Problem 5

Víctor Alcázar
Kosmas Palios
Albert Ribes

February 27, 2017

Problem statement

The Experimental Cuisine problem

We are feeling experimental and want to create a new dish. There are various ingredients we can choose from and we'd like to use as many of them as possible, but some ingredients don't go well with others. If there are n possible ingredients (numbered 1 to n, we write down the $n \times n$ matrix D giving the *discord* between any pair of ingredients. This *discord* is a real number between 0.0 and 1.0, where 0.0 means "they go together perfectly" and 1.0 means "they don't go together". For example, if $D[2, 3] = 0.1$ and $D[1, 5] = 1.0$, then ingredients 2 and 3 go together pretty well whereas 1 and 5 clash badly.

Notice that D is necessarely symmetric and that the diagonal entries are always 0.0. Any set of ingredients always incurs a penalty which is the sum of all discors values between pairs of ingredients. For instance the set of ingredients $\{1, 2, 3\}$ incurs a penalty of $D[1, 2] + D[1, 3] + D[2, 3]$. We want the penalty to be small.

Exercise

Given n ingredients, and the discord $n \times n$ matrix D and some number p, compute the maximum number of ingredients we can choose with penaly $\leq p$

Show that if Experimental Cuisine is solvable in polinomial time, then is so 3SAT.

Answer

A way to prove that if Experimental Cuisine is solvable in polinomial time, then so is 3sat is by reducing in polynomial time 3sat to Experimental Cuisine.

Assuming a reduction is found, it means that Experimental Cuisine is harder or as hard to solve as 3sat. Once this has been established, there is no doubt that if we assume that Experimental Cuisine is solvable in polynomial time, we have that 3sat must be solvable in polynomial time, because it is easier or as easy to solve as Experimental Cuisine, based on our previous assumptions.

The problem is that 3SAT is a search problem while EXPERIMENTAL CUISINE is an **optimisation** problem (Given a set of restrictions, find the better solution to the problem). This makes it very difficult if not impossible to find a reduction directly from one another. Fortunately, we know that a search problem reduces in polynomial time to its optimisation version and vice versa[1, p. 250], so we can easily re-formulate EXPERIMENTAL CUISINE to be a search problem:

EXPERIMENTAL CUISINE Given n ingredients, a number $k \leq n$, the discord $n \times n$ matrix D and some number p, compute if it is possible to choose k ingredients which the total discord between them is $\leq p$.

From now on, when we refer to the problem, we'll be referring to its search version.

We have solved our first problem. Unfortunately, the reduction still is yet to be found, and there is no clear path that leads us to it despite having changed the very nature of our problem. Despite our best efforts, 3SAT and EXPERIMENTAL CUISINE are still very different problems one to another. 3SAT is about boolean formula satisfactibility and EXPERIMENTAL CUISINE is much more similar to a LONGEST PATH problem, for example. Because of this, we'll have to use another problem to help us with the reduction: INDEPENDENT SET.

INDEPENDENT SET A subset of nodes $S \subset V$ is an *independent set* of graph $G = (V, E)$ if there are no edges between them. Given a graph $G = (V, E)$ and a number $s \leq |V|$, say if there is a subset S size s that is an *independent set* of G.

INDEPENDENT SET is very similar (or at least more similar than 3SAT) to EXPERIMENTAL CUISINE. If we interpret EXPERIMENTAL CUISINE as a graph problem, with D as its adjacency matrix, the simiarities begin to arise. And what is more convenient: 3SAT can be reduced to INDEPENDENT SET, and it has aleady been done[1, p. 262]!

Now it is only left for us to reduce INDEPENDENT SET to EXPERIMENTAL CUISINE.

INDEPENDENT SET $\longrightarrow$ EXPERIMENTAL CUISINE

Building the reduction For a reduction to be effective, we have to create a function F that modifies an instance I of INDEPENDENT SET to a valid one of EXPERIMENTAL CUISINE. Additionally, we have to create a function h that takes in a valid solution S to EXPERIMENTAL CUISINE and transforms it to be a valid solution of INDEPENDENT SET. Finally, if $F(I)$ is not a solution to EXPERIMENTAL CUISINE, then I should not be a solution of INDEPENDENT SET.

Let us begin with the reduction. Given an instance of INDEPENDENT SET, G, where G is a graph, construct the following:

Set D to be an incidence matrix of G, where $a_{ij} = 0$ if the graph's vertices do not share a common edge, and $a_{ij} = 1$ if they are, and assign to the maximum discordance d the value of 0. We should then set k to s.

If we feed this values to an algoritm that solves EXPERIMENTAL CUISINE, the result that it yields is a subset of verticies V such that V is an INDEPENDENT SET of G.

Proving that this reduction is correct Given a graph $G = (V, E)$ and a vertex set $I \subseteq V$ that is the independent set of G with highest cardinality, it is easy to see that, when we transform this configuration using the steps described above, it is a valid solution to EXPERIMENTAL CUISINE, and adding one more vertex to is not. This means it is the maximum subset that satisfies the problem.

References

[1] C.H. Papadimitriou S. Dasgupta and U.V. Vazirani. *Algorithms*. McGraw-Hill Education, 2006.