Lookahead Pathologies for Single Agent Search

Vadim Bulitko and LihongLi and RussGreiner and Ilya Levner
Department of Computing Science
University of Alberta
Edmonton, Alberta T6G 2E8
CANADA
{bulitko|lihong|greiner|ilya}@cs.ualberta.ca

Abstract

Admissible and consistent heuristic functions are usually preferred in single-agent heuristic search as they guarantee optimal solutions with complete search methods such as A* and IDA*. Larger problems, however, frequently make a complete search intractable due to space and/or time limitations. In particular, a path-planning agent in a real-time strategy game may need to take an action before its complete search has the time to finish. In such cases, incomplete search techniques (such as RTA*, SRTA*, RTDP, DTA*) can be used. Such algorithms conduct a limited ply lookahead and then evaluate the states envisioned using a heuristic function. The action selected on the basis of such evaluations can be suboptimal due to the incompleteness of search and inaccuracies in the heuristic. It is usually believed that deeper lookahead increases the chances of taking the optimal action. In this paper, we demonstrate that this is not necessarily the case, even when admissible and consistent heuristic functions are used.

1 Lookahead Pathologies in Real-time Single-agent Search

Complete search methods such as A* [Hart *et al,* 1968] and IDA* [Korf, 1985] produce optimal solutions when based on an admissible and monotonic heuristic function. The primary drawbacks are the exponential running time and the necessity to wait until the search completes before the first action can be taken [Korf, 1990], This limits the applicability of complete search in practice as the deliberation time per action can be severely limited [Higgins, 2002], the domain model can be expensive [Bulitko and Wilkins, 2002], the goal states can be difficult to recognize [Levner *et al,* 2002]. Consequently, despite numerous advances in improving heuristic functions [Korf and Taylor, 1996; Culberson and Schaeffer, 1994; Reinefeld, 1993; Korf, 1997], incomplete real-time/on-line search methods remain the practical choice for complex real-life problems.

Various incomplete search methods have been proposed including: RTA* [Korf, 1990], RTDP [Barto *et al,* 1995], SRTA*, and DTA* [Russell and Wefald, 1991]. Such algorithms base their decisions on heuristic information collected from a partial tree expansion (lookahead) prior to reaching the goal state. Since the heuristic function is generally inaccurate and the search is incomplete, suboptimal solutions can be produced even with admissible and consistent heuristics.

It is widely believed that looking ahead deeper improves the chances of taking the right action. Consequently, a considerable amount of effort has been put into increasing the lookahead depth by using selective search (search extensions) and hardware and software optimizations.

In this paper we demonstrate that looking ahead deeper can actually *decrease* the chances of taking the optimal action even when admissible and consistent heuristic functions are used.

2 Related Past Research & Our Novel Contributions

Lookahead pathologies within the mini-max search in two-player games have been investigated extensively in the past. In [Nau, 1982; 1983; Beal, 1980; 1982; 1983; Bratko and Gams, 1982; Pearl, 1983], the primary cause of pathologies was deemed to be the independence of heuristic values of the leaf nodes. Such games were called *non-incremental* Large branching factors were also considered contributing to a pathology. Later, [Michon, 1983] added *non-inertness* (i.e., the constant branching factor) to the "black list" of properties that can cause a pathology.

The efforts presented in this paper differ from the past work in several key areas: (i) we demonstrate lookahead pathologies in *single-agent* heuristic search; (ii) pathologies are shown to take place even when the true and heuristic values of sibling states are correlated (i.e., in *incremental domains);* (iii) unlike research in [Schriifer, 1986; Beal, 1980] that dealt with two-valued (win/loss) heuristics, we consider *real-valued heuristic functions;* (iv) we show pathologies with the *smallest non-trivial branching factor* (two); (v) pathologies are demonstrated for *admissible and consistent* heuristic functions.

3 Framework

In this study we consider a single-agent heuristic search in a discrete state domain with a finite number of deterministic actions. The states (set *S)* and actions (set *A)* form a directed graph with certain specified vertices representing the goal states. The edges (actions) are weighed with action costs:

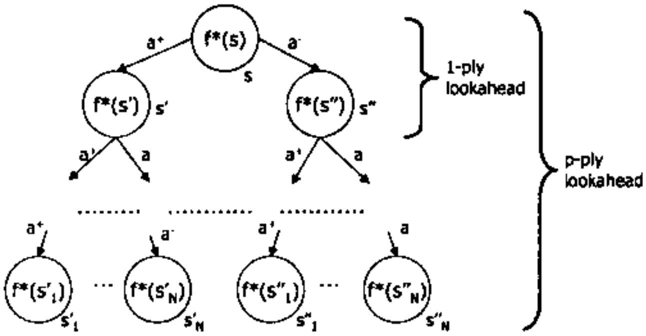

Figure 1: Illustration of p-ply binary lookahead tree. Throughout the paper we display the true f* in the circles, the states or heuristic f values are shown next to the circles, and the edges are labeled with actions. Here u^+ and a' are the optimal and suboptimal actions respectively.

$c : A \to R$. The agent is provided with a perfect domain model: $\delta : S \times A \to S$.

We define the true distance-to-goal function $h^*(s)$ as the sum of action costs along the shortest path from state s to the closest goal state. Generally speaking, the agent uses an approximation h to the unavailable h^*. The approximation is typically inaccurate insomuch as: $\exists s \in S\,[h^*(s) \neq h(s)]$.

For a fixed starting state .s, function $g(s')$ is defined as the sum of action costs along the shortest path from *s* to *s'*. Finally, the sum of *h* or *h** and *g* is typically denoted by / or /*. It easy to see that /* remains constant along any optimal path from a fixed state *s* to the closest goal. Also note that, for any state .s-', action *a* is inferior to action *u.2* iff $f^*(\delta(s', a_1)) > f^*(\delta(s', a_2))$.

Located in state s, the agent can use its perfect model *S* to predict the states it will get to upon taking various sequences of actions. A binary lookahead tree is illustrated in Figure 1. For the sake of clarity and simplicity we will be using small binary lookahead trees to illustrate the discussion throughout the paper.

The lookahead policy $\pi(s,p)$ operates as follows: (i) start in the current state *s;* (ii) construct the lookahead search tree of;; plies deep by envisioning terminal states of all action sequences *of p* actions (whenever possible); (iii) evaluate the leaf nodes of the lookahead tree using the / function and select the minimum-valued state; (iv) output the single action leading to selected leaf state (resolve ties randomly).

Depending on the lookahead tree, random tie resolution, and the approximate heuristic /, the action *a* = $\pi(s,p)$ **output by** $\pi(s,p)$ **ca** be suboptimal: $\exists a^* \neq a\,[f^*(\delta(s,a^*)) < f^*(\delta(s,a))]$. The probability of such an error for state $s \in S$ is denoted by *Err(s,p).* Additionally, we will consider the expected value of $Err(s,p)$ over the states .s. Such state-independent quantity will be referred to as *Err(p).*

The combination of domain and heuristic function *h* (/) is called *pathological* iff $\exists p_1 < p2$ *[Err(s,p1) < Err(s,p2)].* The corresponding state-independent version is: $\exists p_1 < P_2$ *[Err(p_1) < Err(p2)].* The intuitive meaning is quite transparent: lookahead search is pathological iff looking deeper ahead is expected to *increase* the chances of taking

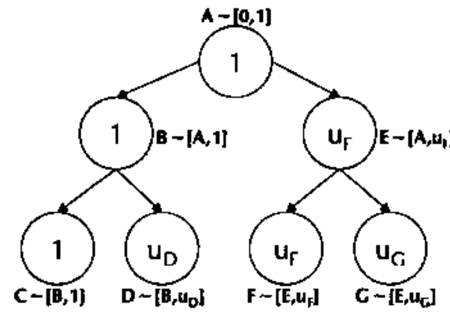

Figure 2: The 7-node 2-ply lookahead tree used in the analysis of pathologies with admissible and consistent heuristics.

a suboptimal action.

We adopt the following standard terminology. Function h is called admissible iff $\forall s \in S\,[h(s) \leq h^*(s)]$. It is called consistent iff $\forall a \in A, s \in S\,[h(s) \leq c(a) + h(\delta(s,a))]$. It can be shown that consistency of h is equivalent to non-decreasing monotonicity of $f = g + h$: $\forall a \in A, s \in S\,[f(s) \leq f(\delta(s,a))]$. In the following we will assume that the true h^* satisfies the mini-min relation: $\forall s\,[h^*(s) = \min_{a \in A}(c(a) + h^*(\delta(s,a)))]$ and h^* is, therefore, consistent and, trivially, admissible.

4 Admissible & Consistent Heuristics

Complete searches (e.g., A* and IDA*) produce optimal solutions with admissible heuristics. Thus, much effort has gone into derivation of admissible heuristics either manually or automatically (e.g., [Korf and Taylor, 1996; Prieditis, 19931). Consistency *of h* leads to non-decreasing monotonia ity of / and often speeds up complete searches conducted by A*/IDA*. Incomplete searches (e.g., RTA*) can produce suboptimal solutions even with consistent and admissible heuristics. It is widely believed, however, that deeper lookahead reduces chances of such errors. Remarkably, this is not necessarily the case.

Consider the binary lookahead tree in Figure 2. Each node is shown with its true value f*(s) inside the circle. For each state .s, heuristic *f(s)* is drawn uniformly from $[f(s_{\text{parent}}), f^*(s)]$ (indicated with the '~' in the boxes next to the circles). The value *(A)* of the root state is drawn from [0, /*(s_{root})]. This makes the / function admissible and monotonically non-decreasing.

We will refer to the drawn values of / as *A,B,C,D,E,F,G* (illustrated in the figure). Assume that $u_D > 1$, $u_G > u_F > 1$. By definition, the lookahead will be pathological iff $Err(s_{\text{root}}, 2) > Err(s_{\text{root}}, 1)$. The probability of taking the suboptimal action (i.e., going to the right child of the root node) with the lookahead of one is:

$$\begin{aligned} Err(s_{\text{root}}, 1) &= Pr(E < B) \\ &= \int_0^1 \int_A^1 \int_E^1 \frac{1}{(1-A)(u_F - A)}\, dB\, dE\, dA \\ &= \frac{1}{2}\left[1 + (u_F - 1)\ln\left(1 - \frac{1}{u_F}\right)\right]. \end{aligned}$$

Similarly we derive an expression for the lookahead of two: *Err(s_{root},2)* which can not be reproduced in this short pa-

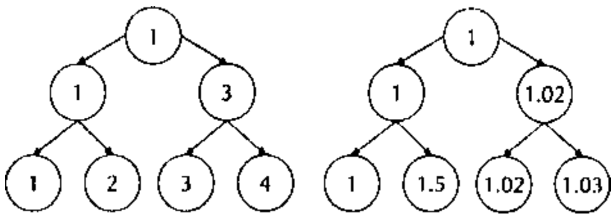

Figure 3: Admissible and consistent heuristics: non-pathological and pathological lookahead trees.

per*. By analyzing the difference of the two expressions we arrive at a criterion of pathology.

A concrete illustration of this phenomenon can be found in Figure 3. The tree on the left is non-pathological as: $Err(s_{root}, 1) \approx 0.0945 > 0.0226 \approx Err(s_{ront}, 2)$. The tree on the right, however, is indeed pathological. Averaged over 10 million trials: $Err(s_{root}, 1) \approx 0.4607 < 0.4861 \approx Err(s_{root}, 2)$.

5 Future Work & Conclusions

In practice, lookahead pathologies with real-time (single-agent or two-player) search are *infrequently* observed even when inadmissible heuristics are used (cf, [Thompson, 1982] for the game of chess). Consequently, future work directions include: (i) identification of the properties of practically used heuristic functions (e.g., Manhattan distance in the 8 puzzle) that are responsible for the *lack* of pathologies, (ii) extension of the current formal analysis to lookahead trees with a variable branching factor and a deeper lookahead, (iii) methods for on-line and off-line pathology detection and correction and (iv) connections between the probability of taking sub-optimal actions and a decrease in the cumulative expected reward.

Acknowledgements

Rene Levia, Dan Lizotte, Omid Madani, Xin Li, and Yiping Zhou have contributed in various ways. We are grateful for the funding from the University of Alberta, NSFRC, and the Alberta Ingenuity Centre for Machine Learning.

References

[Barto et al., 1995] A. G. Barto, S. J. Bradtke, and S. P. Singh. Learning to act using real-time dynamic programming. *Artificial Intelligence,* 72(1):81-138, 1995.

[Beal, 1980] D.F. Beal. An analysis of minimax. In M.R.B. Clarke, editor, *Advances in Computer Chess,* volume 2, pages 103 109. Edinburgh University Press, 1980.

[Beal, 1982] D.F. Beal. Benefits of minimax search. In M.R.B. Clarke, editor, *Advances in Computer Chess,* volume 3, pages 17 24. Oxford : Pergamon, 1982.

[Beal, 1983] D.F. Beal. Recent progress in understanding minimax search. In *Proceedings of ACM Annual Conference,* pages 164-169, New York, 1983. Association for Computing Machinery.

The interested reader is referred to the project website at http://www.cs.ualberta.ca/ircl.

[Bratko and Gams, 1982] I. Bratko and M. Gams. Error analysis of the minimax principle. In M.R.B. Clarke, editor, *Advances in Computer Chess,* volume 3, pages 1-15. Oxford : Pergamon, 1982.

[Bulitko and Wilkins, 2002] V. Bulitko and D.C. Wilkins. Real-time decision making for shipboard damage control. In *Proceedings of the AAAI/KDD/UAI-2002 Joint Workshop on Real-Time Decision Support and Diagnosis Systems,* pages 37 46, Edmonton, Alberta, 2002. AAAI Press.

[Culberson and Schaeffer, 1994] J. Culberson and J. SchaelTer. Efficiently searching the 15-puzzle. Technical Report TR 94-08, Department of Computing Science, University of Alberta, 1994.

[Hart *et ai,* 1968] P.E. Hart, N.J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans, on Systems Science and Cybernetics,* 4(2): 100-107, 1968.

[Higgins, 2002] Dan Higgins. Pathfinding design architecture. In Steve Rabin, editor, *AI Game Programming Wisdom,* pages 122-132. Charles River Media, 2002.

[Korf and Taylor, 1996] Richard E. Korf and Larry A. Taylor. Finding optimal solutions to the twenty-four puzzle. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI-96),* pages 1202 1207, Portland, Oregon, USA, 1996. AAAI Press / The MIT Press.

iKorf, 1985] R. Korf. Depth-first iterative deepening : An optimal admissible tree search. *Artificial Intelligence,* 27(1), 1985.

[Korf, 1990] Richard E. Korf. Real-time heuristic search. *Artificial Intelligence,* 42(2-3): 189 211, 1990.

[Korf, 1997] R. Korf. Finding optimal solutions to rubik's cube using pattern databases. In *Proceedings of the Workshop on Computer Games (W31) at IJCAI-97,* pages 21-26, Nagoya, Japan, 1997.

[Levner et al., 2002] I. Levner, V. Bulitko, O. Madani, and R. Greiner. Performance of lookahead control policies in the face of abstractions and approximations. In S. Koenig and R. Holte, editors, *Lecture Notes in Artificial Intelligence (LNAI),* volume 2371, pages 299-308. Springer-Verlag, Berlin, Heidelberg, 2002.

[Michon, 1983] G. Michon. *Recursive Random Games: A Probablistic Model for Perfect Information Games.* PhD thesis, University of California at Los Angeles, 1983.

[Nau, 1982] D.S. Nau. An investigation of the causes of pathology in games. *Artificial Intelligence,* 19:257-278, 1982.

I Nau, 1983] D.S. Nau. Pathology on game trees revisited, and an alternative to minimaxing. *Artificial Intelligence,* 21(1,2):221 244,1983.

[Pearl, 1983] J. Pearl. On the Nature of Pathology in Game Searching. *Artificial Intelligence,* 20:427 453, 1983.

[Prieditis, 1993] A. E. Prieditis. Machine discovery of effective admissible heuristics. *Machine Learning,* 12:117-141, 1993.

[Reincfcld, 1993] Alexander Reinefeld. Complete solution of the eight-puzzle and the benefit of node ordering in IDA. In *IJCAI,* pages 248-253, 1993.

[Russell and Wefald, 1991] Stuart Jonathan Russell and Eric H. Wefald. *Do the right thing: Studies in limited rationality.* MIT Press, 1991.

[Schriifer, 1986] G. Schriifer. Presence and absence of pathology on game trees. In D.F. Beal, editor, *Advances in Computer Chess,* volume 4, pages 101-112. Oxford: Pergamon, 1986.

[Thompson, 1982] K. Thompson. Computer chess strength. In M.R.B. Clarke, editor, *Advances in Computer Chess,* volume 3, pages 55 56. Oxford : Pergamon, 1982.