Scale-free adaptive PLANNING for deterministic dynamics & discounted rewards

Peter Bartlett, Victor Gabillon, Jennifer Healey, Michal Valko

ICML - June 13th, 2019

An MCTS setting

MDP with **starting state** $x_0 \in X$, action space A

n **interactions:** At time t playing a_t in x_t leads to

Deterministic dynamics g**:** $x_{t+1} \triangleq g(x_t, a_t)$,

Reward: $r_t(x_t, a_t) + \varepsilon_t$ with ε_t being the noise

Objective: Recommend action $a(n)$ that minimizes

$$r_n \triangleq \max_{a \in A} Q^\star(x, a) - Q^\star(x, a(n)) \quad \textbf{simple regret}$$

where $Q^\star(x, a) \triangleq r(x, a) + \sup_\pi \sum \gamma^t r(x_t, \pi(x_t))$

Assumption: $r_t \in [0, R_{\max}]$ and $|\varepsilon_t| \leq b$

Approach: Trying to explore without the parameters $R_{\max}$ and b

An MCTS setting

MDP with **starting state** $x_0 \in X$, action space A

n **interactions:** At time t playing a_t in x_t leads to

Deterministic dynamics g**:** $x_{t+1} \triangleq g(x_t, a_t)$,

Reward: $r_t(x_t, a_t) + \varepsilon_t$ with ε_t being the noise

Objective: Recommend action $a(n)$ that minimizes

$$r_n \triangleq \max_{a \in A} Q^\star(x, a) - Q^\star(x, a(n)) \quad \textbf{simple regret}$$

where $Q^\star(x, a) \triangleq r(x, a) + \sup_\pi \sum \gamma^t r(x_t, \pi(x_t))$

Assumption: $r_t \in [0, R_{\max}]$ and $|\varepsilon_t| \leq b$

Approach: Trying to explore without the parameters $R_{\max}$ and b

An MCTS setting

MDP with **starting state** $x_0 \in X$, action space A

n **interactions:** At time t playing a_t in x_t leads to

Deterministic dynamics g**:** $x_{t+1} \triangleq g(x_t, a_t)$,

Reward: $r_t(x_t, a_t) + \varepsilon_t$ with ε_t being the noise

Objective: Recommend action $a(n)$ that minimizes

$$r_n \triangleq \max_{a \in A} Q^\star(x, a) - Q^\star(x, a(n)) \quad \textbf{simple regret}$$

where $Q^\star(x, a) \triangleq r(x, a) + \sup_\pi \sum \gamma^t r(x_t, \pi(x_t))$

Assumption: $r_t \in [0, R_{\max}]$ and $|\varepsilon_t| \leq b$

Approach: Trying to explore without the parameters $R_{\max}$ and b

OLOP (Bubeck and Munos, 2010)

OLOP implements Optimistic Planning using Upper Confidence Bound (UCB) on the Q value of a sequence of q actions $a_1, \ldots, a_q$:

$$\widehat{Q}_t^{UCB}(a_{1:q}) \triangleq \underbrace{\sum_{h=1}^{q}\left(\gamma^h \widehat{r}_h(t) + \gamma^h b \sqrt{\frac{1}{T_{a_h}(t)}}\right)}_{\text{estimation of observed reward}} + \underbrace{\frac{R_{\mathbf{max}}\gamma^{q+1}}{1-\gamma}}_{\text{unseen reward}}$$

in optimization under a fixed budget n, **excellent strategies** allocate samples to actions **without knowing** $R_{\max}$ or b

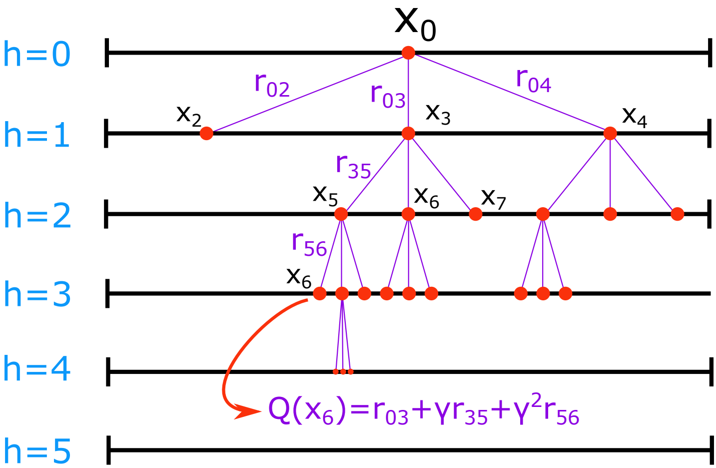
OLOP (Bubeck and Munos, 2010)

OLOP implements Optimistic Planning using Upper Confidence Bound (UCB) on the Q value of a sequence of q actions $a_1, \ldots, a_q$:

$$\widehat{Q}_t^{UCB}(a_{1:q}) \triangleq \underbrace{\sum_{h=1}^{q}\left(\gamma^h \widehat{r}_h(t) + \gamma^h b \sqrt{\frac{1}{T_{a_h}(t)}}\right)}_{\text{estimation of observed reward}} + \underbrace{\frac{R_{\max}\gamma^{q+1}}{1-\gamma}}_{\text{unseen reward}}$$

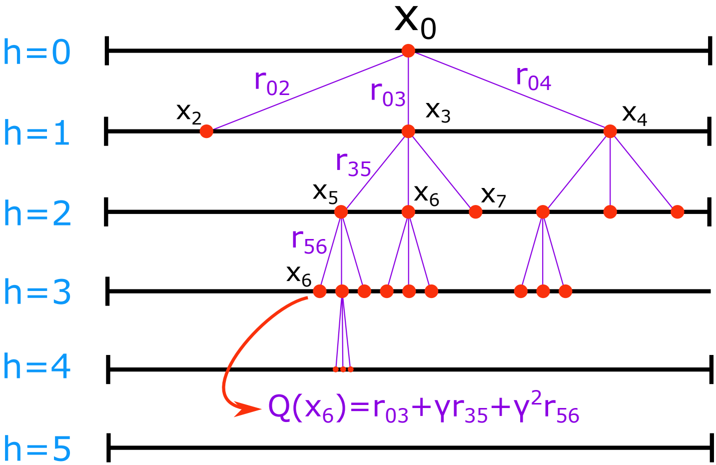
in optimization under a fixed budget n, **excellent strategies** allocate samples to actions **without knowing** $R_{\max}$ or b

Tree Search



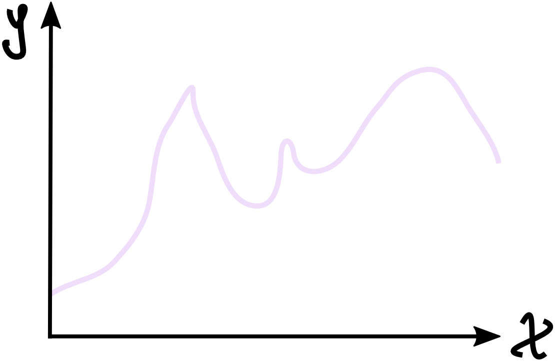
h=0
h=1
h=2
h=3
h=4
h=5
x_0
r_{02}
r_{03}
r_{04}
x_2
x_3
x_4
r_{35}
x_5
x_6
x_7
r_{56}
x_6
$Q(x_6)=r_{03}+\gamma r_{35}+\gamma^2 r_{56}$

Tree Search

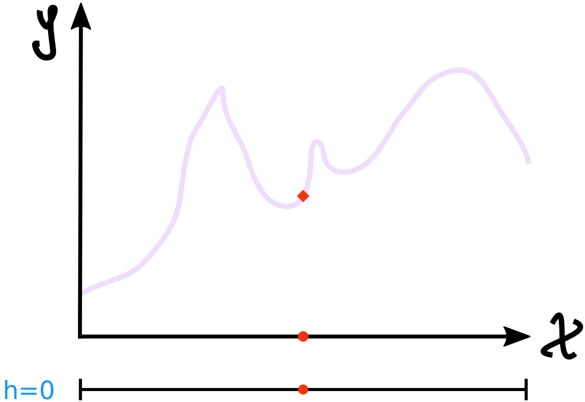


This is a zero order optimization!

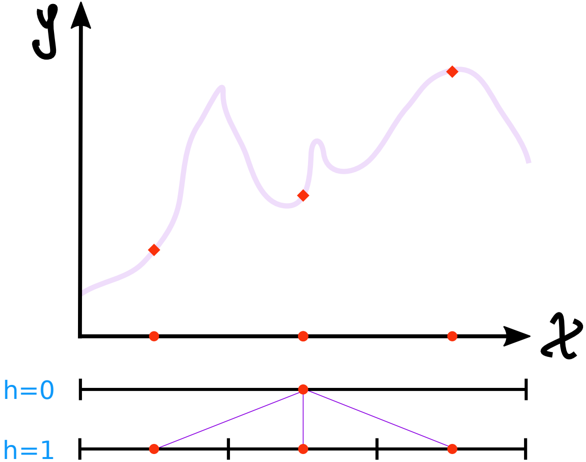
Black-box optimization: use the partitioning to explore f (uniformly)



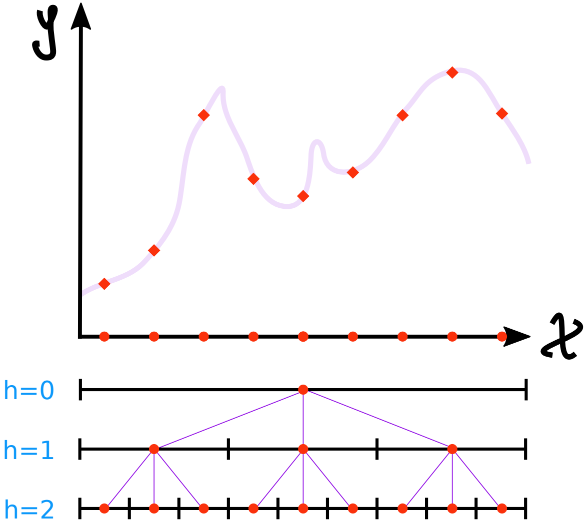
Black-box optimization: use the partitioning to explore f (uniformly)



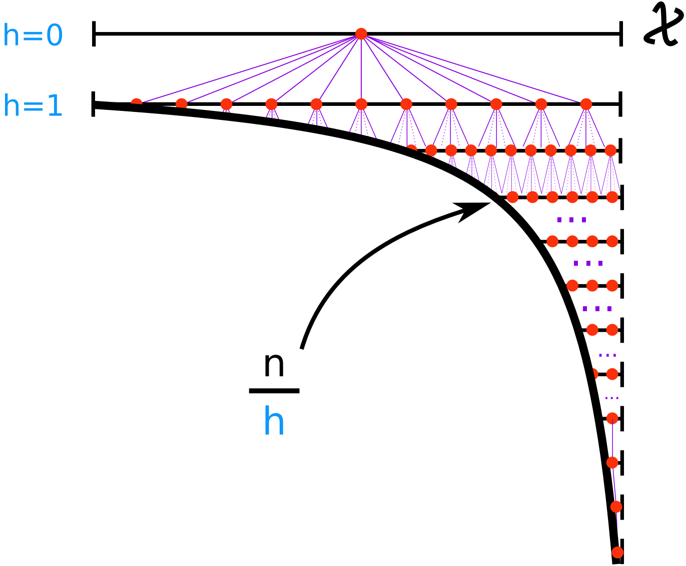
Black-box optimization: use the partitioning to explore f (uniformly)



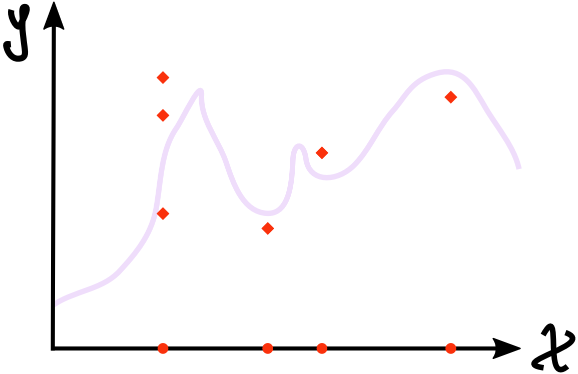
Black-box optimization: use the partitioning to explore f (uniformly)



Zipf exploration: Open best $\frac{n}{h}$ cells at depth h



Noisy case

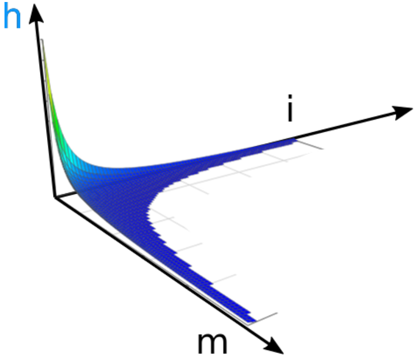


- need to pull more each x to limit uncertainty
- **tradeoff:** the more you pull each x the shallower you can explore

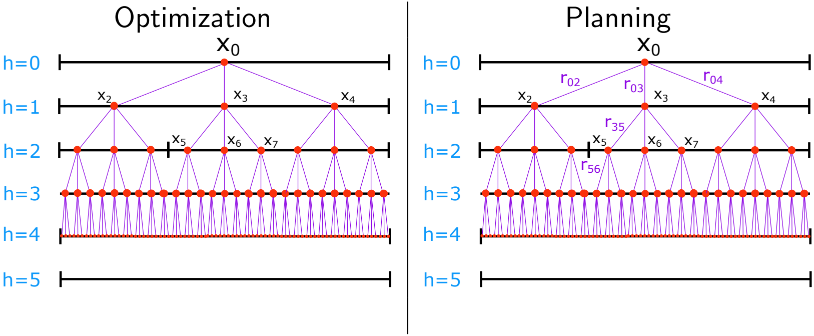
Noisy case: `StroquOOL` (Bartlett et al. 2019)

At depth h:

- order the cells by decreasing value *and*
- open the i-th best cell with $m = \frac{n}{hi}$ estimations

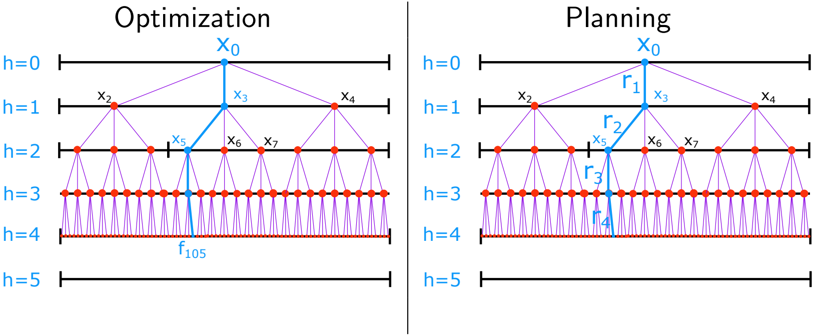


Black-box optimization vs planning:
Reuse of samples and γ



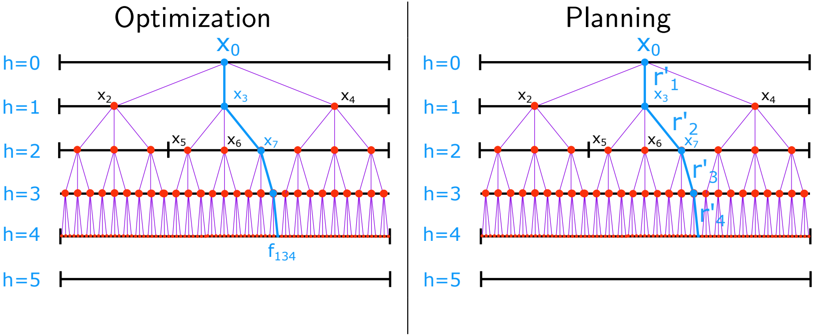
Lower regret for planning! (Bubeck & Munos 2010)

Black-box optimization vs planning:
Reuse of samples and γ



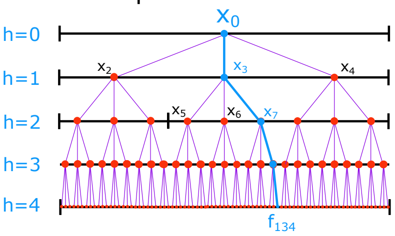
Lower regret for planning! (Bubeck & Munos 2010)

Black-box optimization vs planning:
Reuse of samples and γ

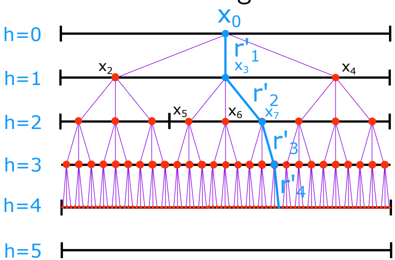


Lower regret for planning! (Bubeck & Munos 2010)

Black-box optimization vs planning:
Reuse of samples and γ



K^H samples near the root



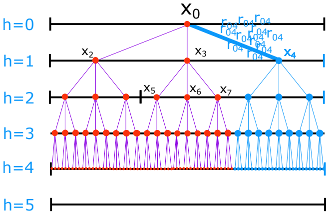
How many samples near the root?

Lower regret for planning! (Bubeck & Munos 2010)

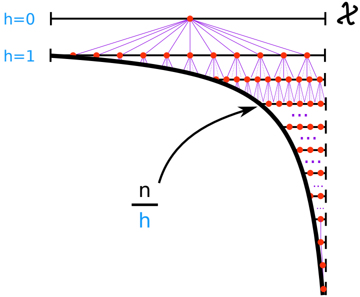
Black-box optimization vs. planning:

Reuse samples and take advantage of γ

Uniform exploration



Zipf exploration



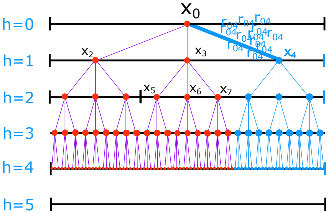
Bubeck & Munos: Only for uniform strategies ...

We figured the amount the samples needed!

Black-box optimization vs. planning:

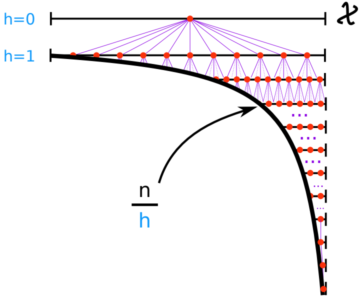
Reuse samples and take advantage of γ

Uniform exploration



not sharing information

Zipf exploration



Sharing information

Bubeck & Munos: Only for uniform strategies . . .

We figured the amount the samples needed!

The power of PlaTγPOOS

- implements **Zipf** exploration for MCTS StroquOOL,
- explicitly pulls an action at depth $h+1$, γ times less than action at depth h, ($Q^\star(x,a) = r(x,a) + \sup_\pi \sum \gamma^t r(x_t, \pi(x_t))$),
- does not use UCB & no use of $R_{\max}$ and b,)
- improves over OLOP with **adaptation to low noise** and **additional unknown smoothness**
- gets exponential speedups when no noise is present!