

Planning and Optimization

F9. Monte-Carlo Tree Search Algorithms (Part II)

Malte Helmert and Gabriele Röger

Universität Basel

ε -greedy
●○○○○

Softmax
○○○○○

UCB1
○○○○○○○○○○

Summary
○○

ε -greedy

ε -greedy: Idea

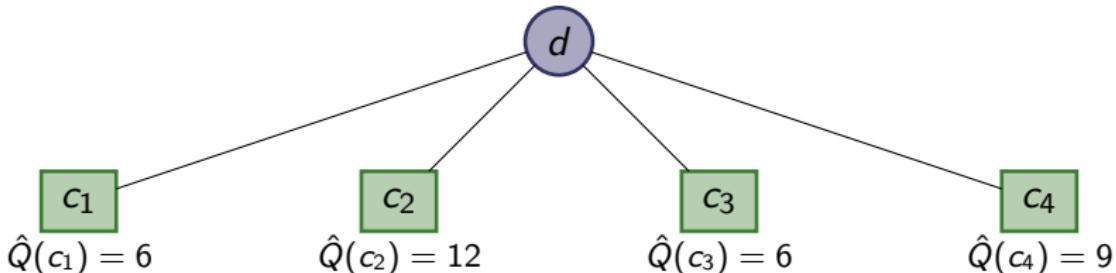
- tree policy parametrized with constant parameter ε
- with probability $1 - \varepsilon$, pick one of the **greedy** actions uniformly at random
- otherwise, pick a non-greedy successor **uniformly at random**

ε -greedy Tree Policy

$$\pi(a | d) = \begin{cases} \frac{1-\varepsilon}{|A_*^k(d)|} & \text{if } a \in A_*^k(d) \\ \frac{\varepsilon}{|A(d(s)) \setminus A_*^k(d)|} & \text{otherwise,} \end{cases}$$

with $A_*^k(d) = \{a(c) \in A(s(d)) \mid c \in \arg \min_{c' \in \text{children}(d)} \hat{Q}^k(c')\}$.

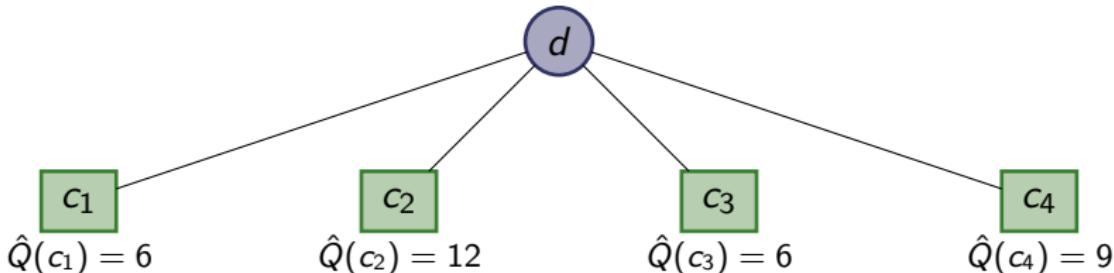
ε -greedy: Example



Assuming $a(c_i) = a_i$ and $\varepsilon = 0.2$, we get:

- $\pi(a_1 | d) =$
- $\pi(a_2 | d) =$
- $\pi(a_3 | d) =$
- $\pi(a_4 | d) =$

ε -greedy: Example



Assuming $a(c_i) = a_i$ and $\varepsilon = 0.2$, we get:

- $\pi(a_1 | d) = 0.4$
- $\pi(a_2 | d) = 0.1$
- $\pi(a_3 | d) = 0.4$
- $\pi(a_4 | d) = 0.1$

ε -greedy: Asymptotic Optimality

Asymptotic Optimality of ε -greedy

- explores forever
- not greedy in the limit

~~> not asymptotically optimal

ε -greedy: Asymptotic Optimality

Asymptotic Optimality of ε -greedy

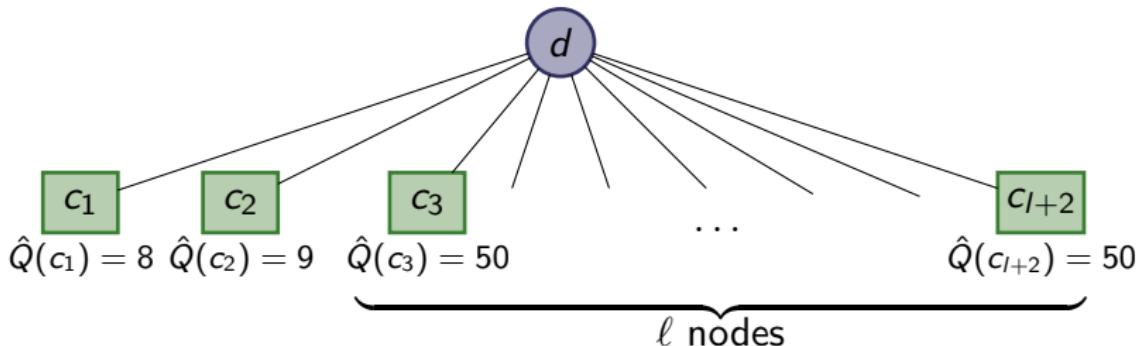
- explores forever
- not greedy in the limit
- ~~> not asymptotically optimal

asymptotically optimal variant uses **decaying ε** , e.g. $\varepsilon = \frac{1}{k}$

ε -greedy: Weakness

Problem:

when ε -greedy explores, all non-greedy actions are treated **equally**

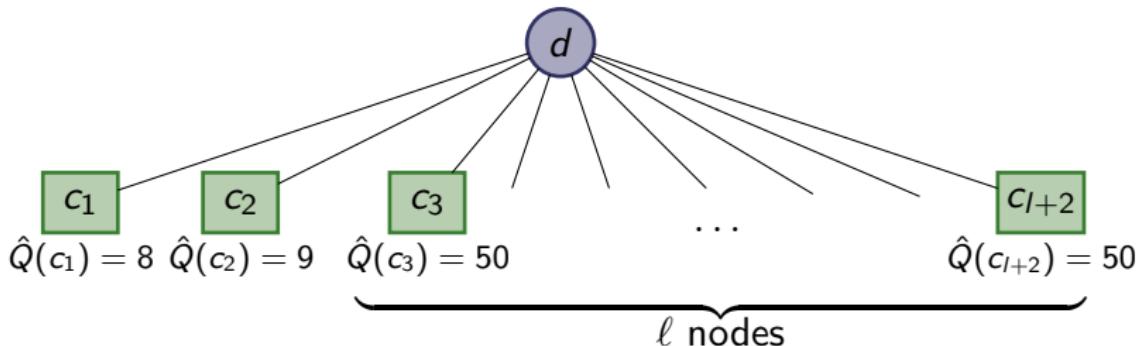


Assuming $a(c_i) = a_i$, $\varepsilon = 0.2$ and $\ell = 9$, we get:

ε -greedy: Weakness

Problem:

when ε -greedy explores, all non-greedy actions are treated **equally**



Assuming $a(c_i) = a_i$, $\varepsilon = 0.2$ and $\ell = 9$, we get:

- $\pi(a_1 | d) = 0.8$
- $\pi(a_2 | d) = \pi(a_3 | d) = \dots = \pi(a_{11} | d) = 0.02$

ε -greedy
ooooo

Softmax
●oooo

UCB1
oooooooo

Summary
oo

Softmax

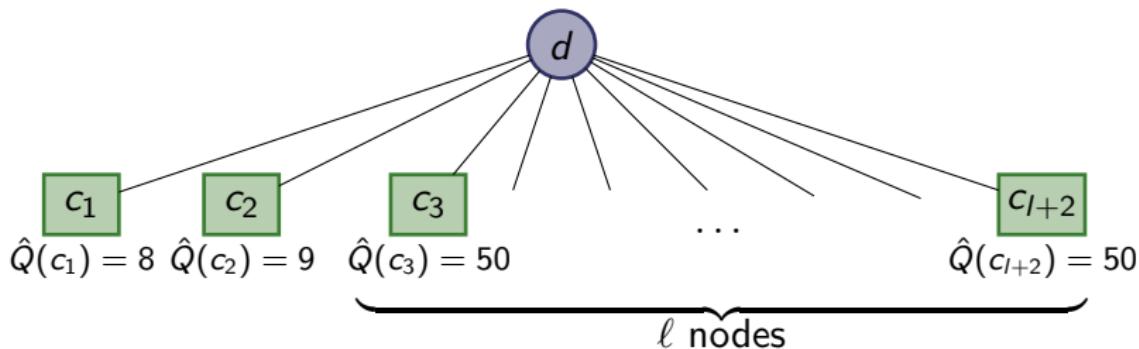
Softmax: Idea

- tree policy with constant parameter τ
- select actions **proportionally** to their action-value estimate
- most popular softmax tree policy uses **Boltzmann exploration**
- \Rightarrow selects actions proportionally to $e^{\frac{-\hat{Q}_k(c)}{\tau}}$

Tree Policy based on Boltzmann Exploration

$$\pi(a(c) | d) = \frac{e^{\frac{-\hat{Q}_k(c)}{\tau}}}{\sum_{c' \in \text{children}(d)} e^{\frac{-\hat{Q}_k(c')}{\tau}}}$$

Softmax: Example



Assuming $a(c_i) = a_i$, $\tau = 10$ and $\ell = 9$, we get:

- $\pi(a_1 | d) = 0.49$
- $\pi(a_2 | d) = 0.45$
- $\pi(a_3 | d) = \dots = \pi(a_{11} | d) = 0.007$

Boltzmann Exploration: Asymptotic Optimality

Asymptotic Optimality of Boltzmann Exploration

- explores forever
- not greedy in the limit:
 - state- and action-value estimates converge to finite values
 - therefore, probabilities also converge to positive, finite values

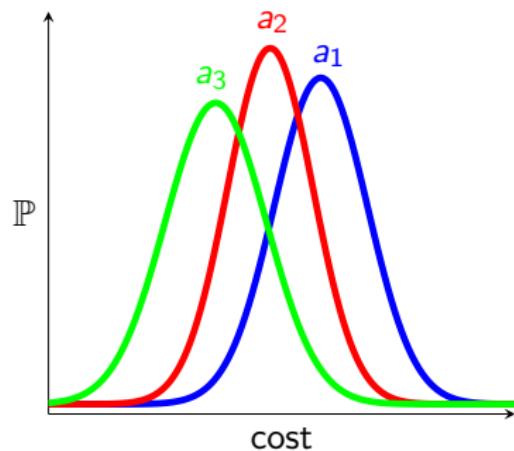
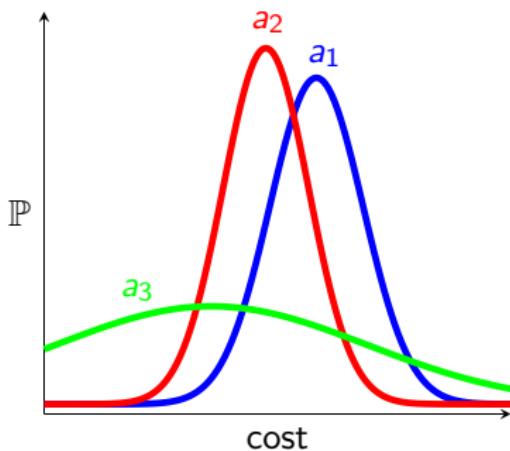
~~> not asymptotically optimal

asymptotically optimal variant uses **decaying τ** , e.g. $\tau = \frac{1}{\log k}$

careful: τ must not decay faster than logarithmically

(i.e., must have $\tau \geq \frac{\text{const}}{\log k}$) to explore infinitely

Boltzmann Exploration: Weakness



- Boltzmann exploration and ε -greedy only consider **mean** of sampled action-values
- as we sample the same node many times, we can also gather information about variance (how **reliable** the information is)
- Boltzmann exploration ignores the variance, treating the two scenarios equally

ε -greedy
ooooo

Softmax
ooooo

UCB1
●oooooooo

Summary
oo

UCB1

Upper Confidence Bounds: Idea

Balance **exploration** and **exploitation** by preferring actions that

- have been **successful in earlier iterations** (exploit)
- have been **selected rarely** (explore)

Upper Confidence Bounds: Idea

- select successor c of d that minimizes $\hat{Q}^k(c) - E^k(d) \cdot B^k(c)$
 - based on **action-value estimate** $\hat{Q}^k(c)$,
 - **exploration factor** $E^k(d)$ and
 - **bonus term** $B^k(c)$.
- select $B^k(c)$ such that
$$Q_*(s(c), a(c)) \geq \hat{Q}^k(c) - E^k(d) \cdot B^k(c)$$
with high probability
- **Idea:** $\hat{Q}^k(c) - E^k(d) \cdot B^k(c)$ is a **lower confidence bound** on $Q_*(s(c), a(c))$ under the collected information

Bonus Term of UCB1

- use $B^k(c) = \sqrt{\frac{2 \cdot \ln N^k(d)}{N^k(c)}}$ as bonus term
- bonus term is derived from **Chernoff-Hoeffding bound**:
 - gives the probability that a **sampled value** (here: $\hat{Q}^k(c)$)
 - is far from its **true expected value** (here: $Q_*(s(c), a(c))$)
 - in dependence of the **number of samples** (here: $N^k(c)$)
- picks the optimal action **exponentially** more often
- concrete MCTS algorithm that uses UCB1 is called **UCT**

Exploration Factor (1)

Exploration factor $E^k(d)$ serves **two roles** in SSPs:

- UCB1 designed for MAB with **reward in $[0, 1]$**
 $\Rightarrow \hat{Q}^k(c) \in [0; 1]$ for all k and c
- bonus term $B^k(c) = \sqrt{\frac{2 \cdot \ln N^k(d)}{N^k(c)}}$ always ≥ 0
- when d is visited,
 - $B^{k+1}(c) > B^k(c)$ if $a(c)$ is not selected
 - $B^{k+1}(c) < B^k(c)$ if $a(c)$ is selected
- if $B^k(c) \geq 2$ for some c , UCB1 **must explore**
- hence, $\hat{Q}^k(c)$ and $B^k(c)$ are always of **similar size**
 \Rightarrow set $E^k(d)$ to a value that **depends on $\hat{V}^k(d)$**

Exploration Factor (2)

Exploration factor $E^k(d)$ serves **two roles** in SSPs:

- $E^k(d)$ allows to adjust **balance** between exploration and exploitation
- search with $E^k(d) = \hat{V}^k(d)$ very greedy
- in practice, $E^k(d)$ is often **multiplied** with constant > 1
- UCB1 often requires **hand-tailored** $E^k(d)$ to work well

Asymptotic Optimality

Asymptotic Optimality of UCB1

- explores forever
- greedy in the limit

~~> asymptotically optimal

Asymptotic Optimality

Asymptotic Optimality of UCB1

- explores forever
- greedy in the limit
- ~~> asymptotically optimal

However:

- no theoretical justification to use UCB1 for SSPs/MDPs
(MAB proof requires stationary rewards)
- development of tree policies is an active research topic

ε -greedy
ooooo

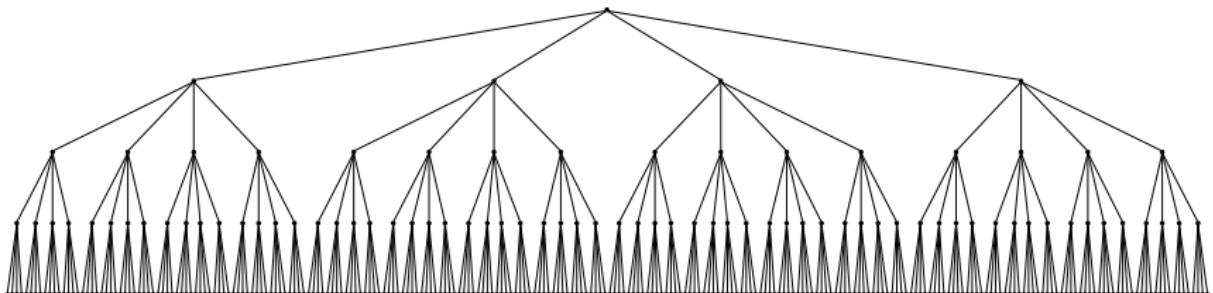
Softmax
ooooo

UCB1
oooooooo●○

Summary
oo

Symmetric Search Tree up to depth 4

full tree up to depth 4



ε -greedy
ooooo

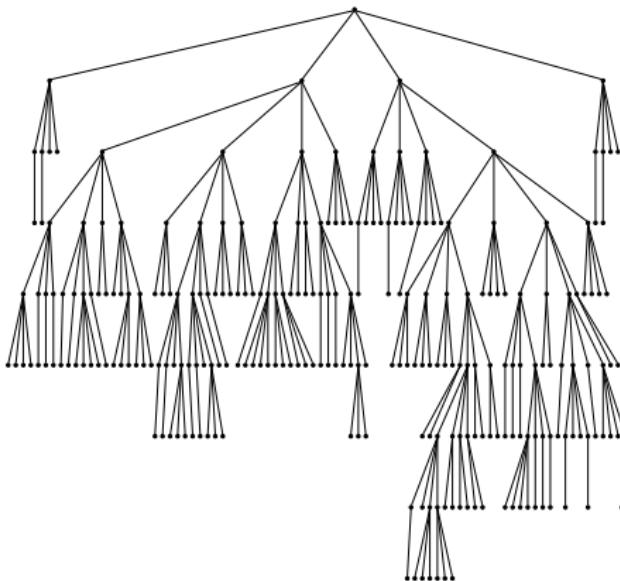
Softmax
ooooo

UCB1
oooooooo●

Summary
oo

Asymmetric Search Tree of UCB1

(equal number of search nodes)



ε -greedy
ooooo

Softmax
ooooo

UCB1
oooooooo

Summary
●○

Summary

Summary

- ε -greedy, Boltzmann exploration and UCB1 **balance exploration and exploitation**
- ε -greedy selects **greedy action** with probability $1 - \varepsilon$ and another action uniformly at random otherwise
- ε -greedy selects non-greedy actions with **same probability**
- Boltzmann exploration selects each action **proportional to its action-value estimate**
- Boltzmann exploration does not take **confidence of estimate** into account
- UCB1 selects actions greedily w.r.t. **upper confidence bound** on action-value estimate