CS 6200: Algorithmics II	**Fall 2020**
Instructor: Avah Banerjee, S&T.	
Lecture: 18	**Date: 10/7**

1 Linear Programming

This lecture looks at the second example of linear programming, where we formulate Maximum flow as a linear program.

1.1 Duality of Linear Programming

Duality is one of the central concepts when it comes to convex programming. This is because, whenever there is duality, there is a possibility of a polynomial time solution. Whenever the dual structure is not possible, then the possibility of a polynomial time is very small.

The objective here is to show that a given linear program is a dual linear program. The linear program objective function and constraints are defined as follows:
maximize: $c^T x$
subject to: $Ax \leq b$
$x \geq 0$
The dual linear program is defined as follows:
maximize: $b^T y$
subject to: $A^T y \geq c$
$y \geq 0$

Here the linear program is a maximization problem and so the dual linear program will be a minimization problem and vice versa if otherwise. For a maximization problem, the dual linear program will give the upper bound and in such a case the linear program is known as the primal and the other the dual of this primal. So, the dual here gives the upper bound for the primal.

Taking an example here to clearly understand the duality concept. In this example consider the following as the linear program objective function and the constraints:
maximize: $2x_1 - 3x_2 + 3x_3$
subject to: $x_1 + x_2 - x_3 \leq 7$
$-x_1 - x_2 + x_3 \leq 7$
$x_1 - 2x_2 + 2x_3 \leq 4$
$x_1, x_2, x_3 \geq 0$
The dual linear program objective function and constraints are as follows:
maximize: $7y_1 - 7y_2 + 4y_3$
subject to: $y_1 - y_2 + x_3 \geq 2$
$y_1 - y_2 - 2y_3 \geq -3$
$-y_1 + y_2 + 2y_3 \geq 3$
$y_1, y_2, y_3 \geq 0$
Here the dual linear program is arrived from the dual linear program described above.

Now if the equations are cross multiplied, we arrive at,

$(y_1 - y_2 + y_3)x_1 - (y_1 - y_2 - 2y_3)x_2 + (-y_1 + y_2 + 2y_3)x_3 \leq 7y_1 - 7y_2 + 4y_3$

Here, if we give some values to $y_1, y_2 and y_3$ and we substitute the values back in the above equation, then we arrive at the equation as below,

$\alpha_1 x_1 + \alpha_2 x_3 + \alpha_3 x_3 \leq \beta$

where, $\alpha_1 \geq 2, \alpha_2 \geq -3 and \alpha_3 \geq 3$ and even if the values of $\alpha_1, \alpha_2 and \alpha_3$ are greater than the given values it will still be less than or equal to β.

1.2 Example: Maximum Flow

We take the problem of finding the maximum flow between source s and sink t and solve it using linear programming.

The objective function here is to maximize the flow from s to t. The flow value from s is the difference of the amount of flow going out of s to the sum of flow coming into s.

$$\sum_{v \in V} f_{sv} - \sum_{v \in V} f_{vs} \tag{1}$$

The constraints that this is maximized are as follows:

- Capacity constraint

  The capacity constraint says that the flow between two vertices should always be less than or equal to the capacity for all edges in the graph

$$f_{uv} \leq c_{uv}, \forall (u, v) \in E \tag{2}$$

- Flow conservation

  Flow conservation says that for every vertex which is neither the source s or the sink t, the amount of flow coming into the vertex should be equal to the amount of flow going ot of the vertex.

$$\sum_{v \in V} f_{vu} = \sum_{v \in V} f_{uv}, \forall u \in V - \{s, t\} \tag{3}$$

- Flow constraint

  The flow must never be negative.

$$f_{vu} \geq 0, \forall (u, v) \geq 0 \tag{4}$$

These set of constraints apply to all the edges in the graph. For the first and the last constraint, there are atmost $2m$ constraints and n-2 for the second one. So totally the size of the program is of the order O(m+n).

1.2.1 Duality of Linear Program of Maxflow

The objective here is to convert the maximum flow problem into a circulation problem so that the primal and the dual objective functions and constraints can be defined. In order for this, we need to add an edge from sink t to source s which has infinite capacity. In the case of circulation here, it is the same as that if the flow but the net flow going in and going out of each vertex should be 0 and also the flow conservation should be satisfied for s and t, and this makes the formulation of the problem much simpler.

The problem here is basically to find the maximum flow from t to s and this here is equal to the flow going out from s to t in the original graph. So, here we maximize the flow from t to s and define the primal as:

maximize: f_{ts}
subject to: $f_{uv} \leq c_{uv}, \forall (u,v) \in E$
$\sum_v f_{vu} - \sum_v f_{uv} \leq 0$
$f_{uv} \geq 0, \forall (u,v) \in E$

The second constraint here is similar to the one in the original, because every cycle present in the the new graph has flows and the flows inside every cycle is 0 and this can be shown to this inequality constraint.

Here, the co-efficient of f_{ts} is 1, which we can derive from looking at the original primal-dual definition. So this means that the vector c is 1. Now, for each of the capacity constrains, a dual variable is defined. One such variable we define with respect to the first constraint on the primal definition is d_{uv} that interprets as *distance* where (u,v) is an edge. Another variable is created corresponding the the second constraint in the primal definition is P_u that interprets as *potential*, where u is any vertex.

Now, the dual part of the problem is defined as follows:

maximize: $\sum_{uv \in E} c_{uv} d_{uv}$
subject to: $d_{uv} + P_v - P_u \geq 0, \forall (u,v) \in E$
$P_s - P_t \geq 1$
$d_{uv} \geq 0, \forall (u,v) \in E \;\; \& \;\; P_u \geq 0, \forall u \in V$

If we compare this dual definition with the original general definition, we can see that the b vector here is a column vector consisting of all the capacities c_{uv} and all 0s. Then y will be a column vector as well, consisting of all d_{uv} and all P_u values. Now, if b^T is taken we get the objective function which is $\sum_{uv \in E} c_{uv} d_{uv}$.

Here, we are talking about the sum of all co-efficients of the variables and making the corresponding co-efficient function in the objective function greater than the variable. f_{ts} is the only variable here for which the co-efficient is non-zero. Now, there is not constraint for f_{ts} in the first constraints since it is not part of the original set of the edges and the capacity is infinite and hence it doesn't have any constraints.

In the second constraint, the variable f_{ts} appears on the left side of the difference operator as well as on the right side and this is the only two times it appears and so we get, $P_s - P_t \geq 1$. From this for each d_{uv}, we get $P_v - P_u$ and hence we get the second constraint in the dual part

References and Further Reading

[1] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2009. Introduction to Algorithms, Third Edition (3rd. ed.). The MIT Press.