

Learning on the Phase Transition Edge

Alessandro Serra, Attilio Giordana and Lorenza Saitta

Dipartimento di Scienze e Tecnologie Avanzate,
Università del Piemonte Orientale,
Corso Borsalino 54, 15100, Alessandria, Italy

Abstract

A previous research has shown that most learning strategies fail to learn relational concepts when descriptions involving more than three variables are required. The reason resides in the emergence of a phase transition in the *covering test*. After an in depth analysis of this aspect, this paper proposes an alternative learning strategy, combining a Monte Carlo stochastic search with local deterministic search. This approach offers two main benefits: on the one hand, substantial advantages over more traditional search algorithms, in terms of increased learning ability, and, on the other, the possibility of an a-priori estimation of the cost for solving a learning problem, under specific assumptions about the target concept.

1 Introduction

In a recent paper Giordana et al. [Giordana et al., 2000] have shown that relational learning becomes very hard when the target concept requires descriptions involving more than three variables. The reason is related to the presence of a phase transition in the covering test [Prosser, 1996; Giordana and Saitta, 2000], i.e., an abrupt change in the probability that an inductive hypothesis covers a given example, when the hypothesis and the example sizes reach some critical values. Moreover, any top-down learner will search for discriminant hypotheses in the phase transition region [Giordana et al., 2000; Giordana and Saitta, 2000], and, finally, heuristics commonly used to guide top-down relational learning [Quinlan, 1990; Botta and Giordana, 1993] become useful only in the same region. The consequence is that the top-down induction process is blind in its first steps, so that the path to the correct concept definition is very easily lost.

This paper offers two major contributions. First, the results reported by [Giordana et al., 2000] are extended, providing a more thorough analysis of the behavior of a top-down learner. A formal estimate of the "difficulty" of a learning problem is proposed, in terms of the density of sub-formulas of the target concept in the hypothesis space. Moreover, an estimate of the probability of detecting one of these is also given, as a function of the target concept location with respect to the phase transition.

Second, an induction algorithm, combining a Monte Carlo stochastic search [Brassard and Bratley, 1988] with local deterministic search, is proposed to (partially) avoid the pitfall that causes top-down search to fail. The new algorithm directly jumps into a region of the hypothesis space where the information gain heuristics has good chance of being successful, continuing its search exploiting a classical hill climbing strategy. The complexity of this algorithm is analyzed in a probabilistic framework, and it is proposed as a measure of the difficulty of the induction task. Finally, the algorithm has been experimentally evaluated on the set of hard induction problems provided by [Giordana et al., 2000], and it shows a good agreement with the theoretical estimates.

2 Covering Test in Relational Learning

A key step in Machine Learning is the *covering test*, i.e., checking whether an example e belonging to a learning set E verifies a hypothesis φ [Mitchell, 1997]. In relational learning e is usually represented as a conjunction of ground literals $e(a_1, \dots, a_r)$, where the a_i 's are constants belonging to a given set [Muggleton and De Raedt, 1994], and φ is a conjunction of non-ground literals $\varphi(x_1, \dots, x_n)$ containing the variables $x_1, \dots, x_n$. The covering test checks whether a First Order Logic formula φ has at least one model in the universe e . To this aim, a substitution θ of the variables $x_1, \dots, x_n$ with constants $a_1, \dots, a_r$ that satisfies φ is searched for.¹

This search process shows a phase transition for critical values of the number L of constants a_i 's in the universe, and the number m of literals in the hypothesis φ [Hogg et al., 1996; Prosser, 1996; Botta et al., 1999]. The graph in the (m, L) plane of the probability that a model of φ exists in e is reported in Figure 1(a), for a set of covering problems generated according to a random model described in [Botta et al., 1999]. In the following, we will refer to "phase transition" as to the locus of the points (m_c, L_c) for which the probability that a model exists is $P_{sol} = 0.5$. The phase transition is located in correspondence of the "waterfall" in Figure 1(a). To the left of the phase transition (in the YES-region, i.e., the high plateau in Figure 1(a)) almost any example verifies any hypothesis. To the right of the phase tran-

¹Verifying φ on e is a Constraint Satisfaction Problem, which exhibits a typical phase transition [Williams and Hogg, 1994; Smith and Dyer, 1996; Prosser, 1996].

sition (in the low plateau, i.e., the NO-region) almost no example verifies any hypothesis. The steep step corresponds to the "mushy" region, where the probability of a model existing drops from 0.99 to 0.01. In this region, the complexity of the search reaches its maximum, as reported in Figure 1(b). More details can be found in [Giordana *et al.*, 2000; Botta *et al.*, 1999].

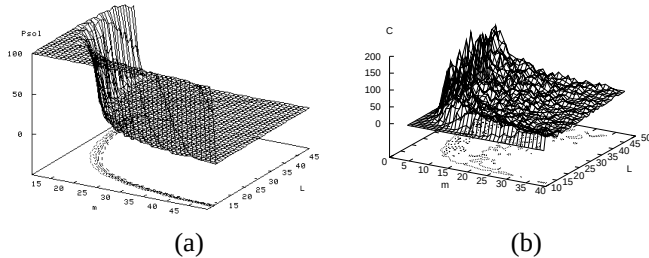


Figure 1: Example of phase transition in the covering test problem. All hypotheses have four variables and contain only binary predicates. Each predicate is associated to a table with 100 tuples in each example. (a) Probability that a model exists for 4 variables. (b) Corresponding complexity peak for the search of a model.

If the number of variables in the formulas changes, the location of the phase transition moves, as described in Figure 2.

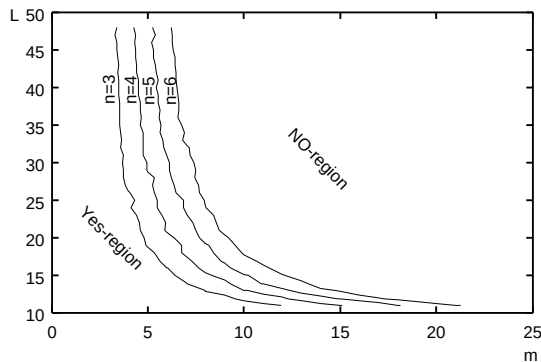


Figure 2: Dependency of the phase transition location upon the number of variables in a hypothesis.

3 Top-Down Induction

Let us consider a learning set $\mathcal{E}$ and a conjunctive concept ω . Every pair (e, ω) , with $e \in \mathcal{E}$, represents a covering test (m_ω, L_e) in the (m, L) plane. A learning problem corresponds then to a cloud of points [Giordana *et al.*, 2000]. In the special case that the number of constants in all examples is the same, the entire learning problem $(m_\omega, L_\mathcal{E})$ corresponds to a single point. In a similar way, any inductive hypothesis φ , built up by a learner in the attempt of finding ω , or at least a good approximation $\hat{\omega}$ of it, can be represented as a cloud of points (or a single point) in the (m, L) plane. Then, an

induction process that proceeds by generating a sequence of hypotheses will describe a family of lines (or a single line) parallel to the m axis.

In order to investigate how the presence of a phase transition in the (m, L) plane may affect the behavior of a relational learner, an extensive experimentation has been done using a large set of artificially generated learning problems². The results have been reported and discussed in [Giordana *et al.*, 2000]. Here experimental setting and the findings are recalled, in order to state the base line for the work described in this paper.

The artificial learning problem set has been generated according to the following procedure. Given n and N , let (m, L) be a point in the plane (m, L) . A set of 451 points has been sampled in the plane (m, L) . Then, for each point, a learning problem $\Pi_{m,L}$ has been built up from the 4-tuple $(n = 4, N = 100, m, L)$. A target concept ω with m literals and n variables has been built up, and then a training set $\mathcal{E}_L$ and a test set $\mathcal{E}_T$ have been generated. Let Λ be a set of L constants; every example is a collection of m relational tables of size N obtained by sampling the binary table $\Lambda \times \Lambda$. In order to generate balanced training and test sets in each point of the (m, L) plane, the random generation of the examples has been modified in order to obtain examples with models also in the NO-region, and examples without models also in the YES region (see [Giordana *et al.*, 2000] for more details). The result has been the generation of training and test sets, $\mathcal{E}_L$ and $\mathcal{E}_T$, each one with 100 positive and 100 negative examples in each (m, L) point.

The experimentation has been done primarily using FOIL [Quinlan, 1990]. A problem has been considered solved when the concept description found on $\mathcal{E}_L$ was at least 80% accurate on $\mathcal{E}_T$. However, other learners, such as SMART+ [Botta and Giordana, 1993] and G-NET [Anglano *et al.*, 1998], have also been tried for comparison, reporting a strong agreement with FOIL both for the positive and for the negative cases. The results can be summarized as in the following. First, all tried learning strategies always end up generating concept descriptions lying on, or very close to, the edge of the phase transition. As discussed by [Giordana and Saitta, 2000], this phenomenon systematically occurs both in artificial and in real learning problems, as soon as the generated hypotheses involve more than three variables.

Second, a large "blind spot" located across the mushy region has been found: when the target concept lies there, no tried learner has been able to find any good generalization of the target concept (see Figure 3). Quite surprisingly, learning problems become again solvable by top-down induction when they are located in the NO-region, to the far right of the phase transition.

Finally, the computational complexity of matching inductive hypotheses lying in the mushy region may render the whole induction process infeasible. A possible way out has been proposed by [Giordana and Saitta, 2000], who suggested to use a stochastic algorithm for on-line estimation of the

²All the artificial learning problems defined in this experimental setting are available at: <http://www.mfn.unipmn.it/~attilio/PROJECTS/phase-trans.html>.

complexity, at the possible expenses of correctness.

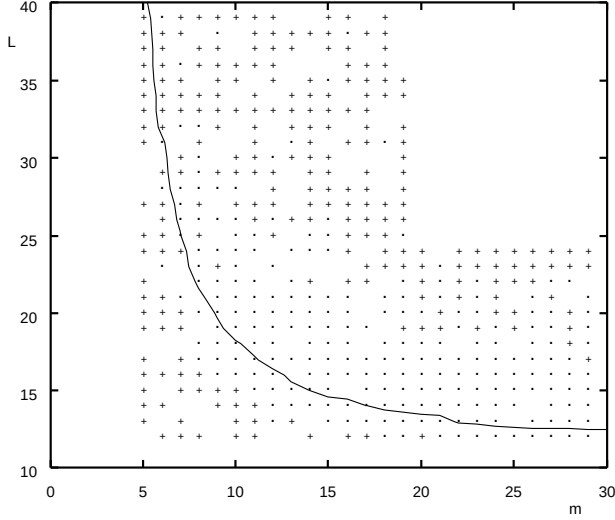


Figure 3: Relational learning with FOIL: *Failure region* (legend “.”) and *success region* (legend “+”), for $n = 4$ and $N = 100$. The contour plot corresponds to the value $P_{sol} = 0.5$ of the probability that randomly generated covering test is positive.

4 Why Top-Down Induction is Misled

Before possibly designing new heuristics, it is necessary to clearly understand why FOIL and the other learners fail in the blind spot. A preliminary analysis showed that heuristics relying on the number of models, hypotheses have on the examples [Quinlan, 1990; Botta and Giordana, 1993; Rissanen, 1978], are not reliable in the YES-region, because both wrong and correct generalizations of the target concept have there similar numbers of models both on the positive and on the negative examples. In support to this claim, Figure 4(a) plots the average number μ of models a random hypothesis has as a function of m , for $L = 18$. It is clear that an increase by 1 in the number n of variables induces a large increase of μ and an even larger one on μ 's standard deviation σ . Actually, when $n = 4$ the standard deviation is larger than the number of positive examples in the learning set. Then, even if we assume that generalizations of the target concept ω have at least one model more on the positive examples than on the negative ones, these generalizations cannot be distinguished from a purely random hypothesis. On the contrary, for hypotheses with 2 or 3 variables only, μ is much smaller, and we may expect that, provided that a correct generalization $\hat{\omega}$ of ω with only two or three variables exists, it should be far easier to find it than any other one having four variables. This conjecture agrees with the fact that many solutions actually generated by FOIL have only three variables, even though the target concept is described by a formula with four variables.

A more detailed analysis is reported in Figure 5, where the line corresponds to the phase transition for $n = 4$. In Figure 5(a) the “+” denote the locations of a set of target concepts ω , whose description contains four variables. Figure 5(a') shows the location of the solutions found by FOIL. Most of

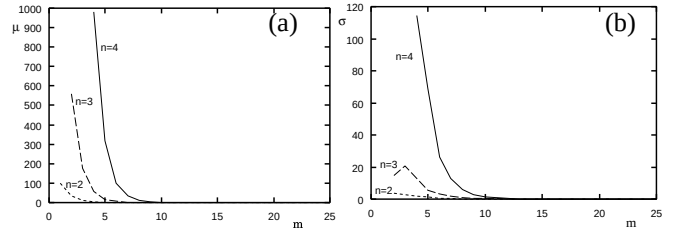


Figure 4: Number μ of models of a random hypothesis versus m , for various values of n . (a) μ for $n = 2, n = 3$ and $n = 4$. (b) Standard deviation of μ .

these solutions are generalizations of ω with four variables, because no acceptable generalizations with three or less variables could be found. On the contrary, Figure 5(b) shows the locations of a set of target concepts whose description still contains four variables, but for which FOIL was able to find generalizations with no more than three variables (reported in Figure 5(b')). Problems in Figure 5(b) are then easier to solve than problems in Figure 5(a): this result corresponds to the fact that problems in Figure 5(a) are closer to the phase transition than problems in Figure 5(b). We observe that the probability of a problem being correctly approximated by a formula with only three variables increases with the number of literals in ω . In fact, when ω contains more literals, the number of its subformulas that are correct generalizations of ω increases as well; then, we may expect that at least some of them contains a number of variables smaller than ω 's. Moreover, the difficulty of a learning problem increases when L decreases, whereas the critical value m_c increases. When $L < 20$ the only solutions found have three variables. In conclusion, most learning problems in the NO-region have been solved only because approximations with only three variables existed. When more than three variables are required, m_c becomes larger, and available heuristics for top-down search are unable to grow valid inductive hypotheses starting from the YES-region. On the other hand, we could not find any bottom-up learning algorithm capable to cope with the complexity of working directly inside the NO-region.

5 A Stochastic Approach

In the absence of reliable heuristics, stochastic search may be a valid alternative, especially if combined with deterministic search. An example of an effective combination of a Monte Carlo search with deterministic search in the n-Queens problem can be found in [Brassard and Bratley, 1988]. The algorithm we propose is based on a two-step strategy. The first step creates a hypothesis φ_γ with a complexity (number of literals) γ sufficient to reach the border between the YES-region and the mushy region; φ_γ is the result of random sampling of the hypothesis space. The second step consists of a general to specific search, starting from φ_γ according to a hill-climbing strategy guided by the information gain heuristics.

Algorithm SFind

Let $\hat{\omega}_{cur} = \emptyset$

while halt condition does not hold **do**

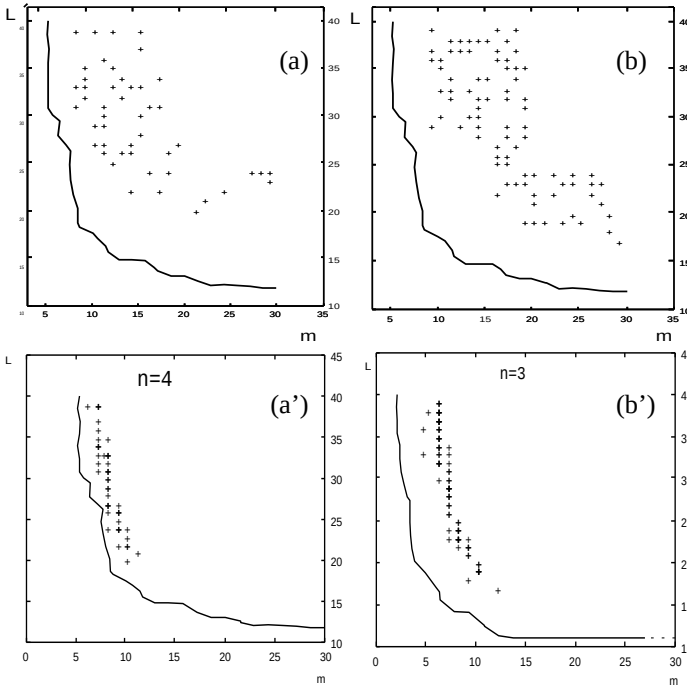


Figure 5: Solutions generated by FOIL for problems in the NO region. (a) Problems solved with four variables; (b) problems solved with hypothesis containing only three variables; (a') and (b') location in the (m, L) plane of the solutions generated by FOIL corresponding to the problems in Figura (a) and (b), respectively.

1. Randomly generate a hypothesis φ_γ close to the mushy region
2. Make φ_γ more specific by following a hill-climbing strategy guided by the information gain. Let $\hat{\omega}_t$ be the locally best hypothesis found in trial t .
3. **if** $\hat{\omega}_t$ is better than $\hat{\omega}_{cur}$ **then** replace $\hat{\omega}_{cur}$ with $\hat{\omega}_t$.

end

If the target concept ω has a conjunctive description in the hypothesis space H , the Monte Carlo algorithm *SFind* is very likely, in the long run, to find ω , or at least a good generalization $\hat{\omega}$ of it, correct on the learning set $\mathcal{E}$. Now, we want to answer the following question: assuming that the complexity m_ω and the number of variables n_ω in ω are known, what is the complexity we would expect from Algorithm *SFind*? As "complexity" of the algorithm we mean the minimum number T_α of trials necessary to reach a probability $(1 - \alpha)$ of finding $\hat{\omega}$ (or ω , as a specially lucky case). If we choose $\gamma = m_c - 1$, there is a strong experimental evidence that a hill-climbing search, guided by the information gain, almost surely will find $\hat{\omega}$. This happens because the search explores the region where the information gain becomes reliable.

Let P_G be the probability that φ_γ is a sub-formula of ω , then the probability $P_{\hat{\omega}}^{(t)}$ of finding $\hat{\omega}$ in no more than t steps

is given by the expression:

$$P_{\hat{\omega}}^{(t)} = 1 - (1 - P_G)^t \quad (1)$$

Given m_ω and n_ω , the probability P_G can be estimated, for a generic number τ of literals, as the ratio $P_G = N_S(m_\omega, n_\omega, \tau, v) / N_\varphi(m_\omega, \tau, v)$, where $N_S(m_\omega, n_\omega, \tau, v)$ is the number of sub-formulas of ω with τ literals and v variables ($2 \leq v \leq n$), and $N_\varphi(m_\omega, \tau, v)$ is the total number of formulas in the hypothesis space with τ literals and v variables. An analytical expression for $N_S(m_\omega, n_\omega, \tau, v)$ and $N_\varphi(m_\omega, \tau, v)$ has been given in [Serra, 2000], under the assumption that all predicates are binary. Using such expres-

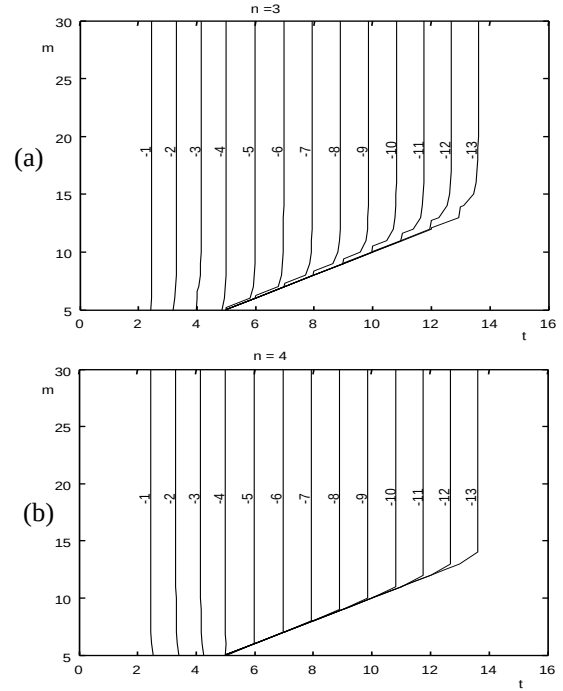


Figure 6: Contour plot of the average proportion of sub-formulas of a target concept ω as a function of m and τ , for $n = 3$ and $n = 4$. (the parameters on the curves are the exponent of 10)

sions, the contour plot of Figure 6 have been obtained for the ratio $R(m_\omega, n_\omega, \tau, v) = N_S(m_\omega, n_\omega, \tau, v) / N_\varphi(m_\omega, \tau, v)$ setting $n_\omega = 4$, $v = 3$ and $v = 4$. It is immediate to verify that $R(m_\omega, n_\omega, \tau, v)$ decreases exponentially with τ , while it remains constant with respect to m , after an initial exponential decay for small values of m . The value P_G can be obtained from Figure 6 by setting $\tau = \gamma$. As $\gamma = m_c - 1$ depends upon L (see Figure 1), P_G depends upon L_c , too. The plots of Figure 7 report the dependency of P_G upon L_c for in the case of $v = 3$ (dotted line), and in the case of $v = 4$ (continuous line). For $n = 3$ the phase transition occurs for smaller values of m_c than for $n = 4$. Consequently, the values of P_G for $v = 3$ are order of magnitude larger than for $v = 4$. This explain why learning a concept with three variables only is much easier than learning a concept with four variables, as it has been stated in the previous section. We

notice that the value L_c is known when the learning examples are given, because L_c is the average number of constants occurring in them.

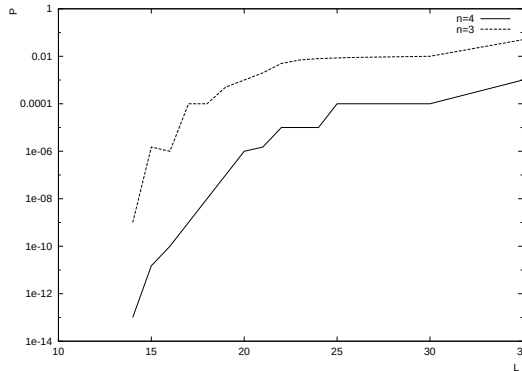


Figure 7: Proportion of correct generalizations of ω evaluated on the edge of the phase transition, i.e., when the probability of solution is $P_{sol} = 0.9$, versus L_c .

6 Improving the Stochastic Search Algorithm

Expression (1) and the graphs in Figure 7 allow one to estimate the number T_α of trials required to reach a confidence $(1-\alpha)$ that Algorithm *SFind* has found an approximation $\hat{\omega}$, provided that such an approximation exists. Then, the search can be planned as follows:

1. Assume an initial value n_ω for the number of variables in ω .
2. Estimate the number T_α of trials necessary to reach the confidence level $(1-\alpha)$. If T_α is too high, stop, otherwise go to the next step.
3. Run *SFind* for T_α trials.
4. If the result returned by *SFind* is acceptable, stop, otherwise increase n_ω by 1 and go to step 2.

We notice that the cost for generating and evaluating a hypothesis in the initial random sampling phase is much lower than the cost for the subsequent hill-climbing search. More specifically, the number of hypotheses to be generated and evaluated for the hill-climbing step can be estimated by:

$$O((1+\nu)mn(n-1)) \quad (2)$$

being ν an integer that experimentally has been observed to range from 0 to 3. In expression (2) the term $(1+\nu)$ is an estimate of the number of literals to be added to a hypothesis φ_γ in order to cross the phase transition, whereas the term $mn(n-1)$ estimates the number of alternative specializations to be considered at each hill-climbing step.

Then, the number of hypotheses to be evaluated in the hill-climbing phase is from one to two orders of magnitude larger than the number of stochastic trials.

A second point worth noticing is that, assuming that the information gain heuristics is reliable for any hypothesis $\varphi_\gamma \wedge \psi$, it is also likely that φ_γ itself be scored higher than the average when it is a sub-formula of ω .

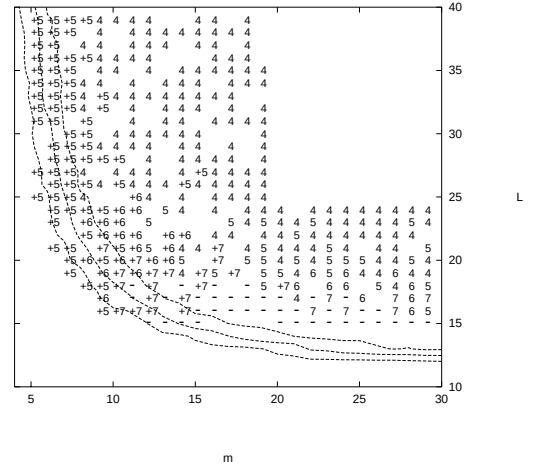


Figure 8: Results obtained by Algorithm T^4 . The numbers denote the minimum value that was necessary to assume for γ in order to solve the learning problem. When the number is prefixed by "+", it means that $n = 4$ has been assumed, otherwise $n = 3$. Symbol "." means that the problem has not been solved.

On the basis of the previous considerations we introduce a new algorithm, which can be more effective in practice than Algorithm *SFind*. Let T_α be the number of trials estimated by (1) in order to reach a confidence $(1-\alpha)$ on the output of algorithm *SFind*. Let moreover K ($1 \leq K \leq T_\alpha$) be a user-defined parameter.

Algorithm T^4

1. Create a set Φ of T_α hypotheses with complexity γ .
2. Rank hypotheses in Φ according to their information gain with respect to the formula *True*.
3. Starting from the top ranked hypothesis, apply the hill-climbing specialization step to the K best ranked hypotheses.
4. Return the best description $\hat{\omega}$.

Algorithm T^4 tries to limit the number of hill-climbing steps to the hypotheses looking more promising, reducing thus the computational complexity. Of course, parameter K is an arbitrary choice, and the confidence level $(1-\alpha)$ is guaranteed to be reached only when $K = T_\alpha$. In practice, we have observed that using values of K relatively small ($K = 1\%$ of T_α) T^4 tends to produce the same results as for $T_\alpha = K$.

Algorithm T^4 has been tested on the set of learning problems, reported in Figure 3, trying to solve the problems laying in the NO-region following the strategy described above. We started with the minimal hypothesis $n_\omega = 3$. Parameter γ has been set $\gamma = 4$, which approximately corresponds to the edge of the phase transition for $n = 3$ and $L \geq 30$. Even if for smaller values of L our strategy foresees larger values of γ , it has been found that T^4 was able to find a solution, with this

setting, also for many problems where $20 \leq L \leq 30$. Afterwards more expensive hypotheses have been progressively considered firstly increasing γ , until the value foreseen by the phase transition has been reached (or an excessive cost was foreseen) and then setting $n_\omega = 4$. A value $K = 1\%$ of T_α has been used everywhere. The results are reported in Figure 8. Comparing to Figure 3, we see that many problems lying in the blind spot have been solved. In practice almost all the problems above the phase transition and the line $L = 18$ have been solved with a complexity larger as FOIL but still affordable. A cluster of 20 Pentium III (800 Mhz) has been used for the experiments, where every single problem required a time ranging from few minutes to several hours (an elapsed time comparable to FOIL on a sequential machine). When the problem was not solved we progressively increased the complexity of the hypothesis increasing γ .

7 Discussion

We have shown that combining stochastic search with local deterministic search it is possible to learn approximated concept descriptions where no known classical algorithm was successful. Even if the algorithm is used under the stringent assumption that a conjunctive concept description exists, it is not difficult to extend it in order to cope with more general concept descriptions. For instance, disjunctive descriptions can be learned by integrating T^4 with a set covering algorithm as it is made in most relational learner [Quinlan, 1990; Botta and Giordana, 1993].

However, this is not the fundamental result that emerges from the framework we propose. In our opinion, the most important outcome is the method for estimating the complexity of a learning problem: given a specific hypothesis about the structure of the concept, we have a method for predicting the expected cost for testing the hypothesis. Moreover, a criterion for deciding on-line when stop testing the hypothesis is provided.

A second important result is a negative one, and concerns the possibility of learning descriptions with many variables. Even under the simplistic assumption that all predicates are relevant to the concept description, the task looks hard for many concepts requiring at least 4 variables. Increasing the number of variables, the complexity rises up exponentially. Considering the presence of irrelevant predicates, the analysis we performed still holds, but the density of sub-formulas of the target concept close to the phase transition becomes even more tiny, and so the difficulty will increase further.

References

- [Anglano *et al.*, 1998] C. Anglano, A. Giordana, G. Lobello, and L. Saitta. An experimental evaluation of coevolutionary concept learning. In *Proceedings of the 15th International Conference on Machine Learning*, pages 19–23, Madison, WI, July 1998.
- [Botta and Giordana, 1993] M. Botta and A. Giordana. SMART+: A multi-strategy learning tool. In *IJCAI-93, Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, pages 937–943, Chambéry, France, 1993.
- [Botta *et al.*, 1999] M. Botta, A. Giordana, and L. Saitta. An experimental study of phase transitions in matching. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence*, pages 1198–1203, Stockholm, Sweden, 1999.
- [Brassard and Bratley, 1988] G. Brassard and P. Bratley. *Algorithmics: Theory and Practice*. Prentice Hall, Englewood Cliffs, NJ, 1988.
- [Giordana and Saitta, 2000] A. Giordana and L. Saitta. Phase transitions in relational learning. *Machine Learning*, x:to appear, 2000.
- [Giordana *et al.*, 2000] A. Giordana, L. Saitta, and M. Sebag and M. Botta. An experimental study of phase transitions in matching. In *Proceedings of the 15th International Conference on Machine Learning*, Stanford, CA, 2000.
- [Hogg *et al.*, 1996] T. Hogg, B.A. Huberman, and C.P. Williams, editors. *Artificial Intelligence: Special Issue on Frontiers in Problem Solving: Phase Transitions and Complexity*, volume 81(1-2). Elsevier, 1996.
- [Mitchell, 1997] T.M. Mitchell. *Machine Learning*. McGraw Hill, 1997.
- [Muggleton and De Raedt, 1994] S. Muggleton and L. De Raedt. Inductive logic programming: theory and methods. *Journal of Logic Programming*, 19/20:629–679, 1994.
- [Prosser, 1996] P. Prosser. An empirical study of phase transitions in binary constraint satisfaction problems. *Artificial Intelligence*, 81:81–110, 1996.
- [Quinlan, 1990] R. Quinlan. Learning logical definitions from relations. *Machine Learning*, 5:239–266, 1990.
- [Rissanen, 1978] J. Rissanen. Modeling by shortest data description. *Automatica*, 14:465–471, 1978.
- [Serra, 2000] A. Serra. Apprendimento di concetti relazionali. Technical Report TR00-12, DISTA, Università del Piemonte Orientale, Feb 2000.
- [Smith and Dyer, 1996] B.M. Smith and M.E. Dyer. Locating the phase transition in binary constraint satisfaction problems. *Artificial Intelligence*, 81:155–181, 1996.
- [Williams and Hogg, 1994] C.P. Williams and T. Hogg. Exploiting the deep structure of constraint problems. *Artificial Intelligence*, 70:73–117, 1994.