The Very Particular Structure of the Very Hard Instances

Dan R. Vlasie

I3S Laboratory, University of Nice
Bât ESSI, Route des Colles, BP 145
06903 Sophia Antipolis, France
vlasier@essi.fr

Abstract

We show that the algorithms which behave well on average may have difficulty only for highly structured, non-random inputs, except in a finite number of cases. The formal framework is provided by the theory of Kolmogorov complexity. An experimental verification is done for graph 3-colorability with Brélaz's algorithm.

Introduction

Let P be a problem and A an algorithm solving it, given that the instances are distributed according to some known input distribution. An important question for AI researchers is to *predict* the algorithmic time spent by the algorithm A, *before* launching it on problem instances. Obviously, this prediction would be possible only if there is some decidable, feasible relation between the *syntactical structure* of instances and their *computational complexity*.

Following this idea, much experimental work (Cheeseman, Kanefsky, & Taylor 1991; Mitchell, Selman, & Levesque 1992; Crawford & Auton 1993; Hogg & Williams 1994; Vlasie 1995b) was done in order to identify *critical values* of so called *order parameters* on which the computational complexity may depend. On the theoretical side, the research focused on modeling the way in which the algorithms work in the search space, depending on the input structure (Williams & Hogg 1992; Minton & al. 1992; Musick & Russell 1992; Vlasie 1995a; Goldberg & al. 1982).

In this paper we address the new possibilities offered by the theory of *Kolmogorov complexity*(Li & Vitányi 1993; Calude 1994; Watanabe 1992) to investigate the relation between the *instance structure* and the *algorithmic time*. Since the theory of Kolmogorov complexity deals with the effective descriptional complexity of individual objects, it becomes a natural framework for considering the structure of problem instances and its influence on the solving cost.

Two results are presented. The theoretical one (Claim 7) shows that when a NP-complete problem is easy on average, then the overwhelming majority of hard cases exhibit regularities in their structure. The second result effectively identifies such a regular structure for 3-COL problem and Brélaz's algorithm, with input graphs having a fixed, small connectivity. Namely, the very hard cases occur when the input graphs contain a sufficiently big number of disjoint subgraphs with 4 nodes and 5 edges.

Preliminaries

In this section we recall some definitions and facts that we use in the remainder of the paper. Details may be found in (Li & Vitányi 1990; 1993) and in the cited papers. In the followings, $|x|$ denote the length of the finite binary string x, $d(S)$ denotes the number of elements in the set S and $\log n$ is the logarithm in base 2 of n, rounded upwards.

Definition 1 (Kolmogorov complexity.) *Assume an effective encoding of all Turing machines over the binary alphabet* $\{0,1\}$*:* $M_1, M_2, \ldots$ *The* Kolmogorov complexity *of each string* $x \in \{0,1\}^*$ *is defined as:*

$$K(x) = min\{|M_j| : M_j(\epsilon) = x\}.$$

That is, $K(x)$ is the length of the smallest program that starting "from scratch" outputs x. It can be shown that $K(x)$ does not depend on the particular encoding $M_1, M_2, \ldots$, up to an additive constant. If some supplementary information y about x is supplied, the *conditional* Kolmogorov complexity $K(x|y)$ is defined by $K(x|y) = min\{|M_j| : M_j$ starting on $y \in \{0,1\}^*$ outputs $x\}$.

Definition 2 (Incompressibility.) *Let* c *be an integer constant. The string* $x \in \{0,1\}^n$ *is* c-incompressible *if* $K(x) \geq n - c$.

Incompressibility (say, c-incompressible with small c) offers an elegant way to capture the randomness of finite binary strings: x is random if and only if it is incompressible (Li & Vitányi 1990; 1993; Calude 1994). In other words, in order to describe a random string, one need at least as many bits as the length of the string. Also, if a string is compressible then it is not random because it exhibits regularities allowing a more compact description (that is, requiring a number of bits less than its length). It can be shown that almost all finite strings are incompressible.

Definition 3 (Sparse set.) *Let S be a subset of $\{0,1\}^*$. Let $S^{\leq n}$ denote the set $\{x \in S : |x| \leq n\}$. If the limit of $d(S^{\leq n})/2^n$ goes to zero for n going to infinity, then we call S* sparse.

Fact 4 (Sipser.) *If S is recursive and sparse, then for all constant c there are only finitely many x in S with $K(x) \geq |x| - c$.*

Definition 5 (Universal distribution.) *The distribution* $\mathbf{m}$ *assigning at each x the probability* $\mathbf{m}(x) = 2^{-K(x)}$ *is called the* universal distribution.

Under the universal distribution $\mathbf{m}$, easily describable objects have high probability, and complex or random objects have low probability. Levin has proved that $\mathbf{m}$ dominates multiplicatively each enumerable distribution.

The following theorem is important as it makes a connection between the Kolmogorov complexity and the computational complexity:

Theorem 6 (Li&Vitànyi.) *If the inputs to any complete algorithm are distributed according to* $\mathbf{m}$*, then the average time complexity is of the same order of magnitude as the worst-case time complexity.*

The Hard Instances Are Compressible

Let P be a problem and A any complete algorithm solving it. Theorem 6 shows that the inputs having small Kolmogorov complexity contain the worst cases with high probability. The proof (Li & Vitányi 1992) is based on the fact that $\mathbf{m}$ dominates multiplicatively any other enumerable distribution and then in particular it dominates the distribution assigning high probability to worst cases. In the following we study under which conditions the reciprocal of Theorem 6 holds, i.e. almost all the worst cases admit short descriptions.

In the case of the algorithms for NP-complete problems, their tractable behavior on average implies short descriptions for almost all the worst cases, as stated in the following

Claim 7 *If under a given enumerable input distribution, an NP-complete problem admits an algorithm with polynomial average-case complexity, then all but finitely many superpolynomial input cases, if they exist, are compressible.*

Proof: Let us consider a standard encoding of the inputs as binary strings. An instance x has length n if its encoding string has n bits. Fix also any algorithm A, provided it terminates on all inputs, and let us note by $t(x)$ the algorithmic time spent by A to solve the instance x. This allows us to consider the average complexity $\langle t_n \rangle$ obtained by running A on all instances of length n.

Let h_n be an integer function on n. It is convenient to say that the *hardness degree* of an instance x of length n is at least h_n, regarding the algorithm A, if $t(x) \geq h_n$. Let us consider the set H_n of instances of length n having hardness degree at least h_n : $H_n = \{x$ is n bits long : $t(x) \geq h_n\}$. From the Markov inequality the cardinal of H_n is bounded by: $d(H_n) \leq 2^n \langle t_n \rangle / h_n$.

Let H equal the set $\cup_n H_n$. H is recursive, provided that the input distribution is enumerable (H can be enumerated by running A on all instances). Let us study the sparseness of the set H. This comes to compute the limit $\lim d(H^{\leq n})/2^n$ when $n \to \infty$:

$$\lim_{n\to\infty} \frac{d(H^{\leq n})}{2^n} = \lim_{n\to\infty} \frac{\sum_{i\leq n} d(H_i)}{2^n} \leq \lim_{n\to\infty} \frac{\sum_{i\leq n} \frac{2^i \langle t_i \rangle}{h_i}}{2^n}.$$

If the above limit vanishes, by applying Fact 4 we conclude that for almost all $x \in H, K(x) < |x|$. The proof is simply done by choosing $\langle t_i \rangle$ of polynomial scaling and h_i of superpolynomial scaling.
□

Example: It is well known that under the uniform distribution 3-COL and 3-SAT have polynomial average-case complexity (see (Papadimitriou 1994) page 297 and (Koutsoupias & Papadimitriou 1992)). It follows that the instances leading to superpolynomial computations for 3-COL and 3-SAT are not random (assuming $P \neq NP$).

We remark that the hypothesis of Claim 7 is essential. For instance, it was shown (Levin 1986) that for the Tiling problem, the simple, uniform selection of instances leads to computations which are superpolynomial on average (unless there are polynomial on the average algorithms for every NP-complete problem and every simple distribution of its instances).

The intuition behind Claim 7 is that with the low value of the average complexity, there are very few cases with large computational cost. Then the hard cases can be described by giving their index in this small set. Similar considerations can be made for other classes of problems. In (Li & Vitányi 1992) it is showed that only non-random lists can achieve the worst-case complexity for Quicksort (for instance, the sorted or almost sorted lists).

Searching for Hard Graphs

It is generally accepted that hard instances of 3-COL occur near the point defining the step from one to zero in the probability to have solutions. These graphs hard to color are globally characterized by a critical value γ_{crit} of their connectivity[1] γ. Experimental (Cheeseman, Kanefsky, & Taylor 1991; Hogg & Williams 1994) and theoretical (Williams & Hogg 1994) results agree on locating this point at being somewhere with $\gamma_{crit} > 4$. On the other hand, a well-known theorem (see (Garey & Johnson 1979) page 85) states that with no node degree exceeding four, 3-COL remains NP-complete.

Since almost all hard graphs are concentrated near $\gamma_{crit} > 4$ (by accepting the conclusions from (Cheeseman, Kanefsky, & Taylor 1991; Williams & Hogg 1994)), it follows that the superpolynomial graphs (by

[1]The connectivity γ of a graph is defined as being the average of nodes degrees.

assuming $P \neq NP$) with $\gamma \leq 4$ are very rare. Claim 7 give us a characterization of these graphs, by implying that they are non-random and that they exhibit regularities allowing a compact description.

Let us proceed with an experimental confirmation. We focus on 3-COL problem and Brélaz's algorithm, with the input graphs having the same, fixed connectivity $\gamma < 4$. Brélaz's algorithm was also used to establish the experimental results about $\gamma_{crit} > 4$ from (Cheeseman, Kanefsky, & Taylor 1991). This is a complete backtracking procedure (Brelaz 1979; Turner 1988), which use heuristics: at each step an uncolored node with fewest remaining colors is chosen; ties are broken by selecting the node with maximal degree in the subgraph of uncolored nodes; remaining choices are made randomly.

A recent experiment (Vlasie 1995b) shows that inside the class of graphs having the same number of nodes μ and the same number of edges m, there is a direct relation between the number of 3-paths[2] and the coloring difficulty, provided that the graph connectivity $\gamma = 2m/\mu$ is smaller than 4. Namely, as the number of 3-paths decreases, the hardness and the density of hard graphs increase, until the probability of 4-cliques is very high[3].

The reported hard graphs are near this *phase transition* defined by the step in the probability to have 4-cliques, as the number of 3-paths decreases. Let us denote by $\sigma(g)$ the number of disjoint subgraphs with 4 nodes and 5 edges (that is, near to become 4-cliques) in the graph g. The experiment from (Vlasie 1995b) suggests that σ increases as the number of 3-paths decreases, but such a relation could be proved formally. In particular, the very hard cases have high values of σ.

Graph Description

The previous experiment suggests that the number σ of disjoint subgraphs with 4 nodes and 5 edges can discriminate between the easy and the hard graphs with given connectivity.

Starting from $\sigma = \sigma(g)$, an effective description $D(g)$ for the graph g can be done as follows:

1. draw σ disjoint graphs, each with 4 nodes and 5 edges;
2. draw the remaining $\mu - 4\sigma$ nodes;
3. draw the remaining $m - 5\sigma$ edges: these edges are given as a list of pairs of nodes.

Suppose that μ and m are fixed, and let us note by $G_{\mu,m}$ the set of all graphs with μ nodes and m edges. The length of the description $D(g)$ of a graph $g \in G_{\mu,m}$ can be computed as follows. A first block of bits records the *self-delimiting description* (see (Li & Vitányi 1993) page 72) of σ in $\log\sigma + 2\log\log\sigma$ bits. This information is sufficient to execute step 1 and step 2 given above. The encoding of σ is immediately followed by the list of the not yet drawn edges. This list is given as a succession of $m - 5\sigma$ blocks, each block encoding two nodes in $2\log\mu$ bits. The Kolmogorov complexity of a graph $g \in G_{\mu,m}$ is $K(g|\mu,m) \leq |D(g)|$, or:

$$K(g|\mu,m) \leq \log\sigma + 2\log\log\sigma + 2(m - 5\sigma)\log\mu. \quad (1)$$

Experimental Results

In order to show the influence of the graph structure on the coloring cost, we generated graphs with different values of parameter σ, while the number of nodes and the number of edges were fixed. The generating procedure is simple: first, σ disjoint subgraphs with 4 nodes and 5 edges are drawn, then the remaining number of edges is obtained by random drawings between nodes from different subgraphs. The graphs containing 4-cliques are rejected. One hundred graphs were generated for each value of σ. The searches exceeding three millions Brélaz steps were stopped, by considering this bound as illustrative for the coloring difficulty of graphs with 80 nodes (thus the average is underestimated). Figure 1 summarize the experimental results when $\mu = 80$ and $m = 136$ (i.e. $\gamma = 3.4$). In the top of the figure we plotted the average costs as a function of σ, while the bottom part of the figure shows the worst-case values at each value of σ.

The main observation is that the very hard instances occurs *only* when the graphs are highly structured. There is a transition point in the graph structure which distinctly separates a easy region of graphs from a hard one. In our case this transition is located at $\sigma = 15$.

Interpretation

According to Claim 7, if an NP-complete problem shows a tractable average behavior, regarding a given algorithm and a given enumerable input distribution, then for some n onwards, the input cases of size n showing a $O(2^n)$ behavior are necessarily compressible. Let us remark that for γ small almost all graphs are 3-colorable, with consequence that Brélaz's algorithm performs well on average, regarding the random, uniform distribution of graphs with a fixed, small connectivity (this is implied in essence by the results from (Turner 1988): almost all 3-colorable graphs are easy to color). Experimental confirmation is added by results from (Cheeseman, Kanefsky, & Taylor 1991; Hogg & Williams 1994) and also by our experiment(when σ is small). Hence the condition about the tractable average behavior holds.

Knowing the number of nodes μ and the number of edges m, any graph $g \in G_{\mu,m}$ can be encoded in a string of n bits, where

$$n = \log \left(\binom{\binom{\mu}{2}}{m} \right).$$

[2] A 3-path is any path of length three, denoted by an alternate succession of nodes and edges $x_1e_1x_2e_2x_3e_3x_4$, $x_1 \neq x_4$.

[3] A graph containing 4-cliques is trivially non-3-colorable.

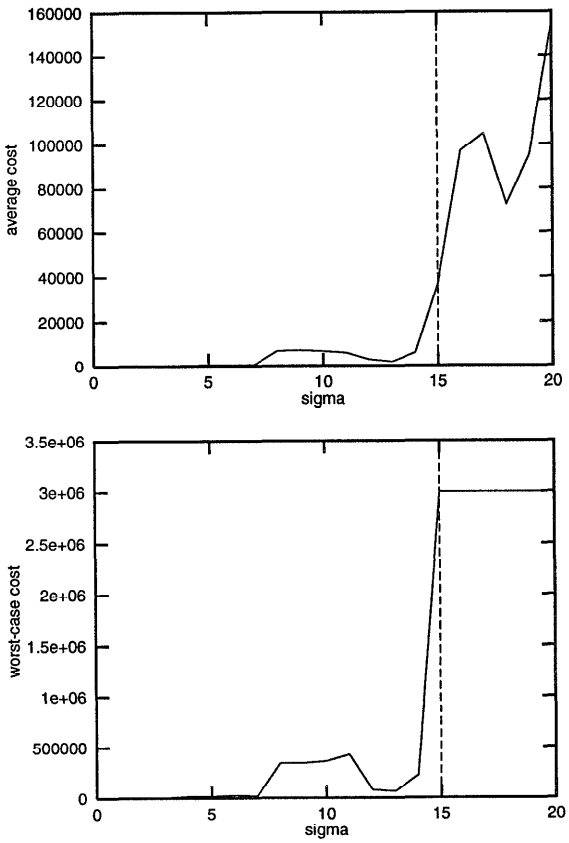

Figure 1: *Experimental results showing the dependence of coloring cost on graph structure; data were plotted for $\mu = 80$, $\gamma = 3.4$ and 100 graphs at each value of σ.*

According to Claim 7, the very hard graphs mostly occur only when $K(g|\mu, m) < n$. By considering the bound given by (1), it is sufficient to solve the following inequality in σ:

$$\log\sigma + 2\log\log\sigma + 2(m - 5\sigma)\log\mu < \log\left(\begin{pmatrix}\mu\\2\end{pmatrix}\right)$$
$$m$$

Letting $\mu = 80, m = 136$ in the above inequality gives $\sigma \geq 15$, which is in perfect correlation with the experimental result.

This experiment allowed us to illustrate Claim 7. By the way, let us remark that the experiment in Figure 1 illustrate also Theorem 6, i.e. the average complexity tends to worst-case complexity as the Kolmogorov complexity of the samples decreases.

More Questions

We claimed that inside the region of graphs with connectivity $\gamma < 4$, only the compressible graphs may be hard to color, with the exception of a finite number of cases. On the other side, in (Hogg & Williams 1994) the authors experimentally found that Brélaz's algorithm is erratically put into difficulty by very few random graphs located in the same region $\gamma < 4$ (the so called "second point transition"). Since the probability that a random graph generator produces a compressible graph is astronomically small, it seems that these random graphs contradict Claim 7. Our opinion is that experiments in (Hogg & Williams 1994) simply put in evidence members of the small set of exceptions allowed by Claim 7. These exceptions are likely to occur at small values of the number of nodes and they should simply vanish as the number of nodes is increased.

Another question is to see whether other algorithms do better on the instances on which Brélaz's algorithm has poor performances. We effectively tested on the *same* input instances the coloring algorithm described in (Vlasie 1995a), which has the particularity to combine a non-systematic search method with a complete one. The conclusion was that there are very few instances which remain hard for both algorithms, but there are many hard instances for the new algorithm alone and all these instances are compressible.

Furthermore, one may imagine a specific method to solve a particular type of compressible instances which are hard for a given algorithm. For example, one can exploit the repeated pattern of subgraphs with 4 nodes and 5 edges in order to facilitate the coloring of graphs which are hard for Brélaz's algorithm. But one should not forget that there are many other manners to get compressible instances, eventually hard for the algorithm in question and on which the specific method would be of little interest. Moreover, we remember that the Kolmogorov complexity as function of binary strings is not computable and that the incompressibility is not a decidable property, so there is little hope that one can imagine a general method for solving the hard cases by detecting and then exploiting their regularities.

Conclusion

With this paper we tried to provide theoretical and experimental evidence for the fact that sufficiently clever algorithms may be put in difficulty only by input data showing a high degree of regularity. This immediately implies that for problems like 3-SAT or 3-COL, the random generation of instances is not a reliable source of hardness. Whether a given instance of an easy on average problem is hard is a matter of the particular algorithm used to solve it. What is independent of the algorithm is the fact that the Kolmogorov complexity of a hard instance is small, a notion depending only of the instance itself.

Acknowledgments. The author would to thank Bruno Martin for his useful comments on previous versions of this paper.

References

Brelaz, D. 1979. *New methods to color the vertices of a graph. Comm. ACM* (22):251–256.

Calude, C. 1994. *Information and Randomness.* Springer Verlag.

Cheeseman, P.; Kanefsky, B.; and Taylor, W. 1991. *Where the Really Hard Problems Are.* In Kaufmann, M., ed., *Proceedings of IJCAI-91*, 331–337.

Crawford, J., and Auton, L. 1993. *Experimental results on the crossover point in satisfiability problems.* In *Proceedings of AAAI-93.*

Garey, M. R., and Johnson, D. S. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* Freeman.

Goldberg, A., and al. 1982. *Average time analysis of simplified Davis-Putnam procedures. Inf. Proc. Letters* (15):72–75.

Hogg, T., and Williams, C. 1994. *The hardest constraint problems: a double phase transition. Artif. Intell.* (69):359–377.

Koutsoupias, E., and Papadimitriou, C. H. 1992. *On the greedy heuristic for satisfiability. Inf. Proc. Letters* (43):53–55.

Levin, L. A. 1986. *Average Case Complete Problems. SIAM J. Comput.* (15):285–286.

Li, M., and Vitányi, P. M. 1990. *Kolmogorov Complexity and Its Applications.* In Leeuwen, J. V., ed., *Handbook of Theoretical Computer Science.* Elsevier - The MIT Press. 187–254.

Li, M., and Vitányi, P. M. 1992. *Average case complexity under the universal distribution equals worst-case complexity. Inf. Proc. Letters* (3):145–149.

Li, M., and Vitányi, P. M. 1993. *An Introduction to Kolmogorov Complexity and Its Applications.* Springer Verlag.

Minton, S., and al. 1992. *Minimizing conflicts: a heuristic repair method for constraint satisfaction and scheduling problems. Artificial Intelligence* (58):161–205.

Mitchell, D.; Selman, B.; and Levesque, H. 1992. *Hard and easy distributions of SAT problems.* In *Proceedings of AAAI-92.*

Musick, R., and Russell, S. 1992. How Long Will It Take? In *Proceedings of AAAI-92*, 466–471.

Papadimitriou, C. H. 1994. *Computational Complexity.* Addison-Wesley.

Turner, J. S. 1988. *Almost All k-Colorable Graphs Are Easy to Color. Journal of Algorithms* (9):63–82.

Vlasie, D. R. 1995a. *Combining Hill Climbing and Forward Checking for Handling Disjunctive Constraints.* In *Constraint Processing - Selected Papers.* LNCS 923 Springer Verlag Heidelberg. 247–266.

Vlasie, D. R. 1995b. *Systematic Generation of Very Hard Cases for Graph 3-Colorability.* In *Proceedings of 7-th IEEE ICTAI*, 114–119.

Watanabe, O., ed. 1992. *Kolmogorov Complexity and Computational Complexity.* EATCS Monographs, Springer-Verlag.

Williams, C. P., and Hogg, T. 1992. *Using Deep Structure to Locate Hard Problems.* In *Proceedings of AAAI-92*, 472–477.

Williams, C. P., and Hogg, T. 1994. *Exploiting the deep structure of constraint problems. Artificial Intelligence* (70):73–117.