An Analysis of the Configuration Space of the Maximal Constraint Satisfaction Problem

Meriema Belaidouni[1] and Jin-Kao Hao[2]

[1] LGI2P, EMA-EERIE, Parc Scientifique Georges Besse, F-30000 Nîmes
[2] LERIA, Université d'Angers, 2 bd Lavoisier F-49045 Angers Cedex 01
{meriema@site-eerie.ema.fr, Jin-Kao.Hao@univ-angers.fr}

Abstract. This paper presents an analysis of the configuration space of a well-known combinatorial search problem called MAX-CSP. The analysis is based on a measure called "density of states (d.o.s)". We show experimentally that the configurations of a random MAX-CSP instance follow a normal distribution. This distribution allows us to get some insights about the behavior of a random walk search.

1 Introduction

The notion of landscape has been widely used to study the behavior of heuristic search algorithms such as genetic algorithms and local search. Landscape measures like autocorrelation indicate when a landscape is easy or difficult to search. Similarly, studies of local optima contribute to explaining why a landscape is difficult.

Recently, a new measure has been proposed aiming to study the configuration space of search problems [1, 10]. This measure, called the density of states (d.o.s.), gives complementary and valuable information to predict or explain the dynamics of heuristic search methods. The d.o.s. simply counts the number of configurations per cost value in the configuration space. The counting may be carried out in different ways - analytically on some problems [7, 8], by enumeration on small instances [4] or by approximation on large instances [1, 3].

This paper undertakes the study of the configuration space of the Maximal Constraint Satisfaction Problem (MAX-CSP), a well-known combinatorial search problem widely studied by people working in artificial intelligence. The MAX-CSP is a general model allowing the formulation of numerous problems such as the MAX-SAT and the k-Coloring problem as well as many real-world applications such as scheduling and planning.

Using the d.o.s., we show that the density of states of random MAX-CSP instances follows a normal distribution characterized by a mean and a variance. This result allows us to obtain some intriguing insights into Random Walk (RW) dynamic. Indeed, previous experiments with RW have shown that the search invariably stagnates within zones having cost values which differ little, independent of the start point of the search. The normal distribution of the configuration space allows us to explain this RW phenomenon.

This article is organized as follows: After a brief recall of the Maximal Constraint Satisfaction Problem in Section 2, Section 3 presents some experiments with the Random Walk method. Section 4 deals with the principles of the measure of density of states. Section 5 establishes the relation between Random Walk and density of states with the help of intensive experiments. Section 6 concludes and suggests some perspectives.

2 Maximal Constraint Satisfaction Problem (MAX-CSP)

2.1 Definition

The MAX-CSP can be defined by means of the notion of constraint networks. A constraint network is a triplet $< V, D, C >$ where:

- $V = \{V_1 ... V_n\}$ is a finite set of variables;
- $D = \{D_1 ... D_n\}$ is a finite set of value domains associated with the variables;
- $C = \{C_1 ... C_n\}$ is a finite set of constraints, each constraint being a subset of the Cartesian product of the domains of several variables, specifying the forbidden value tuples.

Given a constraint network $< V, D, C >$, the problem of maximal constraint satisfaction consists in finding a value in D_i for each variable V_i such that the number of satisfied constraints is maximal [11]. In practice, instead of maximizing the number of satisfied constraints, one minimizes the number of unsatisfied constraints - which is rigorously equivalent. Subsequently, we adopt this minimization version of the problem. The objective function is noted f.

2.2 Instance generation

The test instances that are used in this work correspond to random, binary constraint networks (each constraint concerns only two variables) generated according to a standard model. A network class is defined by $< n, d, p_1, p_2 >$ which has n variables, d values per variable, $p_1.n.(n-1)/2$ constraints taken randomly from $n.(n-1)/2$ possible ones (p_1 is called the density), and $p_2.d^2$ forbidden pairs of values taken randomly from d^2 possible ones for each constraint (p_2 is called the tightness). For each given class $< n, d, p_1, p_2 >$, different instances can be generated using different random seeds s. A constraint network can be under-constrained or over-constrained [6]. These different regions are characterized by a factor called *constrainedness*: $\kappa = \frac{n-1}{2} p_1 \log_d(\frac{1}{1-p_2})$ $\kappa = 1$ separates under- ($\kappa < 1$) from over- ($\kappa > 1$) constrained networks. Networks with $\kappa \approx 1$ correspond to critically-constrained ones.

3 Experiments with the Random Walk

In this section, we carry out some experiments with Random Walk. We are especially interested in simple questions such as: Does RW follow any direction during its search? Where does a RW search go when it begins with very different starting points? Where does the search stop?

3.1 Random Walk

The Random Walk (RW) is a simple yet important search process for studying search spaces. RW is particularly relevant for an important class of optimization methods based on local search. RW is defined by a an iterative process. It begins with an initial point s in the configuration space S and then moves repeatedly from the current s to a randomly chosen neighboring one $s' \in S$ according to a neighborhood relation N. The experiments described below are based on this simple RW.

3.2 Experiments

The experiments aim to demonstrate the relevance of density of states to the dynamic of local search heuristics. Density of states carries new information which is not covered by measures based on correlation measure or local optima density. The experiments are carried out as follows. We take a random MAX-CSP instance I=(S, f). We run a Random Walk search on the instance I search space from three different kind of configurations:

- RW-I: from a *random* configuration,
- RW-II: from a *random optimal* or near-optimal configuration,
- RW-III: from a *random very bad* configuration.

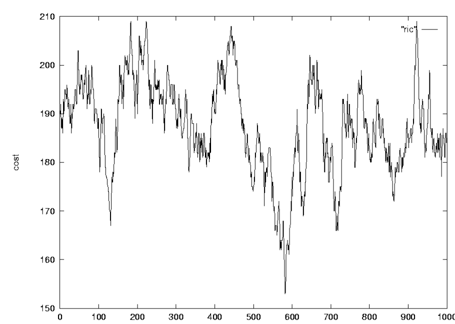

Fig. 1. Cost evolution for RW-I (RW from a random starting configuration)

The instance used here has a known optimal cost of $f^* = 0$. Fig. 1 and Fig. 2 summarize the evolution of cost values obtained with RW-I, RW-II and RW-III. RW-I starts its evolution most probably around $f \approx 190$. All generated cost values oscillate around the cost area [150,210]. RW-II leaves the optimal cost area around $f = 0$ and goes toward the cost area [150,210]. RW-III leaves the bad cost values $f \approx 438$ and moves toward the cost area [150,210].

These experiments are intriguing because they demonstrate RW follows a particular direction independently of its starting configuration. Indeed all these search processes stagnate around the same cost area. It seems that RW is strongly attracted by the configurations of this cost area. It is therefore interesting to consider the configurations according to their cost values.

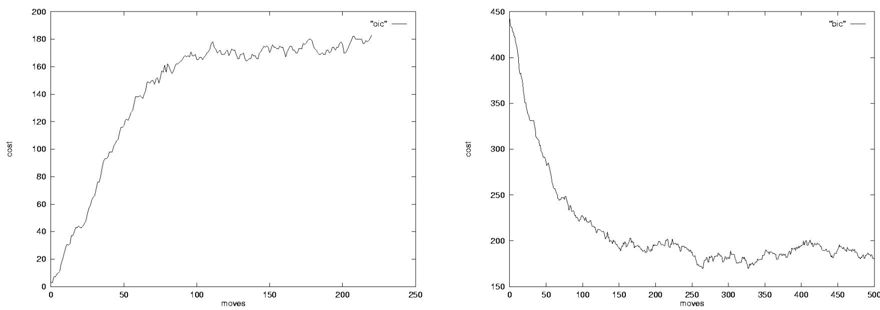

Fig. 2. Cost evolution for RW-II (left, from an optimal configuration) and RW-III (right, from a very bad configuration)

3.3 Cost Density of Random Walk

Now we re-run the above RW search and count the number of times a cost is encountered. This leads to a distribution that can be called cost density (Figure 3). Figure 3 displays clearly a normal distribution. Consequently it appears

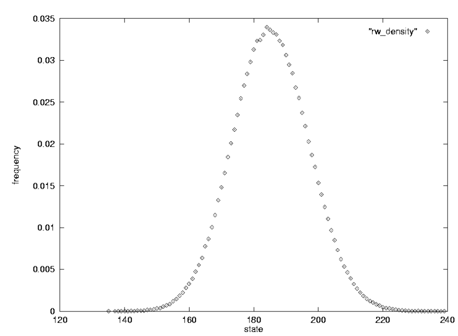

Fig. 3. Cost density of Random Walk

that RW-II and RW-III leave their initial cost areas (low and high) to coincide with this distribution. The remark holds also true for RW-I. Two factors may be responsible for this distribution: 1) the neighborhood relation and 2) the number of configurations per cost (density of states). In what follows, we present a general approximation method for finding density of states and show tight relations between density of states and the dynamic of Random Walk.

4 Density of Cost States

For a given configuration space (S, f) associated with a combinatorial problem, the density of states (d.o.s) gives the number of configurations per cost value.

Asselmeyer et al. [1, 10] have proposed a statistical method to approximate d.o.s that emits no hypothesis on the configuration space studied. The main purpose of the d.o.s. is to establish a classification of problem difficulty based on the mean and the variance of the density of states.

4.1 Method for Approximating Density of States

The approximation technique for d.o.s draws an analogy between energy particles F in thermodynamics and configurations s of cost $f(s)$ of the space S. The method bases itself on the following law:

$$P_{eq}(F) \sim N(F)e^{-\triangle F/T} \tag{1}$$

- which links the number $N(F)$ of particles of energy F to the frequency of their appearance P_{eq} in a sampling process using the Metropolis method at temperature T (Metropolis method was introduced in 1953 by N. Metropolis et al. [9]. This process moves from configuration s to configuration s' with probability p= $e^{-\triangle f/T}$ if $\triangle f = f(s') - f(s) \geq 0$ and probability p=1 otherwise. The approximation proceeds via the following stages:

1. Run the Metropolis sampling process at temperature T. On the sample of size N and for each energy state F, count the number $N(F)$ of configurations having value F, approximate the density $P_{eq}(F)$ by the *frequency distribution* $\frac{N(F)}{N}$.
2. Adjust the density $P_{eq}(F)$ to the scale by applying a multiplier coefficient of $e^{F/T}$ as follows:
$$w(F) = P_{eq}(F)e^{F/T} \tag{2}$$
3. Normalize to obtain an approximation of the state density
$$W(F) = \mathbf{N}\frac{w(F)}{\sum\limits_{F=0}^{F=c} w(F)} \tag{3}$$
where $\mathbf{N}$ is the total number of configurations, and c the number of values that F can take.
4. Execute the above simulation at different temperatures in order to scan the whole range of possible states.

5 Experiments

We carry out our experimental tests on random, binary MAX-CSP landscapes. The first goal is to confirm the behavior (see section 3) of Random Walk for different initializations. The second goal is to explain this behavior using d.o.s.

5.1 Random Walk Behavior and Cost Density

Table 1 presents 7 classes of instances used in our experiments. Each class includes ten different instances. The constrainedness is given by κ, f^* is the best known or optimal cost. On each class, we run Random Walk with three different starting configurations: (I) a random configuration, (II) an optimal configuration, (III) a very bad configuration (see section 3.2). f_{begin} (repect. f_{end}) indicates the initial (respect. final) cost value. The neighborhood relation involved is N_1; two configurations are neighbors if they differ by only one variable value.

Now if we consider the case of $< 100.10.15\%.25\% >$, all initializations converge to the cost area $[130, 240]$. The same phenomenon occurs for the other classes of instances with different final cost intervals f_{end}. Fig. 4 shows the RW cost density of instance classes (a) to (g). It appears that the final cost intervals for each class of instances correspond to the most probable costs. This explains why RW is attracted by these cost values. It remains to explain why this density occurs for costs generated by Random Walk and why the random initial cost values are never very far from each other. The reason for these two phenomena lies, in large part, in the density of states (d.o.s.).

$Classes$	κ	f^*	I		II		III	
			f_{begin}	f_{end}	f_{begin}	f_{end}	f_{begin}	f_{end}
(a)100.10.15%.25%	0.93	0	≈ 204	$[130, 240]$	≈ 2	$[130, 240]$	≈ 438	$[130, 240]$
(b)100.15.20%.30%	1.22	18	≈ 293	$[240, 360]$	≈ 25	$[240, 360]$	≈ 621	$[240, 360]$
(c)100.10.20%.25%	1.24	19	≈ 259	$[180, 320]$	≈ 21	$[180, 320]$	≈ 540	$[180, 320]$
(d)100.15.10%.45%	1.64	11	≈ 234	$[170, 270]$	≈ 16	$[170, 270]$	≈ 448	$[170, 270]$
(e)200.20.25%.04%	0.34	0	≈ 194	$[140, 260]$	≈ 0	$[140, 260]$	≈ 778	$[140, 260]$
(f)200.20.05%.20%	0.37	0	≈ 218	$[150, 260]$	≈ 0	$[150, 260]$	≈ 669	$[150, 260]$
(g)200.20.18%.14%	0.9	0	≈ 518	$[420, 580]$	≈ 2	$[420, 580]$	≈ 1261	$[420, 580]$

Table 1. Classes of different size for confirming the behavior of Random Walk

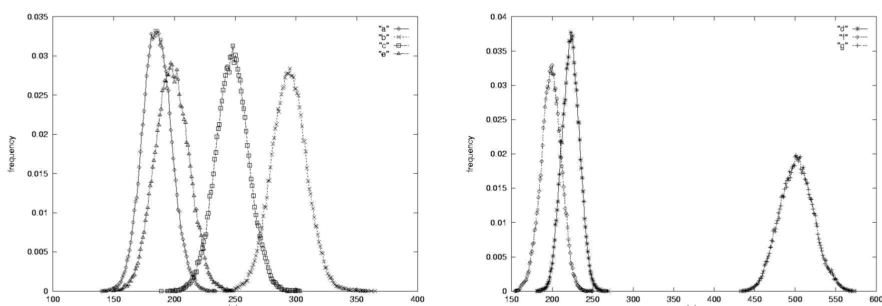

Fig. 4. Cost density of Random Walk for classes (a) to (g)

5.2 Approximation of Density of States

In what follows, we propose to approximate the d.o.s of the class of instances $< 100.10.15\%.25\% >$ using the method described in section 4.1. The density of states will help us to understand the cost density of Random Walk and the random initial cost values.

Tuning of parameters Applying the Metropolis algorithm to approximate a frequency distribution requires fixing the following parameters: the size of the sample, the neighborhood and the temperature. For a given instance, the best values of these parameters are usually determined empirically.

1. *size of sample:* This parameter is very important for the validity of an approximation. In the absence of analytical formulae giving a minimum value, trials are conducted on successively larger sizes with a view to stabilizing the approximation. It can happen though that this doesn't bring a solution - notably when the minimum size is too large. For our experiments, we start typically with a size of 1,000 and end at 500,000. Beyond 500,000, the curves of frequency distribution are no longer sensitive to increase in sample size.
2. *neighborhood:* Two configurations are neighboring if they differ by the value of a single variable.
3. *temperature:* Compared with the sample size, temperature plays an even more critical role. It is this parameter that allows the search to reach zones of low cost[1]. In our experiments, we conduct trials at successively lower temperatures with a view to the greatest coverage of low cost zones. We typically use temperatures between $T = 35$ and $T = 0.5$. For values below 0.5 or above 35, the frequency distributions furnish no further information to the d.o.s.. For $T > 35$ the distributions differ little from that of $T = 35$ while below $T = 0.5$, multiplication by the scale factor obliterates the results.

Results We show now detailed results obtained on the class $< 100.10.15\%.25\% >$ (This class has a known optimum ($f^* = 0$) [5]). Fig. 5 presents the configuration distribution for this class: on the x-axis, the set of costs - going from 70 to 240 with a step of 20; on the y-axis, the estimated number of configurations for each cost. In Fig. 5, one notices that the configuration distribution approaches a normal law. Of the 10^{100} configurations in the space, the majority, or $\approx 3,5 \times 10^{98}$, seems to concentrate around the average of 185. The number of configurations diminishes when one departs from this average. The significant information lies in the area between 140 and 220. The extensions of the curve from 140 to 80 and from 220 to 240 correspond to regions where approximation has yielded figures which are almost null in relation to the other values.

At this point, one can explain why the random initial cost value for RW-I was $f \approx 190$. In fact, the selected random cost responds to the cost distribution in the configuration space. More precisely, it belongs to the most probable cost values of the corresponding instance, here $[130, 240]$. This conclusion is confirmed by the

[1] Recall that we are minimizing the number of violated constraints.

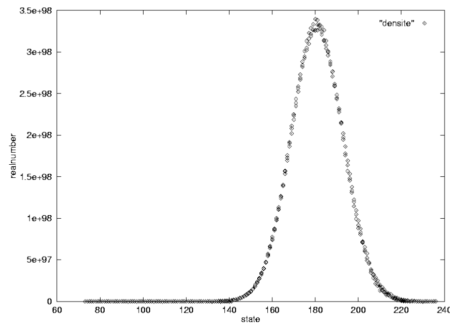

Fig. 5. Density of states of the instance < 100.10.15%.25% >: on the x-axis, costs of the objective function and on the y-axis, estimated number of configurations

class of instances (a) to (g). Indeed, all the initializations of Table 1 agree with the d.o.s. of these instances (204 $\in$ [130,240]; 293 $\in$ [240,360]; 259 $\in$[180,320]; 234 $\in$ [170,270]; 194 $\in$ [140,260]; 218 $\in$ [150,260]; 518 $\in$ [420,580]).

5.3 Does d.o.s Explain the Cost Density of Random Walk?

After explaining the effect of d.o.s on the random initial cost, we try to establish its relation with the cost density of Random Walk.

Similarity - Shape Observing the d.o.s (Fig. 5) and the cost density of RW (Fig 3.), one can notice a common point: the normal shape. To show that this remark is not limited to the class of instances < 100.10.15%.25% >, we undertake one more experiment on the 7 classes of instances of Table 1. Notice that Fig. 4 shows already normal shapes for Random Walk cost density. To confirm the normality of the d.o.s. for these instances, we re-run the above described sampling procedure and then carry out a normality test using Matlab's 'normplot'. This test gives perfect linear curves for these instances, confirming thus the normality nature of the d.o.s. Let us mention that we carried out experiments on other MAX-CSP instances and used a "Random Selection" sampling technique which simply takes a set of configurations in S in a random and independent manner. All these experiments have led to the same conclusion.

At this point, we wish to mention the study presented in [12]. It shows, using an analytical approach, the following result: taking the independence of constraint satisfaction, the random variable nuc "number of unsatisfied constraints" follows a binomial law $nuc \sim B(nc, p_2)$; where $nc = p_1 \frac{n.(n-1)}{2}$ is the number of constraints, n the number of variables, p_1 the density and p_2 the hardness. We notice that this analytical result agrees perfectly with our experimental results since it is possible to approximate a binomial by a normal under certain conditions (conditions satisfied in this case).

Differences - Mean and Variance Now we turn to consider the differences between density of states and RW cost density using the average and the variance. In table 2 one can find the mean and the standard deviation of the binomial $B(nc, p_2)$ and the Random Selection—(data based on 10000 random configurations). Both are considered to be approximations of the exact mean and standard deviation of the classes (a) to (g).

We compare these approximated mean and deviation with the RW's mean and standard deviation (data based on 1000000 configurations generated by Random Walk). The first neighborhood relation involved is N_1 (two configurations are neighbors if they differ by one variable value), the second is N_2 (two configurations are neighbors if they differ at a single conflicting variable). The results

Classes	$B(nc, p_2)$		RandomSelection		RandomWalk(N_1)		RandomWalk(N_2)	
	μ	σ	μ95%Conf.	σ95%Conf.	μ95%Conf.	σ95%Conf.	μ95%Conf.	σ95%Conf.
(a)100.10.15%.25%	185.62	11.78	[185.41, 185.56]	[11.76, 11.86]	[185.54, 185.58]	[11.76, 11.79]	[180.89, 180.93]	[12.07, 12.11]
(b)100.15.20%.30%	297.00	14.42	[294.76, 294.94]	[14.38, 14.51]	[295.09, 295.14]	[14.02, 14.24]	[294.21, 294.26]	[14.53, 14.57]
(c)100.10.20%.25%	247.50	15.73	[247.42, 247.59]	[13.36, 13.74]	[247.62, 247.68]	[13.57, 13.61]	[246.06, 246.11]	[13.72, 13.76]
(d)100.15.10%.45%	222.50	11.06	[222.13, 222.27]	[11.04, 11.14]	[222.11, 222.15]	[11.11, 11.14]	[220.18, 220.22]	[11.17, 11.20]
(e)200.20.25%.04%	199.00	13.82	[198.85, 199.03]	[13.77, 13.89]	[198.80, 198.86]	[13.86, 13.89]	[145.12, 145.18]	[14.81, 14.85]
(f)200.20.05%.20%	199.00	12.62	[198.87, 199.02]	[12.57, 12.68]	[198.98, 199.03]	[12.65, 12.69]	[156.28, 156.33]	[13.73, 13.77]
(g)200.20.18%.14%	501.48	20.77	[501.32, 501.58]	[20.61, 20.79]	[501.39, 501.47]	[20.76, 20.82]	[498.17, 498.26]	[20.77, 20.83]

Table 2. Approximation of mean and standard deviation for different neighborhood relations.

show that the mean and the standard deviation of Random Walk depend on the mean and standard deviation of the density of states (represented by the Binomial and Random Selection mean and standard deviation). They are also sensitive to the neighborhood relation. Thus, the samplings involving $N1$ and N_2 are biased and present different results comparing with the Random Selection and the Binomial law. This phenomenon has an explanation: the use of a neighborhood relation favors the neighboring configurations and so introduces a correlation between the generated configurations while a pure random sampling (here Random Selection) using no neighborhood is a process without memory. One can also notice that the bias introduced by N_2 is larger than the one introduced by N_1. Indeed we obtained smaller means values with N_2. Moreover, the bias of N_2 is not regular: it is much more significant in instances (e) (whose mean is about 144 instead of 198 for N_1) and (f) (where the mean is around 156 instead of 198 for N_1) than in others.

We conclude from this experiment that the mean and the standard deviation of the cost density of RW result from the mean and the standard deviation of the density of states plus a bias introduced by the neighborhood relation. This bias depends on the instance. Before presenting our future work, it is worth pointing out that the mean and standard deviation of Random Walk can be considered as measures for neighborhood efficiency. These measures have the particularity (or quality) of being instance dependent.

6 Conclusions and Perspectives

In this paper, we have analyzed the MAX-CSP configuration space. We have shown that the density of states (d.o.s) of random instances approaches a normal law. This distribution sheds light on some interesting questions related to the dynamic of random walks: (1) we understand now why RW is attracted by some cost areas. In fact, these cost areas correspond to space areas that contain large concentrations of configurations. However the mean cost of random walks and d.o.s. are not (generally) equal. This is due to the bias that the neighborhood relation introduces. (2) We learn that a random initial configuration will have a cost around the mean of the d.o.s. Density of states appears thus to be the first ingredient for understanding the behavior of search methods, the second and third ingredients being the neighborhood relation and the stochastic local search strategy chosen.

In our future work we want to approximate the cost densities of advanced local search methods such as Tabu Search and Simulated Annealing and analyze their relation with d.o.s. Also, we start to examine other problems like Graph Coloring and SAT. Finally, let's note that configuration analysis, like cost analysis, can contribute to understanding stochastic local search [2].

Acknowledgements The authors highly appreciate the valuable and detailed comments of the anonymous reviewers.

References

1. T. Asselmeyer, H. Rosé, and W. Ebeling. Smoothing representation of fitness landscape - the genetype - phenotype map of evolution. *Biol. Systems*, 1995.
2. M. Belaidouni and J. K. Hao. Landscapes of the maximal constraint satisfaction problem. *LNCS Springer Verlag*, 1829:244–255, 2000.
3. J. Bresina, M. Drummond, and K. Swanson. Search space characterization for telescope scheduling application. *AAAI'94*, 1994.
4. J. Cupal, C. Flamm, Renner A., and Stadler P.F. Density of states, metastable states and saddle points exploring the energy landscape of an rna molecule. *Int. Conf. Int. Sys. Mol. Biol.*, pages 88–91, 1997.
5. P. Galinier and J. K. Hao. Tabu search for maximal constraint satisfaction problems. *LNCS Springer-Verlag*, 1330:196–208, 1997.
6. I. P. Gent, E. MacIntyre, P. Prosser, and T. Walsh. Scaling effects in the CSP phase transition. *Proc. of CP95*, pages 70–87, 1995.
7. S. A. Kauffman. Adaptation on rugged fitness landscapes. *Lectures in the Sciences of Complexity*, pages 527–618, 1989. SFI Studies in the Sciences of Complexity, ed. D. Stein, Addison-Wesley Longman.
8. S. Kirkpatrick and G. Toulouse. Configuration space analysis of travelling salesman problem. *J. Physics*, 46:1277–1292, 1985.
9. N. Metropolis, A. W. Rosenbluth, M.N. Rosenbluth, and A. H. Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21:1087–1092, 1953.
10. H. Rosé, W. Ebeling, and T. Asselmeyer. Density of states - a measure of the difficulty of optimization problems. *LNCS Springer*, 1141:208–226, 1996.
11. E. Tsang. Foundation of constraint satisfaction. *Academic Press*, 1993.
12. G. Verfaillé. CSP aléatoires : Estimation de l'optimum. *Research Notes*, 1998.