Exercizes on combinatoric optimisation

Bert Kappen

November 15, 2020

Run the program `makedata.m` to generate an instance of the following combinatoric optimisation problem:

$$E = -\frac{1}{2} x^t w x,$$

with w an $n \times n$ symmetric matrix with zero diagonal and $x = (x_1, \ldots, x_n)$ a binary vector: $x_i = \pm 1$. Finding the minium of E is intractable in general because x is binary (what is the solution when x is real and $\|x\|^2 = 1$?).

For binary x and for specific choices of w, the problem can be significantly more or less difficult. For instance, if all elements w_{ij} are positive or zero, there are two optimal solutions:

$$x = \pm(1, \ldots, 1)$$

(show this result). This solution minimizes the cost for each interaction term separately. These systems are called ferro-magnetic.

Instead, when w_{ij} has arbitrary sign, there is typically no global solution x that minimizes each term $w_{ij} x_i x_j$. Because not all terms can be satisfied simultaneously, these systems are called frustrated. A simple example is the interaction matrix

$$\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & -1 \\ 1 & -1 & 0 \end{pmatrix}$$

the global minimum is the best compromise for all interaction terms taken together.

Exercises

1. (a) When x is real, the minimum of $E(x)$ with $\|x\|^2 = 1$ is easily computed. What is the solution? What is the computational complexity to find the solution?

   (b) Show that when $w_{ij} > 0$ for all i, j, the minimum of $E(x)$ is $x = \pm(1, \ldots, 1)$.

2. Write a computer program for the iterative improvement method. Take as neighborhood R single spin flips. Use K multiple restarts using different initial states and compute the lowest energy. Use multiple runs $N_{\text{runs}} = 20$ of the algorithm to assess whether the results are reproducible.

(a) Use `makedata.m` to generate problem instances of ferro-magnetic and frustrated problems for $n = 500$. Compare the performance of the interative improvement method for these problems in terms of number of restarts K that are need to obtain reproducible results. As performance measure take energy of the final solution averaged over the N_{runs} and its standard deviation.

(b) For the frustrated problem `w500` (file included), produce a table or curve of the quality of the solution versus the CPU time. I find

K	CPU (sec)	E
20	0.1	-6281 ± 62
100	0.5	-6322 ± 40
200	1.0	-6341 ± 43
500	2.5	-6386 ± 45
1000	5.0	-6405 ± 51
2000	10.3	-6429 ± 44
4000	20.5	-6442 ± 29

The CPU time is for a single run of the algorithm (not for the N_{runs} runs).

The best performance that I found in any run with iterative improvement on this problem is -6528.

3. Write a computer program for the simulated annealing method. Take as neighborhood R single spin flips. The SA method does not use restarts ($K = 1$). Implement both the exponential schedule and the Aarts and Korst (AK) schedule.

(a) Assess the performance of the SA algorithm on the `w500` problem using the AK schedule. Produce a table or curve of the quality of the solution versus the CPU time. I find

$\Delta\beta$	L	CPU (sec)	E
0.1	500	0.03	-6330 ± 90
0.01	500	0.7	-6550 ± 35
0.001	500	7.8	-6570 ± 30
0.001	1000	20.6	-6594 ± 28

(b) Assess the performance of the SA algorithm on the `w500` problem using the exponential schedule. Produce a table or curve of the quality of the solution versus the CPU time. I find

f	L	CPU (sec)	E
1.01	500	0.25	-6495 ± 55
1.001	500	2.3	-6558 ± 31
1.0002	500	11.0	-6577 ± 36
1.0002	1000	31.0	-6598 ± 29

The best performance that I found in any run with simulated annealing on this problem is -6624.