T-79.4101 Programming Assignment 2: Spin Glass Ground State Problem

Remy Rojas 486404

April 12, 2015

1 Problem Description

The problem consists of the so called ”Spin Glass” problem. It can be interpreted as a graph problem, by stating that nodes attached to one interact with eachother. The weight of the edge, negative or positive, translates the interaction in sucha a way that on a negative edge, nodes should be of oposite nature, and on a positive edge, nodes should have the same nature. The proposed equation defines how to acheive an optimal (minimum) setting with such specifications.

Unfortunately, the equation is not linear and thus not possible to solve with the tool provided: lpsolve. The solution is then to provide a mixed integer linear program MIP to lpsolve which is equivalent to the former problem.

2 Algorithm

There is not much to discus in the algorithm of the program since all the work is handled first by the library functions to interpret the graph, then printing the output function, readable by lpsolve. The difficulty of this assignment is to convert a non-linear equation into an MIP.

A linear equation is another way of saying first degree equation. No variable is multiplied by itself or other variables. The original equation is the sum of variable products. These variable products are the values of the nodes, two by two. Different nodes output -1, and equal nodes output 1 when multiplied. This can be translated into 0 for -1 and 1 to 1, since it’s easier to work with binary values.

We soon realize the problem has transformed into defining exclusive "or" XOR for each pair of nodes connected by an edge. Luckily, XOR can be encoded into a MIP [?].

```
x1, x2 are 2 nodes connected by and edge
x1*x2 is not possible, instead we try to model
x1 XOR x2
(x1 AND !x2) OR (!x1 AND x2)
mathematically, this can be written as:
x1*(1-x2) + x2*(1-x1)
= x1 + x2 -2*x1*x2
```

We still have a product term that needs to change. By applying linear programming:

```
b12 replaces x1*x2
b12 < x1 max value of b12 will be 1
b12 < x2
b12 > x1 + x2 -1 ensures b12 will not be 1 and not 0 in case x1
    +x2=2
b12 > 0
```

This way, we have encoded XOR in linear programming. To obtain the -1 and 1 desired output:

```
let a12 = x1 + x2 -2*b12
writing 2*a12 - 1
ensures values 1 for XOR 1 and -1 for XOR 0
```

3 Experiments

The programming and experimentation were carried out on a Linux Kernell System, Running Ubuntu 14.04 Xfce Desktop, Kernell version 3.13.0-48 with 8GB RAM memory and Intel i7 processor at 2.4GHz.

The program works properly and the solutions are found within 5seconds for the longest graphs, almost instantly for the shorter.

4 Conclusions

The assigment implied a lot more reasoning than actual code. The output format is simple and very close to reality. The Graph library also helped a lot of by interpreting effectively the instance graph provided. The solution was to reduce the problem to a logical proposition, and interpreting the proposition back as a mathematical formula.

References

[1] FICO Express Optimization Suite: MIP formulations and linearizations Quick Reference