Optimistic optimization algorithms for planning in MDPs: a general algorithm

Thomas Bonis András György Csaba Szepesvári

École Centrale Paris, France University of Alberta, Edmonton, Canada

8th Barbados Workshop on Reinforcement Learning

General framework

Search process

- Process is governed by a set of search states Σ and a next state function $N : \Sigma \to 2^{\Sigma}$.
- Start from search state $\sigma_0 \in \Sigma_f$. For $t = 1, 2, \ldots$ repeat
  - Observe $b(\sigma)$ for all $\sigma \in N_f(\sigma_{t-1})$.
  - Select $\sigma_t \in N_f(\sigma_{t-1})$.
- Return (optionally) σ_t, x_{σ_t}, $f(x_{\sigma_t})$, $\mathcal{F}_{\sigma_t}$.

General framework

Search process

- Process is governed by a set of search states Σ and a next state function $N : \Sigma \to 2^{\Sigma}$.
- Start from search state $\sigma_0 \in \Sigma_f$. For $t = 1, 2, \ldots$ repeat
  - Observe $b(\sigma)$ for all $\sigma \in N_f(\sigma_{t-1})$.
  - Select $\sigma_t \in N_f(\sigma_{t-1})$.
- Return (optionally) σ_t, x_{σ_t}, $f(x_{\sigma_t})$, $\mathcal{F}_{\sigma_t}$.

Cost: Number of search states visited.

General framework

Goal: maximize a function $f : \mathcal{X} \to \mathbb{R}$

- $\mathcal{X}$ is an arbitrary set.
- f belongs to a known class of functions $\mathcal{F} \subset \{g : \mathcal{X} \to \mathbb{R}\}$.

General framework

Goal: maximize a function $f : \mathcal{X} \to \mathbb{R}$

- $\mathcal{X}$ is an arbitrary set.
- f belongs to a known class of functions $\mathcal{F} \subset \{g : \mathcal{X} \to \mathbb{R}\}$.

Search structure

- Set of possible search states Σ, and $\Sigma_f \subset \Sigma$ for a specific f.
- Set of functions that can lead to σ: $\mathcal{F}_\sigma = \{f \in \mathcal{F} : \sigma \in \Sigma_f\}$.
- For all $f \in \mathcal{F}$ and any $\sigma \in \Sigma_f$ there is an estimate $x_\sigma \in \mathcal{X}$ of the maximum of f.
- Neighborhood function: $N_f : \Sigma_f \to 2^{\Sigma_f}$.
- Best estimate available from σ is $f(\sigma)$.
- Optimistic bound: $b : \Sigma \to \mathbb{R}$ such that $b(\sigma) \geq f(\sigma)$ for all f such that $\sigma \in \Sigma_f$.

General optimistic search algorithm

GOS Algorithm

Initialization: $t = 0, \sigma_0 = O$.

For $t = 1, 2, \ldots$

- $\sigma_t = \arg\min_{\sigma \in N_f(\sigma_{t-1})} b(\sigma)$, ties are broken according to some predetermined rule B;
- return x_{σ_t}.

General optimistic search algorithm

GOS Algorithm

Initialization: $t = 0, \sigma_0 = O$.

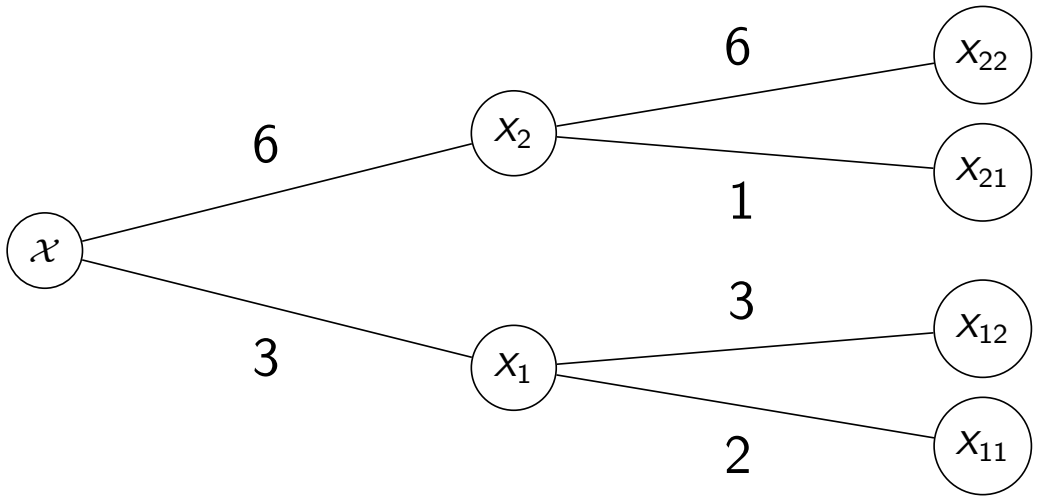
For $t = 1, 2, \ldots$

- $\sigma_t = \arg\min_{\sigma \in N_f(\sigma_{t-1})} b(\sigma)$, ties are broken according to some predetermined rule B;
- return x_{σ_t}.

Why is it optimistic?

Example: branch and bound

- Assume a hierarchical partitioning $\mathcal{P}$ of $\mathcal{X}$ is available, and for any $X \in \mathcal{P}$ the partition of X is $P(X)$.
- $\Sigma = \{\text{partitions } \{X_1, \ldots, X_k\} \in \mathcal{P} \text{ with observations}\}$.
- $N_f(\sigma) = \{\sigma' \in \Sigma : \sigma' = \sigma \setminus X \cup P(X), X \in \sigma\}$.
- $b_f(X) \geq \sup_{x \in X} f(x)$.
- $b(\sigma) = \sup_{f \in \mathcal{F}_\sigma} \max_{X \in \sigma} b_f(X)$.
- minimizing $b(\sigma')$ for $\sigma' \in N_f(\sigma) \Leftrightarrow$ splitting $X \in \sigma$ with maximal $b(X)$.



Assumptions

For every $f \in \mathcal{F}$ and $\sigma \in \Sigma_f$

(a) Achievability of the optimum: $f(\sigma) = \sup_{x \in \mathcal{X}} f(x)$.

(b) Optimistic heuristic function: $b(\sigma) \geq f(\sigma)$.

(c) Existence of minimizers: there exists a $\sigma' \in N_f(\sigma)$ such that $b(\sigma') = \inf_{\sigma'' \in N_f(\sigma)} b(\sigma'')$.

Assumptions

For every $f \in \mathcal{F}$ and $\sigma \in \Sigma_f$

(a) Achievability of the optimum: $f(\sigma) = \sup_{x \in \mathcal{X}} f(x)$.

(b) Optimistic heuristic function: $b(\sigma) \geq f(\sigma)$.

(c) Existence of minimizers: there exists a $\sigma' \in N_f(\sigma)$ such that $b(\sigma') = \inf_{\sigma'' \in N_f(\sigma)} b(\sigma'')$.

Consequences:

- GOS algorithm is well-defined.
- $b(\sigma) \geq \sup_{f \in \mathcal{F}_\sigma} \sup_{x \in \mathcal{X}} f(x)$.

Optimality

Assumption

- $b(\sigma) = \sup_{f \in \mathcal{F}_\sigma} \sup_{x \in \mathcal{X}} f(x)$.

Optimality

Assumption

- $b(\sigma) = \sup_{f \in \mathcal{F}_\sigma} \sup_{x \in \mathcal{X}} f(x)$.

Theorem: optimality of GOS

There exists no search algorithm A that, for all $f \in \mathcal{F}$, explores less search states than GOS with optimal tie-breaking while achieving the same accuracy.

General search: do we need to be optimistic?

GS Algorithm

Initialization: $t = 0, \sigma_0 = O$.

For $t = 1, 2, \ldots$

- Pick $\sigma_t \in N_f(\sigma_{t-1})$ with $b(\sigma_t) \leq b(\sigma_{t-1})$.
- Return x_{σ_t}.

Accuracy

Assumptions: No loops: $\sigma \notin N_f^*(\sigma)$.
Improving search process: $f(x_\sigma) \leq f(x_{\sigma'})$ for any $\sigma' \in N_f(\sigma)$.

Accuracy

Assumptions: No loops: $\sigma \notin N_f^*(\sigma)$.
Improving search process: $f(x_\sigma) \leq f(x_{\sigma'})$ for any $\sigma' \in N_f(\sigma)$.

Notation: Set of states with high uncertainty:

$$\Sigma(f, \varepsilon) = \{\sigma \in \Sigma : b(\sigma) - f(x_\sigma) \geq \varepsilon\}.$$

Set of search states visited: $E_{\Sigma,f} = \cup_{n \geq 0}\{\sigma_n\}$.

Accuracy

Assumptions: No loops: $\sigma \notin N_f^*(\sigma)$.
Improving search process: $f(x_\sigma) \leq f(x_{\sigma'})$ for any $\sigma' \in N_f(\sigma)$.

Notation: Set of states with high uncertainty:

$$\Sigma(f, \varepsilon) = \{\sigma \in \Sigma : b(\sigma) - f(x_\sigma) \geq \varepsilon\}.$$

Set of search states visited: $E_{\Sigma,f} = \cup_{n \geq 0}\{\sigma_n\}$.

Theorem: accuracy of GS

Let $\varepsilon_f(n) = \sup\{\varepsilon \geq 0 : |E_{\Sigma,f} \cap \Sigma(f, \varepsilon)| \geq n\}$. Then

$$f(x_{\sigma_n}) \geq \sup_{x \in \mathcal{X}} f(x) - \varepsilon_f(n).$$

Accuracy

Assumptions: No loops: $\sigma \notin N_f^*(\sigma)$.
Improving search process: $f(x_\sigma) \leq f(x_{\sigma'})$ for any $\sigma' \in N_f(\sigma)$.

Notation: Set of states with high uncertainty:

$$\Sigma(f, \varepsilon) = \{\sigma \in \Sigma : b(\sigma) - f(x_\sigma) \geq \varepsilon\}.$$

Set of search states visited: $E_{\Sigma,f} = \cup_{n \geq 0}\{\sigma_n\}$.

Theorem: accuracy of GS

Let $\varepsilon_f(n) = \sup\{\varepsilon \geq 0 : |E_{\Sigma,f} \cap \Sigma(f, \varepsilon)| \geq n\}$. Then

$$f(x_{\sigma_n}) \geq \sup_{x \in \mathcal{X}} f(x) - \varepsilon_f(n).$$

Proof idea: To achieve ε-optimality, search-states with uncertainty larger than ε should either be visited or excluded.

Using the structure of the space: packing numbers

Assumptions:

- Σ has a δ-refining partitioning (Σ_h): $\delta(h) \searrow 0$
  $f(x_\sigma) + \delta(h) \geq \sup_{x \in \mathcal{X}} f(x)$ if $\sigma \in \Sigma_h$.
- Representative points in search states: $\hat{x}_\sigma$ – the points actually sampled.
- Distance $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$: sampled points at the same level visited by GS are far: $d(\hat{x}_\sigma, \hat{x}_{\sigma'}) \geq \delta(h)$ if $\sigma, \sigma' \in E_{\Sigma,f} \cap \Sigma_h$.
- ε-optimal points $\mathcal{X}_\varepsilon = \{x \in \mathcal{X} : f(x) \geq f^*(x) - \varepsilon\}$.
  Their ε-packing number is

$$g(\varepsilon) = \sup \left\{ |\mathcal{Y} \cap \mathcal{X}_\varepsilon| \ : \ \mathcal{Y} \subseteq \mathcal{X} \text{ s.t. } \min_{x \neq x' \in \mathcal{Y}} d(x, x') \geq \varepsilon \right\}.$$

Using the structure of the space: packing numbers

Assumptions:

- Σ has a δ-refining partitioning (Σ_h): $\delta(h) \searrow 0$
  $f(x_\sigma) + \delta(h) \geq \sup_{x \in \mathcal{X}} f(x)$ if $\sigma \in \Sigma_h$.
- Representative points in search states: $\hat{x}_\sigma$ – the points actually sampled.
- Distance $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$: sampled points at the same level visited by GS are far: $d(\hat{x}_\sigma, \hat{x}_{\sigma'}) \geq \delta(h)$ if $\sigma, \sigma' \in E_{\Sigma,f} \cap \Sigma_h$.
- ε-optimal points $\mathcal{X}_\varepsilon = \{x \in \mathcal{X} : f(x) \geq f^*(x) - \varepsilon\}$.
  Their ε-packing number is

$$g(\varepsilon) = \sup \left\{ |\mathcal{Y} \cap \mathcal{X}_\varepsilon| \ : \ \mathcal{Y} \subseteq \mathcal{X} \text{ s.t. } \min_{x \neq x' \in \mathcal{Y}} d(x, x') \geq \varepsilon \right\}.$$

Consequence: $|E_{\Sigma,f} \cap \Sigma_h| \leq g(\delta(h))$.

Using the structure of the space: packing numbers

Corollary

Let $h_g(n) = \min\{h \geq 0 : \sum_{l=0}^{h} g(\delta(l)) \geq n\}$. Then, under the previous assumptions,

$$f(x_{\sigma_n}) \geq \sup_{x \in \mathcal{X}} f(x) - \delta(h_g(n)).$$

Finally: back to MDPs!

- Earlier notation:
  - $\mathcal{P}$: set of transition probabilities (given, S, A).
  - $\mathcal{R}$: set of reward functions (given, S, A).
  - $M_K^+(P_{|K}, r_{|K}), M_K^-(P_{|K}, r_{|K})$: optimistic and pessimistic MDPs when the states $K \subset S$ have been expanded.
- The function class f:
  - $\mathcal{X} = 2^S$.
  - $f_{P,r}$ is parametrized by $P \in \mathcal{P}$, $r \in \mathcal{R}$.
  - $f_{P,r}(K) = V^*(M_K^-(P, r))$.
- Search states:
  - $\Sigma = \{(K, P_{|K}, r_{|K}) : K \in 2^S, P \in \mathcal{P}, r \in \mathcal{R}\}$.
  - $\Sigma_{f_{P,r}} = \{(K, P_{|K}, r_{|K}) : K \in 2^S\}$.

Finally: back to MDPs!

- Next-state function:

$$N_{f_{P,r}}((K, P_{|K}, r_{|K}))$$
$$= \{(K', P_{|K'}, r_{|K'}) : \ K' \text{ is obtained from } K \text{ by expanded a fringe state}\}$$

- What we see:
  - Search states: $\sigma = (K, P_{|K}, r_{|K}) \in \Sigma$
  - $f(K) = V^*(M_K^-(P_{|K}, r_{|K}))$ for all $f \in \mathcal{F}_\sigma$.

- Bound: $b((K, P_{|K}, r_{|K})) = V^*(M_K^+(P_{|K}, r_{|K}))$.

Consequences for MDPs

Optimality of the optimistic planning algorithm

Consequences for MDPs

Optimality of the optimistic planning algorithm

Consistency of LAO*-type algorithms

- if $b(\sigma_t) - V^*(M) \to 0$.
- Achievable, e.g., if always a state closest to the start state is expanded.

Consequences for MDPs

Optimality of the optimistic planning algorithm

Consistency of LAO*-type algorithms

- if $b(\sigma_t) - V^*(M) \to 0$.
- Achievable, e.g., if always a state closest to the start state is expanded.

Open questions:

- New node expansion rules
- Monte Carlo algorithms
- Packing number bounds
- ...