Hierarchical Testing Designs for Pattern Recognition

Gilles Blanchard * Donald Geman †

Abstract

We explore the theoretical foundations of a "twenty questions" approach to pattern recognition. The object of analysis is the computational process itself rather than probability distributions (Bayesian inference) or decision boundaries (statistical learning). Our formulation is motivated by applications to scene interpretation in which there are a great many possible explanations for the data, one ("background") is statistically dominant, and it is imperative to restrict intensive computation to genuinely ambiguous regions.

The focus here is on detection: Given a large set $\mathcal{Y}$ of possible patterns or explanations, narrow down the true one Y to a small (random) subset $\widehat{Y} \subset \mathcal{Y}$ of "detected" patterns to be subjected to further, more intense, processing. To this end, we consider a family of hypothesis tests for $Y \in A$ versus the non-specific alternatives $Y \in A^c$. Each test has null type I error and the candidate sets $A \subset \mathcal{Y}$ are arranged in a hierarchy of nested partitions. These tests are then characterized by scope ($|A|$), power (type II error) and algorithmic cost.

We consider sequential testing strategies in which decisions are made iteratively, based on past outcomes, about which test to perform next and when to stop testing. The set $\widehat{Y}$ is then taken to be the set of patterns that have not been ruled out by the tests performed. The total cost of a strategy is the sum of the "testing cost" and the "postprocessing cost" (proportional to $|\widehat{Y}|$) and the corresponding optimization problem is analyzed. As might be expected, under mild assumptions good designs for sequential testing strategies exhibit a steady progression from broad scope coupled with low power to high power coupled with dedication to specific explanations. In the assumptions ensuring this property a key role is played by the ratio cost/power. These ideas are illustrated in the context of detecting rectangles amidst clutter.

*CNRS (France) and Laboratoire de Mathématiques, Univerité Paris-Sud, 91405 Orsay Cedex, France. Email: Gilles.Blanchard@math.u-psud.fr.

†Department of Mathematical Sciences, Johns Hopkins University, Baltimore, MD 21218. Email:geman@cis.jhu.edu. Supported in part by ONR under contract N000120210053, ARO under grant DAAD19-02-1-0337, and NSF ITR DMS-0219016.

Contents

1 Introduction **4**

2 Organization of the Paper **8**

3 Overview of Results **9**

4 Previous Work **11**
4.1 Decision trees . . . 11
4.2 Pattern recognition . . . 13

5 Problem Formulation **15**
5.1 Goals . . . 15
5.2 Attributes and attribute tests . . . 16
5.3 Test hierarchies . . . 18
5.3.1 Fixed hierarchy . . . 18
5.3.2 Extended hierarchy . . . 19
5.4 The probabilistic model . . . 19
5.5 Testing strategies and their cost . . . 20
5.5.1 Cost of testing . . . 20
5.5.2 Cost of postprocessing . . . 21
5.5.3 Optimization problem . . . 22
5.5.4 Equivalent model with perfect tests . . . 22
5.6 Cost of a test . . . 23
5.6.1 Power-based cost . . . 23
5.6.2 Usage-based cost . . . 24
5.7 Special strategies . . . 25

6 Optimal Strategies for Fixed Costs and Powers **29**
6.1 Reformulation of the cost . . . 29
6.2 Detecting one special pattern . . . 31
6.3 Detecting any pattern . . . 32
6.4 Simulations with an elementary dependency model . . . 37

7 Optimal Strategies for Power-Based Cost and Variable Powers **38**
7.1 Model and motivations . . . 38
7.2 Basic results . . . 40

7.2.1 Optimal power selection . . . 41
7.2.2 Properties of the CTF strategy . . . 42
7.3 Is the CTF strategy optimal? . . . 44
7.4 CTF optimality for the harmonic cost function . . . 45
7.5 Simulations . . . 48

8 Optimal Strategies for Usage-Based Cost 51
8.1 Detecting one special pattern . . . 52
8.2 Detecting any pattern . . . 53

9 Extended Scenario: Multiple Searches 56
9.1 Background model . . . 57
9.2 Hierarchies . . . 57
9.3 Optimal testing strategies: usage-based cost . . . 58
9.3.1 Randomized patterns . . . 58
9.3.2 Randomized hierarchies . . . 59

10 Application: Rectangle Detection 60
10.1 Problem formulation . . . 61
10.2 Patterns and attributes . . . 62
10.3 Tests . . . 63
10.4 Detection results . . . 65

11 Discussion and Conclusion 68

A Proofs for Section 7 69

B Proofs for Section 9 72

1 Introduction

Motivated by problems in machine perception, specifically scene interpretation, we investigate the theoretical foundations of an approach to pattern recognition based on adaptive sequential testing. The basic scenario is familiar to everybody – identify one "pattern" (or "explanation") from among many by posing a sequence of subset questions. In other words, play a game of "twenty questions." Intuitively, we should ask more and more precise questions, progressing from general ones which "cover" many explanations, but are therefore not very discriminating, to those which are highly dedicated and decisive. Although the efficiency of coarse-to-fine (CTF) search drives the design of codes and many numerical routines, there has been surprisingly little work of a theoretical nature outside information theory to understand why this strategy is advantageous. We explore this question within the framework of sequential hypothesis testing, putting the emphasis on the modeling and optimization of computational cost: *In what sense and under what assumptions are the strategies which minimize total computation CTF?*

Needless to say, in order to have a feasible formulation of the problem one must make specific assumptions about the structure of the available tests (or "questions"). In this paper, we will therefore consider a particular structure based on an *a priori* multiresolution representation for the individual patterns and a corresponding hierarchy of hypothesis tests. Other important assumptions concern the statistical distribution of the tests and how cost varies with scope and power.

Our formulation is influenced by applications to pattern recognition, although we believe it remains sensible for other complex search tasks and we would argue that computational efficiency and CTF search are linked in a fundamental way. In both natural and artificial systems, many tasks do not require immediate, complete explanations of the input data. Nonetheless, the usual approach to machine perception is static: Intermediate results, when they exist, generally do not provide clear and useful provisional explanations. In contrast, we consider a sequence of increasingly precise interpretations (subsets of patterns), noting that experiments in biological vision (e.g., studies on "pop-out") report evidence for graded interpretations, e.g., very fast identification of visual categories (Thorpe, Fize & Marlot 1996) and "regions of interest" (Desimone, Miller, Chelazzi & Lueschow 1995) ("visual selection").

Our formulation is also influenced by what we perceive to be some fundamental limitations in purely learning-based methods in pattern recognition in spite of recent advances (e.g., multiple classifiers, boosting and theoretical bounds on generalization error). We do not believe that very complex problems in machine perception, such as full-scale scene interpretation, will yield directly to improved methods of statistical learning. Some organizational framework is needed to confront the sheer number of explanations and complexity of the data. (See e.g. the discussion in (Geman, Bienenstock & Doursat 1992).) In our approach learning comes into play in actually constructing the individual hypothesis tests from training data; in other words, one learns the individual components of an overall design.

The hypothesis-testing framework is as follows. Consider many patterns (or pattern classes) $y \in \mathcal{Y}$ as well as a special, dominating class 0 which represents "background." There is one true state $Y \in \{0\} \cup \mathcal{Y}$. In the highlighted applications, Y refers to a semantic explanation of image data, for instance the names and poses (geometrical presentations) of members belonging to a repertoire of actual objects appearing in an image. The explanation $Y = 0$ represents "no pattern of interest" and is exceedingly more likely a priori; class 0 is also exceedingly more varied. Ultimately, we want to determine Y (*classification* or *identification*). Ideally, this task would be accomplished rapidly and without error.

However, in machine perception, and many other domains, near-perfect classification is often very difficult, even with sizeable computational resources, and virtually impossible without resorting to a "contextual analysis" of competing explanations. In other words, we eventually need to test precise hypotheses $Y \in A$ against precise alternatives $Y \in B$, where $A, B \subset \mathcal{Y}$ (*"Is it an apple or a pear?"*). In view of the large number of possible explanations, it is not computationally feasible to anticipate all such scenarios. This argues for starting, and going as far as one can, with a "noncontextual analysis," meaning testing the hypothesis $Y \in A$ against the nonspecific alternative, $Y \notin A$ (or, what is often almost the same, against the background alternative, $Y = 0$) for a distinguished family of subsets $A \subset \mathcal{Y}$. Of course this only makes sense if there are *natural groupings* of explanations, which is certainly the case for pattern recognition (e.g., involving real objects and their spatial presentations).

Let X_A denote the result of such a test, with $X_A = 1$ (resp. $X_A = 0$) indicating acceptance (resp. rejection). Indeed, it then makes sense to construct a family $\mathcal{X}$ of such tests *in advance*, say of order $\mathcal{O}(|\mathcal{Y}|)$. Throughout the paper, we assume that the family $\mathcal{A}$ of sets $A \subset \mathcal{Y}$ for which (noncontextual) tests are built has a hierarchical, nested cell structure. These sets will be called *attributes*, their depth in the cell hierarchy referred to as their *resolution level* and their cardinality as their *scope*. In this scheme, the contextual analysis – testing against specific alternatives – begins only after the number of candidate explanations is greatly reduced, at which point tests may be created *on-line* to address the specific ambiguities encountered.

To pin thing down, consider a toy example: Suppose $\mathcal{Y} = \{A, P, O\}$, standing for *Apple*, *Pear*, *Other*, and the most likely explanation is $Y = O$. Suppose also there are four "tests":

- $X_{A,P}$ for testing $Y \in \{A, P\}$ vs. $Y = O$ (something like *"Is it a fruit ?"*);
- X_A (resp., X_P) for testing $Y = A$ vs. $Y = O$ (resp., $Y = P$ vs. $Y = O$);
- X_{AvP} for testing $Y = A$ vs. $Y = P$.

Tests $X_{A,P}, X_A, X_P$ are "noncontextual"; X_{AvP} is "contextual". Suppose all noncontextual tests have null false negative error. The type of CTF strategy that typically emerges from minimizing the "cost" of determining Y under natural assumptions about how cost, scope and error are balanced is the intuitively obvious one: Perform $X_{A,P}$ first; then, if the result is positive ($X_{A,P} = 1$), perform X_A and X_P; finally, perform X_{AvP} if both the previous results are again positive.

In this paper we consider efficient designs for the noncontextual phase only; the full problem, including contextual disambiguation, will be analyzed elsewhere. However, we anticipate the complexity of this contextual analysis by incorporating into our measurement of computation a "post-processing" penalty which is proportional to the number of remaining explanations.

Our objective, then, is efficient pattern "detection". The reduced set of explanations, denoted by $\widehat{Y}$ and called the set of *detected patterns*, is a *random subset* of $\mathcal{Y}$ that also depends on the chosen *strategy*, i.e. the sequence of tests chosen to be performed. The tests are performed sequentially, and the choice of the next test to perform (or the decision to stop the search) depends on the outcomes of the past tests and is prescribed by the strategy. If strategy T has performed the tests $X_{A_1}, \ldots, X_{A_k}$ before terminating (note that k and $A_2, \ldots, A_k$ are themselves random variables), then set of detected patterns is determined in a simple way from the outcomes of the tests: $\widehat{Y}(T)$ consists of all patterns $y \in \mathcal{Y}$ which are "accepted" by every test X_{A_i} for which $y \in A_i, 1 \leq i \leq k$. In other words, a pattern is said to be detected if it is not ruled out by one of the tests performed.

The fundamental constraint is no missed detections:

$$P(Y \in \widehat{Y} \cup \{0\}) = 1.$$

This condition is satisfied if each individual test X_A has zero type I error, and we make this assumption about every test X_A, recognizing that we must pay for it in terms of cost and power (or equivalently type II error). Although we shall not be explicitly concerned with standard estimators such as

$$\widehat{Y}_{MLE}(\mathcal{X}) = \arg\max_y P(\mathcal{X}|Y = y) \ \ \textit{and} \ \ \widehat{Y}_{MAP}(\mathcal{X}) = \arg\max_y P(Y = y|\mathcal{X}),$$

or even formulate a prior distribution for Y, it then follows that

$$P(\widehat{Y}_{MLE} \in \widehat{Y} \cup \{0\}) = P(\widehat{Y}_{MAP} \in \widehat{Y} \cup \{0\}) = 1.$$

Tests $X_A \in \mathcal{X}$ are then characterized by their scope ($|A|$), power (type II error) and computational cost, and certain fundamental tradeoffs are assumed to hold among these quantities. A fundamental assumption is that mean computation is well-approximated by conditioning on $Y = 0$, and that, in this case, the tests are conditionally independent. In order to accommodate differing applications and establish general principles, we will consider several scenarios, including both "fixed" and "variable" powers and two models – "power-based" and "usage-based" – for how the cost of a test is determined. Except for a concluding illustration, we do not consider how these hypothesis tests X_A are actually constructed, i.e., depend functionally on the raw data In the applications cited in §4 this typically involves statistical learning, for instance inducing a decision tree or support vector machine from positive ($Y \in A$) and negative ($Y \notin A$) examples. *We are designing the specifications rather than the tests themselves, and modeling the computational process rather than learning decision boundaries for classification.* Presumably standard techniques can be used to build tests to the desired specifications if the tradeoffs are reasonable. In Section 10 we will provide one recipe in an image analysis framework.

Although we will assume throughout that the true Y is a single pattern belonging to $\{0\} \cup \mathcal{Y}$, our analysis would remain valid if we allowed Y to be an entire subset of patterns $Y \subset \mathcal{Y}$ (with $Y = \emptyset$ representing "no pattern of interest" or "background"). In this case, X_A would test the hypothesis $Y \cap A \neq \emptyset$ against $Y \cap A = \emptyset$, or against the nonspecific alternative $Y = \emptyset$. This setting might be more useful in some applications, such as scene interpretation, although in the end these subsets are simply more complex individual explanations.

Finally, besides pattern detection, two natural subtasks are:

- *Background-Pattern Separation:* Determine if $Y \neq 0$;
- *Single-Pattern Recognition:* Determine if $Y = y^*$ for some distinguished $y^* \in \mathcal{Y}$.

(We separate these due to the special role of the background class.) Again, we only entertain testing non-specific alternatives, leading to an analogous $\widehat{Y}$. What distinguishes these tasks from pattern detection is the postprocessing cost: For pattern detection, it will be taken proportional to the cardinality of $\widehat{Y}$, whereas for background-pattern separation (resp. single-pattern recognition), it is a fixed constant if $\widehat{Y} \neq \emptyset$ (resp. $y^* \in \widehat{Y}$) and zero otherwise. In each case, strategies are ranked according to the amount of total computation. Background-pattern separation will not be analyzed in this paper since it has been studied elsewhere in a very similar framework. In contrast, detecting a single pattern of interest will often serve as a first step before turning to the detection of all possible patterns.

2 Organization of the Paper

In §3 we provide a non-technical overview of the results obtained in the paper, and in §4 we review some previous work on related problems, including a few previous applications of this methodology to scene interpretation in order to see how all this plays out in practice.

The precise mathematical setup is presented in §5, which ends with a summary of our notation. Our principal results appear in §6-9. In most of these sections, we deal first with detecting a special pattern of interest and then with the general problem of detecting any pattern. In §6, we consider the simplest case: There is one single test X_A of fixed power and cost for each attribute $A \in \mathcal{A}$, and we present a fairly general sufficient condition under which CTF strategies are optimal. The "extended hierarchy" is examined in §7, namely a whole family of tests $(X_{A,\beta})$ for each attribute $A \in \mathcal{A}$ indexed by their power β. Assuming the cost of a test depends in a natural way on its scope and power, we study testing strategies under the restriction that each attribute can only be tested once. We first give general results concerning optimal choice of powers and properties of the CTF strategies. We also demonstrate the optimality of the CTF procedure for a particular form of the cost function. As these results are decidedly not comprehensive, we attempt to strengthen the case for the "optimality" of CTF search with a variety of simulations at the end of §7.

In §8-9, we present analytical results for a substantially different cost model. We again consider a fixed-power hierarchy, but the cost of a test is not fixed in advance; instead, it depends on the resources devoted to this test, which can be chosen in accordance with the strategy employed but subject to a global resource limit. As a consequence, the cost of a test then depends on the frequency with which it is used. In §9, we argue that it makes sense in this latter framework to consider an extended scenario where repeated search tasks are undertaken for different sets of target patterns, whereas the resources are distributed in advance among all tests.

In order to illustrate in a controlled setting the quantities which figure in our analysis, especially how computation is measured, we sketch an algorithm in §10 for a synthetic example of detecting rectangles in images against a background of "clutter". Finally, in §11, discuss the results obtained, and some conclusions and directions for future research.

3 Overview of Results

To keep things simple, and consistent with our main results, we shall focus here on detecting any pattern; similar results hold for a single pattern of interest.

A strategy T can be represented as a binary tree with a test $X \in \mathcal{X}$ at each internal node and a subset $\widehat{Y}(t)$ at each external node or leaf t. The computation due to testing, $C_{test}(T)$, is a random variable – the sum of the costs of the tests performed before $\widehat{Y}$ is determined. The mean cost is then the average over all tests $X \in \mathcal{X}$ of the cost of X weighted by the probability that X is performed in T; these quantities will be defined more carefully in §5.

In anticipation of resolving the ambiguities in $\widehat{Y}$ in order to determine Y, we add to the mean testing cost a quantity which reflects the postprocessing cost, taken simply as $C_{post}(\widehat{Y}(T)) = c^*|\widehat{Y}(T)|$, where c^* is a constant called the unit postprocessing cost. This charge may also be (formally) interpreted as the cost of performing perfect, albeit costly, tests for each individual non-background explanation in $\widehat{Y}$ in order to remove any remaining error under the background hypothesis (i.e., render $P(\widehat{Y} = \emptyset | Y = 0) = 1$). The constant c^* then represents the cost of a perfect individual test. Again, all tests have null false negative error, so "perfect" refers to full power.

The natural optimization question is then to find the strategy T^* which minimizes the mean total computation:

$$T^* = \arg\min_{T} EC(T), \quad C(T) = C_{test}(T) + C_{post}(\widehat{Y}(T)).$$

We are particularly interested in determining when T^* is CTF in scope (meaning scope is decreasing along any root-to-leaf branch) and CTF in power (meaning power increases as scope decreases). Informally, the assumptions we impose are:

- *A multiresolution, nested cell representation:* The family of attributes $\mathcal{A}$ has the structure of a tree (see e.g. Figure 3, below).
- *Background domination:* Mean computation $EC(T)$ and power $P(X_A = 0 | Y \notin A)$ are well-approximated by taking $P = P_0 = P(.|Y = 0)$.
- *Conditional independence:* Under P_0 families of tests over distinct attributes are independent. This is the strongest assumption and the one most likely to be violated in practice.

In the case of a fixed-powers hierarchy considered in §6, we assume that the test for attribute A has cost $c(A)$ and power $\beta(A)$. We show that the ratios $c(A)/\beta(A)$ play a crucial role in the analysis of the optimization problem, and give the following general sufficient condition: *CTF optimality holds whenever, for any attribute A, the ratio of cost to power is less than the sum of the corresponding ratios over all direct children of A in the test hierarchy* (including if necessary the perfect tests representing the postprocessing cost, having cost c^* and power 1).

In the case of an extended hierarchy (§7), we consider a multiplicative model for the cost of $X_{A,\beta}$:

$$c(X_{A,\beta}) = \Gamma(|A|) \times \Psi(\beta),$$

where Γ is subadditive and Ψ is convex. We prove that the CTF strategies always perform a specific test with the same power and that this power does not depend of the particular CTF strategy. A rigorous result about CTF optimality is only obtained for one particular Ψ, but simulations strongly indicate that the observed behavior is more widely true. In summary, CTF strategies seem to be optimal for a wide range of situations.

In §8-9, we study a somewhat different framework; we consider only the case of a fixed-powers hierarchy, but the cost of a test depends of the "resource" allocated to it (through a negative exponential function) and there is a global resource constraint. In this framework the optimal resource allocation gives rise to a *usage-based* cost; the cheapest tests are the ones used the most often in a given strategy. No postprocessing cost is taken into account, but we suppose that $\widehat{Y}$ results in the minimum postprocessing burden, namely the set of detected patterns that would be obtained if all of the tests were performed at once. In §8 we prove that for a dyadic hierarchy for which the powers are increasing with the resolution level, the CTF strategy is optimal if we assume that all tests have power greater than some constant $\beta_1 = 7/8$.

In §9, we propose an extended scenario that is especially relevant with usage-based cost, namely repeated detection tasks for which $\mathcal{Y}$ changes from task to task but the attribute tests are reusable. The patterns are identified with conjunctions of abstract attributes at different resolution levels, taken from a possibly very large pool. Whereas the analysis in the fixed-cost model remains unchanged, there is a significant difference under usage-based cost since we must distribute the resources over a larger number of tests. In order to simplify the analysis, we suppose the set of target patterns $\mathcal{Y}$ is randomized and again present some fairly mild sufficient conditions (about the dependence of power on resolution and the size of the attribute pools) ensuring the optimality of CTF strategies.

4 Previous Work

Our work is a natural outgrowth of an ongoing project on scene analysis, especially object recognition and largely of an algorithmic nature (see e.g., (Amit & Geman 1999)). The current objective is to explore a suitable mathematical foundation. This was begun in (Fleuret 2000) and (Fleuret & Geman 2001) where the computational complexity of traversing abstract hierarchies was analyzed in the context of *purely* power-based cost – assuming that cost is an increasing, convex function of power. (The model here is more realistic because cost depends on scope as well as power.) It was continued in (Jung 2001), in which the optimality of depth-first CTF search for background/pattern separation (checking if $\widehat{Y} = \emptyset$) was established under the same model.

4.1 Decision trees

Of course "twenty questions," and the search strategies T discussed in the introduction for a fixed family $\mathcal{X}$ of binary tests, invoke decision (or classification) trees – adaptive procedures for discriminating amongst hypotheses based on sequential testing.

Early work was inspired by applications to fault-testing and medical diagnosis and involves errorless binary questions of varying costs; see e.g. (Garey 1972) and the references therein. Collectively, the tests determine the true hypothesis and vice-versa (i.e., the tests are conditionally degenerate). The goal then is to find the minimal-cost tree for identifying the true hypothesis, a very diffuclt combinatorial optimization problem.

Most of the literature on decision trees is about an inductive framework with nonperfect tests. Trees are induced from a training set of i.i.d. samples of $(\mathbf{X}, Y)$, where $\mathbf{X}$ is a measurement or "feature" vector and the binary tests result from comparing one component of $\mathbf{X}$ to a threshold. A tree T_{loc} is built in a top-down, greedy, recursive fashion based on some splitting criterion, usually entropy reduction (Breiman, Friedman, Olshen & Stone 1984): First the root is assigned a test, then each child of the root, and so forth until a stopping rule is enforced. The construction is then *data-driven* and *locally optimized*, guided by uncertainty reduction. There is a large literature on application of decision trees to pattern recognition which is outside the scope of this paper; see (Amit 2002). In (Amit & Geman 1999), faces were detected using multiple decision trees and a version of our division into "noncontextual" and "contextual" was proposed.

Generally, efficient (online) execution is not a criterion for construction or performance; for instance, the CART algorithm doesn't account for mean path length, let alone "costs" for the tests. Not surprisingly, recursive greedy designs are often globally inefficient, for instance in terms of the mean depth necessary to reach a given classification rate. A rarely studied alternative is to begin with an explicit statistical model for $(\mathbf{X}, Y)$ and compute a tree T_{glo} according to a global criterion involving *both* accuracy *and* (online) computation. The construction is then *model-driven* and *globally optimized.* Our approach to calculating $\widehat{Y}$ is of this general nature. Such a framework can also be found in (Trouvé & Yu 2002), motivated by query-driven retrieval algorithms for very large databases, where the number of queries performed is the quantity to be minimized under the constraint of exact retrieval. In (Jung 2001), the depth-first CTF strategy for background/pattern separation was compared with vanilla CART, i.e., recursive splitting driven by uncertainty reduction. In general, the strategy resulting from CART is *not* CTF; in particular doing the coarsest test first may result in a poor entropy drop and the cost of the CART tree is naturally far from optimal.

Despite well-known, if scattered, evidence about the superiority of global strategies, calculating T_{glo} is simply computationally prohibitive, whether model-driven or data-driven, and the literature is correspondingly sparse. A notable (if unrealistic) exception is when i) the tests conditionally independent given Y; ii) the cost of a tree is a linear combination of the average terminal entropy and the average depth; and iii) the tests are "repeatable" – there is an unlimited number of independent "copies" of each test and hence the same one (in distribution) may appear several times along the same branch of T. Computing T_{glo} is then sometimes feasible (although intensive) because the optimal test to perform at any (interior) node is entirely determined by the depth of the node and the conditional distribution of Y at the node. In other words, the posterior distribution on Y is a "sufficient statistic" for the node history. Consequently, T_{glo} can then be constructed from dynamic programming and variants thereof for "small problems." But backwards recursion fails without repeatability, an assumption which is at best very dubious in practice.

Some of these observations can be traced back to (DeGroot 1970), where the setting is precisely i)-iii) above, sitting at the intersection of sequential statistics (Chernoff 1972), game theory (Blackwell & Girschick 1954) and adaptive control processes (Bellman 1961). Usually, the emphasis is on asymptotic results, for instance as mean tree depth grows. In (Geman & Jedynak 2001), some comparisons in accuracy (resp. mean depth) between T_{loc} and T_{glo} are given at a fixed mean depth (resp. accuracy), revealing an enormous difference in favor of T_{glo}, especially with skewed priors, i.e., when *a priori* some classes are much more likely than others.

Figure 1: Left: a "natural" image. Right: group photograph used in an experiment of face detection.

4.2 Pattern recognition

Consider the scenes in Figure 1. The semantic interpretation of the left image (town, shops, pedestrians, etc.) is effortless for humans but far beyond what any artificial system can do. For the image on the right, the goal might be more modest – detect and localize the faces. Enriching the description with information about the precise pose (scale, orientation, etc.), identities or expressions would be more ambitious. Many methods have been proposed for face detection, including artificial neural networks (Rowley, Baluja & Kanade 1998), Gaussian models (Sung & Poggio 1998), support vector machines (Osuna, Freund & Girosi 1997), Bayesian inference (Cootes & Taylor 1996) and deformable templates (Yuille, Cohen & Hallinan 1992).

To relate these tasks to the framework of this paper, imagine attempting to characterize a (randomly selected) subimage containing at most one object from a predetermined repertoire. (The whole scene can then be searched by a divide-and-conquer strategy; see §10 and (Dietterich 2000).) The dominating explanation $Y = 0$ corresponds to "background" or "clutter" and each of the others, $Y \in \mathcal{Y}$, to the instantiation of an object wholly visible in the subimage. Even with only one (generic) object class, the number of possible instantiations is very large, i.e., there is still considerable within-class variability. For instance, detecting a face at a fixed position, scale and orientation might not be terribly difficult, even given variations in lighting and nonlinear variations due to expressions; it can be accomplished with standard learning algorithms such as multilayer perceptrons, decision trees and support vector machines. However, the amount of computation required to do this separately for every possible pose is prohibitive. Instead, we propose to search simultaneously for many instantiations, say over a range of locations, scales and orientations. In our simplified mathematical analysis, that range of poses is $A = \mathcal{Y}$, which is the "scope" of our coarsest test X_A. (It may not be practical to envision a totally invariant test, in which case there are multiple hierarchies.)

This approach to scene interpretation has been shown to be highly effective in practice. A version involving successive partitions of object/pose pairings, rank-based tests for the corresponding (classes of) hypotheses and breadth-first CTF search appears in (Geman, Manbeck & McClure 1995). The detection results shown in Figure 2 were obtained by an algorithm (Fleuret & Geman 2001) based on the strategy proposed here – traversing a multiresolution hierarchy of $\mathcal{X}$ binary hypothesis tests $\{X_A, A \in \mathcal{A}\}$, where each A represents a family of shapes with some common properties and X_A is an image functional designed to detect shapes in this family. In the face detection experiments, A is a subset of affine poses and X_A is based on checking for special local features (e.g., edges) which are likely to be present for faces with poses in A. In fact, X_A can be interpreted as a likelihood ratio test (Amit & Geman 2003). Recently, researchers in the computer vision community have started using similar methods for similar problems; see for example (Socolinsky, Neuheisel, Priebe, De Vinney & Marchette 2002) and (Viola & Jones 2001). Ideas related to CTF processing have also been proposed by (Frisch & Finke 1998) in a Bayesian classification framework where a hierarchy of estimators is built for the posterior of recursively clustered classes.

In Figure 2, the efficiency of sequential testing is illustrated for the group photo by counting, for each pixel, the amount of computation performed in its vicinity; clearly the spatial "density of work" is highly skewed. The corresponding density would be flat for nearly all other methods, e.g., those based on multilayer perceptrons or support vector machines.

Finally, a more or less exact implementation of the methodology here is provided in §10 in the context of a synthetic example introduced in (Jung 2001) for detecting rectangles against background clutter. A modification of that algorithm has been used to detect the roofs of buildings in aerial photographs in order to partially automate cartography (Jung 2002).

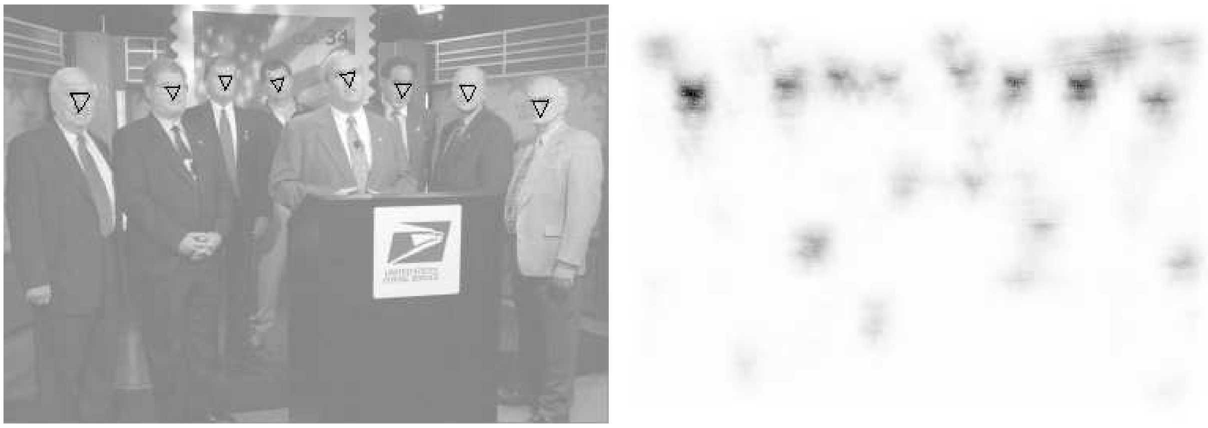


Figure 2: The detections (left) and "density of work" (right) for the group photo.

5 Problem Formulation

In this section we formulate efficient pattern detection as an appropriate optimization problem. In particular, we define the fundamental quantities which appear in this formulation, including attributes, tests and strategies, and how cost is measured both for individual tests and for testing designs. We also state our main assumptions about the test statistics and the relationships among cost, power and invariance which drive the optimization results in §§6-9.

5.1 Goals

The background probability space Ω represents the raw data – collections of numerical measurements – and $\mathcal{Y}$ denotes a set of *patterns* (or *classes* or *explanations*). We imagine the patterns $y \in \mathcal{Y}$ to be rather precise interpretations of the data and consequently $|\mathcal{Y}|$ to be very large. There is also a special explanation called *background*, denoted by 0, which represents "no pattern of interest" and is typically the most prevalent explanation by far.

We suppose there is a true state Y which takes values in $\{0\} \cup \mathcal{Y}$ and which, for simplicity, is determined by the raw data. In other words, we regard Y as a random variable on Ω. Most of what follows could be generalized to the case in which $Y \subset \mathcal{Y}$ and $Y = \emptyset$ represents background.

Example: In the context of machine perception, the raw data represent signals or images and the explanations represent the presentations of special entities, such as words in acoustical signals or physical objects in images (e.g., face instantiations or printed characters at a particular font and pose). The level of specificity of the explanations is problem-specific. However, we do assume that the data have in fact a unique interpretation at the level of precision of Y. Clearly this assumption eventually breaks down in the case of highly detailed semantic descriptions – at some point the subjectivity of the observer cannot be ignored.

The ultimate goal is *pattern identification*: Determine Y. However, for the reasons stated earlier, we shall focus instead on

Pattern Detection: *Reduce the set of possible explanations to a relatively small, data-driven subset* $\widehat{Y} \subset \mathcal{Y}$ *such that* $Y \in \widehat{Y} \cup \{0\}$ *with probability (almost) one.*

We shall also consider the special case of detecting one single, fixed pattern y^*. A related problem of interest, studied in (Fleuret 2000), (Fleuret & Geman 2001) and (Jung 2001), is *Background Filtering*: Determine whether or not $Y = 0$.

As discussed earlier, the rationale behind pattern detection is that requiring that $Y \in \widehat{Y} \cup \{0\}$ ensures, by definition, that no pattern is missed. Hence, the ensuing analysis, which is aimed at determining Y with high precision and is likely to be computationally intensive, can be limited to $\widehat{Y}$. Additional computation might involve a *contextual analysis*, such as constructing hypothesis tests on the fly for distinguishing between competing alternatives belonging to $\widehat{Y}$. This "postprocessing stage" will not be analyzed in this paper, *except* that we shall explicitly anticipate additional computation in the form of a penalty for unfinished business: We impose a "postprocessing cost" $C_{post}(\widehat{Y})$ proportional to the size of $\widehat{Y}$. The goal then is to find an optimal tradeoff between the costs of "testing" and "postprocessing."

5.2 Attributes and attribute tests

Any subset of patterns $A \subset \mathcal{Y}$ can be regarded as an "interpretation" of the data and we assume there are certain "natural groupings" of this nature (for instance, "writer" in a "Guess Who" version of twenty questions, "noun" in speech recognition and "character" in visual recognition). We call these distinguished subsets *attributes* and we denote the family of attributes by $\mathcal{A} \subset \mathcal{P}(\mathcal{Y})$ and suppose $|\mathcal{A}|$ is of order $\mathcal{O}(|\mathcal{Y}|)$. For every $y \in \mathcal{Y}$, we will assume that

$$\{y\} = \bigcap_{A \ni y} A. \tag{1}$$

One of our main assumptions is that $\mathcal{A}$ has a multiresolution, hierarchical structure with

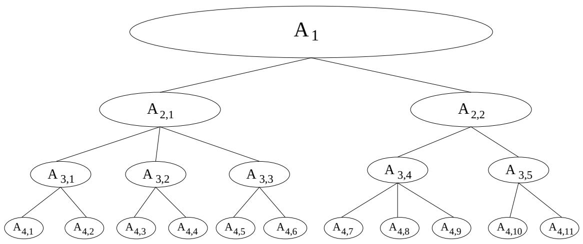


Figure 3: Example of a (non-regular) tree-structured hierarchy of attributes.

attributes at varying levels of precision. Formally, we assume that

$$\forall A, A' \in \mathcal{A},\ A \cap A' \neq \emptyset \Rightarrow (A' \subset A) \text{ or } (A \subset A').$$

Note that the set of attributes thus has a tree structure (see Figure 3 for an example). Furthermore, assumption (1) implies that the set of leaves of the corresponding tree is exactly the set of all singleton attributes.

For every attribute $A \in \mathcal{A}$ we can build one or more binary *tests* X - the result of testing the hypothesis $Y \in A$ against either[1] $Y \notin A$ or $Y = 0$; the value $X = 1$ corresponds to choosing $Y \in A$ and $X = 0$ to choosing the alternative. Due to the domination of the background class, at least at the beginning of the search, and due to the simplification afforded by measuring total computation cost under $P_0 = P(.|Y = 0)$, the alternative hypothesis will hereafter be $Y = 0$ and we define the power of the test accordingly:

$$\beta(X) = P(X = 0|Y = 0).$$

In order to make the notation more informative, we shall write either X_A to indicate the attribute being tested or $X_{A,\beta}$ to signal both the attribute and the power.

The first main assumption we will make about these tests is that *their false negative rate is negligible.* In other words, if a pattern (i.e., non-background explanation) is present then any attribute test which covers this pattern must respond positively:

$$P(X_A = 1|Y \in A) = 1,\ \ \forall A \in \mathcal{A}. \tag{2}$$

For this reason, and due to the origins of this work in visual object recognition, we sometimes refer to the size of A as the *level of invariance* of $X_{A,\beta}$, but usually just as the *scope* or *level*

[1]Which alternative is more appropriate is application-dependent. For example, in inductive learning, the two cases correspond to the nature of the "negative" examples in the training set - whether they represent a random sample under $Y \in A^c$ or under "background". In the applications cited earlier, the tests are constructed based *entirely* on the statistical properties of the patterns in A; neither alternative is explicitly represented.

of resolution. In general, however,

$$P(X_A = 1|Y = y) > 0, \;\; \forall y \in \{0\} \cup \mathcal{Y} \setminus A.$$

In other words, the tests are usually not *perfect* or *two-sided invariants*.

Formally, assumption (2) is not necessary for the mathematical results in the coming sections to hold, because we will only make computations under the "background probability" (when $Y = 0$); see §5.4. However, this assumption is necessary for our formulation of pattern identification to make sense; indeed, it implies that if one has performed tests $X_{A_1}, \ldots, X_{A_k}$, then necessarily

$$Y \in \mathcal{Y} \setminus \bigcup_{k:X_{A_k}=0} A_k.$$

We will refer to the patterns above as "compatible" with tests $X_{A_1}, \ldots, X_{A_k}$ and focus on sequential testing designs for which the chosen $\widehat{Y}$ is the set of patterns compatible with all the tests actually performed. This choice coheres with our requirement that $Y \in \widehat{Y} \cup \{0\}$ with probability one, while at the same time ensuring that $\widehat{Y}$ is of minimum size given the available information. $\widehat{Y}$ – the set of remaining compatible patterns at the end of the testing stage – is called the set of *detected patterns* (relative to the testing strategy used).

Finally, each test $X_{A,\beta}$ has a *cost* or *complexity* $c(X_{A,\beta})$ which represents the amount of online computation (or time) necessary to evaluate $X_{A,\beta}$. In §5.6 we shall consider different cost models, one in which costs are predetermined quantities related to power and scope, and the other in which cost is "usage-based".

5.3 Test hierarchies

We consider two types of families of tests, one with exactly one test (at some fixed power) per attribute and referred to as a *fixed test hierarchy*, and one with a one-parameter family of tests $\{X_{A,\beta}, 0 \leq \beta \leq 1\}$ for each $A \in \mathcal{A}$ indexed by power and referred to as an *extended test hierarchy*.

5.3.1 Fixed hierarchy

We will denote such a hierarchy by $\mathcal{X} = \{X_A, A \in \mathcal{A}\}$ and write $\beta(A)$ for the power of X_A and $c(A)$ for its cost. Optimal testing strategies for fixed hierarchies is the subject of §6,§8 and §9 for two different cost models. In the analysis in those sections a central role is played by the (random) set $\widehat{Y}(\mathcal{X})$ of patterns which are compatible with *all* the tests in $\mathcal{X}$, i.e., those patterns which are verified at all levels of resolution. More precisely:

$$\widehat{Y}(\mathcal{X}) = \mathcal{Y} \setminus \bigcup \{A \in \mathcal{A} | X_A = 0\}.$$

Recall that under our constraint on the false negative error, we necessarily have $P[Y \in \{0\} \cup \widehat{Y}(\mathcal{X})] = 1$. Clearly, $\widehat{Y}(\mathcal{X})$ leads to a smaller postprocessing cost than any $\widehat{Y}$ based on only *some* of the tests in the hierarchy, but, of course, requires more computation to evaluate in general.

5.3.2 Extended hierarchy

The extended test hierarchy is

$$\widetilde{\mathcal{X}} = \{X_{A,\beta} | A \in \mathcal{A}, \beta \in [0,1]\}.$$

In §7 we will consider testing strategies in which, at each step in a sequential procedure, *both* an attribute *and* a power may be selected. This clearly leads to a more complex optimization problem and our results in this direction are correspondingly far less complete than those in the case of a fixed hierarchy. From another point of view, extracting a subset of tests from an extended hierarchy (e.g., specifiying a testing strategy) is a type of *model selection* problem.

5.4 The probabilistic model

In order for the upcoming optimization problems to be well-defined, we need to specify the joint distribution of the random variables in $\widetilde{\mathcal{X}}$.

The first hypothesis we make is that we are going to measure mean computation relative to $P_0(.) = P(.|Y=0)$ – the "background distribution." This is justified by the assumption that, *a priori*, the probability of the explanation $Y = 0$ is far greater than the compound alternative $Y \neq 0$ let alone any single, nonbackground explanation. For instance, in visual processing, a randomly selected subimage is very unlikely to support a precise explanation in terms of visible patterns; in other words, most of the time all we observe is clutter.

The second hypothesis we make is that, under P_0, any family of tests $X_{A_1,\beta_1}, \ldots, X_{A_k,\beta_k}$ for *distinct* attributes $A_1, \ldots, A_k$ are independent. This is probably the strongest assumption in this paper but is not altogether unreasonable under P_0 in view of the structure of $\mathcal{A}$ since two distinct tests are either testing for disjoint attributes (if they are at the same level of resolution) or testing for attributes at different levels of resolution. In §6 we shall briefly consider simulations for a non-trivial dependency structure – a Markov hierarchy.

No assumptions are made about the dependency structure among tests for the same attribute but at different powers. Instead, the assumption to be made in the following section that no attribute can be tested twice in the same procedure allows us to compare the cost of testing strategies regardless of this dependency structure.

5.5 Testing strategies and their cost

We consider sequential testing processes, where tests are performed one after another and the choice of the next test to be performed (or the decision to stop the testing process) can depend on the outcomes of the previously performed tests. We will make the important assumption that in any sequence of tests, a given attribute can only be tested once.

Definition 1 (Testing Strategy). *A strategy is a finite labeled binary tree T where each internal node $t \in T^\circ$ is labeled by a test $X(t) = X_{A(t),\beta(t)}$ and where $A(t) \neq A(s)$ for any two nodes t, s along the same branch. At each internal node t the right branch corresponds to $X(t) = 1$ and the left branch to $X(t) = 0$.*

The restriction to at most one test per attribute A along any given branch, whereas of course automatically satisfied in the case of a fixed hierarchy (§§6,8,9), does limit the set of possible strategies for an extended hierarchy since tests $X_{A,\beta}$ of varying power are available for each attribute A. In that case the purpose of this assumption is essentially to simplify the analysis by guaranteeing that all the tests actually performed are independent.

The leaves (terminal nodes) of T will be labeled in accordance with the answers to the tests: Every leaf of T is labeled by the subset $\widehat{Y} \subset \mathcal{Y}$ of compatible patterns that have not been ruled out by the tests performed by the strategy (along the branch leading to this leaf). In other words, for any strategy T and leaf s of T, if $\mathcal{X}(s)$ denotes the set of tests along the branch leading to s, we put

$$\widehat{Y}(s) = \mathcal{Y} \setminus \bigcup \{X \in \mathcal{X}(s) | X = 0\}.$$

The random set $\widehat{Y}(T)$ is then defined by interpreting T as a function of the tests which takes values among its leaves. However, how the leaves are labeled is irrelevant for the purposes of defining the testing cost, $C_{test}(T)$, of a strategy; it will only influence the postprocessing cost $C_{post}(\widehat{Y})$.

5.5.1 Cost of testing

There are several equivalent definitions of the testing cost of T, another random variable. One is

$$C_{test}(T) = \sum_{t \in T^\circ} c(X(t)) \mathbf{1}_{H_t}$$

where H_t is the history of node t - the event that t is reached. This is clearly the same as aggregating the costs over the branch traversed or adding the costs of all tests performed. Given a probability distribution P on Ω, and in particular $P = P_0$, two equivalent expressions for the *mean cost* are then:

$$
\begin{aligned}
E_0 C_{test}(T) &= \sum_{t \in T^\circ} c(X(t)) P_0(H_t) \\
&= \sum_X c(X) q_X(T) \qquad (3)
\end{aligned}
$$

where

$$
\begin{aligned}
q_X(T) &= P_0(X \ performed \ in \ T) \\
&= \sum_{t \in T^\circ} \mathbf{1}_{\{X(t)=X\}} P_0(H_t).
\end{aligned}
$$

Expression (3) is particularly useful in proving some of our results; in §6 we will transform it into yet another expression that will anchor the analysis there.

5.5.2 Cost of postprocessing

It is natural to define the postprocessing cost in the following, goal-dependent manner:

- *Detecting a Special Pattern*: $C_{post}(\widehat{Y}(T)) = c^* \mathbf{1}_{\{y^* \in \widehat{Y}(T)\}}$ where y^* is the target pattern;
- *Detecting Any Pattern*: $C_{post}(\widehat{Y}(T)) = c^* |\widehat{Y}(T)|$.

Here, c^* is some constant called the *unit postprocessing cost*.

In the case of a single target pattern, note that this choice of postprocessing cost naturally leads us to disregard any attribute not containing the target y^* as those tests are irrelevant to the goal at hand and can only augment the total cost. Consequently, the set of relevant attributes reduces to a "vine" $A_1 \supset A_2 \supset \ldots \supset A_L$. In this case, choosing a testing strategy boils down to choosing a subset of these relevant attributes and an order in which to test for them. If a test returns a null answer the search terminates with the outcome $y^* \notin \widehat{Y}$ and there is no postprocessing charge; on the other hand, if all the selected tests respond positively, then $y^* \in \widehat{Y}$ is declared (which still may not be true) and the charge is c^*. In particular, the testing strategy T itself has in this case the structure of a vine (see Figure 4). In contrast, in the case of general pattern detection the testing strategies are of course tree-structured.

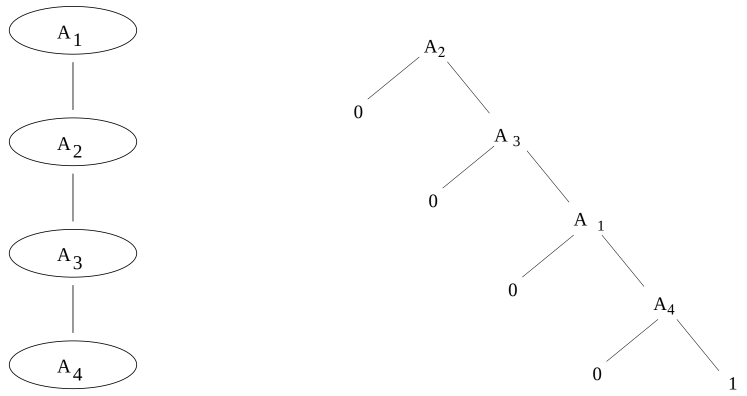


Figure 4: Left: A vine-structured hierarchy of attributes for detecting one pattern. Right: An example of a vine-structured testing strategy for this hierarchy.

5.5.3 Optimization problem

The *total computational cost* for the task at hand is $C_{test}(T) + C_{post}(\widehat{Y}(T))$. The corresponding optimization problem, our central focus, is then to find a strategy attaining

$$\min_{T \in \mathcal{T}} \Big(E_0 C_{test}(T) + E_0 C_{post}(\widehat{Y}(T)) \Big) \tag{4}$$

where $\mathcal{T}$ is the family of all strategies. We emphasize that in the case of extended hierarchies we are therefore optimizing over both power and resolution.

5.5.4 Equivalent model with perfect tests

There is an equivalent way to interpret the postprocessing cost which is technically more convenient. We can think of c^* as the cost of performing a *perfect* test (i.e. without errors under P_0) for any individual pattern. Therefore, the postprocessing cost model is *formally equivalent* to supposing there is no postprocessing stage, but that *no errors* (under P_0) are allowed at the end of the procedure, enforced by performing, as needed, some additional, perfect tests at the end of the search. Since we have assumed that no attribute, and in particular no singleton $\{y\}$, cannot be tested at two different powers along the same branch, we can incorporate perfect testing into the previous framework simply by adding a final layer to the original hierarchy $\mathcal{A}$ which copies the original leaves, thereby accommodating a battery of perfect singleton tests having cost c^*. (Conditional independence is actually maintained since the new tests are deterministic under P_0.) We denote by $\overline{\mathcal{A}}$ the resulting augmented hierarchy. This formal[2] construction allows us include the postprocessing cost in the testing framework. Furthermore, in the augmented model it is not difficult to show that for any strategy T there exists a strategy T' performing exactly the same tests, but with the perfect tests performed at the end only, so that the optimization problem is in fact unchanged by allowing the perfect tests to be performed at any time. *In summary, the equivalent optimization problem is to minimize the amount of computation necessary to achieve no error under P_0 based on the augmented hierarchy.*

5.6 Cost of a test

Two models are considered, one in which the costs of the tests are fixed *a priori* and, in particular, do not depend on the testing strategy, and one in which the costs can adapt to the frequency of utilization. In the former case, there are certain natural tradeoffs among cost, power and invariance:

- At a given cost, power should be a *decreasing* function of invariance;
- At a given power, cost should be an *increasing* function of invariance;
- At a given invariance, cost should be an *increasing* function of power.

5.6.1 Power-based cost

In §6, we will first deal with a generic setting where the test associated to a given attribute A has power $\beta(A)$ and cost $c(A)$. In §7 we will use a more specific model reflecting the tradeoffs among cost, power and invariance mentioned above:

$$c(X_{A,\beta}) = \Gamma(|A|) \times \Psi(\beta) \tag{5}$$

where the *complexity function* Γ is sub-additive and the *power function* Ψ is convex. Consequently, we evaluate the cost of a test much like the merit of a dive in the Olympics: at any given level of difficulty (Γ), a score (Ψ) is assigned based on performance alone. For normalization, we can assume that $\Gamma(1) = 1$. Then, with the equivalent model where the postprocessing cost is replaced by "perfect" tests in mind, it is consistent to assume $c^* = \Psi(1)$. This multiplicative model is supported (at least roughly) by what is observed in actual experiments (see section §10).

[2] Due to this construction there is a slight abuse of notation when identifying an attribute with a subset of $\mathcal{Y}$, since in the augmented hierarchy we would like (in order to be entirely consistent) to consider some attributes as distinct although they correspond to the same set $\{y\}$. However we will stick to the notation introduced before to avoid cumbersome changes.

One special case, treated in §7, is $\Gamma(n) = n$, i.e., the complexity is simply the level of invariance. This case is the *least* favorable to CTF strategies since, in effect, no "credit" is given for shared properties among two disjoint attributes $A, B \in \mathcal{A}$. If, for instance, $|A| = |B|$ with A, B disjoint, a test for $A \cup B$ at a given power β has the same cost as testing separately for both A or B at power β.

A particular case, treated in (Fleuret 2000) and (Jung 2001), in the setting of a fixed hierarchy, is to assume $c(X_{A,\beta_A}) = \Psi(\beta_A)$ for some function Ψ. The model considered here is more general.

5.6.2 Usage-based cost

In §8-9 we consider a different cost model in which cost of a test can be chosen arbitrarily depending on the testing strategy and subject to a global resource constraint. The basic idea is that in some circumstances it might not be efficient to fix the costs of the tests in advance, regardless of their inherent complexity. It may be more efficient to allow the utilization of computing resources to be partitioned in accordance with the frequency with which certain routines are performed, and that frequency will depend on the *order* in which the tests are evaluated.

Suppose we have a fixed amount of resources $R \leq 1$ to be distributed among the tests in accordance with a testing strategy T. Let $r(X)$ denote the allocation to test X. We suppose the cost $c(X)$ of test X is given by $c(X) = -\log(r(X))$. From (3), The cost of testing is then

$$E_0 C_{test}(T) = -\sum_X \log(r(X)) q_X(T)$$

subject to the constraint:

$$\sum_X r(X) \leq R.$$

From standard arguments it is clear that for a fixed strategy T, the optimal choice for the allocation of resources to the tests (i.e., the choice leading to the minimal average testing cost) is given by choosing $r(X)$ to satisfy

$$\frac{r(X)}{R} = \frac{q_X(T)}{Q(T)} \qquad (6)$$

where

$$Q(T) = \sum_X q_X(T).$$

Notice that $Q(T) \geq 1$ (if T is not empty) and $Q(T)$ represents the average number of tests performed.

Substituting $c(X) = -\log(r(X)) = -\log((q_X(T)R/Q(T)))$ into the cost of testing, we obtain

$$E_0 C_{test}(T) = -\sum_X q_X(T)\log(q_X(T)) + Q(T)\log(Q(T)/R). \tag{7}$$

In §8-9, to make the analysis easier we will restrain the optimization of the cost to those strategies T satisfying the constraint that $\widehat{Y}(T) = \widehat{Y}(\mathcal{X})$ (these will be called "complete strategies" below), thereby allowing one to disregard the postprocessing cost. In that case the functional to be minimized over the considered strategies is just (7) above.

5.7 Special strategies

In the following sections our main goal will be to determine under what additional hypotheses the optimal strategies are "coarse-to-fine" (CTF).

Definition 2 (Coarse-to-Fine). *A strategy $T \in \mathcal{T}$ is* **CTF in resolution** *or just* **CTF***, if an attribute is tested if and only if each of its ancestors has already been tested and returned a positive answer. A strategy $T \in \mathcal{T}$ is* **CTF in power** *if, for any two nodes s, t along the same branch, $\beta(s) \geq \beta(t)$ whenever $A(s) \subset A(t)$.*

In the case of detecting a single pattern, this simply means that a CTF strategy performs all the relevant tests in the order of increasing resolution, i.e. $X_{A_1}, \ldots, X_{A_L}$. For pattern detection, several different strategies have the CTF property, for instance "breadth-first" and "depth-first" search. In Figure 5 these two CTF strategies are illustrated in the case of a hierarchy of depth $L = 5$ and test outcomes such that $\widehat{Y}(T_{ctf}) = \emptyset$, i.e., no patterns are verified at all resolutions due to the "null covering" $\{X_{3,1} = 0, X_{4,3} = 0, X_{4,4} = 0, X_{4,5} = 0, X_{4,6} = 0, X_{3,4} = 0\}$ (writing $X_{l,k}$ for the k-th test at depth l). Notice that the *breadth-first* CTF strategy has the nice feature that the tests are always performed in the order of nondecreasing depth.

For a fixed hierarchy, all CTF strategies for pattern detection perform exactly the same tests (although perhaps not in the same order). Whatever the order chosen, in the end, along any branch of the attribute hierarchy, every test has been performed starting from the root until the first null answer encountered on this branch. It is therefore possible to speak of "the" CTF strategy, it being understood that the precise order in which the tests are performed does not affect the mean cost.

Note: Whereas we do not consider the problem of separating patterns from background in and of itself (as in (Jung 2001)), it is interesting to observe that the situation is more

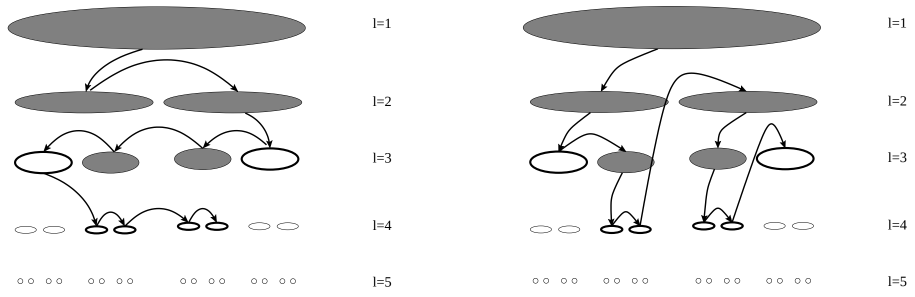


Figure 5: Example of typical CTF strategies. Left: breadth-first; Right: depth-first.

complex in that case since all CTF strategies are not equivalent. Indeed, in any optimal strategy, testing stops as soon as any complete "1-chain" is found and, consequently, depth-first CTF strategies are generally optimal, as shown in (Jung 2001).

The probability of performing a test X_A in a CTF strategy has a simple expression:

$$\begin{aligned} q_A(T) &= P_0(X_A \; performed \; in \; T) \\ &= P_0(X_B = 1, B \supset A, B \neq A) \\ &= \prod_{B \supset A} (1 - \beta_B) \end{aligned}$$

Moreover, under the CTF strategy, $\widehat{Y}$ minimizes $C_{post}(\widehat{Y}(T))$, and, in fact,

$$\widehat{Y}(T_{ctf}) = \widehat{Y}(\mathcal{X}) \; a.s. \tag{8}$$

the set of all "1-chains" in the hierarchy. It follows that the total mean cost of the CTF strategy is then given by

$$E_0 C(T_{ctf}) = \sum_{A \in \mathcal{A}} c_A \prod_{B \supset A} (1 - \beta_B) + E_0 |\widehat{Y}(\mathcal{X})|.$$

Still in the case of a fixed hierarchy, it will be useful to delineate all strategies with property (8).

Definition 3 (Complete Strategies). *A strategy* $T \in \mathcal{T}$ *is complete if* $\widehat{Y}(T) = \widehat{Y}(\mathcal{X})$. *The family of complete strategies is denoted by* $\overline{\mathcal{T}}$.

When dealing with the usage-based cost model in §§8,9, in order to simplify the analysis, we will restrict the set of considered strategies to complete ones; in that case the postprocessing cost does not have an influence over the optimization problem, and will be disregarded.

Remark: Under the hypotheses we have made, *for a complete strategy* it is possible to compute explicitly the probability of error under the null hypothesis before the postprocessing step, i.e., to calculate $P_0(\widehat{Y} \neq \emptyset)$. (This is the probability that at least one non-null pattern is detected when only background clutter is actually observed.) For single-pattern detection, it is just the probability under P_0 that all the tests along the vine respond positively: $P_0(\widehat{Y} \neq \emptyset) = \prod_{k=1}^{L}(1-\beta_k)$ (where $\beta_k = P(X_{A_k} = 0)$); for detection of all possible patterns, it is exactly the probability that there exists a "path of ones" leading from the root of the attribute tree to one of its leaves; given the independence assumption on the tests under P_0, this in turn is *exactly the probability of non-extinction of an inhomogeneous branching process at generation L, which can be computed explicitly once the branching probabilities (i.e.* $\beta(A), A \in \mathcal{A}$*) are known.*

Finally, for an extended hierarchy $\widetilde{\mathcal{X}}$, there are many different, non-cost-equivalent, CTF strategies depending on the powers chosen for the tests along each branch. Nonetheless, surprisingly, the optimal CTF strategy can sometimes be precisely characterized, being CTF in power with, in fact, a unique power assigned to each attribute (see §7).

INDEX OF NOTATION

Objects:

$\mathcal{Y}$: set of all possible objects or explanations

$Y \subset \mathcal{Y}$: true (data-dependent) set of objects

$Y = \emptyset$: background explanation

$P_0(.) = P(.|Y = \emptyset)$: the background distribution

Attributes:

A: a grouping of objects (a.k.a. attribute)

$\mathcal{A}$: hierarchy of attributes

$\overline{\mathcal{A}}$: "augmented" hierachy of attributes (see §5.5.4)

$\mathcal{Z}(\mathcal{A})$: coverings of $\mathcal{A}$: $\cup_{A \in Z} = \mathcal{Y}$ for all $Z \in \mathcal{Z}(\mathcal{A})$

$\mathcal{A}_l, l = 1, ..., L$: attributes at level l in tree-structured case

A_1: coarsest attribute(s); the root in the tree-structured case: $\{A_1\} = \mathcal{A}_1$

Tests:

X: binary random variable

- $\sigma(X) \in \mathcal{A}$: scope of X
- $\beta(X) \in [0, 1]$: power of X
- $c(X) \in [0, \infty)$: cost of X

$c(X) = \Gamma(\sigma(X)) \times \Psi(\beta(X))$: power-based cost

- Γ: increasing, subadditive complexity function
- Ψ: increasing, convex power function: $\Psi(0) = 0, \Psi(1) = 1$

$X_{A,\beta}$: test with scope A and power β

$\mathcal{X}(\mathcal{A})$: family of tests indexed by $\mathcal{A}$; "fixed hierarchy"

- $\beta(A)$: power of X_A
- $c(A)$: cost of X_A

Strategies:

T: labeled binary tree

- $A(s)$: scope of the test at interior node s of T
- $\beta(s)$: power of the test at s
- $\widehat{Y}(t)$: surviving explanations at exterior node t of T

$\widehat{Y}(T)$: detected set of objects (surviving explanations) after testing

$q_X(T)$: probability of performing X in T under P_0

$C(T) = C_{test}(T) + C_{post}(T)$: total cost:

- $C_{test}(T)$: sum of the costs of the tests perfomred in T
- $C_{post}(T)$: postprocessing cost $c^*|\widehat{Y}(T)|$

6 Optimal Strategies for Fixed Costs and Powers

Throughout this section we assume a fixed test hierarchy $\mathcal{X} = \{X_A, A \in \mathcal{A}\}$ and we write $c(A), \beta(A)$ for the cost and power, respectively, of X_A. We will then refer to "testing an attribute A" or "attaching an attribute" to a node of T without ambiguity. Our goal is to identify conditions (tradeoffs) involving $\{c(A), \beta(A), A \in \mathcal{A}\}$ under which optimal strategies may be characterized.

For parts of this section it will be easier to actually consider the equivalent model with perfect tests *en lieu* of the postprocessing cost, as described in §5.5.4. From here on, $\overline{\mathcal{A}}$ will denote the augmented hierarchy, and the considered strategies T for $\overline{\mathcal{A}}$ will satisfy the no-error constraint. In other words, in the augmented model, when the strategy ends, all patterns $y \in \mathcal{Y}$ must have been covered by at least one test which has been performed and returned 0 (again, it may be one of the perfect, artificial tests representing postprocessing). We start this section with a fundamental formula for the average cost $E_0 C(T)$ that will useful for all of the results to follow.

6.1 Reformulation of the cost

As just pointed out, in the augmented hierarchy model, strategies must find a way to "cover" all patterns with attributes whose associated test is negative. Therefore the notion of *covering* will play a central role in the analysis to come, motivating the following definitions:

Definition 4 (Covering). *A set of attributes $\mathcal{Z} \subset \overline{\mathcal{A}}$ is a* covering *if*

$$\bigcup \{A, A \in \mathcal{Z}\} = \mathcal{Y}.$$

The set of coverings for the augmented hierarchy $\overline{\mathcal{A}}$ is denoted $\mathcal{Z}(\overline{\mathcal{A}})$.

Definition 5 (Tested Attributes). *For a given strategy T, denote by $\mathcal{X}(T)$ the (random) set of attributes tested by T, and by $\mathcal{X}_0(T)$ the set of attributes in $\mathcal{X}(T)$ for which the corresponding test returned the answer 0, called the* zero set *of T.*

Of course, the no-error constraint for a strategy T now reads simply: $\mathcal{X}_0(T)$ is (a.s.) a covering. We now turn to an important formula:

Lemma 1 (Cost Reformulation). *For any (no-error) strategy T for the augmented hierarchy $\overline{\mathcal{A}}$:*

$$E_0 C(T) = \sum_{Z \in \mathcal{Z}(\overline{\mathcal{A}})} \left(P_0(\mathcal{X}_0(T) = Z) \sum_{A \in Z} \frac{c(A)}{\beta(A)} \right). \tag{9}$$

Proof. For any attribute $A \in \overline{\mathcal{A}}$, let $\lambda_A(T) = P_0(A \in \mathcal{X}_0(T))$ and let $q_A(T) = P_0(X_A \text{ performed by } T)$. Note that we have two useful expressions for $\lambda_A(T)$:

$$\lambda_A(T) = \sum_{\substack{Z \in \mathcal{Z}(\overline{\mathcal{A}}),\\ Z \ni A}} P_0(\mathcal{X}_0(T) = Z); \tag{10}$$

and

$$\lambda_A(T) = P_0(A \in \mathcal{X}(T), X_A = 0) = P_0(A \in \mathcal{X}(T))P_0(X_A = 0) = q_A(T)\beta(A), \tag{11}$$

where the second equality comes from the fact that the event that A is performed by T only depends on the values of tests for other attributes, and is thus independent of X_A by the independence assumption.

Now recalling expression (3) we have

$$\begin{aligned} E_0C(T) &= \sum_{A \in \overline{\mathcal{A}}} c(A) q_A(T) \\ &= \sum_{A \in \overline{\mathcal{A}}} \frac{c(A)}{\beta(A)} q_A(T)\beta(A) \\ &= \sum_{A \in \overline{\mathcal{A}}} \frac{c(A)}{\beta(A)} \lambda_A(T) \\ &= \sum_{A \in \overline{\mathcal{A}}} \frac{c(A)}{\beta(A)} \sum_{\substack{Z \in \mathcal{Z}(\overline{\mathcal{A}}),\\ Z \ni A}} P_0(\mathcal{X}_0(T) = Z) \\ &= \sum_{Z \in \mathcal{Z}(\overline{\mathcal{A}})} \left(P_0(\mathcal{X}_0(T) = Z) \sum_{A \in Z} \frac{c(A)}{\beta(A)} \right). \end{aligned}$$

□

This lemma combines two straightforward observations. First, the cost "generated" by a specific attribute A using strategy T can be written as

$$c(A)P_0(A \in \mathcal{X}(T)) = \frac{c(A)}{\beta(A)} P_0(A \in \mathcal{X}(T))P_0(X_A = 0) = \frac{c(A)}{\beta(A)} P_0(A \in \mathcal{X}_0(T)). \tag{12}$$

Second, the sum over attributes of the last expression can be reformulated as a sum over coverings (using the no-error property). Note in particular that (12) above has the following interpretation: As far as average cost is concerned, it is equivalent to i) pay the cost $c(A)$ every time test X_A is performed or ii) pay the cost $c(A)/\beta(A)$ when X_A is performed and *and returns the answer 0* but pay nothing otherwise.

Note also that the lemma implies that the average cost $E_0C(T)$ is therefore a *convex combination* of the quantities $\sum_{A \in Z} \frac{c(A)}{\beta(A)}$ for $Z \in \mathcal{Z}(\overline{\mathcal{A}})$.

6.2 Detecting one special pattern

Recall this corresponds to the case where the set of attributes has the structure of a vine (see Figure 4). We can imagine two broad scenarios: In one case, there is really only one pattern of interest, and hence no issue of invariance other than guaranteeing that every test is positive whenever $Y = y^*$. Imagine, for example, constructing a sequence of increasingly precise "templates" for a given shape, in which case *both* power *and* cost would typically increase with precision. In another scenario, one could imagine utilizing a hierarchy of tests originally constructed for multiple patterns in order to check for the presence of a single pattern y^*. Clearly, only one particular branch of the hierarchy is then relevant, namely the branch along which all the attributes contain y^*. Obviously, such tests would typically be less dedicated to y^* than in the first scenario, except at the final level. In either case the natural framework is a *sequence of tests*, say X_ℓ for attributes A_ℓ, with costs c_ℓ and powers β_ℓ for $\ell = 1, ..., L$, and the natural background measure is conditional on $Y \neq y^*$. Also, it is simpler here to consider the augmented hierarchy setting, so that we assume that there is a test at level $L+1$ with $\beta_{L+1} = 1, c_{L+1} = c^*$.

The important quantity is the cost normalized by the power: $\{\frac{c_\ell}{\beta_\ell}\}$. Let $n(\ell), \ell = 1, ..., L+1$, denote the ordering of these ratios:

$$\frac{c_{n(1)}}{\beta_{n(1)}} \leq \frac{c_{n(2)}}{\beta_{n(2)}} \leq \cdots \leq \frac{c_{n(L+1)}}{\beta_{n(L+1)}}. \tag{13}$$

Since we are in the setting of the augmented hierarchy, there exists a distinguished index ℓ^* corresponding to the perfect test, for which $c_{n(\ell^*)} = c^*, \beta_{n(\ell^*)} = 1$.

Theorem 1. *The optimal strategy for detecting a single target pattern is to order the tests in accordance with* $(n(1), n(2), ..., n(\ell^*))$*, i.e., perform* $X_{n(1)}$ *first, then* $X_{n(2)}$ *whenever* $X_{n(1)} = 1$*, etc., and stop with* $X_{n(\ell^*)}$*. The tests* $X_{n(k)}$ *for* $k > \ell^*$ *are never performed.*

Note that the last test, $X_{n(\ell^*)}$, is the perfect one, and always returns the answer 0 under P_0. Reinterpreted in the original model, this would mean that if $X_{n(\ell^*-1)}$ is reached in the strategy and returns answer 1, then the testing procedure ends and the postprocessing stage is performed.

This theorem is a consequence of a straightforward recursion (proof omitted) applied to the following lemma:

Lemma 2. *There exists an optimal strategy for which the first test performed is* $X_{n(1)}$*.*

Proof. Let T be some strategy performing the tests in the order $n'(1), n'(2), \ldots, n'(k^*)$ (for some $k^* \leq L+1$, with $n'(k^*) = n(\ell^*) = L+1$). Assume $n'(1) \neq n(1)$ and consider strategy

T_0 obtained by "switching" $X_{n(1)}$ to the first position, i.e. performing $X_{n(1)}$ first, and then whenever $X_{n(1)} = 1$ continuing through strategy T normally except if an index i is encountered for which $n'(i) = n(1)$, in which case $X_{n(1)}$ is not performed again, but just skipped.

Compare the costs of T and T_0 using equation (12): clearly the mean cost of these strategies is a convex combination of the $(c_\ell/\beta_\ell), \ell = 1, \ldots, L+1$, since $\sum_{\ell=1}^{L+1} P(A_\ell \in \mathcal{X}_0(T)) = 1$ in the single pattern case. More explicitly,

$$P(A_k \in \mathcal{X}_0(T)) = \beta_k \prod_{\ell : n'(\ell) < n'(k)} (1 - \beta_\ell)$$

with the corresponding formula for T_0. From this formula it is clear that the weight for the ratio $c_{n(1)}/\beta_{n(1)}$ is higher in T_0 than in T, while all the other weights either are smaller or stay the same (depending whether the corresponding tests were placed before or after $X_{n(1)}$ in T). Since $c_{n(1)}/\beta_{n(1)}$ is the smallest of the ratios, the average cost of T_0 is lower than the cost of T. □

6.3 Detecting any pattern

Our goal is to determine conditions under which (4) is minimized by the CTF strategy. First, we consider a simple sufficient condition which guarantees that the optimal strategy is complete, meaning $T \in \overline{\mathcal{T}}$. (Recall that $T \in \overline{\mathcal{T}}$ if $\widehat{Y}(T) = \widehat{Y}(\mathcal{X})$; in other words, testing is halted if and only if all "1-chains" in $\mathcal{X}$ are determined.) This condition is by no means necessary since we will prove the optimality of the CTF strategy (which belongs to $\overline{\mathcal{T}}$) under a much weaker condition, but is however informative.

Proposition 1. *If for any attribute $A \in \mathcal{A}$, $\dfrac{c_A}{\beta_A} \leq c^*$, then the optimal strategy must belong to $\overline{\mathcal{T}}$.*

Proof. Let T be an optimal strategy and let s denote a leaf of T. Recall that $\mathcal{X}(s)$ is the set of tests along the branch terminating in s and $\widehat{Y}(s) = \mathcal{Y} \setminus \bigcup \{X_A \in \mathcal{X}(s) | X_A = 0\}$. The expected cost of T is then of the form

$$E_0 C(T) = C + p_s c^* |\widehat{Y}(s)|, \tag{14}$$

where p_s is the probability of reaching s and the second term is the contribution to the mean postprocessing cost at leaf s. In general $\widehat{Y}(\mathcal{X}) \subset \widehat{Y}(s)$, and if these sets do not coincide (possibly empty in the case of a null covering), then, by definition, there must be a test

$X_A \notin \mathcal{X}(s)$ for which $A \cap \widehat{Y}(s) \neq \emptyset$. Consider the strategy T' obtained by adding this test to T at node s. Then

$$E_0C(T') = C + p_s \left[c_A + \beta_A c^* |\widehat{Y}(s) \setminus A| + (1 - \beta_A) c^* |\widehat{Y}(s)| \right] \tag{15}$$

Since $|\widehat{Y}(s)| - |\widehat{Y}(s) \setminus A| \geq 1$ it follows easily from the hypothesis, (14) and (15) that $E_0C(T) - E_0C(T') > 0$ which contradicts the optimality of T. □

We now turn to the problem of optimality of CTF strategies. The method of proof used in §6.1, although very simple in that case, will still serve as a template for most of the results to come. More precisely, under different assumptions about the models, we will always try to first establish the following property denoted **(CF)** for "coarsest first":

Definition 6 ((CF) Property). *There exists an optimal strategy for which the first test performed is the coarsest one.*

In most cases, we will establish the optimality of T_{ctf} as a consequence of **(CF)** for the various models considered. The current model – fixed, power-based cost – is the simplest and allows us to present the main ideas behind the arguments based on the **(CF)** property – a recursion based on "subhierarchies" and the concept of a "conditional strategy". As always, $\mathcal{A}$ is a nested hierarchy of attributes.

Definition 7 (Subhierarchy). *We call $\mathcal{B} \subset \mathcal{A}$ a subhierarchy of $\mathcal{A}$ if there exists an attribute $B_0 \in \mathcal{A}$ such that*

$$\mathcal{B} = \{A \in \mathcal{A} | A \subseteq B_0\}.$$

More specifically, we call $\mathcal{B}$ the subhierarchy rooted in B_0 *and we refer to B_0 as the set of patterns spanned by $\mathcal{B}$, also denoted $\mathcal{Y}_{\mathcal{B}}$.*

Definition 8 (Conditional strategy). *Let A_1 be the root of $\mathcal{A}$ and $\mathcal{B}$ be the subhierarchy of $\mathcal{A}$ rooted in one of the children of A_1. Then $\mathcal{A}$ can be written as a disjoint union $\mathcal{A} = \{A_1\} \dot\cup \mathcal{B} \dot\cup \overline{\mathcal{B}}$. Let $x_{\overline{\mathcal{B}}}$ be a set of numbers in $\{0, 1\}$ indexed by $\overline{\mathcal{B}}$. Consider a testing strategy T for $\mathcal{A}$. The conditional strategy $T_{\mathcal{B}}(x_{\overline{\mathcal{B}}})$ on subhierarchy $\mathcal{B}$ is defined as follows: For every internal node t of T,*

- *If $X(t)$ is a test for an attribute $B \in \mathcal{B}$, leave it unchanged;*
- *If $X(t) = X_{A_1}$, cut the strategy subtree rooted at t and replace it by the right subtree of t;*

- *If $X(t)$ is a test for an attribute $A \in \overline{\mathcal{B}}$, cut the strategy subtree rooted at t and replace it by the right subtree of t if $x_A = 1$, and by the left subtree of t if $x_A = 0$.*

Finally, relabel every remaining leaf s by $\widehat{Y}(s) \cap \mathcal{Y}_{\mathcal{B}}$.

This rather involved definition simply says that $T_{\mathcal{B}}(x_{\overline{\mathcal{B}}})$ is the testing strategy on subhierarchy $\mathcal{B}$ obtained from T when $X_{A_1} = 1$ and the answers to $X_{\overline{\mathcal{B}}} = \{X_B, B \in \overline{\mathcal{B}}\}$ are fixed to be $x_{\overline{\mathcal{B}}}$, and T is pruned accordingly. An obvious but nevertheless crucial observation is that $T_{\mathcal{B}}(x_{\overline{\mathcal{B}}})$ is indeed a valid testing strategy for the subset of attributes $\mathcal{B}$ and the corresponding subset of patterns $\mathcal{Y}_{\mathcal{B}}$.

Theorem 2. *If property* **(CF)** *holds for any subhierarchy $\mathcal{B}$ of $\mathcal{A}$ (including $\mathcal{A}$ itself), then the CTF strategy is optimal.*

Proof. The proof is based on a simple recursion. Let L be the depth of $\mathcal{A}$. The case $L = 1$ is obvious from the **(CF)** property. Suppose the theorem is valid for any $L < L_0$ with $L_0 \geq 2$. Now consider the case $L = L_0$.

Let T be an optimal testing strategy. From the **(CF)** property, we can assume that the test at the root of T is X_{A_1}, the attribute at the root of $\mathcal{A}$. Denote by $\mathcal{B}_1, \ldots, \mathcal{B}_k$ the subhierarchies rooted at the children of A_1, which are of depth at most $L_0 - 1$. Since $\mathcal{A} = \{A_1\} \dot\cup \mathcal{B}_1 \dot\cup \ldots \dot\cup \mathcal{B}_k$ (a disjoint union), we can partition the cost of T as follows:

$$
\begin{aligned}
E_0 C(T) &= \sum_{A \in \mathcal{A}} q_A(T) c_A + E_0 c^* |\widehat{Y}(T)| \\
&= q_{A_1}(T) c_{A_1} + \sum_{A \in \mathcal{B}_1} q_A(T) c_A + E_0 c^* |\widehat{Y}(T) \cap \mathcal{Y}_{\mathcal{B}_1}| + \ldots \\
&\qquad + \sum_{A \in \mathcal{B}_k} q_A(T) c_A + E_0 c^* |\widehat{Y}(T) \cap \mathcal{Y}_{\mathcal{B}_k}|. \qquad (16)
\end{aligned}
$$

Let us focus on the first sum. Consider the conditional strategy $T^{(1)} = T_{\mathcal{B}_1}(x_{\overline{\mathcal{B}_1}})$ and let $q_A(T^{(1)}; x_{\overline{\mathcal{B}_1}})$ be the probability (under P_0) of performing the test for $A \in \mathcal{B}_1$ using $T^{(1)}$. The tests $\{X_A, A \in \mathcal{B}_1\}$ are conditionally independent given $\{X_{\overline{\mathcal{B}_1}} = x_{\overline{\mathcal{B}_1}}, X_{A_1} = 1\}$, with powers $\{\beta_A, A \in \mathcal{B}_1\}$. By the recurrence hypothesis, we can apply the theorem to subhierarchy $\mathcal{B}_1$ and conclude that the cost of strategy $T^{(1)}$ satisfies,

$$
\begin{aligned}
E_0 \left[C(T^{(1)}) | X_{\overline{\mathcal{B}_1}} = x_{\overline{\mathcal{B}_1}} \right] &= \sum_{A \in \mathcal{B}_1} c_A q_A(T^{(1)}; x_{\overline{\mathcal{B}_1}}) + E_0 \left[c^* |\widehat{Y}(T) \cap \mathcal{Y}_{B_1}| \,\middle|\, X_{\overline{\mathcal{B}_1}} = x_{\overline{\mathcal{B}_1}}, X_{A_1} = 1 \right] \\
&\geq E_0 C(T^{(1)}_{ctf}), \qquad (17)
\end{aligned}
$$

where $T_{ctf}^{(1)}$ is the CTF strategy for hierarchy $\mathcal{B}_1$. Now, by construction of the conditional strategy, and denoting β_1 the power of test X_{A_1},

$$\forall A \in \mathcal{B}_1, \qquad E_0 q_A(T^{(1)}; x_{\overline{\mathcal{B}_1}}) = P_0[X_A \text{ performed by } T | X_{A_1} = 1] = q_A(T)(1-\beta_1)^{-1},$$

where of course the expectation is over the possible values of $x_{\overline{\mathcal{B}_1}}$, and the last equality holds beccause X_{A_1} is the first test to be performed in T. Similarly, we have

$$\begin{aligned} E_0\left[E_0 \left[c^* |\widehat{Y}(T) \cap \mathcal{Y}_{B_1}| \,\middle|\, X_{\overline{\mathcal{B}_1}}, X_{A_1} = 1 \right] \,\middle|\, X_{A_1} = 1 \right] &= E_0 \left[c^* |\widehat{Y}(T) \cap \mathcal{Y}_{B_1}| \,\middle|\, X_{A_1} = 1 \right] \\ &= E_0 \left[c^* |\widehat{Y}(T) \cap \mathcal{Y}_{B_1}| \right] (1-\beta_1)^{-1}. \end{aligned}$$

Therefore, taking expectations in (17) we obtain

$$E_0[C(T_{\mathcal{B}_1}(X_{\overline{\mathcal{B}_1}}))] = (1-\beta_1)^{-1} \left(\sum_{A \in \mathcal{B}_1} c_A q_A(T) + E_0 c^* |\widehat{Y}(T) \cap \mathcal{Y}_{B_1}| \right) \geq E_0 C(T_{ctf}^{(1)}).$$

Applying the same reasoning to the other terms of (16), we now obtain

$$E_0 C(T) \geq c_{A_1} q_{A_1}(T) + (1-\beta_1) E_0 \left[C(T_{ctf}^{(1)}) + \ldots + C(T_{ctf}^{(k)}) \right].$$

Finally, the righthand side is precisely the total cost of the CTF strategy for $\mathcal{A}$. Therefore the CTF strategy is optimal. □

We now give a sufficient condition ensuring the **(CF)** property:

Theorem 3. *Let A_1 be the coarsest test. Then the* **(CF)** *property holds under the following condition:*

$$\frac{c(A_1)}{\beta(A_1)} \leq \inf_{Z \in \mathcal{Z}(\overline{\mathcal{A}})} \sum_{A \in Z} \frac{c(A)}{\beta(A)}.$$

Corollary 1. *Consider the augmented hierarchy $\overline{\mathcal{A}}$ as a tree structure (the original hierarchy $\mathcal{A}$ can then be seen as the set of internal nodes of $\overline{\mathcal{A}}$). For any $A \in \mathcal{A}$, let $\mathcal{C}(A)$ be the set of direct children of A in $\overline{\mathcal{A}}$. Then the CTF strategy is optimal if the following condition is satisfied:*

$$\forall A \in \mathcal{A}, \frac{c(A)}{\beta(A)} \leq \sum_{B \in \mathcal{C}(A)} \frac{c(B)}{\beta(B)}. \tag{18}$$

Proof of the theorem. For this proof, it is easier to work with the "augmented" model put forward in section 5.5.4. Let T be a testing strategy for $\overline{\mathcal{A}}$ such that the first attribute to be tested is not the coarsest attribute A_1. From T, construct the strategy T_0 by "switching" test X_{A_1} to the root, i.e., perform X_{A_1} first, and when the result is 1, proceed normally through strategy T, except when test X_{A_1} is encountered in T, in which case it is not performed again and one jumps directly to its right child (corresponding to $X_{A_1} = 1$ in the original T).

Now compare the means cost of T and T_0 using formula (9). Similarly to the proof of Lemma 2, we will prove that in the convex combination defining the cost in (9), the weight of the term $c(A_1)/\beta(A_1)$ is higher in T_0 than in T, while the weights of all the other terms of the form $(\sum_{A\in Z} c(A)/\beta(A))$ are smaller or stay unchanged for all other coverings $Z \in \mathcal{Z}(\overline{\mathcal{A}})$. This together with the hypothesis of the theorem establishes property **(CF)**.

To verify the above statements about the weights of the different coverings, first call the "covering support" $CS(T)$ of a strategy T the set of coverings $Z \in \mathcal{Z}(\overline{\mathcal{A}})$ such that $P_0(\mathcal{X}_0(T) = Z) \neq 0$. It is clear from the construction of T_0 that $CS(T_0) \subset CS(T) \cup \{\{A_1\}\}$. Therefore we can restrict the analysis to the coverings in $\mathcal{Z}_0 = CS(T) \cup \{\{A_1\}\}$.

Note that $CS(T)$ is in one-to-one correspondence with the set of leaves of T having non-zero probability to be reached; for any $Z \in CS(T)$, $P(\mathcal{X}_0(T) = Z)$ is precisely the probability to reach the leaf $s_T(Z)$ of T associated with the covering Z. Along the branch leading to this leaf one finds all the events $\{X_A = 0\}$ for $A \in Z$, along with a number of other events $\{X_A = 1\}$ for A in a certain set $\mathcal{X}_1(s_T(Z))$. Therefore this probability is of the form

$$P_0(\mathcal{X}_0(T) = Z) = P_0(s_T(Z) \text{ is reached }) = \prod_{A\in Z} \beta_A \prod_{A'\in\mathcal{X}_1(s_T(Z))} (1 - \beta(A')).$$

Now with this formula in mind, any $Z \in \mathcal{Z}_0$ falls into one of the following cases:

- $Z = \{A_1\}$, in which case obviously $P_0(\mathcal{X}_0(T_0) = Z) \geq P_0(\mathcal{X}_0(T) = Z)$;
- $A_1 \in Z$ but $Z \neq \{A_1\}$, in which case $P_0(\mathcal{X}_0(T_0) = Z) = 0$;
- $A_1 \notin Z$ and $A_1 \notin \mathcal{X}_1(s_T(Z))$, in which case $P_0(\mathcal{X}_0(T_0) = Z) = (1-\beta_1)P_0(\mathcal{X}_0(T) = Z)$;
- $A_1 \notin Z$ and $A_1 \in \mathcal{X}_1(s_T(Z))$, in which case $P_0(\mathcal{X}_0(T_0) = Z) = P_0(\mathcal{X}_0(T) = Z)$.

Together, these different cases prove the desired property: $\{A_1\}$ is the only covering having higher weight in the cost of T_0 than in the cost of T. □

Corollary 1 follows immediately: Its hypothesis clearly implies that the hypothesis of Theorem 3 is satisfied for any subhierarchy of $\mathcal{A}$ and the conclusion then follows from Theorem 2.

Note that, in contrast to what happened in the case of single target detection, condition (18) falls short of being a necessary condition for ensuring the optimality of CTF strategies. To obtain a counterexample, consider the case of a depth 2 hierarchy with a coarsest attribute A_1 and two children B_1, B_2, and suppose that c^* is large enough so that the condition of Proposition 1 is satisfied, so that we may restrict our attention to complete strategies. Then one can show (by explicitly listing all possible strategies) that the CTF strategy is optimal iff

$$\frac{c(A_1)}{\beta(A_1)} \leq \inf\left(\frac{c(B_1)}{\beta(B_1)\beta(B_2)} + \frac{c(B_2)}{\beta(B_2)}, \frac{c(B_1)}{\beta(B_1)} + \frac{c(B_2)}{\beta(B_1)\beta(B_2)}\right).$$

Clearly this condition is weaker than (18).

Application to the power-based cost model: We can now look at the consequences of these results if we assume the cost model given by (5), in which case the following Corollary is straightforward:

Corollary 2. *Assume the cost of the attribute tests obeys the model given by* (5)*, with* Γ *subadditive and* $\Psi(x)/x$ *is increasing. Then the CTF strategy is optimal for any hierarchy* $\overline{\mathcal{A}}$ *for which* $\beta(A) \leq \beta(B)$ *whenever* $B \subset A$*. In that case, the optimal strategy is CTF in both resolution and power.*

Similarly, in the case of detecting a single pattern of interest, if we assume $\Gamma \equiv 1$, the CTF strategy is optimal when $\Psi(x)/x$ is increasing, a result that was already proved in (Fleuret 2000).

6.4 Simulations with an elementary dependency model

We also performed limited simulations in the case where the tests are not independent under P_0 but obey a very simple Markov dependency structure. Suppose the power of the coarsest test is β_0; the powers of subsequent tests follow a first-order Markov model depending on their direct ancestor. More precisely, the probability that a test returns 0 is γ (resp. λ) given that his father returned 0, (resp. 1) with $\gamma \geq \lambda$. The cost model used is the multiplicative cost model given $c(X_{A,\beta})$ in (5), with β the average power of the test.

We performed experiments for a set of 4 patterns and a corresponding depth 3 dyadic hierarchy, comparing the cost of the CTF strategy to the best cost among a set of 5000 randomly sampled strategies. In our experience, due to the restrained size of the problem, when there are in fact strategies better than CTF one, then this is usually detected in the simulation.

What we found was that, for a given value of γ and λ, the CTF strategy is generally optimal when $\beta_1 \leq \lambda$ (for various choices of the power function Ψ). However, when β_1 becomes too large, then the CTF strategy is not any longer optimal. Heuristically, this is because the coarse questions are then more powerful but also much too costly. The limiting value of β_1 for which CTF is optimal does not appear to be equal to the value $\beta^* = \lambda/(1+\lambda-\gamma)$, the invariant probability for the Markov model. In particular there are cases where $\lambda < \beta_1 \leq \beta^*$, (meaning that the average powers are increasing with depth) and yet CTF is not optimal.

To conclude, these very limited simulations seem to suggest that, even though the optimization problem is already somewhat complex even with a simple dependency structure and leads to challenging questions, still the optimality of CTF strategies can be expected to persist over a fairly wide range of models.

7 Optimal Strategies for Power-Based Cost and Variable Powers

7.1 Model and motivations

In this section we only consider searching for all possible patterns. The previous section dealt with a fixed hierarchy – a single test X_A at a given power $\beta(A)$ for each $A \in \mathcal{A}$. Now suppose we can have, for each $A \in \mathcal{A}$, tests of varying power; of course, a more powerful test at the same level of invariance will be more expensive. (In §10 we illustrate this tradeoff for a particular data-driven construction.) In fact, for each attribute $A \in \mathcal{A}$, we suppose there is a test for every possible power, whose cost is determined as follows:

Cost Model: Let $\Psi : [0,1] \to [0,1]$ be convex and strictly increasing with $\Psi(0) = 0$ and $\Psi(1) = 1$ and let $\Gamma : \mathbb{N}^* \to \mathbb{R}_+$ be sub-additive, with $\Gamma(1) = 1$. We suppose

$$c(X_{A,\beta}) = c(A,\beta) = c \times \Gamma(|A|) \times \Psi(\beta). \tag{19}$$

Recall that the total cost of a strategy T is given by

$$C_{test}(T) + c^*|\widehat{Y}(T)|.$$

The constant c in (19) represents the cost of a P_0-perfect test for a single pattern and the constant c^* represents the cost per pattern of disambiguating among the patterns remaining after detection. Evidently, only the ratio c/c^* matters. We are going to assume that $c^* =$

$c = \Psi(1) = 1$; note that this choice coheres with the formal interpretation of postprocessing cost as the cost of "errorless testing" put forward in §5.5.4.

For the rest of this section, we will implicitly adopt this point of view, i.e. replacing effective postprocessing cost by formal perfect tests corresponding to an additional layer of formal attributes copying the original leaves (this formal doubling of the leaf attributes allows to keep untouched the rule that no attribute can be tested twice). For these special tests only, the power cannot be chosen arbitrarily and is fixed to 1; and the strategies considered must make no errors, enforced by performing at the end of the search some of these perfect tests if needed.

We are going to focus primarily on the case $\Gamma(k) = k$. Consequently,

$$c(X_{A\cup B,\beta}) = c(X_{A,\beta}) + c(X_{B,\beta}), \quad \text{when } A \cap B = \emptyset. \tag{20}$$

This is, in effect, the choice of Γ least favorable to CTF strategies since there is no savings in cost due to shared properties among disjoint attributes. For instance, in practice, it should not be twice as costly to build a test at power β for the explanation $\{E, F\}$ as for $\{E\}$ or $\{F\}$ separately at power β, since (upon registration) these shapes share many "features" (e.g., edges; see §10). Nonetheless, with this choice of Γ the convexity assumption can now be justified as follows:

Motivation for Convexity: As usual, two tests for disjoint attributes are independent under P_0. Consider the following situation: For A and B disjoint, first test A with power β_1 and stop if the answer is positive ($X_A = 1$); otherwise, test B with power β_2 and stop. This produces a randomized, composite test for $A \cup B$ with power $\beta_1\beta_2$ and (mean) cost

$$|A|\Psi(\beta_1) + \beta_1|B|\Psi(\beta_2).$$

Contrast this with directly testing $A \cup B$ with power $\beta_1\beta_2$, which should not have greater cost than the composite test since, presumably, we have already selected the "best" tests at any given power and invariance; see §10 for an illustration. Under our cost model, this implies

$$(|A| + |B|)\Psi(\beta_1\beta_2) \leq |A|\Psi(\beta_1) + \beta_1|B|\Psi(\beta_2). \tag{21}$$

Demanding (21) for any two attributes implies (by letting $|A|/|B| \to 0$) that we should have

$$\Psi(\beta_1\beta_2) \leq \beta_1\Psi(\beta_2). \tag{22}$$

(Conversely, it is easy to see that if (22) is satisfied, then (21) holds for any $|A|, |B|$.) Since we want (22) to hold for any $\beta_1, \beta_2 \in [0, 1]$ we see (after dividing by $\beta_1\beta_2$) that (22) implies that $\Psi(x)/x$ is an increasing function. In our model, we make the stronger hypothesis that Ψ is convex in order to simplify the analysis.

Remark on Independence: It would be irrealistic to assume the independence of *all* the tests in the extended hierarchy $\widetilde{\mathcal{X}}$ were independent, rather than for families corresponding to different attributes. If it were the case, then near-perfect detection would be possible in the sense of obtaining arbitrarily low cost and error by performing enough cheap tests of high invariance. This is easy to see in the case in which $\Psi'(0) = 0$:

Example: Let $A = \mathcal{Y}$, the coarsest attribute, and suppose $\{X_{A,\beta_j}, j = 1, 2, ...\}$ are independent with $\beta_j \searrow \delta$. Consider the vine-structured testing strategy T_n which successively executes $X_{A,\beta_j}, j = 1, 2, ...n$, stopping (with label $\widehat{Y}(T_n) = \emptyset$) as soon as a null response is found and otherwise yielding $\widehat{Y}(T) = \mathcal{Y}$. Then

$$P_0(\widehat{Y}(T_n) = \mathcal{Y}) = \prod_{j=1}^{n}(1 - \beta_j) \leq (1 - \delta)^n$$

and

$$\begin{aligned}
E_0 C_{test}(T_n) &= c(X_{A,\beta_1}) + (1 - \beta_1)c(X_{A,\beta_2}) + \cdots + \prod_{j=1}^{n-1}(1 - \beta_j)c(X_{A,\beta_n}) \\
&\leq c(X_{A,\beta_1})\left[1 + \sum_{i=1}^{n-1}\prod_{j=1}^{i}(1 - \beta_j)\right] \\
&\leq |\mathcal{Y}|\Psi(\beta_1)\left[1 + \sum_{i=1}^{n-1}(1 - \delta)^i\right] \\
&\leq |\mathcal{Y}|\frac{\Psi(\beta_1)}{\delta}.
\end{aligned}$$

Since $\Psi(\delta)/\delta \to 0$, given $\epsilon > 0$, we can choose n such that $P_0(\widehat{Y}(T_n) \neq \emptyset) < \epsilon$ and $E_0 C(T_n) = E_0 C_{test}(T_n) + E_0|\widehat{Y}(T_n)| < \epsilon$.

7.2 Basic results

In the sequel, Ψ^* will denote the Legendre transform of Ψ:

$$\Psi^*(x) = \sup_{\beta\in[0,1]} (x\beta - \Psi(\beta))$$

In addition, for any $a > 0$, define:

$$\Psi_a^*(x) = a\Psi^*\left(\frac{x}{a}\right);$$
$$\Phi_a(x) = x - \Psi_a^*(x).$$

7.2.1 Optimal power selection

Consider partially specificing a strategy T by fixing the attribute A to be tested at each (internal) node but not the power. What assignment of powers (to the non-perfect) tests minimizes the average cost of T? As with dynamic programming, it is easily seen that the answer is given as follows: Start by optimizing the powers of the last, non-perfect tests performed along each branch (since the left and right subtrees of such a node have fixed, known cost), and then climb recursively up each branch of the tree, optimizing the power of the parent at each step. The actual optimization at each step is a simple calculation, summed up by the following lemma:

Lemma 3. *Consider a (sub)strategy T consisting of a test $X_{A,\beta}$ at the root, a left subtree T_L of average cost x and a right subtree T_R of average cost y. Let $\Gamma(|A|) = a$. Then, under the cost model* (19)*, the average cost of T using the optimal choice of β is given by*

$$E_0C(T) = y - \Psi_a^*(y - x) = x + \Phi_a(y - x). \tag{23}$$

In particular, if T_L is empty, then $x = 0$ and $E_0C(T) = \Phi_a(y)$. If Ψ is differentiable, the optimal choice of β is:

$$\beta^* = \begin{cases} (\Psi')^{-1}((y-x)/a) & \text{if } (y-x)/a \in \Psi'([0,1]); \\ 0 & \text{if } (y-x)/a < \Psi'(0); \\ 1 & \text{if } (y-x)/a > \Psi'(1). \end{cases}$$

More generally, Ψ admits $(y - x)/a$ as a subgradient at point β^*.

Proof. Let $T(\beta)$ denote the strategy using power β, and calculate the average cost of $T(\beta)$ as a function of β, x, y, a:

$$\begin{aligned} E_0C(T(\beta)) &= c(X_{A,\beta}) + \beta x + (1-\beta)y \\ &= a\Psi(\beta) + \beta(x - y) + y \\ &= y - a\left(\left(\frac{y-x}{a}\right)\beta - \Psi(\beta)\right). \end{aligned}$$

Now minimizing over β leads directly to (23), and the formulae for β^*, using the definitions of Ψ^* and Ψ_a^*. ☐

7.2.2 Properties of the CTF strategy

In previous sections, with fixed powers, all variations on CTF exploration (e.g., depth-first and breadth-first) had the same average cost, and hence we spoke of "the" CTF strategy. With variable powers the situation might appear different: The bottom-up optimization process in §7.2.1 for assigning the powers may lead to different mean costs for different CTF strategies. More specifically, recall that $A(s), \beta(s)$ denote the attribute and power assigned to an internal node s in a tree T. For CTF trees, it may be that $\beta^*(s)$, the optimal power at s, may depend on the position of s within T as well as $A(s)$.

The following theorem states that, in fact, as in the fixed powers case, among CTF strategies, the order of testing is irrelevant *when the powers are optimally chosen.* More precisely, the optimal power of a test depends only on the attribute being tested, specifically on the structure of the subhierarchy rooted at the attribute. Consequently, in CTF strategies, a given attribute will always be tested at the same power, which means that CTF designs can be implemented by constructing only one test per attribute – a considerable practical advantage.

Theorem 4. *For any CTF strategy T, and for any two nodes s, t in T with $A(s) = A(t)$, the optimal choices of powers are identical: $\beta^*(s) = \beta^*(t)$. In fact, the unique power assigned to an attribute $A \in \mathcal{A}$ depends only on the structure of subhierarchy $\mathcal{B}(A)$ rooted in $\mathcal{A}$. As a consequence, all CTF strategies have the same average cost.*

Whereas the principle of the proof is simple (a recursion on the size of $\mathcal{A}$), it does require some auxiliary notation, and hence we postpone it to the appendix.

Turning to the cost of the CTF strategy, it can easily be computed recursively for regular attribute hierarchies and the simple complexity function $\Gamma(k) = k$. More precisely we have the following proposition for dyadic hierarchies, in which $\beta^*_\ell(L)$ denotes the optimal power for the $2^{\ell-1}$ attributes at level $\ell = 1, ..., L$ for a hierachy of total depth L:

Theorem 5. *Let C_L denote the average CTF cost of a regular, complete dyadic hierarchy of depth L. Then:*

$$C_{L+1} = \Phi_{2^L}(2C_L) \tag{24}$$

with (formally) $C_0 = \Psi(1)/2$. Furthermore,

$$C_L/2^{L-1} \searrow \Psi'(0), \ \ L \to \infty$$

and

$$\beta^*_1(L) \searrow 0, \ \ L \to \infty. \tag{25}$$

Finally,

$$\beta_\ell^*(L) = \beta_1^*(L - \ell + 1), \;\; \ell = 1, ..., L, \tag{26}$$

from which it follows that the CTF strategy is CTF in power, i.e., power increases with depth.

Proof. Consider a (complete, dyadic) hierarchy of depth $L + 1$. The coarsest attribute has cardinality $|A_1| = 2^L$ and the (optimized, breadth-first) CTF strategy starts with the corresponding test. If $X_{A_1} = 0$, the search is over; if not, it is necessary to pay the mean cost for the two subhierarchies of depth L. We thus apply formula (23) with $x = 0, y = 2C_L$ to obtain (24). When $L = 1$ (one pattern), it is easy to check that we retrieve the right value of C_1 from formula (24) with $C_0 = \Psi(1)/2$ by noting that, in this case, $y = \Psi(1)$, which is the cost of a perfect test.

Let $U_L = C_L/2^{L-1}$. Then (24) can be rewritten as

$$U_{L+1} = \Phi_1(U_L),$$

which allows us to study the asymptotic behavior of U_L when L is large based on the function $\Phi_1(x) = x - \Psi^*(x)$. Since Ψ is convex, it follows that $\frac{\Psi(\beta)}{\beta}$ is increasing, and hence $x\beta - \Psi(\beta) \leq 0$ for all $0 \leq \beta \leq 1$ (with equality at $\beta = 0$) whenever $0 \leq x \leq \Psi'(0)$. Consequently, $\Psi^*(x) = 0$ and $\Phi_1(x) = x$ for $x \in [0, \Psi'(0)]$. Similarly, $\Phi_1(x) < x$ for $x > \Psi'(0)$. We have $U_0 = \Psi(1) \geq \Psi'(0)$ because Ψ is convex, and hence since Φ_1 is concave, $U_L \searrow \Psi'(0)$ as $L \to \infty$. Finally, from Lemma 3 we can also conclude that $\beta_1^*(L) = (\Psi')^{-1}(U_L \wedge \Psi'(1))$. The last assertion (26) of the theorem follows directly from theorem 4. □

Remarks:

- We deduce from the above results that if $\Psi'(0) = \delta > 0$, we have $C_L \sim \delta 2^{L-1}$. If, on the other hand, $\Psi'(0) = 0$, then $C_L = o(2^{L-1})$. This should be compared to the strategy of performing only (all) the perfect tests, which costs 2^{L-1}.

- Since the optimal powers are increasing with depth, if we now consider them as fixed we are in the framework of Corollary 2 ensuring that, for these choices of powers, the CTF strategy is indeed optimal.

- Finally, note that the cost of individual tests (with optimal powers) may not vary monotonically with their depth; however, the *cumulated* cost of all tests at a given depth is increasing with depth.

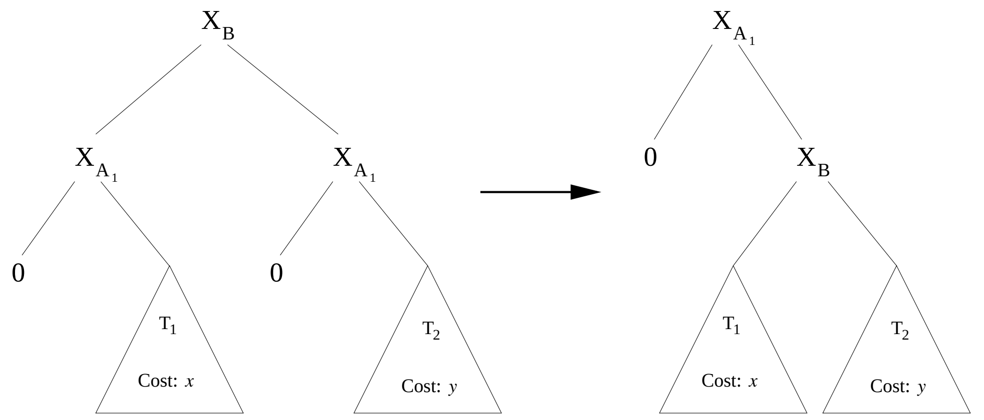


Figure 6: The context of the switching property. Attribute A_1 is the coarsest attribute in the hierarchy; hence $\Gamma(|B|) = b < \Gamma(|A_1|) = a$.

7.3 Is the CTF strategy optimal?

We have not able to prove the optimality of the CTF strategy under general conditions on Ψ, but rather only for one specific example. This is disappointing because the simulations presented later in this section strongly indicate a more general phenomenon.

If we try to follow our usual method for proving optimality, it turns out that the most difficult step is actually to prove the (CF) property. Under the (CF) property, the optimality of CTF would readily follow – it suffices to follow the lines of the proof of theorem 2 with minor adaptations, mainly replacing families $(X_A)_{A\in\mathcal{B}}$ by $(X_{A,\beta})_{A\in\mathcal{B},\beta\in[0,1]}$.

One way to prove the (CF) property is to proceed iteratively, repeatedly applying the "switching property":

Definition 9 (Switching property). *A power function Ψ has the switching property if any (sub)tree T of the form shown on the lefthand side of Figure 6, with any powers, has a larger mean cost than the tree obtained by switching the two first tests of T (shown on the righthand side of Figure 6), with optimal powers. Using Lemma 3, this inequality may be expressed as follows:*

$$\forall y \geq x \geq 0, \forall a \geq b \geq 0, \qquad \Phi_a(x + \Phi_b(y - x)) \leq \Phi_a(x) + \Phi_b(\Phi_a(y) - \Phi_a(x)). \tag{27}$$

We then have the following lemma:

Lemma 4. *The (CF) property is implied by the switching property.*

Proof. Note first that we can assume that X_{A_1}, the coarsest test, is performed at some point (at some power) along every branch of any T. If this is not the case, it can simply be added, with zero power, at the end of any branch where it does not appear without changing the cost. Now let T be a strategy such that X_{A_1} is not performed *first*. Apply the switching lemma to any subtree of T of the form shown on the lefthand side of Figure 6. In this way, X_{A_1} is pushed up in the tree while reducing the cost. This can be done repeatedly until no such subtree exists, i.e., the situation depicted in Figure 6 does not occur anywhere in T. But then the resulting tree must have X_{A_1} at the root. Otherwise, let k be the maximum depth in T were X_{A_1} appears, and let s the corresponding node. Let s' be the direct sibling of s, which exists since $k > 1$. Consider a branch b containing s'. Since X_{A_1} is performed along any branch, it must be performed somewhere in b, say at node t. But t cannot be an ancestor of s', since otherwise X_{A_1} would be performed twice along branch b, a contradiction. Nor can t be a descendant of s', since that would contradict the definition of k. Therefore X_{A_1} is performed at s', which contradicts the assumption that there is no subtree of the form shown on the left of Figure 6. This concludes the proof. □

From numerical experiments, we know however that the switching property is not satisfied for an arbitrary (convex) power function Ψ. Whereas we believe that it should be possible to prove the switching lemma under some additional conditions on Ψ, we have so far only been able to prove it for one case we refer to as the "harmonic" cost function:

$$\Psi(x) = 2 - 2\sqrt{1-x} - x \tag{28}$$

which we now investigate.

7.4 CTF optimality for the harmonic cost function

Throughout this section Ψ is given by (28). This function has the following properties:

- Ψ is convex and increasing;
- $\Psi(0) = \Psi'(0) = 0$ and $\Psi(1) = 1,\ \Psi'(1) = \infty$;
- $\Psi^*(x) = x - \frac{x}{x+1};\ \Phi_a(x) = \frac{ax}{x+a} = (x^{-1} + a^{-1})^{-1}$.

Note that x and a have symmetric roles in Φ_a, and that $\Phi_a(x)$ is the "harmonic sum" of x and a.

We first study the switching lemma in the case of an empty left subtree T_L.

Lemma 5. *Consider two tests X_A and X_B with $\Gamma(|A|) = a$ and $\Gamma(|B|) = b$. Let T_{AB} be the tree shown on the righthand side of Figure 6 with $T_1 = \emptyset$ and let T_{BA} have the same structure with X_A and X_B reversed. Then, with the optimal assignment of powers to X_A and X_B, both T_{AB} and T_{BA} have the same cost.*

Proof. By applying Lemma 3 (with $x = 0$) twice, the cost of T_{AB} is $\Phi_a \circ \Phi_b(y)$ and the cost of T_{BA} is $\Phi_b \circ \Phi_a(y)$. It is then easy to check that

$$\Phi_a \circ \Phi_b(y) = \Phi_b \circ \Phi_a(y) = \frac{aby}{ay + by + ab} = (a^{-1} + b^{-1} + y^{-1})^{-1}.$$

□

Note: Clearly, $\Phi_a \circ \Phi_b(x)$ is the harmonic sum of x, a and b. More generally, consider any "right vine" T consisting of at most one test per level of resolution. Then, under Ψ, the average cost of T is *independent* of the order in which the tests are performed; moreover, this average cost is simply the harmonic mean of the values $\Gamma(|A_i|)$ for the tests performed. In particular, this result is totally *independent* of the choice of the complexity function Γ.

We now return to the "full" switching lemma:

Theorem 6. *The switching property – and hence the optimality of the CTF strategy – holds for the harmonic power function with any complexity function* Γ.

Proof. See Appendix. □

Analogy with Resistor Networks: We conclude this section with a curious connection: Consider a hierarchy of depth L with coarsest attribute A_1 and $a_1 = \Gamma(|A_1|)$. Let C_1 be the average cost of the CTF strategy when A_1 is removed. From lemma 3, with $x = 0$ and $y = C_1$:

$$E_0(C(T_{ctf})) = \Phi_{a_1}(C_1) = \frac{a_1 C_1}{a_1 + C_1} = \left(\frac{1}{a_1} + \frac{1}{C_1}\right)^{-1}.$$

This is exactly the conductance of an electrical circuit composed of two serial resistors of conductances C_1 and a_1. Continuing, C_1 is the sum of the (CTF) costs over the two subhierarchies of depth $L - 1$; if C_1' denotes the cost of these hierarchies, the cost C_1 can be interpreted as the conductance of an electrical circuit formed from two parallel resistors, each of conductance C_1'. By an immediate recurrence that the global cost of the CTF strategy is therefore equal to the conductance of the tree-structured resistor network depicted on figure 7 (wherein a row of resistors is added at the bottom of the tree in order to represent the cost of the postprocessing, or, equivalently, perfect testing). We observe that nothing would be changed in the case of a non-symmetric, tree-structured hierarchy, even with attributes of varying complexities at the same level.

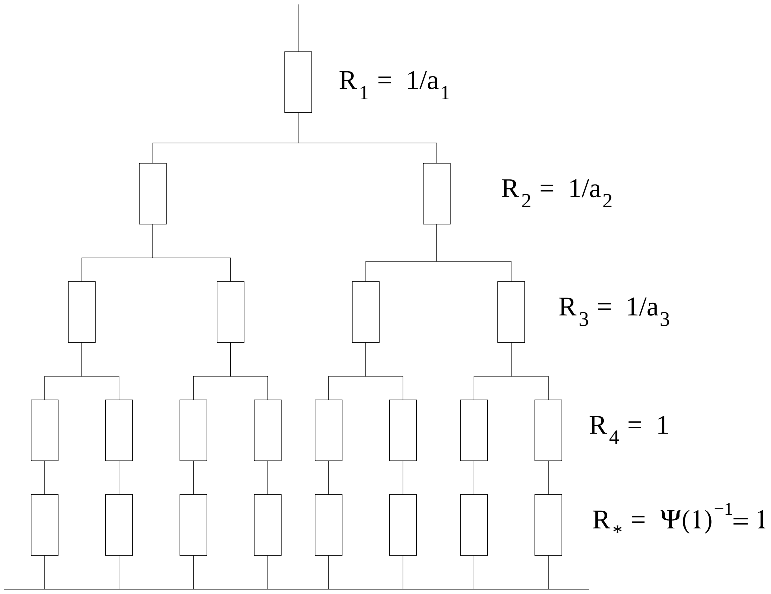


Figure 7: Tree-structured resistor network identified with the attribute hierarchy, where $a_l = \Gamma(|A_l|)$ is the complexity of attributes of level l and $R_l = 1/a_l$ is the associated resistance; note that $a_4 = 1$ by convention for the bottom attributes. The last row of resistors represents the postprocessing stage. The conductance of this circuit is exactly the CTF testing cost of the attribute hierarchy when Ψ is the harmonic power function.

Number	Ψ	Φ_1
1	$x(1-\sqrt{1-x})$	$x-(1-(1-x+\frac{1}{9}\sqrt{(1-x)^2+3})^2)(\frac{2}{3}(x-1)+\frac{1}{3}\sqrt{(1-x)^2+3})$
2	$x^2/2$	$\begin{cases} x-x^2/2 & \text{if } x<1 \\ \frac{1}{2} & \text{otherwise} \end{cases}$
3	$1-\sqrt{1-x^2}$	$1+x-\sqrt{x^2+1}$
4	$\exp(\lambda x)-1$	$\begin{cases} x & \text{if } x<\lambda \\ x-1-\frac{x}{\lambda}\left(\log\left(\frac{x}{\lambda}\right)-1\right) & \text{if } \lambda \leq x \leq \lambda e^{\lambda} \\ x-e^{\lambda}+1 & \text{if } x>\lambda e^{\lambda} \end{cases}$
5	$2-x-2\sqrt{1-x}$	$x/(1+x)$
6	$1-\sqrt{1-x}$	$\begin{cases} x & \text{if } x<\frac{1}{2} \\ 1-\frac{1}{4x} & \text{otherwise} \end{cases}$
7	$\exp(\mu x)-1-\mu x$	$\begin{cases} x\left(1+\frac{1}{\mu}\right)-\left(1+\frac{x}{\mu}\right)\log\left(1+\frac{x}{\mu}\right) & \text{if } x<\mu(e^{\mu}-1) \\ e^{\mu}-1-\mu & \text{otherwise} \end{cases}$

Table 1: Convex power functions used in our simulations. Note that Ψ_5 is the harmonic function.

7.5 Simulations

In this section we investigate the optimality of CTF search by way of simulations involving several different power functions Ψ. In every case we take $\Gamma(k)=k$. The various choices of Ψ, and corresponding functions $\Phi_1(x)=x-\Psi^*(x)$, are presented in Table 1; obviously we have chosen functions with closed-form Legendre transforms. We took $\lambda=1$ for Ψ_4 and $\mu=8$ for Ψ_7. The graphs of the different functions Ψ and Φ_1 are plotted in Figure 8.

First, we investigated the switching property, which we know to be sufficient for the optimality of T_{ctf}. To this end, we plotted the difference $\Delta(a,b,x,y)$ between the lefthand side and the righthand side of the key inequality (27). Without loss of generality, we put $a=1$. Shown in Figure 9 are the plots of $\Delta(1,b,x,y)$ for the particular choice $b=2$. (The harmonic function is not shown as we already know it has the switching property.) The switching property is satisfied if the surface lies below the $xy-plane$. One can readily see that some of these surfaces (corresponding to Ψ_2,Ψ_4,Ψ_6) clearly do not, whereas the others appear to satisfy this inequality (at least all sampled values are negative). In other experiments with other values of b for Ψ_1 and Ψ_3, we always found $\Delta\leq 0$. However we

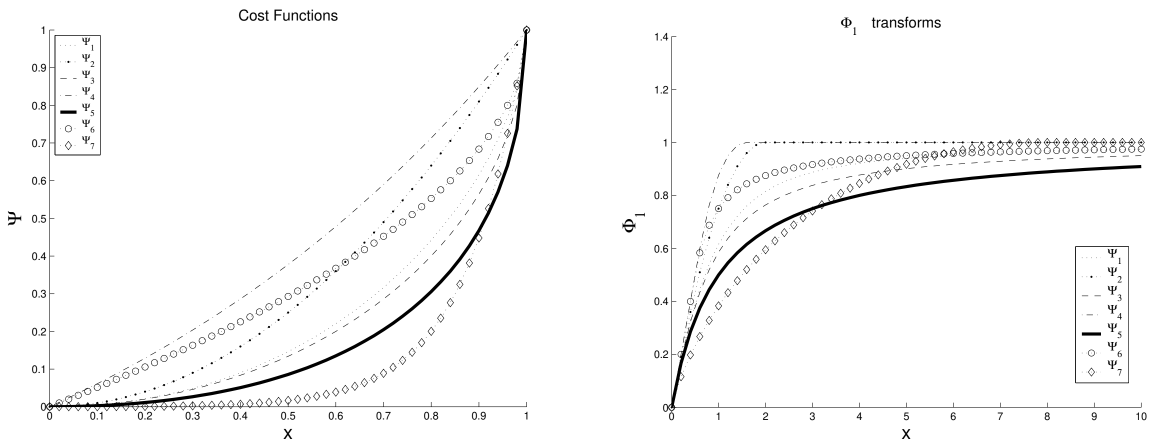


Figure 8: Graphs of the power functions Ψ and associated Φ_1-transforms used in the simulations (see Table 1) normalized so that $\Psi(1) = 1$. The bold curve corresponds to the harmonic function.

found regions with $\Delta > 0$ for Ψ_7 for higher values of b, and hence this cost function does not satisfy the switching property.

From these plots it is tempting to speculate that only power functions Ψ such that $\Psi'(0) = 0$ and $\Psi'(1) = +\infty$ can satisfy the full switching property; however these conditions are very likely not sufficient. Note that $\Psi'(0) = 0$ means that, at any given level of invariance, one can have an arbitrarily small cost-to-power ratios and $\Psi'(1) = +\infty$ means that very high powers are likely not worth the increased cost. Intuitively, both of these properties favor CTF strategies.

The second type of simulation was more direct. Strategies were sampled at random by the simplest method possible: we sampled purely attribute-based strategies T by recursively visiting nodes and choosing an attribute $A \in \mathcal{A}$ at random subject to the two obvious constraints: i) no attribute is repeated along the same branch and ii) no "useless" attribute is chosen, meaning that A consists entirely of patterns already ruled out by the previous tests. Then, for each such T, powers were individually assigned to the tests at each node in order to minimize the cost, which was compared with that of the CTF strategy. This procedure was repeated for various choices of Ψ (with $\Gamma(k) = k$) for regular, dyadic hierarchies for $|\mathcal{Y}| = 4$ patterns (i.e., $L = 3$) and for $|\mathcal{Y}| = 8$ patterns (i.e., $L = 4$). For each Ψ, we sampled several tens of thousands of trees T. (Of course the sheer number of possible strategies (modulo power assignments) in the case $L = 4$ is several orders of magnitude larger.) Summarizing

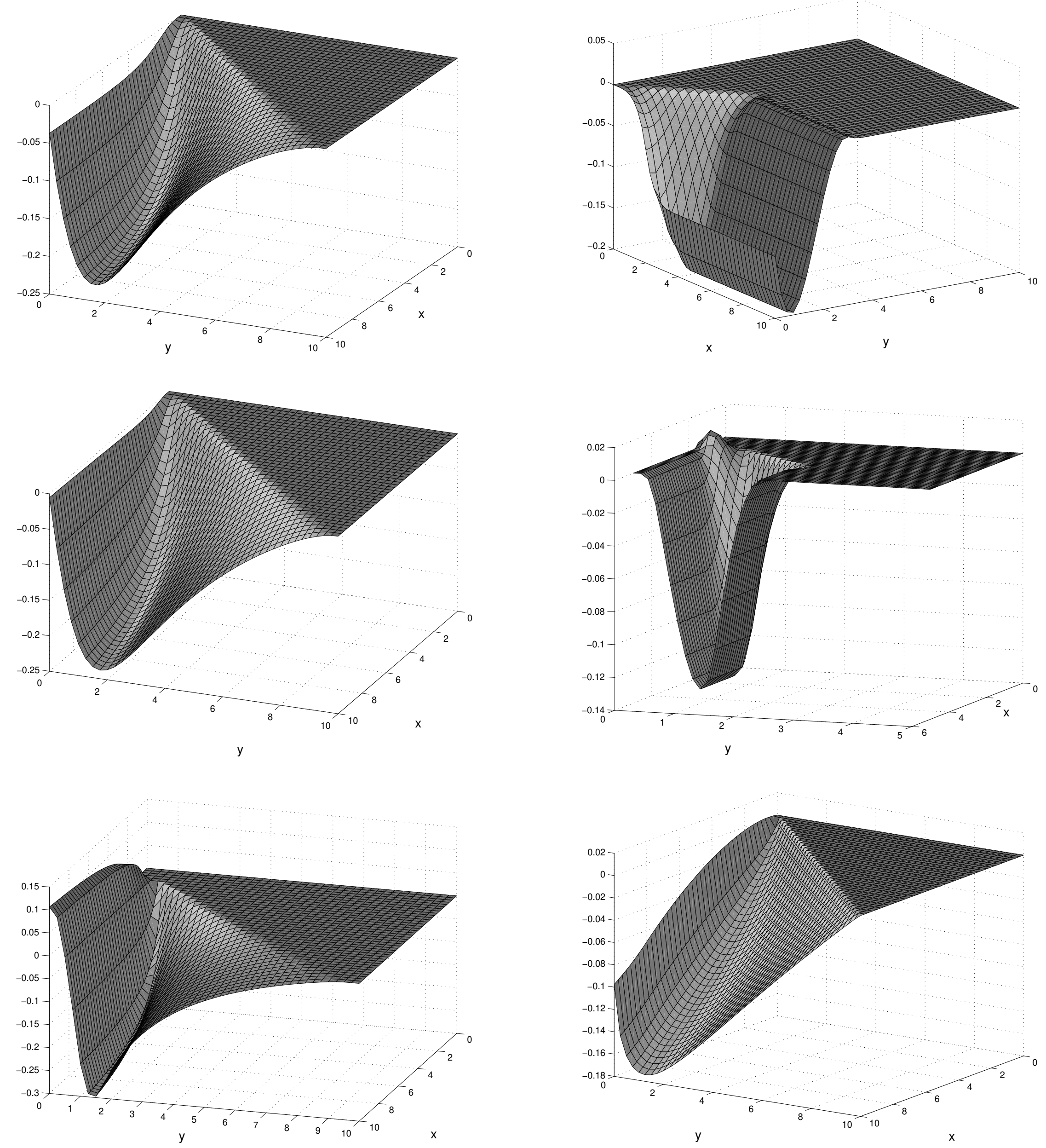


Figure 9: Empirical investigation of the switching property for six of the functions in Table 1. The surfaces represent the difference of the lefthand and righthand sides of (27) as a function of x, y for $a = 1$ and $b = 2$. Left-to-right, top-to- bottom: Ψ_1 to Ψ_7 without Ψ_5. The surfaces corresponding to Ψ_2, Ψ_4, Ψ_6 are not everywhere negative, and hence these *do not* satisfy the switching property.

our observations:

- *In all cases, the CTF strategy had lower cost than any other strategy sampled*;
- *Upon visual inspection, the best sampled strategies seemed close to the CTF strategy in the sense of only differing at relatively deep nodes.*

In conclusion, and bearing in mind the limited scope of both types of simulations, we believe the following conclusions are reasonable:

1. *The switching property is quite likely valid for cost models other than the harmonic function; however, it requires hypotheses in addition to convexity;*
2. *The optimality of the CTF strategy probably holds for a very wide range of cost models, including those which do not satisfy the switching property (for all values of a, b, x, y). As a result, requiring the switching property is likely too restrictive and, more generally, arguments based on the (CF) property may not be the most efficacious.*

8 Optimal Strategies for Usage-Based Cost

Recall from §5 that with the usage-based model the cost of a test may adapt to the strategy, subject to an overall constraint on the total amount of (computational) resources distributed among the tests. We assume that $c(X) = -\log(r(X))$, where $r(X)$ is the allocation to X, and that $\sum_X r(X) \leq R \leq 1$. Minimizing cost subject to this constraint leads to the solution $r(X)/R = q_X(T)/Q(T)$, where $Q(T) = \sum_X q_X(T)$ and (7) for the mean cost of T.

Our goal in this section is mainly to illustrate this cost model. Consequently, our analysis will be less comprehensive than in the case of power-based cost. In particular, we will make the following simplifcations:

- A fixed, tree-structured hierarchy. In this case, (7) reduces to

$$E_0 C_{test}(T) = -\sum_{A \in \mathcal{A}} q_A(T) \log(q_A(T)) + Q(T) \log(Q(T)/R) \tag{29}$$

- All tests at the same resolution have the same power:

$$P_0(X_A = 0) \equiv \beta_l, \ \ \forall A \in \mathcal{A}_l, \ \ l = 1, ..., L$$

- Power is increasing with resolution:
  (H1) $\quad \beta_1 \leq \beta_2 \leq \cdots \leq \beta_L$

- Finally, the post-processing cost c^* per pattern is so large that only complete strategies are feasible, i.e., minimize the overall cost. (Notice that, in any event, the previous, equivalent construction with perfect tests is not consistent with the usage-based cost.)

Writing $\mathbf{q} = (q_A)_{A\in\mathcal{A}}$ for a collection of real numbers in $[0,1]$ indexed by $\mathcal{A}$, let

$$H(\mathbf{q}, R) = \sum_{A\in\mathcal{A}} -\log(q_A)q_A + |\mathbf{q}|\log(|\mathbf{q}|/R), \tag{30}$$

where $|\mathbf{q}| = \sum_{A\in\mathcal{A}} q_A$. Hence

$$E_0 C_{test}(T) = H(\mathbf{q}(T), R).$$

The following elementary proposition will be useful:

Proposition 2. *The function H satisfies:*
(i) It is decreasing in R.
(ii) It is homogeneous and concave in $\mathbf{q}$.
(iii) If $\mathbf{q}, \mathbf{q}'$ have disjoint supports, then, for any R, R',

$$H(\mathbf{q}+\mathbf{q}', R+R') \leq H(\mathbf{q}, R) + H\left(\mathbf{q}', R'\right)$$

with equality iff $R = \frac{|\mathbf{q}|}{|\mathbf{q}'|}R'$.

Proof. Property (i) is immediate. For (ii), it is easily checked that $H(\lambda\mathbf{q}, R) = \lambda H(\mathbf{q}, R)$ for any $\lambda \geq 0$ and concavity results from the characterization

$$H(\mathbf{q}, R) = \inf_{\{c(A), A\in\mathcal{A}\}} \sum_{A\in\mathcal{A}} c_A q_A(T)$$

subject to the constraint

$$\sum_{A\in\mathcal{A}} \exp(-c_A) \leq R$$

as the infimum of concave (actually linear) functionals. Finally, (iii) is straightforward from equation (30), using the convexity of the function $x\log(x)$. □

8.1 Detecting one special pattern

In this case it is easy to see that, under (**H1**), the optimal strategy is in fact fine-to-coarse. Since there is no issue of invariance, the best allocation of resources is evidently to render the most powerful tests the least expensive. Clearly any test which is simultaneously cheaper and more discriminating than another is performed earlier.

In §9 we shall consider a more realistic scenario in which detecting a single pattern is to be carried out repeatedly for many different targets and we seek the fixed strategy which minimizes the *average* testing cost over all targets (without redistributing the resources for each new search). One is therefore obliged to distribute resources to all attribute tests, in which case (see §9), coarse-to-fine is generically optimal. This analysis there involves randomized procedures both for choosing the target patterns *and* for how the data are generated, and hence how the background model P_0 is determined.

8.2 Detecting any pattern

The scenario is the same one we considered in §6: There is a fixed nested hierarchy of attributes $\mathcal{A}$ and we wish to minimize $E_0 C_{test}(T)$, as given by (29), over all strategies. If all the tests are sufficiently powerful, then the CTF strategy is again optimal. The proof again utilizes the **(CF)** property and a recursion.

Lemma 6. *Under hypothesis* (**H1**) *and for the usage-based cost model, the* **(CF)** *property holds whenever:*

(i) The hierarchy $\mathcal{A}$ is tree-structured and at least dyadic;

(ii) $\beta(A_1) \geq 7/8$.

The condition that the power of the coarsest test must exceed 7/8 could likely be relaxed somewhat since the bounds used in the proof are rather crude.

Proof. We can suppose that the hierarchy has depth at least two; otherwise the lemma is trivial. Let T be an optimal strategy such that the first test performed is not the coarsest one, X_{A_1}. Then T must perform at least two tests before reaching a terminal node: Since A_1 has at least two children, the only way to finish – to determine $\widehat{Y}(\mathcal{X})$ – with one test would be to perform X_{A_1} first and obtain a negative answer. In all other cases at least two tests are required. As a result, $Q(T) \geq 2$ and, consequently, $H(\mathbf{q}(T), R) \geq 2\log(2/R)$.

On the other hand, we can compute explicitly the cost of T_{ctf} and show it is at most $2\log(2/R)$. In T_{ctf}, test X_A, $A \in \mathcal{A}_l$, is performed if and only if all the tests corresponding to his ancestors in the hierarchy are positive. Therefore, with $p_i = 1 - \beta_i$,

$$A \in \mathcal{A}_l \Rightarrow P_0(X_A \text{ performed by } T_{ctf}) = p_1 \dots p_{l-1};$$

so that

$$\begin{aligned} H(\mathbf{q}(T_{ctf}), R) &= -2p_1 \log(p_1) - 4p_1p_2 \log(p_1p_2) - \dots - 2^{L-1} p_1 \dots p_{L-1} \log(p_1 \dots p_{L-1}) \\ &\quad + Q(T_{ctf}) \log(Q(T_{ctf})) - Q(T_{ctf}) \log(R), \end{aligned}$$

where

$$Q(T_{ctf}) = 1 + 2p_1 + \ldots + 2^{L-1}p_1 \ldots p_{L-1}.$$

Using the fact that the p_i's are decreasing, that $p_1 \leq 1/8$ and that the function $-x\log(x)$ is increasing for $x \leq 1/e$ we have:

$$-\sum_{i=1}^{L-1} 2^i p_1 \ldots p_i \log(p_1 \ldots p_i) \leq -\sum_{i\geq 1}(2p_1)^i i \log(p_1) = -\log(p_1)\frac{2p_1}{(1-2p_1)^2}$$

and

$$Q(T_{ctf}) \leq \sum_{i\geq 0}(2p_1)^i = \frac{1}{1-2p_1},$$

so that

$$\begin{aligned} H(\mathbf{q}(T_{ctf}), R) &\leq -\log(p_1)\frac{2p_1}{(1-2p_1)^2} - \frac{\log(1-2p_1)}{1-2p_1} - \frac{1}{1-2p_1}\log(R) && (31)\\ &\leq 2\log(2) - 2\log(R) \leq H(\mathbf{q}(T), R). && (32) \end{aligned}$$

The last bound (32) is due to the fact the first two terms of (31) are decreasing in p_1 and since $p_1 \leq 1/8$, it suffices to check e.g. numerically that the inequality holds for $p_1 = 1/8$. Hence T cannot be an optimal strategy, which concludes the proof. □

We then have the following main theorem:

Theorem 7. *Under hypothesis* **(H1)** *and the assumptions (i) and (ii) of Lemma 6, the CTF strategy is optimal:*

$$\forall R \leq 1, \quad \forall T, \qquad H(\mathbf{q}(T_{ctf}), R) \leq H(\mathbf{q}(T), R).$$

Proof. As in the proof of Theorem 2 in §6, we proceed by recurrence on the depth L of the attribute hierarchy $\mathcal{A}$. The result is trivial for $L = 1$; suppose it has been established for $L < L_0$ and consider a hierarchy of depth L_0.

Let T be an optimal strategy. From Lemma 6, the first test to be performed has to be X_{A_1}. Let us denote by $\mathcal{B}_1, \ldots, \mathcal{B}_k$ the subhierarchies of attributes rooted at the children of A_1, which are of depth at most $L_0 - 1$, so that $\mathcal{A} = \{A_1\} \,\dot\cup\, \mathcal{B}_1 \,\dot\cup \ldots \dot\cup\, \mathcal{B}_k$.

Consider the conditional strategy $T^{(1)} = T_{\mathcal{B}_1}(X_{\overline{\mathcal{B}_1}})$. Let $q_A(T^{(1)}; X_{\overline{\mathcal{B}_1}})$ be the probability of performing test $A \in \mathcal{A}$ using strategy $T^{(1)}$. Let R_1 denote the amount of resources spent in $T^{(1)}$, i.e., the sum of the resources allocated to the tests for attributes in $\mathcal{B}_1$ under T:

$$R_1 \doteq R\frac{\sum_{A\in\mathcal{B}_1} q_A(T)}{Q(T)} \leq 1.$$

Since the tests are independent, the joint distribution of $\{X_A, A \in \mathcal{B}_1\}$ remains unchanged conditional on $(X_{\overline{\mathcal{B}_1}}, X_{A_1} = 1)$. Therefore hypotheses **(H1)**, (i) and (ii) are satisfied for subhierarchy $\mathcal{B}_1$ and we can apply the hypothesis of recurrence to this subhierarchy, with available resource R_1, and thereby obtain, for any $X_{\overline{\mathcal{B}_1}}$,

$$H(\mathbf{q}(T^{(1)}; X_{\overline{\mathcal{B}_1}}), R_1) \geq H(\mathbf{q}(T^{(1)}_{ctf}), R_1), \tag{33}$$

where $T^{(1)}_{ctf}$ is the CTF strategy for subhierarchy $\mathcal{B}_1$.

Now, let $\mathbf{q}^{(1)} = (q^{(1)}_A)_{A \in \mathcal{A}}$ where

$$q^{(1)}_A = \begin{cases} q_A(T) & \text{if } A \in \mathcal{B}_1, \\ 0 & \text{otherwise.} \end{cases}$$

Note that, by definition, we have $\mathbf{q}^{(1)} = (1 - \beta_1) E_0[q_A(T^{(1)}; X_{\overline{\mathcal{B}_1}})]$, where the expectation is over the possible values of the tests $X_{\overline{\mathcal{B}_1}}$.

By concavity of H in $\mathbf{q}$, we then have

$$\begin{aligned} H(\mathbf{q}^{(1)}, R_1) = H((1-\beta_1)E_0[\mathbf{q}(T^{(1)}; X_{\overline{\mathcal{B}_1}})], R_1) &\geq E_0\left[H((1-\beta_1)\mathbf{q}(T^{(1)}; X_{\overline{\mathcal{B}_1}}), R_1)\right] \\ &\geq H((1-\beta_1)\mathbf{q}(T^{(1)}_{ctf}), R_1). \end{aligned}$$

Now the same reasoning can be applied to each of the projections $T_{\mathcal{B}_i}(X_{\overline{\mathcal{B}_i}}), 1 \leq i \leq k$ of strategy T on subhierarchies $\mathcal{B}_i$ with resources R_i, so that

$$H(\mathbf{q}^{(i)}, R_i) \geq H((1-\beta_1)\mathbf{q}(T^{(i)}_{ctf}), R_i).$$

Denote by $\mathbf{q}^{(0)}$ the collection of reals indexed by $\mathcal{A}$ such that $q^{(0)}_{A_1} = q_{A_1}(T)$, and $q^{(0)}_A = 0$ for $A \neq A_1$, and put $R_0 = R\, q_{A_1}(T)/Q(T)$. The cost of strategy T can then be written as

$$\begin{aligned} E_0 C_{test}(T) &= H(\mathbf{q}(T), R) \\ &= H(\mathbf{q}^{(0)} + \mathbf{q}^{(1)} + \ldots + \mathbf{q}^{(k)}, R_0 + R_1 + \ldots + R_k) \\ &= H(\mathbf{q}^{(0)}, R_0) + H(\mathbf{q}^{(1)}, R_1) + \ldots + H(\mathbf{q}^{(k)}, R_k) \\ &\geq H(\mathbf{q}^{(0)}, R_0) + H((1-\beta_1)\mathbf{q}(T^{(1)}_{ctf}), R_1) + \ldots + H((1-\beta_1)\mathbf{q}(T^{(k)}_{ctf}), R_k) \\ &\geq H(\mathbf{q}^{(0)} + (1-\beta_1)(\mathbf{q}(T^{(1)}_{ctf}) + \ldots + \mathbf{q}(T^{(k)}_{ctf})), R_0 + R_1 + \ldots + R_k), \end{aligned}$$

where the third equality and the last inequality hold by property (iii) of Proposition 2.

Finally, since $\mathbf{q}^0 + (1-\beta_1)(\mathbf{q}(T^{(1)}_{ctf}) + \ldots + \mathbf{q}(T^{(k)}_{ctf}))$ is the vector of probabilities corresponding to the CTF strategy on $\mathcal{A}$, the proof is finished. $\square$

9 Extended Scenario: Multiple Searches

Previously, $\mathcal{Y}$ represented a particular family of patterns (or explanations) and $\mathcal{A}$ was the corresponding family of attributes (or partial explanations). In this section we consider strategies which are optimal when mean computation is itself averaged over many detection experiments corresponding to different subsets $\mathcal{Y}$ with hierarchies $\mathcal{A} = \mathcal{A}(\mathcal{Y})$. In fact, to make the averaging tractable, we will select hierarchies at random from a very large pool representing all attributes of interest for all patterns. In terms of optimal strategies, nothing changes in the case of power-based cost because those results are valid hierarchy by hierarchy. However, for the usage-based cost, assuming the chosen strategy does not depend on the specific hierarchy (i.e., we are not going to "rewire" the system for each new search – see the discussion in §9.3), the results for fixed hierarchies are different from those for random ones because the resources must be distributed over a larger number of tests.

Let $\widetilde{\mathcal{Y}}$ represent a very large family of patterns and let $\widetilde{\mathcal{A}}$ be a very large pool of attributes divided into disjoint levels of resolution

$$\widetilde{\mathcal{A}} = \widetilde{\mathcal{A}}_1 \,\dot{\cup}\, \ldots \,\dot{\cup}\, \widetilde{\mathcal{A}}_L.$$

With $N_l = |\widetilde{\mathcal{A}}_l|, l = 1, ..., L$, we suppose that $N_1 \leq \cdots \leq N_L$ and that $N_l \geq 2^{l-1}$. The attributes in $\widetilde{\mathcal{A}}_1$ are the coarsest in the sense of being the most commonly observed and belonging to the greatest number of patterns. We write $\widetilde{\mathcal{A}}$ instead of $\mathcal{A}$ to emphasize that $\widetilde{\mathcal{A}}$ is much larger: It represents *all* the attributes (natural groupings) for *all* conceivable patterns of interest (or perhaps the union of attribute hierarchies $\mathcal{A}$ over many different subfamilies $\mathcal{Y}$) whereas, previously, $\mathcal{A}$ represented natural groupings for some *particular subfamily* $\mathcal{Y} \subset \widetilde{\mathcal{Y}}$ of patterns.

Parts and Clutter: In visual recognition, it is not uncommon to conceive of patterns as constructed from "parts" which are "reusable" in the sense of being common to many different objects. Regarding each $A \in \widetilde{\mathcal{A}}$ as such a "part" (or "feature") is basically the "dual" outlook of the previous sections: Rather than beginning with abstract patterns and defining attributes as distinguished subsets, we start instead with abstract attributes and generate patterns by randomly joining attributes. In other words, *patterns are distinguished conjunctions of attributes.* In this setting, one can imagine tens of thousands of patterns of interest (such as couples $\{physical\, object, pose\}$) constructed from some thousands of attributes.

Similarly, one can conceive of "clutter" as constructed from *the same components* as patterns. This provides an "alternative hypothesis" to the existence of patterns which is more realistic than, for instance, white noise models. It is unrealistically easy to separate patterns from white noise, and much more difficult to separate them from highly structured noise in the sense of parts of patterns arranged in a non-distinguished manner.

9.1 Background model

As before, there is a binary test X_A for each attribute $A \in \widetilde{\mathcal{A}}$. Let $\mathbf{X}$ denote this family of tests, which we can think of as replacing the data themselves. A backround model refers to the distribution of $\mathbf{X}$ under the assumption that $\mathbf{Y} = \emptyset$; or that $Y = \emptyset$ for a subfamily of patterns, allowing patterns outside $\mathcal{Y}$ to be present.

We imagine background data as generated from randomized selections of attributes: For each $l = 1, ..., L$, each attribute $A \in \widetilde{\mathcal{A}}_l$ has probability $1 - \beta_l$ to appear. As in §8, we suppose that **(H1)** holds: $\beta_1 \leq \beta_2 \leq \cdots \leq \beta_L$. Moreover, under $\mathbf{Y} = \emptyset$, the appearence of an attribute is independent of that of all other attributes in the system. Thus,

$$P_0(X_A = x_A, A \in \widetilde{\mathcal{A}}) = \prod_{l=1}^{L} \prod_{A \in \widetilde{\mathcal{A}}_l} \beta_l^{1-x_A} (1 - \beta_l)^{x_A}. \tag{34}$$

9.2 Hierarchies

In general, one should expect that $|\widetilde{\mathcal{Y}}| \ll N_1 \cdots N_L$; that is, the number of "interesting" patterns is far smaller than the number of ways in which attributes can be combined across levels into conjunctions. A *hierarchy* of attributes $\mathcal{A}$ will refer to tree-structured subset: Exactly one attribute, A_1, is selected from $\widetilde{\mathcal{A}}_1$, then exactly ν attributes are selected from $\widetilde{\mathcal{A}}_2$, and so forth where at each step ν attributes are selected from $\widetilde{\mathcal{A}}_{l+1}$ for each attribute previously chosen from $\widetilde{\mathcal{A}}_l$. The hierarchy reduces to a vine when $\nu = 1$. As usual, we imagine these hierarchies to be very special in the sense of representing coherent partial explanations for a particular family $\mathcal{Y}$ of patterns; therefore we still identify each complete chain with the *detection* of a pattern. Of course if the corresponding pattern is present, then all the tests in the chain must be positive, but not conversely: a chain of positive test results does not imply the existence of the pattern and, indeed, there may be many such positive chains with positive P_0 probability.

9.3 Optimal testing strategies: usage-based cost

9.3.1 Randomized patterns

First consider detecting a single pattern. If the resources are allocated to exclusively to the corresponding attributes we then have already observed in §8.1 that the fine-to-coarse strategy is optimal. This scenario is not interesting because we want a strategy which in some sense is optimal over many repetitions with many different patterns. In principle, one could fix a set of patterns $\widetilde{\mathcal{Y}}$ and average over choices of the target. However, this would be difficult to analyze mathematically because the results might depend on the particular $\widetilde{\mathcal{Y}}$.

As a result, we will identify choosing a pattern at random with (uniformly) sampling exactly one attribute from each level $\widetilde{\mathcal{A}}_l, l = 1, ..., L$. While this contradicts the fact that the set of possible patterns $\widetilde{\mathcal{Y}}$ should in fact be much smaller than the set of all possible combinations of attributes accross resolution levels, this should serve as a reasonable first approximation to first selecting one large family $\widetilde{\mathcal{Y}}$ and averaging computation only over the corresponding detections. Now resources must be allocated in advance to the entire pool of attributes tests. Of course we still want to minimize the average cost (7), assuming that the same strategy is applied at each round, independently of the selected pattern. If we assume that (N_l) is rapidly increasing with l, it is no longer obvious what is the optimal strategy since more powerful tests are also more numerous, and hence it is less of an advantage to devote resources to them. The following theorem tells us that if (N_l) increases quickly enough, the optimal strategy is in fact CTF.

In the remainder of §9 $p_l = 1 - \beta_l, l = 1, ..., L$.

Theorem 8. *Assume that hypothesis* **(H1)** *is satisfied* ($p_1 \geq \cdots \geq p_L$). *Then, if R denotes the available resources, the following condition is sufficient for the optimality of the CTF strategy:*

$$\forall l : 1 \leq l \leq L-1,$$

$$\log\left(\frac{N_{l+1}}{N_l}\right) \geq \frac{1}{1-p_l}\left\{\max_{l\leq k\leq L-1} p_k \log(N_{k+1}) - \frac{p_l \log(p_l)}{1-p_l} + p_l\left(1 + \log(2) - \log(R) + \sum_{k=1}^{l-1}(p_k - \log p_k)\right)\right\}. \tag{35}$$

Proof. Again, we start with the **(CF)** property; see the Appendix. □

The condition of the Theorem, although involved, can be easily checked for given sequences (N_l) and (p_l). If we suppose (N_l) grows exponentially, the condition is satisfied, for example, for $N_l = 2^{l-1}, p_l = 1/(8l^{1.1})$, and for $N_l = 3^{l-1}, p_l = 1/(6l^{1.1})$. More generally, if $p_l = C/l^{\alpha}$, with $\alpha > 1$, and the $N_l's$ have at most an exponential growth, then the right-hand side of (35) converges to zero as $l \to \infty$, and therefore a sub-exponential growth of the sequence (N_l) is actually sufficient.

Special Case: A "Check Who" Game: As a particular instance of the randomized target model in this section, consider a "Check Who" game played as follows: Two "targets" y^* and y are chosen independently and at random by picking one of the N_l attributes at random from each level l. The player is given y^* (the "template") and must determine, with minimum average cost, whether or not $y = y^*$ by sequentially asking checking for the attributes which characterize y^*; of course the answers are provided by y. The cost of the questions follows the usage-based model. It is readily seen that this is a special case of the above framework with $p_l = 1 - \beta_l = 1/N_l$. There is actually a small difference in that here the "tests" X_A are not independent at a given level of resolution (since there is exactly one positive test at each level), although independence among levels still holds, which is actually sufficient. In other words, it is straightforward to see that the optimality results above remain unchanged. The hypothesis of the theorem is, for instance, satisfied for $N_l = 24\lceil 1.5^l \rceil$.

9.3.2 Randomized hierarchies

The situation is entirely analogous to the previous case of a single target, except that now there are many hierarchies and we desire the best strategy an average sense. Consequently, we shall suppose that the selection procedure for a hierarchy $\mathcal{A}$ that was described in §9.2 is randomized. We consider only binary trees. Thus we draw 2^{l-1} attributes at random from $\widetilde{\mathcal{A}}_l$ for each $l = 1, ..., L$ and form a corresponding binary tree. Of course the set of patterns for this iteration is identified with the set of branches of the resulting labeled binary tree. The goal is then to determine $\widehat{Y}$ as usual – the set of all patterns confirmed at every resolution. The strategy must not depend on the specific attributes drawn and performance is measured, as usual, by the mean of $C_{test}(T)$ relative to the background law P_0 described in §9.2.

In this case again we prove the optimality of the CTF strategy under mild conditions on the behavior of the sequences (N_l) and (β_l).

Let $\mathcal{D}$ denote a complete binary tree of depth L. For any node s of $\mathcal{D}$, denote by $A(s)$ the attribute attached to s at the current round; hence $A(s)$ is a random variable since $\mathcal{A}$ is randomly selected. For simplification, we will denote by X_s the test $X_{A(s)}$. Note that the family $(X_s)_{s\in D}$ are (still) independent random variables under P_0 (because we assumed that the powers of all tests at the same level of resolution have the same power). This is really the only hypothesis that is needed.

Theorem 9. *For any resource $R \leq 1$, the CTF strategy is optimal if* **(H1)** *and the following assumptions are satisfied:*

$$(i)\forall 1 \leq l \leq L-1,$$
$$\log\left(4\frac{N_{l+1}}{N_l}\right) \geq \frac{2}{1-2p_l}\left(\max_{l\leq k\leq L-1} p_k \log(N_{k+1}) + \frac{-\log(2p_l)p_l}{1-2p_l}; -\frac{1}{2}\log(1-2p_l)\right)$$
$$(ii) p_1 \leq 1/(2\sqrt{e}).$$

The proof is slightly more complex that of Theorem 8 (due to the randomization process), but uses some of the same tools, including a recursion based on **(CF)**, conditional strategies and the concavity of H; it appears in the Appendix. Again, the conditions of the Theorem are mild enough; an example of sequences satisfying them is $N_l = 2^l, p_l = 0.15/l$.

10 Application: Rectangle Detection

In order to illustrate numerically the quantities appearing in our analysis, and to check whether the cost model is reasonable in at least one concrete setting, we outline an algorithm for detecting rectangles amidst clutter due to Franck Jung and based on the framework in this paper. (It was developed in order to automate cartography by detecting buildings in aerial photographs (Jung 2002).) Only those aspects which shed light on the mathematical analysis are described, and hence many details are omitted. The interested reader can consult Jung 2001 for some additional information about scene synthesis and test construction, and a complete accounting will appear elsewhere.

The goal is to find and localize rectangles in a "scene" of the type shown in Figure 10. There are many ways to do this (automatically). For instance one could imagine a Bayesian approach based on two distributions: a prior on interpretations (e.g., a generative model involving marked point processes) and a data model for the observed pixel intensities given an interpretation (e.g., allowing for clutter and imperfect, "noisy" rectangles). Or one could train a multilayer perceptron or support vector machine based on labeled subimages. For the artificial problem illustrated in Figure 10, with limited noise and clutter, it would not be surprising to obtain a decent solution with standard methods. Our intention is only to demonstrate how this might be done in an especially efficient manner with a sequential testing design.

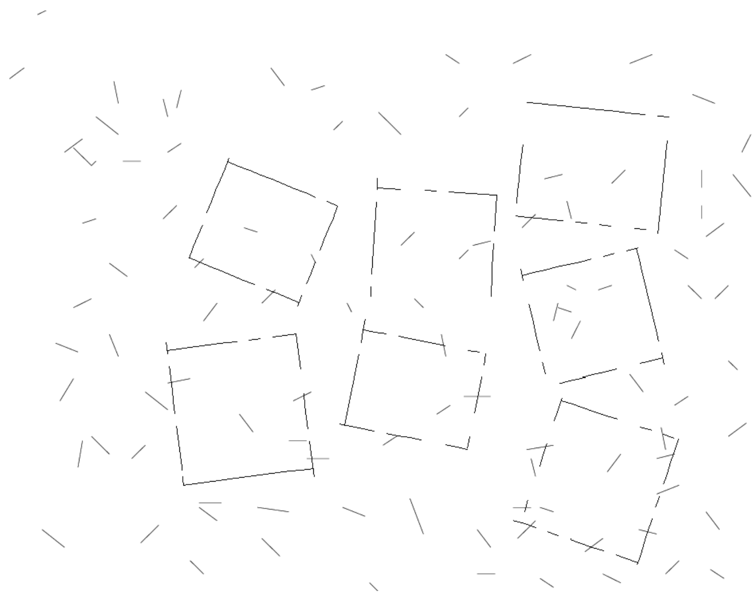


Figure 10: Example of a scene with clutter.

10.1 Problem formulation

It is clearly impossible to find common but localized attributes of two rectangles with significantly different (geometric) poses, say far apart in the scene. Consequently, we divide the whole scene into non-overlapping 5×5 regions and apply a simple, "divide-and-conquer" strategy based on location. Each 5×5 region R is visited in order to determine if there is a rectangle in the scene whose distinguished point (say the center) lies in R; depending on its scale, the rectangle itself will enclose some portion the of scene surrounding R. We can assume that the scale of the rectangle (defined below) is restricted to a given range whose lower end represents the smallest rectangles we attempt to find. Larger rectangles are found by repeatedly downsampling the image and parsing the scene in the same way; this is how the faces in Figure 2 were detected. Similarly, the orientation of the rectangle is restricted to a given range of angles; other orientations could be found by repeating the process with suitably transformed detectors.

The loop over regions R is the "parallel component" of the algorithm and not of interest here. The serial component is a CTF search to determine if there is a rectangle whose center lies in a fixed region R. This the heart of the algorithm and the real source of efficient computation.

The "pose" θ of a rectangle is characterized by four parameters: orientation (ϕ), center

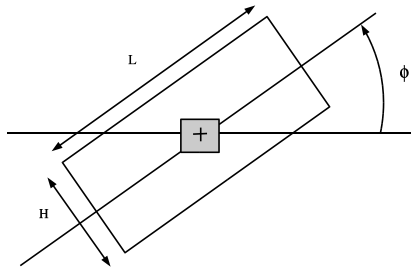


Figure 11: Each rectangle is characterized by its location, height, length and orientation.

(U), height (H) and length (L); see Figure 11. Actually, the scenes are generated in the continuum (high resolution) and a "pixel" simply represents a small square; this explains the fractions of pixels appearing in the discussion below.

Relative to a fixed, reference region R_0, the pose space $\Theta = \{\phi, U, H, L\}$ is restricted as follows:

$$\phi \in [-\frac{\pi}{8}, \frac{\pi}{8}],\ U \in R_0,\ \ H, L \in [10, 14]. \tag{36}$$

The hypothesis $Y = 0$ stands for "no rectangle with these parameters" and is evidently a complex mixture of configurations due to clutter, larger rectangles and nearby ones.

Scenes are synthesized as follows: First, rectangles are inserted by randomly choosing a location, scale and orientation. A rectangle at a given pose is a high resolution silhouette. The rectangles are degraded by dividing the silhouette into small pieces ("edges") and independently removing each one with probability 0.15. Noise is added by randomly selecting locations and orientations at which to add an edge (see below). Finally, "clutter" (structured noise, a first-order obstacle in visual recognition) is introduced by removing entire sides of rectangles with probability 0.2, in which case the resulting structure is considered part of the background (although achieving robustness to occlusion would argue for maintaining the pattern label).

10.2 Patterns and attributes

In order to define the set of explanations $\mathcal{Y}$, we partition the pose space Θ into small subsets. A "pattern" or "explanation" $y \in \mathcal{Y}$ is then a subset of poses at approximately the resolution of the pixel lattice. In fact, these subsets are, by definition, the cells at the finest layer of the attribute hierarchy - a recursive partitioning of Θ of the type used throughout the paper, yielding $\Theta = \{A_{l,k}\}$. In this case Y represents the true pose at the pixel resolution.

There are $L = 6$ levels which corresponds to five splits: two (binary) on orientation, one (quaternary) on position and two (binary) on scale (one on height and one on length). In particular there are $|\mathcal{Y}| = 64$ finest cells, each with resolution 1.25 pixels in location, two pixels in length and height, and $\pi/16$ radians in tilt. Let $\eta_l = |A|, A \in \mathcal{A}_l$. The quaternary split happens to be the second one, and hence $(\eta_1, ..., \eta_6) = (64, 32, 8, 4, 2, 1)$.

10.3 Tests

As in the references cited in §4, the tests X_A are extremely simple image functionals based on local features called "spread edges." Starting with virtually any standard "edge detector" (a local operator which identifies the position and orientation of "significant" intensity transitions), and given a position U, an orientation ϕ and a "spread" $\sigma \in \{1, 2, ...\}$, the "spread edge" ξ indexed by (U, ϕ, σ) is the binary image functional which takes the value "1" if there is an edge of orientation ϕ anywhere along a strip of pixels of length σ orthogonal to ϕ and centered at U. This situation is depicted in Figure 12 in the case of a vertical orientation. In this case $\xi = 1$ since indeed the (horizontal) strip does cross a vertical boundary (shown at low resolution). The parameter σ adapts the spread edge to any given level of affine invariance – the larger σ, the greater the number of possible boundary segments detected by ξ, of course at the expense of precision and sensitivity to clutter.

Each test X_A is based on a threshold τ and a collection $\mathcal{S}$ of spread edges with a common spread σ but varying positions and orientations:

$$X_A = \begin{cases} 1 & \text{if } \sum_{\xi \in \mathcal{S}} \xi \geq \tau \\ 0 & \text{otherwise} \end{cases}$$

Thus, evaluating X_A consists of checking for at least τ spread edges among a special ensemble dedicated to A. The parameter σ is adjusted to achieve the desired level of invariance.

Recall our basic constraint: $P(X_A = 1|Y \in A) = 1$ for every test X_A. In particular, we demand that $X_A = 1$ when the image data surrounding R_0 contains a rectangle whose pose belongs to A. (Obviously a region R is checked for the center of a rectangle by translating the tests accordingly and processing the surrounding image data.) Of course the test may also respond positively in the absence of such a rectangle, due to clutter and nearby rectangles; the likelihood of this happening is precisely $1 - \beta_A = P(X_A = 1|Y = 0)$. Intuitively, we expect that high power will only be possible at low invariance (specific poses). The power β_A is estimated from large samples of randomly selected background subimages.

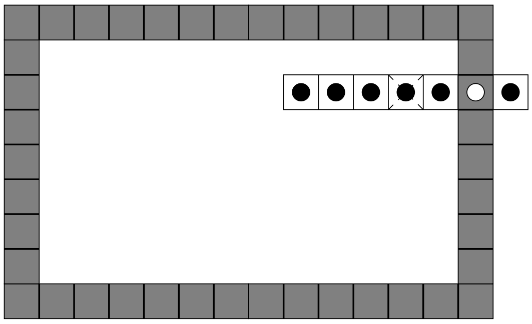


Figure 12: Example of a rectangle boundary "detected" by a spread edge.

Actually, we build many tests of varying powers for each $A \in \mathcal{A}$, each one corresponding to a different collection $\mathcal{S}$ of spread edges. Identifying $\mathcal{S}$ and τ is a problem in statistical learning. We use a fairly simple procedure, briefly summarized here; the details may be found in (Jung 2002). Fix $A \in \mathcal{A}$ and make a training set of subimages, each containing a rectangle with a pose in A; for example, this can be done by first making large scenes and then extracting appropriate subimages. The corresponding estimate of $P(D|Y \in A)$ for an event D is denoted $\widehat{P}(D|Y \in A)$. Assemble a very large family of spread edges ξ for which $\widehat{P}(\xi = 1|Y \in A)$ is reasonably large, say at least 0.5. Now subsample this family to produce a set $\mathcal{S}$ (say, 100 spread edges) and choose τ to be the maximum threshold for which $\widehat{P}(X_A = 1|Y \in A) = 1$; thus, for every training subimage, at least τ of the spread edges in $\mathcal{S}$ respond positively. Selecting $\mathcal{S}$ can be done recursively, adding one ξ at a time, guided by maximizing the current threshold to preserve invariance. Repeating this process, we can make a whole family $\{X_{A,\beta}, \beta \in \mathcal{B}(A)\}$ of tests for attribute A of varying powers.

The cost $c(X_A)$ is defined as the number of pixels involved in evaluating X_A, which is the number of pixels which participate in the definition of any $\xi \in \mathcal{S}(X_A)$. Assuming no preprocessing other than extracting and storing all the edges in the scene (and no other shortcuts in evaluating a test), this is roughly proportional to the actual algorithmic cost in CPU terms.

In Figure 13 we plot cost vs. power for the family of all tests generated for the root cell, A_0, referred to as "cell 1", and one of its two daughter cells, referred to as "cell 2". Thus each point is a pair $(\beta, c(X_{A,\beta}))$. For the root cell we cannot make tests with arbitrarily large power, at least not with such simple functionals. Figure 14 shows all the "best tests" for the depth two cell in Figure 13 – those which are not strictly dominated by another test with respect to both cost and power. Plots for cells at other depths are very similar, and

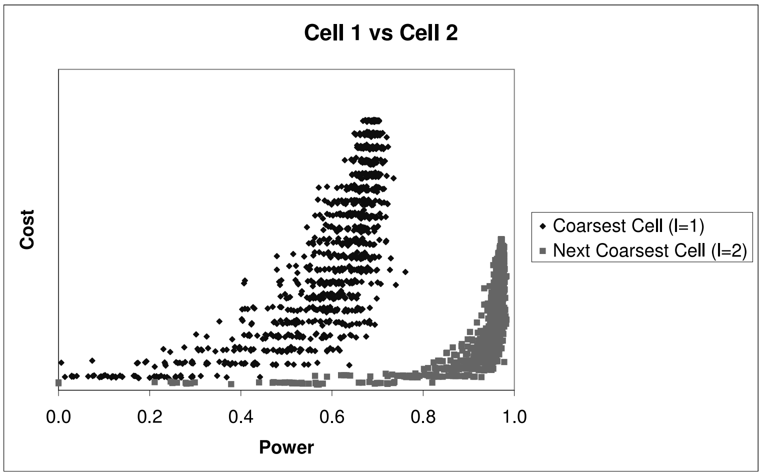


Figure 13: Cost vs power curve for attributes of depth one and two.

the convexity assumption made in §6 and §7 seems to be roughly satisfied.

Finally, one can ask whether the functional form of our global cost model, namely $c(X_{A,\beta}) = \Gamma(|A|) \times \Psi(\beta)$, is consistent with the data. This means an additive model for the log of the cost. In Figure 15 we plot the (base 2) logarithm of cost against the (base 2) logarithm of η_l for five selected powers. Each point is one test – the one with lowest cost among those with power very close to a selected value. The fact that the curves are roughly translations of each other is consistent with the additive model for the log-cost. The roughly linear dependence of the log cost with respect to $\log \Gamma(|A|)$ suggests a power dependence as a first approximation ($\Psi(x) \propto x^\alpha$ for some $\alpha \in [0, 1]$).

10.4 Detection results

We use the framework of §6 – power-based cost for a fixed hierarchy. More specifically, from all the "best tests" created, we extracted one for each cell $A \in \mathcal{A}$ such that all the powers and costs are (approximately) the same at each level, which yields one sequence $(\beta_l, c_l), l = 1, ..., 6$ which is increasing in both components and plotted in Figure 16 (left). Since the powers are increasing, the conditions of Corollary 2 are satisfied under the cost model. However, we needn't assume that the cost model is valid; we can directly check whether (β_l, c_l) satisfies the hypotheses of Corollary 1. In Figure 16 (right) we show, level by level, the (logarithms of the) values representing the two sides of (18). Clearly the conditions of Corollary 1 are easily satisfied.

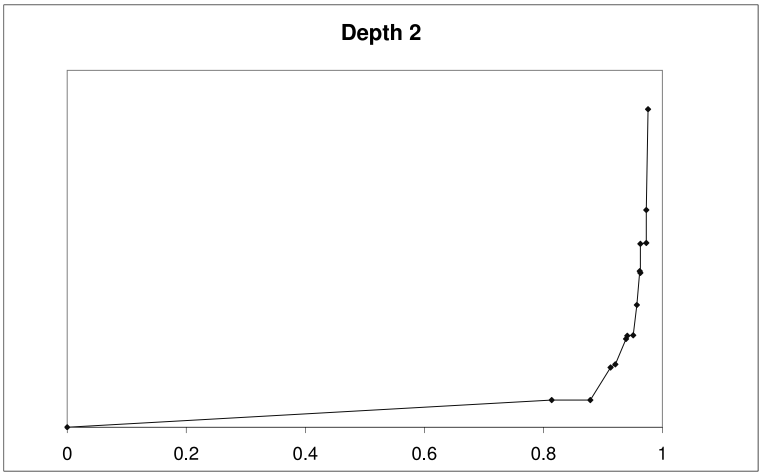


Figure 14: Best tests for a depth two cell.

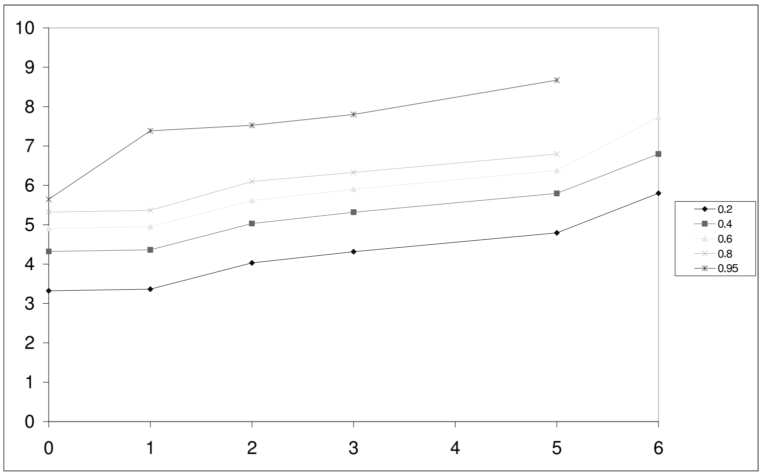
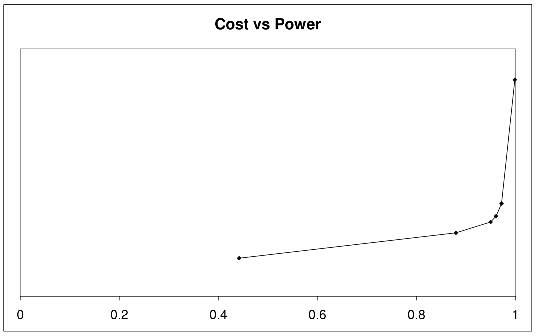


Figure 15: Log cost vs. log invariance for various powers.



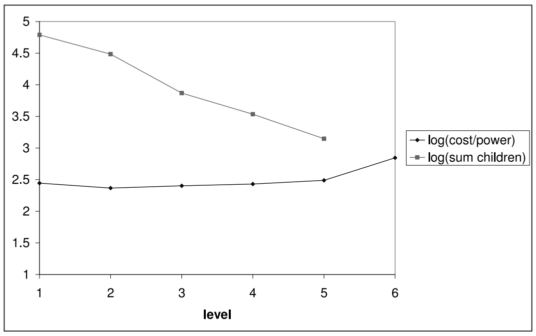


Figure 16: Left: The pairs (β_l, c_l) for the fixed hierarchy used in the experiments. Right, top curve: $l \to \log(C_l \times (c_{l+1}/\beta_{l+1}))$ where C_l is the the number of children of a node at level l. Bottom curve: $l \to \log(c_l/\beta_l)$. The conditions of Corollary 1 are clearly satisfied.

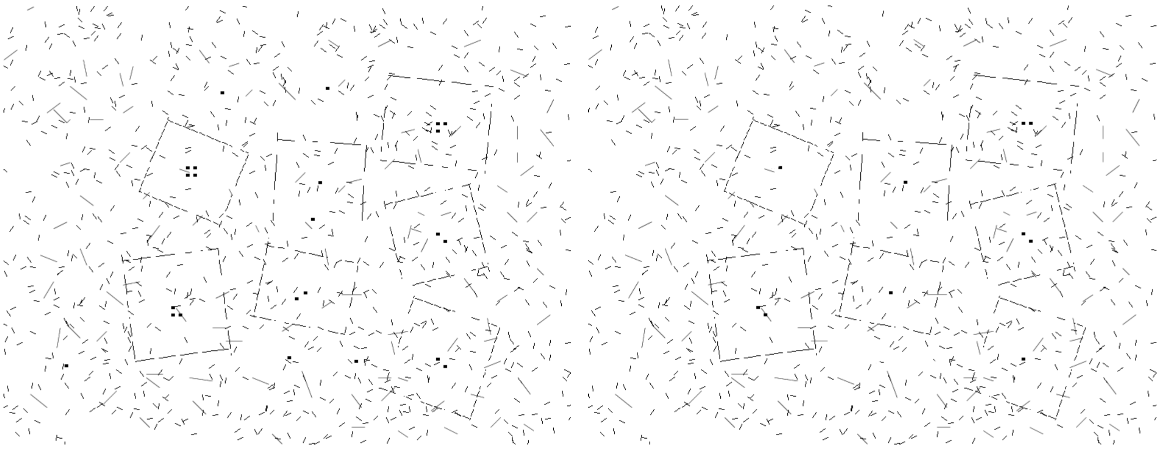


Figure 17: Example of a detection result; the small squares indicate the detected locations. Left: CTF detection only. Notice there are scattered false positives. Right: CTF search followed by template-matching. Nearly all the false positives are removed with virtually no increase in computation.

The detection results for one scene is shown in Figure 17. In order to estimate total computation, we processed a 858×626 scene 100 times. The average time is $3.25s$ on a Pentium 1.5GHz. For comparison, we can perform ideal hypothesis test for each fine cell ($Y \in A_{6,k}, k = 1, ..., 64$) based on simply counting *all* the edges in the region generated by the union of silhouettes over the poses in $A_{6,k}$ (a form of template-matching) and setting a threshold to obtain no false negatives. (This is a more discriminating test than X_A for a fine cell A because the latter uses only *some* of the edges.) The average processing time for this brute force approach is far larger ($2338s$) but the results are virtually perfect. Finally, we can perform a two-stage analysis, first executing the CTF search and then doing the template-matching only at the detected poses. The processing time is virtually the same as for the CTF search (about 3s) but most of the false positives removed; see Figure 17.

11 Discussion and Conclusion

There are many problems in machine learning and perception which come down to differentiating among an enormous number of competing explanations, some very similar to each other and far too many to examine one-by-one. In these cases, efficient representations may be as important as statistical learning (Geman et al. 1992), and thinking about computation at the start of the day may be essential. It then seems prudent to model the computational process itself and hierarchical designs are a natural way to do this. Moreover, there is plenty of evidence that this works in practice. On the mathematical side, the questions that naturally arise from thinking about CTF representations and CTF search are of interest in themselves. We have provided one possible formulation; others could be envisioned.

Within our formulation there are some unanswered but fundamental mathematical questions and a few dubious assumptions. To begin with, we have divided the whole classification problem into two distinct and successive phases, first non-contextual (testing against non-specific alternatives) and second contextual (testing one subset of explanations against another). We have shown that CTF search is effective, even optimal, in the first phase and preliminary results (not reported here) indicate the same is true of the second phase. However, whereas sensible, this division was artificially imposed; in particular, we have not shown that it emerges naturally from a global formulation of the problem. One might, for example, expand the family $\mathcal{X} = \{X_A, A \in \mathcal{A}\}$ into a much larger family of hypothesis tests for testing $Y \in A$ vs. $Y \in B$ for various subsets A, B and levels of error, and then attempt to *prove* that it is in fact computationally efficient to start with $B = A^c$ under some distributional assumptions, and reasonable tradeoffs among scope, error and cost.

Whereas our results on fixed-cost hierarchies are fairly comprehensive, the results on extended hierarchies are evidently not. What is special, if anything, about the "harmonic cost function"? Simulations suggest that the CTF is generically optimal but we have not been able to prove this in general.

On the other hand, several of our model assumptions can be considered as too simplistic. Perhaps the cost model should be revisited; in simulations high power is not always attainable at high invariance (regardless of cost), at least for relatively simple tests (recall Figure 13). As pointed out earlier, supposing conditional independence under P_0 is disputable. Ideally, one should examine non-trivial dependency structures for $\mathcal{X}$, one appealing model being a first-order Markov structure of the tests as already depicted in the simulations of section 6.4. Also, measuring computation under P_0 only is suspect. At some point in the computational process, as evidence accumulates from positive test results for the presence of a pattern of interest, the background hypothesis ceases to be dominant and all the class-conditional distributions must enter the story.

More ambitiously, an even more general optimization problem could be considered: Design the entire system including the subsets to be tested (not requiring a hierarchical structure *a priori*) as well as the levels of discrimination. This would likely involve a dependency structure for overlapping tests. Some of these questions are currently being investigated.

Appendix

A Proofs for Section 7

Proof of Theorem 4. Consider a given tree-structured hierarchy $\mathcal{A}$. In this proof, we are mainly interested in the graph structure of $\mathcal{A}$. Here again it will be easier to consider the equivalent "augmented" model $\overline{\mathcal{A}}$ (see section 5.5.4), thereby assuming the original $\mathcal{A}$ has been extended one level by adding a single child to each original leaf (in order to accommodate the perfect tests $X_{\{y\},1}$ which are performed at the end of the search for all $y \in \widehat{Y}$). Except for the power-one constraint for the final, singleton tests: For any node s in a strategy tree, the assigned power $\beta(s)$ may be freely chosen independently of how it is chosen when the corresponding attribute $A(s)$ appears at other nodes. Of course there must be no errors under P_0, but this is automatically satisfied by definition for any CTF strategy.

To prove the theorem we will proceed by recurrence over the size of subhierarchies of $\overline{\mathcal{A}}$. We actually need slightly more general objects than conventional subhierarchies (i.e., subtrees). We will call $\mathcal{H}$ a *generalized subhierarchy* if $\mathcal{H}$ is a finite union of subhierarchies of $\overline{\mathcal{A}}$. The cardinality of $\mathcal{H}$ is defined as the number of its nodes (internal or leaves). A CTF strategy for $\mathcal{H}$ satisfies the usual hypothesis that an attribute is tested if and only if all of its ancestors in $\mathcal{H}$ have been tested and returned a positive answer. Finally, for a node B of $\overline{\mathcal{A}}$, denote by $\mathcal{H}_B$ the generalized subhierarchy composed of all strict descendents of B, in other words the union of all the subhierarchies rooted in direct children of B.

Now we prove by recurrence on the size c of generalized subhierarchies which have the following property:

($\mathbf{P}$(c)) *For any generalized subhierarchy $\mathcal{H}$ of $\overline{\mathcal{A}}$ of cardinality at most c, every CTF strategy with optimal choice of powers has the same cost $\mathcal{C}_{ctf}(\mathcal{H})$. Furthermore, for any node $B \in \mathcal{H}$, the test X_B is always performed in such a CTF strategy with the same power β_B, and this value depends only on $\mathcal{H}_B$, being therefore independent of the CTF strategy considered. Finally, if $\mathcal{H}$ is the union of several disjoint subhierarchies of $\overline{\mathcal{A}}$, then the CTF cost of $\mathcal{H}$ is the sum of the CTF costs of these subhierarchies.*

For $c = 1$, any generalized subhierarchy $\mathcal{H}$ must be a single node (attribute) corresponding to a perfect test, in which case the property is trivial.

Suppose $\mathbf{P}(c)$ is true and consider a generalized subhierarchy $\mathcal{H}$ of cardinality $c+1$. Let T be a CTF strategy for $\mathcal{H}$ with optimally chosen powers and let B be the attribute which is tested at the root of T; necessarily B has no ancestors in $\mathcal{H}$. Write $\overline{\mathcal{H}}_B$ for the generalized subhierarchy $\mathcal{H} \setminus (\{B\} \cup \mathcal{H}_B)$.

If B is a leaf, then, by construction, its power is fixed to 1 and $\mathcal{H}_B = \emptyset$. Hence, after B is tested with power 1 (thus returning a null answer under P_0), the remaining part of T is a CTF strategy for subhierarchy $\overline{\mathcal{H}}_B$, and therefore, by the hypothesis of recurrence,

$$E_0 C(T) = \Psi(1) + \mathcal{C}_{ctf}(\overline{\mathcal{H}}_B). \tag{37}$$

Suppose now that B is not a leaf. If the test $X_B = 0$, the subsequent part of strategy T must be a CTF exploration, with optimal powers, of the subhierarchy $\overline{\mathcal{H}}_B$. Similarly, if $X_B = 1$, the subsequent part of T is a CTF strategy for $\overline{\mathcal{H}}_B \cup \mathcal{H}_B$, a disjoint union. By Lemma 3 and the recurrence hypothesis concerning cost additivity over disjoint subhierarchies, we therefore have

$$\begin{aligned} E_0 C(T) &= \mathcal{C}_{ctf}(\overline{\mathcal{H}}_B) + \Phi_{\Gamma(|B|)}(\mathcal{C}_{ctf}(\overline{\mathcal{H}}_B \dot{\cup} \mathcal{H}_B) - \mathcal{C}_{ctf}(\overline{\mathcal{H}}_B)) \\ &= \mathcal{C}_{ctf}(\overline{\mathcal{H}}_B) + \Phi_{\Gamma(|B|)}(\mathcal{C}_{ctf}(\mathcal{H}_B)). \end{aligned} \tag{38}$$

Furthermore, the second part of Lemma 3 shows that the optimal power chosen for X_B only depends on $\mathcal{C}_{ctf}(\mathcal{H}_B)$.

Property $\mathbf{P}(c+1)$ now an immediate consequence of (37) and (38), which concludes the proof. $\square$

Proof of Theorem 6. Recall the situation of Figure 6: let A_1 be the coarsest attribute, of complexity $\Gamma(|A_1|) = a$, B is some other attribute, of complexity $\Gamma(|B|) = b$, and we wish to show that doing A_1 before B has lower cost. Let T_L (respectively, T_R) be the tree on the lefthand side (resp. righthand side) of Figure 6. Recall that x (respectively, y) is the mean cost of the tree encountered when $X_{A_1} = 1, X_B = 0$ (resp., $X_{A_1} = 1, X_B = 1$). We can assume $y \geq x$ (otherwise X_B has optimal power 0 and the result is trivial).

From Lemma 3, we know that the mean cost of T_L with optimal choices of powers for X_{A_1} and X_B is given by

$$C_L(y;x) = \Phi_a(x) + \Phi_b(\Phi_a(y) - \Phi_a(x)),$$

and the mean cost of T_R with optimal powers is

$$C_R(y;x) = \Phi_a(x + \Phi_b(y - x)).$$

We wish to show that, for any a, b:

$$C_L(y;x) \geq C_R(y;x), \ \ y \geq x.$$

This is obviously satisfied when $x = y$ (for any choice of a and b). We will show that

$$\frac{\partial C_L(y;x)}{\partial y} \geq \frac{\partial C_R(y;x)}{\partial y}, \ \ y \geq x. \tag{39}$$

Taking derivatives, (39) becomes

$$\Phi_b'(\Phi_a(y) - \Phi_a(x))\Phi_a'(y) \geq \Phi_a'(x + \Phi_b(y - x))\Phi_b'(y - x). \tag{40}$$

Now,

$$\Phi_b'(x) = \left(\frac{b}{x+b}\right)^2$$

and, since all the quantities involved are positive, it is therefore equivalent to take square roots in (40). Since

$$\Phi_a(y) - \Phi_a(x) = \frac{ay}{a+y} - \frac{ax}{a+x} = \frac{a^2(y-x)}{(x+a)(y+a)},$$

the square root of the lefthand side of (40) is given by

$$\frac{ab(x+a)}{a^2(y-x) + b(x+a)(y+a)}.$$

Similarly, the square root of the righthand side of (40) is

$$\frac{ab}{(a+b+x)(y-x)+b(x+a)}.$$

so that (40) is equivalent to

$$\frac{x+a}{a^2(y-x)+b(y+a)(x+a)} \geq \frac{1}{(a+b+x)(y-x)+b(x+a)}$$

which, after some algebra, is in turn equivalent to

$$(y-x)[(x+a)^2-a^2] \geq 0$$

which is true since $y \geq x$. This concludes the proof. $\square$

B Proofs for Section 9

For the proof of Theorem 8, we will first establish the following lemma:

Lemma 7. *Assume that hypothesis* **(H1)** *is satisfied and that for some constant $R_0 \leq 1$:*

$$\log\left(\frac{N_2}{N_1}\right) \geq \frac{1}{1-p_1}\left(\max_{1\leq i\leq L-1} p_i \log(N_{i+1}) - \frac{p_1\log(p_1)}{1-p_1} + p_1(1+\log(2)-\log(R_0))\right).$$

Then if $R \geq R_0$, the first attribute tested in the optimal strategy must belong to the coarsest level.

Proof. First observe that the assumed condition implies that $p_1 < 1/2$ since it implies that

$$\log(N_2) \geq \frac{1}{1-p_1}\left(p_1\log(N_2) - \frac{p_1\log(p_1)}{1-p_1} + p_1(1+\log(2))\right),$$

and therefore that

$$(1-2p_1)\log(N_2) \geq -\frac{p_1\log(p_1)}{1-p_1} + p_1(1+\log(2)).$$

It is easy to see that the righthand side of the last inequality is increasing in p_1 and strictly positive for $1/2 \leq p_1 \leq 1$, whereas the lefthand side is negative on this interval, which implies $p_1 < 1/2$.

Next, we calculate the cost of the strategy T which performs the tests in the order $i_1, i_2, ..., i_L$. Each test at level i_1 has probability $1/N_{i_1}$ of being performed; then each test at level i_2 has probability p_{i_1}/N_{i_2} of being performed (since it is performed only if it is chosen and the answer to the first test is positive); and so forth. Consequently,

$$\begin{aligned}E_0 C_{test}(T) &= H(\mathbf{q}(T), R) = \log(N_{i_1}) + p_{i_1}\log(N_{i_2}/p_{i_1}) + \ldots \\ &\quad + p_{i_1}\ldots p_{i_{L-1}}\log(N_{i_L}/p_{i_1}\ldots p_{i_{L-1}}) \\ &\quad + |\mathbf{q}(T)|\log|\mathbf{q}(T)| - |\mathbf{q}(T)|\log R,\end{aligned}$$

where

$$|\mathbf{q}(T)| = Q(T) = 1 + p_{i_1} + p_{i_1}p_{i_2} + \ldots + p_{i_1}\ldots p_{i_{L-1}}. \tag{41}$$

Let T_{ctf} be the CTF strategy, which performs the tests in the order $1, 2, \ldots, L$. Suppose $i_1 \neq 1$. Then we want to prove that, under the hypotheses made, $H(\mathbf{q}(T), R) \geq H(\mathbf{q}(T_{ctf}), R)$.

By dropping the term

$$p_{i_1}\log(N_{i_2}/p_{i_1}) + \ldots + p_{i_1}\ldots p_{i_{L-1}}\log(N_{i_L}/p_{i_1}\ldots p_{i_{L-1}})$$

in the expression for $H(\mathbf{q}(T), R)$ we have

$$\begin{aligned}H(\mathbf{q}(T), R) - H(\mathbf{q}(T_{ctf}), R) &\geq \log\left(\frac{N_{i_1}}{N_1}\right) - p_1\log\left(\frac{N_2}{p_1}\right) - \ldots \\ &\quad - p_1\ldots p_{L-1}\log\left(\frac{N_L}{p_1\ldots p_{L-1}}\right) + Q(T)\log(Q(T)) \\ &\quad - Q(T_{ctf})\log(Q(T_{ctf})) - (Q(T) - Q(T_{ctf}))\log R.\end{aligned}$$

We can bound the various terms as follows: Since (N_l) is nondecreasing,

$$\log\left(\frac{N_{i_1}}{N_1}\right) \geq \log\left(\frac{N_2}{N_1}\right). \tag{42}$$

Putting $\gamma = \max_{1\leq i\leq L-1} p_i\log(N_{i+1})$, and using the fact that the p_i's are decreasing, we have:

$$\begin{aligned}p_1\log(N_2) + p_1p_2\log(N_3) + \ldots + p_1\ldots p_{L-1}\log(N_L) &= \sum_{i=2}^{L} p_1\ldots p_{i-2}(p_{i-1}\log(N_i)) \\ &\leq \sum_{j\geq 0} p_1^j\gamma \\ &= \frac{1}{1-p_1}\gamma.\end{aligned} \tag{43}$$

Using $p_l \searrow$ again, together with $p_1 \leq 1/2$ and the fact that $-x\log(x)$ is increasing for $x \leq 1/e$ we obtain:

$$\begin{aligned}
-p_1\log(p_1) - \ldots - p_1 \ldots p_{L-1}\log(p_1 \ldots p_{L-1}) &\leq -\sum_{i=1}^{L-1} p_1^i \log(p_1^i) \\
&\leq -\log(p_1)\sum_{i\geq 1} i p_1^i \\
&= -\log(p_1)\frac{p_1}{(1-p_1)^2}. \qquad (44)
\end{aligned}$$

Now, from (41), since $p_1 \leq 1/2$, for any strategy T, $Q(T) \in [1,2]$. Since the function $x\log(x)$ is Lipschitz in $[1,2]$ with constant $(1+\log(2))$, we have

$$|Q(T)\log(Q(T)) - Q(T_{ctf})\log(Q(T_{ctf}))| \leq (1+\log(2)|)|Q(T) - Q(T_{ctf})|.$$

Moreover,

$$Q(T_{ctf}) - Q(T) \leq p_1 + \ldots + p_1 \ldots p_{L-1} \leq \sum_{i\geq 1} p_1^i = \frac{p_1}{1-p_1}.$$

Therefore

$$Q(T)\log(Q(T)) - Q(T_{ctf})\log(Q(T_{ctf})) \geq -\frac{(1+\log(2))p_1}{1-p_1} \qquad (45)$$

and

$$-(Q(T) - Q(T_{ctf}))\log(R) \geq \left(\frac{p_1}{1-p_1}\right)\log(R). \qquad (46)$$

Finally, putting inequalities (42)-(46) into the expression for the cost difference, we obtain

$$\begin{aligned}
H(\mathbf{q}(T),R) - H(\mathbf{q}(T_{ctf}),R) &\geq \log\left(\frac{N_2}{N_1}\right) \\
&\qquad - \frac{1}{1-p_1}\left(\gamma - \frac{p_1\log(p_1)}{1-p_1} + p_1(1+\log(2)-\log(R))\right) \\
&\geq 0,
\end{aligned}$$

by the assumptions of the lemma (since $R_0 \leq R$). $\square$

Proof of Theorem 8. We proceed by recurrence. The case $L = 1$ is obvious. Suppose the theorem is proved for any $L < L_0$ with $L_0 \geq 2$. Now consider the case $L = L_0$. Suppose T is an optimal strategy. Using the hypotheses of Theorem 8 and Lemma 7, we conclude that the first attribute tested in T must belong to the coarsest level. If the response of this test is positive, by independence of the levels, the remaining part of strategy T should be optimal for the problem with $L_0 - 1$ levels of sizes $N_2, \ldots, N_{L_0}$ and remaining available resource

$$R' = R - \sum_{A \in \widetilde{\mathcal{A}}_1} r(A) = R - N_1 \left(\frac{R}{N_1 Q(T)} \right) = R \left(1 - \frac{1}{Q(T)} \right).$$

(Recall that $r(A) = Rq_A(T)/Q(T)$ and $q_A(T) = 1/N_1$ for $A \in \widetilde{\mathcal{A}}_1$.) Now since the first attribute tested belongs to $\widetilde{\mathcal{A}}_1$, necessarily $Q(T) \geq 1 + p_1$ so that $R' \geq Rp_1/(p_1 + 1)$, and

$$-\log(R') \leq -\log(R) - \log(p_1) + p_1.$$

As a consequence, we can apply the hypothesis of recurrence (with resource R'), since the hypothesis of the theorem carries over thanks to the above computation. $\square$

In preparation for the proof of Theorem 9 we need:

Lemma 8. *Suppose* **(H1)** *and the following conditions are satisfied:*

$$(i) \log\left(4\frac{N_2}{N_1}\right) \geq \frac{2}{1-2p_1}\left(\max_{1 \leq l \leq L-1} p_l \log(N_{l+1}) + \frac{-\log(2p_1)p_1}{1-2p_1} - \frac{1}{2}\log(1-2p_1)\right)$$

$(ii) p_1 \leq 1/(2\sqrt{e})$

Then the first test performed by the optimal strategy must belong to the coarsest level.

Proof of Lemma 8. First write down the cost of the CTF strategy T_{ctf}. Let $A_l \in \widetilde{\mathcal{A}}_l$ be some attribute. The probability that its associated test X_{A_l} is performed is

$$P(X_{A_l} \text{ performed}) = P(X_{A_l} \text{ performed} | A_l \in \mathcal{D}) P(A_l \in \mathcal{D}),$$

where "$A_l \in \mathcal{D}$" denotes the event that A_l has been attached to some node of the tree $\mathcal{D}$:

$$\{A_l \in \mathcal{D}\} = \cup_{s \in \mathcal{D}} \{A(s) = A\}.$$

Now $P(A_l \in \mathcal{D}) = 2^{l-1}/N_l$, and a test in the hierarchy is performed by the CTF strategy if and only if all of its ancestors in the hierarchy are positive. Therefore, since the random variables $(X_s)_{s \in \mathcal{D}}$ are independent,

$$P(X_{A_l} \text{ performed by } T_{ctf} | A_l \in \mathcal{D}) = p_1 \ldots p_{l-1},$$

so that finally

$$H(\mathbf{q}(T_{ctf}), R) = \log(N_1) + 2p_1 \log\left(\frac{N_2}{2p_1}\right) + \ldots + 2^{L-1} p_1 \ldots p_{L-1} \log\left(\frac{N_L}{2^{L-1} p_1 \ldots p_{L-1}}\right)$$
$$+ Q(T_{ctf}) \log Q(T_{ctf}) - Q(T_{ctf}) \log R,$$

where

$$Q(T_{ctf}) = 1 + 2p_1 + \ldots + 2^{L-1}p_1 \ldots p_{L-1}.$$

Now suppose we are given a strategy T, with first test $X_{A_i} \in \mathcal{A}_i$ and $i \neq 1$. So we have

$$\begin{aligned} H(\mathbf{q}(T), R) - H(\mathbf{q}(T_{ctf}), R) \quad \geq \quad & \log\left(\frac{N_i}{N_1}\right) - 2p_1 \log\left(\frac{N_2}{2p_1}\right) - \ldots \\ & -2^{L-1}p_1 \ldots p_{L-1} \log\left(\frac{N_L}{2^{L-1}p_1 \ldots p_{L-1}}\right) + Q(T)\log(Q(T)) \\ & -Q(T_{ctf})\log(Q(T_{ctf})) - (Q(T) - Q(T_{ctf}))\log R. \qquad (47) \end{aligned}$$

Now using the hypotheses we bound the different terms. Since the N_i's are increasing we have

$$\log\left(\frac{N_i}{N_1}\right) \geq \log\left(\frac{N_2}{N_1}\right). \qquad (48)$$

Denoting $\gamma = \max_{1\leq i\leq L-1} p_i \log(N_{i+1})$ and using the fact that the p_i's are decreasing we have:

$$\begin{aligned} 2p_1\log(N_2) + 4p_1p_2\log(N_3) + \ldots + 2^{L-1}p_1 \ldots p_{L-1}\log(N_L) =& \sum_{i=2}^{L} 2^{i-1}p_1 \ldots p_{i-2}(p_{i-1}\log(N_i)) \\ \leq& 2\sum_{j\geq 0}(2p_1)^j\gamma \\ =& \frac{2}{1-2p_1}\gamma. \qquad (49) \end{aligned}$$

Using the fact that the p_i's are decreasing, that $p_1 \leq 1/(2\sqrt{e})$ and that the function $x\log(x)$ is decreasing for $x \leq 1/e$ we have:

$$\begin{aligned} -2p_1\log(2p_1) - \ldots - 2^{L-1}p_1 \ldots p_{L-1}\log - \left(2^{L-1}p_1 \ldots p_{L-1}\right) =& -\sum_{i=1}^{L-1}(2p_1)^i \log\left(2^i p_1^i\right) \\ \leq& -\log(2p_1)\sum_{i\geq 1} i2^i p_1^i \\ =& -\log(2p_1)\frac{2p_1}{(1-2p_1)^2}. \end{aligned} \qquad (50)$$

Now, $1 \leq Q(T_{ctf}) \leq \sum_{i\geq 0}(2p_1)^i \leq 1/(1-2p_1)$, and on the other hand $Q(T) \geq 2$ (since any strategy that does not begin with the coarsest test must perform at least 2 tests in any case), so $Q(T) > Q(T_{ctf})$ and

$$-Q(T)\log(Q(T)) + Q(T_{ctf})\log(Q(T_{ctf})) + (Q(T) - Q(T_{ctf}))\log R \leq \frac{-\log(1-2p_1)}{1-2p_1} - 2\log(2). \qquad (51)$$

Finally, putting together inequalities (48)-(51) into (47), we obtain

$$H(\mathbf{q}(T), R) - H(\mathbf{q}(T_{ctf}), R) \geq \log\left(\frac{N_2}{N_1}\right) + \log(4)$$
$$- \frac{2}{1-2p_1}\left(\gamma + \frac{-\log(2p_1)p_1}{1-2p_1} - \frac{1}{2}\log(1-2p_1)\right) \geq 0,$$

using hypothesis (i). □

Proof of Theorem 9. As usual, we proceed by recurrence on the depth L of the hierarchy $\mathcal{D}$. The case $L = 1$ is obvious. Suppose the theorem is true for any $L < L_0$ with $L_0 \geq 2$ and consider the case $L = L_0$.

Suppose T is an optimal strategy. By Lemma 8, we conclude that the first test to be performed at each round is for the attribute attached to the root s_0 of $\mathcal{D}$, X_{s_0}.

Let $\mathcal{D}_1, \mathcal{D}_2$ be the two subtrees rooted at the children of s_0 which are of depth at most $L_0 - 1$, so that, if a tree is identified with the set of its nodes, we have $\mathcal{D} = \{s_0\} \dot\cup \mathcal{D}_1 \dot\cup \mathcal{D}_2$. For an attribute A, and with some abuse of notation, write $A \in \mathcal{D}_1$ for the event "attribute A is attached to one of the nodes of $\mathcal{D}_1$".

For $A \in \widetilde{\mathcal{A}}_2 \cup \ldots \cup \widetilde{\mathcal{A}}_L$, define the quantity

$$q_A^{(1)} = P(X_A \text{ is performed by strategy } T, A \in \mathcal{D}_1),$$

and define $q_A^{(2)}$ in the same way, replacing $\mathcal{D}_1$ by $\mathcal{D}_2$; finally, define the family $\mathbf{q}^{(0)}$ by $q_A^{(0)} = q_A(T)$ if $A \in \mathcal{A}_1$ and $q_A^{(0)} = 0$ otherwise. Thus, $\mathbf{q}(T) = \mathbf{q}^{(0)} + \mathbf{q}^{(1)} + \mathbf{q}^{(2)}$ (where, as usual, bold letters indicate collections of variables indexed by $\mathcal{D}$).

By concavity of H, for any $R' \leq 1$,

$$H\left(\frac{1}{2}(\mathbf{q}^{(1)} + \mathbf{q}^{(2)}), R'\right) \geq \frac{1}{2}\left(H(\mathbf{q}^{(1)}, R') + H(\mathbf{q}^{(2)}, R')\right).$$

Multiplying by 2 and using the fact that H is homogeneous in its first argument we conclude that

$$H(\mathbf{q}^{(1)} + \mathbf{q}^{(2)}, R') \geq H(\mathbf{q}^{(1)}, R') + H(\mathbf{q}^{(2)}, R'). \quad (52)$$

Consider now the conditional strategy $S = T_{\mathcal{D}_1}(X_{\mathcal{D}_2})$, where $X_{\mathcal{D}_2}$ is the set of tests corresponding to the attributes attached to subhierarchy $\mathcal{D}_2$. Let $q_A(S; X_{\mathcal{D}_2})$ be the probability of performing test X_A using strategy S. By definition we have

$$\mathbf{q}^{(1)} = (1 - \beta_1)E_0[\mathbf{q}(S; X_{\mathcal{D}_2})].$$

Now note that, since all attribute tests at the same resolution level ℓ are assumed to have the same power β_ℓ and are independent, by a symmetry argument, conditional on $X_{\mathcal{D}_2}$ (the response values of the attribute tests attached to hierarchy $\mathcal{D}_2$ – but without the information of which tests are attached to $\mathcal{D}_2$), the probability distribution of the variables $(X_{\mathcal{D}_1})$ and of the events $(A \in \mathcal{D}_1)$ remains the same as without conditioning.

Therefore, conditional on $X_{\mathcal{D}_2}$, we can apply the hypothesis of recurrence to subhierarchy $\mathcal{D}_1$ if conditions (i) and (ii) are satisfied. Obviously, (i) is still valid; and hypothesis (ii) carries over because the sequence (p_l) is decreasing (hypothesis **(H1)**).

We therefore conclude that, for any $R' \leq 1$,

$$H(\mathbf{q}(S; X_{\mathcal{D}_2}), R') \geq H(\mathbf{q}(T_{ctf}^{(1)}), R'),$$

where $\mathbf{q}(T_{ctf}^{(1)})$ denotes the vector of probabilities (of performing tests indexed by $\mathcal{A}$) for the CTF strategy applied to subhierarchy $\mathcal{D}_1$. By concavity of the cost function,

$$\begin{aligned} H(\mathbf{q}^{(1)}, R') &= H\left((1-\beta_1)E_0[\mathbf{q}(S; X_{\mathcal{D}_2})], R'\right) \\ &\geq E_0\left[H((1-\beta_1)\mathbf{q}(S; X_{\mathcal{D}_2}), R')\right] \\ &\geq H((1-\beta_1)\mathbf{q}(T_{ctf}^{(1)}), R'). \end{aligned}$$

Applying the same reasoning to subhierarchy $\mathcal{D}_2$ and using (52),

$$H(\mathbf{q}^{(1)} + \mathbf{q}^{(2)}, R') \geq H((1-\beta_1)\mathbf{q}(T_{ctf}^{(1)}), R') + H((1-\beta_1)\mathbf{q}(T_{ctf}^{(2)}), R')$$

(where $T_{ctf}^{(2)}$ denotes the CTF strategy applied to hierarchy $\mathcal{D}_2$).

Now, due to the randomized construction of the hierarchies, and to the symmetry between $\mathcal{D}_1$ and $\mathcal{D}_2$, we have $\mathbf{q}(T_{ctf}^{(1)}) = \mathbf{q}(T_{ctf}^{(2)})$. Using again the homogeneity of the cost function, we then have

$$H(\mathbf{q}^{(1)} + \mathbf{q}^{(2)}, R') \geq H((1-\beta_1)(\mathbf{q}(T_{ctf}^{(1)}) + \mathbf{q}(T_{ctf}^{(2)})), R').$$

Finally, let R_0 be the total amount of resources distributed among the tests for attributes in $\mathcal{A}_1$ and let $R' = R - R_0$. Then

$$\begin{aligned} E_0 C_{test}(T) = H(\mathbf{q}(T), R) &= H(\mathbf{q}^{(0)} + \mathbf{q}^{(1)} + \ldots + \mathbf{q}^{(k)}, R_0 + R') \\ &= H(\mathbf{q}^{(0)}, R_0) + H(\mathbf{q}^{(1)} + \mathbf{q}^{(2)}, R') \\ &\geq H(\mathbf{q}^{(0)}, R_0) + H((1-\beta_1)(\mathbf{q}(T_{ctf}^{(1)}) + \mathbf{q}(T_{ctf}^{(2)})), R') \\ &\geq H(\mathbf{q}^{(0)} + (1-\beta_1)(\mathbf{q}(T_{ctf}^{(1)}) + \mathbf{q}(T_{ctf}^{(2)})), R) \end{aligned}$$

where the third equality and the last inequality hold by Proposition 2-(iii). But $\mathbf{q}^{(0)} + (1 - \beta_1)(\mathbf{q}(T_{ctf}^{(1)}) + \mathbf{q}(T_{ctf}^{(2)}))$ is precisely the probability vector corresponding to the CTF strategy, which means this strategy is optimal. □

References

Amit, Y. (2002), *2D Object Detection and Recognition*, M.I.T. Press.

Amit, Y. & Geman, D. (1999), 'A computational model for visual selection', *Neural Computation* **11**, 1691–1715.

Amit, Y. & Geman, D. (2003), Coarse-to-fine templates and bayesian scene analysis, Technical report, University of Chicago.

Bellman, R. (1961), *Adaptive Conrol Process: A Guided Tour*, Princeton University Press.

Blackwell, D. & Girschick, M. A. (1954), *Theory of Games and Statistical Decisions*, John Wiley.

Breiman, L., Friedman, J., Olshen, R. & Stone, C. (1984), *Classification And Regression Trees*, Wadsworth, Statistics/probability series.

Chernoff, H. (1972), *Sequential Analysis and Optimal Design*, SIAM.

Cootes, T. F. & Taylor, C. J. (1996), Locating faces using statistical feature detectors, *in* 'Proceedings, Second International Conference on Automatic Face and Gesture Recognition', IEEE Computer Society Press, pp. 204–209.

DeGroot, M. H. (1970), *Optimal Statistical Decisions*, McGraw-Hill, New York.

Desimone, R., Miller, E. K., Chelazzi, L. & Lueschow, A. (1995), Multiple memory systems in visual cortex, *in* M. S. Gazzaniga, ed., 'The Cognitive Neurosciences', MIT Press, Cambridge, Massachusetts, pp. 475–486.

Dietterich, T. (2000), The divide-and-conquer manifesto, *in* 'Proc. Eleventh Int'l Conf. Algorithmic Learning Theory', Springer-Verlag, pp. 13–16.

Fleuret, F. (2000), Détection hiérarchique de visages par apprentissage statistique, PhD thesis, University of Paris VI, Jussieu, France.

Fleuret, F. & Geman, D. (2001), 'Coarse-to-fine face detection', *Inter. J. Comp. Vision* **41**, 85–107.

Frisch, J. & Finke, M. (1998), Applying divide and conquer to large scale pattern recognition tasks, *in* G. Orr & K.-R. Müller, eds, 'Neural Networks: Tricks of the Trade', Vol. 1524 of *Lecture notes in Computer Science*, Springer, pp. 315–342.

Garey, M. R. (1972), 'Optimal binary identification procedures', *SIAM J. Appl. Math.* **23**, 173–186.

Geman, D. & Jedynak, B. (2001), 'Model-based classification trees', *IEEE Trans. Info. Theory* **47**, 1075–1082.

Geman, S., Bienenstock, E. & Doursat, R. (1992), 'Neural networks and the bias/variance dilemma', *Neural Computation* **4**, 1–58.

Geman, S., Manbeck, K. & McClure, E. (1995), Coarse-to-fine search and rank-sum statistics in object recognitio, Technical report, Brown University.

Jung, F. (2001), Reconnaissance d'objets par focalisation et detection de changements, PhD thesis, Ecole Polytechnique, Paris, France.

Jung, F. (2002), Detecting new buildings from time-varying aerial stereo pairs, Technical report, IGN.

Osuna, E., Freund, R. & Girosi, F. (1997), Training support vector machines: an application to face detection, *in* 'Proceedings, CVPR', IEEE Computer Society Press, pp. 130–136.

Rowley, H. A., Baluja, S. & Kanade, T. (1998), 'Neural network-based face detection', *IEEE Trans. PAMI* **20**, 23–38.

Socolinsky, D. A., Neuheisel, J. D., Priebe, C. E., De Vinney, J. & Marchette, D. (2002), Fast face detection with a boosted cccd classifier, Technical report, Johns Hopkins University.

Sung, K. K. & Poggio, T. (1998), 'Example-based learning for view-based face detection', *IEEE Trans. PAMI* **20**, 39–51.

Thorpe, S., Fize, D. & Marlot, C. (1996), 'Speed of processing in the human visual system', *Nature* **381**, 520–522.

Trouvé, A. & Yu, Y. (2002), Entropy reduction strategies on tree structured retrieval spaces, *in* 'Proc. Colloquium on Mathematics and Computer Science: Algorithms, Trees, Combinatorics and Probabilities', Birkhauser, pp. 513–525.

Viola, P. & Jones, M. J. (2001), Robust real-time face detection, *in* 'Proc. ICCV01', p. II: 747.

Yuille, A. L., Cohen, D. S. & Hallinan, P. (1992), 'Feature extraction from faces using deformable templates', *Inter. J. Comp. Vision* **8**, 104–109.