Improved Decision-Making in Game Trees: Recovering from Pathology

Arthur L. Delcher* and **Simon Kasif***

Computer Science Department
Loyola College in Maryland
Baltimore, MD 21210

Department of Computer Science
The Johns Hopkins University
Baltimore, MD 21218

Abstract

In this paper we address the problem of making correct decisions in the context of game-playing. Specifically, we address the problem of reducing or eliminating pathology in game trees. However, the framework used in the paper applies to decision making that depends on evaluating complex Boolean expressions. The main contribution of this paper is in casting general evaluation of game trees as belief propagation in causal trees. This allows us to draw several theoretically and practically interesting corollaries.

- In the Bayesian framework we typically do not want to ignore any evidence, even if it may be inaccurate. Therefore, we evaluate the game tree on several levels rather than just the deepest one.
- Choosing the correct move in a game can be implemented in a straightforward fashion by an efficient linear-time algorithm adapted from the procedure for belief propagation in causal trees.
- We propose a probabilistically sound heuristic that allows us to reduce the effects of pathology significantly.

Introduction

Decision-making in the presence of uncertainty is one of the most fundamental problems in Artificial Intelligence (common-sense reasoning), Economics, and the Social Sciences [Savage, 1972; von Neumann and Morgenstern, 1944; Pearl, 1988; Horvitz, 1988; Russell and Wefald, 1989]. In this paper we address the problem of making correct decisions in the context of game-playing. Specifically, we address the problem of reducing or eliminating pathology in game trees. However, the framework used in the paper applies to decision making that depends on evaluating complex Boolean expressions.

The main contribution of this paper is in casting evaluation of game trees as belief propagation in causal trees. This allows us to draw several theoretically and practically interesting corollaries.

- In the Bayesian framework we typically do not want to ignore any evidence, even if it may be inaccurate. Therefore, we evaluate the game tree on several levels rather than just the deepest one.
- Choosing the correct move in a game can be implemented in a straightforward fashion by an efficient linear-time algorithm adapted from the procedure for belief propagation in causal trees.
- Given estimates on the accuracy of the evaluation function in a game tree, we achieve drastic improvements in the quality of move decisions.
- We propose a probabilistically sound heuristic that allows us to reduce the effects of pathology significantly.

The paper consists of three parts. In the first part we develop a Bayesian decision procedure to choose correct moves in game trees. The main contribution of this section is in showing that optimal decision-making in game trees is a simple application of belief propagation in causal trees. This belief propagation allows us to derive a simple and computationally efficient decision procedure. We believe that this is a new application of belief propagation in causal trees. There are, however, two drawbacks in applying belief propagation in game trees.

1. The underlying assumption in Bayesian decision-making is that we have good estimates of all prior and conditional probabilities. In practice, these often are difficult to obtain.
2. The procedure requires evaluating every node in the tree (as in MIN/MAX). In practice, for large trees this is impractical.

Pruning search trees is an essential component in computer game playing. Therefore, we next propose a simple heuristic decision rule. This rule is motivated by probabilistic considerations, and based on our experiments appears to be effective. The rule helps reduce pathological behavior in "noisy" game trees and in some cases removes it altogether. The rule is based on the following observation. Given a set of witnesses

* Supported by NSF DARPA Grant CCR-8908092 and AFOSR Grant AFOSR-89-1151

for a binary random variable whose underlying statistics are unknown, it makes sense to use a weighted majority of the witnesses' reports to estimate the value of the underlying event. We analyze the error probability of this procedure.

Finally, we report the results of our preliminary experiments using the above procedures. We implemented our procedure for a one-dimensional version of the board-splitting game. We report very encouraging performance results and compare Bayesian analysis with our heuristic procedure.

Decision-making in game trees in a probabilistic setting has been considered before (for example, see [Fischer and Paleologou, 1991; Palay, 1983; Pearl, 1988; Nau, 1980; Russell and Wefald, 1989]). Our approach differs, however, in that it relies on evaluating multiple levels in the game tree, and uses both rules and heuristics to combine the evidence accumulated by belief propagation.

Pathology in Game Trees

Pathology in game trees is an interesting phenomenon observed by Nau and Beal. (See [Beal, 1980; Nau, 1980; Nau, 1983; Pearl, 1984] for many references on the subject.) It was observed that, given certain assumptions, when we search deeper, the performance of a MIN/MAX procedure for choosing the best move actually decreases. On the surface, this appears counterintuitive and even somewhat paradoxical. When the assumptions that make pathology appear are understood, however, it is a perfectly logical behavior.

Consider the following simple scenario. You are deciding whether to buy stock in XYZ Corporation. Your decision is based on the probability of the stock price rising. You consult n experts, who each have some probability of making an erroneous prediction. Suppose you are a very conservative investor and you will buy the stock iff all of the experts say it will go up. Then clearly, as the number of experts grows, the probability of at least one of them predicting the stock will fall, when it actually will rise, grows along with the number of experts. Accordingly, the probability that you will not buy the stock if its price rises (*i.e.*, the error probability) increases. In the Bayesian framework, however, if we know each expert's probability of error, it is usually better to have more experts.

Random Game Trees and Pathology

In this section we sketch a formal model of game trees where pathology is present. Our discussion follows Pearl [Pearl, 1984]. We assume the reader is familiar with the basic notation of computer game playing (2-player games) and MIN/MAX trees. Game trees can be defined inductively as follows:

1. The root of the tree (level 0) represents the initial position of the game (with player MAX to play).
2. If node X is on an even level ℓ, its successors on odd level $\ell + 1$ represent the positions that result after each possible move by player MAX at position X.
3. If node X is on an odd level ℓ, its successors on even level $\ell + 1$ represent the positions that result after each possible move by player MIN (the opponent) at position X.
4. The leaves of the tree represent terminal positions of the game and are labelled with either WIN or LOSS (1 and 0, respectively), depending on the outcome (from MAX's point of view).

MIN/MAXing is a standard procedure to decide the correct move. It is generally not possible to evaluate the complete MIN/MAX tree to the terminal positions, as its size grows exponentially. Note that if by some magical trick we actually had the correct evaluations of all nodes on level d (*i.e.*, we had the values that MIN/MAX would propagate up to these nodes from the terminal nodes) we could do the MIN/MAX from that level on. Therefore, it is a standard procedure to use a heuristic evaluation function to estimate the value (strength) of each node on some level d, and then use MIN/MAX to propagate these values to the root. Since this evaluation may be misleading, it is commonly assumed that employing the evaluation function at deeper levels of the tree produces better choices of moves at the root of the tree.

A random tree model of a game is a tree where we assume a probability distribution on the labels of the terminal nodes (leaves). In our case, we assume that the value of each leaf is WIN or LOSS with independent uniform probabilities p and $1 - p$, respectively.

The inaccuracy of the evaluation function on level d is modelled probabilistically by assuming that the error of the evaluation function *Eval* on level d is given by the probability p_e given by

$$\begin{aligned} p_e &= Pr(Eval(X) = \text{WIN} | X = \text{LOSS}) \\ &= Pr(Eval(X) = \text{LOSS} | X = \text{WIN}) \end{aligned}$$

That is, we assume that the evaluation errors on level d are distributed uniformly with probability p_e. In other words, p_e is the probability that the evaluation function on level d returns the incorrect result. Our results easily generalize to the cases where the one-sided probabilities of error are not the same and where p_e is a function of d. Generally, as one searches deeper the probability of error decreases. That is, p_e is the function of d. Our analysis in the next section applies to this case as well.

Given the model described above, pathology arises [Nau, 1980; Pearl, 1984]. Namely, when the evaluation function is employed deeper in the tree, the cumulative effect of errors may offset any increase in the accuracy of the evaluation function. Note, that this model postulates two fairly strong independence assumptions:

1. Independence of WIN's and LOSS'es of adjacent nodes on the terminal level.

2. Independence of the results of the evaluation function at different nodes on level d (*i.e.*, independence of the errors of adjacent nodes).

For proofs and further information on pathology see Nau [Nau, 1980] and Pearl [Pearl, 1984]. However, pathology arises in practice even when assumption 2 is removed (for examples, see the experiments section below). It does not seem to be a problem in many real games such as chess, checkers and others. A rationale for this behavior can be found in [Pearl, 1984].

Game Trees as Bayesian Networks

When making decisions in the presence of uncertainty, it is well-known that Bayes rule provides an optimal decision procedure, if we are given all prior and conditional probabilities. In this section we show how to model decision-making in games as belief propagation in causal trees. The result gives us a simple and efficient procedure to derive optimal decisions.

We model each node in the game tree as a binary random variable whose values are WIN and LOSS. At any given node, choosing the correct move in this model is identified with finding the successor node that has the highest probability of being a WIN. Under this assumption, our task is to find the probability of each successor of the root node being a WIN node, given all the evidence that we have.

Our evidence takes the form of an evaluation function that we can apply to any node in the tree to estimate whether it is a WIN or LOSS node. More precisely, with each node X in the game tree, we associate another random variable X' that represents the value of the evaluation function at node X. Thus, in general we assume that, instead of evaluating the game tree on only a single level d, we can evaluate it on all levels $1, 2, \ldots, d$. The evaluation value at X' is considered evidence for the actual (unknown) value of node X. Note also that the evaluation values of nodes that are ancestors and descendants of node X in the game tree may strongly influence our estimate of the true value of node X.

All the conditional dependencies, therefore, can be captured in a causal tree. For an exact definition of such a tree see Pearl [Pearl, 1988]. In such a tree, a directed arc from node X to node Y indicates that we know the conditional probability distribution $Pr(Y|X)$.

Pearl gives an efficient procedure to compute the belief distribution of every node in such a tree. Most importantly, this procedure operates by a simple propagation mechanism. In our case, we wish to compute, for each successor node of the root node of our game tree, the probability that it is a WIN, given all the evidence (*i.e.*, the values of the evaluation functions) contained in the tree down to some given depth d.

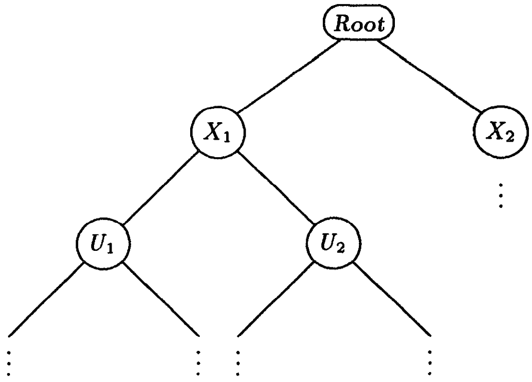

Figure 1: A fragment of a typical game tree.

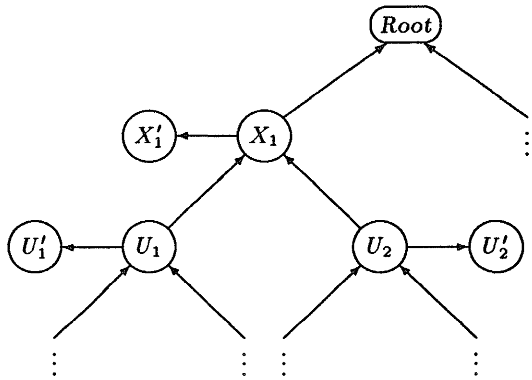

Figure 2: The corresponding causal tree.

Belief-Propagation in Game Trees

In this section we describe the basic ideas used to perform the Bayesian analysis of the tree by an efficient belief-propagation procedure. We describe the evaluation in game trees, where the value of each intermediate node can be expressed as the Boolean NAND of its immediate successors. Our technique generalizes in a straightforward fashion to a tree with an arbitrary Boolean function at the intermediate nodes. For simplicity, we sketch only the main ideas in such an evaluation. Consider a typical game tree as in Figure 1. The causal tree associated with it is shown in Figure 2. Our procedure evaluates the two probabilities that $x_1 = $ WIN and $x_2 = $ WIN given all the available evidence.

Without loss of generality consider X_1. Following Pearl, the belief distribution of x_1, defined by

$$Bel(x_1) = Pr(x_1|E_{below x_1}, x_1')$$

where $E_{below x_1}$ denotes all evidence nodes (*i.e.*, evaluation values) contained in X_1's subtree, can be computed as follows:

$$Bel(x_1) = \alpha Pr(x_1'|x) \cdot \sum_{u_1,u_2} Pr(x_1|u_1,u_2)Bel(u_1)Bel(u_2)$$

where α is a constant that makes the beliefs sum to one. Note that we can assume that the values of $Bel(u_1)$ and $Bel(u_2)$ in this equation have been computed recursively by the same calculation. The base case for this recursion occurs at the deepest level of the evaluation where, in the absence of any evidence below these nodes, we use the prior probabilities.

For the NAND function this calculation can be seen from the following table, where we denote $Pr(u_1 = 1|E_{below u_1}, u_1')$ and $Pr(u_2 = 1|E_{below u_2}, u_2')$ by p_{u_1} and p_{u_2}, respectively, and p_e denotes $Pr(x_1' \neq x_1)$:

u_1	u_2	x_1	x_1'	Probability
0	0	1	0	$(1-p_{u_1})(1-p_{u_2})p_e$
			1	$(1-p_{u_1})(1-p_{u_2})(1-p_e)$
0	1	1	0	$(1-p_{u_1})p_{u_2}p_e$
			1	$(1-p_{u_1})p_{u_2}(1-p_e)$
1	0	1	0	$p_{u_1}(1-p_{u_2})p_e$
			1	$p_{u_1}(1-p_{u_2})(1-p_e)$
1	1	0	0	$p_{u_1}p_{u_2}(1-p_e)$
			1	$p_{u_1}p_{u_2}p_e$

Thus, for example, if $x_1' = 1$ then

$$\begin{aligned} Bel(x_1 = 1) &= \alpha[(1-p_{u_1})(1-p_{u_2})(1-p_e) \\ &\quad + (1-p_{u_1})p_{u_2}(1-p_e) \\ &\quad + p_{u_1}(1-p_{u_2})(1-p_e)] \\ Bel(x_1 = 0) &= \alpha p_{u_1}p_{u_2}p_e \end{aligned}$$

where α is simply chosen to make these values sum to one. A similar calculation applies for any Boolean function.

Thus, we have the following:

Proposition 1: The probability distribution of X_1 is accurately modelled by the causal tree.

Proposition 2: (Pearl) $Bel(x_i)$ can be calculated in linear time.

A Heuristic Procedure

As mentioned in the introduction there are two drawbacks to the above procedure. While it provides a very effective way to make decisions in game trees (see the experimental results in the next section), it requires a complete evaluation of the tree. This is generally impossible. Pruning (such as alpha-beta) is an essential component of game tree evaluation. In addition, the Bayesian procedure requires knowing all prior and conditional probabilities. While, we can perhaps deal with the latter problem by using estimates of the probabilities (*cf.* the full version of this paper), the former problem appears to be fundamental. In fact, we believe that developing pruning mechanisms for very large Bayesian systems is a challenging and important open question.

In this section we propose a "quick fix" to the problem. Consider the following strategy. Rather then performing the belief propagation sketched in the previous section, we still use the standard pruning methods such as alpha-beta. However, we perform the evaluation more than once, each time searching to a different depth in the tree. Let M_i denote the move chosen by the i^{th} evaluation. Then the move we shall select is the one that represents the majority vote of the M_i's.

Evaluating the tree on several levels is not a new idea. This is precisely the strategy employed in existing chess programs, namely iteratively deepening alpha-beta. The motivation there is to use the previous levels to determine node ordering to improve the performance of subsequent deeper searches. However, we actually use the results of the shallow-level evaluation in choosing the move. This idea is motivated by probabilistic considerations. If we consider each one of the evaluations an independent Bernoulli trial, succeeding (*i.e.*, giving the correct result) with probability p, we actually can derive bounds on the error probability of this evaluation. The analysis is given in the full version of the paper. The experimental results given below support this intuition.

Experiments

In this section we apply the algorithms described in the previous section to the one-dimensional board-splitting game. The most important aspect of these results is that the approach outlined above works not only for an idealized model but also for a reasonable evaluation function.

Board-Splitting Game

The board-splitting game was suggested by Nau because it accurately models the assumptions that exhibit pathology in game trees. Most importantly, it exhibits pathology even when a plausible evaluation function is used, rather than using randomly generated noise to pollute the propagation of MIN/MAX values. We describe a one-dimensional version of the game. We start play on a Boolean linear array of length 2^n. We then randomly distribute ones and zeros to each cell on the board, choosing one and zero uniformly and independently with probabilities p and $1-p$, respectively. Players alternate turns. On his turn, a player splits the board exactly in the middle and chooses either the left or the right board. The game is continued on the chosen board, with the other side being discarded. The first player (MAX) wins iff the final remaining single-cell board contains a one; otherwise, MAX loses. It is clear that the number of ones in a given board is strongly correlated with the probability of MAX winning. Therefore, we use the function SUM, which simply sums the values on a given board, as a heuristic function in our experiments. Note that this function creates a strong

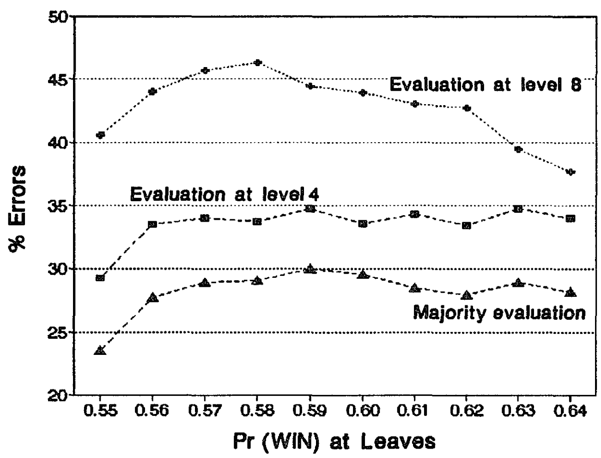

Figure 3: Results obtained from 10,000 trials of the board-splitting game on a depth-10 tree using the SUM heuristic evaluation.

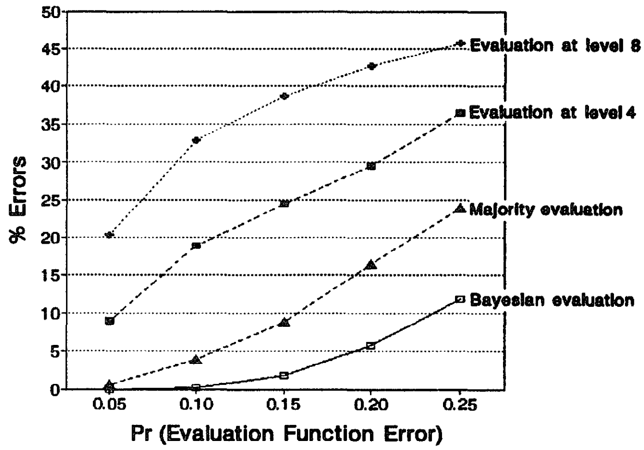

Figure 4: Results obtained from 10,000 trials of the board-splitting game on a tree of depth 10, using $p = Pr(\text{terminal node} = \text{WIN}) = 0.58$.

dependency between the evaluation of parent and child nodes in the game tree.

Experimental Results

We performed two types of experiments. First, we experimented with the SUM evaluation function. Second, we experimented with a noisy evaluation function that more closely matches the model where pathology occurs. Each experiment with the SUM was conducted as follows: In phase I, we generated a random board with 2^n cells (*i.e.*, a depth-n game tree). Using standard MIN/MAX propagation we computed the values at each of the two children of the root, *Left* and *Right*. In phase II, we evaluated the tree to a specified level $d < n$ and applied the evaluation function to each node on that level. Then, again we used MIN/MAX to propagate those values to *Left* and *Right*. Finally we compared the results of the two phases of the experiment. Repeating the above procedure for a specified number of trials, we incremented an error counter as follows:

Level-d Evaluation:	Level-n Evaluation:	Error
Left $\geq$ *Right*	*Left* $<$ *Right*	Yes
Left $<$ *Right*	*Left* $\geq$ *Right*	Yes
Otherwise		No

To compute the majority heuristic, we computed the values of *Left* and *Right* by separately evaluating the tree to each of several successive depths. Then we chose the move recommended by the majority of the separate evaluations. The error was computed as above. Some typical results are shown in Figure 3.

Note, that aggregating evidence using the majority heuristic outperforms the decision made by any of the individual evaluations. Additionally, the graph clearly shows the quality of our heuristic decision degrading as the frequency of ones increases. This behavior can be predicted from the probabilistic model. In these preliminary experiments we have not yet incorporated this information in the decision rule.

The experiments with inserting random noise were performed similarly, except that on level d we randomly perturbed the correct evaluation propagated from the terminal nodes by the MIN/MAX procedure. In other words, for a given evaluation error probability p_e, we would flip the correct value of a node on level d with probability p_e. In this fashion, for each node x, we generated the evidence node x' such that $Pr(x' = 1|x = 0) = Pr(x' = 0|x = 1) = p_e$. We repeated this on several levels and also compared the errors with the majority heuristic. The Bayesian analysis was performed using the belief propagation described earlier. Typical results are shown in Figure 4. We also compared the performance of our Bayesian analysis using evidence from a single (bottom) level *vs.* evidence obtained by board evaluations on multiple levels. Some typical results are shown in Figure 5.

Discussion

It is clear that when the evaluation function used by our program is inaccurate, the MIN/MAX procedure does not provide an effective way to make optimal choices of moves, and pathology occurs. Given an accurate probabilistic model of the game we can determine the move that is most likely to win by a complete Bayesian analysis of the evidence generated by our board evaluators applied to multiple levels of the game tree. In this paper we have pointed out that this evaluation can be done by adapting the known belief-propagation method used in causal trees. As expected we have shown experimentally that this yields

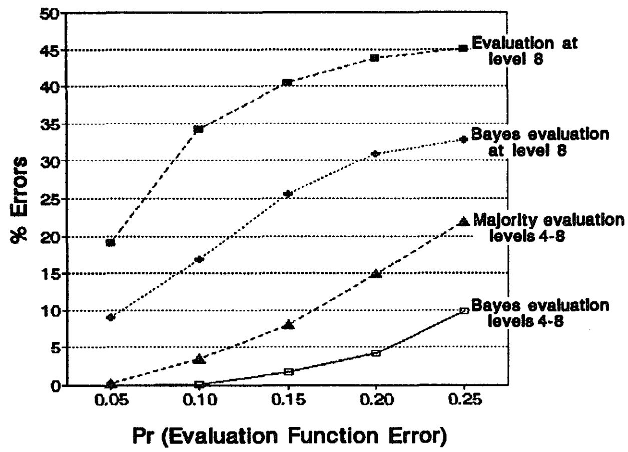

Figure 5: Results obtained from 10,000 trials of the board-splitting game on a tree of depth 12, using $p = Pr(\text{terminal node} = \text{WIN}) = 0.58$.

a very effective decision-making mechanism. Most importantly, it significantly reduces pathology. Pearl has pointed out [Pearl, 1984] that in the presence of a noisy evaluation function we may benefit from using Bayesian product-propagation rules to propagate probabilistic estimates of the quality of positions on the bottom level. We have extended this technique relying on the fact that in a probabilistic setting it is better to consider all available evidence. We also have pointed out two drawbacks of Bayesian analysis in the context of game trees. First, the analysis is sensitive to the degree of accuracy of the probabilities involved in the computation. Second, and most important, like the MIN/MAX procedure, we would need to generate very large trees for a typical popular game. It would be interesting to develop pruning strategies for probabilistic game evaluation.

In view of the drawbacks, we have proposed a heuristic strategy that simply takes a vote from independent iteratively deepening alpha-beta evaluations of the game tree. The procedure then selects the move recommended by a (weighted) majority of the evaluations. This procedure performs well in our experiments and, in fact, performs better than any of the separate evaluations from which it is computed.

References

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

Fischer, M.J. and Paleologou, S. A. 1991. Decision making in the presence of noise. Technical Report YALEU/DCS/TR-875, Yale University.

Horvitz, E. J. 1988. Reasoning under varying and uncertain resource constraints. In *Proceedings of the Seventh National Conference on Artificial Intelligence*. 111–116.

Nau, D. S. 1980. Pathology on game trees: Summary of results. In *Proceedings of AAAI-80*. 102–104.

Nau, D. S. 1983. Pathology on game trees revisited and an alternative to minimaxing. *Artificial Intelligence* 221–244.

Palay, A. 1983. *Searching with Probabilities*. Ph.D. Dissertation, Carnegie-Mellon University.

Pearl, J. 1984. *Heuristics*. Addison-Wesley.

Pearl, J. 1988. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann.

Russell, S. J. and Wefald, E. H. 1989. On optimal game-tree search using rational meta-reasoning. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*. 334–340.

Savage, L. J. 1972. *The Foundations of Statistics*. Dover.

von Neumann, J. and Morgenstern, O. 1944. *Theories of Games and Ecomomic Behavior*. Princeton University Press.