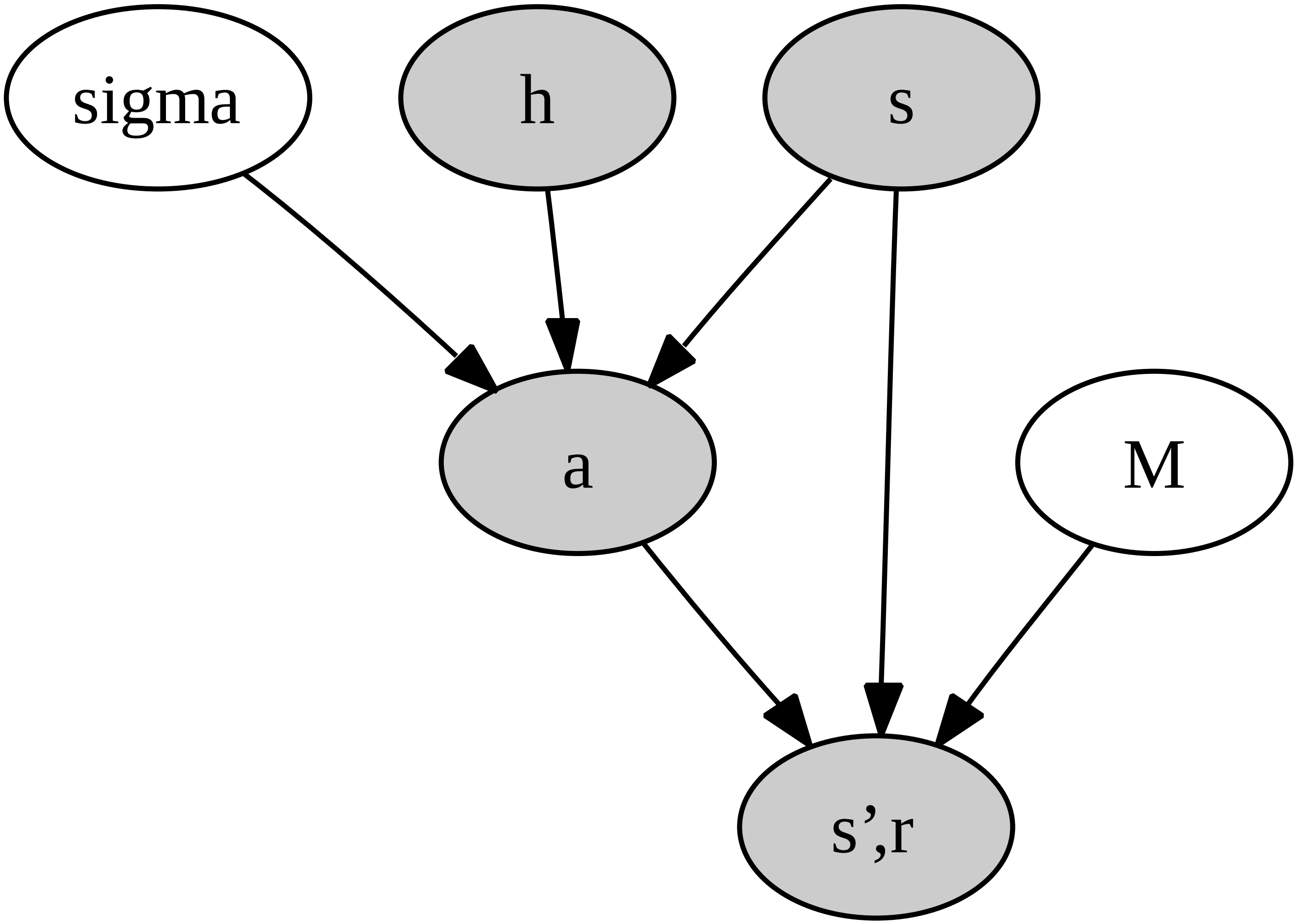
- As well as models of transition, reward functions, maintain models of the other agents
- Belief state: $\{\sigma, M, s, h\}$

- Action (best response) :

$$
\begin{aligned}
Q(a_i, b) \;=\; & \sum_{\mathbf{a_{-i}}} P(\mathbf{a_{-i}}|b) \sum_{s'} P(s'|a_i \circ \mathbf{a_{-i}}, b) \\
& \sum_{r} P(r|s', a_i \circ \mathbf{a_{-i}}, b) \\
& [r + \gamma V(b < s, \mathbf{a}, r, s' >)]
\end{aligned}
$$

- Updates using Bayes' rule

Network diagram



Updates

$$P(M|obs) \propto P(s', r|\mathbf{a}, s, M)P(M) \tag{5}$$

$$P(\sigma_j|obs) \propto P(\mathbf{a_j}|h, s, \sigma_j)P(\sigma_j) \tag{6}$$

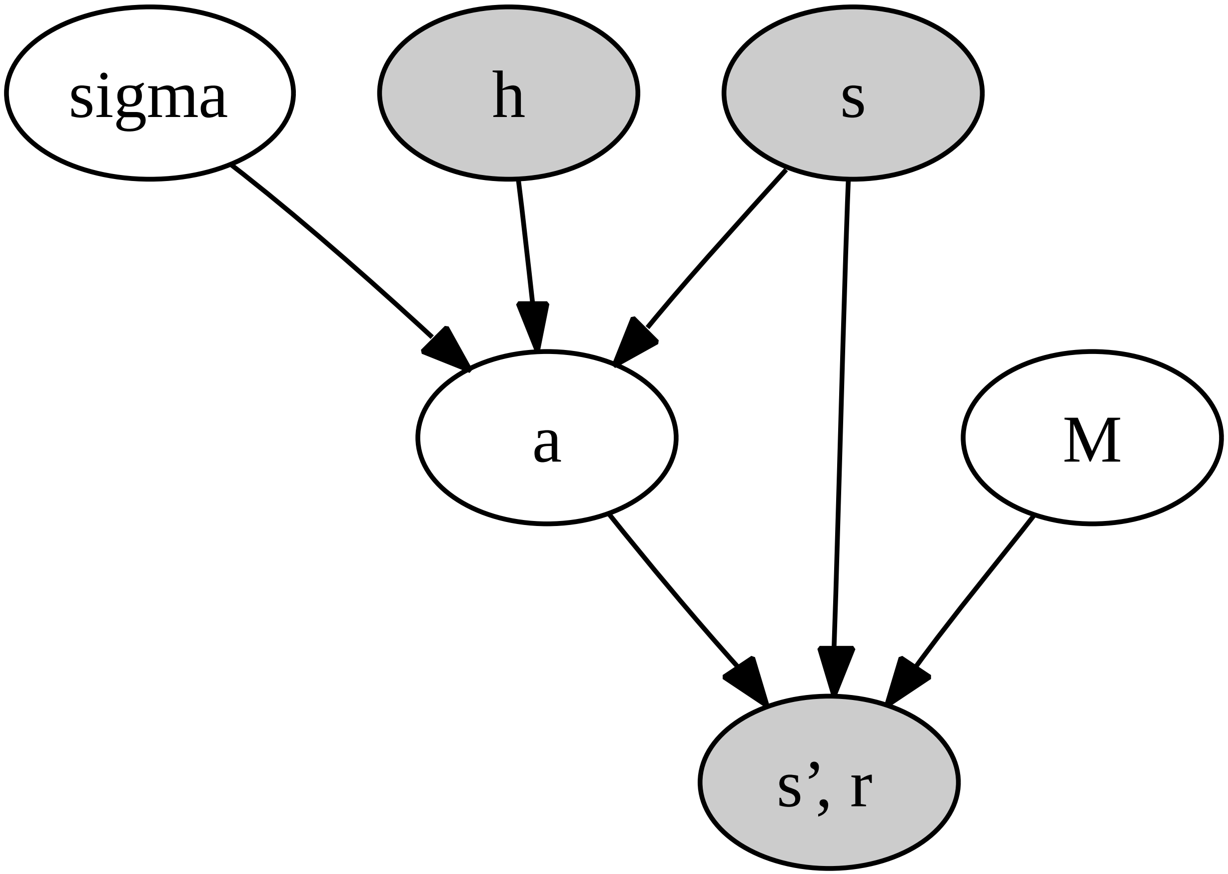
Partial observability

- e.g. Robocup
- May have local information which contributes to the state
- May not be able to see what everyone else is doing

Partially observable actions

- Can’t observe other agents’ actions
- But can perhaps see some, or guess something about them from the state (e.g. state is described by several variables)
- Still use best response
- But now, Bayesian updates are more complex

Partially observable actions: network



Partially observable actions: update

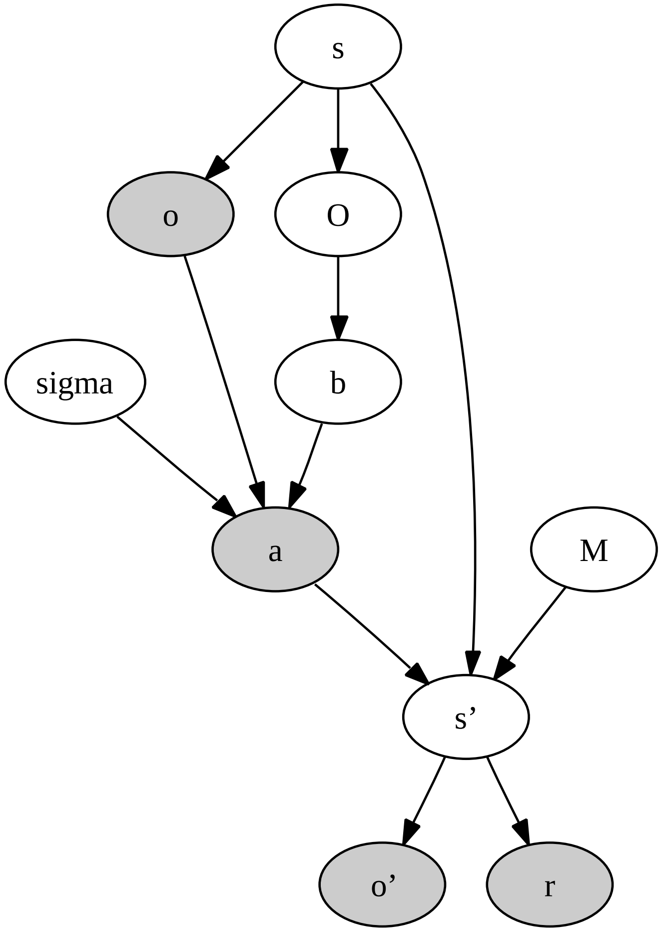
$$P(M|obs) \;\; \propto \;\; P(M)\sum_{\mathbf{a}} P(s', r|M, \mathbf{a}, s) \int_{\sigma} P(\mathbf{a}|\sigma, h, s) P(\sigma)$$

$$P(\sigma|obs) \;\; \propto \;\; P(\sigma)\sum_{\mathbf{a}} P(\mathbf{a}|\sigma, h, s) \int_{M} P(s', r|M, \mathbf{a}, s) P(M)$$

Partially observable states

- Local observations derived from the state
- Single-agent case: POMDP
- Multi-agent case: have to maintain beliefs about the other agents' belief states.

Partially observable states: network



Partially observable states: updates

- Horrible!
- Also have to modify best response to sum over possible belief states

Implementing?

- Theory is all very well, but is it implementable?
- Finite state/action space: multinomials with Dirichlet priors
- Sampling for continuous integrals
- Myopic best response
- (poa) Maximum likelihood ...
- But still very slow
- And haven't even /tried/ the pos ...

So: Approximations

- No details yet
- Sparse Dirichlet priors (Dearden)
- One-step game (Emery-Montemerlo et.al)
- PCA (Roy and Gordon)
- Hierarchies
- Hoey's thing (exploiting variable independences)

Next

- Spec for approximations
- Combining pos and poa
- ... open systems
- ... individual rewards
- ... ?