Impact of structure on parallel local search for SAT

Andrea Roli
DEIS – Università degli Studi di Bologna
Viale Risorgimento, 2 – Bologna (Italia)
aroli@deis.unibo.it

1 Introduction

The phenomenon called *criticality and parallelism* has been observed in the context of local search algorithms applied to combinatorial optimization problems [7, 5, 4], where local search is modified by applying more than one local move in parallel. It has been shown that the effectiveness of these algorithms depends on the parallelism degree τ (number of simultaneous moves): if τ increases, the solution quality also increases up to a maximal point (corresponding to τ_{opt}) at which it starts to decrease. It has also been shown that τ_{opt} is negatively correlated with the *connectivity* among variables of the problem: the higher the connectivity, the lower τ_{opt}. The average connectivity of a problem estimates the direct influence among variables. For instance, in *NK models* [5, 4], each of the N variables is connected with exactly other K variables, therefore the connectivity is K.

The same phenomenon has also been recently observed in local search for SAT [8] and MAX-SAT [9] problems. In this case, the average connectivity is defined as the average node degree of the constraint graph associated with the problem instance.

Recent studies [11] pointed out the different distribution of node degree in random and structured problems and its impact on search. The aim of this work is to investigate the effects of problem structure on the critical parallelization of local search for SAT problems, and compare them with previous results which concerned random instances [8, 9].

2 Preliminaries

In this work we use the expression "parallel local search" with the meaning of local search in which more than one local move is synchronously performed[1]. Typical tackled problems are NP-hard combinatorial optimization problems and NP-complete constraint satisfaction problems. The parallelization of local search can be achieved in different ways and the most important applied so far are:

[1]These algorithms can be both sequentially and parallel implemented.

Algorithm 1 Parallel GSAT

Input: a set of clauses α, τ, $MAXMOVES$
Output: a truth assignment of α
$T \leftarrow$ initial truth assignment
$MOVES \leftarrow 0$; $besterror \leftarrow Eval(\alpha, T)$; $bestsolution \leftarrow T$
while $MOVES < MAXMOVES$ **do** {The stop condition depends on the maximum number of non-improving moves}
  if T satisfies α **then**
    return T
  end if
  Divide the set of variables in τ subsets (randomly)
  for all subset X_k of variables ($k = 1, 2, \ldots, \tau$) **do**
    $p_k \leftarrow$ a propositional variable in X_k such that a change in its truth assignment gives the largest decrease in the number of clauses of α that are not satisfied by T
  end for
  $T \leftarrow T$ with $p_1, p_2, \ldots, p_\tau$ reversed;
  $MOVES \leftarrow MOVES + 1$
  $error \leftarrow Eval(\alpha, T)$
  if $error < besterror$ **then**
    $MOVES \leftarrow 0$; $bestsolution \leftarrow T$; $besterror \leftarrow error$
  end if
end while
return $bestsolution$

- At each iteration, apply a local move on each variable (or a solution component) with probability p. This results in an average parallelism of pn, where n is the number of variables.
- Divide the problem in τ subsystems (which are, in general, not independent) and apply local search to optimize each of them independently.

An example of the first process is given in [7], where Simulated Annealing [6] is applied on problems defined over binary variables. Instead of performing one random flip, every variable is flipped with probability p. The resulting average parallelism is pn, where n is the number of variables. Examples of the second approach can be found in [5, 4], where NK lattice models are optimized respectively by subdividing them in τ *patches*, independently optimized.

The most important results can be summarized in the following properties:

- The algorithm performance increases as parallelism increases, up to a parallelism degree (τ_{opt}) at which it starts to decrease.
- The optimal parallelism depends on a parameter which measures the connectivity of the system.

The same phenomenon has been discovered in parallel local search for SAT [8] and MAX-SAT [9], where a parallel version of GSAT [10] has been applied (see Algorithm 1).

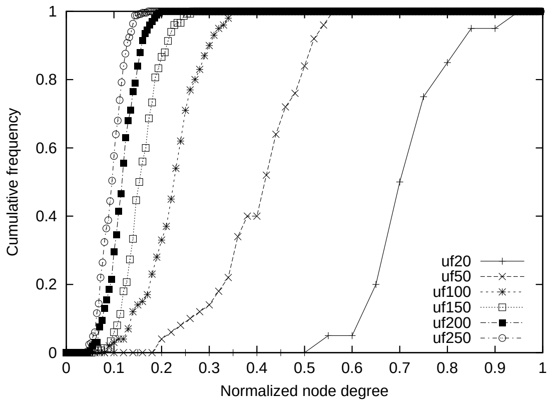

Figure 1: Cumulative frequency vs. normalized node degree for Uniform Random 3-SAT instances in the threshold region.

Variables are divided in τ subsets and, for each subset, the variable that, if flipped, decreases the greatest number of unsatisfied clauses, is flipped. In short, we can say that, for every subset, the "best variable" according to the GSAT criterion is flipped. For both SAT and MAX-SAT the objective function, to be minimized, is the number of unsatisfied clauses.

Previous work concerned random instances, for which the average connectivity can be enough informative. In the following sections the differences in node degree distribution between random and structured instances will be analyzed and their impact on Parallel GSAT will be investigated.

3 Random Instances

The graph associated with a SAT instance is an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$, where each $v_i \in \mathcal{V}$ corresponds to a variable and edge $(v_i, v_j) \in \mathcal{A}$ $(i \neq j)$ if and only if variables v_i and v_j are in a same clause. In an instance with n variables, each node v_i, $i = 1, \ldots, n$, has a degree $q_i \in [0, n-1]$. For k-SAT problems, defined as conjunction of clauses with exactly k literal each, it holds $k - 1 \leq q_i \leq n - 1$.

We define $q = \frac{1}{n} \sum_{i=1}^{n} q_i$, the average connectivity of the instance and, to make direct comparisons among instances with different number of variables, we also introduce the normalization of q: $\overline{q} = \frac{q}{n-1}$. In order to compare the node degree distribution between instances, we consider the cumulative frequency $Freq(j) = \{$frequency of a node connected to not more than j nodes$\}$. Fig.1 shows the cumulative frequency vs. the normalized node degree for random 3-SAT instances retrieved from SATLIB [3]. The instances belong to the threshold region (clauses/variables ≈ 4.3) and are satisfiable. Note that the curves are quite regular and, as the number of variables increases, they converge to a step function located at the average node degree.

We analyzed the behavior of parallel GSAT (hereafter referred to as PGSAT) as a

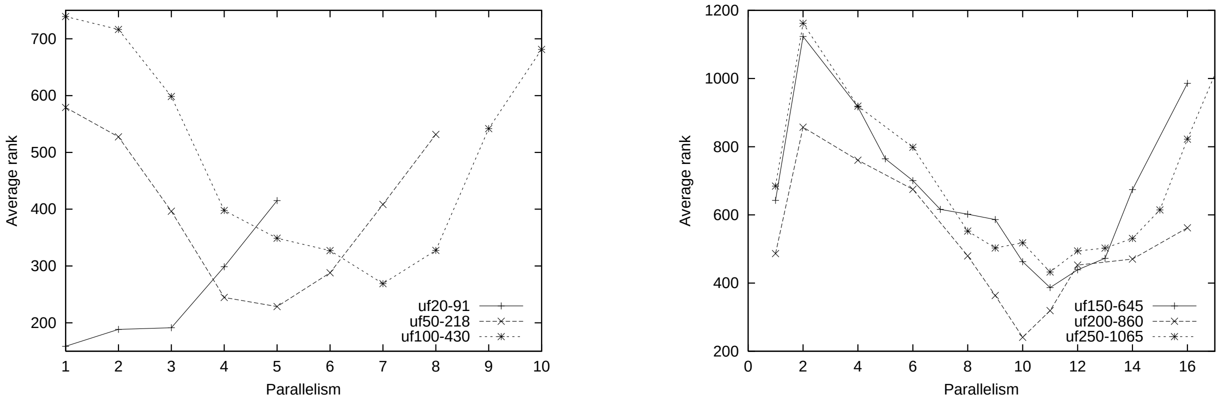

Figure 2: Average rank against parallelism in random 3-SAT instances. Results are ranked on the basis of solution quality and number of iterations. The lower the rank, the better. The ranks of each parallelism degree are then averaged. Results are averaged over 100 trials.

function of parallelism degree τ, i.e., the number of parallel flips. We performed a non-parametric statistics [1] to obtain a significative comparison even with a small number of trials. PGSAT has been executed 100 times on each instance for each value of τ. The results have been ranked on the basis of the quality of solution, computed as number of unsatisfied clauses[2]. The lower the rank, the better. The ranks of the trials corresponding to each value of τ have been averaged and results are shown in Fig.2. If the ranks of the trials belonging to the same τ were all in sequence[3], the best value of τ would have an average rank of 50.5, the second of 150.5, the third of 250.5 and so on.

We can observe that, for every instance type, there is an optimal parallelism τ_{opt} and τ_{opt} increases with n (though for larger instances the behavior is less marked because of very small differences in connectivity). The fraction of trials terminated with a satisfying assignment follows approximately the same behavior, with an optimum of 100% solved corresponding to the optimal parallelism. These results are in accord with previous ones on random forced 3-SAT and MAX-SAT [8, 9]. In the next section we will analyze structured instances, compare them with random ones and show PGSAT experimental results.

4 Structured instances

Structured instances are characterized by the presence of some regularity in their components, for example graph problems based on ring or lattice topology, SAT problems obtained by encoding logic problems (circuit testing, inductive inference, planning, etc.). Fig.3 shows the curve of cumulative frequency for structured SAT instances taken from SATLIB, produced by encoding a Blocks World Planning Problem (*huge*), a Logistics

[2]Ties have been broken on the basis of the number of iterations, favoring runs with smaller number of iterations.

[3]i.e., for every value of τ, $\forall\, i, j = 1, \ldots,$number of trials, $|\text{rank}_\tau(i\text{-th trial}) - \text{rank}_\tau(j\text{-th trial})| \leq$ number of trials.

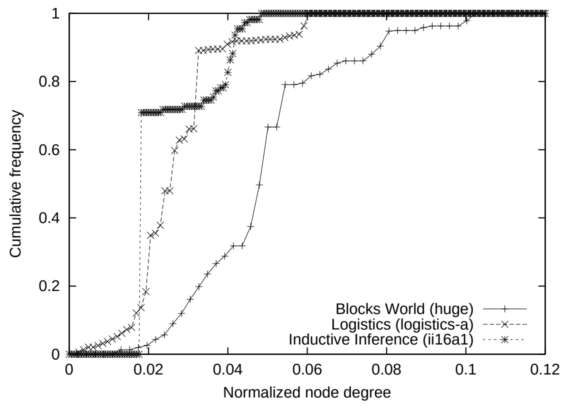

Figure 3: Cumulative frequency vs. normalized node degree in structured instances.

Planning Problem (*logistics-a*) and Inductive Inference (*ii16a1*). The plotted curves show apparent differences with those of random SAT problems. They are not as much regular as random ones and they have gaps and plateaus, especially in the uppermost part of the curve. Structured instances thus have a more spread and non-uniform connectivity distribution.

The node degree cumulative frequency of *ii16a1* is the most peculiar and differs the most from that of random instances. This instance has 1650 variables and a normalized average connectivity $\overline{q}_{ii16a1} = 0.024$. Its cumulative frequency is shown in Fig.4, along with the cumulative frequency of a random 3-SAT instance of the same size and normalized connectivity. Fig.5 shows the rank (averaged over 10 trials) against the parallelism. First of all, we observe that the optimal parallelism is higher for *ii16a1* than for *3SAT-1650*. Furthermore, the shape of the *ii16a1* curve is dramatically more rugged than the other one. Note also that since the *ii16a1* curve is above the *3SAT-1650* curve, the results on the structured instance are more spread and ranks of different parallelism degrees more interleaved than for the random one. These differences seems to be correlated with the differing distribution of connectivity.

To investigate more deeply this behavior, we generated instances with a controlled amount of structure by means of a technique called *morphing* [2]. This methods permits to generate instances gradually morphing from source to destination instance by varying a parameter $p \in [0, 1]$. The higher p used to generate an instance, the more similar to the source. To be applied on SAT problems, the method needs instances with the same number of variables and clauses. We generated a satisfiable random 3-SAT instance with 1650 variables and 19368 clauses (*3SAT-1650_large*), the same number as *ii16a1*. With $p = 1$ we obtain *ii16a1* and with $p = 0$ *3SAT-1650_large*, thus p controls the amount of structure in the instance. The generated instances are no more guaranteed to be satisfiable, but, as shown in [9], the results are qualitatively the same both for SAT and MAX-SAT. Moreover, the use of approximate algorithm and nonparametric statistics permits to directly compare results. Fig.6 shows the node degree cumulative frequency of source (structure), destination

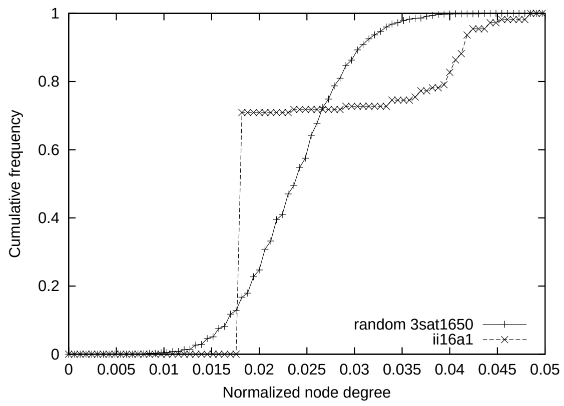

Figure 4: Cumulative frequency vs. normalized node degree in a structured instance (inductive inference) and a random 3-SAT instance of same size ($n = 1650$) and connectivity ($\overline{q} \approx 0.024$).

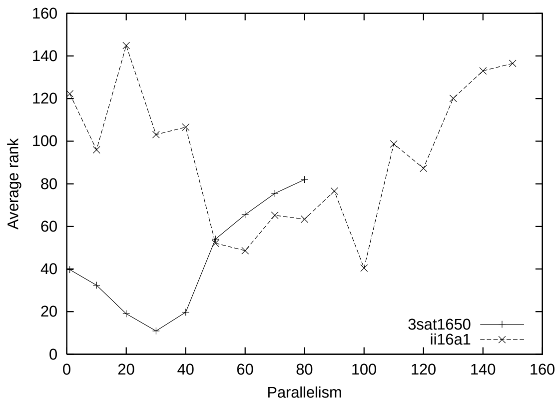

Figure 5: Average rank vs. parallelism for structured instance *ii16a1* and random 3-SAT instance *3sat1650*. Results are averaged over 10 trials.

(random) and instances generated by the morphing method. The results of 10 trials of PGSAT for different values of parallelism are shown in Fig.7. We first observe that the optimal parallelism increases with p, therefore we can conjecture that the more similar an instance to *ii16a1*, the closer should be their optimal parallelism degrees. We can also note that the more structured an instance is, the more rugged the curve. This indicates that an algorithm with fixed parallelism is more sensitive to non-uniform connectivity distribution. This is also proved by the fact that the more random the instance, the lower is the average rank around the optimum, i.e., the less interleaved the ranks.

5 Conclusion and future work

In this work we have analyzed the connectivity distribution of structured SAT instances and observed their impact on the parallelization of local search. We observed that instances with differing connectivity distribution, but equal normalized average connectivity, have different optimal parallelism. Furthermore, by experimenting on instances generated by morphing, we have shown that the more similar the node degree distribution of two instances, the closer their optimal parallelism values. Finally, structure might introduce high variance on the optimal parallelism. This last property can be beneficial in applications of local search with varying parallelism, as noted in [9]. This last topic is subject of ongoing work. Future work is focused on the analysis of structural characteristics and graph properties of SAT and MAX-SAT problems and on the formal investigation of *criticality and parallelism* phenomenon with the goal of providing analytical relations between optimal parallelism and connectivity (and other problem instance characteristics).

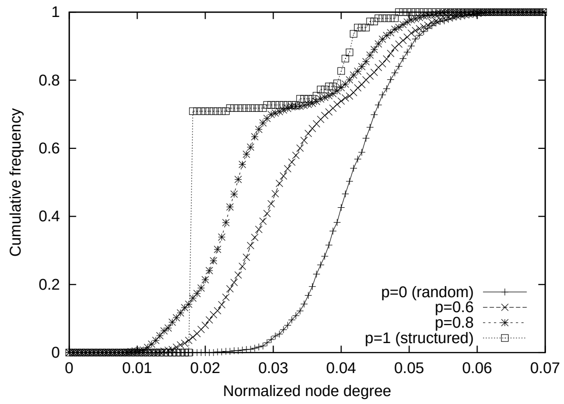

Figure 6: Cumulative frequency vs. normalized node degree for instances generated by morphing. $p \in [0, 1]$ controls the randomness/structure ratio: the higher p, the higher the amount of structure.

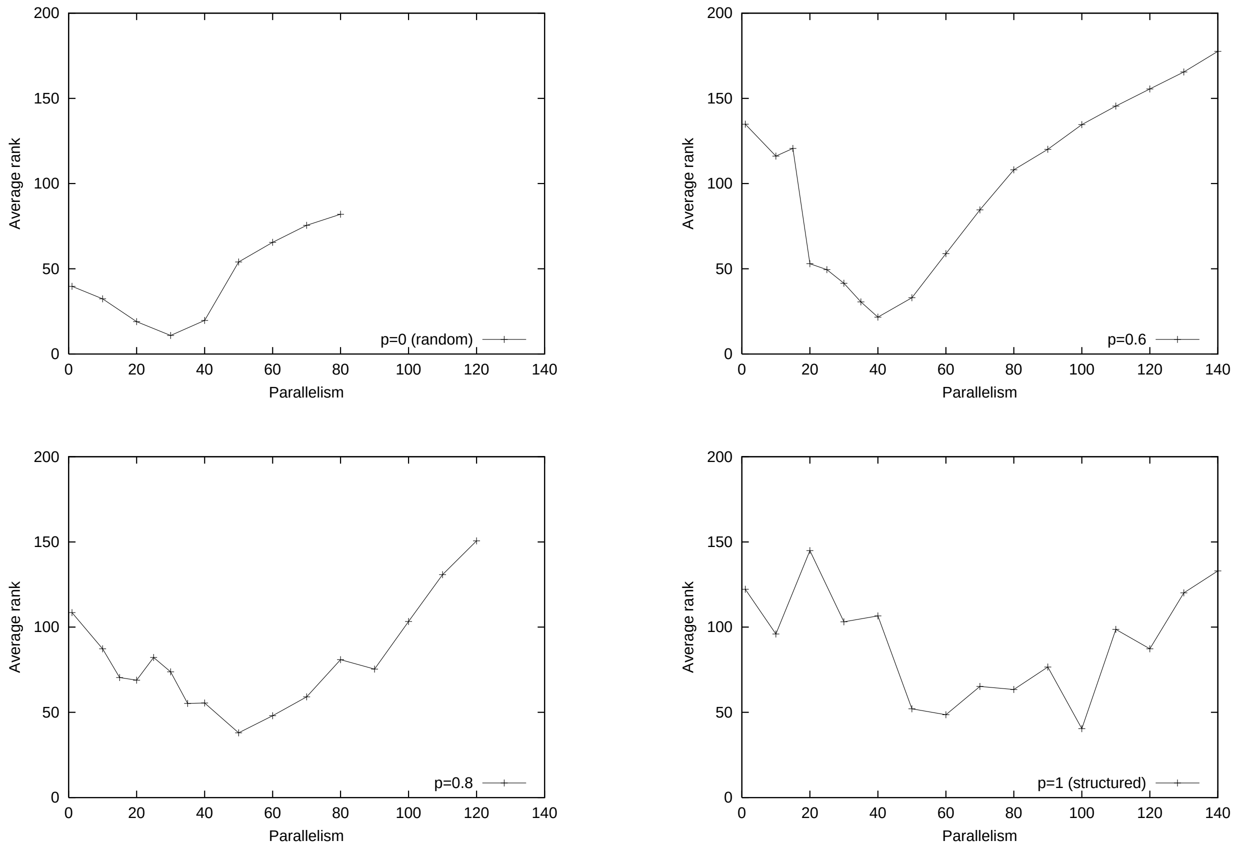

Figure 7: Average rank vs. parallelism for instances generated by morphing procedure controlled by p. The higher p, the higher the amount of structure. Results are averaged over 10 trials.

Acknowledgments

I would like to thank Michela Milano for her support and suggestions on this work.

References

[1] W. J. Conover. *Practical Nonparametric Statistics.* John Wiley & Sons, 3rd edition, 1999.

[2] I. P. Gent, H. H. Hoos, P. Prosser, and T. Walsh. Morphing: Combining structure and randomness. In *Proceedings of AAAI99*, pages 654–660, 1999.

[3] H. Hoos and T. Stützle. SATLIB: An online resource for research on SAT. In I. Gent, H. van Maaren, and T. Walsh, editors, *SAT2000: Highlights of Satisfiability Research in the Year 2000*, pages 283– 292. IOS Press, 2000.

[4] S.A. Kauffman. *At home in the universe.* Oxford Press, 1995.

[5] S.A. Kauffman and W. Macready. Technological evolution and adaptive organizations. *Complexity*, 26(2):26–43, March 1995.

[6] S. Kirkpartick, C.D. Gelatt, and M.P. Vecchi. Optimization by simulated annealing. *Science, 13 May 1983*, 220(4598):671–680, 1983.

[7] W.G. Macready and S.A. Kauffman A.G. Siapas. Criticality and parallelism in combinatorial optimization. *Science*, 271:56–59, January 1996.

[8] A. Roli. Criticality and parallelism in GSAT. *Electronic Notes in Discrete Mathematics*, 9, 2001.

[9] A. Roli and C. Blum. Critical Parallelization of Local Search for MAX–SAT. In F. Esposito, editor, *AI*IA2001: Advances in Artificial Intelligence*, volume 2175 of *Lecture Notes in Artificial Intelligence*, pages 147–158. Springer, 2001.

[10] B. Selman, H.J. Levesque, and D. Mitchell. A new method for solving hard satisfiability problems. In Paul Rosenbloom and Peter Szolovits, editors, *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 440–446, Menlo Park, California, 1992. American Association for Artificial Intelligence, AAAI Press.

[11] T. Walsh. Search on high degree graphs. In *Proceedings of IJCAI-2001*, 2001.