Lecture 23: An example illustrating the LP duality based MST algorithm

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

In this note we illustrate the MST algorithm derived in the class using LP duality.

1 Relaxed LP for the MST of an undirected graph and its Dual

The *relaxed* LP formulation (i.e. after removing the constraints x_e *integer*, $x_e \leq 1$) for the MST of a graph $G(V, E)$ is the following:

$$\begin{aligned}
\min \quad & \sum_e c_e x_e \\
\text{s.t.} \quad & \sum_{e \text{ crosses } \pi} x_e \geq \#(\pi) - 1 \quad && \forall \pi \\
& x_e \geq 0 && \forall e \in E
\end{aligned}$$

where

- $c_e \geq 0$ is the weight of the edge $e \in E$
- π denotes a partition of the vertex set V and can be represented as the set of disjoint subsets of the vertex set V, i.e. $\pi \equiv \{p_1, p_2, \ldots, p_k\}, p_i \subset V, \cup p_i = V, p_i \cap p_j = \phi$. # is the number of 'parts' in π i.e. $= k$
- e crosses π means one end vertex of e belongs to p_i and other end belongs to $p_j, i \neq j$.

The dual LP is given by

$$\begin{aligned}
\max \quad & \sum_e y_\pi (\#(\pi) - 1) \\
\text{s.t.} \quad & \sum_{e \text{ crosses } \pi} y_\pi \leq c_e \quad && \forall e \in E \\
& y_\pi \geq 0 && \forall \pi
\end{aligned}$$

2 Formulations for an example graph

Let us see the formulation for an example graph in figure 1(i).

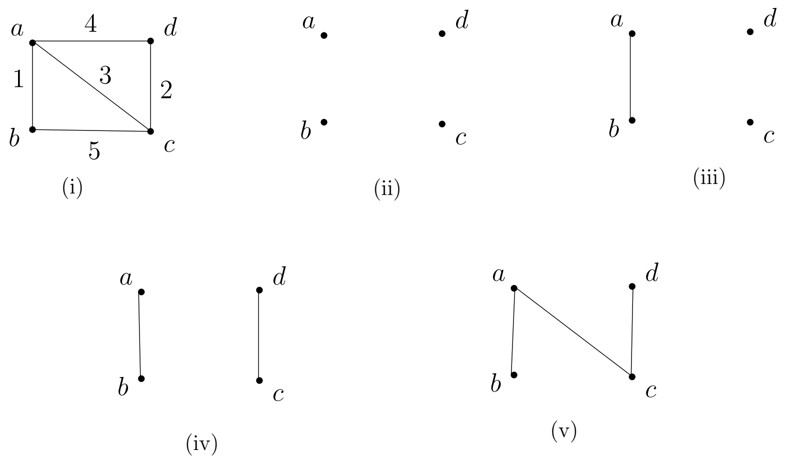

Figure 1: An example graph and intermediate steps of the algorithm

The relaxed primal LP is:

$$
\begin{array}{lrl}
\min & x_{ab}+2x_{cd}+3x_{ac}+4x_{ad}+5x_{bc} & \\
\text{s.t.} & x_{ab}+x_{cd}+x_{ac}+x_{ad}+x_{bc} \geq 4-1 & \text{for } \pi=\{a,b,c,d\} \\
& x_{cd}+x_{ac}+x_{ad}+x_{bc} \geq 3-1 & \text{for } \pi=\{ab,c,d\} \\
& x_{ab}+x_{cd}+x_{ac}+x_{bc} \geq 3-1 & \text{for } \pi=\{ac,b,d\} \\
& x_{ab}+x_{cd}+x_{ad}+x_{bc} \geq 3-1 & \text{for } \pi=\{ad,b,c\} \\
& x_{ab}+x_{cd}+x_{ac}+x_{ad}+x_{bc} \geq 3-1 & \text{for } \pi=\{bc,a,d\} \\
& x_{ab}+x_{cd}+x_{ac}+x_{ad} \geq 3-1 & \text{for } \pi=\{bd,a,c\} \\
& x_{ab}+x_{ac}+x_{ad}+x_{bc} \geq 3-1 & \text{for } \pi=\{cd,a,b\} \\
& x_{ac}+x_{ad}+x_{bc} \geq 2-1 & \text{for } \pi=\{ab,cd\} \\
& x_{ab}+x_{cd}+x_{ac} \geq 2-1 & \text{for } \pi=\{ac,bd\} \\
& x_{ab}+x_{cd}+x_{ad}+x_{bc} \geq 2-1 & \text{for } \pi=\{ad,bc\} \\
& x_{cd}+x_{ac}+x_{bc} \geq 2-1 & \text{for } \pi=\{abc,d\} \\
& x_{cd}+x_{ad} \geq 2-1 & \text{for } \pi=\{abd,c\} \\
& x_{ab}+x_{bc} \geq 2-1 & \text{for } \pi=\{acd,b\} \\
& x_{ab}+x_{ac}+x_{ad} \geq 2-1 & \text{for } \pi=\{bcd,a\} \\
& x_e \geq 0 & \forall e=1..5
\end{array}
$$

And the corresponding dual is

max

$$
\begin{aligned}
3y_{\{a,b,c,d\}} + 2y_{\{ab,c,d\}} + 2y_{\{ac,b,d\}} + 2y_{\{ad,b,c\}} + 2y_{\{bc,a,d\}} + 2y_{\{bd,a,c\}} + 2y_{\{cd,a,b\}} \\
+ y_{\{ab,cd\}} + y_{\{ac,bd\}} + y_{\{ad,bc\}} + y_{\{abc,d\}} + y_{\{abd,c\}} + y_{\{acd,b\}} + y_{\{bcd,a\}}
\end{aligned}
$$

subj. to

$$
\begin{aligned}
e=ab: & \; y_{\{a,b,c,d\}} + y_{\{ac,b,d\}} + y_{\{ad,b,c\}} + y_{\{bc,a,d\}} + y_{\{bd,a,c\}} + y_{\{cd,a,b\}} + y_{\{ac,bd\}} + y_{\{ad,bc\}} + y_{\{acd,b\}} + y_{\{bcd,a\}} \leq 1 && (1) \\
e=cd: & \; y_{\{a,b,c,d\}} + y_{\{ab,c,d\}} + y_{\{ac,b,d\}} + y_{\{ad,b,c\}} + y_{\{bc,a,d\}} + y_{\{bd,a,c\}} + y_{\{ac,bd\}} + y_{\{ad,bc\}} + y_{\{abc,d\}} + y_{\{abd,c\}} \leq 2 && (2) \\
e=ac: & \; y_{\{a,b,c,d\}} + y_{\{ab,c,d\}} + y_{\{ac,b,d\}} + y_{\{bc,a,d\}} + y_{\{bd,a,c\}} + y_{\{cd,a,b\}} + y_{\{ab,cd\}} + y_{\{ac,bd\}} + y_{\{abc,d\}} + y_{\{bcd,a\}} \leq 3 && (3) \\
e=ad: & \; y_{\{a,b,c,d\}} + y_{\{ab,c,d\}} + y_{\{ad,b,c\}} + y_{\{bc,a,d\}} + y_{\{bd,a,c\}} + y_{\{cd,a,b\}} + y_{\{ab,cd\}} + y_{\{ad,bc\}} + y_{\{abd,c\}} + y_{\{bcd,a\}} \leq 4 && (4) \\
e=bc: & \; y_{\{a,b,c,d\}} + y_{\{ab,c,d\}} + y_{\{ac,b,d\}} + y_{\{ad,b,c\}} + y_{\{bc,a,d\}} + y_{\{cd,a,b\}} + y_{\{ab,cd\}} + y_{\{ad,bc\}} + y_{\{abc,d\}} + y_{\{acd,b\}} \leq 5 && (5) \\
& \forall\pi \qquad y_\pi \geq 0 && (6)
\end{aligned}
$$

3 The primal-dual algorithm

The algorithm that we designed is the following:

Algorithm 1 MST algorithm developed using LP duality

Initialize all $y_\pi \leftarrow 0$
Set $i \leftarrow 0, \pi_0 \leftarrow$ the partition in which each vertex is a part by itself
repeat
  Raise y_{π_i} until dual constraint corresponding to some edge e becomes tight
  Set $x_e \leftarrow 1$
  Set $\pi_{i+1} \leftarrow$ the partition got from π_i by merging end points of e
  Set $i \leftarrow i+1$
until edges with $x_e = 1$ forms a spanning tree

Let us apply the technique on the example graph in figure 1(i). Intermediate steps of the algorithm 1 on this instance are given in table 1. Initially all $y_\pi = 0$. This corresponds to figure 1(ii).

Property \ Iteration	1^{st}	2^{nd}	3^{rd}
π considered	$\{a, b, c, d\}$	$\{ab, c, d\}$	$\{ab, cd\}$
[constraint involved, max raise in y_π possible]	[(1),1], [(2),2], [(3),3] [(4),4], [(5),5]	[(2),2-1], [(3),3-1], [(4),4-1],[(5),5-1]	[(3),3-1-1], [(4),4-1-1] [(5),5-1-1]
constraint tightens first ie for which max raise is min	(1)	(2)	(3)
y_π raised	$y_{\{a,b,c,d\}} = 1$	$y_{\{ab,c,d\}} = 2 - 1 = 1$	$y_{\{ab,cd\}} = 3 - 1 - 1 = 1$
x_e set	$x_{ab} = 1$	$x_{cd} = 1$	$x_{ac} = 1$
Vertices of e	a, b	c, d	a, c
New π	$\{ab, c, d\}$	$\{ab, cd\}$	$\{abcd\}$, MST found
Sub-figure no.	(iii)	(iv)	(v)

Table 1: Trace of the primal-dual algorithm on the example graph