Vers une quantification de la notion d'attraction en recherche locale stochastique

A. SIDANER
L.I.R.S.I.A.,
Université de Bourgogne
e-mail: sidaner@crid.u-bourgogne.fr

J.-J. CHABRIER
L.I.R.S.I.A.,
Université de Bourgogne
e-mail: chabrier@crid.u-bourgogne.fr

O. BAILLEUX
C.R.I.L.,
Université d'Artois
e-mail: bailleux@cril.univ-artois.fr

Résumé

L'algorithme Walksat est l'un des meilleurs algorithmes de recherche locale stochastique pour les instances du problème 3-SAT créées par le générateur aléatoire standard. Pourtant, pour des instances de même taille, son comportement est très variable.
Nous avons voulu étudier cet aspect de la recherche en terme d'attraction. Dans ce papier, nous proposons la définition d'un coefficient d'attraction qui quantifie la tendance d'un processus de recherche à se "diriger" vers une configuration donnée de l'espace de recherche. Nous montrons expérimentalement que cette grandeur permet de discriminer des régions des paysages de recherche qui "attirent" le processus spécifié par Walksat. Nos résultats expérimentaux montrent également l'existence, dans certaines instances, de configurations plus attractives que les solutions.

I. Introduction

Le problème SAT, archétype des problèmes NP-complets, est connecté à plusieurs domaines de recherche en Intelligence Artificielle : planification, résolution de problèmes pratiques, raisonnement automatique, ... Pour résoudre des instances du problème SAT, il existe de nombreux algorithmes que l'on peut classer en deux catégories : les algorithmes complets, comme Davis et Putnam [DP60], CSAT [DABC96] ou Satz [mLA97], et les algorithmes incomplets comme Gsat [SM92], tabou [Her90] ou le recuit simulé [VL87]. Ces derniers sont basés sur des heuristiques de recherche locale. Lorsque l'heuristique utilisée est probabiliste, l'algorithme est dit stochastique.
L'un des meilleurs algorithmes de recherche locale stochastique pour les instances de problèmes 3-SAT générés aléatoirement est Walksat [SKC94]. Pourtant, pour certaines de ces instances, l'algorithme mettra en moyenne beaucoup plus de temps (plusieurs ordres de grandeur) pour donner un résultat que pour d'autres instances de même taille. D'où vient cette différence ?
Afin d'étudier le comportement de l'algorithme vis-à-vis d'une instance de problème donnée, Catoni [Cat92] puis Jones [Jon95] ont introduit la notion de paysage de recherche. Un paysage de recherche est la donnée de trois paramètres : l'espace de recherche $\mathcal{E}$ (ensemble des configurations possibles), une fonction d'évaluation $\mathcal{A} : \mathcal{E} \rightarrow I\!N$ (ou fonction altitude) et une fonction de transition $\mathcal{P} : (\mathcal{E} \times \mathcal{E}) \rightarrow [0, 1]$ qui associe à tout couple de configurations (Cfg_1, Cfg_2) la probabilité que la configuration suivante soit Cfg_2 si la configuration courante est Cfg_1.
Grâce à cette abstraction, il est possible d'interpréter certains aspects du comportement du processus de recherche : regroupement des solutions en ensembles connexes, zones d'attraction autour des solutions [Sid99], plateaux, bassins d'attraction [FRPT97], etc. Par exemple, dans certains cas, le processus de recherche semble se "diriger" directement vers une solution, alors que dans d'autres, il a l'air "d'errer" dans des zones d'altitude constante. Un tel comportement peut être interprété comme la faculté qu'a le processus de recherche d'être plus ou moins "attiré" par les solutions.

Le but de ce papier est d'une part, de donner une définition plus formelle de la notion d'attraction, et d'autre part, de valider cette définition par une première série d'expérimentations.
A cette fin, nous introduisons deux grandeurs, l'indice d'attraction et le coefficient d'attraction, permettant de quantifier la tendance d'un processus à se diriger vers une configuration donnée de l'espace de recherche. L'idée sous-jacente à la définition de ces deux grandeurs est la comparaison du comportement du processus étudié avec le comportement d'un processus de référence aléatoire (*i.e.* qui n'est pas guidé par une heuristique de recherche). Alors que dans la définition de l'indice d'attraction toutes les configurations de l'espace de recherche interviennent de manière équiprobable, le coefficient d'attraction permet de prendre en compte la distribution des configurations dans les différentes trajectoires possibles du processus. La valeur obtenue est donc plus représentative du comportement du processus étudié.
La section suivante présente le cadre dans lequel nous nous plaçons pour effectuer nos expérimentations. La troisième section concerne les définitions de l'attraction. La quatrième section regroupe les résultats expérimentaux. Enfin, la cinquième section résume les avancées obtenues et présente des perspectives quant à leur exploitation.

II. Cadre de travail

1. Instances de problème

Le problème SAT est le problème de décision relatif à la satisfiabilité d'une expression booléenne sous forme normale conjonctive (forme CNF). Une telle expression est une conjonction de clauses, chaque clause étant une disjonction de littéraux, un littéral étant une variable booléenne ou sa négation. Chaque formule booléenne sous forme CNF est une *instance* du problème SAT.
On appelle *configuration* (ou interprétation) une affectation de valeurs aux variables booléennes. Une *solution* (ou modèle) est une configuration telle que la formule soit vraie. Une instance est dite *satisfiable* s'il existe au moins une solution. Sinon l'instance est dite *insatisfiable*. Résoudre l'instance, c'est dire si elle est satisfiable ou non.
Les instances 3-SAT sont les instances du problème SAT dont chaque clause comporte au plus 3 littéraux. Il a été montré que le problème 3-SAT est NP-complet [Coo71].

2. Instances 3-SAT et seuil

Dans le cadre de la recherche expérimentale, on utilise souvent des instances de problème 3-SAT produites par le générateur aléatoire standard [CS88]. Ce dernier construit chaque clause de l'instance en choisissant 3 variables différentes de manière aléatoire, par un tirage équiprobable, et en niant chaque variable avec une probabilité $1/2$.
Pour les instances 3-SAT aléatoires, il semble y avoir un phénomène de seuil quand le rapport $\frac{nombre\ de\ clauses}{nombre\ de\ variables}$ est proche de 4.25. Avant cette valeur, les instances sont presque toutes satisfiables, alors que c'est l'inverse après. Au niveau du seuil, environ la moitié des instances est satisfiable. Il a été montré que ces instances sont les plus difficiles à résoudre en moyenne, aussi bien pour les algorithmes systématiques que pour les algorithmes de recherche locale [DABC96].

3. Walksat

Pour notre étude, nous avons choisi l'algorithme Walksat avec son heuristique de base (best). Le principe de fonctionnement de l'algorithme est le suivant :

choisir aléatoirement une configuration Cfg de l'espace $\mathcal{E}$
flip $\leftarrow$ 1
tant que ((Cfg n'est pas solution) **ET** (flip < MAXFLIP)) **faire**
    choisir aléatoirement une clause $C = v_1 \vee v_2 \vee v_3$ falsifiée par Cfg
        soit Cfg_1, Cfg_2 et Cfg_3 les 3 configurations telles que Cfg et Cfg_i
        ne diffèrent que par la valeur de la variable v_i
    à l'aide de l'heuristique, choisir une configuration Cfg' parmi Cfg_1, Cfg_2, Cfg_3
    $Cfg \leftarrow Cfg'$
    flip $\leftarrow$ flip + 1
ftq

Pour une instance de problème donnée, ce processus est répété MAXTRIES fois si aucune solution n'est trouvée. Il est à noter que si l'instance est satisfiable, lorsque MAXFLIP

tend vers l'infini, la probabilité que le processus trouve une solution à chaque exécution tend vers 1 [H.H99]. Le lecteur intéressé pourra trouver une description détaillée de l'heuristique "best", ainsi que d'autres heuritiques de Walksat, dans [MSK97].
Chaque exécution du processus correspond à une *trajectoire* dans le paysage de recherche. Nous appellerons *difficulté* d'une instance pour Walksat la longueur moyenne des trajectoires nécessaires au processus pour trouver une solution.

III. L'attraction

1. Indice d'attraction

De manière intuitive, l'attraction est la faculté qu'a le processus de recherche, étant sur une configuration, de se diriger vers une configuration particulière.
Plus spécifiquement :
Soit M_j un point de l'espace $\mathcal{E}$; nous appellerons *indice d'attraction* de M_j la valeur

$$I(M_j) = \frac{\sum_{i=1,2,\ldots,j-1,j+1,\ldots,2^n} \frac{p_{ij}}{p^a_{ij}}}{2^n-1},$$

où n est le nombre de variables, p_{ij} la probabilité qu'a le processus en se trouvant sur M_i de se rapprocher de M_j, et p^a_{ij} la probabilité p_{ij} pour un processus purement aléatoire.
Remarque: $p^a_{ij} = \frac{d_{ij}}{n}$ où d_{ij} est la distance entre les configurations M_i et M_j et n le nombre de variables de l'instance. L'indice d'attraction est donc directement lié à la distance entre les configurations. S'il existe deux configurations M_{i_1} et M_{i_2} telles que $p_{i_1j} = p_{i_2j}$, la configuration la plus proche de M_j aura plus de poids dans le calcul de $I(M_j)$. Ceci traduit le fait que plus le processus est loin d'une configuration particulière, plus l'éventualité qu'il puisse s'éloigner de cette configuration au cours de la recherche est élevée.
Il est à noter enfin que l'indice d'attraction dépend de la distance et de l'algorithme choisis.

Nous avons mesuré l'indice d'attraction des configurations pour une centaine d'instances de problèmes. Pour chaque instance, nous avons mesuré cet indice pour deux groupes de configurations : d'une part, pour une centaine de configurations choisies aléatoirement (de manière équiprobable) dans l'espace de recherche, d'autre part, pour une centaine de configurations choisies aléatoirement (de manière équiprobable) parmi les solutions. L'indice de chaque configuration a été calculé à partir de 100 configurations choisies aléatoirement (de manière équiprobable) dans l'espace des configurations.
La distance utilisée pour le calcul de p^a_{ij} est la distance de Hamming (*i.e.* la distance entre M_i et M_j est le nombre de variables ayant une valeur différente dans M_i et dans M_j).
Les résultats en moyenne sont présentés figure 1.

On constate que les solutions ont un indice d'attraction plus élevé que celui des configurations aléatoires.

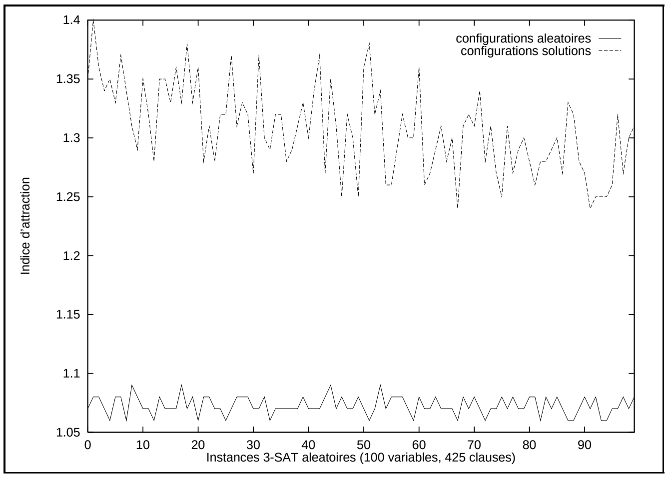

FIG. 1 – *Indice d'attraction des configurations.*
Instances 3-SAT aléatoires (100 variables, 425 clauses) classées par difficulté croissante.

2. Coefficient d'attraction

L'indice d'attraction est calculé sur tout l'espace de recherche, aucune configuration n'étant prépondérante par rapport aux autres. Or le processus de recherche ne parcourt pas l'espace des configurations de manière uniforme. Pour tenir compte de cette particularité, nous avons introduit la notion de *coefficient d'attraction*, défini par :
Soit M_j un point de l'espace $\mathcal{E}$; le coefficient d'attraction de M_j est la valeur

$$C(M_j) = \frac{\sum_{i=1,2,\ldots,j-1,j+1,\ldots,2^n} Prop(M_i).\frac{p_{ij}}{p^a_{ij}}}{2^n-1},$$

où n est le nombre de variables de l'instance, p_{ij} la probabilité qu'a le processus en se trouvant sur M_i de se rapprocher de M_j, p^a_{ij} la probabilité p_{ij} pour un processus purement aléatoire et $Prop(M_i)$ un terme correctif pour tenir compte des spécificités du processus.
De manière plus précise,

$$Prop(M_i) = \sum_t Proba(t).\frac{nombre\ de\ fois\ que\ M_i\ apparait\ dans\ t}{longueur\ de\ t},$$

où t représente une trajectoire quelconque dans le paysage et Proba(t) la probabilité que t soit réalisée.
Remarque : cette définition n'est applicable que si les trajectoires ont une longueur bornée, ce qui est le cas en pratique. Ceci implique que le processus comprenne un nombre fini de trajectoires. Avec l'algorithme Walksat et l'heuristique *best*, ces trajectoires sont de deux types : celles qui aboutissent à une solution avant d'atteindre la longueur limite et celles qui atteignent la longueur limite.

Il n'est pas possible expérimentalement de faire toutes les trajectoires et donc de calculer le coefficient d'attraction. Toutefois, nous estimons qu'une bonne approximation de ce coefficient peut être obtenu à partir d'un nombre significatif de trajectoires.

IV. Expérimentations

Nos expérimentations portent sur 100 instances 3-SAT aléatoires satisfiables au seuil (instances de 100 variables et 425 clauses). Les instances sont classées par ordre croissant de difficulté pour Walksat. La difficulté des instances est comprise entre 341.26 et 17000.44 (pour chaque instance, il s'agit d'une moyenne calculée à l'aide de 10000 trajectoires). Chaque mesure d'attraction correspond à une moyenne effectuée sur 100 trajectoires.

1. Première expérimentation

Nous avons calculé le coefficient d'attraction d'un certain nombre de configurations particulières des paysages de recherche.
Plus spécifiquement, pour chaque trajectoire, nous avons retenu la configuration initiale, celle atteinte après 100 flips, après 500 flips (si la recherche n'est pas terminée), après 1000 flips (si la recherche n'est pas terminée) et la solution trouvée. Nous avons mesuré le coefficient d'attraction de chaque configuration retenue en prenant en compte les 100 trajectoires effectuées, puis nous avons calculé l'attraction moyenne par type de configurations. Les résultats sont présentés en annexe.
Les problèmes étant ordonnés par ordre croissant de difficulté pour l'algorithme, nous avons effectué une moyenne par ensemble de 10 instances de problèmes pour mettre en évidence la relation qui semble exister entre progression de l'algorithme et coefficient d'attraction des configurations rencontrées. Les résultats sont présentés figure 2.

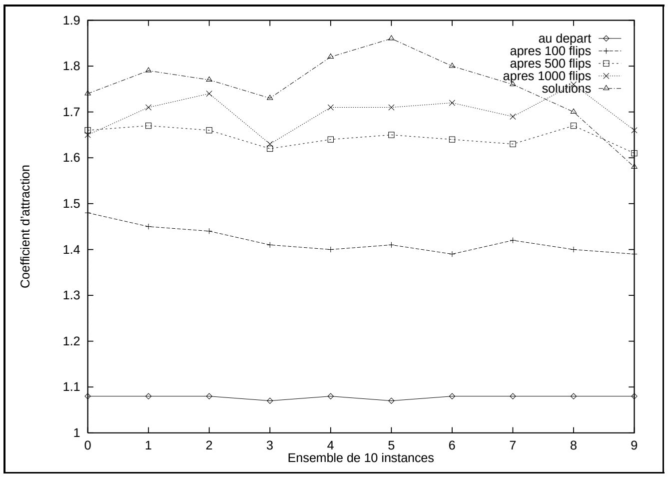

FIG. 2 – *Coefficient d'attraction de points des trajectoires.*
Instances 3-SAT aléatoires (100 variables, 425 clauses) classées par difficulté croissante.
Chaque point représente une moyenne pour 10 instances consécutives.

On constate que l'algorithme atteint au cours de la recherche des configurations qui ont des coefficients d'attraction de plus en plus élevés. Cette progression est plus importante en début de recherche qu'en fin.

On constate également que pour les problèmes les plus difficiles, certaines configurations ont un coefficient d'attraction plus élevé que celui des solutions.

2. Expérimentation avancée

Afin de confirmer l'existence de configurations plus attractives que les solutions, nous avons construit un ensemble F de configurations spécifiques.
Tout d'abord, pour une instance donnée I, à partir de l'expérience précédente, nous avons mémorisé les trois clauses qui ont été le plus souvent falsifiées. Ensuite, nous avons construit l'instance de problème I' à partir de I en retirant ces trois clauses. Enfin, de manière équiprobable, nous avons choisi 100 configurations solutions de I' qui ne sont pas solutions de I. Pour I, ces configurations de l'ensemble F sont donc d'altitude 1, 2 ou 3.
Nous avons mesuré le coefficient d'attraction de chacune de ces configurations en reprenant les trajectoires de l'expérience précédente. Une moyenne est ensuite calculée pour l'ensemble F. Les résultats sont présentés dans l'annexe.
Comme auparavant, nous avons calculé une moyenne par ensemble de 10 instances de problème. Les résultats sont présentés figure 3.

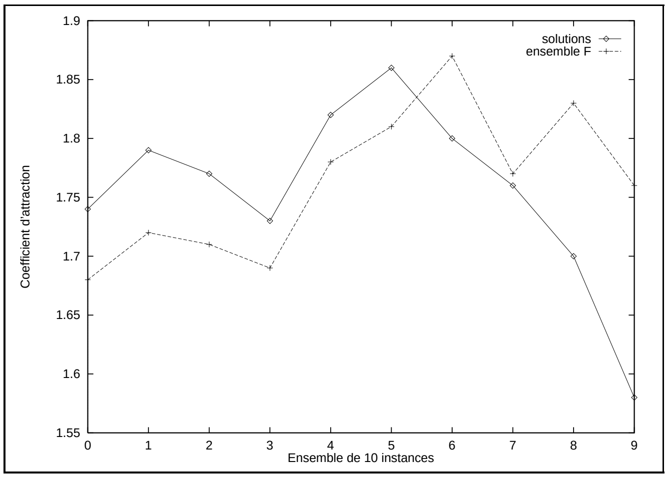

FIG. 3 – *Coefficient d'attraction des configurations de F.*
Instances 3-SAT aléatoires (100 variables, 425 clauses) classées par difficulté croissante.
Chaque point représente une moyenne pour 10 instances consécutives.

On constate que les configurations de l'ensemble F ont un coefficient d'attraction comparable plus faible que celui des solutions pour les instances les plus faciles, et nettement supérieur à celui des solutions pour les instances les plus difficiles.

V. Résumé et perspectives

Dans le but de mieux comprendre les mécanismes de recherche des algorithmes de recherche locale stochastique, nous avons introduit deux nouvelles définitions : celle de l'indice d'attraction et celle du coefficient d'attraction d'une configuration.
Une étude empirique a été réalisée sur des problèmes 3-SAT aléatoires standard à 100 variables au seuil. Les résultats obtenus montrent que l'indice d'attraction et le coefficient d'attraction sont des paramètres permettant de *discriminer* certaines régions des paysages de recherche. Ainsi, nous avons mis en évidence que les trajectoires de Walksat (avec l'heuristique "best") passent par des configurations qui ont un coefficient d'attraction de plus en plus élevé. Ceci dénote une *progression* qui, très nette en début de recherche, s'atténue ensuite. Notre étude a aussi révélé qu'il existe des configurations qui ne sont pas solution et qui ont un coefficient d'attraction très élevé, parfois supérieur à celui des solutions. Quelle influence ces configurations attractives "parasites" ont-elles sur le processus de recherche ? Quelle est leur dispersion dans l'espace de recherche (sont-elles groupées ou dispersées, proches ou loin des solutions,...) ? Existe-t-il d'autres ensembles ou d'autres configurations ayant la même caractéristique ? Les questions ouvertes sont nombreuses. Répondre à certaines d'entre-elles permettrait peut-être de mieux comprendre la "difficulté" de certaines instances, et les résultats obtenus pourraient être exploités dans d'autres problèmes, comme #SAT par exemple.

Références

[Cat92] Catoni. Rough large deviation estimates for simulated annealing : application to exponential schedules. In *The Annals of Probability*, volume 20, pages 1109–1146, 1992.

[Coo71] Stephen A. Cook. The complexity of theorem proving procedures. In *Conference Records of Third Annual ACM Symposium on Theory of Computing*, pages 151–158, 1971.

[CS88] V. Chvátal and E. Szemerédi. Many hard example for resolution. *Journal of the ACM*, pages 759–768, 1988.

[DABC96] O. Dubois, P. Andre, Y. Boufkhad, and J. Carlier. Sat versus unsat. In *Clique, Coloring and Salisfiability: Second DIMACS implementation challenge, Volume 26 of DIMACS series in Discrete Mathematics and Computer Science*, pages 415–436. American Mathematical Society, 1996.

[DP60] Davis and Putnam. A computing procedure for quantification theory. *J. ACM*, 7:201–215, 1960.

[FRPT97] Fonlupt, Robilliard, Preux, and Talbi. Fitness landscapes and performance of meta-heuristics. In *2nd International Conference on Metaheuristics*, 1997.

[Her90] Werra Hertz. The tabu search metaheuristic : how we used it. *The annals of Mathematics and Artificial Intelligence*, 1:111–121, 1990.

[H.H99] H.H.Hoos. On the run-time behaviour of stochastic local search algorithms for sat. In *Proceedings of AAAI 99*, pages 661–666, 1999.

[Jon95] Jones. *Evolutionary Algorithms, Fitness Landscapes and Search.* PhD thesis, University of New Mexico, 1995.

[mLA97] Chu min Li and Anbulagan. Heuristic based on unit propagation for satisfiability problems. In *Proceedings of IJCAI 97*, pages 366–371, 1997.

[MSK97] D. McAllester, B. Selman, and H. Kautz. Evidence for invariants in local search. In *Proceedings of AAAI 97*, pages 321–326, 1997.

[Sid99] Bailleux Sidaner, Chabrier. Etude du paysage en fin de recherche pour l'algorithme walksat. In *JNPC'99*, pages 77–84, 1999.

[SKC94] Bart Selman, Henry Kautz, and Bram Cohen. Noise strategies for improving local search. In *Proceedings of AAAI 94*, pages 337–343, 1994.

[SM92] Levesque Selman and Mitchell. A new method for solving hard satisfiability problems. In *Proceedings of AAAI 92*, pages 440–446, 1992.

[VL87] Aarts Van Laarhoven. Simulated annealing: theory and applications. In *Kluwer Academic Publisher*, 1987.

Annexe

Le tableau suivant présente les résultats obtenus lors de nos expérimentations.
La première colonne contient le numéro de l'instance, la deuxième sa difficulté. Les troisième et quatrième colonnes comprennent l'indice d'attraction moyen pour 100 configurations choisies aléatoirement et pour 100 solutions. Les colonnes 5 à 9 contiennent le coefficient d'attraction moyen calculé pour les configurations rencontrées par l'algorithme au départ, après 100 flips, après 500 flips, après 1000 flips et pour les solutions trouvées. La dernière colonne contient le coefficient d'attraction moyen de l'ensemble F.
Chaque mesure d'attraction a été réalisée à partir de 100 trajectoires dans l'espace de recherche. La difficulté de chaque instance est une moyenne calculée à partir de 10000 trajectoires indépendantes.

Num.	Difficulté	Ind. Att.		Coeff. Att.					
		aléat.	solut.	0	+ 100	+ 500	+ 1000	solutions	ens. F
00	341.26	1.07	1.35	1.08	1.51	1.66	1.38	1.77	1.70
01	360.58	1.08	1.40	1.09	1.55	1.70	1.68	1.80	1.70
02	361.35	1.08	1.36	1.08	1.46	1.70	1.75	1.68	1.65
03	392.64	1.07	1.34	1.08	1.47	1.65	1.61	1.75	1.67
04	422.16	1.06	1.35	1.09	1.51	1.68	1.72	1.67	1.61
05	525.50	1.08	1.33	1.08	1.45	1.60	1.62	1.70	1.65
06	570.77	1.08	1.37	1.08	1.51	1.74	1.77	1.84	1.80
07	624.88	1.06	1.34	1.07	1.44	1.61	1.62	1.65	1.63
08	628.44	1.09	1.31	1.07	1.41	1.60	1.69	1.70	1.66
09	655.43	1.08	1.29	1.09	1.48	1.66	1.67	1.83	1.69
10	743.03	1.07	1.35	1.07	1.44	1.68	1.83	1.77	1.75
11	748.14	1.07	1.32	1.08	1.41	1.55	1.53	1.68	1.60
12	777.38	1.06	1.28	1.07	1.41	1.75	1.83	1.88	1.83
13	812.47	1.08	1.35	1.08	1.49	1.74	1.72	1.84	1.74
14	822.47	1.07	1.35	1.08	1.44	1.58	1.61	1.64	1.60
15	869.39	1.07	1.33	1.08	1.45	1.69	1.74	1.63	1.76
16	952.82	1.07	1.36	1.07	1.48	1.69	1.71	1.86	1.67
17	1046.63	1.09	1.33	1.08	1.42	1.60	1.62	1.79	1.64
18	1072.68	1.07	1.38	1.08	1.51	1.80	1.87	1.99	1.94
19	1084.89	1.08	1.33	1.08	1.46	1.64	1.66	1.77	1.70
20	1159.97	1.06	1.36	1.08	1.47	1.74	1.91	1.78	1.86
21	1214.33	1.08	1.28	1.08	1.33	1.56	1.59	1.82	1.69
22	1230.74	1.08	1.31	1.09	1.45	1.68	1.76	1.83	1.74
23	1231.35	1.07	1.28	1.08	1.48	1.77	1.96	1.91	1.74
24	1234.34	1.07	1.32	1.08	1.47	1.67	1.68	1.65	1.71
25	1286.75	1.06	1.32	1.07	1.47	1.79	1.83	1.73	1.76
26	1340.32	1.07	1.37	1.08	1.47	1.65	1.71	1.77	1.67
27	1342.93	1.08	1.31	1.07	1.37	1.51	1.57	1.78	1.60
28	1380.55	1.08	1.33	1.07	1.45	1.61	1.72	1.71	1.77
29	1428.61	1.08	1.32	1.07	1.41	1.62	1.71	1.73	1.52
30	1440.11	1.07	1.27	1.07	1.41	1.77	1.77	2.04	1.98
31	1461.80	1.07	1.37	1.07	1.48	1.68	1.72	1.66	1.66
32	1463.45	1.08	1.30	1.08	1.39	1.57	1.62	1.57	1.65
33	1510.10	1.06	1.29	1.07	1.38	1.53	1.57	1.74	1.75
34	1517.37	1.07	1.32	1.08	1.40	1.53	1.55	1.63	1.63
35	1520.48	1.07	1.32	1.08	1.40	1.54	1.55	1.65	1.42
36	1531.42	1.07	1.28	1.06	1.36	1.54	1.54	1.68	1.61
37	1560.51	1.07	1.29	1.07	1.46	1.65	1.63	1.83	1.73
38	1614.28	1.07	1.31	1.06	1.40	1.63	1.70	1.72	1.69
39	1629.77	1.08	1.33	1.08	1.46	1.71	1.70	1.75	1.74
40	1638.85	1.07	1.30	1.08	1.39	1.64	1.77	2.17	1.97
41	1661.53	1.07	1.34	1.07	1.47	1.89	2.05	2.16	1.95
42	1721.39	1.07	1.37	1.08	1.38	1.50	1.52	1.63	1.56
43	1723.38	1.08	1.27	1.10	1.41	1.76	1.82	2.12	1.93
44	1727.94	1.09	1.35	1.08	1.45	1.82	1.95	1.88	2.06
45	1782.66	1.07	1.31	1.08	1.36	1.51	1.55	1.49	1.60
46	1865.56	1.08	1.25	1.07	1.34	1.47	1.50	1.47	1.46
47	1921.12	1.07	1.32	1.08	1.43	1.65	1.66	1.78	1.69
48	1942.52	1.07	1.30	1.07	1.42	1.65	1.71	1.82	1.85
49	1953.74	1.08	1.25	1.07	1.34	1.53	1.61	1.69	1.73

Num.	Difficulté	Ind. Att.		Coeff. Att.					
		aléat.	solut.	0	+ 100	+ 500	+ 1000	solutions	ens. F
50	1981.99	1.07	1.36	1.07	1.39	1.58	1.58	1.84	1.72
51	2025.73	1.06	1.38	1.08	1.54	1.96	1.96	2.77	1.97
52	2030.47	1.07	1.32	1.07	1.39	1.66	1.74	1.80	1.85
53	2055.24	1.09	1.34	1.08	1.40	1.60	1.64	1.70	1.80
54	2056.65	1.07	1.26	1.07	1.36	1.58	1.57	1.67	1.75
55	2161.29	1.08	1.26	1.07	1.37	1.55	1.56	1.64	1.75
56	2170.41	1.08	1.29	1.07	1.42	1.76	1.94	2.23	2.04
57	2184.76	1.08	1.32	1.07	1.47	1.79	1.94	1.89	2.02
58	2214.86	1.07	1.30	1.08	1.37	1.50	1.52	1.55	1.58
59	2286.75	1.06	1.30	1.08	1.38	1.55	1.66	1.53	1.63
60	2354.92	1.08	1.36	1.08	1.42	1.73	1.88	1.88	1.72
61	2394.16	1.07	1.26	1.08	1.38	1.67	1.75	1.76	1.81
62	2425.59	1.07	1.27	1.07	1.36	1.61	1.68	1.71	2.01
63	2712.86	1.08	1.29	1.08	1.41	1.62	1.69	1.74	1.86
64	2741.31	1.07	1.31	1.08	1.40	1.66	1.71	2.10	2.06
65	2792.70	1.07	1.28	1.07	1.42	1.64	1.73	1.89	1.91
66	2795.97	1.07	1.30	1.08	1.35	1.49	1.52	1.52	1.52
67	2890.40	1.06	1.24	1.07	1.36	1.74	1.82	1.74	2.20
68	2896.18	1.08	1.31	1.07	1.39	1.62	1.67	1.88	1.79
69	3042.49	1.07	1.32	1.09	1.44	1.66	1.72	1.82	1.82
70	3067.76	1.08	1.31	1.09	1.41	1.59	1.63	1.98	1.73
71	3091.62	1.07	1.34	1.07	1.43	1.61	1.70	1.68	1.82
72	3125.48	1.06	1.28	1.08	1.44	1.62	1.67	1.76	1.68
73	3324.67	1.07	1.31	1.08	1.43	1.65	1.72	1.68	1.87
74	3400.89	1.07	1.27	1.08	1.35	1.49	1.50	1.42	1.52
75	3741.97	1.08	1.25	1.10	1.41	1.61	1.72	1.88	1.84
76	3771.05	1.07	1.31	1.09	1.46	1.68	1.74	1.85	1.69
77	3998.52	1.08	1.27	1.07	1.42	1.65	1.73	1.70	1.81
78	4210.43	1.07	1.29	1.08	1.41	1.62	1.68	1.75	1.72
79	4277.73	1.07	1.30	1.09	1.45	1.76	1.85	1.89	1.98
80	4469.04	1.08	1.28	1.08	1.38	1.55	1.58	1.61	1.56
81	4506.26	1.08	1.26	1.08	1.57	2.15	2.34	2.20	2.25
82	4596.32	1.06	1.28	1.09	1.40	1.62	1.64	1.80	1.64
83	4701.03	1.08	1.28	1.06	1.38	1.75	1.83	1.63	1.96
84	5384.20	1.07	1.29	1.08	1.35	1.51	1.57	1.66	1.63
85	5861.22	1.08	1.30	1.07	1.37	1.69	1.79	1.50	1.87
86	6268.62	1.07	1.27	1.08	1.39	1.63	1.65	1.55	1.70
87	6394.94	1.06	1.33	1.09	1.44	1.68	1.74	2.01	1.90
88	6842.41	1.06	1.32	1.08	1.44	1.61	1.72	1.50	2.10
89	7189.87	1.07	1.28	1.06	1.32	1.56	1.77	1.58	1.68
90	7364.18	1.08	1.27	1.08	1.41	1.67	1.74	1.82	1.77
91	8162.69	1.07	1.24	1.08	1.36	1.54	1.57	1.43	1.67
92	8879.06	1.08	1.25	1.09	1.33	1.51	1.57	1.55	1.86
93	9746.80	1.06	1.25	1.08	1.39	1.59	1.62	1.52	1.65
94	9883.12	1.06	1.25	1.08	1.34	1.47	1.48	1.43	1.86
95	10090.92	1.07	1.26	1.09	1.50	1.89	1.97	1.78	1.88
96	12062.58	1.07	1.32	1.09	1.44	1.69	1.75	1.59	1.83
97	12102.93	1.08	1.27	1.08	1.34	1.54	1.58	1.51	1.60
98	14398.51	1.07	1.30	1.08	1.38	1.55	1.62	1.58	1.70
99	17000.44	1.08	1.31	1.07	1.42	1.62	1.71	1.59	1.78