Department of Computer Science
University of Strathclyde

The Number Partition Phase Transition

by

Ian P. Gent and Toby Walsh

RR-95-185 May 1995

The Number Partition Phase Transition

Ian P. Gent* and Toby Walsh†

May 1995

Abstract

We identify a natural parameter for random number partitioning, and show that there is a rapid transition between soluble and insoluble problems at a critical value of this parameter. Hard problems are associated with this transition. As is seen in other computational problems, finite-size scaling methods developed in statistical mechanics describe the behaviour around the critical value. Such phase transition phenomena appear to be ubiquitous. Indeed, we have yet to find a NP-complete problem which lacks a similar phase transition. The identification of a phase transition in number partitioning is of particular interest since it had been suggested that one would not occur.

*Department of Computer Science, University of Strathclyde, Glasgow G1 1XH, Scotland. ipg@cs.strath.ac.uk

†Mechanized Reasoning Group, IRST, Loc. Panté di Povo, 38100 Trento & DIST, University of Genoa, 16143 Genoa, Italy. toby@irst.it

1 Introduction

Phase transitions have been observed in the solubility of many different NP-complete problems. One reason for interest in such phase transitions is that the hardest problems tend to occur at the phase boundary [CKT91]. In the soluble phase, problem usually have a very large number of solutions, and it is typically easy to find one. In the insoluble phase, it is usually comparatively easy to show that no solution exists. By comparison, problems from the phase transition are typically hard since they cannot easily be proved soluble or insoluble. Such phase transition behaviour has been observed in propositional satisfiability [KS94, CKT91, MSL92], graph colouring [CKT91], the traveling salesman problem [GW95], constraint satisfaction [Smi94, Pro94], independent set problems [GW94b], and Hamiltonian circuits [CKT91].

In this paper, we continue our exploration of the ubiquity of phase transition behaviour in NP-complete problems by considering number partitioning. Fu [Fu89] has claimed that a phase transition does not occur in number partitioning. The support for this claim comes from analysing the statistical mechanics of the problem. If we accept his argument, Fu then claims that "... we must accept as a conclusion that no connection whatsoever exists between the spin glass transition and NP-completeness ... the connection between phase transition and computational complexity, if any, must be of a form very different from what has been imagined ...". We show here that phase transition behaviour does occur in number partitioning problems. As a consequence, we are not required to separate phase transition behaviour from computational complexity.

2 Number Partitioning

Fu [Fu89] considers the optimization version of the number partioning problem. That is, given a bag, S of N positive integers (N even), find a partition of S into two disjoint bags, S_1 and S_2 both of size $N/2$ such that we minimize,

$$|\sum_{i \in S_1} a_i - \sum_{i \in S_2} a_i|$$

The integers a_i are taken from an independent Poisson distribution. Fu claims that other distributions give essentially the same results.

As is usual, when looking for phase transition behaviour [GW95], we begin by focusing on the decision version of the problem. However, see section 7 for some discussion and analysis of the optmization problem. Given a fixed k, the decision problem is to determine if there is a partition of S into S_1 and S_2 such that,

$$|\sum_{i \in S_1} a_i - \sum_{i \in S_2} a_i| \leq k$$

To simplify the problem, we will look just at perfect partitions. That is, we fix $k = 0$ when the sum of S is even, and $k = 1$ when the sum of S is odd. Finally, we drop the restrictions on the size of the bags (that N is even and that S_1 and S_2 are both the same size). We expect similar results both for other k and if we reinstated the restrictions on the size of the bags.

The resulting problem for $k = 0$ is what Garey and Johnson [GJ79] call the PARTITION problem. It is one of their six basic NP-complete problems that lie at the core of NP-completeness theory. These six problems are some of the most frequently used problems in proving NP-completeness. Phase transition behaviour has been observed in three out of the six problems (3-SAT [MSL92], CLIQUE via the dual independent set problem [GW94b], and HAMILTONIAN CIRCUIT [CKT91]). Here, we show it occuring in the fourth problem, PARTITION. We predict that the two remaining problems, 3-DIMENSIONAL MATCHING (3-DM) and VERTEX COVER will also display similar phase transition behaviour. In support of this conjecture, we note that 3-DM naturally "lies between" 3-SAT and PARTITION since Garey and Johnson demonstrate the NP-completeness of PARTITION by transforming 3-SAT into 3-DM and then 3-DM into PARTITION. Similarly VERTEX COVER naturally lies between 3-SAT and both HAMILTONIAN CIRCUIT and CLIQUE.

From a theoretical point of view, number partitioning has various interesting properties. First, it is the only one of Garey and Johnson's six basic problem about numbers. It is often therefore the natural choice for NP-completeness proofs of other number problems (*e.g.* bin packing, multiprocessor scheduling, open-shop scheduling, quadratic programming, and knapsack problems). Second, whilst it is a NP-complete problem, it is not NP-complete in the "strong" sense [GJ79]. That is, there exists a pseudo-polynomial time algorithm for solving the PARTITION problem. Given N numbers to partition which sum to M, there exists a simple algorithm based on dynamic programming which solves the PARTITION problem in time and space bounded by a low polynomial in $N.M$. Note, however, that the size of the input need only be $O(N.\log(M))$. The NP-completeness of the PARTITION problem depends largely on the fact that we can supply extremely large integers to partition. For this reason, the PARTITION problem may offer a strong test of the connection between phase transitions and NP-completeness.

3 Uniform distribution

We first consider number partition problems where random numbers are drawn uniformly from $[1, l]$. We restrict attention to those bags of random numbers which have an even sum. If a bag has an odd sum then it cannot be partitioned into two bags with equal sum. This can be checked in polynomial time. Except trivially at $l = 1$, half the problems generated at any l or N have an odd sum. We have observed almost identical results with bags with an odd sum, and with bags with both odd and even sums if we let $k = 1$. Similarly, we restrict attention to even N in this paper. We do observe broadly similar behaviour as described below for odd N, but there are some small N effects for odd parity groups: for example if $N = 3$ then the two numbers will almost certainly outweigh the third.

In Figure 1A, we plot the probability that a problem is soluble (*i.e.* the bag has an perfect partition) against l for N from 6 to 12. 1000 problems were generated for each value of l and N. It does not appear from this figure that there this is a phase transition in solubility. As problems become larger, the transition between soluble and insoluble problems appears to become more gradual. Whilst N is one of the natural sizes of problems, l is not since we need just $\log_2(l)$ bits to represent a number l. In search of a better view, in Figure 1B we plot $\log_2(l)$ along the x-axis. In addition, we use a wider range of N. We now can see that there is in fact a rapid transition between soluble and insoluble problems as $\log_2(l)$ increases.

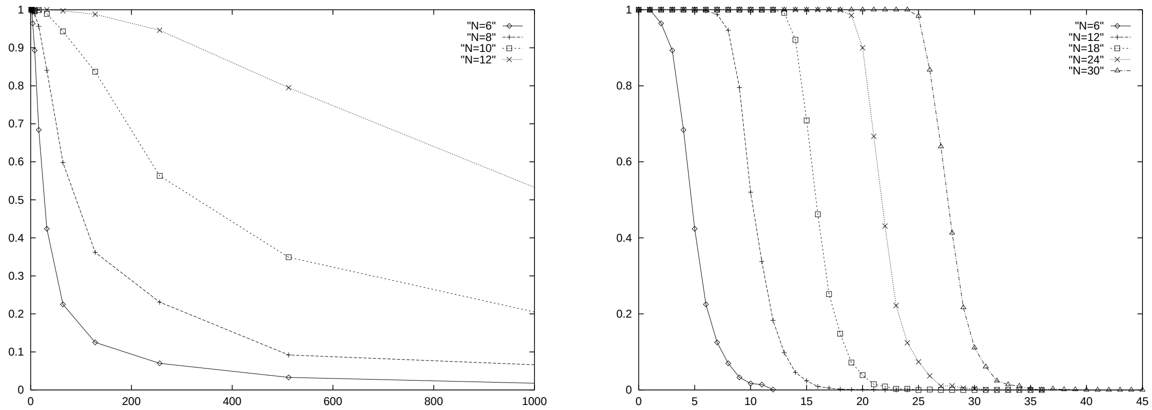

Figure 1. The probability of a bag of N integers randomly drawn from $[1, l]$ having a perfect partition. Bags are restricted to those with an even sum. **Figure 1A** Probability plotted against l for $N = 6$, 8, 10 and 12. **Figure 1B** Probability plotted against $\log_2(l)$ for $N = 6$, 12, 18, 24 and 30.

Next, we identify a control parameter, α for number partitioning problems. The phase transition appears to occur at a constant value, α_c of this parameter irrespective of the size of l and N. As with satisfiability [KS94] and constraint satisfaction [GMPW95], we can identify the control parameter via an "annealed" theory[1].

Consider the binary representation of the N integers that are being partitioned. The least significant bit in each bag has to sum to the same value modulo 2: i.e. the parity of the number of 1's in each bag has to be identical. On average, we expect just 1/2 the possible partitions to give the same parity for the least significant bit. For the next binary digit in each bag, we must sum the second most significant bits and the carry from the least significant bit. Again we expect 1/2 the possible partitions to give the same parity for this binary digit. The same argument applies to each binary digit in turn, of which there are $\log_2(l)$. If we now assume independence of each bit position, a partition is perfect if the sum of each of the binary digits has the correct parity. This will occur in a fraction, $(1/2)^{\log_2(l)}$ of the 2^N possible partitions. We predict that the phase transition occurs when just one of the possible partitions is perfect. That is, when,

$$2^N.(\frac{1}{2})^{\log_2(l)} = 1.$$

[1] We call it an annealed theory since we average probabilities independently over the different digit positions. This is by analogy with an annealed theory of materials which averages independently over sources of disorder. Both give good approximations in the thermodynamic limit.

Taking logarithms and simplifying gives,

$$\frac{\log_2(l)}{N} = 1.$$

This suggests that a suitable control parameter, α for number partitioning might be,

$$\alpha \ =_{\text{def}} \ \frac{\log_2(l)}{N}$$

We expect the phase transition to occur around the value $\alpha \approx 1$.

The choice of representing numbers in base 2 in fact does not affect our conclusions. In number base b then we expect each digit position to match modulo b with probability $1/b$, and there are $\log_b(l)$ digits base b. Both the resulting control parameter and the predicted position of the phase transition are unaffected.

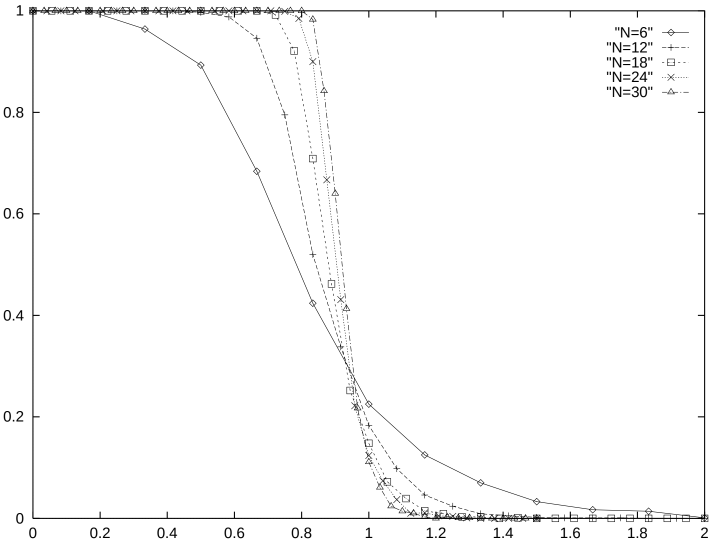

Figure 2. The probability of a bag of N integers randomly drawn from $[1, l]$ having a perfect partition for N = 6, 12, 18, 24 and 30. The probability is plotted against the control parameter, $\alpha = \log_2(l)/N$. Bags are restricted to those with an even sum.

In Figure 2, we plot the probability of a perfect partition exisiting against α. As predicted, the phase transition occurs close to $\alpha = 1$. In addition, we can now see that the transition sharpens as N increases.

4 Finite-size scaling

We next applied finite-size scaling methods [Bar83] to determine how the probability scales with problem size. Around some critical point, problems of all sizes should be indistinguishable except for a change of scale. This suggests that,

$$Prob(\text{perfect partition}) = f((\frac{\alpha - \alpha_c}{\alpha_c}) \,.\, N^{1/\nu}) \tag{1}$$

where f is some fundamental function, α_c is the critical point, and $N^{1/\nu}$ provides the change of scale [Bar83]. The fraction, $(\alpha - \alpha_c)/\alpha_c$ plays the same rôle as the reduced temperature, $(T - T_c)/T_c$ in physical systems. Equation (1) has a fixed point where the control parameter α equals α_c and for all N, the probability is the constant value $f(0)$. To estimate α_c, we take the fixed point to be the point where there is the minimum spread in probabilities. This gives $\alpha_c = 0.96 \pm 0.02$, where the errors indicate the range giving less than 9% spread. To compute ν, we assume (1) holds at the point of 50% probability, and calculate the median estimate for ν. This gives $\nu = 1 \pm 0.3$ where errors represent the upper and lower quartiles of estimates of ν. We define a rescaled parameter,

$$\gamma =_{\text{def}} \frac{\alpha - 0.96}{0.96} \,.\, N.$$

In Figure 3, we plot the probability of a perfect partition existing against γ. This graph suggest that finite-size scaling provides both a simple and accurate model for the scaling of probability with problem size.

5 Search cost

As in many other combinatorial problems, a peak in search cost is associated with the phase transition in solubility. Korf has proposed a branch and bound algorithm for number partitioning, called CKK [Kor95]. This uses the Karmarkar Karp (KK) heuristic based on set differencing to branch. He has shown this can give orders of magnitude better performance than a simple branch and bound algorithm with a greedy heuristic for branching. He claims that CKK outpeforms the best previously-known

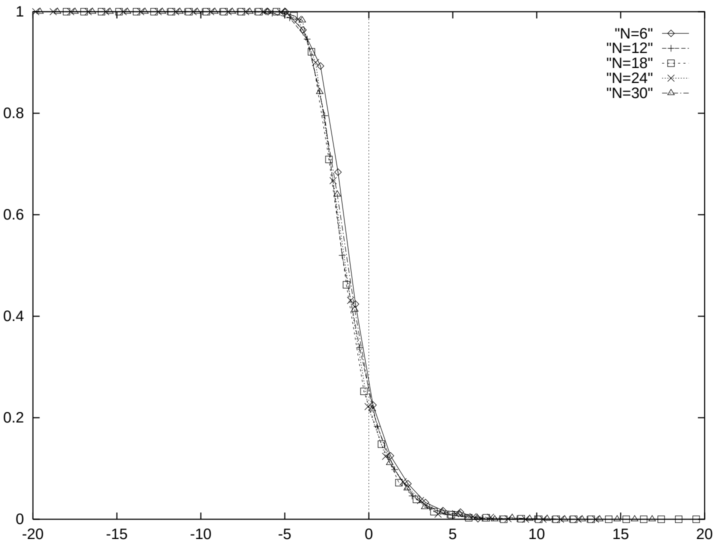

Figure 3. The probability of a bag of N integers randomly drawn from $[1, l]$ having a perfect partition for $N = 6$, 12, 18, 24 and 30. Probability plotted against rescaled parameter γ . Bags are restricted to those with an even sum.

algorithms, and will partition arbitrarily large bags of integers with up to 12 decimal digits. The dynamic programming algorithm for number partitioning has unacceptable time and space requirements for such problems.

In Figure 4A, we plot the average number of nodes searched by the CKK algorithm [2] against the rescaled parameter, γ, for N from 6 to 30. Plotting search cost against the rescaled parameter rather than the control parameter has been shown to be useful in other problem classes [SK, GMPW95]. For comparison, in Figure 4B we plot the average number of nodes searched by the simple greedy branch and bound algorithm also against the rescaled parameter, γ. In the soluble phase problems are, on average, easy. Problem hardness increases rapidly as we approach the phase boundary. Interestingly, problems appear to remain uniformly hard in the insoluble phase away from the phase boundary. Experiments out to larger l do not show problems becoming significantly easier (or harder) well away from the phase transition. This reflects the fact that both

[2] Note that we are here using this optimisation procedure to solve the decision problems. We discuss this point further in Section 7.

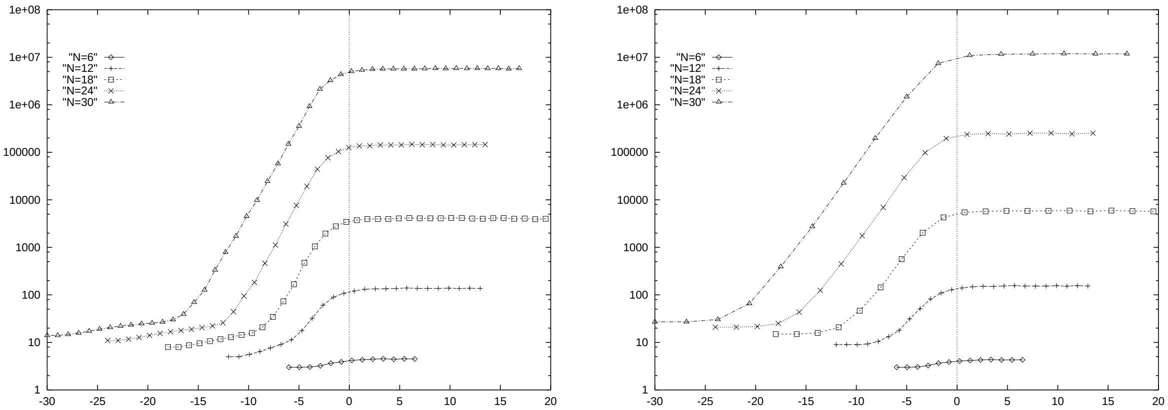

Figure 4. The average number of nodes searched plotted against the rescaled parameter, γ. N = 6, 12, 18, 24 and 30. Bags are restricted to those with an even sum. **Figure 4A** the CKK algorithm. **Figure 4B** the greedy branch and bound algorithm.

algorithms must perform a lot of exhaustive search to determine that a problem is insoluble. Algorithms which compute tighter bounds may perform significantly better on such overconstrained and insoluble problems.

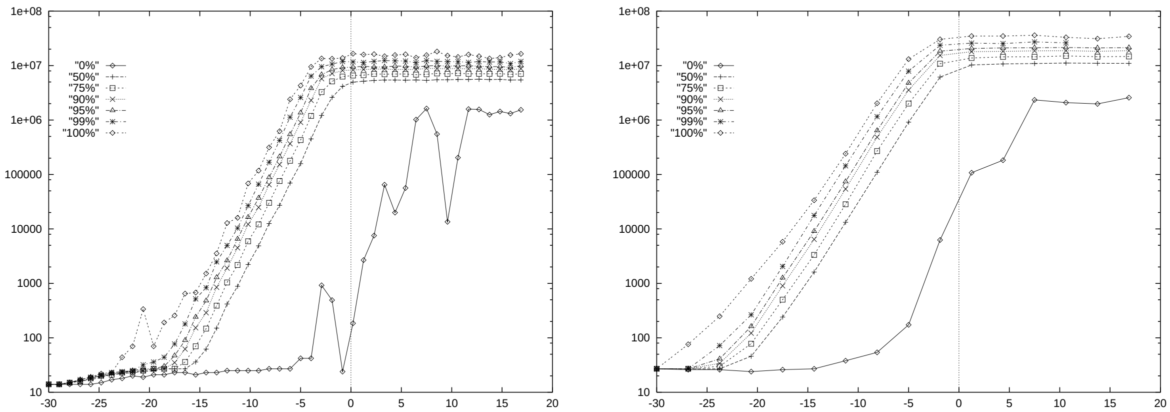

Figure 5. Percentiles of number of nodes searched plotted against the rescaled parameter, γ for $N = 30$. Bags are restricted to those with an even sum. **Figure 5A** the CKK algorithm. **Figure 5B** the greedy branch and bound algorithm.

The breakdown in performance can often reveal interesting effects. In Figure 5, we plot percentiles for the performance of the CKK algorithm and the greedy branch and bound algorithm for $N = 30$ again using the rescaled parameter along the x-axis. There is a hint that hard problems may occasionally turn up in the soluble phase for the CKK algorithm. To determine if hard problems turn up in the soluble phase at larger N and sample sizes, we tested 100,000 problems at N=100 and $\log_2(l) = 32$ (corresponding to $\gamma = -66.7$). Results are given in Table 1. Problem difficulty varies by 4 orders of magnitude with the easiest problem requiring less than 100 nodes of search and the hardest more than 3 million. The mean was 96795.0 nodes, nearly twice as large as the median problem which took 49300 nodes. This suggests that occasional hard problems can turn up in the soluble phase, especially at larger N, and that these problems can significantly influence the mean. Similar behaviour, to a more dramatic degree, has been observed in graph-colouring and propositional satisfiability [HW94, GW94a].

Percentile	Nodes
0.001%	94
0.01%	95
0.1%	97
1%	251
10%	4251
50%	49300
90%	243605
99%	634630
99.9%	1086803
99.99%	1709947
99.999%	3048765

Table 1. Percentiles for CKK algorithm on 100,000 problems at N=100 and $\log_2(l) = 32$. The mean was 96795.0 nodes. Bags are restricted to those with an even sum. All problems had a perfect partition.

Finally, in Figure 6 we plot maximum average search cost for all l at each value of N. From the gradient of Figure 6, we estimate that the worst average search costs grow as approximately $2^{0.85N}$ for the CKK algorithm and approximately $2^{0.90N}$ for the greedy branch and bound algorithm. Note that simply computing all possible partitions would give a maximum search costs that grows as 2^N.

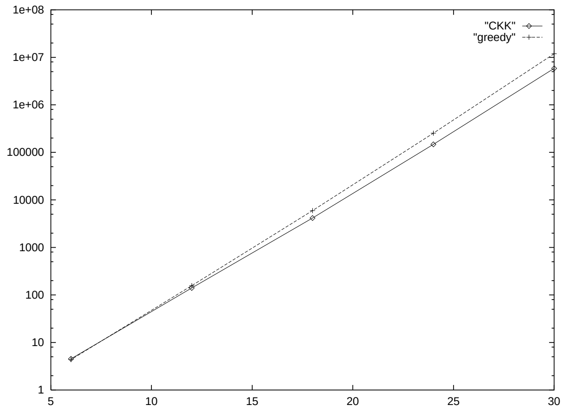

Figure 6. Maximum average nodes searched to find optimal partition of a bag of N integers randomly drawn from $[1, l]$. N plotted along x-axis. Bags are restricted to those with an even sum.

6 Poisson distribution

We next consider number partition problems where, as in [Fu89], random numbers are drawn from a Poisson distribution with parameter l. That is, where the probability of having a positive integer j is $\frac{e^{-l}.l^j}{j!}$. The mean and variance of such a distribution is l. For large l, we approximate the Poisson distribution by a normal distribution with mean and variance l. In addition, the Poisson distribution approximates the Binomial distribution for large n and small p. We therefore expect similar results (at least asymptotical) for certain types of normal and Binomial distributions. We again restrict attention to those bags of random numbers which have an even sum. We have observed similar results with bags with an odd sum, and with bags with both odd and even sums if we let $k = 1$. Results in this section are preliminary and are currently being extended.

In Figure 7A, we plot the probability of a perfect partition exisiting against the control parameter α, for samples of size 1000 at each point. The phase transition occurs at a larger value of α than with an uniform distribution. We expect the noise in this graph to disappear with larger sample sizes and with larger N. In Figure 7B, we plot the average number of nodes searched by the CKK algorithm against α, for N from 6 to 18. Problems become harder in the phase transition. As in graph-colouring and propositional satisfiability [HW94, GW94a], occasional hard problems turn up in the soluble phase, and these distort the mean. For example at $\alpha = 1/3$ and $N = 18$, all problems were soluble but one took 6624 nodes. This compares to a median of 12 and a mean of 52.0 nodes at this point. The worst median of 12281.5 nodes, and the worst mean of 12086.274 nodes both occured in the phase transition at $\alpha = 1.944$, close to the fixed point α_c.

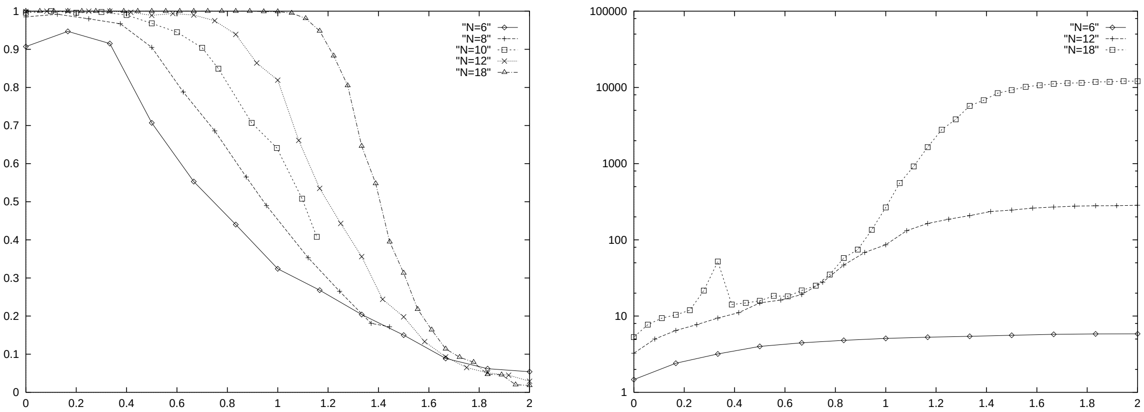

Figure 7. Partition problems with numbers drawn from a Poisson distribution with parameter l. Bags are restricted to those with an even sum. **Figure 7A** Probablility of a perfect partition existing against α. **Figure 7B** Average number of nodes searched by the CKK algorithm against α.

All our experiments so far have used even N. For random numbers drawn from a uniform distribution, we have seen almost identical results for odd N. However, for random numbers drawn from a Poisson distribution, we have observed significantly different results for odd N. In particular, the phase transition in solubility for odd N occurs at a noticeably smaller value of α than for even N. For odd N, it appears that $\alpha_c < 1$, whilst for even N, α_c appears nearer to 2 than to 1. This difference is perhaps not too surprising. Consider $2n+1$ random numbers drawn from a Poisson distribution. As we increase l (and thus α), the numbers will tend to cluster more tightly around the value l. It will therefore be hard to find a perfect partition in which n of the numbers add up to l or so more than the remaining $n+1$ numbers. By comparison, with a uniform distribution, increasing l (and thus α) increases the spread.

7 Optimization problems

There is not a phase transition in the solubility of the optimization version of number partitioning since, by definition, optimization problems are always soluble. However, we do see a transition in problem hardness for the optimization problem. This transition coincides with the phase transition in solubility for the decision problem. This can be seen, for example, in Figure 5 of [Kor95] around $N = 40$. The CKK and the greedy branch and bound algorithms simultaneously solve the decision problem and the optimization problem. Because we used an optimization procedure to solve the decision problem, our graphs of search cost could have been labelled "optimization search cost" just as accurately as "decision problem search cost". Optimization is usually easy in the soluble phase as we can immediately terminate search when a perfect partition is found. By comparison, optimization is hard in the insoluble phase as branches cannot usually be closed till deep in the search. We must therefore search through a large number of possible partitions before concluding that we have found the optimal one. The transition between hard and easy optimization problems coincides with the appearance of perfect partitions and thus with the phase transition in the solubility of the decision problem.[3].

8 Imperfect partitioning

We can repeat our derivation of an "annealed" theory to identify a control parameter for partitioning into less than perfect partitions (*i.e.* for $k > 1$). For simplicity assume k is a power of 2 (although this argument probably extends with little change to numbers that are not powers of 2). We now don't care about the bottom $\log_2(k)$ bits. We do, however, insist that the top $\log_2(l) - \log_2(k)$ bits in each bag add up with the same parity. The phase transition therefore occurs when,

$$2^N.(\frac{1}{2})^{\log_2(l)-\log_2(k)} = 1.$$

Taking logarithms and simplifying gives,

$$\frac{\log_2(l/k)}{N} = 1.$$

[3]This section has benefited from discussions with Barbara Smith.

This suggests that a suitable control parameter, α for imperfect partitions would be,

$$\alpha =_{\text{def}} \frac{\log_2(l/k)}{N}$$

We expect the phase transition to occur again around the value $\alpha \approx 1$.

9 Multi-way partitioning

We can also repeat our derivation of an "annealed" theory to identify a control parameter for multi-way partitioning. Given a natural number m, we wish to partition the bag S of N numbers into m bags $S_1, S_2, \ldots S_m$ such that

$$|\sum_{i \in S_{m_1}} a_i - \sum_{i \in S_{m_2}} a_i| \leq k$$

for each $1 \leq m_1 < m_2 \leq m$.

We first assume that the sum of the numbers in S is an exact multiple of m and $k = 0$. As a result each S_i must have the same sum. We repeat our argument about digit positions to derive a control parameter and expected phase transition point. If we are partitioning a bag of numbers m-ways, then at each digit position, the first $m-1$ bags must each have a given sum modulo 2 and this is expected to happen with probability $(1/2)^{m-1}$. (The last bag is guaranteed to have the right digit sum if the first $m-1$ do.) As there are m^N possible m-way partitions and $\log_2(l)$ bit positions, we now predict the phase transition when

$$m^N.(\frac{1}{2})^{(m-1)\log_2(l)} = 1.$$

Rearranging and simplifying this is equivalent to:

$$\frac{(m-1)\log_m(l)}{N} = 1$$

This suggests that the general control parameter for m-way partitioning should be:

$$\alpha =_{\text{def}} \frac{(m-1)\log_m(l)}{N}$$

and again we expect the phase transition to occur around $\alpha \approx 1$. Note that this coincides with the control parameter derived in Section 3 for 2-way partitioning.

It may be necessary to include a correction to the above discussion, because there are in fact only $N^m/m!$ essentially different partitions, the order of bags being irrelevant. Although this only leads to a constant correction term and therefore does not affect the annealed limit, it may prove to be helpful in the study of comparatively small finite problem sizes such as studied in this paper.

Finally, we can simply drop the assumption that the sum r of the numbers in S is an exact multiple of m if we let $k = 1$. In this case, r modulo m of the bags will have a sum 1 greater than the rest. Our analysis, however, remains essentially unchanged.

It will obviously be very interesting to see experimentally if the analysis given here and in the previous section leads to appropriate parameters for examining more general partitioning problems than those studied in this paper, and we intend to do so.

10 Conclusions

The phase transition phenomena described here appear to be widespread. One common feature of this and other combinatorial problems is "frustration". A frustrated system has a large number of near optimal equilibrium configurations which cannot be improved by local changes (for example, by swapping a pair of integers between partitions). Frustration also arises in statistical mechanics when describing spin-glasses and their phase transition. Because of this frustration, it is very difficult to find the global optimum. It remains an open question if phase transitions and frustration will be seen in all NP-complete problems: we do not know of any evidence suggesting it will not. Whatever the answer, computational complexity appears to be closely associated with phase transitions and frustration.

Acknowledgments

The second author is supported by a HCM Postdoctoral Fellowship. We thank the members of the Mechanized Reasoning Group at Trento for many CPU cycles donated to these and other experiments. We also thank Richard Korf for providing us with his code for the CKK and greedy branch and bound algorithms, and for suggesting number partitioning as one of the experimental themes at the First AI and OR Workshop, Timberline, 1995. Finally we thank the members of the APES research group for discussion and feedback.

References

[Bar83] Michael N. Barber. Finite-size scaling. In *Phase Transitions and Critical Phenomena, Volume 8*, pages 145–266. Academic Press, 1983.

[CKT91] P. Cheeseman, B. Kanefsky, and W.M. Taylor. Where the really hard problems are. In *Proceedings of the 12th IJCAI*, pages 331–337. International Joint Conference on Artificial Intelligence, 1991.

[Fu89] Y. Fu. The uses and abuses of statistical mechanics in computational complexity. In D. Stein, editor, *Lectures in the Sciences of Complexity*, pages 815–826. Addison-Wesley Longman, 1989.

[GJ79] Michael R. Garey and David S. Johnson. *Computers and intractability : a guide to the theory of NP-completeness.* W H Freeman, 1979.

[GMPW95] I.P. Gent, E. MacIntyre, P. Prosser, and T. Walsh. Scaling effects in the CSP phase transition. Research report 95/179, Department of Computer Science, University of Strathclyde, March 1995. To appear in proceedings of PPCP-95.

[GW94a] Ian P. Gent and Toby Walsh. Easy problems are sometimes hard. *Artificial Intelligence*, 70:335–345, 1994.

[GW94b] Ian P. Gent and Toby Walsh. The hardest random SAT problems. In Bernhard Nebel and Leonie Dreschler-Fischer, editors, *KI-94: Advances in Artificial Intelligence. 18th German Annual Conference on Artificial Intelligence*, pages 355–366. Springer-Verlag, 1994.

[GW95] I.P. Gent and T. Walsh. The TSP phase transition. Research report 95/178, Department of Computer Science, University of Strathclyde, 1995. To be presented at the First International workshop on AI and OR.

[HW94] T. Hogg and C. Williams. The hardest constraint problems: A double phase transition. *Artificial Intelligence*, 69:359–377, 1994.

[Kor95] R.E. Korf. From approximate to optimal solutions: A case study of number partitioning. In *Proceedings of the 14th IJCAI*. International Joint Conference on Artificial Intelligence, 1995.

[KS94] S. Kirkpatrick and B. Selman. Critical behavior in the satisfiability of random boolean expressions. *Science*, 264:1297–1301, May 27 1994.

[MSL92] David Mitchell, Bart Selman, and Hector Levesque. Hard and easy distributions of SAT problems. In *Proceedings, 10th National Conference on Artificial Intelligence*. AAAI Press/The MIT Press, July 12-16 1992.

[Pro94] Patrick Prosser. Binary constraint satisfaction problems: Some are harder than others. In *Proceedings of ECAI-94*, pages 95–99, 1994.

[SK] B. Selman and S. Kirkpatrick. Critical behavior in the computational cost of satisfiability testing. *Artificial Intelligence*. To appear.

[Smi94] Barbara M Smith. Phase transition and the mushy region in constraint satisfaction problems. In *Proceedings of ECAI-94*, pages 100–104, 1994.