

**Solving large sparse MAXCUT problems  
using an interior point cutting plane  
algorithm**

**John E. Mitchell**

Mathematical Sciences,  
Rensselaer Polytechnic Institute,  
Troy NY 12180 USA

email: [mitchj@rpi.edu](mailto:mitchj@rpi.edu)

<http://www.math.rpi.edu/~mitchj>

Seattle INFORMS Conference

October 25–28, 1998

Session MC25

**Abstract**

We describe computational experience with interior point cutting plane algorithms for linear ordering problems and MAXCUT problems, two classical integer programming problems. We discuss warm starting an interior point method in such a setting, as well as other important features of our methods. We show that the use of an interior point method has enabled the solution of certain problems in far less time than that required by a simplex cutting plane algorithm.

## **1 Overview**

The Ising spin glass problem:

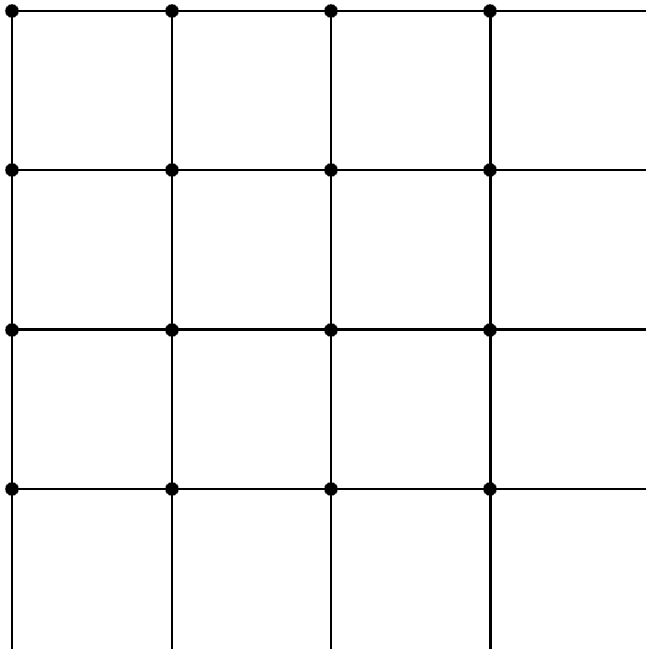
- Definition, polyhedral theory
- Algorithm
- Computational results

## 2 Finding the ground state of an Ising Spin Glass

- References: Grötschel *et al.*, 1985, De Simone *et al.*, 1996.
- Problem in glassy dynamics in statistical physics
- An *Ising spin glass* is a model of a magnetic material, and it consists of a grid of magnetic spins.
- Each spin  $S_i$  is in one of two states, which we call “up” and “down”; we assign  $S_i$  the value  $+1$  if the spin is up and  $-1$  if the spin is down.
- We assume the grid is an  $L \times L$  square grid embedded on a torus.
- Further, we assume that the interactions between spins are restricted to neighbours, that is, we consider the short range model with Ising spins  $S_i$ .

## Graph representation:

$4 \times 4$  grid on a torus:



- Know **node interactions**,  $\pm 1$ . These are edge weights.
- Find **state** of each vertex.
- If edge between neighbouring vertices has value  $J_{ij} = +1$ , want vertices to be in *same* state.
- If edge between neighbouring vertices has value  $J_{ij} = -1$ , want vertices to be in *opposite* states.

### 3 Integer programming model

- Minimize the Hamiltonian of the energy:

$$H := - \sum_{\text{neighbours } i,j} J_{ij} S_i S_j$$

with  $S_i = \pm 1$ .

- Can be modelled as a **Max Cut** problem: all the *up* vertices on one side of the cut, and all the *down* vertices on the other side, with edge weights derived from  $J_{ij}$ .
- So solve

$$\min\{c^T x : x \text{ is the incidence vector of a cut}\}$$

Here,  $x$  has one component for each *edge*. The optimal value of this problem is *even*.

## Polyhedral theory:

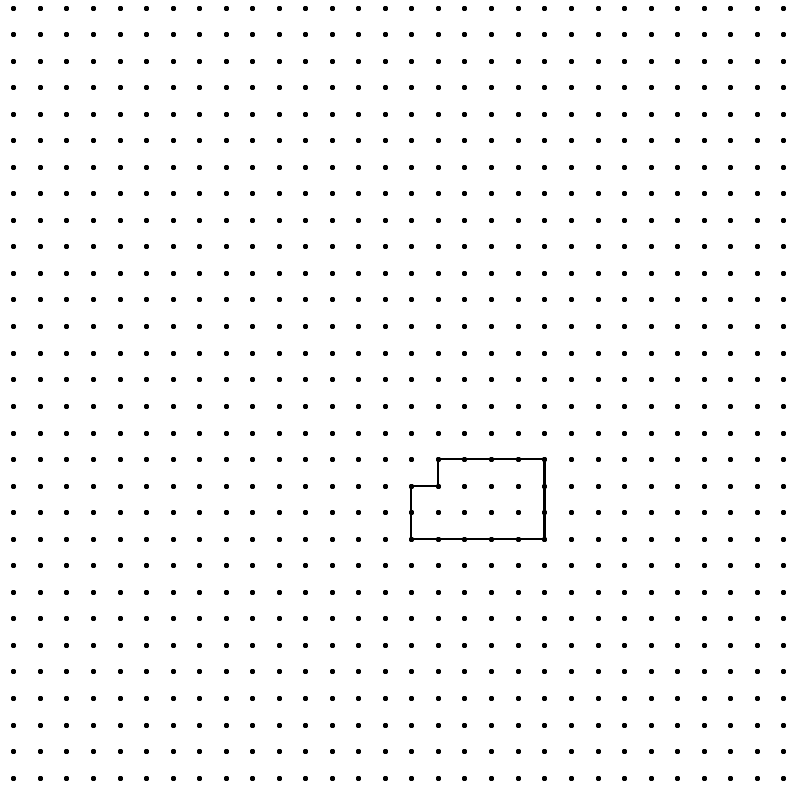
- **Initial relaxation:**

$$\min\{c^T x : 0 \leq x_e \leq 1\}$$

- **Cutting planes from cycles:** Every cycle and every cut intersect in an even number of edges. Gives the following facet defining inequality for every subset  $F$  of odd cardinality of every chordless cycle  $C$ :

$$x(F) - x(C \setminus F) \leq |F| - 1$$

- All the **squares** in the grid are chordless cycles. There are also many other chordless cycles.
- Any integral vector that satisfies all the cycle inequalities must be the incidence vector of a cut.
- There are also other families of valid inequalities.



**Cycle for added inequality 673**  
**for a  $30 \times 30$  grid**







## 4 Cutting plane algorithm

1. **Initialize**
2. **Solve current relaxation**, using a primal-dual interior point method. This gives a lower bound on the optimal value of the Ising spin glass problem.
3. **Separation:** Check all cycle inequalities corresponding to squares. Bucket sort resulting violated inequalities by violation and add a subset of constraints to the relaxation. If want more constraints, look at longer cycles. Drop any constraints that no longer appear important.
4. **Primal heuristic:** Look for the incidence vector of a cut close to the solution to the current relaxation. Store the resulting solution if it is better than the best cut found previously.
5. **Check for termination:** If the difference between the lower bound and the value of the best cut found so far is less than two, STOP with optimality. If no violated cycle inequalities are found and if the difference is greater than two, use branch and bound to complete the solution.
6. **Loop:** return to step 2

### **Refinements to the cutting plane method:**

- Suffices to solve the relaxations **approximately**, gradually tightening the degree of accuracy.
- If primal and dual values are sufficiently close that it appears that solving this relaxation will solve the MaxCut problem, do not look for cutting planes.
- **Restart** after adding cutting planes by backing up the primal solution to a convex combination of the vector of halves and the current point. The dual solution is set to an earlier dual solution.
- About 40% of the runtime is spent on the primal heuristic. This heuristic looks for chains of vertices which can be switched.
- Every tenth relaxation is solved to an accuracy of  $10^{-8}$ .

### Adding a cut algebraically:

- The problem is formulated as the **primal** problem.
- Cutting planes are **rows** of  $A$ .
- Add constraints  $Gx \leq g$ .
- Modify relaxation with additional slack variables  $s_g$ :

$$\begin{aligned}
 \min \quad & c^T x \\
 \text{subject to} \quad & Ax + s = b \\
 & \mathbf{G}\mathbf{x} + \mathbf{s}_g = \mathbf{g} \\
 & 0 \leq x \leq e, \quad 0 \leq s \leq u, \quad \mathbf{0} \leq \mathbf{s}_g \leq \mathbf{u}_g
 \end{aligned}$$

- **Restart**: any point in the interior of the convex hull of feasible integer points is feasible in the new relaxation. For MAXCUT, can take  $0.5e$ . This point can be improved as the iterations proceed, or a sequence of possible restart points can be stored.

## Restarting the dual

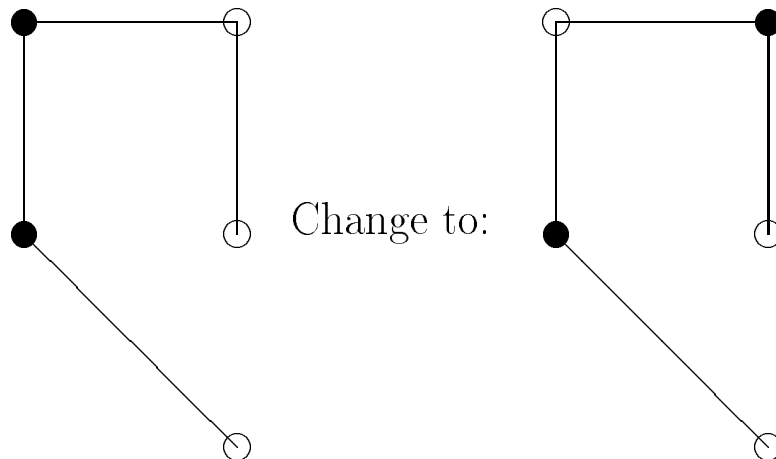
- Dual problem is:

$$\begin{aligned}
 \max \quad & b^T y + \mathbf{g}^T \mathbf{y}_g - e^T w_x - u^T w_s - \mathbf{u}_g^T \mathbf{w}_g \\
 \text{subject to} \quad & A^T y + \mathbf{G}^T \mathbf{y}_g + z - w = c \\
 & y + z_s - w_s = 0 \\
 & \mathbf{y}_g + \mathbf{z}_g - \mathbf{w}_g = \mathbf{0} \\
 & z, z_s, \mathbf{z}_g \geq 0, \quad w, w_s, \mathbf{w}_g \geq 0.
 \end{aligned}$$

- Restart with  $y_g = 0$ ,  $z_g = w_g = \epsilon e$ , say  $\epsilon = 10^{-3}$ .

## Primal heuristic

- Construct a cut from the fractional solution  $x$ .
- Use local improvement to get a better solution:
  - 1-change: If moving one vertex to the other side of the cut improves the solution, make the change.
  - 2-change: If swapping two vertices on either side of the cut, make the change.
  - $k$ -change: Look for chains of vertices where each vertex can be switched from one side to the other, and switching everything results in an improvement, even though individual switches do nothing. We look for chains of length up to 10 vertices.
  - Example:



Here, no 1-change gives improvement, need a 2-change.

## 5 Computational results

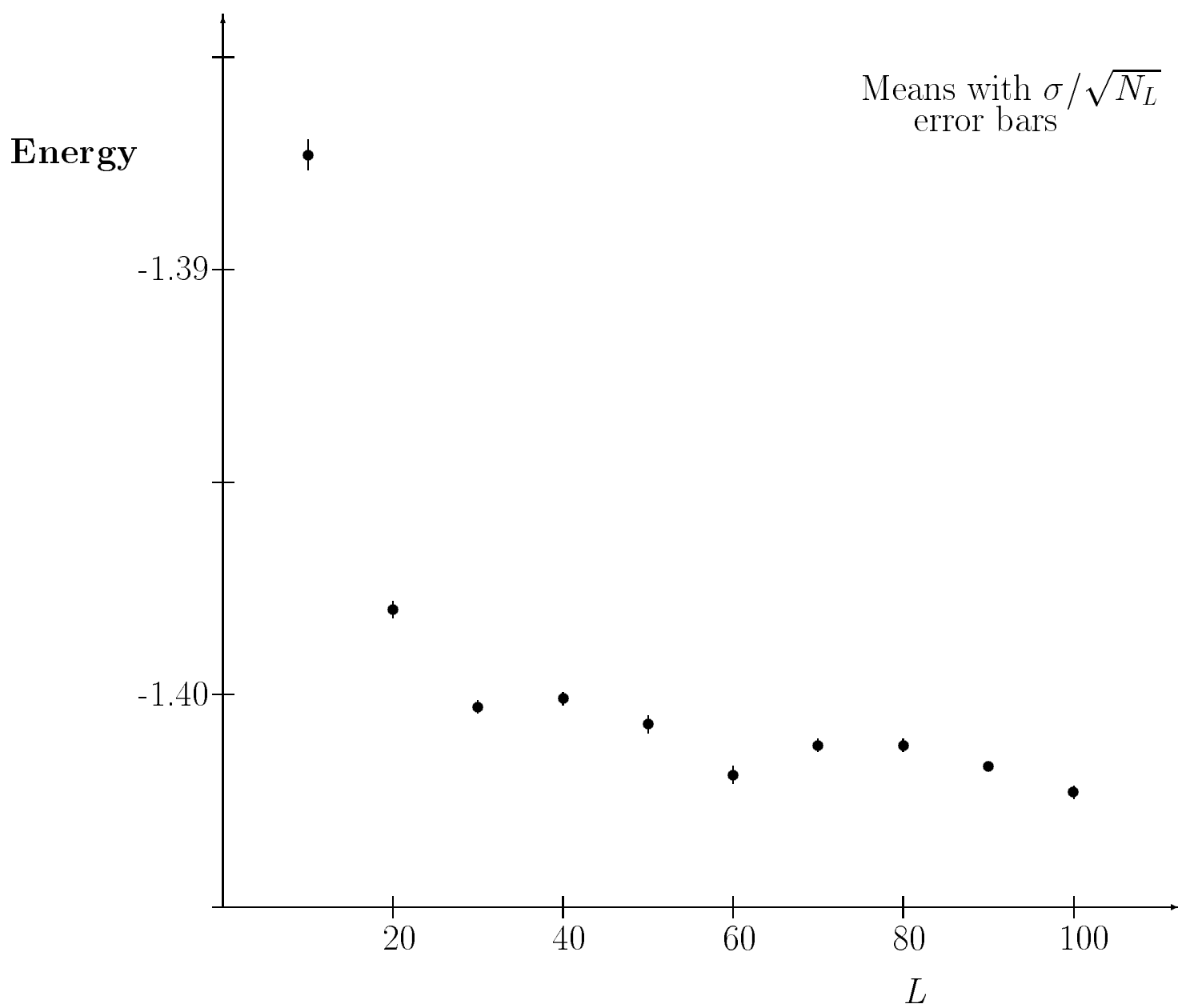
- **Randomly generated problems**

- Grid sizes up to  $100 \times 100$ .
- Interactions  $J_{ij}$  equally likely to be  $+1$  or  $-1$ .

- **Computational environment**

- All runs performed on a Sun SPARC 20/71 using SunOS. All runtimes will be quoted in **seconds**.
- Interior point code written in Fortran. Fortran command **ETIME** used for timings.

## Ground state energy



## 6 Ground state energy

- **Exponential model:**

$$E_e(L) = E_e^\infty + ce^{-aL}$$

Best fit:  $E_e^\infty = -1.3997$ .

With just  $L \geq 70$ : Best fit:  $E_e^\infty = -1.4017$ .

(De Simone *et al.* (1996) give  $-1.4007 \pm 0.0003$ .)

- **Quadratic model:**

$$E_q(L) = E_q^\infty + cL^{-2}$$

Best fit:  $E_q^\infty = -1.4016$ .

With just  $L \geq 70$ : Best fit:  $E_q^\infty = -1.4031$ .

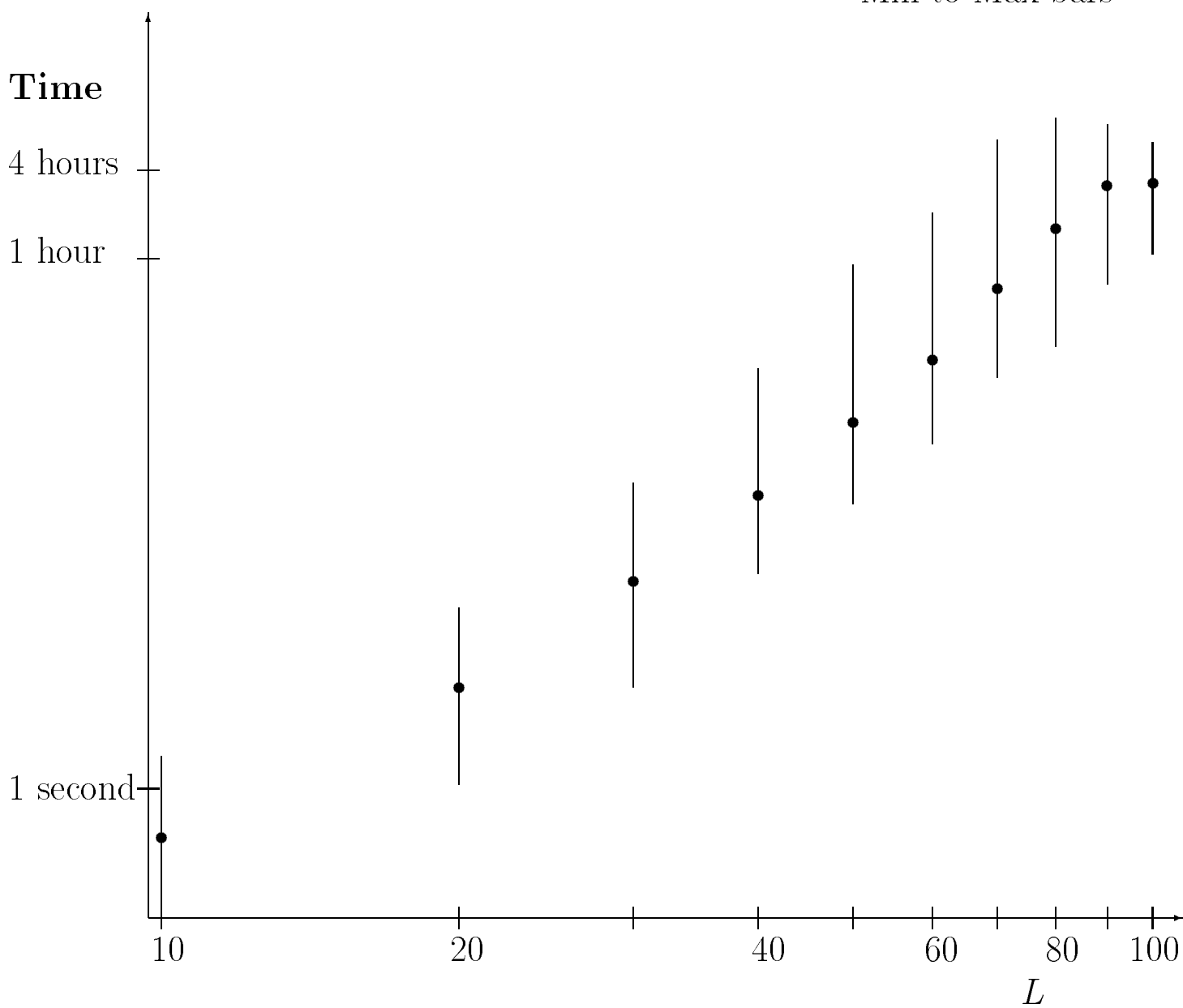
(De Simone *et al.* (1996) give  $-1.4022 \pm 0.0003$ .)

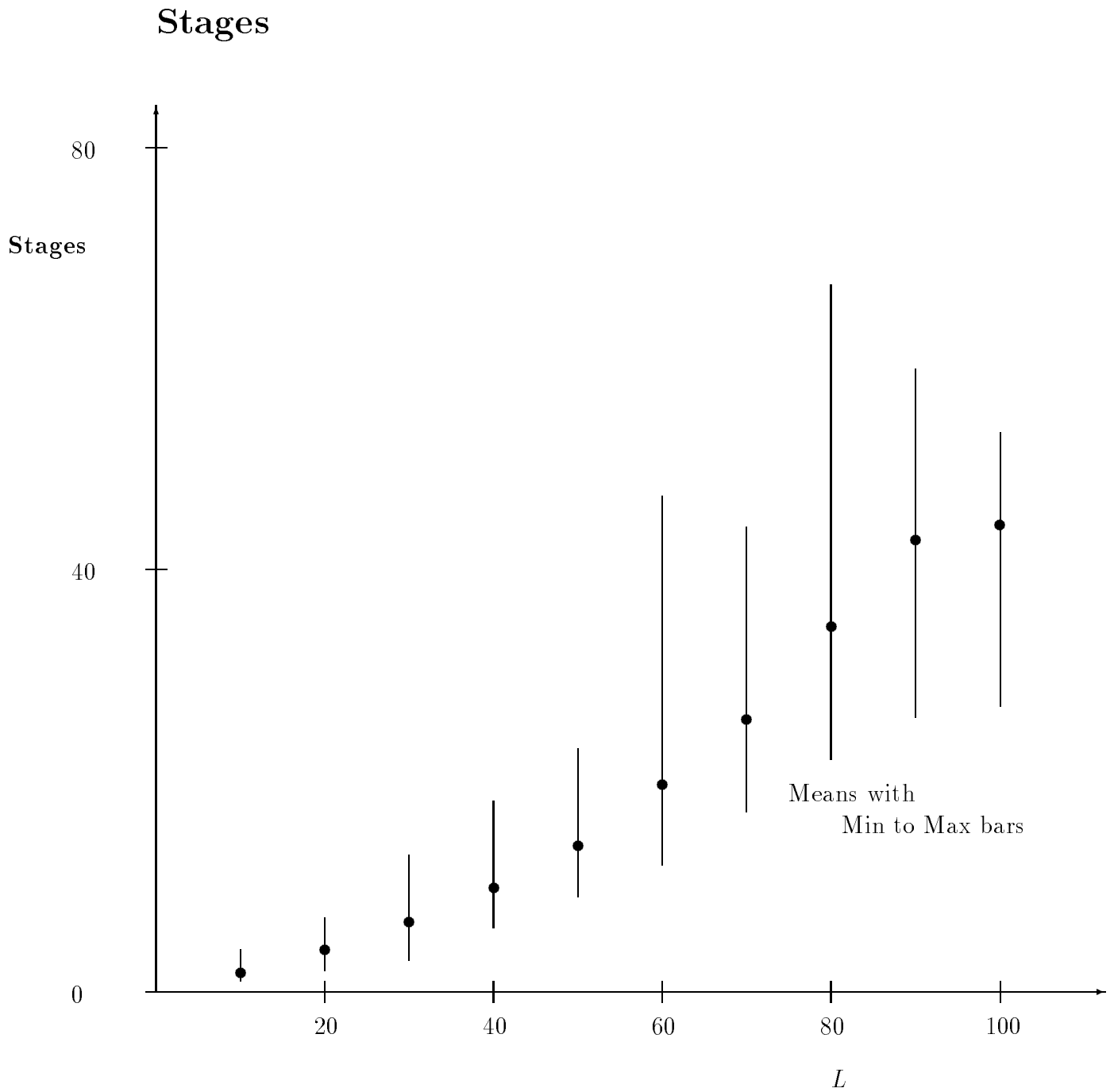
$L$	$N_L$	Mean	Std Dev	Minimum	Maximum
10	5100	.47	.24	.15	1.68
20	3100	4.84	2.11	1.07	16.55
30	2100	24.49	11.67	4.80	114.80
40	1200	93.08	54.34	27.57	650.43
50	720	290.75	201.46	82.23	3370.88
60	440	772.37	707.68	207.42	7450.18
70	384	2245.92	2277.75	580.68	22768.40
80	310	5787.28	5369.17	935.38	32470.40
90	280	11320.24	6254.31	2474.20	28785.30
100	229	11873.59	3609.73	3855.02	21922.60

**Table 2: Time (seconds) to solve Ising spin glass problems**

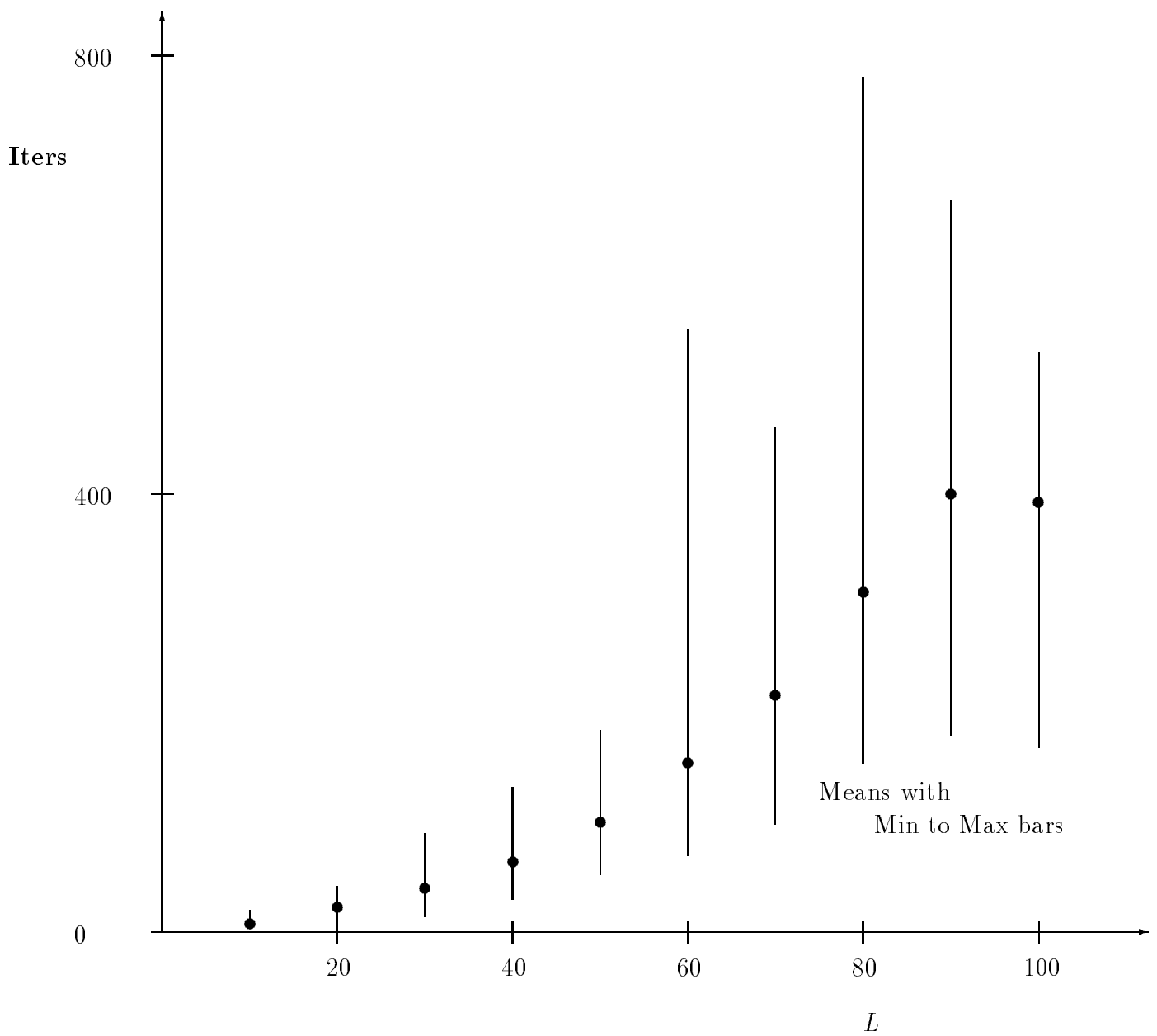
### Run times

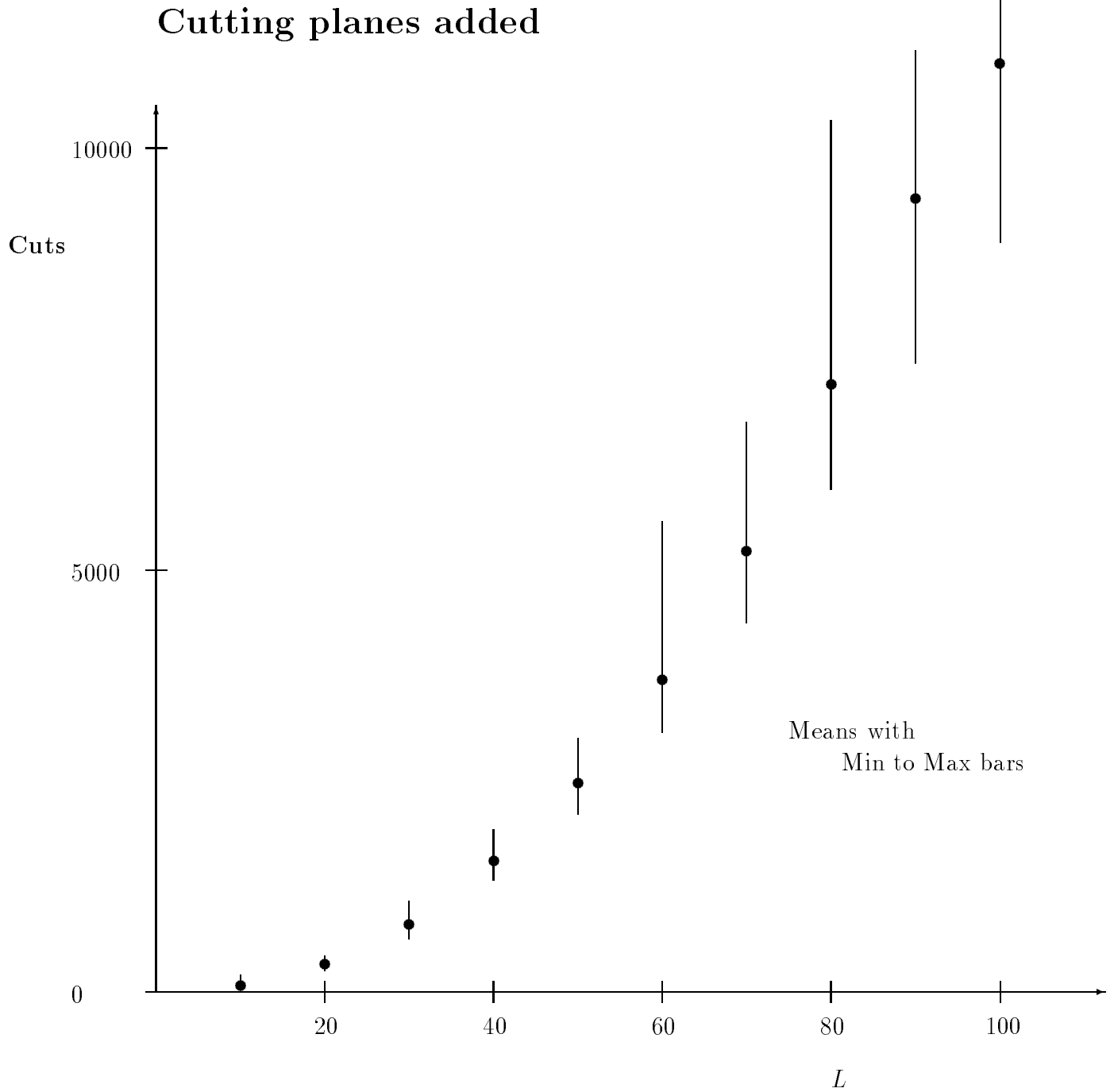
Means with  
Min to Max bars





### Interior point iterations





$L$	Primal Heuristic	Integer Prog	Linear Prog
10	41	55	39
20	36	54	42
30	33	51	46
40	34	49	49
50	36	49	50
60	38	49	50
70	36	44	55
80	38	44	56
90	42	47	52
100	54	61	38

**Table 3: Percentage of runtime spent on each part of algorithm**

## 7 Run times

- Problems of size  $100 \times 100$  are solved in an average of **3hrs, 20 minutes**. By comparison, a simplex cutting plane code using CPLEX 3.0 on a Sun SPARC-station 10 required up to a **day** to solve problems of size  $70 \times 70$ . (De Simone *et al*, 1996.)
- Fitting  $\log(\text{Time})$  versus  $\log(L)$  gives a slope of approximately 4. Thus, runtime only grows at rate  $L^4$ . Note that the number of vertices increases at a rate of  $L^2$ . The simplex runtime appeared to increase at a rate of approximately  $L^6$ .
- The number of iterations per linear program averages out to around 8. Fewer iterations are required in earlier relaxations and more iterations for later relaxations.
- One possible way to improve the algorithm would be to **crossover** from an interior point method to a simplex method after a certain number of stages.

## 8 Conclusions

- The interior point code is able to solve far larger instances than those previously reported.
- The ground state energy estimates provided by this model using just the data for large  $L$  are lower than previous estimates.
- The interior point solver is a research code. For example, it does not use supernodes when calculating the Cholesky factorization. We believe that current high quality interior point solvers are at least 2–3 times faster than our code for linear programming problems.

$L$	$N$	Need B&B	Mean	Std Dev	Minimum	Maximum
10	5100	0	-1.3873	.0466	-1.5600	-1.2400
20	3100	0	-1.3980	.0228	-1.4700	-1.3200
30	2100	0	-1.4003	.0151	-1.4444	-1.3467
40	1200	0	-1.4001	.0109	-1.4325	-1.3700
50	720	0	-1.4005	.0098	-1.4368	-1.3744
60	440	0	-1.4019	.0077	-1.4233	-1.3783
70	386	2	-1.4012	.0064	-1.4196	-1.3829
80	328	18	-1.4012	.0059	-1.4163	-1.3869
90	290	10	-1.4017	.0043	-1.4128	-1.3916
100	236	7	-1.4023	.0047	-1.4134	-1.3864

**Table 1: Ground state energy of Ising spin glass problems**