



HAL
open science

Estimations de satisfaisabilité

Thomas Hugel

► **To cite this version:**

Thomas Hugel. Estimations de satisfaisabilité. Mathématiques [math]. Université Paris-Diderot - Paris VII, 2010. Français. NNT: . tel-00582571

HAL Id: tel-00582571

<https://theses.hal.science/tel-00582571>

Submitted on 1 Apr 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Université Paris Diderot (Paris 7)

École Doctorale de Sciences Mathématiques de Paris Centre (ED386)

Thèse de Doctorat

Informatique

Thomas HUGEL

Estimations de satisfaisabilité

Thèse encadrée par Yacine BOUFKHAD

Soutenue publiquement le 7 décembre 2010 devant le jury composé de :

Mme. Nadia Creignou, Rapporteur

M. Lefteris Kirousis, Rapporteur

M. Yacine Boufkhad

M. Arnaud Durand

M. Michel Habib

M. Marc Mézard

Résumé

Le problème de satisfaisabilité booléenne 3-SAT est connu pour présenter un phénomène de seuil en fonction du quotient entre le nombre de clauses et le nombre de variables. Nous donnons des estimations de la valeur de ce seuil au moyen de méthodes combinatoires et probabilistes : la méthode du premier moment et la méthode du second moment. Ces méthodes mettent en jeu des problèmes d'optimisation sous contraintes et nous amènent à employer de façon intensive la méthode des multiplicateurs de Lagrange.

Nous mettons en œuvre une forme pondérée de la méthode du premier moment sur les affectations partielles valides de Maneva ainsi que des variantes. Cela nous conduit à élaborer une pondération générale pour les problèmes de satisfaction de contraintes qui soit compatible avec la méthode du premier moment. Cette pondération est constituée d'une graine et d'un répartiteur, et nous permet d'obtenir une pondération des affectations partielles valides meilleure que celle de Maneva. Nous comparons aussi dans certains cas les performances de la pondération et de l'orientation de l'espace des solutions des problèmes de satisfaction de contraintes relativement à la méthode du premier moment.

Nous développons la première sélection non uniforme de solutions pour majorer le seuil de 3-SAT et nous montrons sa supériorité sur ses prédécesseurs.

Nous construisons un cadre général pour appliquer la méthode du second moment à k-SAT et nous discutons des conditions qui la font fonctionner. Nous faisons notamment fonctionner la méthode du second moment sur les solutions booléennes et sur les impliquants. Nous étendons cela au modèle distributionnel de k-SAT.

Mots-clés : Satisfaisabilité, Problème de Satisfaction de Contraintes, Seuil, Transition de Phase, Méthode du Premier Moment, Méthode du Second Moment, Multiplicateurs de Lagrange, Orientation, Pondération, Répartiteur, Sélection non Uniforme

Laboratoire d'accueil

Laboratoire d'Informatique Algorithmique : Fondements et Applications (LIAFA)
CNRS UMR 7089, Université Paris VII - Denis Diderot
175 rue du Chevaleret
75013 Paris - France

Esquisse

Le problème de satisfaisabilité booléenne 3-SAT est NP-complet, ce qui signifie qu'il est algorithmiquement difficile. Sur des instances aléatoires on peut observer un phénomène de transition de phase pour la satisfaisabilité à une valeur du rapport $\frac{\text{nombre de clauses}}{\text{nombre de variables}}$ voisine de 4.25. Il se trouve que c'est au voisinage de cette transition de phase que le problème semble le plus difficile algorithmiquement.

La localisation du seuil est un problème ouvert depuis une vingtaine d'années. Il a été démontré que ce seuil se situe entre 3.5 et 4.5. Notre objectif est de contribuer à resserrer l'encadrement de ce seuil. Pour ce faire nous décortiquons le fonctionnement de deux méthodes probabilistes :

1. la méthode du premier moment, qui permet d'obtenir des majorants du seuil ;
2. la méthode du second moment, qui permet éventuellement d'obtenir des minorants du seuil.

Cela nous amène à utiliser de façon récurrente la méthode des multiplicateurs de Lagrange. Ce travail est au croisement de la combinatoire, des probabilités et de l'optimisation.

Faire des expériences avec **SATLab**

Puisque nous travaillons sur k-SAT aléatoire et que le comportement typique des instances est difficile à calculer au moyen d'outils mathématiques, nous avons entrepris de développer un logiciel (que nous avons appelé **SATLab**) qui puisse nous permettre d'avoir un aperçu direct de ce qui se passe dans les formules et leur solutions. Ce logiciel réalise les tâches répétitives et pénibles consistant à tirer des formules au hasard avec les paramètres voulus, appeler dessus un "solver" (programme résolvant l'instance) et rassembler les résultats dans un seul et même graphique.

La première observation que nous faisons est le fameux pic de complexité autour du seuil. Ensuite nous allons un peu plus loin dans l'étude de la difficulté de modèles plantés (c'est-à-dire où une solution a été cachée), en les comparant les uns aux autres et en mettant en évidence des produits dérivés de la difficulté algorithmique tels que les variables gelées et les "cores" (noyaux). Cette petite étude expérimentale permet de rendre compte des possibilités qu'offre notre logiciel **SATLab**.

Nous revenons plus tard à **SATLab**, à l'occasion d'une discussion sur l'opportunité de la méthode du second moment.

Méthode du premier moment pondéré

La méthode du premier moment est le principal outil utilisé pour majorer le seuil de 3-SAT.

Nous utilisons les affectations partielles valides et le théorème de préservation de poids de Maneva, Mossel & Wainwright - 2007 [MMW07] pour construire d'autres systèmes similaires et leur appliquer la méthode du premier moment.

Une affectation partielle valide est une affectation sur le domaine $\{0, 1, *\}$ telle qu'il n'y ait pas de clause dans la formule ayant (tous ses littéraux faux) ou (tous ses littéraux faux sauf un à $*$). Dans une affectation partielle valide, nous appelons étoilable une variable qui a la valeur 0 ou 1 et qui, si on la mettait à la valeur $*$, donnerait une autre affectation partielle valide. Étant donné deux réels positifs ω_0 et ω_* , le poids d'une affectation partielle valide σ est $W(\sigma) = \omega_0^{n_0(\sigma)} \omega_*^{n_*(\sigma)}$, où $n_0(\sigma)$ est le nombre de variables étoilables dans σ et $n_*(\sigma)$ est le nombre de variables étoilées dans σ . Le théorème de préservation de poids de [MMW07, MS08, AM09] affirme que si σ est une solution booléenne et que $\omega_0 + \omega_* = 1$, alors $\sum_{\tau \leq \sigma} W(\tau) = 1$. Par conséquent, le poids W convient parfaitement à la méthode du premier moment. Par un calcul formel et numérique, nous obtenons un majorant de 4.883.

La propriété de validité de Maneva élimine les clauses de type FFF et FF*, mais qu'advient-il si l'on choisit d'éliminer FFF, FF*, F** et ***, ou tout simplement FFF? Quel est le majorant obtenu dans chacun de ces cas? Nous donnons les calculs mathématiques correspondants et nous déterminons numériquement que nous ne pouvons pas obtenir de majorant meilleur que 4.866 dans ce cadre.

Alors nous faisons une autre tentative d'utiliser les affectations partielles valides, en éliminant celles qui n'ont aucune chance de se trouver sous un NPS (un NPS est une solution où toutes les variables libres sont à 1). Les NPS ont été présentés en 1997 par Dubois & Boufkhad [DB97] et par Kirousis, Kranakis & Krizanc [KKK97]. Leur décompte a permis à [DB97] d'obtenir un majorant de 4.643, contre 4.667 pour [KKK97]. Il s'avère qu'en cherchant à combiner les affectations partielles valides aux NPS, nous obtenons le meilleur majorant dans le cas particulier où il n'y a plus d'étoiles, ce qui correspond à compter les NPS seuls (le majorant correspondant étant 4.643).

C'est ce qui nous a amenés à nous interroger sur la performance de la pondération par rapport à l'orientation des solutions.

Pondérer et ordonner les solutions

Dans ce chapitre nous considérons les problèmes de satisfaction de contraintes généraux (CSP). Notre objectif est de définir sur l'espace de leurs solutions des pondérations et des orientations qui soient compatibles avec la méthode du premier moment.

- les orientations de l'espace des solutions doivent présenter des éléments minimaux, car ce sont eux qui sont comptés par la méthode du premier moment ;
- les pondérations de l'espace des solutions doivent avoir un poids global d'au moins 1 ;
- les orientations tout comme les pondérations doivent pouvoir être calculées *localement*, c'est-à-dire en tenant compte seulement du voisinage immédiat de la solution considérée, afin que le premier moment puisse tout bonnement être mis en équations !

Ainsi nous définissons l'orientation suivante : pour chaque variable x , nous munissons le domaine D d'un ordre total strict $<_{F,x}$. À partir de là, pour comparer deux solutions voisines, il suffit de comparer leurs valeurs sur la variable x où elles diffèrent.

Pondérer les solutions des CSP en accord avec les contraintes susmentionnées n'est pas si simple. Nous avons recours à une graine de poids $s_F(x, a)$ et à un répartiteur $d_F(x, a)$ définis pour chaque variable x et chaque valeur a . Ensuite, étant donné une clique de voisinage dans l'espace des solutions, c'est-à-dire un ensemble de solutions différant seulement sur la même

variable x , nous donnons à chacune des solutions présentes σ un poids $w_F(\sigma, x)$ égal à la graine $s_F(x, \sigma(x))$, puis nous répartissons entre elles les graines de poids des solutions absentes au moyen du répartiteur d_F . Finalement, le poids d'une solution est le produit sur x des poids $w_F(\sigma, x)$.

Nous démontrons que le système de poids ci-dessus a bien un poids global d'au moins 1, en appliquant un théorème plus général de conservation de poids, par lequel nous établissons des conditions suffisantes pour qu'une pondération ait un poids global d'au moins 1.

La portée de notre théorème de conservation de poids est effectivement plus générale, dans la mesure où nous l'utilisons aussi pour valider un autre système de poids que nous définissons sur les affectations partielles valides de Maneva (en les considérant comme des solutions d'un CSP sur le domaine $\{0, 1, *\}$). L'intérêt de notre nouveau système de poids est qu'il est plus léger que celui de Maneva, tout en ayant bien un poids global d'au moins 1. Nous accomplissons alors le calcul du premier moment avec notre tout nouveau système de poids sur les affectations partielles valides. Avec la pondération de Maneva, le majorant obtenu était de 4.883, alors que notre nouvelle pondération nous donne dans le meilleur réglage un majorant de 4.643. Donc notre pondération est meilleure, mais pas fracassante. En effet, le lecteur attentif aura remarqué que le meilleur réglage de notre pondération revient à compter les NPS!

Tout cela nous amène à nous poser la question suivante : quel est donc l'intérêt de pondérer, si le meilleur réglage des poids revient toujours à orienter? Nous répondons à cette question dans deux cas particuliers :

1. pondérations homogènes (i.e. lorsque $d_F = s_F$) : étant donné une pondération homogène, nous montrons qu'il existe toujours une orientation au moins aussi bonne ;
2. sur un ensemble d'instances de CSP clos par renommage (i.e. un ensemble clos par permutation des valeurs du domaine) : nous montrons que sur un tel ensemble, les pondérations et les orientations sont équivalentes en moyenne.

Comparer les pondérations et les orientations dans le cas général reste un problème ouvert.

Modèles distributionnels et non uniformité

Puisque pondérer les affectations partielles valides ne semble pas être mieux que compter les NPS, nous avons décidé d'emprunter une autre direction pour obtenir des majorants du seuil de 3-SAT. La première chose que nous eussions à faire était de passer dans le modèle distributionnel (où les occurrences et les signes des variables sont donnés par une certaine distribution de probabilité plutôt que par tirage aléatoire). En effet, ce modèle donne intrinsèquement de meilleurs majorants que le modèle à tirage (voir le tableau 6.2). De plus, le modèle à distribution permet des réglages plus fins parce qu'il donne du contrôle sur le nombre d'occurrences de littéraux.

Nous avons observé que les orientations des espaces de solutions vers des solutions intéressantes (telles que les NPS) qui avaient été mises en œuvre jusqu'alors pour majorer le seuil (4.643 par Dubois & Boufkhad - 1997 [DB97], 4.667 par Kirousis, Kranakis & Krizanc - 1997 [KKK97], 4.506 par Dubois, Boufkhad & Mandler - 2000 [DBM00, DBM03] ou encore 4.490 par Díaz, Kirousis, Mitsche and Pérez-Giménez - 2009 [DKMPG09]) étaient toutes uniformes : toutes les arêtes correspondant à une variable x donnée dans le graphe de Hamming des solutions sont orientées de la même manière, i.e. soit de 0 vers 1, soit de 1 vers 0. Il en va de même pour notre pondération générale des CSP!

Nous cherchons donc à élaborer une orientation non uniforme de l'espace des solutions d'une manière qui soit compatible avec la méthode du premier moment (i.e. garantissant l'existence d'éléments minimaux) mais aussi qui donne de meilleurs majorants que les orientations uniformes connues.

À cet effet, nous utilisons un jeu de 5 nombres par variable et nous l'utilisons pour départager les solutions voisines. Ces 5 nombres renseignent sur la répartition des occurrences vraies et fausses de chaque variable dans chaque type de clause (le type d'une clause étant son nombre de littéraux à vrai : 1, 2 ou 3). De même que [DBM03] orientait les solutions vers une quantité croissante de littéraux à vrai, nous allons partir de cette idée et la pousser plus loin en distinguant les occurrences des littéraux en fonction de leur *état critique*. Notre intuition nous dit que nous devrions sélectionner les solutions qui ont le moins possible d'occurrences critiques de littéraux. Une occurrence de littéral à vrai est dans un état d'autant plus critique que la clause où elle apparaît a plus d'occurrences de littéraux à faux. Une telle propriété est par nature non uniforme.

Nous appliquons notre technique à 4 modèles aléatoires différents de 3-SAT : le modèle standard, un modèle où les occurrences des variables sont équilibrées, un modèle où les signes sont équilibrés et un modèle où à la fois les signes et les variables sont équilibrés (le modèle introduit par Boufkhad, Dubois, Interian & Selman - 2005 [BDIS05]). Sur tous ces modèles, notre approche donne des majorants inférieurs à ceux obtenus par les méthodes antérieures (et uniformes) de sélection.

Il faut dire que dans le modèle standard, notre borne de référence était le 4.506 de [DBM00] ; en effet le 4.490 de [DKMPG09] est dérivé du 4.506 par l'ajout de deux ingrédients supplémentaires : l'élimination des littéraux purs et la typicité des signatures des clauses. Ces deux ingrédients supplémentaires sont en fait d'un intérêt indépendant : ils pourraient être ajoutés à nos calculs, mais le surcoût requis en calculs serait disproportionné au regard du modeste gain auquel on peut prétendre. Nous nous sommes donc focalisés sur la sélection des solutions proprement dite, et nous avons trouvé

- une manière non uniforme de sélectionner des solutions (la première à notre connaissance),
- qui est meilleure que toutes les autres manières connues.

Au niveau des calculs proprement dits, nous aboutissons à un problème classique de maximisation sous contraintes, que nous résolvons classiquement et en suivant la même démarche que [DKMPG09], c'est-à-dire par la méthode des multiplicateurs de Lagrange. Cela nous oblige à prouver que la fonction objectif ne se maximise pas aux frontières du polytope ; de plus nous contrôlons que la solution obtenue est bien un maximum global en balayant ce polytope.

Un cadre général pour la méthode du second moment sur k-SAT

La méthode du second moment n'est pas l'outil principal ayant servi à minorer le seuil de 3-SAT. En effet la plupart des minorants ont été obtenus par l'analyse d'algorithmes trouvant des solutions avec grande probabilité, y compris le meilleur minorant connu (3.52), obtenu par Kaporis, Kirousis & Lalas [KKL06] et par Hajiaghayi & Sorkin [HS03]. Il est particulièrement difficile de faire fonctionner la méthode du second moment sur k-SAT, parce qu'elle requiert de compter un ensemble de solutions de variance faible.

Par conséquent, on peut dire que le travail d'Achlioptas & Peres - 2004 [AP04] est un grand succès dans la mise en œuvre de la méthode du second moment sur k-SAT ; ils établissent un minorant de 2.68 pour 3-SAT et un minorant asymptotiquement quasi-optimal de $2^k \ln 2 - O(k)$ pour k-SAT.

Notre objectif est de tirer le meilleur parti de la méthode du second moment sur k-SAT. À cet effet nous suivons une approche différente de celle d'Achlioptas & Peres. Dans notre cadre de travail, nous sélectionnons les solutions en fonction de la fraction de variables à 1 et des fractions des différents types de clauses (le type d'une clause étant défini par le nombre d'occurrences de littéraux vrais dans cette clause). Ce cadre est suffisamment général pour permettre de traiter les solutions booléennes et les impliquants, entre autres. Néanmoins, le

meilleur réglage des paramètres que nous ayons pu trouver ne nous a pas permis de trouver numériquement un minorant meilleur que 2.83.

La pierre d'achoppement sur laquelle nous avons buté de façon récurrente est ce que nous appelons le point d'indépendance. Ce point correspond aux couples de solutions indépendantes dans le sous-ensemble des solutions sélectionnées. Bien que les solutions soient indépendantes, la proportion des occurrences de littéraux ayant une certaine valeur de vérité peut ne pas être indépendante entre les solutions. Par des tests numériques, nous avons acquis la conviction que la méthode du second moment ne peut pas fonctionner si les valeurs de vérité des occurrences de littéraux ne sont pas indépendantes entre les solutions indépendantes. D'un autre côté, lorsqu'elles sont indépendantes, nous donnons une condition nécessaire pour que la méthode du second moment fonctionne, en utilisant juste l'équivalent exponentiel du second moment en ce point. Cette condition nous dit qu'il faut sélectionner des solutions ayant des surfaces vraie et fausse égales (la surface étant tout simplement le nombre total d'occurrences de littéraux), ce qui est très artificiel compte tenu de ce qu'on peut observer avec **SATLab**. En utilisant cette condition nous avons réussi à faire marcher numériquement la méthode du second moment ; cependant, puisque les minorants que nous obtenons (2.83) sont loin au-dessous du meilleur minorant actuel (3.52), nous n'avons pas pris la peine de donner une preuve détaillée de nos minorants (pour ce faire, l'équivalent exponentiel ne suffirait plus, et les calculs deviendraient terriblement compliqués).

Les conditions très restrictives sur lesquelles nous sommes tombés pour faire marcher la méthode du second moment sont peut-être dues à des faiblesses présentes dans notre cadre de travail. Nous ne prétendons pas que la méthode du second moment soit condamnée à échouer perpétuellement sur **k-SAT**. Nous espérons seulement y apporter notre éclairage.

Second moment sur le modèle distributionnel de **k-SAT**

L'utilisation du modèle distributionnel standard à la place du modèle à tirages standard permet d'obtenir de meilleurs majorants du seuil de satisfaisabilité (cf. tableau 6.2). De plus, nous aimerions avoir plus de contrôle sur les proportions de variables à 1 en fonction du déséquilibre entre les occurrences positives et négatives des variables. En effet, nous pensons qu'une variable a d'autant plus de chances de se retrouver à 1 dans une solution qu'elle a plus d'occurrences positives, et inversement. C'est du moins ce qui semble se produire dans la réalité, comme l'atteste **SATLab**. Toutes ces raisons nous amènent à mettre en œuvre la méthode du second moment sur le modèle distributionnel de **k-SAT**.

Les grandes lignes des calculs sont à peu près les mêmes que sur le modèle à tirages. Une différence majeure cependant est que s'il n'y a pas indépendance des valeurs de vérité des occurrences de littéraux au point d'indépendance, alors le point d'indépendance lui-même viole une des contraintes. Néanmoins, le résultat est le même que dans le modèle à tirages : nous n'arrivons à faire fonctionner la méthode du second moment que lorsqu'il y a indépendance des valeurs de vérité des occurrences de littéraux au point d'indépendance.

Ce calcul est très décevant parce que non seulement il mène au même genre de contraintes que dans le modèle à tirages, contraintes assurément artificielles au regard de ce qu'on peut observer avec **SATLab**, mais en outre il donne la même valeur critique de 2.83 !

Satisfiability Estimations

Abstract

The boolean satisfiability problem 3-SAT is known to exhibit a threshold phenomenon with respect to the clauses to variables ratio. We give some estimations of the location of this threshold through combinatorial and probabilistic methods: the First Moment Method and the Second Moment Method. These methods involve constrained optimization and lead us to extensively use the Lagrange multipliers method.

We implement a weighted form of the First Moment Method on Maneva's valid partial assignments as well as some variants. That leads us to design a general weighting for Constraint Satisfaction Problems compliant with the requirements of the First Moment Method. This weighting is made up of a seed and a dispatcher, and yields as a by-product a better weighting system on valid partial assignments than Maneva's. We also compare in some cases the performances of weighting and ordering the solutions space of Constraint Satisfaction Problems with respect to the First Moment Method.

We exhibit the very first non-uniform selection of solutions for upper-bounding the threshold of 3-SAT and show that it is better than all previous selection methods.

We give a general framework implementing the Second Moment Method on k-SAT and discuss the conditions making the Second Moment Method work in this framework. As applications, we make the Second Moment Method work on boolean solutions and implicants. We extend this to the distributional model of k-SAT.

Keywords: Satisfiability, Constraint Satisfaction Problem, Threshold, Phase Transition, First Moment Method, Second Moment Method, Lagrange Multipliers, Ordering, Weighting, Dispatcher, Non-Uniform Selection

Acknowledgments

“Bless the Lord, O my soul!”

Psalm 103.

First of all I am very indebted to Yacine Boufkhad, who was my supervisor during these four years. He was involved in all the work presented in this thesis. I would like to stress the huge amount of time he spent with me: we racked our brains together for hours and hours trying to step forward. I salute both his patience and his intellectual rigor. Thanks, thanks and thanks again.

I want to thank specially Nadia Creignou and Lefteris Kirousis, who accepted to review this thesis and to come from far away for the defense in Paris. I also thank the other committee members: Arnaud Durand, Marc Mézard, and Michel Habib who kindly accepted to be my official supervisor.

I want to thank LIAFA’s engineers, Houy Kuoy and Laifa Ahmadi. Whenever I had a request for a software they considered it carefully and struggled against the system to make it work correctly.

I want to thank Noëlle Delgado, who was in charge of the administrative stuff, and who was always there when I needed a room to work with Yacine!

I thank some of my friends junior researchers who cheered me up, Alexis Ballier and Julien Robert, as well as some senior researchers, for their encouragements: Nicolas Schabanel, Daniel Hirschhoff, Mathieu Raffinot and Vlady Ravelomanana.

I also thank my young colleagues for bringing some relaxing breaks in my everyday life: Claire David, my elder-sister at LIAFA, the always smiling Christophe Crespelle, the astute Thu-Hien To, the smart Hoang Anh Phan, the crafty Djamel Belazzougui, as well as Eudes Petonnet and Cezara Dragoi.

I thank my Latin American friends of the canteen, among which Cristhian Emmanuel Garay López, Victor Torres Perez, Elhoim Sumano, Daniel Barrera, Lili Guadarrama Bustos and Arturo Giles Flores, who always ate as late and almost as slowly as me.

I thank all my former teachers, especially Mr. Viain, Mr. Bartier, Mr. Fauve, Mrs. Thirode, Mr. Franck Taïeb, Mr. Bernard Randé, Marianne Delorme, Jacques Mazoyer and Pascal Koiran.

Finally I would like to thank my parents and grandparents, who taught me lots of knowledge, and obstinacy as well, and who were always a support for me.

Contents

Contents	13
List of Figures	17
List of Tables	18
1 Introduction	19
1.1 Framework - Random 3-SAT	19
1.1.1 k-CNF SAT	19
1.1.2 Goldberg's Random-Clause-Width Model	20
1.1.3 Drawing Model	20
1.1.4 Distributional Model	20
1.2 Motivations	21
1.2.1 A Universal Model	21
1.2.2 A Difficult Problem	21
1.3 Sharpness of Thresholds	21
1.3.1 Influence and Complexity	22
1.3.2 The Quest for Sharpness	22
1.3.3 Sharpness vs. Complexity	23
1.4 The Solutions Space	23
1.4.1 The Number of k-CNF Functions	23
1.4.2 Clustering	24
1.5 Outline of the Thesis	25
1.5.1 Experimenting through SATLab	25
1.5.2 Weighted First Moment Method	25
1.5.3 Weighting and Ordering Solutions	26
1.5.4 Distributional Models and Non-Uniformity	26
1.5.5 A General Framework for the Second Moment Method on k-SAT	27
1.5.6 Second Moment on Distributional Random k-SAT	28
2 Experimenting through SATLab	29
2.1 A Brief Description of SATLab	29
2.2 Some Noteworthy Observations through SATLab	29
2.2.1 Algorithmic Complexity	29

2.2.2	Some Hard Satisfiable Models	30
I	First Moment	35
3	Introduction to the First Moment Method	37
3.1	How the First Moment Method Works	37
3.2	Use of the First Moment Method for Upper-Bounding the Threshold of k-SAT	38
3.2.1	First Moment of Solutions	38
3.2.2	First Moment of Implicants	39
3.2.3	First Moment of Prime Implicants	39
3.2.4	First Moment of Negatively Prime Solutions	39
4	Weighted First Moment Method	41
4.1	Partial Assignments	41
4.1.1	Directed Graph and Partial Ordering	41
4.1.2	Weight Preservation Theorem	42
4.2	Trivial Kind of Validity: FFF	43
4.3	Generic Calculation for Various Kinds of Validity	44
4.3.1	Application to FFF + FF*	46
4.3.2	Application to FFF + FF* + F**	46
4.3.3	Application to FFF + FF* + F** + ***	47
4.4	Combination of Stars and NPSs	49
4.5	Conclusion and Perspectives	52
5	Weighting and Ordering Solutions	53
5.1	Overview of Results	53
5.2	Framework	54
5.3	Partial Ordering of Solutions	55
5.4	Weighting of Solutions	57
5.4.1	Weighting Seeds	57
5.4.2	Decomposers	58
5.4.3	Weight Conservation Theorem	58
5.4.4	Generators	59
5.4.5	Dispatchers	59
5.5	Homogeneous Case: Weighting Is not Better than Ordering	64
5.6	Instance Independent Case: Ordering and Weighting Are Equivalent	68
5.7	Boolean Case: a Better Weighting for Partial Valuations	70
5.7.1	Calculation of the First Moment in this Framework	73
5.7.2	Calculation of the First Moment with Restricted Validity	76
5.8	Conclusion and Perspectives	79
6	Distributional Models and Non-Uniformity	81
6.1	Definitions and Notations	83
6.1.1	Overview of Models	83
6.1.2	Types of Clauses and Variables	84
6.2	Selection of Solutions	85
6.2.1	Construction of a Correct Weighting Scheme	85
6.2.1.1	Recapitulation of Existing Methods	85
6.2.1.2	Our Method	86

<i>CONTENTS</i>	15
6.2.2 Summary of Results	86
6.3 The First Moment Method	87
6.3.1 Types of Variables	87
6.3.2 Expression of the First Moment and its Constraints	88
6.3.3 Maximization of $\ln F$	91
6.3.4 Minimization of Global Weight	91
6.4 Conclusion	92
II Second Moment	95
7 Introduction to the Second Moment Method	97
7.1 How the Second Moment Method Is Supposed to Work	98
7.2 Use of the Second Moment Method for Lower-Bounding the Threshold of k-SAT	99
7.2.1 Second Moment of Solutions	99
7.2.2 Balancing True and False Surfaces	100
8 A General Framework for the Second Moment Method on k-SAT	101
8.1 Preliminaries	101
8.1.1 Values	101
8.1.2 Signs	102
8.1.3 Truth Values	102
8.1.4 Clauses Types	103
8.1.5 Surfaces	103
8.1.6 Symmetry of Occurrences	103
8.2 Expression of the First Moment	104
8.3 Expression of the Second Moment	105
8.4 Expression of the Lagrangian	106
8.4.1 Derivative with respect to $\gamma_{t,\mu}$	107
8.4.2 Derivative with respect to μ	107
8.5 Independence Point - Discussion about $\varepsilon_{v,w}$	107
8.6 Applications	109
8.6.1 Boolean Solutions	109
8.6.1.1 Preliminaries	109
8.6.1.2 Condition for $\varepsilon_{v,w} = \eta_v \eta_w$ at the Independence Point	110
8.6.1.3 Condition for the Second Moment Method to Work at $\delta = \frac{1}{2}$	110
8.6.1.4 Attempts to evade the $\delta = \frac{1}{2}$ condition	111
8.6.2 Implicants	113
8.6.2.1 Preliminaries	113
8.6.2.2 Condition for $\varepsilon_{v,w} = \eta_v \eta_w$ at the Independence Point	114
8.6.2.3 Condition for the Second Moment Method to Work at $\delta = \frac{1-\alpha}{2}$	115
8.7 Confrontation of the Second Moment Method with Reality	117
8.7.1 Distances between Solutions	117
8.7.2 Surface of True Literals	120
8.7.3 Non-Independence of True / False Surfaces	120
8.7.4 Non-Independence of Clauses Types	122
9 Second Moment Method on Distributional Random k-SAT	125
9.1 Preliminaries	125
9.1.1 Occurrences and Signs	125

9.1.2	Values	126
9.1.3	Truth Values	126
9.1.4	Types of Clauses and Surfaces	126
9.2	Expression of the First Moment	127
9.3	Expression of the Second Moment	127
9.4	Expression of the Lagrangian	128
9.4.1	Derivative with respect to $\gamma_{t,u}$	129
9.4.2	Derivative with respect to $\psi_{v,w}$	129
9.4.3	Derivative with respect to $\mu_{p,q}$	130
9.5	Independence Point	130
9.5.1	Condition for $\varepsilon_{v,w} = \eta_v \eta_w$ at the Independence Point	131
9.5.2	Attempts to evade the $\delta_{p,q} = \frac{1}{2}$ condition	132
9.6	Confrontation with Reality	132
9.6.1	Non-Independence of Values	132
9.6.2	Non-Independence of Surfaces	134
Bibliography		137
A Maximization of $\ln F$(Non-Uniform Selection of Solutions)		143
A.1	Resolution of the Global Lagrange Multipliers Problem	144
A.2	Inspection of the Boundary of \mathcal{P}	151
A.3	Inspection of the Interior of \mathcal{P}	163
B Some Useful Asymptotic Estimates		169
B.1	Multinomial Coefficients	169
B.2	Stirling Numbers of the Second Kind	169
Index		171

List of Figures

1.1	Threshold of satisfiability and peak of complexity.	22
2.1	Algorithmic complexity of random 3-SAT (standard & planted) with respect to c . . .	30
2.2	Complexity increases with p_0 in model Ambiguously Planted SAT.	32
2.3	Number of variables according to the proportion of solutions assigning them 1. . . .	32
2.4	Number of variables according to the proportion of solutions assigning them 1. . . .	33
2.5	Size of the cores in Ambiguously Planted SAT.	33
2.6	Complexity of Fixedly XOR Planted SAT with respect to p	34
2.7	Effect of fixation on the complexity for WalkSat.	34
4.1	Coarsening graph of some valid partial assignments.	42
4.2	FFF: $c = 5.191$ for $\omega_* = 0$	44
4.3	FFF+FF*: $c = 4.883$ for $\omega_* = 0.587$	47
4.4	FFF+FF*+F**: $c = 4.866$ for $\omega_* = 0.999$	48
4.5	FFF+FF*+F**+***: $c = 4.883$ for $\omega_* = 0.9999$	49
4.6	Stars+NPSs: $c = 4.643$ for $\omega_* = 0.001$	52
5.1	A bad orientation.	56
5.2	A good orientation.	56
5.3	A very good orientation.	56
5.4	A bad weighting system.	60
5.5	An homogeneous weighting system.	60
5.6	An heterogeneous weighting system.	60
5.7	Our weighting system with FFF+FF*: $c = 4.643$ for $\omega_* = 0.0003$	76
5.8	Our weighting system with FFF+FF*+F**+***: $c = 4.643$ for $\omega_* = 0.001$	79
6.1	Hamming graph of solutions for formula Φ	82
6.2	Uniform orientation.	83
6.3	Non-uniform orientation.	83
8.1	$\Sigma_T = \Sigma_F = \frac{k}{2}$: $-\infty$ is obtained only when $\delta = \frac{1}{2}$	112
8.2	$\Sigma_T = k\eta_T$ and $\Sigma_F = k\eta_F$: $-\infty$ is obtained only when $\delta = \frac{1}{2}$	112
8.3	Hamming similarity among solutions in NAE-SAT and in SAT.	118

8.4	Hamming similarity is greater among fixed variables.	118
8.5	Hamming similarity among free variables increases with c	119
8.6	Hamming similarity among fixed variables increases with c	119
8.7	The true surface of fixed variables decreases with c	120
8.8	The true surface of free variables increases with c	121
8.9	(Non-)independence of surfaces in NAE-SAT and in SAT.	121
8.10	The non-independence of surfaces comes rather from fixed variables.	122
8.11	(Non-)independence of uniquely satisfied clauses in NAE-SAT and in SAT.	123
9.1	Setting $\delta_{p,q} = \frac{1}{1+\omega^{p-q}}$ makes the Second Moment Method work iff $\omega = 1$	132
9.2	$\delta_{p,q}$ and $\mu_{p,q} = \delta_{p,q} (1 - \delta_{p,q})$ at $c = 2$	133
9.3	$\delta_{p,q}$ and $\mu_{p,q} \neq \delta_{p,q} (1 - \delta_{p,q})$ at $c = 4$	133
9.4	The distributional model slightly curbs the non-independence of surfaces.	134
A.1	Standard model: $c = 4.500$ for $\beta_1 \simeq 0.443$ and $\beta_2 \simeq 0.422$	166
A.2	Model with balanced signs: $c = 3.509$ for $\beta_1 \simeq 0.558$ and $\beta_2 \simeq 0.366$	167
A.3	Model with balanced occurrences: $c = 4.623$ for $\beta_1 \simeq 0.445$ and $\beta_2 \simeq 0.421$	168
A.4	Model with balanced signs & occurrences: $c = 3.546$ for $\beta_1 \simeq 0.568$ and $\beta_2 \simeq 0.363$	168

List of Tables

1.1	Sharpness vs. complexity of some problems.	23
4.1	Bounds obtained by weighted First Moment Method on partial assignments.	52
6.1	Upper bounds obtained in various distributional models.	87
6.2	Upper bounds obtained in the drawing and distributional models of 3-SAT.	93
8.1	Critical ratios of implicants obtained by the Second Moment Method.	116
A.1	Solutions to the Lagrange multipliers problem of the first moment.	151
A.2	Summary of bounds on β_t	152
A.3	Interior point when $\beta_2 = 0$	156
A.4	Interior point when $\beta_3 = 0$	162
A.5	Values of parameters obtained by a sweep over the interior of \mathcal{P}	166

Introduction

“In mathematics, approximation or estimation typically means finding upper or lower bounds of a quantity that cannot readily be computed precisely.” Wikipedia page for *Estimation* - July 26th 2010.

1.1 Framework - Random **3-SAT**

1.1.1 **k-CNF SAT**

We consider n variables x_1, \dots, x_n , each taking a *value* in the domain $\{0, 1\}$. A *formula* or *instance* is built from variables by logical connectives: \neg , \vee and \wedge . A formula in the Conjunctive Normal Form (in short CNF) appears as follows:

- \neg , the negation, applies to a single variable to build a *literal*; $\neg x_1$ as well as x_1 are called *literals*;
- \vee , the disjunction, applies to several literals to build a *clause*, e.g. $x_1 \vee \neg x_2 \vee x_4$;
- \wedge , the conjunction, applies to several clauses to build a *formula*, e.g. $(x_1 \vee \neg x_2 \vee x_4) \wedge (x_{10} \vee x_3)$.

A clause is *satisfied* iff one of its literals is satisfied (this is an inclusive *or*). **SAT** is the problem of deciding whether it is possible to assign each variable a value in the domain $\{0, 1\}$ so that the formula is true (i.e. each clause is satisfied). **SAT** was the very first problem to be shown NP-complete by Cook - 1971 [Coo71] and Levin - 1973 [Lev73].

k-SAT is a restriction of **SAT** to instances having clauses of width k . It turns out that **k-SAT** is NP-complete as soon as $k \geq 3$ but polynomial if $k \leq 2$. We shall particularly focus on **3-SAT**. A **3-CNF** instance thus looks like that:

$$\begin{array}{ccccccc}
 x_3 & \vee & \neg x_6 & \vee & x_7 & & \\
 \neg x_{12} & \vee & x_1 & \vee & \neg x_2 & & \\
 & & \dots & & & & \\
 x_5 & \vee & x_{18} & \vee & x_3 & &
 \end{array}$$

Let us now introduce two well-studied variants of the SAT problem:

- **NAE-SAT**, where a clause is satisfied iff not all literals have the same truth value (Not All Equal), i.e. at least one literal is true and at least one literal is false. So this is a kind of symmetrization of SAT. It turns out that k -NAE-SAT is also NP-complete when $k \geq 3$;
- **XOR-SAT**, where the inclusive *or* of SAT become an exclusive *or*. Consequently the problem amounts to solving a linear system of equations. Thus Gaussian elimination makes it a polynomial problem.

1.1.2 Goldberg’s Random-Clause-Width Model

One of the first attempts to build a random model of formulas was made by Goldberg in 1979 [Gol79]. In this model, each of the m clauses of a formula is built as follows: each of the $2n$ literals are present with probability p . Goldberg showed that in this model, the SAT problem is polynomial on average. Moreover, Franco & Paull - 1983 [FP83] showed that even a constant number of random guesses can find a satisfying assignment almost surely.

However, Franco & Swaminathan - 1997 [FS97] noticed that formulas become harder when their clause width is less than $\ln m$.

We shall see in the sequel that when clause width is bounded (k -SAT), formulas can be very hard, and that is the reason why we focus on random k -SAT.

1.1.3 Drawing Model

In the classical drawing model of k -SAT, there are n variables and $m = cn$ clauses. Clauses are drawn independently as follows: pick k literals uniformly at random. A variation on this model requires all variables in the same clause to be distinct. It turns out that both models are equivalent for the threshold properties we shall consider.

- in the first model, there are $(2n)^{kcn}$ possible formulas;
- in the second one, there are $\left(2^k \binom{n}{k}\right)^{cn}$ possible formulas.

1.1.4 Distributional Model

We consider a generic random model of k -CNF formulas having n variables and cn clauses, parametrized by a probability distribution $(d_{p,q})_{p,q \in \mathbb{N}}$ such that $\sum_{p,q \in \mathbb{N}} d_{p,q} = kc$. Before we get formulas we draw *configurations* as follows:

1. each of the n variables is given p labeled positive occurrences and q labeled negative occurrences in a way that the overall proportion of variables with p positive occurrences and q negatives occurrences is $d_{p,q}$;
2. a configuration can be seen as a matrix of kcn bins containing literals occurrences; the repartition of literals into the kcn bins is drawn uniformly among all $(kcn)!$ permutations of labeled literals occurrences.

A legal *formula* is a configuration where occurrences are unlabeled and each clause contains at most one occurrence of each variable. For the models we consider in this thesis (as described in section 6.1.1), it was shown that an upper bound on the satisfiability threshold obtained

for configurations also applies to legal formulas (see Díaz, Kirousis, Mitsche & Pérez-Giménez - 2009 [DKMPG09] for the standard model and Boufkhad, Dubois, Interian & Selman - 2005 [BDIS05] for models where p and q are bounded).

Note in particular that for the standard model discussed in section (1.1.3) where all literals are drawn uniformly and independently, it was shown by Dubois, Boufkhad & Mandler - 2003 [DBM03] and Díaz, Kirousis, Mitsche & Pérez-Giménez - 2009 [DKMPG09] that the resulting distribution is the 2D Poisson distribution:

$$d_{p,q} = \binom{p+q}{p} \frac{e^{-kc}}{(p+q)!} \left(\frac{kc}{2}\right)^{p+q}. \quad (1.1)$$

1.2 Motivations

1.2.1 A Universal Model

Since SAT is NP-complete, all NP-complete problems can be reduced to it in polynomial time. Moreover, the very simple yet powerful framework offered by SAT has drained lots of efforts to elaborate efficient SAT-solvers. So in practice lots of problems are encoded into a SAT instance and then submitted to a SAT-solver.

For example, let us consider software distributions. Such distributions as Ubuntu or Debian have over 20,000 available packages, and there exist different kinds of dependencies between packages: “A requires B” or “A is incompatible with C”. The problem of deciding whether a given package is installable in a given repository is called *INSTALLABILITY* and was shown to be NP-complete by Di Cosmo, Durak, Leroy, Mancinelli & Vouillon - 2006 [DDL⁺06]. In another paper, the same authors plus Boender & Treinen [MBC⁺06] showed that encoding this problem into SAT and submitting it to the FGrasp solver is dramatically better than encoding it into a Constraint Programming instance submitted to the Mozart-Oz solver.

1.2.2 A Difficult Problem

As mentioned earlier in section 1.1.1, SAT was the very first problem to be shown NP-complete by Cook - 1971 [Coo71] and Levin - 1973 [Lev73]. However we saw in section 1.1.2 that Goldberg’s model is easy on average! However for random 3-SAT the situation is quite different:

1. as was first observed by Simon & Dubois - 1989 [SD89] there exists a phase transition in the satisfiability property: for low values of c (the ratio $\frac{\text{\#clauses}}{\text{\#variables}}$), the probability of satisfiability of formulas tends to 1 as n tends to infinity. On the contrary, it tends to 0 for high values of c . But it is experimentally observed that there is a threshold (or phase transition) at around $c = 4.25$ where the probability suddenly goes from 1 to 0;
2. a few years later, Mitchell, Selman & Levesque - 1992 [MSL92] showed that a complexity pattern seems to be connected to this threshold phenomenon: instances are easy when c is far below 4.25, harder when c is far above 4.25, and considerably harder when c is around 4.25. This situation is depicted on figure 1.1.

More details on hard instances are available in the survey of Cook & Mitchell - 1997 [CM97].

1.3 Sharpness of Thresholds

Look at figure 1.1 again. What would you like to say about both curves? The second one looks like the d...e of the first one. The derivative, yes (in absolute value). This does make sense in fact.

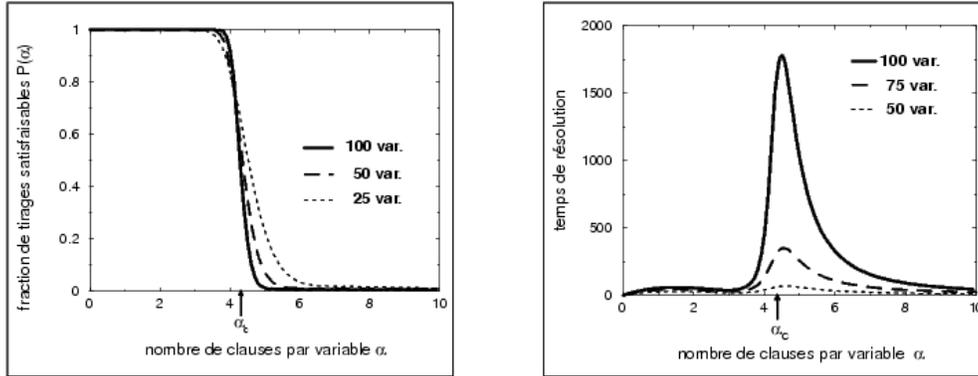


Figure 1.1: On the left picture is the probability for a formula to be satisfiable with respect to c . On the right picture is the algorithmic (resolution) complexity with respect to c . We can see that the peak in complexity occurs in the neighborhood the threshold of satisfiability. These pictures come from Biroli, Cocco & Monasson - 2002 [BCM02].

1.3.1 Influence and Complexity

Let us enumerate clauses from 1 to $M = 2^k \binom{n}{k}$ and define an indicator variable γ_j telling if clause number j is present in the formula. In this framework, a formula can be seen as a boolean vector $\gamma = (\gamma_1, \dots, \gamma_M)$. The probability space considered here is defined by a measure $\mu_p(\gamma) = p^k (1-p)^{M-k}$ where $p \in (0, 1)$ and $k = \sum_{i=1}^M \gamma_i$. Given a boolean function f on formulas, we call *influence* of clause j on f the quantity $I_j(f) = \mu_p(f(\gamma \oplus \mathbf{1}_j) \neq f(\gamma))$, i.e. the probability over all formulas that clause j is *critical* for f . For us f will be of course SATISFIABILITY. The total influence of f is defined as $I(f) = \sum_j I_j(f)$.

Margulis-Russo's lemma [Mar74, Rus78] states that the total influence is equal (in absolute value) to the slope of the probability curve $\frac{d\mu_p(f)}{dp}$, where $\mu_p(f)$ denotes the probability that $f(\gamma) = 1$. So is there a connection between influence and algorithmic complexity?

To some extent, yes. Boppana's theorem [Bop97] states that if f is computable by a circuit of depth d and of width w , then $I(f) \leq 2(10 \log_2(4w))^{d-1}$, assuming that $I(f)$ is taken under uniform probability: $p = \frac{1}{2}$. It turns out that in our case, $p = \Theta(M^{\frac{1}{k}-1}) = o(1)$, so this theorem is not applicable. Moreover in the circuit complexity of Boppana, the most difficult function is PARITY... However it is a hint that there is a connection between influence and complexity, and we think this is a direction of research which deserves to be further investigated.

1.3.2 The Quest for Sharpness

We denote by $t_\varepsilon(f)$ the threshold interval, that is $|p_2 - p_1|$ where $\mu_{p_1}(f) = \varepsilon$ and $\mu_{p_2}(f) = 1 - \varepsilon$. The critical probability p_c is defined by $\mu_{p_c}(f) = \frac{1}{2}$. Bollobás & Thomasson's theorem [BT87] states that for each monotone boolean function f , $t_\varepsilon(f) = O(\min(p_c, 1 - p_c))$ for asymptotic M . It follows a dichotomy: a threshold is called *sharp* if $t_\varepsilon(f) = o(\min(p_c, 1 - p_c))$ and *coarse* otherwise.

There were then several lower bounds of influence (i.e. the slope $\frac{d\mu_p(f)}{dp}$) according to the

Table 1.1: Sharpness vs. complexity of some problems.

problem	threshold	complexity
2-SAT	sharp	P
3-SAT	sharp	NP-complete

Margulis-Russo lemma) by a function of $\mu_p(f)$ itself, often referred to as isoperimetric inequalities. See for example the inequalities of Kahn, Kalai & Linial - 1988 [KKL88]: $\max_j I_j(f) \geq K \min\left(\mu_{\frac{1}{2}}(f), 1 - \mu_{\frac{1}{2}}(f)\right) \frac{\log M}{M}$ or Talagrand - 1994 [Tal94]: $\sum_{j=1}^M \frac{I_j(f)}{\ln \frac{1}{I_j(f)}} \geq K \frac{\mu_p(f)(1-\mu_p(f))}{\ln \frac{2}{p(1-p)}}$.

More information on this topic can be found in the excellent survey of Kalai & Safra - 2006 [KS06].

The sharpness and the value of the 2-SAT threshold at $c = 1$ has been known since the early 90's (see Chvátal & Reed - 1992 [CR92] and Goerdt - 1996 [Goe96]), but only in 1999 did Friedgut & Bourgain [FB99] prove that the threshold of 3-SAT is sharp, a very deep result. However the question of the location of this threshold is still open, and this thesis is dedicated to the quest for upper and lower bounds on this threshold location.

1.3.3 Sharpness vs. Complexity

Schaefer's dichotomy theorem [Sch78] characterizes those Constraint Satisfaction Problems (see definition in section 5.2) which are in P (trivial, bijunctive, Horn, affine) and states that all other ones are NP-complete.

Building on Friedgut-Bourgain's theorem, Creignou & Daudé [CD03, CD04] established a similar dichotomy regarding the sharpness of thresholds. They showed that if there are some types of constraints f such that $f(x_1, \dots, x_k) = 1 \Rightarrow x_i = \varepsilon$ (f strongly depends on one component) or $f(x_1, \dots, x_k) = 1 \Rightarrow x_i \oplus x_j = \varepsilon$ (f strongly depends on a 2-XOR relation) for some $\varepsilon \in \{0, 1\}$, then the threshold is coarse, and otherwise the threshold is sharp.

The reader will certainly have noticed that both dichotomies are different, which means that sharpness and complexity do not seem to be correlated, see table 1.1. So can't random 3-SAT tell anything interesting about the computational complexity of 3-SAT?

1.4 The Solutions Space

Here is maybe the major hope in using random 3-SAT as a tool to establish computational complexity results on 3-SAT.

1.4.1 The Number of k-CNF Functions

A boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$ is characterized by the set $f^{-1}(\{1\})$ of its solutions. By the way, how many boolean functions are there? 2^{2^n} . Now, how many k-CNF formulas are there? there are $2^k \binom{n}{k}$ possible clauses, each one may be present in the formula or not, so there are $2^{2^k \binom{n}{k}}$ k-CNF formulas. But how many boolean functions are represented by these formulas? At least $2^{\binom{n}{k}}$, because two different monotone formulas have different sets of solutions. A *monotone formula* is a formula without negation \neg . Let $\mathbf{0}_{i,j,k}$ be the assignment where all variables are at 1 except x_i, x_j and x_k . Then $\mathbf{0}_{i,j,k}$ is a solution of a monotone formula Φ iff clause $x_i \vee x_j \vee x_k$ does not appear in Φ .

So there are between $2^{\binom{n}{k}}$ and $2^{2^{\binom{n}{k}}}$ boolean functions which are expressible by a k -CNF formula. This is negligible compared to 2^{2^n} , the number of boolean functions. Consequently the set of solutions of a k -CNF formula must be somehow structured! Moreover in our random model we only allow $O(n)$ clauses out of $2^{\binom{n}{k}}$. So what does the solutions space look like?

And what if k may depend on n ? Of course every boolean function is expressible by a n -CNF formula. Bollobás & Brightwell - 2003 [BB03] give different estimates for various ranges of k ; in particular we can notice that the proportion of boolean functions expressible by a $n - 1$ -CNF formula is about 61% and it is still negligible - about $\exp\left(-2^{\frac{\log_2^2 n}{2}}\right)$ - for $k = n - 2$. They establish a kind of phase transition with respect to $\alpha = \frac{k}{n}$: if $\alpha < \frac{1}{2}$, then there are $2^{o(2^n)}$ k -CNF functions, whereas if $\alpha > \frac{1}{2}$, there are $2^{\Omega(2^n)}$ k -CNF functions. Moreover the authors conjecture that the number of k -CNF functions is $2^{\binom{n}{k}(1+o(1))}$ if $k \leq \left(\frac{1}{2} - \varepsilon\right)n$ for any $\varepsilon > 0$, which they proved with Leader [BBL03] for $k = 2$. Once more 3-SAT seems to be much more difficult to handle than 2-SAT...

1.4.2 Clustering

Statistical Physics has brought some helpful insights into the geometry of the solutions space. For example, Mertens, Mézard & Zecchina - 2006 [MMZ06] successfully applied the so-called "One-Step Replica Symmetry Breaking Cavity Method" from Statistical Physics to random k -SAT, which enabled them to identify at least 3 phases: in random 3-SAT, solutions form a connected mist for $c < 3.927$ and then group together into an exponential number of clusters until they vanish at $c = 4.267$. Moreover, for $k \geq 4$, there is an extra phase just below the satisfiability threshold, where there is a finite number of clusters, see Mora's thesis [Mor07].

Another phenomenon is quite interesting: just above the threshold, some *frozen variables* appear, i.e. variables which take the same value in all solutions; they form a so-called *backbone*. Monasson, Zecchina, Kirkpatrick, Selman & Troyansky - 1999 [MZK⁺99] showed that there is a remarkable difference between 2-SAT and 3-SAT: the proportion of frozen variables is continuous (thus zero) at the threshold for 2-SAT (second-order phase transition), whereas it is discontinuous for 3-SAT (first order phase transition). That might be an explanation for the algorithmic complexity of 3-SAT, but it turns out that 3-XOR-SAT (which is solvable in polynomial time by Gaussian elimination) also has a first order transition; however the way variables freeze in each cluster in the satisfiable phase might make the difference, see Semerjian - 2007 [Sem07]. Moreover Mora emphasizes a fundamental difference in the fragmentation process: a condensation transition exists for k -SAT but not for k -XOR-SAT.

On the other hand, Istrate, Boettcher & Percus - 2005 [IBP05] make a connection between first-order transitions of the *spine* (slightly different from the backbone, see [BBC⁺01]) and the hardness for resolution-based algorithms (it turns out that 3-XOR-SAT is hard for resolution).

Although their methods are usually not mathematically rigorous, physicists were able to elaborate the best algorithm so far for random 3-SAT: Survey Propagation, by Braunstein, Mézard & Zecchina - 2005 [BMZ05]. This a fixed-point algorithm where variables and clauses pass messages to each other.

There are some mathematically established results about clustering in the solutions space. For example, Achlioptas & Ricci-Tersenghi - 2006 [ART06] showed that, somewhere below the satisfiability threshold, there is an exponential number of separate clusters (for $k \geq 8$) and every cluster have some (locally) frozen variables (for $k \geq 9$).

1.5 Outline of the Thesis

1.5.1 Experimenting through SATLab

Since we work on random k-SAT and typical behavior of instances is difficult to compute with mathematical tools, we decided to build a software (which we called SATLab) to enable us to have some insight into formulas and their solutions. This software achieves the repetitive and painful tasks of drawing random instances with the desired parameters, calling a solver on the drawn instances, and bringing together all results into a single graph.

We first make the classical observation of the complexity peak around the threshold, and go a little further into the study of the hardness of some planted models (i.e. models with a hidden solution), comparing them to each other, and emphasizing such spin-offs of computational hardness as frozen variables or cores. This experimental study is just a sample of what can be performed by SATLab.

We go back to SATLab later in a discussion about the relevance of the Second Moment Method.

1.5.2 Weighted First Moment Method

The First Moment Method has been the main tool to compute upper bounds on the 3-SAT threshold.

We use the weighted valid partial assignments and the Weight Preservation Theorem introduced by Maneva, Mossel & Wainwright - 2007 [MMW07] to build some other similar systems and apply the First Moment Method to them.

A valid partial assignment of a formula is an assignment over the domain $\{0, 1, *\}$ such that there is no clause in the formula having (only false literals) or (only false literals but one assigned $*$). In a valid partial assignment, we call a variable starrable if it is assigned 0 or 1 and assigning it $*$ would still yield another valid partial assignment. Given two non-negative reals ω_0 and ω_* , the weight of a valid partial assignment σ is $W(\sigma) = \omega_0^{n_0(\sigma)} \omega_*^{n_*(\sigma)}$, where $n_0(\sigma)$ is the number of starrable variables in σ and $n_*(\sigma)$ is the number of starred variables in σ . The Weight Preservation Theorem [MMW07, MS08, AM09] insures that if σ is a boolean solution and $\omega_0 + \omega_* = 1$, then $\sum_{\tau \leq \sigma} W(\tau) = 1$. Consequently the weight W perfectly fits in with the First Moment Method. Doing the calculation formally and then numerically, we get an upper bound of 4.883.

The validity property of Maneva *et al.* discards clauses of type FFF and FF*, but what happens if we choose to discard FFF, FF*, F** and ***, or just FFF? What is the upper bound in each of these cases? We give the mathematical corresponding calculations and find out numerically that we cannot get a better upper bound than 4.866 in this framework.

So we make a further attempt to use the valid partial assignments, discarding some of them which cannot lay below a NPS (a NPS is a solution where every free variable are assigned 1). The NPSs were presented in 1997 by Dubois & Boufkhad [DB97] and by Kirousis, Kranakis & Krizanc [KKK97]. Counting them enabled [DB97] to get an upper bound of 4.643 and [KKK97] to get an upper bound of 4.667. It turns out that combining NPSs and valid partial assignments, the best upper bound we obtain corresponds to the special case where the $*$'s disappear, i.e. the mere counting of NPSs (and thus the upper bound of 4.643).

That led us to question the performance of weighting versus merely counting solutions.

1.5.3 Weighting and Ordering Solutions

In this chapter we consider general Constraint Satisfaction Problems (CSPs). We aim at defining on their solutions space some orderings and weightings which may be suitable for the First Moment Method:

- orderings of the solutions space must have some minimal elements, which will be counted by the First Moment Method;
- weightings of the solutions space must have a global weight of at least 1;
- both orderings and weightings must be *locally* computable, that is looking only at the immediate neighborhood, so that the First Moment calculation may be tractable.

So we define the following ordering: for each variable x , the domain D is given a total strict order $<_{F,x}$. Now to compare two neighboring solutions, it suffices to compare their values at the variable x on which they differ.

Weighting the CSPs solutions in compliance with the aforementioned constraints is somewhat trickier. We use a weighting seed $s_F(x, a)$ and a dispatcher $d_F(x, a)$ defined for each variable x and each value a . Now given a clique of neighborhood in the solutions space, that is a set of solutions differing only on the same variable x , we give each of the present solutions σ a weight $w_F(\sigma, x)$ equal to the weighting seed $s_F(x, \sigma(x))$, and we dispatch among them the weighting seeds of the absent solutions in proportions following d_F . Finally the weight of a solution is the product of the weights $w_F(\sigma, x)$ over x .

We prove that the above weighting scheme has a global weight of at least 1 as a particular case of a more general Weight Conservation Theorem, by which we give sufficient conditions for a weighting to have a global weight of at least 1.

The scope of our Weight Conservation Theorem is indeed more general, since we derive from it another weighting system fitted to Maneva's valid partial assignments (we regard them as solutions of a CSP over the domain $\{0, 1, *\}$). The purpose of our new weighting of valid partial assignments is that it weighs less than Maneva's while keeping a global weight of at least 1. So we performed the calculation of the first moment with our brand new weighting on valid partial assignments. With Maneva's weighting, the upper bound obtained was 4.883, whereas our new weighting gives us an upper bound of 4.643. So it is better, but not outstanding. Namely the attentive reader will remark that the best tuning of our weighting enables us in fact to count NPSs!

So this leads us to the following question: what is the purpose of weighting, if the best tuning of weights always amounts to an ordering? We are able to answer this question in two special cases:

1. homogeneous weightings (i.e. when $d_F = s_F$): given an homogeneous weighting, we show that there always exists a ordering which is at least as good as it;
2. on a set of CSP instances closed under renaming (i.e. on a set closed under permutations of the values of the domain): we show that on such a set, weightings and orderings are equivalent on average.

The general question of comparing weightings and orderings remains open.

1.5.4 Distributional Models and Non-Uniformity

Since weighting valid partial assignments does not seem to be better than counting NPSs, we decided to try another direction to get upper bounds on the threshold of 3-SAT. First of all we took the distributional model, which intrinsically gives better upper bounds than the drawing model (see table 6.2). Moreover the distributional model allows more precise tunings because it gives some control on the number of occurrences of the literals.

We observed that the orientations of the solutions space towards interesting solutions (such as NPSs) that had been implemented to get upper bounds on the threshold (4.643 by Dubois & Boufkhad - 1997 [DB97], 4.667 by Kirousis, Kranakis & Krizanc - 1997 [KKK97], 4.506 by Dubois, Boufkhad & Mandler - 2000 [DBM00, DBM03] as well as 4.490 by Díaz, Kirousis, Mitsche and Pérez-Giménez - 2009 [DKMPG09]) were all uniform: all edges relative to a given variable x in the Hamming graph of the solutions are oriented in the same way, i.e. from 0 to 1 or from 1 to 0). And so is the general weighting we put on CSPs!

So the purpose of our investigations is to build a non-uniform orientation of the solutions space, in a way that is still suitable with the First Moment Method requirements (i.e. insuring that minimal elements will exist) and also which may give lower upper bounds than the previous ones.

To do so, we keep track of a set of 5 numbers associated with each variable and use it to discriminate among neighboring solutions. These 5 numbers provide information on the repartition of true and false occurrences of each variable in each type of clauses (clauses having 1, 2 or 3 true literals). Just as [DBM03] oriented solutions towards a greater quantity of true literals, so shall we build on this idea, but we want to be able to distinguish between them on the basis of their *criticality*. Our intuition is that we should select solutions in which the least occurrences of true literals are *critical*. The less a clause has true literals, the more its true literals are critical. Such a property is by nature non-uniform.

We apply our technique to 4 different models of random 3-SAT: the standard model, a model where occurrences of variables are balanced, a model where signs are balanced and a model where both signs and variables are balanced (the model introduced by Boufkhad, Dubois, Interian & Selman - 2005 [BDIS05]). Our approach gives lower upper bounds than the previous methods for selecting solutions in all of these models.

It turns out that in the standard model, our benchmark was the 4.506 of [DBM00]; namely the 4.490 of [DKMPG09] was obtained from the 4.506 by adding two extra ingredients: elimination of pure literals and typicality of clauses signatures. These extra ingredients are in fact of independent interest: they could be added to our calculations, but the additional computational cost would be too high with respect to the expected gain. So we focused on the selection of solutions itself and found

- a non-uniform way of selecting solutions, the first one to our knowledge,
- which is better than the best known (uniform) ways of selecting solutions.

Our calculations lead us to as classical problem of constrained maximization, and we solve it classically as well, through the Lagrange multipliers method and following the approach of [DKMPG09]. Thus we must prove that the objective function does not maximize at the boundary of the polytope. Moreover we check that the solution we get is indeed a global maximum by a sweep over the polytope.

1.5.5 A General Framework for the Second Moment Method on k-SAT

The Second Moment Method has not been the main tool to compute lower bounds on the 3-SAT threshold. Namely most of the lower bounds were obtained by analyzing algorithms finding solutions with high probability, and in particular the best currently known lower bound (3.52), obtained by Kaporis, Kirousis & Lalas [KKL06] and by Hajiaghayi & Sorkin [HS03]. It is rather difficult to make the Second Moment Method work on k-SAT, because it requires to count a set of solutions with low variance.

Consequently, the work of Achlioptas & Peres - 2004 [AP04] can be seen as a great success in the implementation of the Second Moment Method on k-SAT, establishing a lower bound of 2.68 for 3-SAT and an asymptotically tight lower bound of $2^k \ln 2 - O(k)$ for k-SAT.

We aim at making the most of the Second Moment Method on k-SAT. To do so we take a

different approach from Achlioptas & Peres'. In our framework we select solutions according to the fraction of variables assigned 1 and the fractions of the different types of clauses (i.e. the number of true literals occurrences in the clauses). This framework is general enough to include boolean solutions, implicants etc. However, tuning our parameters the best way we could, we got numerical evidence that we could not obtain better lower bounds than 2.83.

The stumbling block we recurrently encountered is what we call the independence point. It corresponds to the couples of independent solutions in the subset of selected solutions. Even though solutions are independent, the proportion of literals occurrences having a certain truth value may not be independent between solutions. We got numerical evidence that the Second Moment Method does not work if truth values of literals occurrences are not independent. On the other hand, when they are independent, we give a necessary condition for the Second Moment Method to work, taking into account just the exponential equivalent of the second moment at this point. This condition tells us that we must select solutions having equal true and false surfaces (the surface is just the total number of literals occurrences), which is very artificial with respect to what we can observe with **SATLab**. Using this condition, we could make the Second Moment Method work numerically; however, since the lower bounds we get (2.83) are far below the currently best lower bound (3.52), we do not give a rigorous (and tedious) proof of our lower bounds (to do so, the exponential equivalent would not be enough, and calculations would become quite involved).

The very restrictive conditions we encountered to make the Second Moment Method work may be due to some weaknesses of our framework. We do not claim that the Second Moment Method is doomed to perpetual failure on **k-SAT**. We only hope to shed a small ray of light onto it.

1.5.6 Second Moment on Distributional Random **k-SAT**

Using the standard distributional model instead of the standard drawing model yields better upper bounds on the satisfiability threshold (cf. table 6.2). Moreover, we would like to gain some more control over the proportion of variables assigned 1 according to the imbalance between their positive and negative occurrences. Namely, a variable is all the more expected to be assigned 1 in a solution as it has more positive occurrences, and vice-versa. At least this seems to happen on real solutions, as we can see with **SATLab**. That is the reasons why we implement the Second Moment Method on the distributional model.

The outline of the calculations is roughly the same as in the drawing model. A major difference though is that if the independence of truth values of literals occurrences does not hold at the independence point, then the independence point violates one of the constraints. However, the result is the same as in the drawing model: we are able to make the Second Moment Method work only when the independence of truth values of literals occurrences holds at the independence point.

This calculation is very disappointing, because it leads not only to the same kind of constraints as in the drawing model, which are again very artificial with respect to what can be seen through **SATLab**, but also to the same critical value of 2.83!

Experimenting through SATLab

E shall see in this thesis that it is rather difficult to get estimates by calculations. In particular access to what happens almost surely is most of the time not possible by calculations. Calculations involve rather moments than probabilities. For example it is very difficult to give rigorous upper and lower bounds on the 3-SAT threshold location, whereas it is mere experimental evidence that this threshold is located at around $c \simeq 4.25$.

Consequently, if we want to have some more insights into what happens in reality, we have to make some experiments, that is, draw some random formulas with a computer program, find some solutions with a SAT-solver, and finally analyze all of these raw data to infer properties we want to investigate. To this end we built a software called SATLab, which is available and documented (more thoroughly than here) at the following web page [Hug10]: <http://www.pratum.org/satlab/>

2.1 A Brief Description of SATLab

SATLab is written in object-oriented Perl, with a Graphical User Interface in Tk. The purpose of this software is to avoid to the experimenter the burden of repetitive tasks: generate some random formulas, then call some SAT-solvers, extract complexity, solutions, and gather all results in a data file which is then passed to gnuplot. Moreover the GUI is designed in order to eliminate as far as possible unnecessary parameters.

A description of parameters and variables is available at the aforementioned web page.

2.2 Some Noteworthy Observations through SATLab

2.2.1 Algorithmic Complexity

The first observation to make is the famous “easy-hard-easy” pattern (see figure 2.1), which was already observed by Mitchell, Selman & Levesque in 1992 [MSL92].

Since the hard region corresponds to the threshold region (in fact hardness occurs just above the threshold), this observation aroused interest in the threshold phenomenon of random 3-SAT. The solver used here is `Dew_Satz`, a solver based on resolution written by Slaney & Anbulagan [SA05, Anb05].

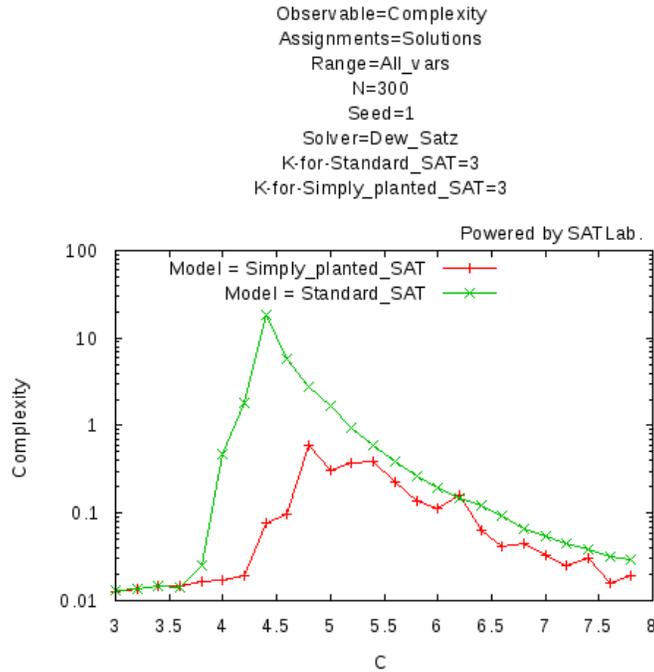


Figure 2.1: Algorithmic complexity of random 3-SAT (standard & planted) with respect to c .

It is also noteworthy that the same pattern occurs in random *planted* 3-SAT, which is a model with a hidden solution (so always satisfiable), see figure 2.1. However the relative peak in complexity is much coarser in the planted model than in the standard model. It is also noticeable that in the planted model, the peak occurs in the neighborhood of $c = 5$, which is significantly above the threshold.

2.2.2 Some Hard Satisfiable Models

Solvers based on a random walk (such as WalkSat, introduced in 1995 by Selman, Kautz & Cohen [SKC95]) eventually find a solution if some solutions exist, but they are unable to prove that there is no solution otherwise. That is why they are called *incomplete solvers*. Thus benchmarks for these solvers must be satisfiable instances. However it turns out that just hiding one solution (the so-called *planted model*) yields very easy instances (see figure 2.1). This may be explained by the fact that there is an imbalance in the number of positive and negative occurrences of each literal (since 1 clause signature out of 8 is discarded). Then a simple majority-based assignment (0 for variables having more negative than positive occurrences and vice-versa) is not far from a solution.

Some physicists, Barthel, Hartmann, Leone, Ricci-Tersenghi, Weigt & Zecchina - 2001 [BHL⁺01] made a thorough study of planted models of 3-SAT having a balance between positive and negative occurrences; they define the following probabilities:

- the signature making 3 true literals has probability p_0 ;
- each of the 3 signatures making 2 true literals has probability p_1 ;
- each of the 3 signatures making 1 true literal has probability p_2 .

Probabilities must have a total sum of 1: $p_0 + 3p_1 + 3p_2 = 1$. Moreover, the balance between positive and negative occurrences gives the following condition: $3p_0 + 6p_1 + 3p_2 = 3p_1 + 6p_2$, i.e. $p_0 + p_1 = p_2$. Thus we must have:

$$\begin{aligned} p_1 &= \frac{1 - 4p_0}{6} ; \\ p_2 &= \frac{1 + 2p_0}{6} . \end{aligned}$$

It follows that $0 \leq p_0 \leq \frac{1}{4}$; in **SATLab** we call that model **Ambiguously Planted SAT**:

- the case when $p_0 = 0$ makes the hidden assignment an **NAE-SAT** hidden assignment. It amounts to hiding 2 opposite assignments. This special case was studied by Moore, Achlioptas & Jia - 2005 [MAJ05]; in **SATLab** we call this model **Doubly Planted SAT**.
- the case when $p_0 = 0.073$ corresponds to the case when the model is *deceptively* planted, as proposed by Jia, Moore & Strain - 2007 [JMS07]. They put onto each clause a signature making $t > 0$ literals true with a probability proportional to q^t , where q is a positive parameter. The balance for 3-SAT is thus obtained when $q^3 + q^2 = q$, i.e. $q \simeq 0.618$ (the golden ratio). We shall refer to this model as **Deceptively Planted SAT**.
- the case when $p_0 = 0.25$ (and thus $p_1 = 0$) makes the hidden assignment a **XOR-SAT** hidden assignment. Thus this assignment can be found in polynomial time by Gaussian elimination. We shall refer to this model as **XOR Planted SAT**.

Barthel *et al.* note that the case $p_0 = 0$ is the easiest one and explain this by a second-order phase transition. In the other cases, the algorithmic difficulty for **WalkSat** is maximal at the ratio $c \simeq 4.25$ and increases with p_0 and they explain this fact by the appearance of a backbone of frozen variables at $p_0 = 0.077$ which size increases with p_0 . We observe on figure 2.2 that the algorithmic difficulty for **kcnfs** (a complete solver based on resolution, by Dequen & Dubois [DD03]) increases with p_0 as well. That is the limit: the hardest model for these algorithms is just the polynomial one!

Figure 2.3 and 2.4 represent the number of variables according to the proportion of solutions assigning them 1. In particular, frozen variables are those variables assigned 1 by none or all solutions. Note however that here we use **WalkSat** to find solutions, and it may happen that **WalkSat** finds several times the same solution. We can see that when $p_0 = 0$ there are few frozen variables (figure 2.3), whereas they are numerous when $p_0 = 0.25$ (figure 2.4).

Another indication of algorithmic difficulty is the presence of cores. The *core* of a solution is obtained as follows: start from the solution, remove all free variables and the clauses where they appear, and iterate until a fix point is reached. The remaining set of variables is called the core. We can see on figure 2.5 that for **Ambiguously Planted SAT**, the relative size of the solutions cores is 0 when $p_0 = 0$ but approaches 100% when $p_0 = 0.25$.

We add to these models an extra ingredient: fixing variables which are not already fixed in priority. More precisely, when a clause is signed so that it contains 1 true and 2 false literals (and consequently the true literal is fixed), with probability p we fix a variable which is already fixed. Complexity is difficult to measure for **WalkSat** for low values of p , because about half of the instances are very hard (cutoff of 10^8 iterations is reached for 90 variables in **Fixedly XOR Planted SAT**), and half are very easy, but for **kcnfs** complexity does not vary much, see figure 2.6. When $p = 0.2$, we obtain models which are significantly more difficult than without fixing for **WalkSat**, see figure 2.7.

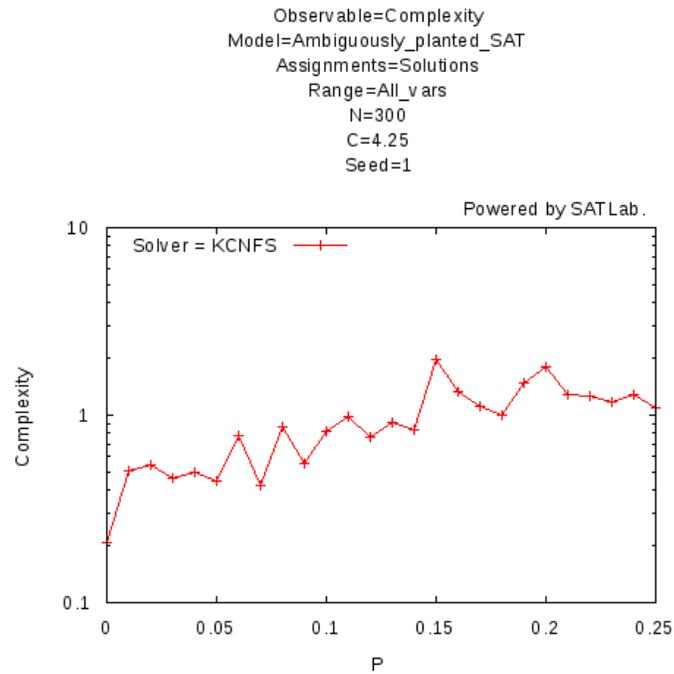


Figure 2.2: Complexity increases with p_0 in model Ambiguously Planted SAT.

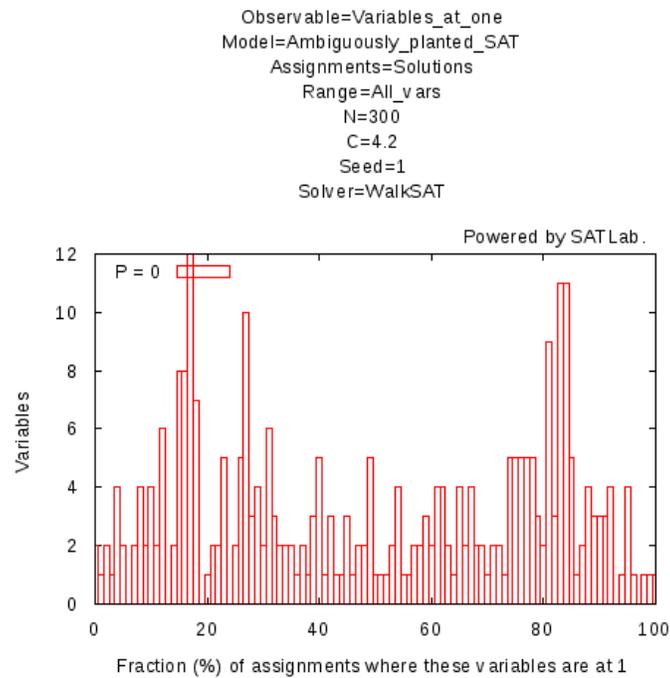


Figure 2.3: Number of variables according to the proportion of solutions assigning them 1.

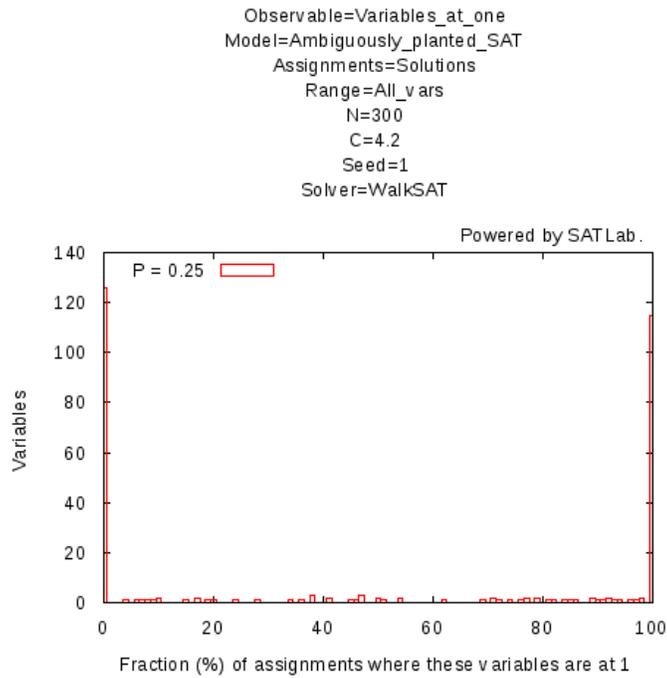


Figure 2.4: Number of variables according to the proportion of solutions assigning them 1.

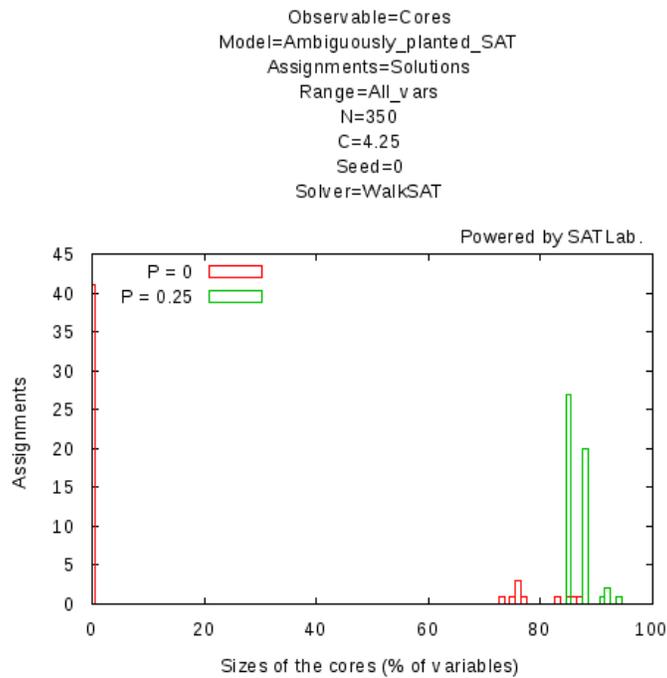


Figure 2.5: Size of the cores in Ambiguously Planted SAT.

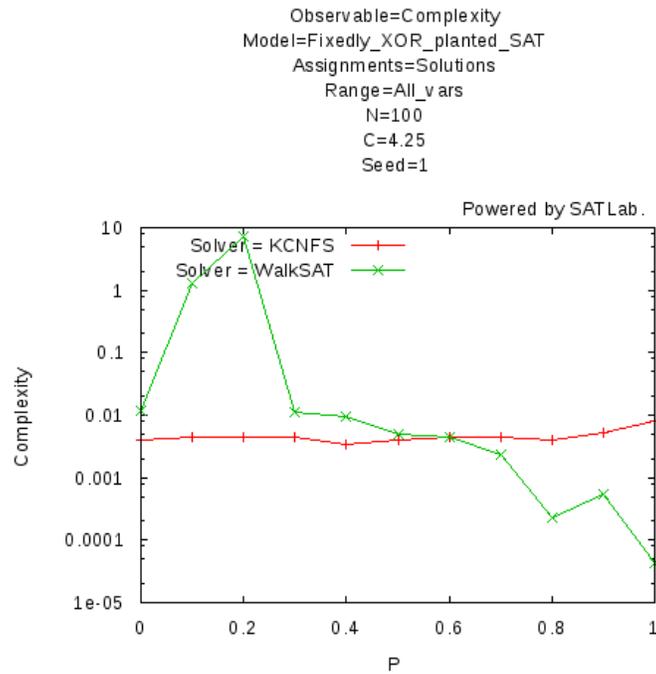
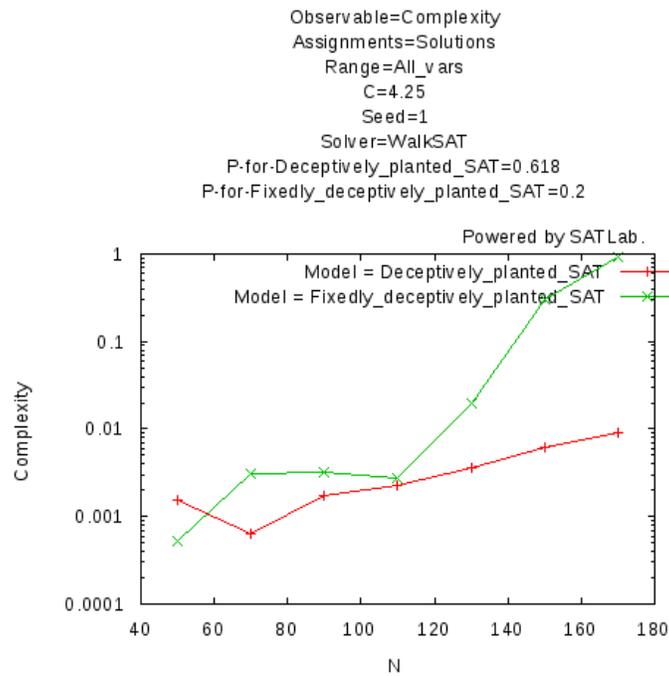
Figure 2.6: Complexity of Fixedly XOR Planted SAT with respect to p .

Figure 2.7: Effect of fixation on the complexity for WalkSat.

Part I

First Moment

Introduction to the First Moment Method

FIRST we make a brief technical introduction to the First Moment Method in section 3.1; then we present in section 3.2 a survey of the early attempts to upper bound the threshold of 3-SAT through the First Moment Method. The general idea is to count special kinds of solutions, which are present whenever there exist some solutions, but which are hopefully not too numerous. Ideally, to get exactly the threshold, one should count exactly 1 solution for all satisfiable formulas (and 0 for all unsatisfiable formulas). However, in practice, calculations lead us to count several if not many solutions per satisfiable formula.

In the next chapters we present two implementations of the First Moment Method: ordering and weighting. Ordering amounts to selecting some subset of solutions existing in all satisfiable formulas, whereas weighting consists in putting some weights onto solutions in order to count a weight of at least 1 for all satisfiable solutions. In chapter 4 we present several attempts to implement the First Moment Method by weighting valid partial assignments, along the same lines as Maneva, Mossel, Sinclair & Wainwright [MMW07, MS08]. In chapter 5, we introduce a general weighting scheme for CSPs, try to compare ordering and weighting, and give an improvement on Maneva’s weighting as an application of our general weighting scheme for CSPs. In chapter 6, we implement the First Moment Method on the distributional model of 3-SAT, transforming a weighting into an ordering to improve on the upper bound of 4.506 obtained by Dubois, Boufkhad & Mandler [DBM03].

3.1 How the First Moment Method Works

Let us recall how the First Moment Method works: given an event A , we want to show that $\Pr(A)$ is small but we don’t have access to $\Pr(A)$. Instead we use some EX . It suffices then to ensure that $\Pr(A) \leq EX$. For our problem 3-SAT, A is the event “a formula is satisfiable” and X is a non-negative random variable to be defined. Since $X \geq 0$, Markov’s inequality yields that $\Pr(X \geq 1) \leq EX$; so if we choose X such that A implies $X \geq 1$, we have

$$\Pr(A) \leq \Pr(X \geq 1) \leq EX .$$

Then our goal will be to tune X so that $EX \rightarrow 0$ for the least ratio $c = \frac{\text{\#clauses}}{\text{\#variables}}$, which gives an upper bound for the threshold of satisfiability.

What is the purpose of working with expectation instead of probability? In general expectation is much easier to handle than probability, because expectation is linear. Let X be the number of assignments having some property \mathcal{P} (\mathcal{P} might be “be a solution” or “be a black and red polka-dot solution”):

$$\begin{aligned} EX &= E \left(\sum_{\sigma \text{ assignment}} \mathbf{1}_{\sigma \in \mathcal{P}} \right) \\ &= \sum_{\sigma \text{ assignment}} E \mathbf{1}_{\sigma \in \mathcal{P}} \\ &= \sum_{\sigma \text{ assignment}} \Pr(\sigma \in \mathcal{P}) . \end{aligned}$$

In the case of satisfiability, due to NP-completeness, it is very difficult to estimate the probability for a formula to have some solutions; but it is much easier to estimate the probability for a formula to be satisfied by a *given* assignment.

3.2 Use of the First Moment Method for Upper-Bounding the Threshold of k-SAT

In this section we make a survey of different applications of the First Moment Method to upper bound the threshold of random k-SAT. The model considered here is uniform random drawing of k-SAT, as defined in section 1.1.3.

3.2.1 First Moment of Solutions

The first idea for X is simply the number of solutions. So what is the probability for a formula to be satisfied by a given assignment?

1. number of satisfied formulas:
 - a) we consider how each clause can be constructed:
 - i. all variables are allowed: n^k ;
 - ii. all signatures are allowed except the one which makes all literals false: $2^k - 1$;
 - b) there are cn clauses,
so the number of satisfied formulas is $\left((2^k - 1) n^k \right)^{cn}$;
2. total number of formulas: $(2^k n^k)^{cn}$.

It turns out that the probability for a formula to be satisfied by a given assignment does not depend on the considered assignment. There are 2^n possible assignments, so we have the following expression of EX :

$$EX = 2^n \left(1 - \frac{1}{2^k} \right)^{cn} .$$

It follows that EX tends to 0 as n tends to infinity as soon as $2 \left(1 - \frac{1}{2^k} \right)^c < 1$, i.e. $c > -\frac{\ln 2}{\ln \left(1 - \frac{1}{2^k} \right)}$. This is thus an upper bound of the satisfiability threshold of k-SAT. Note also that $-\frac{\ln 2}{\ln \left(1 - \frac{1}{2^k} \right)} \leq 2^k \ln 2$ and $-\frac{\ln 2}{\ln \left(1 - \frac{1}{2^k} \right)} \sim 2^k \ln 2$, so this upper bound is often referred to as $2^k \ln 2$.

Numerically, for 3-SAT this bound is 5.191, which is much above the experimentally observed value of around 4.25. All progress made so far in the quest of the threshold has consisted in a more refined choice of the random variable X , in combination with a more refined model of formulas (shown to be equivalent to the original one - for example models defined by a distribution, which we shall explore later in chapter 6).

Note that this very simple random variable was successfully used by [DM02] to establish a tight upper-bound on the threshold of 3-XOR-SAT, in combination with a slight modification of the random model, requiring each variable to have at least 2 occurrences.

3.2.2 First Moment of Implicants

The second idea for X is the number of implicants. An *implicant* is a partial assignment such that every assignment of the non-assigned variables will yield a solution. Of course, solutions are implicants. So in fact X is the number of implicants having a certain proportion $\alpha > 0$ of non-assigned variables.

In the standard drawing model of 3-SAT (as defined in section 1.1.3), El Maftouhi & de la Vega - 1995 [ED95] showed that at the ratio $c = 5.081$ almost all solutions may be grouped in a way that emphasizes a proportion of $\alpha = 0.02137$ so-called *dispensable* variables, which correspond to non-assigned variables of an implicant. Thus it is licit to count only those implicants having this fraction of non-assigned variables, which amounts to counting 1 solution out of $2^{\alpha n}$, and it turns out that this ratio $c = 5.081$ is an upper bound on the threshold.

As a refinement of this calculation, Kamath, Motwani, Palem & Spirakis - 1995 [KMPS95] show that at the ratio $c = 4.87$ (and numerically at $c = 4.762$) the same phenomenon occurs.

3.2.3 First Moment of Prime Implicants

Another way to restrict implicants (which by default contain solutions) is to consider prime implicants. A *prime implicant* is an implicant such that if any of the assigned variables is flipped, this is no longer an implicant (thus it is a kind of minimal implicant). As shown by Boufkhad & Dubois [BD99], the number of prime implicants of a random formula is exponentially smaller than the number of solutions. It turns out that in the standard drawing model of 3-SAT (as defined in section 1.1.3), the First Moment Method with X taken to be the number of prime implicants yields an upper bound on the threshold of $c = 4.88$.

3.2.4 First Moment of Negatively Prime Solutions

A major breakthrough in the use of the First Moment Method for upper-bounding the threshold of k-SAT consisted in counting the Negatively Prime Solutions (NPSs) of a formula. This technique was presented in 1997 by Dubois & Boufkhad [DB97] and by Kirousis, Kranakis & Krizanc [KKK97]. A *Negatively Prime Solution* (NPS) is a solution where every free variable are assigned 1. A *free variable* in a solution is a variable which when flipped (from 0 to 1 or conversely) yields another solution. It turns out that in the standard drawing model of 3-SAT (as defined in section 1.1.3), the First Moment Method with X taken to be the number of negatively prime solutions yields an upper bound on the threshold of $c = 4.667$ or $c = 4.643$, depending on the chosen parameters. A refinement of this technique was made by Kirousis, Kranakis, Krizanc & Stamatiou - 1998 [KKKS98], as they counted maximal solutions within a distance of 2 (thus allowing double-flips). They got an upper bound of 4.601, but calculations become quite involved.

All better bounds shown so far are based on the framework of NPSs, as can be seen in the sequel.

Weighted First Moment Method

LET us recall the fundamental requirement for the First Moment Method, as emphasized in section 3.1: find a non-negative random variable X such that for each satisfiable formula, $X \geq 1$. Note in particular that X is not required to be integer-valued! That leads us to consider the possibility of putting non-negative real weights onto solutions. X would be the total weight of all solutions. To do so we use the notion of (valid) partial assignment and the Weight Preservation Theorem introduced by Maneva, Mossel & Wainwright - 2007 [MMW07].

4.1 Partial Assignments

A *partial assignment* or *valuation* is an assignment taking its values in the domain $\{0, 1, *\}$. A *valid partial assignment* or *valid valuation* is a partial assignment such that no clause contain literals which are all false or all false but one starred. Thus the meaning of the $*$ value is that to some extent starred variable may be given a certain value 0 or 1 to build a solution. However the presence of some valid partial assignments does not guarantee the presence of solutions, since the all-star assignment is always valid.

We call *starrable variable* a variable of a valid partial assignment assigned 0 or 1 such that assigning it $*$ instead still yields a valid partial assignment. We call *invertible variable* a variable of a valid partial assignment assigned 0 or 1 such that flipping its value (to 1 or 0 respectively) still yields a valid partial assignment. Note that an invertible variable is starrable but the converse may not be true. It turns out that Maneva, Mossel & Wainwright [MMW07] considered only starrable variables (which they called *unconstrained*) but we shall need to make the distinction between starrable and invertible variables later in section 5.7.

4.1.1 Directed Graph and Partial Ordering

Let us consider the following *coarsening* process: given a valid partial assignment σ , choose a starrable variable x and put a star instead of its current value. This yields another valid partial assignment τ differing from σ only on variable x . We put an arc $\sigma \rightarrow \tau$ and equivalently we define a relation $\sigma > \tau$. Finally we embed into $>$ its transitive closure. Since coarsening strictly increases the number of stars in the assignments, the oriented graph obtained by \rightarrow is circuit-free, and $>$ is a partial strict order, whose minimal elements are valid assignments without any

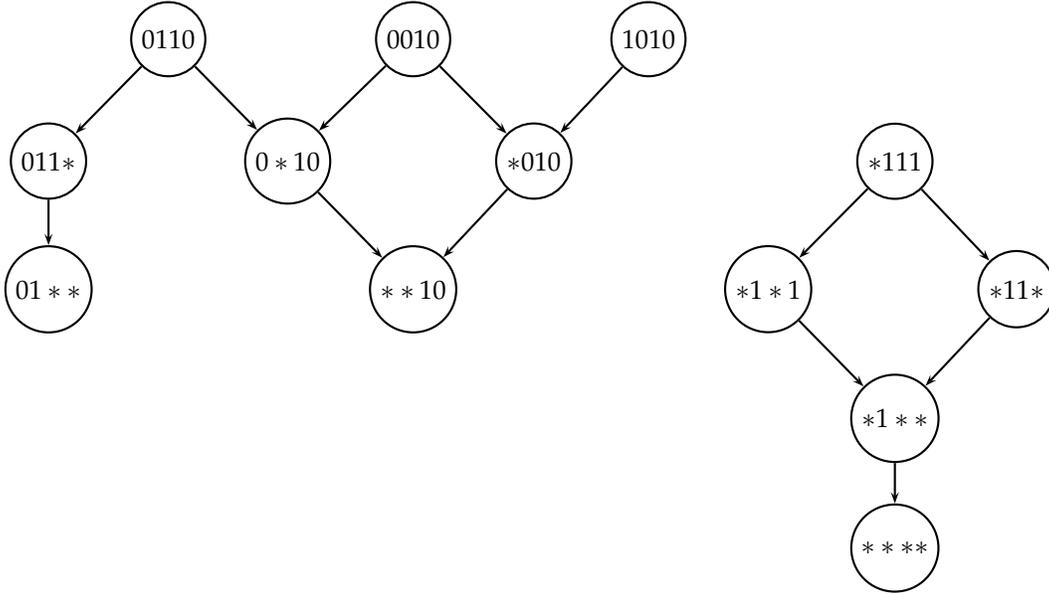


Figure 4.1: Coarsening graph of some valid partial assignments. Each (horizontal) level corresponds to a given number of stars. There are 3 minimal elements: 01^{**} , $^{**}10$ and *** . Note that this particular graph is not connected.

starrable variables. Moreover the graph may be sliced into levels characterized by the number of stars, as shown in figure 4.1. On the top level are the boolean solutions, and on the bottom level is the all-star assignment (because it is always valid).

4.1.2 Weight Preservation Theorem

Maneva, Mossel & Wainwright - 2007 [MMW07], and later Maneva & Sinclair [MS08] and Ardila & Maneva [AM09] define the following weighting system: given two non-negative reals ω_o and ω_* , the weight of a valid partial assignment σ is $W(\sigma) = \omega_o^{n_o(\sigma)} \omega_*^{n_*(\sigma)}$, where $n_o(\sigma)$ is the number of starrable variables in σ and $n_*(\sigma)$ is the number of starred variables in σ . And here is the fundamental result they establish:

Theorem 4.1. (Weight Preservation Theorem, [MMW07, MS08, AM09]). *If σ is a valid partial assignment and $\omega_o + \omega_* = 1$, then $\sum_{\tau \leq \sigma} W(\tau) = \omega_*^{n_*(\sigma)}$.*

In fact Ardila & Maneva - 2009 [AM09] show that this identity characterizes the structure of convex geometries, and it is immediate to see that coarsening graphs in this framework are convex geometries, because a starrable variable remains starrable when other variables get starred. We shall refer to this property as the *convexity condition*.

Corollary 4.2. *If σ is a boolean solution and $\omega_o + \omega_* = 1$, then $\sum_{\tau \leq \sigma} W(\tau) = 1$.*

Hurrah! This is a wonderful candidate for the First Moment Method: take X to be the weight of all valid partial assignments. Corollary 4.2 guarantees that $X \geq 1$ whenever the formula is satisfiable. The incentive of this method is that several solutions may share the same valid partial assignments below them. However, we may count too many valid partial

assignments, because some of them are not reachable from any solution. For example, the all-star assignment will always be counted with a weight of ω_*^n , even if the formula is not satisfiable. So what bound shall we get here?

We shall focus our attention on 3-SAT. We shall denote FFF a clause with 3 false literals, FF* a clause with 2 false and 1 starred literal etc. Maneva, Mossel & Wainwright consider that a partial assignment is valid when there no clause of type FFF or FF*, but we can imagine some other kinds of validity. It is enough to check that the convexity condition is satisfied.

4.2 Trivial Kind of Validity: FFF

Here is the most simple kind of validity we can imagine. A partial assignment is valid iff it does not build FFF clauses. In particular FF* is allowed, and every non-starred variable is starrable. So the convexity condition holds.

Notations.

X : total weight of valid partial assignments;

α_o : proportion of starrable variables (with an associated weight of ω_o);

α_* : proportion of starred variables (with an associated weight of ω_*).

Expression of the first moment. The first moment of X can be split up according to the repartition of variables into starrable and starred: $\sum_{\alpha_* + \alpha_o = 1} \binom{n}{\alpha_* n, \alpha_o n}$;

1. the $\alpha_o n$ non-starred variables are assigned 0 or 1: $2^{\alpha_o n}$ possibilities;
2. weight of $\omega_*^{\alpha_* n} \omega_o^{\alpha_o n}$;
3. probability for a partial assignment to be valid:
 - a) number of ways to build a satisfied formula: $(8 \binom{n}{3} - \binom{\alpha_o n}{3})^{cn}$;
 - b) number of ways to build a formula: $(8 \binom{n}{3})^{cn}$.

We denote by \mathcal{P} the set of all families of non-negative numbers (α_*, α_o) satisfying the following constraint:

$$\alpha_* + \alpha_o = 1 .$$

We denote by $\mathcal{I}(n)$ the intersection of \mathcal{P} with the multiples of $\frac{1}{n}$; we get the following expression of the first moment:

$$EX = \sum_{(\alpha_*, \alpha_o) \in \mathcal{I}(n)} T_1(n)$$

where

$$T_1(n) = \sum_{\alpha_* + \alpha_o = 1} \binom{n}{\alpha_* n, \alpha_o n} 2^{\alpha_o n} \omega_*^{\alpha_* n} \omega_o^{\alpha_o n} \frac{(8 \binom{n}{3} - \binom{\alpha_o n}{3})^{cn}}{(8 \binom{n}{3})^{cn}} .$$

We get rid of multinomials thanks to inequalities B.2, so $EX \leq \text{poly}(n) F_1^n$ where

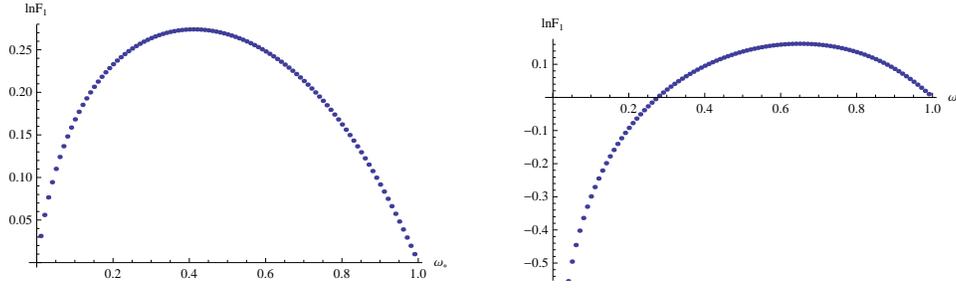


Figure 4.2: Here is a plot of $\ln F_1$ with respect to ω_* at $c = 5.191$ and at $c = 15$. The best choice for ω_* is thus 0.

$$F_1 = \alpha_*^{-\alpha_*} \alpha_0^{-\alpha_0} 2^{\alpha_0} \omega_*^{\alpha_*} \omega_0^{\alpha_0} \left(1 - \frac{\alpha_0^3}{8}\right)^c ;$$

$$F_1 = \alpha_*^{-\alpha_*} (1 - \alpha_*)^{-(1-\alpha_*)} 2^{(1-\alpha_*)} \omega_*^{\alpha_*} \omega_0^{(1-\alpha_*)} \left(1 - \frac{(1-\alpha_*)^3}{8}\right)^c .$$

Maximization.

- if $\omega_* = 0$, then we are precisely counting solutions, and the critical c is $c = 5.191$;
- if $\omega_* = 1$, then the all-star assignment has a weight of 1 and the critical c is $c = +\infty$;
- otherwise, numerical calculations reveal that it is even worse than counting only solutions, since the critical value of c increases from 5.191 to $+\infty$ when ω_* goes from 0 to 1. So the best choice for ω_* is 0, as suggests figure 4.2.

4.3 Generic Calculation for Various Kinds of Validity

The previous calculation shows that defining validity by excluding FFF alone is not very relevant. So in this section we are going to redo the calculations in a framework allowing more elaborated forms of validity (such as FFF+FF*).

Notations.

X : total weight of valid partial assignments

α_0 : proportion of starrable variables (with an associated weight of ω_0);

α_c : proportion of unstarrable variables (with an associated weight of $\omega_c = 1$);

α_* : proportion of starred variables (with an associated weight of ω_*);

β_1 : proportion of constraining clauses (i.e. clauses that make a variable unstarrable);

β_2 : proportion of other clauses.

Expression of the first moment. The first moment of X can be split up according to the repartition of variables into starrable, unstarrable and starred: $\sum_{\alpha_* + \alpha_c + \alpha_0 = 1} \binom{n}{\alpha_* n, \alpha_c n, \alpha_0 n}$;

1. the $(1 - \alpha_*) n$ non-starred variables are assigned 0 or 1: $2^{(1-\alpha_*)n}$ possibilities;

2. weight of $\omega_*^{\alpha_* n} \omega_0^{\alpha_0 n}$;
3. probability for a partial assignment to be valid:
 - a) number of ways to build a satisfied formula, decomposed by the proportion of constraining clauses: $\sum_{\beta_1 + \beta_2 = 1} \binom{cn}{\beta_1 cn, \beta_2 cn}$;
 - i. choice of the variable made unstarrable in each constraining clause:
 $S(\beta_1 cn, \alpha_c n) (\alpha_c n)!$ (Stirling number of the second kind, see appendix B.2);
 - ii. choice of the other literals in constraining clauses: $\hat{C}(\alpha_*)^{\beta_1 cn}$ (to be precised below);
 - iii. choice of the literals in the other clauses: $\hat{U}(\alpha_*)^{\beta_2 cn}$ (to be precised below);
 - b) number of ways to build a formula: $(8 \binom{n}{3})^{cn}$.

We denote by \mathcal{P} the set of all families of non-negative numbers $(\alpha_*, \alpha_o, \alpha_c, \beta_1, \beta_2)$ satisfying the following constraints:

$$\alpha_* + \alpha_o + \alpha_c = 1 ; \quad (4.1)$$

$$\beta_1 + \beta_2 = 1 . \quad (4.2)$$

We denote by $\mathcal{I}(n)$ the intersection of \mathcal{P} with the multiples of $\frac{1}{n}$; we get the following expression of the first moment:

$$EX = \sum_{(\alpha_*, \alpha_o, \alpha_c, \beta_1, \beta_2) \in \mathcal{I}(n)} T_1(n)$$

where

$$T_1(n) = \sum_{\alpha_* + \alpha_c + \alpha_o = 1} \binom{n}{\alpha_* n, \alpha_c n, \alpha_o n} 2^{(1-\alpha_*)n} \omega_*^{\alpha_* n} \omega_0^{\alpha_o n} \cdot \sum_{\beta_1 + \beta_2 = 1} \binom{cn}{\beta_1 cn, \beta_2 cn} S(\beta_1 cn, \alpha_c n) (\alpha_c n)! \frac{\hat{C}(\alpha_*)^{\beta_1 cn} \hat{U}(\alpha_*)^{\beta_2 cn}}{(8 \binom{n}{3})^{cn}} .$$

We get rid of multinomials thanks to inequalities B.2 and of Stirling numbers thanks to Temme's estimates (see appendix B.2), so $T_1(n) \leq \text{poly}(n) F_1^n$ where

$$F_1 = \alpha_*^{-\alpha_*} \alpha_c^{-\alpha_c} \alpha_o^{-\alpha_o} 2^{1-\alpha_*} \omega_*^{\alpha_*} \omega_0^{\alpha_o} \left(\beta_1^{-\beta_1} \beta_2^{-\beta_2} \right)^c \cdot (e^x - 1)^{\alpha_c} \left(\frac{\beta_1 c}{e^x} \right)^{\beta_1 c} \left(\frac{6}{8} \right)^c C(\alpha_*)^{\beta_1 c} U(\alpha_*)^{\beta_2 c} ,$$

with x defined as the positive solution to the following equation:

$$\frac{\alpha_c x}{\beta_1 c} = 1 - e^{-x} .$$

4.3.1 Application to FFF + FF*

Valid partial assignment: assignment without clause FFF or FF*;

Constraining clauses: FFT.

Note that this kind of validity is precisely the one introduced by Maneva, Mossel & Wainwright - 2007 [MMW07].

What are the expressions of \hat{C} and \hat{U} ?

1. for \hat{C} , choose 2 false literals;
2. for \hat{U} , remove from all possibilities clauses of the following types: FFF, FF*, and FFT.

$$\begin{aligned}\hat{C}(\alpha_*) &= \binom{(1-\alpha_*)n}{2}; \\ \hat{U}(\alpha_*) &= 8\binom{n}{3} - \binom{(1-\alpha_*)n}{3} - 2\alpha_*n\binom{(1-\alpha_*)n}{2} - 3\binom{(1-\alpha_*)n}{3}.\end{aligned}$$

Thus

$$\begin{aligned}C(\alpha_*) &= \frac{(1-\alpha_*)^2}{2}; \\ U(\alpha_*) &= \frac{8}{6} - \frac{(1-\alpha_*)^3}{6} - 2\alpha_*\frac{(1-\alpha_*)^2}{2} - 3\frac{(1-\alpha_*)^3}{6};\end{aligned}$$

i.e.

$$\begin{aligned}C(\alpha_*) &= \frac{(1-\alpha_*)^2}{2}; \\ U(\alpha_*) &= \frac{8}{6} - 4\frac{(1-\alpha_*)^3}{6} - \alpha_*(1-\alpha_*)^2.\end{aligned}$$

Maximization.

- if $\omega_* = 0$, then we are precisely counting solutions, and the critical c is $c = 5.191$;
- if $\omega_* = 1$, then the all-star assignment has a weight of 1 and the critical c is $c = +\infty$;
- otherwise, numerical calculations reveal that the best upper bound we can get is 4.883, which is obtained for $\omega_* = 0.587$, as shown on figure 4.3.

4.3.2 Application to FFF + FF* + F**

Valid partial assignment: assignment without clause FFF, FF* or F**;

Constraining clauses: FFT and F*T.

Note that the convexity condition does not hold, since T variables are starrable in FTT, but not in F*T. So it is not sure whether we can apply the Weight Preservation Theorem. However the result we get (4.866) is not good either...

What are the expressions of \hat{C} and \hat{U} ?

1. for \hat{C} , choose 2 false literals or (1 false and 1 starred literals);
2. for \hat{U} , remove from all possibilities clauses of the following types: FFF, FF*, F**, FFT and F*T.

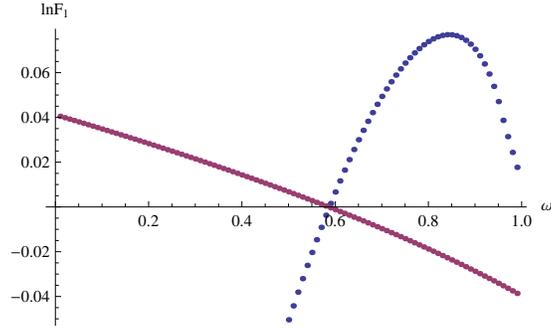


Figure 4.3: FFF+FF*: here is a plot of the value of $\ln F_1$ with respect to ω_* for 2 families of solutions canceling out the derivative of $\ln F_1$ and satisfying the constraints, at $c = 4.883$. In fact there exists at least another family of solutions, but we can see here that we cannot get a better upper bound than 4.883.

$$\begin{aligned}\hat{C}(\alpha_*) &= \binom{(1-\alpha_*)n}{2} + 2\alpha_* (1-\alpha_*)n^2 ; \\ \hat{U}(\alpha_*) &= 8\binom{n}{3} - \binom{(1-\alpha_*)n}{3} - 2\alpha_*n \binom{(1-\alpha_*)n}{2} - 4\binom{\alpha_*n}{2} (1-\alpha_*)n \\ &\quad - 3\binom{(1-\alpha_*)n}{3} - 4\binom{(1-\alpha_*)n}{2} \alpha_*n .\end{aligned}$$

It follows that:

$$\begin{aligned}C(\alpha_*) &= \frac{(1-\alpha_*)(1+3\alpha_*)}{2} ; \\ U(\alpha_*) &= \frac{2-3\alpha_*+6\alpha_*^2-\alpha_*^3}{3} .\end{aligned}$$

Maximization.

- if $\omega_* = 0$, then we are precisely counting solutions, and the critical c is $c = 5.191$;
- if $\omega_* = 1$, then the all-star assignment has a weight of 1 and the critical c is $c = +\infty$;
- otherwise, numerical calculations reveal that the best critical c we can get is $c = 4.866$, obtained for $\omega_* = 0.999$, as shown on figure 4.4.

4.3.3 Application to FFF + FF* + F** + ***

Valid partial assignment: assignment without clause FFF, FF*, F** or ***;

Constraining clauses: FFT, F*T and **T.

Note that the convexity condition does not hold, since T variables are starrable in FTT, but not in F*T. However this kind of validity is well-known. . . since valid partial assignments here are precisely implicants. So in fact we make a calculation on weighted implicants, and we are not sure it is correct with respect to the First Moment Method.

What are the expressions of \hat{C} and \hat{U} ?

1. for \hat{C} , choose 2 false literals or (1 false and 1 starred literals) or 2 starred literals;
2. for \hat{U} , remove from all possibilities clauses of the following types: FFF, FF*, F**, ***, FFT, F*T and **T.

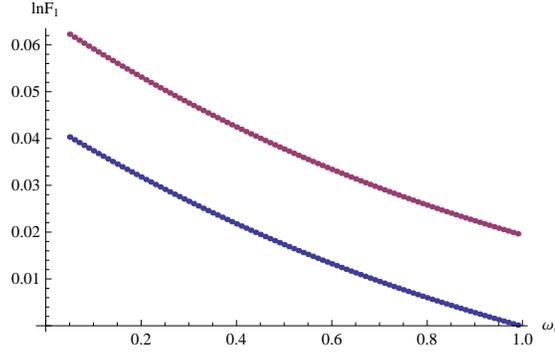


Figure 4.4: FFF+FF*+F***: here is a plot of the value of $\ln F_1$ with respect to ω_* for 1 family of solutions canceling out the derivative of $\ln F_1$ and satisfying the constraints, at $c = 4.866$ (below) and at $c = 4.7$ (above). In fact there exist at least two other families of solutions, but we can see here that we cannot get a better upper bound than 4.866.

$$\begin{aligned}\hat{C}(\alpha_*) &= \binom{(1-\alpha_*)n}{2} + 2\alpha_* (1-\alpha_*)n^2 + 4\binom{\alpha_*n}{2}; \\ \hat{U}(\alpha_*) &= 8\binom{n}{3} - \binom{(1-\alpha_*)n}{3} - 2\alpha_*n\binom{(1-\alpha_*)n}{2} - 4\binom{\alpha_*n}{2}(1-\alpha_*)n - 8\binom{\alpha_*n}{3} \\ &\quad - 3\binom{(1-\alpha_*)n}{3} - 4\binom{(1-\alpha_*)n}{2}\alpha_*n - 4\binom{\alpha_*n}{2}(1-\alpha_*)n.\end{aligned}$$

Thus

$$\begin{aligned}C(\alpha_*) &= \frac{(1-\alpha_*)^2}{2} + 2\alpha_*(1-\alpha_*) + 4\frac{\alpha_*^2}{2} \\ U(\alpha_*) &= \frac{8}{6} - \frac{(1-\alpha_*)^3}{6} - 2\alpha_*\frac{(1-\alpha_*)^2}{2} - 4\frac{\alpha_*^2}{2}(1-\alpha_*) - 8\frac{\alpha_*^3}{6} \\ &\quad - 3\frac{(1-\alpha_*)^3}{6} - 4\frac{(1-\alpha_*)^2}{2}\alpha_* - 4\frac{\alpha_*^2}{2}(1-\alpha_*)\end{aligned}$$

and after simplifications:

$$\begin{aligned}C(\alpha_*) &= \frac{(1+\alpha_*)^2}{2}; \\ U(\alpha_*) &= \frac{1}{3}(1-\alpha_*)^2(2+\alpha_*) .\end{aligned}$$

Maximization.

- if $\omega_* = 0$, then we are precisely counting solutions, and the critical c is $c = 5.191$;
- if $0 < \omega_* < 1$, then numerical calculations reveal that the best critical c we can get is 4.883, obtained for $\omega_* = 0.9999$, see figure 4.5.
- if $\omega_* = 1$, this case must be dealt with separately, because here the all-star assignment is not valid. What does it represent? Precisely prime implicants! Why? because we put a zero weight onto implicants which have some starrable variables.

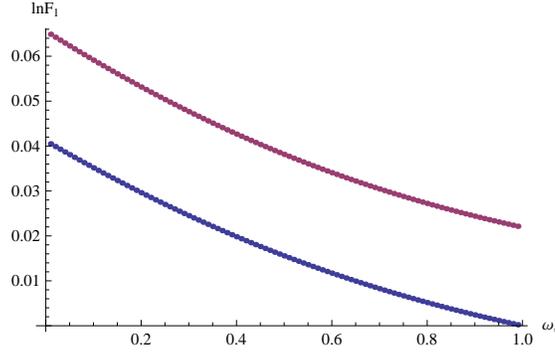


Figure 4.5: FFF+FF*+F**+***: here is a plot of the value of $\ln F_1$ with respect to ω_* for 1 family of solutions canceling out the derivative of $\ln F_1$ and satisfying the constraints, at $c = 4.883$ (below) and at $c = 4.7$ (above). In fact there exist at least two other families of solutions, but we can see here that we cannot get a better upper bound than 4.883.

Here we have $\omega_o = 0$, so $\alpha_o = 0$ as well, and thus

$$F_1 = \alpha_*^{-\alpha_*} \alpha_c^{-\alpha_c} 2^{1-\alpha_*} \left(\beta_1^{-\beta_1} \beta_2^{-\beta_2} \right)^c \cdot (e^x - 1)^{\alpha_c} \left(\frac{\beta_1 c}{ex} \right)^{\beta_1 c} \left(\frac{6}{8} \right)^c C (\alpha_*)^{\beta_1 c} U (\alpha_*)^{\beta_2 c} .$$

Numerically, we find that c is critical at $c = 4.883$.

4.4 Combination of Stars and NPSs

The Weight Preservation Theorem (theorem 4.1) insures that the total weight of partial assignments below a given solution is 1. Since NPSs (see definition in section 3.2.4) are special solutions, the total weight below them is 1 as well. So the idea here is to discard those valid partial assignments which cannot be under any NPS. How to do so? We require any variable at 0 to be T either in some TFF clause (so it is blocked) or in some TF* or T** clause (so it may be under a solution where it is blocked). This restriction still counts all valid partial assignments located under any NPS, so we count 1 under each NPS.

Notations.

X : total weight of valid partial assignments

$\alpha_{o,1}$: proportion of starrable variables at 1 (with an associated weight of ω_o);

$\alpha_{o,0}$: proportion of starrable variables at 0 (with an associated weight of ω_o);

α_c : proportion of unstarrable variables (with an associated weight of $\omega_c = 1$);

α_* : proportion of starred variables (with an associated weight of ω_*);

β_1 : proportion of clauses making a (true) variable unstarrable (i.e. the clauses of type TFF);

β_2 : proportion of clauses making a (true) starrable variable at 0 potentially below a NPS (i.e. some of the clauses of type TF* and T** where T is at value 0);

β_3 : proportion of other clauses.

Expression of the first moment. The first moment of X can be split up according to the repartition of variables into starrable, unstarrable and starred: $\sum_{\alpha_* + \alpha_c + \alpha_{o,0} + \alpha_{o,1} = 1} \binom{n}{\alpha_* n, \alpha_c n, \alpha_{o,0} n, \alpha_{o,1} n}$;

1. the $\alpha_c n$ non-starred and non-assigned variables are assigned 0 or 1: $2^{\alpha_c n}$ possibilities;
2. weight of $\omega_*^{\alpha_* n} \omega_o^{\alpha_{o,0} n + \alpha_{o,1} n}$;
3. probability for a partial assignment to be valid:
 - a) number of ways to build a satisfied formula, decomposed by the proportion of constraining clauses: $\sum_{\beta_1 + \beta_2 + \beta_3 = 1} \binom{cn}{\beta_1 cn, \beta_2 cn, \beta_3 cn}$;
 - i. choice of the variable made unstarrable in each constraining clause: $S(\beta_1 cn, \alpha_c n) (\alpha_c n)!$;
 - ii. choice of the other literals in constraining clauses: $\hat{C}(\alpha_*)^{\beta_1 cn}$ (to be precised below);
 - iii. choice of the free true variable at 0 potentially below a NPS: $S(\beta_2 cn, \alpha_{o,0} n) (\alpha_{o,0} n)!$
 - iv. choice of the other literals in these clauses: $\hat{P}(\alpha_*)^{\beta_2 cn}$ (to be precised below);
 - v. choice of the literals in the other clauses: $\hat{U}(\alpha_*, \alpha_{o,0})^{\beta_3 cn}$ (to be precised below);
 - b) number of ways to build a formula: $(8 \binom{n}{3})^{cn}$.

We denote by \mathcal{P} the set of all families of non-negative numbers $(\alpha_*, \alpha_{o,0}, \alpha_{o,1}, \alpha_c, \beta_1, \beta_2)$ satisfying the following constraints:

$$\alpha_* + \alpha_{o,0} + \alpha_{o,1} + \alpha_c = 1 ; \quad (4.3)$$

$$\beta_1 + \beta_2 + \beta_3 = 1 . \quad (4.4)$$

Note that \mathcal{P} is convex (by linearity of constraints). We denote by $\mathcal{I}(n)$ the intersection of \mathcal{P} with the multiples of $\frac{1}{n}$; we get the following expression of the first moment:

$$EX = \sum_{(\alpha_*, \alpha_{i,0}, \alpha_{i,1}, \alpha_s, \alpha_c, \beta_1, \beta_2, \beta_3) \in \mathcal{I}(n)} T_1(n)$$

where

$$T_1(n) = \sum_{\alpha_* + \alpha_c + \alpha_{o,0} + \alpha_{o,1} = 1} \binom{n}{\alpha_* n, \alpha_c n, \alpha_{o,0} n, \alpha_{o,1} n} 2^{\alpha_c n} \omega_*^{\alpha_* n} \omega_o^{\alpha_{o,0} n + \alpha_{o,1} n} \cdot \sum_{\beta_1 + \beta_2 + \beta_3 = 1} \binom{cn}{\beta_1 cn, \beta_2 cn, \beta_3 cn} S(\beta_1 cn, \alpha_c n) (\alpha_c n)! S(\beta_2 cn, \alpha_{o,0} n) (\alpha_{o,0} n)! \cdot \frac{\hat{C}(\alpha_*)^{\beta_1 cn} \hat{P}(\alpha_*)^{\beta_2 cn} \hat{U}(\alpha_*, \alpha_{o,0})^{\beta_3 cn}}{(8 \binom{n}{3})^{cn}} .$$

We get rid of multinomials thanks to inequalities B.2 and of Stirling numbers thanks to Temme's estimates (see appendix B.2), so $T_1(n) \leq \text{poly}(n) F_1^n$ where

$$F_1 = \alpha_*^{-\alpha_*} \alpha_c^{-\alpha_c} \alpha_{o,0}^{-\alpha_{o,0}} \alpha_{o,1}^{-\alpha_{o,1}} 2^{\alpha_c} \omega_*^{\alpha_*} \omega_o^{\alpha_{o,0} + \alpha_{o,1}} \left(\beta_1^{-\beta_1} \beta_2^{-\beta_2} \beta_3^{-\beta_3} \right)^c \\ \cdot (e^x - 1)^{\alpha_c} \left(\frac{\beta_{1c}}{ex} \right)^{\beta_{1c}} (e^y - 1)^{\alpha_{o,0}} \left(\frac{\beta_{2c}}{ey} \right)^{\beta_{2c}} \left(\frac{6}{8} \right)^c C(\alpha_*)^{\beta_{1c}} P(\alpha_*)^{\beta_{2c}} U(\alpha_*, \alpha_{o,0})^{\beta_{3c}},$$

with x and y defined as the positive solutions to the following equations:

$$\frac{\alpha_c x}{\beta_{1c}} = 1 - e^{-x}; \\ \frac{\alpha_{o,0} y}{\beta_{2c}} = 1 - e^{-y}.$$

What are the expressions of \hat{C} , \hat{P} and \hat{U} ?

1. for \hat{C} , just choose 2 false literals;
2. for \hat{P} , just choose 1 false and 1 starred literals, or 2 starred literals;
3. for \hat{U} , it is a little trickier: we remove from all possibilities clauses of the following types: FFF, FF*, TFF, T** with T among $\alpha_{o,0}$, TF* with T among $\alpha_{o,0}$ and F not among $\alpha_{o,0}$, and TF* with both T and F among $\alpha_{o,0}$.

$$\hat{C}(\alpha_*) = \binom{(1 - \alpha_*)n}{2}; \\ \hat{P}(\alpha_*) = 4 \binom{\alpha_* n}{2} + 2\alpha_* (1 - \alpha_*) n^2; \\ \hat{U}(\alpha_*, \alpha_{o,0}) = 8 \binom{n}{3} - \binom{(1 - \alpha_*)n}{3} - 2\alpha_* n \binom{(1 - \alpha_*)n}{2} \\ - 3 \binom{(1 - \alpha_*)n}{3} - 4\alpha_{o,0} n \binom{\alpha_* n}{2} - 2\alpha_* (1 - \alpha_* - \alpha_{o,0}) \alpha_{o,0} n^3 - 4\alpha_* n \binom{\alpha_{o,0} n}{2}.$$

Thus

$$C(\alpha_*) = \frac{(1 - \alpha_*)^2}{2} \\ P(\alpha_*) = 4 \frac{\alpha_*^2}{2} + 2\alpha_* (1 - \alpha_*) \\ U(\alpha_*, \alpha_{o,0}) = \frac{8}{6} - \frac{(1 - \alpha_*)^3}{6} - 2\alpha_* \frac{(1 - \alpha_*)^2}{2} \\ - 3 \frac{(1 - \alpha_*)^3}{6} - 4\alpha_{o,0} \frac{\alpha_*^2}{2} - 2\alpha_* (1 - \alpha_* - \alpha_{o,0}) \alpha_{o,0} - 4\alpha_* \frac{\alpha_{o,0}^2}{2}$$

and finally

$$C(\alpha_*) = \frac{(1 - \alpha_*)^2}{2}; \\ P(\alpha_*) = 2\alpha_*; \\ U(\alpha_*, \alpha_{o,0}) = \frac{4}{3} - \frac{2(1 - \alpha_*)^3}{3} - \alpha_* (1 - \alpha_*)^2 - 2\alpha_{o,0} \alpha_*.$$

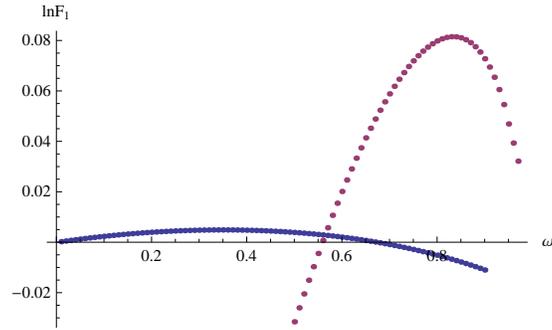


Figure 4.6: Here is a plot of the value of $\ln F_1$ with respect to ω_* for 2 families of solutions canceling out the derivative of $\ln F_1$ and satisfying the constraints, at $c = 4.643$. In fact there exists at least another family of solutions, but we can see here that we cannot get a better upper bound than 4.643.

Table 4.1: Recap of the bounds obtained by weighted First Moment Method on partial assignments.

framework	best ω_*	equivalent	upper bound
FFF	0	solutions	5.191
FFF+FF*	0.587	-	4.883
FFF+FF*+F**	0.999	-	4.866
FFF+FF*+F**+***	1	prime implicants	4.883
partial assignments under NPSs	0	NPSs	4.643

Maximization.

- if $\omega_* = 0$, then we are precisely counting NPSs, and the critical c is $c = 4.643$;
- if $\omega_* = 1$, then the all-star assignment has a weight of 1 and the critical c is $c = +\infty$;
- otherwise, numerical calculations reveal that the best critical c we obtain is $c = 4.643$, obtained for $\omega_* = 0.001$, as can be checked on figure 4.6.

4.5 Conclusion and Perspectives

In this chapter we presented some attempts we made to implement the First Moment Method using weights and partial assignments. All bounds we found are summed up in table 4.1.

We are not able to explain why both FFF+FF* and FFF+FF*+F**+*** give the same bound of 4.883. Is there a formal kind of equivalence between both frameworks?

It turns out that in the end we could not establish a better upper bound than 4.643, which is the bound obtained by merely counting NPSs. In the next chapter we investigate some connections between weighting and counting, and we build a better weighting system on partial assignments than Maneva's.

Weighting and Ordering Solutions

 THE First Moment Method requires to count 1 for each satisfiable instance, as discussed in section 3.1. In section 3.2 we saw the early attempts to upper bound satisfiability by selecting some varieties of solutions (implicants or NPSs) which must exist in every satisfiable instance; we shall refer to this process as *ordering*. In chapter 4 we investigated an attempt to put some weights onto solutions in a way that gives a weight of at least 1 to satisfiable instances; we shall refer to this process as *weighting*. Now the purpose of this chapter is to introduce a correct weighting for multi-valued CSPs (see definition in section 5.2 below) and to compare ordering and weighting when possible.

The NP-completeness of these problems in general makes it difficult to determine whether a given instance is satisfiable; that may explain why direct counting of satisfiable instances is currently unfeasible. However, precisely because these problems are in NP, it is easy to determine whether some instance is satisfied by a given valuation and then to count the formulas satisfied by this valuation. Thus counting couples (formulas, solutions) is only accessible starting from a solution; moreover, given a solution, it is not complicated to investigate also its immediate neighborhood. But even at a distance of 2, i.e. with neighbors of neighbors, calculations become quite complicated (see Kirousis, Kranakis, Krizanc & Stamatiou - 1998 [KKKS98]). This fact imposes a strong restriction on the design of both estimation techniques studied: they can only make use of local information. We shall refer to this as *the locality condition*.

So in this framework, ordering consists in putting a total order on the domain, which induces an orientation between neighboring solutions in a way that prevents circuits from appearing, and then counting only minimal elements. Weighting consists in putting onto each solution a non-negative real value based on its neighborhood in a way that the total weight is at least 1 for each satisfiable instance. We investigate the combinatorial properties of these two systems of estimation. First we give some sufficient conditions for a weighting system to be correct, and then we compare it to ordering under different conditions.

The material contained in this chapter was written in collaboration with Yacine Boufkhad and submitted for publication.

5.1 Overview of Results

Our first result consists in giving some sufficient conditions to make a weighting scheme correct for the estimation of satisfiability on general CSPs (theorem 5.9, *Weight Conservation*

Theorem). Then we propose a general weighting scheme obeying these conditions (theorem 5.15). This scheme is based on:

1. a *weighting seed* that expresses the relative importance of each value with respect to a variable and an instance; the seed is such that if all valuations were solutions, then their total weight would be exactly 1;
2. a *dispatching function* expressing how the weights of forbidden valuations are dispatched among solutions to insure that counting weighted solutions will yield at least 1 for any satisfiable instance.

We will refer to this method as *solution weighting* or for short *weighting*. Contrary to the weighting introduced by Maneva et al. which needs both solutions and partial valuations, our weighting addresses solutions only.

Till now the only way to compare ordering and weighting was to compute the estimations of satisfiability obtained by each of them on a certain set of instances and to choose the best one. We give some results comparing these two ways of estimating satisfiability in the following cases:

- weighting and ordering can be instance dependent when such syntactic properties as the number of occurrences of variables and values etc. can guide the design of weighting functions and orderings. We show that in the general case where the weighting function is instance dependent and when the weighting is *homogeneous* (i.e. when weighting and dispatching functions are equal), weighting cannot be better than a well chosen instance dependent ordering (theorem 5.25);
- in the case where ordering and weighting are instance independent (which is the case of problems where the values are indistinguishable like graph coloring for example) and in the case of sets of instances closed under value renaming (which is the case of almost all sets of instances considered in the literature), we show that weighting and ordering are equivalent (theorem 5.31).

Finally, in theorem 5.37, we show that the estimation of satisfiability used by Maneva & Sinclair - 2008 [MS08] can be improved upon by using a weighting scheme based on a 3-valued CSP and obeying the conditions of our *Weight Conservation Theorem* (which shows that these conditions are somehow relevant).

5.2 Framework

A CSP (Constraint Satisfaction Problem) is a triple $F = \langle X, D, C \rangle$ where X is a set of *variables* taking their values in the same finite *domain* D of values, and C is a set of *constraints*. A *constraint* is a couple $\langle x, R \rangle$ where $x \in X^k$ and $R \subseteq D^k$ for some integer k . R is interpreted as the tuples of allowed values. A *valuation* is a vector $v \in D^X$; access to coordinate $x \in X$ of v will be denoted as $v(x)$. It satisfies some constraint $\langle (x_1, x_2, \dots, x_k), R \rangle$ iff $(v(x_1), v(x_2), \dots, v(x_k)) \in R$. A valuation is said to be a *solution* of a CSP instance iff it satisfies all of its constraints.

We consider some sets \mathcal{F} of CSP instances sharing the same set X of variables and the same domain D . In the rest of the paper $n = |X|$ denotes the number of variables, $d = |D|$ the size of the domain. Given a CSP instance F , let $\mathcal{S}(F)$ denotes the set of its solutions.

We are interested in the neighborhood of valuations. Given a valuation v and $a \in D$, we define $v_{x \leftarrow a}$ as the valuation obtained from v by changing the value of x to a (including the case when already $a = v(x)$). Given a variable x , two solutions are called *x-adjacent* if they agree on all variables but x : in other words σ and τ are *x-adjacent* iff $\tau = \sigma_{x \leftarrow \tau(x)}$. Note that for each variable x , *x-adjacency* is an equivalence relation on solutions. Bringing together the *x-adjacency* relations with respect to every variable and removing the loops (σ, σ) we get an

non-oriented graph on $\mathcal{S}(F)$ that we call *solutions network*. Let $N_F(\sigma, x)$ denote the equivalence class of σ under x -adjacency (i.e. the neighborhood of σ for variable x); note that $N_F(\sigma, x)$ is a clique for x -adjacency. Such a clique will play a central role in our weighting system. We are also interested in the different values that x takes in this equivalence class, so we define $A_F(\sigma, x) = \{\tau(x)\}_{\tau \in N_F(\sigma, x)}$. For example in figure 5.1, solutions ab and aa are y -adjacent, $N_F(ab, y) = \{ab, aa\}$ and $A_F(ab, y) = \{b, a\}$.

Most of the results in this paper apply to any set of solutions regardless of which CSP instance has generated them. The sole solutions network can be thought of as the input of the problem. However it should be borne in mind that weightings and orderings cannot be defined using the global knowledge of the whole set of solutions, because of the locality condition: one can only count instances having a given solution and for each instance the solutions that are neighbors of this solution (rather than all solutions of a given instance). A convenient way to visualize this limitation is to imagine a network of processors (a processor representing a solution) where each processor has knowledge of its neighbors only and must compute from this knowledge its own weight or determine the orientation with respect to its neighbors.

5.3 Partial Ordering of Solutions

Given a CSP instance F , various partial orders \prec_F can be defined on the set of solutions such that for every two adjacent solutions σ and τ of F , we have either $\sigma \prec_F \tau$ or $\tau \prec_F \sigma$. The aim of the partial order here is to provide a measure on the solutions network through the number of its minimal elements. Let $\mathcal{M}_{\prec_F}(F)$ be the set of minimal solutions of F with respect to the order \prec_F .

In the solutions network of F , a partial order \prec_F can be seen as a circuit-free orientation of the edges of the graph such that an edge goes from τ to σ iff $\sigma \prec_F \tau$; then minimal elements are vertices with no outgoing edges. In general one seeks partial orderings that have the least number of minimal elements; however the choice is limited because orderings must be chosen according to local criteria only.

Construction of an Ordering.

Definition 5.1. Given a variable $x \in X$, a total strict order $<_{F,x}$ on D gives an orientation between neighboring solutions: $\sigma \prec_{F,x} \tau$ iff σ and τ are x -adjacent and $\sigma(x) <_{F,x} \tau(x)$. Note that $\prec_{F,x}$ is a partial strict order on the set of solutions, but a total strict order in each clique $N_F(\sigma, x)$.

We can bring all partial orders $\prec_{F,x}$ together on the set of solutions, as follows: if σ and τ are x -adjacent and different, then $\sigma \prec_F \tau$ iff $\sigma \prec_{F,x} \tau$. This is possible because two different solutions σ and τ cannot be both x -adjacent and y -adjacent for two different variables x and y . We say that \prec_F is the orientation on $\mathcal{S}(F)$ induced by the set $\{(x, <_{F,x})\}_{x \in X}$.

Lemma 5.2. If \prec_F is the orientation on $\mathcal{S}(F)$ induced by a set $\{(x, <_{F,x})\}_{x \in X}$, then \prec_F is circuit-free.

Proof. Suppose on the contrary that there exists a circuit $\sigma_1 \prec_F \cdots \prec_F \sigma_l \prec_F \sigma_1$ for some $l \geq 2$. Let us consider the variable x such that $\sigma_1 \prec_{F,x} \sigma_2$. For any $i \leq l$, either $\sigma_i(x) = \sigma_{i+1}(x)$ (if σ_i and σ_{i+1} are not x -adjacent) or $\sigma_i(x) <_{F,x} \sigma_{i+1}(x)$ (if σ_i and σ_{i+1} are x -adjacent). Thus $\sigma_1(x) <_{F,x} \sigma_2(x)$ and $\sigma_2(x) \leq_{F,x} \sigma_3(x) \leq_{F,x} \cdots \leq_{F,x} \sigma_l(x) \leq_{F,x} \sigma_1(x)$: a contradiction. \square

Corollary 5.3. The transitive closure of \prec_F is a strict order relation.

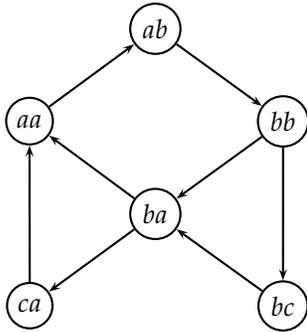


Figure 5.1: A bad orientation. There is no minimal element under the orientation.

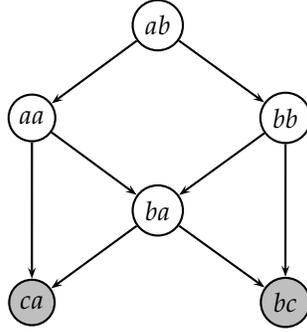


Figure 5.2: A good orientation. There are some minimal elements under the orientation (in gray).

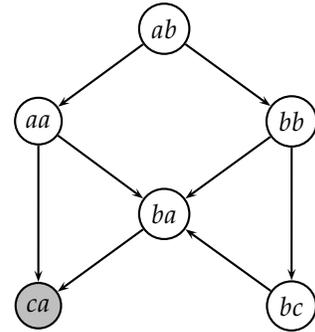


Figure 5.3: A very good orientation. There is only 1 minimal element under the orientation.

Instance Dependent or not.

- *Instance dependent ordering.* In this case, we put for each variable $x \in X$ and each CSP instance F a total order $<_{F,x}$ onto the domain D of possible values. As mentioned above, we (partially) order solutions as follows: let $\sigma \in S(F)$ and $\tau \in N_F(\sigma, x)$; we have $\sigma \prec_F \tau$ if and only if $\sigma(x) <_{F,x} \tau(x)$. The motivation for the instance dependent ordering is that some syntactic properties of the CSP instance F can be exploited to define a suitable order for that instance.
- *Instance independent ordering.* This is a particular case of the above ordering, when the total order $<_x$ on D does not depend on F . For some problems, no preferred order can be defined given some instance. This happens in particular when values are indistinguishable because of the symmetry of the problem (e.g. colors in graph coloring).

Examples of Orientations. In figures 5.1, 5.2 and 5.3, we consider a network of 6 solutions over the domain $D = \{a, b, c\}$ for a set of variables $X = \{x, y\}$; by shortcut ab we mean that variable x takes value a and variable y takes value b . For example $\{aa, ba, ca\}$ is a clique for variable x .

We first give an example (figure 5.1) of an orientation which is not circuit-free, even though it was built from the following local orderings on each individual clique:

- in cliques $\{a, b\}$ for variables x and y , we have $b < a$;
- in cliques $\{a, b, c\}$ for variables x and y , we have $a < c < b$.

The problem comes from the fact that a and b are ordered differently in clique $\{a, b, c\}$ and its sub-clique $\{a, b\}$, which led us to consider only orientations built in the following way: we choose for each variable x a total order $<_x$ on the domain D and use it for each sub-clique of D . This is what we shall call a *uniform* orientation in chapter 6. Example in figure 5.2 was obtained by the following orders: $c <_x b <_x a$ and $c <_y a <_y b$. This orientation is circuit-free and has two minimal elements. Now among good orientations, the less minimal elements they have, the better they are; figure 5.3, which was obtained by the following orders: $c <_x b <_x a$ and $a <_y c <_y b$, gives an example of an orientation with just one minimal element.

5.4 Weighting of Solutions

First we define a weighting system for all valuations (solutions or not) which sums up to 1. Then we give sufficient conditions on a weighting system on solutions only, such that a transfer between this weighting system and the previous one may be possible. Doing this we establish a general framework for putting weights onto solutions, and use it to derive two particular weighting systems: the first one addresses general CSPs and the second one is built to improve on the weighting system introduced by Maneva, Mossel & Wainwright - 2007 [MMW07] and further explored by Maneva & Sinclair - 2008 [MS08] and Ardila & Maneva - 2009 [AM09]. The purpose of such a transfer is to estimate the global weight in the weighting system on solutions by means of the global weight of the weighting system on all valuations (which is easier to compute).

5.4.1 Weighting Seeds

Definition 5.4. For a CSP $F = \langle X, D, C \rangle$ a *weighting seed* is a function $s_F : X \times D \rightarrow \mathbf{R}^+$. We say that s_F is *unitary* iff $\forall x \in X, \sum_{a \in D} s_F(x, a) = 1$.

Now we define the *unladen weight* of any valuation v (solution or not) with respect to some weighting seed s_F as :

$$U_F(v) = \prod_{x \in X} s_F(x, v(x)) . \quad (5.1)$$

As for the *actual weight* of a solution, we want to take into account the neighborhood of the solution, so we put the weight $w_F(\sigma, x)$ on each variable x of solution σ . We will see later how to build w_F from s_F .

The actual weight of a solution is:

$$W_F(\sigma) = \prod_{x \in X} w_F(\sigma, x) . \quad (5.2)$$

By extension, the weight of a set S of solutions is:

$$W_F(S) = \sum_{\sigma \in S} W_F(\sigma) . \quad (5.3)$$

Lemma 5.5. *If the weighting seed s_F is unitary, then the total unladen weight of all valuations is 1: $\sum_{v \in D^X} U_F(v) = 1$.*

Proof.

$$\begin{aligned} \sum_{v \in D^X} U_F(v) &= \sum_{v \in D^X} \prod_{x \in X} s_F(x, v(x)) \\ &= \prod_{x \in X} \sum_{a \in D} s_F(x, a) \\ &= \prod_{x \in X} 1 \\ &= 1 . \end{aligned}$$

□

This weight U_F is indeed simple to handle. The purpose is now to connect it with W_F . Just as we defined weights W_F of solutions in a product form variable per variable, so shall we build our transfer system.

5.4.2 Decomposers

Definition 5.6. We say that w_F is *decomposable* by a family $(\delta_{F,\sigma,x,a})$ iff for all solution σ of F and all variable x , $w_F(\sigma, x) = \sum_{a \in D} \delta_{F,\sigma,x,a}$. Such a family will be referred to as a *decomposer*. We define onto it the following transfer quantities between a solution σ and a valuation v :

$$T_{F,\sigma \rightarrow v} = \prod_{x \in X} \delta_{F,\sigma,x,v(x)} . \quad (5.4)$$

Lemma 5.7. (*Transfer lemma*). Let F be a CSP instance and σ any of its solutions. If w_F is decomposable by family $(\delta_{F,\sigma,x,a})$, then

$$W_F(\sigma) = \sum_{v \in D^X} T_{F,\sigma \rightarrow v} . \quad (5.5)$$

Proof. It is sufficient to expand the weight of a solution as follows:

$$\begin{aligned} W_F(\sigma) &= \prod_{x \in X} w_F(\sigma, x) \\ &= \prod_{x \in X} \sum_{a \in D} \delta_{F,\sigma,x,a} \\ &= \sum_{v \in D^X} \prod_{x \in X} \delta_{F,\sigma,x,v(x)} \\ &= \sum_{v \in D^X} T_{F,\sigma \rightarrow v} . \end{aligned}$$

□

We want to insure that transfers made towards a valuation are at least its unladen weight, hence we define the following property of covering.

Definition 5.8. Let S be a subset of $\mathcal{S}(F)$; we say that (T_F, S) *covers* U_F iff $\forall v \in D^X, \sum_{\sigma \in S} T_{F,\sigma \rightarrow v} \geq U_F(v)$.

We can now state some general conditions that are sufficient for a weighting scheme to be correct.

5.4.3 Weight Conservation Theorem

Theorem 5.9. (*Weight Conservation Theorem*). If the following assumptions hold:

1. the weighting seed s_F is unitary,
2. the actual weight w_F is decomposable by family $(\delta_{F,\sigma,x,a})$,
3. (T_F, S) covers U_F ,

then $W_F(S) \geq 1$.

Proof. Since w_F is decomposable by family $(\delta_{F,\sigma,x,a})$, lemma 5.7 asserts that $\forall \sigma \in S, W_F(\sigma) = \sum_{v \in D^X} T_{F,\sigma \rightarrow v}$. Thus

$$\begin{aligned} W_F(S) &= \sum_{\sigma \in S} W_F(\sigma) \\ &= \sum_{\sigma \in S} \sum_{v \in D^X} T_{F,\sigma \rightarrow v} \text{ by lemma 5.7} \\ &= \sum_{v \in D^X} \sum_{\sigma \in S} T_{F,\sigma \rightarrow v} \\ &\geq \sum_{v \in D^X} U_F(v) \text{ since } (T_F, S) \text{ covers } U_F . \end{aligned}$$

Moreover by lemma 5.5, since s_F is unitary, $\sum_{v \in D^X} U_F(v) = 1$. \square

Thus we have exhibited three sufficient conditions to get a weight conservation theorem. These conditions might not be necessary; however not any weighting system w_F will be correct, as shown in example on figure 5.4. So let us introduce a way to build w_F from s_F in a way that is intended to match the conditions of our Weight Conservation Theorem.

5.4.4 Generators

All weights we put onto solutions (either in section 5.4.5 or in section 5.7) are built from a *weight generator*, as follows.

Definition 5.10. A *generator* is a function $\omega_F : X \times D \times \mathcal{P}(D) \rightarrow \mathbf{R}^+$. We say that ω_F is *unitary* iff for all variable x and all nonempty subset Δ of D , $\sum_{a \in \Delta} \omega_F(x, a, \Delta) = 1$.

From the weight generator ω_F we now define the *actual weight* w_F of a variable in a solution:

$$w_F(\sigma, x) = \omega_F(x, \sigma(x), A_F(\sigma, x)) . \quad (5.6)$$

Remark. If σ and τ are 2 solutions such that $\sigma(x) = \tau(x)$ and $A_F(\sigma, x) = A_F(\tau, x)$, then $w_F(\sigma, x) = w_F(\tau, x)$. This is what we shall call a *uniform weighting* in chapter 6.

This may suggest that it could be sufficient to put any weights such that the sum of weights on any clique would be 1; but it is not the case (cf. example on figure 5.4).

5.4.5 Dispatchers

Definition 5.11. A *dispatcher* is a function $d_F : X \times D \rightarrow \mathbf{R}_*^+$.

Using the weighting seed s_F and the dispatcher d_F we now build the weight generator ω_F of variables in a solution. Each variable will keep its seed s_F ; moreover the weights of forbidden values will be dispatched to allowed values thanks to d_F , in the following way:

$$\omega_F(x, a, \Delta) = \begin{cases} s_F(x, a) + \frac{d_F(x, a)}{\sum_{b \in \Delta} d_F(x, b)} \sum_{b \in D \setminus \Delta} s_F(x, b) & \text{if } a \in \Delta; \\ 0 & \text{otherwise .} \end{cases} \quad (5.7)$$

Δ represents a category of set of allowed values; so the dispatcher d_F dispatches the total weighting seed of forbidden values among allowed values.

Fact. If s_F is unitary, so is ω_F .

Definition 5.12. We say that the weighting system is *homogeneous* when $d_F = s_F$. In this noticeable case the same function is used to assign a weighting seed and to dispatch remaining weights among neighbors.

Examples of Weightings. As one can see in figure 5.4, even if we put a total weight of 1 on each clique, the overall weight can be less than 1. To prevent such bad configurations we let our weights take the form of seeds+dispatchers (figures 5.5 and 5.6). The purpose of building weights from seeds and dispatchers is to prevent the same kind of inversions that we encountered for orientations (which led to circuits): in figure 5.4, in clique $\{a, b\}$ for variable x , a is given a much smaller weight than b , whereas in clique $\{a, b, c\}$ for the same variable x , the opposite occurs. In fact dispatchers allow some reshuffling of weights between different cliques of the same variable (see the weights given to values a and b in cliques $\{a, b\}$ and $\{a, b, c\}$ for variable x on figure 5.6), but the fact that seeds and dispatchers are assigned to each individual couple (variable, value) enables a kind of consistency between a clique and its sub-cliques, preventing circuit-like structures from appearing.

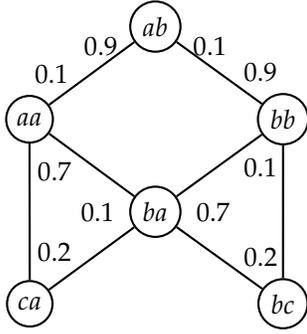


Figure 5.4: A bad weighting system. The total weight is $2 * (0.1 * 0.9 + 0.1 * 0.7 + 0.2) = 0.72$.

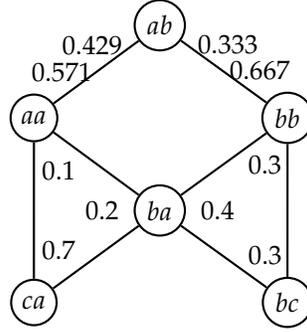


Figure 5.5: An homogeneous weighting system. The total weight is $0.7 + 0.3 + 0.2 * 0.4 + 0.1 * 0.571 + 0.3 * 0.667 + 0.333 * 0.429 \simeq 1.48$.

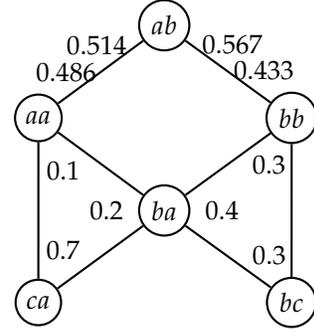


Figure 5.6: An heterogeneous weighting system. The total weight is $0.7 + 0.3 + 0.2 * 0.4 + 0.1 * 0.486 + 0.3 * 0.433 + 0.567 * 0.514 \simeq 1.55$.

- figure 5.5 was obtained by the following choice of s_F and d_F (homogeneous case, so $d_F = s_F$):

s_F	a	b	c
x	0.1	0.2	0.7
y	0.4	0.3	0.3

d_F	a	b	c
x	0.1	0.2	0.7
y	0.4	0.3	0.3

- figure 5.6 was obtained by the following choice of s_F and d_F :

s_F	a	b	c
x	0.1	0.2	0.7
y	0.4	0.3	0.3

d_F	a	b	c
x	0.6	0.3	0.1
y	0.2	0.5	0.3

We come back to our weighting system w_F built from s_F and d_F and show that it may be used to estimate satisfiability if s_F is unitary. So our first result concerning this weighting system states that this system is correct for the estimation of satisfiability (theorem 5.15 below). To prove it, we use our Weight Conservation Theorem, using the following decomposers:

$$\delta_{F,\sigma,x,a} = \begin{cases} s_F(x,a) & \text{if } \sigma(x) = a; \\ \frac{d_F(x,\sigma(x))}{\sum_{a \in A_F(\sigma,x)} d_F(x,a)} s_F(x,a) & \text{if } a \notin A_F(\sigma,x); \\ 0 & \text{otherwise .} \end{cases} \quad (5.8)$$

We must now prove that the conditions of our Weight Conservation Theorem are satisfied: w_F is decomposable family $(\delta_{F,\sigma,x,a})$ and (T_F, g) covers U_F .

Lemma 5.13. w_F is decomposable by family $(\delta_{F,\sigma,x,a})$.

Proof. By definitions:

$$\begin{aligned}
\sum_{a \in D} \delta_{F, \sigma, x, a} &= \sum_{a \in D} \left(s_F(x, a) \mathbf{1}_{a=\sigma(x)} + \frac{d_F(x, \sigma(x))}{\sum_{a \in A_F(\sigma, x)} d_F(x, a)} s_F(x, a) \mathbf{1}_{a \notin A_F(\sigma, x)} \right) \\
&= s_F(x, \sigma(x)) + \frac{d_F(x, \sigma(x))}{\sum_{a \in A_F(\sigma, x)} d_F(x, a)} \sum_{a \notin A_F(\sigma, x)} s_F(x, a) \\
&= w_F(\sigma, x) .
\end{aligned}$$

□

As the unladen weight of a valuation is scattered among lots of solutions, in the proof of the following lemma we use an algorithm building a tree in order to catch enough solutions to insure the covering condition. The proof is somewhat technical and may be skipped at first reading.

Lemma 5.14. *Let g be any connected component of the solutions network $\mathcal{S}(F)$. Then (T_F, g) covers U_F .*

Proof. First we need some definitions. A *partial valuation* η over $Y \subseteq X$ is a function from Y to the set D . The *domain* of η is $\text{Dom}(\eta) = Y$. The *level* (of undetermination) of η is $\text{Level}(\eta) = |X \setminus Y|$. Let $Z \subseteq Y \subseteq X$, let ι be a partial valuation over Z and η be a partial valuation over Y . Since $Z \subseteq Y$, we say that $\iota \leq_{\text{Dom}} \eta$. Of course \leq_{Dom} is a partial order relation. We say that η is an *extension* of ι iff $\forall z \in Z, \eta(z) = \iota(z)$, in which case we also say that ι is the *restriction* of η to Z : $\iota = \eta|_Z$. In the particular case when $Y = Z \cup \{x\}$ with $x \notin Z$, we denote by $\iota_{x \mapsto a}$, the extension of ι to Y assigning value a to x . Let g be a connected component of the solutions network. Note that the empty valuation ϵ (with domain \emptyset) is extensible to a solution in g as soon as $g \neq \emptyset$. Given a partial valuation η , we call $E_g(\eta)$ the set of its extensions which are elements of g and $r_g(\eta)$ the set of restrictions of η extensible to a solution in g (i.e. restrictions r of η such that $E_g(r) \neq \emptyset$).

Let us take any valuation v . We must prove that $\sum_{\sigma \in g} T_{F, \sigma \rightarrow v} \geq U_F(v)$. Since $g \neq \emptyset$, $\epsilon \in r_g(v)$ so $r_g(v) \neq \emptyset$ and we can pick an element v_0 in $r_g(v)$ maximal with respect to the order \leq_{Dom} . We arbitrarily put indices $1 \dots n_0$ onto the remaining $n_0 = \text{Level}(v_0)$ variables: x_1, \dots, x_{n_0} (i.e. variables not set by v_0). In the following algorithm we shall bind a *factitious weight* $f(\eta)$ and a solution $\tau(\eta)$ to a partial valuation η . At the beginning $f(v_0) = U_F(v)$, and we make a call of $\text{Extend}(v_0)$.

Algorithm 5.1 Extensions of a partial valuation.

```

1: procedure Extend( $\eta$ )
2:    $i \leftarrow \text{Level}(\eta)$ 
3:   if  $i = 0$  then
4:      $S \leftarrow S \cup \{\eta\}$ 
5:   else
6:      $\tau(\eta) \leftarrow$  a solution maximizing  $\sum_{b \in A_F(\sigma, x_i)} d_F(x_i, b)$  among  $\sigma \in E_g(\eta)$ 
7:     for all  $a \in A_F(\tau(\eta), x_i)$  do
8:        $f(\eta_{x_i \mapsto a}) \leftarrow \frac{d_F(x_i, a)}{\sum_{b \in A_F(\tau(\eta), x)} d_F(x_i, b)} f(\eta)$ 
9:     Extend ( $\eta_{x_i \mapsto a}$ )

```

Informally we are building a tree and propagating weights from the root v_0 (at level n_0) to leaves which are solutions (at level 0) in a conservative way: the total factitious weight on level i will be the same as that of level $i + 1$.

Formally, what can we insure along this process?

1. The first thing to notice is that the algorithm stops; namely the nested calls of $\text{Extend}(\eta)$ decrement $\text{Level}(\eta)$ till it reaches 0.
2. Secondly S is indeed a set of solutions in g extending v_0 . Namely at each call of $\text{Extend}(\eta)$, η is extensible to a solution in g and the set of unset variables of η is $\{x_1, \dots, x_i\}$, where $i = \text{Level}(\eta)$. Thus when $i = 0$, η is a solution in g . We prove this by induction:
 - a) at the beginning: $v_0 \in r_g(v)$, v_0 trivially extends itself, $E_g(v_0) \neq \emptyset$ and the set of unset variables of v_0 is $\{x_1, \dots, x_{n_0}\}$;
 - b) now suppose that $E_g(\eta) \neq \emptyset$, η extends v_0 and the unset variables of η are $\{x_1, \dots, x_i\}$; given $\tau(\eta) \in E_g(\eta)$, let $a \in A_F(\tau(\eta), x_i)$; then the valuation $\tau(\eta)_{x_i \leftarrow a}$ is a solution by definition of $A_F(\tau(\eta), x_i)$; moreover it is connected to $\tau(\eta)$, thus $\tau(\eta)_{x_i \leftarrow a}$ is an element of component g . Moreover since $\tau(\eta)$ is an extension of η and x_i is unset in η , $\tau(\eta)_{x_i \leftarrow a}$ is an extension of $\eta_{x_i \rightarrow a}$. Thus $\tau(\eta)_{x_i \leftarrow a} \in E_g(\eta_{x_i \rightarrow a})$, so $E_g(\eta_{x_i \rightarrow a}) \neq \emptyset$. Of course, $\eta_{x_i \rightarrow a}$ extends v_0 , the unset variables of $\eta_{x_i \rightarrow a}$ are $\{x_1, \dots, x_{i-1}\}$ and $\text{Level}(\eta_{x_i \rightarrow a}) = \text{Level}(\eta) - 1 = i - 1$.
3. $\sum_{\sigma \in S} f(\sigma) = U_F(v)$; namely among partial valuations considered in the process, $\eta \in S$ iff $\text{Level}(\eta) = 0$. Moreover we now prove by induction that $\sum_{\text{Level}(\eta)=i} f(\eta) = U_F(v)$:
 - a) at the beginning when $i = n_0$, the only partial valuation of level n_0 is v_0 and $f(v_0) = U_F(v)$;
 - b) now suppose that $\sum_{\text{Level}(\eta)=i} f(\eta) = U_F(v)$; in our process each partial valuation η of level $i - 1$ has one and only one parent in level i , which is given by the restriction η' of η to $\text{Dom}(v_0) \cup \{x_{i+1}, \dots, x_{n_0}\}$; thus

$$\begin{aligned}
 \sum_{\text{Level}(\eta)=i-1} f(\eta) &= \sum_{\text{Level}(\eta')=i} \sum_{a \in A_F(\tau(\eta'), x_i)} f(\eta'_{x_i \rightarrow a}) \\
 &= \sum_{\text{Level}(\eta')=i} \sum_{a \in A_F(\tau(\eta'), x_i)} \frac{d_F(x_i, a)}{\sum_{b \in A_F(\tau(\eta'), x_i)} d_F(x_i, b)} f(\eta') \\
 &= \sum_{\text{Level}(\eta')=i} f(\eta') \\
 &= U_F(v) .
 \end{aligned}$$

4. $\forall \sigma \in S, \forall i \in \{1, \dots, n_0\}, v(x_i) \notin A_F(\sigma, x_i)$. Suppose on the contrary that $\exists \sigma \in S, \exists i \in \{1, \dots, n_0\}, v(x_i) \in A_F(\sigma, x_i)$; the partial valuation $v_0_{x_i \rightarrow v(x_i)}$ is still a restriction of v ; moreover, since by item 2, σ is an extension of v_0 , $\sigma_{x_i \leftarrow v(x_i)}$ is an extension of $v_0_{x_i \rightarrow v(x_i)}$; and since $v(x_i) \in A_F(\sigma, x_i)$, $\sigma_{x_i \leftarrow v(x_i)}$ is a solution. Thus $v_0_{x_i \rightarrow v(x_i)} \in r_g(v)$ and $v_0_{x_i \rightarrow v(x_i)} \succ_{\text{Dom}} v_0$, contradicting the maximality of v_0 in $r_g(v)$.

5. $\forall \sigma \in S, f(\sigma) \leq T_{F, \sigma \rightarrow v}$; namely, let us take any $\sigma \in S$:

$$\begin{aligned}
T_{F, \sigma \rightarrow v} &= \prod_{x \in X} \delta_{F, \sigma, x, v(x)} \text{ by definition 5.6} \\
&= \prod_{x \in X} s_F(x, v(x)) \left(\mathbf{1}_{v(x)=\sigma(x)} + \frac{d_F(x, \sigma(x))}{\sum_{a \in A_F(\sigma, x)} d_F(x, a)} \mathbf{1}_{v(x) \notin A_F(\sigma, x)} \right) \text{ by eq. 5.8} \\
&= \prod_{x \in \text{Dom}(v_0)} s_F(x, v(x)) \prod_{i=1}^{n_0} \frac{d_F(x_i, \sigma(x_i)) s_F(x_i, v(x_i))}{\sum_{a \in A_F(\sigma, x_i)} d_F(x_i, a)} \text{ by item 2 and 4} \\
&= U_F(v) \prod_{i=1}^{n_0} \frac{d_F(x_i, \sigma(x_i))}{\sum_{a \in A_F(\sigma, x_i)} d_F(x_i, a)} \text{ by definition 5.4 .}
\end{aligned}$$

Moreover note that

$$\begin{aligned}
f(\sigma) &= f(\sigma|_{\text{Dom}(v_0)}) \prod_{i=1}^{n_0} \frac{f(\sigma|_{\text{Dom}(v_0) \cup \{x_i, \dots, x_{n_0}\}})}{f(\sigma|_{\text{Dom}(v_0) \cup \{x_{i+1}, \dots, x_{n_0}\}})} \\
&= f(v_0) \prod_{i=1}^{n_0} \frac{f(\sigma|_{\text{Dom}(v_0) \cup \{x_{i+1}, \dots, x_{n_0}\}}, x_i \rightarrow \sigma(x_i))}{f(\sigma|_{\text{Dom}(v_0) \cup \{x_{i+1}, \dots, x_{n_0}\}})} \\
&= U_F(v) \prod_{i=1}^{n_0} \frac{d_F(x_i, \sigma(x_i))}{\sum_{a \in A_F(\tau(\sigma|_{\text{Dom}(v_0) \cup \{x_{i+1}, \dots, x_{n_0}\}}, x_i))} d_F(x_i, a)} .
\end{aligned}$$

Since of course, for all i between 1 and n_0 , $\sigma \in E_g(\sigma|_{\text{Dom}(v_0) \cup \{x_{i+1}, \dots, x_{n_0}\}})$, by choice of $\tau(\eta)$ in line 6 of algorithm 5.1, we have that $\sum_{a \in A_F(\tau(\sigma|_{\text{Dom}(v_0) \cup \{x_{i+1}, \dots, x_{n_0}\}}, x_i))} d_F(x_i, a) \geq$

$$\sum_{a \in A_F(\sigma, x_i)} d_F(x_i, a), \text{ whence } f(\sigma) \leq U_F(v) \prod_{i=1}^{n_0} \frac{d_F(x_i, \sigma(x_i))}{\sum_{a \in A_F(\sigma, x_i)} d_F(x_i, a)} = T_{F, \sigma \rightarrow v}.$$

Thus we finally get that

$$\begin{aligned}
\sum_{\sigma \in g} T_{F, \sigma \rightarrow v} &\geq \sum_{\sigma \in g} f(\sigma) \text{ by item 5} \\
&\geq \sum_{\sigma \in S} f(\sigma) \text{ because } S \subseteq g .
\end{aligned}$$

Moreover, by item 3, $\sum_{\sigma \in S} f(\sigma) = U_F(v)$; thus (T_F, g) covers U_F . \square

From lemmas 5.13 and 5.14, we conclude that our weighting system built from seeds and dispatchers obeys the conditions of the Weight Conservation Theorem.

Theorem 5.15. *Let F be a satisfiable CSP instance and g the solutions in a connected component of the solutions network of F . Weights w_F are built from seeds s_F and dispatchers d_F , as in definition 5.11. If the weighting seed s_F is unitary, then $W_F(g) \geq 1$.*

Remark. In this paper we do not address the question of choosing the best s_F and d_F for a given instance F or for a given family of instances, which must be custom-tailored depending on the considered problem.

5.5 Homogeneous Case: Weighting Is not Better than Ordering

As we have seen, the weighting is based upon two functions:

1. the weighting seed s_F that determines the intrinsic weight of each value and then allows to compute the intrinsic unladen weight of each valuation;
2. the dispatcher d_F that represents how the weights of forbidden valuations are scattered among the authorized ones.

A natural case to investigate is when these two quantities are equal, namely when each allowed value is dispatched a complimentary weight proportional to its intrinsic weight. So we deal here with the homogeneous case $d_F = s_F$ and show that whatever s_F may be, there will exist an ordering which is at least as good as the weighting system, as will be stated in theorem 5.25. The proof consists in choosing variable per variable the order $\prec_{F,x}$ in a way that does not increase the global weight. For our recurrence to work we use the homogeneity property. Just as we defined a generator ω_F for a weight w_F , so need we now to define a generator μ_F for an orientation m_F .

Definition 5.16. We define the following binary weight function:

$$\mu_F(x, a, \Delta) = \begin{cases} 1 & \text{if } a \text{ is the minimum of } \Delta \text{ for } \prec_{F,x}; \\ 0 & \text{otherwise.} \end{cases} \quad (5.9)$$

$$m_F(\sigma, x) = \mu_F(x, \sigma(x), A_F(\sigma, x)) . \quad (5.10)$$

At each step of the recurrence, some variables are ordered while the other ones are weighted. That leads us to introduce the following definitions. We are going to substitute binary weights m_F 's to original weights w_F 's variable per variable, so we call Ξ the set of couples of (variables x , orders $\prec_{F,x}$) where m_F 's are used and we define

Definition 5.17.

$$\Omega_F(\sigma, \Xi) = \prod_{x \in \Xi} m_F(\sigma, x) \prod_{x \in X \setminus \Xi} w_F(\sigma, x) \quad (5.11)$$

and we extend it to a set S of solutions by

$$\Omega_F(S, \Xi) = \sum_{\sigma \in S} \Omega_F(\sigma, \Xi) . \quad (5.12)$$

Remark 5.18. What happens when Ξ is empty?

$$\Omega_F(\mathcal{S}(F), \emptyset) = W_F(\mathcal{S}(F)) . \quad (5.13)$$

Namely, by definition, for any solution $\sigma \in \mathcal{S}(F)$, $\Omega_F(\sigma, \emptyset) = \prod_{x \in X} w_F(\sigma, x) = W_F(\sigma)$. But $\Omega_F(\mathcal{S}(F), \emptyset) = \sum_{\sigma \in \mathcal{S}(F)} \Omega_F(\sigma, \emptyset)$ and $W_F(\mathcal{S}(F)) = \sum_{\sigma \in \mathcal{S}(F)} W_F(\sigma)$.

Remark 5.19. What happens when Ξ is full? Suppose that for all variable x , $\prec_{F,x}$ is a total order on D . Let \prec_F be the orientation induced by $\{(x, \prec_{F,x})\}_{x \in X}$. Then

$$\Omega_F(\mathcal{S}(F), \{(x, \prec_{F,x})\}_{x \in X}) = |\mathcal{M}_{\prec_F}(F)| . \quad (5.14)$$

Namely, let us recall that for any solution $\sigma \in \mathcal{S}(F)$, $\Omega_F(\sigma, \{(x, \prec_{F,x})\}_{x \in X}) = \prod_{x \in X} m_F(\sigma, x)$. Thus $\Omega_F(\sigma, \{(x, \prec_{F,x})\}_{x \in X}) = 1$ iff $\forall x \in X$, σ is the minimum of $N_F(\sigma, x)$ for $\prec_{F,x}$ (or equivalently for \prec_F); in other words σ is minimal among all of its neighbors, which means that σ is minimal (since \prec_F compares neighboring solutions only). Thus $\Omega_F(\{(x, \prec_{F,x})\}_{x \in X})$ is the number of minimal elements of the underlying orientation \prec_F .

We are now ready to state the main lemma in this section.

Lemma 5.20. *Suppose that s_F is unitary and $d_F = s_F$. Then for each set Ξ , each variable $x_0 \notin \Xi$, there exists a total order $<_{F,x_0}$ on D such that $\Omega_F(\mathcal{S}(F), \Xi \cup \{(x_0, <_{F,x_0})\}) \leq \Omega_F(\mathcal{S}(F), \Xi)$.*

At first reading it might be convenient to jump directly to theorem 5.25, because the proof of lemma 5.20 is somewhat technical and requires some more notations and sub-lemmas. We fix a variable $x_0 \notin \Xi$. Let a be an element of D and Δ be a subset of D . We consider the preimages of (a, Δ) obtained through mapping a solution σ of instance F to $(\sigma(x_0), A_F(\sigma, x_0))$. We denote these preimages as follows:

$$\Sigma_{F,x_0}(a, \Delta) = \{\sigma \in \mathcal{S}(F), \sigma(x_0) = a \text{ and } A_F(\sigma, x_0) = \Delta\} . \quad (5.15)$$

Note that:

1. when $a \notin \Delta$, $\Sigma_{F,x_0}(a, \Delta) = \emptyset$;
2. the $\Sigma_{F,x_0}(a, \Delta)$ are pairwise disjoint and $\bigsqcup_{a \in \Delta} \Sigma_{F,x_0}(a, \Delta) = \mathcal{S}(F)$;
3. if $\sigma, \tau \in \Sigma_{F,x_0}(a, \Delta)$, then $w_F(\sigma, x_0) = w_F(\tau, x_0) = \omega_F(x_0, a, \Delta)$
and $m_F(\sigma, x_0) = m_F(\tau, x_0) = \mu_F(x_0, a, \Delta)$;
4. we call

$$Z_{F,\Xi,x_0}(a, \Delta) = \sum_{\sigma \in \Sigma_{F,x_0}(a, \Delta)} \prod_{x \in \Xi} m_F(\sigma, x) \prod_{x \in X \setminus (\Xi \cup \{x_0\})} w_F(\sigma, x) ; \quad (5.16)$$

then by item 3,

$$\Omega_F(\Sigma_{F,x_0}(a, \Delta), \Xi) = \omega_F(x_0, a, \Delta) \cdot Z_{F,\Xi,x_0}(a, \Delta) ; \quad (5.17)$$

$$\Omega_F(\Sigma_{F,x_0}(a, \Delta), \Xi \cup \{(x_0, <_{F,x_0})\}) = \mu_F(x_0, a, \Delta) \cdot Z_{F,\Xi,x_0}(a, \Delta) . \quad (5.18)$$

We now need to explore further both quantities we want to compare. It will be convenient to use the following quantities: let $E \subseteq D$ and $a \in E$; we define the following quantities:

$$\zeta_{F,\Xi,x_0}(a, E) = \sum_{\substack{\Delta \subseteq E \\ \Delta \ni a}} Z_{F,\Xi,x_0}(a, \Delta) ; \quad (5.19)$$

$$\check{\zeta}_{F,\Xi,x_0}(E) = \sum_{\Delta \subseteq E} \sum_{a \in \Delta} \omega_F(x_0, a, \Delta) \cdot Z_{F,\Xi,x_0}(a, \Delta) . \quad (5.20)$$

So what is the purpose of introducing these extra quantities? They will help us prove lemma 5.20 through the following facts.

Fact 5.21. *If $a_1 <_{F,x_0} a_2 <_{F,x_0} \dots <_{F,x_0} a_d$, then*

$$\Omega_F(\mathcal{S}(F), \Xi \cup \{(x_0, <_{F,x_0})\}) = \sum_{i=1}^d \zeta_{F,\Xi,x_0}(a_i, D \setminus \{a_1, \dots, a_{i-1}\}) .$$

Proof. We use the partition mentioned in item 2:

$$\begin{aligned}
\Omega_F(\mathcal{S}(F), \Xi \cup \{(x_0, <_{F,x_0})\}) &= \Omega_F\left(\bigsqcup_{\substack{\Delta \subseteq D \\ a \in \Delta}} \Sigma_{F,x_0}(a, \Delta), \Xi \cup \{(x_0, <_{F,x_0})\}\right) \\
&= \sum_{\substack{\Delta \subseteq D \\ a \in \Delta}} \Omega_F(\Sigma_{F,x_0}(a, \Delta), \Xi \cup \{(x_0, <_{F,x_0})\}) \\
&= \sum_{\substack{\Delta \subseteq D \\ a \in \Delta}} \mu_F(x_0, a, \Delta) \cdot Z_{F,\Xi,x_0}(a, \Delta) \text{ by eq. 5.18} \\
&= \sum_{a \in D} \sum_{\substack{\Delta \subseteq D \\ \Delta \ni a}} \mu_F(x_0, a, \Delta) \cdot Z_{F,\Xi,x_0}(a, \Delta) \\
&= \sum_{a \in D} \sum_{\substack{\Delta \subseteq D \\ \Delta \ni a}} \mathbf{1}_{a \text{ is the minimum of } \Delta \text{ for } <_{F,x_0}} \cdot Z_{F,\Xi,x_0}(a, \Delta) \\
&= \sum_{i=1}^d \sum_{\substack{\Delta \subseteq D \setminus \{a_1, \dots, a_{i-1}\} \\ \Delta \ni a_i}} Z_{F,\Xi,x_0}(a_i, \Delta) \text{ since } a_1 <_{F,x_0} \dots <_{F,x_0} a_d \\
&= \sum_{i=1}^d \zeta_{F,\Xi,x_0}(a_i, D \setminus \{a_1, \dots, a_{i-1}\}) .
\end{aligned}$$

□

Fact 5.22. For all $x_0 \notin \Xi$, $\Omega_F(\mathcal{S}(F), \Xi) = \zeta_{F,\Xi,x_0}(D)$.

Proof. We use again the partition mentioned in item 2:

$$\begin{aligned}
\Omega_F(\mathcal{S}(F), \Xi) &= \Omega_F\left(\bigsqcup_{\substack{\Delta \subseteq D \\ a \in \Delta}} \Sigma_{F,x_0}(a, \Delta), \Xi\right) \\
&= \sum_{\substack{\Delta \subseteq D \\ a \in \Delta}} \Omega_F(\Sigma_{F,x_0}(a, \Delta), \Xi) \\
&= \sum_{\substack{\Delta \subseteq D \\ a \in \Delta}} \omega_F(x_0, a, \Delta) \cdot Z_{F,\Xi,x_0}(a, \Delta) \text{ by eq. 5.17} \\
&= \zeta_{F,\Xi,x_0}(D) .
\end{aligned}$$

□

Fact 5.23. If $E \subseteq D$, $\Delta \subseteq E$, $a \in \Delta$, s_F is unitary and $d_F = s_F$ then

$$\omega_F(x, a, \Delta) \sum_{b \in E} d_F(x, b) = d_F(x, a) + \omega_F(x, a, \Delta) \sum_{b \in E \setminus \Delta} d_F(x, b) .$$

Proof. If $a \in \Delta$, then by equation 5.7, $\omega_F(x, a, \Delta) \sum_{b \in \Delta} d_F(x, b) = s_F(x, a) \sum_{b \in \Delta} d_F(x, b) + d_F(x, a) \sum_{b \in D \setminus \Delta} s_F(x, b)$. By equality $d_F = s_F$ and the fact that s_F is unitary, we get $d_F(x, a) = \omega_F(x, a, \Delta) \sum_{b \in \Delta} d_F(x, b)$. □

Fact 5.24. Let $x_0 \notin \Xi$ and E any nonempty subset of D . Suppose that s_F is unitary and $d_F = s_F$. Then there exists $a \in E$ such that $\zeta_{F,\Xi,x_0}(E) \geq \zeta_{F,\Xi,x_0}(a, E) + \zeta_{F,\Xi,x_0}(E \setminus \{a\})$.

Proof. Let us call a_0 an element of E minimizing $\zeta_{F,\Xi,x_0}(a, E) + \zeta_{F,\Xi,x_0}(E \setminus \{a\})$ when $a \in E$:

$$\begin{aligned}
\zeta_{F,\Xi,x_0}(E) \sum_{b \in E} d_F(x_0, b) &= \sum_{b \in E} d_F(x_0, b) \sum_{\substack{\Delta \subseteq E \\ a \in \Delta}} \omega_F(x_0, a, \Delta) Z_{F,\Xi,x_0}(a, \Delta) \\
&= \sum_{\substack{\Delta \subseteq E \\ a \in \Delta}} d_F(x_0, a) Z_{F,\Xi,x_0}(a, \Delta) \\
&\quad + \sum_{\substack{\Delta \subseteq E \\ a \in \Delta}} \omega_F(x_0, a, \Delta) \sum_{b \in E \setminus \Delta} d_F(x_0, b) Z_{F,\Xi,x_0}(a, \Delta) \text{ by fact 5.23} \\
&= \sum_{\substack{\Delta \subseteq E \\ a \in \Delta}} d_F(x_0, a) Z_{F,\Xi,x_0}(a, \Delta) \\
&\quad + \sum_{\substack{\Delta \subseteq E \\ a \in E \setminus \Delta \\ b \in \Delta}} \omega_F(x_0, b, \Delta) d_F(x_0, a) Z_{F,\Xi,x_0}(b, \Delta) \\
&= \sum_{a \in E} d_F(x_0, a) \sum_{\substack{\Delta \subseteq E \\ \Delta \ni a}} Z_{F,\Xi,x_0}(a, \Delta) \\
&\quad + \sum_{a \in E} d_F(x_0, a) \sum_{\Delta \subseteq E \setminus \{a\}} \sum_{b \in \Delta} \omega_F(x_0, b, \Delta) Z_{F,\Xi,x_0}(b, \Delta) \\
&= \sum_{a \in E} d_F(x_0, a) (\zeta_{F,\Xi,x_0}(a, E) + \zeta_{F,\Xi,x_0}(E \setminus \{a\})) \\
&\geq \sum_{a \in E} d_F(x_0, a) (\zeta_{F,\Xi,x_0}(a_0, E) + \zeta_{F,\Xi,x_0}(E \setminus \{a_0\})) .
\end{aligned}$$

That gives what we want since $\sum_{b \in E} d_F(x_0, b) \neq 0$ (by definition 5.11, dispatchers must be positive). □

Proof of Lemma 5.20. By fact 5.22, $\Omega_F(\mathcal{S}(F), \Xi) = \zeta_{F,\Xi,x_0}(D)$. From D we successively remove what we call a_1, a_2, \dots, a_d till we reach the empty set; applying at each step fact 5.24 yields that $\zeta_{F,\Xi,x_0}(D) \geq \sum_{i=1}^d \zeta_{F,\Xi,x_0}(a_i, D \setminus \{a_1, \dots, a_{i-1}\}) + \zeta_{F,\Xi,x_0}(\emptyset)$. By definition, $\zeta_{F,\Xi,x_0}(\emptyset) = 0$. What order \prec_{F,x_0} shall we choose on D ?

Of course: $a_1 \prec_{F,x_0} a_2 \prec_{F,x_0} \dots \prec_{F,x_0} a_d$.

Then by fact 5.21, $\Omega_F(\mathcal{S}(F), \Xi \cup \{(x_0, \prec_{F,x_0})\}) = \sum_{i=1}^d \zeta_{F,\Xi,x_0}(a_i, D \setminus \{a_1, \dots, a_{i-1}\})$. So in the end $\Omega_F(\mathcal{S}(F), \Xi) \geq \Omega_F(\mathcal{S}(F), \Xi \cup \{(x_0, \prec_{F,x_0})\})$.

Theorem 5.25. For any instance F , any positive and unitary weighting seed s_F , when $d_F = s_F$, there exists an instance dependent orientation \prec_F induced by a set $\{(x, \prec_{F,x})\}_{x \in X}$ of total orders on D , such that $|\mathcal{M}_{\prec_F}(F)| \leq W_F(\mathcal{S}(F))$.

Proof. By remark 5.18, $W_F(\mathcal{S}(F)) = \Omega_F(\mathcal{S}(F), \emptyset)$. Starting with $\Xi = \emptyset$, we add elements (x_0, \prec_{F,x_0}) to Ξ such that $\Omega_F(\mathcal{S}(F), \Xi) \geq \Omega_F(\mathcal{S}(F), \Xi \cup \{(x_0, \prec_{F,x_0})\})$, which is possible by lemma 5.20. At the end of the process we have thus $\Omega_F(\mathcal{S}(F), \emptyset) \geq \Omega_F(\mathcal{S}(F), \{(x, \prec_{F,x})\}_{x \in X})$. Let \prec_F be the orientation on $\mathcal{S}(F)$ induced by $\{(x, \prec_{F,x})\}_{x \in X}$.

By remark 5.19, $\Omega_F(\mathcal{S}(F), \{(x, \prec_{F,x})\}_{x \in X}) = |\mathcal{M}_{\prec_F}(F)|$. So $W_F(\mathcal{S}(F)) \geq |\mathcal{M}_{\prec_F}(F)|$. □

Whether this theorem is true for non homogeneous weights remains an open question.

Remark. In the particular case of boolean satisfiability (i.e. when $D = \{0, 1\}$), there is no choice on d_F : the weighting system is necessarily homogeneous. Thus in this case weighting is not better than ordering.

5.6 Instance Independent Case: Ordering and Weighting Are Equivalent

Definition 5.26. The weight of a CSP instance F is:

$$\gamma(F) = W_F(\mathcal{S}(F)) \quad . \quad (5.21)$$

By extension, the weight of a set \mathcal{F} of CSP instances is:

$$\gamma(\mathcal{F}) = \sum_{F \in \mathcal{F}} \gamma(F) \quad . \quad (5.22)$$

A *permutation* over the domain of values is a bijection $\pi : D \rightarrow D$. A *renaming* of values is a family of permutations $\Pi = (\pi_x)_{x \in X}$ over the domain D . For a CSP instance F , let $\Pi(F)$ be the instance where every occurrence of a value a for every variable x are replaced by $\pi_x(a)$. A set of CSP instances \mathcal{F} is said to be *closed under renaming* if for any renaming Π , if $F \in \mathcal{F}$ then $\Pi(F) \in \mathcal{F}$. By abuse of notation, for any valuation v , we denote by $\Pi(v)$ the valuation that assigns value $\pi_x(v(x))$ to variable x .

Let us first give a very simple yet useful fact:

Fact 5.27. Let Π be a renaming, F and G be CSP instances. Then

1. $\sigma \in \mathcal{S}(F)$ iff $\Pi(\sigma) \in \mathcal{S}(\Pi(F))$;
2. $A_{\Pi(F)}(\Pi(\sigma), x) = \pi_x(A_F(\sigma, x))$.

Note that almost all sets of CSP instances we know to be dealt with in the literature are closed under renaming.

Let \mathcal{F} be some set of instances closed under renaming. We prove in the sequel that $\gamma(\mathcal{F}) = \sum_{F \in \mathcal{F}} |\mathcal{M}_{\prec}(F)|$ for any instance independent orientation \prec on solutions as defined in section 5.3. That can be interpreted as follows: on average on \mathcal{F} , the weight of all solutions is equal to the number of minimal solutions, independently of the orientation \prec . The proof idea is to partition the couples (solutions, instances) in a way that the weight of each class of the partition has a weight of 1 and corresponds to a minimal element for \prec .

We define the set \mathcal{C} of couples (σ, F) where F is an element of \mathcal{F} and σ a solution of F :

$$\mathcal{C} = \{(\sigma, F)\}_{\substack{F \in \mathcal{F} \\ \sigma \in \mathcal{S}(F)}} \quad . \quad (5.23)$$

$\gamma(\mathcal{F})$ can be written as

$$\gamma(\mathcal{F}) = \sum_{(\sigma, F) \in \mathcal{C}} W_F(\sigma) \quad . \quad (5.24)$$

For some variable x and some valuations v_1 and v_2 , we define the permutation π_{x, v_1, v_2} on D as the transposition which swaps $v_1(x)$ and $v_2(x)$, and the renaming Π_{v_1, v_2} as the collection of these permutations, variable per variable:

$$\pi_{x, v_1, v_2}(a) = \begin{cases} v_1(x) & \text{if } a = v_2(x) \\ v_2(x) & \text{if } a = v_1(x) \\ a & \text{otherwise} \end{cases} ; \quad (5.25)$$

$$\Pi_{v_1, v_2} = (\pi_{x, v_1, v_2})_{x \in X} \quad . \quad (5.26)$$

Note that these definitions are symmetric in v_1 and v_2 . Moreover note that $\Pi_{v_1, v_2}(v_1) = v_2$ and $\Pi_{v_1, v_2}(v_2) = v_1$.

Consider a formula $F \in \mathcal{F}$ and a solution τ of F . We denote by $\chi_F(\tau)$ the set of valuations σ assigning each variable x one of the values in the set $A_F(\tau, x)$:

$$\chi_F(\tau) = \prod_{x \in X} A_F(\tau, x) . \quad (5.27)$$

When τ is a solution of F , we denote by $C(\tau, F)$ the set of all renamings of (τ, F) ranging in $\chi_F(\tau)$:

$$C(\tau, F) = \{(\sigma, \Pi_{\sigma, \tau}(F))\}_{\sigma \in \chi_F(\tau)} . \quad (5.28)$$

Lemma 5.28. *If τ is a solution of F and $\sigma \in \chi_F(\tau)$, then σ is a solution of $G = \Pi_{\sigma, \tau}(F)$ and for all variable $x \in X$, $A_G(\sigma, x) = A_F(\tau, x)$.*

Proof. Since $G = \Pi_{\sigma, \tau}(F)$, $\sigma = \Pi_{\sigma, \tau}(\tau)$ and τ is a solution of F , by fact 5.27 we know that σ is a solution of G . Moreover by fact 5.27, for every variable x , $A_G(\sigma, x) = \pi_{x, \sigma, \tau}(A_F(\tau, x))$. By definition of χ_F , $\sigma(x) \in A_F(\tau, x)$. Since $\pi_{x, \sigma, \tau}$ swaps two values $\tau(x)$ and $\sigma(x)$ that are both elements of $A_F(\tau, x)$, $\pi_{x, \sigma, \tau}(A_F(\tau, x)) = A_F(\tau, x)$, hence $A_G(\sigma, x) = A_F(\tau, x)$. \square

Lemma 5.29. *The set $\{C(\tau, F)\}_{\substack{F \in \mathcal{F} \\ \tau \in \mathcal{M}_{\prec}(F)}}$ is a partition of \mathcal{C} .*

Proof. If $(\sigma, G) \in C(\tau, F)$, then by lemma 5.28, σ is a solution of G . Moreover, by closure of \mathcal{F} under renaming, $G \in \mathcal{F}$. Thus $C(\tau, F) \subseteq \mathcal{C}$. Now it is sufficient to prove that $\forall (\sigma, G) \in \mathcal{C}$ there exists a unique (τ, F) where $F \in \mathcal{F}$, τ is a minimal solution of F and $(\sigma, G) \in C(\tau, F)$.

- Existence of (τ, F) : for every x , let $\tau(x)$ be the minimal value in $A_G(\sigma, x)$ according to the order $<_x$ underlying \prec ; by construction, $\tau \in \chi_G(\sigma)$. Consider the renaming $\Pi_{\sigma, \tau}$ and let $F = \Pi_{\sigma, \tau}(G)$. By lemma 5.28, τ is a solution of F and for all variable x , $A_F(\tau, x) = A_G(\sigma, x)$. Since for all $x \in X$, $\tau(x)$ is the minimal value in $A_F(\tau, x)$, τ is minimal for the orientation \prec . Moreover for all x , $\sigma(x) \in A_F(\tau, x)$, thus $\sigma \in \chi_F(\tau)$; and since $G = \Pi_{\sigma, \tau}(F)$, we have that $(\sigma, G) \in C(\tau, F)$.
- Uniqueness of (τ, F) : let (τ', F') be such that $C(\tau', F') \ni (\sigma, G)$, i.e. $\sigma \in \chi_{F'}(\tau')$ and $G = \Pi_{\sigma, \tau'}(F')$; then by lemma 5.28, for all variable x , $A_G(\sigma, x) = A_{F'}(\tau', x)$. By minimality of τ' , $\tau'(x)$ must be the minimum of $A_G(\sigma, x)$ for each variable x . \square

Lemma 5.30. *Suppose that the weight w_F is obtained from a unitary and instance independent generator ω . Let (τ, F) be an element of \mathcal{C} ; then $\sum_{(\sigma, G) \in C(\tau, F)} W_G(\sigma) = 1$.*

Proof. First note that by lemma 5.28, for all $(\sigma, G) \in C(\tau, F)$, we have $A_G(\sigma, x) = A_F(\tau, x)$.

Thus:

$$\begin{aligned}
\sum_{(\sigma, G) \in C(\tau, F)} W_G(\sigma) &= \sum_{(\sigma, G) \in C(\tau, F)} \prod_{x \in X} w_G(\sigma, x) \\
&= \sum_{(\sigma, G) \in C(\tau, F)} \prod_{x \in X} \omega(x, \sigma(x), A_G(\sigma, x)) \text{ since } \omega \text{ is instance independent} \\
&= \sum_{\sigma \in \chi_F(\tau)} \prod_{x \in X} \omega(x, \sigma(x), A_F(\tau, x)) \text{ since } A_G(\sigma, x) = A_F(\tau, x) \\
&= \prod_{x \in X} \sum_{\sigma(x) \in A_F(\tau, x)} \omega(x, \sigma(x), A_F(\tau, x)) \\
&= \prod_{x \in X} 1 \text{ since } \omega \text{ is unitary} \\
&= 1 .
\end{aligned}$$

□

Theorem 5.31. *Let \mathcal{F} be a set of CSP instances which is closed under renaming. Let w_F be a weighting system built from a unitary and instance independent weight generator ω . Let \prec be an instance independent orientation. Then it holds that $\sum_{F \in \mathcal{F}} |\mathcal{M}_\prec(F)| = \gamma(\mathcal{F})$.*

Proof. It is a mere combination of lemmas 5.29 and 5.30:

$$\begin{aligned}
\gamma(\mathcal{F}) &= \sum_{F \in \mathcal{F}} \gamma(F) \\
&= \sum_{F \in \mathcal{F}} \sum_{\sigma \in \mathcal{S}(F)} W_F(\sigma) \\
&= \sum_{(\sigma, F) \in C} W_F(\sigma) \\
&= \sum_{\substack{F \in \mathcal{F} \\ \tau \in \mathcal{M}_\prec(F)}} \sum_{(\sigma, G) \in C(\tau, F)} W_G(\sigma) \text{ by lemma 5.29} \\
&= \sum_{\substack{F \in \mathcal{F} \\ \tau \in \mathcal{M}_\prec(F)}} 1 \text{ by lemma 5.30} \\
&= \sum_{F \in \mathcal{F}} |\mathcal{M}_\prec(F)| .
\end{aligned}$$

□

Closure by renaming involves symmetry, so it is not surprising that on average all weightings on the one hand and all orderings on the other hand should be equivalent. What is more surprising though, is the fact that weightings and orderings are equivalent. This is noteworthy because weights are simpler to handle in calculations (they yield more compact and tractable formulas, see chapter 6 for example).

5.7 Boolean Case: a Better Weighting for Partial Valuations

In order to estimate boolean satisfiability of formulas, Maneva & Sinclair - 2008 [MS08] use a so called *Weight Preservation Theorem*. Valuations here are mappings from X to $D = \{0, 1, *\}$.

The value $*$ is to be interpreted as 0 or 1. They call a valuation *valid* iff each clause contains at least one true literal or two starred literals. In this section, one has to be aware about the fact that we define a *boolean solution* as a valid valuation taking its values in $\{0, 1\}$ only! $\mathcal{S}(F)$ still denotes the set of valid valuations of instance F (with values in $\{0, 1, *\}$) and $A_F(\sigma, x)$ still refers to neighborhood in $\mathcal{S}(F)$. Note that any formula has at least one valid valuation: the one that gives the value $*$ to every variable (the so-called trivial cores by Maneva and Sinclair) so the existence of valid assignments does not guarantee the existence of boolean solutions. Nevertheless, counting weighted valid valuations can be used to estimate boolean satisfiability.

Maneva and Sinclair choose their weights as follows: each variable has a weighting seed $s_0(x), s_*(x)$ such that $s_0(x) + s_*(x) = 1$, and for all valid valuation σ and all variable x they put the following weight:

$$q_F(\sigma, x) = \begin{cases} s_*(x) & \text{if } \sigma(x) = * ; \\ s_0(x) & \text{if } \sigma(x) \neq * \text{ and } * \in A_F(\sigma, x) ; \\ s_0(x) + s_*(x) & \text{otherwise .} \end{cases} \quad (5.29)$$

As shown by Maneva, Mossel & Wainwright - 2007 [MMW07], the sum of the weights of all valid valuations reachable from any boolean solution is exactly 1. The reachability property is defined as the existence of a path from the boolean solution to the valid valuation where at each step a variable is given the value $*$ while maintaining the validity property. Since a given valid valuation may be reachable from lots of different boolean solutions (but sometimes from no one), counting the weighted partial valuations hopefully enables to count less than the number of boolean solutions. To get an upper bound for the satisfiability threshold, Maneva & Sinclair - 2008 [MS08] count the so called non trivial cores σ ; a non trivial core σ is a valid assignment with a linear number of non starred variables, all of them such that $A_F(\sigma, x) = \{\sigma(x)\}$. Many non trivial cores are not extensible to solutions; a core is extensible to a solution when there is a boolean valuation of the starred literals which is a boolean solution. They manage to count only cores which are extensible to a boolean solution, and they estimate the satisfiability of the starred part of the formula by weighting valid assignments as defined in equation 5.29. What we show in the following is that this weighting can be improved upon using our framework.

Before we give this improvement and show its correctness, we want to stress an important difference between the weighting of solutions of general CSPs as defined in the previous sections and the weighting defined in this section: in the previous sections, an unsatisfiable formula has always a total weight of 0 while in the present one, an unsatisfiable formula (a formula with no boolean solution) will have a non-zero total weight (provided the weights of the value $*$ are not 0). This is the price one has to pay to lower the weights of satisfiable formulas. This fact makes difficult to establish a general comparison between both methods, because they are highly dependent on the set of instances that are considered and in particular on the proportion of unsatisfiable instances among them.

To improve on Maneva et al.'s estimation system, we choose the following weights: each variable x has a unitary weighting seed $s_F(x, 0), s_F(x, 1)$ and $s_F(x, *)$. From this seed s_F we define the weight generator ω_F as follows:

$$\omega_F(x, a, \Delta) = \begin{cases} s_F(x, a) & \text{if } a = * \text{ and } a \in \Delta ; \\ s_F(x, a) + \sum_{b \in D \setminus \Delta} s_F(x, b) & \text{if } a \neq * \text{ and } a \in \Delta ; \\ 0 & \text{if } a \notin \Delta . \end{cases} \quad (5.30)$$

As before in section 5.4.5, we define the actual weight $w_F(\sigma, x) = \omega_F(x, \sigma(x), A_F(\sigma, x))$.

Remark 5.32. Noticeable values of ω_F :

$$\begin{aligned}\omega_F(x, 0, \{0\}) &= \omega_F(x, 1, \{1\}) = s_F(x, 0) + s_F(x, 1) + s_F(x, *) = 1 ; \\ \omega_F(x, 0, \{0, *\}) &= \omega_F(x, 1, \{1, *\}) = s_F(x, 0) + s_F(x, 1) ; \\ \omega_F(x, a, \{0, 1, *\}) &= s_F(x, a) ; \\ \omega_F(x, *, \Delta) &= s_F(x, *) \text{ if } * \in \Delta .\end{aligned}$$

Remark 5.33. ω_F is almost unitary, since for all nonempty $\Delta \subseteq D$, $\Delta \neq \{*\}$, $\Delta \neq \{0, 1\}$, $\sum_{a \in \Delta} \omega(x, a, \Delta) = 1$; $\{0, 1\}$ cannot be a clique in this model of validity, because if both 0 and 1 are allowed, so is *. However $\{*\}$ can be a clique, and in this case $\omega(x, *, \{*\}) = s_F(x, *)$.

Our system can be seen as a split of $1 - s_*(x)$ into $s_F(x, 0)$ and $s_F(x, 1)$ (instead of just $s_0(x)$ for Maneva) in the case when $\sigma(x) \neq *$ and $A_F(\sigma, x) = \{0, 1, *\}$; thus our weights are smaller than Maneva's, though we are able to insure that they are correct.

This system is different from the system seeds+dispatchers, because here a fixed variable at value * is given a weight of $s_F(x, *)$, whereas dispatchers would give it a weight of 1. However we are able to use our Weight Conservation Theorem, using the following decomposers:

$$\delta_{F, \sigma, x, a} = \begin{cases} s_F(x, a) & \text{if } \begin{cases} \sigma(x) = a \\ \text{or } (\sigma(x) \neq * \text{ and } a \notin A_F(\sigma, x)) \end{cases} \\ 0 & \text{otherwise} . \end{cases} \quad ; \quad (5.31)$$

We must now prove that the conditions of our Weight Conservation Theorem are satisfied: w_F is decomposable family $(\delta_{F, \sigma, x, a})$ and (T_F, g) covers U_F .

Lemma 5.34. w_F is decomposable by family $(\delta_{F, \sigma, x, a})$.

Proof. By definitions:

1. if $\sigma(x) = *$: $\sum_{a \in D} \delta_{F, \sigma, x, a} = \sum_{a \in D} s_F(x, a) \mathbf{1}_{a=\sigma(x)} = s_F(x, \sigma(x)) = w_F(\sigma, x)$;
2. if $\sigma(x) \neq *$:

$$\begin{aligned}\sum_{a \in D} \delta_{F, \sigma, x, a} &= \sum_{a \in D} s_F(x, a) (\mathbf{1}_{a=\sigma(x)} + \mathbf{1}_{a \notin A_F(\sigma, x)}) \\ &= s_F(x, \sigma(x)) + \sum_{b \notin A_F(\sigma, x)} s_F(x, b) \\ &= w_F(\sigma, x) .\end{aligned}$$

□

Lemma 5.35. Let v be a valuation and g be any connected component of the network of valid valuations $\mathcal{S}(F)$ containing a boolean solution. Then there exists a valid valuation $\sigma \in g$ such that $U_F(v) = T_{F, \sigma \rightarrow v}$.

Proof. Let us take any $v \in D^X$ and a boolean solution $\sigma_0 \in g$. At the beginning we put $\sigma = \sigma_0$. Consider the following procedure:

- If there is a variable $x \in X$ such that $\sigma(x) \neq v(x)$ and $\sigma_{x \leftarrow v(x)}$ remains a valid valuation, then change σ to $\sigma_{x \leftarrow v(x)}$.

We iterate this procedure till there is no variable $x \in X$ such that $\sigma(x) \neq v(x)$ and $\sigma_{x \leftarrow v(x)}$ remains a valid valuation. This eventually happens because at each step we make a move towards v , and X is finite. So in the end, each variable in σ has either its initial boolean value in σ_0 or the value given by v . In other words, for any $x \in X$, either $\sigma(x) = v(x)$ or $(\sigma(x) \neq * \text{ and } v(x) \notin A_F(\sigma, x))$. Thus by equation 5.31, $\delta_{F, \sigma, x, v(x)} = s_F(x, v(x))$, which in turn by definitions 5.4 and 5.6 yields $T_{F, \sigma \rightarrow v} = U_F(v)$. Moreover, by construction, the ending σ is also in g . \square

Corollary 5.36. *Let g be any connected component of the network of valid valuations $S(F)$ containing a boolean solution. Then (T_F, g) covers U_F .*

Thus our weighting system obeys the Weight Conservation Theorem, and we can conclude that $\gamma(F) = W_F(S(F)) \geq W_F(g) \geq 1$ and state the following theorem:

Theorem 5.37. *w_F as defined in equation 5.30 yields $\gamma(F) \geq 1$ whenever F admits a boolean solution.*

Remark. We cannot apply theorem 5.31 because there is no closure by renaming; namely $\{0, 1\}$ cannot be a clique (if both 0 and 1 are allowed, so is $*$), whereas $\{0, *\}$ and $\{1, *\}$ can.

Note that in the particular case where for all $x \in X$, $s_F(x, 0) = 0$, $s_F(x, 1) = 1$ and $s_F(x, *) = 0$, we count what Dubois & Boufkhad - 1997 [DB97] call Negatively Prime Solutions (NPSs). Moreover, as soon as $s_F(x, *) = 0$, this weighting can be seen as seeds+dispatchers on a boolean domain (so this weighting is homogeneous).

We used the weighting defined in equation 5.30 to compute an upper bound of the threshold of random 3-SAT: taking seeds independent of F and x , we obtained the best estimation when $s_F(x, *) = 0$, see section 5.7.1 below. We conjecture that even if one takes seeds dependent on F or x , the best choice of $s_F(x, *)$ to estimate boolean unsatisfiability remains indeed 0. The reason why we think so, is that, as described in remark 5.33, ω_F is almost unitary, except for clique $\{*\}$, in which case $\omega_F(x, *, \{*\}) = s_F(x, *)$.

5.7.1 Calculation of the First Moment in this Framework

Notations.

X : total weight of valid partial assignments

α_* : proportion of variables at value $*$ (they get a weight of ω_*);

$\alpha_{i,0}$: proportion of invertible variables at value 0 (they get a weight of ω_0);

$\alpha_{i,1}$: proportion of invertible variables at value 1 (they get a weight of ω_1);

α_s : proportion of starrable but non-invertible variables (they get a weight of $\omega_0 + \omega_1$);

α_c : proportion of non-starrable variables (they get a weight of 1);

β_1 : proportion of clauses making a (true) variable non-starrable (i.e. the clauses of type TFF);

β_2 : proportion of clauses making a (true) starrable variable non-invertible (i.e. some of the clauses of type TF*);

β_3 : proportion of other clauses.

Expression of the first moment.

The first moment of X can be split up into the following factors: total number of assignments and probability for an assignment to be a solution.

1. total number of assignments:

- a) choose subsets of the different types of variables: $\binom{n}{\alpha_* n, \alpha_{i,0} n, \alpha_{i,1} n, \alpha_s n, \alpha_c n}$;
 - b) choose 0 or 1 for not yet assigned variables: $2^{(\alpha_s + \alpha_c)n}$;
2. weighting: we put a weight of $\omega_*^{\alpha_* n} \omega_0^{\alpha_{i,0} n} \omega_1^{\alpha_{i,1} n} (\omega_0 + \omega_1)^{\alpha_s n}$;
 3. probability for an assignment to be a solution: quotient of the number of satisfied formulas by the total number of formulas:
 - a) number of satisfied formulas:
 - i. we give each clause its type: $\binom{cn}{\beta_1 cn, \beta_2 cn, \beta_3 cn}$;
 - ii. we choose in each clause of type β_1 which variable is made non starrable: $S(\beta_1 cn, \alpha_c n) (\alpha_c n)!$;
 - iii. we draw 2 false literals for each of these clauses: $\hat{C}(\alpha_*)^{\beta_1 cn}$;
 - iv. we choose in each clause of type β_2 which starrable variable is made non invertible: $S(\beta_2 cn, \alpha_s n) (\alpha_s n)!$;
 - v. we draw 1 false and 1 starred literals for each of these clauses : $\hat{P}(\alpha_*)^{\beta_2 cn}$;
 - vi. we draw all other literals: $\hat{U}(\alpha_*, \alpha_s)^{\beta_3 cn}$;
 - b) total number of formulas: $(8 \binom{n}{3})^{cn}$.

We denote by \mathcal{P} the set of all families of non-negative numbers $(\alpha_*, \alpha_{i,0}, \alpha_{i,1}, \alpha_s, \alpha_c, \beta_1, \beta_2, \beta_3)$ satisfying the following constraints:

$$\begin{aligned} \alpha_* + \alpha_{i,0} + \alpha_{i,1} + \alpha_s + \alpha_c &= 1 ; \\ \beta_1 + \beta_2 + \beta_3 &= 1 . \end{aligned}$$

Note that \mathcal{P} is convex (by linearity of constraints). We denote by $\mathcal{I}(n)$ the intersection of \mathcal{P} with the multiples of $\frac{1}{n}$; we get the following expression of the first moment:

$$EX = \sum_{(\alpha_*, \alpha_{i,0}, \alpha_{i,1}, \alpha_s, \alpha_c, \beta_1, \beta_2, \beta_3) \in \mathcal{I}(n)} T_1(n)$$

where

$$\begin{aligned} T_1(n) &= \binom{n}{\alpha_* n, \alpha_{i,0} n, \alpha_{i,1} n, \alpha_s n, \alpha_c n} 2^{(\alpha_s + \alpha_c)n} \omega_*^{\alpha_* n} \omega_0^{\alpha_{i,0} n} \omega_1^{\alpha_{i,1} n} (\omega_0 + \omega_1)^{\alpha_s n} \\ &\cdot \binom{cn}{\beta_1 cn, \beta_2 cn, \beta_3 cn} S(\beta_1 cn, \alpha_c n) (\alpha_c n)! S(\beta_2 cn, \alpha_s n) (\alpha_s n)! \\ &\cdot \frac{\hat{C}(\alpha_*)^{\beta_1 cn} \hat{P}(\alpha_*)^{\beta_2 cn} \hat{U}(\alpha_*, \alpha_s)^{\beta_3 cn}}{(8 \binom{n}{3})^{cn}} . \end{aligned}$$

We get rid of multinomials thanks to inequalities B.2 and of Stirling numbers thanks to Temme's estimates (see appendix B.2), so $T_1(n) \leq \text{poly}(n) F_1^n$ where

$$\begin{aligned}
F_1 &= \alpha_*^{-\alpha_*} \alpha_{i,0}^{-\alpha_{i,0}} \alpha_{i,1}^{-\alpha_{i,1}} \alpha_s^{-\alpha_s} \alpha_c^{-\alpha_c} 2^{\alpha_s + \alpha_c} \omega_*^{\alpha_*} \omega_0^{\alpha_{i,0}} \omega_1^{\alpha_{i,1}} (\omega_0 + \omega_1)^{\alpha_s} \\
&\cdot \left(\beta_1^{-\beta_1} \beta_2^{-\beta_2} \beta_3^{-\beta_3} \right)^c (e^x - 1)^{\alpha_c} \left(\frac{\beta_1 c}{ex} \right)^{\beta_1 c} (e^y - 1)^{\alpha_s} \left(\frac{\beta_2 c}{ey} \right)^{\beta_2 c} \\
&\cdot \left(\frac{6}{8} \right)^c C(\alpha_*)^{\beta_1 c} P(\alpha_*)^{\beta_2 c} U(\alpha_*, \alpha_s)^{\beta_3 c}
\end{aligned}$$

with x and y defined as the positive solutions to the following equations:

$$\begin{aligned}
\frac{\alpha_c x}{\beta_1 c} &= 1 - e^{-x} ; \\
\frac{\alpha_s y}{\beta_2 c} &= 1 - e^{-y} .
\end{aligned}$$

Since $\mathcal{I}(n)$ consists of a variable which can take at most $n + 1$ values (as a multiple of $\frac{1}{n}$ ranging between 0 and 1), there is a polynomial $\text{poly}_1(n)$ such that

$$EX \leq \text{poly}_1(n) \left(\max_{\delta \in \mathcal{I}(n)} F_1 \right)^n .$$

What are the expressions of \hat{C} , \hat{P} and \hat{U} ?

1. for \hat{C} , just choose 2 false literals;
2. for \hat{P} , just choose 1 false and 1 starred literals;
3. for \hat{U} , it is a little trickier: we remove from all possibilities clauses of the following types: FFF, FF*, TFF, TF* with T among $1 - \alpha_* - \alpha_c$ and F not among $1 - \alpha_* - \alpha_c$, and TF* with both T and F among $1 - \alpha_* - \alpha_c$:

$$\begin{aligned}
\hat{C}(\alpha_*) &= \binom{(1 - \alpha_*)n}{2} ; \\
\hat{P}(\alpha_*) &= 2\alpha_* (1 - \alpha_*) n^2 ; \\
\hat{U}(\alpha_*, \alpha_s) &= 8 \binom{n}{3} - \binom{(1 - \alpha_*)n}{3} - 2\alpha_* n \binom{(1 - \alpha_*)n}{2} - 3 \binom{(1 - \alpha_*)n}{3} \\
&\quad - 2\alpha_* (1 - \alpha_* - \alpha_c) \alpha_c n^3 - 4\alpha_* n \binom{(1 - \alpha_* - \alpha_c)n}{2} .
\end{aligned}$$

What are the expressions of C , P and U ?

$$\begin{aligned}
C(\alpha_*) &= \frac{(1 - \alpha_*)^2}{2} \\
P(\alpha_*) &= 2\alpha_* (1 - \alpha_*) \\
U(\alpha_*, \alpha_s) &= \frac{8}{6} - \frac{(1 - \alpha_*)^3}{6} - 2\alpha_* \frac{(1 - \alpha_*)^2}{2} - 3 \frac{(1 - \alpha_*)^3}{6} \\
&\quad - 2\alpha_* (1 - \alpha_* - \alpha_c) \alpha_c - 4\alpha_* \frac{(1 - \alpha_* - \alpha_c)^2}{2}
\end{aligned}$$

i.e., after simplifications:

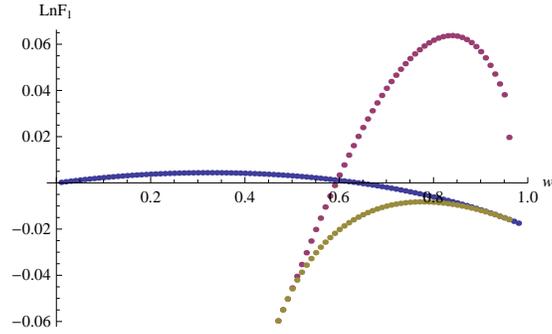


Figure 5.7: Here is a plot of $\ln F_1$ with respect to ω_* for 3 families of solutions canceling out the derivative of $\ln F_1$ and satisfying the constraints, at $c = 4.643$. We can see here that we cannot get a better upper bound than 4.643.

$$\begin{aligned} C(\alpha_*) &= \frac{(1 - \alpha_*)^2}{2} ; \\ P(\alpha_*) &= 2\alpha_*(1 - \alpha_*) ; \\ U(\alpha_*, \alpha_s) &= \frac{4}{3} - \frac{2(1 - \alpha_*)^3}{3} - \alpha_*(1 - \alpha_*)(3 - 3\alpha_* - 2\alpha_c) . \end{aligned}$$

Maximization.

- if $\omega_* = 0$, which amounts to taking the binary domain $D = \{0, 1\}$, any choice of ω_0 will yield an homogeneous weighting, and theorem 5.31 tells us that weightings and orderings are equivalent on average. In this case they correspond to the Negatively Prime Solutions introduced in section 3.2.4. Thus we are precisely counting NPSs, and the critical c is $c = 4.643$;
- if $\omega_* = 1$, then the all-star assignment has a weight of 1 and the critical c is $c = +\infty$;
- otherwise, numerical calculations reveal that the best critical c we can get is $c = 4.643$, obtained for $\omega_* = 0.0003$, as depicted on figure 5.7.

5.7.2 Calculation of the First Moment with Restricted Validity

Here we consider partial assignment to be valid when there are no clauses of type FFF, FF*, F** or ***. Here starrable is equivalent to invertible. Why is it interesting to do this calculation? Because we saw just above in section 5.7.1 that in this framework the best ω_* is $\omega_* = 0$ (NPSs) whereas with restricted validity FFF+FF*+F**+***, we saw in section 4.3.3 that the best ω_* is $\omega_* = 1$ (prime implicants). So we are doing here a combination of NPSs and prime implicants.

Notations.

X : total weight of valid partial assignments

α_* : proportion of variables at value $*$ (they get a weight of ω_*);

$\alpha_{i,0}$: proportion of invertible variables at value 0 (they get a weight of ω_0);

$\alpha_{i,1}$: proportion of invertible variables at value 1 (they get a weight of ω_1);

- α_c : proportion of non-starrable variables (they get a weight of 1);
 β_1 : proportion of clauses making a (true) variable non-starrable (i.e. the clauses of type TFF, TF* or T**);
 β_2 : proportion of other clauses.

Expression of the first moment.

The first moment of X can be split up into the following factors: total number of assignments and probability for an assignment to be a solution.

1. total number of assignments:
 - a) choose subsets of the different types of variables: $\binom{n}{\alpha_*n, \alpha_{i,0}n, \alpha_{i,1}n, \alpha_cn}$;
 - b) choose 0 or 1 for not yet assigned variables: 2^{α_cn} ;
2. weighting: we put a weight of $\omega_*^{\alpha_*n} \omega_0^{\alpha_{i,0}n} \omega_1^{\alpha_{i,1}n}$;
3. probability for an assignment to be a solution: quotient of the number of satisfied formulas by the total number of formulas:
 - a) number of satisfied formulas:
 - i. we give each clause its type: $\binom{cn}{\beta_1cn, \beta_2cn}$;
 - ii. we choose in each clause of type β_1 which variable is made non-starrable: $S(\beta_1cn, \alpha_cn) (\alpha_cn)!$;
 - iii. we draw 2 false literals for each of these clauses: $\hat{C}(\alpha_*)^{\beta_1cn}$;
 - iv. we draw all other literals: $\hat{U}(\alpha_*)^{\beta_2cn}$;
 - b) total number of formulas: $(8\binom{n}{3})^{cn}$.

We denote by \mathcal{P} the set of all families of non-negative numbers $(\alpha_*, \alpha_{i,0}, \alpha_{i,1}, \alpha_c, \beta_1, \beta_2)$ satisfying the following constraints:

$$\begin{aligned} \alpha_* + \alpha_{i,0} + \alpha_{i,1} + \alpha_c &= 1 ; \\ \beta_1 + \beta_2 &= 1 . \end{aligned}$$

Note that \mathcal{P} is convex (by linearity of constraints). We denote by $\mathcal{I}(n)$ the intersection of \mathcal{P} with the multiples of $\frac{1}{n}$; we get the following expression of the first moment:

$$EX = \sum_{(\alpha_*, \alpha_{i,0}, \alpha_{i,1}, \alpha_c, \beta_1, \beta_2) \in \mathcal{I}(n)} T_1(n)$$

where

$$\begin{aligned} T_1(n) &= \binom{n}{\alpha_*n, \alpha_{i,0}n, \alpha_{i,1}n, \alpha_cn} 2^{\alpha_cn} \omega_*^{\alpha_*n} \omega_0^{\alpha_{i,0}n} \omega_1^{\alpha_{i,1}n} \\ &\cdot \binom{cn}{\beta_1cn, \beta_2cn} S(\beta_1cn, \alpha_cn) (\alpha_cn)! \\ &\cdot \frac{\hat{C}(\alpha_*)^{\beta_1cn} \hat{U}(\alpha_*)^{\beta_2cn}}{(8\binom{n}{3})^{cn}} . \end{aligned}$$

We get rid of multinomials thanks to inequalities B.2 and of Stirling numbers thanks to Temme's estimates (see appendix B.2), so $T_1(n) \leq \text{poly}(n) F_1^n$ where

$$\begin{aligned} F_1 &= \alpha_*^{-\alpha_*} \alpha_{i,0}^{-\alpha_{i,0}} \alpha_{i,1}^{-\alpha_{i,1}} \alpha_c^{-\alpha_c} 2^{\alpha_c} \omega_*^{\alpha_*} \omega_0^{\alpha_{i,0}} \omega_1^{\alpha_{i,1}} \\ &\quad \cdot \left(\beta_1^{-\beta_1} \beta_2^{-\beta_2} \right)^c (e^x - 1)^{\alpha_c} \left(\frac{\beta_1 c}{ex} \right)^{\beta_1 c} \\ &\quad \cdot \left(\frac{6}{8} \right)^c C(\alpha_*)^{\beta_1 c} U(\alpha_*)^{\beta_2 c} \end{aligned}$$

with x defined as the positive solution to the following equation:

$$\frac{\alpha_c x}{\beta_1 c} = 1 - e^{-x} .$$

Since $\mathcal{I}(n)$ consists of a variable which can take at most $n+1$ values (as a multiple of $\frac{1}{n}$ ranging between 0 and 1), there is a polynomial $\text{poly}_1(n)$ such that

$$EX \leq \text{poly}_1(n) \left(\max_{\delta \in \mathcal{I}(n)} F_1 \right)^n .$$

What are the expressions of \hat{C} , \hat{P} and \hat{U} ?

1. for \hat{C} , choose 2 false literals or (1 false and 1 starred literals) or 2 starred literals;
2. for \hat{U} , remove from all possibilities clauses of the following types: FFF, FF*, F**, ***, TFF, TF* and T**:

$$\begin{aligned} \hat{C}(\alpha_*) &= \binom{(1-\alpha_*)n}{2} + 2\alpha_* (1-\alpha_*)n^2 + 4 \binom{\alpha_* n}{2} ; \\ \hat{U}(\alpha_*) &= 8 \binom{n}{3} - \binom{(1-\alpha_*)n}{3} - 2\alpha_* n \binom{(1-\alpha_*)n}{2} - 4 \binom{\alpha_* n}{2} (1-\alpha_*)n - 8 \binom{\alpha_* n}{3} \\ &\quad - 3 \binom{(1-\alpha_*)n}{3} - 4 \binom{(1-\alpha_*)n}{2} \alpha_* n - 4 (1-\alpha_*)n \binom{\alpha_* n}{2} . \end{aligned}$$

What are the expressions of C , P and U ?

$$\begin{aligned} C(\alpha_*) &= \frac{(1-\alpha_*)^2}{2} + 2\alpha_* (1-\alpha_*) + 4 \frac{\alpha_*^2}{2} \\ U(\alpha_*) &= \frac{8}{6} - \frac{(1-\alpha_*)^3}{6} - 2\alpha_* \frac{(1-\alpha_*)^2}{2} - 4 \frac{\alpha_*^2}{2} (1-\alpha_*) - 8 \frac{\alpha_*^3}{6} \\ &\quad - 3 \frac{(1-\alpha_*)^3}{6} - 4 \frac{(1-\alpha_*)^2}{2} \alpha_* - 4 (1-\alpha_*) \frac{\alpha_*^2}{2} \end{aligned}$$

i.e., after simplifications:

$$\begin{aligned} C(\alpha_*) &= \frac{(1-\alpha_*)^2}{2} + 2\alpha_* ; \\ U(\alpha_*) &= \frac{4}{3} - \frac{2(1-\alpha_*)^3}{3} - 3\alpha_* (1-\alpha_*)^2 - 4\alpha_*^2 (1-\alpha_*) - \frac{4\alpha_*^3}{3} . \end{aligned}$$

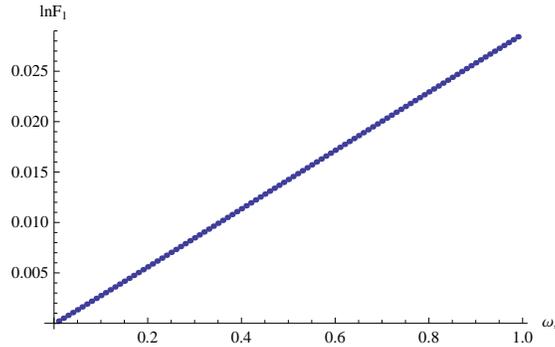


Figure 5.8: Here is a plot of $\ln F_1$ with respect to ω_* for a family of solutions canceling out the derivative of $\ln F_1$ and satisfying the constraints, at $c = 4.643$.

Maximization.

- if $\omega_* = 0$, which amounts to taking the binary domain $D = \{0,1\}$, any choice of ω_0 will yield an homogeneous weighting, and theorem 5.31 tells us that weightings and orderings are equivalent on average. In this case they correspond to the negatively prime solutions. Since we are precisely counting NPSs, and the critical c is $c = 4.643$;
- if $\omega_* = 1$, then we count prime implicants, and the critical c is $c = 4.883$;
- otherwise, numerical calculations reveal that the best critical c we can get is $c = 4.643$, obtained for $\omega_* = 0.001$, as depicted on figure 5.8.

5.8 Conclusion and Perspectives

Through our Weight Conservation Theorem we gave sufficient conditions to have a correct weighting on solutions of CSPs. We were able to apply it to two different weightings: the first one, which is very general, was built from seeds and dispatchers; the second one was specifically designed to improve on Maneva et al.'s weighting. We also showed an equivalence between weighting and ordering over a set closed under renaming when they are instance independent. On the contrary, when weighting and ordering may depend on instances, we showed that given an homogeneous weighting it is possible to find an ordering which is not worse, but what happens for heterogeneous weightings? is it always possible to find for a given weighting a corresponding ordering?

Other perspectives include: is it possible to define a correct non-uniform weighting? how to generalize boolean partial valuations to general CSPs? how to extend our weighting when considering neighbors of neighbors, or more generally neighbors at bounded distance?

Distributional Models and Non-Uniformity

SINCE we were not able in the previous chapters to get lower upper bounds than 4.643 on the threshold of 3-SAT through the First Moment Method, despite lots of efforts to tune the weights, we are going now to change models! Namely it turns out that there is a framework which is better (but trickier to handle) than drawing models (introduced in section (1.1.3)): distributional models (introduced in section (1.1.4)). Indeed upper bounds obtained in distributional models are significantly lower than in drawing models (see table (6.2) for a comparison). Moreover distributional models allow us to make more precise tunings on the types of variables present in the formulas and consequently to define more precise weights.

We obtain some new upper bounds in a variety of models of 3-CNF formulas (which we introduce later in section 6.1.1) thanks to a new technique for selecting solutions. In the particular case of the standard model we get an upper bound of 4.500. We must mention here the work of Díaz, Kirousis, Mitsche & Pérez-Giménez - 2009 [DKMPG09]; gathering the technique of Dubois, Boufkhad & Mandler [DBM00, DBM03] with a pure literal elimination and a filtering on the typicality of clauses, they got an upper bound of 4.490. The fact is that our new technique is quite compatible with the pure literal elimination and the filtering on the typicality of clauses, but we only aim at emphasizing the positive effect of our new technique for selecting solutions, by comparing it to previous analogous techniques in several models of formulas.

The best implementations of the first moment method approximating the threshold of 3-SAT use local relationships between solutions, which involves solutions agreeing on the values of all variables but a constant number of them, in general one variable (see Dubois & Boufkhad - 1997 [DB97] or Kirousis, Kranakis & Krizanc - 1997 [KKK97]) or two (see Kirousis, Kranakis, Krizanc & Stamatiou - 1998 [KKKS98]).

We shall consider the set of solutions with local relationship as a graph which nodes are the solutions and an edge exists between two solutions if and only if both solutions agree on the values of all variables except one. Each edge will be labeled by the variable differing between both solutions (the so-called *free* variable).

For example the formula

$$\Phi = \{a \vee b \vee c, a \vee c \vee \bar{d}, a \vee \bar{c} \vee \bar{d}, a \vee \bar{b} \vee \bar{d}, \bar{b} \vee c \vee \bar{d}, \bar{a} \vee \bar{b} \vee \bar{d}, \bar{a} \vee b \vee \bar{c}\}$$

has 7 solutions that can be represented by the non-oriented Hamming graph of figure 6.1.

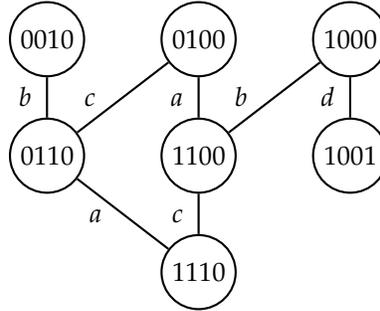


Figure 6.1: Hamming graph of solutions for formula Φ . The label of an edge is the name of the variable differing between both solutions.

The techniques used so far amount to making an acyclic orientation of the above graph and to counting only the minimal solutions (those that do not have outgoing edges). The least is the number of minimal solutions the best is the upper bound obtained. In general, any graph can be oriented so as to obtain only one minimal element for every connected component (e.g. by a depth first search), but this orientation is obtained thanks to a sophisticated algorithm that is aware of the whole graph while in our case, the orientation must be decided locally.

The very first orientation introduced by Dubois & Boufkhad - 1997 [DB97] and Kirousis, Kranakis, Krizanc - 1997 [KKK97] consisted in orienting an edge from the solution where the label variable is assigned 0 to the one where it is 1 regardless of which variable is considered. Later, in the works of Dubois, Boufkhad & Mandler [DBM03] and Díaz, Kirousis, Mitsche & Pérez-Giménez - 2009 [DKMPG09], an edge is oriented towards the value that makes true the most literals and this can be known thanks to the syntactic property of the number of occurrences of each variable in the formula. In both these types of orientation, the edges having the same labels are oriented the same way (e.g. from 0 to 1) anywhere in the graph. So we call such orientations *uniform* (see Figure 6.2).

The orientation that we use in this paper is less rigid: two edges labeled with the same variable can be oriented differently depending on the solutions involved (that is what we call *non-uniform* orientation, see Figure 6.3). Indeed we keep track of a set of 5 numbers associated with each variable and use it to discriminate among neighboring solutions. These 5 numbers provide information on the repartition of true and false occurrences of each variable in each type of clauses (clauses having 1, 2 or 3 true literals). Our intuition is that we should select solutions in which the least occurrences of true literals are *critical*. The less a clause has true literals, the more its true literals are critical. Such a property is by nature non-uniform.

We develop our technique in a general framework allowing us to apply it to a wide variety of 3-CNF models of formulas defined by their distributions; thus we derive new bounds for some known models of formulas (see Boufkhad, Dubois, Interian & Selman - 2005 [BDIS05]). The existence of other non-uniform orientations that may give a smaller number of minimal elements and then better bounds remains to be investigated.

In section 6.1 we present our framework and four different models of formulas; in section 6.2 we show how we make our non-uniform selection of solutions, and sum up the bounds we obtain for each model. We give details on the calculation of the first moment and its constraints in section 6.3, as well as some hints on what led us to the weights we took for our non-uniform

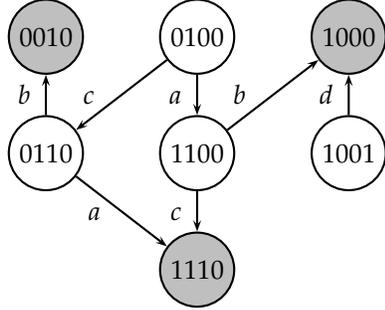


Figure 6.2: Uniform orientation. For example b has 2 positive occurrences and 3 negative ones, so every edge labeled by b is oriented from 1 to 0. Minimal solutions are in gray.

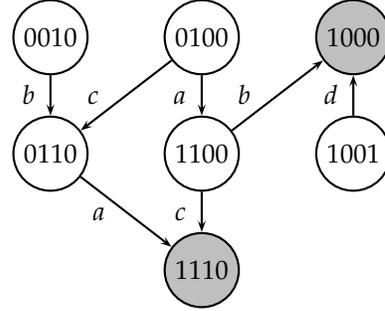


Figure 6.3: Non-uniform orientation, obtained in this example by minimizing $4\beta_1 + 2\beta_2 + \beta_3$ (see definition in section 6.1). Both edges labeled by b are oriented differently (i.e. from 0 to 1 as well as from 1 to 0). Minimal solutions are in gray.

selection.

Our calculations lead us to as classical problem of constrained maximization, and we solve it classically as well, through the Lagrange multipliers method (appendice A.1) and following the approach of [DKMPG09]. Thus we must prove that the objective function does not maximize at the boundary of the polytope (appendice A.2). Moreover we check that the solution we get is indeed a global maximum by a sweep over the polytope (appendice A.3).

The material contained in this chapter was written in collaboration with Yacine Boufkhad and is published as [BH10] and [HB10].

6.1 Definitions and Notations

6.1.1 Overview of Models

Standard Model: see section (1.1.4). We recall equation (1.1):

$$d_{p,q} = \binom{p+q}{p} \frac{e^{-3c}}{(p+q)!} \left(\frac{3c}{2}\right)^{p+q}.$$

By analogy with the standard model we now define several other models where we force an equilibrium between variables occurrences and/or signs. These can be seen as *regular* variants of 3-SAT (just like regular graphs). The equilibrium cannot be perfect because of parity or truncation reasons, but we circumvent it as follows. Of course one can check that all of these distributions sum up to 1 and have an average of $3c$.

Model with Almost Balanced Signs: every variable appear with (almost) the same number of positive and negative occurrences; we define $d_{p,q}$ by

$$\begin{aligned} d_{p,p} &= \frac{e^{-3c} (3c)^{2p}}{(2p)!} \\ d_{p+1,p} = d_{p,p+1} &= \frac{1}{2} \frac{e^{-3c} (3c)^{2p+1}}{(2p+1)!} \end{aligned}$$

and zero elsewhere.

Model with Almost Balanced Occurrences: every variable appear with (almost) the same number occurrences; let $t^* = \lfloor 3c \rfloor$ and $r^* = 3c - t^*$; we define $d_{p,q}$ by

$$\begin{aligned} d_{p,t^*-p} &= (1 - r^*) \frac{\binom{t^*}{p}}{2^{t^*}} \\ d_{p,t^*+1-p} &= r^* \frac{\binom{t^*+1}{p}}{2^{t^*+1}} \end{aligned}$$

and zero elsewhere.

Model with Almost Balanced Signs and Occurrences: every variable appear with (almost) the same number occurrences and have strictly the same number of positive as negative occurrences (this model was introduced and examined by Boufkhad, Dubois, Interian & Selman - 2005 [BDIS05]); let $p^* = \lfloor \frac{3c}{2} \rfloor$ and $r^* = \frac{3c}{2} - p^*$. We define $d_{p,q}$ by

$$\begin{aligned} d_{p^*,p^*} &= 1 - r^* \\ d_{p^*+1,p^*+1} &= r^* \end{aligned}$$

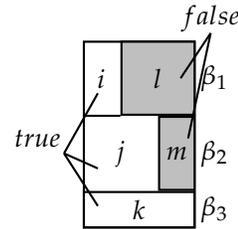
and zero elsewhere.

6.1.2 Types of Clauses and Variables

Our selection method is based on different types of clauses: given any assignment, we call clause of *type* t a clause having t true literals under this assignment, and β_t the proportion of clauses of type t .

Moreover we want to have some control on the number of occurrences of variables in the different types of clauses; to do so we need 6 numbers per variable, so we say that a variable is of *type* (i, j, k, l, m, v) if it is assigned v and has:

- i** true occurrences in clauses of type 1;
- j** true occurrences in clauses of type 2;
- k** true occurrences in clauses of type 3;
- l** false occurrences in clauses of type 1;
- m** false occurrences in clauses of type 2;



Remark 6.1. For each variable we have $i + j + k = p$ and $l + m = q$ or vice versa (according to the value v assigned to the variable).

Then we put some weights onto the solutions as follows: in a given solution each variable of type (i, j, k, l, m, v) receives a weight $\omega_{i,j,k,l,m,v}$. The weight of a solution will be the product of the weights of all variables. It turns out that in the end we shall take binary weights, yielding in fact an orientation between solutions. We explain the choice of the weights in sections 6.2 and 6.3.4. Then we apply the first moment method to the random variable X equal to the sum of the weights of the solutions.

6.2 Selection of Solutions

6.2.1 Construction of a Correct Weighting Scheme

Of course we must put some constraints onto the weights in order that the weighting scheme can be correct for the first moment method: namely the sum of the weights of the solutions of a satisfiable formula must be at least 1. However the constraints we choose here might not be necessary for the first moment method to hold.

Let us recall that given a solution, a variable is called *free* when the assignment obtained by inverting its value (0/1) remains a solution. Thus in our framework, a variable is free iff its i number is 0. How does the tuple $(0, j, k, l, m, v)$ for a free variable x behave when the value v is inverted to $1 - v$?

$$\begin{aligned} i(x) &\leftarrow 0 \\ j(x) &\leftrightarrow l(x) \\ k(x) &\leftrightarrow m(x) \\ v(x) &\leftarrow 1 - v(x) \end{aligned}$$

1. the first constraint we put is that $\omega_{i,j,k,l,m,v} = 1$ as soon as $i \geq 1$; that is, we put significant weights only onto free variables. The reason for this is that free variables allow to move between solutions.
2. the second constraint is that

$$\omega_{0,j,k,l,m,v} + \omega_{0,l,m,j,k,1-v} = 1 ; \quad (6.1)$$

that is, the sum of the weights of a free variable in a couple of solutions differing only on that variable is 1. We impose this condition by analogy with the conditions on weights given by Ardila & Maneva - 2009 [AM09] and in chapter (5).

As suggested by the analysis given in section 6.3.4, we shall take $\omega_{0,j,k,l,m,v} = \mathbf{1}_{P(j,k,l,m,v)}$ for a certain predicate $P(j, k, l, m, v)$ linked with the sign of $\alpha_1 \rho_{j,l} + \alpha_3 \rho_{k,m}$ (where α_1 and α_3 are any real constants and ρ is an operator defined as $\rho_{a,b} = a - b$).

The fact that we imposed $\omega_{0,j,k,l,m,v} + \omega_{0,l,m,j,k,1-v} = 1$ tells us that given a solution and a free variable x at the value v , the predicate P is satisfied by x at the value v or (exclusively) by x at the value $1 - v$. Thus we are able to define an orientation between neighboring solutions.

Let us say that variable x is *obedient* when P is satisfied. We put an arc between 2 solutions differing only on 1 (free) variable x from the solution S_d (where x is disobedient) to the solution S_o (where x is obedient), and we call that relation $S_d > S_o$. The notation $>$ is not randomly chosen.

Namely our weighting scheme counts 1 for a solution when it does not have any disobedient free variables, and 0 otherwise; but what can ensure that whenever there is a solution, there is also a solution where all free variables are obedient? It suffices that the relation $>$ is circuit-free. Then the transitive closure of $>$ is an order, and we are precisely counting the minimal solutions in that order. Minimal solutions exist because the set of all solutions is finite. So let us see how we can make the relation $>$ circuit-free.

6.2.1.1 Recapitulation of Existing Methods

All Solutions: This method consists in computing the first moment on all solutions:

$$P(j, k, l, m, v) \equiv 1 .$$

Negatively Prime Solutions (NPS): This method consists in counting only solutions which free variables are assigned 1. That is

$$P(j, k, l, m, v) \equiv v > 0 .$$

This method was introduced by Dubois & Boufkhad - 1997 [DB97].

NPS with Imbalance: This method was introduced by Dubois, Boufkhad & Mandler - 2000 [DBM00] and combined to some other ingredients by Díaz, Kirousis, Mitsche & Pérez-Giménez - 2009 [DKMPG09]. This method consists in allowing free variables to take only a value such that the number of true occurrences is larger than the number of negative occurrences of this variable (and in case of equality, ties are broken in favor of the value 1). In other words

$$P(j, k, l, m, v) \equiv (\rho_{j,l} + \rho_{k,m}, v) >_{\text{lex}} (0, 0)$$

where $>_{\text{lex}}$ denotes the lexicographical order.

6.2.1.2 Our Method

May we choose arbitrary real coefficients α_1 and α_3 in the expression of $\alpha_1\rho_{j,l} + \alpha_3\rho_{k,m}$ in order that the first moment method should hold? It turns out that it is the case, and here is a proof of it.

We make the following observation: how does the population of the 3 different types of clauses evolve when a free variable x is flipped?

$$\begin{aligned} \beta_1 cn &+ = \rho_{j,l}(x) \\ \beta_2 cn &+ = (\rho_{k,m} - \rho_{j,l})(x) \\ \beta_3 cn &+ = -\rho_{k,m}(x) \end{aligned}$$

Thus $\alpha_1\rho_{j,l} + \alpha_3\rho_{k,m}$ is the variation of $\alpha_1\beta_1 - \alpha_3\beta_3$; so we may define our predicate P in the following way: $P(j, k, l, m, v) \equiv (\alpha_1\rho_{j,l} + \alpha_3\rho_{k,m}, v) >_{\text{lex}} (0, 0)$; thanks to v we break ties when $\alpha_1\rho_{j,l} + \alpha_3\rho_{k,m} = 0$, so that the underlying relation $>$ between solutions is circuit-free: namely going from S_d to S_o when $S_d > S_o$ strictly increases $(-\alpha_1\beta_1 + \alpha_3\beta_3, v)$ for $>_{\text{lex}}$.

Moreover the exclusion between $P(j, k, l, m, v)$ and $P(l, m, j, k, 1 - v)$ is satisfied, which means that whenever there is a solution with a disobedient free variable, it suffices to flip the value of this variable so that it becomes obedient.

We investigated the best ratio between α_1 and α_3 by numerical experiments.

6.2.2 Summary of Results

As one can see in table 6.1, our method yields in all models a slight improvement on the bounds obtained by former methods. Note that for some models there is a range of values for α which give the same upper bound.

In the model where signs as well as occurrences are balanced, the method of NPS+imbalance is of course the same as the method of NPS, whereas our method is somewhat better than the method of NPS.

The bound we obtain in the standard model is 4.500; this is not better than the bound of 4.490 obtained by Díaz, Kirousis, Mitsche & Pérez-Giménez - 2009 [DKMPG09]. Their calculation adds 2 ingredients to the method of Dubois, Boufkhad & Mandler - 2003 [DBM03]:

Table 6.1: Upper bounds obtained in various distributional models.

model	standard	almost balanced signs	almost balanced occurrences	almost balanced signs and occurrences
all solutions	5.040	3.858	5.046	3.783
NPS $v > 0$	4.552	3.521	4.662	3.548
NPS+imbalance $(\rho_{j,l} + \rho_{k,m}, v) >$ $(0, 0)$	4.506	3.514	4.628	3.548
our method $(\alpha\rho_{j,l} + \rho_{k,m}, v) >$ $(0, 0)$	4.500	3.509	4.623	3.546
our α	$\alpha = 2.00$	$1.01 \leq \alpha \leq$ 1.16	$2.01 \leq \alpha \leq$ 2.24	$\alpha \geq 1.01$

typicality of clauses and elimination of pure literals. These 2 ingredients might be combined to our approach to improve on the 4.490, but this would involve too complicated calculations with respect to the expected improvement. However in models where signs are balanced it is irrelevant to eliminate pure literals.

6.3 The First Moment Method

6.3.1 Types of Variables

We split the set of variables into several sets and subsets of variables. In order to be able to match the original random 3-CNF model of formulas where all literals are drawn independently, we should consider p and q to range in N . For convenience of our forthcoming maximization, we only take into account bounded values of p and q . So we are going to consider 2 kinds of variables, according to their numbers of occurrences. We follow the notations of Díaz, Kirousis, Mitsche & Pérez-Giménez - 2009 [DKMPG09]. We denote by M some integer whose value will be determined according to the required accuracy of the calculations; in practice we shall take $M = 21$. M enables us to define 2 kinds of variables:

1. the set of *light* variables, that is variables which indices are in the set

$$\mathcal{L} = \left\{ (p, q) \in \mathbb{N}^2, p \leq M \wedge q \leq M \wedge d_{p,q} > 0 \right\}; \quad (6.2)$$

they are the most important variables since almost all variables are light in the models we consider; we call $\delta_{p,q}$ the proportion of light variables having p positive occurrences, q negative occurrences, and assigned 1. As a further refinement, we call $\pi_{i,j,k,l,m,v}$ the proportion of variables of type (i, j, k, l, m, v) whose corresponding weight $\omega_{i,j,k,l,m,v}$ is nonzero, and omit the other ones because we shall need all active $\pi_{i,j,k,l,m,v}$ to be nonzero. To connect $\pi_{i,j,k,l,m,v}$'s with $\delta_{p,q}$'s we introduce the following set of tuples of integers:

$Q_{p,q} = \{(i, j, k, l, m) \in \mathbf{N}^5, i + j + k = p \wedge l + m = q\}$; thus we have

$$\sum_{(i,j,k,l,m) \in Q_{p,q}} \pi_{i,j,k,l,m,1} = \delta_{p,q} ; \quad (6.3)$$

$$\sum_{(i,j,k,l,m) \in Q_{q,p}} \pi_{i,j,k,l,m,0} = d_{p,q} - \delta_{p,q} . \quad (6.4)$$

Note that equality 6.4 involves $Q_{q,p}$ whereas equality 6.3 involves $Q_{p,q}$.

2. the set of *heavy* variables, that is all other variables; their indices are thus in the set

$$\mathcal{H} = \left\{ (p, q) \in \mathbf{N}^2, p > M \vee q > M \vee d_{p,q} = 0 \right\} ; \quad (6.5)$$

we weaken the notion of satisfiability by considering that heavy variables are always satisfied, regardless of their signs and values. Doing so is harmless for the validity of the first moment method because we can only increase the number of solutions. In other words we are going to consider heavy variables as indistinguishable members of a tote bag. We call τ the global scaled number of heavy variables: $\tau = \sum_{(p,q) \in \mathcal{H}} d_{p,q}$.

We also need to distinguish some types of occurrences of heavy variables. We call H the global scaled number of occurrences of heavy variables:

$$\begin{aligned} H &= \sum_{(p,q) \in \mathcal{H}} (p + q) d_{p,q} \\ &= 3c - \sum_{(p,q) \in \mathcal{L}} (p + q) d_{p,q} . \end{aligned}$$

According to the types of clauses where occurrences appear, H is divided into H_t 's, where H_t is the scaled number of occurrences of heavy variables in clauses of type t .

We are now ready to write down the expression of the first moment of X , the weight of all solutions.

6.3.2 Expression of the First Moment and its Constraints

We recall that all occurrences of literals are drawn according to the distribution $d_{p,q}$ (see section 6.1). Thus the sample space we consider consists in the $(3cn)!$ permutations of labeled occurrences of literals, and our parameters are $n, c, d_{t,p}, \tau, H$ and $\omega_{i,j,k,l,m,v}$'s (although we must carefully choose the weights $\omega_{i,j,k,l,m,v}$, as explained below in section 6.3.4).

All other quantities: $\beta_t, H_t, \delta_{t,p}$ and $\pi_{i,j,k,l,m,v}$ are variables, and the first moment of X can be split up into a big sum over all variables of the product of the following factors depending on variables: number of assignments, weight of an assignment and probability for an assignment to be a solution.

1. number of assignments: each variable is assigned 0 or 1: $2^{\tau n} \prod_{(p,q) \in \mathcal{L}} \binom{d_{p,q} n}{\delta_{p,q} n}$;
2. weight of an assignment: $\prod_{(p,q) \in \mathcal{L}} \prod_{(i,j,k,l,m) \in Q_{p,q}} \prod_{v \in \{0,1\}} \omega_{i,j,k,l,m,v}^{\pi_{i,j,k,l,m,v} n}$;
3. probability for an assignment to be a solution: quotient of the number of satisfied configurations by the total number of configurations:
 - a) number of satisfied configurations: a configuration can be seen as a set of bins filled with occurrences of literals:

- i. each of the $3cn$ bins is first given a truth value:
there are $\binom{cn}{\beta_1 cn, \beta_2 cn, \beta_3 cn} 3^{(\beta_1 + \beta_2)cn}$ possibilities, and the following constraint appears:

$$\beta_1 + \beta_2 + \beta_3 = 1 . \quad (6.6)$$

- ii. each light literal is given a tuple (i, j, k, l, m) consistently with $d_{p,q}$ and $\delta_{p,q}$. This gives a series of constraints:

$$\sum_{(i,j,k,l,m) \in Q_{p,q}} \pi_{i,j,k,l,m,1} + \sum_{(i,j,k,l,m) \in Q_{q,p}} \pi_{i,j,k,l,m,0} = d_{p,q} . \quad (6.7)$$

Note that $\delta_{p,q} = \sum_{(i,j,k,l,m) \in Q_{p,q}} \pi_{i,j,k,l,m,1}$. Thus, given a family $(\pi_{i,j,k,l,m,v})$, there are

$$\prod_{(p,q) \in \mathcal{L}} \binom{\delta_{p,q} n}{\dots \pi_{i,j,k,l,m,1} n \dots}_{(i,j,k,l,m) \in Q_{p,q}} \cdot \prod_{(p,q) \in \mathcal{L}} \binom{(d_{p,q} - \delta_{p,q}) n}{\dots \pi_{i,j,k,l,m,0} n \dots}_{(i,j,k,l,m) \in Q_{q,p}}$$

possible allocations. Moreover the following constraints appear, so that all occurrences of literals can fit into the destined types of clauses:

$$\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} i \pi_{i,j,k,l,m,v} + H_1 = \beta_1 c ; \quad (6.8)$$

$$\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} j \pi_{i,j,k,l,m,v} + H_2 = 2\beta_2 c ; \quad (6.9)$$

$$\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} k \pi_{i,j,k,l,m,v} + H_3 = 3\beta_3 c ; \quad (6.10)$$

$$\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} l \pi_{i,j,k,l,m,v} = 2\beta_1 c ; \quad (6.11)$$

$$\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} m \pi_{i,j,k,l,m,v} = \beta_2 c . \quad (6.12)$$

- iii. all occurrences of light variables are allocated to the 5 regions:

$$\prod_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} \binom{(i+j+k)(l+m)}{i,j,k} \pi_{i,j,k,l,m,v}^n \text{ allocations are possible;}$$

- iv. all occurrences of heavy variables are allocated to the 3 satisfied regions, which yields $\binom{Hn}{H_1 n, H_2 n, H_3 n}$ possible allocations; and we must add the following constraint:

$$H_1 + H_2 + H_3 = H . \quad (6.13)$$

- v. all permutations of occurrences of literals are possible inside the 5 regions: their number is $(\beta_1 cn)! (2\beta_2 cn)! (3\beta_3 cn)! (2\beta_1 cn)! (\beta_2 cn)!$;
- b) total number of configurations: the occurrences of literals can be in any order: $(3cn)!$ permutations are possible.

We denote by \mathcal{P} the set of all families ζ of non-negative numbers

$$\left(\left(\pi_{i,j,k,l,m,v} \right)_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} , (H_1, H_2, H_3), (\beta_1, \beta_2, \beta_3) \right) \quad (6.14)$$

satisfying the above constraints; note that \mathcal{P} is convex (by linearity of constraints). We denote by $\mathcal{I}(n)$ the intersection of \mathcal{P} with the multiples of $\frac{1}{n}$; we get the following expression of the first moment: $EX = \sum_{\zeta \in \mathcal{I}(n)} T(n)$ where

$$\begin{aligned} T(n) &= 2^{\tau n} \binom{Hn}{H_1 n, H_2 n, H_3 n} \binom{cn}{\beta_1 cn, \beta_2 cn, \beta_3 cn} 3^{(\beta_1 + \beta_2)cn} \\ &\quad \frac{(\beta_1 cn)! (2\beta_2 cn)! (3\beta_3 cn)! (2\beta_1 cn)! (\beta_2 cn)!}{(3cn)!} \\ &\quad \cdot \prod_{(p,q) \in \mathcal{L}} \binom{d_{p,q} n}{\delta_{p,q} n} \prod_{(p,q) \in \mathcal{L}} \left(\dots \pi_{i,j,k,l,m,1} n \dots \right)_{(i,j,k,l,m) \in Q_{p,q}} \\ &\quad \cdot \prod_{(p,q) \in \mathcal{L}} \left(\binom{d_{p,q} - \delta_{p,q}}{\dots \pi_{i,j,k,l,m,0} n \dots} \right)_{(i,j,k,l,m) \in Q_{p,q}} \\ &\quad \cdot \prod_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} \left(\omega_{i,j,k,l,m,v} \binom{i+j+k}{i,j,k} \binom{l+m}{l,m} \right)^{\pi_{i,j,k,l,m,v} n}. \end{aligned} \quad (6.15)$$

We get rid of all factorials thanks to the following Stirling's inequalities due to Batir - 2008 [Bat08]: $\left(\frac{k}{e}\right)^k \sqrt{2\pi \left(k + \frac{1}{6}\right)} < k! < \left(\frac{k}{e}\right)^k \sqrt{2\pi \left(k + \left(\frac{e^2}{2\pi} - 1\right)\right)}$.

The boundedness of the set \mathcal{L} of light variables (and thus the boundedness of the sets $Q_{p,q}$) allows to write that $T(n) \leq \text{poly}_1(n) F^n$ where

$$\begin{aligned} F &= 2^\tau \frac{H^H}{H_1^{H_1} H_2^{H_2} H_3^{H_3}} \left(\frac{1}{3} (2\beta_1)^{\beta_1} (2\beta_2)^{\beta_2} (3\beta_3)^{\beta_3} \right)^{2c} \\ &\quad \prod_{(p,q) \in \mathcal{L}} d_{p,q}^{d_{p,q}} \prod_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} \left(\omega_{i,j,k,l,m,v} \frac{\binom{i+j+k}{i,j,k} \binom{l+m}{l,m}}{\pi_{i,j,k,l,m,v}} \right)^{\pi_{i,j,k,l,m,v}}. \end{aligned} \quad (6.16)$$

Once again, by the lightness property, $\mathcal{I}(n)$ consists of a bounded number of variables, each of which can take at most $n+1$ values (as a multiple of $\frac{1}{n}$ ranging between 0 and 1). It follows that the size of $\mathcal{I}(n)$ is bounded by a polynomial $\text{poly}_2(n)$. And since $\mathcal{I}(n) \subseteq \mathcal{P}$, we have $EX \leq \text{poly}_2(n) \text{poly}_1(n) (\max_{\zeta \in \mathcal{P}} F)^n$.

6.3.3 Maximization of $\ln F$

This is the technical part of our work. We mainly use the same techniques as [DKMPG09].

1. In order to maximize $\ln F$ under our constraints, we use the standard Lagrange multipliers technique. This is appendix A.1. The following equations come from the Lagrange derivations and are important for our study:

$$\pi_{i,j,k,l,m,1} = \omega_{i,j,k,l,m,1} \binom{i+j+k}{i,j,k} \binom{l+m}{l,m} r_{i+j+k,l+m} x_1^{2i} x_2^j y_1^l y_2^{2m} \quad (6.17)$$

$$\pi_{i,j,k,l,m,0} = \omega_{i,j,k,l,m,0} \binom{i+j+k}{i,j,k} \binom{l+m}{l,m} r_{l+m,i+j+k} x_1^{2i} x_2^j y_1^l y_2^{2m} \quad (6.18)$$

x_1, x_2, y_1 and y_2 are Lagrange multipliers, that is positive numbers; moreover $r_{p,q}$ is defined as follows:

$$r_{p,q} = \frac{d_{p,q}}{A_{p,q}} ; \quad (6.19)$$

$$\begin{aligned} A_{p,q} = & \sum_{(i,j,k,l,m) \in Q_{p,q}} \omega_{i,j,k,l,m,1} \binom{p}{i,j,k} \binom{q}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \\ & + \sum_{(i,j,k,l,m) \in Q_{q,p}} \omega_{i,j,k,l,m,0} \binom{q}{i,j,k} \binom{p}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} . \end{aligned} \quad (6.20)$$

2. In order to justify the use of this technique we must show that the function $\ln F$ does not maximize on the boundary of the polytope of constraints; to do so we show that starting at a boundary point there is always a “good” direction inside the polytope which makes $\ln F$ greater. This is appendix A.2.
3. Finally we must ensure that the solution we found by the Lagrange multiplier technique is indeed a global maximum; to do so we make a sweep over different values of the parameters β_t ; indeed when these β_t are fixed the function $\ln F$ is strictly concave relative to the remaining variables, thus easier to maximize. This is appendix A.3.

6.3.4 Minimization of Global Weight

Let us see how one can minimize F (or equivalently $\ln F$) by a good choice of the weights. The following reasoning is not rigorous; we only aim at giving some hints to explain the choice of the weights we made in section 6.2.

Remember that F is given by equation 6.16. We want to minimize $\ln F$ by tuning the weights $\omega_{0,j,k,l,m,v}$, so we are going to differentiate $\ln F$ with respect to an individual $\omega_{0,j,k,l,m,1}$. Of course due to the constraints every variable depend on $\omega_{0,j,k,l,m,1}$ in the process of maximizing $\ln F$ under these constraints. But we consider that the variations on all variables are negligible except for $\pi_{0,j,k,l,m,1}$ (because of equation A.10) and $\pi_{0,l,m,j,k,0}$ (because of equations A.11 and 6.1), so we can write:

$$\frac{\partial (\ln F)}{\partial \omega_{0,j,k,l,m,1}} \simeq \frac{\partial (\ln F)}{\partial \pi_{0,j,k,l,m,1}} \frac{\partial \pi_{0,j,k,l,m,1}}{\partial \omega_{0,j,k,l,m,1}} + \frac{\partial (\ln F)}{\partial \pi_{0,l,m,j,k,0}} \frac{\partial \pi_{0,l,m,j,k,0}}{\partial \omega_{0,j,k,l,m,1}} . \quad (6.21)$$

Using equations A.10, A.11 and 6.1 we find that:

$$\begin{aligned} \frac{\partial(\ln F)}{\partial\omega_{0,j,k,l,m,1}} &\simeq -\binom{j+k}{j,k}\binom{l+m}{l,m}r_{j+k,l+m}x_2^jy_1^ly_2^{2m}\ln\left(r_{j+k,l+m}x_2^jy_1^ly_2^{2m}\right) \\ &\quad +\binom{j+k}{j,k}\binom{l+m}{l,m}r_{j+k,l+m}x_2^ly_1^jy_2^{2k}\ln\left(r_{j+k,l+m}x_2^ly_1^jy_2^{2k}\right). \end{aligned} \quad (6.22)$$

Now due to equations A.13 and A.12 and numerical experiments we make the following approximations:

$r_{j+k,l+m}x_2^jy_1^ly_2^{2m} \ll 1$ and $r_{j+k,l+m}x_2^ly_1^jy_2^{2k} \ll 1$. As the function $x \mapsto x \ln(ax)$ is strictly decreasing between 0 and $\frac{1}{ea}$, we can infer the following property: $\frac{\partial(\ln F)}{\partial\omega_{0,j,k,l,m,1}} > 0$ iff $x_2^ly_1^jy_2^{2k} < x_2^jy_1^ly_2^{2m}$, i.e. $\left(\frac{y_1}{x_2}\right)^{j-l}(y_2^2)^{k-m} < 1$.

Now let us consider we are at the minimum point of $\ln F$. If $\frac{\partial \ln(F)}{\partial \omega_{0,j,k,l,m,1}} \neq 0$, then $\omega_{0,j,k,l,m,1}$ must be at the boundary, i.e. 0 or 1.

$\frac{\partial(\ln F)}{\partial\omega_{0,j,k,l,m,1}} > 0$ iff $\alpha_1\rho_{j,l} + \alpha_3\rho_{k,m} < 0$, where $\alpha_1 = \ln \frac{y_1}{x_2}$ and $\alpha_3 = \ln(y_2^2)$. Thus:

1. if $\alpha_1\rho_{j,l} + \alpha_3\rho_{k,m} < 0$, then $\omega_{0,j,k,l,m,1} = 0$;
2. if $\alpha_1\rho_{j,l} + \alpha_3\rho_{k,m} > 0$, then $\omega_{0,j,k,l,m,1} = 1$;
3. if $\alpha_1\rho_{j,l} + \alpha_3\rho_{k,m} = 0$, nothing can be said about $\omega_{0,j,k,l,m,1}$.

What about $\omega_{0,j,k,l,m,0}$?

1. if $\alpha_1\rho_{j,l} + \alpha_3\rho_{k,m} < 0$, then $\alpha_1\rho_{l,j} + \alpha_3\rho_{m,k} > 0$, thus $\omega_{0,l,m,j,k,1} = 1$, so $\omega_{0,j,k,l,m,0} = 0$;
2. if $\alpha_1\rho_{j,l} + \alpha_3\rho_{k,m} > 0$, then by the same argument, $\omega_{0,j,k,l,m,0} = 1$;
3. if $\alpha_1\rho_{j,l} + \alpha_3\rho_{k,m} = 0$, nothing can be said about $\omega_{0,j,k,l,m,0}$.

6.4 Conclusion

We hope that the new track we opened will help gain some more insight and some more decimals in the quest of the 3-SAT threshold. In particular note that we required the relation $>$ between solutions to be circuit-free although this might not be necessary; indeed we only used the fact that this relation had at least one minimal element. The same remark holds for the constraints we put onto the weights of two neighboring solutions as introduced in equation 6.1, since this might be too strong. Thus there may be better orientations or weighting schemes than ours.

Conclusion on the First Moment Method

We saw in chapter 3 that the First Moment Method can be used either through an appropriate selection of solutions or through an appropriate weighting of solutions. Moreover we extensively used the weighting introduced by Maneva, Mossel & Wainwright [MMW07] onto valid partial assignments, without any success. Even with the improvement we gave in section 5.7, we were not able to achieve a better upper bound than the one obtained by a mere counting of NPSs (see a recap in table 6.2). Moreover we were able to prove in section 5.5 that in some particular cases weighting cannot be better than ordering. In fact we have no example where weighting is better than ordering. So we have little hope in this direction.

With lots of efforts on distributional models (chapter 6) we performed a non-uniform selection of solutions, slightly improving (4.500) on the Dubois-Boufkhad-Mandler [DBM00] upper bound of 4.506. This might thus be a more fruitful direction for future investigations: find an astute (non-uniform?) way of selecting solutions.

Table 6.2: Summary of upper bounds obtained on the threshold of standard 3-SAT by the First Moment Method in the drawing and distributional models.

		drawing	distribution
solutions	sections 3.2.1 and chapter 6	5.191	5.040
prime implicants	section 4.3.3	4.883	-
Maneva's weighted valid partial assignments	section 4.3.1	4.883	-
valid partial assignments weighted w.r.t. starrable vs. invertible	section 5.7	4.643	-
NPSs	[DB97] and chapter 6	4.643	4.552
NPSs with imbalance	[DBM00, DBM03] and chapter 6	-	4.506
our non-uniform selection of solutions	chapter 6	-	4.500
NPSs with imbalance + elimination of pure literals + typicality of clauses	[DKMPG09]	-	4.490

Part II

Second Moment

Introduction to the Second Moment Method

UST as the First Moment Method is a way to upper-bound the threshold of k -SAT, so is the Second Moment Method a way to lower-bound it. After a brief technical introduction to the Second Moment Method in section 7.1, we present in section 7.2 a survey of the early attempts to lower-bound the threshold of 3-SAT through the Second Moment Method. As in the First Moment Method, the general idea is to count special kinds of solutions. However, the selection of solutions is not the same as in the First Moment Method:

1. in the First Moment Method, we considered random variables X such that satisfiability implies $X \geq 1$ (see section 3.1); setting X to be the number of solutions yields an upper bound of 5.191 for the threshold of 3-SAT (see section 3.2.1); and we tried to select the least solutions;
2. in the Second Moment Method, we shall consider random variables X such that unsatisfiability implies $X = 0$ (see section 7.1); setting X to be the number of solutions yields a lower bound of 0 for the threshold of 3-SAT (see section 7.2.1); here the criterion to select solutions is quite different: we are interested in subsets of solutions having low variance.

In a breakthrough paper, Achlioptas & Peres - 2004 [AP04] succeeded with the Second Moment Method on k -SAT, establishing a lower bound of 2.68 for 3-SAT and an asymptotically tight lower bound of $2^k \ln 2 - O(k)$ for k -SAT. It turns out that the currently best lower bound of the 3-SAT threshold (3.52) was obtained by another way: analyzing algorithms finding solutions with high probability, cf. Kaporis, Kirousis & Lalas [KKL06] and Hajiaghayi & Sorkin [HS03].

The purpose of the next chapters is to make the most of the Second Moment Method on k -SAT. To do so we take a different approach from Achlioptas & Peres'. In our framework we select solutions according to the fraction of variables assigned 1 and the fractions of the different types of clauses (i.e. the number of true literals occurrences in the clauses). This framework is general enough to include boolean solutions, implicants etc. However, tuning our parameters the best way we could, we got numerical evidence that we could not obtain better lower bounds than 2.83.

The stumbling block we recurrently encountered is what we call the independence point. It corresponds to the couples of independent solutions in the subset of selected solutions. Even though solutions are independent, the proportion of literals occurrences having a certain truth value may not be independent between solutions. We got numerical evidence that the Second Moment Method does not work if truth values of literals occurrences are not independent. On the other hand, when they are independent, we give a necessary condition for the Second Moment Method to work, taking into account just the exponential equivalent of the second moment at this point. This condition tells us that we must select solutions having equal true and false surfaces (the surface is just the total number of literals occurrences), which is very artificial with respect to what we can observe with SATLab. Using this condition, we could make the Second Moment Method work numerically; however, since the lower bounds we get (2.83) are far below the currently best lower bound (3.52), we do not give a rigorous (and tedious) proof of our lower bounds (to do so, the exponential equivalent would not be enough, and calculations would become quite involved).

The very restrictive conditions we encountered to make the Second Moment Method work may be due to some weaknesses of our framework. We do not claim that the Second Moment Method is doomed to perpetual failure on k -SAT. We only hope to shed a small ray of light onto it. This work is very fresh, still in progress, and has not been published.

7.1 How the Second Moment Method Is Supposed to Work

Let us recall how the Second Moment Method is supposed to work: given an event A , we want to show that $\Pr(A)$ tends to 1 but we don't have access to $\Pr(A)$. Instead we use the first and the second moments of a non-negative random variable X such that $\Pr(A) \geq \Pr(X > 0)$, i.e. X must be 0 when A does not hold. For our problem 3-SAT, A is the event "a formula is satisfiable". The simplest choice for X is of course the number of solutions.

1. The first thing to notice is that if we show that $\Pr(A)$ is lower-bounded by a positive constant, then it tends to 1. Why? Because Friedgut & Bourgain's theorem [FB99] established a sharp threshold for random k -SAT;
2. In order to prove that $\Pr(X > 0)$ is bounded away from zero, we use the following classical identity:

$$\Pr(X > 0) \geq \frac{(EX)^2}{EX^2} . \quad (7.1)$$

To prove it, use the fact that $EX = E(X\mathbf{1}_{X>0})$ since $X \geq 0$, and apply the Cauchy-Schwartz inequality to it: $(EX)^2 \leq EX^2 E\mathbf{1}_{X>0}^2$. So in particular note that

$$\frac{(EX)^2}{EX^2} \leq 1 . \quad (7.2)$$

Just as the first moment is fairly easy to compute, so is the second moment. Let X be the number of assignments having some property \mathcal{P} (\mathcal{P} might be "be a solution" or "be a black

and red polka-dot solution”):

$$\begin{aligned}
 E(X^2) &= E\left(\left(\sum_{\sigma \text{ assignment}} \mathbf{1}_{\sigma \in \mathcal{P}}\right)^2\right) \\
 &= E\left(\sum_{\sigma, \tau \text{ assignments}} \mathbf{1}_{\sigma \in \mathcal{P}} \mathbf{1}_{\tau \in \mathcal{P}}\right) \\
 &= \sum_{\sigma, \tau \text{ assignments}} E \mathbf{1}_{\sigma \in \mathcal{P} \wedge \tau \in \mathcal{P}} \\
 &= \sum_{\sigma, \tau \text{ assignments}} \Pr(\sigma \in \mathcal{P} \wedge \tau \in \mathcal{P}) .
 \end{aligned}$$

In the case of satisfiability however, we are going to see that the Second Moment Method is much more difficult to implement than the First Moment Method. The reason is that in general EX^2 tends to be exponentially greater than $(EX)^2$, so equation 7.1 just says that $\Pr(X > 0) \geq 0$, which is not very informative... Thus the challenge is to find out a set of solutions having low variance.

7.2 Use of the Second Moment Method for Lower-Bounding the Threshold of k-SAT

In this section we make a survey of different applications of the Second Moment Method to lower-bound the threshold of random k-SAT. The model considered here is uniform random drawing of k-SAT, as defined in section 1.1.3.

7.2.1 Second Moment of Solutions

Here is a perfect example of the failure of the Second Moment Method. Namely the lower bound obtained by the Second Moment Method of solutions is $c = 0$. We saw in section 3.2.1 that if X is just the number of solutions, then $EX = 2^n \left(1 - \frac{1}{2^k}\right)^{cn}$. So let us now compute the second moment.

To do so we need an extra parameter μ , representing the proportion of variables changing values between two solutions.

1. total number of couples of assignments:
 - a) choose the value of variables assigned 0 or 1 in the first assignment: 2^n ;
 - b) choose the subset of variables assigned different values in both assignments: $\binom{n}{\mu n}$;
2. probability for a couple of assignments to be a couple of solutions: as noted by Achlioptas & Peres [AP04], it is easier to compute the probability that a clause breaks two given assignments, since it is $\left(\frac{1-\mu}{2}\right)^k$. Using then the fact that $\Pr(A \cap B) = 1 - \Pr(\overline{A} \cup \overline{B}) = 1 - \Pr(\overline{A}) - \Pr(\overline{B}) + \Pr(\overline{A} \cap \overline{B})$, it follows that the probability for a clause not to break any of both assignments is $g(\mu) = 1 - \frac{2}{2^k} + \left(\frac{1-\mu}{2}\right)^k$.

Thus the second moment is:

$$EX^2 = 2^n \sum_{\substack{0 \leq \mu \leq 1 \\ \mu n \in \mathbb{N}}} \binom{n}{\mu n} g(\mu)^{cn} .$$

Let us look at the exponential equivalent of this quantity:

$$EX^2 \asymp \max_{0 \leq \mu \leq 1} \left(\frac{2}{\mu^\mu (1-\mu)^\mu} g(\mu)^c \right)^n .$$

As explained by Achlioptas & Peres [AP04], it turns out that when $\mu = \frac{1}{2}$, this is precisely $(EX)^2$. Now the function $\mu \mapsto \frac{1}{\mu^\mu (1-\mu)^{1-\mu}}$ has its maximum at $\mu = \frac{1}{2}$, whereas $g(\mu)$ is strictly decreasing over $(0, 1)$. Consequently, at any positive ratio c , the maximum of $\frac{2}{\mu^\mu (1-\mu)^\mu} g(\mu)^c$ occurs at $\mu < \frac{1}{2}$ and EX^2 is exponentially greater than $(EX)^2$. So $\frac{(EX)^2}{EX^2}$ tends to zero, and we only get that $\Pr(X > 0) \geq 0 \dots$

7.2.2 Balancing True and False Surfaces

Achlioptas & Moore - 2002 [AM02] noticed that $g(\mu)$ is locally maximal at $\mu = \frac{1}{2}$ in k -NAE-SAT because in this problem $g(\mu) = 1 - \frac{4}{2^k} + \frac{1}{2^k} \left((1-\mu)^k + \mu^k \right)$ is symmetric in μ . We recall that in NAE-SAT, when an assignment is a solution, then the opposite assignment is a solution as well. Thus this problem contains some symmetry. Using this remark, Achlioptas & Moore were able to establish a tight lower bound on the k -NAE-SAT threshold. And since a NAE-SAT solution is a solution of standard SAT, they got the following lower bound of the k -SAT threshold: $2^{k-1} \ln 2 - O(1)$.

Achlioptas & Peres - 2004 [AP04] put some weights onto the solutions of standard SAT and got a lower bound of $2^k \ln 2 - O(k)$. (thus almost matching the asymptotic upper bound of $2^k \ln 2$ we saw in section 3.2.1). The weights they put are of the form $\lambda^{\text{true surface}} \mathbf{1}_{\text{SAT}}$, where the true surface is the number of occurrences of true literals under the solution. Assignments which are not solutions must be discarded because the Second Moment Method requires to count 0 when there is no solution, as explained in section 7.1. In the particular case of 3-SAT they got a lower bound of 2.54 (and even 2.68 with a refinement).

We are going to implement the Second Moment Method without any weights; so how shall we control the balance of true and false surfaces? Our control parameters will be β_t , the fraction of clauses having t true literals. Then the true surface will be $\beta_1 + 2\beta_2 + 3\beta_3$ and the false surface will be $2\beta_1 + \beta_2$. With this parameters we are able to make the Second Moment work. However, we are not able to achieve a better lower bound than 2.833, see section 8.6.1.

Moreover our approach is quite general and enables us to make the Second Moment Method work on implicants as well.

A General Framework for the Second Moment Method on k-SAT

HERE we present a general framework for the Second Moment Method on k-SAT. Section 8.1 introduces all ingredients we need: values, signs, truth values, types of clauses and surfaces. Then in section 8.2 we give the expression of the first moment of the solutions under these settings; the expression of the second moment is given in section 8.3. Bringing together the second moment and the constraints, we use the Lagrange multipliers method in section 8.4.

One point in the space of the variables is very important in the Second Moment Method: this is what we call the independence point. It is important because it makes $\frac{EX^2}{(EX)^2} = 1$ (see conditions in theorem 8.5 of section 8.5). Thus if we want the Second Moment to work, we must be careful that this point should be stationary.

We apply this general framework to boolean solutions (section 8.6.1) and to implicants (section 8.6.2).

We discuss the relevance of the Second Moment Method for lower-bounding the k-SAT threshold in section 8.7, where we use SATLab to confront the theoretical requirements we obtained with reality.

8.1 Preliminaries

8.1.1 Values

First of all we have n variables. An assignment gives each variable a value taken from a given domain D :

- in the case of boolean satisfiability, $D = \{0, 1\}$;
- in the case of implicants, $D = \{0, 1, *\}$.

Given an assignment, for all $a \in D$, we denote by δ_a the proportion of variables assigned value a :

$$\sum_{a \in D} \delta_a = 1 . \tag{8.1}$$

Given two assignments S_1 and S_2 , for all $(a, b) \in D^2$, we denote by $\mu_{a,b}$ the proportion of variables assigned value a in S_1 and value b in S_2 :

$$\sum_{a \in D} \mu_{a,b} = \delta_b ; \quad (8.2)$$

$$\sum_{b \in D} \mu_{a,b} = \delta_a . \quad (8.3)$$

So we are going to consider some of the $\mu_{a,b}$'s as functions of the other ones, assuming that equations 8.2 and 8.3 are satisfied. We shall refer to the remaining $\mu_{a,b}$'s as a generic variable μ , cf. section 8.4.2.

8.1.2 Signs

In a k-CNF formula with cn clauses, we have kcn occurrences of variables, each having a sign $s \in S$. In the case of boolean satisfiability as well as in the case of implicants, $S = \{+, -\}$;

For all $s \in S$, we denote by ρ_s the proportion of occurrences having sign s :

$$\sum_{s \in S} \rho_s = 1 . \quad (8.4)$$

8.1.3 Truth Values

The combination of a sign and a value yields a truth value $v \in \mathcal{V}$. Here is an example of a

classical truth table with $\mathcal{V} = \{T, F, *\}$:

	+	-
0	F	T
1	T	F
*	*	*

We use the following notation, for $a \in D, s \in S$ and $v \in \mathcal{V}$:

$$\chi_{a,s,v} = \begin{cases} 1 & \text{if value } a \text{ and sign } s \text{ yield the truth value } v ; \\ 0 & \text{otherwise .} \end{cases}$$

Of course one sign and one value yield exactly one truth value:

$$\sum_{v \in \mathcal{V}} \chi_{a,s,v} = 1 .$$

So we shall denote by $a \otimes s$ the unique v such that $\chi_{a,s,v} = 1$.

Given an assignment, for all $v \in \mathcal{V}$, we denote by η_v the proportion of literals occurrences having the truth value v :

$$\eta_v = \sum_{\substack{a \in D \\ s \in S}} \chi_{a,s,v} \delta_a \rho_s .$$

Given two assignments S_1 and S_2 , for all $(v, w) \in \mathcal{V}^2$, we denote by $\varepsilon_{v,w}$ the proportion of literals occurrences having the truth value v in S_1 and the truth value w in S_2 :

$$\varepsilon_{v,w} = \sum_{\substack{(a,b) \in D^2 \\ s \in S}} \chi_{a,s,v} \chi_{b,s,w} \mu_{a,b} \rho_s .$$

8.1.4 Clauses Types

A *clause type* is an element of \mathcal{V}^k , e.g. TTF or *TF. Some types of clauses will be forbidden, such as FFF and FF*. We denote by \mathcal{T} the set of allowed types of clauses. \mathcal{T} is a subset of \mathcal{V}^k .

- in the case of boolean solutions of 3-SAT: $\mathcal{V} = \{T, F\}$ and FFF is forbidden;
- in the case of implicants of 3-SAT: $\mathcal{V} = \{T, F, *\}$ and the allowed types of clauses are those containing at least one T .

Given an assignment, for all $t \in \mathcal{T}$, we denote by β_t the proportion of clauses of type t (which is zero for all forbidden types of clauses):

$$\sum_{t \in \mathcal{T}} \beta_t = 1. \quad (8.5)$$

Given two assignments S_1 and S_2 , for all $(t, u) \in \mathcal{T}^2$, we denote by $\gamma_{t,u}$ the proportion of clauses of type t in solution S_1 and of type u in solution S_2 :

$$\sum_{t \in \mathcal{T}} \gamma_{t,u} = \beta_u; \quad (8.6)$$

$$\sum_{u \in \mathcal{T}} \gamma_{t,u} = \beta_t. \quad (8.7)$$

8.1.5 Surfaces

Given an assignment, the *surface* occupied by a truth value v is obtained by summing all occurrences of v in the different types of clauses: $\Sigma_v = \sum_{t \in \mathcal{T}} \beta_t \sum_{i=1}^k \mathbf{1}_{t_i=v}$.

Given two assignments S_1 and S_2 , the *surface* occupied by a couple of truth values (v, w) is obtained by summing all occurrences of (v, w) in the different types of clauses: $\Xi_{v,w} = \sum_{(t,u) \in \mathcal{T}^2} \gamma_{t,u} \sum_{i=1}^k \mathbf{1}_{t_i=v \wedge u_i=w}$.

Surfaces are normalized to k because the β_t 's sum up to 1:

Fact 8.1. $\sum_{v \in \mathcal{V}} \Sigma_v = k$ and $\sum_{(v,w) \in \mathcal{V}^2} \Xi_{v,w} = k$.

Proof.

$$\begin{aligned} \sum_{v \in \mathcal{V}} \Sigma_v &= \sum_{v \in \mathcal{V}} \sum_{t \in \mathcal{T}} \beta_t \sum_{i=1}^k \mathbf{1}_{t_i=v} \\ &= \sum_{t \in \mathcal{T}} \beta_t \sum_{i=1}^k \sum_{v \in \mathcal{V}} \mathbf{1}_{t_i=v} \\ &= \sum_{t \in \mathcal{T}} \beta_t \sum_{i=1}^k 1 \\ &= k \sum_{t \in \mathcal{T}} \beta_t \\ &= k \text{ by constraint 8.5.} \end{aligned}$$

The same proof works for the other sum, using constraints 8.6 and 8.5. \square

8.1.6 Symmetry of Occurrences

We say that there is *symmetry of occurrences* when for all permutation σ of \mathcal{V}^k , $\beta_t = \beta_{\sigma(t)}$ (it follows that \mathcal{T} is closed by permutation).

Fact 8.2. *Symmetry of occurrences implies that $\Sigma_v = k \sum_{t \in \mathcal{T}} \mathbf{1}_{t_1=v} \beta_t$.*

Proof. Let us call σ_i the permutation of \mathcal{V}^k swapping t_1 and t_i .

$$\begin{aligned}
 \Sigma_v &= \sum_{t \in \mathcal{T}} \beta_t \sum_{i=1}^k \mathbf{1}_{t_i=v} \\
 &= \sum_{i=1}^k \sum_{t \in \mathcal{T}} \mathbf{1}_{t_i=v} \beta_t \\
 &= \sum_{i=1}^k \sum_{t \in \mathcal{T}} \mathbf{1}_{\sigma_i(t)_1=v} \beta_{\sigma_i(t)} \text{ by symmetry of occurrences} \\
 &= \sum_{i=1}^k \sum_{\sigma_i(t) \in \mathcal{T}} \mathbf{1}_{\sigma_i(t)_1=v} \beta_{\sigma_i(t)} \text{ because } \mathcal{T} \text{ is closed by permutation} \\
 &= k \sum_{t \in \mathcal{T}} \mathbf{1}_{t_1=v} \beta_t .
 \end{aligned}$$

□

Fact 8.3. *If $\gamma_{t,u} = \beta_t \beta_u$, symmetry of occurrences implies that $k \Xi_{v,w} = \Sigma_v \Sigma_w$.*

Proof. Let us call σ_i the permutation of \mathcal{V}^k swapping t_1 and t_i .

$$\begin{aligned}
 k \Xi_{v,w} &= k \sum_{(t,u) \in \mathcal{T}^2} \gamma_{t,u} \sum_{i=1}^k \mathbf{1}_{t_i=v \wedge u_i=w} \\
 &= k \sum_{(t,u) \in \mathcal{T}^2} \beta_t \beta_u \sum_{i=1}^k \mathbf{1}_{t_i=v} \mathbf{1}_{u_i=w} \text{ by independence} \\
 &= k \sum_{i=1}^k \sum_{(\sigma_i(t), \sigma_i(u)) \in \mathcal{T}^2} \beta_{\sigma_i(t)} \beta_{\sigma_i(u)} \mathbf{1}_{\sigma_i(t)_1=v} \mathbf{1}_{\sigma_i(u)_1=w} \text{ by symmetry of occurrences} \\
 &= k^2 \sum_{(t,u) \in \mathcal{T}^2} \beta_t \beta_u \mathbf{1}_{t_1=v} \mathbf{1}_{u_1=w} \text{ because } \mathcal{T} \text{ is closed by permutation} \\
 &= \Sigma_v \Sigma_w \text{ by fact 8.2.}
 \end{aligned}$$

□

Symmetry of occurrences is quite natural and will be assumed from now on. Note that δ_a 's and β_t 's are parameters of the first moment, so they may be chosen without any restriction, except that they must sum up to 1. They are our control parameters: we can tune them as we wish in order to take into account only some solutions. However, when the set of solutions defined by δ_a 's and β_t 's is determined, we have to consider all possible couples of solutions. So the variables $\mu_{a,b}$'s and $\gamma_{t,u}$'s of the second moment may not be chosen, but result from a maximization process, as investigated in section 8.4.

8.2 Expression of the First Moment

The first moment of the number X of solutions can be split up into the following factors: total number of assignments and probability for an assignment to be a solution.

1. total number of assignments: choose subsets of variables assigned $a \in D$: $(\dots(\delta_a n)_{a \in D} \dots)$.
2. probability for an assignment to be a solution:
 - a) we give each clause an allowed type $t \in \mathcal{T}$: $(\dots(\beta_t cn)_{t \in \mathcal{T}} \dots)$.
 - b) probability for clauses to be constructed (variables + signs) according to their types:

$$\prod_{t \in \mathcal{T}} \left(\prod_{i=1}^k \eta_{t_i} \right)^{\beta_t cn}.$$

We denote by \mathcal{P} the set of all families of non-negative numbers $((\delta_a)_{a \in D}, (\beta_t)_{t \in \mathcal{T}})$ satisfying constraints 8.1 and 8.5. We denote by $\mathcal{I}(n)$ the intersection of \mathcal{P} with the multiples of $\frac{1}{n}$; we get the following expression of the first moment:

$$EX = \sum_{((\delta_a)_{a \in D}, (\beta_t)_{t \in \mathcal{T}}) \in \mathcal{I}(n)} T_1(n)$$

where

$$\begin{aligned} T_1(n) &= \left(\dots (\delta_a n)_{a \in D} \dots \right) \left(\dots (\beta_t cn)_{t \in \mathcal{T}} \dots \right) \left(\prod_{t \in \mathcal{T}} \left(\prod_{i=1}^k \eta_{t_i} \right)^{\beta_t} \right)^{cn} \\ &= \left(\dots (\delta_a n)_{a \in D} \dots \right) \left(\dots (\beta_t cn)_{t \in \mathcal{T}} \dots \right) \left(\prod_{t \in \mathcal{T}} \prod_{i=1}^k \prod_{v \in \mathcal{V}} \eta_v^{\mathbf{1}_{t_i=v} \beta_t} \right)^{cn} \\ &= \left(\dots (\delta_a n)_{a \in D} \dots \right) \left(\dots (\beta_t cn)_{t \in \mathcal{T}} \dots \right) \left(\prod_{v \in \mathcal{V}} \eta_v^{\sum_{t \in \mathcal{T}} \sum_{i=1}^k \mathbf{1}_{t_i=v} \beta_t} \right)^{cn} \\ &= \left(\dots (\delta_a n)_{a \in D} \dots \right) \left(\dots (\beta_t cn)_{t \in \mathcal{T}} \dots \right) \left(\prod_{v \in \mathcal{V}} \eta_v^{\sum_v} \right)^{cn}. \end{aligned}$$

From equation B.2, the exponential equivalent of $T_1(n)$ is T_1^n , with the following two equivalent forms:

$$\begin{aligned} T_1 &= \frac{1}{\prod_{a \in D} \delta_a^{\delta_a}} \left(\prod_{t \in \mathcal{T}} \left(\frac{\prod_{i=1}^k \eta_{t_i}}{\beta_t} \right)^{\beta_t} \right)^c \\ &= \frac{1}{\prod_{a \in D} \delta_a^{\delta_a}} \left(\frac{\prod_{v \in \mathcal{V}} \eta_v^{\sum_v}}{\prod_{t \in \mathcal{T}} \beta_t^{\beta_t}} \right)^c. \end{aligned}$$

8.3 Expression of the Second Moment

The second moment of X can be split up into the following factors: total number of couples of assignments and probability for a couple of assignments to be a couple of solutions.

1. total number of couples of assignments: $(\dots(\mu_{a,b} n)_{(a,b) \in D^2} \dots)$;
2. probability for a couple of assignments to be a couple of solutions:

- a) we give each clause an allowed type $t \in \mathcal{T}$ in solution S_1 and another $u \in \mathcal{T}$ in solution S_2 : $(\dots(\gamma_{t,u} cn)_{(t,u) \in \mathcal{T}^2} \dots)$;
- b) probability for clauses to be constructed (variables + signs) according to their types: $\prod_{(t,u) \in \mathcal{T}^2} \left(\prod_{i=1}^k \varepsilon_{t_i, u_i} \right)^{\gamma_{t,u} cn}$.

We denote by \mathcal{P}_2 the set of all families of non-negative numbers $((\mu_{a,b})_{(a,b) \in D^2}, (\gamma_{t,u})_{(t,u) \in \mathcal{T}^2})$ satisfying constraints 8.2, 8.3, 8.6 and 8.7. We denote by $\mathcal{I}_2(n)$ the intersection of \mathcal{P}_2 with the multiples of $\frac{1}{n}$; we get the following expression of the second moment:

$$EX^2 = \sum_{((\mu_{a,b})_{(a,b) \in D^2}, (\gamma_{t,u})_{(t,u) \in \mathcal{T}^2}) \in \mathcal{I}_2(n)} T_2(n)$$

where

$$\begin{aligned} & T_2(n) \\ = & \left(\dots (\mu_{a,b} n)_{(a,b) \in D^2} \dots \right) \left(\dots (\gamma_{t,u} cn)_{(t,u) \in \mathcal{T}^2} \dots \right) \left(\prod_{(t,u) \in \mathcal{T}^2} \left(\prod_{i=1}^k \varepsilon_{t_i, u_i} \right)^{\gamma_{t,u}} \right)^{cn} \\ = & \left(\dots (\mu_{a,b} n)_{(a,b) \in D^2} \dots \right) \left(\dots (\gamma_{t,u} cn)_{(t,u) \in \mathcal{T}^2} \dots \right) \left(\prod_{(t,u) \in \mathcal{T}^2} \prod_{i=1}^k \prod_{(v,w) \in \mathcal{V}^2} \varepsilon_{v,w}^{\mathbf{1}_{t_i=v \wedge u_i=w}} \gamma_{t,u} \right)^{cn} \\ = & \left(\dots (\mu_{a,b} n)_{(a,b) \in D^2} \dots \right) \left(\dots (\gamma_{t,u} cn)_{(t,u) \in \mathcal{T}^2} \dots \right) \left(\prod_{(v,w) \in \mathcal{V}^2} \varepsilon_{v,w}^{\sum_{(t,u) \in \mathcal{T}^2} \sum_{i=1}^k \mathbf{1}_{t_i=v \wedge u_i=w} \gamma_{t,u}} \right)^{cn} \\ = & \left(\dots (\mu_{a,b} n)_{(a,b) \in D^2} \dots \right) \left(\dots (\gamma_{t,u} cn)_{(t,u) \in \mathcal{T}^2} \dots \right) \left(\prod_{(v,w) \in \mathcal{V}^2} \varepsilon_{v,w}^{\Xi_{v,w}} \right)^{cn}. \end{aligned}$$

From equation B.2, the exponential equivalent of $T_2(n)$ is T_2^n with the following two equivalent forms:

$$\begin{aligned} T_2 &= \frac{1}{\prod_{(a,b) \in D^2} \mu_{a,b}^{\mu_{a,b}}} \left(\prod_{(t,u) \in \mathcal{T}^2} \left(\frac{\prod_{i=1}^k \varepsilon_{t_i, u_i}}{\gamma_{t,u}} \right)^{\gamma_{t,u}} \right)^c \\ &= \frac{1}{\prod_{(a,b) \in D^2} \mu_{a,b}^{\mu_{a,b}}} \left(\frac{\prod_{(v,w) \in \mathcal{V}^2} \varepsilon_{v,w}^{\Xi_{v,w}}}{\prod_{(t,u) \in \mathcal{T}^2} \gamma_{t,u}^{\gamma_{t,u}}} \right)^c. \end{aligned}$$

8.4 Expression of the Lagrangian

When the parameters of the first moment (i.e. $(\delta_a)_{a \in D}, (\beta_t)_{t \in \mathcal{T}}$) are chosen, T_2 must be maximized under constraints 8.2, 8.3, 8.6 and 8.7. That leads us to use the Lagrange multipliers method.

As explained in section 8.1.1, we are going to consider some of the $\mu_{a,b}$'s as functions of the other ones, assuming that equations 8.2 and 8.3 are satisfied. We shall refer to the remaining $\mu_{a,b}$'s as a generic variable μ . So we define the following Lagrangian:

$$\begin{aligned} \Lambda = & - \sum_{(a,b) \in D^2} \mu_{a,b} \ln \frac{\mu_{a,b}}{e} - c \sum_{(t,u) \in \mathcal{T}^2} \gamma_{t,u} \ln \frac{\gamma_{t,u}}{e} + c \sum_{(v,w) \in \mathcal{V}^2} \Xi_{v,w} \ln \varepsilon_{v,w} \\ & + c \sum_{t \in \mathcal{T}} (\ln f_t) \left(\sum_{u \in \mathcal{T}} \gamma_{t,u} - \beta_t \right) + c \sum_{u \in \mathcal{T}} (\ln g_u) \left(\sum_{t \in \mathcal{T}} \gamma_{t,u} - \beta_u \right) . \end{aligned}$$

8.4.1 Derivative with respect to $\gamma_{t,u}$

$$\frac{\partial \Lambda}{\partial \gamma_{t,u}} = -c \ln \gamma_{t,u} + c \sum_{i=1}^k \ln \varepsilon_{t_i, u_i} + c \ln f_t + c \ln g_u .$$

Canceling out this derivative yields:

$$\gamma_{t,u} = f_t g_u \prod_{i=1}^k \varepsilon_{t_i, u_i} . \quad (8.8)$$

8.4.2 Derivative with respect to μ

$$\begin{aligned} \frac{\partial \Lambda}{\partial \mu} &= \sum_{(a,b) \in D^2} \frac{\partial \Lambda}{\partial \mu_{a,b}} \frac{\partial \mu_{a,b}}{\partial \mu} \\ &= \sum_{(a,b) \in D^2} \frac{\partial \mu_{a,b}}{\partial \mu} \left(-\ln \mu_{a,b} + c \sum_{(v,w) \in \mathcal{V}^2} \frac{\partial \varepsilon_{v,w}}{\partial \mu_{a,b}} \frac{\Xi_{v,w}}{\varepsilon_{v,w}} \right) \\ &= - \sum_{(a,b) \in D^2} \frac{\partial \mu_{a,b}}{\partial \mu} \ln \mu_{a,b} + c \sum_{(a,b) \in D^2} \frac{\partial \mu_{a,b}}{\partial \mu} \sum_{(v,w) \in \mathcal{V}^2} \frac{\Xi_{v,w}}{\varepsilon_{v,w}} \sum_{s \in S} \chi_{a,s,v} \chi_{b,s,w} \rho_s \\ &= - \sum_{(a,b) \in D^2} \frac{\partial \mu_{a,b}}{\partial \mu} \ln \mu_{a,b} + c \sum_{(a,b) \in D^2} \frac{\partial \mu_{a,b}}{\partial \mu} \sum_{s \in S} \rho_s \frac{\Xi_{a \otimes s, b \otimes s}}{\varepsilon_{a \otimes s, b \otimes s}} . \end{aligned} \quad (8.9)$$

Canceling out this derivative is somewhat tricky in general, so we are going to focus on some simple particular cases and otherwise end up calculations numerically with Mathematica...

So let us consider first a particularly simplifying case, i.e. when $\varepsilon_{v,w} = \eta_v \eta_w$.

8.5 Independence Point - Discussion about $\varepsilon_{v,w}$

We define the independence point in the polytope \mathcal{P}_2 to be the point where $\mu_{a,b} = \delta_a \delta_b$ and $\gamma_{t,u} = \beta_t \beta_u$. This point is of major interest because it make $\frac{T_2}{T_1^2} = 1$ if $\varepsilon_{v,w} = \eta_v \eta_w$ (see theorem 8.5). More surprisingly, it turns out that there seems to be a dichotomy in the success / failure of the Second Moment Method, regarding $\varepsilon_{v,w}$ at the independence point:

- if $\varepsilon_{v,w} = \eta_v \eta_w$ then we are able to find a necessary and sufficient condition on the first moment parameters for the Second Moment Method to give a non trivial lower bound in all models we considered:
 - boolean solutions, see section 8.6.1;
 - implicants, see section 8.6.2;
 - distributional model, see chapter 9;
- otherwise numerical calculations give us empirical evidence that the Second Moment Method fails to give any non trivial lower bound to the threshold; indeed even when we have almost this identity, the ratio $\frac{T_2}{T_1^2}$ is strictly greater than 1 for any positive ratio c (see figures 8.1 and 8.2 at the end of section 8.6.1 and figure 9.1 in chapter 9).

Conjecture 8.4. *The Second Moment Method works only if $\varepsilon_{v,w} = \eta_v \eta_w$.*

This conjecture echoes the following theorem.

Theorem 8.5. *At the independence point (i.e. $\mu_{a,b} = \delta_a \delta_b$ and $\gamma_{t,u} = \beta_t \beta_u$), if $\varepsilon_{v,w} = \eta_v \eta_w$, then $\frac{T_2}{T_1^2} = 1$.*

Proof. Let us recall that

$$\frac{T_2}{T_1^2} = \frac{\left(\prod_{a \in D} \delta_a^{\delta_a}\right)^2}{\prod_{(a,b) \in D^2} \mu_{a,b}^{\mu_{a,b}}} \left(\left(\prod_{t \in \mathcal{T}} \left(\frac{\beta_t}{\prod_{i=1}^k \eta_{t_i}} \right)^{\beta_t} \right)^2 \prod_{(t,u) \in \mathcal{T}^2} \left(\frac{\prod_{i=1}^k \varepsilon_{t_i, u_i}}{\gamma_{t,u}} \right)^{\gamma_{t,u}} \right)^c.$$

So at the independence point:

$$\begin{aligned} \frac{T_2}{T_1^2} &= \frac{\left(\prod_{a \in D} \delta_a^{\delta_a}\right) \left(\prod_{b \in D} \delta_b^{\delta_b}\right)}{\prod_{(a,b) \in D^2} (\delta_a \delta_b)^{\delta_a \delta_b}} \\ &\cdot \left(\frac{\left(\prod_{t \in \mathcal{T}} \beta_t^{\beta_t}\right) \left(\prod_{u \in \mathcal{T}} \beta_u^{\beta_u}\right)}{\prod_{(t,u) \in \mathcal{T}^2} (\beta_t \beta_u)^{\beta_t \beta_u}} \cdot \frac{\prod_{(t,u) \in \mathcal{T}^2} \left(\left(\prod_{i=1}^k \eta_{t_i}\right) \left(\prod_{i=1}^k \eta_{u_i}\right) \right)^{\beta_t \beta_u}}{\left(\prod_{t \in \mathcal{T}} \left(\prod_{i=1}^k \eta_{t_i}\right)^{\beta_t}\right) \left(\prod_{u \in \mathcal{T}} \left(\prod_{i=1}^k \eta_{u_i}\right)^{\beta_u}\right)} \right)^c. \end{aligned}$$

To show that this quantity is indeed 1, we shall use the following fact:

Fact 8.6. *Let \mathcal{X} and \mathcal{Y} be two finite sets. We assume that $\sum_{x \in \mathcal{X}} e_x = \sum_{y \in \mathcal{Y}} f_y = 1$.*

Then $\prod_{(x,y) \in \mathcal{X} \times \mathcal{Y}} (b_x c_y)^{e_x f_y} = \left(\prod_{x \in \mathcal{X}} b_x^{e_x}\right) \left(\prod_{y \in \mathcal{Y}} c_y^{f_y}\right)$.

Proof.

$$\begin{aligned} \prod_{(x,y) \in \mathcal{X} \times \mathcal{Y}} (b_x c_y)^{e_x f_y} &= \prod_{(x,y) \in \mathcal{X} \times \mathcal{Y}} b_x^{e_x f_y} \prod_{(x,y) \in \mathcal{X} \times \mathcal{Y}} c_y^{e_x f_y} \\ &= \prod_{x \in \mathcal{X}} b_x^{e_x \sum_{y \in \mathcal{Y}} f_y} \prod_{y \in \mathcal{Y}} c_y^{f_y \sum_{x \in \mathcal{X}} e_x} \\ &= \prod_{x \in \mathcal{X}} b_x^{e_x} \prod_{y \in \mathcal{Y}} c_y^{f_y}. \end{aligned}$$

□

Now to prove that the previous ratio $\frac{T_2}{T_1^2}$ is 1, it suffices to apply fact 8.6 3 times:

with $\mathcal{X} = \mathcal{Y} = D$, $b_x = e_x = \delta_a$ and $c_y = f_y = \delta_b$, which is possible thanks to equation 8.1;

with $\mathcal{X} = \mathcal{Y} = \mathcal{T}$, $b_x = e_x = \beta_t$ and $c_y = f_y = \beta_u$, which is possible thanks to equation 8.5;

with $\mathcal{X} = \mathcal{Y} = \mathcal{T}$, $b_x = \prod_{i=1}^k \eta_{t_i}$, $e_x = \beta_t$, $c_y = \prod_{i=1}^k \eta_{u_i}$ and $f_y = \beta_u$, which is possible again thanks to equation 8.5. \square

Moreover, it turns out that the independence point satisfies constraints 8.2, 8.3, 8.6 and 8.7, thus T_2 must be stationary at the independence point if we want the Second Moment Method to work (because $\frac{T_2}{T_1^2}$ must not exceed 1 if we want to avoid the pitfall we encountered in section 7.2.1). Thus we have the following necessary condition to make the Second Moment Method work:

$$\frac{\partial \Lambda}{\partial \mu} = 0 \text{ at the independence point.} \quad (8.10)$$

Remark 8.7. We show rigorously only the fact that the above conditions ($\varepsilon_{v,w} = \eta_v \eta_w$ and equation 8.10) are necessary to make the Second Moment Method work, but not that they are sufficient. This would require to handle the polynomial residues of the multinomials, and the complete expressions of EX^2 and $(EX)^2$. Since we show only negative results (i.e. bad lower bounds), this tricky part is omitted.

8.6 Applications

8.6.1 Boolean Solutions

8.6.1.1 Preliminaries

1. the domain of values is $D = \{0, 1\}$; given a solution, we call δ the fraction of variables assigned 1; constraints 8.2 and 8.3 become:

$$\begin{aligned} \delta &= \mu_{1,1} + \mu_{1,0} \\ \delta &= \mu_{1,1} + \mu_{0,1} \\ 1 - \delta &= \mu_{0,0} + \mu_{1,0} \\ 1 - \delta &= \mu_{0,0} + \mu_{0,1} \end{aligned}$$

so, if we define $\mu = \mu_{0,1}$, we have

$$\begin{aligned} \mu_{1,1} &= \delta - \mu \\ \mu_{0,0} &= 1 - \delta - \mu \\ \mu_{1,0} &= \mu \\ \mu_{0,1} &= \mu \end{aligned}$$

2. the set of signs is $S = \{+, -\}$; we call ρ the fraction of positive occurrences;

3. the set of truth values is $\mathcal{V} = \{T, F\}$ and the truth table is:

	+	-
0	F	T
1	T	F

; so

$$\begin{aligned}\eta_T &= \rho\delta + (1-\rho)(1-\delta) \\ \eta_F &= (1-\rho)\delta + \rho(1-\delta) \\ \varepsilon_{T,F} = \varepsilon_{F,T} &= \mu\end{aligned}\tag{8.11}$$

$$\varepsilon_{T,T} = \eta_T - \mu\tag{8.12}$$

$$\varepsilon_{F,F} = \eta_F - \mu\tag{8.13}$$

4. the set of allowed types of clauses is $\mathcal{T} = D^k \setminus \{F^k\}$.

8.6.1.2 Condition for $\varepsilon_{v,w} = \eta_v\eta_w$ at the Independence Point

The first thing to notice is that if $\varepsilon_{T,F} = \eta_T\eta_F$, then the three other identities follow, because $\varepsilon_{T,F} + \varepsilon_{T,T} = \eta_T$ etc.

At the independence point, we have $\mu = \delta(1-\delta)$. Thus

$$\begin{aligned}\varepsilon_{T,F} - \eta_T\eta_F &= \delta(1-\delta) - (\rho\delta + (1-\rho)(1-\delta))((1-\rho)\delta + \rho(1-\delta)) \\ &= \delta(1-\delta) - \rho(1-\rho)(\delta^2 + (1-\delta)^2) - (\rho^2 + (1-\rho)^2)\delta(1-\delta) \\ &= 2\rho(1-\rho)\delta(1-\delta) - \rho(1-\rho)(\delta^2 + (1-\delta)^2) \\ &= -\rho(1-\rho)(2\delta - 1)^2.\end{aligned}$$

Consequently, $\varepsilon_{T,F} \leq \eta_T\eta_F$, with equality iff $\rho \in \{0, 1\}$ or $\delta = \frac{1}{2}$.

We discard the particular case of $\rho \in \{0, 1\}$ (which corresponds to monotone k-SAT, always trivially satisfiable). It turns out that as soon as $\delta \neq \frac{1}{2}$, we could not make the Second Moment Method work: our numerical attempts revealed that the ratio $\frac{T_2}{T_1^2}$ is strictly greater than 1 for any positive ratio c .

On the other hand, when $\delta = \frac{1}{2}$ we could make the Second Moment Method work, as follows.

8.6.1.3 Condition for the Second Moment Method to Work at $\delta = \frac{1}{2}$

As mentioned in section 8.5, stationarity of the independence point implies that $\frac{\partial \Lambda}{\partial \mu} = 0$. Using equation 8.9:

$$\begin{aligned}\frac{\partial \Lambda}{\partial \mu} &= - \sum_{(a,b) \in D^2} \frac{\partial \mu_{a,b}}{\partial \mu} \ln \mu_{a,b} + c \sum_{(a,b) \in D^2} \frac{\partial \mu_{a,b}}{\partial \mu} \sum_{s \in S} \rho_s \frac{\Xi_{a \otimes s, b \otimes s}}{\varepsilon_{a \otimes s, b \otimes s}} \\ &= \ln \frac{\mu_{1,1}\mu_{0,0}}{\mu_{1,0}\mu_{0,1}} + c \left((\rho + (1-\rho)) \left(\frac{\Xi_{T,F}}{\varepsilon_{T,F}} + \frac{\Xi_{F,T}}{\varepsilon_{F,T}} - \frac{\Xi_{T,T}}{\varepsilon_{T,T}} - \frac{\Xi_{F,F}}{\varepsilon_{F,F}} \right) \right) \\ &= \ln \frac{\mu_{1,1}\mu_{0,0}}{\mu_{1,0}\mu_{0,1}} + c \left(\frac{\Xi_{T,F}}{\varepsilon_{T,F}} + \frac{\Xi_{F,T}}{\varepsilon_{F,T}} - \frac{\Xi_{T,T}}{\varepsilon_{T,T}} - \frac{\Xi_{F,F}}{\varepsilon_{F,F}} \right).\end{aligned}$$

At independence $\frac{\mu_{1,1}\mu_{0,0}}{\mu_{1,0}\mu_{0,1}} = \frac{\delta_1^2\delta_0^2}{\delta_1\delta_0\delta_0\delta_1} = 1$; moreover, assuming symmetry of occurrences, we may use independence of surfaces (cf. fact 8.3):

$$\begin{aligned}\frac{\partial\Lambda}{\partial\mu} &= c \left(-\frac{\Sigma_F^2}{\eta_F^2} - \frac{\Sigma_T^2}{\eta_T^2} + 2\frac{\Sigma_F\Sigma_T}{\eta_F\eta_T} \right) \\ &= c \left(\frac{\Sigma_T}{\eta_T} - \frac{\Sigma_F}{\eta_F} \right)^2.\end{aligned}$$

Canceling out this derivative yields:

$$\frac{\Sigma_T}{\eta_T} = \frac{\Sigma_F}{\eta_F}.$$

Since we assume $\delta = \frac{1}{2}$, we have $\eta_T = \eta_F = \frac{1}{2}$, thus $\Sigma_T = \Sigma_F = \frac{k}{2}$.

It turns out that this condition is sufficient to make the Second Moment Method work, and numerically we found a critical ratio $c = 2.833$ for $\beta_{TFE} = \beta_{FTF} = \beta_{FFT} = 0.197633$, $\beta_{TTF} = \beta_{FTT} = \beta_{FTT} = 0.104733$ and $\beta_{TTT} = 0.0929$. It is noticeable that this critical ratio is the same for any value of ρ ; this comes from the fact that laying down $\eta_T = \eta_F = \frac{1}{2}$, equations 8.11, 8.12 and 8.13 imply that equation 9.8 has no dependence in ρ .

The First Moment Method applied with these settings (i.e. $\delta = \frac{1}{2}$) yields a critical ratio of 3.783 when $\beta_{TFE} = 0.191$, which means that such balanced solutions disappear far below the conjectured threshold ratio of 4.25. Thus we would like to evade the $\delta = \frac{1}{2}$ condition.

Moreover SATLab enables us to see that real solutions do not have $\delta = \frac{1}{2}$, see our discussion in section 8.7.1.

8.6.1.4 Attempts to evade the $\delta = \frac{1}{2}$ condition

We plot $\ln \ln \frac{F_2}{F_1}$ for different values of δ and ρ , at a point satisfying our constraints 8.6 and 8.7. The expected value is $-\infty$ iff $\frac{F_2}{F_1} = 1$. We set the ratio $c = 0.1$ (to be compared with 2.833, where $\delta = \frac{1}{2}$ works).

1. setting $\Sigma_T = \Sigma_F = \frac{k}{2}$: only $\delta = \frac{1}{2}$ seems to make $\frac{F_2}{F_1} = 1$, cf. figure 8.1;
2. setting $\Sigma_T = k\eta_T$ and $\Sigma_F = k\eta_F$: once more, only $\delta = \frac{1}{2}$ seems to make $\frac{F_2}{F_1} = 1$, cf. figure 8.2.

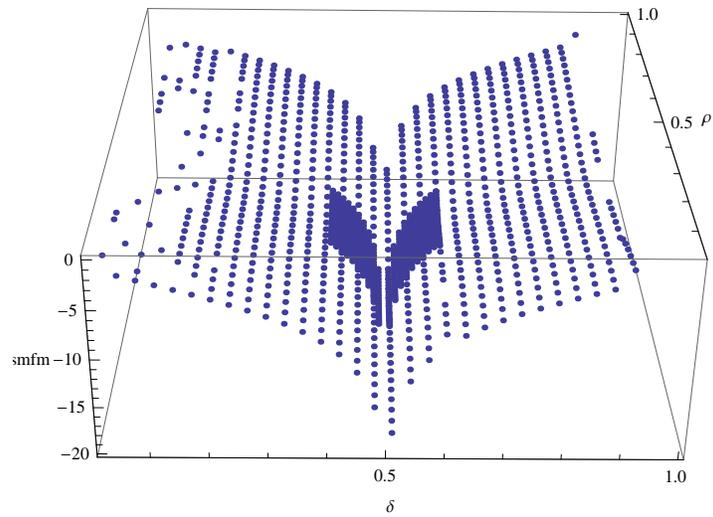


Figure 8.1: $\Sigma_T = \Sigma_F = \frac{k}{2}$: $-\infty$ is obtained only when $\delta = \frac{1}{2}$.

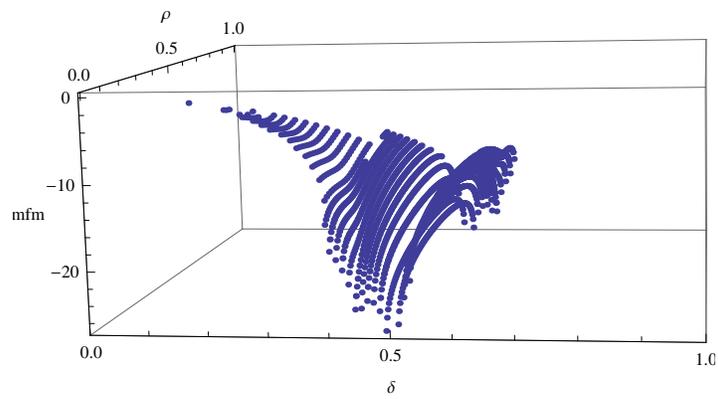


Figure 8.2: $\Sigma_T = k\eta_T$ and $\Sigma_F = k\eta_F$: $-\infty$ is obtained only when $\delta = \frac{1}{2}$.

8.6.2 Implicants

We recall (see section 3.2.2) that an *implicant* is a partial assignment such that every assignment of the non-assigned variables will yield a solution. We represent the non-assigned value of variables by a *. We performed the calculations on implicants with the hope that their variance might be lower than the solutions'.

8.6.2.1 Preliminaries

1. the domain of values is $D = \{0, 1, *\}$; given a solution, we call δ the fraction of variables assigned 1 and α the fraction of variables assigned *; constraints 8.2 and 8.3 become:

$$\begin{aligned}
 \delta &= \mu_{1,1} + \mu_{1,0} + \mu_{1,*} \\
 \delta &= \mu_{1,1} + \mu_{0,1} + \mu_{*,1} \\
 \alpha &= \mu_{*,1} + \mu_{*,0} + \mu_{*,*} \\
 \alpha &= \mu_{1,*} + \mu_{0,*} + \mu_{*,*} \\
 1 - \delta - \alpha &= \mu_{0,0} + \mu_{0,1} + \mu_{0,*} \\
 1 - \delta - \alpha &= \mu_{0,0} + \mu_{1,0} + \mu_{*,0} ,
 \end{aligned}$$

so, if we define $\mu = \mu_{*,*}$, $\nu = \mu_{1,1}$, $\pi = \mu_{1,*}$ and $\pi' = \mu_{*,1}$, we have

$$\begin{aligned}
 \mu_{1,0} &= \delta - \nu - \pi \\
 \mu_{0,1} &= \delta - \nu - \pi' \\
 \mu_{*,0} &= \alpha - \mu - \pi' \\
 \mu_{0,*} &= \alpha - \mu - \pi \\
 \mu_{0,0} &= 1 - 2\delta - 2\alpha + \mu + \nu + \pi + \pi' ;
 \end{aligned}$$

2. the set of signs is $S = \{+, -\}$; we call ρ the fraction of positive occurrences;

3. the set of truth values is $\mathcal{V} = \{T, F, *\}$ and the truth table is:

	+	-
0	F	T
1	T	F
*	*	*

so

$$\begin{aligned}
 \eta_T &= \rho\delta + (1 - \rho)(1 - \delta - \alpha) \\
 \eta_F &= (1 - \rho)\delta + \rho(1 - \delta - \alpha) \\
 \eta_* &= \alpha
 \end{aligned}$$

and

$$\begin{aligned}
 \varepsilon_{T,T} &= \rho\nu + (1-\rho)(1-2\delta-2\alpha+\mu+\nu+\pi+\pi') \\
 \varepsilon_{T,F} &= \rho(\delta-\nu-\pi) + (1-\rho)(\delta-\nu-\pi') \\
 \varepsilon_{F,T} &= \rho(\delta-\nu-\pi') + (1-\rho)(\delta-\nu-\pi) \\
 \varepsilon_{F,F} &= \rho(1-2\delta-2\alpha+\mu+\nu+\pi+\pi') + (1-\rho)\nu \\
 \varepsilon_{*,*} &= \mu \\
 \varepsilon_{T,*} &= \rho\pi + (1-\rho)(\alpha-\mu-\pi) \\
 \varepsilon_{*,T} &= \rho\pi' + (1-\rho)(\alpha-\mu-\pi') \\
 \varepsilon_{F,*} &= \rho(\alpha-\mu-\pi) + (1-\rho)\pi \\
 \varepsilon_{*,F} &= \rho(\alpha-\mu-\pi') + (1-\rho)\pi' ;
 \end{aligned}$$

4. a clause type is allowed iff it contains at least one T .

8.6.2.2 Condition for $\varepsilon_{v,w} = \eta_v\eta_w$ at the Independence Point

The independence point is defined by $\mu = \alpha^2, \nu = \delta^2$ and $\pi = \pi' = \alpha\delta$. So at this point we have:

$$\begin{aligned}
 \varepsilon_{T,T} &= \rho\delta^2 + (1-\rho)(1-\delta-\alpha)^2 \\
 \varepsilon_{T,F} = \varepsilon_{F,T} &= \delta(1-\delta-\alpha) \\
 \varepsilon_{F,F} &= \rho(1-\delta-\alpha)^2 + (1-\rho)\delta^2 \\
 \varepsilon_{*,*} &= \alpha^2 = \eta_*^2 \\
 \varepsilon_{T,*} = \varepsilon_{*,T} &= \alpha(\rho\delta + (1-\rho)(1-\delta-\alpha)) = \eta_T\eta_* \\
 \varepsilon_{F,*} = \varepsilon_{*,F} &= \alpha(\rho(1-\delta-\alpha) + (1-\rho)\delta) = \eta_F\eta_* .
 \end{aligned}$$

The first thing to notice is that all identities involving $*$ satisfy $\varepsilon_{v,w} = \eta_v\eta_w$. The second thing to notice is that if $\varepsilon_{T,F} = \eta_T\eta_F$, then the three remaining identities follow, because $\varepsilon_{T,F} + \varepsilon_{T,T} + \varepsilon_{T,*} = \eta_T$ etc.

$$\begin{aligned}
 \varepsilon_{T,F} - \eta_T\eta_F &= \delta(1-\delta-\alpha) - (\rho\delta + (1-\rho)(1-\delta-\alpha))((1-\rho)\delta + \rho(1-\delta-\alpha)) \\
 &= \delta(1-\delta-\alpha) - \rho(1-\rho)(\delta^2 + (1-\delta-\alpha)^2) - (\rho^2 + (1-\rho)^2)\delta(1-\delta-\alpha) \\
 &= 2\rho(1-\rho)\delta(1-\delta-\alpha) - \rho(1-\rho)(\delta^2 + (1-\delta-\alpha)^2) \\
 &= -\rho(1-\rho)(2\delta + \alpha - 1)^2 .
 \end{aligned}$$

Consequently, $\varepsilon_{T,F} \leq \eta_T\eta_F$, with equality iff $\rho \in \{0, 1\}$ or $\delta = \frac{1-\alpha}{2}$.

We discard the particular case of $\rho \in \{0, 1\}$ (which corresponds to monotone k-SAT, always trivially satisfiable). It turns out that as soon as $\delta \neq \frac{1-\alpha}{2}$, we could not make the Second Moment Method work.

On the other hand, when $\delta = \frac{1-\alpha}{2}$ we could make the Second Moment Method work.

Remark 8.8. Setting $\alpha = 0$ corresponds in fact to solutions (which are some trivial implicants), thus to some extent we get back to the $\delta = \frac{1}{2}$ condition. But is there a positive α yielding a better lower bound than $\alpha = 0$?

8.6.2.3 Condition for the Second Moment Method to Work at $\delta = \frac{1-\alpha}{2}$

As mentioned in section 8.5, stationarity of the independence point implies that $\frac{\partial \Lambda}{\partial \mu} = \frac{\partial \Lambda}{\partial v} = \frac{\partial \Lambda}{\partial \pi} = \frac{\partial \Lambda}{\partial \pi'} = 0$.

Using equation 8.9, i.e.

$$\frac{\partial \Lambda}{\partial \mu} = - \sum_{(a,b) \in D^2} \frac{\partial \mu_{a,b}}{\partial \mu} \ln \mu_{a,b} + c \sum_{(a,b) \in D^2} \frac{\partial \mu_{a,b}}{\partial \mu} \sum_{s \in S} \rho_s \frac{\Xi_{a \otimes s, b \otimes s}}{\varepsilon_{a \otimes s, b \otimes s}},$$

we get:

$$\begin{aligned} \frac{\partial \Lambda}{\partial \mu} &= \ln \frac{\mu_{*,0} \mu_{0,*}}{\mu_{*,*} \mu_{0,0}} \\ &\quad + c \left((1-\rho) \left(\frac{\Xi_{T,T}}{\varepsilon_{T,T}} + \frac{\Xi_{*,*}}{\eta_{*,*}} - \frac{\Xi_{T,*}}{\varepsilon_{T,*}} - \frac{\Xi_{*,T}}{\varepsilon_{*,T}} \right) + \rho \left(\frac{\Xi_{F,F}}{\varepsilon_{F,F}} + \frac{\Xi_{*,*}}{\eta_{*,*}} - \frac{\Xi_{F,*}}{\varepsilon_{F,*}} - \frac{\Xi_{*,F}}{\varepsilon_{*,F}} \right) \right); \\ \frac{\partial \Lambda}{\partial v} &= \ln \frac{\mu_{1,0} \mu_{0,1}}{\mu_{1,1} \mu_{0,0}} + c \left(\frac{\Xi_{T,T}}{\varepsilon_{T,T}} + \frac{\Xi_{F,F}}{\varepsilon_{F,F}} - \frac{\Xi_{T,F}}{\varepsilon_{T,F}} - \frac{\Xi_{F,T}}{\varepsilon_{F,T}} \right); \\ \frac{\partial \Lambda}{\partial \pi} &= \ln \frac{\mu_{1,0} \mu_{0,*}}{\mu_{1,*} \mu_{0,0}} \\ &\quad + c \left((1-\rho) \left(\frac{\Xi_{T,T}}{\varepsilon_{T,T}} + \frac{\Xi_{F,*}}{\varepsilon_{F,*}} - \frac{\Xi_{F,T}}{\varepsilon_{F,T}} - \frac{\Xi_{T,*}}{\varepsilon_{T,*}} \right) + \rho \left(\frac{\Xi_{F,F}}{\varepsilon_{F,F}} + \frac{\Xi_{T,*}}{\varepsilon_{T,*}} - \frac{\Xi_{T,F}}{\varepsilon_{T,F}} - \frac{\Xi_{F,*}}{\varepsilon_{F,*}} \right) \right); \\ \frac{\partial \Lambda}{\partial \pi'} &= \ln \frac{\mu_{0,1} \mu_{*,0}}{\mu_{*,1} \mu_{0,0}} \\ &\quad + c \left((1-\rho) \left(\frac{\Xi_{T,T}}{\varepsilon_{T,T}} + \frac{\Xi_{*,F}}{\varepsilon_{*,F}} - \frac{\Xi_{T,F}}{\varepsilon_{T,F}} - \frac{\Xi_{*,T}}{\varepsilon_{*,T}} \right) + \rho \left(\frac{\Xi_{F,F}}{\varepsilon_{F,F}} + \frac{\Xi_{*,T}}{\varepsilon_{*,T}} - \frac{\Xi_{F,T}}{\varepsilon_{F,T}} - \frac{\Xi_{*,F}}{\varepsilon_{*,F}} \right) \right). \end{aligned}$$

We consider the independence point, thus $\mu_{a,b} = \delta_a \delta_b$. Assuming symmetry of occurrences, we may use independence of surfaces (cf. fact 8.3); moreover, using condition $\varepsilon_{v,w} = \eta_v \eta_w$, we get that:

$$\begin{aligned} \frac{\partial \Lambda}{\partial \mu} &= c \left((1-\rho) \left(\frac{\Sigma_T}{\eta_T} - \frac{\Sigma_*}{\eta_*} \right)^2 + \rho \left(\frac{\Sigma_F}{\eta_F} - \frac{\Sigma_*}{\eta_*} \right)^2 \right); \\ \frac{\partial \Lambda}{\partial v} &= c \left(\frac{\Sigma_T}{\eta_T} - \frac{\Sigma_F}{\eta_F} \right)^2; \\ \frac{\partial \Lambda}{\partial \pi} = \frac{\partial \Lambda}{\partial \pi'} &= c \left((1-\rho) \left(\frac{\Sigma_T}{\eta_T} - \frac{\Sigma_F}{\eta_F} \right) \left(\frac{\Sigma_T}{\eta_T} - \frac{\Sigma_*}{\eta_*} \right) + \rho \left(\frac{\Sigma_F}{\eta_F} - \frac{\Sigma_T}{\eta_T} \right) \left(\frac{\Sigma_F}{\eta_F} - \frac{\Sigma_*}{\eta_*} \right) \right). \end{aligned}$$

Canceling out these derivatives yields:

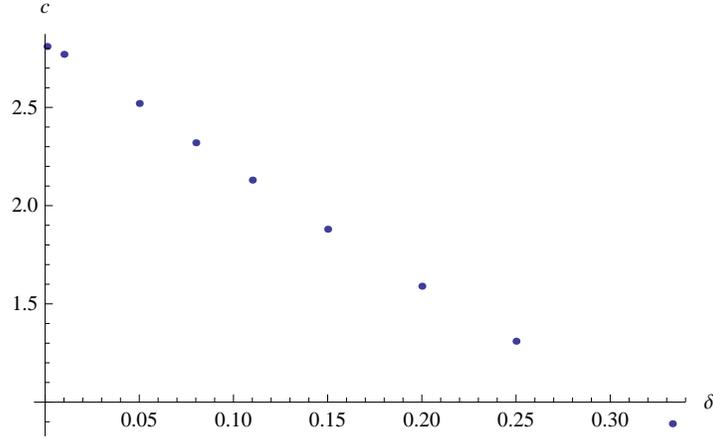
$$\frac{\Sigma_*}{\eta_*} = \frac{\Sigma_T}{\eta_T} = \frac{\Sigma_F}{\eta_F}.$$

Since we assume $\delta = \frac{1-\alpha}{2}$, we have $\eta_T = \eta_F = \frac{1-\alpha}{2}$; moreover $\eta_* = \alpha$. Thus $\Sigma_T = \Sigma_F = k \frac{1-\alpha}{2}$ and $\Sigma_* = k\alpha$.

It turns out that this condition is sufficient to make the Second Moment Method work, and numerically we found the critical ratios laid in table 8.1 for standard 3-SAT with symmetry of occurrences at $\rho = \frac{1}{2}$.

Table 8.1: Critical ratios c of implicants obtained for a given α (and the corresponding choice of the free β parameters).

α	c	β_{TFF}	β_{TTF}	$\beta_{T^{**}}$	ratio c of [BD99]
0.001	2.81	0.195	0.10867	3.33×10^{-5}	4.5
0.01	2.77	0.1942	0.098833	3.33×10^{-5}	4.5
0.05	2.52	0.17767	0.08167	0.001633	2
0.08	2.32	0.1633	0.07	0.00233	1.5
0.11	2.13	0.1533	0.05467	0.0033	1
0.15	1.88	0.13833	0.041	0.012	-
0.2	1.59	0.1233	0.02567	0.02833	-
0.25	1.31	0.10833	0.0133	0.04767	-
0.333	0.89	0.094433	3.33×10^{-5}	0.094167	-



These values are to be compared with those of Boufkhad & Dubois - 1999 [BD99], who proved for example that at the ratio $c = 4.5$, any satisfiable instance will have prime implicants with $\alpha = 0.01$. Combined with the lower bound of 3.52 of [KKL06] and [HS03], this proves that such implicants exist almost surely when $c \leq 3.52$.

Thus in the range $c \in (2.81, 3.52)$ Boufkhad & Dubois prove that implicants with $\alpha = 0.01$ exist. However, in the range $c \leq 2.81$ the Second Moment Method enable us to establish the existence of implicants with a α significantly greater than Boufkhad & Dubois's.

How can we interpret the fact that the critical c obtained decreases with α ? Looking at the set of allowed types of clauses:

$$\mathcal{T} = \{TTT, TTF, TFT, FTT, TFF, FTF, FFT, TT^*, T^*T, ^*TT, T^{**}, ^*T^*, **T\} \\ \cup \{TF^*, T^*F, FT^*, F^*T, ^*TF, ^*FT\} ,$$

we can see that there are 27 T 's, 15 F 's and 15 $*$'s. Thus when $*$'s are present, the ratio T/F is $\frac{27}{15} = 1.8$. Without $*$'s, T/F would be $\frac{12}{9} \simeq 1.33$ (see section 8.7.2). Now, since the Second Moment Method requires $\Sigma_T = \Sigma_F$, we can see that it is all the more artificial as T/F is large. Thus adding $*$'s should cut down the Second Moment Method's performance.

8.7 Confrontation of the Second Moment Method with Reality

Using SATLab, we investigate the behavior of real solutions and we emphasize how it differs from the conditions required by the Second Moment Method that we laid down just above.

Based on numerical calculations of figures 8.1 and 8.2, we conjectured in section 8.5 that the Second Moment Method might work only if $\varepsilon_{v,w} = \eta_v \eta_w$. In this setting we showed that the independence point defined by $\mu_{a,b} = \delta_a \delta_b$ and $\gamma_{t,u} = \beta_t \beta_u$ must be a maximum of T_2 , the second moment. This led us in section 8.6.1 to the following necessary condition for the Second Moment Method to work on boolean solutions: $\delta = \frac{1}{2}$.

Now using SATLab, we are going to give experimental evidence that:

- real solutions of standard 3-SAT violate all of these conditions: they are not independent at all!
- real solutions of standard 3-NAE-SAT seem to be rather independent.

These observations may explain why the Second Moment Method performs so poorly on standard 3-SAT (cf. section 8.6.1) whereas it works pretty well on 3-NAE-SAT (cf. Achlioptas & Moore - 2002 [AM02]).

8.7.1 Distances between Solutions

It turns out that in random 3-SAT, solutions are correlated with respect to their Hamming distances. Namely their Hamming distances are not centered around 50% contrary to solutions of random 3-NAE-SAT, but narrower to each other (cf. figure 8.3).

What we mean by *Hamming similarity* between two assignments is just the proportion of variables assigned the same value in both assignments. We took all couples of different solutions in a sample of random solutions output by a solver, and we plotted the frequency of the Hamming similarity.

To have a more precise insight into Hamming similarity, we separated fixed and free variables. Let us recall that a variable is free iff flipping it yields another solution. We can notice that Hamming similarity is significantly greater among fixed variables than among free variables (see figure 8.4), and that it increases with c for both types of variables (see figures 8.5 and 8.6).

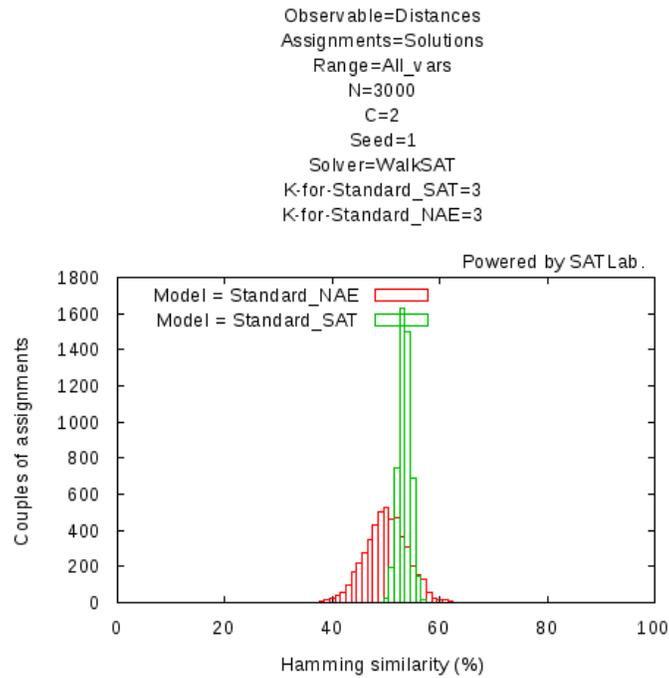


Figure 8.3: Hamming similarity among solutions in NAE-SAT and in SAT.

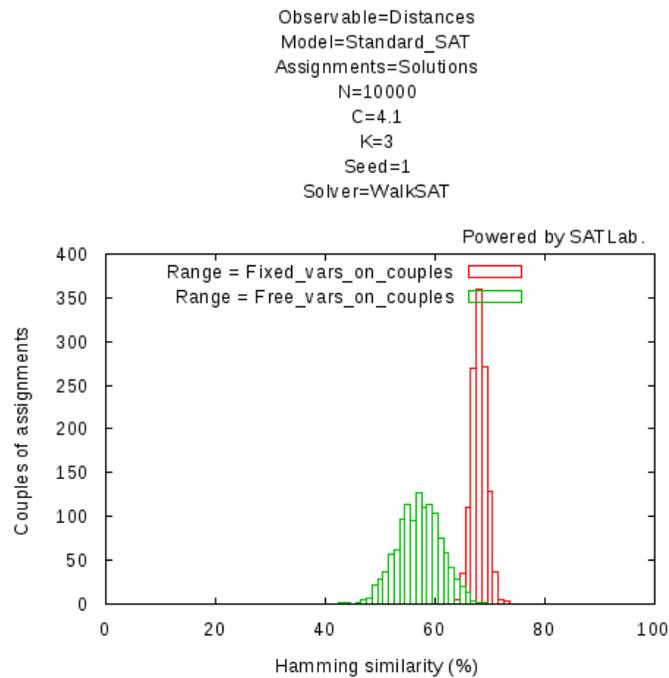


Figure 8.4: Hamming similarity is greater among fixed variables.

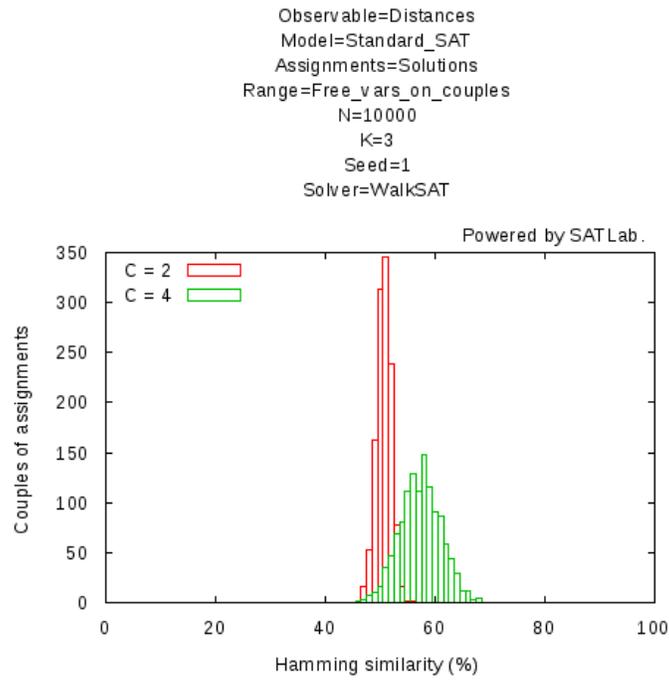


Figure 8.5: Hamming similarity among free variables increases with c .

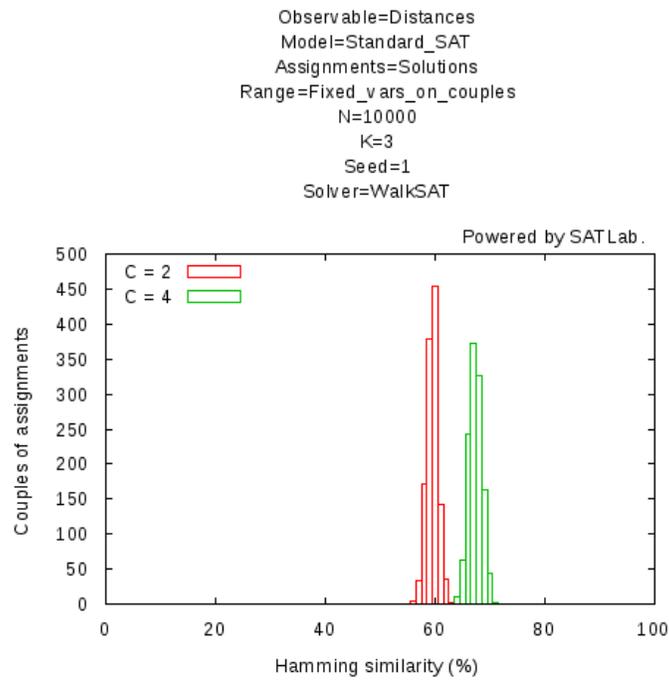


Figure 8.6: Hamming similarity among fixed variables increases with c .

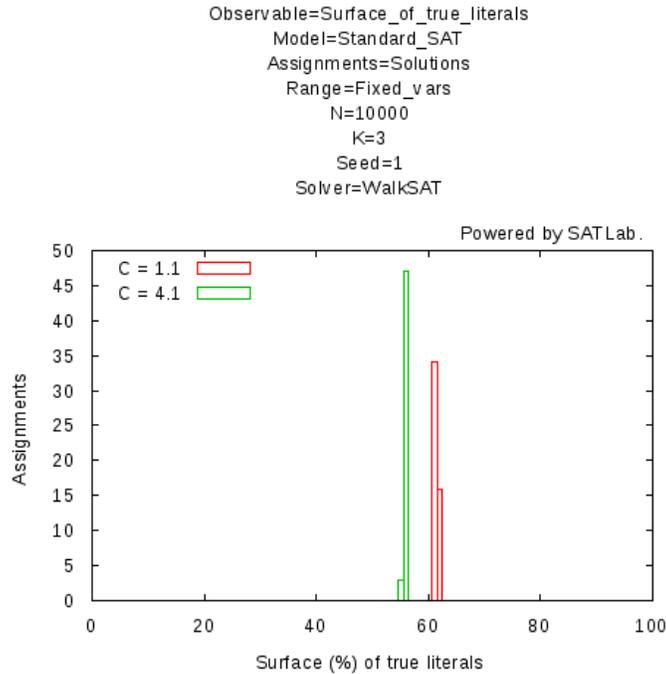


Figure 8.7: The true surface of fixed variables decreases with c .

8.7.2 Surface of True Literals

What we call true *surface* is the scaled number of true occurrences of literals. We can see a fundamental difference between the true surface of fixed variables and the true surface of free variables. Namely the true surface of fixed variables decreases with c (figure 8.7) whereas the true surface of free variables increases with c (figure 8.8). Note that both quantities converge to roughly 0.56 (i.e. roughly $\frac{4}{7}$) when c approaches the threshold ratio, whereas in section 8.6.1 we got the following condition: $\Sigma_T = \Sigma_F = \frac{k}{2}$ to make the Second Moment Method work.

We interpret the ratio $\frac{4}{7}$ as follows: the allowed types of clauses are

$$\{TTT, TTF, TFT, FTT, TFF, FTF, FFT\} ,$$

which amounts to 12 T 's and 9 F 's. Now $\frac{12}{12+9} = \frac{12}{21} = \frac{4}{7}$.

8.7.3 Non-Independence of True / False Surfaces

Let us consider two solutions S_1 and S_2 . We denote by Σ_F the false surface under solution S_1 , Σ_T the true surface under solution S_2 , and Ξ_{FT} the surface which is false under S_1 and true under S_2 . In a given sample of random solutions, we took all couples of different solutions (S_1, S_2) and computed the ratio $\frac{\Sigma_F \Sigma_T}{k \Xi_{FT}}$; the histogram in figure 8.9 plots the frequency of this ratio for the solutions of two different models of formulas: random 3-NAE-SAT and random 3-SAT. Although some independence seems to exist in 3-NAE-SAT (i.e. the ratio is centered around 1), it can be seen that there is no independence of these surfaces for random 3-SAT.

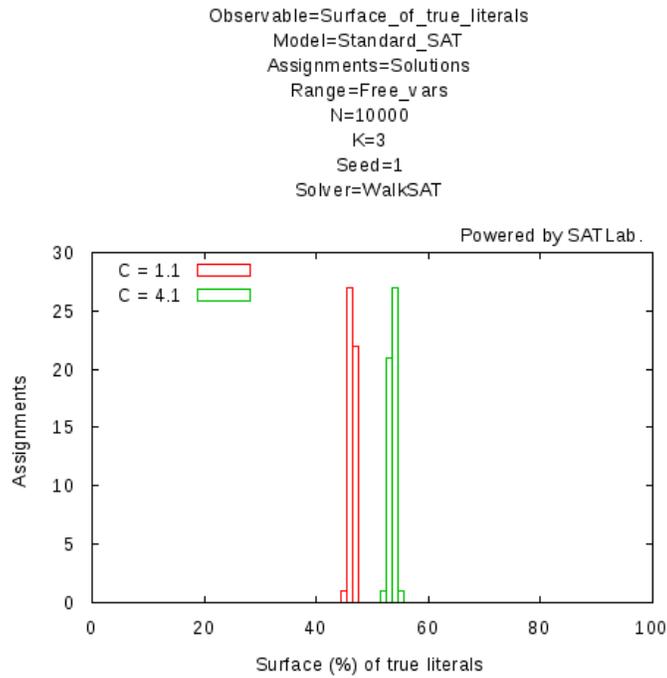


Figure 8.8: The true surface of free variables increases with c .

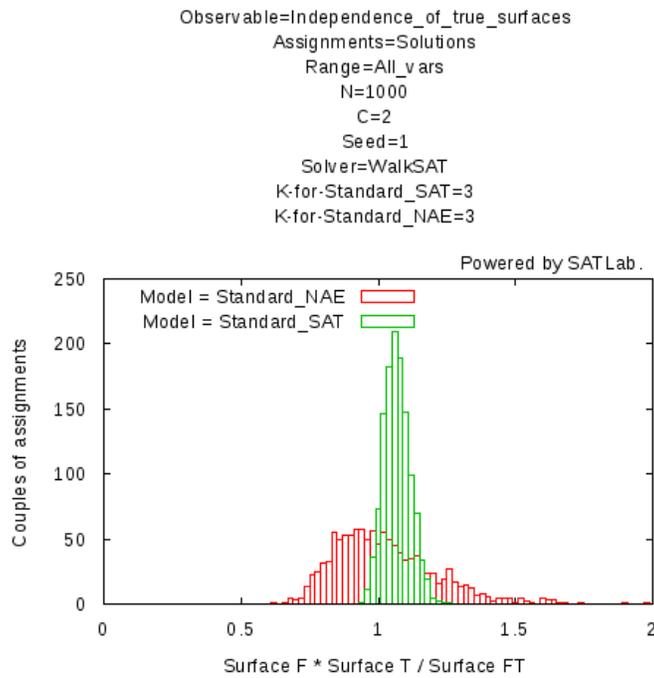


Figure 8.9: (Non-)independence of surfaces in NAE-SAT and in SAT.

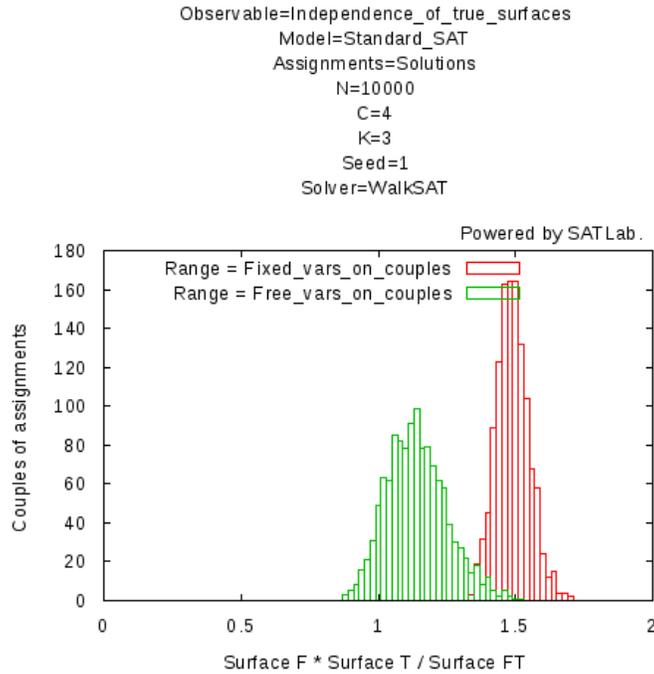


Figure 8.10: The non-independence of surfaces comes rather from fixed variables.

To have a more precise insight into the non-independence, we separated fixed and free variables. Let us recall that a variable is free iff flipping it yields another solution. We can notice that non-independence comes from both free and fixed variables, but rather from fixed variables than from free variables, cf. figure 8.10.

8.7.4 Non-Independence of Clauses Types

Here what we call *clause type* is the number of true occurrences of variables in the clause. Let us consider two solutions S_1 and S_2 . We denote by β_1 the proportion of uniquely satisfied clauses under solution S_1 , b_1 the proportion of uniquely satisfied clauses under solution S_2 , and $\gamma_{1,1}$ the proportion of clauses which are uniquely satisfied under S_1 and under S_2 . In a given sample of random solutions, we took all couples of different solutions (S_1, S_2) and computed the ratio $\frac{\beta_1 b_1}{\gamma_{1,1}}$; the histogram in figure 8.11 plots the frequency of this ratio for the solutions of two different kinds of assignments: solutions of a random 3-NAE-SAT formula and solutions of a random 3-SAT formula. Although independence seems to happen among solutions of random 3-NAE-SAT (i.e. the ratio is centered around 1), it can be seen that there is no independence in random 3-SAT.

In real instances we can assume symmetry of occurrences, in the sense of section 8.1.6; so in the light of fact 8.3, we could conclude from the non-independence of surfaces observed in section 8.7.3 that in the real solutions of 3-SAT independence of clauses types would not hold.

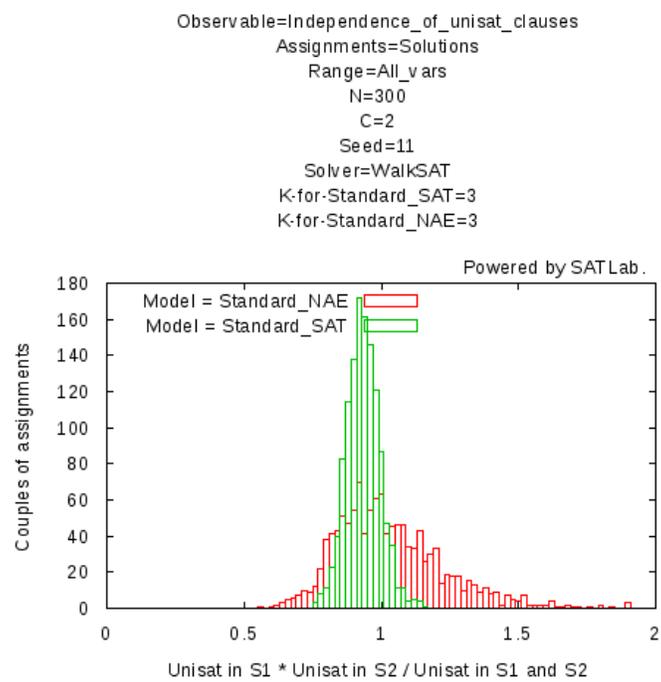


Figure 8.11: (Non-)independence of uniquely satisfied clauses in NAE-SAT and in SAT.

Second Moment Method on Distributional Random k-SAT

REMEMBER that using the standard distributional model (defined in section 1.1.4) instead of the standard drawing model (defined in section 1.1.3) yields better upper bounds on the satisfiability threshold (cf. table 6.2). Moreover, we would like to gain some more control over the proportion of variables assigned 1 according to the imbalance between their positive and negative occurrences. Namely, a variable is all the more expected to be assigned 1 in a solution as it has more positive occurrences, and vice-versa. At least this seems to happen on real solutions, see figure 9.2 in section 9.6.1.

That is the reasons why we are going to implement the Second Moment Method in the distributional model.

In this chapter we follow roughly the same outline than in chapter 8, but we focus on solutions only; we only emphasize the major differences with respect to the general framework of chapter 8.

9.1 Preliminaries

9.1.1 Occurrences and Signs

We still have n variables. We denote by $d_{p,q}$ the fraction of variables having p positive and q negative occurrences.

$$\begin{aligned} \sum_{(p,q) \in \mathbb{N}^2} d_{p,q} &= 1 ; \\ \sum_{(p,q) \in \mathbb{N}^2} (p+q) d_{p,q} &= kc . \end{aligned}$$

Occurrences and signs of variables are determined *a priori*.

As in section 6.3.1, we ought to consider light and heavy variables \mathcal{L} and \mathcal{H} . In fact we are going not to worry about that, because they make the calculation heavier, and in the end we shall see that there is no need to be rigorous since we only have negative results.

9.1.2 Values

Given a boolean assignment, we denote by $\delta_{p,q}$ the proportion of variables with p positive and q negative occurrences which are assigned 1. Thus the proportion of variables with p positive and q negative occurrences which are assigned 0 is $1 - \delta_{p,q}$.

Given two assignments S_1 and S_2 , for all $(a, b) \in D^2$, considering variables with p positive and q negative occurrences, we denote by:

- $\lambda_{p,q}$ the proportion of variables which are assigned 1 in S_1 and S_2 ;
- $\mu_{p,q}$ the proportion of variables which are assigned 1 in S_1 and 0 in S_2 ;
- $\mu'_{p,q}$ the proportion of variables which are assigned 0 in S_1 and 1 in S_2 ;
- $\nu_{p,q}$ the proportion of variables which are assigned 0 in S_1 and S_2 .

We have the following constraints:

$$\begin{aligned}\delta_{p,q} &= \lambda_{p,q} + \mu_{p,q} \\ \delta_{p,q} &= \lambda_{p,q} + \mu'_{p,q} \\ 1 - \delta_{p,q} &= \mu_{p,q} + \nu_{p,q} \\ 1 - \delta_{p,q} &= \mu'_{p,q} + \nu_{p,q}\end{aligned}$$

thus

$$\begin{aligned}\lambda_{p,q} &= \delta_{p,q} - \mu_{p,q} \\ \mu'_{p,q} &= \mu_{p,q} \\ \nu_{p,q} &= 1 - \delta_{p,q} - \mu_{p,q}\end{aligned}$$

which enables us to work only with $\mu_{p,q}$.

9.1.3 Truth Values

$$\begin{aligned}\eta_T &= \frac{1}{kC} \sum_{(p,q) \in \mathcal{N}^2} (p\delta_{p,q} + q(1 - \delta_{p,q})) d_{p,q} \\ \eta_F &= \frac{1}{kC} \sum_{(p,q) \in \mathcal{N}^2} (q\delta_{p,q} + p(1 - \delta_{p,q})) d_{p,q} \\ \varepsilon_{T,T} &= \frac{1}{kC} \sum_{(p,q) \in \mathcal{N}^2} (p(\delta_{p,q} - \mu_{p,q}) + q(1 - \delta_{p,q} - \mu_{p,q})) d_{p,q} \\ \varepsilon_{F,F} &= \frac{1}{kC} \sum_{(p,q) \in \mathcal{N}^2} (q(\delta_{p,q} - \mu_{p,q}) + p(1 - \delta_{p,q} - \mu_{p,q})) d_{p,q} \\ \varepsilon_{T,F} = \varepsilon_{F,T} &= \frac{1}{kC} \sum_{(p,q) \in \mathcal{N}^2} (p + q) \mu_{p,q} d_{p,q}\end{aligned}$$

9.1.4 Types of Clauses and Surfaces

We keep the same definitions as in section 8.1.

However, some extra constraints on surfaces occur in the distributional model, because here all occurrences and signs of variables are determined *a priori*:

$$\Sigma_v = k\eta_v ; \quad (9.1)$$

$$\Xi_{v,w} = k\varepsilon_{v,w} . \quad (9.2)$$

9.2 Expression of the First Moment

The first moment of the number X of solutions can be split up into the following factors: total number of assignments and probability for an assignment to be a solution.

1. total number of assignments: choose subsets of variables assigned 0 or 1:

$$\prod_{(p,q) \in \mathcal{L}} \binom{d_{p,q}n}{\delta_{p,q}d_{p,q}n};$$

2. probability for an assignment to be a solution:

- a) number of satisfied formulas:

- i. we give each clause an allowed type $t \in \mathcal{T}$: $(\dots(\beta_t cn)_{t \in \mathcal{T}} \dots)$

- ii. we find a permutation of the true literals into the true boxes and a permutation of the false literals into the false boxes: $(\eta_T kcn)! (\eta_F kcn)!$

- b) total number of formulas, i.e. number of permutations of the occurrences of literals into the boxes: $(kcn)!$

We denote by \mathcal{P} the set of all families of non-negative numbers $((\delta_{p,q})_{(p,q) \in \mathcal{L}}, (\beta_t)_{t \in \mathcal{T}})$ satisfying constraint 8.5. We denote by $\mathcal{I}(n)$ the intersection of \mathcal{P} with the multiples of $\frac{1}{n}$; we get the following expression of the first moment:

$$EX = \sum_{((\delta_{p,q})_{(p,q) \in \mathcal{L}}, (\beta_t)_{t \in \mathcal{T}}) \in \mathcal{I}(n)} T_1(n)$$

where

$$T_1(n) = \frac{\prod_{(p,q) \in \mathcal{L}} \binom{d_{p,q}n}{\delta_{p,q}d_{p,q}n} (\dots(\beta_t cn)_{t \in \mathcal{T}} \dots)}{\binom{kcn}{\eta_T kcn}} .$$

From equation B.2, the exponential equivalent of $T_1(n)$ is T_1^n , where

$$T_1 = \frac{1}{\prod_{(p,q) \in \mathcal{L}^2} \delta_{p,q}^{\delta_{p,q}d_{p,q}}} \left(\frac{(\eta_T^{\eta_T} \eta_F^{\eta_F})^k}{\prod_{t \in \mathcal{T}} \beta_t^{\beta_t}} \right)^c .$$

9.3 Expression of the Second Moment

The second moment of X can be split up into the following factors: total number of assignments and probability for an assignment to be a solution.

1. total number of assignments: choose subsets of variables assigned 0 or 1:

$$\prod_{(p,q) \in \mathcal{L}} (\lambda_{p,q} d_{p,q} n, \mu_{p,q} d_{p,q} n, \nu_{p,q} d_{p,q} n)^{d_{p,q} n};$$

2. probability for an assignment to be a solution:

- a) number of satisfied formulas:

- i. we give each clause two allowed types: $(\dots(\gamma_{t,u}^{cn})_{(t,u) \in \mathcal{T}^2} \dots)$

- ii. we find a permutation of the literals into the corresponding boxes:
 $(\varepsilon_{T,T} kcn)! (\varepsilon_{T,F} kcn)! (\varepsilon_{F,T} kcn)! (\varepsilon_{F,F} kcn)!$

- b) total number of formulas, i.e. number of permutations of the occurrences of literals into the boxes: $(kcn)!$

We denote by \mathcal{P}_2 the set of all families of non-negative numbers $((\mu_{p,q})_{(p,q) \in \mathcal{L}}, (\gamma_{t,u})_{(t,u) \in \mathcal{T}^2})$ satisfying constraints 8.6 and 8.7. We denote by $\mathcal{I}_2(n)$ the intersection of \mathcal{P}_2 with the multiples of $\frac{1}{n}$; we get the following expression of the second moment:

$$\mathbb{E}X^2 = \sum_{((\mu_{p,q})_{(p,q) \in \mathcal{L}}, (\gamma_{t,u})_{(t,u) \in \mathcal{T}^2}) \in \mathcal{I}_2(n)} T_2(n)$$

where

$$T_2(n) = \frac{\prod_{(p,q) \in \mathcal{L}} (\lambda_{p,q} d_{p,q} n, \mu_{p,q} d_{p,q} n, \nu_{p,q} d_{p,q} n)^{d_{p,q} n} (\dots(\gamma_{t,u}^{cn})_{(t,u) \in \mathcal{T}^2} \dots)}{(\varepsilon_{T,T} kcn, \varepsilon_{T,F} kcn, \varepsilon_{F,T} kcn, \varepsilon_{F,F} kcn)^{kcn}}.$$

From equation B.2, the exponential equivalent of $T_2(n)$ is T_2^n , where

$$T_2^n = \frac{1}{\prod_{(p,q) \in \mathcal{L}} \left((\delta_{p,q} - \mu_{p,q})^{\delta_{p,q} - \mu_{p,q}} \mu_{p,q}^{2\mu_{p,q}} (1 - \delta_{p,q} - \mu_{p,q})^{1 - \delta_{p,q} - \mu_{p,q}} \right)^{d_{p,q}} \cdot \left(\frac{(\varepsilon_{T,T}^{\varepsilon_{T,T}} \varepsilon_{T,F}^{\varepsilon_{T,F}} \varepsilon_{F,T}^{\varepsilon_{F,T}} \varepsilon_{F,F}^{\varepsilon_{F,F}})^k}{\prod_{(t,u) \in \mathcal{T}^2} \gamma_{t,u}^{\gamma_{t,u}}} \right)^c}.$$

9.4 Expression of the Lagrangian

When the parameters of the first moment (i.e. $(\delta_{p,q})_{(p,q) \in \mathcal{L}}, (\beta_t)_{t \in \mathcal{T}}$) are chosen, T_2 must be maximized under constraints 8.6 and 8.7. That leads us to use the Lagrange multipliers method. In order to make the forthcoming maximization easier, we introduce some extra variables $\psi_{v,w}$ which are going to simulate $\varepsilon_{v,w}$. The reason for this is that $\varepsilon_{v,w}$ contains $\mu_{p,q}$, but we need the expression of $\mu_{p,q}$ for our numerical calculations. So, because of equation 9.2, we have the following constraints:

$$\begin{aligned} k\varepsilon_{T,T} &= k\psi_{T,T} &= \Xi_{T,T} \\ k\varepsilon_{T,F} &= k\psi_{T,F} &= \Xi_{T,F} \\ k\varepsilon_{F,T} &= k\psi_{F,T} &= \Xi_{F,T} \\ k\varepsilon_{F,F} &= k\psi_{F,F} &= \Xi_{F,F} \end{aligned}$$

Using the facts that $\varepsilon_{T,T} + \varepsilon_{T,F} = \eta_T$, $\Xi_{T,T} + \Xi_{T,F} = \Sigma_T$ and $\Sigma_T = k\eta_T$, we see that constraint $\varepsilon_{T,T} = \psi_{T,T}$ is redundant. Eliminating $\varepsilon_{F,T}$ and $\varepsilon_{F,F}$ as well, there remain the following 5 constraints:

$$\varepsilon_{T,F} = \psi_{T,F} \quad (9.3)$$

$$\psi_{T,T} = \eta_T - \psi_{T,F} \quad (9.4)$$

$$\psi_{T,F} = \frac{\Xi_{T,F}}{k} \quad (9.5)$$

$$\psi_{F,T} = \psi_{T,F} \quad (9.6)$$

$$\psi_{F,F} = \eta_F - \psi_{T,F} \quad (9.7)$$

So we define the following Lagrangian:

$$\begin{aligned} \Lambda = & - \sum_{(p,q) \in \mathcal{L}} d_{p,q} \left((\delta_{p,q} - \mu_{p,q}) \ln \frac{\delta_{p,q} - \mu_{p,q}}{e} + (1 - \delta_{p,q} - \mu_{p,q}) \ln \frac{1 - \delta_{p,q} - \mu_{p,q}}{e} \right) \\ & - 2 \sum_{(p,q) \in \mathcal{L}} d_{p,q} \mu_{p,q} \ln \frac{\mu_{p,q}}{e} - c \sum_{(t,u) \in \mathcal{T}^2} \gamma_{t,u} \ln \frac{\gamma_{t,u}}{e} + kc \sum_{(v,w) \in \{T,F\}^2} \psi_{v,w} \ln \frac{\psi_{v,w}}{e} \\ & + c \sum_{(v,w) \in \{T,F\}^2} (\ln h_{v,w}) (\Xi_{v,w} - k\psi_{v,w}) + kc (\ln y) (\psi_{T,F} - \varepsilon_{T,F}) \\ & + c \sum_{t \in \mathcal{T}} (\ln f_t) \left(\sum_{u \in \mathcal{T}} \gamma_{t,u} - \beta_t \right) + c \sum_{u \in \mathcal{U}} (\ln g_u) \left(\sum_{t \in \mathcal{T}} \gamma_{t,u} - \beta_u \right) . \end{aligned}$$

9.4.1 Derivative with respect to $\gamma_{t,u}$

$$\frac{\partial \Lambda}{\partial \gamma_{t,u}} = -c \ln \gamma_{t,u} + c \sum_{i=1}^k \ln h_{t_i, u_i} + c \ln f_t + c \ln g_u .$$

Canceling out this derivative yields:

$$\gamma_{t,u} = f_t g_u \prod_{i=1}^k h_{t_i, u_i} . \quad (9.8)$$

9.4.2 Derivative with respect to $\psi_{v,w}$

$$\frac{\partial \Lambda}{\partial \psi_{v,w}} = kc \ln \psi_{v,w} - kc \ln h_{v,w} + kc (\ln y) \mathbf{1}_{v=T \wedge w=F} .$$

Canceling out these derivatives yields:

$$\psi_{v,w} = h_{v,w} y^{-\mathbf{1}_{v=T \wedge w=F}} . \quad (9.9)$$

Thus constraints 9.3, 9.4, 9.5, 9.6 and 9.7 become:

$$\begin{aligned}\varepsilon_{T,F} &= \frac{h_{T,F}}{y} \\ h_{T,T} &= \eta_T - \frac{h_{T,F}}{y} \\ \frac{h_{T,F}}{y} &= \frac{\Xi_{T,F}}{k} \\ h_{F,T} &= \frac{h_{T,F}}{y} \\ h_{F,F} &= \eta_F - \frac{h_{T,F}}{y}\end{aligned}$$

9.4.3 Derivative with respect to $\mu_{p,q}$

$$\frac{\partial \Lambda}{\partial \mu_{p,q}} = d_{p,q} \ln \frac{(\delta_{p,q} - \mu_{p,q})(1 - \delta_{p,q} - \mu_{p,q})}{\mu_{p,q}^2} - (p+q) d_{p,q} \ln y . \quad (9.10)$$

Canceling out this derivative yields:

$$(\delta_{p,q} - \mu_{p,q})(1 - \delta_{p,q} - \mu_{p,q}) = \mu_{p,q}^2 y^{p+q} ,$$

i.e.

$$\mu_{p,q}^2 (1 - y^{p+q}) - \mu_{p,q} + \delta_{p,q} (1 - \delta_{p,q}) = 0 .$$

Thus there are 2 cases to consider:

1. case where $y = 1$ or $p + q = 0$: $\mu_{p,q} = \delta_{p,q} (1 - \delta_{p,q})$;
2. case where $y \neq 1$ and $p + q \neq 0$: $\mu_{p,q} = \frac{1 \pm \sqrt{1 - 4(1 - y^{p+q})\delta_{p,q}(1 - \delta_{p,q})}}{2(1 - y^{p+q})}$; numerically we can find solutions with $\mu_{p,q} = \frac{1 - \sqrt{1 - 4(1 - y^{p+q})\delta_{p,q}(1 - \delta_{p,q})}}{2(1 - y^{p+q})}$.

9.5 Independence Point

As in section 8.5, we define the independence point by $\mu_{p,q} = \delta_{p,q} (1 - \delta_{p,q})$ and $\gamma_{t,u} = \beta_t \beta_u$. Again, we were able to make the Second Moment Method work only if $\varepsilon_{v,w} = \eta_v \eta_w$.

When $\varepsilon_{v,w} = \eta_v \eta_w$, we have $\frac{T_2}{T_1} = 1$ (see proof of theorem 8.5) and the independence point is stationary without any extra condition (plug $h_{v,w} = \eta_v \eta_w$, $y = 1$, $f_t = \frac{\beta_t}{\prod_{i=1}^k \eta_{t_i}}$, $g_u = \frac{\beta_u}{\prod_{i=1}^k \eta_{u_i}}$ into the constraints and equations 9.8, 9.9 and 9.10), assuming symmetry of occurrences as usual (and thus fact 8.3). By comparison with chapter 8, we could say that the extra condition we had there on the surfaces to make the independence point stationary corresponds here to the preliminary extra constraint 9.2.

Moreover, it is noteworthy that, because of fact 8.3 and constraint 9.1, the independence point violates constraint 9.2 when $\varepsilon_{v,w} \neq \eta_v \eta_w$.

So, what is the condition for $\varepsilon_{v,w} = \eta_v \eta_w$?

9.5.1 Condition for $\varepsilon_{v,w} = \eta_v \eta_w$ at the Independence Point

As before in section 8.6.1, the first thing to notice is that if $\varepsilon_{T,F} = \eta_T \eta_F$, then the three other identities follow, because $\varepsilon_{T,F} + \varepsilon_{T,T} = \eta_T$ etc.

We are grateful to Emmanuel Lepage, who gave us the main idea to compare $\varepsilon_{T,F}$ and $\eta_T \eta_F$. Let us make the following change of variables: $\delta_{p,q} = \frac{1}{2} + \delta'_{p,q}$:

$$\begin{aligned}\eta_T &= \frac{1}{2} + \frac{1}{kc} \sum_{(p,q) \in \mathbb{N}^2} (p-q) \delta'_{p,q} d_{p,q} ; \\ \eta_F &= \frac{1}{2} - \frac{1}{kc} \sum_{(p,q) \in \mathbb{N}^2} (p-q) \delta'_{p,q} d_{p,q} ; \\ \varepsilon_{T,F} &= \frac{1}{4} - \frac{1}{kc} \sum_{(p,q) \in \mathbb{N}^2} (p+q) \delta'^2_{p,q} d_{p,q} .\end{aligned}$$

Thus

$$\varepsilon_{T,F} - \eta_T \eta_F = \left(\frac{1}{kc} \sum_{(p,q) \in \mathbb{N}^2} (p-q) \delta'_{p,q} d_{p,q} \right)^2 - \frac{1}{kc} \sum_{(p,q) \in \mathbb{N}^2} (p+q) \delta'^2_{p,q} d_{p,q} .$$

1. $\left(\frac{1}{kc} \sum_{(p,q) \in \mathbb{N}^2} (p-q) \delta'_{p,q} d_{p,q} \right)^2 \leq \left(\frac{1}{kc} \sum_{(p,q) \in \mathbb{N}^2} |(p-q) \delta'_{p,q}| d_{p,q} \right)^2$, with equality iff $(p-q) \delta'_{p,q}$ has the same sign wherever $d_{p,q} \neq 0$;
2. Since $p, q \geq 0$, $|p-q| \leq p+q$ with equality iff $p=0$ or $q=0$;
thus $\left(\frac{1}{kc} \sum_{(p,q) \in \mathbb{N}^2} |(p-q) \delta'_{p,q}| d_{p,q} \right)^2 \leq \left(\frac{1}{kc} \sum_{(p,q) \in \mathbb{N}^2} \sqrt{p+q} |\delta'_{p,q}| \sqrt{d_{p,q}} \right)^2$, with equality iff $\delta'_{p,q} = 0$ wherever $p \neq 0, q \neq 0$ and $d_{p,q} \neq 0$;
3. By the Cauchy-Schwartz inequality,

$$\begin{aligned}& \left(\frac{1}{kc} \sum_{(p,q) \in \mathbb{N}^2} \sqrt{p+q} |\delta'_{p,q}| \sqrt{d_{p,q}} \right)^2 \\ & \leq \left(\frac{1}{kc} \sum_{(p,q) \in \mathbb{N}^2} \sqrt{p+q} \sqrt{d_{p,q}} \right) \left(\frac{1}{kc} \sum_{(p,q) \in \mathbb{N}^2} \sqrt{p+q} |\delta'_{p,q}|^2 \sqrt{d_{p,q}} \right) \\ & = \frac{1}{kc} \sum_{(p,q) \in \mathbb{N}^2} (p+q) \delta'^2_{p,q} d_{p,q} ,\end{aligned}$$

with equality iff $|\delta'_{p,q}|$ has the same value wherever $(p+q) d_{p,q} \neq 0$.

To conclude, $\varepsilon_{T,F} \leq \eta_T \eta_F$ with equality iff $(\delta'_{p,q} = 0$ whenever $(p+q) d_{p,q} \neq 0$) or $(\delta'_{p,q}$ is symmetric in p, q and the model has only pure literals).

This means that in all models allowing non-pure literals (in particular the standard model having a 2D-Poisson $d_{p,q}$), $\varepsilon_{T,F} = \eta_T \eta_F$ iff $\delta_{p,q} = \frac{1}{2}$ whenever $(p+q) d_{p,q} \neq 0$.

Consequently, even in the distributional model, we encounter the very restrictive condition $\delta_{p,q} = \frac{1}{2}$ to make the Second Moment Method work.

Numerically, we found a critical ratio of 2.838, thus very slightly above the 2.833 obtained in the drawing model (cf section 8.6.1).

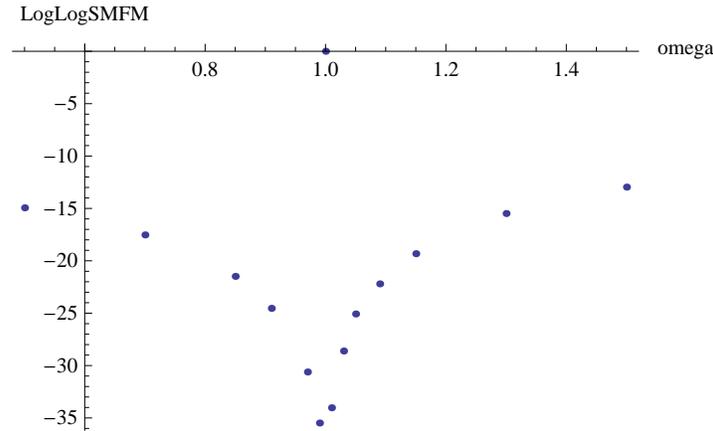


Figure 9.1: Setting $\delta_{p,q} = \frac{1}{1+\omega^{p-q}}$ makes the Second Moment Method work iff $\omega = 1$.

9.5.2 Attempts to evade the $\delta_{p,q} = \frac{1}{2}$ condition

The shape of $\delta_{p,q}$ on figures 9.2 and 9.3 suggested us that on real solutions $\delta_{p,q} = \frac{1}{1+\omega^{p-q}}$. So we tried to evade the $\omega = 1$ case.

We plotted $\ln \ln \frac{E_2}{E_1^2}$ for different values of ω , at a point satisfying constraints 8.6, 8.7, 9.2, 9.3, 9.4, 9.5, 9.6 and 9.7, for the best choice of the β_t 's that we found complying with constraint 9.1. The expected value is $-\infty$ iff $\frac{E_2}{E_1^2} = 1$. We set the ratio $c = 0.1$ (to be compared with 2.838, where $\omega = 1$ works). Only $\omega = 1$ seems to make $\frac{E_2}{E_1^2} = 1$, cf. figure 9.1.

9.6 Confrontation with Reality

We are going to do the same kinds of observations through SATLab as in section 8.7 in order to figure out why the Second Moment Method still fails to give high upper bounds in the distributional model.

9.6.1 Non-Independence of Values

We focus our attention on variables with T occurrences among which U are positive. We denote by d the average proportion of those variables assigned 1 by a solution and u the average proportion of those variables assigned 0 and 1 by a couple of distinct solutions. In a given sample of random solutions, we took all couples of different solutions and computed the following three quantities: d , $d_d = d(1-d)$ and u . At independence we should have $u = d_d$, which happens for $c = 2$ (cf. figure 9.2) but not for $c = 4$ (cf. figure 9.3).

Moreover we can see that d is almost linear in U when $c = 2$ but it curves when $c = 4$. Note also that the range of U may be strictly included in $[0 \dots T]$ (cf. figure 9.3). Determining the shape of d might help do better calculations of first and second moments, even though in section 9.5.2 we took $\delta_{p,q} = \frac{1}{1+\omega^{p-q}}$ but it could not make the Second Moment Method work. It is clear however that the condition $\delta_{p,q} = \frac{1}{2}$ we encountered in section 9.5.1 does not hold on real solutions.

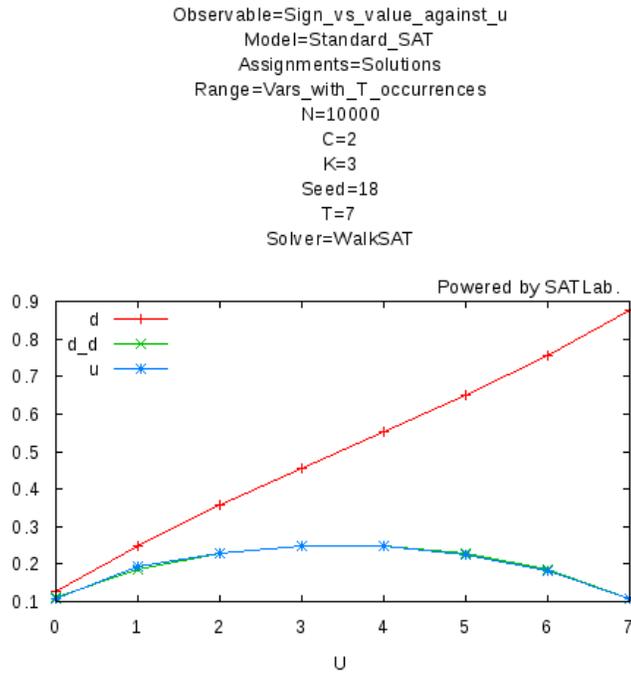


Figure 9.2: $\delta_{p,q}$ and $\mu_{p,q} = \delta_{p,q} (1 - \delta_{p,q})$ at $c = 2$.

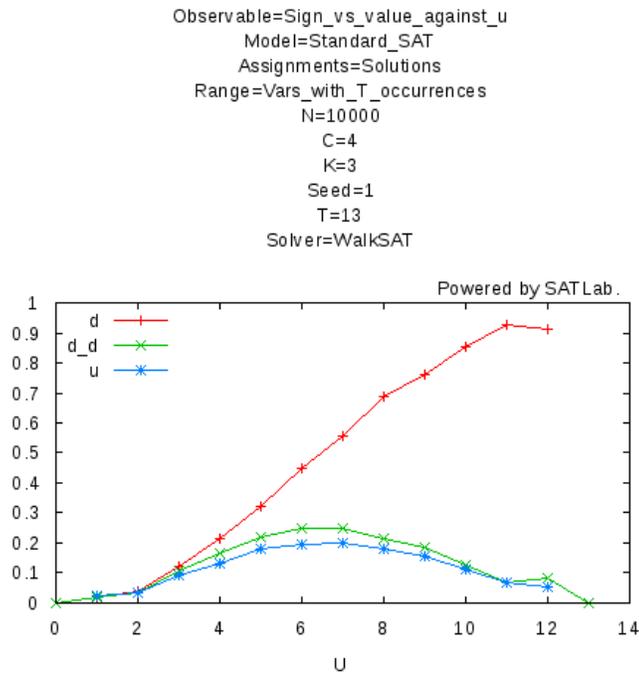


Figure 9.3: $\delta_{p,q}$ and $\mu_{p,q} \neq \delta_{p,q} (1 - \delta_{p,q})$ at $c = 4$.

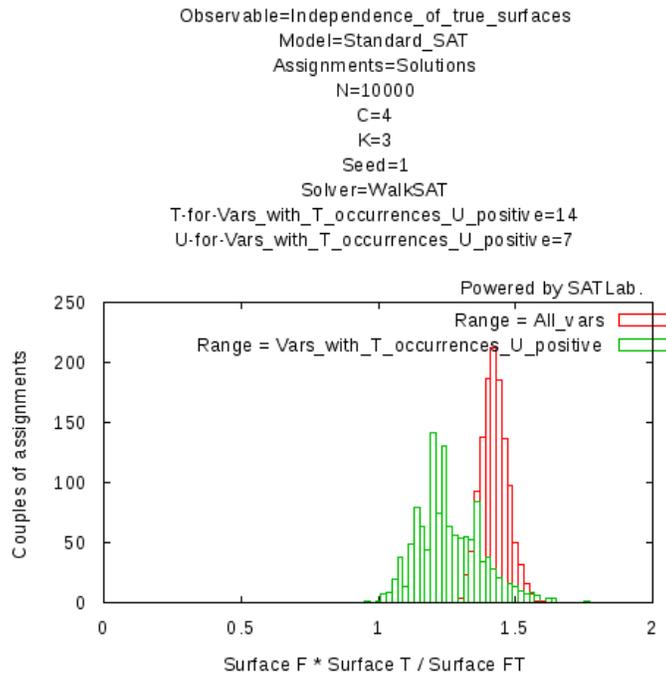


Figure 9.4: The distributional model slightly curbs the non-independence of surfaces.

9.6.2 Non-Independence of Surfaces

We perform the same experiment as in section 8.7.3, but we restrict surfaces to variables having T occurrences among which U are positive. On figure 9.4 we can see that there is still no independence of surfaces, although the restriction of the surfaces to these variables curbs the non-independence.

Conclusion on the Second Moment Method

“Personne n’est jamais assez fort pour ce calcul.” French folklore.

Contrary to the First Moment Method, which works more or less finely, the Second Moment Method will not always work. Furthermore, it is rather difficult to make it work, and we were able to make it work only under very artificial conditions with respect to reality. Moreover, even when it works, we have not been able to find strong lower bounds with it. We got stuck at 2.83 for 3 different models (standard drawing model, implicants and standard distributional model).

However we did not prove that it is impossible to find better lower bounds with our general framework, this is just numerical experiments. Moreover our framework may not be perfect, perhaps the parameters we consider are not relevant for the Second Moment Method, so there is still hope in making the Second Moment Method work and give higher lower bounds on the threshold of 3-SAT.

Bibliography

- [AM02] Dimitris Achlioptas and Cristopher Moore. The asymptotic order of the random k -SAT threshold. In *The 43rd Annual IEEE Symposium on Foundations of Computer Science*, pages 779–788. IEEE Computer Society, 2002.
- [AM09] Federico Ardila and Elitza N. Maneva. Pruning processes and a new characterization of convex geometries. *Discrete Mathematics*, 309(10):3083–3091, 2009.
- [Anb05] Anbulagan. Dew Satz: Integration of Lookahead Saturation with Restrictions into Satz. *SAT Competition*, pages 1–2, 2005.
- [AP04] Dimitris Achlioptas and Yuval Peres. The Threshold for Random k -SAT is $2^k \ln 2 - O(k)$. *JAMS: Journal of the American Mathematical Society*, 17:947–973, 2004.
- [ART06] Dimitris Achlioptas and F. Ricci-Tersenghi. On the solution-space geometry of random constraint satisfaction problems. In *STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 130–139, New York, NY, USA, 2006. ACM Press.
- [Bat08] Necdet Batir. Inequalities for the gamma function. *Archiv der Mathematik*, 91(6):554–563, 2008.
- [BB03] Béla Bollobás and Graham R. Brightwell. The number of k -SAT functions. *Random Structures and Algorithms*, 22(3):227–247, 2003.
- [BBC⁺01] Béla Bollobás, C. Borgs, Jennifer Chayes, J.H. Kim, and D.B. Wilson. The scaling window of the 2-SAT transition. *Random Structures and Algorithms*, 18(3):201–256, 2001.
- [BBL03] Béla Bollobás, G Brightwell, and I Leader. The Number of 2-SAT Functions. *Israel Journal of Mathematics*, 133:45–60, 2003.
- [BCM02] G Biroli, S Cocco, and Rémi Monasson. Phase transitions and complexity in computer science: an overview of the statistical physics approach to the random satisfiability problem. *Physica A: Statistical Mechanics and its Applications*, 306:381–394, April 2002.
- [BD99] Yacine Boufkhad and Olivier Dubois. Length of prime implicants and number of solutions of random CNF formulae. *Theoretical Computer Science*, 215(1-2):1–30, 1999.

- [BDIS05] Yacine Boufkhad, Olivier Dubois, Yannet Interian, and Bart Selman. Regular Random k -SAT: Properties of Balanced Formulas. *J. Autom. Reasoning*, 35(1-3):181–200, 2005.
- [BH10] Yacine Boufkhad and Thomas Hugel. Non Uniform Selection of Solutions for Upper Bounding the 3-SAT Threshold. In Ofer Strichman and Stefan Szeider, editors, *Theory and Applications of Satisfiability Testing - SAT 2010*, volume 6175 of *Lecture Notes in Computer Science*, pages 99–112, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
- [BHL⁺01] W. Barthel, A. K. Hartmann, M. Leone, F. Ricci-Tersenghi, M. Weigt, and Riccardo Zecchina. Hiding solutions in random satisfiability problems: A statistical mechanics approach. *Order A Journal On The Theory Of Ordered Sets And Its Applications*, page 5, 2001.
- [BMZ05] A. Braunstein, Marc Mézard, and Riccardo Zecchina. Survey propagation: An algorithm for satisfiability. *Random Structures and Algorithms*, 27(2):201–226, 2005.
- [Bop97] R B Boppana. The Average Sensitivity of Bounded-Depth Circuits. *Inf. Process. Lett.*, 63(5):257–261, 1997.
- [BT87] B. Bollobás and A. G. Thomason. Threshold functions. *Combinatorica*, 7(1):35–38, March 1987.
- [CD03] N. Creignou and H. Daudé. Generalized satisfiability problems: minimal elements and phase transitions. *Theoretical Computer Science*, 302(1-3):417–430, June 2003.
- [CD04] N. Creignou and H Daudé. Combinatorial sharpness criterion and phase transition classification for random CSPs. *Inf. Comput.*, 190(2):220–238, 2004.
- [CM97] S.A. Cook and D.G. Mitchell. Finding hard instances of the satisfiability problem: A survey. *Satisfiability Problem: Theory and Applications*, 35:1–18, 1997.
- [Coo71] S.A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the third annual ACM symposium on Theory of computing*, page 158. ACM, 1971.
- [CR92] Vašek Chvátal and B. Reed. Mick gets some (the odds are on his side)[satisfiability]. In *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science*, pages 620–627, 1992.
- [DB97] Olivier Dubois and Yacine Boufkhad. A General Upper Bound for the Satisfiability Threshold of Random r -SAT Formulae. *J. Algorithms*, 24(2):395–420, 1997.
- [DBM00] Olivier Dubois, Yacine Boufkhad, and J. Mandler. Typical random 3-SAT formulae and the satisfiability threshold. In *Proceedings of the eleventh annual ACM-SIAM symposium on Discrete algorithms*, pages 126–127. Society for Industrial and Applied Mathematics, 2000.
- [DBM03] Olivier Dubois, Yacine Boufkhad, and J. Mandler. Typical random 3-SAT formulae and the satisfiability threshold. *Electronic Colloquium on Computational Complexity (ECCC)*, pages TR03–007, 2003.
- [DD03] Gilles Dequen and Olivier Dubois. kcnfs: An Efficient Solver for Random k -SAT Formulae. In Enrico Giunchiglia and Armando Tacchella, editors, *SAT*, pages 486–501. Springer, 2003.

- [DDL⁺06] R Di Cosmo, B Durak, X Leroy, F Mancinelli, and J Vouillon. Maintaining large software distributions: new challenges from the FOSS era. In *Proceedings of the FRCSS 2006 workshop*, 2006.
- [DKMPG09] J. Díaz, Lefteris M. Kirousis, D. Mitsche, and X. Pérez-Giménez. On the satisfiability threshold of formulas with three literals per clause. *Theoretical Computer Science*, 410(30-32):2920–2934, 2009.
- [DM02] Olivier Dubois and J. Mandler. The 3-XORSAT threshold. *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science, FOCS'2002 (Vancouver, BC, Canada, November 16-19, 2002)*, pages 769–778, 2002.
- [ED95] A. El Maftouhi and W. Fernandez De La Vega. On Random 3-sat. *Combinatorics, Probability and Computing*, 4(03), September 1995.
- [FB99] Ehud Friedgut and J. Bourgain. Sharp thresholds of graph properties, and the k-sat problem. *Journal of the American Mathematical Society*, 12(4):1017–1054, 1999.
- [FP83] J. Franco and M. Paull. Probabilistic analysis of the Davis Putnam procedure for solving the satisfiability problem. *Discrete Applied Mathematics*, 5:77–87, 1983.
- [FS97] John V Franco and Ramjee P Swaminathan. Average Case Results for Satisfiability Algorithms Under the Random-Clause-Width Model. *Ann. Math. Artif. Intell.*, 20(1-4):357–391, 1997.
- [Goe96] A Goerdt. A Threshold for Unsatisfiability. *J. Comput. Syst. Sci.*, 53(3):469–486, 1996.
- [Gol79] Allen Goldberg. On the Complexity of the Satisfiability Problem, 1979.
- [HB10] Thomas Hugel and Yacine Boufkhad. Non Uniform Selection of Solutions for Upper Bounding the 3-SAT Threshold. *CoRR*, abs/1002.1, 2010.
- [HS03] M.T. Hajiaghayi and G.B. Sorkin. The satisfiability threshold of random 3-SAT is at least 3.52. *IBM Research Report RC22942*, 2003.
- [Hug10] Thomas Hugel. SATLab. <http://www.pratum.org/satlab/>, 2010.
- [IBP05] Gabriel Istrate, Stefan Boettcher, and Allon G. Percus. Spines of random constraint satisfaction problems: definition and connection with computational complexity. *Annals of Mathematics and Artificial Intelligence*, 44(4):353–372, August 2005.
- [JMS07] Haixia Jia, Christopher Moore, and Doug Strain. Generating Hard Satisfiable Formulas by Hiding Solutions Deceptively. *J. Artif. Intell. Res. (JAIR)*, 28:107–118, 2007.
- [KKK97] Lefteris M. Kirousis, Evangelos Kranakis, and Danny Krizanc. A Better Upper Bound for the Unsatisfiability Threshold. In B. Du, J. Gu, and P. M. Pardalos, editors, *Satisfiability Problem: Theory and Applications, DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, volume 00, pages 643–648. American Mathematical Society, 1997.
- [KKKS98] Lefteris M. Kirousis, Evangelos Kranakis, Danny Krizanc, and Yannis C. Stamatiou. Approximating the unsatisfiability threshold of random formulas. *Random Structures and Algorithms*, 12(3):253–269, 1998.

- [KKL88] Jeff Kahn, Gil Kalai, and Nathan Linial. The Influence of Variables on Boolean Functions (Extended Abstract). In *FOCS*, pages 68–80. IEEE Computer Society Press, 1988.
- [KKL06] A.C. Kaporis, Lefteris M. Kirousis, and E.G. Lalas. The probabilistic analysis of a greedy satisfiability algorithm. *Random Structures and Algorithms*, 28(4):444–480, 2006.
- [KMPS95] A. Kamath, R. Motwani, K. Palem, and P. Spirakis. Tail bounds for occupancy and the satisfiability threshold conjecture. *Random Structures and Algorithms*, 7(1):59–80, 1995.
- [KS06] G Kalai and M Safra. Threshold Phenomena and Influence. In Percus, Istrate, and Moore, editors, *Computational Complexity and Statistical Physics*, pages 25–60. Oxford University Press, 2006.
- [Lev73] Leonid A. Levin. Universal search problems. *Problems of Information Transmission*, 9(3):265–266, 1973.
- [MAJ05] Christopher Moore, Dimitris Achlioptas, and Haixia Jia. Hiding Satisfying Assignments: Two are Better than One. *Journal of Artificial Intelligence Research*, 24:623–639, 2005.
- [Mar74] G.A. Margulis. Probabilistic Characteristics of Graphs with Large Connectivity. *Problems of Information Transmission*, 10(2):174–179, 1974.
- [MBC⁺06] F Mancinelli, J Boender, R Di Cosmo, J Vouillon, B Durak, X Leroy, and R Treinen. Managing the Complexity of Large Free and Open Source Package-Based Software Distributions. In *ASE*, pages 199–208. IEEE Computer Society, 2006.
- [MMW07] Elitza N. Maneva, Elchanan Mossel, and Martin J. Wainwright. A new look at survey propagation and its generalizations. *Journal of the ACM*, 54(4):17, July 2007.
- [MMZ06] Stephan Mertens, Marc Mézard, and Riccardo Zecchina. Threshold values of Random K-SAT from the cavity method, 2006.
- [Mor07] Thierry Mora. *Géométrie et Inférence dans l’optimisation et en théorie de l’information*. PhD thesis, Université Paris Sud 11, 2007.
- [MS08] Elitza N. Maneva and Alistair Sinclair. On the satisfiability threshold and clustering of solutions of random 3-SAT formulas. *Theoretical Computer Science*, 407(1-3):359–369, 2008.
- [MSL92] D.G. Mitchell, Bart Selman, and H. Levesque. Hard and easy distributions of SAT problems. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 459–465, 1992.
- [MZK⁺99] Rémi Monasson, Riccardo Zecchina, S. Kirkpatrick, Bart Selman, and L. Troyansky. Determining computational complexity from characteristic ‘phase transitions’. *Nature*, 400(6740):133–137, 1999.
- [Rus78] Lucio Russo. A note on percolation. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 43(1):39–48, 1978.
- [SA05] John Slaney and Anbulagan. Lookahead saturation with restriction for SAT. In Peter van Beek, editor, *Principles and Practice of Constraint Programming-CP*, pages 727–731, Sitges, Spain, 2005. Springer.

- [Sch78] Thomas J. Schaefer. *The complexity of satisfiability problems*. ACM Press, New York, New York, USA, 1978.
- [SD89] J.C. Simon and Olivier Dubois. Number of Solutions of Satisfiability Instance - Applications to Knowledge Bases. *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)*, 3(1):53–65, 1989.
- [Sem07] Guilhem Semerjian. On the Freezing of Variables in Random Constraint Satisfaction Problems. *Journal of Statistical Physics*, 130(2):251–293, October 2007.
- [SKC95] Bart Selman, Henry Kautz, and Bram Cohen. Local Search Strategies for Satisfiability Testing. In Michael Trick and David Stifler Johnson, editors, *Proceedings of the Second {DIMACS} Challenge on Cliques, Coloring, and Satisfiability*, 1995.
- [Tal94] M Talagrand. On Russo’s approximate zero-one law. *The Annals of Probability*, 1994.
- [Tem93] N.M. Temme. Asymptotic estimates of Stirling numbers. *Stud. Appl. Math*, 89(3):233–243, 1993.

Maximization of $\ln F$ (Non-Uniform Selection of Solutions)

Let us recall (cf. section 6.3.3) that under the following constraints:

$$\beta_1 + \beta_2 + \beta_3 = 1 \quad (\text{A.1})$$

$$\sum_{(i,j,k,l,m) \in Q_{p,q}} \pi_{i,j,k,l,m,1} + \sum_{(i,j,k,l,m) \in Q_{q,p}} \pi_{i,j,k,l,m,0} = d_{p,q} \quad (\text{A.2})$$

$$H_1 + H_2 + H_3 = \sum_{(p,q) \in \mathcal{H}} (p+q) d_{p,q} \quad (\text{A.3})$$

$$\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} i \pi_{i,j,k,l,m,v} + H_1 = \beta_1 c \quad (\text{A.4})$$

$$\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} j \pi_{i,j,k,l,m,v} + H_2 = 2\beta_2 c \quad (\text{A.5})$$

$$\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} k \pi_{i,j,k,l,m,v} + H_3 = 3\beta_3 c \quad (\text{A.6})$$

$$\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} l \pi_{i,j,k,l,m,v} = 2\beta_1 c \quad (\text{A.7})$$

$$\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} m \pi_{i,j,k,l,m,v} = \beta_2 c \quad (\text{A.8})$$

we want to maximize the function:

$$F = 2^\tau \frac{H^H}{H_1^{H_1} H_2^{H_2} H_3^{H_3}} \left(\frac{1}{3} (2\beta_1)^{\beta_1} (2\beta_2)^{\beta_2} (3\beta_3)^{\beta_3} \right)^{2c} \prod_{(p,q) \in \mathcal{L}} d_{p,q}^{d_{p,q}} \prod_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} \left(\omega_{i,j,k,l,m,v} \frac{\binom{i+j+k}{i,j,k} \binom{l+m}{l,m}}{\pi_{i,j,k,l,m,v}} \right)^{\pi_{i,j,k,l,m,v}}. \quad (\text{A.9})$$

on variables $\left(\pi_{i,j,k,l,m,v} \right)_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}}$, (H_1, H_2, H_3) , $(\beta_1, \beta_2, \beta_3)$.

To perform such a maximization we use the standard technique of Lagrange multipliers.

A.1 Resolution of the Global Lagrange Multipliers Problem

Elimination of redundant constraints. The first thing to do is to remove redundant constraints. It appears that e.g. constraint (A.6) is redundant with constraints (A.4), (A.5), (A.7), (A.8), because summing these 5 equations and using the previous ones (A.1), (A.3), (A.2) gives a tautology:

$$\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} (i+j+k+l+m) \pi_{i,j,k,l,m,v} + H_1 + H_2 + H_3 = 3(\beta_1 + \beta_2 + \beta_3)c$$

which is equivalent to

$$\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q}}} \left((i+j+k) \pi_{i,j,k,l,m,1} + (l+m) \pi_{i,j,k,l,m,1} \right) + \sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{q,p}}} \left((i+j+k) \pi_{i,j,k,l,m,0} + (l+m) \pi_{i,j,k,l,m,0} \right) + H = 3c$$

which is equivalent to

$$\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q}}} \left(p \pi_{i,j,k,l,m,1} + q \pi_{i,j,k,l,m,1} \right) + \sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{q,p}}} \left(q \pi_{i,j,k,l,m,0} + p \pi_{i,j,k,l,m,0} \right) + H = 3c$$

which is equivalent to

$$\sum_{(p,q) \in \mathcal{L}} (p+q) \left(\sum_{(i,j,k,l,m) \in Q_{p,q}} \pi_{i,j,k,l,m,1} + \sum_{(i,j,k,l,m) \in Q_{q,p}} \pi_{i,j,k,l,m,0} \right) + H = 3c$$

which is equivalent to

$$\sum_{(p,q) \in \mathcal{L}} (p+q) d_{p,q} + \sum_{(p,q) \in \mathcal{H}} (p+q) d_{p,q} = 3c$$

which was a requirement we made on $(d_{p,q})$ (see section (1.1.4)).

Thus we get rid of constraint (A.6) and there remain 7 constraints.

Definition of the Lagrangian.

$$\begin{aligned}
\Lambda = & \tau \ln 2 + H \ln H - H_1 \ln \left(\frac{H_1}{e} \right) - H_2 \ln \left(\frac{H_2}{e} \right) - H_3 \ln \left(\frac{H_3}{e} \right) - H \\
& + \sum_{(p,q) \in \mathcal{L}} d_{p,q} \ln d_{p,q} + \sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} \pi_{i,j,k,l,m,v} \ln \left(\omega_{i,j,k,l,m,v} \frac{e^{(i+j+k)(l+m)}}{\pi_{i,j,k,l,m,v}} \right) - 1 \\
& - 2c \ln 3 + 2c\beta_1 \ln \left(2 \frac{\beta_1}{e} \right) + 2c\beta_2 \ln \left(2 \frac{\beta_2}{e} \right) + 2c\beta_3 \ln \left(3 \frac{\beta_3}{e} \right) + 2c \\
& + (2c \ln b) (\beta_1 + \beta_2 + \beta_3 - 1) \\
& + \sum_{(p,q) \in \mathcal{L}} (\ln r_{p,q}) \left(\sum_{(i,j,k,l,m) \in Q_{p,q}} \pi_{i,j,k,l,m,1} + \sum_{(i,j,k,l,m) \in Q_{q,p}} \pi_{i,j,k,l,m,0} - d_{p,q} \right) \\
& + (\ln h) \left(H_1 + H_2 + H_3 - \sum_{(p,q) \in \mathcal{H}} (p+q) d_{p,q} \right) \\
& + (2 \ln x_1) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} i \pi_{i,j,k,l,m,v} + H_1 - \beta_1 c \right) \\
& + (\ln x_2) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} j \pi_{i,j,k,l,m,v} + H_2 - 2\beta_2 c \right) \\
& + (\ln y_1) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} l \pi_{i,j,k,l,m,v} - 2\beta_1 c \right) \\
& + (2 \ln y_2) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} m \pi_{i,j,k,l,m,v} - \beta_2 c \right) .
\end{aligned}$$

Derivatives with Respect to $\pi_{i,j,k,l,m,v}$.

$$\begin{aligned}\frac{\partial \Lambda}{\partial \pi_{i,j,k,l,m,1}} &= \ln \omega_{i,j,k,l,m,1} + \ln \left(\binom{i+j+k}{i,j,k} \binom{l+m}{l,m} \right) - \ln \pi_{i,j,k,l,m,v} + \ln r_{i+j+k,l+m} \\ &\quad + 2i \ln x_1 + j \ln x_2 + l \ln y_1 + 2m \ln y_2 ; \\ \frac{\partial \Lambda}{\partial \pi_{i,j,k,l,m,0}} &= \ln \omega_{i,j,k,l,m,0} + \ln \left(\binom{i+j+k}{i,j,k} \binom{l+m}{l,m} \right) - \ln \pi_{i,j,k,l,m,v} + \ln r_{l+m,i+j+k} \\ &\quad + 2i \ln x_1 + j \ln x_2 + l \ln y_1 + 2m \ln y_2 .\end{aligned}$$

Canceling out these derivatives yields:

$$\pi_{i,j,k,l,m,1} = \omega_{i,j,k,l,m,1} \binom{i+j+k}{i,j,k} \binom{l+m}{l,m} r_{i+j+k,l+m} x_1^{2i} x_2^j y_1^l y_2^{2m} ; \quad (\text{A.10})$$

$$\pi_{i,j,k,l,m,0} = \omega_{i,j,k,l,m,0} \binom{i+j+k}{i,j,k} \binom{l+m}{l,m} r_{l+m,i+j+k} x_1^{2i} x_2^j y_1^l y_2^{2m} . \quad (\text{A.11})$$

The $r_{p,q}$ constraints become:

$$\begin{aligned}&\sum_{(i,j,k,l,m) \in Q_{p,q}} \omega_{i,j,k,l,m,1} \binom{i+j+k}{i,j,k} \binom{l+m}{l,m} r_{i+j+k,l+m} x_1^{2i} x_2^j y_1^l y_2^{2m} \\ &+ \sum_{(i,j,k,l,m) \in Q_{q,p}} \omega_{i,j,k,l,m,0} \binom{i+j+k}{i,j,k} \binom{l+m}{l,m} r_{l+m,i+j+k} x_1^{2i} x_2^j y_1^l y_2^{2m} = d_{p,q} .\end{aligned}$$

Let us introduce

$$\begin{aligned}A_{p,q} &= \sum_{(i,j,k,l,m) \in Q_{p,q}} \omega_{i,j,k,l,m,1} \binom{p}{i,j,k} \binom{q}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \\ &\quad + \sum_{(i,j,k,l,m) \in Q_{q,p}} \omega_{i,j,k,l,m,0} \binom{q}{i,j,k} \binom{p}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} .\end{aligned} \quad (\text{A.12})$$

We have:

$$r_{p,q} A_{p,q} = d_{p,q}$$

i.e.

$$r_{p,q} = \frac{d_{p,q}}{A_{p,q}} . \quad (\text{A.13})$$

Thus

$$\begin{aligned}\pi_{i,j,k,l,m,1} &= \omega_{i,j,k,l,m,1} \binom{i+j+k}{i,j,k} \binom{l+m}{l,m} \frac{d_{i+j+k,l+m}}{A_{i+j+k,l+m}} x_1^{2i} x_2^j y_1^l y_2^{2m} ; \\ \pi_{i,j,k,l,m,0} &= \omega_{i,j,k,l,m,0} \binom{i+j+k}{i,j,k} \binom{l+m}{l,m} \frac{d_{l+m,i+j+k}}{A_{l+m,i+j+k}} x_1^{2i} x_2^j y_1^l y_2^{2m} .\end{aligned}$$

Derivatives with Respect to β_t .

$$\begin{aligned}\frac{\partial \Lambda}{\partial \beta_1} &= 2c \ln 2 + 2c \ln \beta_1 + 2c \ln b - 2c \ln x_1 - 2c \ln y_1 ; \\ \frac{\partial \Lambda}{\partial \beta_2} &= 2c \ln 2 + 2c \ln \beta_2 + 2c \ln b - 2c \ln x_2 - 2c \ln y_2 ; \\ \frac{\partial \Lambda}{\partial \beta_3} &= 2c \ln 3 + 2c \ln \beta_3 + 2c \ln b .\end{aligned}$$

Canceling out these derivatives yields:

$$\begin{aligned}\beta_1 &= \frac{x_1 y_1}{2b} ; \\ \beta_2 &= \frac{x_2 y_2}{2b} ; \\ \beta_3 &= \frac{1}{3b} .\end{aligned}$$

Constraint (A.1) yields:

$$b = \frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} + \frac{1}{3} .$$

Derivatives with Respect to H_t .

$$\begin{aligned}\frac{\partial \Lambda}{\partial H_1} &= -\ln H_1 + \ln h + 2 \ln x_1 ; \\ \frac{\partial \Lambda}{\partial H_2} &= -\ln H_2 + \ln h + \ln x_2 ; \\ \frac{\partial \Lambda}{\partial H_3} &= -\ln H_3 + \ln h .\end{aligned}$$

Canceling out these derivatives yields:

$$\begin{aligned}H_1 &= h x_1^2 ; \\ H_2 &= h x_2 ; \\ H_3 &= h .\end{aligned}$$

Constraint (A.3) yields:

$$h (x_1^2 + x_2 + 1) = H$$

i.e.

$$h = \frac{H}{x_1^2 + x_2 + 1} .$$

Thus:

$$\begin{aligned} H_1 &= \frac{Hx_1^2}{x_1^2 + x_2 + 1} ; \\ H_2 &= \frac{Hx_2}{x_1^2 + x_2 + 1} ; \\ H_3 &= \frac{H}{x_1^2 + x_2 + 1} . \end{aligned}$$

Further Simplifications.

– 1st Moment:

$$\begin{aligned} F &= \prod_{(p,q) \in \mathcal{L}} d_{p,q}^{d_{p,q}} \prod_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} \left(\omega_{i,j,k,l,m,v} \frac{\binom{i+j+k}{i,j,k} \binom{l+m}{l,m}}{\tau_{i,j,k,l,m,v}} \right)^{\tau_{i,j,k,l,m,v}} \\ &\cdot 2^\tau \frac{H^H}{H_1^{H_1} H_2^{H_2} H_3^{H_3}} \left(\frac{1}{3} (2\beta_1)^{\beta_1} (2\beta_2)^{\beta_2} (3\beta_3)^{\beta_3} \right)^{2c} \end{aligned}$$

i.e.

$$\begin{aligned} F &= \prod_{(p,q) \in \mathcal{L}} d_{p,q}^{d_{p,q}} \prod_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q}}} \left(\frac{A_{i+j+k,l+m}}{d_{i+j+k,l+m} x_1^{2i} x_2^j y_1^l y_2^m} \right)^{\tau_{i,j,k,l,m,1}} \\ &\cdot \prod_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{q,p}}} \left(\frac{A_{l+m,i+j+k}}{d_{l+m,i+j+k} x_1^{2i} x_2^j y_1^l y_2^m} \right)^{\tau_{i,j,k,l,m,0}} \\ &\cdot 2^\tau \frac{H^H}{(hx_1^2)^{H_1} (hx_2)^{H_2} (h)^{H_3}} \left(\frac{1}{3b} (x_1 y_1)^{\beta_1} (x_2 y_2)^{\beta_2} \right)^{2c} \end{aligned}$$

i.e.

$$\begin{aligned} F &= \prod_{(p,q) \in \mathcal{L}} \left(d_{p,q}^{d_{p,q}} \prod_{(i,j,k,l,m) \in Q_{p,q}} \left(\frac{A_{p,q}}{d_{p,q} x_1^{2i} x_2^j y_1^l y_2^m} \right)^{\tau_{i,j,k,l,m,1}} \right) \\ &\cdot \prod_{(p,q) \in \mathcal{L}} \left(\prod_{(i,j,k,l,m) \in Q_{q,p}} \left(\frac{A_{p,q}}{d_{p,q} x_1^{2i} x_2^j y_1^l y_2^m} \right)^{\tau_{i,j,k,l,m,0}} \right) \\ &\cdot 2^\tau H^H h^{-(H_1+H_2+H_3)} x_1^{-2H_1} x_2^{-H_2} \left(\frac{1}{3b} x_1^{\beta_1} y_1^{\beta_1} x_2^{\beta_2} y_2^{\beta_2} \right)^{2c} \end{aligned}$$

i.e.

$$\begin{aligned}
F &= 2^\tau \left(\frac{H}{h}\right)^H \prod_{(p,q) \in \mathcal{L}} \left(d_{p,q}^{d_{p,q}} \left(\frac{A_{p,q}}{d_{p,q}}\right)^{\sum_{(i,j,k,l,m) \in Q_{p,q}} \pi_{i,j,k,l,m,1} + \sum_{(i,j,k,l,m) \in Q_{q,p}} \pi_{i,j,k,l,m,0}} \right) \\
&\quad \cdot x_1^{-2 \left(\sum_{\substack{(i,j,k,l,m) \in Q_n \\ v \in \{0,1\}}} i \pi_{i,j,k,l,m,v} + H_1 \right)} x_2^{-2 \left(\sum_{\substack{(i,j,k,l,m) \in Q_n \\ v \in \{0,1\}}} j \pi_{i,j,k,l,m,v} + H_2 \right)} \\
&\quad \cdot y_1^{-\sum_{\substack{(i,j,k,l,m) \in Q_n \\ v \in \{0,1\}}} l \pi_{i,j,k,l,m,v}} y_2^{-2 \sum_{\substack{(i,j,k,l,m) \in Q_n \\ v \in \{0,1\}}} m \pi_{i,j,k,l,m,v}} \\
&\quad \cdot \left(\frac{1}{3b} x_1^{\beta_1} y_1^{\beta_1} x_2^{\beta_2} y_2^{\beta_2} \right)^{2c}
\end{aligned}$$

i.e.

$$\begin{aligned}
F &= 2^\tau \left(\frac{H}{h}\right)^H \prod_{(p,q) \in \mathcal{L}} \left(d_{p,q}^{d_{p,q}} \left(\frac{A_{p,q}}{d_{p,q}}\right)^{d_{p,q}} \right) x_1^{-2\beta_1 c} x_2^{-2\beta_2 c} y_1^{-2\beta_1 c} y_2^{-2\beta_2 c} \\
&\quad \cdot \left(\frac{1}{3b} x_1^{\beta_1} y_1^{\beta_1} x_2^{\beta_2} y_2^{\beta_2} \right)^{2c}
\end{aligned}$$

i.e.

$$F = 2^\tau (x_1^2 + x_2 + 1)^H \prod_{(p,q) \in \mathcal{L}} \left(A_{p,q}^{d_{p,q}} \right) \left(\frac{3x_1 y_1}{2} + \frac{3x_2 y_2}{2} + 1 \right)^{-2c}.$$

– Remaining constraints:

$$\begin{aligned}
& \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,k,l,m) \in Q_{p,q}} i \binom{p}{i,j,k} \binom{q}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,k,l,m,1} \\
& + \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,k,l,m) \in Q_{q,p}} i \binom{q}{i,j,k} \binom{p}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,k,l,m,0} + \frac{Hx_1^2}{x_1^2 + x_2 + 1} \\
& = \frac{x_1 y_1 c}{2 \left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} + \frac{1}{3} \right)} ; \\
& \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,k,l,m) \in Q_{p,q}} j \binom{p}{i,j,k} \binom{q}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,k,l,m,1} \\
& + \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,k,l,m) \in Q_{q,p}} j \binom{q}{i,j,k} \binom{p}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,k,l,m,0} + \frac{Hx_2}{x_1^2 + x_2 + 1} \\
& = \frac{x_2 y_2 c}{\left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} + \frac{1}{3} \right)} ; \\
& \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,k,l,m) \in Q_{p,q}} l \binom{p}{i,j,k} \binom{q}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,k,l,m,1} \\
& + \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,k,l,m) \in Q_{q,p}} l \binom{q}{i,j,k} \binom{p}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,k,l,m,0} \\
& = \frac{x_1 y_1 c}{\left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} + \frac{1}{3} \right)} ; \\
& \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,k,l,m) \in Q_{p,q}} m \binom{p}{i,j,k} \binom{q}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,k,l,m,1} \\
& + \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,k,l,m) \in Q_{q,p}} m \binom{q}{i,j,k} \binom{p}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,k,l,m,0} \\
& = \frac{x_2 y_2 c}{2 \left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} + \frac{1}{3} \right)} ,
\end{aligned}$$

that is:

$$\begin{aligned}
\sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \frac{x_1}{2} \frac{\partial A_{p,q}}{\partial x_1} + \frac{Hx_1^2}{x_1^2 + x_2 + 1} &= \frac{x_1 y_1 c}{2 \left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} + \frac{1}{3} \right)} ; \\
\sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} x_2 \frac{\partial A_{p,q}}{\partial x_2} + \frac{Hx_2}{x_1^2 + x_2 + 1} &= \frac{x_2 y_2 c}{\left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} + \frac{1}{3} \right)} ; \\
\sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} y_1 \frac{\partial A_{p,q}}{\partial y_1} &= \frac{x_1 y_1 c}{\left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} + \frac{1}{3} \right)} ; \\
\sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \frac{y_2}{2} \frac{\partial A_{p,q}}{\partial y_2} &= \frac{x_2 y_2 c}{2 \left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} + \frac{1}{3} \right)} .
\end{aligned}$$

Table A.1: Solutions to the Lagrange multipliers problem of the first moment.

model	standard	balanced signs	balanced occurrences	balanced signs and occurrences
our method $(\alpha\rho_{j,l} + \rho_{k,m}, v) > (0, 0)$	4.500	3.509	4.623	3.546
our α	2.00	1.01	2.01	1.01
x_1	1.0083	1.47787	1.01694	1.57726
x_2	2.06625	3.09005	2.08256	3.38506
y_1	2.18256	3.27457	2.19038	3.51076
y_2	1.01253	1.02742	1.01221	1.045
β_1	0.44373	0.557479	0.445306	0.568436
β_2	0.421847	0.365723	0.421418	0.363128
β_3	0.134422	0.0767974	0.133276	0.0684362

Then we introduce $Z = \prod_{(p,q) \in \mathcal{L}} A_{p,q}^{d_{p,q}}$ and $Y = \ln Z$:

$$\begin{aligned} \frac{\partial Y}{\partial x_1} + \frac{2Hx_1}{x_1^2 + x_2 + 1} &= \frac{y_1 c}{\left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} + \frac{1}{3}\right)} ; \\ \frac{\partial Y}{\partial x_2} + \frac{H}{x_1^2 + x_2 + 1} &= \frac{y_2 c}{\left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} + \frac{1}{3}\right)} ; \\ \frac{\partial Y}{\partial y_1} &= \frac{x_1 c}{\left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} + \frac{1}{3}\right)} ; \\ \frac{\partial Y}{\partial y_2} &= \frac{x_2 c}{\left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} + \frac{1}{3}\right)} . \end{aligned}$$

To solve these equations we used Mathematica. The bound we obtained for c are summed up in table (A.1).

A.2 Inspection of the Boundary of \mathcal{P}

In order to justify the use of Lagrange multipliers technique we must show that the function $\ln F$ does not maximize on the boundary of the polytope of constraints; to do so we show that starting at a boundary point there is always a “good” direction inside the polytope which makes $\ln F$ greater. The boundary of \mathcal{P} is reached when one of the variables is at 0. We want to

Table A.2: Summary of bounds on β_t .

model	standard	balanced signs	balanced occurrences	balanced signs and occurrences
our method $(\alpha\rho_{j,l} + \rho_{k,m}, v) > (0,0)$	4.500	3.509	4.623	3.546
our α	2.00	1.01	2.01	1.01
our bounds for β_1	$0.177 < \beta_1 < 0.912$	$0.428 < \beta_1 < 0.786$	$0.182 < \beta_1 < 0.909$	$0.5 < \beta_1 < 0.75$
(our bounds for β_2 - deductible from above)	$0 \leq \beta_2 < 0.823$	$0 \leq \beta_2 < 0.572$	$0 \leq \beta_2 < 0.818$	$0 \leq \beta_2 < 0.5$
our bounds for β_3	$0 \leq \beta_3 < 0.412$	$0 \leq \beta_3 < 0.286$	$0 \leq \beta_3 < 0.409$	$0 \leq \beta_3 < 0.25$

be sure that F cannot be maximized by such a configuration. Remember that

$$\begin{aligned} \ln F = & \tau \ln 2 + H \ln H - H_1 \ln \left(\frac{H_1}{e} \right) - H_2 \ln \left(\frac{H_2}{e} \right) - H_3 \ln \left(\frac{H_3}{e} \right) - H \\ & + \sum_{(p,q) \in \mathcal{L}} d_{p,q} \ln d_{p,q} + \sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} \pi_{i,j,k,l,m,v} \ln \left(\omega_{i,j,k,l,m,v} \frac{\binom{i+j+k}{i,j,k} \binom{l+m}{l,m}}{e^{\pi_{i,j,k,l,m,v}}} \right) + 1 \\ & - 2c \ln 3 + 2c\beta_1 \ln \left(2 \frac{\beta_1}{e} \right) + 2c\beta_2 \ln \left(2 \frac{\beta_2}{e} \right) + 2c\beta_3 \ln \left(3 \frac{\beta_3}{e} \right) + 2c . \end{aligned}$$

If we increase an H_t or a $\pi_{i,j,k,l,m,v}$ from 0 to a small $\zeta > 0$ and we change any other nonzero variables, then the variation of $\ln F$ is $f = -\zeta \ln \zeta + \Theta(\zeta)$ is such that $\frac{f}{\zeta} = -\ln \zeta + \Theta(1) \rightarrow +\infty$, so $\ln F$ must increase; but what if we increase a β_t from 0 to a $\zeta > 0$? Then $\frac{f}{\zeta} = +\ln \zeta + \Theta(1) \rightarrow -\infty$. Thus the problem at the boundary of \mathcal{P} comes from the β_t . The technique will be the same as Dubois, Boufkhad & Mandler - 2003 [DBM03] or Díaz, Kirousis, Mitsche & Pérez-Giménez - 2009 [DKMPG09]: make a small move in a well chosen direction in order to circumvent the negative side-effect of increasing a β_t which is at 0. Such a direction will be referred to as an *increasing direction*. However we must ensure that such a direction is indeed in the polytope \mathcal{P} . Note that in case we find the direction by pointing towards another point in \mathcal{P} , this property results from the convexity of \mathcal{P} .

We used *Mathematica* to minimize and maximize β_1 under the above constraints in each model and our corresponding weighting scheme ; the precise bounds we obtained for β_1 in each model are summed up in table (A.2). Noteworthy is the fact that β_1 can be neither 0 nor 1 (thus we can have neither $\beta_1 = 0$ nor $\beta_2 = \beta_3 = 0$).

1. case where $\beta_2 = 0$: then $H_2 = 0$ and $\pi_{i,j,k,l,m,v} = 0$ unless $j = m = 0$; we call these variables *forced* as did Díaz, Kirousis, Mitsche & Pérez-Giménez - 2009 [DKMPG09]; moreover in the models where there are no heavy variables we consider variables H_t to be forced to 0 as well.

- sub-case where there is an unforced variable at zero: we find a feasible point where $\beta_2 = 0$ and all unforced variables are nonzero. Then a move towards this point gives an increasing direction (because $\beta_1 > 0$ and $\beta_3 > 0$). To find such a point, we use the Lagrange multipliers method, as follows:
- Definition of the Lagrangian:

$$\begin{aligned}
\Lambda = & \tau \ln 2 + H \ln H - H_1 \ln \left(\frac{H_1}{e} \right) - H_3 \ln \left(\frac{H_3}{e} \right) - H \\
& + \sum_{(p,q) \in \mathcal{L}} d_{p,q} \ln d_{p,q} + \sum_{\substack{(p,q) \in \mathcal{L} \\ (i,0,k,l,0) \in Q_{p,q} \\ v \in \{0,1\}}} \pi_{i,0,k,l,0,v} \ln \left(\omega_{i,0,k,l,0,v} \frac{\binom{i+k}{i,k}}{e^{\pi_{i,0,k,l,0,v}}} \right) + 1 \\
& - 2c \ln 3 + 2c\beta_1 \ln \left(2 \frac{\beta_1}{e} \right) + 2c\beta_3 \ln \left(3 \frac{\beta_3}{e} \right) + 2c \\
& + (2c \ln b) (\beta_1 + \beta_3 - 1) \\
& + \sum_{(p,q) \in \mathcal{L}} (\ln r_{p,q}) \left(\sum_{(i,0,k,l,0) \in Q_{p,q}} \pi_{i,j,k,l,m,1} + \sum_{(i,0,k,l,0) \in Q_{q,p}} \pi_{i,j,k,l,m,0} - d_{p,q} \right) \\
& + (\ln h) (H_1 + H_3 - H) \\
& + (2 \ln x_1) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,0,k,l,0) \in Q_{p,q} \\ v \in \{0,1\}}} i \pi_{i,0,k,l,0,v} + H_1 - \beta_1 c \right) \\
& + (\ln y_1) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,0,k,l,0) \in Q_{p,q} \\ v \in \{0,1\}}} l \pi_{i,0,k,l,0,v} - 2\beta_1 c \right) .
\end{aligned}$$

- Derivatives with respect to $\pi_{i,0,k,l,0,v}$:

$$\begin{aligned}
\frac{\partial \Lambda}{\partial \pi_{i,0,k,l,0,1}} &= \ln \omega_{i,0,k,l,0,1} + \ln \binom{i+k}{i,k} - \ln \pi_{i,0,k,l,0,v} + \ln r_{i+k,l} + 2i \ln x_1 + l \ln y_1 ; \\
\frac{\partial \Lambda}{\partial \pi_{i,0,k,l,0,0}} &= \ln \omega_{i,0,k,l,0,0} + \ln \binom{i+k}{i,k} - \ln \pi_{i,0,k,l,0,v} + \ln r_{l,i+k} + 2i \ln x_1 + l \ln y_1 .
\end{aligned}$$

Canceling out these derivatives yields:

$$\begin{aligned}
\pi_{i,0,k,l,0,1} &= \omega_{i,0,k,l,0,1} \binom{i+k}{i,k} r_{i+k,l} x_1^{2i} y_1^l ; \\
\pi_{i,0,k,l,0,0} &= \omega_{i,0,k,l,0,0} \binom{i+k}{i,k} r_{l,i+k} x_1^{2i} y_1^l .
\end{aligned}$$

The $r_{p,q}$ constraints become:

$$\begin{aligned} & \sum_{(i,0,k,q,0) \in Q_{p,q}} \omega_{i,0,k,q,0,1} \binom{i+k}{i,k} r_{i+k,q} x_1^{2i} y_1^q \\ + & \sum_{(i,0,k,p,0) \in Q_{q,p}} \omega_{i,0,k,p,0,0} \binom{i+k}{i,k} r_{l,i+k} x_1^{2i} y_1^p = d_{p,q} . \end{aligned}$$

Let us introduce

$$\begin{aligned} A_{p,q} &= \sum_{(i,0,k,q,0) \in Q_{p,q}} \omega_{i,0,k,q,0,1} \binom{p}{i,k} x_1^{2i} y_1^q \\ &+ \sum_{(i,0,k,l,0) \in Q_{q,p}} \omega_{i,0,k,p,0,0} \binom{q}{i,k} x_1^{2i} y_1^p : \\ r_{p,q} A_{p,q} &= d_{p,q} \end{aligned}$$

thus

$$r_{p,q} = \frac{d_{p,q}}{A_{p,q}}$$

and

$$\begin{aligned} \pi_{i,0,k,l,0,1} &= \omega_{i,0,k,l,0,1} \binom{i+k}{i,k} \frac{d_{i+k,l}}{A_{i+k,l}} x_1^{2i} y_1^l ; \\ \pi_{i,0,k,l,0,0} &= \omega_{i,0,k,l,0,0} \binom{i+k}{i,k} \frac{d_{l,i+k}}{A_{l,i+k}} x_1^{2i} y_1^l . \end{aligned}$$

– Derivatives with respect to β_t :

$$\begin{aligned} \frac{\partial \Lambda}{\partial \beta_1} &= 2c \ln 2 + 2c \ln \beta_1 + 2c \ln b - 2c \ln x_1 - 2c \ln y_1 ; \\ \frac{\partial \Lambda}{\partial \beta_3} &= 2c \ln 3 + 2c \ln \beta_3 + 2c \ln b . \end{aligned}$$

Canceling out these derivatives yields:

$$\begin{aligned} \beta_1 &= \frac{x_1 y_1}{2b} ; \\ \beta_3 &= \frac{1}{3b} . \end{aligned}$$

Constraint (A.1) yields:

$$\frac{x_1 y_1}{2b} + \frac{1}{3b} = 1$$

i.e.

$$b = \frac{x_1 y_1}{2} + \frac{1}{3} .$$

– Derivatives with respect to H_i :

$$\begin{aligned}\frac{\partial \Lambda}{\partial H_1} &= -\ln H_1 + \ln h + 2 \ln x_1 ; \\ \frac{\partial \Lambda}{\partial H_3} &= -\ln H_3 + \ln h .\end{aligned}$$

Canceling out these derivatives yields:

$$\begin{aligned}H_1 &= hx_1^2 ; \\ H_3 &= h .\end{aligned}$$

Constraint (A.3) yields then:

$$h(x_1^2 + 1) = H$$

i.e.

$$h = \frac{H}{x_1^2 + 1} .$$

– Remaining constraints:

$$\begin{aligned}& \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,0,k,q,0) \in Q_{p,q}} i \binom{p}{i,k} x_1^{2i} y_1^q \omega_{i,0,k,q,0,1} \\ & + \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,0,k,p,0) \in Q_{q,p}} i \binom{q}{i,k} x_1^{2i} y_1^q \omega_{i,0,k,p,0,0} + \frac{Hx_1^2}{x_1^2 + 1} = \frac{x_1 y_1 c}{2 \left(\frac{x_1 y_1}{2} + \frac{1}{3} \right)} ; \\ & \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,0,k,q,0) \in Q_{p,q}} q \binom{p}{i,k} x_1^{2i} y_1^q \omega_{i,0,k,q,0,1} \\ & + \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,0,k,p,0) \in Q_{q,p}} p \binom{q}{i,k} x_1^{2i} y_1^p \omega_{i,0,k,p,0,0} = \frac{x_1 y_1 c}{\left(\frac{x_1 y_1}{2} + \frac{1}{3} \right)} .\end{aligned}$$

Then we introduce $Z = \prod_{(p,q) \in \mathcal{L}} A_{p,q}^{d_{p,q}}$ and $Y = \ln Z$:

$$\begin{aligned}\frac{\partial Y}{\partial x_1} + \frac{2Hx_1}{x_1^2 + 1} &= \frac{y_1 c}{\left(\frac{x_1 y_1}{2} + \frac{1}{3} \right)} ; \\ \frac{\partial Y}{\partial y_1} &= \frac{x_1 c}{\left(\frac{x_1 y_1}{2} + \frac{1}{3} \right)} .\end{aligned}$$

With **Mathematica** we found the solutions (and the corresponding values of $\ln F$) mentioned in table (A.3).

So we find a feasible point where $\beta_2 = 0$ all unforced variables are nonzero.

Table A.3: Interior point when $\beta_2 = 0$.

model	standard	balanced signs	balanced occurrences	balanced signs and occurrences
c	4.500	3.509	4.623	3.546
x_1	0.997334	0.96803	0.999726	0.970462
y_1	2.07123	2.06284	2.05335	2.06087
$\ln F$	-2.463	-1.78313	-2.53084	-1.79349

- sub-case where all unforced variables are nonzero: we define a function $f(\xi)$ representing the variation of $\ln F$ under a small positive variation ξ in the following direction; remember that $\omega_{i,j,k,l,m,v} = 1$ as soon as $i \geq 1$ thus the corresponding variable $\pi_{i,j,k,l,m,v}$ exists. Let us take some $p \geq 3$ and $q \geq 2$ such that $d_{p,q} > 0$. We make the following move:

$$\begin{aligned}
\beta_1 &\rightarrow \beta_1 - \frac{\xi}{c} \\
\beta_2 &\rightarrow \beta_2 + \frac{2\xi}{c} \\
\beta_3 &\rightarrow \beta_3 - \frac{\xi}{c} \\
\pi_{p-1,0,1,q,0,1} &\rightarrow \pi_{p-1,0,1,q,0,1} - 5\xi \\
\pi_{p-2,1,1,q,0,1} &\rightarrow \pi_{p-2,1,1,q,0,1} + \xi \\
\pi_{p-1,0,1,q-2,2,1} &\rightarrow \pi_{p-1,0,1,q-2,2,1} + \xi \\
\pi_{p-1,1,0,q,0,1} &\rightarrow \pi_{p-1,1,0,q,0,1} + 3\xi
\end{aligned}$$

so that all constraints remain satisfied; in fact we are performing a small move inside the polytope \mathcal{P} and we would like to show that along this direction $\ln F$ is increasing. Note that $\beta_2 = \pi_{p-2,1,1,q,0,1} = \pi_{p-1,0,1,q-2,2,1} = \pi_{p-1,1,0,q,0,1} = 0$ and all other variables here are nonzero, so we have:

$$\begin{aligned}
f(\xi) &= 2c \left(\beta_1 - \frac{\xi}{c} \right) \ln \left(\frac{2}{e} \left(\beta_1 - \frac{\xi}{c} \right) \right) - 2c\beta_1 \ln \left(\frac{2}{e} \beta_1 \right) \\
&+ 2c \left(\beta_2 + \frac{2\xi}{c} \right) \ln \left(\frac{2}{e} \left(\beta_2 + \frac{2\xi}{c} \right) \right) \\
&+ 2c \left(\beta_3 - \frac{\xi}{c} \right) \ln \left(\frac{3}{e} \left(\beta_3 - \frac{\xi}{c} \right) \right) - 2c\beta_3 \ln \left(\frac{3}{e} \beta_3 \right) \\
&- (\pi_{p-1,0,1,q,0,1} - 5\xi) \ln (\pi_{p-1,0,1,q,0,1} - 5\xi) \\
&+ \pi_{p-1,0,1,q,0,1} \ln \pi_{p-1,0,1,q,0,1} \\
&- (\pi_{p-2,1,1,q,0,1} + \xi) \ln (\pi_{p-2,1,1,q,0,1} + \xi) \\
&- (\pi_{p-1,0,1,q-2,2,1} + \xi) \ln (\pi_{p-1,0,1,q-2,2,1} + \xi) \\
&- (\pi_{p-1,1,0,q,0,1} + 3\xi) \ln (\pi_{p-1,1,0,q,0,1} + 3\xi) + \Theta(\xi)
\end{aligned}$$

that is

$$\begin{aligned}
f(\tilde{\zeta}) &= +2c \left(\beta_1 - \frac{\tilde{\zeta}}{c} \right) \ln \left(\frac{2}{e} \left(\beta_1 - \frac{\tilde{\zeta}}{c} \right) \right) - 2c\beta_1 \ln \left(\frac{2}{e}\beta_1 \right) \\
&\quad + 4\tilde{\zeta} \ln \left(\frac{4\tilde{\zeta}}{ec} \right) \\
&\quad + 2c \left(\beta_3 - \frac{\tilde{\zeta}}{c} \right) \ln \left(\frac{3}{e} \left(\beta_3 - \frac{\tilde{\zeta}}{c} \right) \right) - 2c\beta_3 \ln \left(\frac{3}{e}\beta_3 \right) \\
&\quad - (\pi_{p-1,0,1,q,0,1} - 5\tilde{\zeta}) \ln (\pi_{p-1,0,1,q,0,1} - 5\tilde{\zeta}) + \pi_{p-1,0,1,q,0,1} \ln \pi_{p-1,0,1,q,0,1} \\
&\quad - 2\tilde{\zeta} \ln \tilde{\zeta} - 3\tilde{\zeta} \ln (3\tilde{\zeta}) + \Theta(\tilde{\zeta})
\end{aligned}$$

and thus:

$$\lim_{\tilde{\zeta} \rightarrow 0} \frac{f(\tilde{\zeta})}{\tilde{\zeta}} = \lim_{\tilde{\zeta} \rightarrow 0} (-\ln \tilde{\zeta}) + \Theta(1) .$$

Since $\lim_{\tilde{\zeta} \rightarrow 0} \frac{f(\tilde{\zeta})}{\tilde{\zeta}} = +\infty$, we have found an increasing direction.

2. case where $\beta_3 = 0$: then $H_3 = 0$ and $\pi_{i,j,k,l,m,v} = 0$ unless $k = 0$; again we call these variables *forced*.
- sub-case where there is an unforced variable at zero: we find a feasible point where $\beta_3 = 0$ and all unforced variables are nonzero. Then a move towards this point gives an increasing direction (because $\beta_1 > 0$ and $\beta_2 > 0$). To find such a point we use again the Lagrange multipliers method, as follows:
 - Definition of the Lagrangian:

$$\begin{aligned}
\Lambda = & \tau \ln 2 + H \ln H - H_1 \ln \left(\frac{H_1}{e} \right) - H_2 \ln \left(\frac{H_2}{e} \right) - H \\
& + \sum_{(p,q) \in \mathcal{L}} d_{p,q} \ln d_{p,q} + \sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,0,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} \pi_{i,j,0,l,m,v} \ln \left(\omega_{i,j,0,l,m,v} \frac{\binom{i+j}{i,j} \binom{l+m}{l,m}}{e \pi_{i,j,0,l,m,v}} \right) + 1 \\
& - 2c \ln 3 + 2c\beta_1 \ln \left(2 \frac{\beta_1}{e} \right) + 2c\beta_2 \ln \left(2 \frac{\beta_2}{e} \right) + 2c \\
& + (2c \ln b) (\beta_1 + \beta_2 - 1) \\
& + \sum_{(p,q) \in \mathcal{L}} (\ln r_{p,q}) \left(\sum_{(i,j,0,l,m) \in Q_{p,q}} \pi_{i,j,0,l,m,1} + \sum_{(i,j,0,l,m) \in Q_{q,p}} \pi_{i,j,0,l,m,0} - d_{p,q} \right) \\
& + (\ln h) (H_1 + H_2 - H) \\
& + (2 \ln x_1) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,0,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} i \pi_{i,j,0,l,m,v} + H_1 - \beta_1 c \right) \\
& + (\ln x_2) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,0,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} j \pi_{i,j,0,l,m,v} + H_2 - 2\beta_2 c \right) \\
& + (\ln y_1) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,0,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} l \pi_{i,j,0,l,m,v} - 2\beta_1 c \right) \\
& + (2 \ln y_2) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,0,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} m \pi_{i,j,0,l,m,v} - \beta_2 c \right) .
\end{aligned}$$

– Derivatives with respect to $\pi_{i,j,0,l,m,v}$:

$$\begin{aligned} \frac{\partial \Lambda}{\partial \pi_{i,j,0,l,m,1}} &= \ln \omega_{i,j,0,l,m,1} + \ln \left(\binom{i+j}{i,j} \binom{l+m}{l,m} \right) - \ln \pi_{i,j,0,l,m,v} + \ln r_{i+j,l+m} \\ &\quad + 2i \ln x_1 + j \ln x_2 + l \ln y_1 + 2m \ln y_2 ; \\ \frac{\partial \Lambda}{\partial \pi_{i,j,0,l,m,0}} &= \ln \omega_{i,j,0,l,m,0} + \ln \left(\binom{i+j}{i,j} \binom{l+m}{l,m} \right) - \ln \pi_{i,j,0,l,m,v} + \ln r_{l+m,i+j} \\ &\quad + 2i \ln x_1 + j \ln x_2 + l \ln y_1 + 2m \ln y_2 . \end{aligned}$$

Canceling out these derivatives yields:

$$\begin{aligned} \pi_{i,j,0,l,m,1} &= \omega_{i,j,0,l,m,1} \binom{i+j}{i,j} \binom{l+m}{l,m} r_{i+j,l+m} x_1^{2i} x_2^j y_1^l y_2^{2m} ; \\ \pi_{i,j,0,l,m,0} &= \omega_{i,j,0,l,m,0} \binom{i+j}{i,j} \binom{l+m}{l,m} r_{l+m,i+j} x_1^{2i} x_2^j y_1^l y_2^{2m} . \end{aligned}$$

The $r_{p,q}$ constraints become:

$$\begin{aligned} &\sum_{(i,j,0,l,m) \in Q_{p,q}} \omega_{i,j,0,l,m,1} \binom{i+j}{i,j} \binom{l+m}{l,m} r_{i+j,l+m} x_1^{2i} x_2^j y_1^l y_2^{2m} \\ + &\sum_{(i,j,0,l,m) \in Q_{q,p}} \omega_{i,j,0,l,m,0} \binom{i+j}{i,j} \binom{l+m}{l,m} r_{l+m,i+j} x_1^{2i} x_2^j y_1^l y_2^{2m} = d_{p,q} . \end{aligned}$$

Let us introduce

$$\begin{aligned} A_{p,q} &= \sum_{(i,j,0,l,m) \in Q_{p,q}} \omega_{i,j,0,l,m,1} \binom{p}{i,j} \binom{q}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \\ &\quad + \sum_{(i,j,0,l,m) \in Q_{q,p}} \omega_{i,j,0,l,m,0} \binom{q}{i,j} \binom{p}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} . \end{aligned}$$

We have:

$$r_{p,q} A_{p,q} = d_{p,q}$$

i.e.

$$r_{p,q} = \frac{d_{p,q}}{A_{p,q}} .$$

Thus

$$\begin{aligned} \pi_{i,j,0,l,m,1} &= \omega_{i,j,0,l,m,1} \binom{i+j}{i,j} \binom{l+m}{l,m} \frac{d_{i+j,l+m}}{A_{i+j,l+m}} x_1^{2i} x_2^j y_1^l y_2^{2m} ; \\ \pi_{i,j,0,l,m,0} &= \omega_{i,j,0,l,m,0} \binom{i+j}{i,j} \binom{l+m}{l,m} \frac{d_{l+m,i+j}}{A_{l+m,i+j}} x_1^{2i} x_2^j y_1^l y_2^{2m} . \end{aligned}$$

– Derivatives with respect to β_t :

$$\begin{aligned}\frac{\partial \Lambda}{\partial \beta_1} &= 2c \ln 2 + 2c \ln \beta_1 + 2c \ln b - 2c \ln x_1 - 2c \ln y_1 ; \\ \frac{\partial \Lambda}{\partial \beta_2} &= 2c \ln 2 + 2c \ln \beta_2 + 2c \ln b - 2c \ln x_2 - 2c \ln y_2 .\end{aligned}$$

Canceling out these derivatives yields:

$$\begin{aligned}\beta_1 &= \frac{x_1 y_1}{2b} ; \\ \beta_2 &= \frac{x_2 y_2}{2b} .\end{aligned}$$

Constraint (A.1) yields:

$$\frac{x_1 y_1}{2b} + \frac{x_2 y_2}{2b} = 1$$

i.e.

$$b = \frac{x_1 y_1}{2} + \frac{x_2 y_2}{2} .$$

– Derivatives with respect to H_t :

$$\begin{aligned}\frac{\partial \Lambda}{\partial H_1} &= -\ln H_1 + \ln h + 2 \ln x_1 ; \\ \frac{\partial \Lambda}{\partial H_2} &= -\ln H_2 + \ln h + \ln x_2 .\end{aligned}$$

Canceling out these derivatives yields:

$$\begin{aligned}H_1 &= h x_1^2 ; \\ H_2 &= h x_2 .\end{aligned}$$

Constraint (A.3) yields then:

$$h (x_1^2 + x_2) = H$$

i.e.

$$h = \frac{H}{x_1^2 + x_2} .$$

– Remaining constraints:

$$\begin{aligned}
& \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,0,l,m) \in Q_{p,q}} i \binom{p}{i,j} \binom{q}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,0,l,m,1} \\
& + \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,0,l,m) \in Q_{q,p}} i \binom{q}{i,j} \binom{p}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,0,l,m,0} + \frac{Hx_1^2}{x_1^2 + x_2} \\
& \qquad \qquad \qquad = \frac{x_1 y_1 c}{(x_1 y_1 + x_2 y_2)} ; \\
& \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,0,l,m) \in Q_{p,q}} j \binom{p}{i,j} \binom{q}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,0,l,m,1} \\
& + \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,0,l,m) \in Q_{q,p}} j \binom{q}{i,j} \binom{p}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,0,l,m,0} + \frac{Hx_2}{x_1^2 + x_2} \\
& \qquad \qquad \qquad = \frac{2x_2 y_2 c}{(x_1 y_1 + x_2 y_2)} ; \\
& \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,0,l,m) \in Q_{p,q}} l \binom{p}{i,j} \binom{q}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,0,l,m,1} \\
& + \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,0,l,m) \in Q_{q,p}} l \binom{q}{i,j} \binom{p}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,0,l,m,0} \\
& \qquad \qquad \qquad = \frac{2x_1 y_1 c}{(x_1 y_1 + x_2 y_2)} ; \\
& \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,0,l,m) \in Q_{p,q}} m \binom{p}{i,j} \binom{q}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,0,l,m,1} \\
& + \sum_{(p,q) \in \mathcal{L}} \frac{d_{p,q}}{A_{p,q}} \sum_{(i,j,0,l,m) \in Q_{q,p}} m \binom{q}{i,j} \binom{p}{l,m} x_1^{2i} x_2^j y_1^l y_2^{2m} \omega_{i,j,0,l,m,0} \\
& \qquad \qquad \qquad = \frac{x_2 y_2 c}{(x_1 y_1 + x_2 y_2)} .
\end{aligned}$$

Then we introduce $Z = \prod_{(p,q) \in \mathcal{L}} A_{p,q}^{d_{p,q}}$ and $Y = \ln Z$:

$$\begin{aligned}
\frac{\partial Y}{\partial x_1} + \frac{2Hx_1}{x_1^2 + x_2} &= \frac{y_1 c}{\left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2}\right)} ; \\
\frac{\partial Y}{\partial x_2} + \frac{H}{x_1^2 + x_2} &= \frac{y_2 c}{\left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2}\right)} ; \\
\frac{\partial Y}{\partial y_1} &= \frac{x_1 c}{\left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2}\right)} ; \\
\frac{\partial Y}{\partial y_2} &= \frac{x_2 c}{\left(\frac{x_1 y_1}{2} + \frac{x_2 y_2}{2}\right)} .
\end{aligned}$$

With **Mathematica** we found the solutions (and the corresponding values of $\ln F$) mentioned in table (A.4).

So we find a feasible point where $\beta_3 = 0$ and all unforced variables are nonzero.

Table A.4: Interior point when $\beta_3 = 0$.

model	standard	balanced signs	balanced occurrences	balanced signs and occurrences
c	4.500	3.509	4.623	3.546
x_1	0.512382	0.473066	0.493115	0.494614
x_2	0.546014	0.489687	0.501307	0.494614
y_1	0.583465	0.520769	0.531045	0.494614
y_2	0.529328	0.513117	0.505216	0.494614
$\ln F$	-0.682149	-0.375917	-0.695819	-0.33427

- sub-case where all unforced variables are nonzero: we define a function $f(\xi)$ representing the variation of $\ln F$ under a small positive variation ξ in the following direction; remember that $\omega_{i,j,k,l,m,v} = 1$ as soon as $i \geq 1$ thus the corresponding variable $\pi_{i,j,k,l,m,v}$ exists. Let us take some $p \geq 3$ and $q \geq 2$ such that $d_{p,q} > 0$. We make the following move:

$$\begin{aligned}
\beta_1 &\rightarrow \beta_1 + \frac{\xi}{c} \\
\beta_2 &\rightarrow \beta_2 - \frac{2\xi}{c} \\
\beta_3 &\rightarrow \beta_3 + \frac{\xi}{c} \\
\pi_{p-2,2,0,q-1,1,1} &\rightarrow \pi_{p-2,2,0,q-1,1,1} - 3\xi \\
\pi_{p-1,0,1,q-1,1,1} &\rightarrow \pi_{p-1,0,1,q-1,1,1} + \xi \\
\pi_{p-2,1,1,q,0,1} &\rightarrow \pi_{p-2,1,1,q,0,1} + 2\xi
\end{aligned}$$

so that all constraints remain satisfied; in fact we are performing a small move inside the polytope \mathcal{P} and we would like to show that along this direction $\ln F$ is increasing. Note that $\beta_3 = \pi_{p-1,0,1,q-1,1,1} = \pi_{p-2,1,1,q,0,1} = 0$ and all other variables here are nonzero, so we have:

$$\begin{aligned}
f(\xi) &= 2c \left(\beta_1 + \frac{\xi}{c} \right) \ln \left(\frac{2}{e} \left(\beta_1 + \frac{\xi}{c} \right) \right) - 2c\beta_1 \ln \left(\frac{2}{e}\beta_1 \right) \\
&+ 2c \left(\beta_2 - \frac{2\xi}{c} \right) \ln \left(\frac{2}{e} \left(\beta_2 - \frac{2\xi}{c} \right) \right) - 2c\beta_2 \ln \left(\frac{2}{e}\beta_2 \right) \\
&+ 2c \left(\beta_3 + \frac{\xi}{c} \right) \ln \left(\frac{3}{e} \left(\beta_3 + \frac{\xi}{c} \right) \right) \\
&- (\pi_{p-2,2,0,q-1,1,1} - 3\xi) \ln (\pi_{p-2,2,0,q-1,1,1} - 3\xi) \\
&+ \pi_{p-2,2,0,q-1,1,1} \ln \pi_{p-2,2,0,q-1,1,1} \\
&- (\pi_{p-1,0,1,q-1,1,1} + \xi) \ln (\pi_{p-1,0,1,q-1,1,1} + \xi) \\
&- (\pi_{p-2,1,1,q,0,1} + 2\xi) \ln (\pi_{p-2,1,1,q,0,1} + 2\xi) + \Theta(\xi)
\end{aligned}$$

thus

$$\begin{aligned}
f(\xi) &= +2c \left(\beta_1 + \frac{\xi}{c} \right) \ln \left(\frac{2}{e} \left(\beta_1 + \frac{\xi}{c} \right) \right) - 2c\beta_1 \ln \left(\frac{2}{e} \beta_1 \right) \\
&+ 2c \left(\beta_2 - \frac{2\xi}{c} \right) \ln \left(\frac{2}{e} \left(\beta_2 - \frac{2\xi}{c} \right) \right) - 2c\beta_2 \ln \left(\frac{2}{e} \beta_2 \right) \\
&+ 2\xi \ln \left(\frac{3\xi}{ec} \right) \\
&- (\pi_{p-2,2,0,q-1,1,1} - 3\xi) \ln (\pi_{p-2,2,0,q-1,1,1} - 3\xi) \\
&+ \pi_{p-2,2,0,q-1,1,1} \ln \pi_{p-2,2,0,q-1,1,1} \\
&- \xi \ln \xi - 2\xi \ln (2\xi) + \Theta(\xi)
\end{aligned}$$

and thus:

$$\lim_{\xi \rightarrow 0} \frac{f(\xi)}{\xi} = \lim_{\xi \rightarrow 0} (-\ln \xi) + \Theta(1) .$$

Since $\lim_{\xi \rightarrow 0} \frac{f(\xi)}{\xi} = +\infty$, we have found an increasing direction.

3. case where all $\beta_t > 0$; suppose there is another variable at zero; we move towards the general solution we found in section A.1, where all variables are nonzero. Then again $\lim_{\xi \rightarrow 0} \frac{f(\xi)}{\xi} = +\infty$; so this is an increasing direction.

A.3 Inspection of the Interior of \mathcal{P}

As Díaz, Kirousis, Mitsche & Pérez-Giménez [DKMPG09] noticed in their calculation, we can perform a sweep over some coordinates in order to check that the solution of the Lagrange multipliers problem is indeed a global maximum. Namely when we fix all β_1, β_2 (and $\beta_3 = 1 - \beta_1 - \beta_2$), the function $\ln F$ is strictly concave in the other variables. Let $\mathcal{P}_{\beta_1, \beta_2}$ the polytope where the remaining variables are allowed to move; remember that the function to maximize is:

$$\begin{aligned}
\ln F &= \tau \ln 2 + H \ln H - H_1 \ln \left(\frac{H_1}{e} \right) - H_2 \ln \left(\frac{H_2}{e} \right) - H_3 \ln \left(\frac{H_3}{e} \right) - H \\
&+ \sum_{(p,q) \in \mathcal{L}} d_{p,q} \ln d_{p,q} + \sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} \pi_{i,j,k,l,m,v} \ln \left(\omega_{i,j,k,l,m,v} \frac{\binom{i+j+k}{i,j,k} \binom{l+m}{l,m}}{e^{\pi_{i,j,k,l,m,v}}} \right) + 1 \\
&- 2c \ln 3 + 2c\beta_1 \ln \left(2 \frac{\beta_1}{e} \right) + 2c\beta_2 \ln \left(2 \frac{\beta_2}{e} \right) + 2c\beta_3 \ln \left(3 \frac{\beta_3}{e} \right) + 2c .
\end{aligned}$$

If we increase an H_t or a $\pi_{i,j,k,l,m,v}$ from 0 to a small $\xi > 0$ and we change any other nonzero variables, then the variation f of $\ln F$: $f = -\xi \ln \xi + \Theta(\xi)$ is such that $\frac{f}{\xi} = -\ln \xi + \Theta(1) \rightarrow +\infty$, so $\ln F$ must increase; thus the function cannot maximize on the boundary of $\mathcal{P}_{\beta_1, \beta_2}$ and we can apply the Lagrange multiplier technique again. But now by strict concavity of the objective function, we know that the solution we find corresponds to a global maximum.

– Definition of the Lagrangian:

$$\begin{aligned}
\Lambda = & \tau \ln 2 + H \ln H - H_1 \ln \left(\frac{H_1}{e} \right) - H_2 \ln \left(\frac{H_2}{e} \right) - H_3 \ln \left(\frac{H_3}{e} \right) - H \\
& + \sum_{(p,q) \in \mathcal{L}} d_{p,q} \ln d_{p,q} + \sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in \mathcal{Q}_{p,q} \\ v \in \{0,1\}}} \pi_{i,j,k,l,m,v} \ln \left(\omega_{i,j,k,l,m,v} \frac{\binom{i+j+k}{i,j,k} \binom{l+m}{l,m}}{e^{\pi_{i,j,k,l,m,v}}} \right) + 1 \\
& - 2c \ln 3 + 2c\beta_1 \ln \left(2 \frac{\beta_1}{e} \right) + 2c\beta_2 \ln \left(2 \frac{\beta_2}{e} \right) + 2c\beta_3 \ln \left(3 \frac{\beta_3}{e} \right) + 2c \\
& + \sum_{(p,q) \in \mathcal{L}} (\ln r_{p,q}) \left(\sum_{(i,j,k,l,m) \in \mathcal{Q}_{p,q}} \pi_{i,j,k,l,m,1} + \sum_{(i,j,k,l,m) \in \mathcal{Q}_{p,q}} \pi_{i,j,k,l,m,0} - d_{p,q} \right) \\
& + (\ln h) \left(H_1 + H_2 + H_3 - \sum_{(p,q) \in \mathcal{H}} (p+q) d_{p,q} \right) \\
& + (2 \ln x_1) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in \mathcal{Q}_{p,q} \\ v \in \{0,1\}}} i \pi_{i,j,k,l,m,v} + H_1 - \beta_1 c \right) \\
& + (\ln x_2) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in \mathcal{Q}_{p,q} \\ v \in \{0,1\}}} j \pi_{i,j,k,l,m,v} + H_2 - 2\beta_2 c \right) \\
& + (\ln y_1) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in \mathcal{Q}_{p,q} \\ v \in \{0,1\}}} l \pi_{i,j,k,l,m,v} - 2\beta_1 c \right) \\
& + (2 \ln y_2) \left(\sum_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in \mathcal{Q}_{p,q} \\ v \in \{0,1\}}} m \pi_{i,j,k,l,m,v} - \beta_2 c \right) .
\end{aligned}$$

– Derivatives with respect to $\pi_{i,j,k,l,m,v}$:

As in the general case we find that

$$\begin{aligned}
\pi_{i,j,k,l,m,1} &= \omega_{i,j,k,l,m,1} \binom{i+j+k}{i,j,k} \binom{l+m}{l,m} \frac{d_{i+j+k,l+m}}{A_{i+j+k,l+m}} x_1^{2i} x_2^j y_1^l y_2^{2m} ; \\
\pi_{i,j,k,l,m,0} &= \omega_{i,j,k,l,m,0} \binom{i+j+k}{i,j,k} \binom{l+m}{l,m} \frac{d_{l+m,i+j+k}}{A_{l+m,i+j+k}} x_1^{2i} x_2^j y_1^l y_2^{2m} .
\end{aligned}$$

- Derivatives with respect to H_t :
As in the general case we find that

$$\begin{aligned} H_1 &= \frac{Hx_1^2}{x_1^2 + x_2 + 1} ; \\ H_2 &= \frac{Hx_2}{x_1^2 + x_2 + 1} ; \\ H_3 &= \frac{H}{x_1^2 + x_2 + 1} . \end{aligned}$$

- Remaining constraints:

$$\begin{aligned} \frac{\partial Y}{\partial x_1} + \frac{2Hx_1}{x_1^2 + x_2 + 1} &= \frac{2\beta_1 c}{x_1} ; \\ \frac{\partial Y}{\partial x_2} + \frac{H}{x_1^2 + x_2 + 1} &= \frac{2\beta_2 c}{x_2} ; \\ \frac{\partial Y}{\partial y_1} &= \frac{2\beta_1 c}{y_1} ; \\ \frac{\partial Y}{\partial y_2} &= \frac{2\beta_2 c}{y_2} . \end{aligned}$$

- Objective function:

$$\begin{aligned} F &= 2^\tau \frac{H^H}{H_1^{H_1} H_2^{H_2} H_3^{H_3}} \left(\frac{1}{3} (2\beta_1)^{\beta_1} (2\beta_2)^{\beta_2} (3\beta_3)^{\beta_3} \right)^{2c} \\ &\cdot \prod_{(p,q) \in \mathcal{L}} d_{p,q}^{d_{p,q}} \prod_{\substack{(p,q) \in \mathcal{L} \\ (i,j,k,l,m) \in Q_{p,q} \\ v \in \{0,1\}}} \left(\omega_{i,j,k,l,m,v} \frac{\binom{i+j+k}{i,j,k} \binom{l+m}{l,m}}{\pi_{i,j,k,l,m,v}} \right)^{\pi_{i,j,k,l,m,v}} \end{aligned}$$

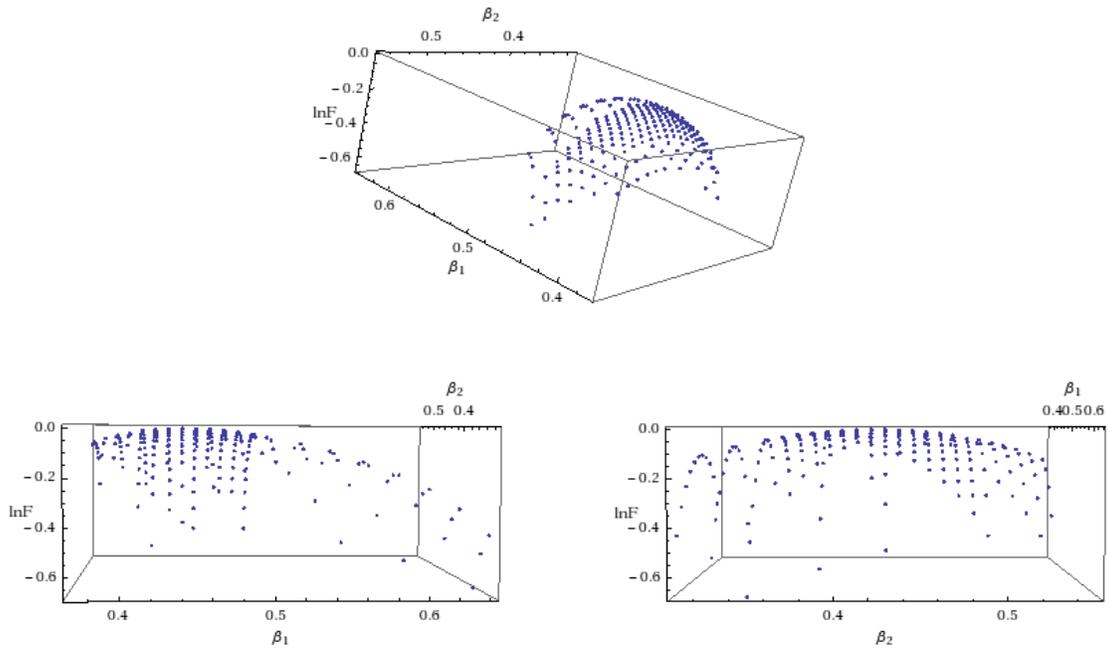
thus

$$F = 2^\tau (x_1^2 + x_2 + 1)^H \prod_{(p,q) \in \mathcal{L}} (A_{p,q}^{d_{p,q}}) \left(\frac{1}{3} \left(\frac{2\beta_1}{x_1 y_1} \right)^{\beta_1} \left(\frac{2\beta_2}{x_2 y_2} \right)^{\beta_2} (3\beta_3)^{\beta_3} \right)^{2c} .$$

So we made a sweep over β_1 and β_2 in the feasible region and plotted the maximum point given as the solution of these equations, which confirmed the fact that the solutions to the global Lagrange system are indeed global maxima.

Table A.5: Values of parameters obtained by a sweep over the interior of \mathcal{P} .

model	standard	balanced signs	balanced occurrences	balanced signs and occurrences
our method $(\alpha\rho_{j,l} + \rho_{k,m}, v) > (0, 0)$	4.500	3.509	4.623	3.546
β_1	0.44373	0.557479	0.445306	0.568436
β_2	0.421847	0.365723	0.421418	0.363128
β_3	0.134422	0.0767974	0.133276	0.0684362

Figure A.1: Maximum of $\ln F$ for different values of β_1 and β_2 in the standard model at $c = 4.500$. Numerically we found that the maximum is at $\beta_1 \simeq 0.44313$ and $\beta_2 \simeq 0.421847$.

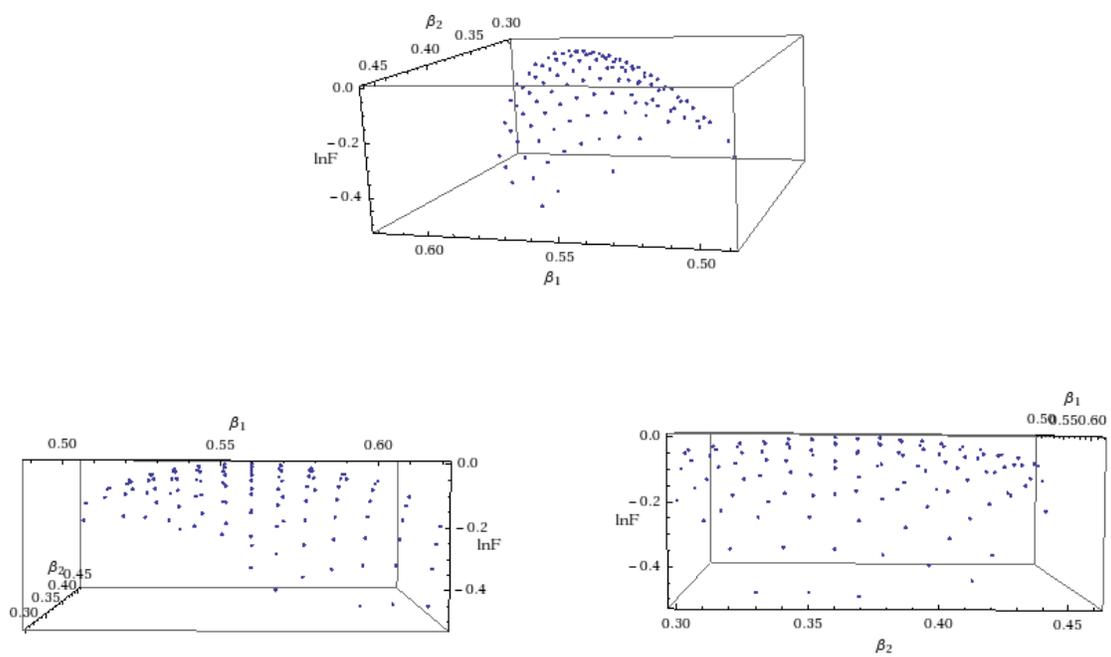


Figure A.2: Maximum of $\ln F$ for different values of β_1 and β_2 in the model with balanced signs at $c = 3.509$. Numerically we found that the maximum is at $\beta_1 \simeq 0.557479$ and $\beta_2 \simeq 0.365723$.

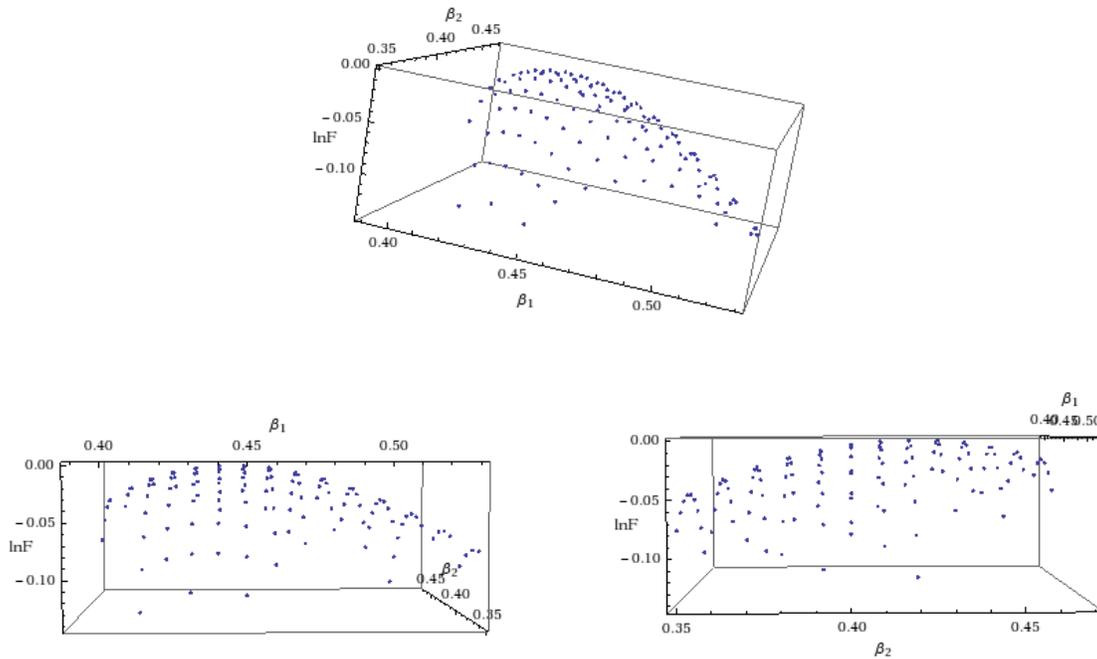


Figure A.3: Maximum of $\ln F$ for different values of β_1 and β_2 in the model with balanced occurrences at $c = 4.623$. Numerically we found that the maximum is at $\beta_1 \simeq 0.445306$ and $\beta_2 \simeq 0.421418$.

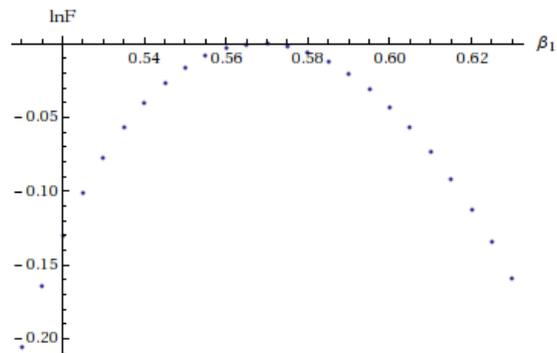


Figure A.4: Maximum of $\ln F$ for different values of β_1 in the model with balanced signs and occurrences at $c = 3.546$. In this particular model where each variable has as many positive occurrences as negative ones, true and false surfaces are equal: $\beta_1 + 2\beta_2 + 3\beta_3 = 2\beta_1 + \beta_2$, thus $\beta_2 = 1.5 - 2\beta_1$. Numerically we found that the maximum is at $\beta_1 \simeq 0.568436$ and $\beta_2 \simeq 0.363128$.

Some Useful Asymptotic Estimates

B.1 Multinomial Coefficients

Stirling's asymptotic estimate of the factorial states that

$$k! \sim \left(\frac{k}{e}\right)^k \sqrt{2\pi k} ; \quad (\text{B.1})$$

it is enough for us to know that $\frac{k!}{\left(\frac{k}{e}\right)^k \sqrt{2\pi k}}$ is bounded and away from 0. But to be more precise, we would like to mention the following inequalities due to Batir - 2008 [Bat08]:

$$\left(\frac{k}{e}\right)^k \sqrt{2\pi \left(k + \frac{1}{6}\right)} < k! < \left(\frac{k}{e}\right)^k \sqrt{2\pi \left(k + \left(\frac{e^2}{2\pi} - 1\right)\right)} .$$

It follows that when k is a constant, there exist two polynomials $\text{poly}_1(n)$ and $\text{poly}_2(n)$ such that:

$$\text{poly}_1(n) \frac{n^n}{n_1^{n_1} \dots n_k^{n_k}} < \binom{n}{n_1, \dots, n_k} < \text{poly}_2(n) \frac{n^n}{n_1^{n_1} \dots n_k^{n_k}} . \quad (\text{B.2})$$

B.2 Stirling Numbers of the Second Kind

We recall that $S(n, m)$ denotes the Stirling number of the second kind with parameters n and m . It represents the number of possible surjections from a set of n elements onto a set of m elements. We are interested in an asymptotic estimate of this number.

According to Temme - 1993 [Tem93],

$$S(n, m) \sim e^A m^{n-m} f(t) \binom{n}{m} ,$$

where $t = \frac{n-m}{m}$, f is bounded by a polynomial, $A = \Phi(x) - mt + (n-m) \ln t$, $\Phi(x) = -n \ln x + m \ln(e^x - 1)$, and x is the positive solution to the following equation:

$$\frac{m}{n} x = 1 - e^{-x} . \quad (\text{B.3})$$

So we have

$$\begin{aligned} S(n, m) &\sim e^{-n \ln x + m \ln(e^x - 1) - n + m + (n-m) \ln \frac{n-m}{m}} m^{n-m} f\left(\frac{n-m}{m}\right) \binom{n}{m} \\ &\sim x^{-n} (e^x - 1)^m \left(\frac{n-m}{e}\right)^{n-m} f\left(\frac{n-m}{m}\right) \binom{n}{m}. \end{aligned}$$

Using equation B.1 and inequalities B.2, we conclude that there exist two polynomials $\text{poly}_1(n)$ and $\text{poly}_2(n)$ such that $\text{poly}_1(n) B < S(n, m) m! < \text{poly}_2(n) B$ where

$$\begin{aligned} B &= \left(\frac{m}{e}\right)^m x^{-n} (e^x - 1)^m \left(\frac{n-m}{e}\right)^{n-m} \frac{n^n}{m^m (n-m)^{n-m}} \\ &= \left(\frac{n}{ex}\right)^n (e^x - 1)^m, \end{aligned}$$

i.e.

$$\text{poly}_1(n) \left(\frac{n}{ex}\right)^n (e^x - 1)^m < S(n, m) m! < \text{poly}_2(n) \left(\frac{n}{ex}\right)^n (e^x - 1)^m. \quad (\text{B.4})$$

First Derivative Cancels out

Suppose now that m and n are variables of some function F to maximize, and that F contains a Stirling number $S(n, m) m!$; suppose that we work only on the exponential equivalent of F : $\ln F = \Lambda + \Delta$, where $\Lambda = m \ln(e^x - 1) + n \ln\left(\frac{n}{ex}\right)$ and x does not appear in Δ . In particular x is an implicit function of m and n as defined by equation B.3.

$$\begin{aligned} \frac{\partial \Lambda}{\partial m} &= \ln(e^x - 1) + \frac{\partial \Lambda}{\partial x} \frac{\partial x}{\partial m}; \\ \frac{\partial \Lambda}{\partial n} &= \ln \frac{n}{x} + \frac{\partial \Lambda}{\partial x} \frac{\partial x}{\partial n}; \\ \frac{\partial \Lambda}{\partial x} &= m \frac{e^x}{e^x - 1} - \frac{n}{x}. \end{aligned}$$

But if x satisfies equation B.3, $m \frac{e^x}{e^x - 1} - \frac{n}{x} = \frac{m}{1 - e^{-x}} - \frac{n}{x} = \frac{n}{x} - \frac{n}{x} = 0$, thus $\frac{\partial \Lambda}{\partial x} = 0$. Consequently maximization of $\ln F$ can be performed as if x were a constant.

Index

- actual weight, 57, 59
- adjacency, 54

- balanced model, 83
- boolean function, 23
- boolean solution, 71
- boundary, 151

- clause, 19
- clause type, 103, 122
- closed under renaming, 68
- CNF, *see* Conjunctive Normal Form
- coarse threshold, 22
- coarsening, 41
- Conjunctive Normal Form, 19
- constraining clause, 44
- constraint, 54
- Constraint Satisfaction Problem, 54
- convexity condition, 42
- core, 31
- covering, 58
- critical, 22, 27, 82
- CSP, *see* Constraint Satisfaction Problem

- decomposer, 58
- dispatcher, 59
- dispensable variable, 39
- distributional model, 20
- domain, 54
- drawing model, 20

- First Moment Method, 37
- forced variable, 152
- formula, 19
- free variable, 39, 81, 85

- frozen variables, 24

- generator, 59

- Hamming graph, 81
- Hamming similarity, 117
- heavy variable, 88
- homogeneous, 59

- implicant, 39, 113
- incomplete solver, 30
- increasing direction, 152
- independence point, 107
- induced orientation, 55
- influence, 22
- instance, 19
- instance dependent ordering, 56
- instance independent ordering, 56
- invertible variable, 41

- light variable, 87
- literal, 19
- locality condition, 53

- monotone formula, 23

- NAE, 20
- Negatively Prime Solution, 39, 76
- neighborhood, 55
- non-uniform, 82
- Not All Equal, *see* NAE
- NPS, *see* Negatively Prime Solution

- ordering, 53
- orientation, 55

partial assignment, 41
permutation, 68
phase transition, 21
planted model, 30
prime implicant, 39

regular model, 83
renaming, 68

SAT, 19
Second Moment Method, 97
seed, 57
sharp threshold, 22
sign, 102
solution, 54
solutions network, 55
standard model, 83
starrable variable, 41
Stirling number, 169
surface, 100, 103, 120
symmetry of occurrences, 103

threshold, 21
Transfer lemma, 58
truth value, 102
type of a clause, 84
type of a variable, 84, 87

uniform, 56, 59, 82
unitary, 57
unladen weight, 57

valid partial assignment, 41
valid valuation, 41, 71
valuation, 41, 54
value, 101

Weight Conservation Theorem, 58
Weight Preservation Theorem, 70
weighting, 53

XOR, 20