An AND-OR-Tree Connected to Leaves via Communication Channels

Toshio Suzuki, *Member, IAENG*

***Abstract*—We introduce a successor model of an AND-OR tree. Leaves are connected to internal nodes via communication channels that possibly have high probability of interruption. By depth-first communication we mean the following protocol: if a given algorithm probes a leaf then it continues to make queries to that leaf until return of an answer. For each such tree, we give a concrete example of interruption probability setting with the following property. For any independent and identical distribution on the truth assignments (probability is assumed to be neither 0 nor 1), any depth-first search algorithm that performs depth-first communication is not optimal. This result makes sharp contrast with the counterpart on the usual AND-OR tree (Tarsi) that optimal and depth-first algorithm exists. A key to the proof is Riemann zeta function.**

***Index Terms*—Analysis of algorithms and problem complexity, AND-OR tree, independent distribution, interruption, Riemann zeta function.**

I. Introduction

An AND-OR tree is a mini-max tree whose evaluation function is Boolean-valued, in other words, value is 1 (true) or 0 (false). The root is labeled by AND, child nodes of an AND-gate (an OR-gate, respectively) are labeled by OR (AND, respectively). Each leaf has Boolean value and their values are hidden. An algorithm probes leaves to find the Boolean value of the root, and during computation, the algorithm skips a leaf if unnecessary. Cost of computation is measured by the number of leaves probed during computation. Given a probabilistic distribution on the truth assignments to the leaves, cost means expected value of the above mentioned cost.

Computational complexity issues on AND-OR trees have been studied from the early stage of artificial intelligence ([2], [4], [5], [13] and [8]).

On the other hand, current systems of artificial intelligence are often consist of many devices that communicate each other. Interruption of communication is one of potential risks in such systems.

We propose a successor model of an AND-OR tree in which each leaf is connected to an internal node via a communication channel. We are interested in the case where each channel has high probability of interruption. Figure 1 is an example of such a tree. Circles are internal nodes, squares are leaves, solid lines are usual wire, and broken lines are communication channels. In our mind, the main body of the tree is in our local computer, but leaves are on remote devices.

The most simple type of probability distribution on an AND-OR tree is an *independent and identical distribution* (IID for short). More precisely, an IID is a distribution such that there is a fixed positive real number $p \leq 1$ and each leaf has independently has value 0 with probability p.

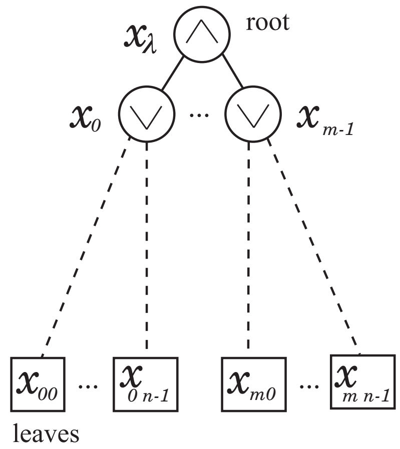

Fig. 1. broken lines are communication channels

A tree is *balanced* (in the sense of Tarsi [13]) if (1) any two internal nodes of the same depth (the distance from the root) have the same number of child nodes and (2) all the leaves have the same depth.

An algorithm A is *depth-first* if for every internal node x, once A probes a leaf that is descendant of x then A does not probe leaves that are non-descendants of x until A finds value of x. A is *directional* if there is a fixed linear order of the leaves, and for any truth assignment to the leaves, the order of probe by A is consistent to the above mentioned linear order [4].

The above (standard) definition of depth-first is not exact for our purpose. In our computation model, "algorithm A makes a query to leaf x" is merely a necessary condition for "A finds value of x".

Thus, we redefine the concept of depth-first, and in addition, introduce concept of depth-first communication.

Definition Let A be an algorithm on a tree.

1) A performs *depth-first search* (or simply, A is depth-first) if for every internal node x, once A finds value of a leaf that is descendant of x then A does not make queries to leaves that are non-descendants of x until A finds value of x.
2) A (possibly does not perform depth-first search) performs *depth-first communication* if for each leaf, once A makes a query to a leaf, A consecutively makes queries to that leaf until return of an answer.

■

About an IID on an AND-OR tree, the following result of Tarsi is important and well-known. If $0 < p < 1$, there is an

Manuscript received December 28, 2017; revised January 16, 2018. This work was supported in part by JSPS KAKENHI 16K05255.

Department of mathematics and Information Sciences, Tokyo Metropolitan University, Hachioji-city, Tokyo 192-0397, Japan e-mail: toshio-suzuki@tmu.ac.jp

optimal algorithm (its cost achieves the minimum among all algorithms) that is depth-first and directional [13].

Suppose that T is a balanced AND-OR tree, and that an attached distribution is an IID with $0 < p < 1$ (p is probability of a leaf having value 0). For any leaf v and any positive integer k, at kth query to v, assume that probability of interruption depends only on k, not depending on v. Let $f(k)$ be the probability. We give a particular example of a function f with the following property: Any depth-first algorithm that performs depth-first communication is not optimal (The main theorem).

This result and the above result of Tarsi contrast sharply.

Our $f(x)$ is $x^2/(x+1)^2$. In the proof, a key tool is Riemann zeta function.

In section III, we observe cost of getting value of a leaf via consecutive access to it through a communication channel. In section IV, we introduce our interruption probability setting on a tree of height 1. In section V, we investigate a tree of general height, and show our main result.

II. Preliminaries

As usual, $\sum$ and $\prod$ denote sum and product, respectively. Throughout the paper, an expression of the form $\sum_{i=k}^{k-1}[\cdots]$ denotes 0, and $\prod_{i=k}^{k-1}[\cdots]$ denotes 1.

We denote Riemann zeta function [1, Chapter 23] by ζ. Thus, for each $s > 1$, $\zeta(s) = \sum_{n=1}^{\infty} n^{-s}$. In particular, $\zeta(2) = \pi^2/6 = 1.6449\cdots$ and $\zeta(4) = \pi^4/90 = 1.0823\cdots$.

For two events E_1 and E_2, we denote conditional probability of E_2 under E_1 by $\mathrm{prob}[E_2|E_1]$.

The paper given in [9] is a concise survey on complexity and equilibria of AND-OR trees, by which the reader can overview the previous research [3] and its subsequent developments [11], [12] and [7]. For more recent works on this line, see the papers [6] and [10].

III. Consecutive queries to a particular leaf

In this section, we investigate a single leaf x_0 with a communication channel (Figure 2). A procedure P consecutively makes queries to x_0 until return of an answer.

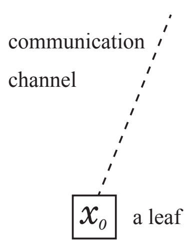

Fig. 2. a single leaf with a communication channel

For each positive integer n, we look at the following events $E_{n,0}$ and $E_{n,1}$.

$E_{n,0}$: "For each j such that $1 \leq j < n$, jth query to x_0 is interrupted (that is, P does not receive an answer)".

$E_{n,1}$: "nth query to x_0 is interrupted".

Lemma 1 (Cost of getting 1-bit information) *Let k be a positive integer. Let α_k be expected cost for P to get an answer, under the assumption that for all n,* (1) *holds.*

$$\mathrm{prob}[E_{n,1}|E_{n,0}] = [(n+k-1)/(n+k)]^2 \tag{1}$$

Then, we have the following.

$$\alpha_k = k^2(\zeta(2) - \sum_{j=1}^{k-1} j^{-2}) \tag{2}$$

Recall that by our convention in the notation section, we have $\sum_{j=1}^{-1} j^{-2} = 0$. Thus, in particular, the following holds.

$$\alpha_1 = \zeta(2) \tag{3}$$

Proof: Let $f(x) = [x/(x+1)]^2$.

$$\begin{aligned} \alpha_k &= \sum_{j=1}^{\infty} \mathrm{prob}[E_{j,0} \wedge \neg E_{j,1}] \times j \\ &= \sum_{j=1}^{\infty} [(\prod_{i=k}^{k+j-2} f(i))(1 - f(k+j-1))j] \end{aligned}$$

Here, we have the following.

$$\begin{aligned} & \sum_{j=1}^{n} [(\prod_{i=k}^{k+j-2} f(i))(1 - f(k+j-1))j] \\ =& \sum_{j=1}^{n} [j \prod_{i=k}^{k+j-2} f(i) - j \prod_{i=k}^{k+j-1} f(i)] \\ =& \sum_{j=1}^{n} \prod_{i=k}^{k+j-2} f(i) - n \prod_{i=k}^{k+n-1} f(i) \\ =& \sum_{j=1}^{n} [k/(k+j-1)]^2 - n[k/(k+n)]^2 \\ =& k^2[\sum_{j=1}^{n} (k+j-1)^{-2} - n/(k+n)^2] \\ =& k^2[\sum_{j=1}^{k+n-1} j^{-2} - \sum_{j=1}^{k-1} j^{-2} - n/(k+n)^2] \\ \to& k^2(\zeta(2) - \sum_{j=1}^{k-1} j^{-2}) \ (n \to \infty) \end{aligned} \tag{4}$$

Hence, (2) holds. ■

IV. Height 1 binary tree

In this section, we investigate a binary OR-tree of height 1 with a communication channel for each leaf (Figure 3). We are going to compare depth-first communication and non-depth-first communication.

Example (Cost of depth-first communication) Let p be a real number such that $0 < p < 1$. Let d_p be an IID such that probability of a leaf having value 0 is p. Assume that interruption probabilities are given by (1) with $k = 1$.

$$\mathrm{prob}[E_{n,1}|E_{n,0}] = [n/(n+1)]^2 \tag{5}$$

ISBN: 978-988-14047-8-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

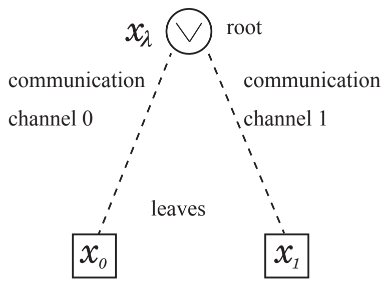

Fig. 3. height 1, binary case

In this model, we investigate the following algorithm $L^\omega R^\omega$: Make queries to x_0 until x_0 returns an answer. If the value is 0 then make queries to x_1 until x_1 returns an answer.

Let $\mathrm{cost}(L^\omega R^\omega, d_p; [n/(n+1)]^2)$ denote the expected cost of $L^\omega R^\omega$ under d_p and the interruption probabilities (5). Then, by Lemma 1, we have the following.

$$\begin{aligned}\mathrm{cost}(L^\omega R^\omega, d_p; [n/(n+1)]^2) &= \alpha_1 + p\alpha_1 \\ &= (1+p)\zeta(2) \quad (6)\end{aligned}$$

■

Let $(LR)^\omega$ denotes the following algorithm. Repeat the following while none of x_0 and x_1 returns an answer: "Make a query to x_0, then make a query to x_1". If x_i returns an answer, break the above loop. If x_i is 0 then make queries to x_{1-i} until x_{1-i} returns an answer.

Let p and d_p be those in Example, and assume that interruption probabilities are given by (5). Let $\mathrm{cost}((LR)^\omega, d_p; [n/(n+1)]^2)$ denote the expected cost of $L^\omega R^\omega$ under d_p and the above interruption probabilities.

We are going to define sequence $\{\beta_k\}_{k=0,1,2,\cdots}$ so that $\mathrm{cost}((LR)^\omega, d_p; [n/(n+1)]^2) = \sum_{k=0}^{\infty} \beta_k$. In the following, $f(x)$ denotes $[x/(x+1)]^2$, and α_k is that defined in Lemma 1.

For each $i \in \{0,1\}$ and $n \geq 1$, let E'_n be the event "The first return from a leaf happens at nth access to the leaves". Let $\beta_n = \mathrm{prob}[E'_n] \times (n + p\alpha_{(n+1)/2})$ (if n is odd), and $\beta_n = \mathrm{prob}[E'_n] \times (n + p\alpha_{(n+2)/2})$ (if n is even). Then the following holds.

$$\beta_1 = (1-f(1))(1+p\alpha_1), \quad (7)$$

$$\beta_2 = f(1)(1-f(1))(2+p\alpha_2), \quad (8)$$

$$\begin{aligned}&\beta_{2k+1} \\ =\ &(\prod_{j=1}^{k} f(j)^2)(1-f(k+1))(2k+1+p\alpha_{k+1}), \quad (9)\end{aligned}$$

$$\begin{aligned}&\beta_{2k+2} \\ =\ &(\prod_{j=1}^{k} f(j)^2)f(k+1)(1-f(k+1))(2k+2+p\alpha_{k+2}) \\ &\quad (10)\end{aligned}$$

Lemma 2

$$\mathrm{cost}((LR)^\omega, d_p; [n/(n+1)]^2) = 2\zeta(2) + (1-p)(\zeta(4)-3) \quad (11)$$

Proof: It is easy to verify the following.

$$\begin{aligned}&\beta_{2k+1} + \beta_{2k+2} \\ =\ &(k+1)^{-4} \\ &+\frac{2k+p\alpha_{k+1}}{(k+1)^4} - \frac{2(k+1)+p\alpha_{k+2}}{(k+2)^4} \\ &+\frac{1}{(k+1)^2(k+2)^2} + \frac{p(\alpha_{k+2}-\alpha_{k+1})}{(k+1)^2(k+2)^2} \quad (12)\end{aligned}$$

Hence, we have the following.

$$\begin{aligned}&\sum_{j=1}^{2k+2} \beta_j \\ =\ &\sum_{j=1}^{k+1} j^{-4} + p\alpha_1 - (2k+2+p\alpha_{k+2})/(k+2)^4 \\ +\ &\sum_{j=0}^{k} \frac{1}{(j+1)^2(j+2)^2} + \sum_{j=0}^{k} \frac{p(\alpha_{j+2}-\alpha_{j+1})}{(j+1)^2(j+2)^2} \\ &\quad (13)\end{aligned}$$

Here, the following holds.

$$\sum_{j=1}^{k+1} j^{-4} \to \zeta(4)\ (k \to \infty) \quad (14)$$

Throughout the rest of the proof, let σ_x denote $\sum_{j=1}^{x} j^{-2}$. By Lemma 1, the following hold.

$$p\alpha_1 = p\,\zeta(2), \quad (15)$$

$$\begin{aligned}&(2k+2+p\alpha_{k+2})/(k+2)^4 \\ =\ &O(k^{-3}) + p(\zeta(2)-\sigma_{k+1})/(k+2)^2 \\ \to\ &0\ (k \to \infty) \quad (16)\end{aligned}$$

The third term of (13) is estimated as follows.

$$\begin{aligned}&\sum_{j=0}^{k} \frac{1}{(j+1)^2(j+2)^2} \\ =\ &\sum_{j=0}^{k}(-\frac{2j+1}{(j+1)^2} + \frac{2j+3}{(j+2)^2} + \frac{2}{(j+2)^2}) \\ =\ &-3 + (2k+3)/(k+2)^2 + 2\sigma_{k+2} \\ \to\ &-3 + 2\zeta(2)\ (k \to \infty) \quad (17)\end{aligned}$$

Again, by Lemma 1, we get the following.

$$\begin{aligned}&(\alpha_{j+2}-\alpha_{j+1})/(j+1)^2(j+2)^2 \\ =\ &[((j+1)^{-2}-(j+2)^{-2})\zeta(2) - (j+1)^{-4}] \\ &-((j+1)^{-2}-(j+2)^{-2})\sigma_j \\ &\quad (18)\end{aligned}$$

Here, the sum of $[\cdots]$ has the following limit.

$$\begin{aligned}&\sum_{j=0}^{k}[((j+1)^{-2}-(j+2)^{-2})\zeta(2) - (j+1)^{-4}] \\ =\ &(1-(k+2)^{-2})\zeta(2) - \sum_{j=1}^{k+1} j^{-4} \\ \to\ &\zeta(2) - \zeta(4)\ (k \to \infty) \quad (19)\end{aligned}$$

We are going to show $\sum_{j=0}^{\infty}[((j+1)^{-2}-(j+2)^{-2})\sigma_j] = -3+2\zeta(2)$. Let a be the left-hand side. Here, we have $\sigma_0 = 0$, thus we may ignore the term for $j = 0$.

$$\begin{aligned} a &= \sum_{j=1}^{\infty}[(j+1)^{-2}\sum_{k=1}^{j}k^{-2} - (j+2)^{-2}\sum_{k=1}^{j+1}k^{-2}] \\ &\quad + \sum_{j=1}^{\infty}(j+1)^{-2}(j+2)^{-2} \\ &= 1/4 - \lim_{n\to\infty}(n+2)^{-2}\sum_{k=1}^{n+1}k^{-2} \\ &\quad + \sum_{j=1}^{\infty}(j+1)^{-2}(j+2)^{-2} \\ &= \sum_{j=0}^{\infty}(j+1)^{-2}(j+2)^{-2} \\ &= -3+2\zeta(2) \qquad \text{[by (17)]} \qquad (20) \end{aligned}$$

By (18), (19) and (20), we can evaluate the fourth term of (13).

$$\sum_{j=0}^{\infty}\frac{p(\alpha_{j+2}-\alpha_{j+1})}{(j+1)^2(j+2)^2} = p(3-\zeta(2)-\zeta(4)) \qquad (21)$$

By (13), (15), (16), (17) and (21), we find the cost.

$$\begin{aligned} \mathrm{cost}((LR)^{\omega}, d_p; [n/(n+1)]^2) &= \sum_{j=1}^{\infty}\beta_j \\ &= 2\zeta(2) + (1-p)(\zeta(4)-3) \qquad (22) \end{aligned}$$

In other words, (11) holds. ■

Corollary *Suppose that* $0 < p < 1$. *Then,* $\mathrm{cost}((LR)^{\omega}, d_p; [n/(n+1)]^2)$ *is less than* $\mathrm{cost}(L^{\omega}R^{\omega}, d_p; [n/(n+1)]^2)$.

Proof: By Example and Lemma 2, $\mathrm{cost}(L^{\omega}R^{\omega}, d_p; [n/(n+1)]^2) - \mathrm{cost}((LR)^{\omega}, d_p; [n/(n+1)]^2) = (1-p)(-\zeta(2)-\zeta(4)+3) > 0$. ■

V. The theorem

Now, we investigate a tree of arbitrary height. Let T be a balanced AND-OR tree or a balanced OR-AND tree of height $h(\geq 1)$. Suppose that p is a real number such that $0 < p < 1$ and let d_p be an IID such that at each leaf, probability of having value 0 is p. Assume that interruption probabilities are given by (5).

Given an algorithm A on T and a real number p, let $\mathrm{cost}(A, d_p; [n/(n+1)]^2)$ denote the expected cost of A under d_p and the above interruption probabilities.

Theorem *In additon to the above setting, suppose that* A *is a depth-first algorithm on* T *and* A *performs depth-first communication (see Definition in Introduction). Then* A *is not optimal.*

Proof: We investigate the case where nodes just above leaves are OR-gates. The other case (they are AND-gates) is treated in a similar way.

The case where T is a binary OR-tree of height 1 is shown by Corollary. In the following, T is assumed to have more than 2 leaves.

Since the distribution d_p is an IID and $0 < p < 1$, there exists an initial segment γ of a computation path of A on T, such that γ has positive probability and after γ, A performs in the same way as $L^{\omega}R^{\omega}$.

More precisely, γ (its length is, say $k \geq 1$) consists of ordered pairs $\langle x^{(i)}, a^{(i)}\rangle$ ($i = 0, \dots, k-1$) of leaves $x^{(i)}$ and truth values $a^{(i)} \in \{0, 1\}$, and in addition there are leaves $x^{(k)}$ and $x^{(k+1)}$, and the following hold.

1) There exists an OR-gate (say, x_u) such that $x^{(k)}$ and $x^{(k+1)}$ are its child.
2) At the beginning of computation, A makes queries to $x^{(0)}$ until return of an answer.
3) For each $i < k$, if an answer of $x^{(i)}$ is $a^{(i)}$ then A makes queries to $x^{(i+1)}$ until return of an answer.
4) In the presence of d_p and the interruption probabilities given by (5), A performs the move γ (until getting the answer $a^{(k-1)}$) with positive probability.
5) If an answer of $x^{(k-1)}$ is $a^{(k-1)}$ then A makes queries to $x^{(k)}$ until return of an answer.
6) If an answer of $x^{(k)}$ is 0 then A makes queries to $x^{(k+1)}$ until return of an answer. Then A finds value of the root.
7) If an answer of $x^{(k)}$ is 1 then A finds value of the root.

Figure 4 illustrates the OR-gate x_u and its child leaves. For example, $x^{(k)}$ and $x^{(k+1)}$ are possibly $x_{u,m-2}$ and $x_{u,m-1}$, respectively. These two leaves are the last two leaves with positive probability.

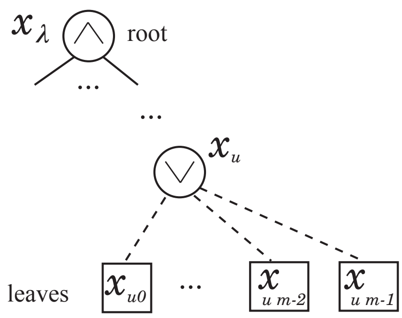

Fig. 4. general case

Now, let B be the following algorithm. B simulates A. However, if history γ (including the instance getting the answer $a^{(k-1)}$) happens then B performs as $(x^{(k)}x^{(k+1)})^{\omega}$, in other words, B obeys the following instructions. Repeat the following while none of x_k and x_{k+1} returns an answer: "Make a query to x_k, then make a query to x_{k+1}". If x_{k+i} ($i \in \{0, 1\}$) returns an answer, break the above loop, and then make queries to x_{k+1-i} until x_{k+1-i} returns an answer.

By Corollary, B has lower cost than A. ■

VI. Summary and future directions

We investigated multi-branching balanced AND-OR trees with communication channels between leaves and the main body. For each such tree, we showed concrete example of interruption probability setting with the following property:

For any independent and identical distribution on the truth values on the leaves (probability is assumed to be neither 0 nor 1), depth-first algorithms of depth-first communication are not optimal. Our main tool is Riemann zeta function.

The following are future directions.

- Characterization of optimal algorithms in the presence of the above interruption probability setting.
- Study on words of infinite length as algorithms of our model. For example, $L^\omega R^\omega$ is an infinite sequence $LL \cdots RR \cdots$, and $(LR)^\omega$ is $(LR)(LR) \cdots$.
- Application to computation model under emergency where batteries of devices have loss of power.

Acknowledgement

We are grateful to anonymous referees for helpful advices.

References

[1] Milton Abramowitz and Irene A. Stegun (eds.). *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*, tenth Printing. U.S. Government Printing Office, 1972.

[2] Donald E. Knuth and Ronald W. Moore. An analysis of alpha-beta pruning. *Artif. Intell.*, 6 (1975) 293–326.

[3] ChenGuang Liu and Kazuyuki Tanaka. Eigen-distribution on random assignments for game trees. *Inform. Process. Lett.*, 104 (2007) 73–77.

[4] Judea Pearl. Asymptotic properties of minimax trees and game-searching procedures. *Artif. Intell.*, 14 (1980) 113–138.

[5] Judea Pearl. The solution for the branching factor of the alpha-beta pruning algorithm and its optimality. *Communications of the ACM*, 25 (1982) 559–564.

[6] Weiguang Peng, NingNing Peng, KengMeng Ng, Kazuyuki Tanaka and Yue Yang. Optimal depth-first algorithms and equilibria of independent distributions on multi-branching trees. *Inform. Process. Lett.*, 125 (2017) 41–45.

[7] Weiguang Peng, Shohei Okisaka, Wenjuan Li and Kazuyuki Tanaka. The Uniqueness of eigen-distribution under non-directional algorithms. *IAENG Int. J. Comput. Sci.*, 43 (2016) 318–325.

[8] Michael Saks and Avi Wigderson. Probabilistic Boolean decision trees and the complexity of evaluating game trees. In: *Proc. 27th IEEE FOCS*, 1986, 29–38.

[9] Toshio Suzuki. Kazuyuki Tanaka's work on AND-OR trees and subsequent developments. *Annals of the Japan Association for Philosophy of Science*, 25 (2017) 79–88.

[10] Toshio Suzuki. Non-Depth-First Search against an Independent Distribution on a Balanced AND-OR Tree. *preprint*, arXiv:1709.07358[cs.DS] (2017).

[11] Toshio Suzuki and Ryota Nakamura. The eigen distribution of an AND-OR tree under directional algorithms. *IAENG Int. J. Appl. Math.*, 42 (2012) 122–128.

[12] Toshio Suzuki and Yoshinao Niida. Equilibrium points of an AND-OR tree: under constraints on probability. *Ann. Pure Appl. Logic* , 166 (2015) 1150–1164.

[13] Michael Tarsi. Optimal search on some game trees. *J. ACM*, 30 (1983) 389–396.