MCTS Based on Simple Regret

David Tolpin, Solomon Eyal Shimony
Department of Computer Science,
Ben-Gurion University of the Negev, Beer Sheva, Israel
{tolpin,shimony}@cs.bgu.ac.il

Abstract

UCT, a state-of-the art algorithm for Monte Carlo tree search (MCTS) in games and Markov decision processes, is based on UCB, a sampling policy for the Multi-armed Bandit problem (MAB) that minimizes the cumulative regret. However, search differs from MAB in that in MCTS it is usually only the final "arm pull" (the actual move selection) that collects a reward, rather than all "arm pulls". Therefore, it makes more sense to minimize the simple regret, as opposed to the cumulative regret. We begin by introducing policies for multi-armed bandits with lower finite-time and asymptotic simple regret than UCB, using it to develop a two-stage scheme (SR+CR) for MCTS which outperforms UCT empirically.

Optimizing the sampling process is itself a metareasoning problem, a solution of which can use value of information (VOI) techniques. Although the theory of VOI for search exists, applying it to MCTS is non-trivial, as typical myopic assumptions fail. Lacking a complete working VOI theory for MCTS, we nevertheless propose a sampling scheme that is "aware" of VOI, achieving an algorithm that in empirical evaluation outperforms both UCT and the other proposed algorithms.

Introduction

Monte-Carlo tree search, and especially a version based on the UCT formula (Kocsis and Szepesvári 2006) appears in numerous search applications, such as (Gelly and Wang 2006; Eyerich, Keller, and Helmert 2010). Although these methods are shown to be successful empirically, most authors appear to be using the UCT formula "because it has been shown to be successful in the past", and "because it does a good job of trading off exploration and exploitation". While the latter statement may be correct for the multi-armed bandit and for the UCB method (Auer, Cesa-Bianchi, and Fischer 2002), we argue that it is inappropriate for search. The problem is not that UCT does not work; rather, a simple reconsideration from basic principles can result in schemes that outperform UCT.

The core issue is that in adversarial search and search in "games against nature" — optimizing behavior under uncertainty, the goal is typically to either find a good (or optimal) strategy, or even just to find the best first action of such a policy. Once such an action is discovered, it is usually not beneficial to further sample that action, "exploitation" is thus meaningless for search problems. Finding a good first action is closer to the pure exploration variant, as seen in the selection problem (Bubeck, Munos, and Stoltz 2011; Tolpin and Shimony 2012). In the selection problem, it is much better to minimize the *simple* regret. However, the simple and the cumulative regret cannot be minimized simultaneously; moreover, (Bubeck, Munos, and Stoltz 2011) shows that in many cases the smaller the cumulative regret, the greater the simple regret.

We begin with background definitions and related work. Some sampling schemes are introduced, and shown to have better bounds for the simple regret on sets than UCB, the first contribution of this paper. The results are applied to sampling in trees by combining the proposed sampling schemes on the first step of a rollout with UCT for the rest of the rollout. An additional sampling scheme based on metareasoning principles is also suggested, another contribution of this paper. Finally, the performance of the proposed sampling schemes is evaluated on sets of Bernoulli arms, in randomly generated 2-level trees, and on the sailing domain, showing where the proposed schemes have improved performance.

Background and Related Work

Monte-Carlo tree search was initially suggested as a scheme for finding approximately optimal policies for Markov Decision Processes (MDP). An MDP is defined by the set of states S, the set of actions A (also called *moves* in this paper), the transition distribution $T(s, a, s')$, the reward function $R(s, a, s')$, the initial state s and an optional goal state t: (S, A, T, R, s, t) (Russell and Norvig 2003). Several MCTS schemes explore an MDP by performing *rollouts*—trajectories from the current state to a state in which a termination condition is satisfied (either the goal state, or a cutoff state for which the reward is evaluated approximately).

Multi-armed bandits and UCT

In the Multi-armed Bandit problem (Vermorel and Mohri 2005) we have a set of K arms (see Figure 1.a). Each arm can be pulled multiple times. Sometimes a cost is associated with each pulling action. When the ith arm is pulled,

Copyright © 2012, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

a random reward X_i from an unknown stationary distribution is encountered. The reward is usually bounded between 0 and 1. In the cumulative setting (the focus of much of the research literature on Multi-armed bandits), all encountered rewards are collected by the agent. The UCB scheme was shown to be near-optimal in this respect (Auer, Cesa-Bianchi, and Fischer 2002):

Definition 1. *Scheme* $\mathbf{UCB(c)}$ *pulls arm* i *that maximizes upper confidence bound* b_i *on the reward:*

$$b_i = \overline{X}_i + \sqrt{\frac{c \log(n)}{n_i}} \tag{1}$$

where $\overline{X}_i$ *is the average sample reward obtained from arm* i*,* n_i *is the number of times arm* i *was pulled, and* n *is the total number of pulls so far.*

The UCT algorithm, an extension of UCB to Monte-Carlo Tree Search is described in (Kocsis and Szepesvári 2006), and shown to outperform many state of the art search algorithms in both MDP and adversarial games (Eyerich, Keller, and Helmert 2010; Gelly and Wang 2006).

In the simple regret (selection) setting, the agent gets to collect only the reward of the last pull.

Definition 2. *The* ***simple regret*** $\mathbb{E}r$ *of a sampling policy for the Multi-armed Bandit Problem is the expected difference between the best true expected reward* μ_* *and the true expected reward* μ_j *of the arm with the greatest sample mean,* $j = \arg\max_i \overline{X}_i$*:*

$$\mathbb{E}r = \sum_{j=1}^{K} \Delta_j \Pr(j = \arg\max_i \overline{X}_i) \tag{2}$$

where $\Delta_j = \mu_* - \mu_j$*.*

Strategies that minimize the simple regret are called pure exploration strategies (Bubeck, Munos, and Stoltz 2011). An upper bound on the simple regret of uniform sampling is exponentially decreasing in the number of samples (see (Bubeck, Munos, and Stoltz 2011), Proposition 1). For UCB(c) the best known respective upper bound on the simple regret of UCB(c) is only polynomially decreasing in the number of samples (see (Bubeck, Munos, and Stoltz 2011), Theorems 2,3). However, empirically UCB(c) appears to yield a lower simple regret than uniform sampling.

Metareasoning

A completely different scheme for control of sampling can use the principles of bounded rationality (Horvitz 1987) and metareasoning — (Russell and Wefald 1991) provided a formal description of rational metareasoning and case studies of applications in several problem domains. In search, under myopic and sub-tree independence assumptions, one maintains a current best move α at the root, and finds the expected gain from finding another move β to be better than the current best (Russell and Wefald 1991). The "cost" of search actions can also be factored in. Ideally, an "optimal" sampling scheme, to be used for selecting what to sample, both at the root node (Hay and Russell 2011) and elsewhere, can be developed using metareasoning. However, this task is daunting for the following reasons:

- The method is in general intractable, necessitating simplifying assumptions. However, using the standard metareasoning myopic assumption, where samples would be selected as though at most one sample can be taken before an action is chosen, we run into serious problems. Even the basic selection problem (Tolpin and Shimony 2012) exhibits a non-concave utility function and results in premature stopping of the standard myopic algorithms. This is due to the fact that the value of information of a single measurement (analogous to a sample in MCTS) is frequently less than its time-cost, even though this is not true for multiple measurements.

  When applying the selection problem to MCTS, the situation is exacerbated. The utility of an action is usually bounded, and thus in many cases a single sample may be insufficient to change the current best action, *regardless* of its outcome. As a result, we frequently get a *zero* "myopic" value of information for a single sample.
- Rational metareasoning requires a known distribution model, which may be difficult to obtain.
- Defining the time-cost of a sample is not trivial.

As the above ultimate goal is extremely difficult to achieve, we introduce in this paper simple schemes more amenable to analysis, loosely based on the metareasoning concept of value of information, and compare them to UCB (on sets) and UCT (in trees).

Sampling Based on Simple Regret

Analysis of Sampling on Sets

We examine two sampling schemes with super-polynomially decreasing upper bounds on the simple regret. The bounds suggest that these schemes achieve a lower simple regret than uniform sampling; indeed, this is confirmed by experiments.

We first consider ε-greedy sampling as a straightforward generalization of uniform sampling:

Definition 3. *The* ε***-greedy*** *sampling scheme pulls the arm that currently has the greatst sample mean, with probability* $0 < \varepsilon < 1$*, and any other arm with probability* $\frac{1-\varepsilon}{K-1}$*.*

This sampling scheme exhibits an exponentially decreasing simple regret:

Theorem 1. *For every* $0 < \eta < 1$ *and* $\gamma > 1$ *there exists* N *such that for any number of samples* $n > N$ *the simple regret of the* ε*-greedy sampling scheme is bounded from above as*

$$\mathbb{E}r_{\varepsilon\text{-}greedy} \leq 2\gamma \sum_{i=1}^{K} \Delta_i \exp\left(\frac{-2\Delta_j^2 n\varepsilon}{\left(1 + \sqrt{\frac{(K-1)\varepsilon}{1-\varepsilon}}\right)^2} \right) \tag{3}$$

with probability at least $1 - \eta$*.*

Proof outline: Bound the probability P_i that a non-optimal arm i is selected. Split the interval $[\mu_i, \mu_*]$ at $\mu_i + \delta_i$. Apply

the Chernoff-Hoeffding bound to get:

$$\begin{aligned} P_i &\leq \Pr[\overline{X}_i > \mu_i + \delta_i] + \Pr[\overline{X}_* < \mu_* - (\Delta_i - \delta_i)] \\ &\leq \exp\left(-2\delta_i^2 n_i\right) + \exp\left(-2(\Delta_i - \delta_i)^2 n_*\right) \end{aligned} \quad (4)$$

Observe that, in probability, $\overline{X}_i \to \mu_i$ as $n \to \infty$, therefore $n_* \to n\varepsilon$, $n_i \to \frac{n(1-\varepsilon)}{K-1}$ as $n \to \infty$. Conclude that for every $0 < \eta < 1$, $\gamma > 1$ there exists N such that for every $n > N$ and all non-optimal arms i:

$$P_i \leq \gamma \left(\exp\left(\frac{-2\delta_i^2 n(1-\varepsilon)}{K-1} \right) + \exp\left(-2(\Delta_i - \delta_i)^2 n\varepsilon\right) \right) \quad (5)$$

Require

$$\begin{aligned} \exp\left(-\frac{2\delta_i^2 n(1-\varepsilon)}{K-1} \right) &= \exp\left(-2(\Delta_i - \delta_i)^2 n\varepsilon\right) \\ \frac{\delta_i}{\Delta_i - \delta_i} &= \sqrt{\frac{(K-1)\varepsilon}{1-\varepsilon}} \end{aligned} \quad (6)$$

Substitute (5) together with (6) into (2) and obtain

$$\mathbb{E}r_{\varepsilon\text{-}greedy} \leq 2\gamma \sum_{i=1}^{K} \Delta_i \exp\left(\frac{-2\Delta_i^2 n\varepsilon}{\left(1 + \sqrt{\frac{(K-1)\varepsilon}{1-\varepsilon}}\right)^2} \right) \quad (7)$$

□

In particular, as the number of arms K grows, the bound for $\frac{1}{2}$-greedy sampling ($\varepsilon = \frac{1}{2}$) becomes considerably tighter than for uniform random sampling ($\varepsilon = \frac{1}{K}$):

Corollary 1. *For uniform random sampling,*

$$\mathbb{E}r_{uniform} \leq 2\gamma \sum_{i=1}^{K} \Delta_i \exp\left(-\frac{\Delta_i^2 n}{K} \right) \quad (8)$$

For $\frac{1}{2}$-greedy sampling,

$$\begin{aligned} \mathbb{E}r_{\frac{1}{2}\text{-}greedy} &\leq 2\gamma \sum_{i=1}^{K} \Delta_i \exp\left(\frac{-2\Delta_i^2 n}{\left(1 + \sqrt{K-1}\right)^2} \right) \\ &\approx 2\gamma \sum_{i=1}^{K} \Delta_i \exp\left(\frac{-2\Delta_i^2 n}{K} \right) \textit{ for } K \gg 1 \end{aligned} \quad (9)$$

ε-greedy is based solely on sampling the arm that has the greatest sample mean (henceforth called the "current best" arm) with a higher probability then the rest of the arms, and ignores information about sample means of other arms. On the other hand, UCB distributes samples in accordance with sample means, but, in order to minimize cumulative regret, chooses the current best arm too often. Intuitively, a better scheme for simple regret minimization would distribute samples in a way similar to UCB, but would sample the current best arm less often. This can be achieved by replacing $\log(\cdot)$ in Equation 1 with a faster growing sublinear function, for example, $\sqrt{\cdot}$.

Definition 4. *Scheme* $\mathbf{UCB}_{\sqrt{\cdot}}(\mathbf{c})$ *pulls arm i that maximizes b_i, where:*

$$b_i = \overline{X}_i + \sqrt{\frac{c\sqrt{n}}{n_i}} \quad (10)$$

where, as before, $\overline{X}_i$ is the average reward obtained from arm i, n_i is the number of times arm i was pulled, and n is the total number of pulls so far.

This scheme also exhibits a super-polynomially decreasing simple regret:

Theorem 2. *For every $0 < \eta < 1$ and $\gamma > 1$ there exists N such that for any number of samples $n > N$ the simple regret of the UCB$_{\sqrt{\cdot}}(c)$ sampling scheme is bounded from above as*

$$\mathbb{E}r_{ucb\sqrt{\cdot}} \leq 2\gamma \sum_{i=1}^{K} \Delta_i \exp\left(-\frac{c\sqrt{n}}{2} \right) \quad (11)$$

with probability at least $1 - \eta$.

Proof outline: Bound the probability P_i that a non-optimal arm i is chosen. Split the interval $[\mu_i, \mu_*]$ at $\mu_i + \frac{\Delta_i}{2}$. Apply the Chernoff-Hoeffding bound to get:

$$\begin{aligned} P_i &\leq Pr\left[\overline{X}_i > \mu_i + \frac{\Delta_i}{2} \right] + \Pr\left[\overline{X}_* < \mu_* - \frac{\Delta_i}{2} \right] \\ &\leq \exp\left(-\frac{\Delta_i^2 n_i}{2} \right) + \exp\left(-\frac{\Delta_i^2 n_*}{2} \right) \end{aligned} \quad (12)$$

Observe that, in probability, $n_i \to \frac{c\sqrt{n}}{\Delta_i^2}$, $n_i \leq n_*$ as $n \to \infty$. Conclude that for every $0 < \eta < 1$, $\gamma > 1$ there exists N such that for every $n > N$ and all non-optimal arms i:

$$P_i \leq 2\gamma \exp\left(-\frac{c\sqrt{n}}{2} \right) \quad (13)$$

Substitute (13) into (2) and obtain

$$\mathbb{E}r_{ucb\sqrt{\cdot}} \leq 2\gamma \sum_{i=1}^{K} \Delta_i \exp\left(-\frac{c\sqrt{n}}{2} \right) \quad (14)$$

□

Sampling in Trees

As mentioned above, UCT (Kocsis and Szepesvári 2006) is an extension of UCB for MCTS, that applies UCB(c) at each step of a rollout. At the root node, the sampling in MCTS is usually aimed at finding the first move to perform. Search is re-started, either from scratch or using some previously collected information, after observing the actual outcome (in MDPs) or the opponent's move (in adversarial games). Once one move is shown to be the best choice with high confidence, the value of information of additional samples of the best move (or, in fact, of any other samples) is low. Therefore, one should be able to do better than UCT by optimizing *simple regret*, rather than *cumulative regret*, at the root node.

Nodes deeper in the search tree are a different matter. In order to support an optimal move choice at the root, it

is beneficial in many cases to find a more precise estimate of the *value* of the state in these search tree nodes. For these internal nodes, optimizing simple regret is not the answer, and cumulative regret optimization is not so far off the mark. Lacking a complete metareasoning for sampling, which would indicate the optimal way to sample both root nodes and internal nodes, our suggested improvement to UCT thus combines different sampling schemes on the first step and during the rest of each rollout:

Definition 5. *The **SR+CR MCTS sampling scheme** selects an action at the current root node according to a scheme suitable for minimizing the simple regret (**SR**), such as $\frac{1}{2}$-greedy or UCB$_{\sqrt{}}$, and (at non-root nodes) then selects actions according to UCB, which approximately minimizes the cumulative regret (**CR**).*

The pseudocode of this two-stage rollout for an undiscounted MDP is in Algorithm 1: FIRSTACTION selects the first step of a rollout (line 5), and NEXTACTION (line 6) selects steps during the rest of the rollout (usually using UCB). The reward statistic for the selected action is updated (line 10), and the sample reward is back-propagated (line 11) towards the current root.

We denote such two-step realizations of SR+CR as *Alg*+UCT, where *Alg* is the sampling scheme employed at the first step of a rollout (e.g. $\frac{1}{2}$-greedy+UCT).

Algorithm 1 Two-stage Monte-Carlo tree search sampling

```
 1: procedure ROLLOUT(node, depth=1)
 2:     if ISLEAF(node, depth) then
 3:         return 0
 4:     else
 5:         if depth=1 then action ← FIRSTACTION(node)
 6:         else action ← NEXTACTION(node)
 7:         next-node ← NEXTSTATE(node, action)
 8:         reward ← REWARD(node, action, next-node)
 9:                  + ROLLOUT(next-node, depth+1)
10:         UPDATESTATS(node, action, reward)
11:         return reward
12:     end if
13: end procedure
```

We expect such two-stage sampling schemes to outperform UCT and be significantly less sensitive to the tuning of the exploration factor c of UCB(c). That is since the contradiction between the need for a larger value of c on the first step (simple regret) and a smaller value for the rest of the rollout (cumulative regret) (Bubeck, Munos, and Stoltz 2011) is resolved. In fact, a sampling scheme that uses UCB(c) at all steps but a larger value of c for the first step than for the rest of the steps, should also outperform UCT.

VOI-aware Sampling

Further improvement can be achieved by computing or estimating the value of information (VOI) of the rollouts and choosing rollouts that maximize the VOI. However, as indicated above, actually computing the VOI is infeasible. Instead we suggest the following scheme based on the following features of value of information:

1. An estimate of the probability that one or more rollouts will make another action appear better than the current best α.
2. An estimate of the gain that may be incurred if such a change occurs.

If the distribution of results generated by the rollouts were known, the above features could be easily computed. However, this is not the case for most MCTS applications. Therefore, we estimate bounds on the feature values from the current set of samples, based on the myopic assumption that the algorithm will only sample one of the actions, and use these bounds as the feature values, to get:

$$\begin{aligned} VOI_\alpha &\approx \frac{\overline{X}_\beta}{n_\alpha + 1} \exp\left(-2(\overline{X}_\alpha - \overline{X}_\beta)^2 n_\alpha\right) \qquad (15) \\ VOI_i &\approx \frac{1 - \overline{X}_\alpha}{n_i + 1} \exp\left(-2(\overline{X}_\alpha - \overline{X}_i)^2 n_i\right),\ i \neq \alpha \\ \text{where} \quad & \alpha = \arg\max_i \overline{X}_i, \quad \beta = \arg\max_{i,\, i \neq \alpha} \overline{X}_i \end{aligned}$$

with VOI_α being the (approximate) value for sampling the current best action, and VOI_i is the (approximate) value for sampling some other action i.

These equations were derived as follows. The gain from switching from the current best action α to another action can be bounded: by the current expectation of the value the current second-best action for the case where we sample only α, and by 1 (the maximum reward) minus the current expectation of α when sampling any other action. The probability that another action be found best can be bounded by an exponential function of the difference in expectations when the true value of the actions becomes known. But the effect of each individual sample on the sample mean is inversely proportional to the current number of samples, hence the current number of samples (plus one in order to handle the initial case of no previous samples) in the denominator.

These VOI estimates are used in the "VOI-aware" sampling scheme as follows: sample the action that has maximum estimated VOI. We judged these estimates to be too crude to be used as "stopping criteria" that can be used to cut off sampling, leaving this issue for future research. Although this scheme appears too complicated to be amenable to a formal analysis, early experiments (Section) with this approach demonstrate a significantly lower simple regret.

Empirical Evaluation

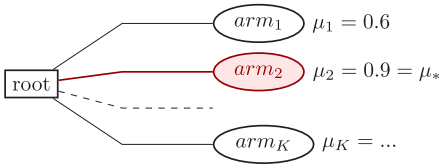
The results were empirically verified on Multi-armed Bandit instances, on search trees, and on the sailing domain, as defined in (Kocsis and Szepesvári 2006). In most cases, the experiments showed a lower average simple regret for $\frac{1}{2}$-greedy an UCB$_{\sqrt{}}$ than for UCB on sets, and for the SR+CR scheme than for UCT in trees.

Simple regret in multi-armed bandits

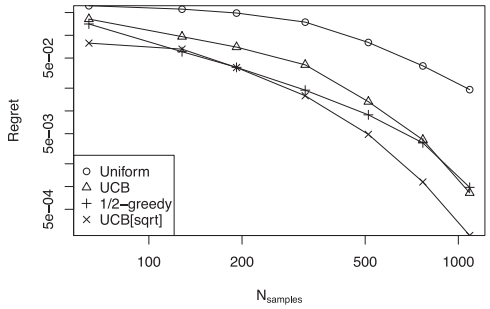
Figure 1 presents a comparison of MCTS sampling schemes on Multi-armed bandits. Figure 1.a shows the search tree corresponding to a problem instance. Each arm returns a random reward drawn from a Bernoulli distribution. The search

selects an arm and compares the expected reward, unknown to the algorithm during the sampling, to the expected reward of the best arm.

Figure 1.b shows the regret vs. the number of samples, averaged over 10000 experiments for randomly generated instances of 32 arms. Either $\frac{1}{2}$-greedy or UCB$_{\sqrt{\cdot}}$ dominate UCB over the whole range. For larger number of samples, the advantage of UCB$_{\sqrt{\cdot}}$ over $\frac{1}{2}$-greedy becomes more significant.



a. search tree, with best arm shaded



b. regret vs. number of samples

Figure 1: Simple regret in MAB

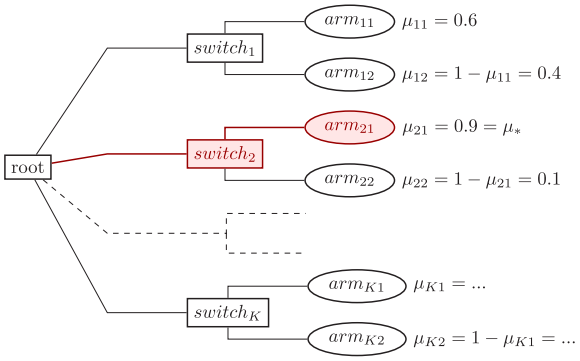
Monte Carlo tree search

The second set of experiments was performed on randomly generated 2-level max-max trees crafted so as to deliberately deceive uniform sampling (Figure 2.a), necessitating an adaptive sampling scheme, such as UCT. That is due to the switch nodes, each with 2 children with anti-symmetric values, which would cause a uniform sampling scheme to incorrectly give them all a value of 0.5.

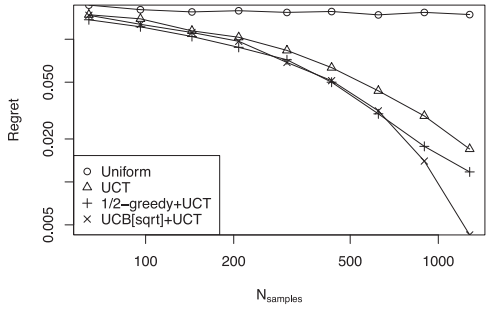
Simple regret vs. the number of samples are shown for trees with root degree 16 (Figure 2.b) and 64 (Figure 2.c). The exploration factor c is set to 2, the default value for rewards in the range $[0, 1]$. The algorithms exhibit a similar relative performance: either $\frac{1}{2}$-greedy+UCT or UCB$_{\sqrt{\cdot}}$+UCT result in the lowest regret, UCB$_{\sqrt{\cdot}}$+UCT dominates UCT everywhere except when the number of samples is small. The advantage of both $\frac{1}{2}$-greedy+UCT and UCB$_{\sqrt{\cdot}}$+UCT grows with the number of arms.

The sailing domain

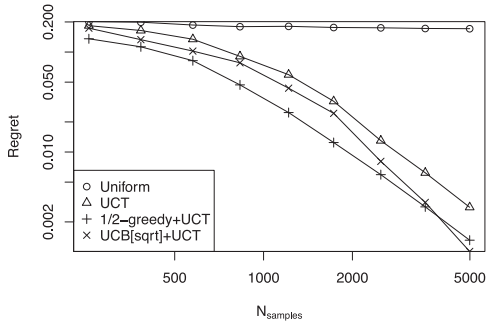
Figures 3–5 show results of experiments on the sailing domain. Figure 3 shows the regret vs. the number of samples, computed for a range of values of c. Figure 3.a shows the median cost, and Figure 3.b — the minimum costs. UCT is always worse than either $\frac{1}{2}$-greedy+UCT or UCB$_{\sqrt{\cdot}}$+UCT, and is sensitive to the value of c: the median cost is much higher than the minimum cost for UCT.



a. search tree, with path to the best arm shaded



b. 16 arms



c. 64 arms

Figure 2: MCTS in random trees

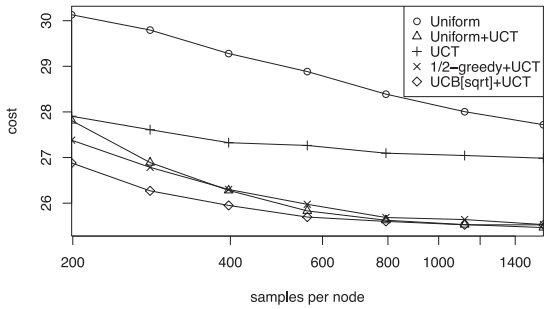
For both $\frac{1}{2}$-greedy+UCT and UCB$_{\sqrt{\cdot}}$+UCT, the difference is significantly less prominent.

Figure 4 shows the regret vs. the exploration factor for different numbers of samples. UCB$_{\sqrt{\cdot}}$+UCT is always better than UCT, and $\frac{1}{2}$-greedy+UCT is better than UCT expect for a small range of values of the exploration factor.

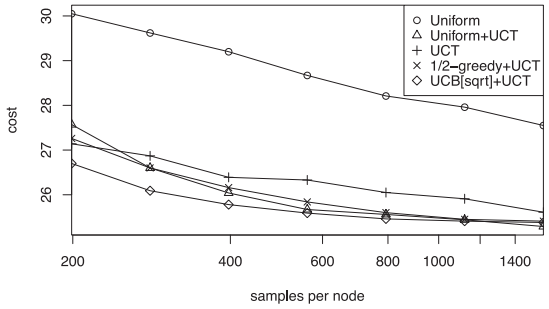
Figure 5 shows the cost vs. the exploration factor for lakes of different sizes. The relative difference between the sampling schemes becomes more prominent when the lake size increases.

VOI-aware MCTS

Finally, the VOI-aware sampling scheme was empirically compared to other sampling schemes (UCT, $\frac{1}{2}$-greedy+UCT, UCT$_{\sqrt{\cdot}}$+UCT). Again, the experiments were performed on randomly generated trees with structure shown in Figure 2.a. Figure 6 shows the results for 32 arms. VOI+UCT, the

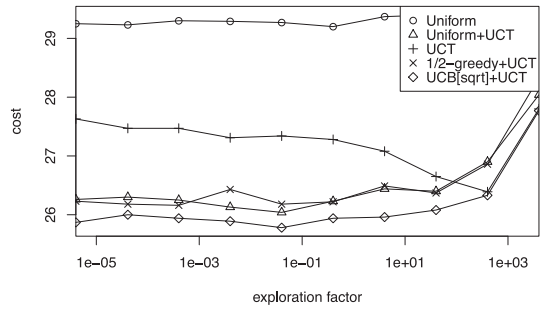


a. median cost

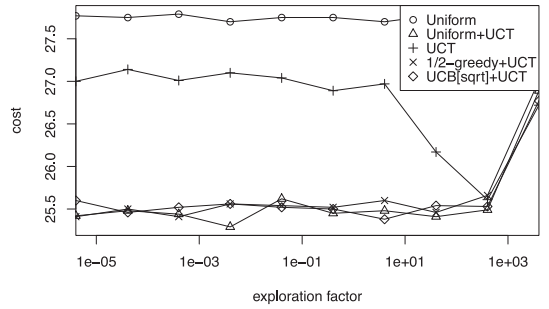


b. minimum cost

Figure 3: The sailing domain, 6×6 lake, cost vs. samples



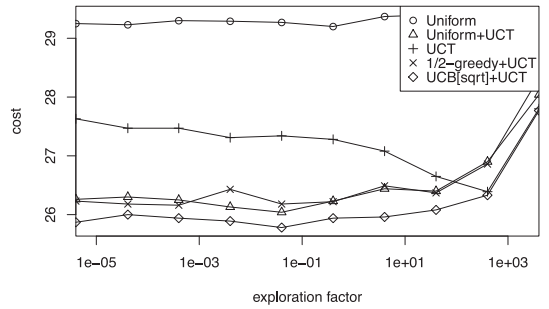
a. 397 rollouts



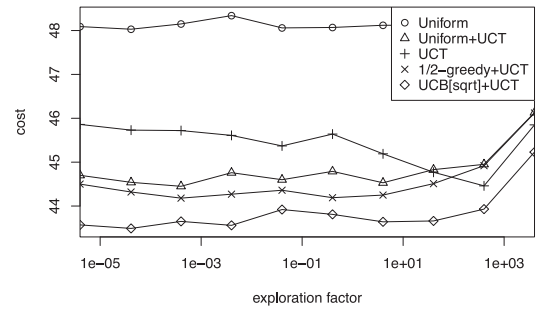
b. 1585 rollouts

Figure 4: The sailing domain, 6×6 lake, cost vs. factor

scheme based on a VOI estimate, outperforms all other sampling schemes in this example. Similar performance improvements (not shown) also occur for the sailing domain.



a. 6×6 lake



b. 10×10 lake

Figure 5: The sailing domain, 397 rollouts, cost vs. factor

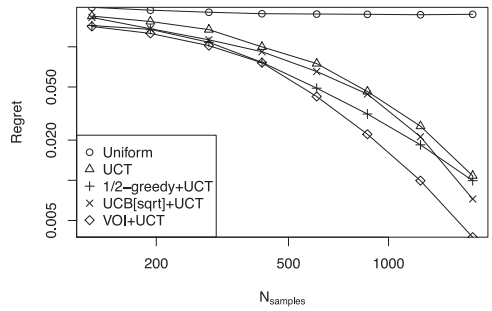


Figure 6: MCTS in random trees, including VOI+UCT.

Conclusion and Future Work

UCT-based Monte-Carlo tree search has been shown to be very effective for finding good actions in both MDPs and adversarial games. Further improvement of the sampling scheme is thus of interest in numerous search applications. We argue that although UCT is already very efficient, one can do better if the sampling scheme is considered from a metareasoning perspective of value of information (VOI).

The MCTS SR+CR scheme presented in the paper differs from UCT mainly in the first step of the rollout, when we attempt to minimize the 'simple' selection regret rather than the cumulative regret. Both the theoretical analysis and the empirical evaluation provide evidence for better general performance of the proposed scheme.

Although SR+CR is inspired by the notion of VOI, the VOI is used there implicitly in the analysis of the algorithm, rather than computed or learned explicitly in order

to plan the rollouts. Ideally, using VOI to control sampling ab-initio should do even better, but the theory for doing that is still not up to speed. Instead we suggest a "VOI-aware" sampling scheme based on crude probability and value estimates, which despite its simplicity already shows a marked improvement in minimizing regret. However, application of the theory of rational metareasoning to Monte Carlo Tree Search is an open problem (Hay and Russell 2011), and both a solid theoretical model and empirically efficient VOI estimates need to be developed.

Finding a better sampling scheme for non-root nodes, as well as the root node, should also be possible. Although cumulative regret does reasonably well there, it is far from optimal, as meta-reasoning principles imply that an optimal scheme for these nodes must be asymmetrical (e.g. it is not helpful to find out that the value of the current best action is even better than previously believed).

Finally, applying VOI methods in complex deployed applications that already use MCTS is a challenge that should be addressed in future work. In particular, UCT is extremely successful in Computer Go (Gelly and Wang 2006; Braudiš and Loup Gailly 2011; Enzenberger and Müller 2009), and the proposed scheme should be evaluated on this domain. This is non-trivial, since Go programs typically use "non-pure" versions of UCT, extended with domain-specific knowledge. For example, Pachi (Braudiš and Loup Gailly 2011) typically re-uses information from rollouts generated for earlier moves, thereby violating our underlying assumption that information is only used for selecting the current move. In early experiments not shown here (disallowing re-use of samples, admittedly not really a fair comparison) the VOI-aware scheme apears to dominate UCT. Nevertheless, it should also be possible to adapt the VOI-aware schemes to take into account expected re-use of samples, another topic for future research.

Acknowledgments

The research is partially supported by Israel Science Foundation grant 305/09, by the Lynne and William Frankel Center for Computer Sciences, and by the Paul Ivanier Center for Robotics Research and Production Management.

References

Auer, P.; Cesa-Bianchi, N.; and Fischer, P. 2002. Finite-time analysis of the multiarmed bandit problem. *Mach. Learn.* 47:235–256.

Braudiš, P., and Loup Gailly, J. 2011. Pachi: State of the art open source Go program. In *ACG 13*.

Bubeck, S.; Munos, R.; and Stoltz, G. 2011. Pure exploration in finitely-armed and continuous-armed bandits. *Theor. Comput. Sci.* 412(19):1832–1852.

Enzenberger, M., and Müller, M. 2009. Fuego - An Open-source Framework for Board Games and Go Engine Based on Monte-Carlo Tree Search. Technical report, University of Alberta, Dept. of Computing Science, TR09-08.

Eyerich, P.; Keller, T.; and Helmert, M. 2010. High-quality policies for the Canadian traveler's problem. In *In Proc. AAAI 2010*, 51–58.

Gelly, S., and Wang, Y. 2006. Exploration exploitation in Go: UCT for Monte-Carlo Go. *Computer*.

Hay, N., and Russell, S. J. 2011. Metareasoning for Monte Carlo Tree Search. Technical Report UCB/EECS-2011-119, EECS Department, University of California, Berkeley.

Horvitz, E. J. 1987. Reasoning about beliefs and actions under computational resource constraints. In *Proceedings of the 1987 Workshop on Uncertainty in Artificial Intelligence*, 429–444.

Kocsis, L., and Szepesvári, C. 2006. Bandit based Monte-Carlo planning. In *ECML*, 282–293.

Russell, S. J., and Norvig, P. 2003. *Artificial Intelligence: A Modern Approach*. Pearson Education.

Russell, S., and Wefald, E. 1991. *Do the right thing: studies in limited rationality*. Cambridge, MA, USA: MIT Press.

Tolpin, D., and Shimony, S. E. 2012. Semimyopic measurement selection for optimization under uncertainty. *IEEE Transactions on Systems, Man, and Cybernetics, Part B* 42(2):565–579.

Vermorel, J., and Mohri, M. 2005. Multi-armed bandit algorithms and empirical evaluation. In *ECML*, 437–448.