Università degli Studi di Bologna
DEIS

Criticality and parallelism in structured SAT instances

Andrea Roli

April 30, 2002

DEIS Technical Report no. DEIS-LIA-02-006 *LIA Series no. 59*

Criticality and parallelism in structured SAT instances

Andrea Roli

DEIS
Università degli Studi di Bologna
Viale Risorgimento 2, 40136 Bologna (Italia)
aroli@deis.unibo.it

April 30, 2002

Contents

1 Introduction

The phenomenon called *criticality and parallelism* has been observed in the context of local search algorithms applied to combinatorial optimization problems [9, 7, 6], 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[1] 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* [7, 6], each of the N variables is connected with exactly other K variables, therefore the average connectivity is K. For SAT problems, the average connectivity is defined as the average node degree of the constraint graph associated with the problem instance.

A phenomenon with analogous characteristics has also been recently observed in local search for SAT [16] and MAXSAT [17] problems. The considered instances are uniformly random generated and have three literals per clause. For these instances, it has been experimentally shown that there is an optimal number of simultaneous local moves (flips) which leads to a maximum in the average quality of solution found.

Recent studies [21] investigate the different distribution of node degree in random and structured problems and its impact on search. The authors show that real world problems frequently have a connectivity distribution strongly different from that of random instances and they point out how this topological characteristic may affect the search effectiveness.

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 concerning random instances [16, 17]. The paper is structured as follows: in Sect.2 we summarize the state of the art in the context of critical parallelization of local search, in particular with respect to SAT problems. In Sect.3 and Sect.4 we respectively analyze the connectivity distribution in random and structured SAT instances and show the results of parallel local search. Similarities and differences between critical parallel local search on random and structured instances are further examined in Sect.5, where, by means of a *morphing* procedure, a random instance is gradually changed to a structured one and the behavior of parallel local search is analyzed. In Sect.6 we propose a method to tune τ to the (near-)optimal value, based on short runs of the algorithms. Finally, after recalling related work in

[1] In the following we will use the expression *solution quality* referring to the value of the objective function; in case of a minimization problem, the lower the objective function value, the higher the quality.

Sect.7, we conclude with Sect.8 by outlining future work.

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. Moves are performed *synchronously* and, in general, they can be implemented both in sequential or parallel algorithms; in this work we have implemented parallel local search as a sequential algorithm. 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:

- 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 approach is given in [9], where Simulated Annealing [8] is applied on problems defined over binary variables. Instead of performing one random flip, every variable is flipped with probability p. Examples of the second approach can be found in [7, 6], where NK lattice models are optimized 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.
- When the decrease in performance for $\tau > \tau_{opt}$ is abrupt, the behavior is accompanied by a phase transition.

The same phenomenon has been discovered in parallel local search for SAT [16] and MAXSAT [17], where a parallel version of GSAT [19] has been applied (see Algorithm 1). Variables are divided in τ subsets and, for each subset, we flip the variable that, if flipped, decreases the greatest number of unsatisfied clauses. In short, we can say that, for every subset, the "best variable" according to the GSAT criterion is flipped. For both SAT and MAXSAT the objective function, to be minimized, is the number of unsatisfied clauses. Those experiments dealt with random

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 $bestsolution$

instances with three literals per clause (3-SAT/MAXSAT instances). For satisfiable SAT instances it has been experimentally shown that the best global performance (time, iterations, fraction of solved instances) is achieved with an optimal parallelism degree τ_{opt}. Furthermore, τ_{opt} is monotonically non increasing with the connectivity. For MAXSAT instances the results are almost the same as for SAT. Though, in this case, the transition between good and bad performance is less marked and the cost of search (time or number of iterations) is slightly increasing with τ.

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 node $v_i \in \mathcal{V}$ 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 appear in a same clause. In an instance with n variables, each node v_i, $i = 1, \ldots, n$, has a degree $q_i \in \{0, 1, \ldots, n-1\}$. For k-SAT problems, defined as conjunction of clauses with exactly k literal each, it

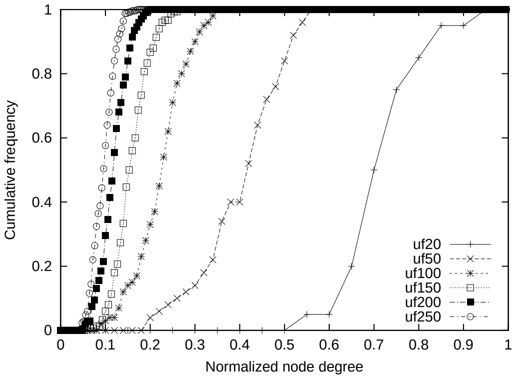


Figure 1. *Cumulative frequency vs. normalized node degree for Uniform Random 3-SAT instances in the threshold region.*

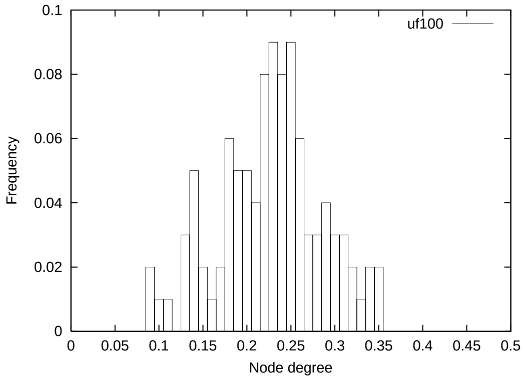


Figure 2. *Frequency against normalized connectivity in random 3-SAT instance uf100-01. The instance has an average normalized connectivity of* 0.229.

holds $k-1 \leq q_i \leq n-1$. We define the *average connectivity* of the instance as the average node degree of the corresponding graph, i.e., $q = \frac{1}{n}\sum_{i=1}^{n} q_i$. Moreover, 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 the following, we will use indifferently the expressions connectivity and node degree of an instance $\mathcal{I}$, being the second defined on the graph associated with $\mathcal{I}$. 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 [4]. 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 can assume that the graph corresponding to a random 3-SAT instance is a random graph $G_{n,p}$ [14], where n is the number of nodes and p is the probability that any pair of nodes are connected. For this kind of random graphs the distribution probability of connectivity follows a Poisson distribution, i.e.,

$$\text{prob}\{\text{a node is connected exactly to other } j \text{ nodes}\} = e^{-\lambda}\lambda^j/j!$$

where the parameter λ is the expected node degree, therefore, in our case, $\lambda = (n-1)\overline{q} = q$. For instance, in Fig.2 the frequency of a 3-SAT instance with 100 variables is plotted.

We analyzed the behavior of parallel GSAT (hereafter referred to as PGSAT) as a function of parallelism degree τ, i.e., the number of parallel flips. We stopped the algorithm when the maximum allowed number of moves without improvement

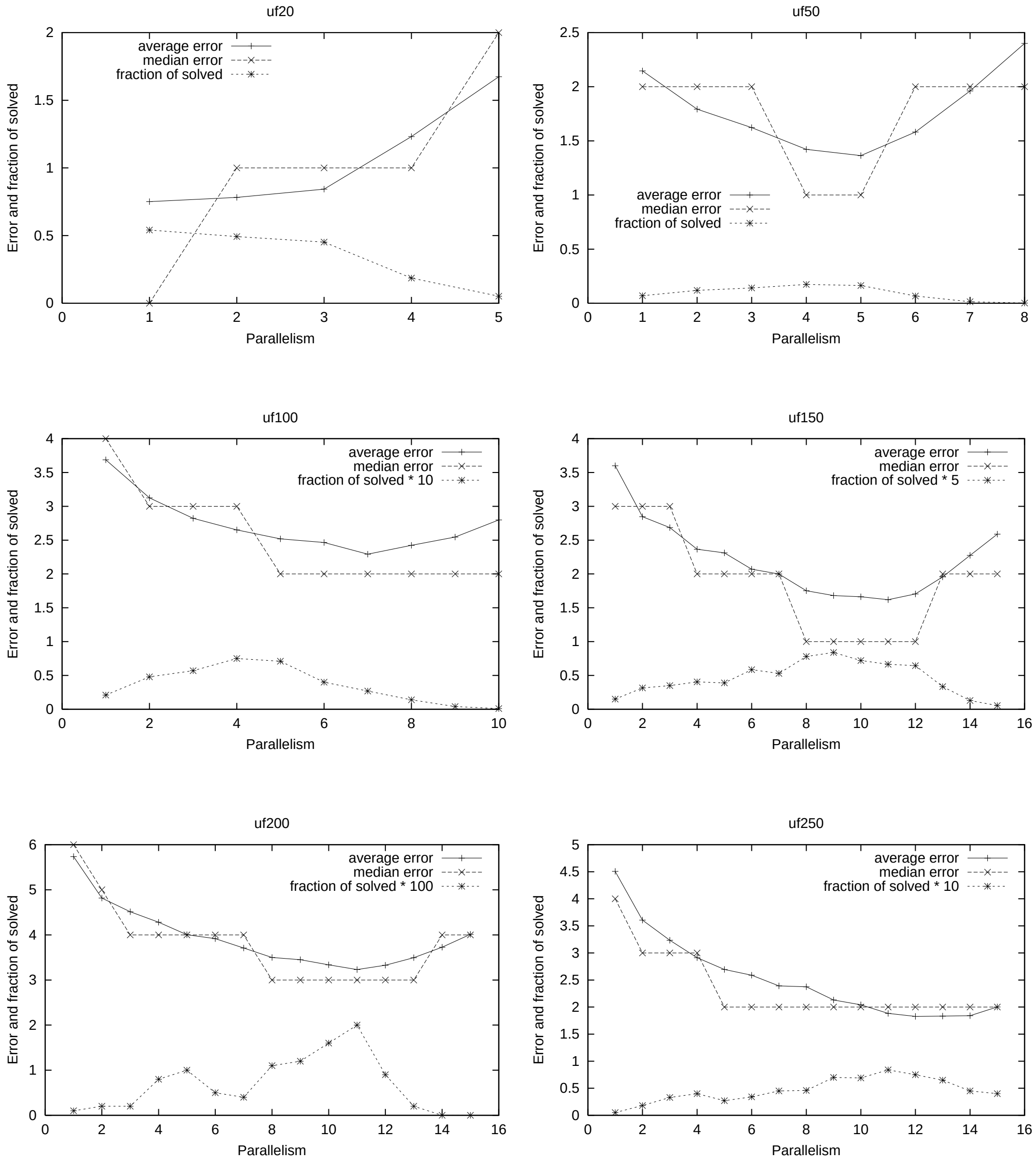


Figure 3. *Average and median error (number of unsatisfied clauses) and fraction of solved instances (rescaled when necessary) vs. parallelism* (τ) *for random 3-SAT instances with* $20, 50, 100, 150, 200$ *and* 250 *variables. Results are averaged over 1000 trials. We can observe that for every value of* n *the average (resp. median) number of unsatisfied clauses returned by the algorithm has a minimum corresponding to a* $\tau_{opt}(n)$. *If we consider the average, the results are:* $\tau_{opt}(20) = 1$, $\tau_{opt}(50) = 5$, $\tau_{opt}(100) = 7$, $\tau_{opt}(150) = 11$, $\tau_{opt}(200) = 11$, $\tau_{opt}(250) = 12$.

of the best solution found has been reached. This cutoff value has been set to n. The variables are divided in τ subsets at random, before each iteration. All subsets have equal cardinality (n/τ), except for one which contains $n/\tau + n \bmod \tau$ variables. Results, averaged over 1000 trials, are shown in Fig.3. We can observe that for every value of n the average (resp. median) number of unsatisfied clauses returned by the algorithm has a minimum corresponding to a $\tau_{opt}(n)$. If we consider the average, the results are: $\tau_{opt}(20) = 1$, $\tau_{opt}(50) = 5$, $\tau_{opt}(100) = 7$, $\tau_{opt}(150) = 11$, $\tau_{opt}(200) = 11$, $\tau_{opt}(250) = 12$. For the hit rate (i.e., the number of solved instances) results are analogous. We can note that the correspondence between the minimum error and the maximum hit rate is fairly good, though it seems that the maximal hit rate is reached just before reaching the minimum error. An important point to consider is that, at least in the context of random 3-SAT instances, both for maximal hit rate and minimal error, the following relation holds: $n_1 \geq n_2 \Rightarrow \tau_{opt}(n_1) \geq \tau_{opt}(n_2)$. As noted in [16], this relation derives from the fact that τ_{opt} is negatively correlated with q and it is not just an effect of size scaling. Indeed, at fixed ratio between clauses and variables (in the threshold region $m/n \approx 4.3$), q decreases as n increases. These results are in accordance with previous ones on random forced 3-SAT and MAXSAT [16, 17]. In the next section we will analyze structured instances, compare them with random ones and show PGSAT experimental results.

4 Structured instances

In this section we first analyze the node degree distribution of graphs associated with structured SAT instances. Then we show and discuss results of PGSAT, run with different values of τ, on two representative structured SAT 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.4 shows the curve of cumulative frequency for structured SAT instances taken from SATLIB, produced by encoding a Blocks World Planning Problem (*huge*), a Logistics Planning Problem (*logistics-a*) and Inductive Inference (*ii16a1*). The plotted curves show apparent differences with those of random SAT problems. They are not as 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 instance *ii16a1* is the most peculiar and differs the most from that of random instances. It has 1650 variables and a normalized average connectivity $\overline{q}_{ii16a1} = 0.0239$. Its cumulative frequency is shown in Fig.5, along with the cumulative frequency of a random 3-SAT instance of the same size and normalized connectivity (instance *3sat1650*). Fig.6 and Fig.7 plot the respective frequency of node degree. We can note that the node degree frequency of the structured instance

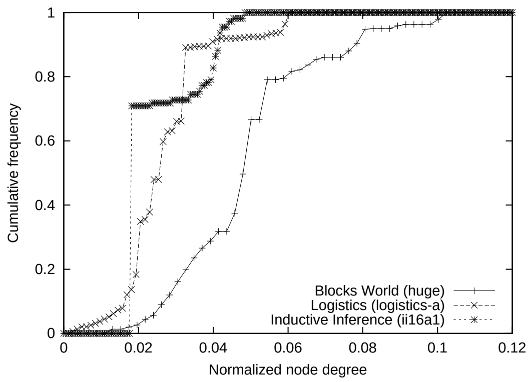


Figure 4. *Cumulative frequency vs. normalized node degree in structured instances.*

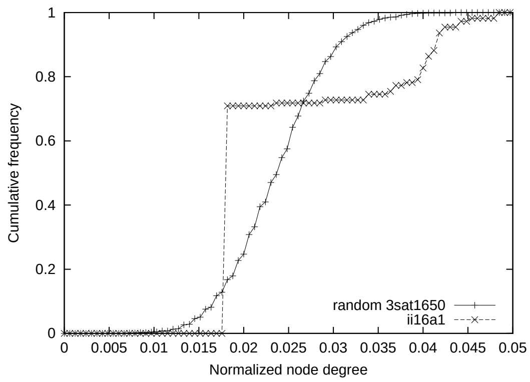
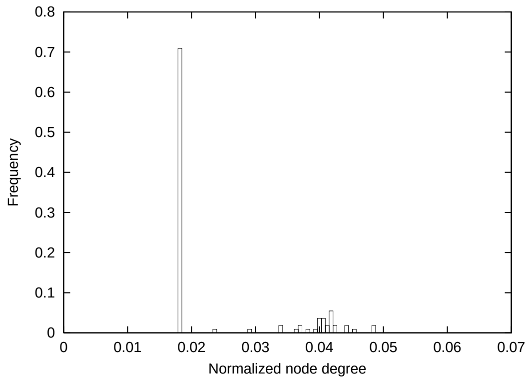


Figure 5. *Cumulative frequency vs. normalized node degree in a structured instance (ii16a1) and in a random 3-SAT instance 3sat1650 of same size ($n = 1650$) and connectivity ($\overline{q} \approx 0.0239$).*

is highly asymmetric and has a peak close to 0.018, corresponding to the large gap in the cumulative frequency. Therefore, *ii16a1* has a very large number of nodes with lower connectivity than the average. Conversely, the node degree frequency of the random instance is regular (it approximately fits the Poisson distribution with high mean) and the highest peak in frequency is very close to the mean.

Fig.8 and Fig.9 show the average and median error (number of unsatisfied clauses) respectively on *ii16a1* and *3sat1650* for PGSAT with different values of τ. Results are averaged over 500 trials. We first observe that also for the structured instance there exists an optimal value of τ. Nevertheless, despite the fact that the two instances have the same average connectivity, the optimal parallelism is higher for *ii16a1* than for *3sat1650*. We conjecture that the high number of nodes with low degree present in the instance *ii16a1* is the cause of higher optimal parallelism.

We performed the same kind of experiments on the *logistics-a* instance, by comparing it with a random 3-SAT instance with the same size (828 variables) and normalized average connectivity $\overline{q} = 0.0275$ (instance *3sat828*). Fig.10 and Fig.11 show the respective node degree frequency. Note that, while the distribution of the instance *3sat828* approximately follows a Poisson distribution, the distribution of *logistics-a* is not regular and has a high peak at normalized node degree 0.0326. The results of PGSAT performance with respect to τ are plotted in Fig.12 and Fig.13. The results are analogous, but dual, to the previous ones on *ii16a1*. We can observe that also for this structured instance there is a value of τ leading to a minimum average error, but in this case the optimal parallelism for *logistics-a* is lower than that of the random instance. This difference may be explained by observing that



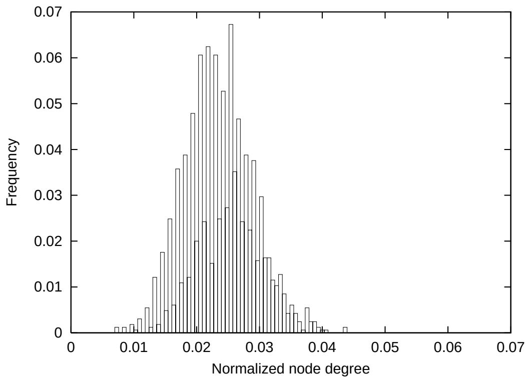
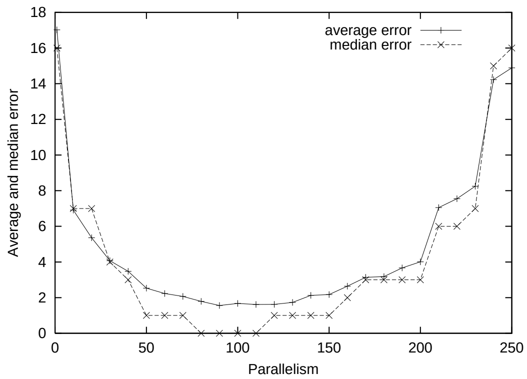


Figure 6. *Frequency vs. normalized node degree in the instance ii16a1.*

Figure 7. *Frequency vs. normalized node degree in the instance 3sat1650.*



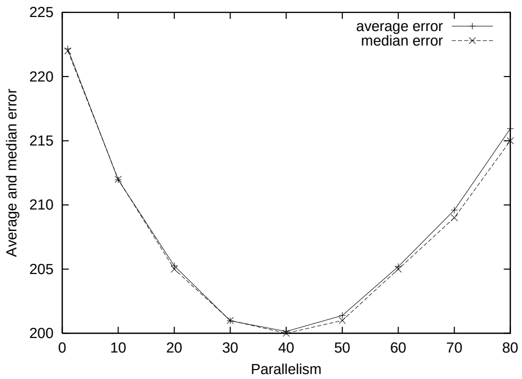


Figure 8. *Average and median error against τ for PGSAT on the instance ii16a1.*

Figure 9. *Average and median error against τ for PGSAT on the instance 3sat1650.*

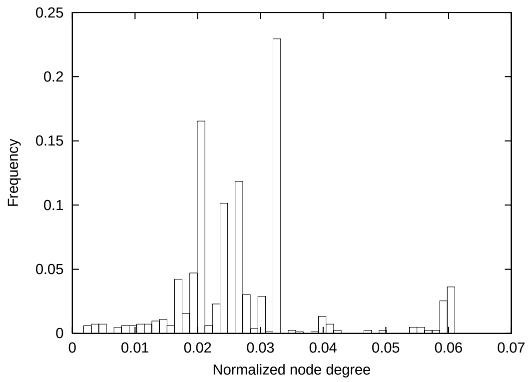


Figure 10. *Frequency vs. normalized node degree in the instance logistics-a.*

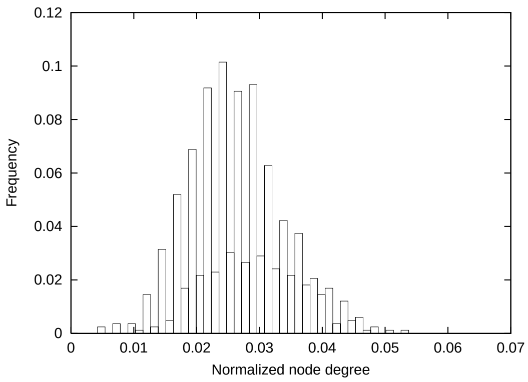


Figure 11. *Frequency vs. normalized node degree in the instance 3sat828.*

the highest peak in *logistics-a* node degree frequency corresponds to a node degree higher than the average value, which characterizes the random instance.

We can conclude this section by considering that the results obtained clearly show that an optimal parallelism value exists also for structured instances. Nevertheless, not surprisingly, this value is strongly affected by the asymmetric frequency of node degree. In particular, we observe that the highest peaks location seem the most relevant characteristic influencing the optimal parallelism.

5 Morphing from random to structure

In Sect.3 we have seen that the average connectivity affects the optimal parallelism for random instances. The results of Sect.4 show that for structured instances, characterized by asymmetric frequency distribution, the optimal parallelism is mostly affected by the highest peaks of node degree distribution. To investigate more deeply this difference, we generated instances with a controlled amount of *structure*, by means of a technique called *morphing* [3]. This methods permits to generate instances gradually morphing from source to destination instance by varying a parameter $p \in [0, 1]$. The lower 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 (n) and clauses (m). A new SAT instance is generated by selecting each of the m clauses either from the source or the destination. The clause is chosen from the destination instance with probability p. We generated a satisfiable random 3-SAT instance with 1650 variables and 19368 clauses (*3sat1650_large*), the same number as *ii16a1*. With $p = 1$ we obtain *ii16a1* and with $p = 0$ *3sat1650_large*. Since p controls the number of clauses belonging to the structured instance *ii16a1*,

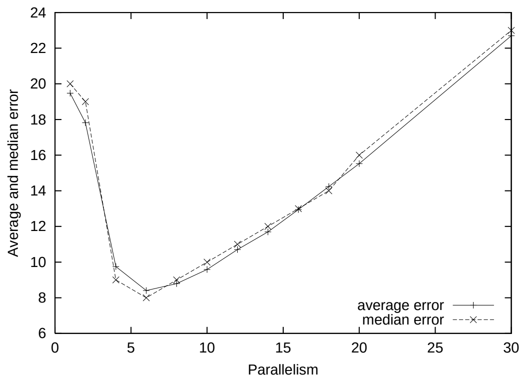


Figure 12. *Average and median error against τ for PGSAT on the instance logistics-a.*

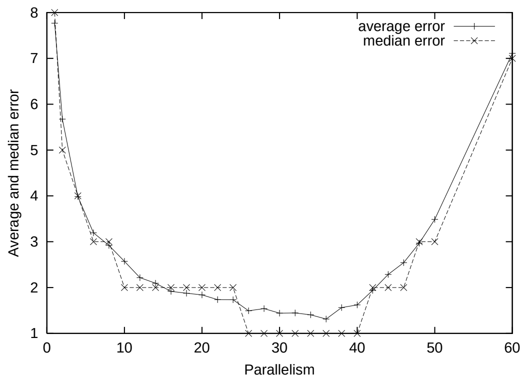


Figure 13. *Average and median error against τ for PGSAT on the instance 3sat828.*

it also measures the amount of structure in the generated instance. Fig.14 shows the node degree cumulative frequency of source (random), destination (structure) and instances generated by the morphing method. The node degree frequency of the considered instances is plotted in Fig.15. The results of 500 trials of PGSAT for different values of parallelism are shown in Fig.16. Observe that the optimal parallelism increases with p, therefore we can conjecture that the more similar the node degree frequency distributions of two instances, the closer should be their optimal parallelism degrees.

We can conclude by asserting two points. First, regardless of the instance type, the average connectivity is a rough, yet indicative, parameter for the optimal parallelism. Second, we have again found experimental evidence that in structured instances the highest peaks in the node degree distribution have a strong impact on the optimal parallelism value.

6 Tuning of τ

In the previous sections we have shown that also structured instances exhibit the behavior found in the criticality and parallelism phenomenon for random SAT problems. Beside the interest about the phenomenon itself, the parallelization can be used to improve local search, namely GSAT. The intuition behind the effectiveness of parallel local moves is that multi-flip moves help to escape from local minima and reach faster search space regions with low objective value. This conjecture is supported by experimental results in [7, 6] and by preliminary tests we performed on large MAXSAT instances. Parallel flips may prevent the search to get trapped in

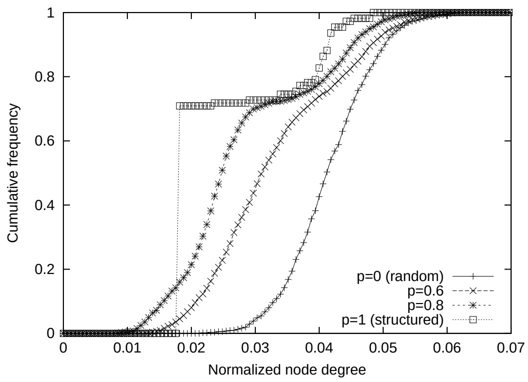


Figure 14. *Cumulative frequency vs. normalized node degree for instances generated by morphing.* $p \in [0,1]$ *controls the randomness/structure ratio: the higher* p*, the higher the amount of structure.*

local minima in the same way as random noise when added to local search [10, 18, 1]. The relations between parallel moves and noise are beyond the scope of this paper. However, it is important to observe that the optimal number of parallel flips and the optimal noise level are related to properties of the search landscape, like the ruggedness, the distribution of local minima and the size of their basins of attraction [11, 5]. Since up to now theoretical results are still missing to compute the optimal parallelism value for a given instance, we need an empirical method to tune τ as close as possible to τ_{opt}. The easiest way is to run PGSAT for $\tau = 1, 2, \ldots$, using a small cutoff value and stop as soon as a minimum average error has been found. Nevertheless, this method is, in general, computationally expensive, as it might require several runs before finding τ_{opt}. We developed another method which requires just one short run for every candidate for τ_{opt}. This method is based on the observation of solution improvements PGSAT achieves during the execution. Fig.17 shows typical results obtained with short runs of PGSAT on the *logistics-a*[2], for which $\tau_{opt} \approx 6$. For every run length $(1, 10, 20, 50)$ we stored the objective function value (number of unsatisfied clauses) reached by PGSAT with different values of τ; we call this value $\epsilon(\tau)$. For each point (x, y) in the plot, the coordinate y is given by $\epsilon(x-1) - \epsilon(x)$. For example, let us consider the 1st iteration curve. After one iteration, the difference of the errors returned by PGSAT with $\tau = 1$ and $\tau = 2$ is 12, i.e., $\epsilon(1) - \epsilon(2) = 12$. All the runs have the same initial state and the algorithm uses the same random number generator seed. Except for the plot of the 1st iteration, we note that the curve decreases approximately linearly up to a point where the slope decreases quite abruptly. This point corresponds, with good ap-

[2] We have chosen *logistics-a* just as a representative example of the experiments we made on the considered instances.

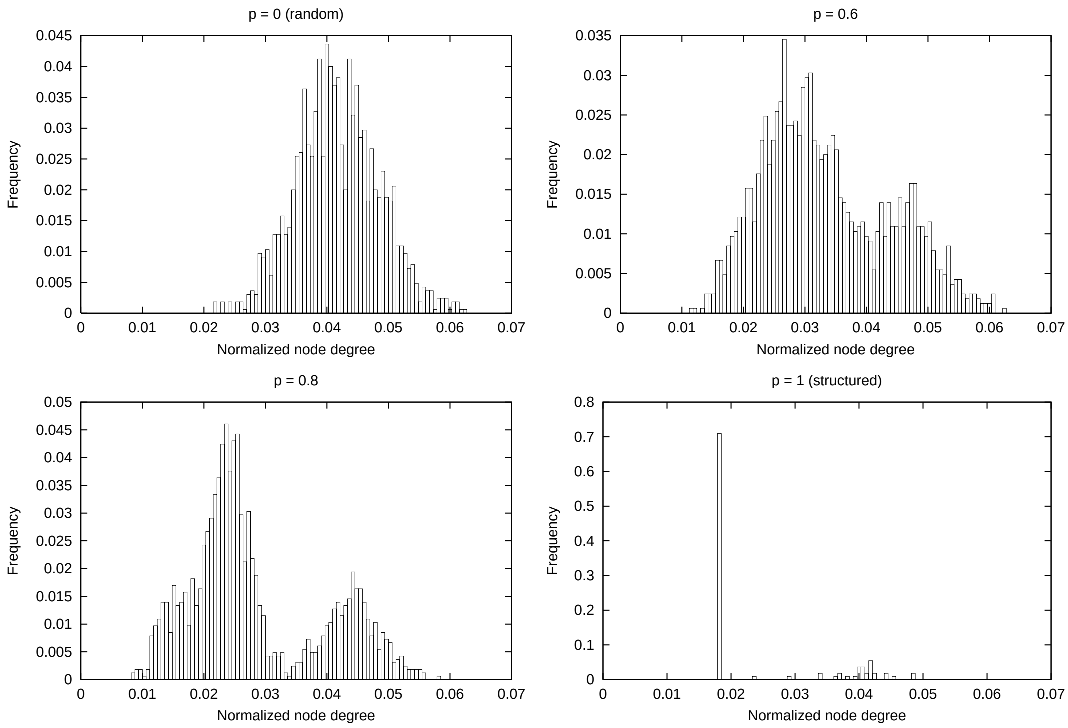


Figure 15. *Frequency against normalized node degree for instances generated by morphing from a random instance (3sat1650_large) to a structured one (ii16a1).*

proximation, to τ_{opt}. This behavior can be presumably explained by the fact that, at τ_{opt}, the distance between a state and its successor, reached after a multi-flip move, enables local search to achieve the optimal trade-off between intensification and diversification[3]. This result yields to the definition of a simple procedure to tune τ: run PGSAT for a very small number of iterations for different values of τ and set it at the value corresponding to the knee in the curve. Up to now, we have determined the (near-)optimal value of τ by directly observing the plotted data, but the development of an automatic procedure is subject of current work.

7 Related Work

This work is inspired by work of Kauffman et al. on criticality and parallelism [9, 7, 6] and Monte Carlo algorithms for NK-models [2]. Even though not with the same objective, other works deal with multiple flips in local search for SAT and MAXSAT

[3] We refer to the informal definition of *intensification* as the greedy exploitation of search history and *diversification* as the exploration of the search space.

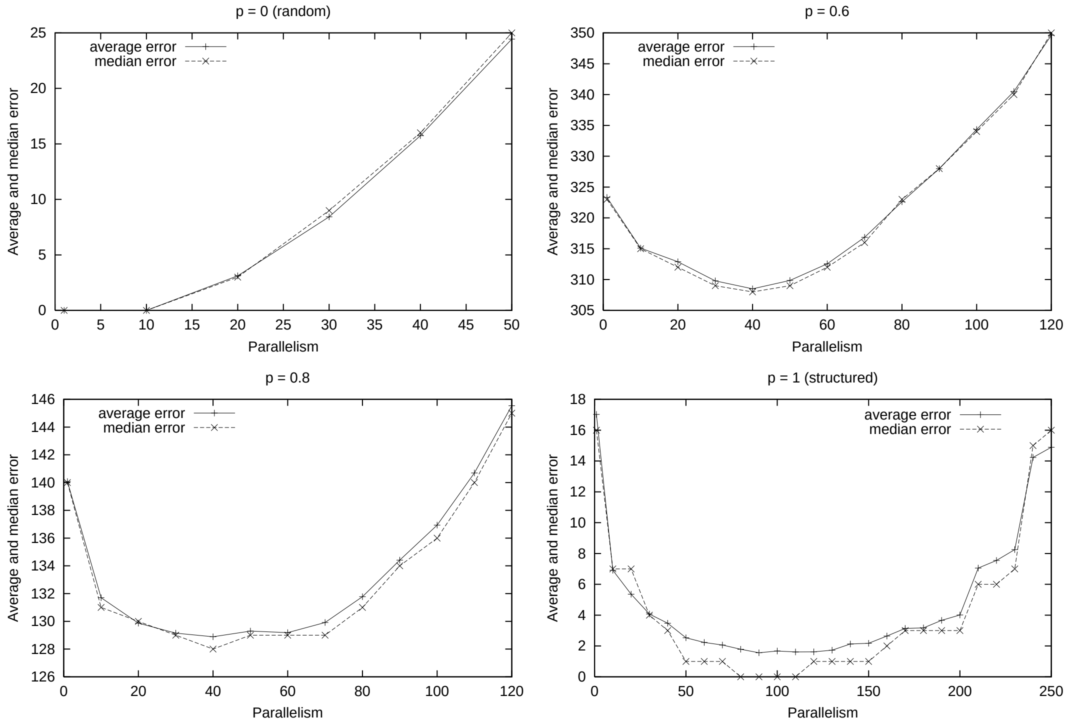


Figure 16. *Average and median error of PGSAT run with different values of τ on instances generated by morphing from a random instance (3sat1650_large) to a structured one (ii16a1).*

problems. Yagiura and Ibaraki [22] present a computational study of r-flips neighborhoods ($r = 2, 3$). Parkes [15] considers the parallelization of WalkSAT [10, 18] for 3-SAT instances in the underconstrained region and observes that parallel flips does not degrade the local search, indirectly confirming our results. Indeed, in the underconstrained region the connectivity is low, therefore the optimal parallelism is high. Finally, Strohmaier [20] discusses the implementation of GSAT on a multi-flip neural network and Milano and Roli [13, 12] present a general method for tackling SAT with boolean networks, explicitly considering the possibility of multi-flips local search algorithms.

8 Conclusion and future work

In this work we have compared the connectivity distribution of random and structured SAT instances and observed its impact on the parallelization of local search. We first showed that also in the case of structured instances there is a critical value

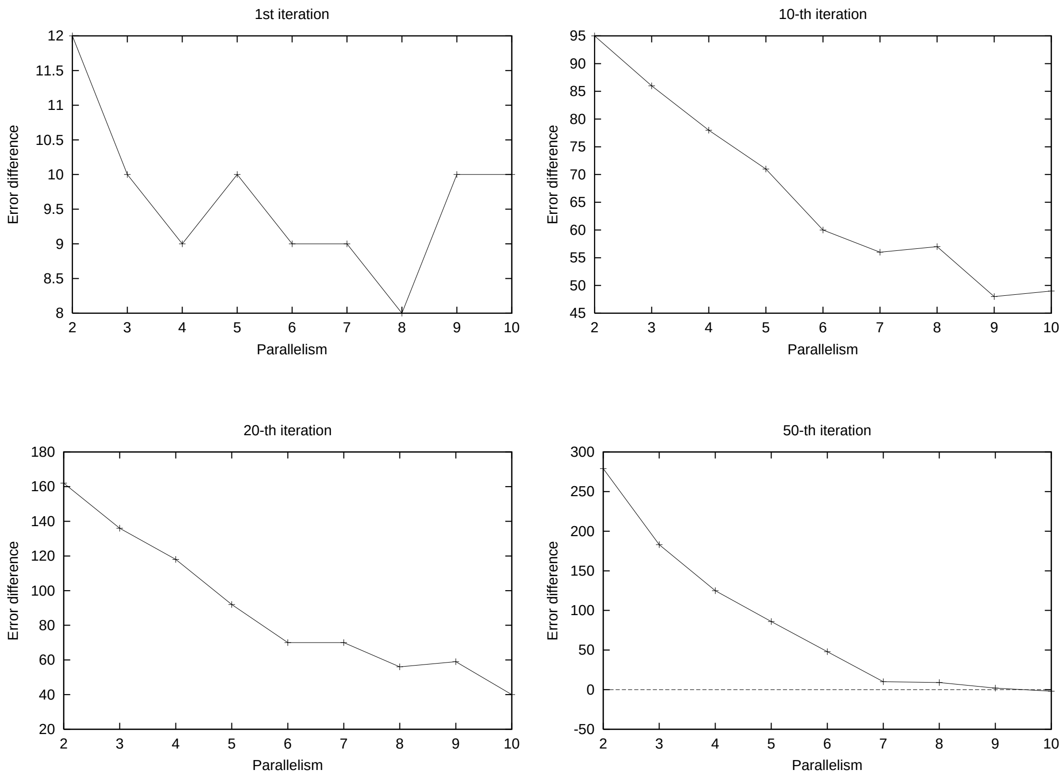


Figure 17. *Difference of the error (number of unsatisfied clauses) reached after* $1, 10, 20, 50$ *iterations between successive values of parallelism. For each point* (τ, y) *in the plot, the coordinate* y *is given by* $\epsilon(\tau - 1) - \epsilon(\tau)$*, where* $\epsilon(\tau)$ *is the number of unsatisfied clauses returned by PGSAT with parallelism* τ*.*

of parallelism τ_{opt} that optimizes the average solution quality. Secondly, we observed that in structured instances τ_{opt} is strongly affected by the highest peaks in the node degree distribution, thus structured instances can be tackled by analyzing their node degree frequency. An empirical procedure to tune τ has been presented. This procedure is based on the observation that the optimal parallelism enables local search to maximize the balance between intensification and diversification. The definition of an automatic procedure to tune τ is subject of ongoing work, along with the use of parallelization as a mechanism to adapt the balance between intensification and diversification. Preliminary experiments on MAXSAT with dynamic τ have shown very promising results.

We are currently exploring different ways of parallelizing GSAT, for example by probabilistic flips, such that each variable flips with a probability which depends on the increase of number of satisfied clauses. Moreover, we are studying the relations between SAT/MAXSAT instance connectivity and properties of the search landscape and we are configuring experiments to check whether a phase transition exists also in the case of PGSAT on SAT/MAXSAT problems.

There are several open issues to explore, like the exploitation of graph properties to define the subsets of strongly connected variables (instead of dividing variables at random), the investigation of the relation between criticality and parallelism and the introduction of noise in local search, the analytical study of connectivity and optimal parallelism.

Acknowledgments

I thank Michela Milano for her support, helpful discussions and suggestions.

References

[1] I. Charon and O. Hudry. The noising methods: A generalization of some metaheuristics. *European Journal of Operational Research*, pages 135:86–101, 2001.

[2] F.M. Dittes. Optimization on rugged landscapes: A new general purpose Monte Carlo approach. *Physical Review Letters*, 76(25):4651–4655, June 1996.

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

[4] 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.

[5] T. Jones. *Evolutionary Algorithms, Fitness Landscapes and Search.* PhD thesis, Univ. of New Mexico, Albuquerque, NM, 1995.

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

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

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

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

[10] D. McAllester, B. Selman, and H. Kautz. Evidence for invariants in local search. In *Proceedings of the 14th National Conference on Artificial Intelligence and 9th Innovative Applications of Artificial Intelligence Conference (AAAI-97/IAAI-97)*, pages 321–326, Menlo Park, July 27–31 1997. AAAI Press.

[11] P. Merz and B. Freisleben. Fitness landscapes and memetic algorithm design. In David Corne, Marco Dorigo, and Fred Glover, editors, *New Ideas in Optimization*, Advanced topics in computer science series. McGraw-Hill, 1999.

[12] M. Milano and A. Roli. Boolean networks-based algorithms for the satisfiability problem. In *Electronic Proceedings of the Workshop on Experimental Analysis of Algorithms for Artificial Intelligence*, 1999. AI*IA working group on Knowledge Representation and Reasoning.

[13] M. Milano and A. Roli. Solving the satisfiability problem through boolean networks. In *Lecture Notes in Artificial Intelligence - AI*IA99: Advances in Artificial Intelligence*, volume 1792, pages 72–83. Springer, 2000.

[14] M.E.J. Newman, S.H. Strogatz, and D.J. Watts. Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E*, 64, 2001.

[15] A.J. Parkes. Distributed local search, phase transitions, and polylog time. In *Workshop "Stochastic Search Algorithms" at IJCAI-01*, 2001.

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

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

[18] B. Selman, H.A. Kautz, and B. Cohen. Noise strategies for local search. In *Proc. 12th National Conference on Artificial Intelligence, AAAI'94, Seattle/WA, USA*, pages 337–343, 1994.

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

[20] A. Strohmaier. Multi-flip networks: Extending symmetric networks to real parallelism. Technical report aida-96-08, FG Intellektik, TU Darmstadt, 1996.

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

[22] M. Yagiura and T.Ibaraki. Analyses on the 2 and 3-flip neighborhoods for the MAXSAT. *Journal of Combinatorial Optimization*, 3:95–114, 1999.