Algorithms for Data Science

CSOR W4246, Fall 2019

Eleni Drinea

Computer Science Department

Columbia University

Final review session

by Ashwin Jayaraman and Yiran Wang

1. Using reductions to design efficient algorithms

You are asked to assist in the following crisis event.

Due to large scale flooding, there is a set of n injured people distributed across a region that need to be rushed to hospitals. There are k hospitals in the region, and each of the n people needs to be brought to a hospital that is within a half-hour's driving time of their current location (so different people will have different options for hospitals, depending on where they are right now). However you do not want to overload any single hospital; instead, you want every hospital to receive at most $\lceil n/k \rceil$ people.

Give a polynomial-time algorithm for this problem or state its decision version and prove it is $\mathcal{NP}$-complete.

1. Solution

Solution: Reduce to max flow.

First construct the following bipartite graph $G = (X \cup Y, E)$

- X is the set of injured people
- Y is the set of hospitals
- Add an edge (x, y) from person $x \in X$ to hospital $y \in Y$ if x is within half-hour's driving time of hospital y.

Note that there are at most nk edges in this graph.

1. Solution (cont'd)

Next construct a flow network $G' = (V', E', c, s, t)$ as follows:

- Introduce two nodes s, t.
- Set $V' = X \cup Y \cup \{s\} \cup \{t\}$.
- Direct all edges in E from X to Y.
  Set $E' = E \cup \{(s, x) \text{ for all } x \in X\} \cup \{(y, t) \text{ for all } y \in Y\}$.
  Thus $|E'| = |E| + n + k$.
- Set $c_e = 1$ for every edge $e \in E$ that does enter t.
- Set $c_e = \lceil n/k \rceil$ for every edge $e = (y, t)$ with $y \in Y$.

Note that construction of the bipartite graph G and the flow network G' take time polynomial in n, k.

1. Solution (cont'd)

Proof of equivalence of instances: G' has a max flow of value n if and only if every injured person can be rushed to a nearby hospital so that no hospital is overloaded *(exercise).*

Running time: Ford-Fulkerson finds max flow in $O(nmU)$, which becomes $O(n \cdot nk \cdot \lceil n/k \rceil) = O(n^3)$ in G'.

2. Using reductions to prove $\mathcal{NP}$-completeness

A large store has m customers and n products and maintains an $m \times n$ matrix A such that $A_{ij} = 1$ if customer i has purchased product j; otherwise, $A_{ij} = 0$.

Two customers are called *orthogonal* if they did not purchase any products in common. Your task is to help the store determine a maximum subset of orthogonal customers.

Give a polynomial-time algorithm for this problem or state its decision version and prove it is $\mathcal{NP}$-complete.

2. Solution

Decision version: Given an $m \times n$ binary matrix A of customers and products, and a target value k, are there at least k orthogonal customers?

We will show that this problem, call it `M(D)`, is $\mathcal{NP}$-complete.

Efficient certifier for `M(D)`*:* On input (A, k) (the instance), and a subset of the customers (the short certificate), the certifier will check that there are at least k customers in the certificate, and that they are pairwise orthogonal. There are at most m customers in the certificate thus at most $\binom{m}{2}$ dot products to compute; each dot product takes $O(n)$ time, hence the certifier is efficient.

2. Solution (cont'd)

Reduction from `IS(D)`*:*

Given an arbitrary input to IS(D), that is, a pair (G, k), we will construct an input to `M(D)`, that is, a matrix A and a target value k' as follows:

- for every node in G, introduce a customer in A;
- for every edge in G, introduce a product in A.

So we have n customers and m products and every product is purchased by at most two customers.
Finally, set $k' = k$.

Clearly the reduction requires polynomial time.

2. Solution (cont'd)

Equivalence of instances: we will show that G has an independent set of size k if and only if A has k orthogonal customers. Both directions are straightforward.

- $\Rightarrow$ Suppose G has an independent set S of size k. Then the k customers in A corresponding to the nodes in S form a set of pairwise orthogonal customers: for any pair of nodes i, j in S, they do not share an edge, hence customers i and j did not purchase a product in common.
- $\Leftarrow$ Suppose there is a set S of k orthogonal customers in A. The nodes in G corresponding to these customers form an independent set of size k: for any pair of customers i, j in S, they do not share a product, hence there is no edge (i, j) in G.

3. Formulating IPs

Formulate integer programs for the following problems.

- `3SAT`

- `Uncapacitated Facility Location`:

  There is a set F of m facilities and a set D of n clients. For each facility $i \in F$ and each client $j \in D$, there is a cost c_{ij} of assigning client j to facility i. Further, there is a one-time cost f_i associated with opening and operating facility i.

  Find a subset F' of facilities to open that minimizes the total cost of (i) operating the facilities in F' and (ii) assigning every client j to one of the facilities in F'.

3. Solution (SAT)

1. Let ϕ be the input formula consisting of m clauses $C_1, C_2, \ldots, C_m$ in 3CNF over n boolean variables.

For clause C_j, let I_j^+ be the set of variables appearing unnegated in C_j and I_j^- the set of variables appearing negated.

- For every boolean variable, introduce one binary variable. Intuitively, the value of the binary variable corresponds to the truth value of the boolean variable and vice versa.
- For every clause, introduce one linear constraint to ensure that at least one literal in the clause evaluates to 1.

3. Solution (SAT, cont'd)

Any setting of the variables that meets the constraints yields a satisfying truth assignment for ϕ. The IP follows.

$$\begin{array}{rll} \max & 0 & \\ \text{subject to} & \displaystyle\sum_{i \in I_j^+} x_i + \sum_{i \in I_j^-} (1 - x_i) \geq 1, & \text{for } j = 1, \ldots, m \\ & x_i \in \{0, 1\}, & \text{for } i = 1, \ldots, n \end{array}$$

3. Solution (UFL)

2. For every facility $i \in F$ and every client $j \in D$, define the binary variable

$$y_{ij} = \begin{cases} 1, & \text{if client } j \text{ is assigned to facility } i \\ 0, & \text{otherwise} \end{cases}$$

Also, for every facility $i \in F$ define the binary variable

$$x_i = \begin{cases} 1, & \text{if facility } i \text{ is open} \\ 0, & \text{otherwise} \end{cases}$$

3. Solution (UFL cont'd)

Constraints: every client must be assigned to one facility. If a client is assigned to a facility, then that facility must be open.

$$\begin{aligned}
\min_{x_i, y_{ij}} \quad & \sum_{i \in F} f_i x_i + \sum_{i \in F, j \in D} c_{ij} y_{ij} & \\
\text{subject to} \quad & \sum_{i \in F} y_{ij} = 1 & \text{for all } j \in D \\
& y_{ij} \leq x_i & \text{for all } j \in D, \text{ for all } i \in F \\
& y_{ij} \in \{0, 1\} & \text{for all } i \in F, j \in D \\
& x_i \in \{0, 1\} & \text{for all } i \in F
\end{aligned}$$