MONASH University

MONASH
INFORMATION
TECHNOLOGY

FIT2004 Algorithms and Data Structures

Ian Wern Han Lim
lim.wern.han@monash.edu

Referencing materials by
Rafael Dowsley, Nathan Companez, Aamir Cheema, Arun Konagurthu and Lloyd Allison

GROUP OF EIGHT AUSTRALIA

Faculty of Information Technology, Monash University

MONASH University

Ready?

Agenda

- Circulation with Demands
- Circulation with Demands and Lower Bound

- Applications Example
  - Survey Design
  - Airline Scheduling

Let us begin…

Network Flow

Recall…

- You have learnt Graph
- You have learnt Network Flow

Network Flow
Recall…

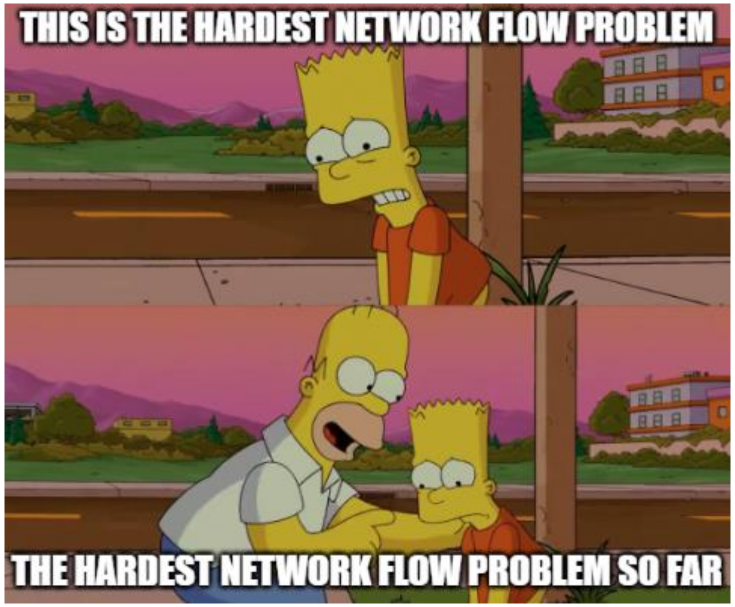
- You have learnt Graph
  - How many problems can be modelled as a Graph, then be solved
- You have learnt Network Flow

Network Flow
Recall…

- You have learnt Graph
  - How many problems can be modelled as a Graph, then be solved.
- You have learnt Network Flow
  - Likewise, we have explore the simple Bipartite Matching problem.

Network Flow
Recall…

- You have learnt Graph
  - How many problems can be modelled as a Graph, then be solved.
- You have learnt Network Flow
  - Likewise, we have explore the simple Bipartite Matching problem.
  - … now let us push 1 step further!


THIS IS THE HARDEST NETWORK FLOW PROBLEM
THE HARDEST NETWORK FLOW PROBLEM SO FAR

Questions?

Circulation with Demands

A Feasibility Problem…

Circulation with Demands

A Feasibility Problem…

- Recall the 2 concepts from Network Flow

Circulation with Demands

A Feasibility Problem…

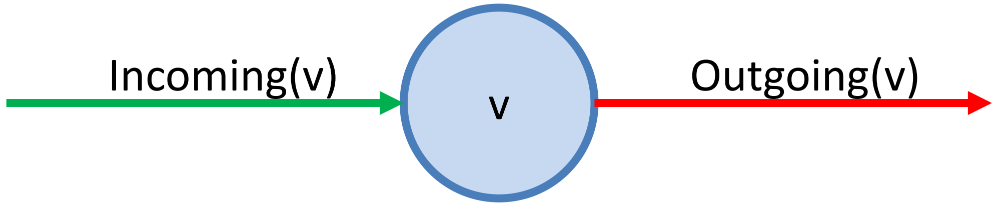
- Recall the 2 concepts from Network Flow
  - Capacity Constraint
    - Flow <= Capacity for an edge
  - Flow conservation
    - Incoming flow to a vertex == outgoing flow from the vertex

Circulation with Demands
A Feasibility Problem…

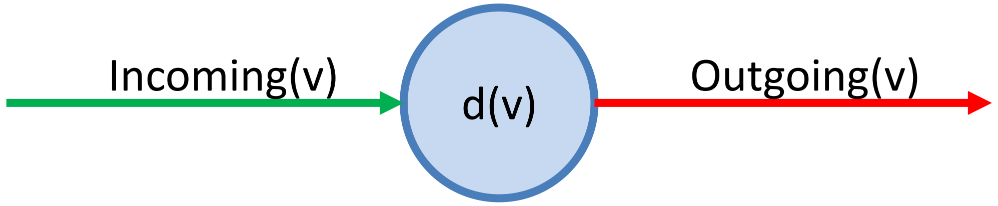
- Recall the 2 concepts from Network Flow
  - Capacity Constraint
    - Flow <= Capacity for an edge
  - Flow conservation
    - Incoming flow to a vertex == outgoing flow from the vertex
    - Now what if we tweak this rule?

Circulation with Demands
A Feasibility Problem…

- Recall the 2 concepts from Network Flow
  - Capacity Constraint
    - Flow <= Capacity for an edge
  - Flow conservation
    - Incoming flow to a vertex == outgoing flow from the vertex

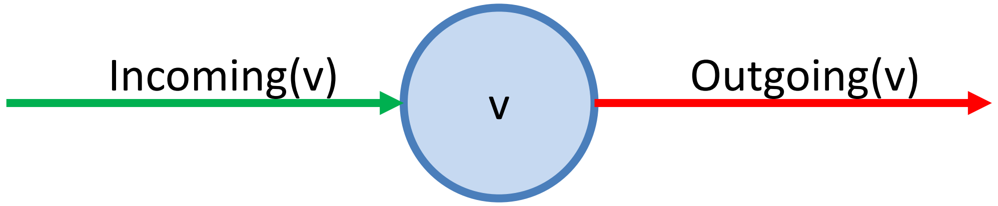


    - Now what if we tweak this rule?

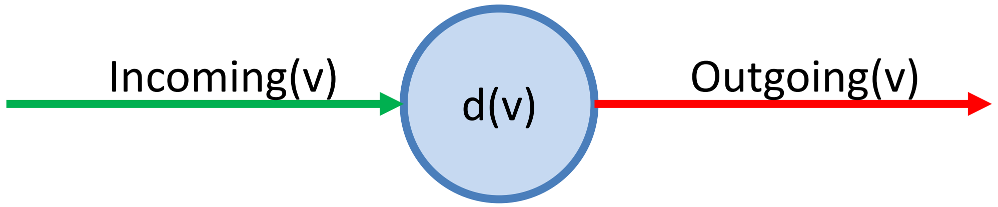


Circulation with Demands
A Feasibility Problem…

- Recall the 2 concepts from Network Flow
  - Capacity Constraint
    - Flow <= Capacity for an edge
  - Flow conservation
    - Incoming flow to a vertex == outgoing flow from the vertex

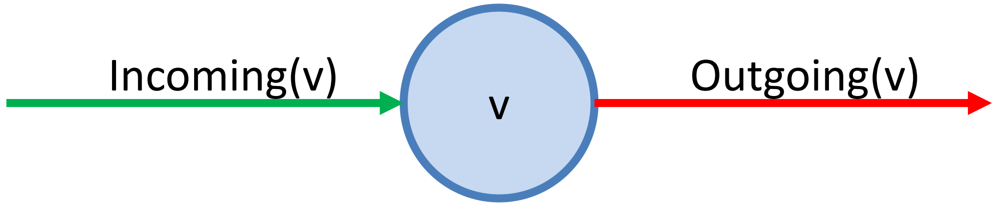


    - Now what if we tweak this rule? incoming(v) - outgoing(v) = d(v)

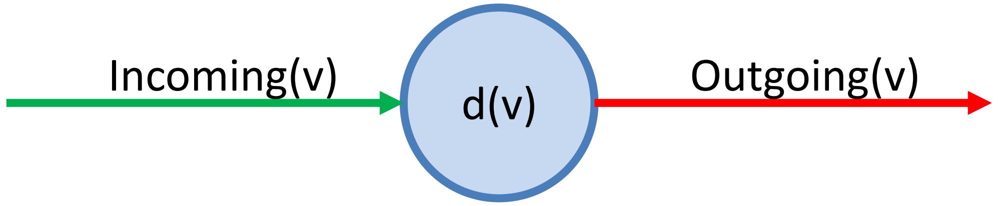


Circulation with Demands
A Feasibility Problem…

- Recall the 2 concepts from Network Flow
  - Capacity Constraint
    - Flow <= Capacity for an edge
  - Flow conservation
    - Incoming flow to a vertex == outgoing flow from the vertex

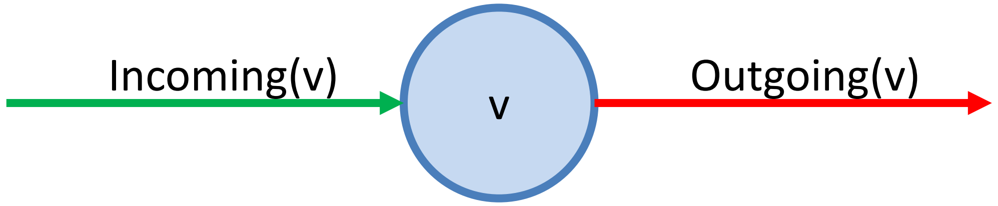


    - Now what if we tweak this rule? incoming(v) - outgoing(v) = demand(v)

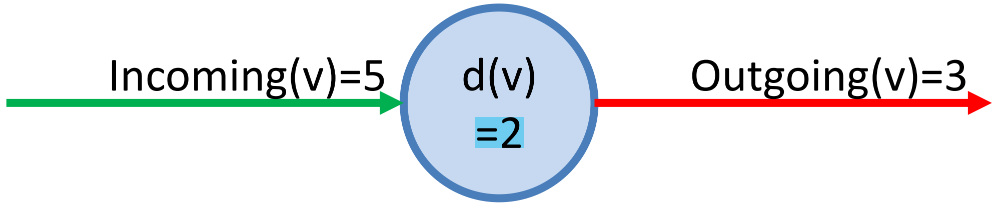


Circulation with Demands
A Feasibility Problem…

- Recall the 2 concepts from Network Flow
  - Capacity Constraint
    - Flow <= Capacity for an edge
  - Flow conservation
    - Incoming flow to a vertex == outgoing flow from the vertex

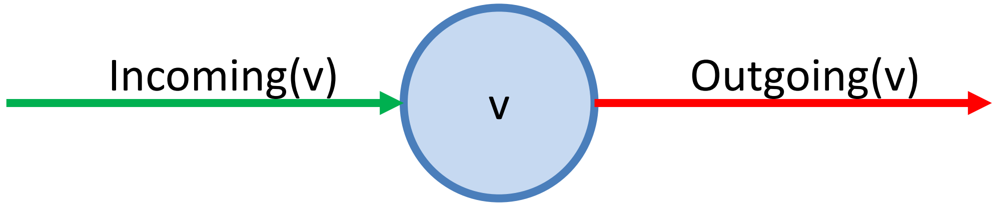


    - Now what if we tweak this rule? incoming(v) - outgoing(v) = demand(v)

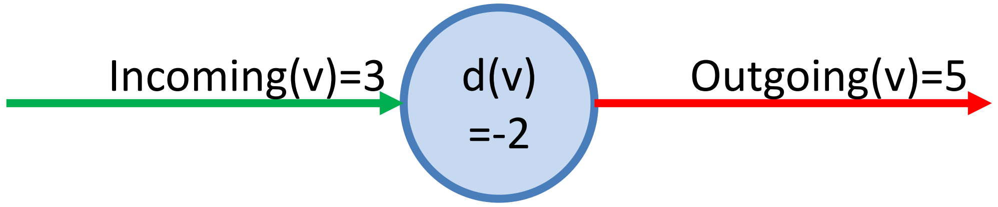


Circulation with Demands
A Feasibility Problem…

- Recall the 2 concepts from Network Flow
  - Capacity Constraint
    - Flow <= Capacity for an edge
  - Flow conservation
    - Incoming flow to a vertex == outgoing flow from the vertex

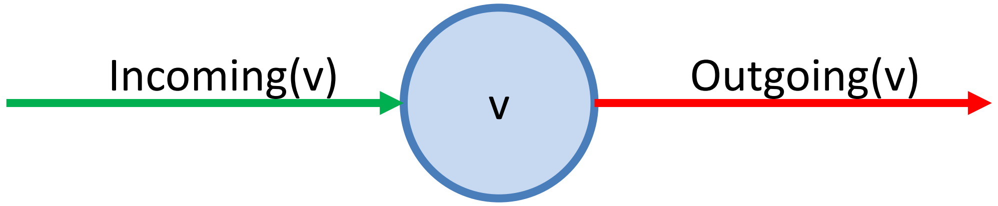


- Now what if we tweak this rule? incoming(v) - outgoing(v) = demand(v)

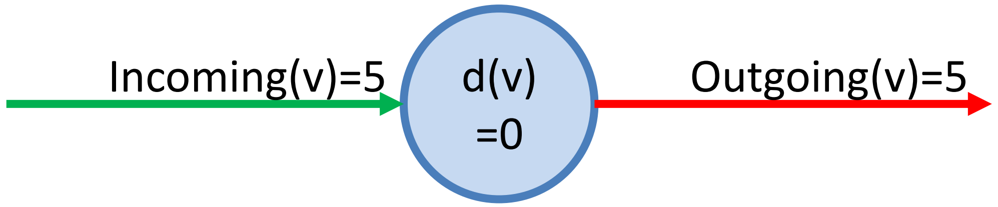


Circulation with Demands
A Feasibility Problem…

- Recall the 2 concepts from Network Flow
  - Capacity Constraint
    - Flow <= Capacity for an edge
  - Flow conservation
    - Incoming flow to a vertex == outgoing flow from the vertex



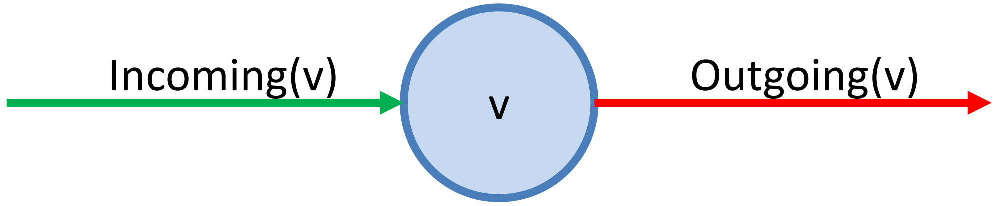
    - Now what if we tweak this rule? incoming(v) - outgoing(v) = demand(v)



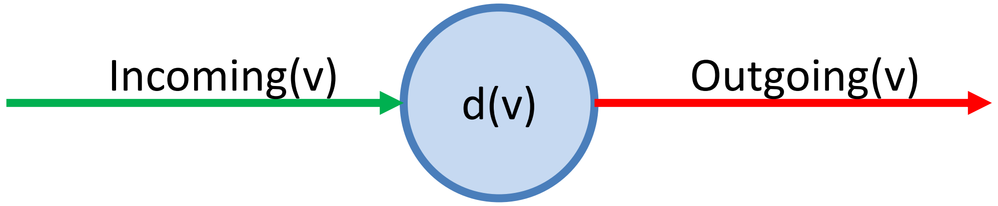
Circulation with Demands

A Feasibility Problem…

- Recall the 2 concepts from Network Flow
  - Capacity Constraint
    - Flow <= Capacity for an edge
  - ~~Flow conservation~~ Demand Constraint
    - Incoming flow to a vertex == outgoing flow from the vertex

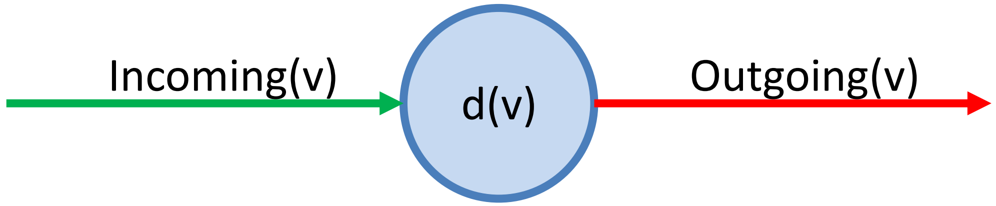


    - Now what if we tweak this rule? incoming(v) - outgoing(v) = demand(v)



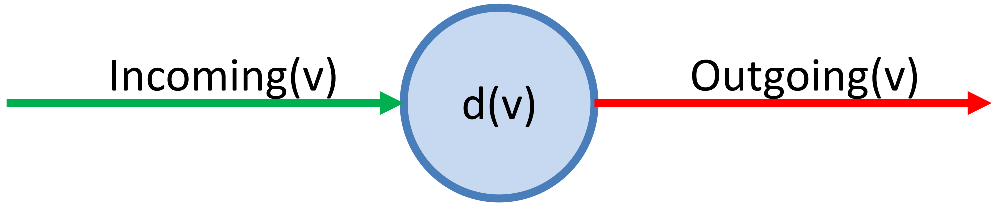
Circulation with Demands
A Feasibility Problem…

- Recall the 2 concepts from Network Flow
  - Capacity Constraint
    - Flow <= Capacity for an edge
  - Demand Constraint
    - Now what if we tweak this rule? incoming(v) - outgoing(v) = demand(v)



Circulation with Demands
A Feasibility Problem…

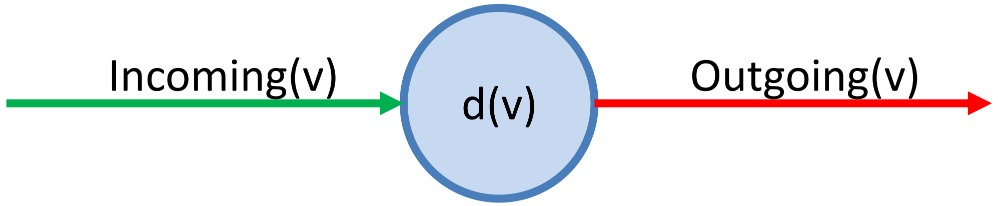
- Recall the 2 concepts from Network Flow
  - Capacity Constraint
    - Flow <= Capacity for an edge
  - Demand Constraint
    - Now what if we tweak this rule? incoming(v) - outgoing(v) = demand(v)



- Circulation with Demands is a feasibility problem

Circulation with Demands
A Feasibility Problem…

- Recall the 2 concepts from Network Flow
  - Capacity Constraint
    - Flow <= Capacity for an edge
  - Demand Constraint
    - Now what if we tweak this rule? incoming(v) - outgoing(v) = demand(v)



- Circulation with Demands is a feasibility problem that satisfy both of the above!

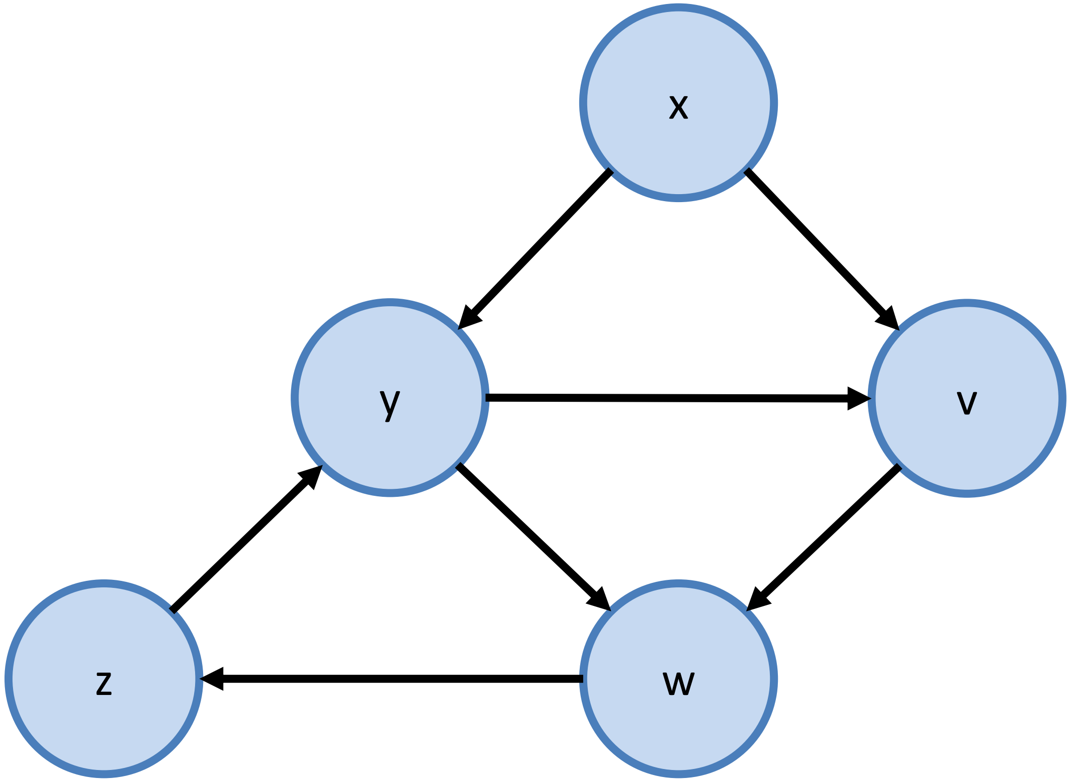
MONASH University

Questions?

Circulation with Demands

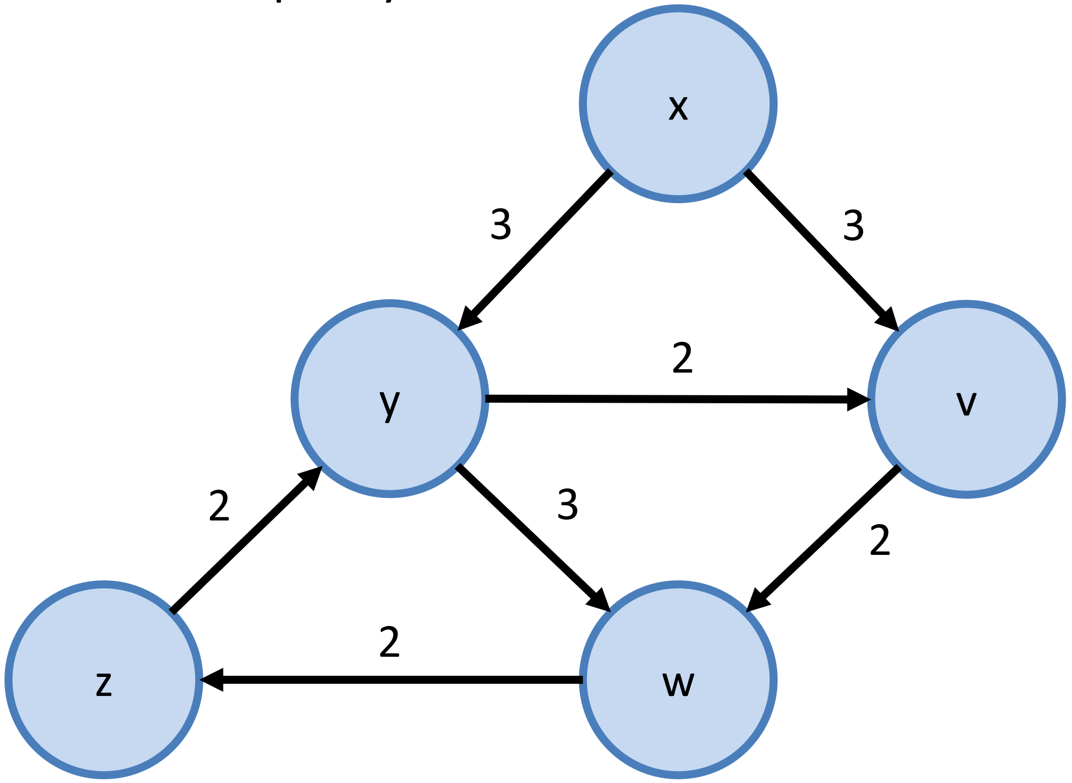
A Feasibility Problem…

- An example below…



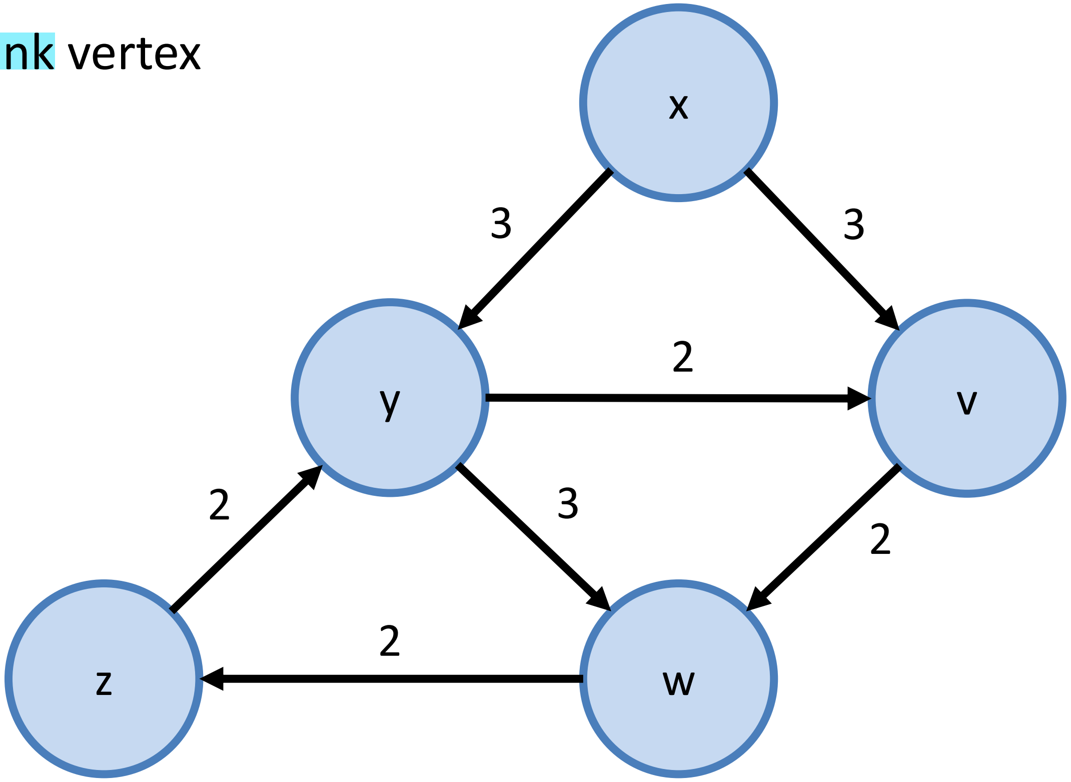
Circulation with Demands
A Feasibility Problem…

- An example below…
  - With the usual capacity



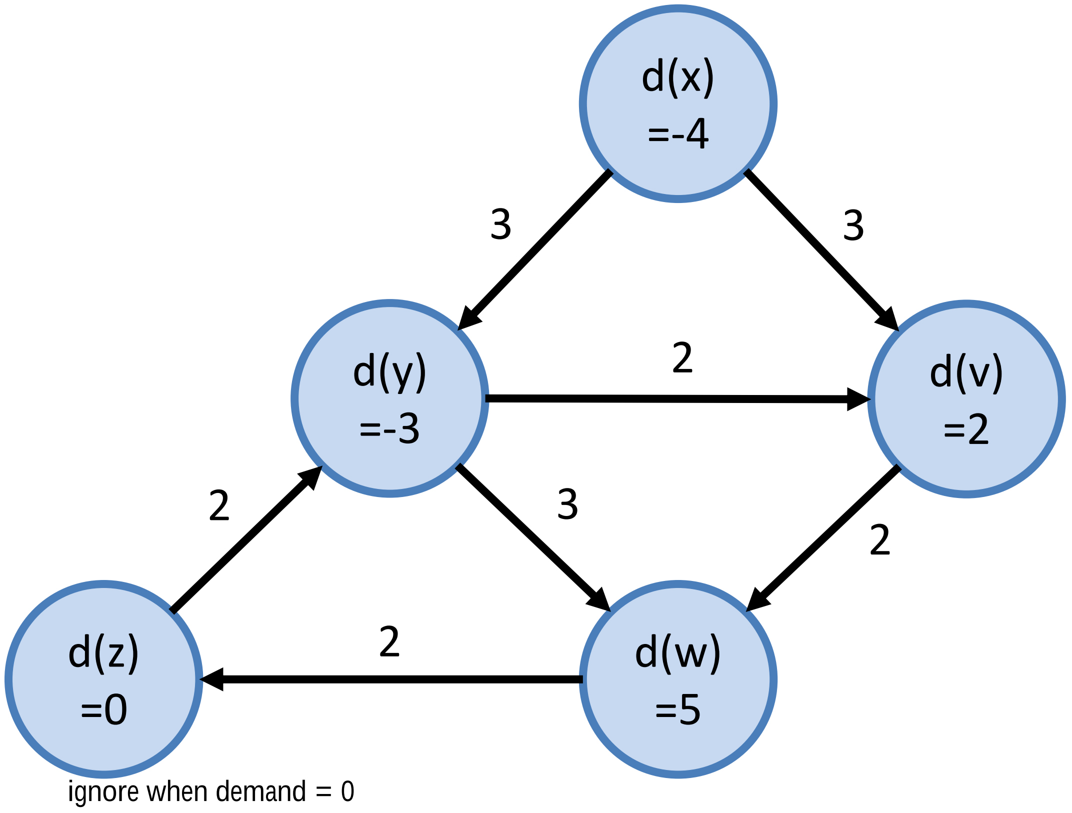
Circulation with Demands
A Feasibility Problem…

- An example below…
  - No source vertex
  - No sink vertex



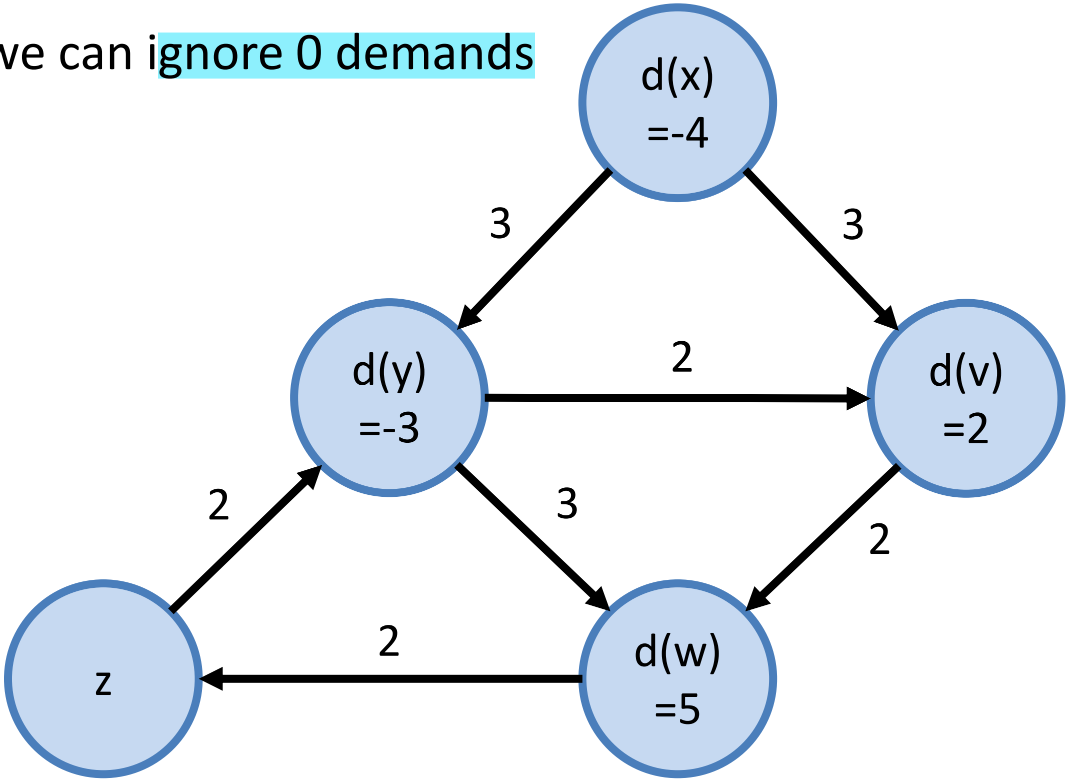
Circulation with Demands
A Feasibility Problem…

- An example below…
  - Vertex with demands



Circulation with Demands
A Feasibility Problem…

- An example below…
  - Vertex with demands
  - But we can ignore 0 demands

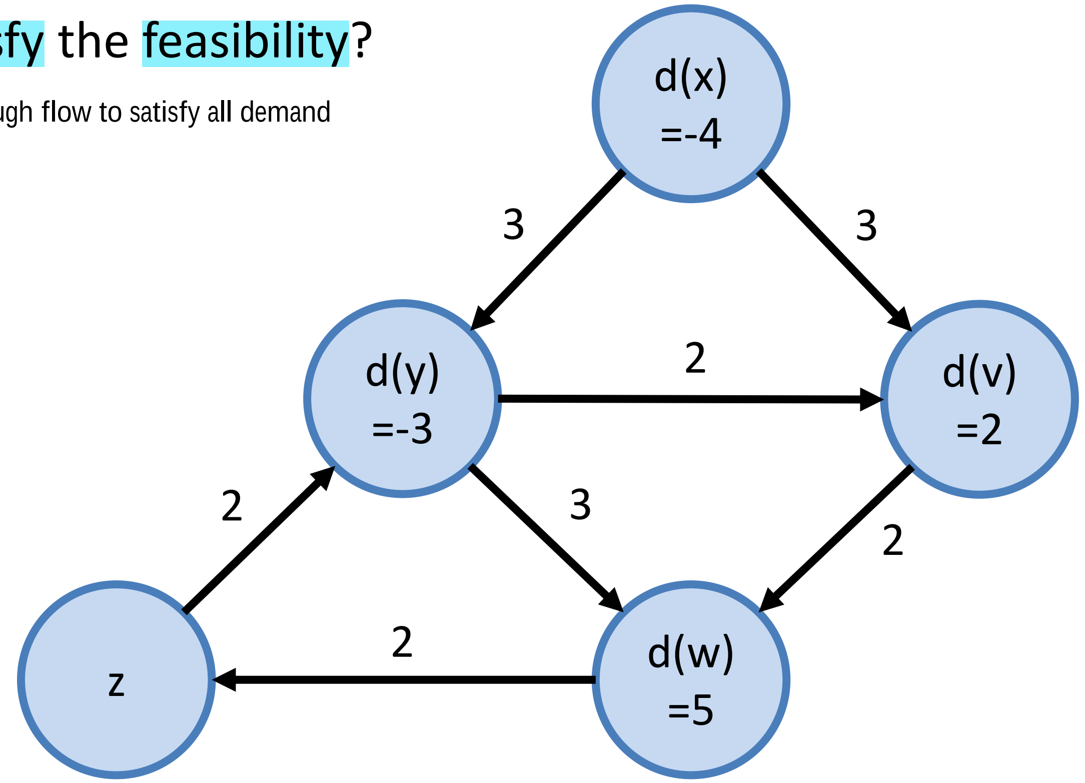


Circulation with Demands

A Feasibility Problem…

- An example below…
  - Can we have a flow that satisfy the feasibility?

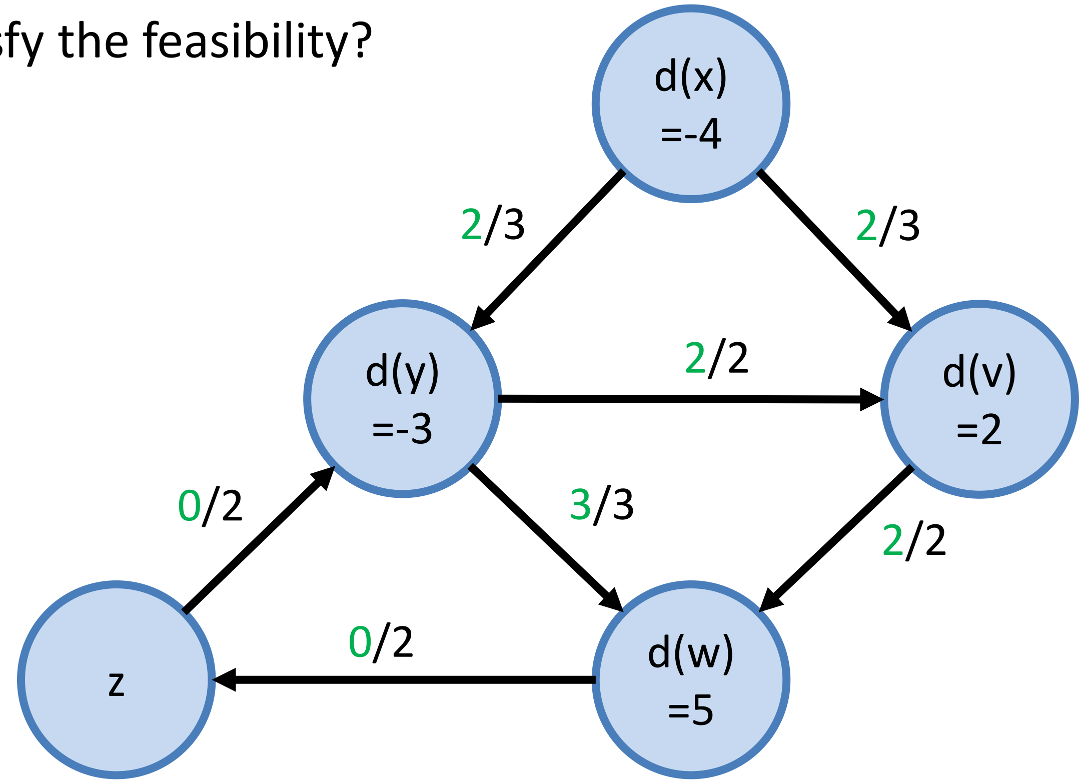
    enough flow to satisfy all demand



Circulation with Demands

A Feasibility Problem…

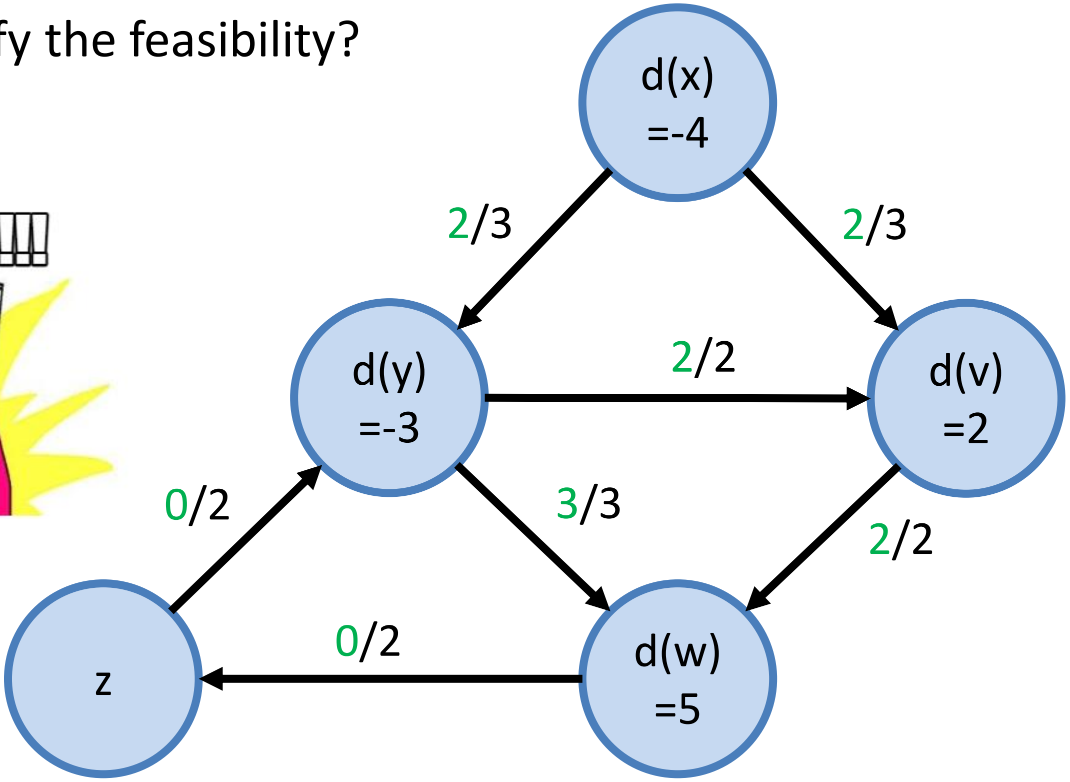
- An example below…
  - Can we have a flow that satisfy the feasibility?



Circulation with Demands

A Feasibility Problem…

- An example below…
  - Can we have a flow that satisfy the feasibility?
  - YES

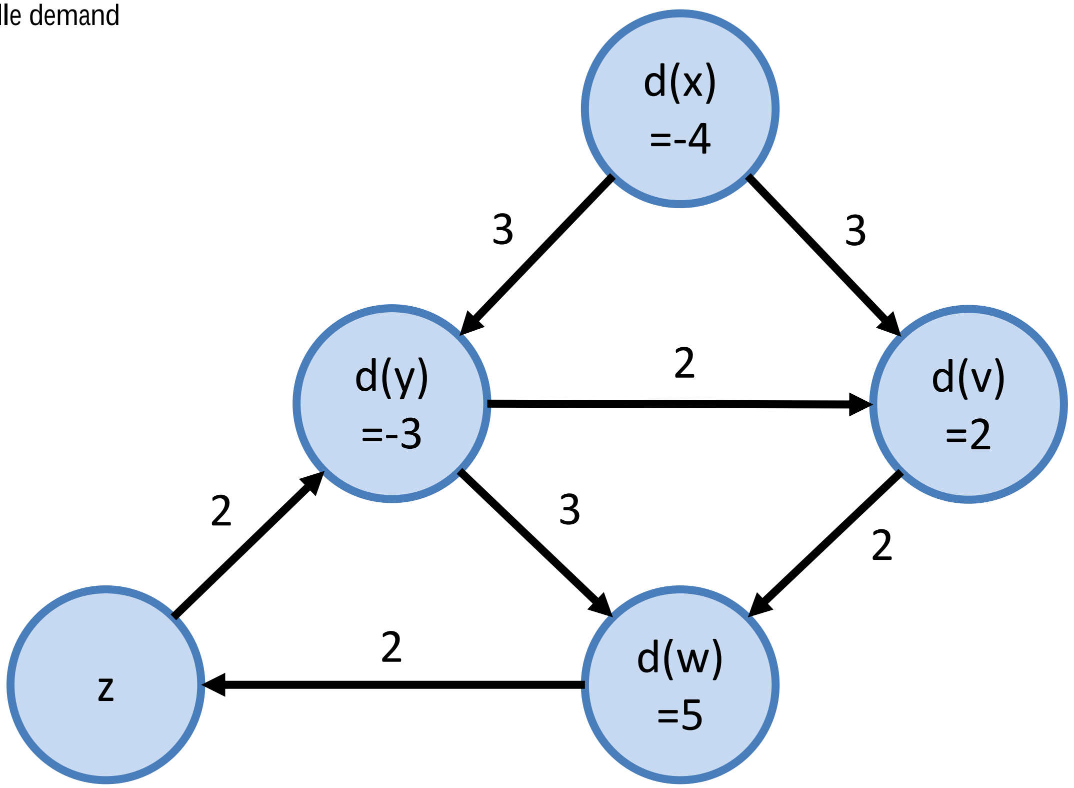


Questions?

Circulation with Demands
How to Check Feasibility…

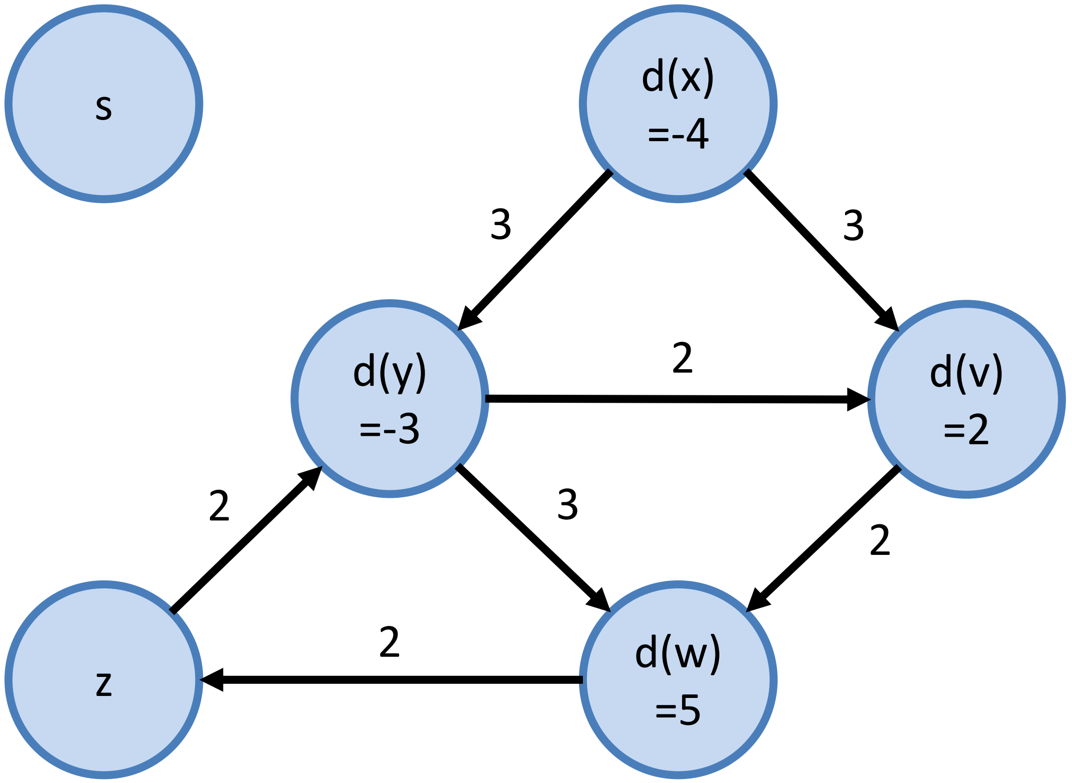
- Given this… How do we solve this?

convert to normal network flow
need to handle demand



Circulation with Demands
How to Check Feasibility…

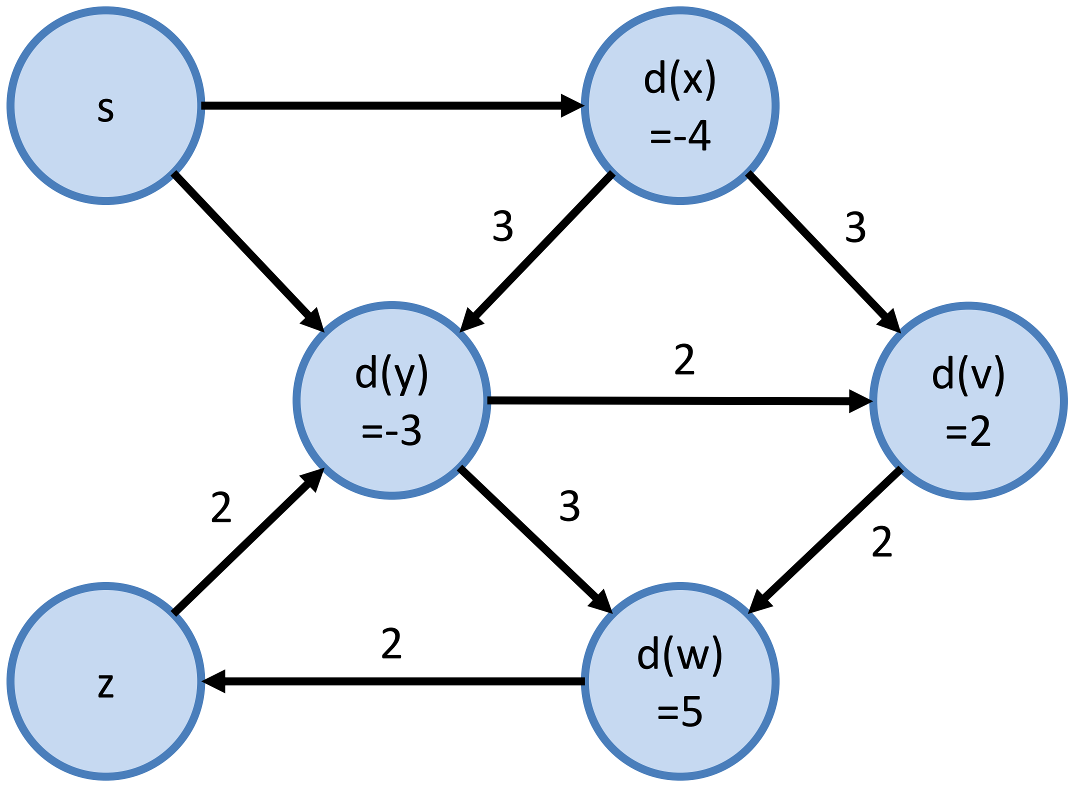
- Make a source



Circulation with Demands

How to Check Feasibility…

- Make a source, link to all negative demand

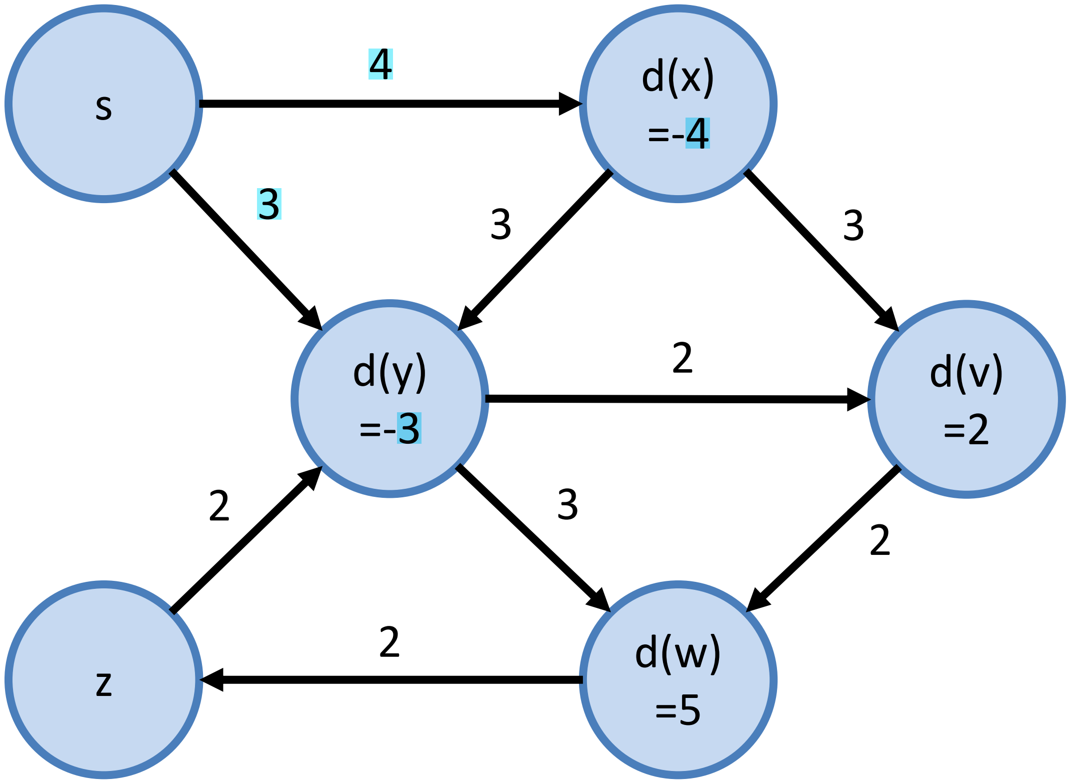


Circulation with Demands

How to Check Feasibility…

- Make a source, link to all negative demand, weighted

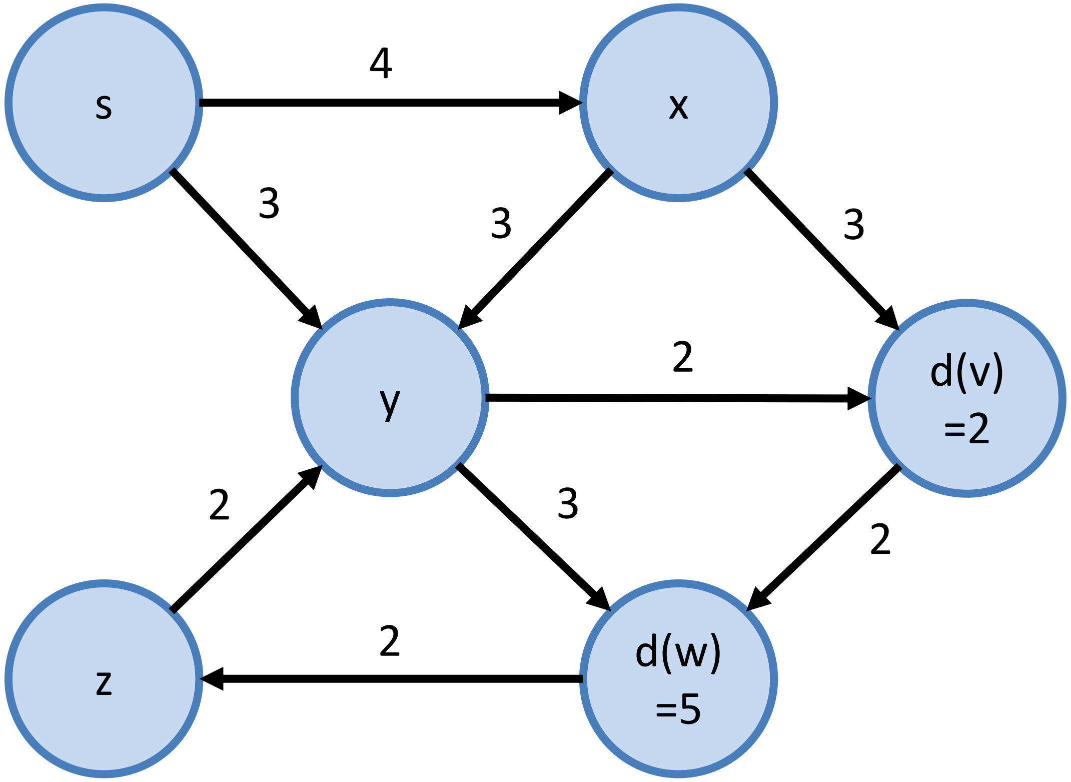
imaginary source to provide demanded flow to vertices that has negative demand to neutralise the negative value



Circulation with Demands

How to Check Feasibility…

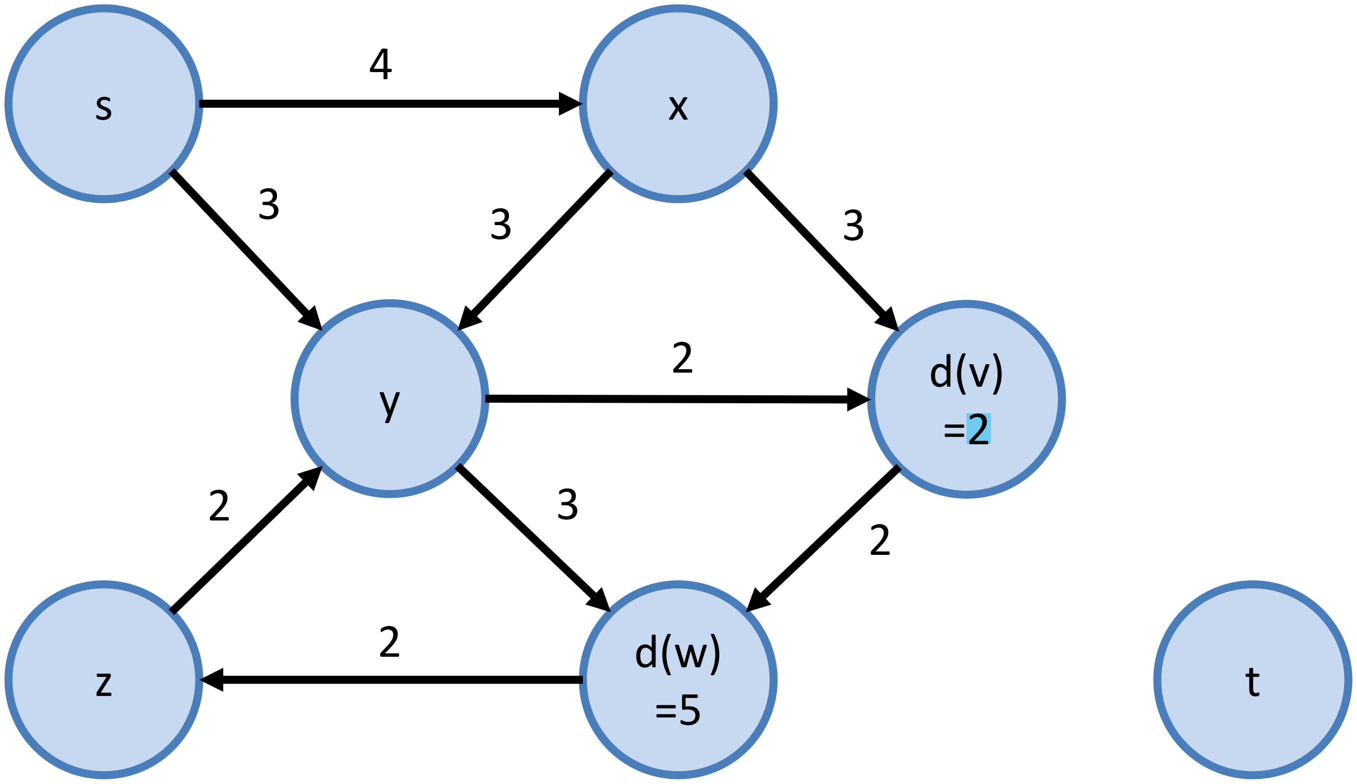
- We are done! for source >.<



Circulation with Demands

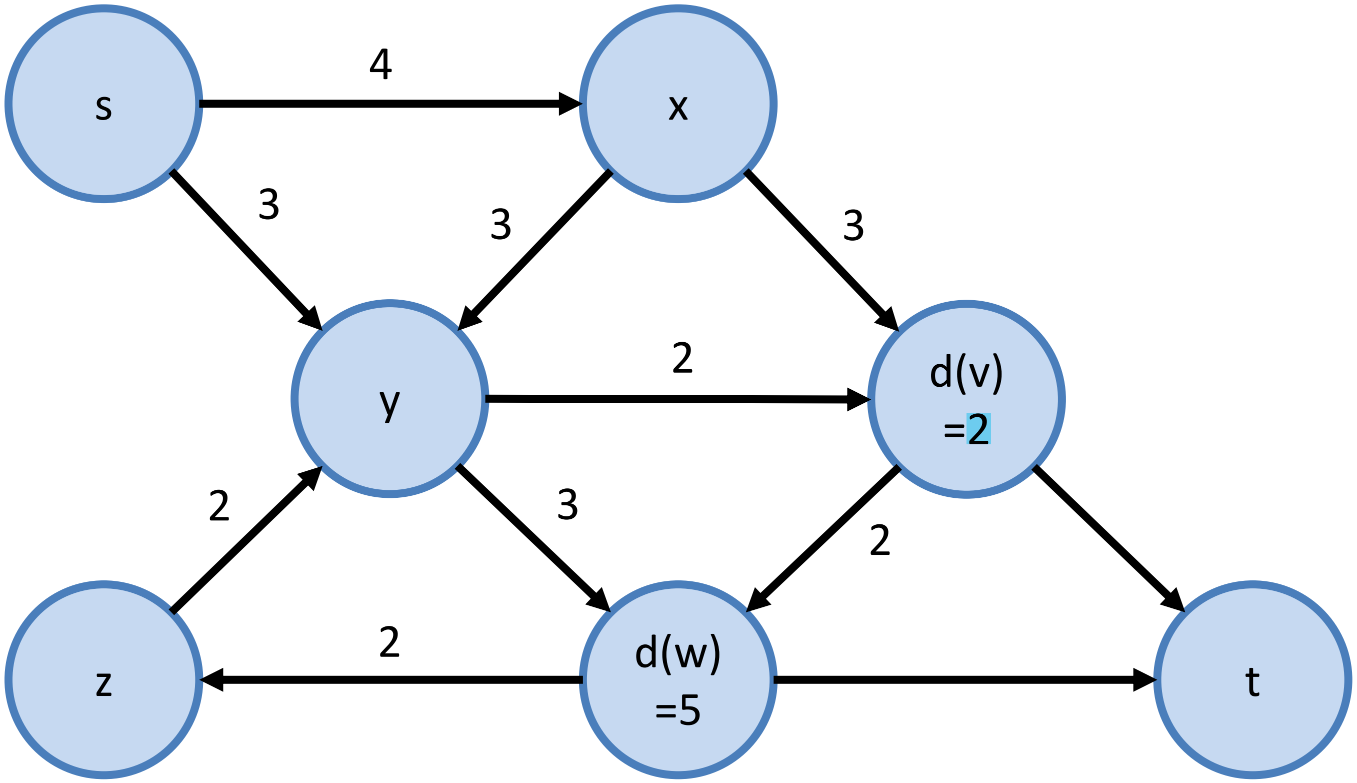
How to Check Feasibility…

- Make a sink



Circulation with Demands
How to Check Feasibility…

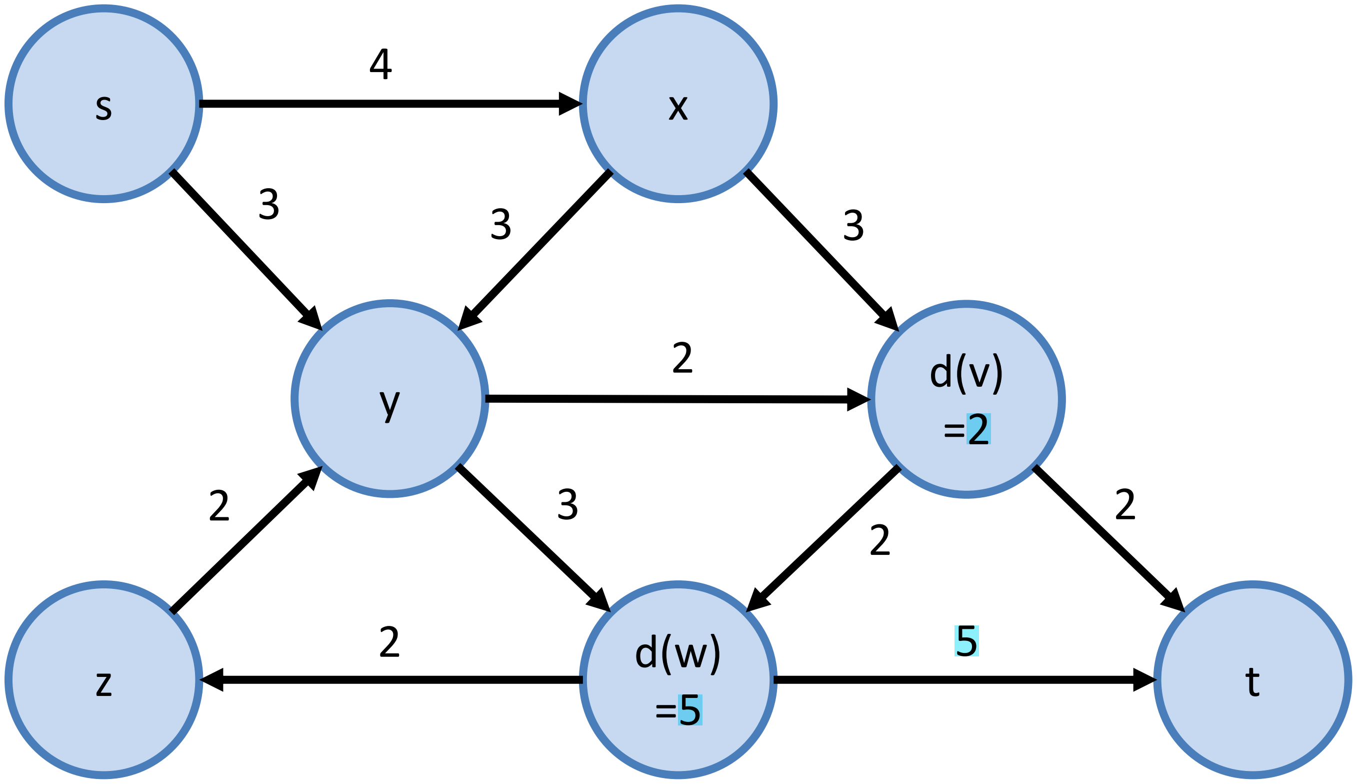
- Make a sink, link from positive demand



Circulation with Demands

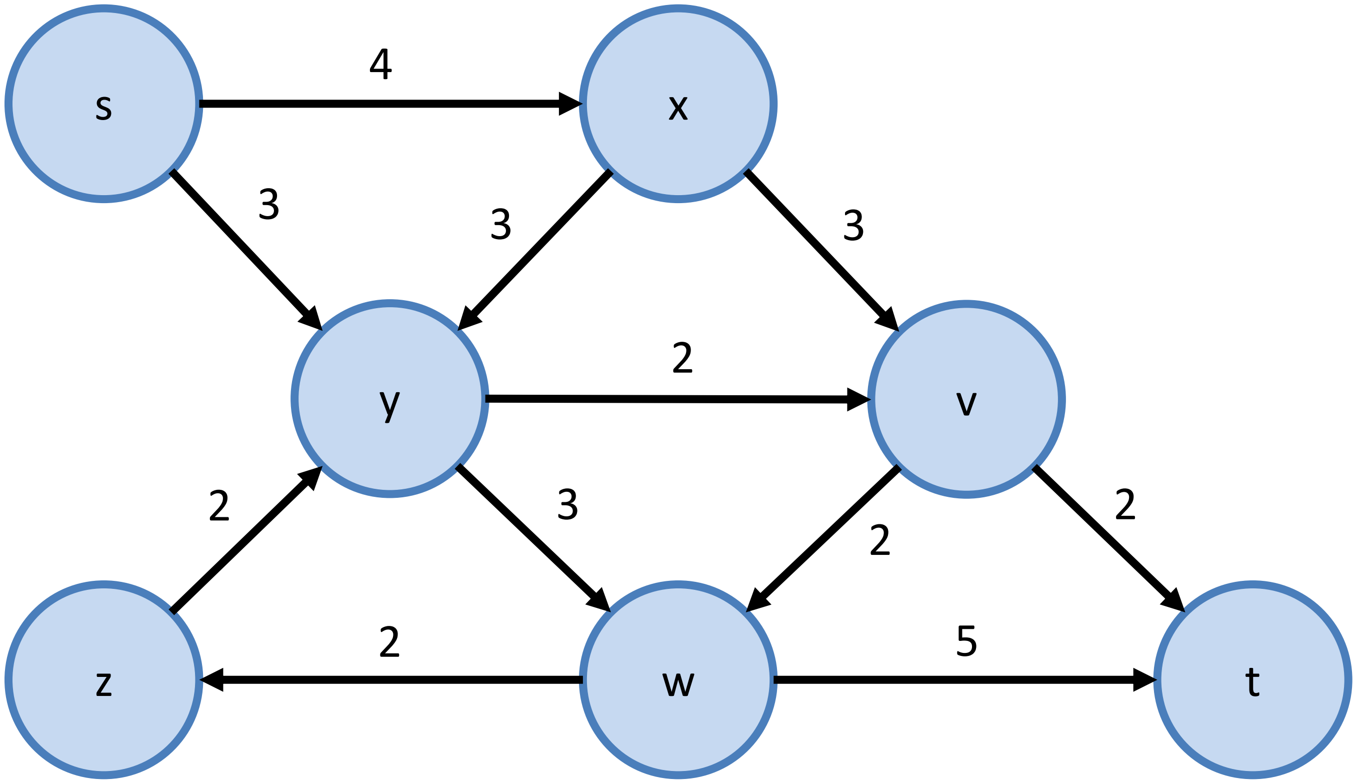
How to Check Feasibility…

- Make a sink, link from positive demand, weighted



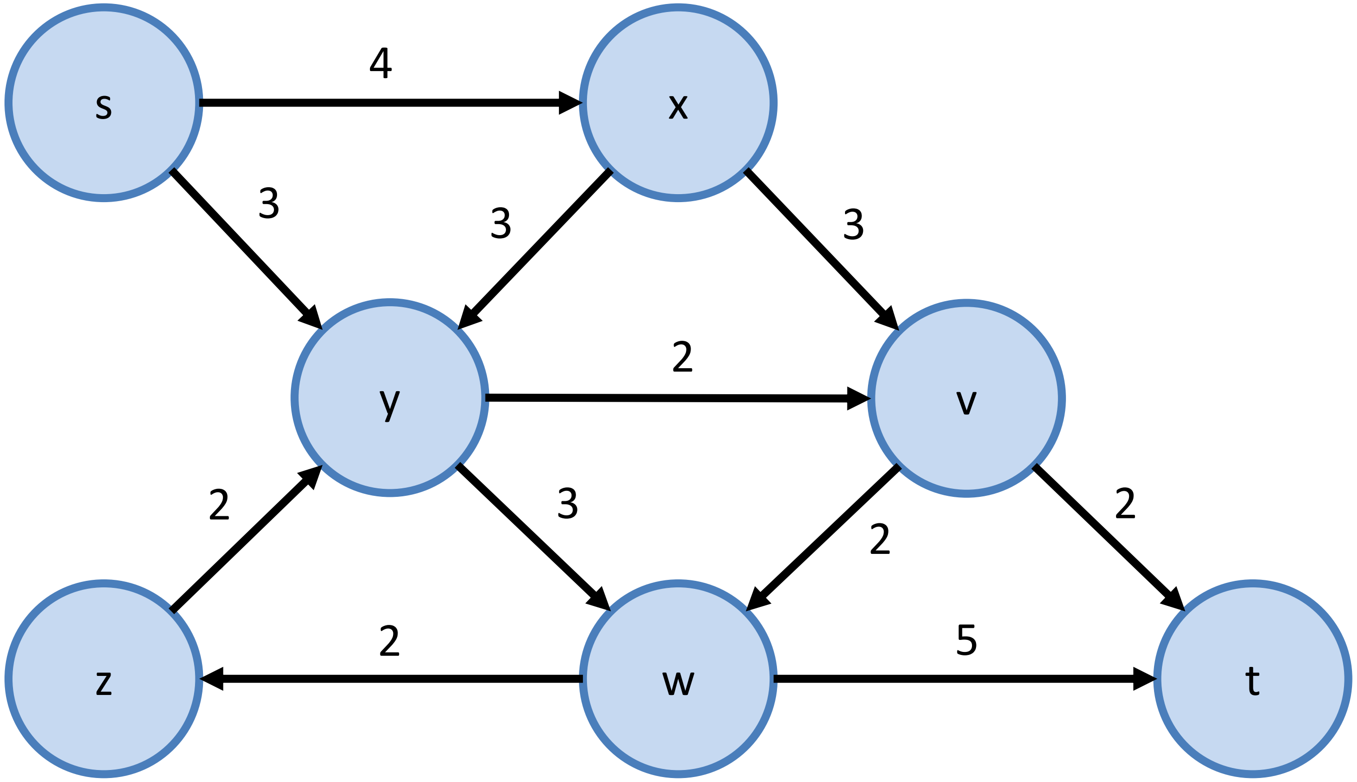
Circulation with Demands
How to Check Feasibility…

- We are done! for sink >.<



Circulation with Demands
How to Check Feasibility…

- Now same as network flow!

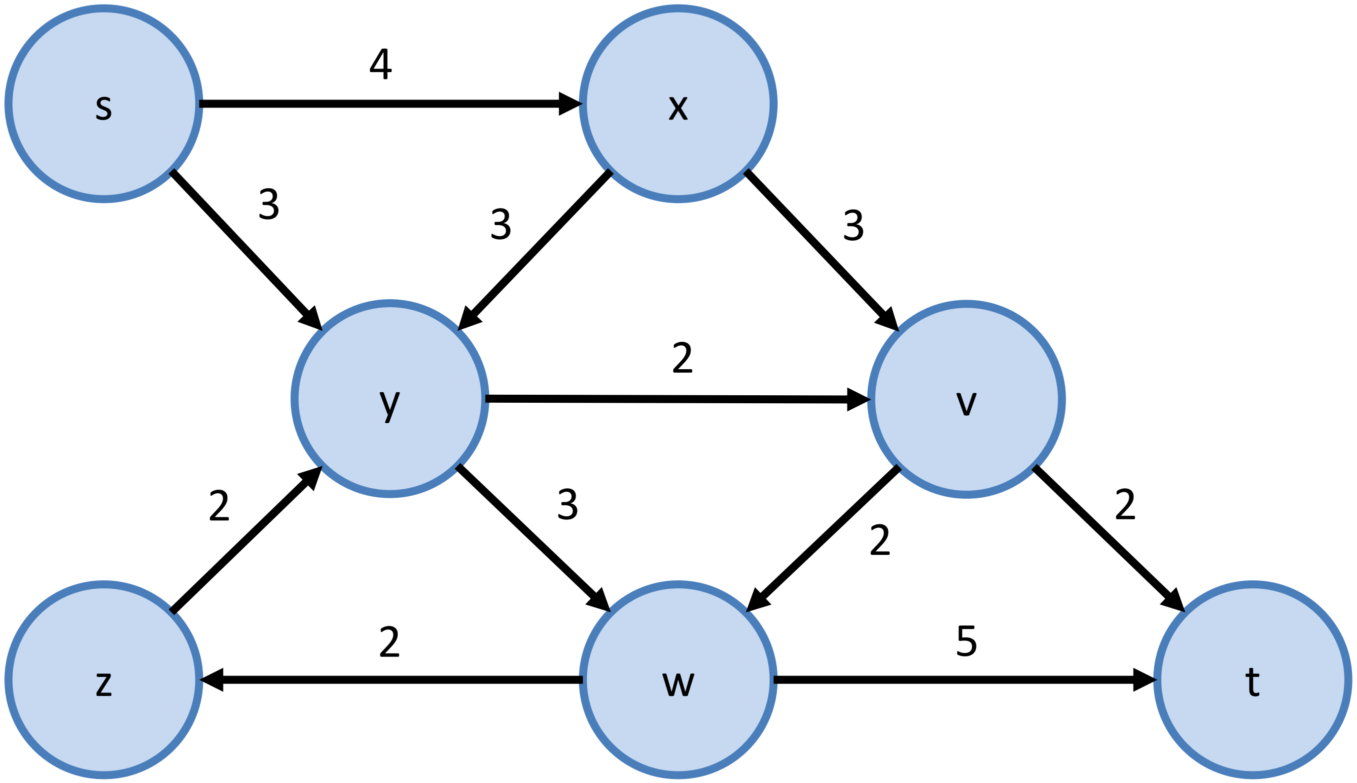

$$\begin{tikzcd}
s \arrow[rr, "4"] \arrow[dr, "3"] & & x \arrow[dl, "3"] \arrow[dr, "3"] & & \\
& y \arrow[rr, "2"] \arrow[dr, "3"] & & v \arrow[dl, "2"] \arrow[dr, "2"] & \\
z \arrow[ur, "2"] & & w \arrow[ll, "2"] \arrow[rr, "5"] & & t
\end{tikzcd}$$

MONASH University

Questions?

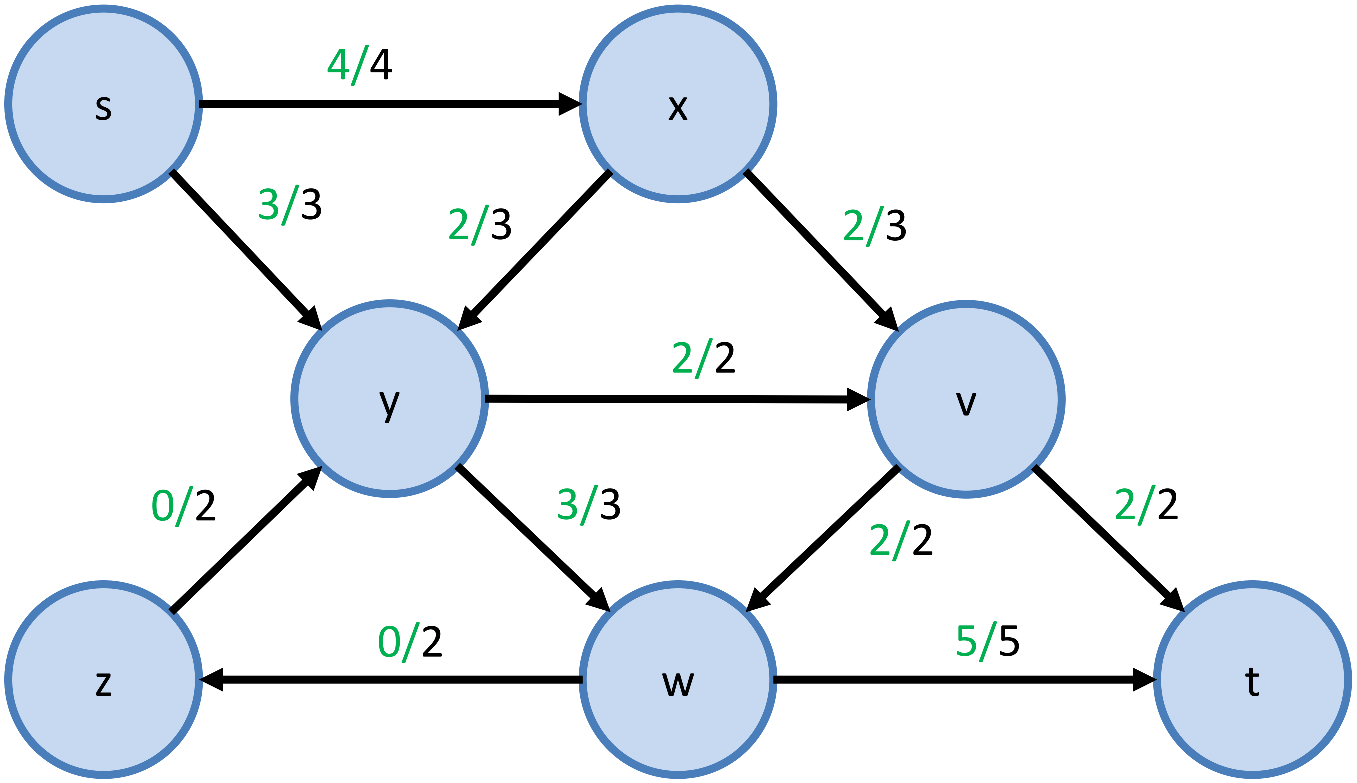
Circulation with Demands
How to Check Feasibility…

- Then we run Ford-Fulkerson



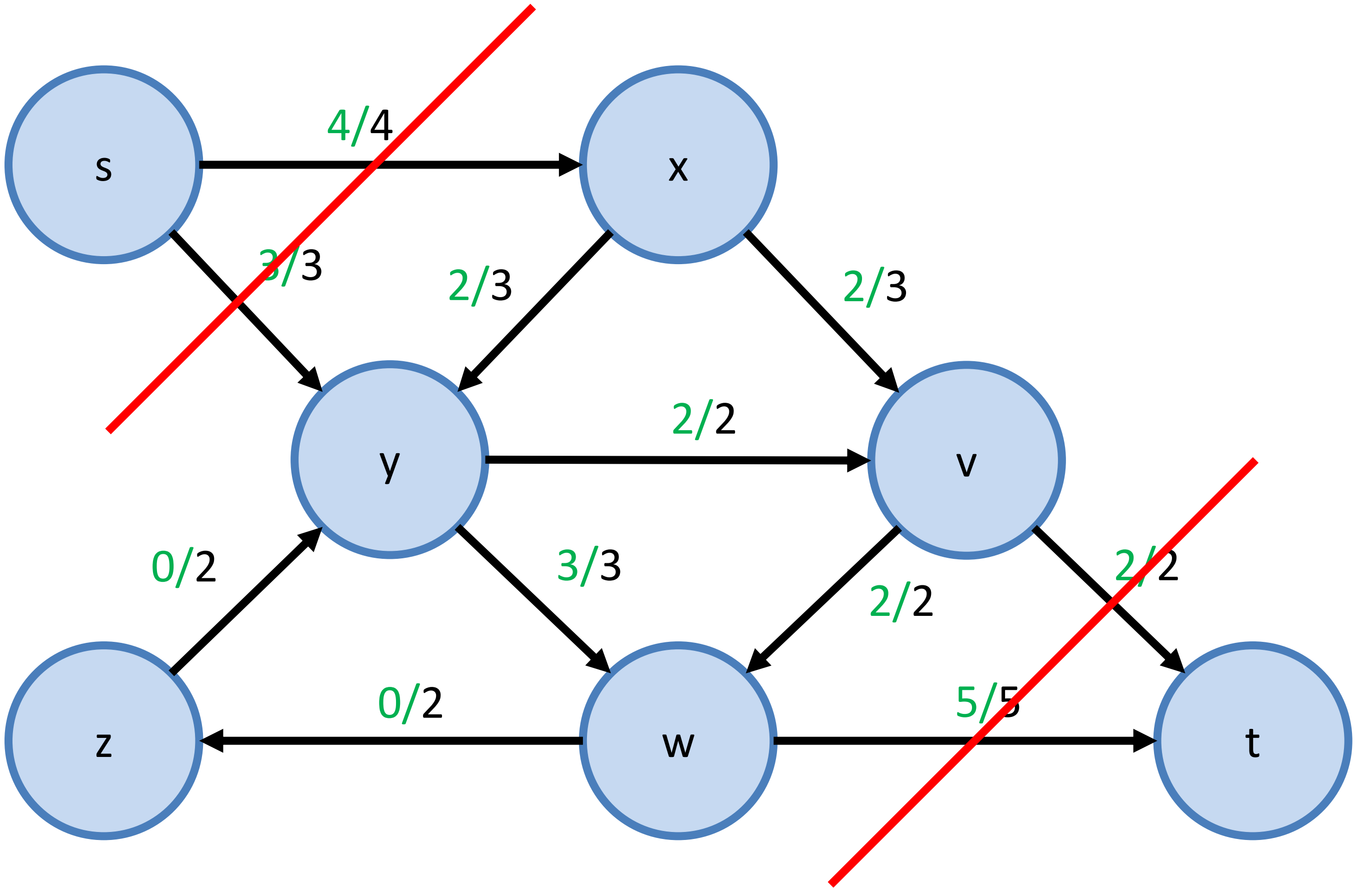
Circulation with Demands
How to Check Feasibility…

- Then we run Ford-Fulkerson



Circulation with Demands
How to Check Feasibility…

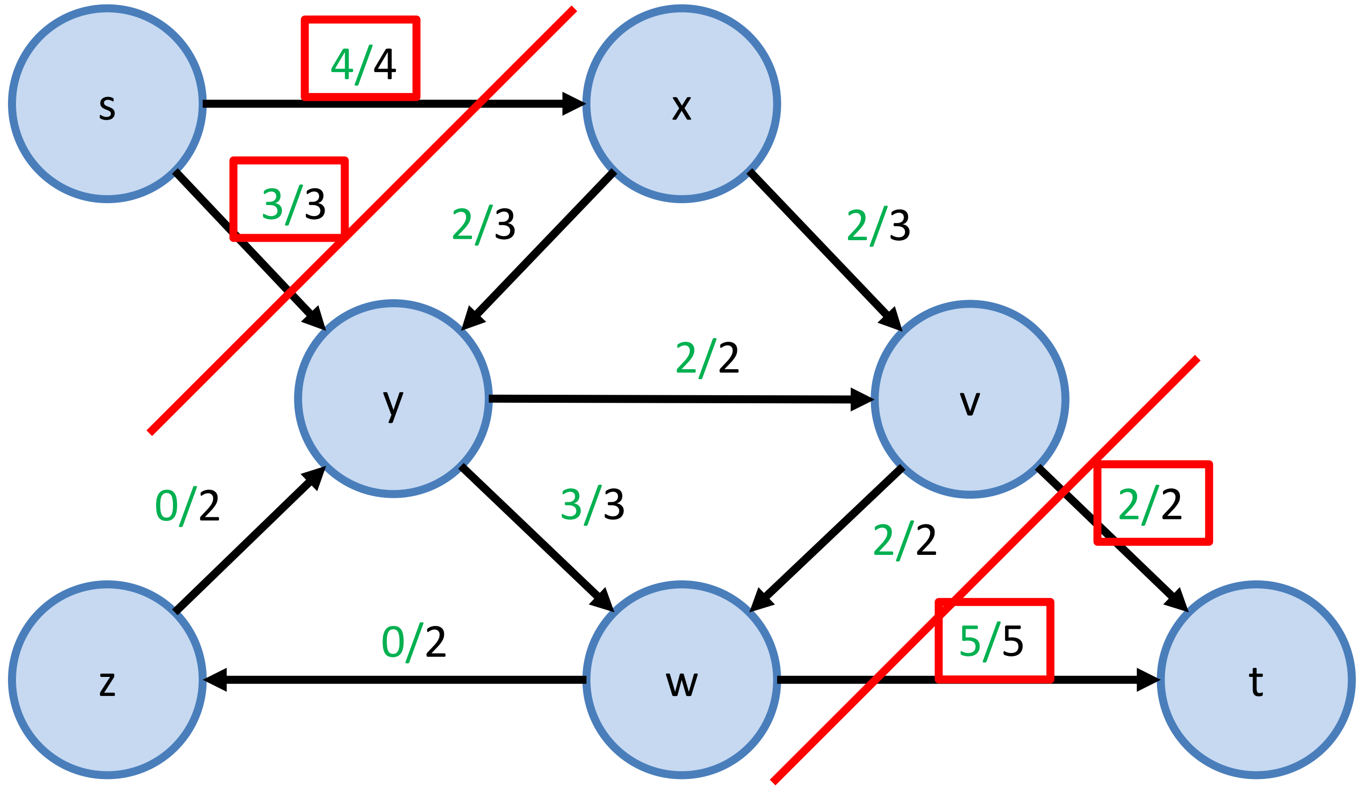
- If we have a cut…



Circulation with Demands

How to Check Feasibility…

- If we have a cut… find min-cut to get max-flow



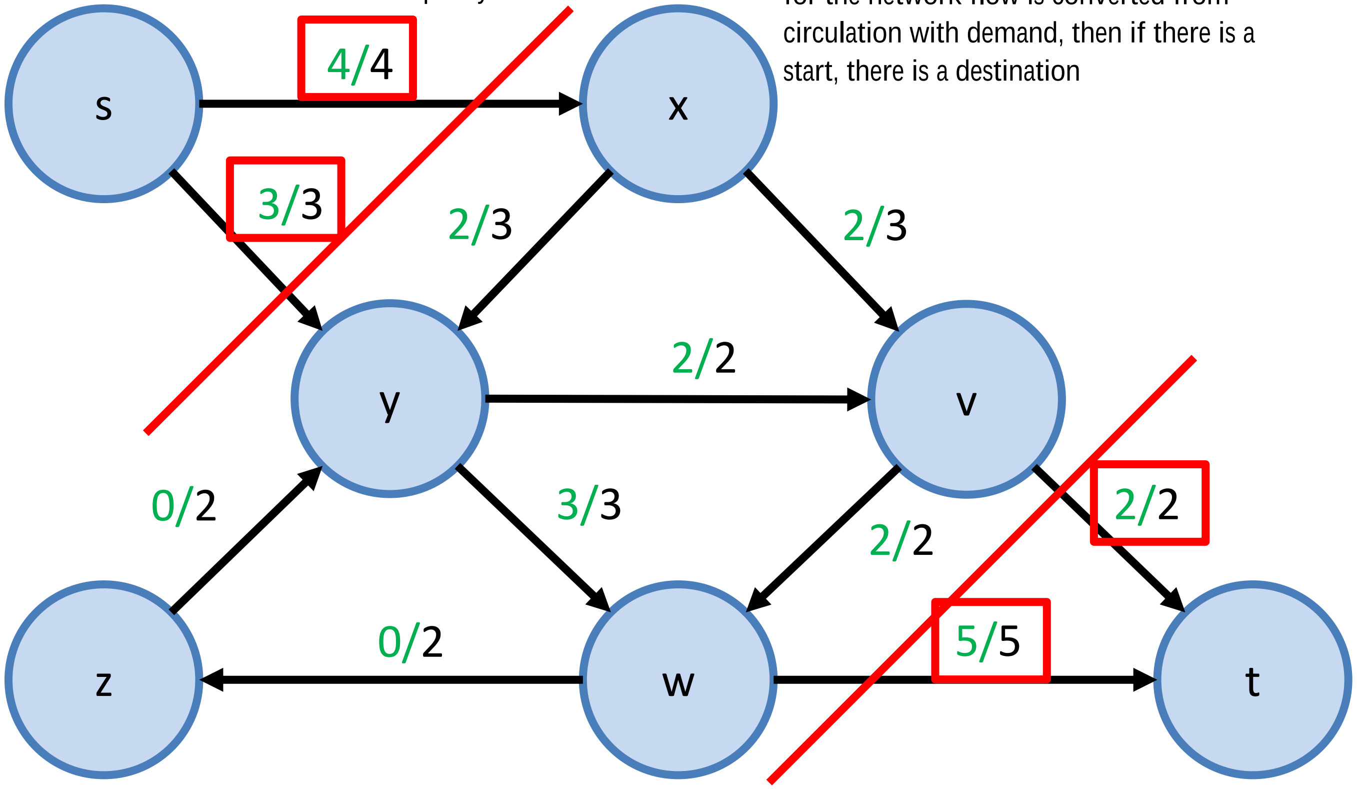
Circulation with Demands

How to Check Feasibility…

- If we have a cut… it is feasible!

as long as there is a cut, then the original network is feasible, flow = capacity

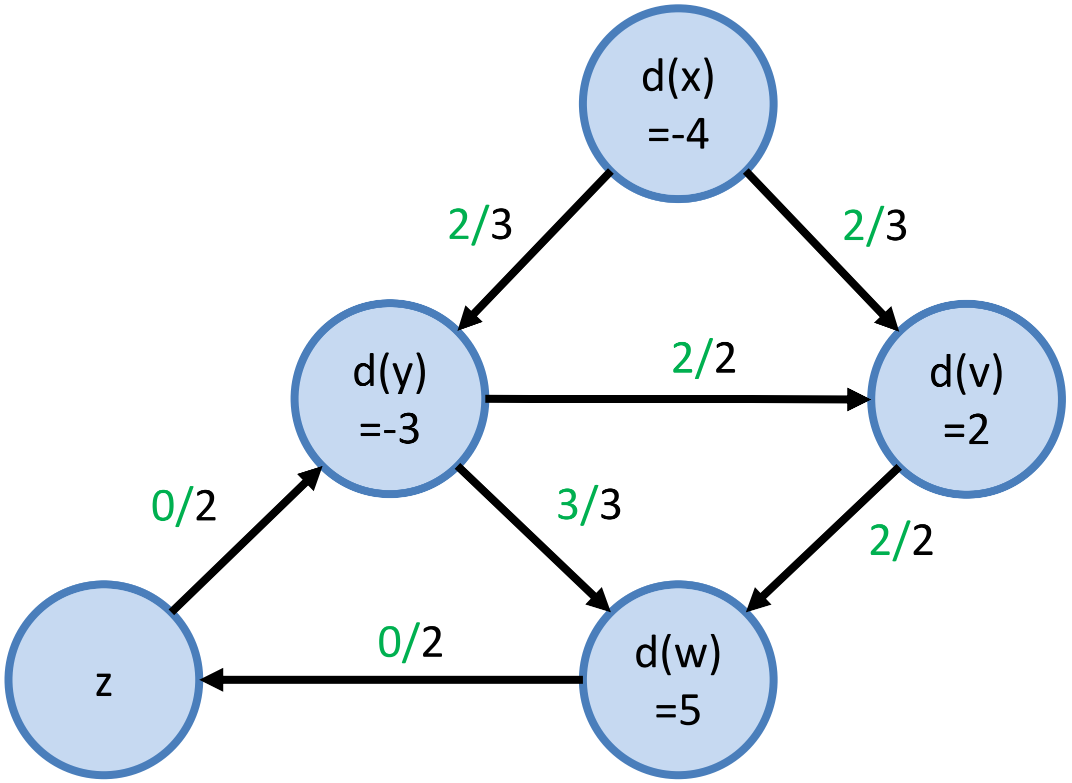
for the network flow is converted from circulation with demand, then if there is a start, there is a destination



Circulation with Demands

How to Check Feasibility…

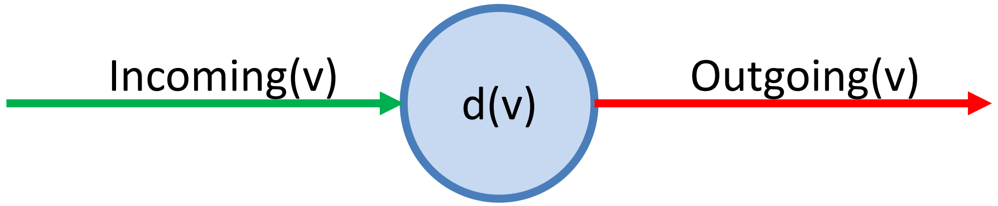
- Then we just clean it up as the solution



Questions?

Circulation with Demands and Lower Bound
A Feasibility Problem…

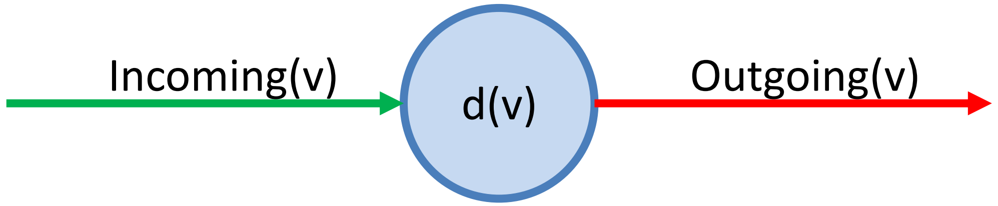
- Recall the 2 concepts from Circulation with Demands
  - Capacity Constraint
    - Flow <= Capacity for an edge
  - Demand Constraint
    - Now what if we tweak this rule? incoming(v) - outgoing(v) = demand(v)



- Circulation with Demands is a feasibility problem that satisfy both of the above!

Circulation with Demands and Lower Bound
A Feasibility Problem…

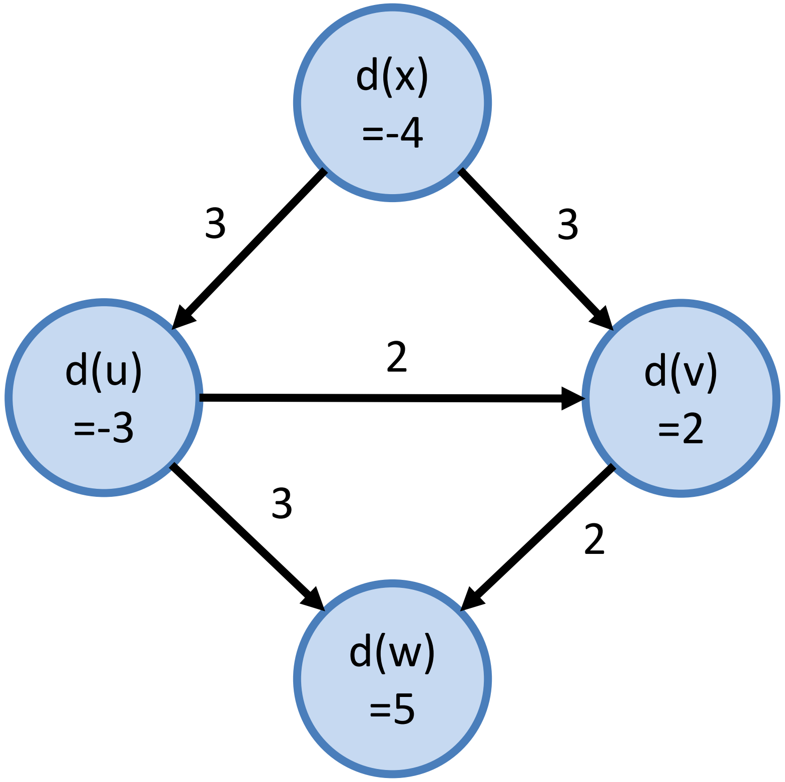
- Recall the 2 concepts from Circulation with Demands
  - Capacity Constraint
    - Lower bound for an edge <= Flow <= $\overset{\text{upper bound}}{\text{Capacity}}$ for an edge
  - Demand Constraint
    - Now what if we tweak this rule? incoming(v) - outgoing(v) = demand(v)



- Circulation with Demands is a feasibility problem that satisfy both of the above!

Circulation with Demands and Lower Bound
A Feasibility Problem…

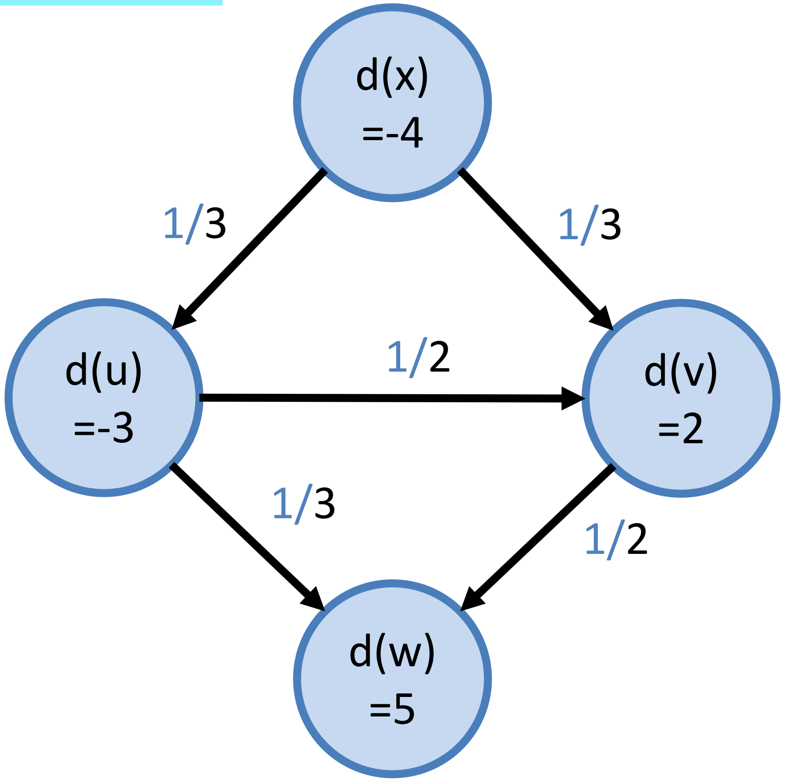
- Consider the following…



Circulation with Demands and Lower Bound

A Feasibility Problem…

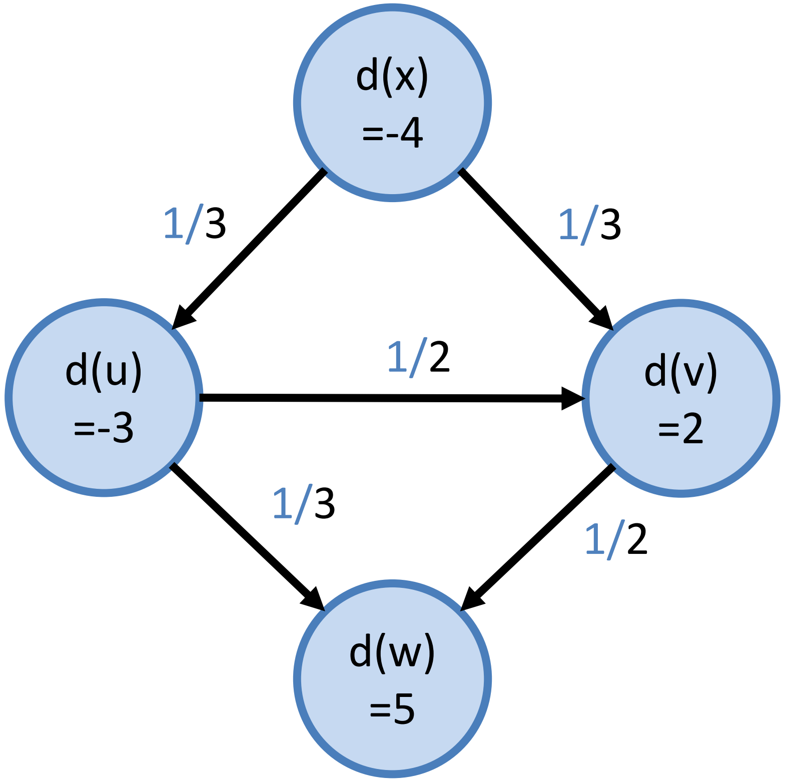
- Consider the following… edges have lower bound of 1



Circulation with Demands and Lower Bound

A Feasibility Problem…

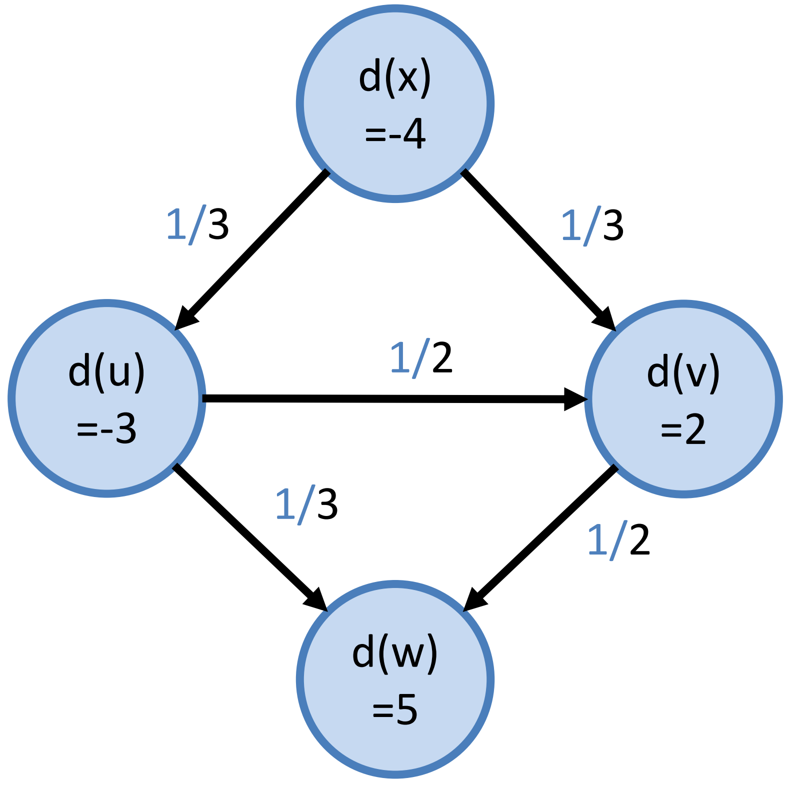
- Can we find a feasible solution?



Circulation with Demands and Lower Bound

A Feasibility Problem…

- Can we find a feasible solution? Of course!

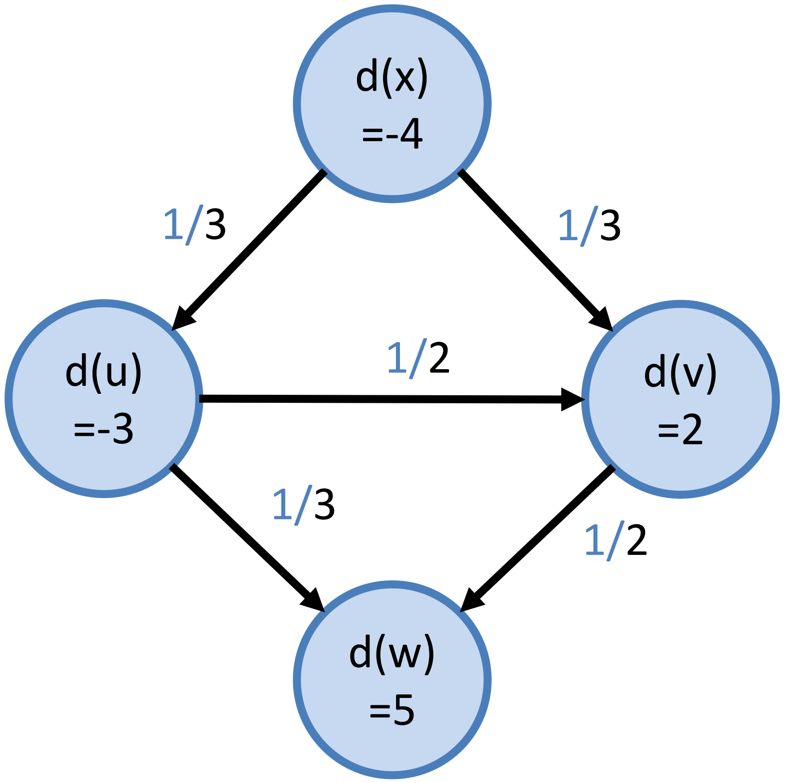


Questions?

Circulation with Demands and Lower Bound

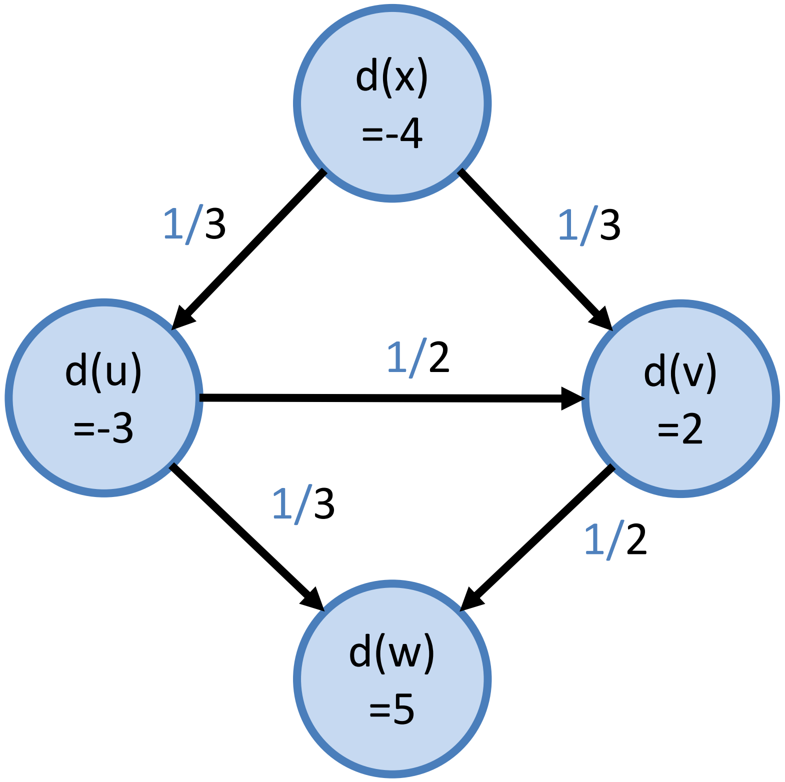
A Feasibility Problem…

- We will need to make some transformation…



Circulation with Demands and Lower Bound
A Feasibility Problem…

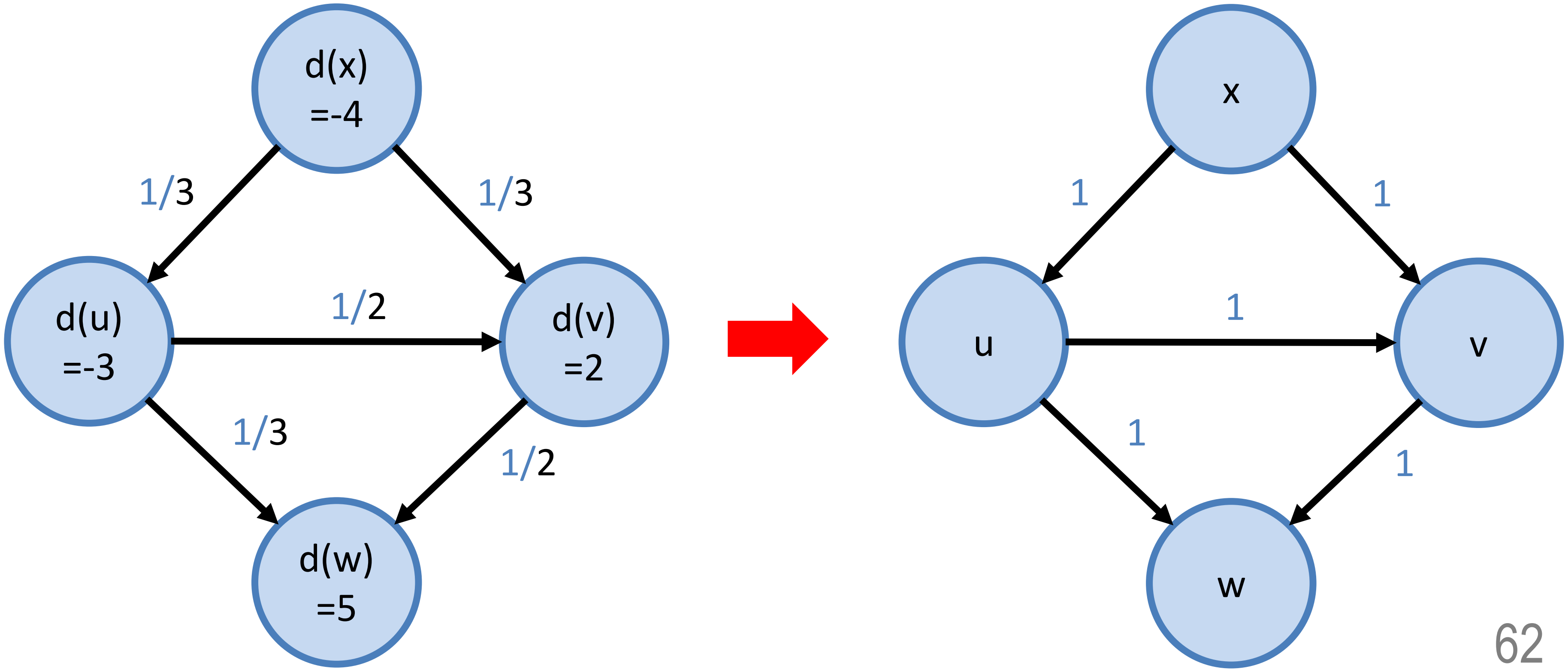
- We will need to make some transformation…



Circulation with Demands and Lower Bound

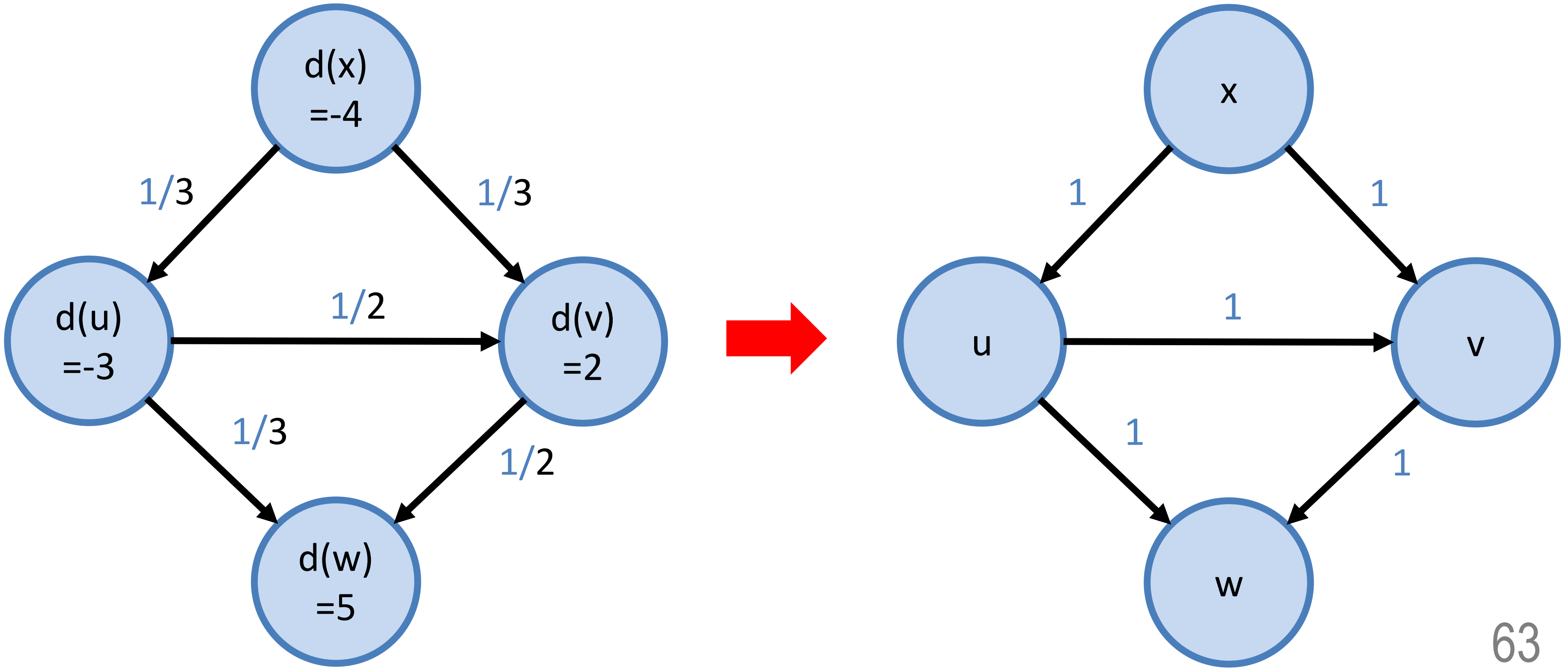
A Feasibility Problem…

- We will need to make some transformation by removing the lower bound to a temp network…



Circulation with Demands and Lower Bound
A Feasibility Problem…

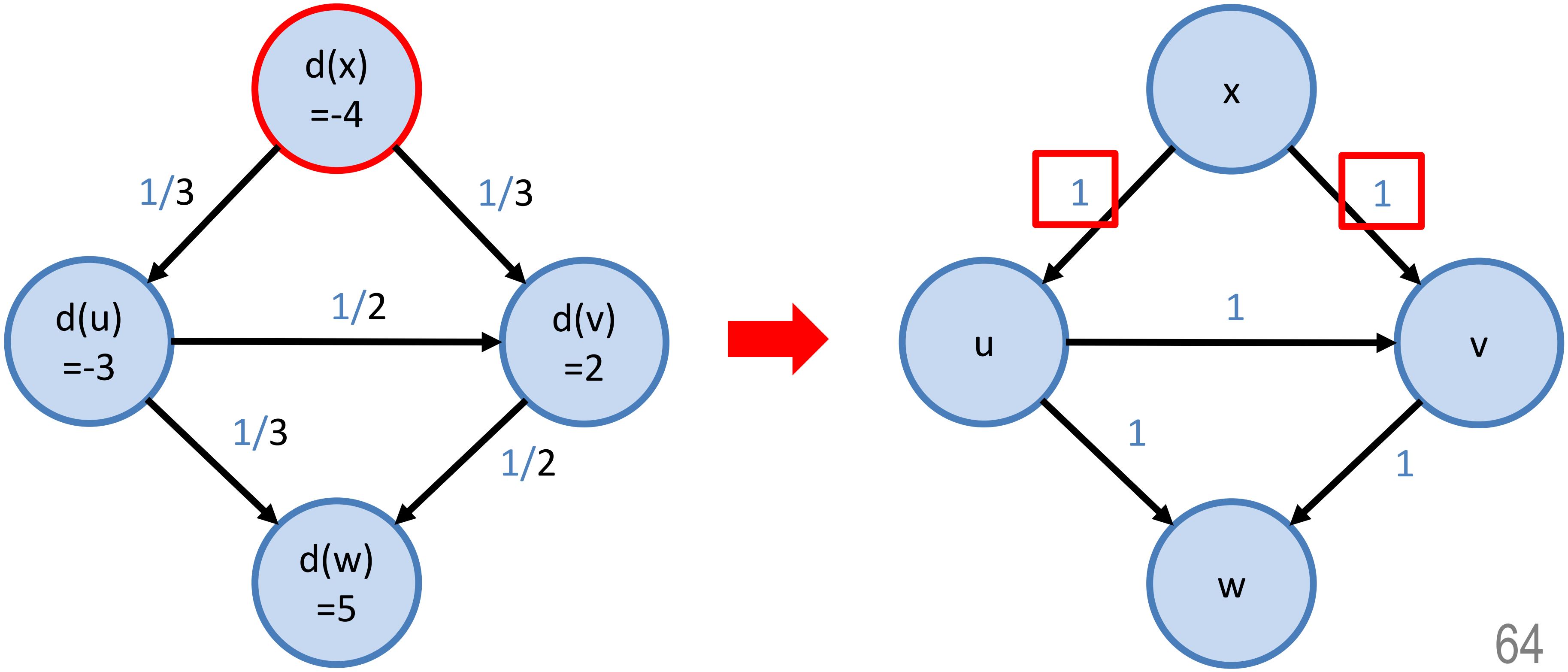
- We will need to make some transformation by removing the lower bound to a temp network…
- Thus, the original reduced…



Circulation with Demands and Lower Bound

A Feasibility Problem…

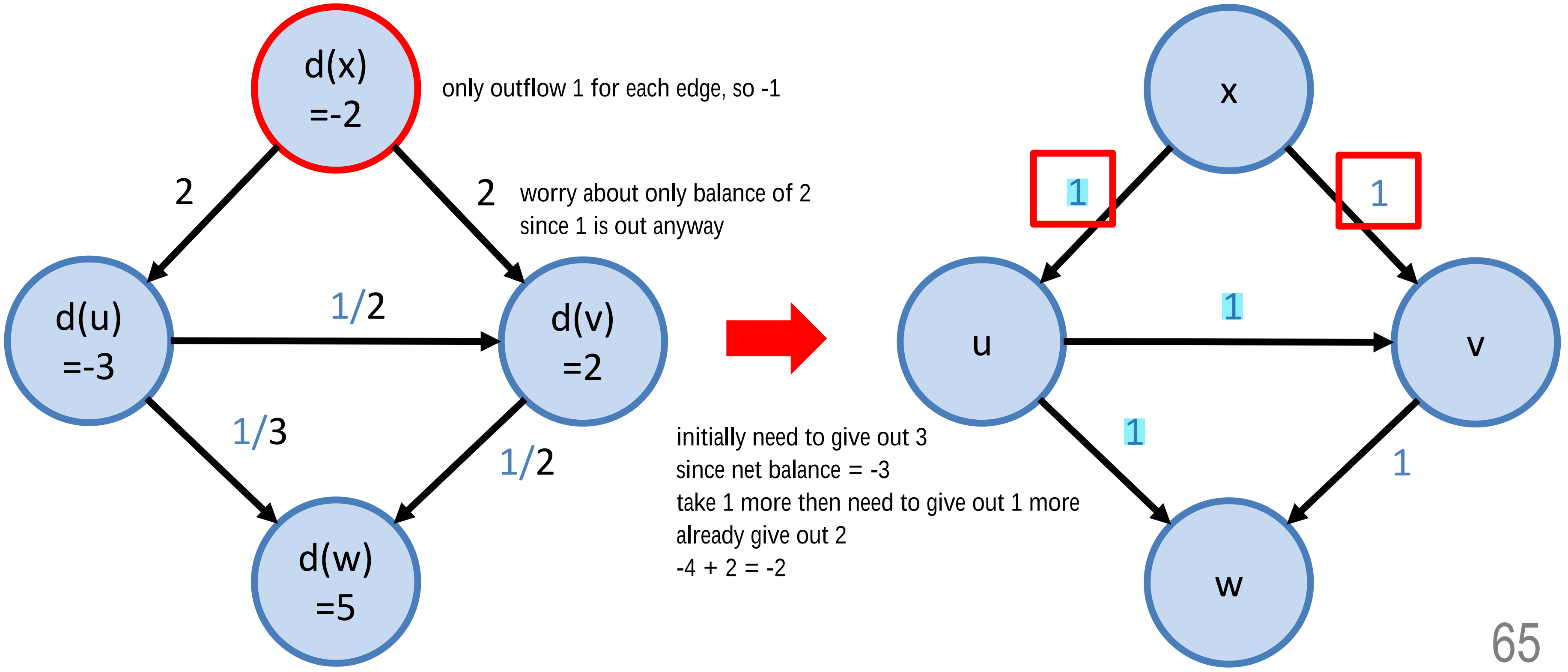
- We will need to make some transformation by removing the lower bound to a temp network…
- Thus, the original reduced…



Circulation with Demands and Lower Bound

A Feasibility Problem…

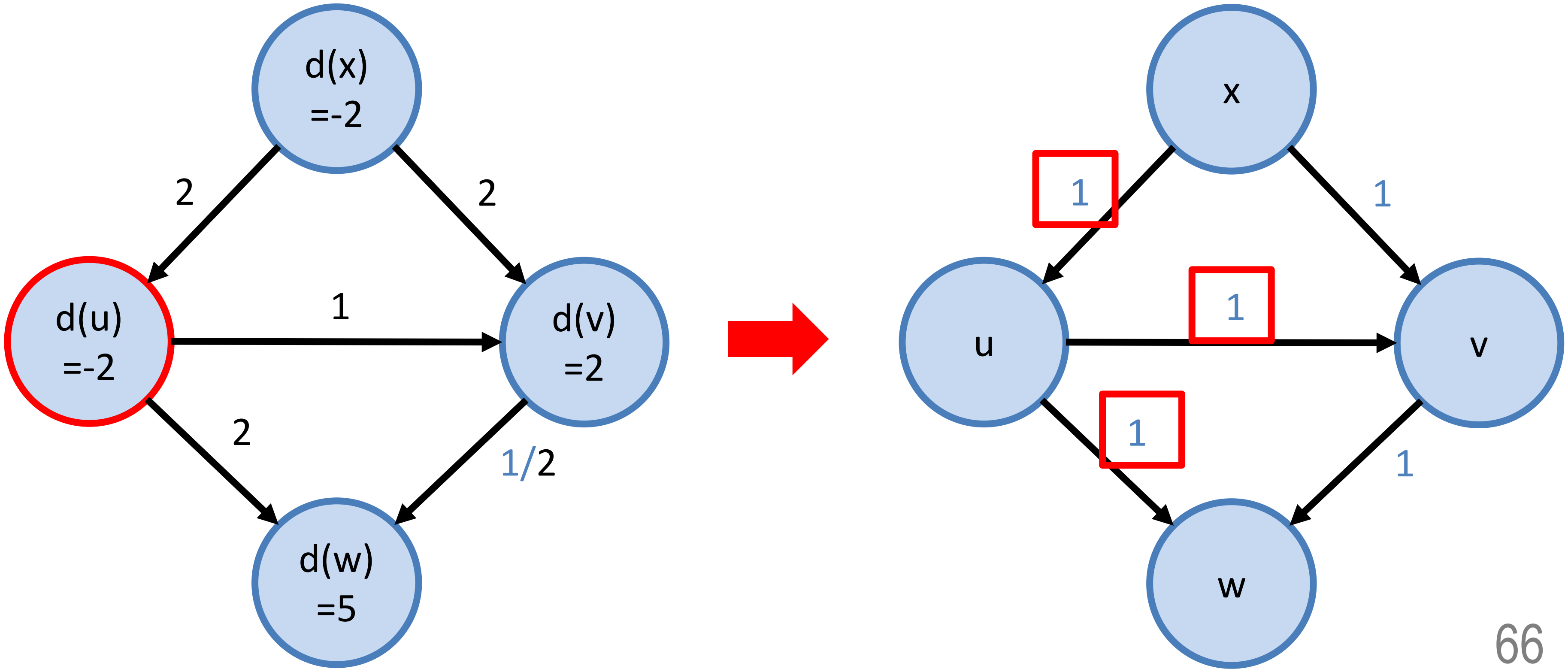
- We will need to make some transformation by removing the lower bound to a temp network…
- Thus, the original reduced…


d(x) =-2
only outflow 1 for each edge, so -1
2
2
worry about only balance of 2 since 1 is out anyway
d(u) =-3
1/2
d(v) =2
1/3
1/2
d(w) =5
initially need to give out 3
since net balance = -3
take 1 more then need to give out 1 more
already give out 2
-4 + 2 = -2
x
1
1
u
1
v
1
1
w

Circulation with Demands and Lower Bound

A Feasibility Problem…

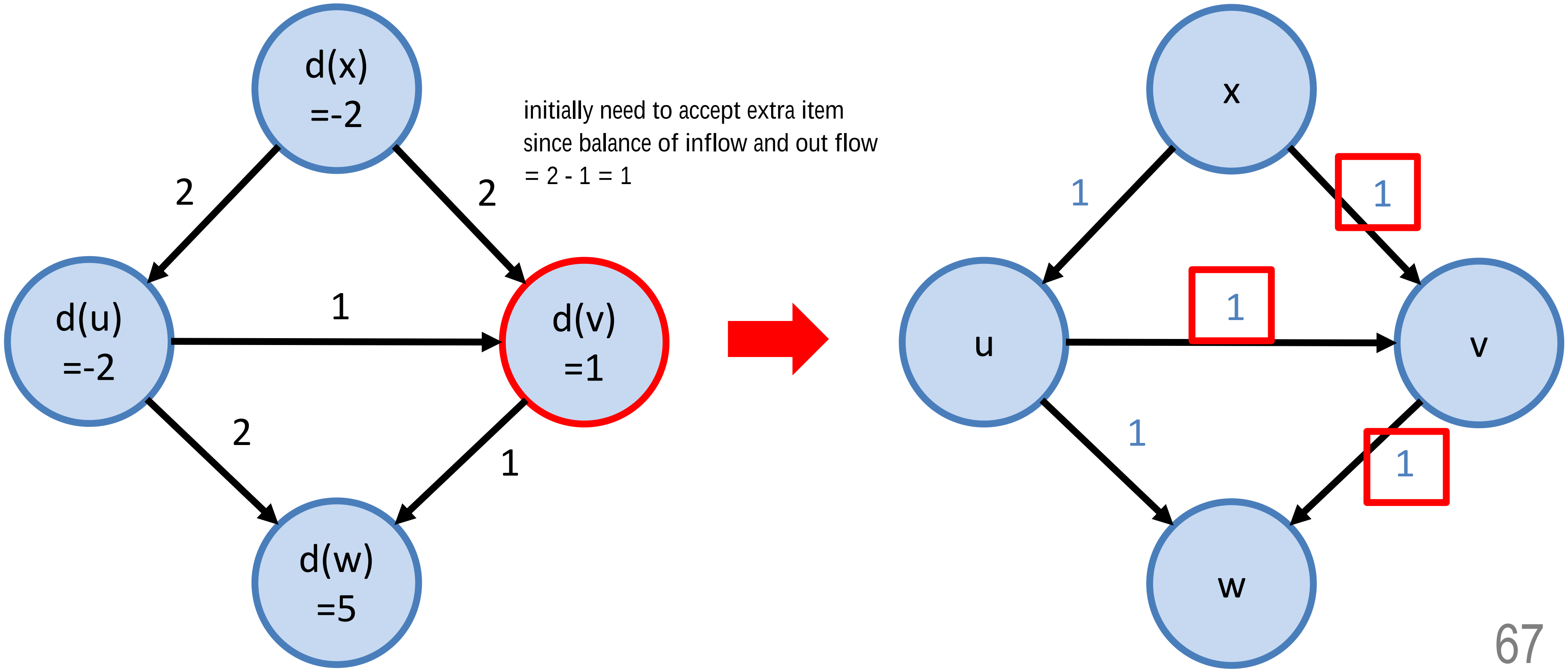
- We will need to make some transformation by removing the lower bound to a temp network…
- Thus, the original reduced…



Circulation with Demands and Lower Bound

A Feasibility Problem…

- We will need to make some transformation by removing the lower bound to a temp network…
- Thus, the original reduced…

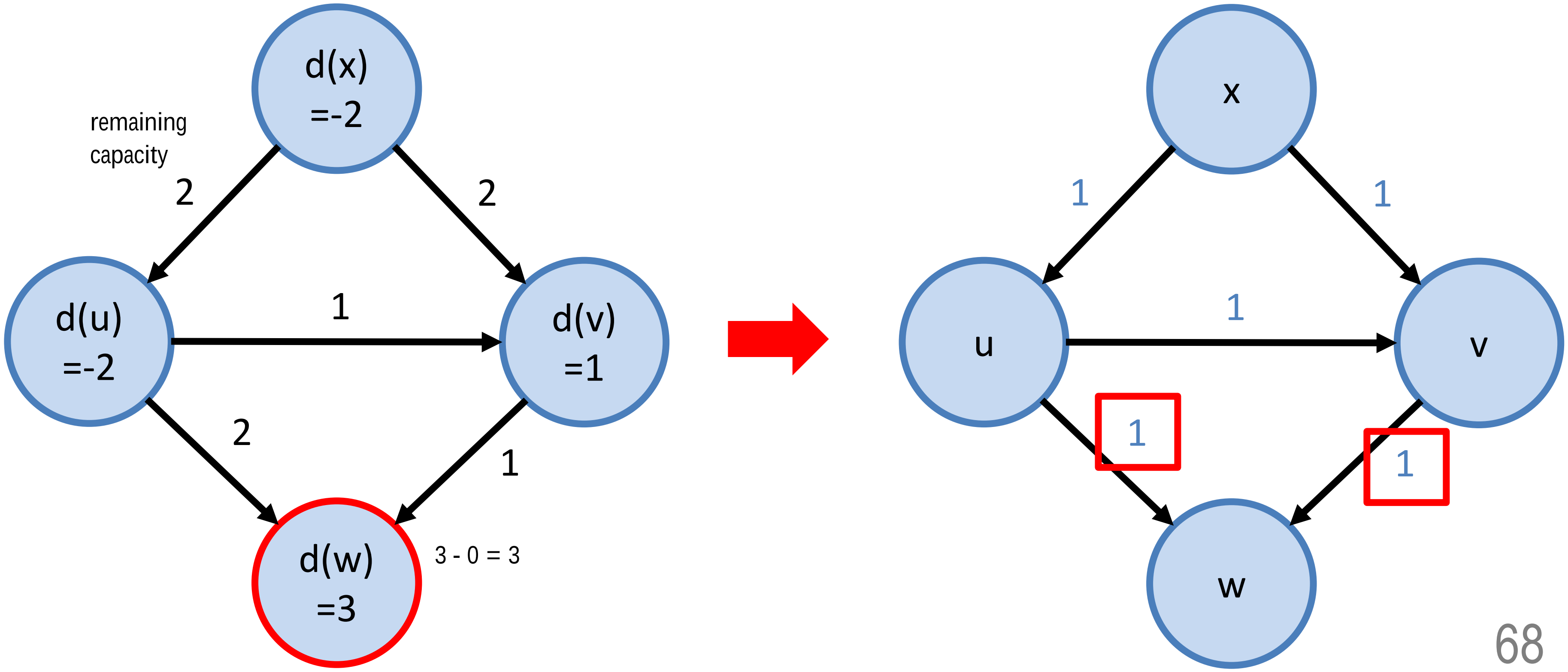


d(x) =-2
2
2
d(u) =-2
1
d(v) =1
2
1
d(w) =5
initially need to accept extra item since balance of inflow and out flow = 2 - 1 = 1
x
1
1
u
1
v
1
1
w

Circulation with Demands and Lower Bound

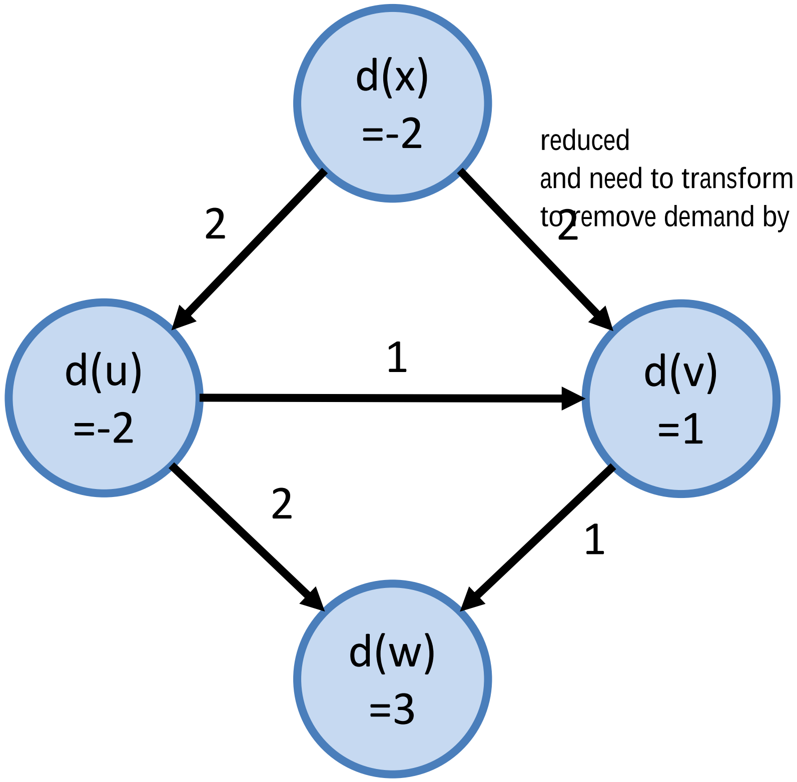
A Feasibility Problem…

- We will need to make some transformation by removing the lower bound to a temp network…
- Thus, the original reduced…



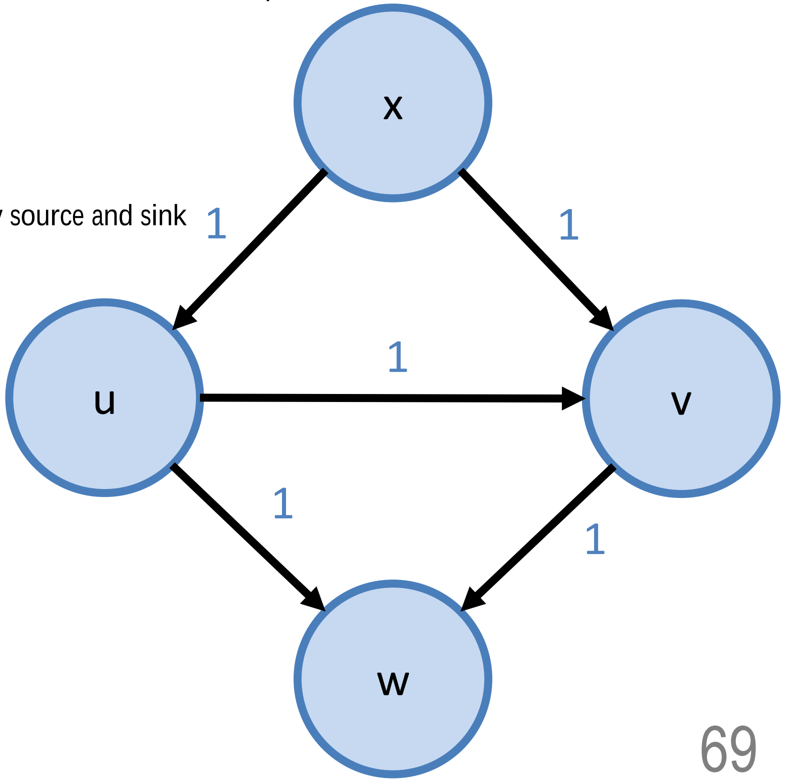
Circulation with Demands and Lower Bound
A Feasibility Problem…

- Thus, the original reduced…



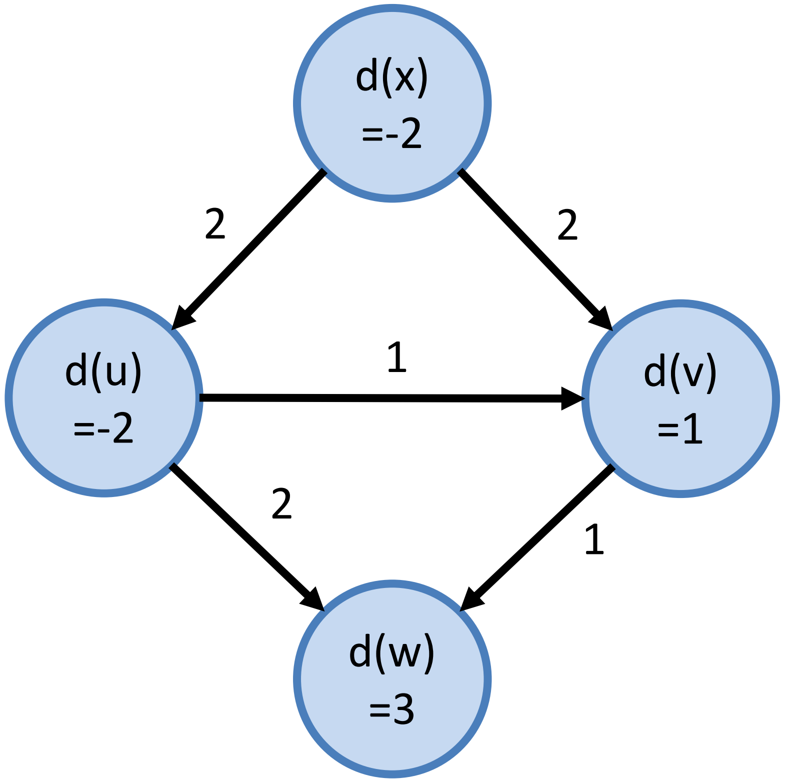
reduced
and need to transform into flow network
to remove demand by introducing imaginary source and sink

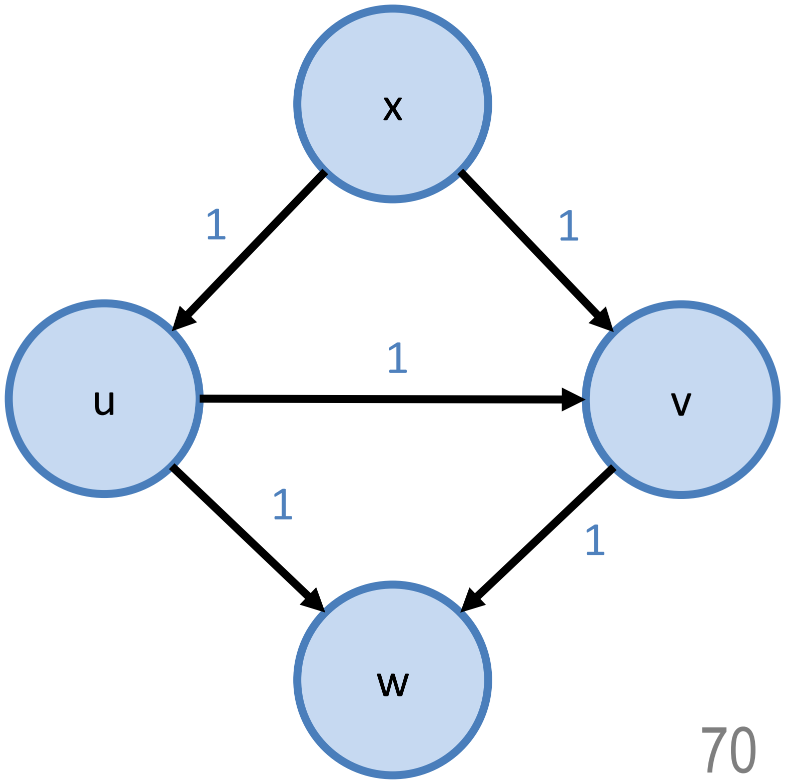
put this aside
at this point



Circulation with Demands and Lower Bound
A Feasibility Problem…

- Thus, the original reduced…





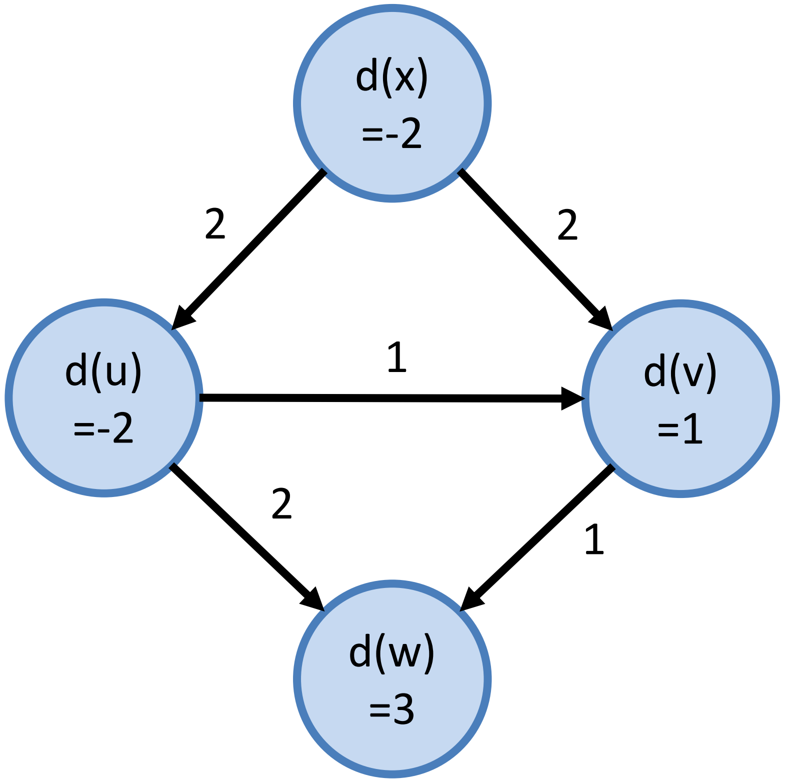
MONASH University

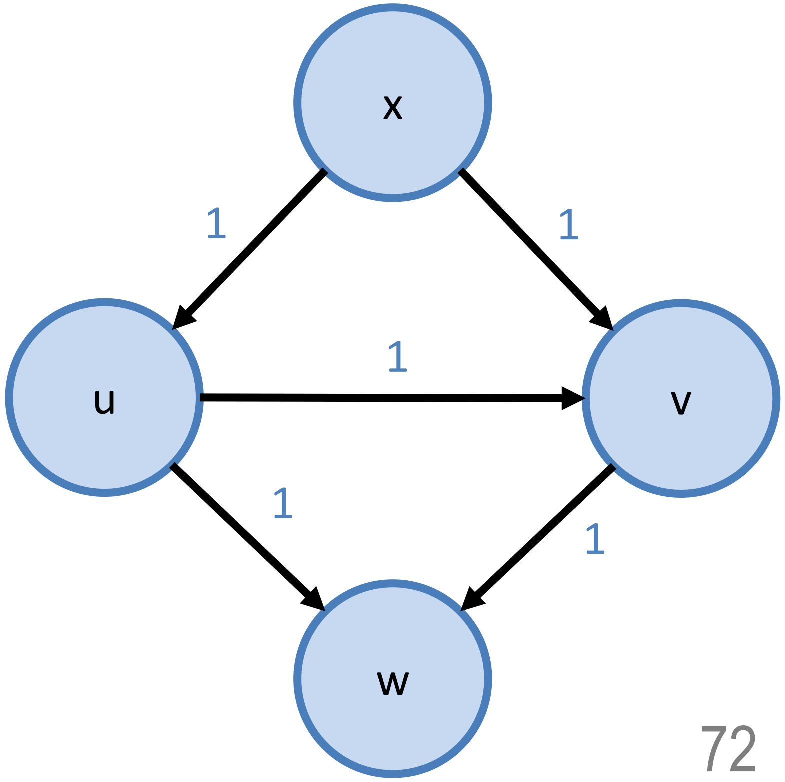
Questions?

Circulation with Demands and Lower Bound

A Feasibility Problem…

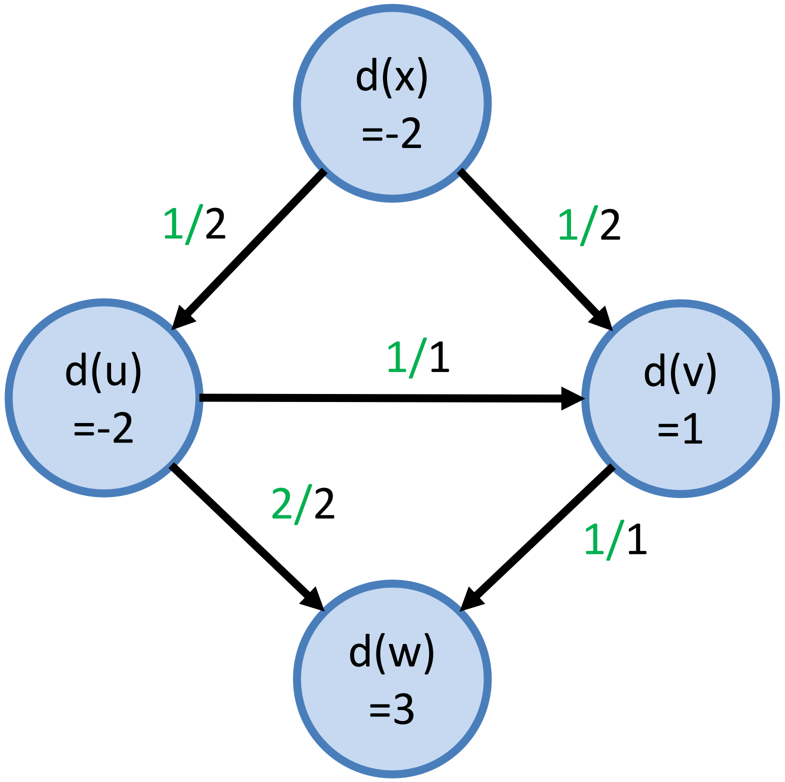
- Then we follow the same Circulation with Demands as earlier for the reduced network…

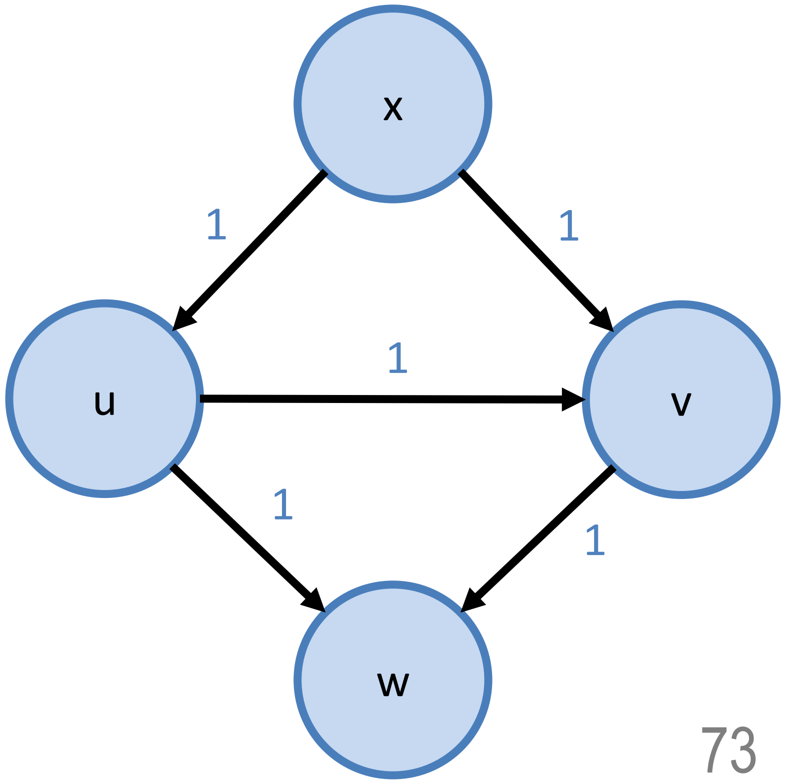




Circulation with Demands and Lower Bound
A Feasibility Problem…

- Then we follow the same Circulation with Demands as earlier for the reduced network…

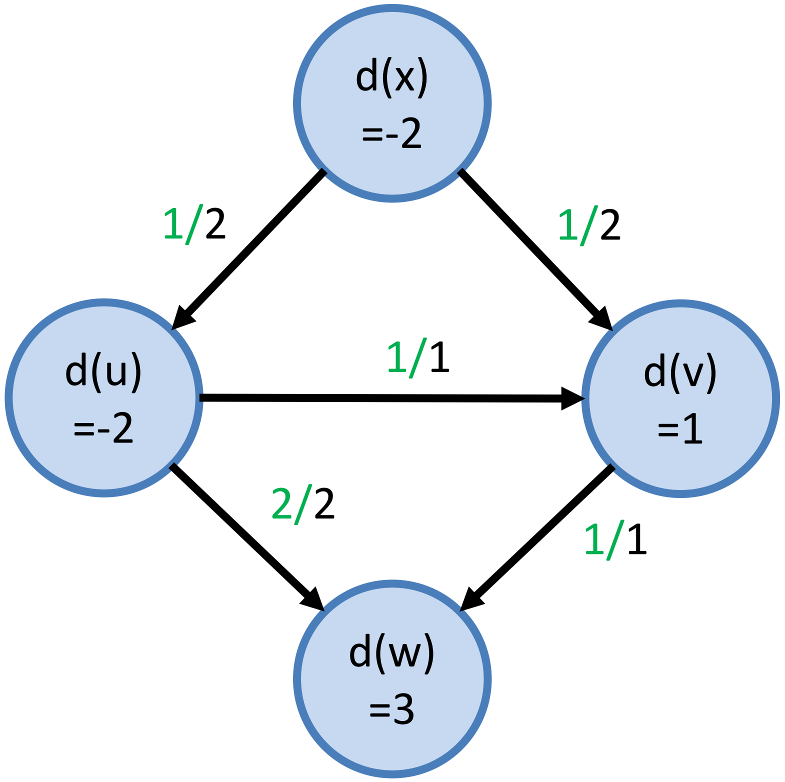


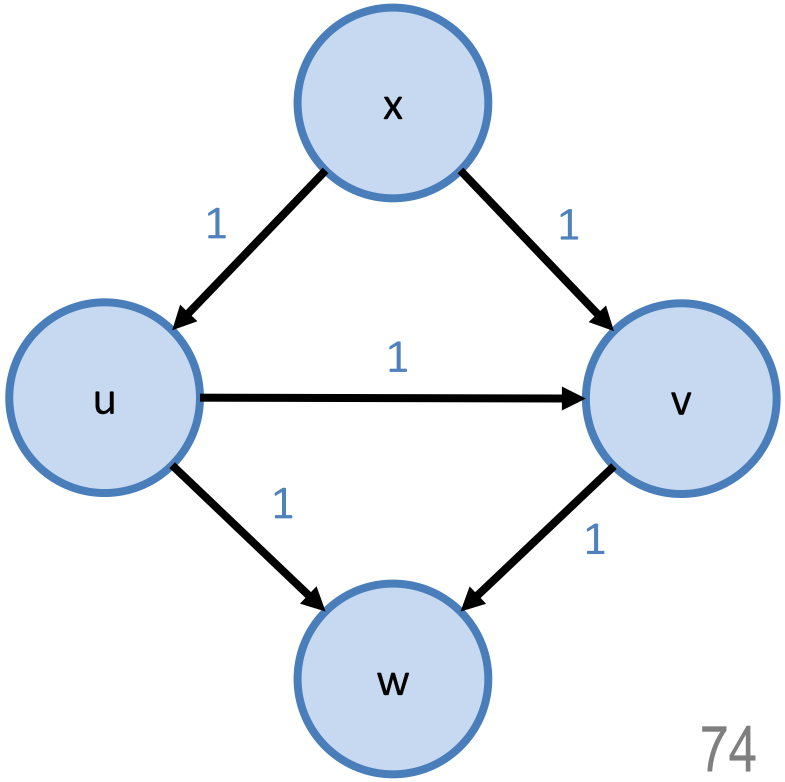


Circulation with Demands and Lower Bound

A Feasibility Problem…

- Then we follow the same Circulation with Demands as earlier for the reduced network… It is feasible!

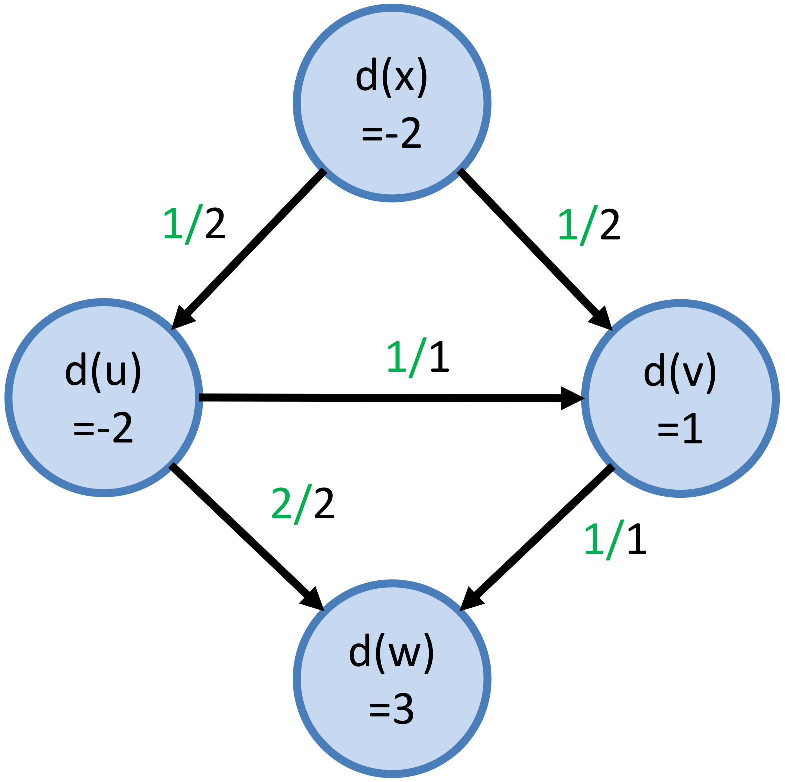


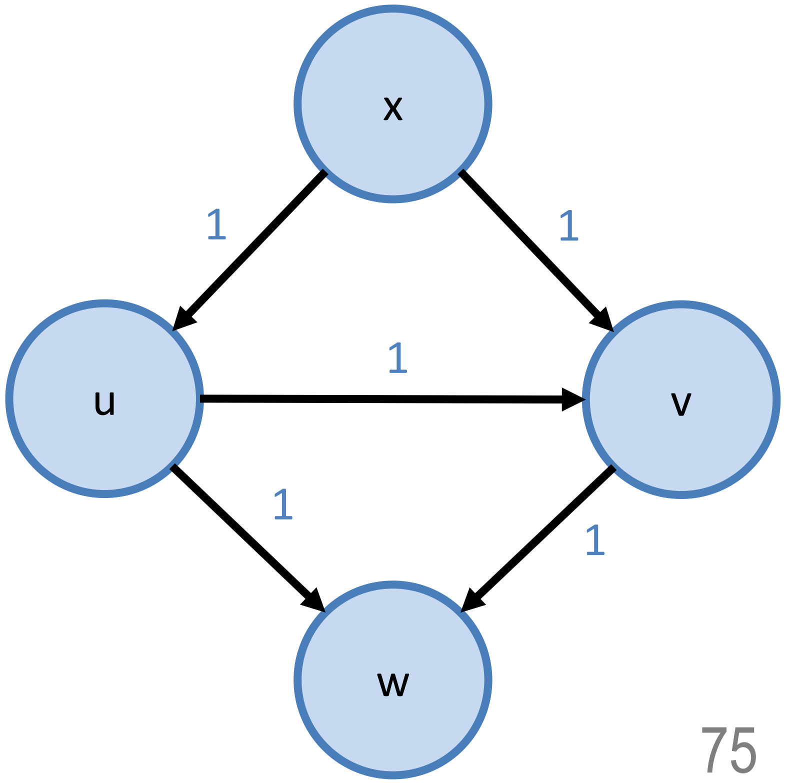


Circulation with Demands and Lower Bound

A Feasibility Problem…

- We finally combine them back but with the additional information using constraint/flow/capacity

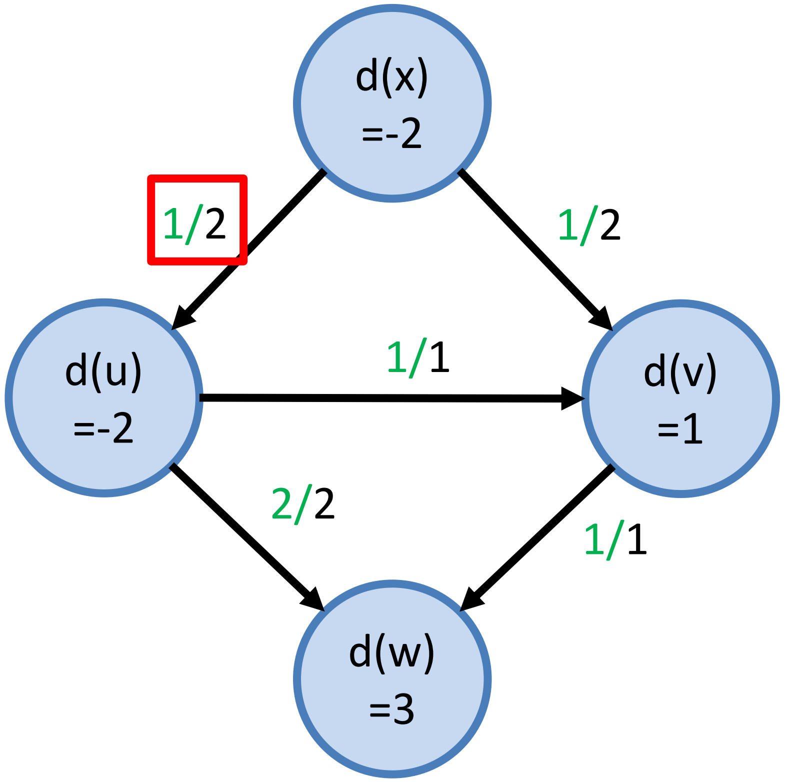


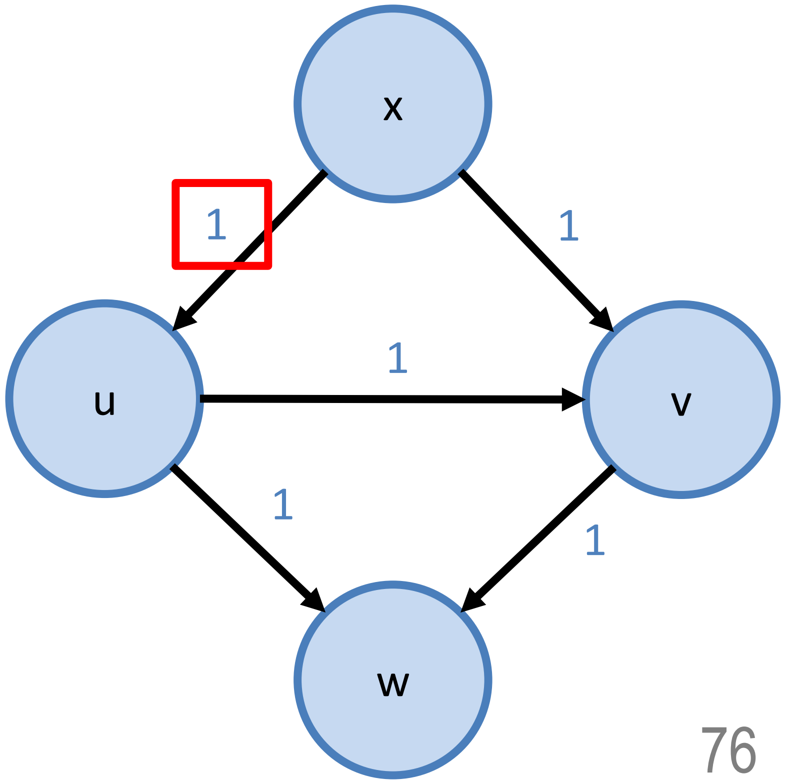


Circulation with Demands and Lower Bound

A Feasibility Problem…

- We finally combine them back but with the additional information using constraint/flow/capacity

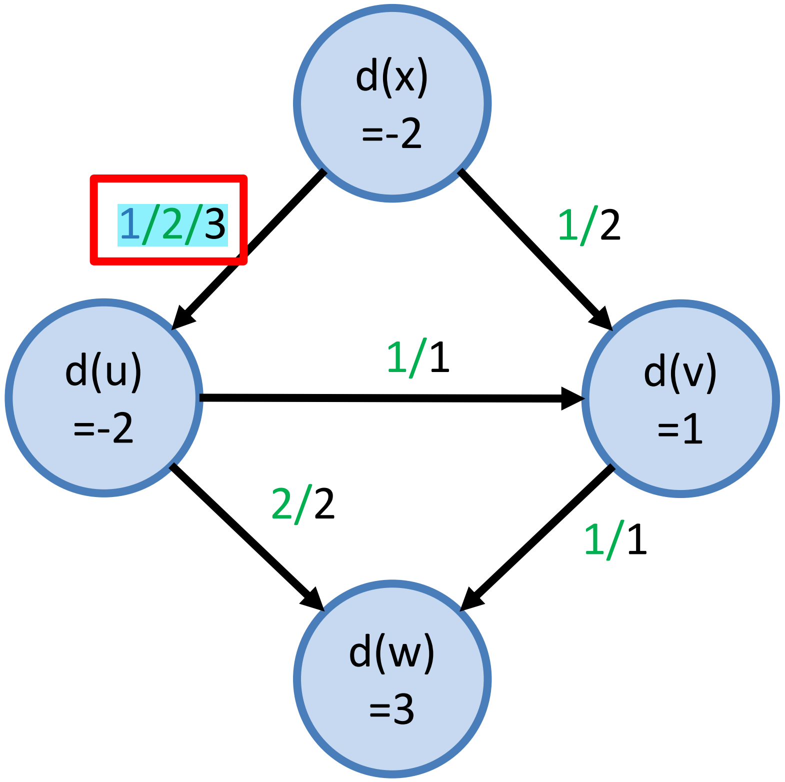


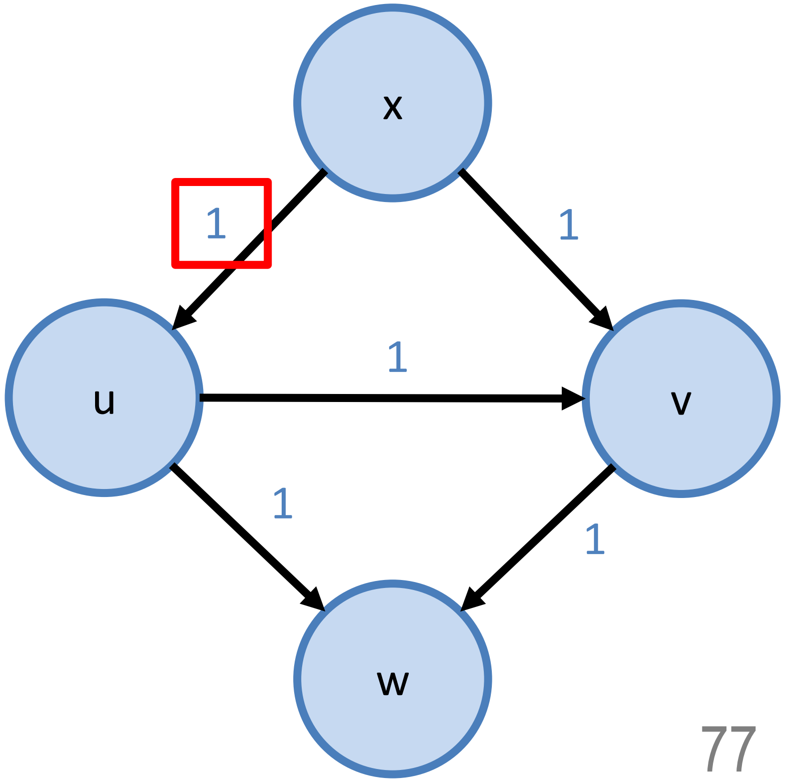


Circulation with Demands and Lower Bound

A Feasibility Problem…

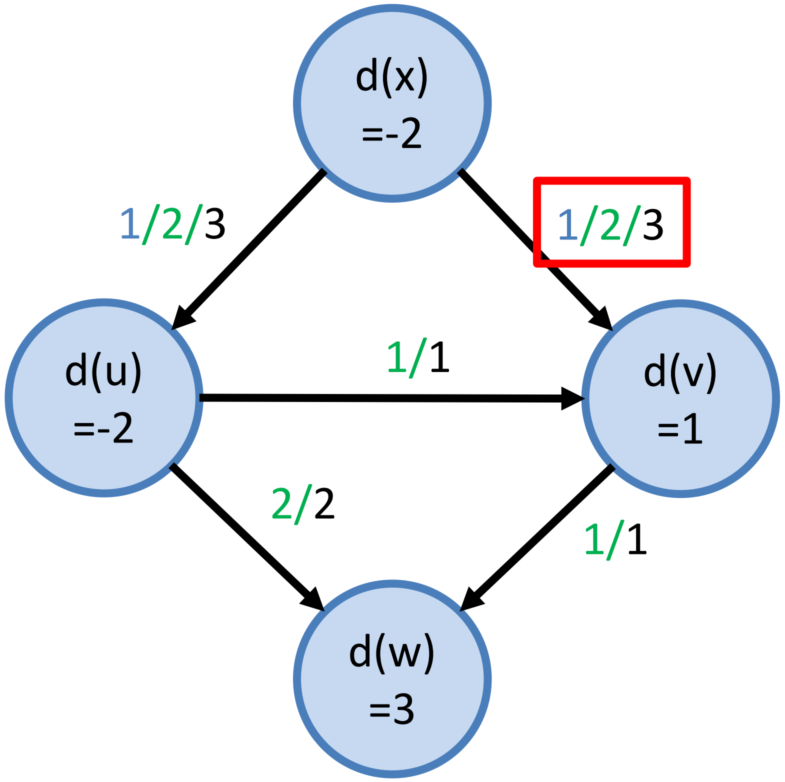
- We finally combine them back but with the additional information using constraint/flow/capacity

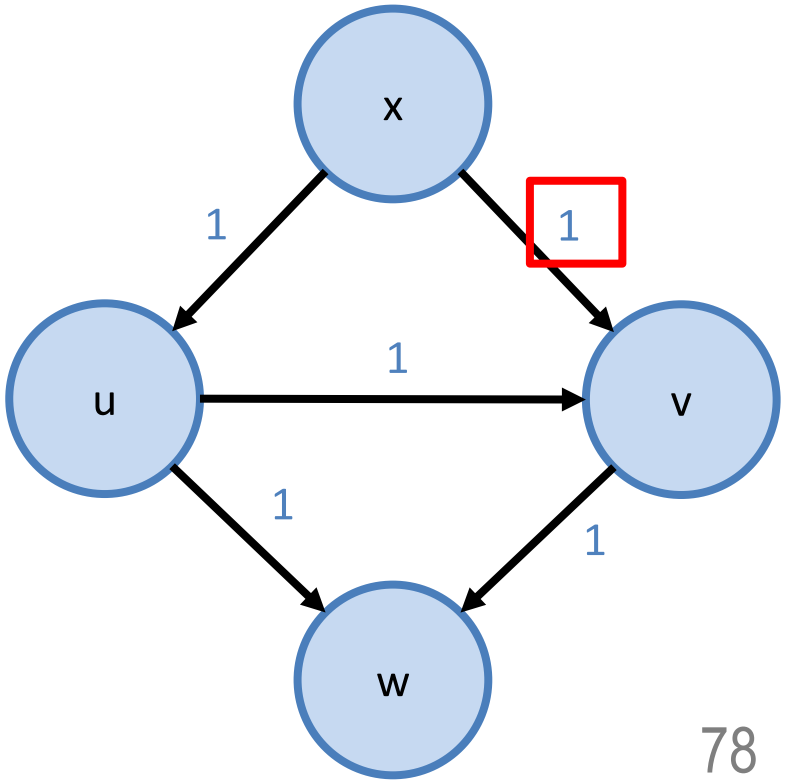




Circulation with Demands and Lower Bound
A Feasibility Problem…

- We finally combine them back but with the additional information using constraint/flow/capacity

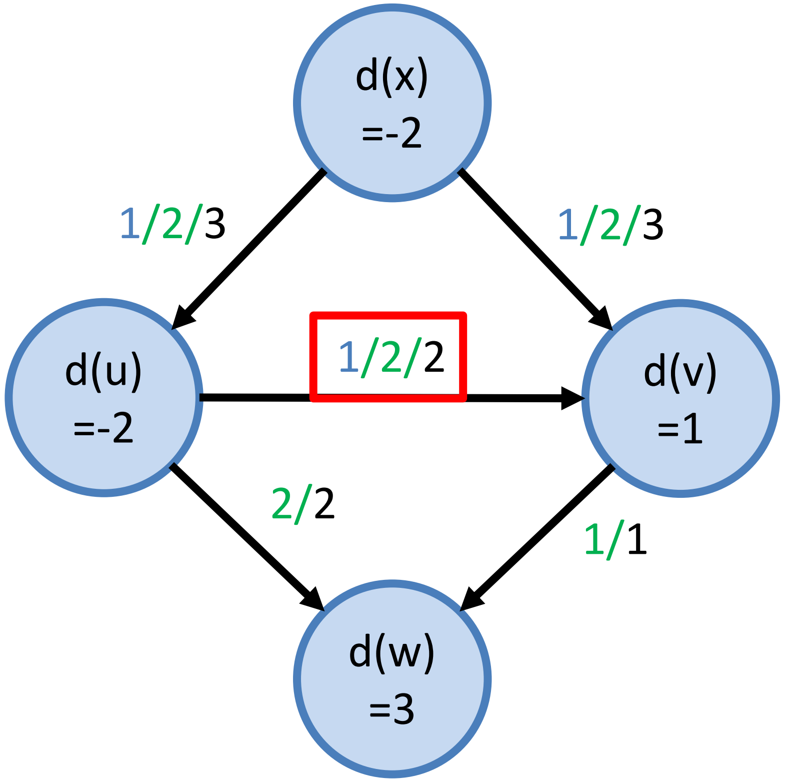


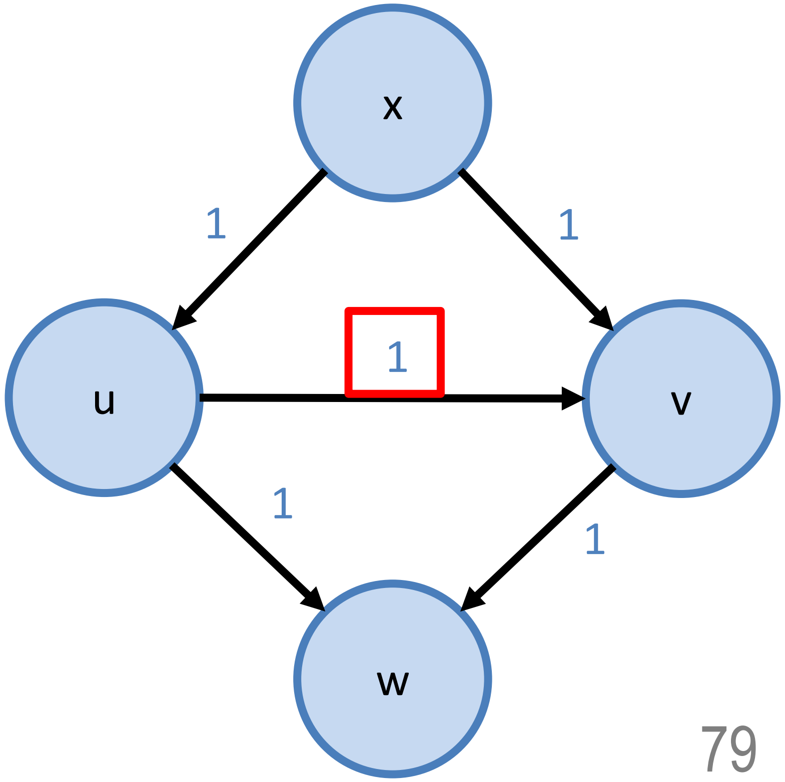


Circulation with Demands and Lower Bound

A Feasibility Problem…

- We finally combine them back but with the additional information using constraint/flow/capacity

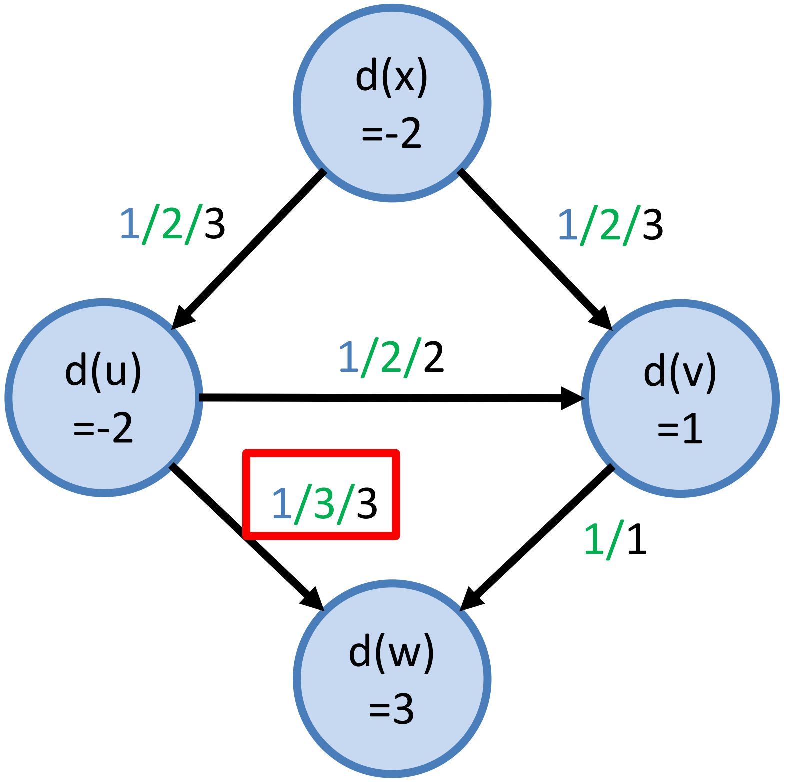


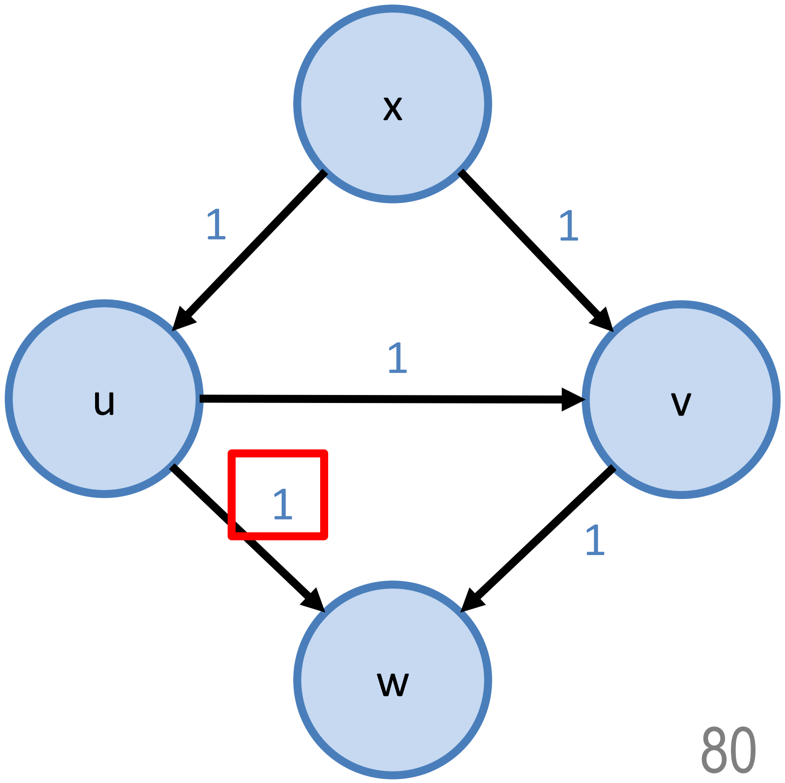


Circulation with Demands and Lower Bound

A Feasibility Problem…

- We finally combine them back but with the additional information using constraint/flow/capacity

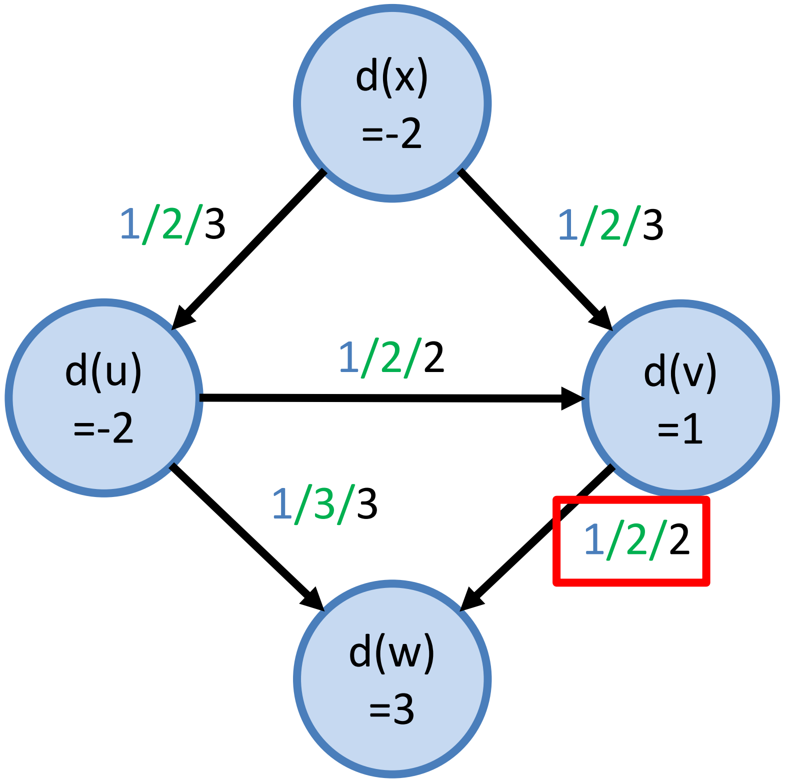


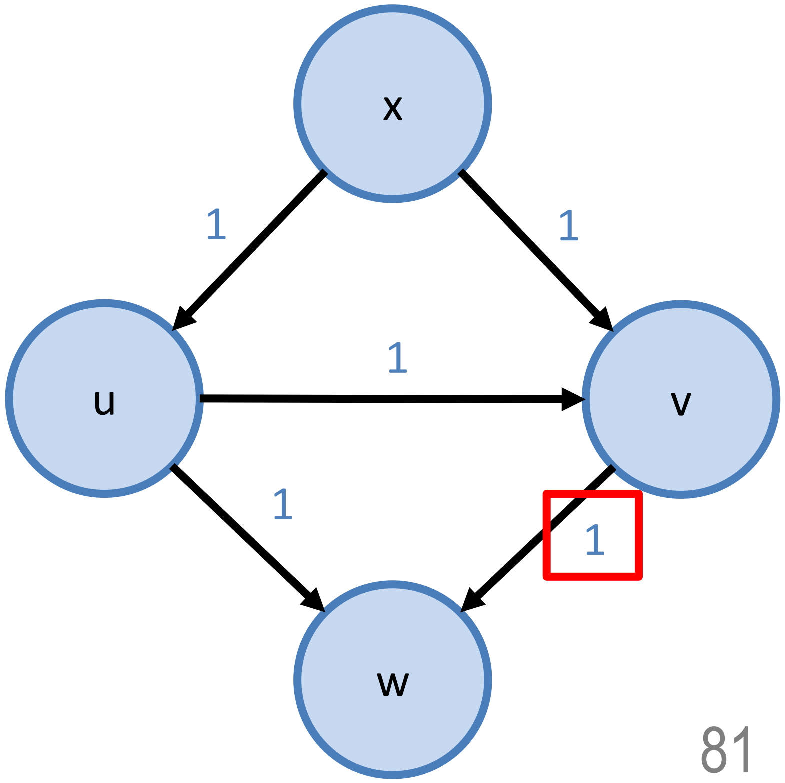


Circulation with Demands and Lower Bound

A Feasibility Problem…

- We finally combine them back but with the additional information using constraint/flow/capacity

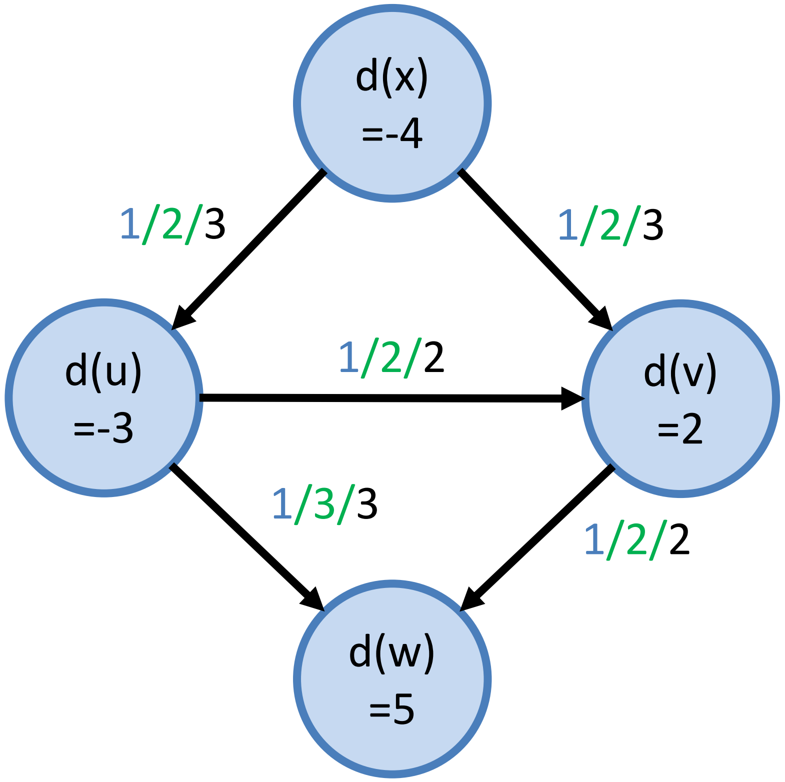


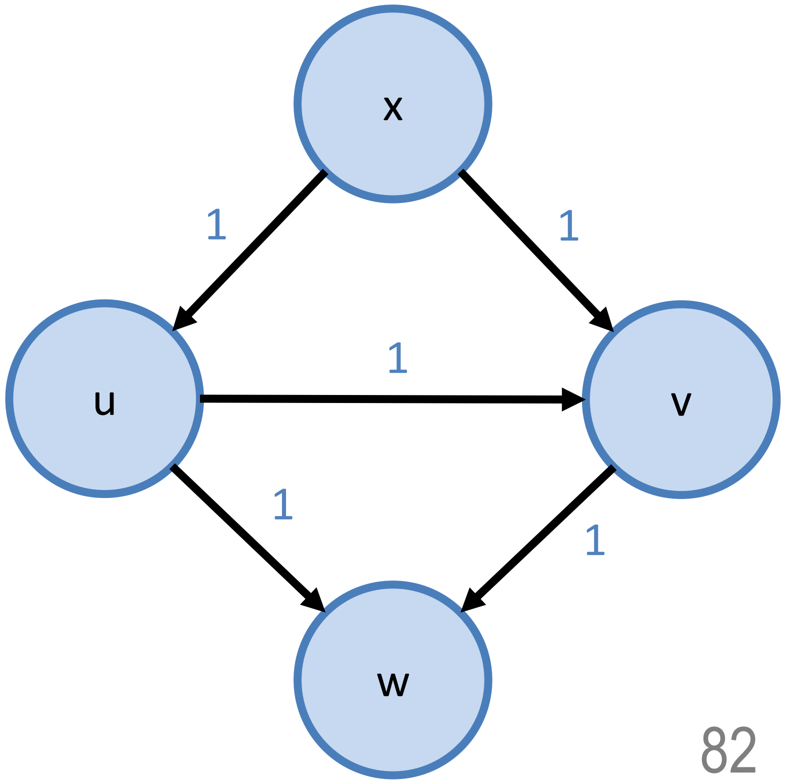


Circulation with Demands and Lower Bound

A Feasibility Problem…

- Don't forget the demand of the vertices as well!

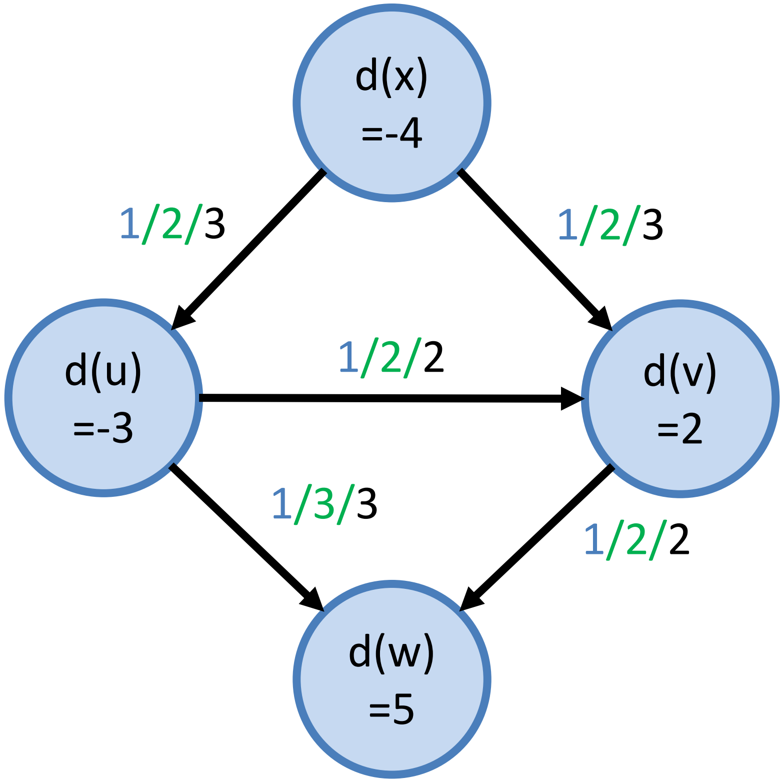




Circulation with Demands and Lower Bound

A Feasibility Problem…

- And we are done!

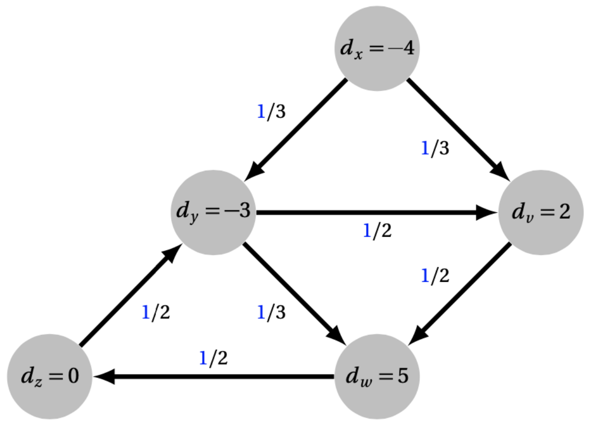


Questions?

Circulation with Demands and Lower Bound

A Feasibility Problem…

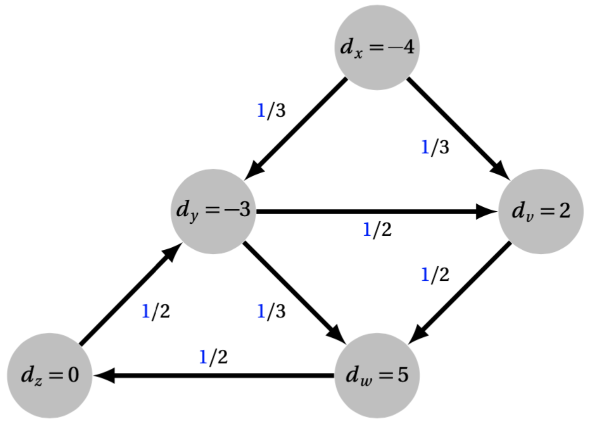
- The following is an example from the Clayton campus



Circulation with Demands and Lower Bound

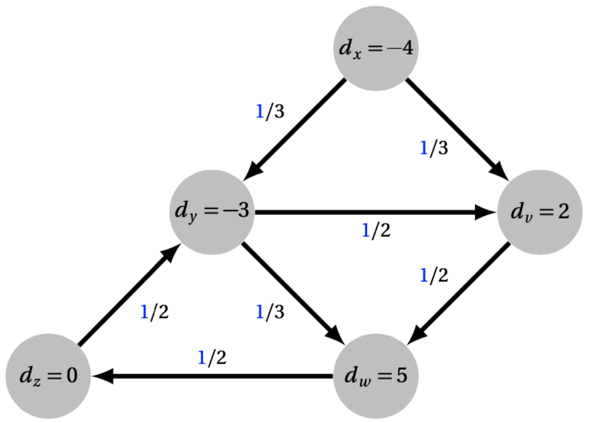
A Feasibility Problem…

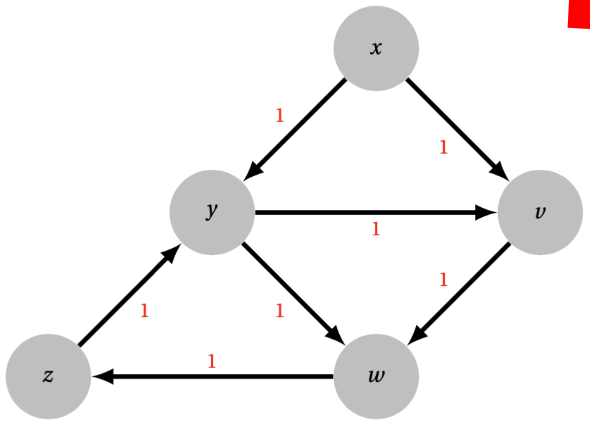
- The following is an example from the Clayton campus
- Work it out on your own to see if it is feasible before we discuss in class



Circulation with Demands and Lower Bound
A Feasibility Problem…

- The following is an example from the Clayton campus
- Work it out on your own to see if it is feasible before we discuss in class

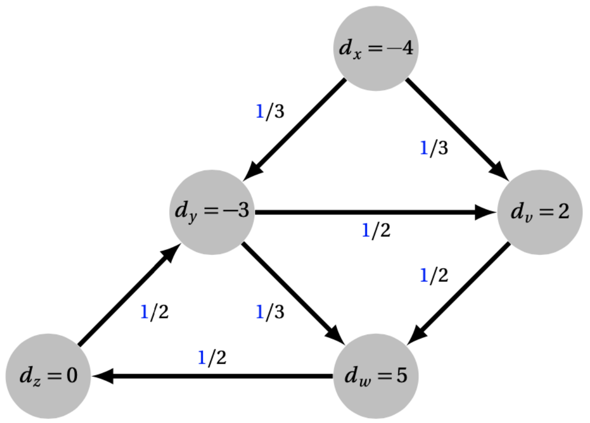


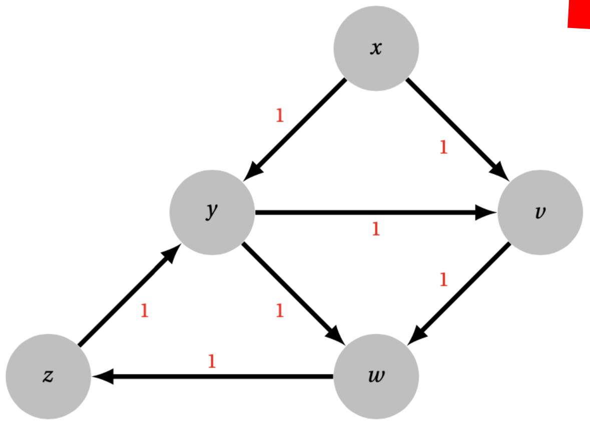


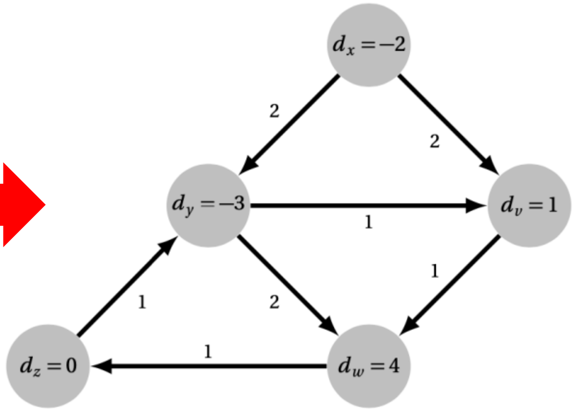
Circulation with Demands and Lower Bound

A Feasibility Problem…

- The following is an example from the Clayton campus
- Work it out on your own to see if it is feasible before we discuss in class.



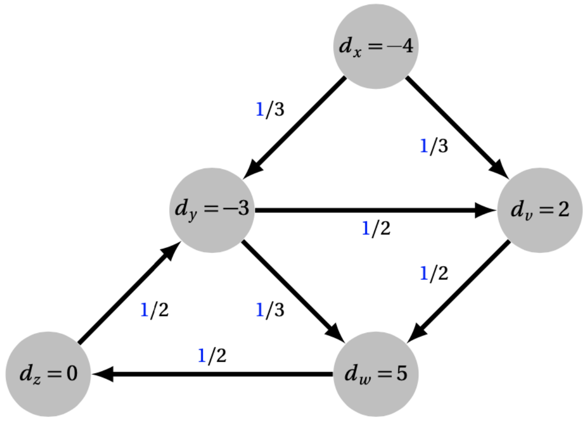


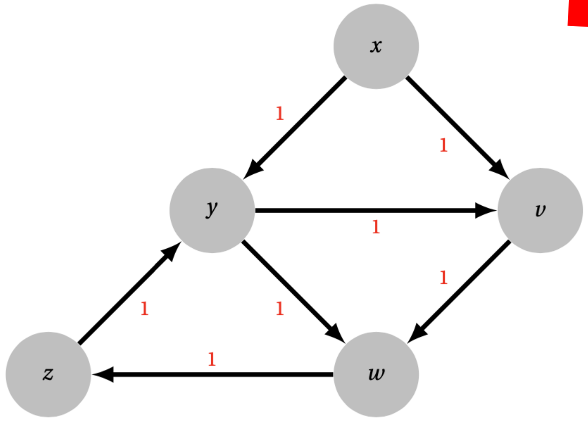


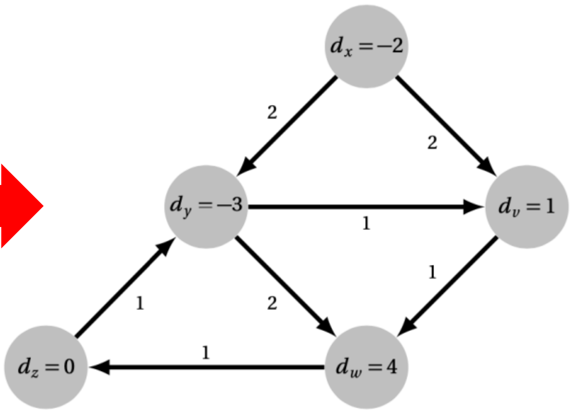
Circulation with Demands and Lower Bound

A Feasibility Problem…

- The following is an example from the Clayton campus
- Work it out on your own to see if it is feasible before we discuss in class. It is it not!

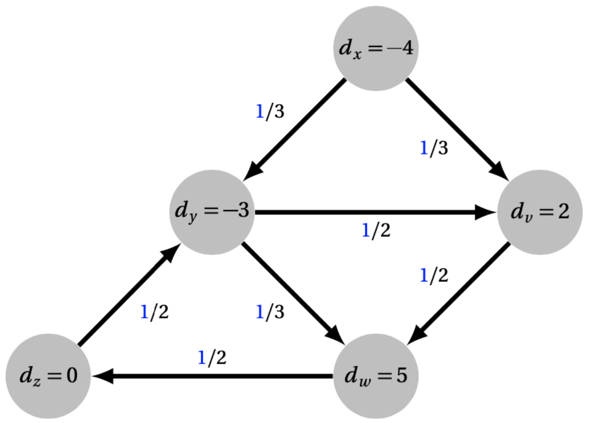


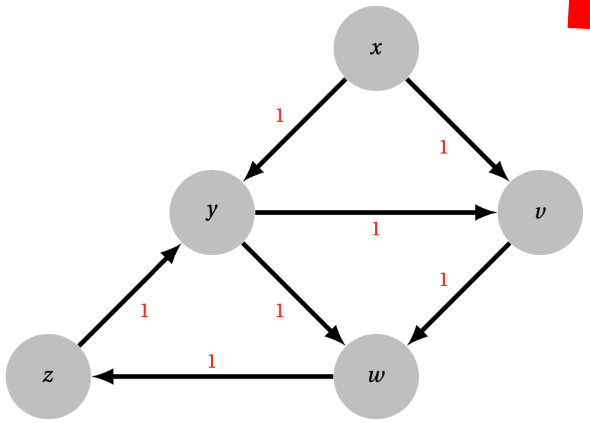


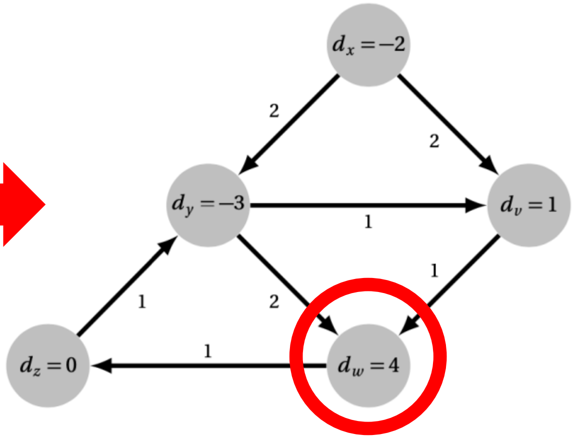


Circulation with Demands and Lower Bound
A Feasibility Problem…

- The following is an example from the Clayton campus
- Work it out on your own to see if it is feasible before we discuss in class. It is it not!







Questions?

Applications with Network Flow

For Real-World Problems

- We know what is a flow network.
- We know how to design flow network for bipartite matching.
- We know how to design flow network for circulation with demands and lower bound?

Applications with Network Flow

For Real-World Problems

- We know what is a flow network.
- We know how to design flow network for bipartite matching.
- We know how to design flow network for circulation with demands and lower bound? We shall see them now…

Applications with Network Flow

For Real-World Problems

MONASH University

- We know what is a flow network.
- We know how to design flow network for bipartite matching.
- We know how to design flow network for circulation with demands and lower bound? We shall see them now…

Applications with Network Flow
For Real-World Problems

- We know what is a flow network.
- We know how to design flow network for bipartite matching.
- We know how to design flow network for circulation with demands and lower bound? We shall see them now…

- Not that we only deal with integers, to make it simpler

Questions?

Applications with Network Flow

Survey Design

Applications with Network Flow

Survey Design

- You have C customers who have used the product
- You have P products

Applications with Network Flow
Survey Design

- You have C customers who have used the product
- You have P products

- You want to conduct a survey, but…

Applications with Network Flow
Survey Design

- You have C customers who have used the product
- You have P products

- You want to conduct a survey, but…
  – You do not want to ask the customer for too many reviews.
  – You do want to ask the customer for at least some reviews.
  – Each product needs to have at least some reviews.
  – Each product do not require more than some reviews.

Applications with Network Flow
Survey Design

- You have C customers who have used the product
- You have P products

- You want to conduct a survey, but…
  - You do not want to ask the customer for too many reviews. capacity
  - You do want to ask the customer for at least some reviews. lower bound
  - Each product needs to have at least some reviews. lower bound
  - Each product do not require more than some reviews.capacity
  - Of course, each customer can only give a review per product.

Applications with Network Flow

Survey Design

- You have C customers who have used the product
- You have P products

- You want to conduct a survey, but…
  - You do not want to ask the customer for too many reviews.
  - You do want to ask the customer for at least some reviews.
  - Each product needs to have at least some reviews.
  - Each product do not require more than some reviews.
  - Of course, each customer can only give a review per product.

- Let us add some notations

Applications with Network Flow

Survey Design

- You have C customers who have used the product , $c_1, c_2, c_3, \ldots, c_n$
- You have P products, $p_1, p_2, p_3, \ldots, p_m$

- You want to conduct a survey, but…
  - You do not want to ask the customer c_i for too many reviews c_i^+.
  - You do want to ask the customer c_i for at least some reviews c_i^-.
  - Each product p_j needs to have at least some reviews p_j^-.
  - Each product p_j do not require more than some reviews p_j^+.
  - Of course, each customer can only give a review per product.

- Let us add some notations

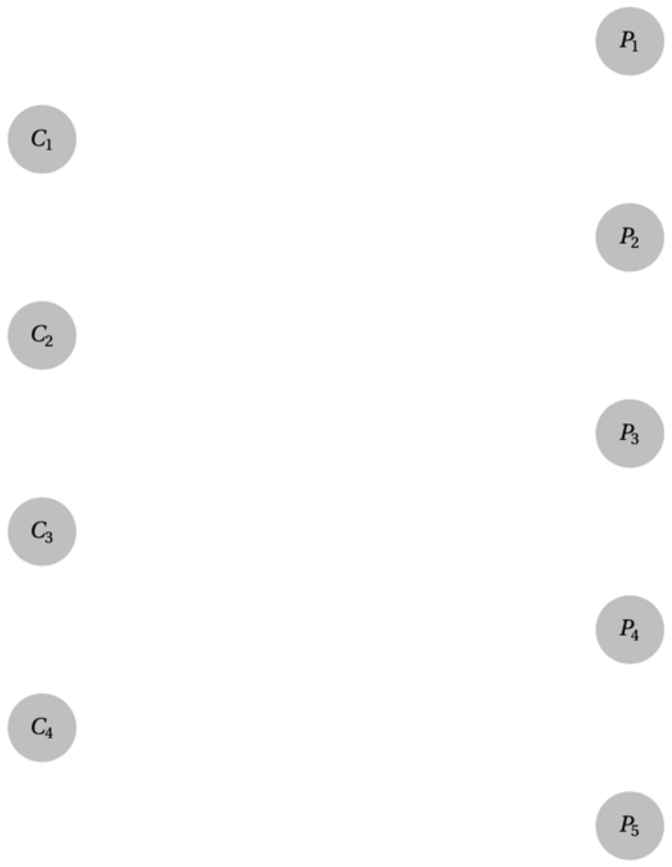
Applications with Network Flow
Survey Design

- Now let us go through 1 by 1

Applications with Network Flow

Survey Design

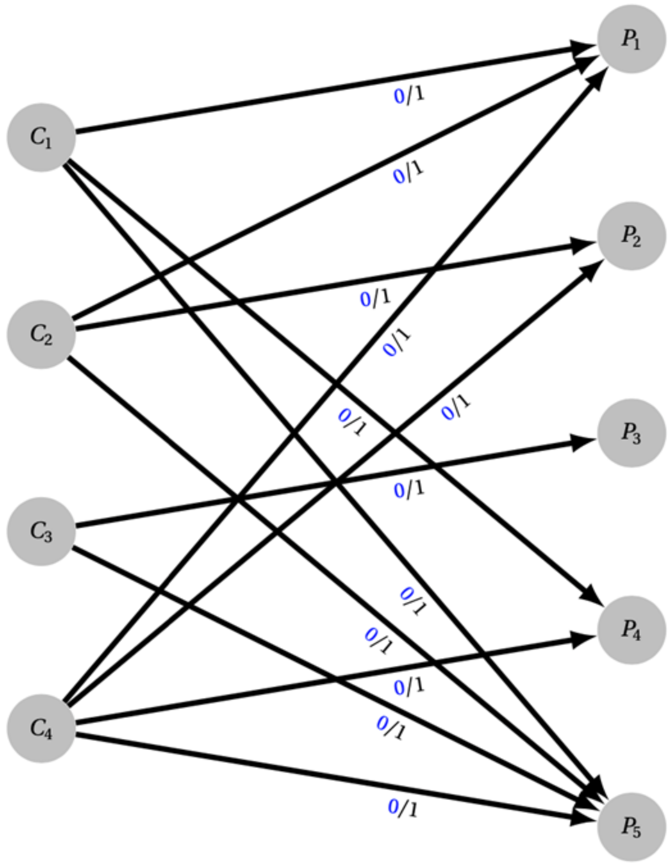
- The customer and the product.



Applications with Network Flow

Survey Design

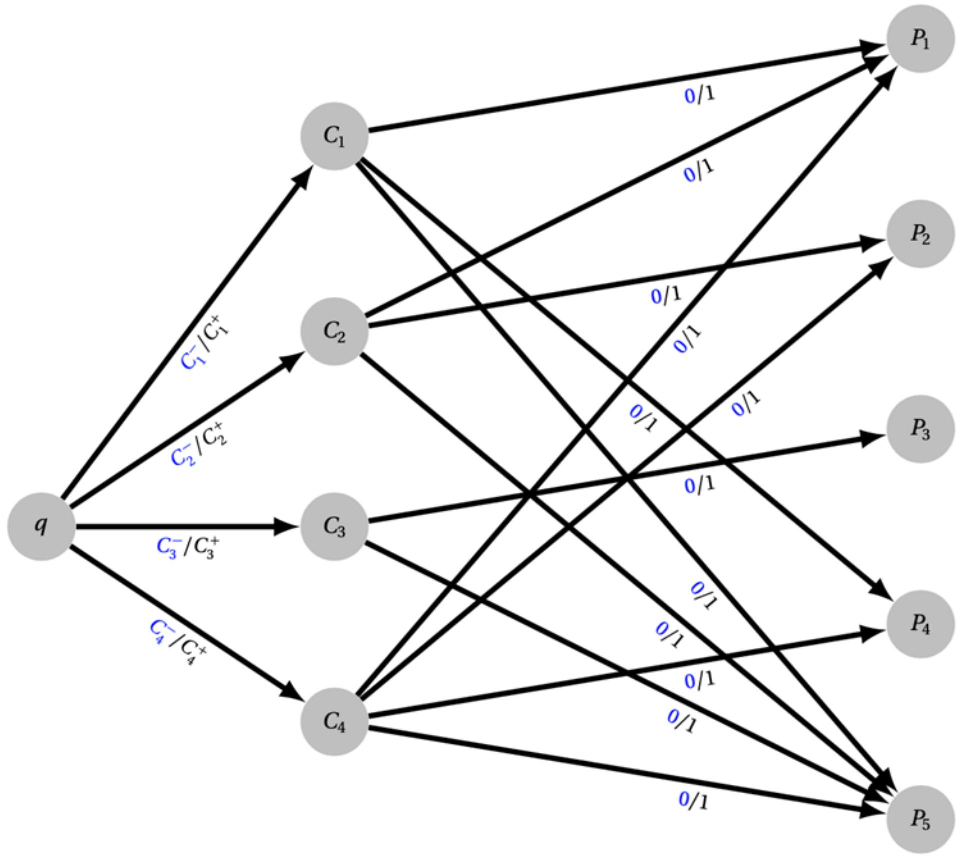
- Each customer can give 0 or 1 review for the product



Applications with Network Flow

Survey Design

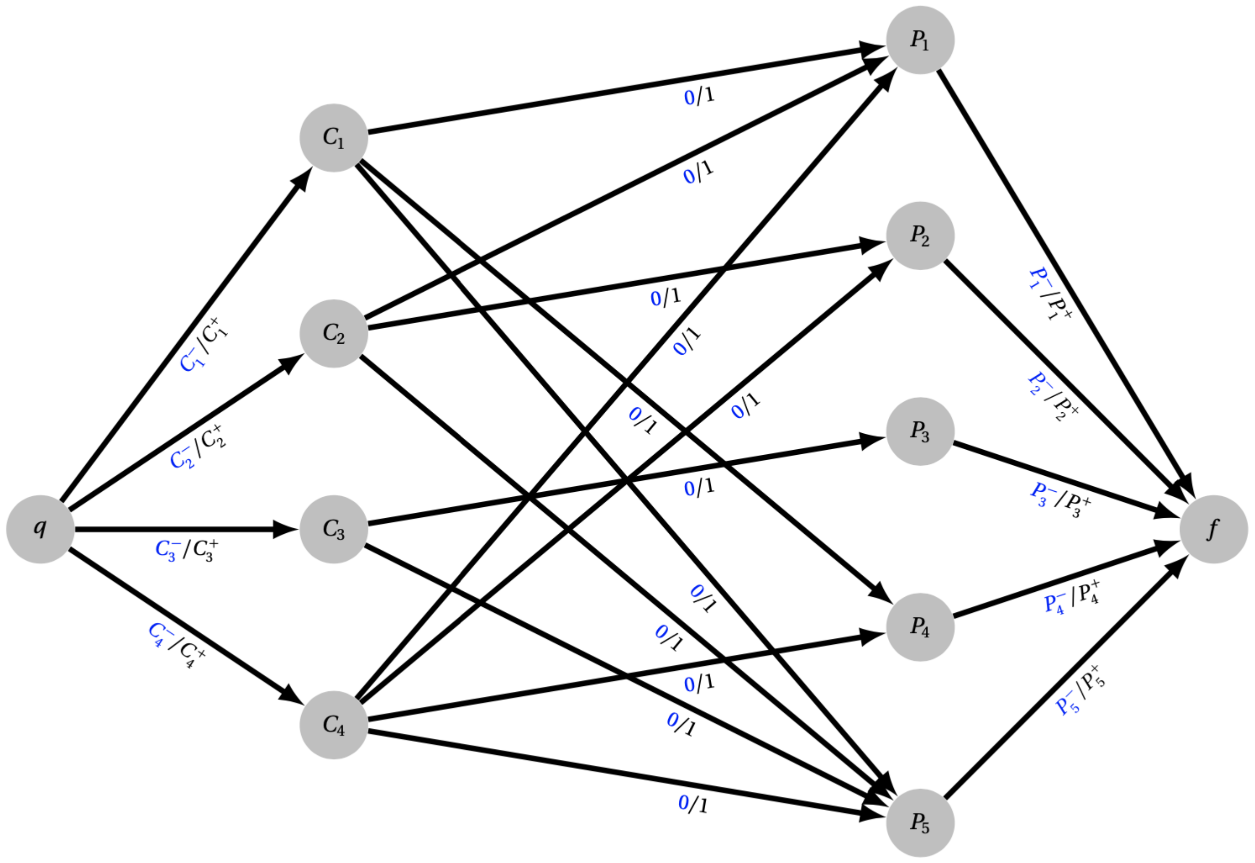
- There is range of review expected from the customer



Applications with Network Flow

Survey Design

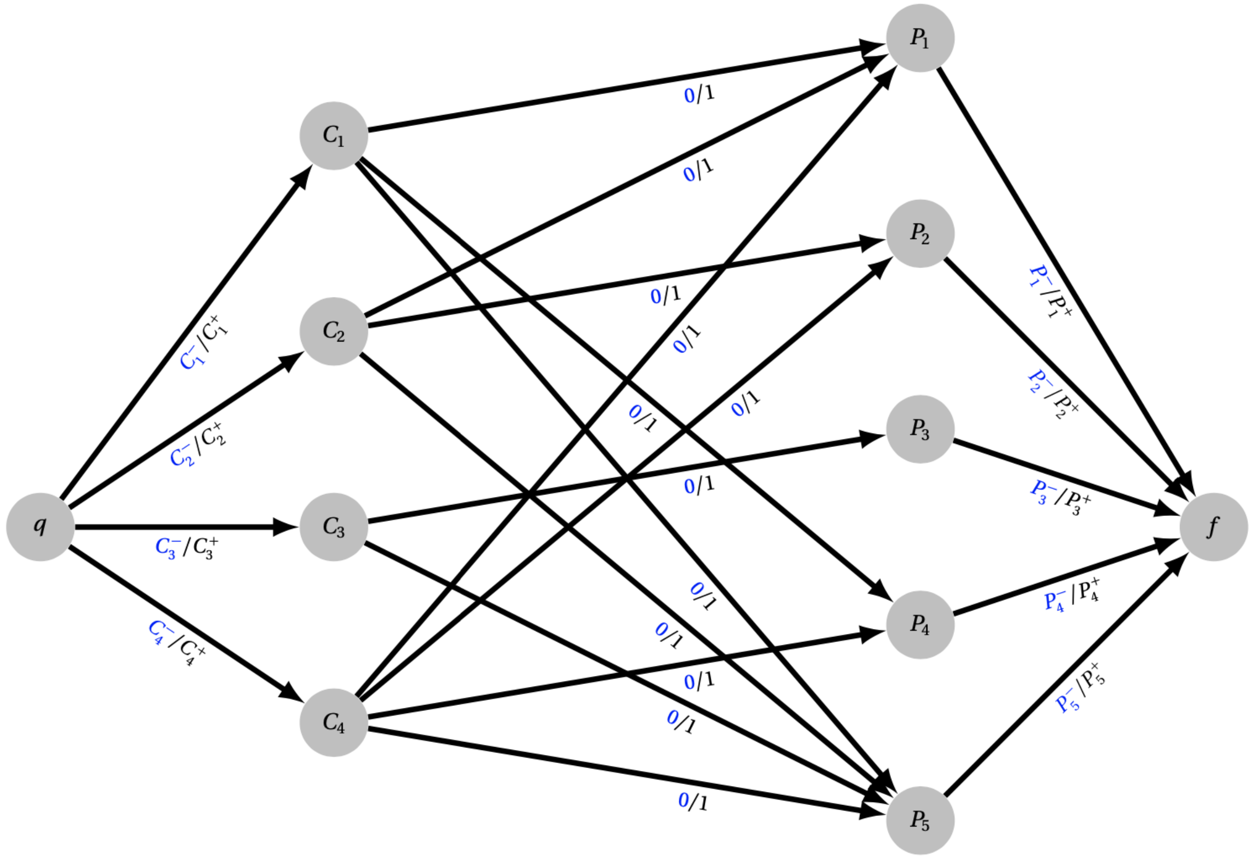
- There is range of review expected from of the product



Applications with Network Flow

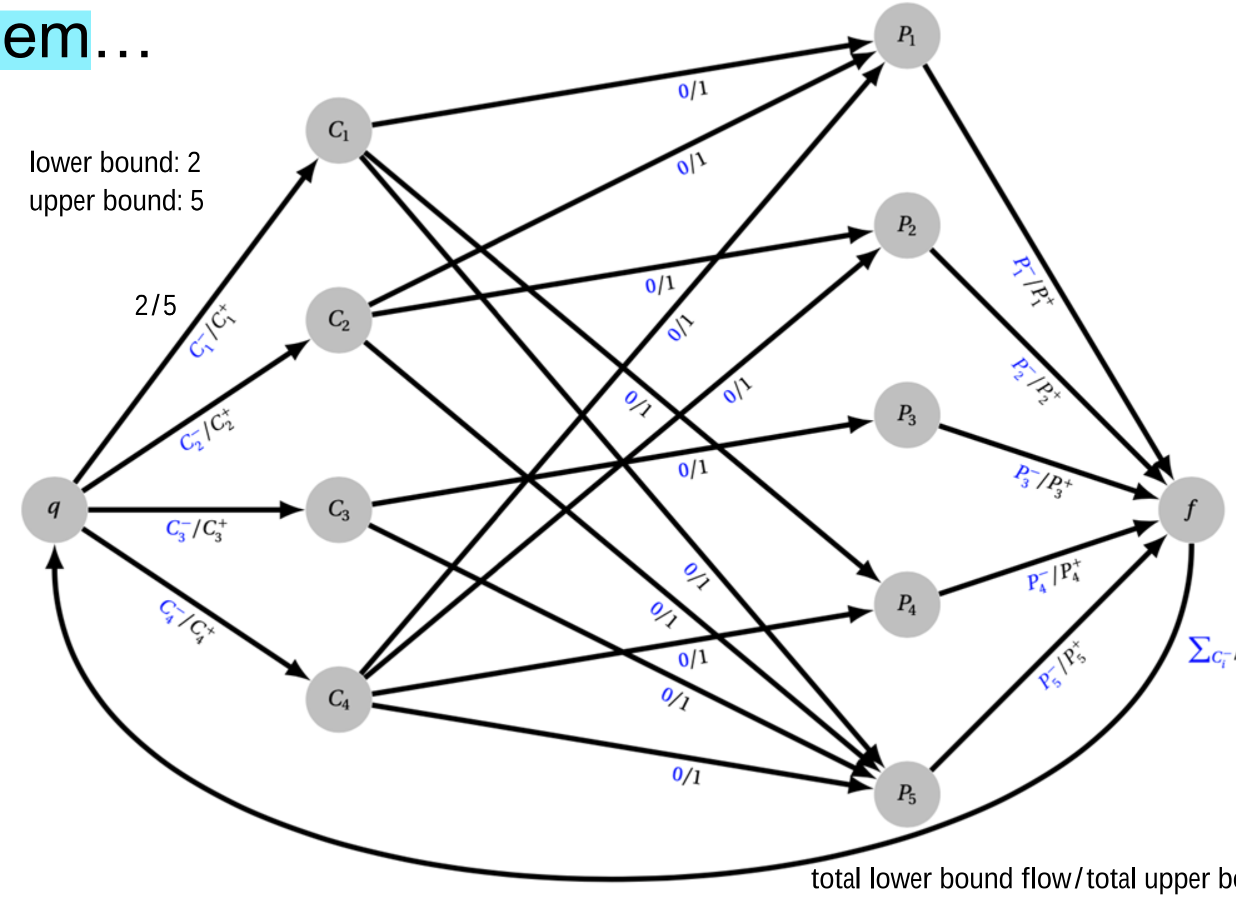
Survey Design

- In a way, it do look like a bipartite matching problem but with lower bound



Applications with Network Flow
Survey Design

- In a way, it do look like a bipartite matching problem but with lower bound, but since it is a circulation problem…



lower bound: 2
upper bound: 5

no demand
so need this edge
to meeting flow
conservation

total lower bound flow/total upper bound capacity

Questions?

Applications with Network Flow

Airline Scheduling

Applications with Network Flow
Airline Scheduling

- You have a collection of airplanes

Applications with Network Flow
Airline Scheduling

- You have a collection of airplanes

- You have a list of routes
- Some of the routes are very profitable, thus you want to fly the routes
  - Departure location
  - Departure time
  - Arrival location
  - Arrival time

Applications with Network Flow
Airline Scheduling

- You have a collection of airplanes

- You have a list of routes
- Some of the routes are very profitable, thus you want to fly the routes
  - Departure location
  - Departure time
  - Arrival location
  - Arrival time

- The airplanes can start flying from any location

Applications with Network Flow

Airline Scheduling

- You have a collection of airplanes, k

- You have a list of routes, $r_1, r_2, \ldots, r_n$
- Some of the routes are very profitable, thus you want to fly the routes
  - Departure location
  - Departure time
  - Arrival location
  - Arrival time

- The airplanes can start flying from any location

Applications with Network Flow
Airline Scheduling

- Imagine you have the following routes:
  - Route 1: SYD 6am – MEL 7am
  - Route 2: CBR 8am – SYD 9am
  - Route 3: MEL 11am – BNE 1pm
  - Route 4: PER 11am – SYD 7pm

Applications with Network Flow
Airline Scheduling

- Imagine you have the following routes:
  - Route 1: SYD 6am – MEL 7am
  - Route 2: CBR 8am – SYD 9am
  - Route 3: MEL 11am – BNE 1pm (can fit route1 and route2 prior)
  - Route 4: PER 11am – SYD 7pm (can fit route1 prior)

Applications with Network Flow
Airline Scheduling

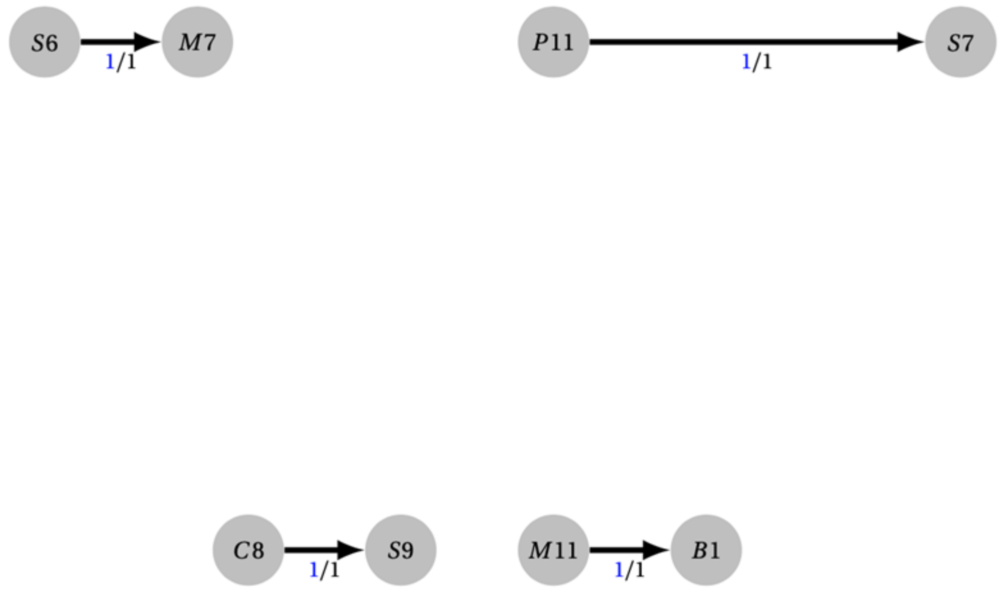
- Imagine you have the following routes:
  - Route 1: SYD 6am – MEL 7am
  - Route 2: CBR 8am – SYD 9am
  - Route 3: MEL 11am – BNE 1pm (can fit route1 and route2 prior)
  - Route 4: PER 11am – SYD 7pm (can fit route1 prior)

- Can you cover these 4 vital routes, using only 2 planes?

Applications with Network Flow

Airline Scheduling

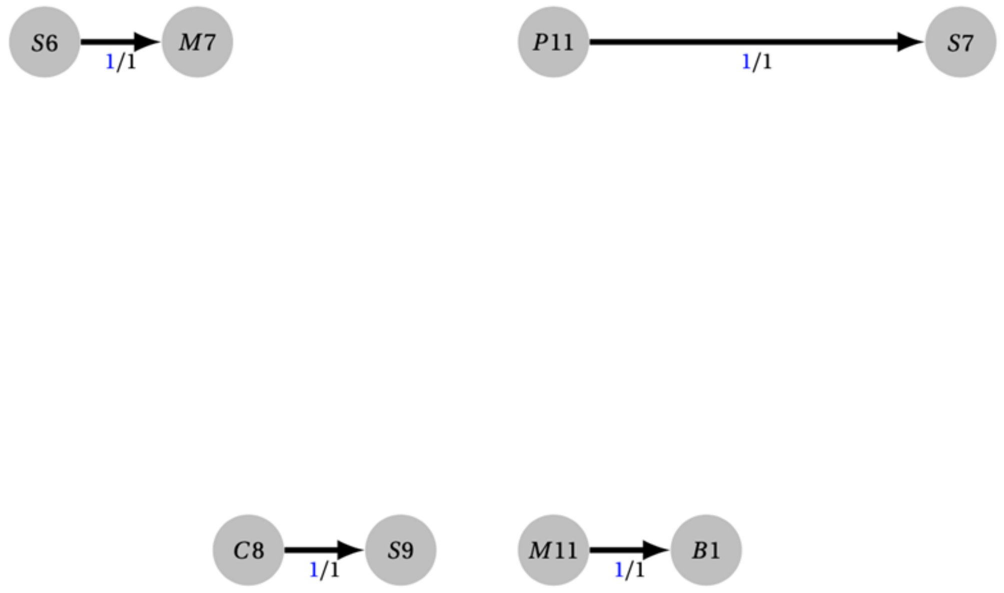
- First, we list down the routes imagine the x-axis as time…



Applications with Network Flow

Airline Scheduling

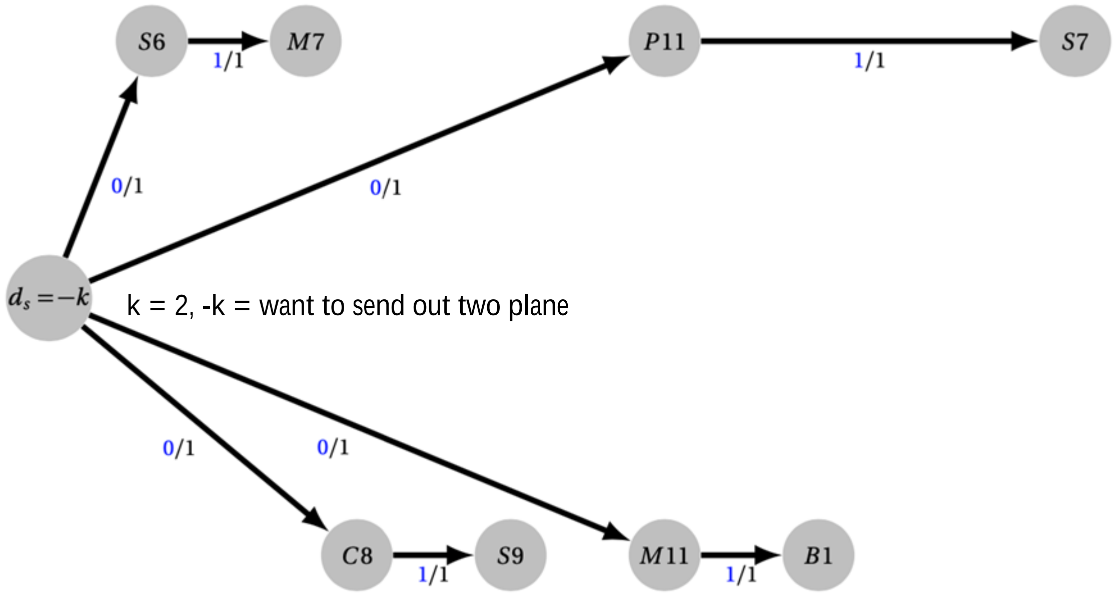
- First, we list down the routes imagine the x-axis as time…
  - Since they are vital flights, we want to always fly and thus lower-bound is set to 1.



Applications with Network Flow

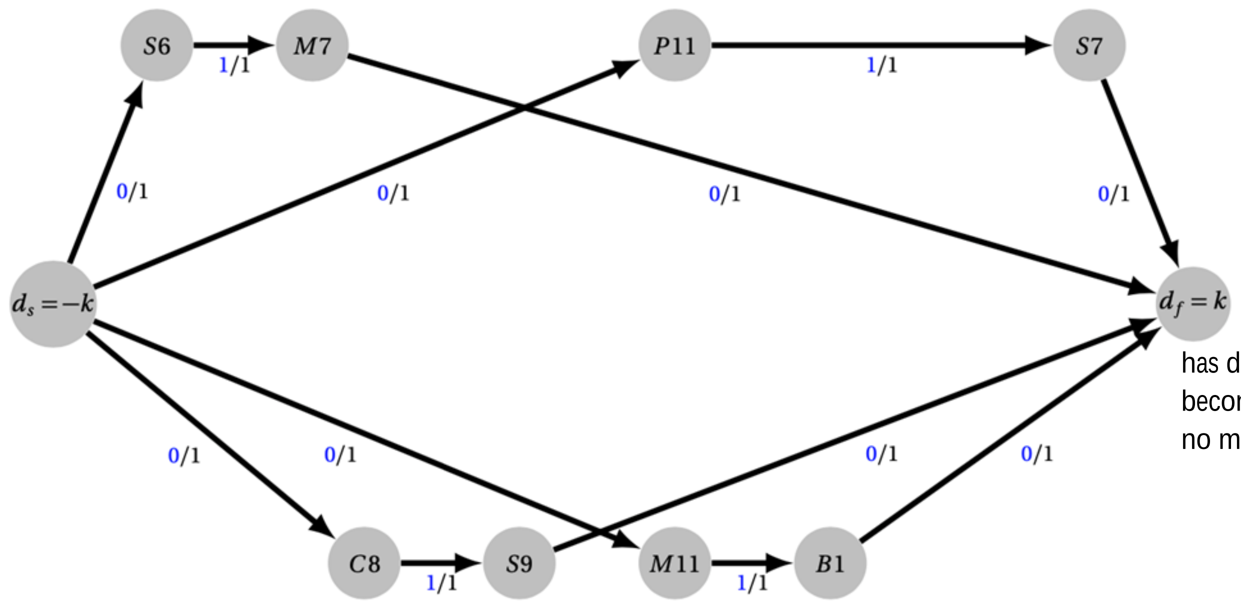
Airline Scheduling

- Then we add a source, which we can place our planes from in any of the locations.
  - Lower bound is 0 because there is not requirement to be placed at which location



Applications with Network Flow
Airline Scheduling

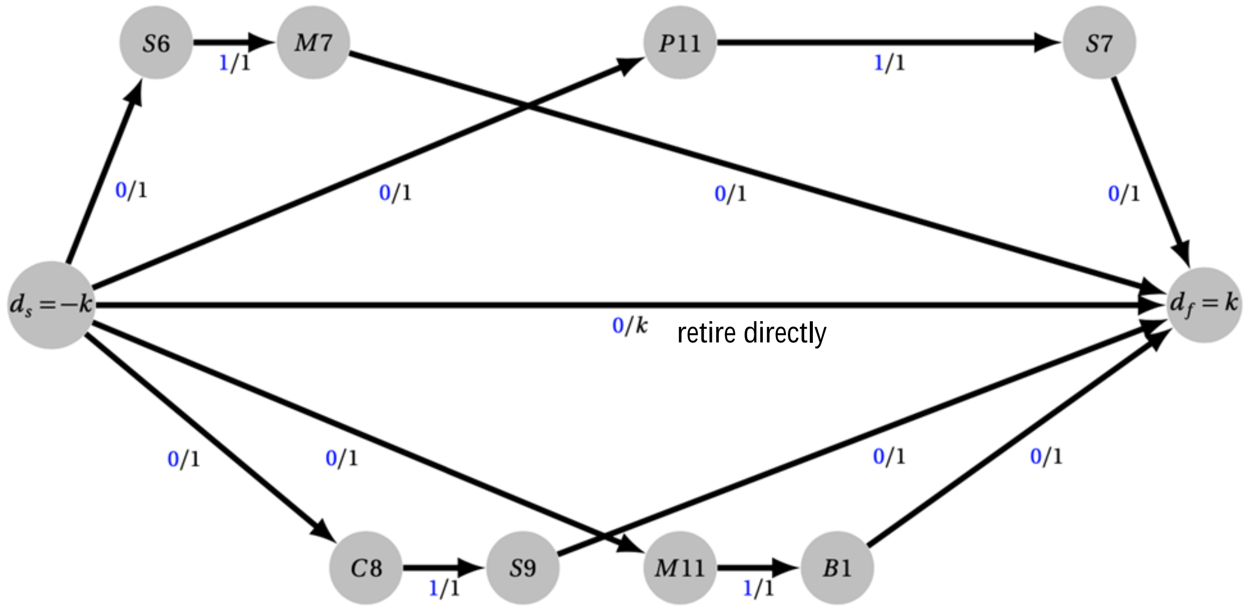
- Next, we add a sink which we can retire our planes at any of the locations.
  - No requirement for the planes to retire from any location



has demand
become demand conservation
no more flow conservation

Applications with Network Flow
Airline Scheduling

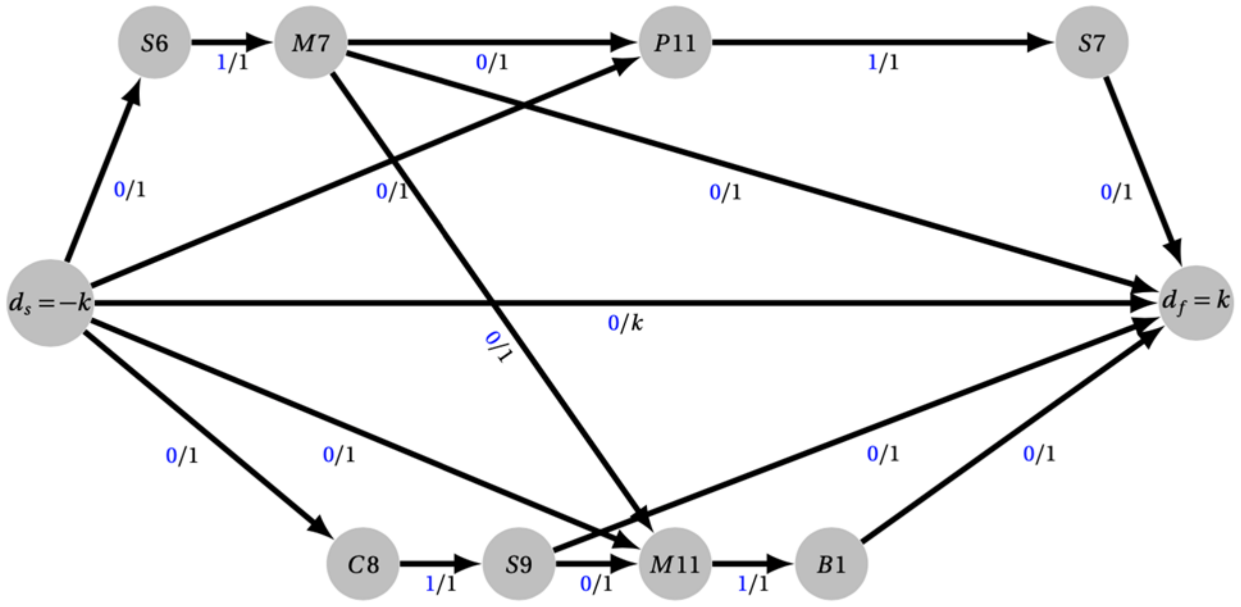
- But what if we don’t need all our planes to cover all vital routes?
  - Thus, they can go from start to retire directly



Applications with Network Flow

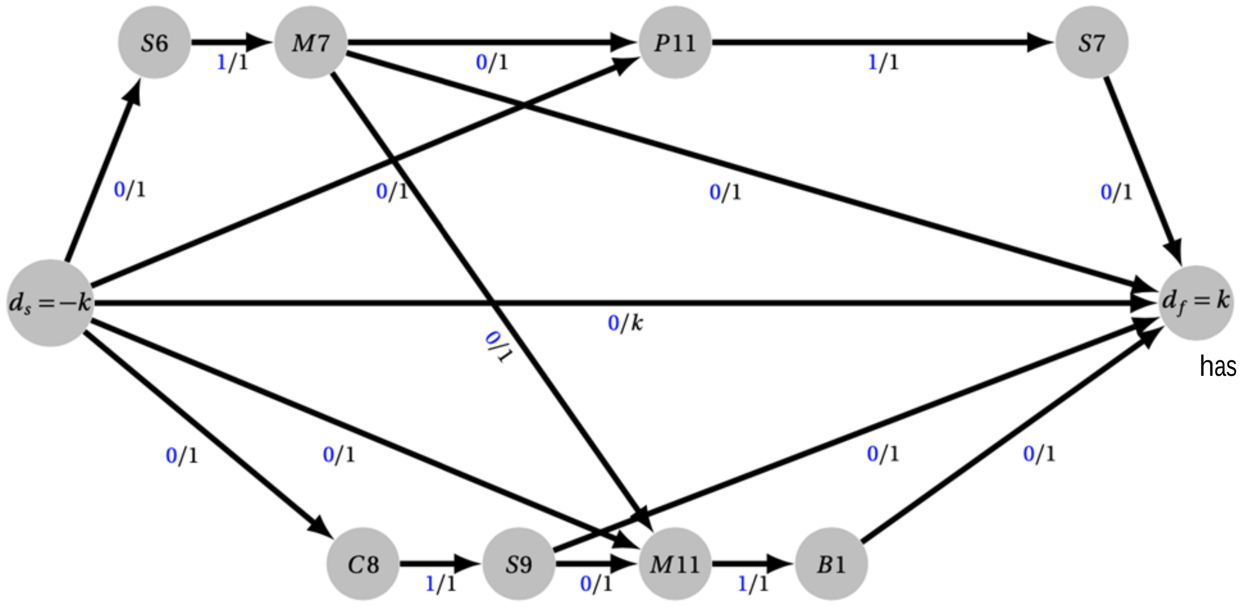
Airline Scheduling

- Since it is possible for a plane to follow a route, then go to another route instead of retiring…
  - We add the edge, again it is optional



Applications with Network Flow
Airline Scheduling

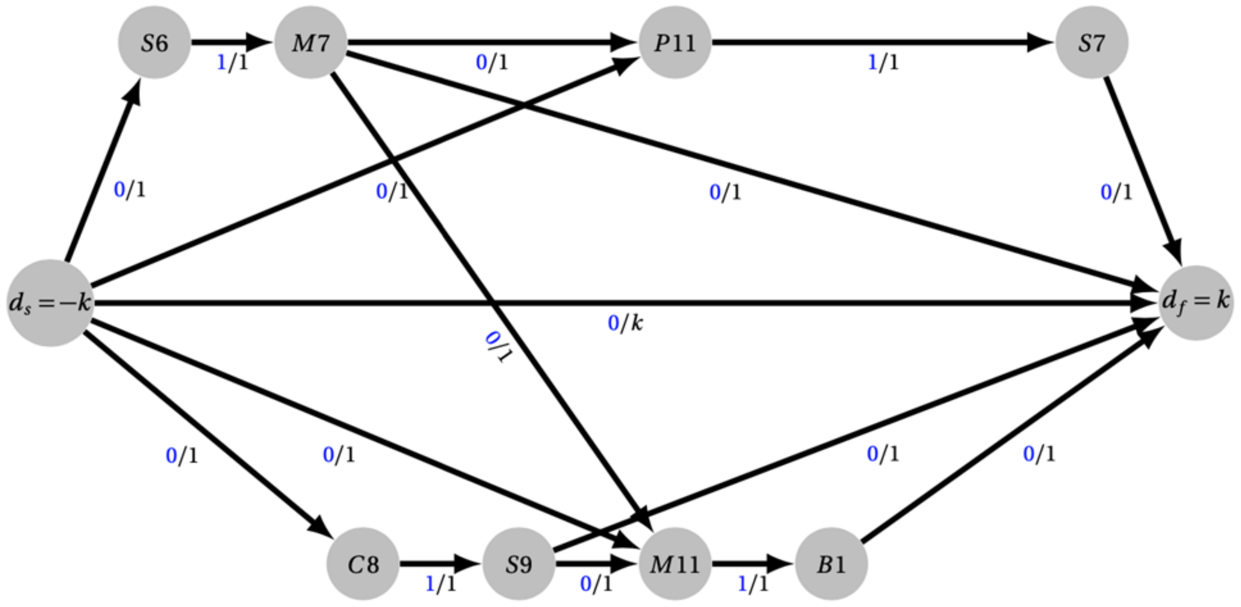
- Then we just solve this as it is!



has demand

Applications with Network Flow
Airline Scheduling

- Then we just solve this as it is!
- Answer is yes with 2 planes
  - Plane1: Route1 (S6->M7), then Route4 (P11->S7)
  - Plane2: Route2 (C8->S9), then Route3 (M11->B1)



Questions?

Applications with Network Flow

Other Examples?

Applications with Network Flow

Other Examples?

- Several examples in the studio

Applications with Network Flow
Other Examples?

- Several examples in the studio
  - Choosing profitable projects
  - Determining if teams/ players can progress in a tournament

Applications with Network Flow
Other Examples?

- Several examples in the studio
  - Choosing profitable projects
  - Determining if teams/ players can progress in a tournament
- … and many more
  - Open-pit mining
  - Image segmentation (e.g., background/foreground segmentation)
  - Network connectivity
  - Data mining
  - Distributed computing
  - Network intrusion detection
  - Edge-disjoint paths in graphs
  - Network reliability
  - Multi-camera scene reconstruction
  - Gene function prediction

Questions?

MONASH University

Thank You