Flows and Matchings

March 17, 2023

1 Introduction

This report will contain 4 sections. This includes Flows(Introduction to Network Flows), Max-Flow Min-Cut(The Theorem that can be used to find the Max-Flow), Matchings(Introduction to Matchings), and Hall's Marriage Theorem(The Theorem that defines conditions to find a complete matching). The Flows Section will define some basic principles. The Max-Flow Min-Cut will state the theorem as well as provide proof for the theorem. The Matchings section will define some basic principles. The Halls Marriage Theorem will provide conditions as to when a complete matching can be found. Finally the concept of Flows and Matchings will be connected at the end. This concept of the combination of Flows and Matchings has many applications in the real world. An example of this would be the Assignment Problem. The Assignment Problem is where a set of tasks is assigned to a set of agents such that each task is assigned to exactly one agent. The goal is to find a matching which can be solved using max-flow min-cut algorithm.

2 Flows

Let V, vertices, be a finite set that contains two special vertices s (the source) and t (the sink). Every ordered pair u,v has a capacity, c_{uv} which is ≥ 0. The notation (V,C) is called a capacitated network.

Given a capacitated network (V,C), E is the set of directed edges. G = (V, E) is the directed graph that represents the available routes.

A flow F over the network (V,C) is a set of non-negative integers. These values have to satisfy 2 conditions:
(1) For any $u, v \in$ V, $f_{uv} \leq c_{uv}$
(2) At any vertex v, the sum of the flow values going in must be the same as the sum of the flow values going out(except vertices s and t). This can also be thought of as the no leaks condition.

The concept of flows can be thought of as transporting some resource from s to t under the assumption that unlimited resource is available; however, the network is constrained by the capacities of the different directed edges. The easiest analogy might be considering the graph as a network of pipes. Furthermore, c(e) = 0 if $e \in in(s)$ and c(e) = 0 if $e \in out(t)$. This means that while directed edges may go into the vertex s or a directed edge may leave the vertex t, their capacity must be 0.

Assumptions:

(1) There are no loops in these graphs (for all u $\in$ V, $c_{uu} = 0$)
(2) A directed path exists from s to t$(v_0, v_1, ..., v_{k-1}, v_k \in V$ where $v_0 = s$ and $v_k = t)$

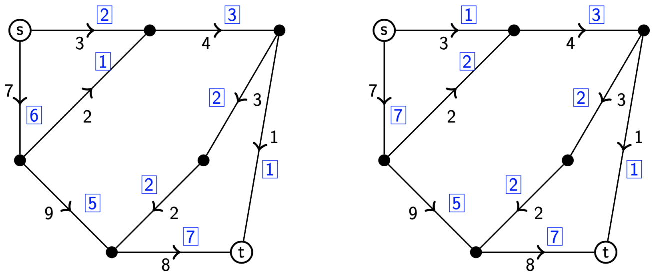

Figure 1: Two graphs that obey all conditions and assumptions

One of the most important questions that arises when looking at flows is: What is the maximum total value of flow that a given network can support? The Max-Flow Min-cut theorem answers that question exactly.

Before the Max-Flow Min-Cut theorem and its proof is stated, it would chronologically make sense to express the second condition in the definition of flows differently:
$\sum_{u \in V} f_{uv}$ - $\sum_{u \in V} f_{vu} = 0$

Proof:
Given a set of vertices V, $\sum_{u \in V} f_{uv} = f_{u_1 v} + f_{u_2 v} + ... + f_{u_k v}$ where v is fixed. $\sum_{u \in V} f_{uv}$ means all the flows which enters into v. This is shown in Figure 2.
Using the same logic, $\sum_{u \in V} f_{vu}$ means all the flows leaving v. This is shown in Figure 3.
Hence, according to the second condition, if $v \notin \{s, t\}$ then $\sum_{u \in V} f_{uv} = \sum_{u \in V} f_{vu}$. This can then be written as $\sum_{u \in V} f_{uv}$ - $\sum_{u \in V} f_{vu} = 0$

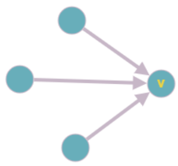

Figure 2: Flow entering v

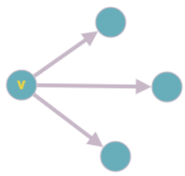

Figure 3: Flow leaving v

$\Phi_{in}(F)$ can be defined as $\Phi_{in}(F) = \sum_{v \in V} f_{sv}$ - $\sum_{u \in V} f_{us}$ and $\Phi_{out}(F) = \sum_{u \in V} f_{ut}$ - $\sum_{v \in V} f_{tv}$
Furthermore, the sum of the flow values leaving for each vertex except s and t can be looked at. This is written as $\sum_{u,v \in V}(f_{uv})$. The sum of all the flow values entering each vertex except s and t can be written as $\sum_{u,v \in V}(f_{vu})$
This is a double sum. $\sum_{u,v \in V}(f_{uv}) = \sum_{u,v \in V}(f_{vu})$ because of the definition of flows. Incorporating s and t in this concept, all the flows that enter vertices can be written as $\sum_{u \in V}(f_{uv}) + \sum_{u \in V}(f_{ut}) + \sum_{u \in V}(f_{us})$. All the flows leaving vertices can be written as $\sum_{u \in V}(f_{vu}) + \sum_{u \in V}(f_{tu}) +$

$\sum_{u \in V}(f_{su})$. These 2 equations can be rearranged with the concept of $\sum_{u \in V} f_{uv} = \sum_{u \in V} f_{vu}$ to get to
$\sum_{v \in V} f_{sv}$ - $\sum_{u \in V} f_{us} = \sum_{u \in V} f_{ut}$ - $\sum_{v \in V} f_{tv}$
Which is the same as $\Phi_{in}(F) = \Phi_{out}(F)$
Definition. $\Phi_{in}(F) = \Phi_{out}(F) = \Phi_{tot}(F)$

3 Max-Flow Min-Cut

The purpose of this theorem is to calculate the maximum total value of flow that a given network can support.
First defining a cut.
Definition:
Let $S \subset V$ with $s \in S$ and $t \notin S$ where the cut associated with S is

$$K(S) := \{uv \in E : u \in S, v \in V \backslash S\}$$

In other words, a cut is a partition of a graph G into 2 disjoint subsets, S and T, such that the source, s, is in the subset S and the sink, t, is in the subset T.
The capacity of K(S) is:

$$k(S) := \sum_{e \in V} c_e = \sum_{u \in V, v \in V \backslash S} c_{uv}$$

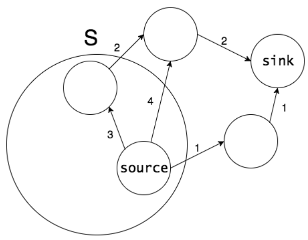

Figure 4: Cut Visual

As shown in Figure 4, there are 2 vertices in the set S where one of them is the source s. This is disjoint to set T which contains the rest of the vertices. The capacity of this cut would be the sum of the capacities of the edges in the cut-set which would be 2 + 4 + 1 = 7. This; however, is not the min cut.

Theorem: Given a capacited network (V,C),

$$\max_F \Phi_{tot}(F) = \min_S k(S)$$

To put it simply, for any network graph and a selected source and sink node, the max-flow from source to sink = the min-cut necessary to separate source from sink.

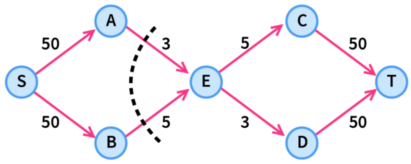

Figure 5: Example if Max-Flow Min-Cut Theorem

In Figure 5, the capacity of the cut shown is 8 which is the min-cut for the graph. This shows that even though the source can transport 100 unit of a resource(50 to A and 50 to B), the max flow that can reach the sink is 8 due to the cut displayed.

Proof: The proof for the 'Max-Flow Min-Cut' theorem can be done using 2 steps:
(1): Proving $\max_F \Phi_{tot}(F) \leq \min_S k(S)$
(2): Proving $\max_F \Phi_{tot}(F) \geq \min_S k(S)$
If these statements are proven then the theorem must be concluded as true.

1. To prove the upper bound on the maximum possible total flow for the minimum cut, the idea of subsets must be introduced.
   This requires further notation.
   For $W \subseteq V$,

   $$\phi_{in}(W) := \sum_{u \in V \setminus W} \sum_{v \in W} f_{uv},$$

   Similarly,

   $$\phi_{out}(W) := \sum_{u \in W} \sum_{v \in V \setminus W} f_{uv}$$

   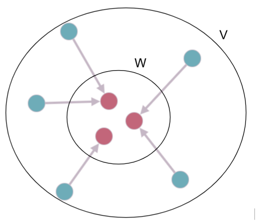

   Figure 6: Flow going from $V \setminus W$ to W showing$\phi_{in}(W)$

   In other words, $\phi_{in}(W)$ is the sum of all flows from u to v where u is in the set V but not in W($V \setminus W$) and v is in the set W. This is also displayed in Figure 6 above.
   $\phi_{out}(W)$ is the sum of all the flows that are leaving W and going into $V \setminus W$.
   Notice that while there may be flows from one vertex in W to another in W, it is not of importance as by definiton, the new notation is only concerned with the flows that go in and out of a subset.
   Before s and t is introduced into this concept, this concept can be looked at with another vertex y where $y \in V \setminus W$. This is shown in Figure 7 below.

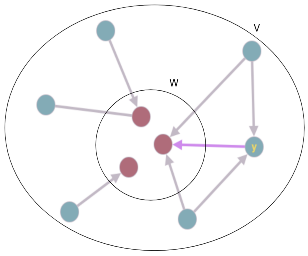

Figure 7: Introducing another Vertex y

Using the definition, ϕ_{in} is the sum of the flows that enter the subset from outside the subset. So if $\phi_{in}(W \cup \{y\})$ would have to take into account the flows that go from y into the subset W. As shown in Figure 7, an example of this would be the pink edge going from the vertex y into the subset W.

$$\phi_{in}(W \cup \{y\}) = \phi_{in}(W) + \sum_{u \in V \setminus W} f_{uy} - \sum_{u \in W} f_{yu}$$

Where $\sum_{u \in V \setminus W} f_{uy}$ is the sum of the flows that go into y from vertices $\in V \setminus W$ and $\sum_{u \in W} f_{yu}$ is the sum of the vertices that go into the subset W from y.
Similarly,

$$\phi_{out}(W \cup \{y\}) = \phi_{out}(W) + \sum_{u \in V \setminus W} f_{yu} - \sum_{u \in W} f_{uy}$$

After defining $\phi_{in}(W \cup \{y\})$ and $\phi_{out}(W \cup \{y\})$, $\Delta(W \cup \{y\})$ is what needs to be defined next. This is because after a cut has been made, the max flow can only be determined if the ΔS is found where S is the subset that contains the source after a cut is made.
Let $\Delta(W) = \phi_{out}(W) - \phi_{in}(W)$
This means that $\Delta(W \cup \{y\}) = \phi_{out}(W \cup \{y\}) - \phi_{in}(W \cup \{y\})$
After expanding and simplifying, this equation can be written as:
$\Delta(W \cup \{y\}) = \phi_{out}(W) - \phi_{in}(W) + \sum_{v \in V} f_{yv} - \sum_{u \in V} f_{uy}$
$= \Delta(W) + \sum_{v \in V} f_{yv} - \sum_{u \in V} f_{uy}$
This can be further simplified as $\Delta(W) = 0$ as the definition of flows says that the sum of all flow values arriving at a vertex is to the sum of all flow values leaving that vertex. This can be applied over all the vertices in W. Using this logic, it can also be shown that $\sum_{v \in V} f_{yv} = \sum_{u \in V} f_{uy}$ and so $Delta(W \cup \{y\}) = 0$
Finally, a representation of $\Delta(W)$ can be shown with regards to if s and t is in the subset W:

$$\Delta(W) = \begin{cases} 0 & \text{if } s \notin W, t \notin W; \\ \Phi_{tot}(F) & \text{if } s \in W, t \notin W; \\ -\Phi_{tot}(F) & \text{if } s \notin W, t \in W; \\ 0 & \text{if } s \in W, t \in W; \end{cases}$$

The subset W, if $s \in W$ and $t \notin W$ can be written as:
$W = W' \cup \{s\}$
This is in the same format as above so the same rules and equations that have have proved can be used.
$\Delta(W) = \Delta(W' \cup \{s\}) = \Delta(W') + \sum_{v \in V} f_{sv} - \sum_{u \in V} f_{us}$

$\Delta(W') = 0$ and $\sum_{u \in V} f_{us} = 0$ because of the initial definition of the vertex s that the capacity of edges that go into s must have a capacity of 0.
Since $\sum_{v \in V} f_{sv} = \Phi_{in}(F) = \Phi_{tot}(F)$
So $\Delta(W) = \Phi_{tot}(F)$ when $s \in W, t \notin W$
Likewise:
$\Delta(W) = -\Phi_{tot}(F)$ when $s \notin W, t \in W$
Likewise:
$\Delta(W) = \Phi_{tot}(F) - \Phi_{tot}(F) = 0$ when $s \in W, t \in W$
Finally, for this proof a cut can be defined by looking at $S \subset V$ where $s \in S, t \notin S$.
$\Delta(S) = \Phi_{out}(S) - \Phi_{in}(S) = \Phi_{tot}(F) = \sum_{u \in S, v \in V \setminus S}(f_{uv}) - \sum_{v \in S, u \in V \setminus S}(f_{uv})$
Capacity of set $k(S)$ is defined according to capacity in S.
$\Phi_{tot}(F) =$ Total flow of network.
By definition $f_{uv} \leq c_{uv}$
So mathematically, $\Phi_{tot}(F) \leq min_S k(S)$

2. A capacited network G, (V,C), may have a maximal flow or it may not. Starting from the source to the sink, an undirected path, p is a sequence of vertices s = $v_0, v_1, \ldots$ where for each r at least one of $v_r v_{r+1}$ or $v_{r+1} v_r$ is an edge.
Definition: An undirected walk p = $(s = v_0, ..., v_k = t)$ from the source to the sink is called usable if for every successive pair of vertices $v_r v_{r+1}$ along the walk, at least one of the following holds
(i) $c_{v_r, v_{r+1}} > 0$ and $c_{v_r, v_{r+1}} > f_{v_r, v_{r+1}}$
(ii) $c_{v_{r+1}, v_r} > 0$ and $f_{v_{r+1}, v_r} > 0$
In other words, for an undirected walk to be usable at least one of the following holds
(i) Forward-oriented edge has spare capacity
(ii) Backward-oriented edge has a non-zero flow
For a vertex v_r in a walk p, let

$$a(v_r) = \begin{cases} c_{v_r, v_{r+1}} - f_{v_r, v_{r+1}} & \text{if } c_{v_{r+1}, v_r} = 0, \\ c_{v_r, v_{r+1}} - f_{v_r, v_{r+1}} + f_{v_{r+1}, v_r} & \text{if } c_{v_{r+1}, v_r} > 0. \end{cases}$$

$a(v_r)$ can be assumed as the excess capacity of the forward-oriented edge, the flow of the backward-oriented edge, or the sum of both if both directed edges are presented. Availability of a usable walk is then is defined as: $A(p) := min_r a(v_r)$
Which means that the availibility is the smallest value from the usable path p that is found. Furthermore, A(p)$\geq$ 1

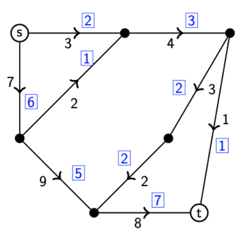

Figure 8: Graph that has usable paths

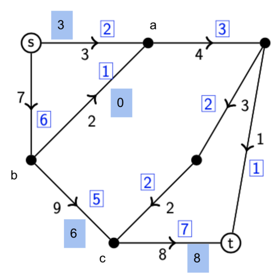

Figure 9: Maximal Flow

Looking at the example of Figure 8, a usable path could be identified would is shown in Figure 9. This is the sequence {sa,ab,bc,ct}. Looking at this sequence $a(v_r)$ can be found for each edge:
$a(s) = 1$, $a(a) = 2$, $a(b) = 4$, $a(t) = 1$
A(p) would be the minimum from these which is 1. You can then increase the current total flow by simply adding A(p) with it. The new flow values are shown in Figure 9 where $\Phi_{tot}(F) = 9$.
Note that after A(p) has been calculated, the path p must be modified to p' using the following rules:
(i) Forward edge - Add flow by A(p)
(ii) Backward edge - Reduce flow by A(p)
Using the example above(Figure 8 and 9), it can be concluded that $\Phi_{tot}(F') = \Phi_{tot}(F) + A(p)$
The introduction of a cut comes in when looking at unusable paths. If a path is unusable then there is a separation between the source and the sink. This is by definition a cut. Furthermore, $\Phi_{tot}(F) = K(S)$ because the amount of the flow coming out of the source vertex is not reaching the sink due to the cut-set. Hence, all the flows are accumulated in the edges of this cut-set and so $\sum_{u \in S, V \in V \setminus S}(f_{uv}) = \Phi_{tot}(F)$

4 Matchings

A matching in a graph G is a subset W of the edges of G with the property that no vertex of G has more than one edge in W incident to it.

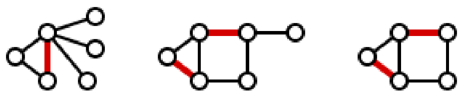

Figure 10: Matching examples

The most common examples of matchings occur in bipartite graphs. Which is displayed in Figure 11. A bipartite graph is a graph whose vertices can be divided into two disjoint and independent sets U and V, that is every edge connects a vertex in U to one in V.

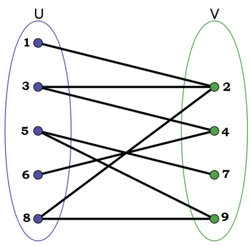

Figure 11: Bipartite Graph

Suppose G is a bipartite graph, with vertex set $V = A \cup B$. Then a complete matching of A is a subset W of the edges such that every $a \in A$ occurs as an endpoint of exactly one edge in W, and no two edges in W are incident to the same vertex in B.

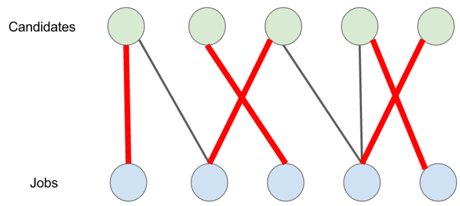

Figure 12: Complete Matching

Clearly a complete matching can only exist if $|A| \leq |B|$
In the example above(figure 12), $|A| = |B|$. So in this example the graph shows a complete set of A and of B.
This leaves us with a question:
Can we describe precisely the conditions under which a complete matching can be found?
This question leads us to a theorem.

5 Hall's Marriage Theorem

Theorem Definition:
Suppose G is a directed bipartite graph with vertex subsets A and B and all edges flow from A to B.
For each $U \subseteq A$, define $i(U) \subseteq B$ by
$i(U) = \{b \in B$: there is an edge in G running to b from some $u \in U.\}$
$i(U)$ is called the out-neighbourhood of U.
A complete matching of A exists if:
(1) $|A| \leq |B|$
(2) $|U| \leq |i(U)|$ for all subsets U of A

This concept can be explained using examples of where this conditions are met in a bipartite graph and when these conditions are not met.

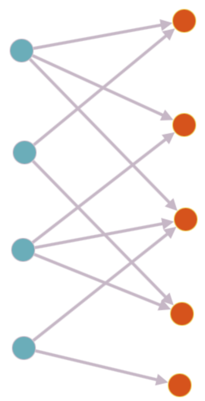

Figure 13: Bipartite Graph

Figure 14: Bipartite Graph with a Complete Matching highlighted

In figure 13, assume the left set as A and the right set as B, then no matter what is chosen as the subset U choose as U from A, $|U| \leq |i(U)|$. This means that Hall's conditions are met and according to Hall's Marriage Theorem, a complete set of A exists.
In figure 14, an example of a complete set of A is shown.

To prove that Hall's Marriage Theorem works, another example can be looked at where Hall's conditions are not met and so a solution for a complete set of A does not exist.

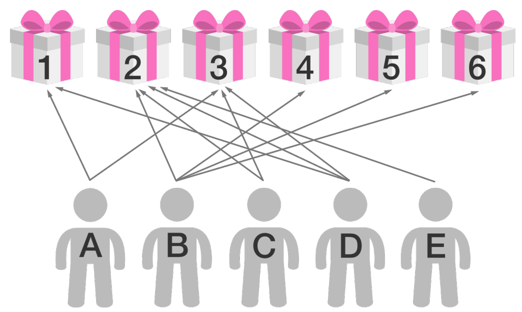

Figure 15: Example of Bipartite Graph that does not follow Hall's Conditions

If you look at the example in Figure 15, the scenario is 'Suppose I have 6 gifts (labeled 1, 2, 3, 4, 5, 6) to give at Christmas, to 5 friends (Alice, Bob, Charles, Dot, Edward). Can I distribute one gift to each person so that everyone gets something they want?'
The bipartite graph does not follow Hall's conditions which can be shown as the gift preference amongst A, C, D, E is 1, 2 and 3. This means it would be impossible to give each person a present that they prefer. Through this it can be concluded that a complete matching cannot be found.
It can easily be shown that $|A|$ is the maximum flow if and only if there is a complete matching of A

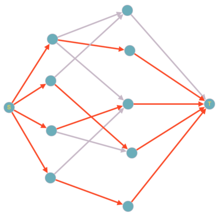

Figure 16: Example of Bipartite Graph that does not follow Hall's Conditions

In Figure 16, it is evident that maximum flow is at most $|A|$ and the maximum flow will also only be equal to $|A|$ if and only if there is a complete matching of A.