A Probabilistic Approach for Reducing the Search Cost in Binary Decision Trees

by

Athanasios Rontogiannis
Ptychion, National Technical University of Athens, 1991

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF APPLIED SCIENCE

in the Department of Electrical and Computer Engineering

We accept this thesis as conforming
to the required standard

[redacted]

Dr. N.J. Dimopoulos, Supervisor (Department of Electrical and Computer Engineering)

[redacted]

Dr. K.F. Li, Departmental Member (Department of Electrical and Computer Engineering)

[redacted]

Dr. Z. Dong, Outside Member (Department of Mechanical Engineering)

[redacted]

Dr. R.N. Horspool, External Examiner (Department of Computer Science)

Supervisor: Dr. N.J. Dimopoulos

Abstract

In many complex problems a particular decisionmaking procedure is often required in order for a final solution to be found. Such a procedure may consist of a large number of intermediate steps where "local" decisions must be taken and can be sometimes represented as a decision tree. When that structure is used the final solutions obtained vary depending on the available information. However, if the same model is applied many times, experimental data can be collected and observations on the acquired knowledge can be made. In this work, we present a probabilistic approach for reducing the number of decisions (tests) that are required in a particular decisionmaking situation. Specifically, we consider that a problem is structured as a decision binary balanced tree the interior nodes of which correspond to decision points; the paths of the tree represent different decisionmaking processes. By assuming that there exists sufficient probabilistic information concerning the decisions at the decision nodes, we attempt to minimize the average number of these decicions when we search for a final solution.

Examiners:

[redacted]

Dr. N.J. Dimopoulos, Supervisor (Department of Electrical and Computer Engineering)

[redacted]

Dr. K.F. Li, Departmental Member (Department of Electrical and Computer Engineering)

[redacted]

Dr. Z. Dong, Outside Member (Department of Mechanical Engineering)

[redacted]

Dr. R.N. Horspool, External Examiner (Department of Computer Science)

Table of Contents

List of Tables

Table of Figures

Acknowledgements

I am deeply indebted to my supervisor, Dr. Nikitas J. Dimopoulos, for guiding this research and providing a congenial work environment. Without his contribution this thesis would not have been completed. Special thanks must be addressed to the Canadian Cable Labs Fund for supporting this work. I would also like to thank my brother Panagiotis for endless discussions and moral support. Last but not least, I would like to express my gratitude to my parents for their encouragement, patience and love. This thesis is dedicated to them as a small recognition of their sacrifices and understanding.

Στους γονεις μου

Chapter 1

Introduction

Uncertainty is a notion which is very often present in many intelligent systems, that is, systems which are capable of making reasonable judgments about specific and complex problems. This is due to the fact that reasoning in some domains cannot be done with absolute certainty because there exists only partial information concerning some components of the problem under consideration. Especially in decision making situations, in which a series of decisions must be taken before a final conclusion is reached, uncertainty constitutes a very significant part of the decision procedure.

Many approaches have been developed attempting to quantify uncertainty in intelligent systems. The most rigorous and commonly used one is the probability theory which provides a complete framework for expressing and manipulating degrees of uncertainty. Particularly, in decision theory applications, probability theory appears to be the main tool for dealing with uncertain or unpredictable situations and events. In such applications the main concern of the human expert is

to follow a certain decision making process which will lead him to the correct final conclusion or action. A particular procedure may consist of many intermediate steps where "local" decisions must be made. The whole task proceeds according to the decisions taken at each and every intermediate step. A decision making procedure may be diagnostic, classifying different situations according to available information. Consider, for example, the case of a patient who has some symptoms. The physician asks the patient about his symptoms and other related information and then based on that information and on his knowledge of the domain ends up with a specific diagnosis and proposes the required treatment for the diagnosed disease. In this example before reaching a conclusion, the expert follows a diagnostic procedure which is determined by the information that is available to him. For different data, different final diagnoses may result. Furthermore, some of the parameters that are required may not be clear or there may be only partial knowledge of them. Therefore, the physician must consider similar cases in the past and propose a treatment despite the uncertainties that are present.

In situations like the above, the given problem can often be structured as a decision tree the nodes of which represent decision points; the paths correspond to different decision making processes. Most of the times, the main purpose of the decision maker is to select and follow the correct path despite the uncertainties that often appear. By correct path we mean the one which ends up with a correct final decision (solution).

After structuring the problem under consideration in an appropriate and clear way, the same model can be applied every time that a similar case arises. By using that model many times, experimental data can be collected and observations on the acquired knowledge can be made. We may observe, for example, that some diagnoses appear more frequently than others or that the outcomes of some "local" decisions are known with a high probability. Thus, all this new knowledge

can be taken into account in order for the diagnostic procedure to be simplified and facilitated.

The last principle is the main notion underlying our work. Specifically, we consider balanced binary decision trees and assume that there exists sufficient probabilistic (statistical) information available concerning the decisions at the interior nodes. This information may come from experimental data collected by using the tree structure many times as a diagnostic tool. Our basic goal is to develop a model in which, relying on the given information, we reduce the number of decisions that is required when we search for a final solution. That can be achieved by selecting the nodes at which an explicit decision is not taken but instead the branch of the tree which is to be followed is determined probabilistically. Such an approach has the effect that we may sometimes end up with wrong final diagnoses.We can then search for the correct path by appropriately backtracking into the tree. In order to achieve our goal, we build up a gain function and try to maximize it. That function depends not only on the selection of the nodes mentioned before but also on the choice of a proper backtracking procedure, and expresses the average number of saved decisions in a single traversal of the tree.

Our work is organized into the following chapters

- In **chapter2** the idea of using the probability theory for quantifying uncertainty is emphasized. The main approaches to probability are highlighted and the basic concepts of probability theory are described.
- In **chapter 3** the main motivation of our work is presented and topics that appear in the literature and are related to our model, are described.
- In **chapter 4** we present the main assumptions that are required in order for our mathematical model to be built up. Based on these assumptions, we then formulate the general expression of the gain function.

- In **chapter 5** the notion of the backtracking cost is introduced and its expression for a single path of the tree is derived. Then, different backtracking approaches are discussed and the procedure for obtaining the one with the minimum cost is described.

- In **chapter 6** we propose some heuristic techniques for selecting the nodes of the tree at which a probabilistic decision is taken. The initiative of these techniques is presented and their results are compared to each other for different tree structures.

- In **chapter 7** we summarize our work, present some conclusions and give suggestions for future research.

Chapter 2

Probability Theory and Bayesian Inference

Probability is a means of quantifying the uncertainty about facts and events occurring in our everyday life. It expresses the degree of belief, concerning the truth or falsity of those events by attaching to them specific numerical values. Since its appearance in the seventeenth century, probability theory has always been a subject of continuous debate about its axiomatic foundation and definition. Different approaches have been developed trying to investigate its actual meaning. Criticisms often arise for each of the approaches that have already appeared and different "schools" exist.

Despite all the above, probability theory remains a very powerful framework for modeling uncertainty. Its application in many fields of human activity , such as business, economics, physical sciences etc., proves this fact. In particular, probability theory is a very important component of the formal theory of decision making under conditions of uncertainty. People must often take decisions or face problems when some of the parameters are not completely known. Instead, numerical values representing probabilities can be attached to those parameters.

The mathematical probability theory contains specific rules for manipulating these numerical values. All those rules constitute the probability calculus.

In the rest of this chapter the main approaches to probability theory are presented. Criticisms for each approach are emphasized. Then, the basic concepts of the probability calculus are described. Special attention is given to Bayesian inference, which is a very important and effective tool for updating degrees of belief under the appearance of new evidence. The chapter concludes with a brief discussion of the notion of a probabilistic model.

2.1 The Main Approaches to Probability

The main approaches to probability in chronological order are

- The Classical Approach.
- The Limiting Frequency Approach.
- The Subjectivistic Approach.

In this section a brief description of all the three of them is given. For a more detailed presentation see also [1]-[6].

2.1.1 The Classical Approach

Probability theory has its origin in gambling games and therefore the classical definition of probability is relative to such situations. When we roll a die, for example, the possible outcome is one of the numbers : 1,2,3,4,5,6. The set of all possible outcomes of an experiment is called the *sample space* and is often denoted by the symbol Ω. Thus, in the case of rolling a die $\Omega = \{1, 2, 3, 4, 5, 6\}$. Every subset of Ω is called an *event*. For instance, the subset $A = \{1, 3, 5\}$ represents the event of an odd outcome after rolling the die. The classical definition of the probability of an event A is due to Laplace and can be expressed as follows (see also [3])

Definition 2.1. *If there are α possible outcomes favorable to the occurrence of an event A and β possible outcomes unfavorable to the occurence of A and all these outcomes are equally likely and mutually exclusive, then the probability that A will occur, denoted $P(A)$ is*

$$P(A) = \frac{\alpha}{\alpha + \beta} \qquad \text{(EQ 2.1)}$$

In the last definition, the assumption of equal likelihood about the outcomes of the experiment means that there is nothing that makes us believe the domination of anyone of the outcomes over the others. A formal definition of "mutually exclusive" outcomes will be given later when we discuss the principles of probability theory.

The main disadvantage of the classical definition is that it assumes a finite sample space. However, this is not the case in many situations where we have to deal with experiments with infinite sample spaces. For example the classical approach cannot give the probability that an item manufactured through a specific production process will be defective. Such situations require the use of empirical data. This fact led to the development of the limiting frequency approach.

2.1.2 The Limiting Frequency Approach

The limiting frequency approach to probability was first established by Von Mises at the beginning of this century. According to Von Mises, probability only has meaning in the case of an experiment that can be repeated ([1]). He defines the probability of an event A as follows

Definition 2.2. *If an experiment is repeated n times under the same conditions and $S^n(A)$ is the number of times that event A occurred, the probability of A is defined as*

$$P(A) = \lim_{n \to \infty} \frac{S^n(A)}{n} \qquad \text{(EQ 2.2)}$$

Von Mises believed that if the experiment is repeated infinitely the limit in (Eq. 2.2) is a specific number which represents the probability of the event A. The problem with the last definition is that there does not exist any mathematical proof that the limit in (Eq. 2.2) actually exists. Only after collecting enough statistical data can the above convergence be revealed empirically. Furthermore, the assumption that the experiment is repeated under the same conditions is not feasible. Instead, there is not any guarantee that the conditions will remain constant during the infinite executions of the experiment. For these reasons, the limiting frequency approach has often been the subject of intense criticism. Nevertheless, that approach is extensively used in many applications nowadays. In particular, when *a priori* statistical information is available, the frequentistic approach can be effectively used for assigning probabilities to events that may occur in the future.

2.1.3 The Subjectivistic Approach

Both of the approaches that we have already discussed cannot handle many situations taking place in our everyday life. We often say : "It is probable that it will rain tonight" or "I will probably meet him there". Such statements are very commonly used and can not be considered as part of an experiment that can be repeated. The development of the subjectivistic approach to probability attempts to deal with such situations.

The subjective concept of probability is relatively recent. The founder of the subjectivistic approach is considered to be De Finneti (1972). According to him, probability is the degree of belief (or degree of uncertainty) that an individual assigns to the statement that a particular event will occur, based on evidence available to him. This evidence may come either from personal experience or from data that have already been collected. A very important point is that the degrees of belief that the individual attaches to different facts, must be consistent with the axioms and laws of the probability theory which will be discussed in the next section. As it is obvious, different people will assign different probabilities

to the same facts because of variations in experience, available data, etc. Thus, in contrast to the approaches presented in the two previous sections in which the numerical values that correspond to probabilities of events are objectively acceptable, a subjective probability represents personal degree of belief or degree of confidence assigned to those events.

2.2 Basic Concepts of the Probability Theory

In a given experiment we denote by S the certain event i.e., the event that occurs in every trial. Given two events A and B, we denote by $A \vee B$ the event that occurs when A and B or both occur. Regardless of the approach that one accepts, the probabilities assigned to events must obey and must be consistent with the three basic axioms of the probability theory ([2])

1. $P(A)$ is positive :

$$P(A) \geq 0$$

2. The probability of the certain event equals 1 :

$$P(S) = 1$$

3. If A and B are mutually exclusive events then

$$P(A \vee B) = P(A) + P(B)$$

Definition 2.3. *The events* $E_1, E_2, \ldots, E_n$ *are called mutually exclusive when the occurrence of anyone of them makes the occurrence of the others impossible.*

The third axiom given above can be generalized as follows

$$P(E_1 \vee E_2 \vee \ldots \vee E_n) = P(E_1) + P(E_2) + \ldots + P(E_n) \qquad \text{(EQ 2.3)}$$

when $E_1, E_2, \ldots, E_n$ are mutually exclusive.

Definition 2.4. *The events* $E_1, E_2, \ldots, E_n$ *are called exhaustive when*

$$P(E_1 \vee E_2 \vee \ldots \vee E_n) = 1 \qquad \text{(EQ 2.4)}$$

We have to emphasize that the meaning of the probability $P(A)$ of an event A is not absolute. Instead, the probability only exists relative to partial information or knowledge. When that knowledge remains constant, we do not need to specify its existence explicitly in the notation $P(A)$. On the other side, when the available information undergoes changes, the probability of an event A must be declared relative to the new information at hand. We use the symbolism $P(A|e)$ to express the probability of A after obtaining evidence e from the new data. More specifically, when an event B is known with absolute certainty, we define the conditional probability of A given the occurrence of B as follows

$$P(A|B) = \frac{P(A,B)}{P(B)} \qquad \text{(EQ 2.5)}$$

In the last equation, $P(A,B)$ is the joint probability of the events A and B, that is, the probability that both A and B occur. From (Eq. 2.5) we get

$$P(A,B) = P(A|B) \cdot P(B) \qquad \text{(EQ 2.6)}$$

The last expression is known as the *product* (or *multiplication*) *rule* of the probability theory. When $P(A|B) = P(A)$ the events A and B are called *independent*. In such a case (Eq. 2.6) can be written as follows

$$P(A,B) = P(A) \cdot P(B)$$

A very useful generalization of the product rule given in (Eq. 2.6) is the so-called *chain rule* formula ([6]). According to that, the joint probability of n events $(E_1, E_2, \ldots, E_n)$ can be expressed as a product of n conditional probabilities

$$P(E_1, E_2, \ldots, E_n) = P(E_n|E_{n-1},\ldots,E_1)\ldots P(E_2|E_1)P(E_1) \qquad \text{(EQ 2.7)}$$

This formula can be derived by repeated application of (Eq. 2.6) in any convenient order and it will be widely used in the chapters to follow.

2.3 Bayesian Inference and Probabilistic Models

Bayesian inference is a very effective framework for updating probabilities of hypotheses in the context of new evidence. Bayes attempted to calculate probabilities of events based on observations about their consequences. For example, in the case of a patient, we can observe particular symptoms. From these symptoms and available data which have been collected from similar cases in the past, the probability that the patient has a specific disease can be calculated. In this example, the symptoms are the consequences of the observed fact, which is the disease. For the manipulation of such situations, the well-known *inversion formula* has been developed. The inversion formula is the base of the Bayesian inference and can be written as follows

$$P(H|e) = \frac{P(e|H) \cdot P(H)}{P(e)} \qquad \text{(EQ 2.8)}$$

where H is the hypothesis and e is the new evidence. The probability $P(H)$ is often called *a priori* (or *prior*) probability of the hypothesis H. $P(H|e)$ is called *a posteriori* (or *posterior*) probability of H. We note that when new information is again available, the probability $P(H|e)$ becomes a priori probability and is used in the right-hand side of (Eq. 2.8) for the calculation of the new posterior probability of H. This way, the degree of belief that we assign to the hypothesis H is updated every time new data are received.

All the statements discussed so far in this chapter are meaningful in the context of a probabilistic model, which can be defined as follows ([6])

Definition 2.5. *A probabilistic model is an encoding of probabilistic information that permits us to compute the probability of every well-formed sentence F in accordance to the three basic axioms of the probability theory.*

In general, starting with a set of atomic propositions, a well-formed sentence is constructed as a conjunction of atomic propositions or their negations. For instance, if A, B, C and D are atomic events, then $F = A \wedge B \wedge \neg C \wedge D$ is a well-

formed sentence. In a probabilistic model each sentence is assigned a probability which either has its origin in a collective (frequentistic approach) or represents the degree of belief of an expert (subjectivistic approach). In order for consistency to be kept, the probabilities of all possible well-formed formulae must add up to 1, that is, they must constitute a set of exhaustive events. Then, by applying some appropriate relation of the probability calculus (for example (Eq. 2.7) or (Eq. 2.8)) the probability of any other formula can be computed. This way, the joint probability of some atomic propositions given the occurrence of other propositions (such as $P(\neg A, C|B, D)$ in the above example), can be calculated.

Summarizing, in a specific problem a probabilistic model must first be determined and constructed. The sufficiency of our information and the consistency of our model can be verified by checking if the probabilities of all the possible well-formed formulas add up to 1. Finally, the probability of any other event can be derived by using the laws and theorems of the probability calculus.

Chapter 3

Motivation and Related Work

People often cope with problems in which they have to make decisions under conditions of uncertainty. This is the case because some components of the specific problem are only partialy known or the cost associated with obtaining all the necessary information is high. When the decision maker faces such a situation, he must base his decisions and conclusions on his experience or on available information. His main purpose is, of course, to find a solution to the given problem. However, when such an outcome is not always feasible because of the uncertainties that often appear, the decision maker attempts to maximize an expected utility which is a function of the problem parameters. The whole procedure can be facilitated when it is properly structured in a way that makes clear the particular of the problem. Decision trees constitute an effective representation for modeling some aspects of human reasoning. Different paths of a decision tree express different policies that can be followed in order for a solution to be found. Probability theory and statistics ([7]-[10]) provide a range of tools by means of which

predictions can be made based on the assumption that future events will fit the predefined model. In our work we are using the theorems of statistical and probability theory in order to minimize the number of decisions taken by the expert or the decision maker when the problem is structured as a decision binary tree.

In the next section we give a brief introduction to the main methods for dealing with uncertainty that can be found in various intelligent systems. The idea of using the probability theory for quantifying uncertainty in human reasoning is especially emphasized. Then, two basic structures for representing uncertainty, namely belief networks and decision trees, are introduced. Two different approaches of the application of decision trees that appear in the literature are presented. The chapter concludes with a discussion of the main motivation of our work and its interrelations with the concepts described in the previous sections of the chapter.

3.1 Quantification of Human Uncertainty

Many theories and methods have been developed so far trying to model human reasoning under conditions of uncertainty. The most important and most commonly used one is probability theory. In this section a brief presentation of the notion of probability theory is given. A more detailed discussion appeared already in the previous chapter. The other basic methods are also briefly described.

3.1.1 Probability Theory

Probability theory assigns numerical values to events and facts and then manipulates these numbers according to the theorems of probability calculus. Even though people do not attach numerical values to events because they cannot remember the exact frequencies of them in the past, "they learn to think in terms of dependencies and independencies which are implied by those frequencies" ([6]). For example when we say : "It is very probable that Tom will be in the class

tomorrow", we attach a high probability to the aforementioned statement. Our belief is based on previous experience and may be dependent on other related events. Thus, if we know that the next day is a deadline for an assignment our belief (or confidence) that Tom will show up in the class is strengthened.

Probability theory provides a complete framework in which degrees of belief can be expressed numerically and updated when new information is received. Dependencies and non-dependencies can also be clearly represented by using conditional probabilities. Therefore, it seems that the language of uncertainty is probability. Furthermore, the literature arguing in favor of direct probabilistic quantification of uncertainty is vast. For these reasons and because of the particularities of our problem, probability theory has been selected and extensively used in our work.

3.1.2 Other Methods for Reasoning under Uncertainty

Some other approaches for dealing with uncertainty are the certainty factors theory, the Dempster-Shafer theory of evidence and fuzzy set theory ([1], [6],[11]).

The certainty factors approach appeared for the first time in the expert system MYCIN ([12]). MYCIN is a rule-based expert system for diagnosing bacterial infections and prescribing treatments for them. MYCIN attaches to its rules numerical values, called certainty factors. Each rule is assigned a value between -1 and 1 where a positive value indicates that the verity of the premises will increase our belief in the conclusion while a negative value indicates that the verity of the premises will decrease our belief in the conclusion. Several regulation principles have then been developed for obtaining and manipulating certainty factors.

The Dempster-Shafer theory of evidence ([13]) can be considered as an extension of the probability theory in the cases where we are unable to determine the probability values needed to obtain the probabilities of propositions. This theory introduces the notion of the belief interval, that is, for the probability of each

proposition a lower and an upper bound is determined. Ways for manipulating belief intervals are then proposed in order for the belief intervals of complex propositions to be obtained.

Fuzzy set theory ([14]-[17]), in contrast to the approaches discussed so far, deals with propositions which have vague meaning. Vague concepts can not be handled according to the classical two-valued logic. The key idea in the fuzzy set theory is that an element has a degree of membership in a fuzzy set. Thus, a proposition need not only be true or false but may be partly true to any degree. We usually assume that this degree is a real number in the interval [0, 1].

3.2 Structuring of Human Uncertainty

As it has been stated in the introduction of this chapter, proper structuring and representation of a problem makes the process of solving it easier. Various structures have been proposed for problems which contain uncertainty. We will present two of them, namely, belief networks and decision trees.

3.2.1 Belief Networks

A *belief* (or *causal* or *bayesian*) network ([1],[6],[18]) is a directed acyclic graph (DAG). The nodes of the graph represent propositions (or variables) which can be either true or false (even though they are not necessarily two-valued) , and the edges between nodes express the dependencies or interrelations between the linked propositions. The strengths of these dependencies are quantified by conditional probabilities. Pearl in [19] describes a procedure in which probabilities can be propagated in a causal network when one or more of the variables are instantiated (that is, when these variables are known to be true or false with absolute certainty). This way the probabilities of the network can change in the context of new information. On the other hand, a causal network can not explicitly address a method for using these probabilities to make a decision. That can be done by using a decision tree structure.

3.2.2 Decision Trees

A decision tree is a representation which is widely used when an explicit decision is required although our knowledge about many components of the problem under consideration may not be complete. Two distinct applications of the decision tree structure are discussed in the next sections.

3.2.2.1 Decision Trees for Maximizing an Expected Utility ([1], [3], [6])

Decision making under conditions of uncertainty does not guarantee that the decision taken is the best one. That is because some events lie outside the control of the decision maker who does not know which event will occur. Such events are often called "states of nature" or "states of the world" because their existence is governed by external factors which the decision maker ignores. Therefore, what is left to the human expert is to base his conclusions on his experience and try to maximize an expected utility function which in many cases is cost related. Decision trees can be used for the structuring of such problems. For instance consider the example tree of figure 3.1. In that tree there exist two kinds of nodes : *decision nodes* (square in the figure) where the decision maker is in control of choice and *chance nodes* (shaded circles) at which Chance is in control. The possible actions that can be taken by the decision maker are denoted by the symbols A_1 and A_2 while θ_i, $i = 1, 2, 3$, represent the states of nature and p_i are the corresponding probabilities of these events. The symbols m_i, $i = 1, 2, \ldots, 6$ stand for different costs. For example, m_2 is the expected profit (gain) or cost when the decision maker selects the action A_1 and the event θ_2 takes place. In general, if the decision maker selects the action A_1 the expected utility is

$$Utility = p_1 \cdot m_1 + p_2 \cdot m_2 + p_3 \cdot m_3$$

If the action A_2 is selected the expected utility is

$$Utility = p_1 \cdot m_4 + p_2 \cdot m_5 + p_3 \cdot m_6$$

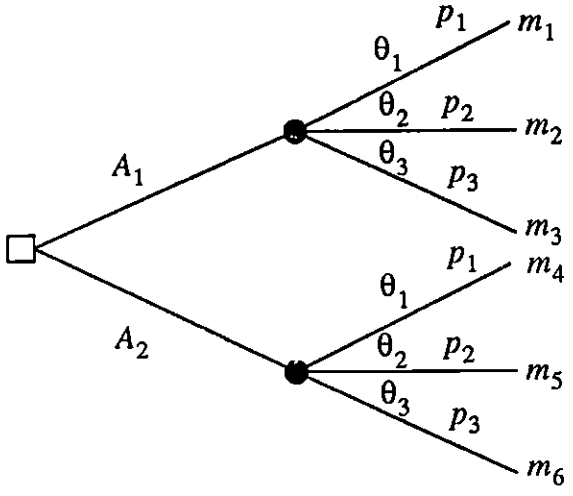

FIGURE 3.1. Example Decision Tree

The action which is finally chosen is the one which maximizes the expected utility given before.

In more complicated problems the decision trees are larger and the problem can be depicted in terms of a series of choices in alternating order by the decision maker and chance.

3.2.2.2 Decision Trees as Probabilistic Classifiers ([20]-[22])

J.R.Quinlan gives the following definition of a decision tree in [20]

Definition 3.1. *A decision tree is a recursive structure for expressing classification rules. Such a tree may be a leaf associated with one class. Alternatively, the tree may consist of a test that has a set of mutually exclusive possible outcomes together with a subsidiary decision tree for each such outcome.*

According to that definition, the leaves of the tree constitute a disjoint set of classes while at each interior node the value of a particular attribute is checked. Such an action can be called an *attribute test.* For example, a decision tree that classifies days as *Play* or *Don't Play* might be represented as shown in figure 3.2 ([20])

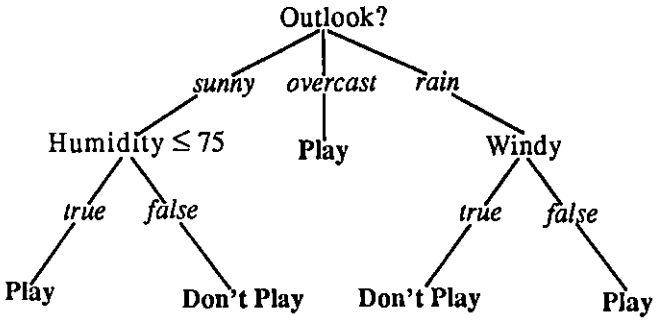

FIGURE 3.2. Decision Tree

In the previously mentioned paper Quinlan discusses how a decision tree can be constructed from a set of initial *objects* often called a *training set*. The objects are distinguished through the values of various attributes. When the construction of the tree has been completed and required simplifications have been made, the problem of classifying a new object arises. In particular when some attribute values of the new object are unknown or the given data are noisy, the object may be assigned to more than one class or may be misclassified. The latter is possible even if all the attribute values are known because of simplifications that have been made to the tree (i.e. pruning).

In [23] the author presents a probabilistic model for dealing with unknown attribute values. According to that, during the construction of the decision tree from the training set, statistics are collected for each attribute test concerning the frequency of appearance of the test outcomes. This way an estimate of the probability of each outcome can be made. Thus, when a new object has some attribute values unknown, it belongs to two or more classes with weights corresponding to probabilities. In other words, the higher the weight is, the more probable is that the object belongs to a specific class. It is then left to the human expert to decide which class will be assigned to the new object.

3.3 Motivation

Cable television distribution networks are widely used nowadays to distribute cable signals from a central site to subscribers' homes. Such networks consist of a large number of amplifiers interconnected with coaxial cable. As the signals propagate into the cable, they deteriorate and therefore amplification of the signals is required. This is achieved through the interconnection of a number of amplifiers into the network. Cable television technology is presented in detail in [24].

A typical cable network consists of several hundred to a few thousand amplifiers covering a metropolitan area. Failures, caused by various factors, sometimes occur in the network and result in bad signal quality received by the subscribers. In particular the failure of an amplifier affects many other neighboring ones making the identification of the failing amplifier a difficult task. A prototype diagnostic environment for Rogers' Victoria Cable Network is described in [25] and [26]. When the amplifier which malfunctions is found, it is normally replaced by a functioning one. The malfunctioning amplifier is then returned to the shop or the manufacturer for further tests and repair.

The amplifier's technology used in a typical cable network is presented in [27] where the possible failures of the amplifiers and the diagnosis of their causes are also discussed. For the latter, some diagnostic flowcharts have been constructed. Each flowchart corresponds to a specific failure of an amplifier and represents the steps that must be followed by the technician in order to identify the problem. Such a flowchart, for example, is given in figure 3.3. From that figure we observe that during the diagnostic procedure the values of some attributes are tested. These tests take place at the decision boxes of the flowchart (diamond boxes) and there exist only two possible outcomes : either an attribute value is verified or not. The similarities with the decision tree structure presented in section 3.2.2.2 are obvious. Indeed, the flowchart of figure 3.3 can be converted to the decision tree of figure 3.4. In that figure the nodes of the tree correspond to the decision

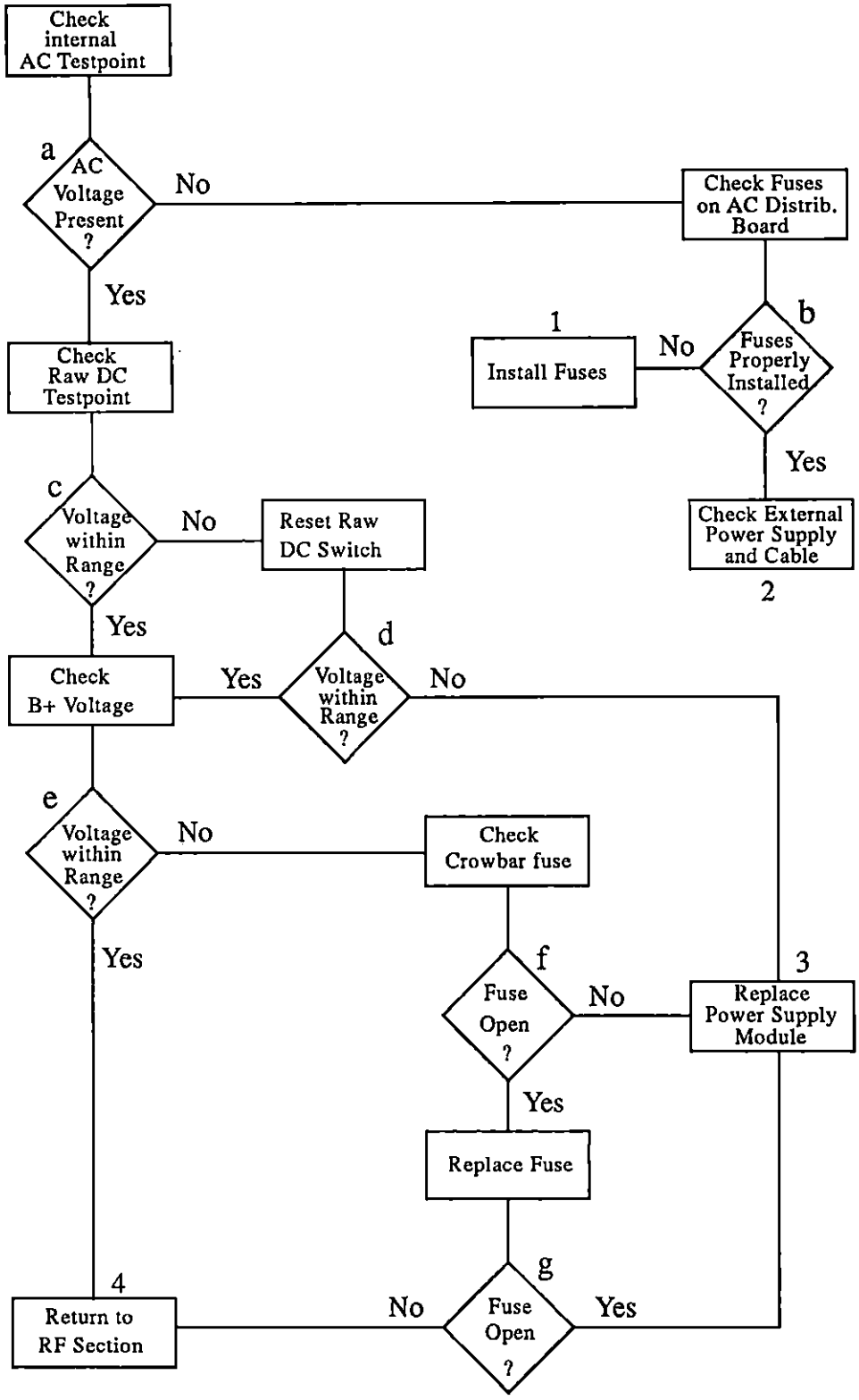

FIGURE 3.3. Diagnostic Flowchart

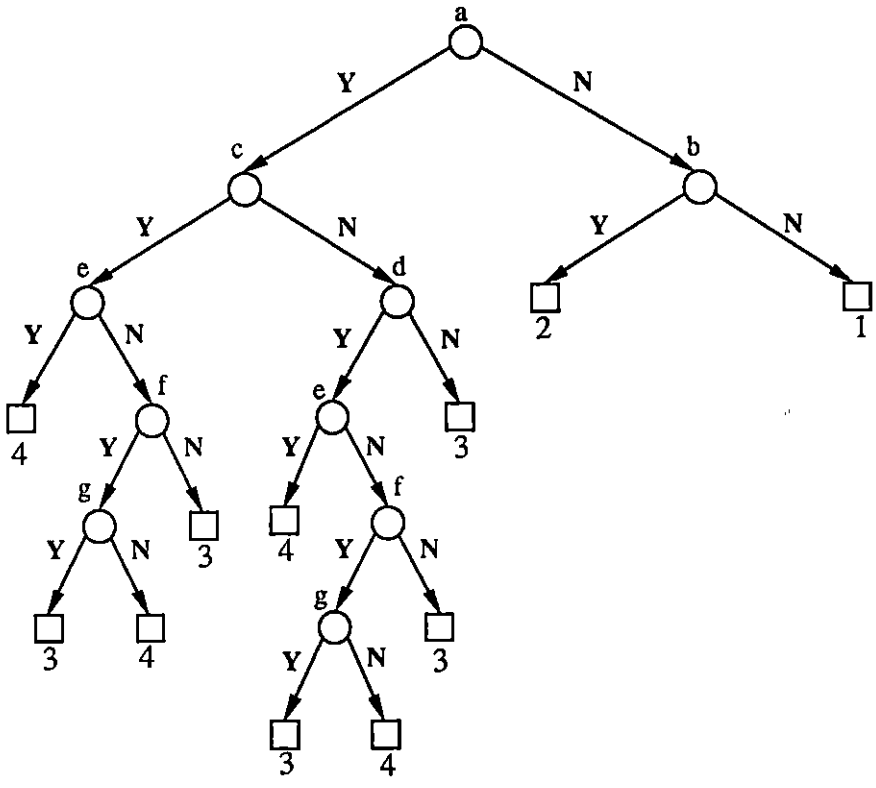

FIGURE 3.4. Decision Tree Representation

boxes of the flowchart as denoted by the attached letters. The leaves of the tree represent the final diagnoses as denoted by the attached numbers in figures 3.3 and 3.4. Observe that a decision box in the flowchart may correspond to more than one decision nodes (as for example decision box *e* in figure 3.3). This is so because more than one path may connect the root node (node *a* in this example) with a specific decision box.

In our work, binary decision trees are considered. The interior nodes of these trees represent attribute tests with two possible outcomes while the leaves constitute a set of distinct final diagnoses. In other words, each path in a tree corresponds to a different diagnostic procedure.

We assume that enough information is available so that we are able to assign two numerical values to each of the interior nodes of the tree which quantify the probabilities of the two possible attribute test outcomes. That information comes

either from statistical data, concerning the specific tree, that have been collected in the past or from the human expert's experience.

When the problem of diagnosing a new case arises, we assume that misclassification is not possible if all the attribute values of the new "object" are known. This means that, if we traverse the tree and there is an explicit answer to each attribute test that we meet then the resulting diagnosis is certainly correct.

The aim of our work is, based on probabilistic information, to minimize the number of attribute tests which take place when we go through a specific decision tree. That can be achieved by properly partitioning the interior nodes of the tree into nodes where the test occurs and nodes where it does not. Different methods for such a partitioning are proposed. In our model, we make the assumption that after a wrong diagnosis we are able to backtrack and search for the correct one. Thus, different backtracking approaches are also presented attempting to minimize our backtracking cost.

From the discussion so far, we observe that in our analysis we make use of the maximization concept discussed in section 3.2.2.1 while retaining the decision tree structure of section 3.2.2.2. In our case the utility that must be maximized is the average number of nodes in the tree where an attribute test was not performed and yet the correct decision was reached. In order to simplify our mathematical model, balanced trees are considered in the chapters to follow.

Chapter 4

Formulation of the Gain Function

In this chapter the problem that we are going to deal with, is presented. The assumptions that are required in order for our mathematical model to be built are emphasized and their correspondence to real-world situations is discussed. Based on these assumptions, a gain function is constructed and its interpretation is analysed. The purpose of this work is the maximization of that gain function. The maximization procedure is the subject of the next two chapters.

4.1 Presentation of the Problem

We consider a complete binary balanced tree of depth n. We assume that the interior nodes of the tree are decision nodes at which an attribute test takes place as discussed in section (3.2.2). There exist only two possible outcomes for each attribute test : either the value of the attribute is verified or not. In other words, we can assume that each interior node of the tree contains a question with only two possible answers : "yes" or "no". The answer "yes" can be arbitrarily assigned to the left "child" and the answer "no" to the right "child" of a particular

node. The leaves of the tree constitute a set of distinct final diagnoses to the problem that is represented by the given tree structure.

Each of the interior nodes of the tree is attached two numerical values which express the probabilities of occurrence of the two test outcomes. These probabilities can be calculated by applying either the frequentistic approach or the subjectivistic approach, presented in section (2.1). In the former case we must assume that the given tree has been used many times in the past and statistical information has been collected for each of the tree nodes. More specifically, let us consider an arbitrary interior node k. If N_k is the number of times we reached this node and N_{ky}, N_{kn} are the number of times we had a positive and a negative answer to the question of the node respectively, we have the following equation

$$N_k = N_{ky} + N_{kn} \qquad \text{(EQ 4.1)}$$

According to Von Mises' definition of probability, if the number N_k is sufficiently high so that the limit of (Eq 2.2) can be approximated, we can assign to the node k the following probabilities (see also figure 4.1)

$$P_{ky} = Pr[YES] = \lim_{N_k \to \infty} \frac{N_{ky}}{N_k} \qquad \text{(EQ 4.2)}$$

$$P_{kn} = Pr[NO] = \lim_{N_k \to \infty} \frac{N_{kn}}{N_k} \qquad \text{(EQ 4.3)}$$

From equations (4.1), (4.2) and (4.3) we obtain

$$P_{ky} + P_{kn} = 1 \qquad \text{(EQ 4.4)}$$

From the last equation we have

$$max\,(P_{ky}, P_{kn}) \geq 0.5 \qquad \text{(INQ 4.5)}$$

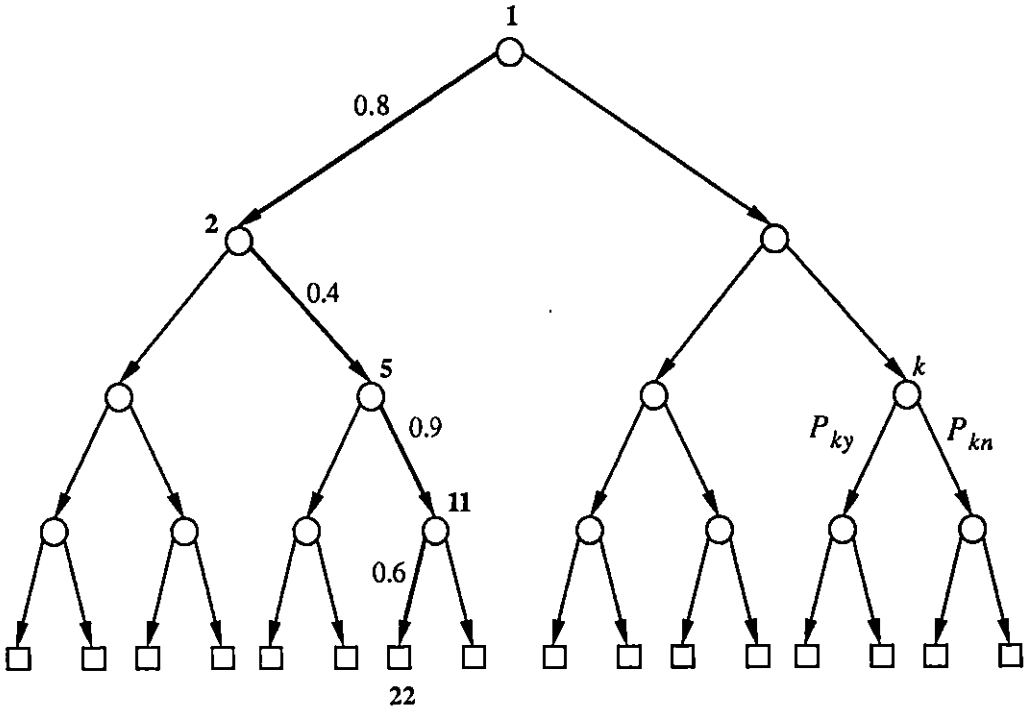

FIGURE 4.1. Example Tree

On the other hand, according to the subjectivistic approach, the human expert based on his experience assigns the probabilities P_{ky} and P_{kn} to node k so that (Eq 4.4) is satisfied. Regardless of the approach accepted, we make the assumption that the probabilities of the interior nodes of the tree are known. We have to emphasize that these probabilities are conditional probabilities. That is, P_{ky} is the probability that the answer in node k is positive given that we have followed the path that leads to this node. For example, in figure 4.1, P_{5n} can be written as

$$P_{5n} = P(E|E_1, E_2) \qquad \text{(EQ 4.6)}$$

where E is the statement that the answer at the node 5 is "no" and E_1, E_2 are the facts that the answers at the nodes 1 and 2 are "yes" and "no" respectively.

The probabilities of the interior nodes of the tree sometimes carry very important information. For instance, if $P_{ky} = 0.9$, our belief is very strong that the outcome of the test at the node k is positive. Thus, when we reach that node, we may decide not to perform any test but instead, we follow the left branch since such a decision is correct with a very high probability. By avoiding a test we have a gain whose value depends on the importance of the test under consideration. In our analysis, all the tests are assumed to have the same significance and therefore we can assign a weight of 1 to each of them.The aim of our work is to partition the interior nodes of the tree into nodes where the test occurs and nodes where it does not, in such a manner that the gain is maximized. In other words, we will attempt to maximize the number of nodes at which probabilistic information is used in order for a decision to be made when we go through the tree.

By partitioning the interior nodes as it was described above, we may sometimes end up with wrong conclusions (final nodes). That is possible because the decisions which are based on probabilistic information may not be correct, resulting in wrong final nodes. In such a case we assume that we are able to backtrack and search for the correct path. We have to note that when a conclusion is reached by using our model, the truth of that conclusion must be verified. Therefore, there may be a cost associated with that process. In order to simplify our model we assume that we are able to distinguish between correct and wrong final conclusions without any cost. Summarizing, in order to build up our mathematical model we make the following assumptions

- The tree is binary and balanced.
- We know the probabilities of the interior nodes.
- All the tests contained in the interior nodes are of the same significance.
- The leaves of the tree constitute a set of distinct final conclusions.
- When we end up with a wrong path we can backtrack and search for the correct one.
- We are able to distinguish between correct and wrong final outcomes without any cost.

4.2 The Gain Function

In a complete binary tree of depth n there are 2^n leaf nodes. Each leaf corresponds to a unique path from the root node to that leaf node. In the following, when we refer to the path j we mean the path with final node j, $j = 2^n, 2^n + 1, \ldots, 2^{n+1} - 1$. For example in figure 4.1 the path which corresponds to the node $j = 22$ is (1,2,5,11,22). If we assume that we know the probabilities of the interior nodes of the tree we can prove the following

Proposition 4.1. *The prior probability that the arbitrary final node j is reached when we traverse the tree can be expressed as follows*

$$P_j = \prod_{i \in path(j)} P_{ir} \qquad \text{(EQ 4.7)}$$

where $r \in \{y, n\}$ *, that is* P_{ir} *is the probability of a positive or a negative answer at the node i which is included in path j.*

Proof. Let us assume that the path j consists of the branches $E_1, E_2, \ldots, E_n$ where E_1 is the first one and E_n is the last one starting from the root. If E_i stands also for the fact that the branch E_i is followed as we go through the tree, then P_j can be expressed as follows

$$P_j = P(E_1, E_2, \ldots, E_n)$$

But according to the chain rule formula (Eq. 2.7) we have

$$P_j = P(E_n | E_1, E_2, \ldots, E_{n-1}) \cdot \ldots \cdot P(E_2 | E_1) \cdot P(E_1)$$

The last expression is equivalent to (Eq. 4.7) as is clear from (Eq.4.6).

■

We call P_j the prior probability of the path j in order to distinguish it from the posterior probability of j which will be defined later in this chapter. According to (Eq. 4.7), in the example of figure 4.1 where $n = 4$ and $j = 22$ we have

$$P_{22} = P_{1y} \cdot P_{2n} \cdot P_{5n} \cdot P_{11y} = 0.1728$$

We note here that the final nodes of the tree constitute a set of mutually exclusive and exhaustive events. They are exclusive because we cannot follow more than one path at the same time. Furthermore, the probabilities of the final nodes add up to 1, that is

$$\sum_{j=2^n}^{2^{n+1}-1} P_j = 1 \qquad \text{(EQ 4.8)}$$

The last is true because we are going to follow, for sure, a path when we go through the tree. Therefore, the P_j's form a probabilistic model as was described in section 2.3. Based on that model the probabilities of other events will be calculated in the chapters to follow.

4.2.1 Gain in the Single-Path Case

Before we proceed to the derivation of the gain function, we give the following definition of a *statistical node*

Definition 4.1. *An interior node of the tree is called statistical when the test is not performed at that node but instead the decision is based on the probabilistic information that we have. The branch of the statistical node which is followed is called a statistical branch. All the other branches are called non-statistical.*

Each statistical node has one statistical and one non-statistical branch. Each non-statistical node has two non-statistical branches. Three methods for partitioning the interior nodes of the tree into statistical and non-statistical are presented in chapter 6. In order for the gain function to be constructed we assume in this chapter that such a partitioning already exists.

It was stated in section 4.1 that all the tests in the tree are equally significant and therefore we can assign to each of them a weight of 1. In general, we define the cost as the number of nodes where the question of the node is asked (the test is performed). As we traverse the tree we have two choices : either we use the sta-

tistical data that we have collected or we do not. In the latter case the cost will be expressed as

$$L_j = n \qquad \text{(EQ 4.9)}$$

Equation 4.9 holds because we are following a path by asking the questions in all the nodes of the path. Furthermore we are certain that the path is correct, that is , backtracking is not required. Let us now consider the former case. Assume that we have followed a path and we have reached a final node j. Let us also assume that m_j out of n nodes of the path are statistical nodes.This means that in m_j out of n nodes of the path (we do not consider the last node which is not a decision node), probabilistic information has been used in order for a decision to be made. We have two exclusive and exhaustive events : a) Path j that we followed is correct and b) Path j is wrong. If the probabilities of these two events are P_{Cj} and P_{Wj} respectively then

$$P_{Cj} + P_{Wj} = 1 \qquad \text{(EQ 4.10)}$$

The cost in the case of a correct path is given by

$$L_{Cj} = n - m_j \qquad \text{(EQ 4.11)}$$

As it was mentioned before, when we end up with a wrong path, we backtrack into the tree in order to find the correct path. Such a procedure results in a cost which will be called *backtracking cost* for the path j. Let B_j stand for the backtracking cost. Different approaches to the calculation of B_j are presented in chapter 5. The cost in the case of a wrong path is

$$L_{Wj} = n - m_j + B_j \qquad \text{(EQ 4.12)}$$

Assume, for example, that after a wrong final conclusion we back up to the first statistical node of the path and then we traverse the tree by asking all the questions at the nodes we meet. This way the correct conclusion is reached because the path from the root node to the first statistical node is certainly cor-

rect. If h_j is the height of that node where n is the height of the root and 0 the height of node j, L_{Wj} can be written as

$$L_{Wj} = n - m_j + h_j \qquad \text{(EQ 4.13)}$$

(Eq. 4.13) holds because h_j more tests take place after backtracking. If we now weight the costs L_{Cj} and L_{Wj} with the corresponding probabilities P_{Cj} and P_{Wj} we can express the cost of using the probabilistic information for the path j as follows

$$L_{Sj} = P_{Cj}L_{Cj} + P_{Wj}L_{Wj} = P_{Cj} \cdot (n - m_j) + (1 - P_{Cj}) \cdot (n - m_j + B_j) \qquad \text{(EQ 4.14)}$$

If we simplify (Eq 4.14) we end up with the following expression for L_{Sj}

$$L_{Sj} = (n - m_j) + B_j \cdot (1 - P_{Cj}) \qquad \text{(EQ 4.15)}$$

In equation 4.15, the probability P_{Cj} can be computed according to the following proposition

Proposition 4.2. *The probability P_{Cj} that the path j is correct, is the product of the probabilities of its statistical branches.*

Proof. Let's assume that the path j consists of the branches $E_1, E_2, \ldots, E_n$. If, in addition, E_i also stands for the fact that branch E_i is correct then P_{Cj} can be expressed as follows

$$P_{Cj} = P(E_1, E_2, \ldots, E_n)$$

According to the chain rule formula (Eq. 2.7) we have

$$P_{Cj} = P(E_n | E_1, \ldots, E_{n-1}) \cdot \ldots \cdot P(E_2 | E_1) \cdot P(E_1) \qquad \text{(EQ 4.16)}$$

But, if E_k is an arbitrary non-statistical branch we have

$$P(E_k|E_1,\ldots,E_{k-1}) = 1 \qquad \text{(EQ 4.17)}$$

because the test has been performed and the result is certainly correct. Thus, if we substitute (Eq. 4.17) in (Eq. 4.16) for each non-statistical branch we obtain

$$P_{Cj} = \prod P(E_l|E_1,\ldots,E_{l-1})$$

for all l such that E_l is a statistical branch. Therefore, P_{Cj} is expressed as the product of the probabilities of the statistical branches. ■

In the tree of figure 4.1, if the nodes 1 and 5 were statistical we would have

$$P_{C22} = 0.8 \cdot 0.9 = 0.72$$

From (Eq. 4.9) and (Eq. 4.15) we can calculate the gain in the case of a single path of the tree as follows

$$G_j = L_j - L_{Sj} = m_j - B_j \cdot (1 - P_{Cj}) \qquad \text{(EQ 4.18)}$$

We must note that the gain G_j can be negative. A negative gain indicates that the cost of using the statistical information is more than the cost of determining the path at each node.

4.2.2 The Tree Gain

In the analysis of section 4.2.1 a single path of the tree was considered. In fact, when we start traversing the tree we do not know which path we are going to follow. As it was mentioned before, an a priori probability P_j is assigned to the arbitrary path j. It was stated that P_j expresses the probability that the final node j is reached when we cross the tree and statistics are not taken into consideration. On the other hand, a partitioning of the interior nodes into statistical and non-statistical ones separates the set of the final nodes F into two disjoint sets : the set of reachable nodes F_1 and the set of non-reachable nodes F_2. This can become more obvious by considering the tree in figure 4.2. The tree has depth $n = 4$. The prob-

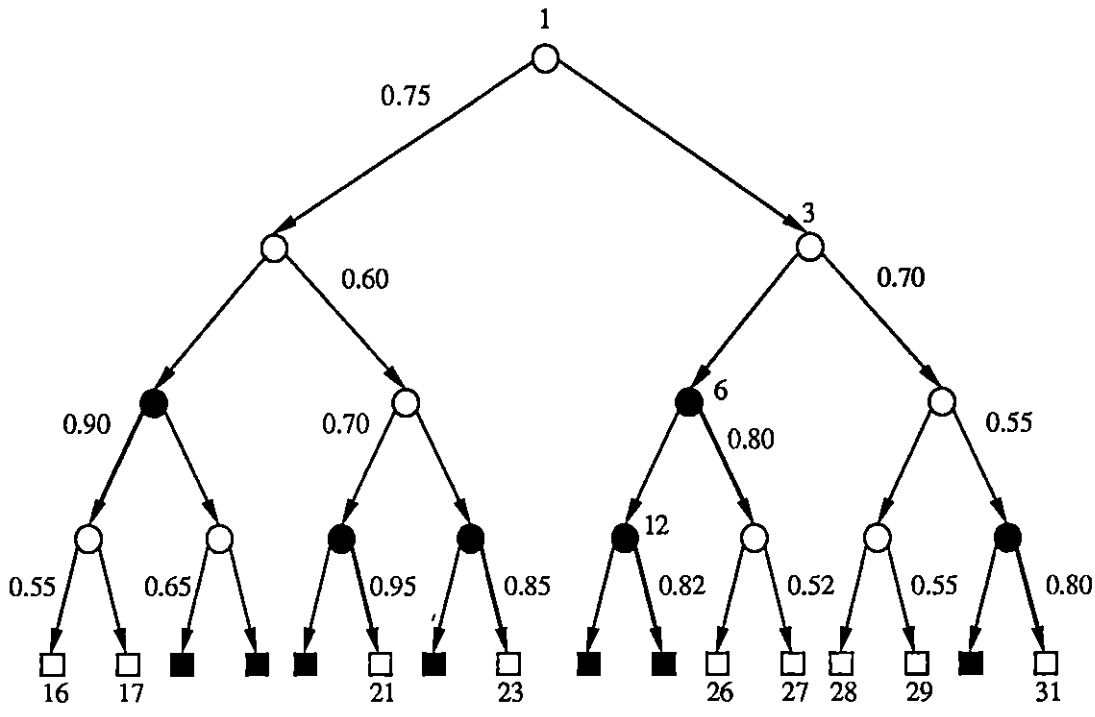

FIGURE 4.2. Example Tree

abilities that are greater than 0.5 appear attached to the corresponding branches. The filled circles represent statistical nodes while the statistical branches are emboldened. The shaded squares constitute the non-reachable final nodes. Observe that the non-statistical branch of a statistical node leads to non-reachable final nodes. This is true because every time we reach such a node we always follow the statistical branch without performing any test. This way many other nodes of the tree become unreachable too. For example, in figure 4.2, the statistical node 12 is unreachable because every time we are at the node 6 we continue with the right branch which holds the probability 0.8.

In order for the tree gain to be derived, the posterior probabilities of the paths must be calculated, that is, the probabilities after partitioning the interior nodes of the tree as it was discussed earlier in this section. If P_{Pj} is the posterior probability that the path j is followed we must have

$$\sum_{j \in F} P_{Pj} = 1 \qquad \text{(EQ 4.19)}$$

We will prove the following proposition for the posterior probabilities P_{Pj}

Proposition 4.3. *In a decision tree, where a certain number of decision nodes have been marked as statistical, the posterior probabilities P_{Pj} of a path j to be followed, are given by the following expressions*

$$P_{Pj} = 0, \qquad \forall j \in F_2 \qquad \text{(EQ 4.20)}$$

$$P_{Pj} = \frac{P_j}{P_{Cj}}, \qquad \forall j \in F_1 \qquad \text{(EQ 4.21)}$$

Proof. A path which ends up with a non-reachable final node will never be followed. That is (Eq. 4.20) always holds. In order to prove (Eq. 4.21), let us suppose that $E_1, E_2, \ldots, E_n$ are the n branches of the path j. If E_i also stands for the fact that the branch E_i is followed we obviously have

$$P_{Pj} = P(E_1, E_2, \ldots, E_n)$$

or

$$P_{Pj} = P(E_n | E_1, \ldots, E_{n-1}) \cdot \ldots \cdot P(E_2 | E_1) \cdot P(E_1) \qquad \text{(EQ 4.22)}$$

But if E_i is a statistical branch then we get

$$P(E_i | E_1, \ldots, E_{i-1}) = 1 \qquad \text{(EQ 4.23)}$$

because such a branch is always followed. Substituting (Eq. 4.23) in (Eq. 4.22) for every statistical branch we obtain

$$P_{Pj} = \prod P(E_v | E_1, \ldots, E_{v-1})$$

for all v such that E_v is a non-statistical branch. Therefore, P_{Pj} is expressed as the product of the probabilities of the non-statistical branches. From propositions 4.1.and 4.2 we have

$$P_j = P_{Pj} \cdot P_{Cj} \Leftrightarrow P_{Pj} = \frac{P_j}{P_{Cj}} \quad \blacksquare$$

Lemma 4.1. *The probabilities P_{Pj} of all the reachable final nodes add up to 1. That is,*

$$\sum_{j \in F_1} P_{Pj} = 1 \qquad \text{(EQ 4.24)}$$

Proof. It directly follows from (4.19) and (4.20). ■

The tree gain which represents the average number of questions that are not asked when we go through the tree once, can be computed by weighting the gain appearing in (Eq 4.18) with the posterior probability P_{Pj} that the path j is followed and then building up the following sum

$$G = \sum_{j \in F_1} P_{Pj} \cdot G_j = \sum_{j \in F_1} P_{Pj} \cdot [m_j - B_j \cdot (1 - P_{Cj})] \qquad \text{(EQ 4.25)}$$

From the last equation we conclude that the expression

$$\frac{G}{n} \times 100$$

gives the gain as the percentage of the depth of the decision tree.

For a specific partition of the interior nodes of the tree, all the components of the above expression except for B_j are known and well-defined. Therefore, in order for the gain G in (Eq. 4.25) to be maximized, two specific tasks must be accomplished : a) the optimal partition of the interior nodes of the tree must be found and b) the backtracking cost B_j for each reachable final node resulting from

that partition must be minimized. The derivation and minimization of B_j is the subject of chapter 5. In chapter 6 two partitioning techniques are proposed. The problem of finding the optimal partitioning appears to be a difficult and complicated one.

4.3 Probability of a Correct Path

When the probabilistic information is not taken into account and all the attribute tests take place as we traverse the tree, we have assumed that the resulting path is correct with absolute certainty. On the other hand, this is not the case when some nodes of the tree are statistical nodes. More specifically, it was shown (see equation 4.21) that for a single path j the probability P_{Cj} that the path is correct given that it has been followed is

$$P_{Cj} = \frac{P_j}{P_{Pj}} \qquad \text{(EQ 4.26)}$$

By weighting the probabilities P_{Cj} with the corresponding posterior probabilities P_{Pj} and then summing up the results we end up with the probability that we follow a correct path when we go through the tree once. We have

$$P_C = \sum_{j \in F_1} P_{Pj} \cdot P_{Cj} = \sum_{j \in F_1} P_{Pj} \cdot \frac{P_j}{P_{Pj}} = \sum_{j \in F_1} P_j \qquad \text{(EQ 4.27)}$$

We observe that the probability of a correct path is equal to the sum of the a priori probabilities of the reachable nodes. In other words, when a path which corresponds to a reachable final node is the correct one, we are going to follow this path with absolute certainty. Indeed, this is true and can be explained as follows : if we start traversing the tree, while we do not meet a node where a probabilistic decision must be taken, we are on the correct path. When we reach such a node and we follow its statistical branch, we are still on the correct path because

its non-statistical branch leads to non-reachable final nodes. But we have made the hypothesis that the correct final node is a reachable node. The above steps are repeated, which means that we are always on the correct path.

The probability P_W of a wrong path will be

$$P_W = 1 - P_C = 1 - \sum_{j \in F_1} P_j = \sum_{j \in F} P_j - \sum_{j \in F_1} P_j = \sum_{j \in F_2} P_j \qquad \text{(EQ 4.28)}$$

that is the sum of the a priori probabilities of the non-reachable nodes of the tree. The probabilities P_C and P_W quantify our degree of belief that we will end up with a correct or wrong final diagnosis respectively, when we traverse the tree once and before we backtrack. As it is clear from (Eq. 4.27) and (Eq. 4.28) these two probabilities depend exclusively on the partitioning of the interior nodes of the tree into statistical and non-statistical ones which determines the sets F_1 and F_2. In particular, P_C is a measure of the probability of the paths selected by a specific partitioning technique. If the probability P_C is high while the number of reachable nodes is small, then it seems that very probable diagnostic paths have been chosen.

What has been discussed so far is illustrated in the following example.

Example 4.1. Let us consider the tree of figure 4.2. The set of reachable nodes F_1 will be as follows

$$F_1 = \{16, 17, 21, 23, 26, 27, 28, 28, 31\}$$

If the reachable final node 26 is the correct one, it is obvious that we are going to reach that node although a statistical decision is taken at the node 6. On the other hand the posterior probability that path 26 is followed is

$$P_{P26} = 0.25 \cdot 0.30 \cdot 0.48 = 0.036$$

The probability that path 26 is correct given that it has been followed is

$$P_{C26} = 0.80$$

The probabilities P_{Cj}, P_{Pj} and P_j for all the reachable nodes are presented in table 1. According to (Eq. 4.28), the probability that we are going to follow a correct path is given as

$$P_C = \sum_{j \in F_1} P_j = 0.89974$$

From the last equation we observe that although only 9 out of 16 final nodes are reachable, the probability P_C is very high.

j	P_{Cj}	P_{Pj}	P_j
16	0.90	0.1650	0.1485
17	0.90	0.1350	0.1215
21	0.95	0.3150	0.2993
23	0.85	0.1350	0.1148
26	0.80	0.0360	0.0288
27	0.80	0.0390	0.0312
28	1.00	0.0354	0.0354
29	1.00	0.0433	0.0433
31	0.80	0.0963	0.0770

TABLE 4.1 Probabilities of the Reachable Nodes

Chapter 5

Derivation and Minimization of the Backtracking Cost

As stated in the previous chapter, when we end up with a wrong diagnosis, we backtrack and search for the correct path. Such a procedure results in a cost which is called the backtracking cost. In this chapter, an expression for the backtracking cost is derived. Starting from simple forms of the cost, a general expression is constructed. Then, the optimal backtracking procedure which corresponds to the minimum cost is obtained. The chapter concludes with the presentation of some characteristics of the optimal backtracking procedure which facilitate the minimization process.

5.1 Probabilistic Considerations for a Single Path

It has already been stated that the aim of our work is to maximize the gain G given by (Eq. 4.25). From that equation we observe that, for a specific partition of the interior nodes of the tree, the gain G is maximized if the backtracking cost B_j is minimized for every reachable final node j. In this chapter a single path of the tree is considered and its backtracking cost is calculated. Clearly, the results obtained can be applied to each path of the set F_1. In order to simplify our nota-

tion the subscript j which refers to a specific path j is avoided in the analysis to follow.

Let us consider an arbitrary path in the tree which ends at a reachable final node. Let $A_1, A_2, \ldots, A_n$ be the n nodes of the path (we do not take into account the final node) where A_1 is the root. Assume that the path contains m statistical nodes ($1 \leq m \leq n$), say $A_{k_1}, A_{k_2}, \ldots, A_{k_m}$ with heights $h_1, h_2, \ldots, h_m$ respectively. If we follow that path and the final diagnosis turns out wrong, it happened because a wrong decision was made in at least one statistical node. We are most interested in the first statistical node of the path where a wrong decision occurred. This is so because, after passing that node, we are already on a wrong path. Let us now define the following facts

- W : The arbitrary path that we followed is wrong.
- A_i : A correct decision was made at node A_i of the path.
- $\bar{A}_i$: A wrong decision was made at node A_i of the path.

We form the following conditional probability

$$P(A_1, A_2, \ldots, A_{i-1}, \bar{A}_i | W) \qquad i = 1, 2, \ldots, n \qquad \text{(EQ 5.1)}$$

The last expression gives the probability that the path is wrong because a wrong decision was taken at node A_i and all other decisions preceding that were correct. Obviously, if A_i is a non-statistical node

$$P(A_1, A_2, \ldots, A_{i-1}, \bar{A}_i | W) = 0$$

On the other hand, for an arbitrary statistical node A_{k_i} of the path, the expression

$$P(A_1, A_2, \ldots, A_{k_i - 1}, \bar{A}_{k_i} | W) \qquad i = 1, 2, \ldots, m \qquad \text{(EQ 5.2)}$$

gives the probability that the first wrong decision occurred at node A_{k_i}. In order to calculate this probability we apply the inversion formula given in (Eq. 2.8). We have

$$P(A_1, \ldots, A_{k_i-1}, \bar{A}_{k_i} | W) = \frac{P(W | A_1, \ldots, A_{k_i-1}, \bar{A}_{k_i}) \cdot P(A_1, \ldots, A_{k_i-1}, \bar{A}_{k_i})}{P(W)} \quad \text{(EQ 5.3)}$$

Clearly,

$$P(W | A_1, \ldots, A_{k_i-1}, \bar{A}_{k_i}) = 1 \qquad i = 1, 2, \ldots, m \quad \text{(EQ 5.4)}$$

because the path is certainly wrong if a wrong decision was made at any node. Furthermore, in equation 5.3

$$P(W) = 1 - P(C)$$

and $P(C)$ is calculated by means of proposition 4.2. Therefore, (Eq. 5.3) can be rewritten as follows

$$P(A_1, \ldots, A_{k_i-1}, \bar{A}_{k_i} | W) = \frac{P(A_1, \ldots, A_{k_i-1}, \bar{A}_{k_i})}{1 - P(C)} \quad \text{(EQ 5.5)}$$

In order to compute the probability $P(A_1, \ldots, A_{k_i-1}, \bar{A}_{k_i})$ we can apply the chain rule formula. We get

$$P(A_1, \ldots, A_{k_i-1}, \bar{A}_{k_i}) = P(\bar{A}_{k_i} | A_1, \ldots, A_{k_i-1}) \cdot \ldots \cdot P(A_2 | A_1) \cdot P(A_1) \quad \text{(EQ 5.6)}$$

But for a non-statistical node A_r, $1 \leq r \leq k_i - 1$

$$P(A_r | A_1, \ldots, A_{r-1}) = 1 \quad \text{(EQ 5.7)}$$

Substituting (Eq. 5.7) in (Eq. 5.6) for every non-statistical node we get

$$P(A_1, \ldots, A_{k_i-1}, \bar{A}_{k_i}) = P(\bar{A}_{k_i} | A_1, \ldots, A_{k_i-1}) \cdot P(A_{k_{i-1}} | A_1, \ldots, A_{k_{i-1}-1})$$

$$\ldots \cdot P(A_{k_2} | A_1, \ldots, A_{k_2-1}) \cdot P(A_{k_1} | A_1, \ldots, A_{k_1-1}) \quad \text{(EQ 5.8)}$$

that is, $P(A_1, \ldots, A_{k_i-1}, \bar{A}_{k_i})$ depends exclusively on the probabilities assigned to the statistical nodes which are located above node A_{k_i} in the path. More specifically, it is derived by multiplying the probabilities of all the statistical branches that are above A_{k_i} and then by multiplying again the outcome with the probability of the non-statistical branch of A_{k_i}. All these probabilities are known from our initial model and therefore the expression in (Eq. 5.5) can be calculated. By applying the procedure which has been described so far, the probability that the first wrong decision occurred at a particular statistical node can be computed (see equation 5.2). We note that

$$\sum_{i=1}^{m} P(A_1, \ldots, A_{k_i-1}, \bar{A}_{k_i} | W) = 1 \qquad \text{(EQ 5.9)}$$

because the probabilities that appear in the above summation correspond to a set of exhaustive events.

We now define P_r, $r = 1, 2, \ldots, m$, to be the probability that the first wrong decision took place either at node A_{k_r} or at a node which follows A_{k_r} in the path $(A_{k_{r+1}}, \ldots, A_{k_m})$. Then P_r can be expressed as follows

$$P_r = \sum_{i=r}^{m} P(A_1, \ldots, A_{k_i-1}, \bar{A}_{k_i} | W) \qquad \text{(EQ 5.10)}$$

If we back up to node A_{k_r} after a wrong final diagnosis, P_r expresses the probability of correct backtracking, that is, the probability that we are going to find the correct final node after going back to node A_{k_r} and traversing the tree from this point by performing all the tests at all the nodes we meet. Obviously $P_1 = 1$. This is the case of backtracking to the node A_{k_1}.

Example 5.1. Assume that we have followed the path shown in figure 5.1. The shaded circles correspond to the statistical nodes of the path. Therefore, $A_{k_1} \equiv A_2$, $A_{k_2} \equiv A_4$ and $A_{k_3} = A_5$. In this example we have $n = 5$ and $m = 3$. The heights of

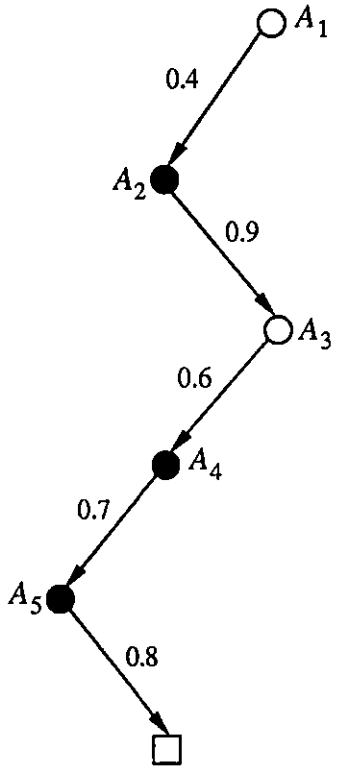

FIGURE 5.1. The path of Example 5.1

the nodes A_{k_1}, A_{k_2} and A_{k_3} are $h_1 = 4$, $h_2 = 2$ and $h_3 = 1$ respectively. According to proposition 4.2, the probability that the path is wrong will be

$$P(W) = 1 - P(C) = 1 - (0.9 \cdot 0.8 \cdot 0.7) = 0.496$$

Given now that the path is wrong, the probabilities that the first wrong decision occurred at node A_{k_1}, A_{k_2} or A_{k_3} can be derived by applying (Eq. 5.5) and (Eq. 5.8). We have

$$P(A_1, \bar{A}_2 | W) = \frac{(1-0.9)}{0.496} \approx 0.2$$

$$P(A_1, A_2, A_3, \bar{A}_4 | W) = \frac{0.9 \cdot (1-0.7)}{0.496} \approx 0.54$$

$$P(A_1, A_2, A_3, A_4, \bar{A}_5 | W) = \frac{0.9 \cdot 0.7 \cdot (1-0.8)}{0.496} \approx 0.26$$

Hence the probabilities P_1, P_2 and P_3 (defined above) are

$$P_1 = P(A_1, \bar{A}_2 | W) + P(A_1, A_2, A_3, \bar{A}_4 | W) + P(A_1, A_2, A_3, A_4, \bar{A}_5 | W) = 1$$

$$P_2 = P(A_1, A_2, A_3, \bar{A}_4 | W) + P(A_1, A_2, A_3, A_4, \bar{A}_5 | W) \approx 0.80$$

$$P_3 = P(A_1, A_2, A_3, A_4, \bar{A}_5 | W) \approx 0.26$$

For instance, if we backtrack to node A_4 after a wrong diagnosis, there is a probability of 0.80 that we are going to find the correct path when traversing the tree from this point by determining the path at each node we meet. Note that not only $\{h_i\}$ but also $\{P_i\}$, $i = 1, 2, \ldots, m$, constitute a decreasing sequence.

5.2 The Backtracking Cost

The backtracking cost represents the number of tests that take place after a wrong path is diagnosed and before the correct path is found. In order to simplify our model we assume that during our search, after backtracking, for the correct path, the probabilistic information that is available at the nodes of the tree is not taken into account any more. Thus, for example, if we back up to the first statistical node of the path which has height h_1 then the backtracking cost equals h_1 because h_1 tests are required before the correct final node is reached. As it will become clear later, the backtracking cost B depends not only on the height of the statistical node A_{k_i} at which we backtrack but also on the probability of successful backtracking P_i which is associated with that node. Before we proceed to the formulation of the general expression of the cost B for a single path of the tree, that cost is derived for a 2-step backtracking procedure, that is, a procedure in which we backtrack at most to one statistical node before we back up to node A_{k_1}.

5.2.1 The Cost B in a 2-Step Backtracking Procedure

In the following, we assume that after a wrong path is taken we backtrack to a certain statistical node, say A_{k_i}. If we end up again with a wrong final node we backtrack to the first statistical node A_{k_1}. We note here that in every backtracking procedure A_{k_1} is always the last node that we may backtrack to. That is so because there is always a non-zero probability that the first wrong decision took place at that node. If h_i is the height of node A_{k_i} then the backtracking cost is h_i with probability P_i (the probability of successful backtracking) and $(h_i + h_1)$ with probability $(1 - P_i)$. Hence we have

$$B = P_i \cdot h_i + (1 - P_i) \cdot (h_i + h_1)$$

or

$$B = h_i + h_1 \cdot (1 - P_i) \qquad \text{(EQ 5.11)}$$

In the last equation, h_1 is constant for a specific path in the tree. On the other hand, h_i and P_i depend on the choice of which node A_{k_i} to back up to. More specifically, the first term of (Eq. 5.11) decreases as the height h_i is decreased. On the other hand, the second term, which contains P_i, increases as the height h_i becomes smaller, that is, as node A_{k_i} becomes neare to the leaf node of the path. Based on these observations, we now present three different backtracking approaches. We note again that, in these approaches, at most one backtracking step occurs before we backtrack to node A_{k_1}.

5.2.1.1 The Maximum-Probability Approach

This is actually the approach already mentioned. In this approach we backtrack to the statistical node with the highest probability P_i. The node which satisfies this requirement is the node A_{k_1} with $P_1 = 1$ and height h_1. If we substitute this result in (Eq. 5.11) we obtain

$$B = h_1 \qquad \text{(EQ 5.12)}$$

5.2.1.2 The Minimum-Height Approach

In this approach we first back up to the last statistical node of the path, that is to the node A_{k_m}. This way we minimize the first term of the cost B. (Eq. 5.11) can be rewritten as follows

$$B = h_m + h_1 \cdot (1 - P_m) \qquad \text{(EQ 5.13)}$$

The probability P_m is given by (Eq. 5.10) for $r = m$. In this case, equations (5.5) and (5.10) are equivalent. They both express the probability that the path is wrong because a wrong decision was made at node A_{k_m}. From (Eq. 5.10) we have

$$P_m = \frac{P(A_1, \ldots, A_{k_m - 1}, \bar{A}_{k_m})}{P(W)} \qquad \text{(EQ 5.14)}$$

If we multiply the numerator and denominator of (Eq. 5.14) by $P(A_{k_m} | A_1, \ldots, A_{k_m - 1})$ then from (Eq. 5.8) and proposition 4.2 we have

$$P_m = \frac{P(C)}{P(W)} \cdot \frac{P(\bar{A}_{k_m} | A_1, \ldots, A_{k_m - 1})}{P(A_{k_m} | A_1, \ldots, A_{k_m - 1})} \qquad \text{(EQ 5.15)}$$

5.2.1.3 The Optimal Approach

Consider now two arbitrary statistical nodes of the path, say A_{k_i} and A_{k_j}. Let us compare their costs as calculated by (Eq. 5.11). We have

$$h_i + h_1 \cdot (1 - P_i) < h_j + h_1 \cdot (1 - P_j)$$

or

$$(h_i - h_j) < h_1 \cdot (P_i - P_j) \qquad \text{(INQ 5.16)}$$

If inequality (5.16) holds, we conclude that the cost associated with node A_{k_i} is less than the corresponding cost of the node A_{k_j}. Therefore, it is preferable that we first backtrack to node A_{k_i} instead of node A_{k_j}. By applying (Inq 5.16) we can find which one of the nodes A_{k_i}, $i = 1, 2, ..., m$, provides the lowest cost B. We then backtrack to that node first. This is illustrated in the following example.

Example 5.2. Let us apply the last method to the path of example 5.1. We have

$$(h_2 - h_1) < h_1 \cdot (P_2 - P_1) \Rightarrow -2 < -0.8$$

which holds. Furthermore

$$(h_2 - h_3) < h_1 \cdot (P_2 - P_3) \Rightarrow 1 < 4 \cdot 0.54$$

which holds as well. Hence we should first backtrack to the node $A_{k_2} \equiv A_4$.

5.2.2 The General Expression for the Cost *B*

In the previous section, three different backtracking approaches were described. In all three of them one backtracking step occurs before we back up to the first statistical node of the path. Actually, when more than two statistical nodes exist, more than two backtracking steps can be considered. Such a procedure may result in a lower cost B than the minimum cost obtained by applying the method described in section 5.2.1.3. We will prove the following theorem in which a general expression for the backtracking cost is derived. In order to avoid confusion we assume in the following that $S_1, S_2, ..., S_m$ represent the statistical nodes of the path with heights $h_1, h_2, ..., h_m$ and probabilities $P_1, P_2, ..., P_m$.

Theorem 5.1. *If $S_{N_1}, S_{N_2}, ..., S_{N_q}$ are the q statistical nodes at which we back up to where $1 \le q \le m$, (that is they constitute a subset of the initial statistical nodes), S_{N_q} is the first node at which we backtrack and $S_{N_1} \equiv S_1$, the cost B can be written in the following form*

$$B = \sum_{i=1}^{q} h_{N_i} \cdot (1 - P_{N_{i+1}}), \qquad P_{N_{q+1}} = 0 \tag{EQ 5.17}$$

Proof. As was stated previously, the cost B gives the number of nodes where the test at the node is performed, during the backtracking procedure. If we assume that we back up to nodes $S_{N_1}, S_{N_2}, \ldots, S_{N_q}$ as was described before, the cost B can be expressed as follows

$$B = P_{N_q} \cdot h_{N_q} + (1 - P_{N_q}) \cdot \{h_{N_q} + P'_{N_{q-1}} \cdot h_{N_{q-1}} + (1 - P'_{N_{q-1}}) \cdot \{h_{N_{q-1}} + \ldots$$

$$\ldots (1 - P'_{N_2}) \cdot \{h_{N_2} + h_{N_1} \cdot P'_{N_2}\} \ldots \} \tag{EQ 5.18}$$

In equation (5.18), the term P'_{N_k}, $2 \le k \le q-1$ represents to the probability that the first wrong decision took place at node S_{N_k}, denoted as fact A, given that it did not take place at the nodes $S_{N_{k+1}}, \ldots, S_{N_q}$, denoted as fact D. From (Eq. 2.5) we get

$$P'_{N_k} = P(A|D) = \frac{P(A, D)}{P(D)} = \frac{P_{N_k} - P_{N_{k+1}}}{1 - P_{N_{k+1}}} \tag{EQ 5.19}$$

Furthermore

$$1 - P'_{N_k} = \frac{1 - P_{N_k}}{1 - P_{N_{k+1}}} \tag{EQ 5.20}$$

If we substitute (5.19) and (5.20) into (5.18), we observe that the denominators of the above fractions cancel out. Thus the cost B can be rewritten as

$$B = P_{N_q} \cdot h_{N_q} + (1-P_{N_q}) \cdot h_{N_q} + (P_{N_{q-1}} - P_{N_q}) \cdot h_{N_{q-1}} + (1-P_{N_{q-1}}) \cdot h_{N_{q-1}} + \ldots$$

$$\ldots + (P_{N_1} - P_{N_2}) \cdot h_{N_2} + (1-P_{N_1}) \cdot h_{N_2} + (1-P_{N_2}) \cdot h_{N_1} =$$

$$h_{N_q} + (1-P_{N_q}) \cdot h_{N_{q-1}} + \ldots + (1-P_{N_3}) \cdot h_{N_2} + (1-P_{N_2}) \cdot h_{N_1} \Leftrightarrow$$

$$B = \sum_{i=1}^{q} h_{N_i} \cdot (1-P_{N_{i+1}}), \qquad P_{N_{q+1}} = 0 \qquad \blacksquare$$

The above result can also be verified intuitively. Indeed, after ending up with a wrong path and following the backtracking process described in theorem 5.1, $(1-P_{N_{i+1}})$ expresses the probability that the first wrong decision occurred at a statistical node of the path before we reach the statistical node $S_{N_{i+1}}$. Therefore, in such a case we always back up to node S_{N_i} and thus in (Eq. 5.17) we weight h_{N_i} (the height of S_{N_i}) with $(1-P_{N_{i+1}})$. Indeed, assuming that the tree is balanced and that after backtracking to a particular node S_{N_i} we traverse the tree by asking all the questions at the nodes we meet, we conclude that the number of these nodes is equal to h_{N_i}. Hence the summation in (Eq. 5.17) represents the cost of the backtracking process according to the definition given before.

In order to achieve the minimum value of the backtracking cost B we have to solve the following optimization problem

$$min\left[\sum_{i=1}^{q} h_{N_i} \cdot (1-P_{N_{i+1}})\right] \qquad \text{(EQ 5.21)}$$

In general, if there are m statistical nodes, we have 2^{m-1} different backtracking procedures which result in 2^{m-1} costs. This is so because the last node to which we back up is always node S_1. Therefore, 2^{m-1} is the number of all possible combinations of the remaining $m-1$ nodes. Out of these 2^{m-1} procedures there is one (or possibly more) with the minimum cost given in (Eq. 5.21). In

other words there is a combination of q nodes, $1 \leq q \leq m$, which minimizes the backtracking cost given in (Eq. 5.17).

By exhaustively examining all the possible combinations mentioned before, the minimum cost B for a particular path through the tree can be obtained. If we then repeat the same procedure for every path j of the set F_1 and substitute the results in (Eq. 4.25) the maximum gain G for a specific partition of the interior nodes of the tree is achieved.

Example 5.2. In the path of figure 5.2 we have $n = 6$ and $m = 3$. The statistical nodes of the path are $S_1 \equiv A_1$, $S_2 \equiv A_3$ and $S_3 \equiv A_6$. The probabilities of correct backtracking P_1, P_2 and P_3 have been calculated in example 5.1. Thus, $P_1 = 1$, $P_2 = 0.80$ and $P_3 = 0.26$. Furthermore we have $h_1 = 6$, $h_2 = 3$ and $h_3 = 1$. Since $m = 3$ there are $2^2 = 4$ different backtracking procedures. In the following we calculate the costs of all the four procedures. When we back up directly to the node S_1 the cost is

$$B = h_1 = 6$$

If we back up to the node S_3 and then to the node S_1 the cost is

$$B = h_3 + h_1 \cdot (1 - P_3) = 5.44$$

If we back up to the node S_2 and then to the node S_1 we have

$$B = h_2 + h_1 \cdot (1 - P_2) = 5.20$$

If we back up to the node S_3 then to the node S_2 and finally to the node S_1 the cost is

$$B = h_3 + h_2 \cdot (1 - P_3) + h_1 (1 - P_2) = 5.16$$

We observe that in this example the minimum cost is achieved by visiting all the statistical nodes during the backtracking process, until we find the correct path.

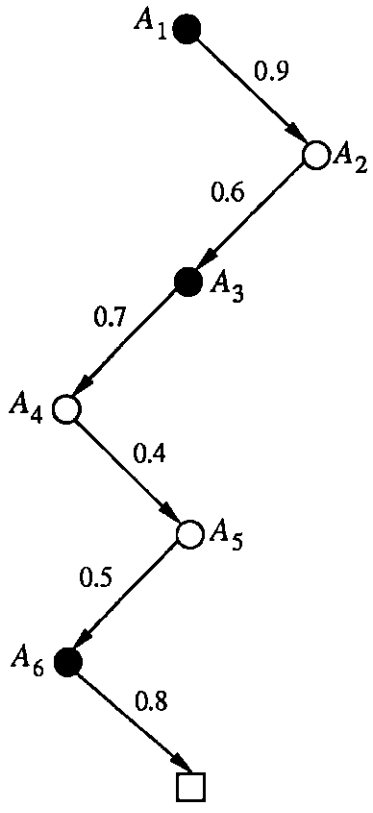

FIGURE 5.2. The Path of Example 5.2

5.2.3 Characteristics of the Optimal Backtracking Procedure

It was stated in section 5.2.2 that the minimization problem given in (5.18) can be solved by calculating 2^{m-1} different costs and choosing the nodes which give the lowest value for the cost. Then, the same procedure must be repeated for every path of the set F_1 in order for the maximum gain G to be obtained. For large trees, such a procedure may result in much computation. However, the optimal backtracking process for a single path has some special properties. These properties are summarized in the following propositions. Their application can reduce significantly the number of cost evaluations needed to find the minimum cost.

Proposition 5.1. *The optimal backtracking procedure, that is, the one which achieves the minimum cost given in (5.18), does not contain a node S_l such that*

$$\frac{h_l}{P_l} > h_1 \qquad \text{(INQ 5.22)}$$

where h_1 is the height of the node S_1.

Proof. Let us assume that the optimal backtracking procedure contains a node S_l which satisfies (5.22). The minimum cost associated with that procedure can be written as follows (see figure 5.3)

$$B_{min} = B_1 + h_l \cdot (1 - P_k) + h_t \cdot (1 - P_l) + B_2$$

In the last expression, we assume that S_k is the node which precedes node S_l (if any) and S_t is the node which follows S_l in the backtracking process. B_1 stands for the cost up to the node S_k and B_2 gives the cost after the node S_t. If, instead of backtracking to S_l after node S_k, we back up directly to node S_t the cost can be expressed as

$$B'_{min} = B_1 + h_t \cdot (1 - P_k) + B_2$$

We shall prove that $B'_{min} < B_{min}$. In other words we shall show that the cost B_{min} cannot be the minimum if S_l satisfies (5.22). We have

$$B'_{min} - B_{min} =$$

$$h_t \cdot (1 - P_k) - h_l \cdot (1 - P_k) - h_t \cdot (1 - P_l) =$$

$$[h_t \cdot P_l + P_k \cdot (h_l - h_t)] - h_l \qquad \text{(EQ 5.23)}$$

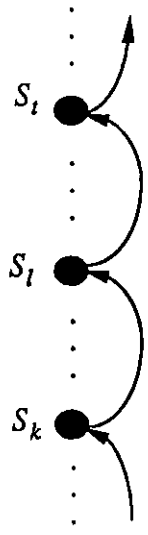

FIGURE 5.3. The Backtracking Procedure which Corresponds to B_{min}

But from (5.22) we have

$$h_l > h_1 \cdot P_l \geq h_t \cdot P_l > h_t \cdot P_l + P_k \cdot (h_l - h_t)$$

because $h_l - h_t < 0$. Hence, from the last inequality and (5.23), we conclude that if (5.22) is true then $B'_{min} < B_{min}$. ■

According to the last proposition, we can form the fractions h_i/P_i for all the statistical nodes of the path and then ignore the nodes which satisfy (5.22) when we search for the optimal backtracking process.

Lemma 5.1. *If all the statistical nodes of the path satisfy (5.22) then we achieve the minimum cost by backtracking directly to the first statistical node of the path* S_1.

Proof. The above lemma is a direct result of proposition 5.1. ■

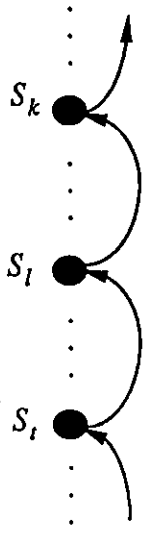

FIGURE 5.4. The Backtracking Procedure which Corresponds to B_{min}

Proposition 5.2. *The backtracking process which results in the minimum cost does not contain two consecutive nodes S_l and S_k with $h_l < h_k$ and $P_l < P_k$ which satisfy the following*

$$h_k < \frac{h_l}{P_l} \qquad \text{(INQ 5.24)}$$

Proof. Let us assume the opposite. In such a case the minimum cost could be written as follows (see figure 5.4)

$$B_{min} = B_1 + h_l \cdot (1 - P_t) + h_k \cdot (1 - P_l) + B_2 \qquad \text{(EQ 5.25)}$$

where nodes S_l and S_k satisfy (5.24). In the last expression, B_1 is the cost up to node S_t and B_2 is the cost after node S_k. If S_l is the first node of the backtracking process then $B_1 = 0$ and $P_t = 0$. If S_k is the last node of the process (that is

$S_k \equiv S_1$), then $B_2 = 0$. Therefore, (5.25) is a general expression for the cost in the case we are dealing with.

If, instead of backtracking to node S_l, we back up to node S_k after node S_t, the cost will be

$$B'_{min} = B_1 + h_k \cdot (1 - P_t) + B_2$$

Let us compare B_{min} and B'_{min}. We have

$$B'_{min} - B_{min} =$$

$$h_k \cdot (1 - P_t) - h_l \cdot (1 - P_t) - h_k \cdot (1 - P_l) =$$

$$(h_k \cdot P_l - h_l) + (h_l - h_k) \cdot P_t < 0$$

The last inequality holds because of (5.24) and the fact that $h_k > h_l$. Hence, we observe that B_{min} is not the minimum cost as it was assumed. ■

Proposition 5.3. *If $S_1, S_2, \ldots, S_m$ are the nodes of the path at which a statistical decision was taken and S_k, $1 \le k \le m$, is the node with the minimum value of h_i/P_i, that is*

$$\forall i, \quad 1 \le i \le m, \quad i \ne k \qquad \frac{h_k}{P_k} < \frac{h_i}{P_i} \qquad \text{(EQ 5.26)}$$

then the optimal backtracking process can not start at a node S_p where $h_p < h_k$.

Proof. Let us assume the opposite. In such a case, if S_l is the last node of the optimal backtracking procedure with $h_l < h_k$, the minimum cost can be written as follows (see figure 5.5)

$$B_{min} = B_1 + h_l \cdot (1 - P_r) + h_t \cdot (1 - P_l) + B_2$$

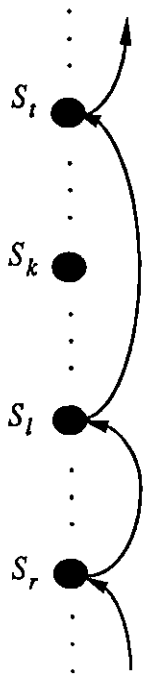

FIGURE 5.5. The Backtracking Procedure which Corresponds to B_{min}

where we can have either $S_k \equiv S_t$ or $h_k < h_t$. In the former case we get

$$\frac{h_k}{P_k} < \frac{h_l}{P_l} \Rightarrow h_k < \frac{h_l}{P_l}$$

which is not possible according to proposition 5.2. In the latter case, consider the cost of backtracking to the node S_k instead of the node S_l

$$B'_{min} = B_1 + h_k \cdot (1 - P_r) + h_t \cdot (1 - P_k) + B_2$$

We will prove that $B'_{min} < B_{min}$. We have

$$B'_{min} - B_{min} =$$

$$h_k \cdot (1-P_r) + h_t \cdot (1-P_k) - h_l \cdot (1-P_r) - h_t \cdot (1-P_l) =$$

$$(h_k - h_l) \cdot (1-P_r) + h_t \cdot (P_l - P_k)$$

We will prove that the last expression is negative. Since we know that $h_k > h_l$ and $0 < 1-P_r \leq 1$ it is sufficient to show that

$$(h_k - h_l) + h_t \cdot (P_l - P_k) < 0 \qquad \text{(EQ 5.27)}$$

From (5.26) and proposition 5.2 we have

$$\frac{h_k}{P_k} < \frac{h_l}{P_l} < h_t \qquad \text{(EQ 5.28)}$$

Therefore

$$\frac{h_k}{P_k} < \frac{h_l}{P_l} \Leftrightarrow h_t - \frac{h_l}{P_l} < h_t - \frac{h_k}{P_k} \Leftrightarrow$$

$$\frac{h_t \cdot P_l - h_l}{P_l} < \frac{h_t \cdot P_k - h_k}{P_k} \Rightarrow \frac{h_t \cdot P_l - h_l}{P_l} < \frac{h_t \cdot P_k - h_k}{P_l}$$

because the numerators of both fractions are positive according to (5.28) and furthermore $P_l < P_k$. Hence

$$h_t \cdot P_l - h_l < h_t \cdot P_k - h_k$$

or

$$(h_k - h_l) + h_t \cdot (P_l - P_k) < 0$$

■

The above propositions can be applied when the number of the statistical nodes of the path is larger than 2. Propositions 5.1 and 5.3 can be used first to

eliminate some statistical nodes. Then proposition 5.2 can be applied. Such a procedure is illustrated in the following example.

Example 5.3. Consider the path of figure 5.6. The characteristics of this path are summarized in table 5.1. In this table, P_{S_i} stands for the probability attached to the "child" of the statistical node S_i. Since, $m = 6$, $2^5 = 32$ different costs should be evaluated. But, according to proposition 5.1, node S_3 can not be in the optimal path because $h_3/P_3 > h_1$. Nodes S_5 and S_6 must also be excluded because from proposition 5.3 and the last column of table 5.1 we conclude that the optimal procedure can not start before node S_4. Finally, from proposition 5.2 we observe that the optimal backtracking procedure can not contain consecutively nodes S_4 and S_2 because $h_2 < h_4/P_4$.

Thus, there are only two costs left that need to be computed. The cost of backtracking to node S_4 and then to node S_1 and that of backtracking to the node S_2 and then to the node S_1. In the former case we have

$$B_1 = h_4 + h_1 \cdot (1 - P_4) = 6.5588$$

In the latter case we have

$$B_2 = h_2 + h_1 \cdot (1 - P_2) = 6.3136$$

We conclude that 6.3136 is the minimum cost. In order to verify this result we present in table 5.2 all the possible backtracking costs associated with the path of figure 5.6. The first 6 columns of the table contain the statistical nodes which constitute the different backtracking procedures. Observe that 6.3136 is indeed the minimum cost.

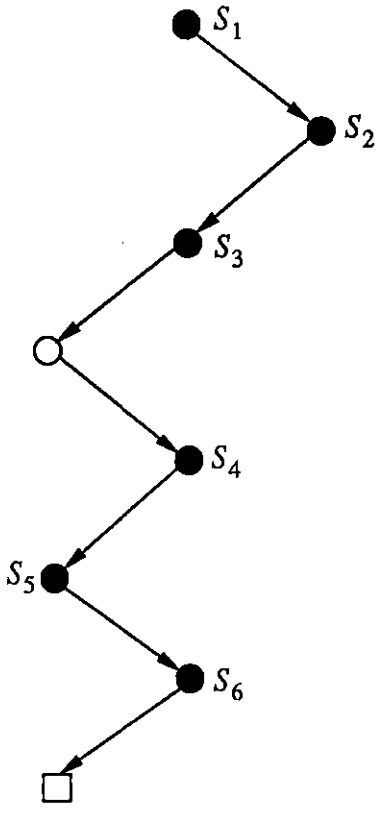

FIGURE 5.6. The Path of Example 5.3

i	h_i	P_{S_i}	P_i	P_i/h_i
1	7	0.9720	1.0000	7.0000
2	6	0.7405	0.9552	6.2813
3	5	0.9464	0.5530	9.0413
4	3	0.8185	0.4916	6.1031
5	2	0.8429	0.2944	6.7928
6	1	0.7934	0.1548	6.4602

TABLE 5.1. Characteristics of the Path of Figure 5.6.

1	2	3	4	5	6	B
S_1	-	-	-	-	-	7
S_6	S_1	-	-	-	-	6.9164
S_5	S_1	-	-	-	-	6.9392
S_4	S_1	-	-	-	-	6.5588
S_3	S_1	-	-	-	-	8.1290
S_2	S_1	-	-	-	-	**6.3136**
S_6	S_5	S_1	-	-	-	7.0944
S_6	S_4	S_1	-	-	-	8.3550
S_6	S_3	S_1	-	-	-	7.6570
S_6	S_2	S_1	-	-	-	6.3848
S_5	S_4	S_1	-	-	-	7.6756
S_5	S_3	S_1	-	-	-	8.6570
S_5	S_2	S_1	-	-	-	6.5472
S_4	S_3	S_1	-	-	-	8.6710
S_4	S_2	S_1	-	-	-	6.3640
S_3	S_2	S_1	-	-	-	7.9956
S_6	S_5	S_4	S_1	-	-	8.3660
S_6	S_5	S_3	S_1	-	-	9.3476
S_6	S_5	S_2	S_1	-	-	7.2376
S_6	S_4	S_3	S_1	-	-	9.2066
S_6	S_4	S_2	S_1	-	-	6.8996
S_6	S_3	S_2	S_1	-	-	8.2216
S_5	S_4	S_3	S_1	-	-	9.7878
S_5	S_4	S_2	S_1	-	-	7.4808
S_5	S_3	S_2	S_1	-	-	8.5236
S_4	S_3	S_2	S_1	-	-	8.5376
S_6	S_5	S_4	S_3	S_1	-	10.4782
S_6	S_5	S_4	S_2	S_1	-	8.1712
S_6	S_5	S_3	S_2	S_1	-	9.2140
S_6	S_4	S_3	S_2	S_1	-	9.0732
S_5	S_4	S_3	S_2	S_1	-	9.6546
S_6	S_5	S_4	S_3	S_2	S_1	10.3450

TABLE 5.2. The Backtracking Costs for the Path of Figure 5.6

Chapter 6

Partitioning of the Interior Nodes of the Tree

In this chapter the problem of partitioning the interior nodes of the tree into statistical and non-statistical ones is considered. The difficulty in obtaining the optimal partition is presented. Then, three specific methods are proposed and the motivation for each method is described. The application of the three techniques to different trees and the corresponding results are given. The chapter concludes with a comparison of the methods in terms of their performance, that is, the value of the gain G achieved.

6.1 The Optimal Partition

It has been stated in the previous chapters that in order for the maximum value of the gain G, given in (Eq. 4.25), to be attained, the appropriate partition of the interior nodes of the tree must be found. This partition can be called the optimal partition. Under the hypothesis that the optimal separation of the nodes is given, the maximum gain G is obtained by minimizing the backtracking cost B_j for all j

in F_1. On the other hand, it must be noted that the backtracking costs B_j depend on the partition of the interior nodes of the tree. Therefore, during the maximization procedure the above two concepts can not be considered separately. That is, for a specific partition of the nodes, the maximum gain may be obtained by minimizing the backtracking costs for all the reachable paths. That gain can then be compared with the maximum gain resulting from another partition of the nodes. The same process may be repeated until the absolute maximum is found.

In general, if there exist k interior nodes, 2^k possible partitions of them into statistical and non-statistical can be considered. For each of them the maximum gain must be calculated according to (4.25) and the results must be compared to each other in order for the optimal gain to be obtained. More specifically, in a tree of depth n there are $2^n - 1$ interior nodes. Clearly, the corresponding number of possible partitions is large ($2^{(2^n-1)}$). Even for a tree with $n = 5$ that number is high (2^{31}) resulting in a large amount of computation during the maximization procedure. For these reasons, an exhaustive evaluation of the gain for each possible partition is to be avoided. However, in the following sections three partitioning methods are proposed. In all of them while the computational complexity is kept relatively low, the gain obtained seems to be sufficiently high.

6.2 The Threshold Method

The main idea underlying this method is the following : the human expert often makes a decision about an attribute value without performing any test but instead, based on his experience, assigns a high probability to a particular outcome and accepts it as being true. Correspondingly, in our model when we reach an arbitrary node k we may decide not to ask the question which is contained in the node, that is , to consider the node as a statistical one. The last can happen, if the probability $max(P_{ky}, P_{kn})$ (see equation 4.5) is sufficiently high. But, how high must that probability be for an arbitrary node for the node to become statistical and the resulting gain G to be maximized ? The answer to that question will be given in the following sections.

6.2.1 Description of the Method

We know from (Eq. 4.5) that for an arbitrary node k the probability $max\,(P_{ky}, P_{kn})$ is greater than or equal to 0.5. We define the *threshold* t to be a number in the interval $[0.5, 1]$. When the node k is reached the quantity $max\,(P_{ky}, P_{kn})$ is compared to the threshold t. If it is greater than or equal to the threshold we follow the path with the maximum probability without asking the question at the node. Otherwise, the branch which will be followed is determined by the answer to the question of the node k. Clearly, a particular threshold t, with $0.5 \leq t \leq 1$, separates the interior nodes of the tree into statistical and non-statistical as follows : all the interior nodes whose probability is greater than or equal to t are considered statistical while the rest become non-statistical. Therefore, the gain G in (4.25) is expressed as a function of the probability threshold t in the interval $[0.5, 1]$. In the threshold method, we attempt to find the value of t which maximizes the gain G for a tree of depth n and a given distribution of probabilities at the nodes. We must note that the maximum of G so obtained is not the absolute maximum discussed in the previous section. It is simply, the optimal gain achieved by applying the threshold method.

6.2.2 The Maximization of G

In a binary balanced tree of depth n there are $s = 2^n - 1$ interior nodes and therefore there exist s probabilities greater than or equal to 0.5. For simplicity and without loss of generality, we have assumed that these probabilities are distinct and different from 0.5 and 1. Under that assumption, these probabilities divide the interval $[0.5, 1]$ into 2^n subintervals. Among these subintervals, illustrated in figure 6.1, there is one (or possibly more) say $(p_k, p_{k+1}]$ such that if $t \in (p_k, p_{k+1}]$ the gain G is maximized. We note that the value of G is constant in each of the subintervals. That is because, for each value of the threshold in a particular interval, the resulting partition of the nodes is the same. In order to get the maximum value of G, (4.25) can be evaluated in each of the subintervals. Starting from the last one, $(p_s, 1]$, we have : $F_1 = F$, $F_2 = \varnothing$ and $G_j = 0$,

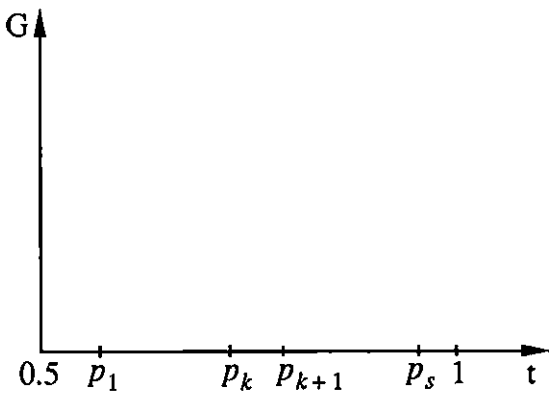

FIGURE 6.1. Representation of the Subintervals

$\forall j \in F_1$. As we move towards the first interval, elements (final nodes) are removed from the set F_1 and are added to the set F_2. At the first interval, $(0.5, 1\ ]$,we have

$$F_1 = \{K\}, \quad F_2 = F - \{K\}$$

where K is the final node of the path all the links of which have probabilities that are greater than 0.5. There is only one such path if we assume that all the probabilities of the tree are different from 0.5. This assumption has been made before.

We must state that not only the gain G but also the probability of a correct path P_C defined in section (4.3) is expressed as a function of the probability threshold t. According to (Eq. 4.27) P_C is an increasing function of t because as we move from 0.5 to 1 the number of reachable nodes is increased. At the same time the probability of a wrong path P_W becomes a decreasing function of t. All the above are illustrated in the following example.

Example 6.1. Consider the tree of figure 6.2. The probabilities that are greater than 0.5 are attached to the corresponding branches. A uniform random number generator has been used to create the probabilities of the branches. More specifically, the probabilities which correspond to the left branches of each node are

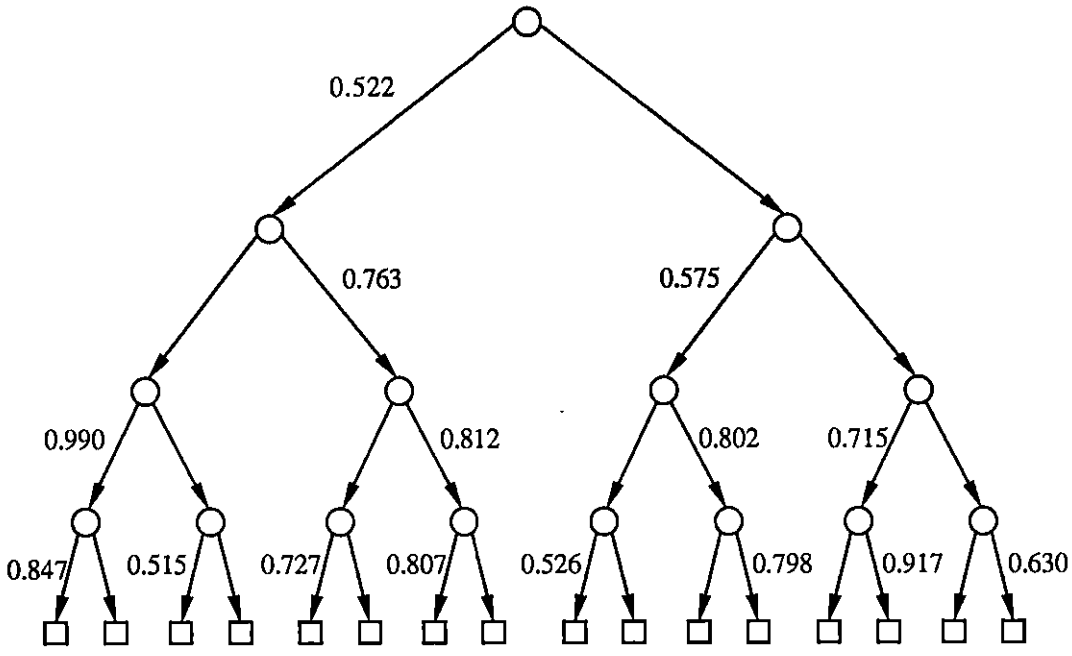

FIGURE 6.2. The Tree of Example 6.1

uniformly distributed in the interval [0, 1]. The gain G as a function of the threshold t in the interval [0.5, 1] is plotted in figure 6.3. In the expression of the gain, the backtracking costs B_j have been calculated by applying the approach described in section 5.2.1.3. The maximum value of the gain so obtained is 1.4015, for $t \in (0.575, 0.712]$. In such a case, the partition of the nodes which results, is shown in figure 6.4. We point out that the non-reachable statistical nodes do not appear in that figure. Observe that there exist only three reachable final nodes. The probability of correct backtracking P_C versus the threshold is plotted in figure 6.5. In that figure we have superimposed the gain shown in figure 6.3. P_C is equal to 0.57 in the interval in which we get the maximum value of G.

From the discussion made so far it seems that during the exhaustive search for the maximum of G, the gain function given in (4.25) must be evaluated in each of the subintervals, resulting in much computation. Actually, this is not the case because, as it is explained in appendix A, we do not need to compute all the terms of the summation (4.25) in each of the subintervals.

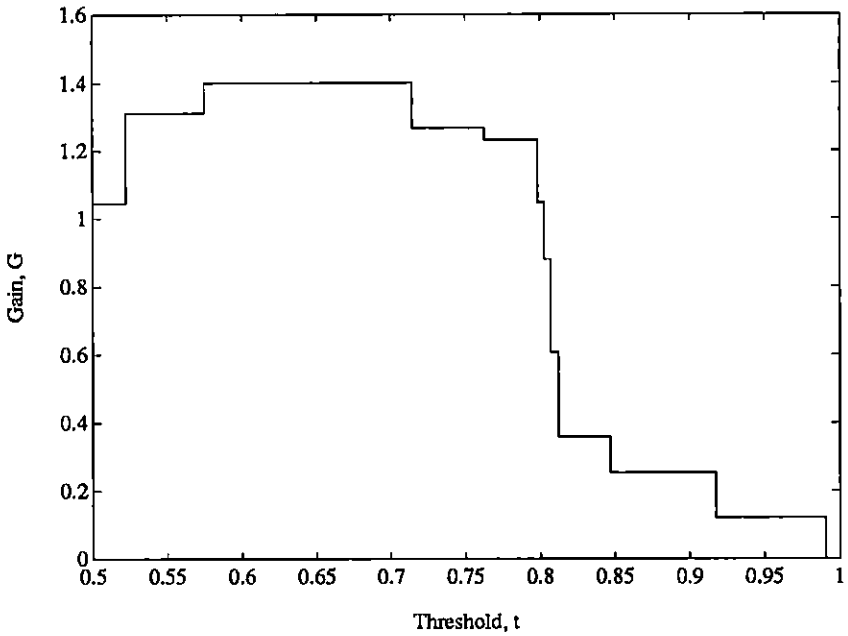

FIGURE 6.3. The Gain as a Function of the Threshold

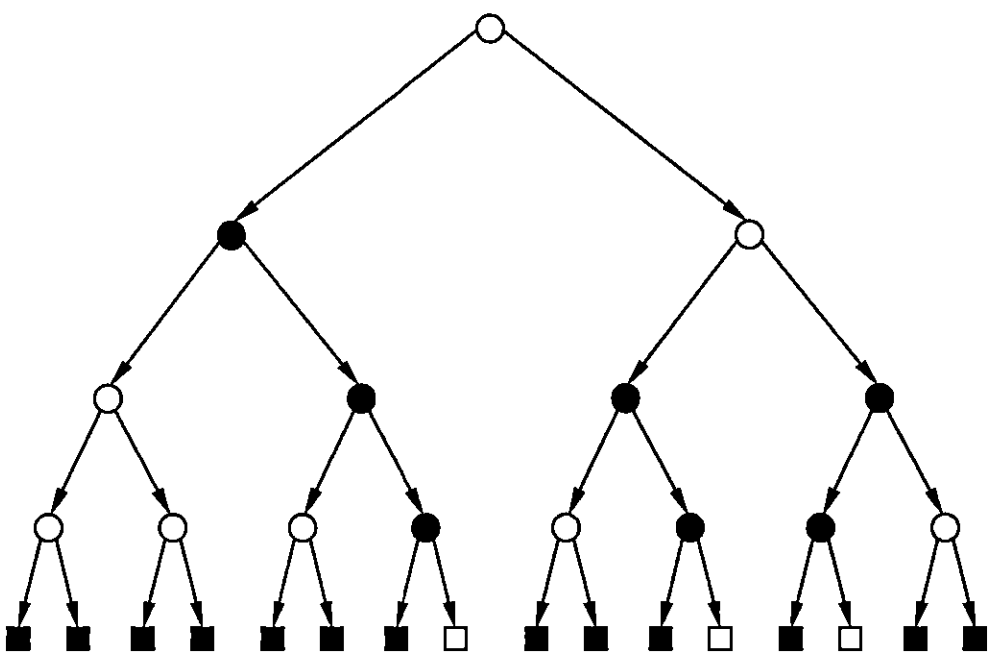

FIGURE 6.4. Partition of the Nodes after Applying the Threshold Method

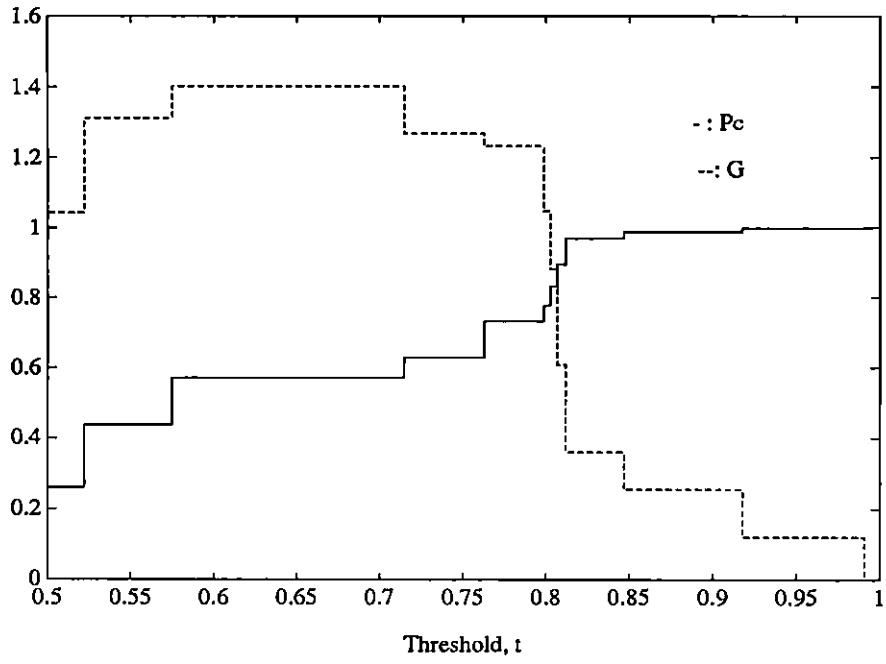

FIGURE 6.5. The Probability P_C versus the Threshold

6.2.3 Application of the Threshold Method

In this section some results of the application of the threshold method are presented. In all the examples given, we consider binary balanced trees of various depths. Since we do not have a specific model, the probabilities in the tree are produced by using a uniform random generator as was described in example 6.1. Clearly our method can be applied independently of the distribution of these probabilities but its performance is closely related to their values and to their locations in the tree structure. Therefore, for a particular distribution of probabilities, the threshold method (as well as the methods which will be described later) may result in a negative maximum gain G. Consequently, it is preferable, in such a case, that we do not take into consideration the probabilistic information that we possess but instead each test of the tree should be explicitly performed when we search for a final diagnosis.

In order to verify our results, a simulator ([28], [29]) has been constructed. For a specific tree structure, a given partition of the nodes and a particular backtrack-

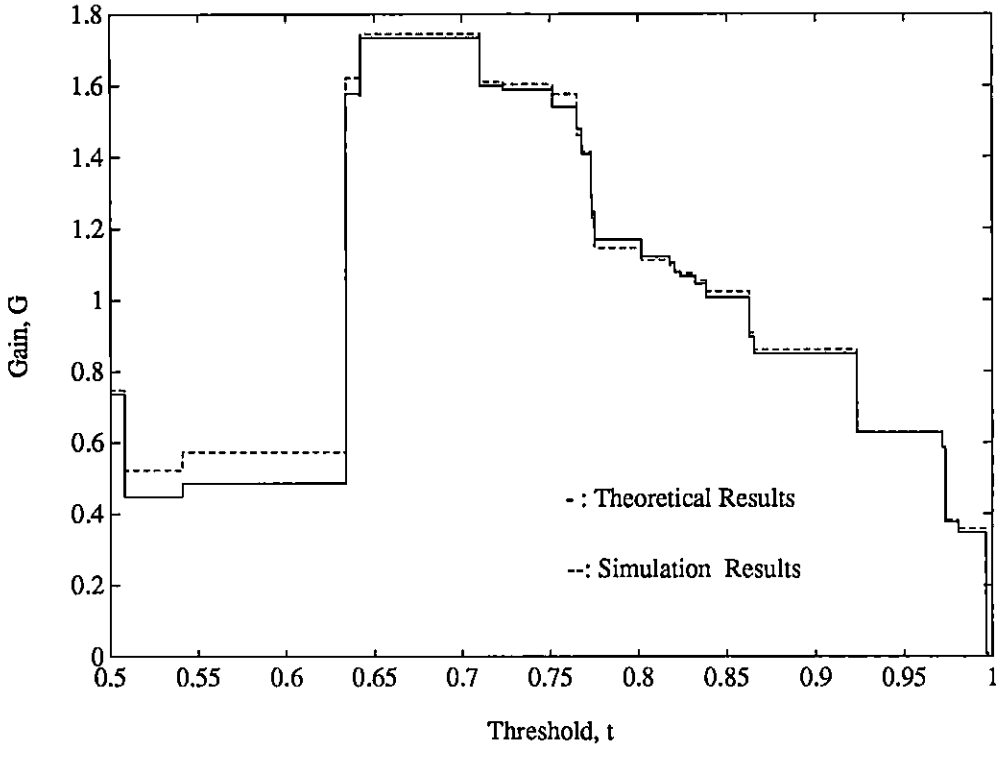

FIGURE 6.6. Comparison of Theoretical and Experimental Results

ing technique, the tree is traversed a sufficiently large number of times. The path, which is correct, is determined according to the probabilities P_j that are obtained by means of proposition 4.1. In the simulation, the total number of tests that are saved is counted and their average number for a single traversal of the tree is calculated. This number is then compared to the theoretical gain derived from (4.25). For a tree of depth $n = 6$ the gain versus the threshold is plotted in figure 6.6. In that figure, the solid line corresponds to the theoretical gain while the dashed line represents the gain observed in the simulation. In the simulation the tree is traversed 10,000 times. Observe how close the two lines are, confirming our theoretical derivation.

As stated before, the backtracking costs B_j in (4.25) depend on the backtracking technique used. The importance of the selection of a particular backtracking technique is highlighted in figure 6.7. A tree of depth 8 has been

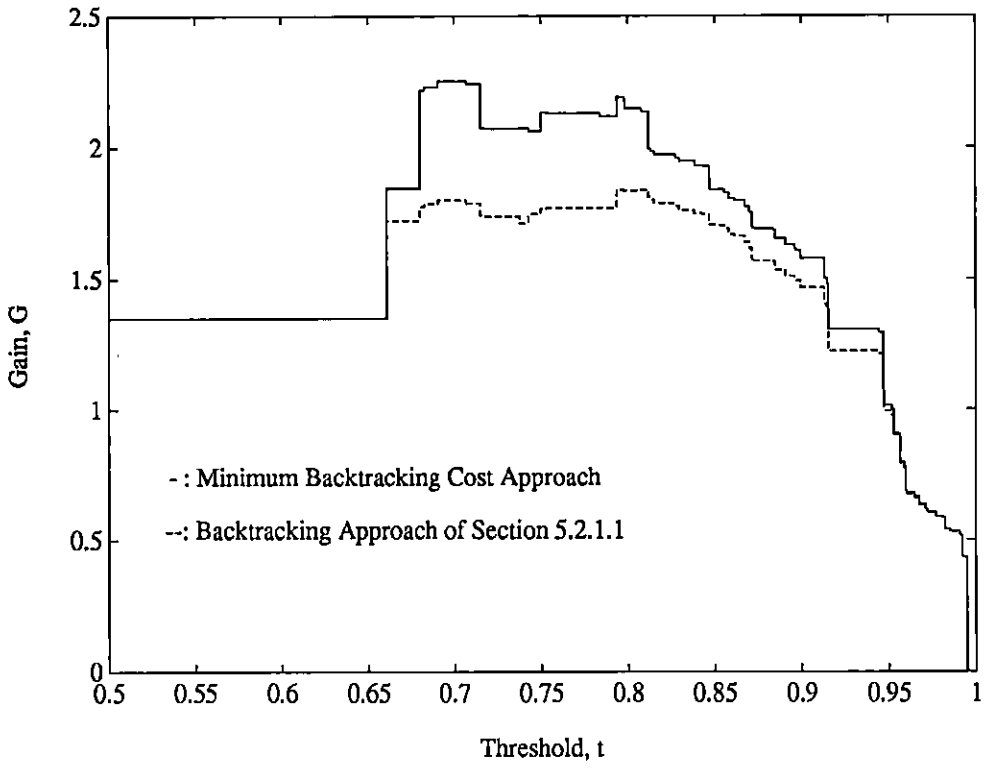

FIGURE 6.7. The Gain for Different Backtracking Techniques

considered. The dashed line corresponds to the gain when the approach presented in section 5.2.1.1 is used. According to this approach, we back up to the first statistical node of a wrong path. The solid line represents the gain when the minimum backtracking cost for each "reachable" path is substituted in (4.25). In the former case the maximum gain achieved is 1.840 while in the latter case it is 2.254. That is, an improvement of more than 20% has been attained.

The performance of the threshold method is illustrated in tables 6.1 and 6.2. In table 6.1 we consider trees of depth 6 while in table 6.2 we deal with trees of depth 10. In both cases a uniform random generator is used for the production of the probabilities of the branches. Ten different uniform distributions are considered in each table. G and P_C stand for the gain and the probability of a cor-

rect path respectively. The third column of each table holds the corresponding percentages of saved tests. The minimum backtracking cost approach is used.

i	G	%	P_C
1	0.884	14.7	0.857
2	1.706	28.4	0.270
3	2.001	33.5	0.261
4	1.289	21.5	0.794
5	1.432	23.9	0.616
6	1.648	27.5	0.825
7	1.644	27.4	0.420
8	1.279	21.3	0.363
9	1.090	18.2	0.763
10	1.863	31.1	0.423

TABLE 6.1. Performance of the Threshold Method for $n = 6$

i	G	%	P_C
1	1.322	13.2	0.802
2	1.393	13.9	0.648
3	1.714	17.1	0.900
4	1.507	15.1	0.638
5	1.308	13.1	0.794
6	1.998	20.0	0.782
7	1.517	15.2	0.845
8	1.442	14.4	0.900
9	1.157	11.6	0.913
10	1.127	11.2	0.881

TABLE 6.2. Performance of the Threshold Method for $n = 10$

6.3 An Alternative Method

Irrespective of their heights, the threshold method handles nodes with the same maximum probability in like manner. But, in general, a node which is located at a higher level of tree has a higher backtracking cost, as will be explained later. Therefore, except for the maximum probability given in (4.5), the height must be also taken into account when we search for a partition of the interior nodes of the tree. In the next section, an alternative method is described which in most cases gives better results than the threshold method in terms of the gain G obtained.

6.3.1 Description of the Method

We observe from equation (5.18) that the backtracking cost associated with a single path is increased when statistical decisions take place at high levels of the tree. That happens because when we end up with a wrong path we back up with some probability to high levels of the tree and then we have to traverse the tree by asking all all the questions at the nodes we meet. Therefore, if we reduce the number of probabilistic decisions in such nodes and at the same time we increase the number of probabilistic decisions taken in nodes with relatively small height, we may obtain a higher value for the gain G. In the new technique that we are going to describe a different partition of the interior nodes of the tree results. In order to be consistent with the previous method we assume, even though it is not necessary, that in the statistical nodes, the branch with the maximum probability is still followed.

In the new method we make a level-order visit of all the interior nodes of the tree starting at the last level. At each node we calculate the total gain of the paths which contain that node by considering it as a statistical one. The gain so obtained is compared with the sum of the gains of the two subtrees originating at the node and if it is greater than the sum, the node becomes a statistical node. We note that the gains of the two subtrees have already been calculated in the previous level of the tree.

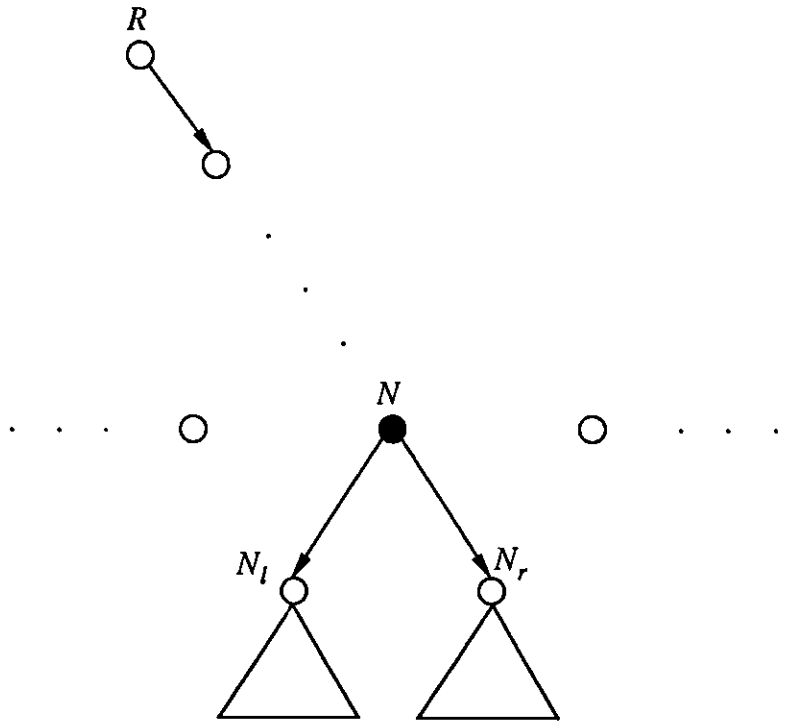

FIGURE 6.8. Illustration of the Alternative Method

The technique can be illustrated in figure 6.8. It is clear from the figure that there is only one path from the root R to the node under consideration, N. If we suppose that the left branch of N has a probability greater than 0.5 then we need to calculate the following gain

$$G_N = \sum_{j \in J} P_{Pj} \cdot G_j$$

where the set J contains all the reachable final nodes which are leaves of the subtree N_l. The G_j terms can be computed using (Eq 4.18). In this expression, the minimum backtracking cost B_j can be derived by following the analysis of the previous chapter. We note that in all these calculations, node N is considered a statistical one. The computed value of G_N is compared with the sum of the gains of the two subtrees N_l and N_r calculated in the same way as in the previous level

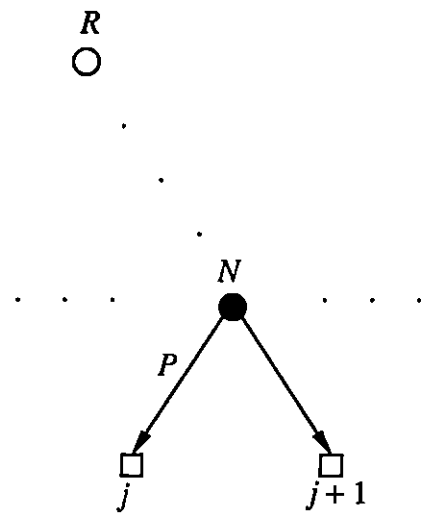

FIGURE 6.9. A Node with Height 1

of the tree. If it is greater than that, N indeed becomes a statistical node. Otherwise, it remains non-statistical and we proceed to the next node of the same level of the tree (if any) or to the first node of the next level.

For the special case of an arbitrary node at the last level of the tree we have the situation shown in figure 6.9. In this figure we assume that $P \geq 0.5$. According to (4.25) the gain associated with the node N is

$$G_N = P_{Pj} \cdot [m_j - B_j \cdot (1 - P_{Cj})] =$$

$$\frac{P_{Pj}}{P} \cdot [1 - (1 - P)] = P_j > 0$$

that is N is converted to a statistical node. In the last expression we consider that $B_j = 1$, even though when the path turns out wrong we know that the path $j+1$ is the correct one and therefore we do not have to perform any test at node N. In general and in order to be consistent with (5.17) we assume that when a path has only one statistical node with height h the backtracking cost associated with that path is h and not h-1.

As the algorithm progresses we always keep track of the current value of the total gain G of the tree. Initially, $G = 0$. When we find an interior node N such that

$$G_N > G_{N_l} + G_{N_r}$$

we add the value of the difference $G_N - (G_{N_l} + G_{N_r})$ to the current value of the total gain, N becomes statistical, and we proceed to the next node.

6.3.2 Application of the Alternative Method

Consider the tree of depth $n = 4$ drawn in figure 6.10. Probabilities that are greater than 0.5 are attached to the corresponding branches. If the alternative method is applied to that tree, the successive values of the gain G appear in table 6.3. The procedure starts at node 15, proceeds from left to right and evaluates the gain by visiting all the nodes of the tree. The value of the gain finally achieved and the corresponding probability of a correct path are $G = 1.327$ and $P_C = 0.560$. The partition of the nodes which results is shown in figure 6.11. If, instead of applying the alternative method, we use the threshold method the maximum gain and the probability P_C would be equal to 1.157 and 0.770, respectively. The state of the tree after applying the threshold method is shown in figure 6.12. The different partitions of the nodes which result from the two methods are obvious from figures 6.11 and 6.12.

6.4 Comparison of the Two Methods

In tables 6.4 and 6.5 we present the results of applying the alternative method for the same trees that appear in tables 1 and 2 respectively. By comparing the corresponding table entries we observe that, in general, the second method performs better in terms of the gain value achieved. This is the case because that method reduces the cost related to the backtracking procedure by preventing statistical decisions from being taken in relatively high levels of the tree. On the other hand, the threshold method usually results in a higher probability

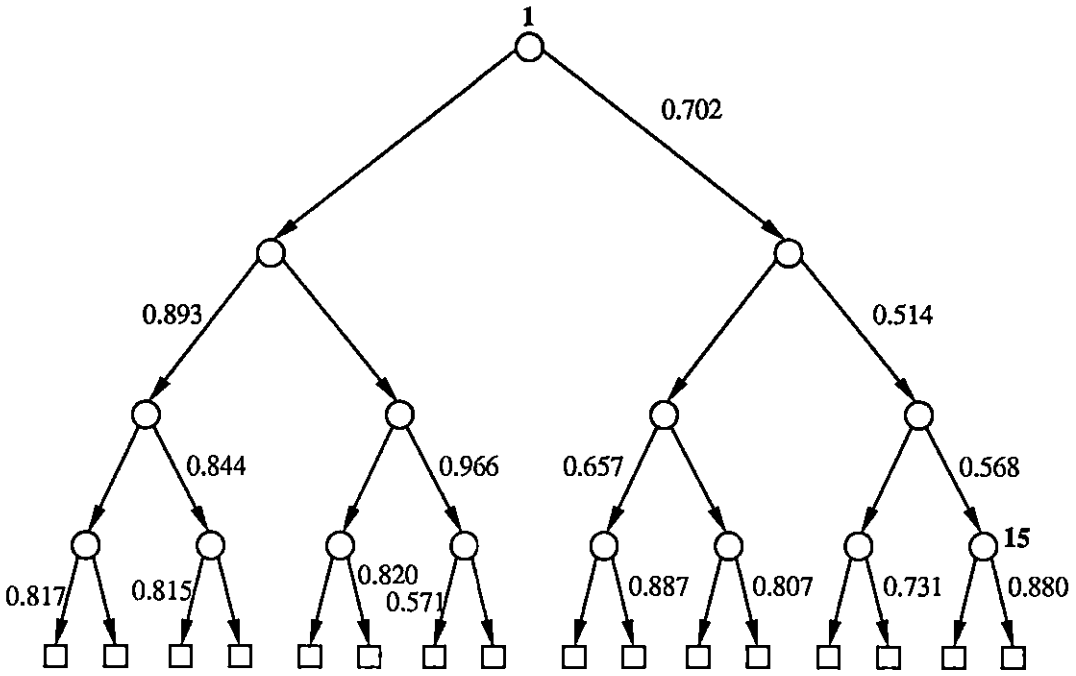

FIGURE 6.10. Example Tree

N	*G*
15	0.180
14	0.294
13	0.389
12	0.587
11	0.605
10	0.789
9	0.823
8	0.889
7	0.993
6	1.022
5	1.022
4	1.172
3	1.172
2	1.327
1	1.327

TABLE 6.3. The Progress of the Alternative Technique

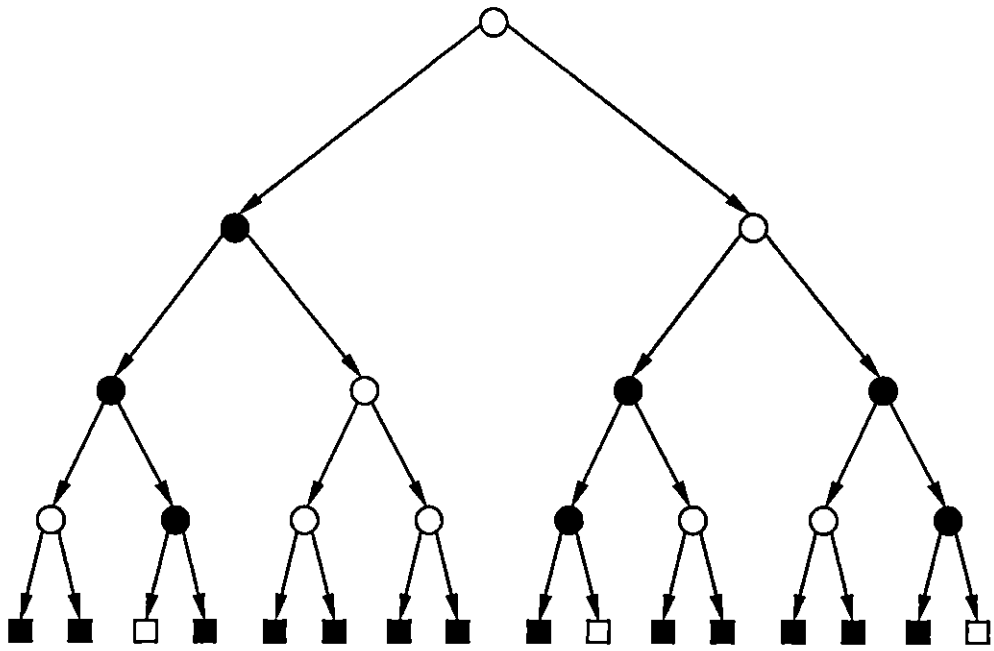

FIGURE 6.11. The state of the Tree after Applying the Alternative Method

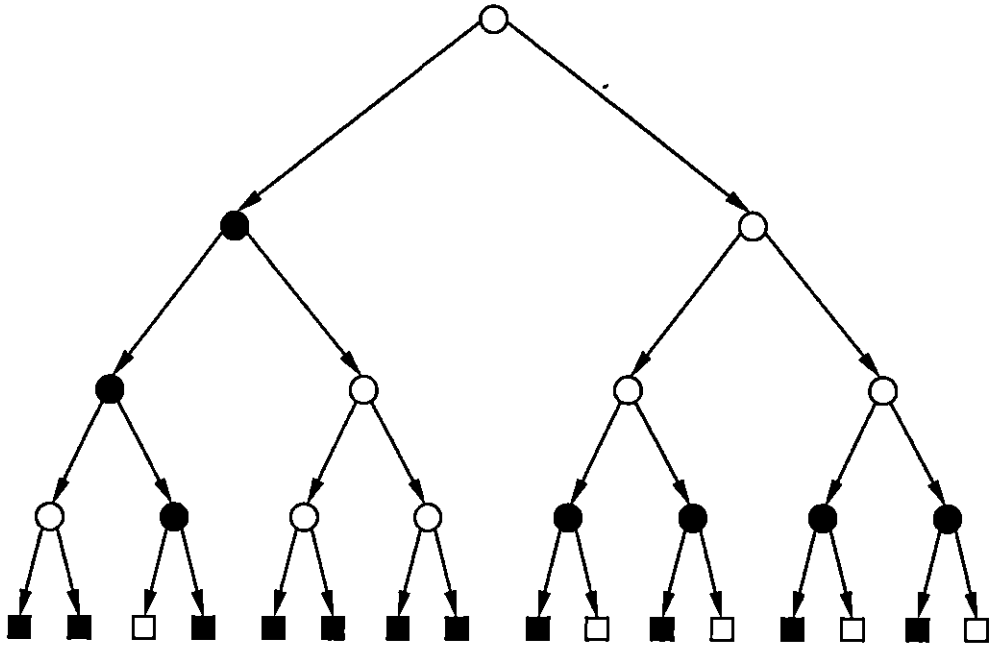

FIGURE 6.12. The state of the Tree after Applying the Threshold Method

P_C. That becomes more obvious if we compare tables 6.2 and 6.5. It has two possible explanations. First, it seems that the number of reachable nodes is higher when the threshold method is used. Moreover, that technique generally ends up with more probable final paths in terms of the probabilities P_j, $j \in F_1$, because the statistical branches that are selected are the most probable ones.

i	G	%	P_C
1	1.328	22.1	0.556
2	2.046	34.1	0.387
3	2.186	36.4	0.450
4	1.716	28.6	0.438
5	1.692	28.2	0.458
6	2.131	35.5	0.600
7	1.931	32.2	0.471
8	1.626	27.1	0.435
9	1.396	23.3	0.488
10	1.890	31.5	0.324

TABLE 6.4. Performance of the Alternative Method for $n = 6$

i	G	%	P_C
1	1.930	19.3	0.465
2	1.917	19.2	0.346
3	2.423	24.2	0.471
4	2.185	21.9	0.456
5	1.932	19.3	0.469
6	2.520	25.2	0.466

TABLE 6.5. Performance of the Alternative Method for $n = 10$

i	G	%	P_C
7	2.202	22.0	0.460
8	2.156	21.5	0.502
9	2.047	20.5	0.487
10	1.828	18.3	0.488

TABLE 6.5. Performance of the Alternative Method for $n = 10$

In table 6.6 the two techniques discussed so far are compared for trees of different depths. In that table $\overline{G}_t, \overline{G}_a$ stand for the average gain of the threshold and the alternative technique respectively. The third and fifth columns contain the corresponding standard deviations. For each value of n, we calculate the gain 5000 times for trees with different uniform distributions of probabilities at the interior nodes. Then the entries of table 6.6 are calculated ([6]). From that table we observe that the alternative method performs better than the threshold method.

n	$\overline{G}_t$	s_t	$\overline{G}_a$	s_a
4	1.586	0.521	1.658	0.472
5	1.660	0.556	1.806	0.483
6	1.691	0.562	1.934	0.467
7	1.700	0.545	2.034	0.448
8	1.696	0.520	2.123	0.427
9	1.688	0.492	2.204	0.408
10	1.672	0.445	2.269	0.382

TABLE 6.6. Performance of the two Techniques for Different Depths of the Tree

6.5 Selection of the Nodes using a Heuristic Function

In the method described in section 6.3 we visit all the nodes starting at the last level of the tree. If there exists an increase of the gain at a particular node, that node becomes a statistical one. In this method the height is considered to be the most important component and therefore it gives priority to nodes of small

height. However, the order in which the nodes are selected could be different. For instance, their selection can be determined according to a heuristic function. That function may depend not only on the height of the node but also on its maximum probability (EQ. 4.5). Thus the node which gives the maximum value of the heuristic function is selected first. Then the procedure discussed in section 6.3 is repeated. That is, the gain is calculated by considering that node as a statistical one. If there is an increase in the gain, the node is indeed converted to a statistical one. Otherwise, it remains non-statistical and we proceed to the node which gives the next highest value of the heuristic function.The algorithm stops after having evaluated the gain function for all the nodes of the tree.

6.5.1 The Heuristic Function

The heuristic function f must depend on both the maximum probability P and the height h. If priority is to be given to nodes of small height and high maximum probability then f can be chosen to be an increasing function of P and at the same time a decreasing function of h. For instance, f could be written as

$$f(h, P) = P \cdot g(h) \qquad \text{(EQ 6.1)}$$

where g is a decreasing function of h and $g(h) \geq 0$, $\forall h = 1, 2, \ldots, n$. For example g can have one of the following expressions

$$g_1(h) = n - h \qquad \text{(EQ 6.2)}$$

$$g_2(h) = \alpha + (\beta + h) \cdot e^{-h} \qquad \text{(EQ 6.3)}$$

where n is the depth of the tree and α, β are positive numbers.

We have applied the method described in this section with different heuristic functions for the trees of table 6.5. The results are summarized in table 6.7. Observe that the alternative method still performs better than any other technique. However, when the function g has the expression given in (Eq. 6.3), β is relatively large and α is small, the results obtained are very close to the ones

appearing in table 6.5. (see third column of table 6.7). This is so because in such a case the nodes are selected in an order that is almost similar to the one used in the alternative method.

In the methods described in the sections 6.3 and 6.5 the gain G is calculated 2^n times. However, we do not need to evaluate all the terms of the summation (4.25) every time. This is explained in Appendix A. In contrast to the threshold method, after the computation of the gain at a particular node, that gain must be also compared with the sum of the gains of the two subtrees originating at that node. Furthermore, in the technique of section 6.5 the complexity is increased because the nodes of the tree need to be ordered according to the value of the heuristic function, before we start evaluating the gain function.

i	G			
	$g_1(h)$	$g_2(h)$		
		$\alpha = 0.3$ $\beta = 0.3$	$\alpha = 0.3$ $\beta = 6$	$\alpha = 4$ $\beta = 6$
1	1.854	1.898	1.930	1.771
2	1.838	1.895	1.917	1.756
3	2.311	2.394	2.423	2.091
4	2.033	2.160	2.180	2.039
5	1.839	1.862	1.921	1.798
6	2.443	2.517	2.519	2.257
7	2.136	2.162	2.200	2.082
8	2.092	2.068	2.156	1.943
9	1.928	2.008	2.045	1.850
10	1.673	1.773	1.826	1.753

TABLE 6.7. The Gain when a Heuristic Function is Used

Chapter 7

Concluding Remarks

7.1 Summary and Conclusions

The main motivation for work is the goal of reducing the number of tests or steps that are required in a specific decision procedure. Namely, we consider that our problem is structured as a binary balanced tree whose interior nodes are decision nodes and whose leaves correspond to distinct final diagnoses. In order to achieve our goal, a probabilistic model has been developed. That was feasible under the assumption that sufficient statistical information is available concerning "local" decisions at the nodes of the tree.

After building up a gain function which expresses the number of decisions that are saved in a single traversal of the tree, our purpose is to obtain its highest possible value. This can be achieved by solving two specific subproblems which determine the parameters of the gain function. First, we must select the interior nodes of the tree at which a decision based on probabilistic information is taken. Then, the cost that arises as a consequence of a wrong final diagnosis, namely the

backtracking cost, has to be minimized. These two problems have been considered separately even though they are closely related. In the first problem, the optimal selection of nodes (that is, the one which gives the maximum gain for the minimum backtracking cost) appears to be computationally complex even for small trees. Therefore, three heuristic methods have been proposed and compared to each other. We have to emphasize that the complexity of the methods is not our main concern. That is because, for a given tree structure, our model is applied once, a particular partition of the nodes is produced and to each reachable final node a specific backtracking procedure is assigned. These facts must be then taken into account when we go through the tree again. Only if, after using the same tree structure many times, the probabilities at the nodes change significantly, should we apply our model again using the new information.

We note that the performance of our techniques depends exclusively on the values of the probabilities and their locations in the tree structure. A negative gain indicates that the probabilistic information should be ignored when we search for a final diagnosis. Our model has been applied to several trees and the results have been presented.

7.2 Future Work

Most of the material presented in the previous chapters can be directly applied to non-balanced binary trees. All the probability related notions (i.e. probability of a correct path, probability of correct backtracking, etc.) remain the same. On the other hand, a slightly different treatment must be given to quantities which are related to path lengths. For instance, the depth of a non-balanced tree which also represents the average number of tests when we go through the tree once, could be written as

$$n = \sum_{j \in F} P_j \cdot n_j$$

where n_j is the length of the path j and P_j its corresponding probability obtained from proposition 4.1. More significant modifications must be made when we are dealing with general tree structures.

Another aspect of our problem that could be investigated is to allow that different weights to be assigned to the tree tests. For instance, in a medical diagnosis problem, it is more costly to perform a c.t. scan or an operation rather than a simple X-ray. Therefore, is it possible to extend our model or develop a similar one in which the decisions of the tree do not have the same importance as has been assumed so far?

In chapter 6, three different techniques for partitioning the interior nodes of the tree were presented and their motivation was given. The performance of our model can probably be improved by considering other methods for dealing with the above problem.

Finally, in our work we made the assumption that there is no cost associated with a wrong final diagnosis. Furthermore we assumed that we are able to distinguish between correct and wrong final outcomes at no cost. These assumptions may not hold in many real decision problems. Extending our model to be able to deal with such situations could be investigated.

Bibliography

[1] R. E. Neapolitan, *Probabilistic Reasoning in Expert Systems*, John Wiley & Sons Inc., 1990.

[2] A. Papoulis, *Probability, Random Variables and Stochastic Processes*, 2nd ed., McGraw-Hill, 1984.

[3] M. Hamburg, *Statistical Analysis for Decision Making*, Harcourt, Brace & World Inc., 1970.

[4] G. Kokolakis, I. Spiliotis, *Probability Theory and Applications*, Athens 1985.

[5] J. R. Blum, J. I. Rasenbalt, *Probability and Statistics*, W. B. Saunders Company, 1972.

[6] J. Pearl, *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufmann, San Mateo, California, 1988.

[7] J. O. Berger, *Statistical Decision Theory and Bayesian Analysis*, 2nd ed., Springer-Verlag, 1985.

[8] G. W. Snedecor, W. G. Cochran, *Statistical Methods*, 7th ed., The Iowa State University Press, 1970.

[9] D. G. Rees, *Foundations of Statistics*, Chapman and Hall Ltd., 1987.

[10] P. E. Lehner, "A Probability Analysis of the Usefulness of Decision Aids", *Uncertainty in Artificial Intelligence*, North-Holland, Amsterdam, pp. 427-436, 1986.

[11] D. J. Spiegelhalter, "A Statistical View of Uncertainty in Expert Systems", *Artificial Intelligence and Statistics*, Addison-Wesley, Reading, Massachusets, pp. 17-55, 1986.

[12] E. H. Shortliffe, *Computer-Based Medical Consultations : MYCIN*, American Elsevier Publishing Company Inc., 1976.

[13] G. Shafer, *A Mathematical Theory of Evidence*, Princeton University Press, 1976.

[14] E. Cox, "Fuzzy Fundamentals", *IEEE Spectrum*, pp. 58-61, October 1992.

[15] C. V. Negoita, *Expert Systems and Fuzzy Systems*, Benjamin/Cummings Pub. Co., 1985.

[16] D. J. Dubois, H. M. Prade, *Fuzzy Sets and Systems : Theory and Applications*, Academic Press, 1980.

[17] B. R. Gaines, "Fuzzy and Probability Uncertainty Logics", *Information and Control 38*, pp. 154-169, 1978.

[18] E. Charniak, "Bayesian Networks without Tears", *AI Magazine*, pp. 50-63, Winter 1992.

[19] J. Pearl, "A Constraint Propagation Approach to Probabilistic Reasoning", *Uncertainty in Artificial Intelligence*, North-Holland, Amsterdam, pp. 357-370, 1986.

[20] J. R. Quinlan, "Decision Trees and Decisionmaking", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 20, no. 2, pp. 339-346, March/April 1990.

[21] C. Carter, J. Catlett, "Assessing Credit Card Applications using Machine Learning", *IEEE Expert*, vol. 2, no. 3, pp. 71-79, Fall 1987.

[22] K. B. Irani, J. Cheng, U. M. Fayyad, Z. Qian, "Applying Machine Learning to Semiconductor Manufacturing", *IEEE Expert*, vol. 8, no. 1, pp. 41-47, February 1993.

[23] J. R. Quinlan, "Decision Trees as Probabilistic Classifiers" *Proc. Fourth Int. Workshop on Machine Learning*, P. Langley, Ed. Los Altos, CA : Morgan Kaufmann, 1987.

[24] K. T. Deschler, *Cable Television Technology*, McGraw-Hill Inc., 1987.

[25] S. Neville, "A Prototype Expert System Based Diagnostic Tool for Cable Trunk Amplifier Networks", Masters Thesis, University of Victoria, 1992.

[26] N. J. Dimopoulos, K. F. Li, A. Watkins, S. Neville, A. Rontogiannis, "An Expert Network Analyser", *Technical Papers : Canadian Cable Television*

Association 35th Annual Convention, Vancouver, B.C., pp. 123-127, May 31-June 3, 1992.

[27] C-COR Electronics Inc., *Preliminary Instruction Manual for the B-507 Remote Bridger*, Rev. 0, July 1983.

[28] H. Maisel, G. Gnugnoli, *Simulation of Discrete Stochastic Systems*, Science Research Associates Inc., 1972.

[29] F. S. Hillier, G. J. Lieberman, *Operations Research*, 2nd Edition, Holden-Day Inc., 1974.

Appendix A

As mentioned in section 6.2.2, during the exhaustive search for the maximum of G, it is not required that we compute all terms of the summation (4.25) in each interval (see figure 6.1). This is because, as we move from one interval to the next starting from the last one, some terms of the summation (4.25) do not need to be evaluated; these are the terms associated with the paths in F_1 which contain the branch with the probability p_k we just evaluated. At the same time some final nodes become unreachable : these are the final nodes of the paths which contain the branch with the probability $1-p_k$ and which were reachable before. We will prove the following

Proposition 1. *The number of term evaluations R_n to find the maximum of G is bounded by*

$$n \leq R_n \leq n \cdot 2^{n-1} \tag{1}$$

for a tree of depth n and an arbitrary distribution of probabilities at the nodes of the tree.

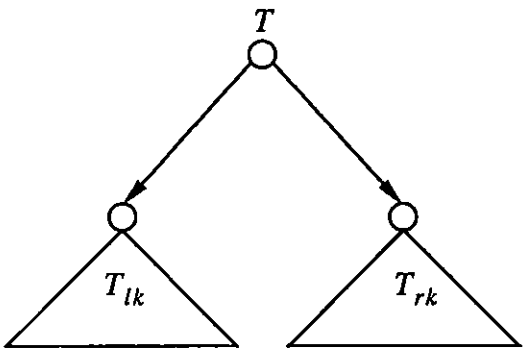

FIGURE 1. Tree of depth $k + 1$

Proof. We will prove the second inequality by induction on the depth of the tree n. For $n = 1$ we have $R_1 = 1$ and (1) holds because $R_1 \leq 1$. Let us suppose that for $n = k$, (Eq. 1) is correct. That is for every distribution of probabilities and a tree of depth k, we assume that the number of term evaluations is bounded as follows

$$R_k \leq k \cdot 2^{k-1}$$

We will prove the follwing

$$R_{k+1} \leq (k+1) \cdot 2^k$$

Consider the binary tree of figure 1. The tree has depth $k+1$ and each of the subtrees T_{lk} and T_{rk} has depth k. In general the number of term evaluations R_{k+1} can be expressed as

$$R_{k+1} = R_{lk} + R_{rk} + R_{root}$$

where R_{lk} and R_{rk} are the number of term evaluations because of the probabilities of the left and the right subtree respectively and R_{root} is the number of term eval-

uations because of the probability of the root. From the induction hypothesis we know that

$$R_{lk} \leq k \cdot 2^{k-1}$$

$$R_{rk} \leq k \cdot 2^{k-1}$$

Furthermore

$$R_{root} \leq 2^k$$

because the branch with the maximum probability of the root (see Eq. 4.5) can belong at most to 2^k paths. From the last four expressions we get

$$R_{k+1} \leq k \cdot 2^{k-1} + k \cdot 2^{k-1} + 2^k = k \cdot 2^k + 2^k \Rightarrow$$

$$R_{k+1} \leq (k+1) \cdot 2^k$$

The first inequality holds because there is at least one reachable final node (path K). Thus when we meet the probabilities of this path, at least one evaluation occurs for each probability. Since the number of these probabilities in the interval $[0.5, 1]$ is n we have

$$n \leq R_n$$ ■

The actual number of term evaluations depends on the positions of the probabilities on the branches of the tree. The best case corresponding to the lower bound of (1), appears when the probabilities of the path K are the highest ones and are located in increasing order from the root to the leaf node. The upper limit of (1) occurs when we have the maximum number of evaluations in both subtrees of depth $n-1$ and the probability of the root is less than the minimum probability of the one subtree and greater than the maximum probability of the other subtree. The maximum number of evaluations in the two subtrees is defined recursively. A tree of depth 4 which will require the maximum number of term evaluations is shown in figure 2. The numbered branches are the branches the probabilities of

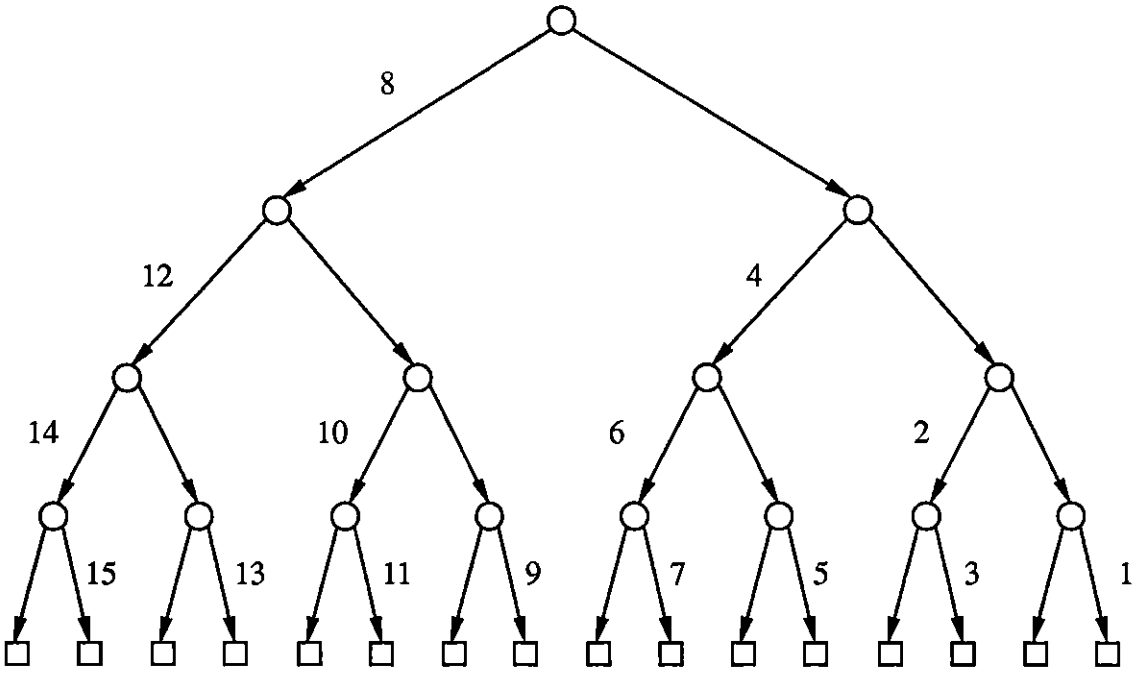

FIGURE 2. The Worst Case in a Tree of Depth 4

which are greater than 0.5. Number 1 corresponds to the highest probability and number 15 to the lowest. According to (1), 32 term evaluations are required during the exhaustive search procedure in this case.

VITA

Surname: Rontogiannis Given Names: Athanasios

Place of Birth: Athens (Greece) Date of Birth: 16 June 1968

Educational Institutions Attended:

University of Victoria	1991 to 1993
National Technical University of Athens	1986 to 1991

Degrees Awarded:

Ptychion	National Technical University of Athens	1991

Honours and Awards:

Technical Chamber of Greece Award	1991

Publications:

N.J.Dimopoulos, K.F.Li, A.Watkins, S.Neville, A.Rontogiannis, "*An Expert Network Analyser*", Technical Papers : Canadian Cable Television Association, 35th Annual Convention, Vancouver, B.C., May 31 - June 3, 1992, pp. 123-127.

A.Rontogiannis, N.J.Dimopoulos, "*A Probabilistic Approach for Reducing the Search Cost in Binary Decision Trees*", (to be presented), IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, Victoria, B.C., May 19th to 21th 1993.

PARTIAL COPYRIGHT LICENSE

I hereby grant the right to lend my thesis to the users of the University of Victoria Library, and to make single copies only for such users or in response to a request of the Library of any other university, or similar institution, on its behalf or for one of each users. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by me or a member of the University designated by me. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Title of Thesis: A Probabilistic Approach for Reducing the Search Cost in Binary Decision Trees

Author [redacted]
(Signature)

ATHANASIOS RONTOGIANNIS
(Name in Block Letters)

MAY 8, 1993
(Date)