Symmetry-breaking and local search

Andrea Roli
Dipartimento di Scienze
Università degli Studi “G.D’Annunzio” Chieti-Pescara
Pescara (Italia)
a.roli@unich.it

SOMMARIO/*ABSTRACT*

Recenti risultati sperimentali riguardanti il comportamento di algoritmi di ricerca locale hanno evidenziato un fenomeno particolarmente interessante: l'uso di vincoli per eliminare le simmetrie del problema ha un effetto negativo sulle prestazioni dell'algoritmo. Il motivo principale di questo effetto risiede nella perturbazione dello spazio di ricerca operata dai vincoli inseriti nel modello che rende più difficile il raggiungimento delle soluzioni. Tuttavia, non è ancora stata formalizzata una teoria che permetta di spiegare il fenomeno. In questo lavoro sosteniamo l'ipotesi che l'effetto negativo sia in primo luogo da attribuirsi al fatto che i vincoli di eliminazione delle simmetrie riducono i bacini di attrazione delle soluzioni ottime. L'ipotesi è supportata da un'analisi sperimentale eseguita sul problema LABS (Low Autocorrelation Binary Sequence).

A recently observed phenomenon in local search algorithms behavior is the negative effect of symmetry breaking constraints on their performance. The reasons for this are poorly understood and they are likely to be found in the structure of the search space, mainly in the reachability of solutions. In this work, we attempt to shed light on the phenomenon by testing the following conjecture: Symmetry-breaking constraints can harm local search because they often reduce the relative size of the basins of attraction of global optima. We provide experimental support to the conjecture by studying local search performance vs. different modelling of the Low Autocorrelation Binary Sequence problem.

Keywords: Local search, Symmetry breaking, Low autocorrelation binary sequence

1 Introduction

The effects of combining search and modelling techniques can be complex and unpredictable, so guidelines are very important for the design and development of effective and robust solvers. This is an issue not only for exact solvers, but also for approximate techniques, such as local search.

A recently observed phenomenon is the negative effect of symmetry breaking constraints on local search performance. Symmetry-breaking (SB) has been proved to be very effective when combined with complete solvers [4, 12]. This can be explained by observing that symmetry-breaking constraints considerably reduce the search space. Nevertheless, the use of symmetry-breaking constraints seem to have opposite effect on local search-based solvers, despite the search space reduction. In [9, 10, 11] some examples of this phenomenon are reported and discussed. When the problem is modeled with symmetry-breaking constraints, the search cost[1] is higher than the one corresponding to the model with symmetries. In [10], and also in [5] in a related context, it is suggested to use models that maximize the number of symmetries, contrarily to the case of complete solvers.

The reasons for this are poorly understood and they are likely to be found in the structure of the search space, mainly in the reachability of solutions. With this work, we attempt to shed light on the phenomenon by testing the following conjecture, composed of two statements: *(i)* Symmetry-breaking constraints can reduce the relative size of the basins of attraction of global optima (basins are defined upon the application of a best improvement local search). Since most local search strategies incorporate a greedy component like best improvement, they should be affected by this reduction of global optima basins of attraction; therefore, *(ii)* we should observe a positive correlation

[1]Runtime and number of moves

between size of global optima basins of attraction and algorithm effectiveness. Moreover, we should observe that complex local search heuristics can dampen this negative effect, as they are equipped with advanced diversification strategies. This conjecture can provide one of the main reasons for the negative effect of symmetry-breaking constraints on local search.

SB constraints can be treated by local search techniques in two ways: They can be either enforced (i.e., they are posted as hard constraints), or they can be penalized in the objective function if violated. The first approach, that can be viewed as a special case of the second when penalties are infinite, is particularly interesting and important to study since it represents the extreme case of modelling with symmetry-breaking constraints and it can be studied independently of the actual penalty values.[2].

Our investigation is focused on *why*, despite the search space reduction, local search-based techniques seem to be penalized by the introduction of symmetry-breaking constraints. Simplifying, the reasons why this happens can be either that *(i)* symmetry-breaking constraints modify the topology of the search landscape (e.g., by making the space not connected), or that *(ii)* they reduce the basins of attraction of global optima, or both. For testing these hypotheses, we introduce a model to study the effect of symmetry-breaking constraints on the search landscape explored by local search.

In this work, we report and discuss a series of experiments on the *Low auto-correlation binary sequence problem* that support our conjecture. We first briefly define a model of the search space explored by a local search algorithm in Section 2. In Section 3, we report results concerning the search space characteristics of small size instances of LABS. Then, in Section 4, we compare the performance of four local search algorithms on both the original model and the model in which some symmetric solutions have been cut and we show that complex local search heuristics can temper the negative effect of symmetry-breaking constraints.

2 The search graph and its main characteristics

The aim of this section is to provide a model of the phenomenon we want to study. First of all, we will define the execution of a local search algorithm as a 'walk' along a graph. This enables us to outline important features of the search space and to relate them with the symmetry-breaking constraints. Then, we will introduce the kind of symmetries and the class of instances we consider.

[2]However, results obtained in the 'hard' case, provide important guidelines for understanding the algorithm behavior also in the case with penalties.

Local search can be used to attack both constraint satisfaction problems and combinatorial optimization problems. The main operative difference is that, in the former case, the algorithm succeeds only if it finds a feasible solution to the original problem, which means that the local search algorithm must find a global optimum in the search landscape. In the latter case, the goal is usually to find, at least, a near-optimal solution. In the following, global optima can be considered either solutions better than any other solution or solutions belonging to the set of 'acceptable solutions', i.e., solutions considered good to the application at hand. The number of local minima can be taken as an indicator of the *ruggedness* of the search space explored by local search. In turn, it is usually recognized that the more rugged a search space is, the poorer is local search performance. Nevertheless, this parameterization of the search space might be sometimes insufficiently explicative or predictive. In this section, we provide a simple model of the search space which not only considers local and global optima, but also their *basins of attraction*, i.e., the set of states from which the optima can be reached.

The local search process can be viewed as an exploration of a landscape aimed at finding an optimal solution, or a *good* solution, i.e., a solution with a quality above a given threshold. We define the search space explored by a local search algorithm as a *search graph*. The topological properties of such a graph are defined upon the neighborhood structure, that generate the *neighborhood graph*.

2.1 Neighborhood and search graphs

A *Neighborhood Graph* (NG), is defined by a triple $\langle S, \mathcal{N}, f\rangle$, where:

- S is the set of feasible states;
- $\mathcal{N}$ is the neighborhood function $\mathcal{N}: S \rightarrow 2^S$ that defines the neighborhood structure, by assigning to every $s \in S$ a set of states $\mathcal{N}(s) \subseteq S$.
- f is the objective function $f\colon S \rightarrow \mathbb{R}^+$

The neighborhood graph can be interpreted as a graph (see Figure 1) in which nodes are states (labeled with their objective value) and arcs represent the neighborhood relation between states. The neighborhood function $\mathcal{N}$ implicitly defines an *operator* φ which takes a state s_1 and transforms it into another state $s_2 \in \mathcal{N}(s_1)$. Conversely, given an operator φ, it is possible to define a neighborhood of a variable $s_1 \in S$:

$$\mathcal{N}_\varphi(s_1) = \{s_2 \in S \setminus \{s_1\} \mid s_2 \text{ can be obtained by one application of } \varphi \text{ on } s_1\}$$

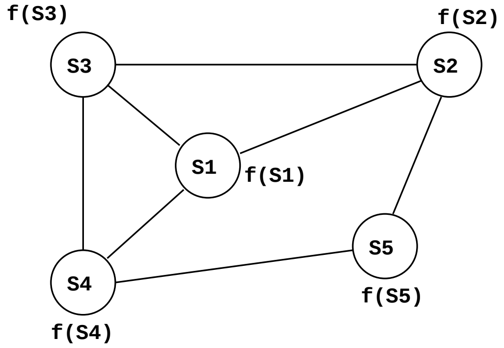

Figure 1: Example of undirected graph representing a neighborhood graph (fitness landscape). Each node is associated with a solution s_i and its corresponding objective value $f(s_i)$. Arcs represent transition between states by means of φ. Undirected arcs correspond to symmetric neighborhood structure.

In most cases, the operator is *symmetric*: if s_1 is a neighbor of s_2 then s_2 is a neighbor of s_1. In a graph representation (like the one depicted in Figure 1) undirected arcs represent symmetric neighborhood structures. A desirable property of the neighborhood structure is to allow a path from every pair of nodes (i.e., the neighborhood is strongly optimally connected) or at least from any node to an optimum (i.e., the neighborhood is weakly optimally connected). Nevertheless, there are some exceptions of effective neighborhood structures which do not enjoy this property [8].

The exploration process of local search methods can be seen as the evolution in (discrete) time of a discrete dynamical system [1]. The algorithm starts from an initial state and describes a trajectory in the state space, that is defined by the neighborhood graph. The system dynamics depends on the strategy used; simple algorithms generate a trajectory composed of two parts: a *transient* phase followed by an *attractor* (a fixed point, a cycle or a complex attractor). Algorithms with advanced strategies generate more complex trajectories which can not be subdivided in those two phases.

It is useful to define the search as a walk on the neighborhood graph. In general, the choice of the next state is a function of the search history (the sequence of the previously visited states) and the iteration step. Formally: $s(t+1) = \phi(\langle s(0), s(1), \ldots, s(t)\rangle, t)$ where the function ϕ is defined on the basis of the search strategy. ϕ could also depend on some parameters and can be either deterministic or stochastic.

For instance, let us consider a deterministic version of the Iterative Improvement local search. The trajectory starts from a point s_0, exhaustively explores its neighborhood, picks the neighboring state s' with minimal objective function value[3] and, if s' is better than s_0, it moves from s_0 to s'. Then this process is repeated, until a minimum $\hat{s}$ (either local or global) is found. The trajectory does not move further and we say that the system has reached a fixed point ($\hat{s}$). The set of points from which $\hat{s}$ can be reached is the basin of attraction of $\hat{s}$. Note that, especially in the case of deterministic local search algorithms, not every initial state is guaranteed to reach a global optimum.

Once we have introduced also the search strategy, the edges of the neighborhood graph can be oriented and labeled with transition probabilities (whenever it is possible to evaluate them). This will lead to the definition of concepts such as basins of attraction, state reachability and graph navigation. In the following, this resulting graph will be referred to as the *search graph*.

2.2 Basins of attraction

The concept of *basin of attraction* (BOA) has been introduced in the context of dynamical systems, in which it is defined referring to an *attractor*. In our model of local search, we will use the concept of basin of attraction of any node of the search graph. For the purposes of this paper we only consider the case of deterministic systems, even if it is possible to extend the definition to stochastic ones.[4]

Definition Given a deterministic algorithm $\mathcal{A}$, the basin of attraction $\mathcal{B}(\mathcal{A}|s)$ of a point s, is defined as the set of states that, taken as initial states, give origin to trajectories that include point s. The cardinality of a basin of attraction represents its size (in this context, we always deal with finite spaces).

Given the set S_{opt} of the global optima, the union of the BOA of global optima $I_{opt} = \bigcup_{i \in S_{opt}} \mathcal{B}(\mathcal{A}|i)$ represents the set of desirable initial states of the search. Indeed, a search starting from $s \in I_{opt}$ will eventually find an optimal solution. Since it is usually not possible to construct an initial solution that is guaranteed to be in I_{opt}, the ratio $rGBOA = |I_{opt}|/|S|$ can be taken as an indicator of the probability to find an optimal solution. In the extreme case, if we start from a random solution, the probability of finding a global optimum is exactly $|I_{opt}|/|S|$. Therefore, the higher this ratio, the higher the probability of success of the algorithm.

Given a local search algorithm $\mathcal{A}$, the topology and structure of the search graph determine the effectiveness of $\mathcal{A}$. In particular, the reachability of optimal

[3] Ties are broken by enforcing a lexicographic order of states.

[4] In this case, not only probabilistic basins of attraction are of interest, but also the probability a given state can be reached. Some studies in this direction are subject of ongoing work.

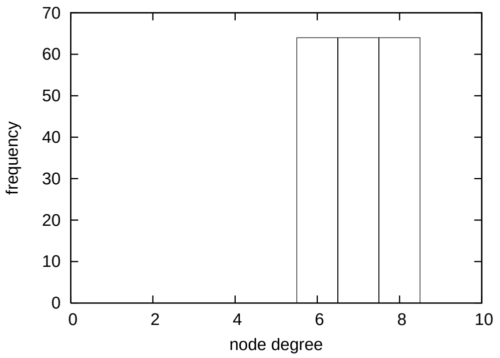

Figure 2: Node degree frequency of the neighborhood graph in the case of $n = 10$ and the model with SB constraints.

solutions is the key issue. Therefore, the characteristics of the BOA of optimal solutions are of dramatic importance.

3 LABS search space

LABS consists of finding an assignment to binary variables such that an energy function defined upon them is minimized. Given n binary variables $x_1, \ldots, x_n$, which can assume a value in $\{-1, +1\}$, we define the k-th correlation coefficient of a complete variable assignment s $C_k(s) = \sum_{i=1}^{n-k} x_i\, x_{i+k}$, $k = 1, \ldots, n-1$ and the total function to be minimized is $E(s) = \sum_{k=1}^{n-1} C_k^2(s)$.

We exhaustively explored the search space of LABS, for n ranging from 6 up to 18.[5] As neighborhood function we chose the one defined upon unitary Hamming distance, that is the most used one for problems defined over binary variables. (We emphasize that this choice determines the fundamental topological properties of the search space.) In the model with SB constraints (hereinafter referred to as $\mathcal{M}_s$, while the original model will be referred to as $\mathcal{M}$), only a subset of symmetric solutions has been cut, by enforcing constraints on the three left-most and right-most variables [7].

We first have to study how the neighborhood graph changes upon the application of SB constraints. In $\mathcal{M}$, the neighborhood graph induced by single variable flips is a hypercube in which each node is connected to other n nodes. This graph has a constant degree equal to n. The neighborhood graph associated to $\mathcal{M}_s$ is characterized by a node degree frequency that varies in a small range, around a mean value slightly small than n. In Figure 2, a representative example of this effect is depicted. The plot shows the number of nodes with a given degree versus the degree, as commonly done when studying the graph degree distribution [2]. We can observe that nodes are equally divided in three sets of degree 6,7, and 8. The topological characteristics of this graph are not affecting the search, since the reachability of nodes is not significantly perturbed. Therefore, we can exclude that SB constraints in LABS affect local search by only perturbing the topological properties of the neighborhood graph.

We have now also to consider the features of the search graph, which, in general, can be algorithm dependent. (This is the case for basins of attraction, while local and global optima only depend on the objective function and the neighborhood.) The search space characteristics of interest are the number of feasible states, the number of global and local optima and the value *rGBOA*. Observe that the concept of density of local and global optima[6], often taken as an indicator of instance hardness, is generalized by the concept of GBOA, because by observing the basins of attraction also the actual algorithm is taken into account. Furthermore, while the density of local optima implicitly assumes that each local (resp.global) optima can be reached by the same number of states, the concept of basin of attraction can capture also non-uniform cases. Table 1 reports the number of feasible states, the number of global and local optima and GBOA in the case without and with SB constraints, respectively. The basins of attraction are defined with respect to deterministic iterative improvement. Observe that in all the cases, except for $n = 11, 13$, *rGBOA*$(\mathcal{M}_s)<$*rGBOA*$(\mathcal{M})$.

4 Local search: Experimental analysis

We attacked LABS (with $n = 6, \ldots, 18$) with four different local search algorithms: Best improvement with randomly broken ties (BI), First improvement with random order among neighbors (FI), Simulated annealing (SA) and Tabu search (TS).[7] From the perspective of search space exploration, the algorithms chosen exhibit a varying explorative attitude, starting from the lowest of BI to the highest of TS, while all keeping a 'greedy' character.[8]

We run each algorithm on the original model and on the model with SB constraints. The algorithms are stopped after $10n$ non-improving moves. This termination condition enables us to compare the algorithms on the basis of the best solution they returned once a steady state is reached. (In the literature of

[5]The size limit is due to the exhaustiveness of the analysis.

[6]i.e., the ratio between number of local (resp. global) optima and number of feasible states

[7]The initial temperature in SA has been set after a simple trial-and-test procedure. The tabu tenure in TS is randomly restarted each iteration in a range between 1 and $n/2$, in the spirit of robust tabu search [14].

[8]A deep discussion on this topic, involving also intensification and diversification, can be found in [3].

Table 1: Search space characteristics of LABS instances.

n	feasible states		global optima		local optima		global BOA ratio	
	no SB	with SB	no SB	with SB	no SB	with SB	no SB	with SB
6	64	12	28	5	0	0	1.0	1.0
7	128	24	4	1	24	5	0.40625	0.33333
8	256	48	16	3	8	2	0.86328	0.77083
9	512	96	24	4	84	16	0.42969	0.37500
10	1024	192	40	7	128	29	0.54590	0.45833
11	2048	384	4	1	240	52	0.03906	0.04427
12	4096	768	16	3	264	61	0.07544	0.06901
13	8192	1536	4	1	496	111	0.01831	0.01953
14	16384	3072	72	11	664	177	0.21240	0.15202
15	32768	6144	8	2	1384	326	0.01956	0.01742
16	65536	12288	32	8	1320	332	0.05037	0.04972
17	131072	24576	44	9	3092	721	0.05531	0.04073
18	262144	49152	16	2	5796	1372	0.02321	0.01068

metaheuristics, this state is also commonly called *stagnation*.) Table 2 gives a synoptic view of the algorithm performance in term of success ratio (out of 1000 runs).

A comparison of the performance of each algorithm on the two problem models is given in Figures 3–6, in which we plotted the difference of solved instances (perc.) against n, i.e., $\Delta_{\%} = 100 \times (\text{solved(noSB)}-\text{solved(SB)})/1000$. A point above (resp. below) the line means that the algorithm achieved a higher success ratio on $\mathcal{M}$ (resp. $\mathcal{M}_s$) than on the other model; the greater the y-value, the greater the difference between the success ratio in the two cases. Note that the performance on $\mathcal{M}$ dominates the one on $\mathcal{M}_s$ in all but the TS case .

The correlation between number of successes and *rGBOA* is particularly interesting. From the plots in Figures 7–10, we observe that for BI and FI the number of successes is proportional to the size of the global optima basin of attraction. In the case of SA and TS, while the correlation is still observable, we note that the performance remains quite high even for low values of *rGBOA*, especially in the case of TS. The *rGBOA* values have been measured on the basis of deterministic best improvement, therefore it is not surprising that both BI and FI show a proportional relation between successes and fraction of states that make the search converging to a global optimum. SA performs a more effective search space exploration than iterative improvement procedures, and even more TS, therefore the number of successes they achieve is much higher than that of BI and FI. It is interesting to note that the performance of both SA and TS starts to degrade (quite abruptly) when the normalized size of the global optima basin of attraction approaches a threshold value. Moreover, for TS this value is smaller than for SA, in other words, the more sophisticated an exploration strategy is, the lower the value of *rGBOA* at which the performance starts to be strongly affected.

We would like to remark that we can use $rGBOA|_{BI}$ as an indication for the hardness of a particular instance since *(i)* a positive correlation between $rGBOA|_{BI}$ and search cost has been experimentally observed, and *(ii)* almost all local search algorithm incorporate –to a certain degree– a greedy mechanism such as best improvement. The use of probabilistic basins of attraction, that would enable a more precise evaluation as linked to the actual algorithm used, introduces the problem of controlling the precision of errors due to search space sampling[9] and it is subject of ongoing work.

5 Discussion and future work

The available data are still not sufficient to provide a complete descriptive theory of the phenomenon, nevertheless we have experimental results to support our conjecture[10]. First of all, it is apparent that *rGBOA* is reduced in the model with SB constraints. Another important observation is that local search performance is strongly affected by the size of the global optima basin of attraction. This relation is in the form of a positive correlation (i.e., the smaller the BOA, the lower the performance) and it is well approximated by a linear relation in the case of simple local search algorithms (BI and FI), while it is nonlinear in the case of more complex search strategies (SA and TS). The

[9] Local optima are indeed *rare events* and sampling methods more efficient than Montecarlo sampling have to be used.

[10] Analogous results have been produced by studying optimization problems characterized by permutation symmetries [13].

Table 2: Synopsis of the number of solved instances (out of 1000 runs) of the four local search algorithms on the original model and the model with SB constraints.

n	**BI**		**FI**		**SA**		**TS**	
	no SB	with SB	no SB	with SB	no SB	with SB	no SB	with SB
6	1000	1000	1000	1000	1000	1000	1000	1000
7	417	324	530	484	1000	999	1000	1000
8	875	751	825	728	1000	960	1000	1000
9	438	342	266	267	913	834	1000	1000
10	561	426	995	720	996	920	1000	1000
11	42	41	47	30	101	136	528	928
12	77	63	39	37	308	318	835	895
13	16	17	3	5	105	111	283	251
14	200	125	202	142	857	731	992	1000
15	25	24	22	20	157	179	363	599
16	66	52	29	39	336	307	819	919
17	60	43	71	45	508	388	909	916
. 18	25	6	35	8	270	131	678	412

nonlinearity of this relation plays a big role when we compare the performance of local search algorithms (in terms of success ratio). In fact, large differences in *rGBOA* imply large deviations of the performance. But on the other side, when the difference is quite small, other factors come into play. Indeed, in some cases we can observe that the variance in the performance can not directly be attributed only to the size of global optima basin of attraction.

The reduction of *rGBOA* and the correlation between performance and *rGBOA* could give a first order empirical explanation of why SB constraints have been observed to be harmful for local search.

We have also to note that the more complex the search strategy is, the more it could take advantage of the search space reduction, even if *rGBOA* decreases. This issue should be investigated in detail, especially by experimenting with large size instances.

One of the most important questions to address at this stage is why do SB constraints reduce GBOA. A partial answer to this question is that *new* local optima are created in M_s: some feasible states in M_s are local optima in M_s, but not in M [11]. Therefore, rGBOA is decreased. Our general conjecture is that SB constraints modifies the trajectories to global optima basins of attraction in such a way that trajectories are more likely to contain high *barriers* [6], very difficult to overcome by simple local search strategies. We are currently studying a formal graph-based model for verifying this conjecture and preliminary observations on some specific properties are in accordance with the experimental results.

Acknowledgments

I thank Steve Prestwich and the anonymous reviewers for useful comments and suggestions on this work.

REFERENCES

[1] Y. Bar–Yam. *Dynamics of Complex Systems.* Studies in nonlinearity. Addison–Wesley, 1997.

[2] A.-L. Barabási. *Linked: The New Science of Networks.* Perseus Books Group, 2002.

[3] C. Blum and A. Roli. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys*, 35(3):268–308, September 2003.

[4] J. Crawford, M.L. Ginsberg, E. Luck, and A. Roy. Symmetry-breaking predicates for search problems. In *KR'96: Principles of Knowledge Representation and Reasoning*, pages 148–159. Morgan Kaufmann, San Francisco, California, 1996.

[5] M. Ebner, M. Shackleton, and R. Shipman. How neutral networks influence evolvability. *Complexity*, 7(2):19–33, 2001.

[6] H.H. Hoos and T. Stützle. *Stochastic Local Search: Foundations and Applications.* Morgan Kaufmann Publishers, 2004.

[7] S. Mertens. Exhaustive search for low-autocorrelation binary sequeneces. *J. Phys. A*, 29:L473–L481, 1996.

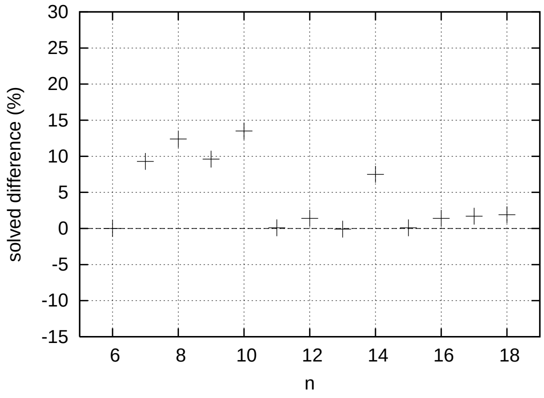

Figure 3: **Best improvement**. Percentage of the difference between solved instances in the original model and solved instances in the model with SB constraints.

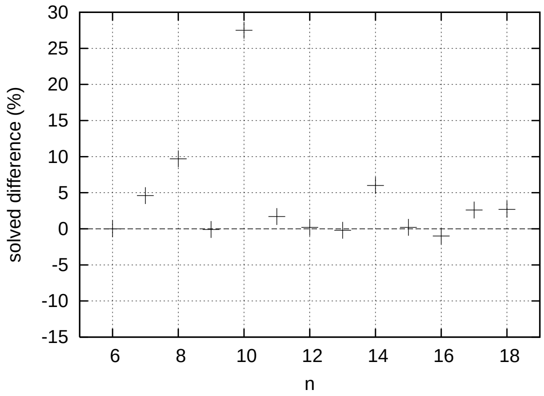

Figure 4: **First improvement**. Percentage of the difference between solved instances in the original model and solved instances in the model with SB constraints.

[8] E. Nowicki and C. Smutnicki. A fast taboo search algorithm for the job-shop problem. *Management Science*, 42(2):797–813, 1996.

[9] S. Prestwich. First-solution search with symmetry breaking and implied constraints. In *Proc. of CP'01 Workshop on Modelling and Problem Formulation*, 2001.

[10] S. Prestwich. Supersymmetric modeling for local search. In *Proc. of SymCon'02 Workshop on Symmetry and Constraint Satisfaction Problems*, 2002.

[11] S. Prestwich and A. Roli. Symmetry breaking and local search spaces. In *Proceedings of CPAIOR 2005*, volume 3524 of *Lecture Notes in Computer Science*. Springer–Verlag, 2005.

[12] J. F. Puget. Symmetry breaking revisited. In *Principle and Practice of Constraint Program-*

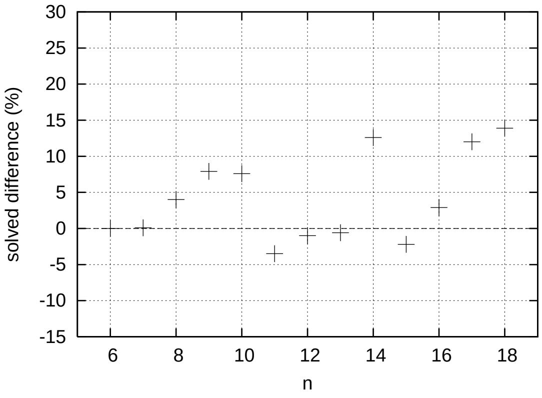

Figure 5: **Simulated annealing**. Percentage of the difference between solved instances in the original model and solved instances in the model with SB constraints.

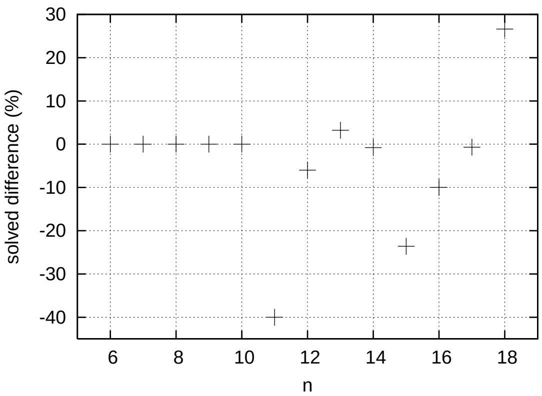

Figure 6: **Tabu search**. Percentage of the difference between solved instances in the original model and solved instances in the model with SB constraints.

ming – CP02, volume 2470 of *Lecture Notes in Computer Science*, pages 446–451. Springer-Verlag, 2002.

[13] A. Roli. Symmetry-breaking and local search: A case study. In *SymCon'04 – 4th International Workshop on Symmetry and Constraint Satisfaction Problems.* 2004.

[14] E. Taillard. Robust taboo search for the quadratic assignment problem. *Parallel Computing*, 17:443–455, 1991.

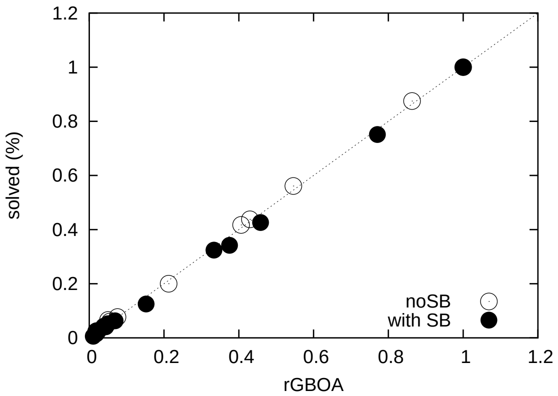

Figure 7: **Best improvement**. Solved instances (perc.) plotted against the size of the global optima BOA.

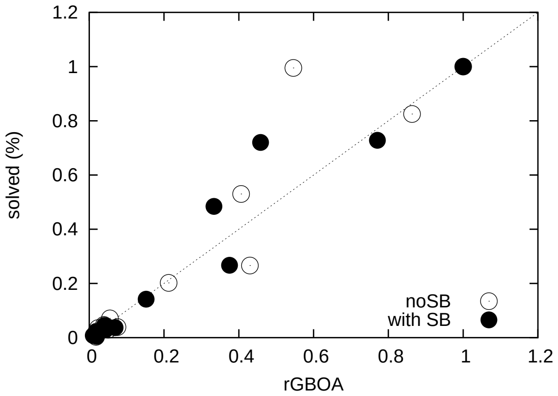

Figure 8: **First improvement**. Solved instances (perc.) plotted against the size of the global optima BOA.

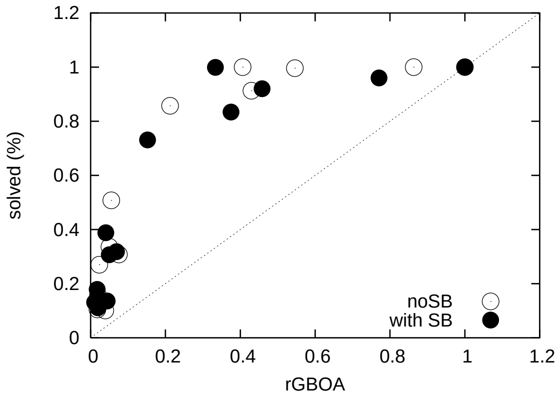

Figure 9: **Simulated annealing**. Solved instances (perc.) plotted against the size of the global optima BOA.

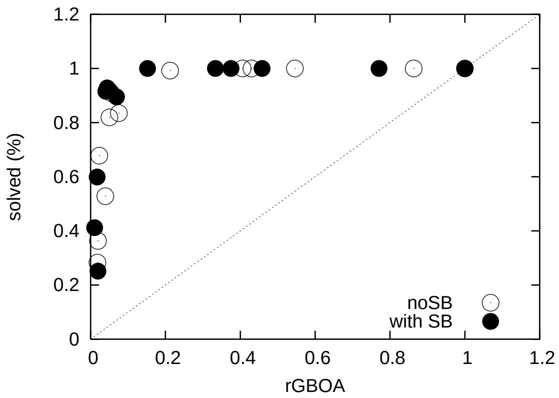

Figure 10: **Tabu search**. Solved instances (perc.) plotted against the size of the global optima BOA.