CS 435 : Linear Optimization — Fall 2006

Lecture 25: ILP formulations for SAT and shortest path problem (contd.)

Lecturer: *Sundar Vishwanathan* — Scribe: *Ajit Burad*
Computer Science & Engineering — Indian Institute of Technology, Bombay

1 SAT (*continuing with previous lecture*)

Input : Conjunction of Clauses
$(x_1 \vee x_2 \vee ...) \wedge (x_3 \vee x_4 \vee ...) \wedge (... \vee ...)$

Variables The literals, x_i's, in the clauses which are being assigned true or false are variables for the ILP.
But for the ILP we do not have True, False in our set so we will have one variable per boolean variable. For instance a variable y_i corresponding to each boolean variable x_i, So

x_i being false will be equivalent to y_i being zero.
x_i being true will be equivalent to y_i being one.

Here $0 \leq y_i \leq 1$, y_i is Integer forces y_i to take values either 1 or 0.

We will represent $\overline{y_i} = 1 - y_i$. So for each clause $(x_i \vee x_j \vee ...)$ the fact that one of the literals has to be true can be expressed as:

$$y_i + y_j + ... \geq 1$$

If we have $\overline{x_i}$, replace it with 1-y_i.
Here cost function is immaterial as for each set of y_i's we can get corresponding x_i's.

Here another approach (which does not work) could have been assigning :

$$y_i = +1 \quad \text{if} \quad x_i = \text{true}$$
$$y_i = \text{-1} \quad \text{if} \quad x_i = \text{false}$$

then -1 $\leq y_i \leq 1$ but $y_i = 0$ is not assigned anywhere.
Hence it is not possible to solve it this way.

2 Shortest Path

Input :
Directed Graph(G) (with each edge having positive integer weight)
Vertices s,t $\in$ V(G)

- Where s and t are start and end vertices respectively for the required path.
- V(G) is vertex set of G

Output : Shortest Path between s and t

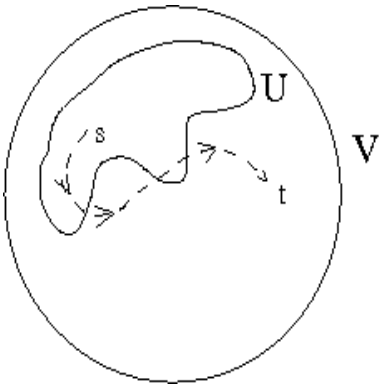

Figure 1:

To construct an ILP we first need to define variables. An assignment to the variables should tell us what the output should be.
For this problem we choose one variable (w_i) per edge.
$x_i = 1 \Rightarrow$ corresponding edge is chosen in shortest path from s to t.
$x_i = 0 \Rightarrow$ edge is not part of shortest path from s to t.

Cost : Let the cost of edge e be c_e
Cost of path $= \sum_e (x_e * c_e)$
Consider subsets of the vertex set which do not contain t but contain s. (See figure 1)
$\forall$ U $\subseteq$ V such that s $\in$ U, t $\notin$ U, at least one edge leaving U must be selected.

The following set of constraints force a path from s to t for every U $\in$ V(G), such that s $\in$U and v $\notin$ U,

$$\sum_{u \in U, v \notin U} X_{uv} \geq 1 \ ;$$

Note that we had an exponential number of costraints. There is another way to write the constraints, so that there are only a polynomial number of them.

$\sum_v x_{sv} = 1$, $\sum_u x_{ut} = 1$ (There are edges one starting from s and one terminating at t in our required path. We assume that there are no edges entering s and leaving t in G).
Also $\forall$ u $\neq$ s,t $\sum_w x_{wu} - \sum_v x_{uv} = 0$; $0 \leq Xuv \leq 1$.
Now we need to write dual for this and we will do the same thing as we did for SAT and we will end up with Dijkstra's.