Primal-Dual algorithm

Step 0: Formulate the problem as an integer program. Relax it to get linear program LP and its dual DLP.

Step 1: Start with an infeasible solution x_0 to LP and a feasible solution y_0 to DLP.

Step 2: while (x_i is not feasible)
$y_{i+1} \leftarrow$ update y_i s.t. atleast one more dual constraint is tight
set corresponding primal varikble to be 1

Step 3: Prove bound on cost of x_{end}.

Remark: Primal-dual algorithm does not involve solving primal or the dual linear programs.

Example: $2\left(1-\frac{1}{k}\right)$ approximation for steiner tree problem.

Problem: Given an undirected graph $G = (V, E)$, a cost function w on the edges $w: E \to Q_+$ and a set of terminals $T \subseteq V$, goal is to find a subgraph $H = (V_H, E_H)$ connecting the terminals of minimum cost $\left(\sum_{e \in E_H} w_e\right)$

Theorem: There exists a primal-dual algorithm for Steiner tree problem with approximation ratio $2\left(1-\frac{1}{|T|}\right)$.

Remark: There exist an algorithm with improved approximatio ratio for steiner tree. However it is not discussed here.

- S separates T if $S \cap T \neq \phi$ and $S \cap T \neq T$.
- S is the set of edges with exactly one end point in S.
- $\delta(S) \subseteq E$ is the set of edges with exactly one end point in S.

Primal

$$\min \sum_{e \in E} w_e x_e$$

s.t.

$$\sum_{e \in \delta(S)} x_e \geq 1 \quad \forall S \text{ separate } T$$

$$x_e \geq 0$$

Dual

$$\max \sum_{S : S \text{ separates } T} y_S$$

s.t.

$$\sum_{e \in \delta(S)} y_S \leq w_e \qquad \forall e \in E$$

$$y_S \geq 0$$

If $c_e = 1 \; \forall e \in E$:- largest connection of separating sets with disjoint edge boundaries

---

Algorithm :-

$\Psi = \{ \{x\} : x \in T \}$ (set of components)

$X_{\{x\}} = \{ (\{x\}, \phi) : x \in T \}$ (For each component $C \in \Psi$, X_C is the tree on C found in the algorithm)

$F = (T, \phi)$ (Steiner forest initialized to a forest on vertex set T and no edges. At the end F will be a Steiner tree)

$(X_C, C \in \Psi)$ vs F: During the algorithm, we would grow components by adding edges and vertices. However, not all the edges added to these components are part of the steiner tree F we return at the end. We only add edges to F, when two components merge.

$y_S = 0 \quad \forall S \subseteq V : S$ separates T (Initial dual feasible sol^n)

Time $t = 0$

$M_{\{x\}} = \{\{x\}\}$ (At any time t and $C \in \Psi$ at time t, M_C contains all subsets S of C s.t. y_S may be non-zero)

While $(|\Psi| > 1)$

For all $C \in \Psi$, increase y_C at the same rate as t till $\sum_{\substack{S : S \text{ separates } T \\ e \in \delta(S)}} y_S \leq c_e$ is tight for some $(u,v) \in E$.

If $u \in C_i$, $v \in C_j$ for some $C_i, C_j \in \Psi$

Add $C_i \cup C_j$ to Ψ and delete C_i, C_j from Ψ.

$X_{C_i \cup C_j} = X_{C_i} \cup X_{C_j} + e$

Let p be a path connecting $F \cap C_i$ and $F \cap C_j$ in $X_{C_i \cup C_j}$

$F = F \cup p$ (add edges and vertices of p to F)

$M_{C_i \cup C_j} = M_{C_i} \cup M_{C_j} \cup \{C_i \cup C_j\}$

If $u \in C_i$ for some $C_i \in \Psi$, $v \notin C_j$ for any $C_j \in \Psi$
add $C_i + v$ to Ψ, delete C_i from Ψ.
$X_{C_i+v} = X_{C_i} + e$ (add vertex v and edge e to C_i)
$M_{C_i+v} = M_{C_i} \cup \{C_i + v\}$

Return F

Analysis

Lemma 1:- At any time t, for any $C \in \Psi$, X_C is a tree.

Lemma 2:- At the end of algorithm, F is a steiner tree.

Lemma 3:- At any time t, $\{y_S,\ S$ separates $T\}$ is a feasible dual solution.

All these lemmas can be easily proved by induction

Lemma 4:- At time $t \geq 0$, for $C \in \Psi$ at time t let F_C be the edges of **F** (at time t) with both end point in C.

$$Z(C) = \sum_{S \in M_C} y_S, \quad Cost(C) = \sum_{e \in F_C} w_e.$$

Then, $$Cost(C) \leq 2\left(Z(C) - t\right).$$

Proof:- At time $t=0$, $F=(T,\phi)$, $\Psi=\{\{x\}: x\in T\}$
$y_S = 0, \forall S: S$ separates T

Hence, $Cost(C)=0, Z(C)=0 \quad \forall C \in \Psi$.
$\Rightarrow Cost(C) \leq 2(Z(C)-t)$ at $t=0$.

*For ease of exposition, we divide the events into three cases:

1. t increases by Δt and no change in the set of components
2. A vertex v is added to some component C_i at time t.
3. Two components C_i, C_j merge at time t.

Case 1:- Set of components Ψ does not change.
By induction, $Cost(C) \leq 2(Z(C)-t)$ at time t.
y_C increases by Δt for each $C\in\Psi$

$Z(C)=\sum_{S\in M_C} y_S$ increases by Δt since $C \in M_C$.
F does not change. Hence, $Cost(C)$ does not change

Hence, $Cost(C) \leq 2(Z(C)-t)$ at time $t+\Delta t$.

Case 2:- A vertex v is added to C_i (t does not change)
$C_i + v$ is added to Ψ and C_i is deleted.
By induction, $Cost(C_i) \leq 2(Z(C_i)-t)$.

$Cost(C_i + v) = Cost(C_i)$ since, no edges are added to F and $F_{C_i+v} = F_{C_i}$

$$Z(C_i+v) = \sum_{S \in M_{C_i+v}} y_S = \sum_{S \in M_{C_i}} y_S + y_{C_i+v}$$

when v is added to C_i, $y_{C_i+v} = 0$. Hence,

$$Z_{C_i+v} = Z_{C_i}, \quad Cost(C_i+v) = Cost(C_i)$$

$$\Rightarrow Cost(C_i+v) \le 2(Z(C_i+v) - t).$$

<u>Case 3</u> :- Two components $C_i, C_j \in \psi$ merge (t does not change)

$C_i \cup C_j$ is added to ψ, C_i, C_j are deleted from ψ.

By induction $Cost^{old}(C_i) \le 2(Z(C_i) - t)$

$Cost^{old}(C_j) \le 2(Z(C_j) - t)$

*old denotes before merging

$$Cost^{new}(C_i \cup C_j) = \sum_{e \in F^{new}_{C_i+C_j}} w_e = \sum_{e \in F^{old}_{C_i}} w_e + \sum_{e \in F^{old}_{C_j}} w_e + \sum_{e \in P} w_e.$$

<u>Claim</u> :- $\sum_{e \in P} w_e \le 2t$

Hence, $Cost^{new}(C_i \cup C_j) \le Cost^{old}(C_i) + Cost^{old}(C_j) + 2t$

$$\le 2(Z(C_i) + Z(C_j) - t).$$

$$Z(C_i \cup C_j) = \sum_{S \in M_{C_i \cup C_j}} y_S = \sum_{S \in M_{C_i}} y_S + \sum_{S \in M_{C_j}} y_S + y_{C_i \cup C_j}$$

Since, $C_i \cup C_j$ is just added to $M_{C_i \cup C_j}$, $y_{C_i \cup C_j} = 0$

Hence,

$$Z_{C_i \cup C_j} = Z(C_i) + Z(C_j)$$

$$\Rightarrow \text{Cost}^{new}(C_i \cup C_j) \leq 2\left(Z(C_i \cup C_j) - t\right).$$

end of proof of Lemma 4.

<u>Theorem</u> :- $\sum_{e \in E(F)} w_e \leq 2\left(1 - \frac{1}{|T|}\right)$ OPT-Steiner-tree cost

<u>Proof</u> :- At the end of the algorithm ($t = t_{end}$)

let the component in ψ be C^*.

Then, at $t = t_{end}$,

$$\text{Cost}(C^*) = \sum_{e \in F_{C^*}} w_e = \sum_{e \in E(F)} w_e.$$

$$Z(C^*) = \sum_{S \in M_{C^*}} y_S = \sum_{S : S \text{ separate } T} y_S.$$

Hence, by lemma 4,

$$\sum_{e \in E(F)} w_e \leq 2\left(\sum_{S: \text{Separates } T} y_S - t_{end}\right) \quad \text{— (1)}$$

At any time t, # of components in $\mathcal{Y}$ is at most $|T|$. Hence, when t increases by Δt, $\sum_{S: S \text{ separates } T} y_S$ increases by atmost $|T| \Delta t$.

$$\Rightarrow \sum_{S: S \text{ separates } T} y_S \leq |T| \cdot t_{end}.$$

or $$t_{end} \geq \frac{\sum_{S: S \text{ separates } T} y_S}{|T|}$$

Substituting in (1)

$$\Rightarrow \sum_{e \in E(F)} w_e \leq 2\left(1 - \frac{1}{|T|}\right) \sum_{S: S \text{ separate } T} y_S.$$

Since $\{y_S, S \text{ separates } T\}$ is a feasible dual solution, $\sum_{S: S \text{ separates } T} y_S \leq$ Optimal-dual-value

By strong duality

Optimal-dual-value = Optimal-primal-value

Since, primal is a relaxation of the steiner tree problem,

Optimal-primal-value $\leq$ Optimal-steiner-tree-cost

Combining all four inequalities, we get,

$$\sum_{e \in E(F)} w_e \leq 2\left(1-\frac{1}{|T|}\right) \text{Optimal-steiner-tree-cost}$$

Hence, we get a $2\left(1-\frac{1}{|T|}\right)$ approximation algorithm for steiner tree problem.