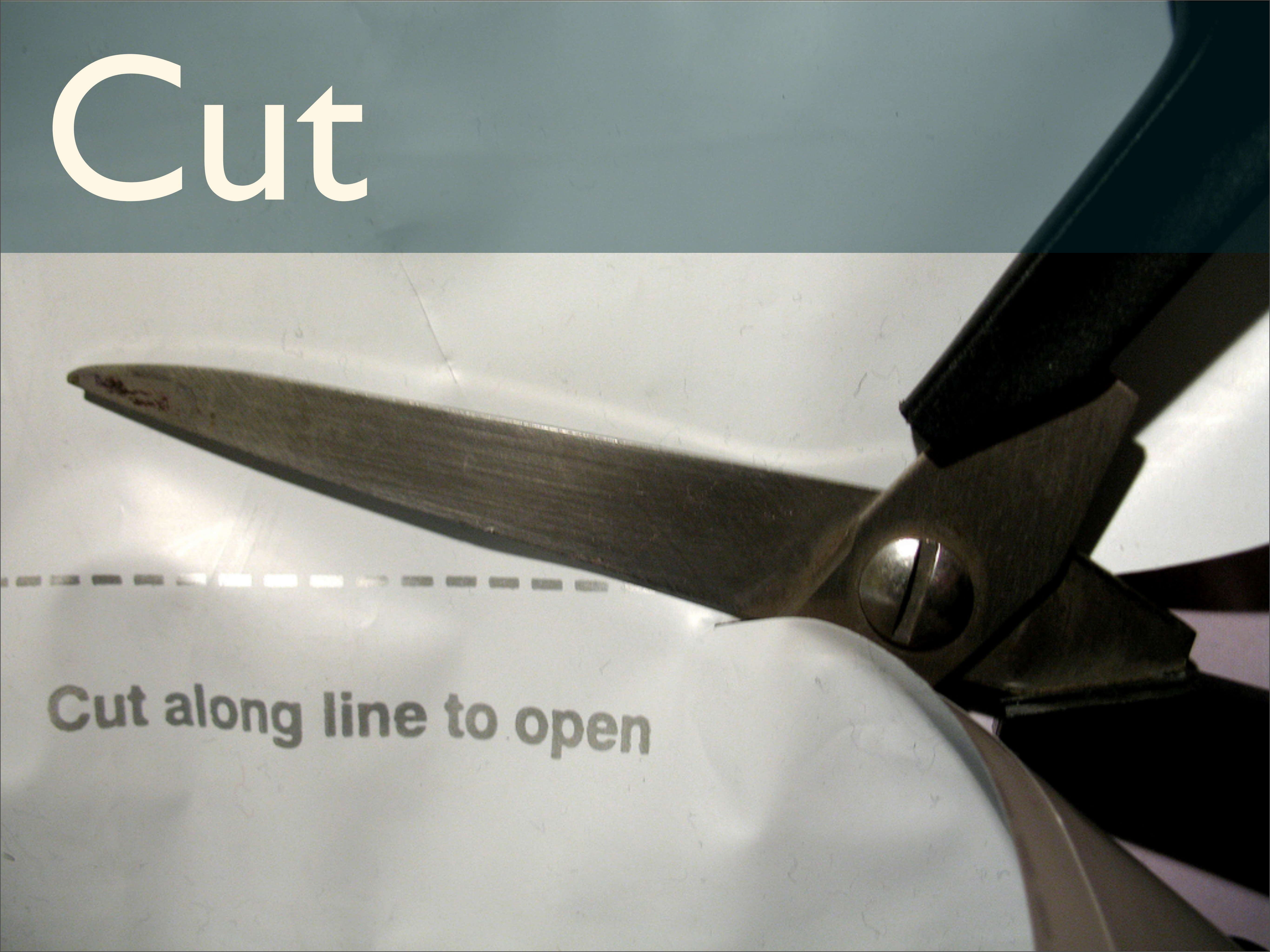
Minimum Cut

郭至軒（KuoE0）

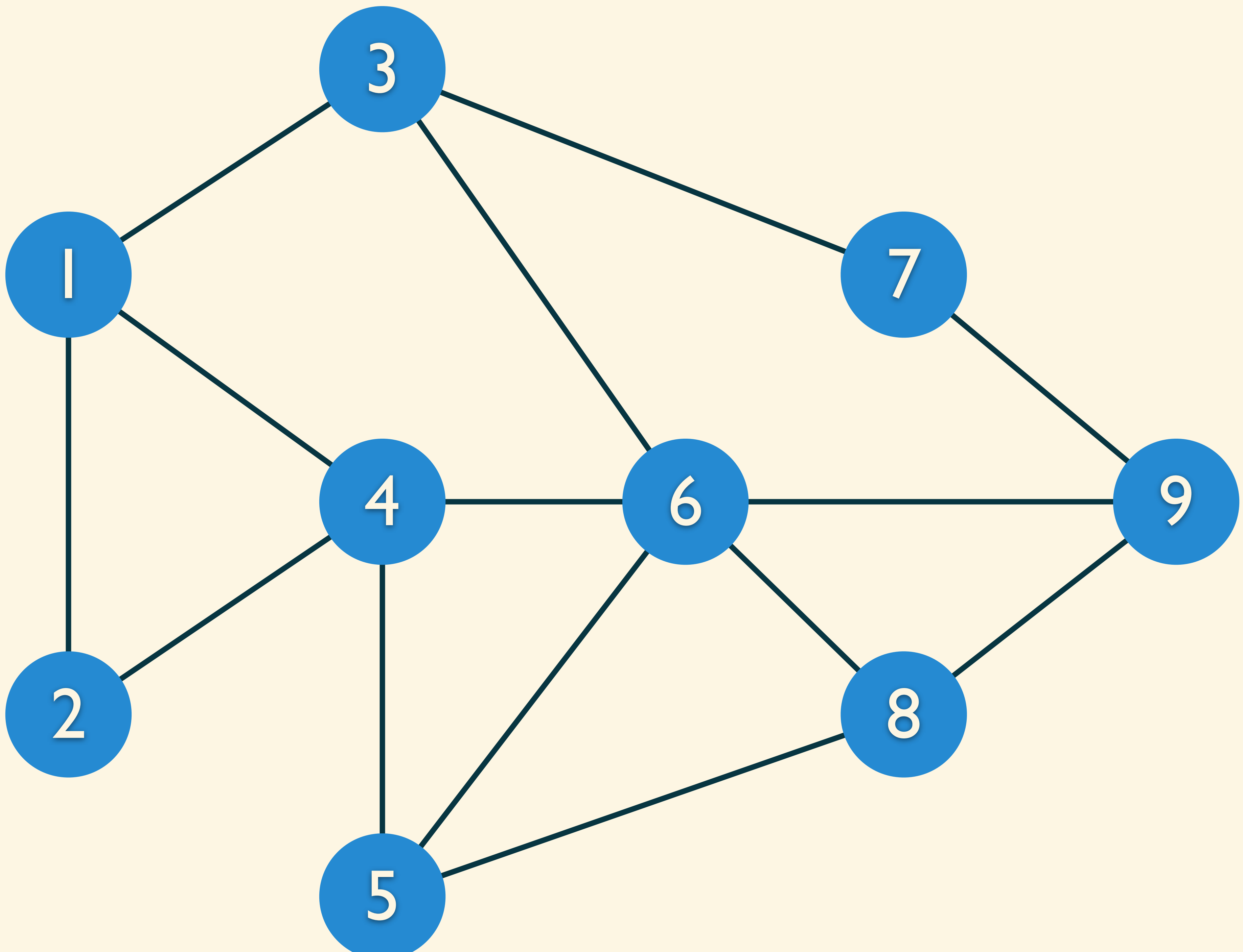
KuoE0.tw@gmail.com
KuoE0.ch

Cut



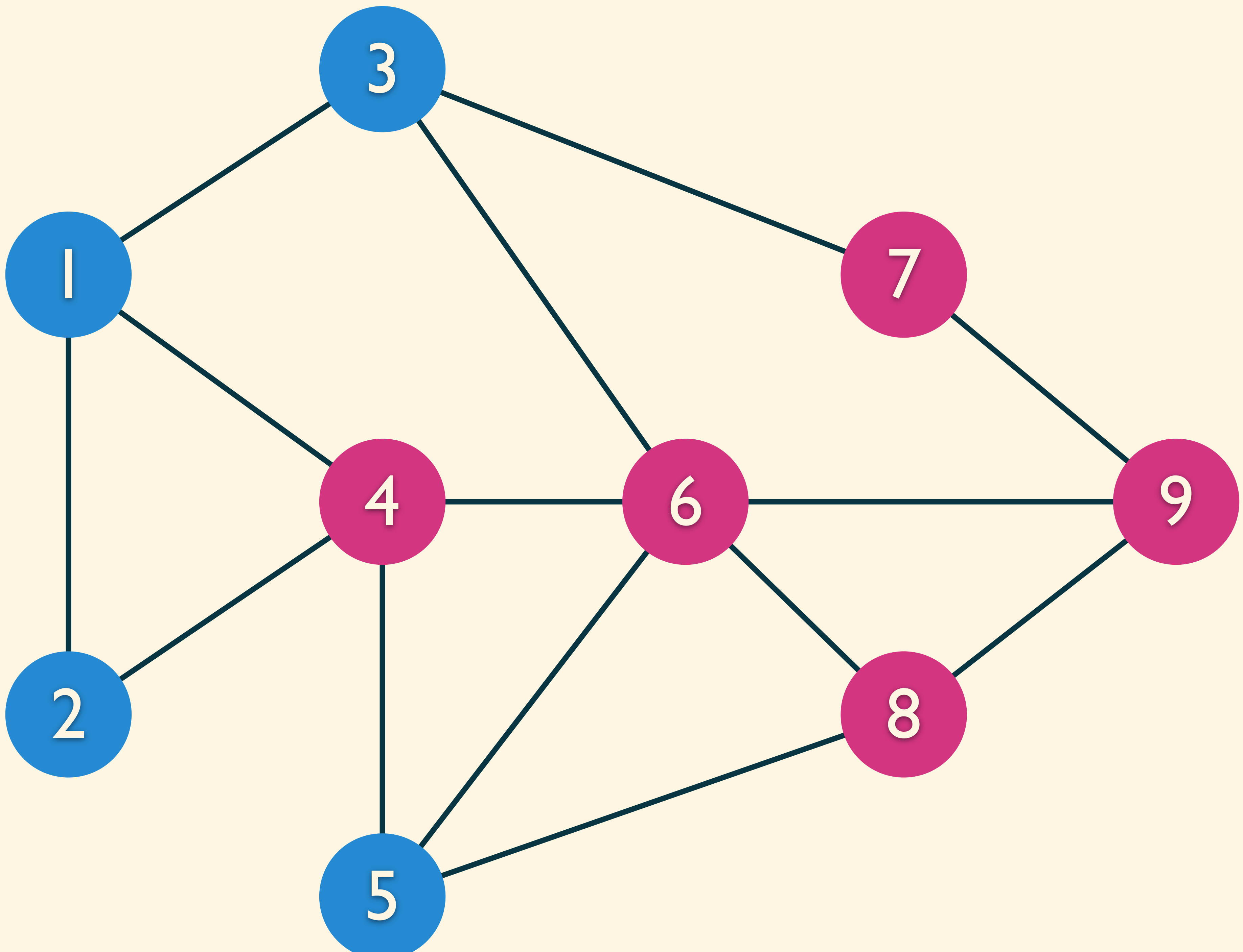
cut (undirected)

A **partition** of the vertices of a graph into **two disjoint subsets**



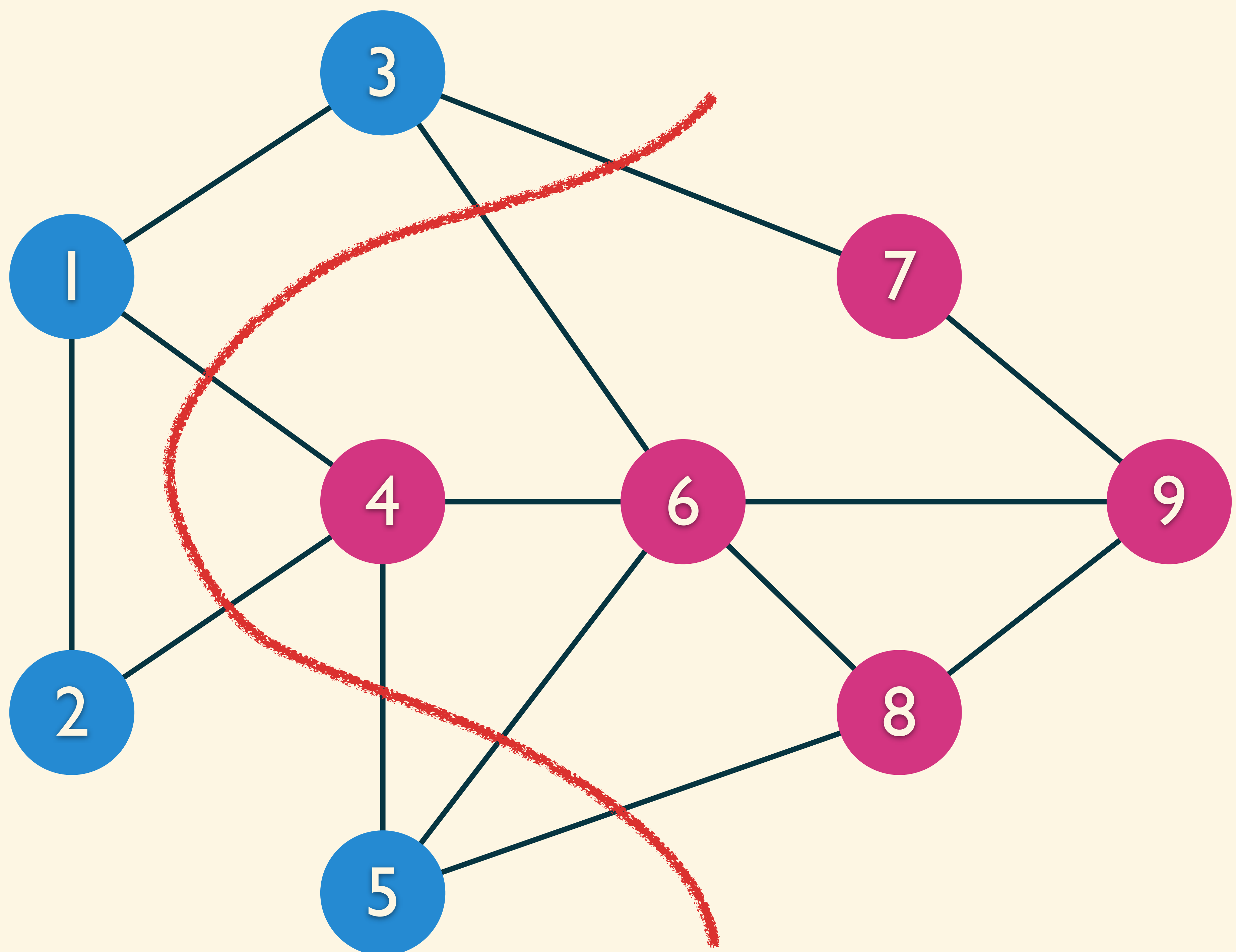
undirected graph

A **partition** of the vertices of a graph into **two disjoint subsets**



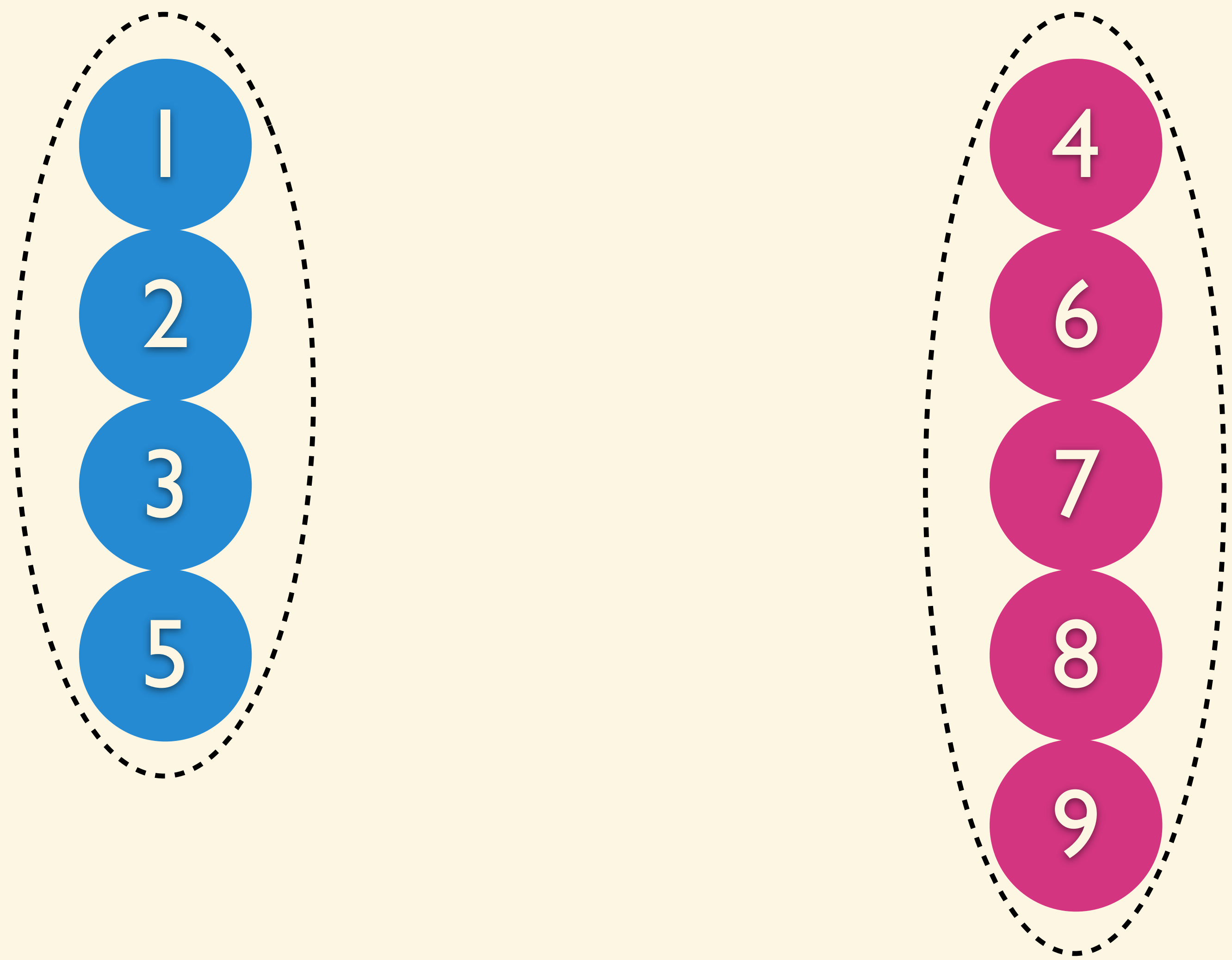
undirected graph

A **partition** of the vertices of a graph into **two disjoint subsets**



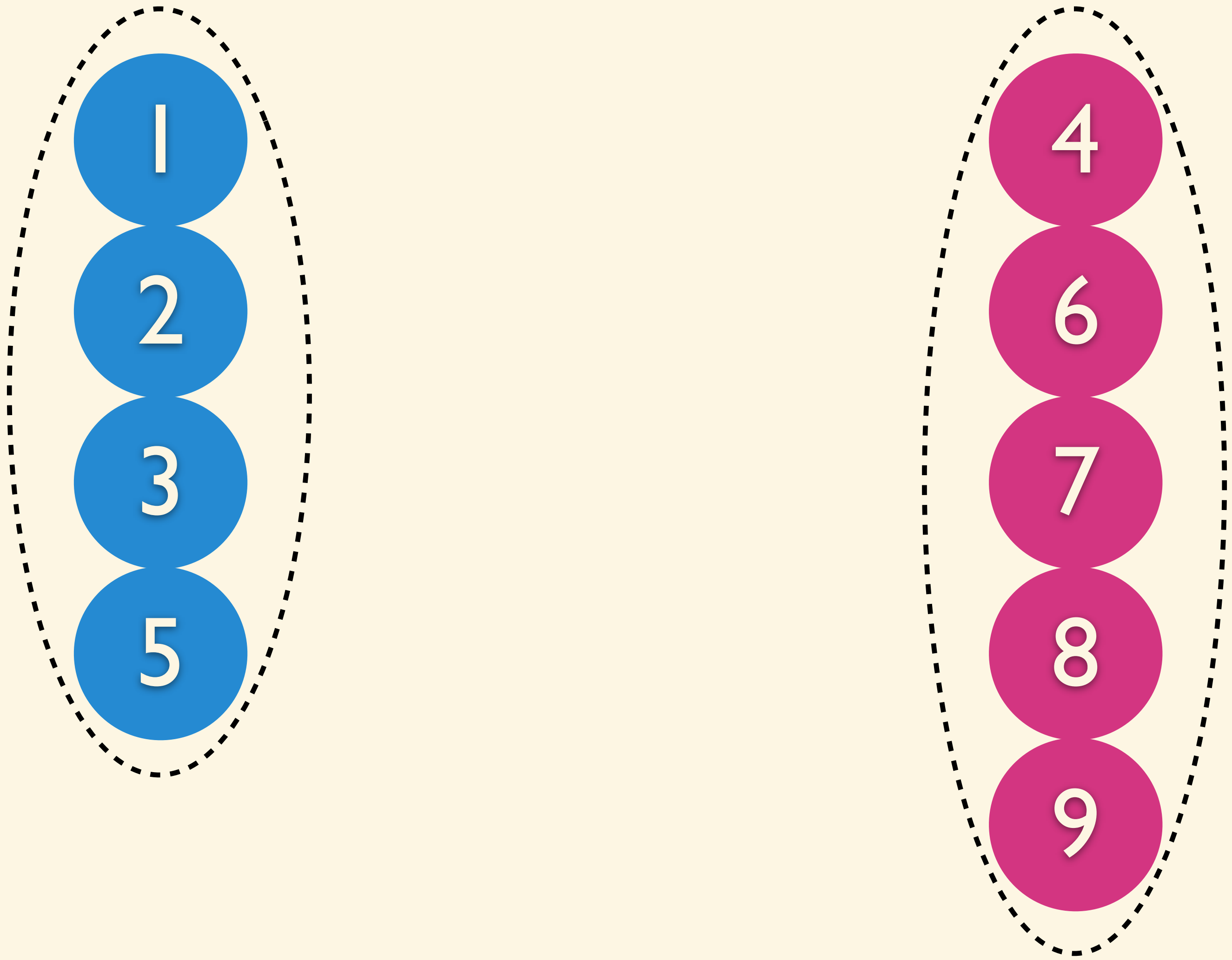
undirected graph

A **partition** of the vertices of a graph into **two disjoint subsets**



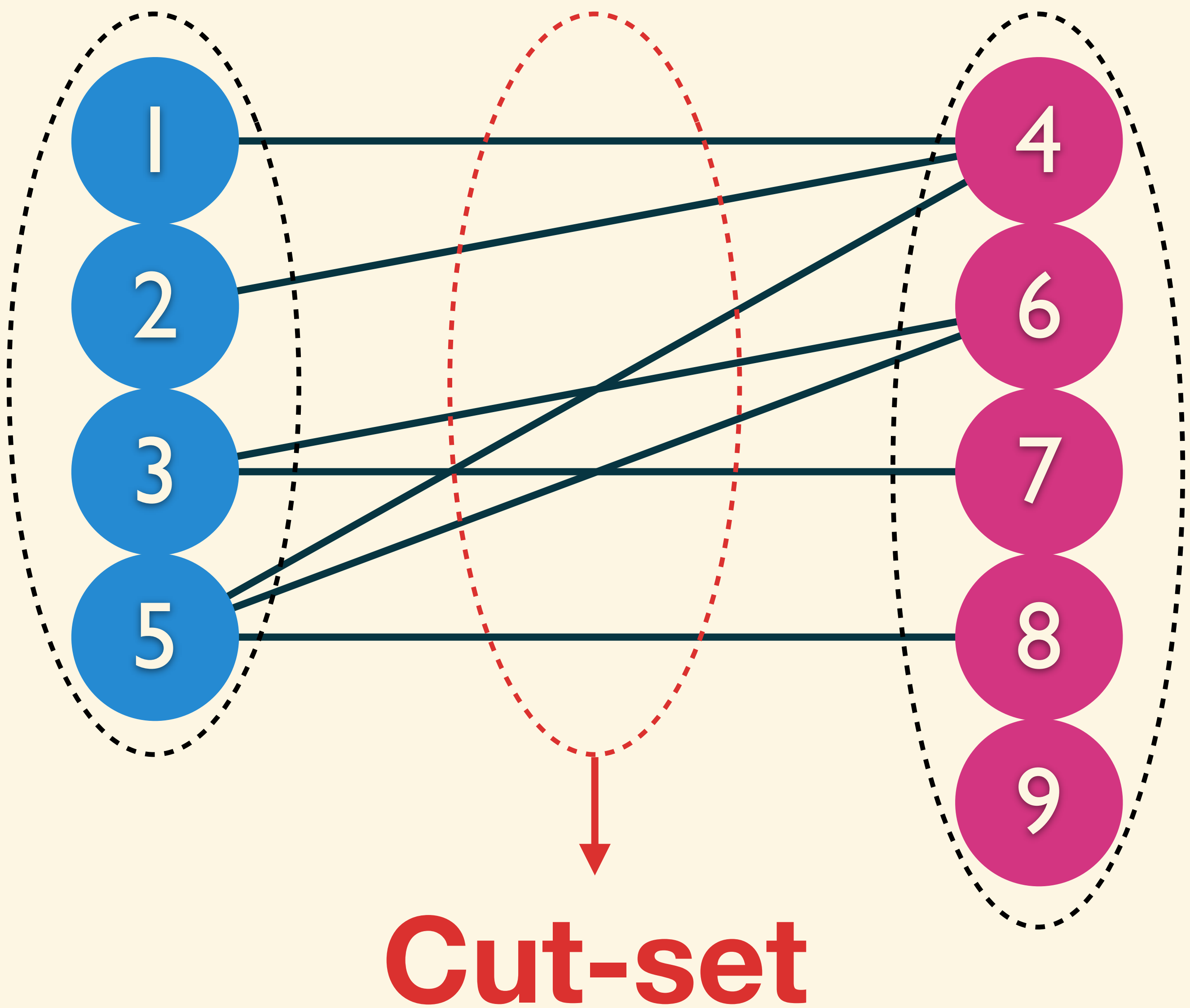
undirected graph

Cut-set of the cut is the set of **edges whose end points are in different subsets**.



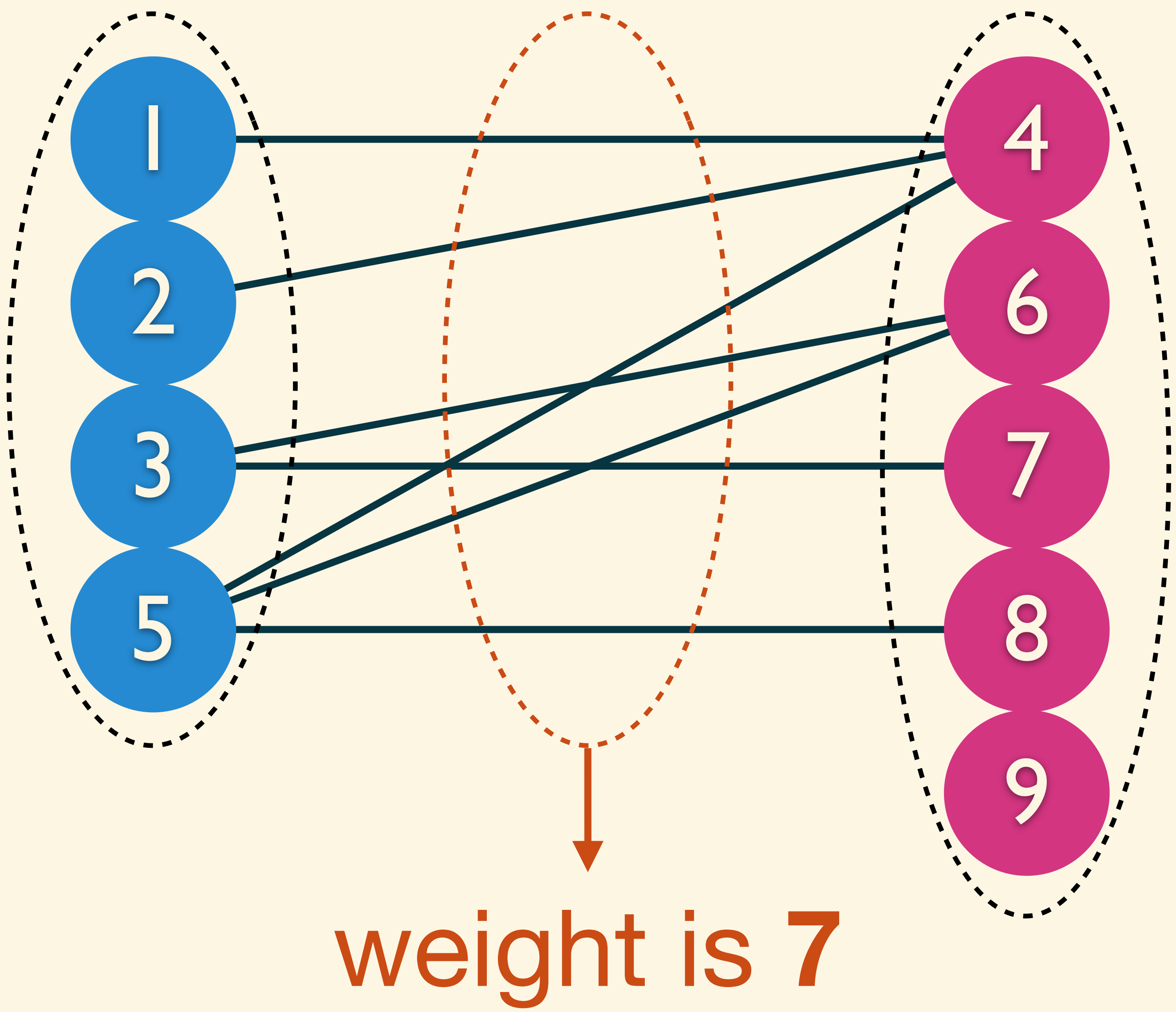
undirected graph

Cut-set of the cut is the set of **edges whose end points are in different subsets**.



undirected graph

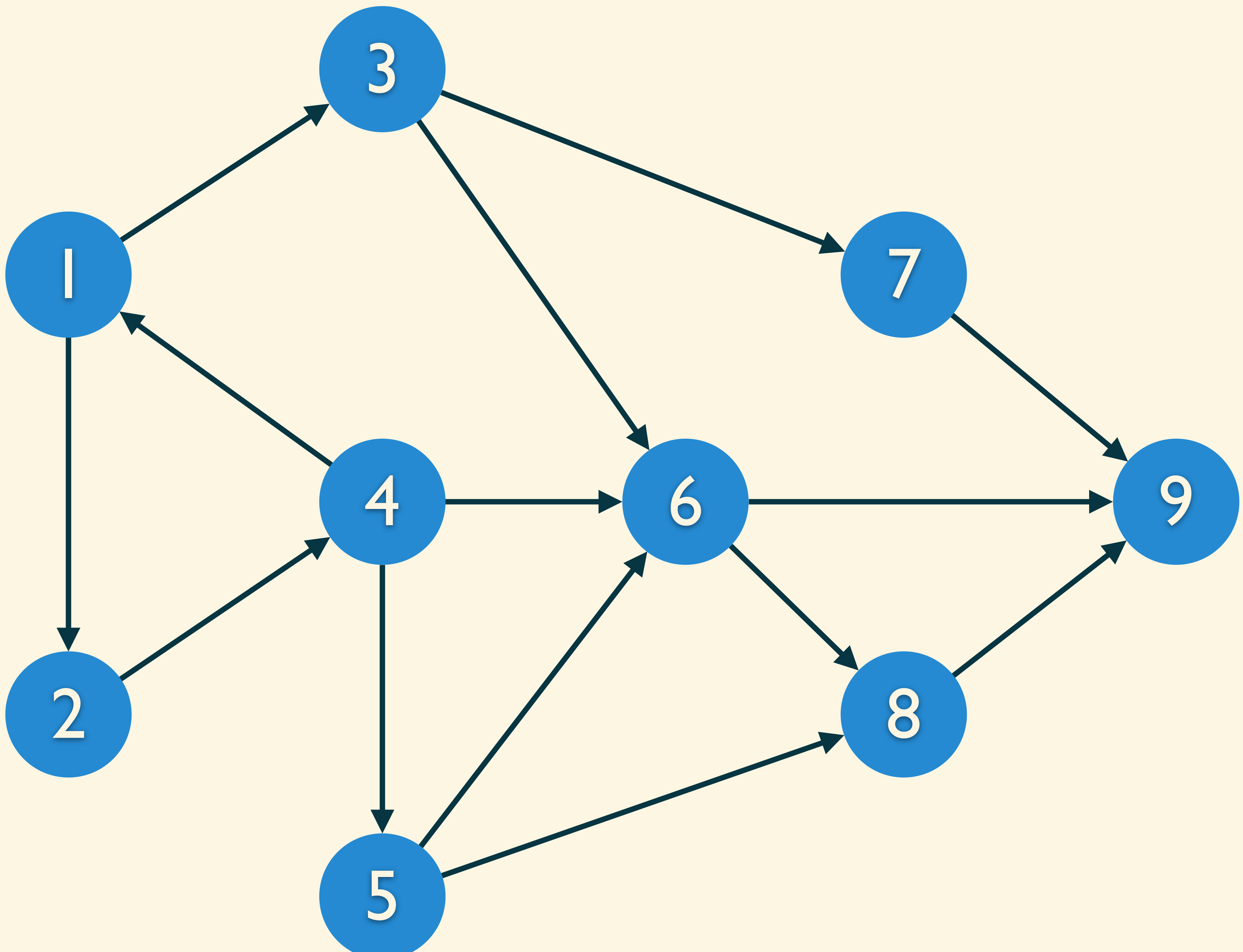
weight = **number of edges** or **sum of weight on edges**



undirected graph

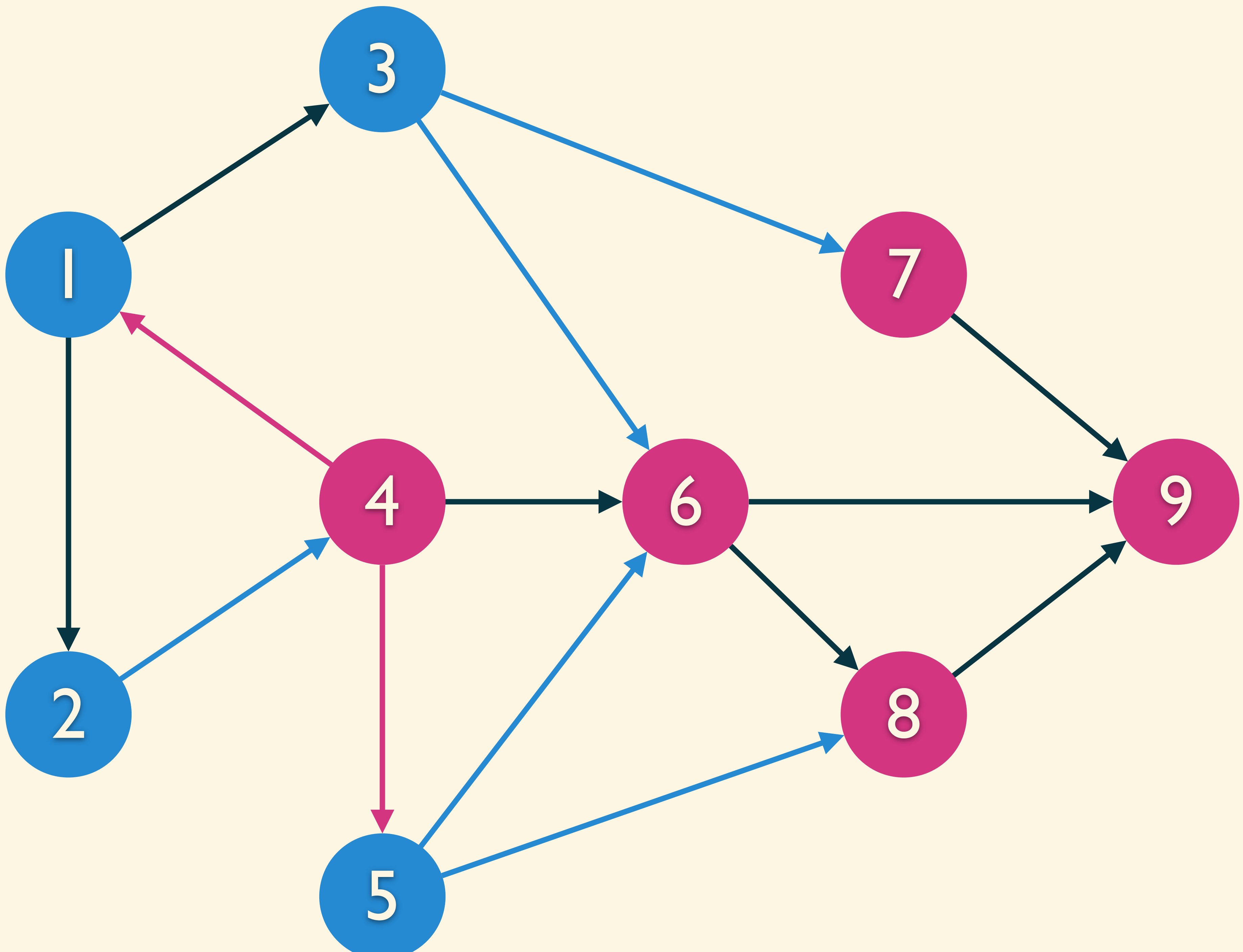
cut (directed)

A **partition** of the vertices of a graph into **two disjoint subsets**



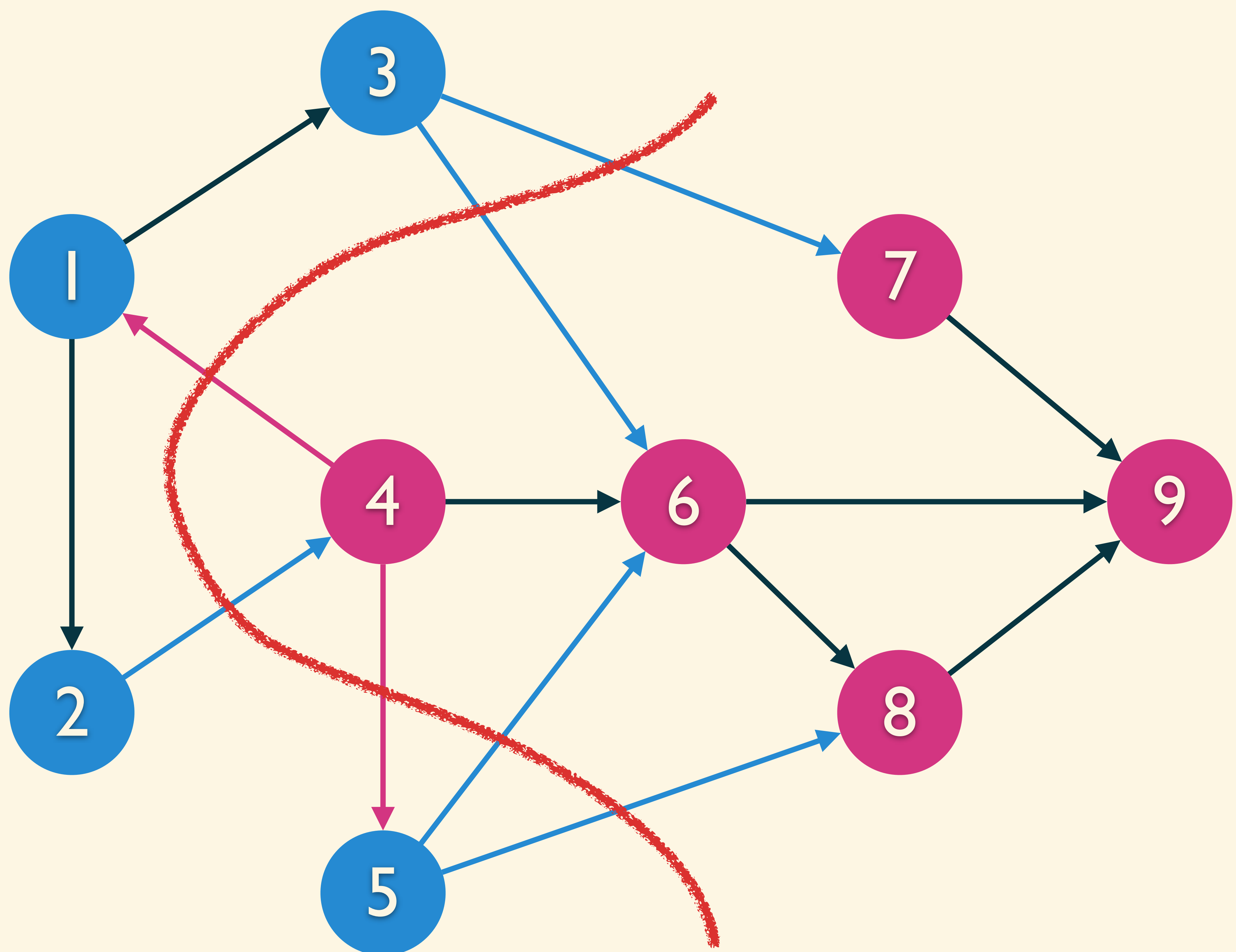
directed graph

A **partition** of the vertices of a graph into **two disjoint subsets**



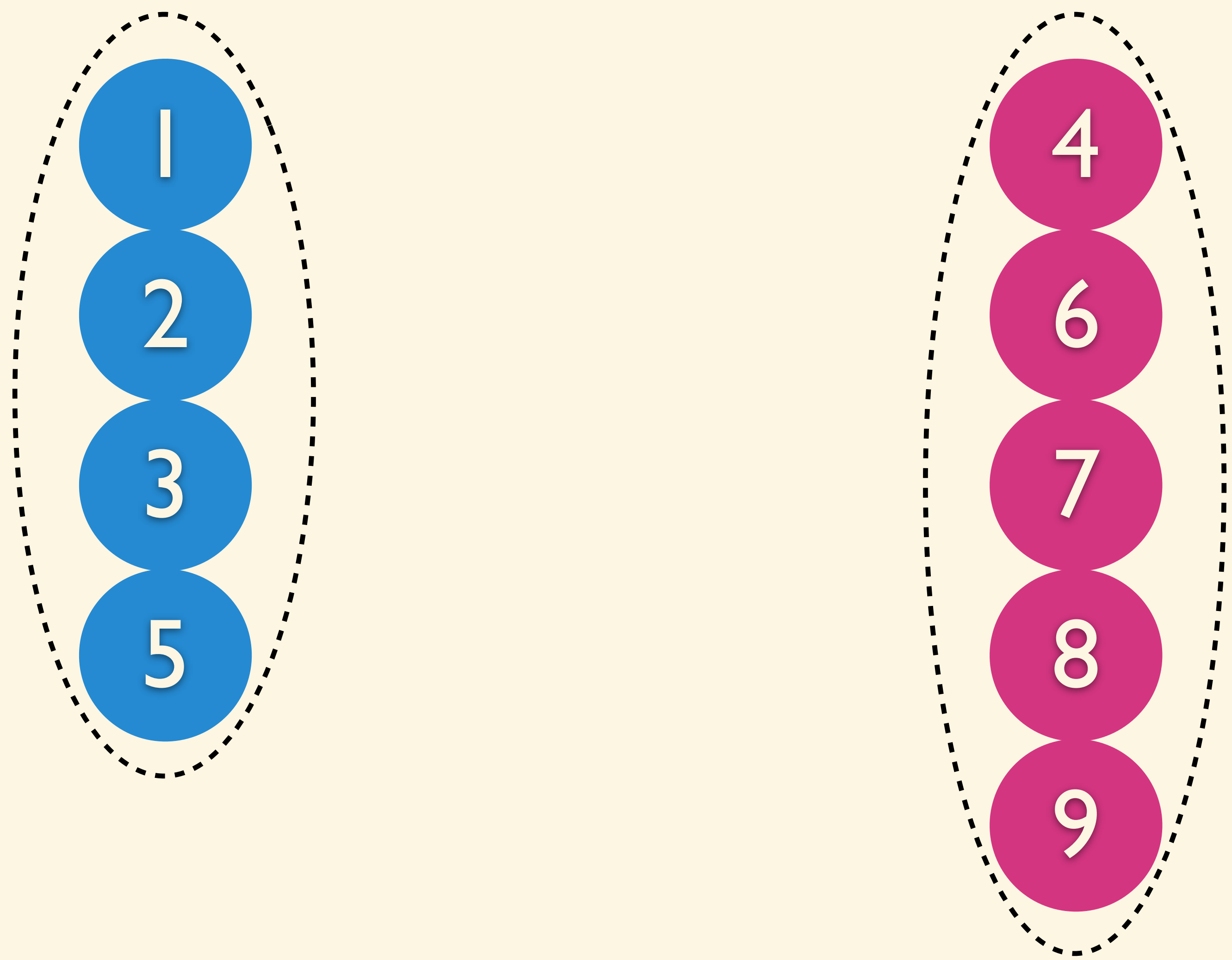
directed graph

A **partition** of the vertices of a graph into **two disjoint subsets**



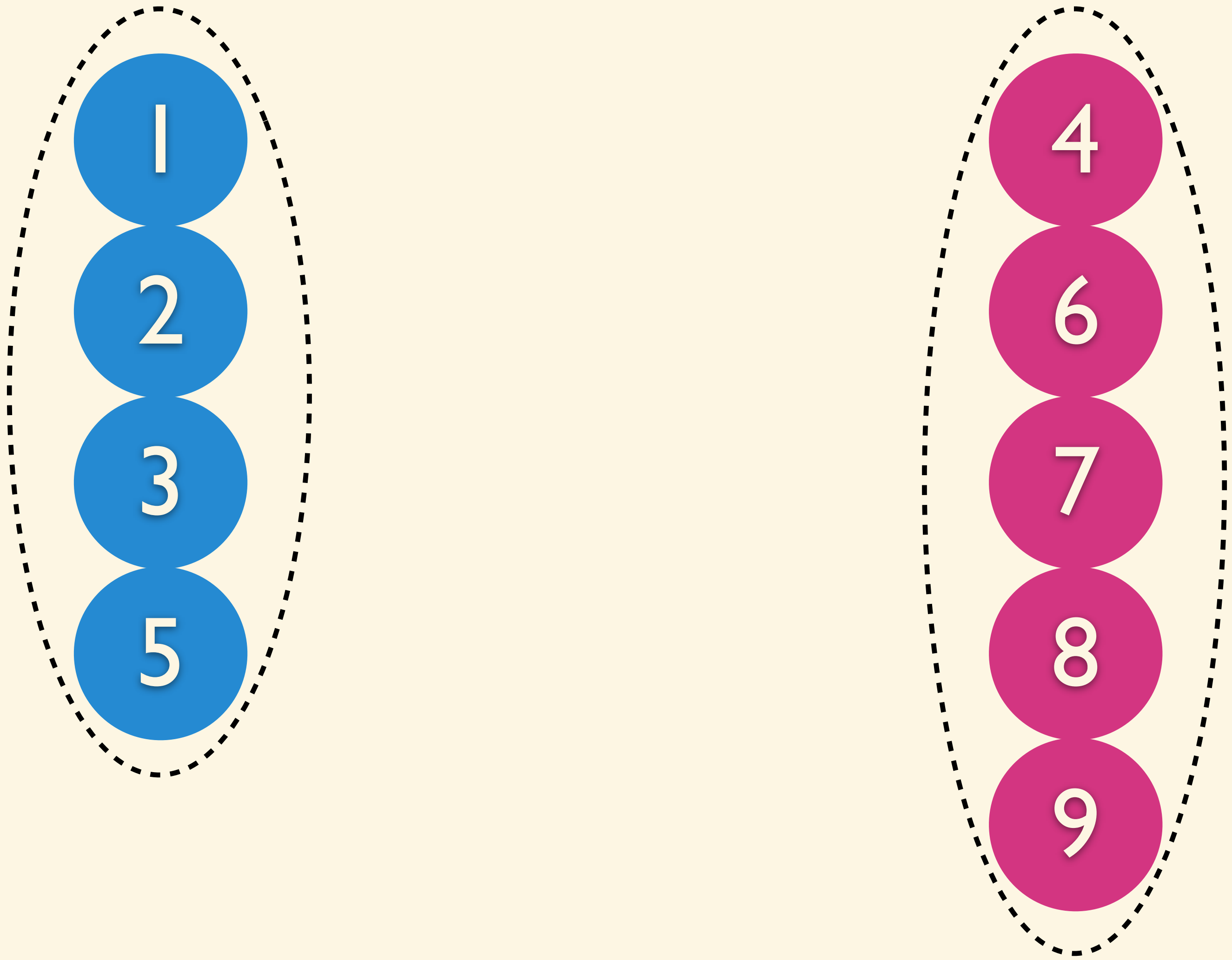
directed graph

A **partition** of the vertices of a graph into **two disjoint subsets**



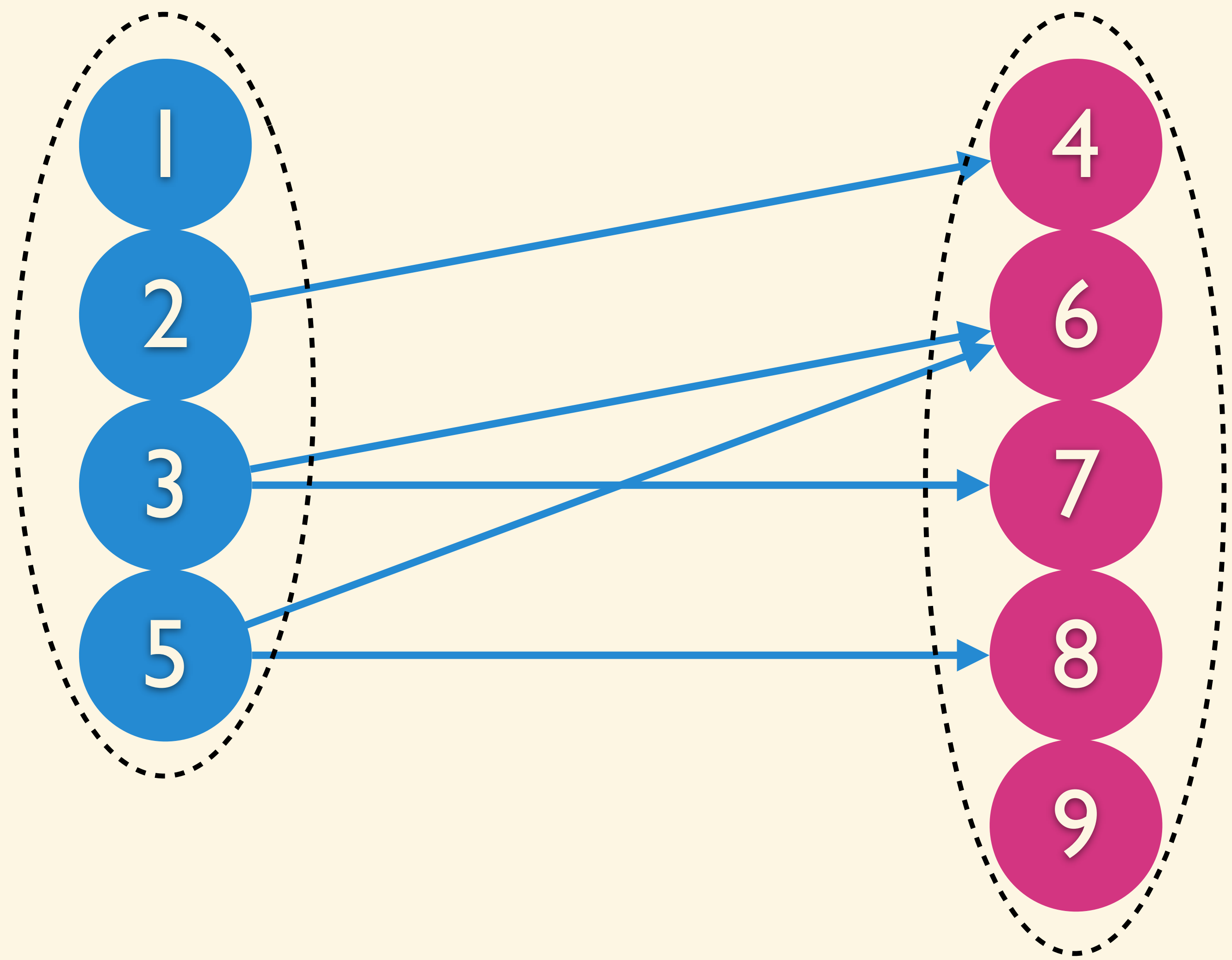
directed graph

Cut-set of the cut is the set of **edges whose end points are in different subsets**.



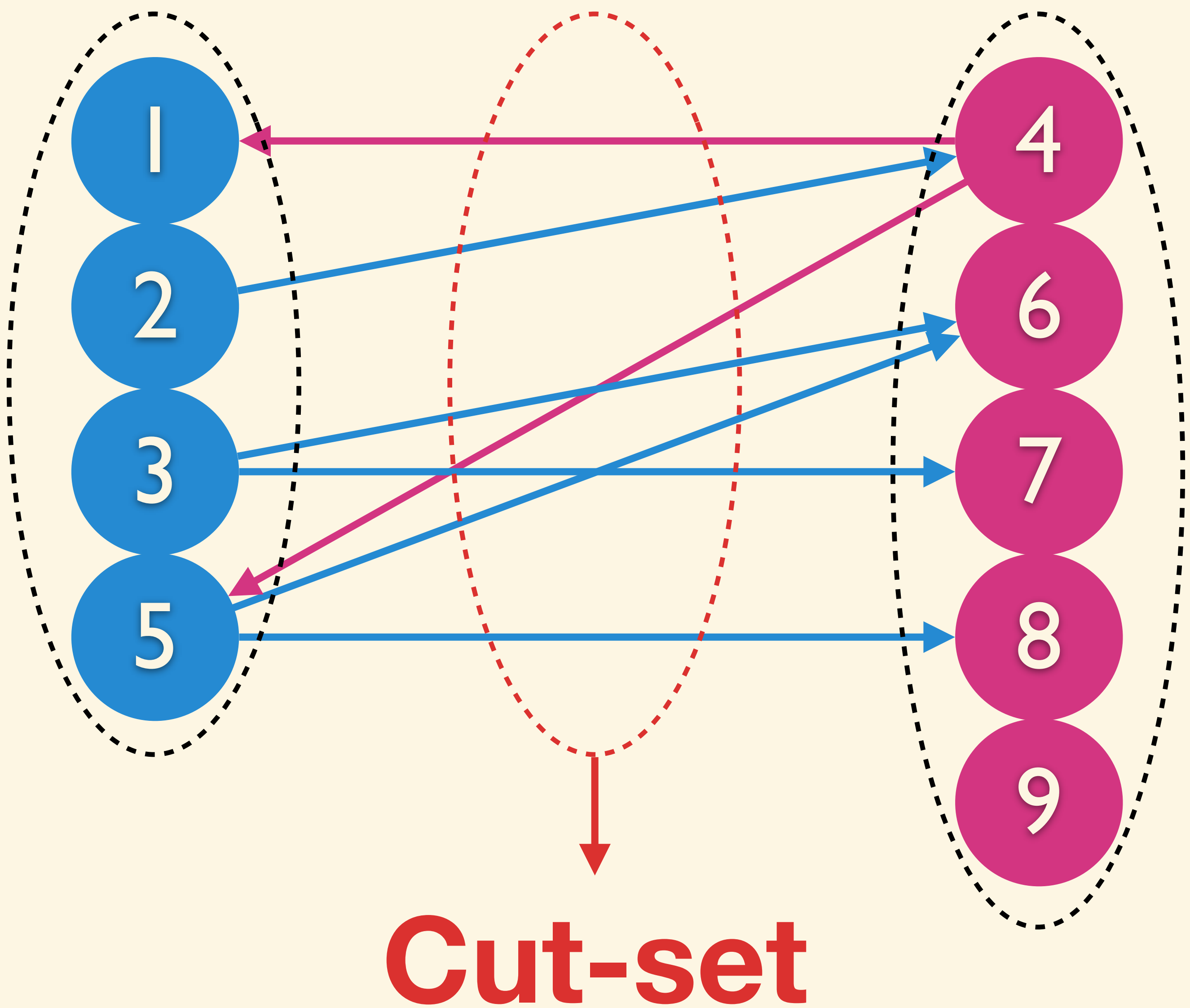
directed graph

Cut-set of the cut is the set of **edges whose end points are in different subsets**.



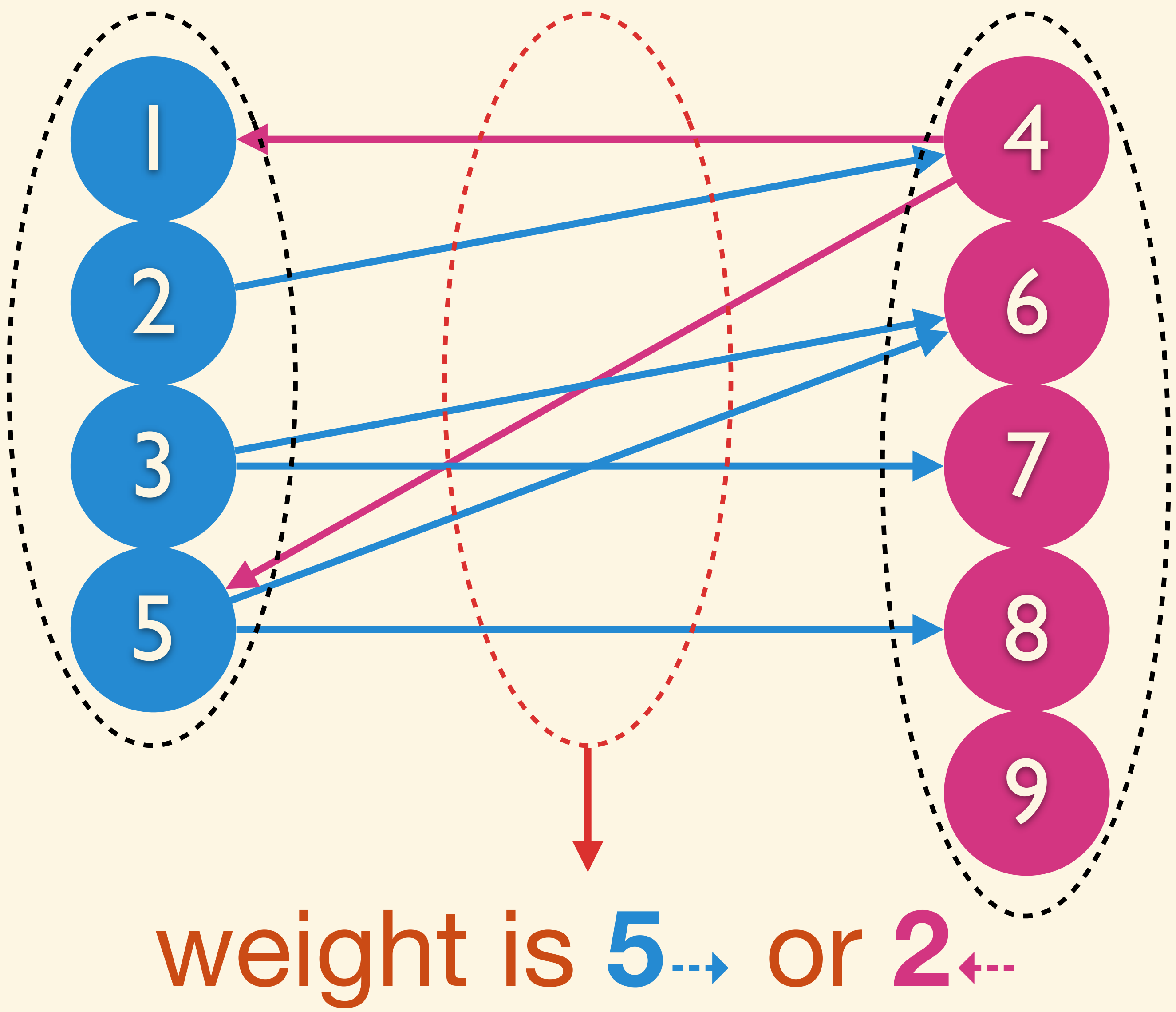
directed graph

Cut-set of the cut is the set of **edges whose end points are in different subsets**.



directed graph

weight = **number of edges** or **sum of weight on edges**



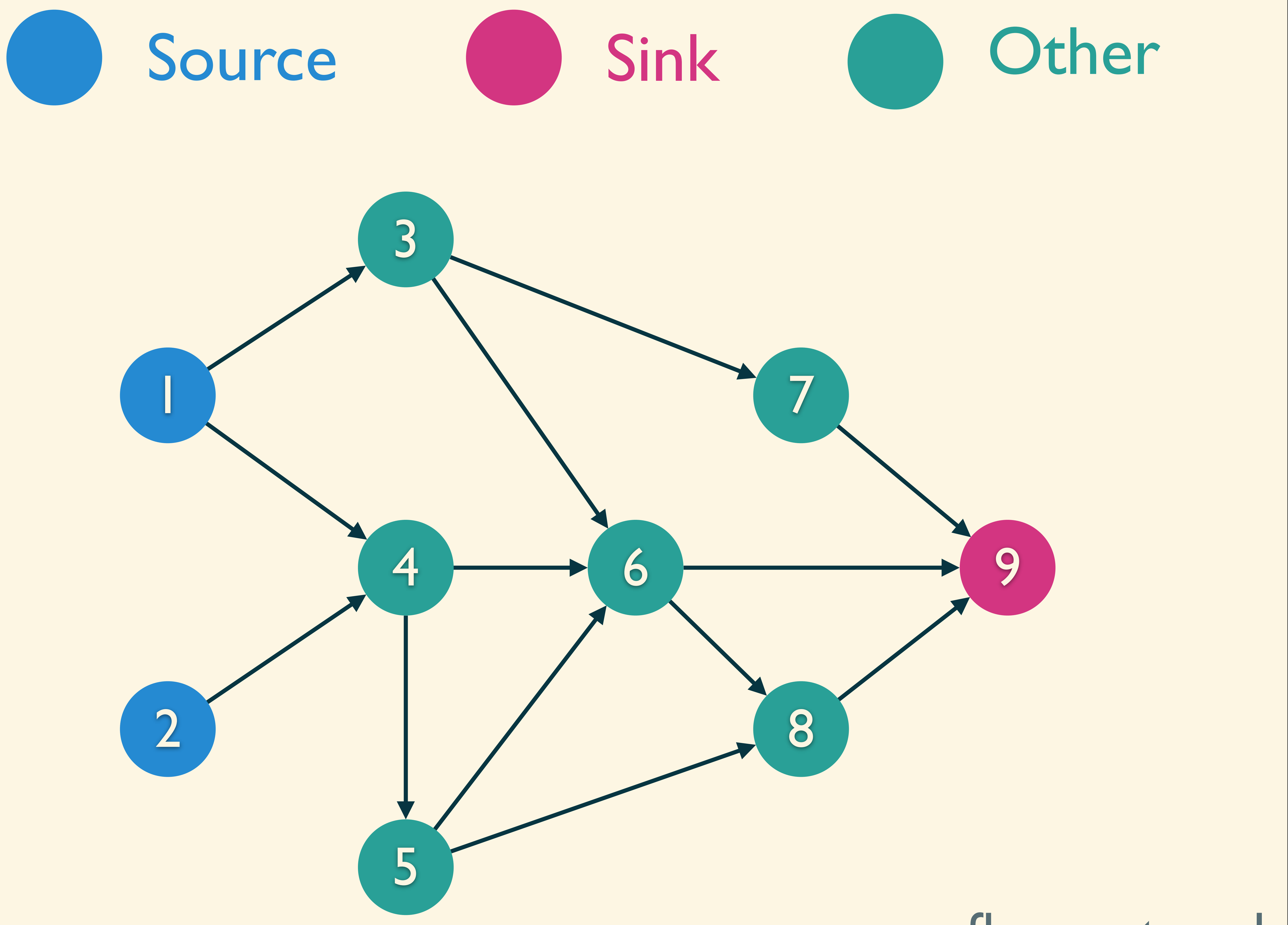
directed graph

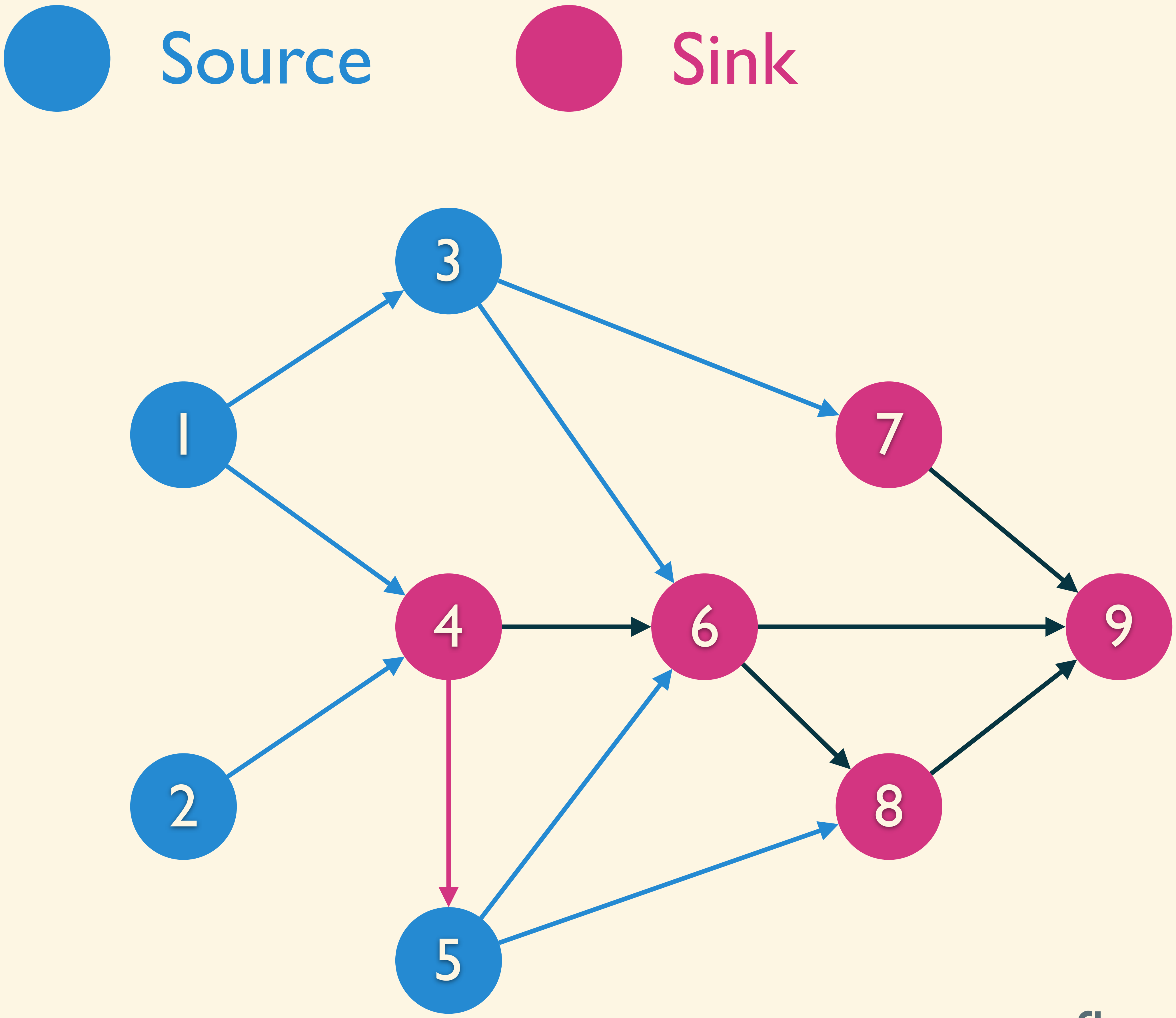
s-t cut

1. one side is **source**

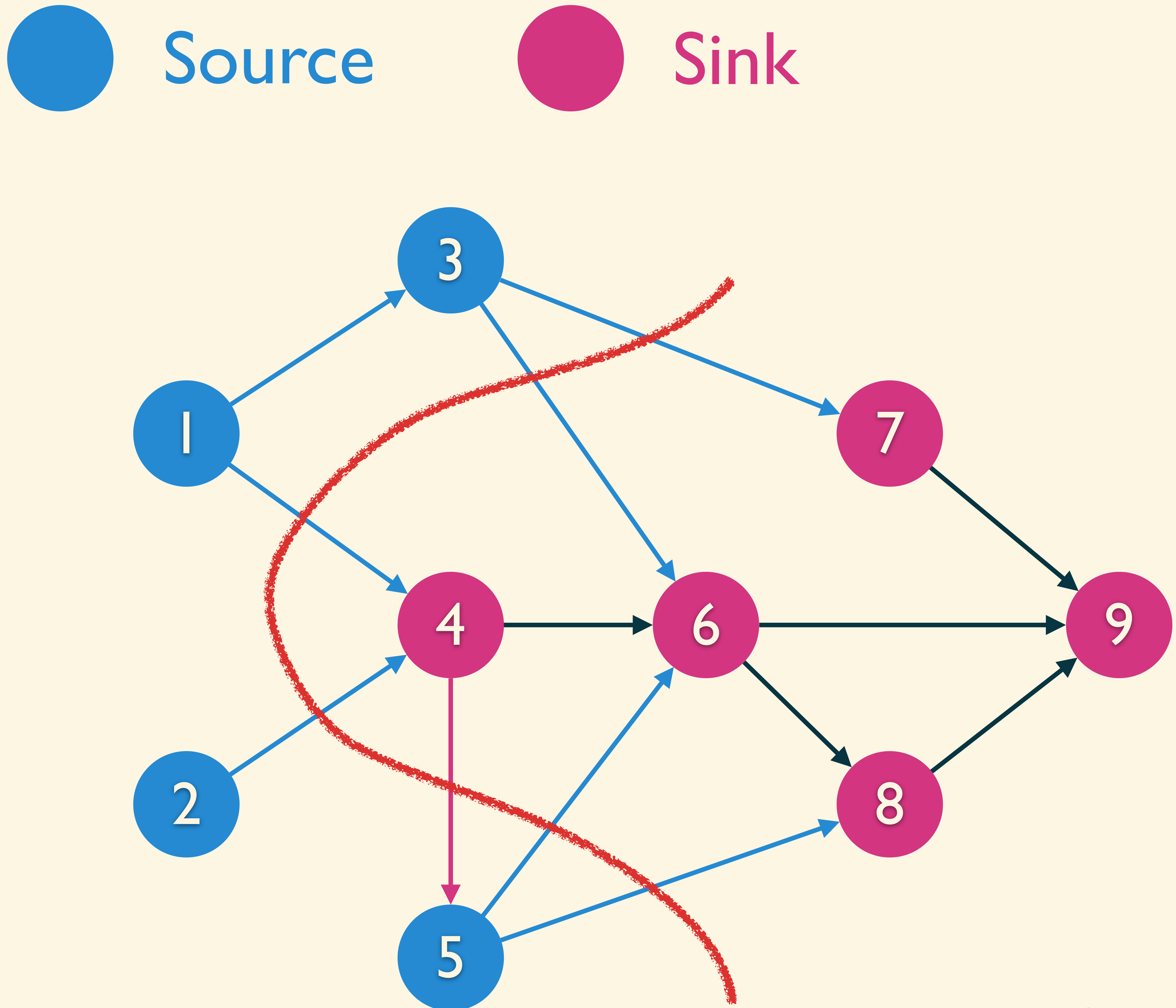
2. another side is **sink**

3. cut-set only consists of edges going **from source's side to sink's side**


Source
Sink
Other
1
2
3
4
5
6
7
8
9
flow network



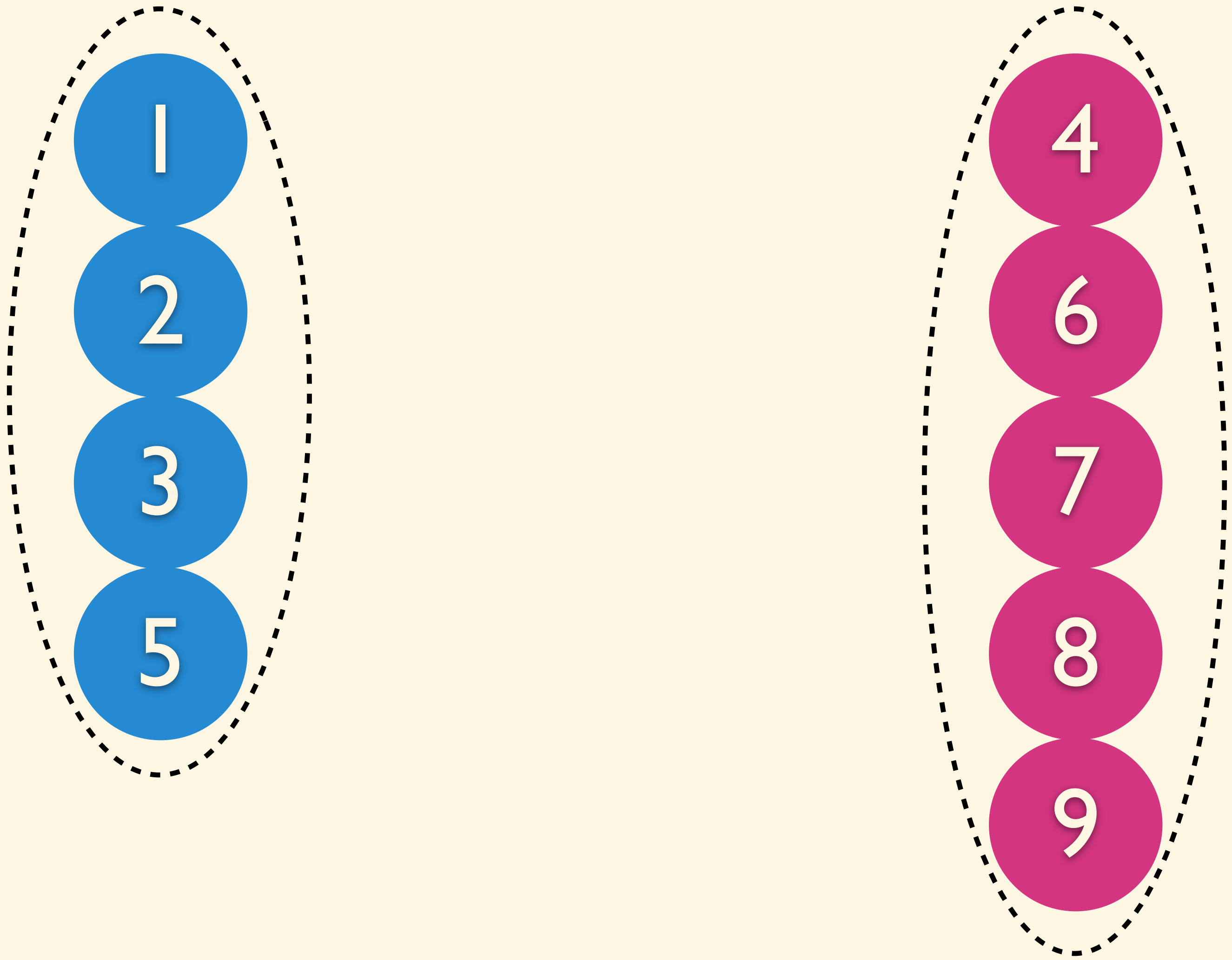
flow network



flow network

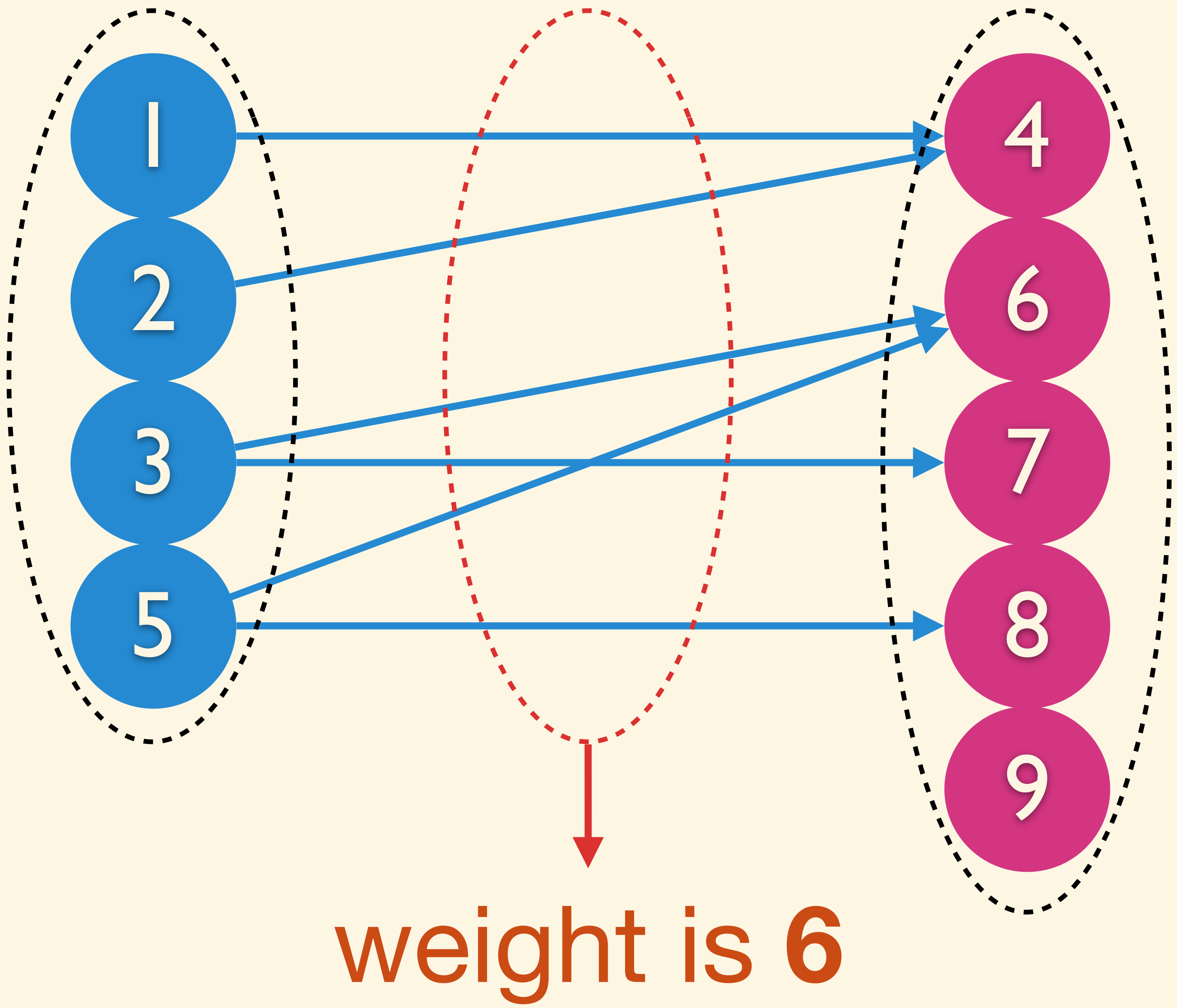
cut-set only consists of edges going

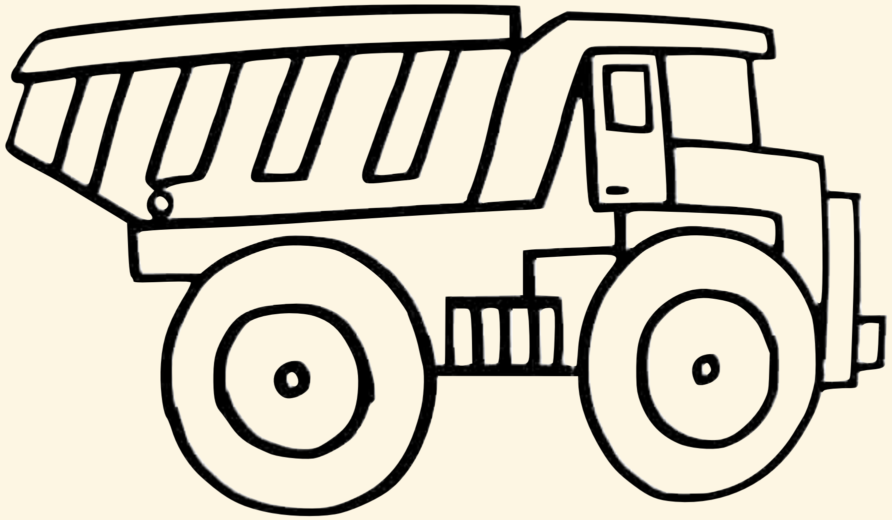
from source's side to sink's side



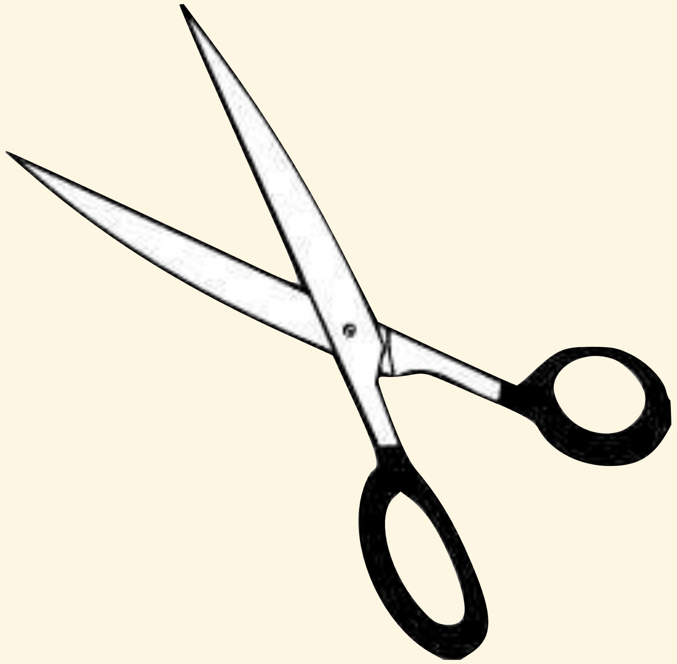
cut-set only consists of edges going

from source's side to sink's side



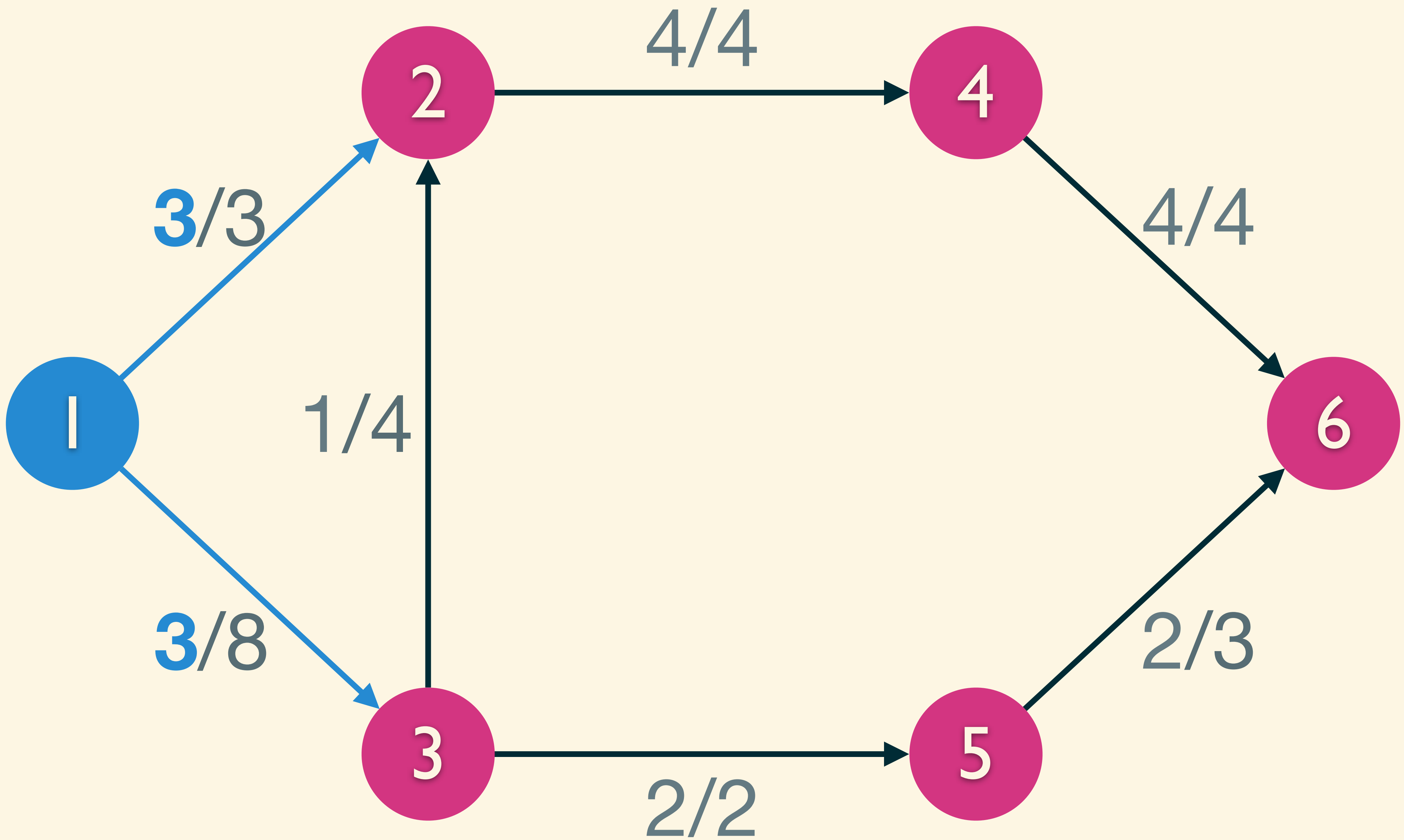


Max-Flow Min-Cut Theorem



Observation 1

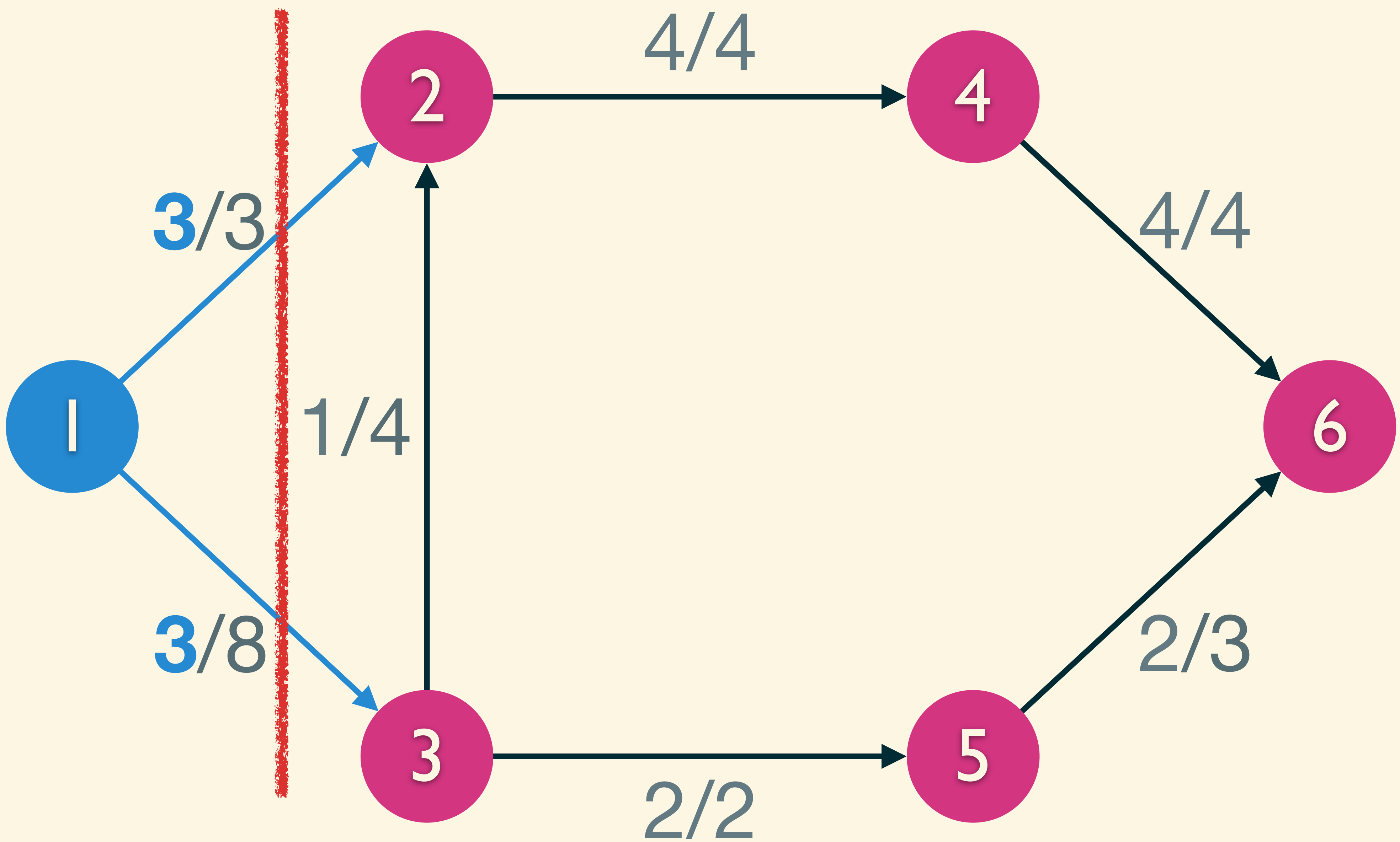
The network flow sent across any cut is equal to the amount reaching sink.



total flow = 6, flow on cut = 3 + 3 = 6

Observation 1

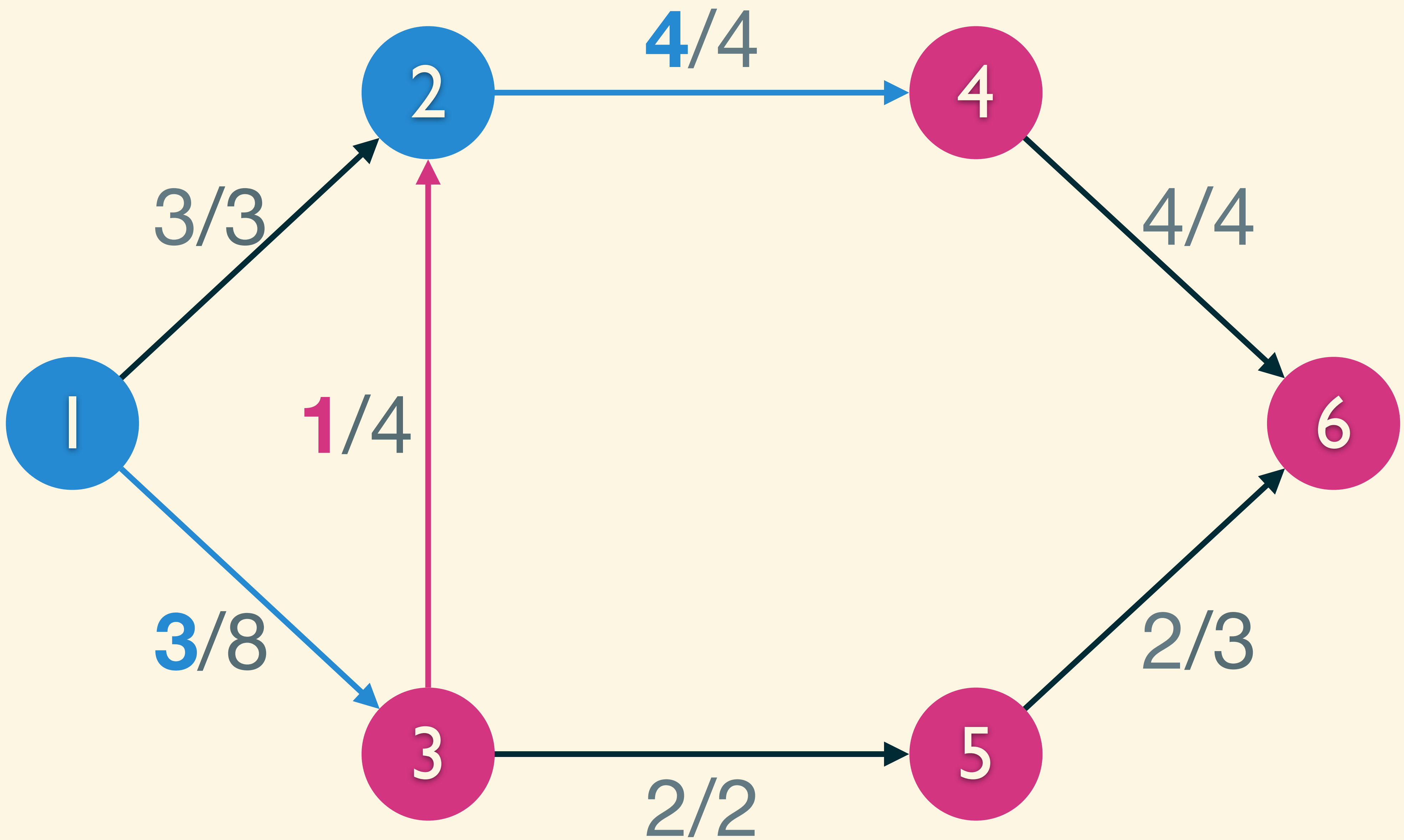
The network flow sent across any cut is equal to the amount reaching sink.



total flow = 6, flow on cut = 3 + 3 = 6

Observation 1

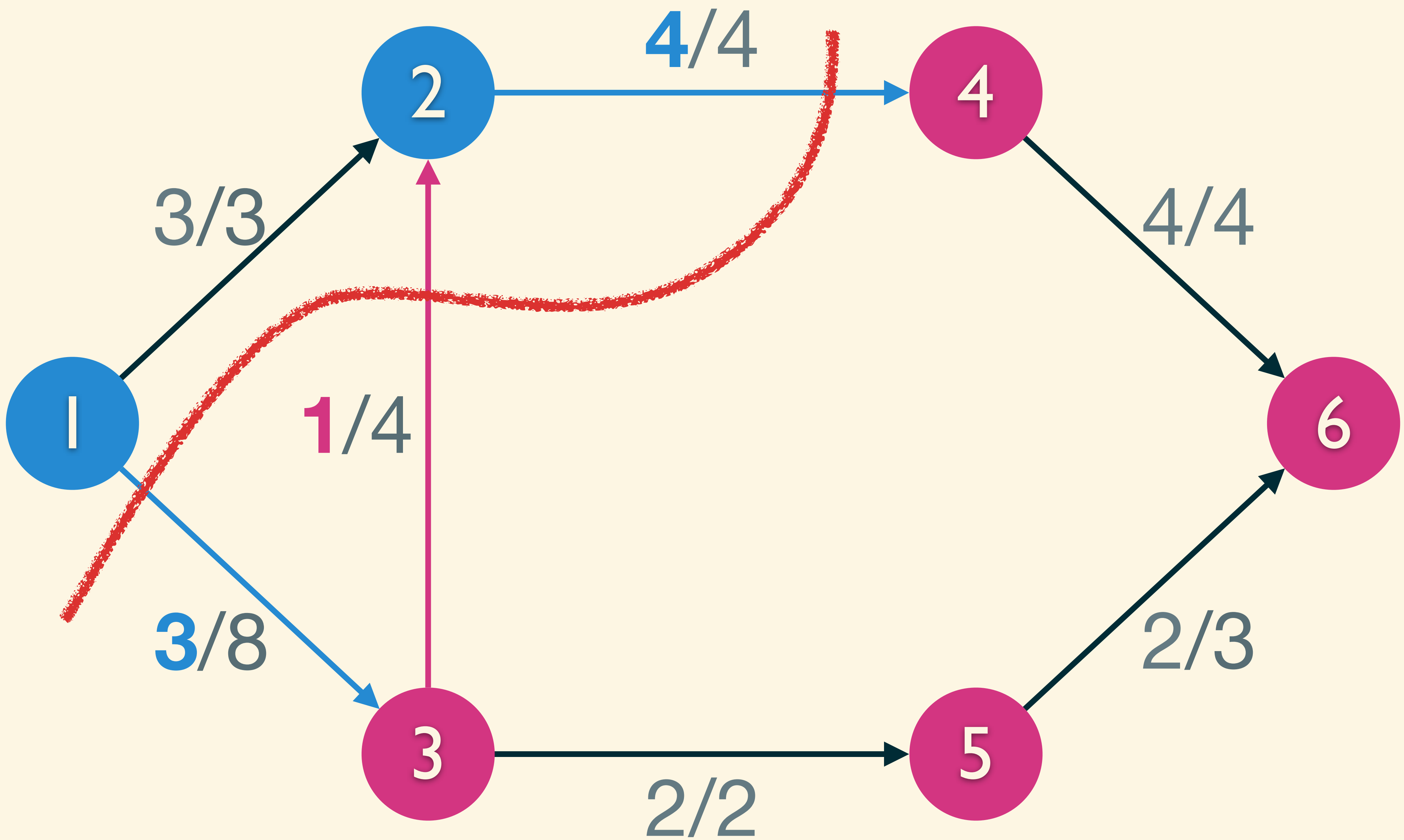
The network flow sent across any cut is equal to the amount reaching sink.



total flow = 6, flow on cut = 3 + 4 - 1 = 6

Observation 1

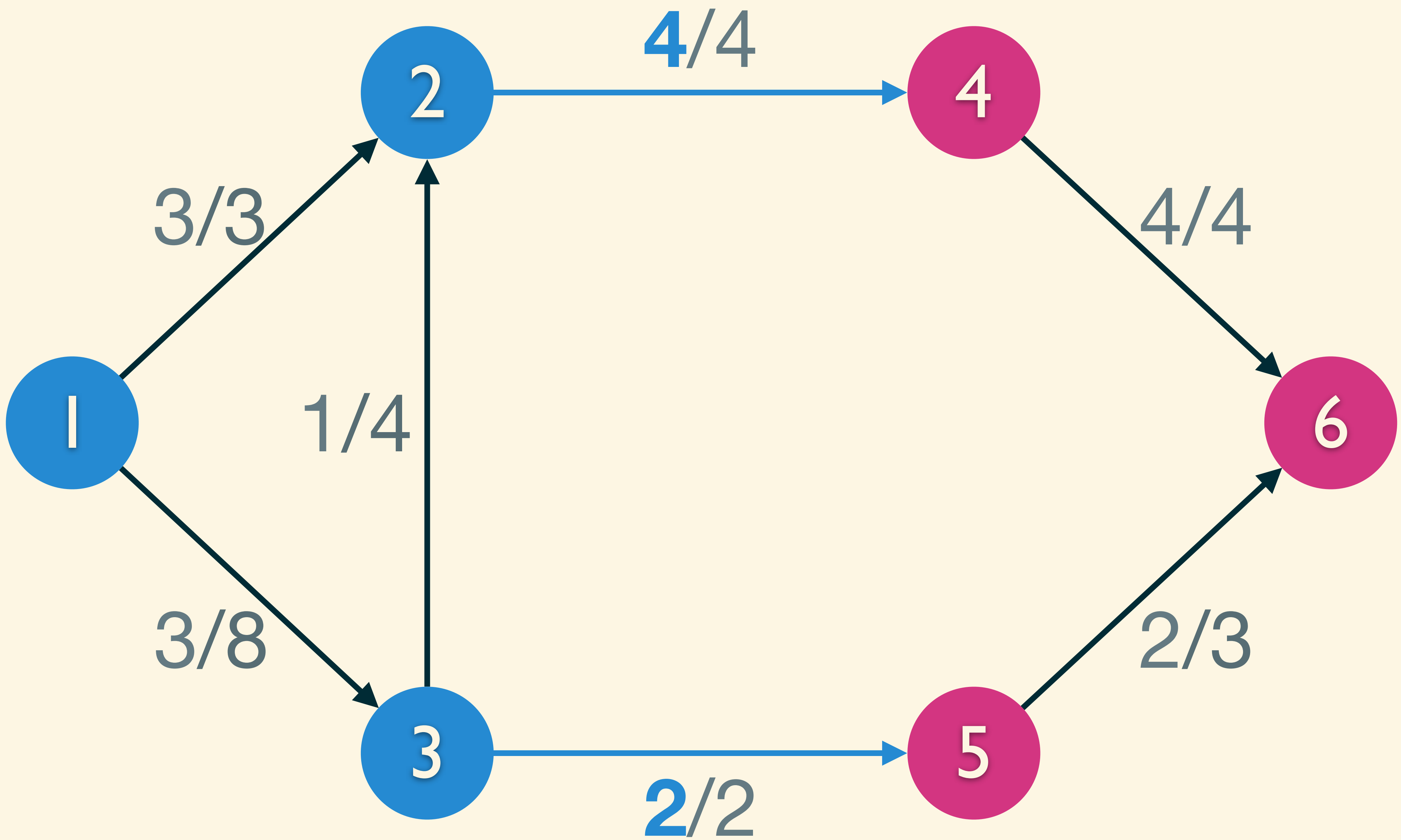
The network flow sent across any cut is equal to the amount reaching sink.



total flow = 6, flow on cut = 3 + 4 - 1 = 6

Observation 1

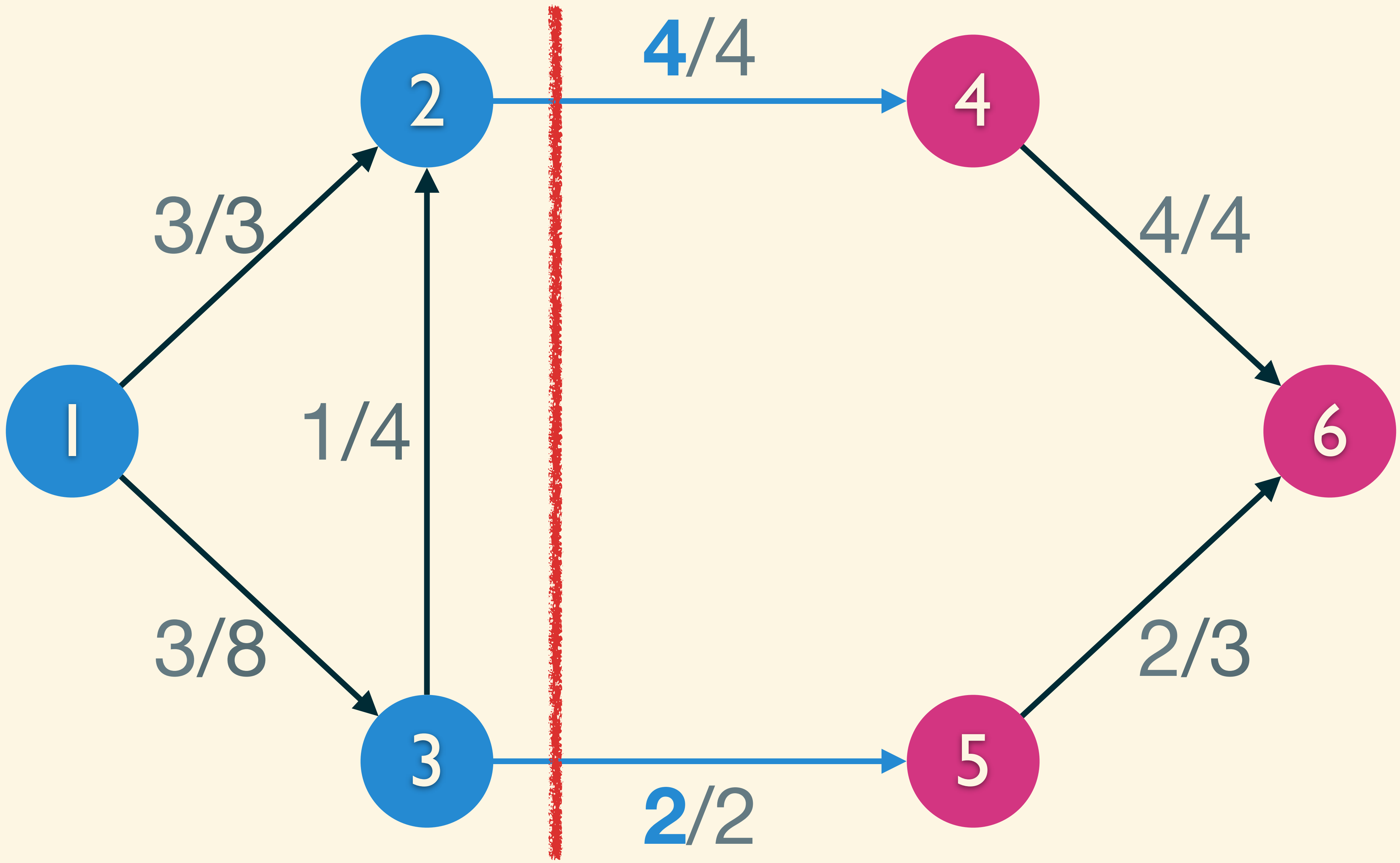
The network flow sent across any cut is equal to the amount reaching sink.



total flow = 6, flow on cut = 4 + 2= 6

Observation 1

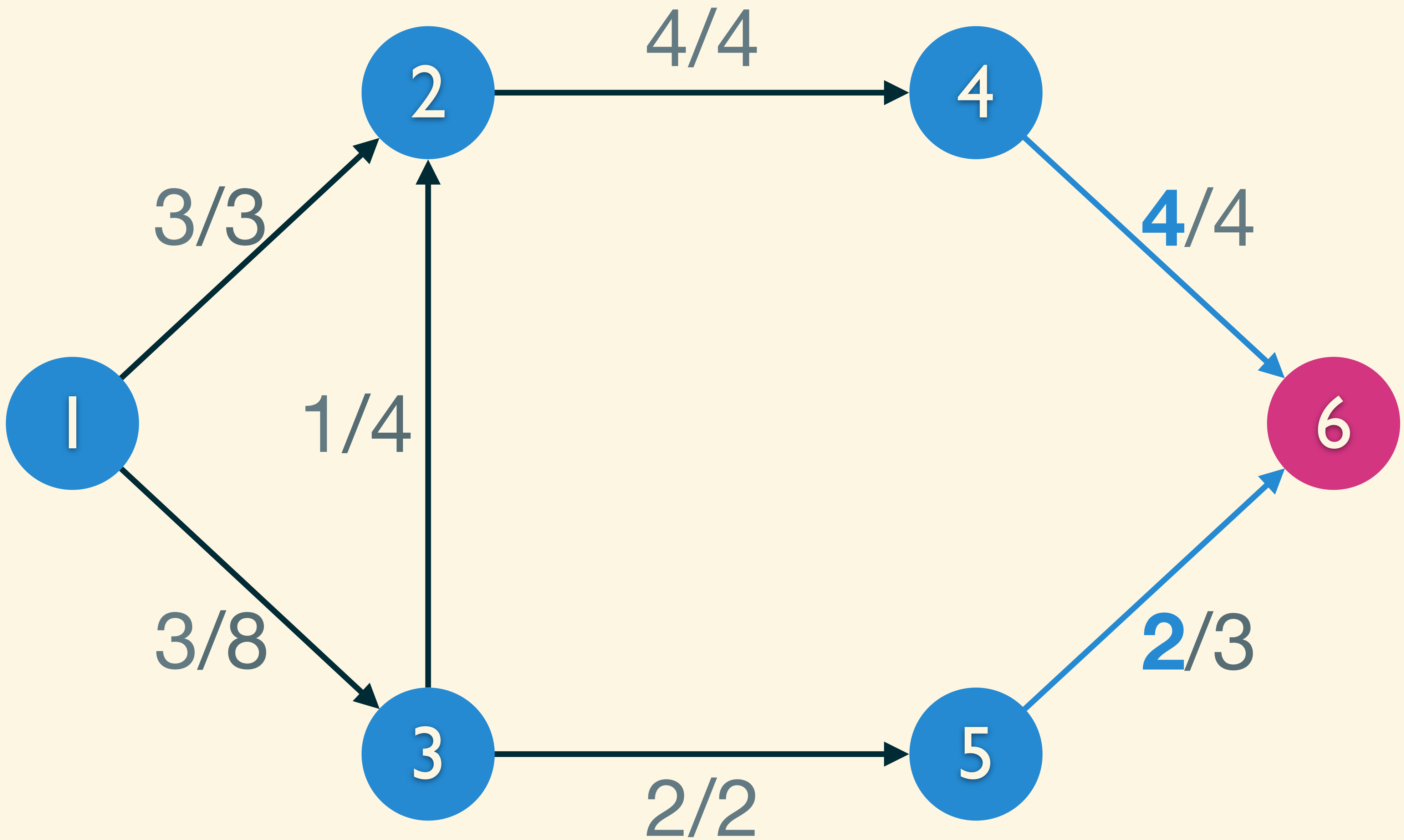
The network flow sent across any cut is equal to the amount reaching sink.



total flow = 6, flow on cut = 4 + 2= 6

Observation 1

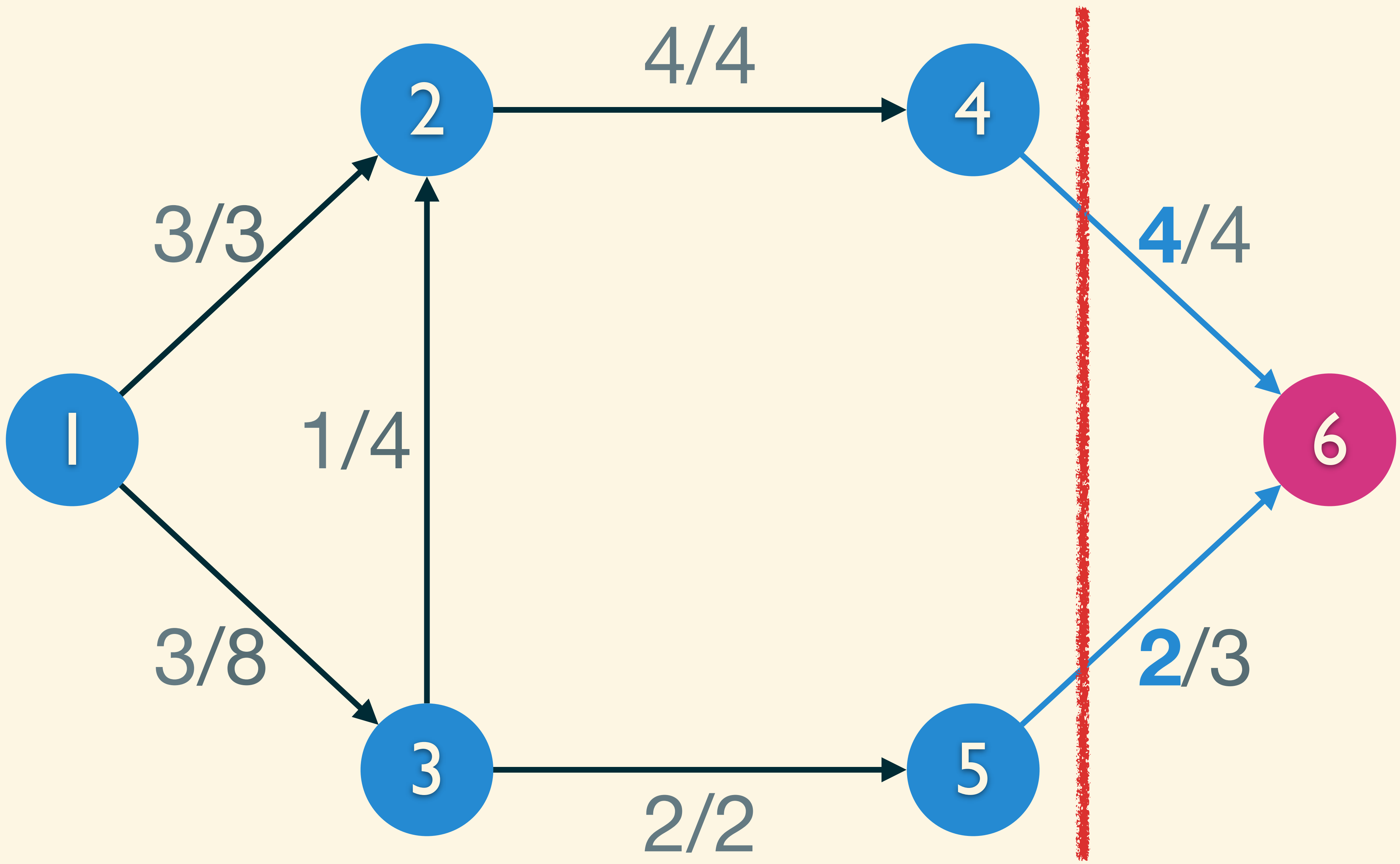
The network flow sent across any cut is equal to the amount reaching sink.



total flow = 6, flow on cut = 4 + 2= 6

Observation 1

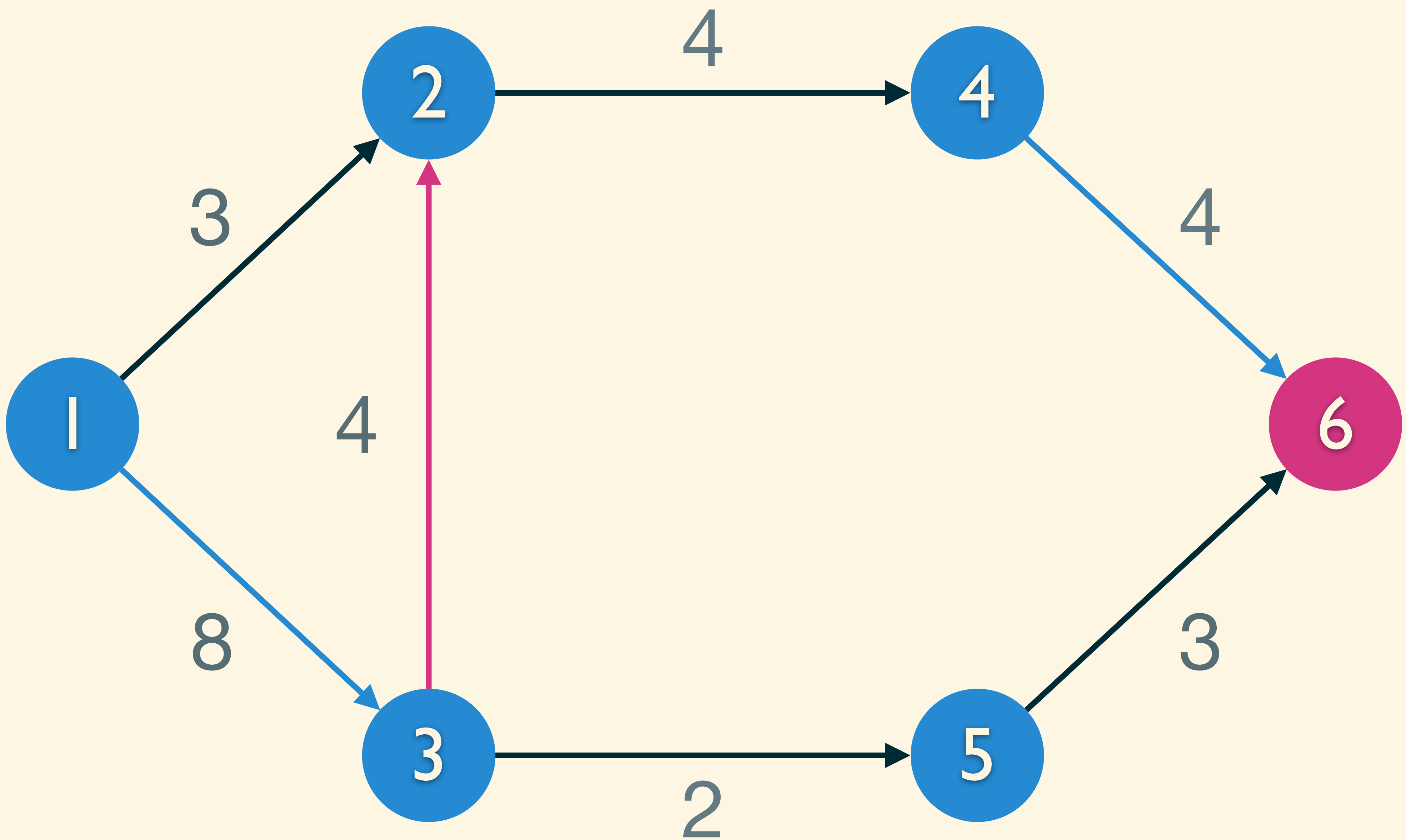
The network flow sent across any cut is equal to the amount reaching sink.



total flow = 6, flow on cut = 4 + 2= 6

Observation 2

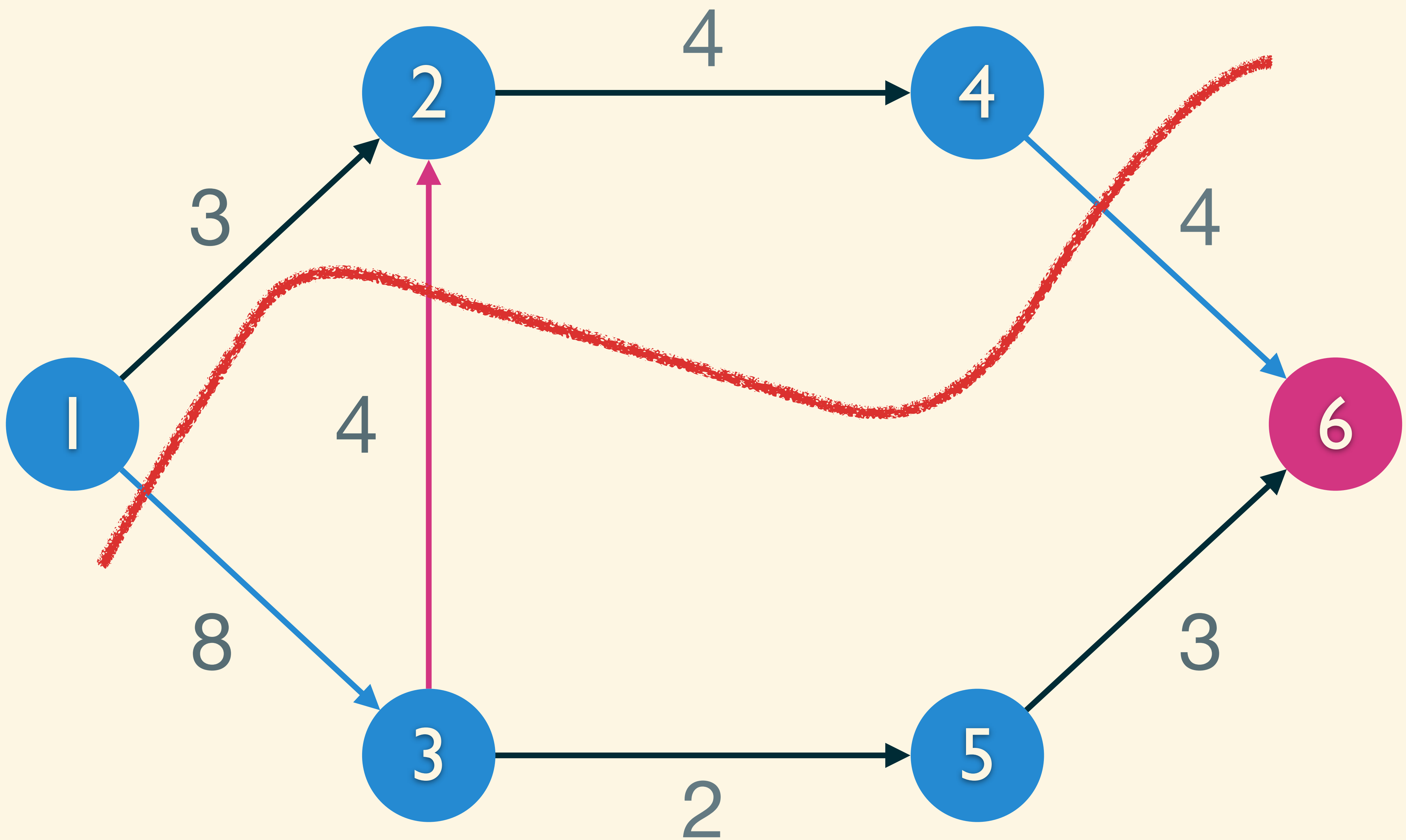
Then the value of the flow is at most the capacity of any cut.



It's trivial!

Observation 2

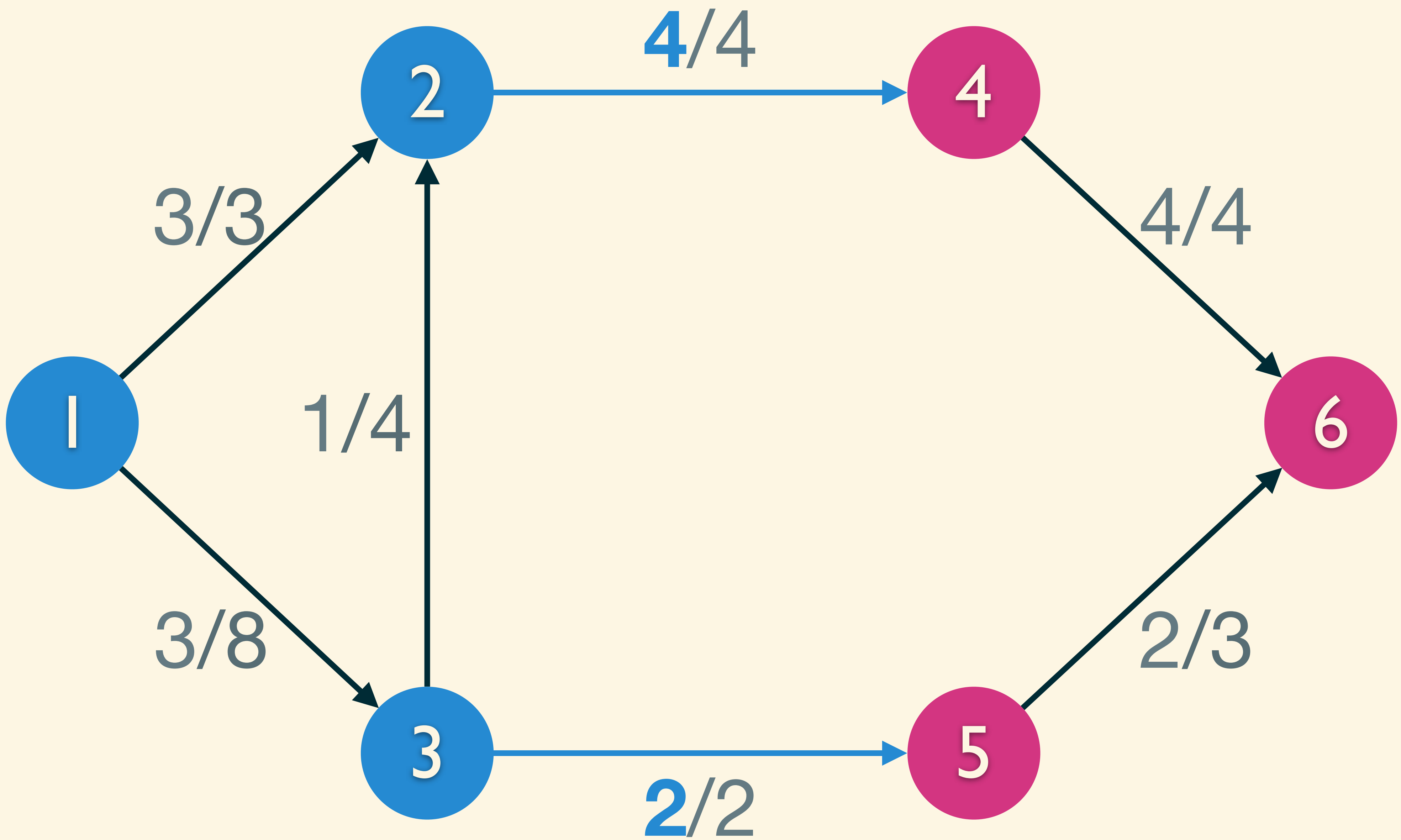
Then the value of the flow is at most the capacity of any cut.



It's trivial!

Observation 3

Let f be a flow, and let (S,T) be an s-t cut whose capacity equals the value of f.

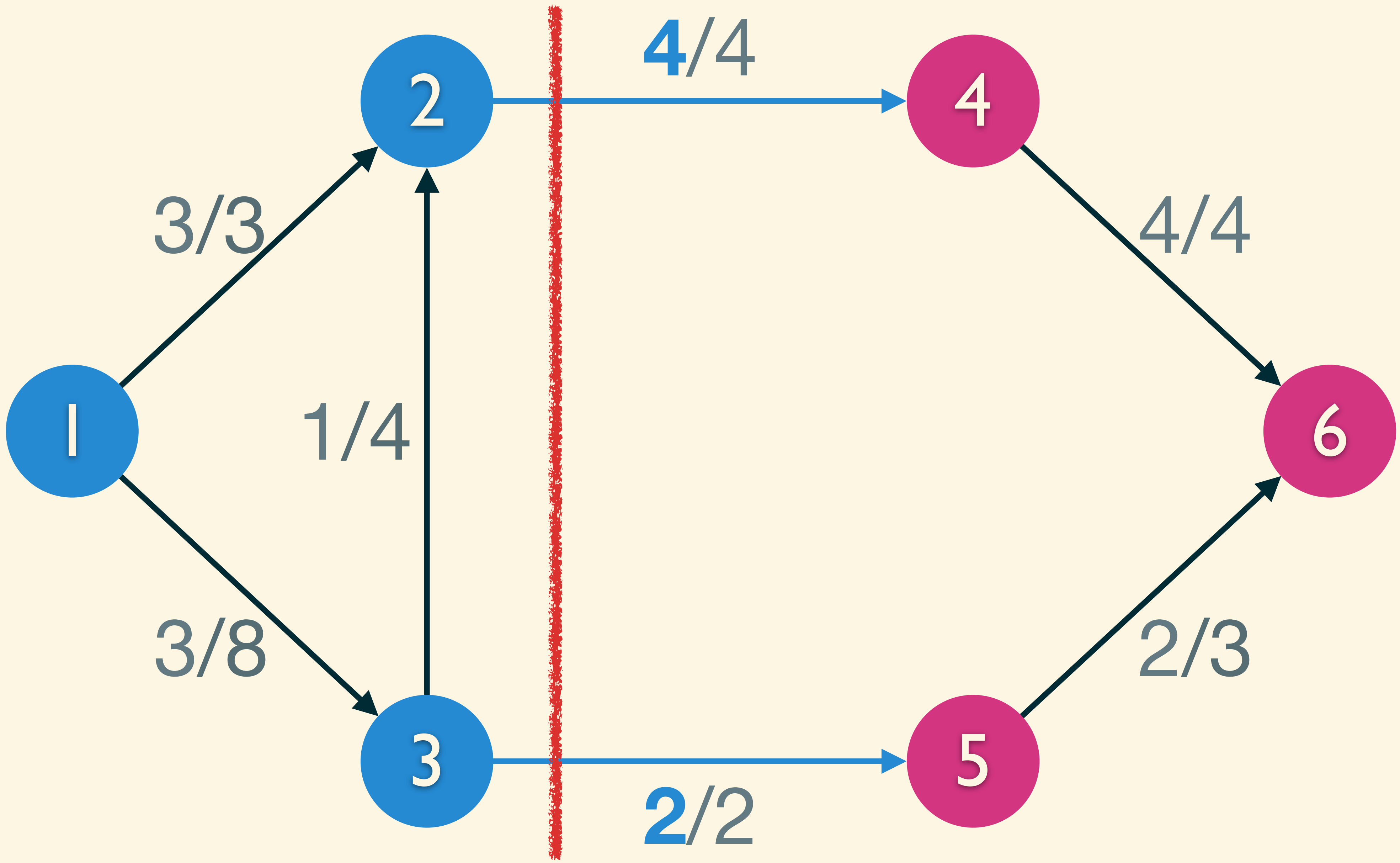


f is the maximum flow

(S,T) is the minimum cut

Observation 3

Let f be a flow, and let (S,T) be an s-t cut whose capacity equals the value of f.



f is the maximum flow

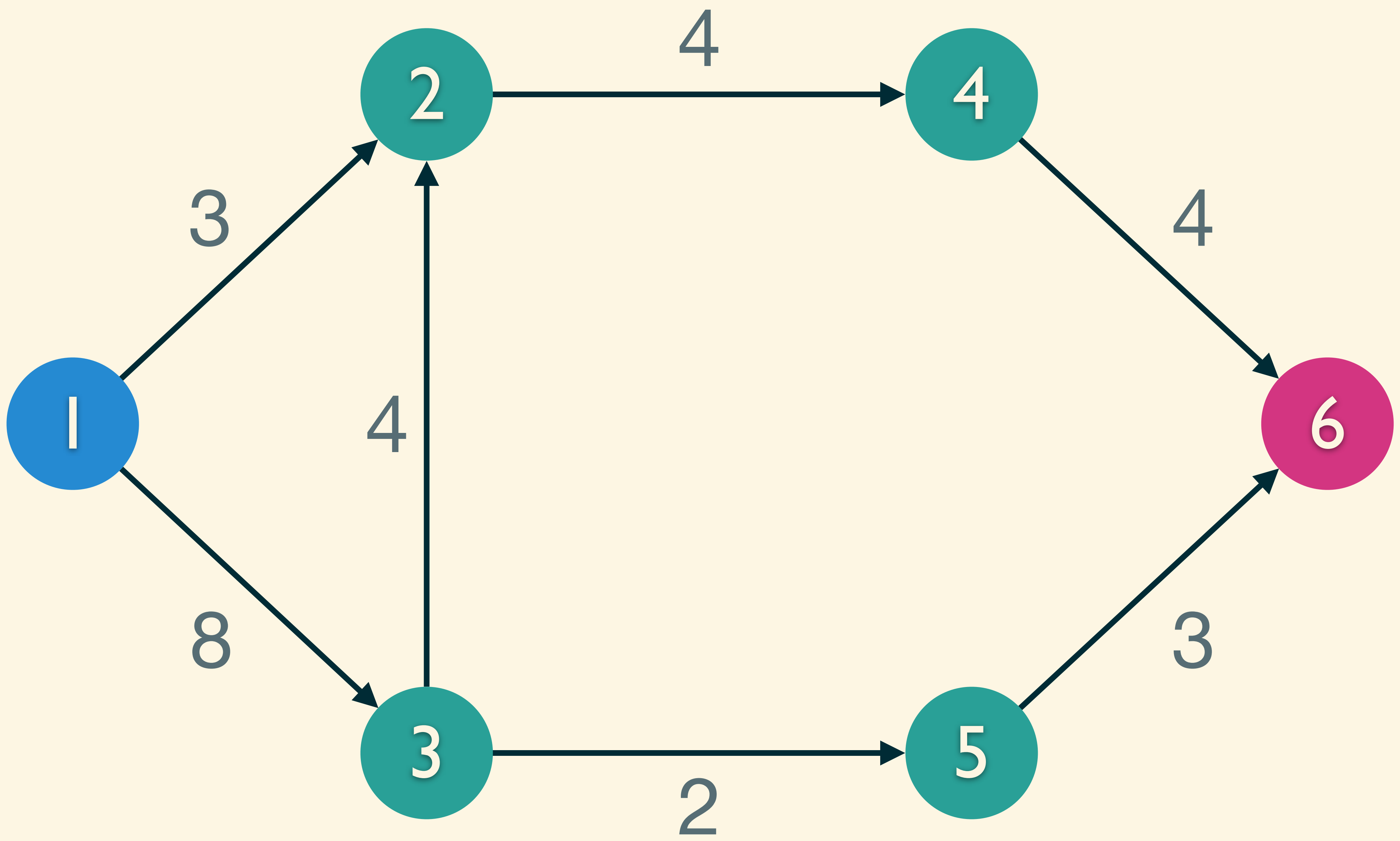
(S,T) is the minimum cut

Max-Flow

EQUAL

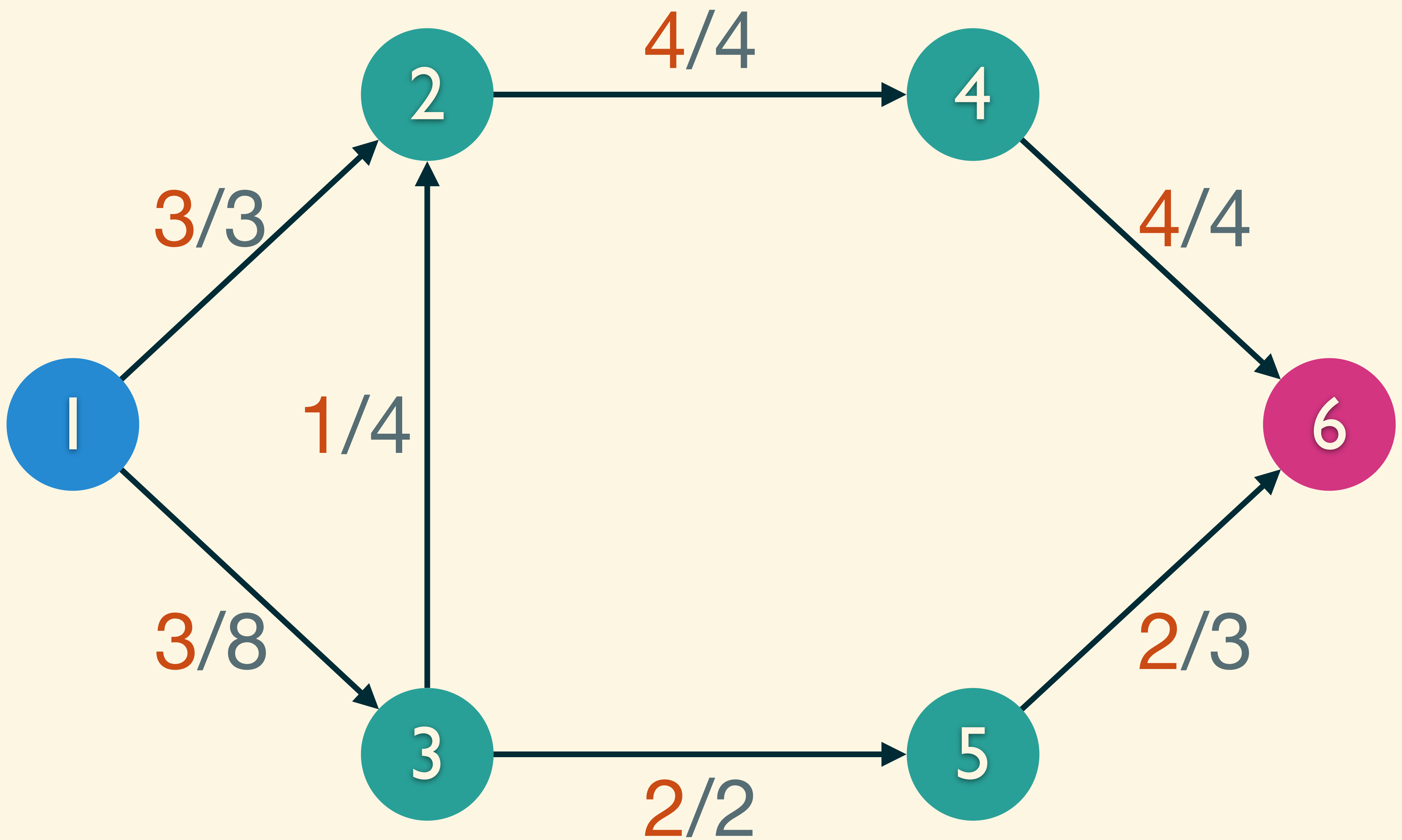
Min-Cut

Example

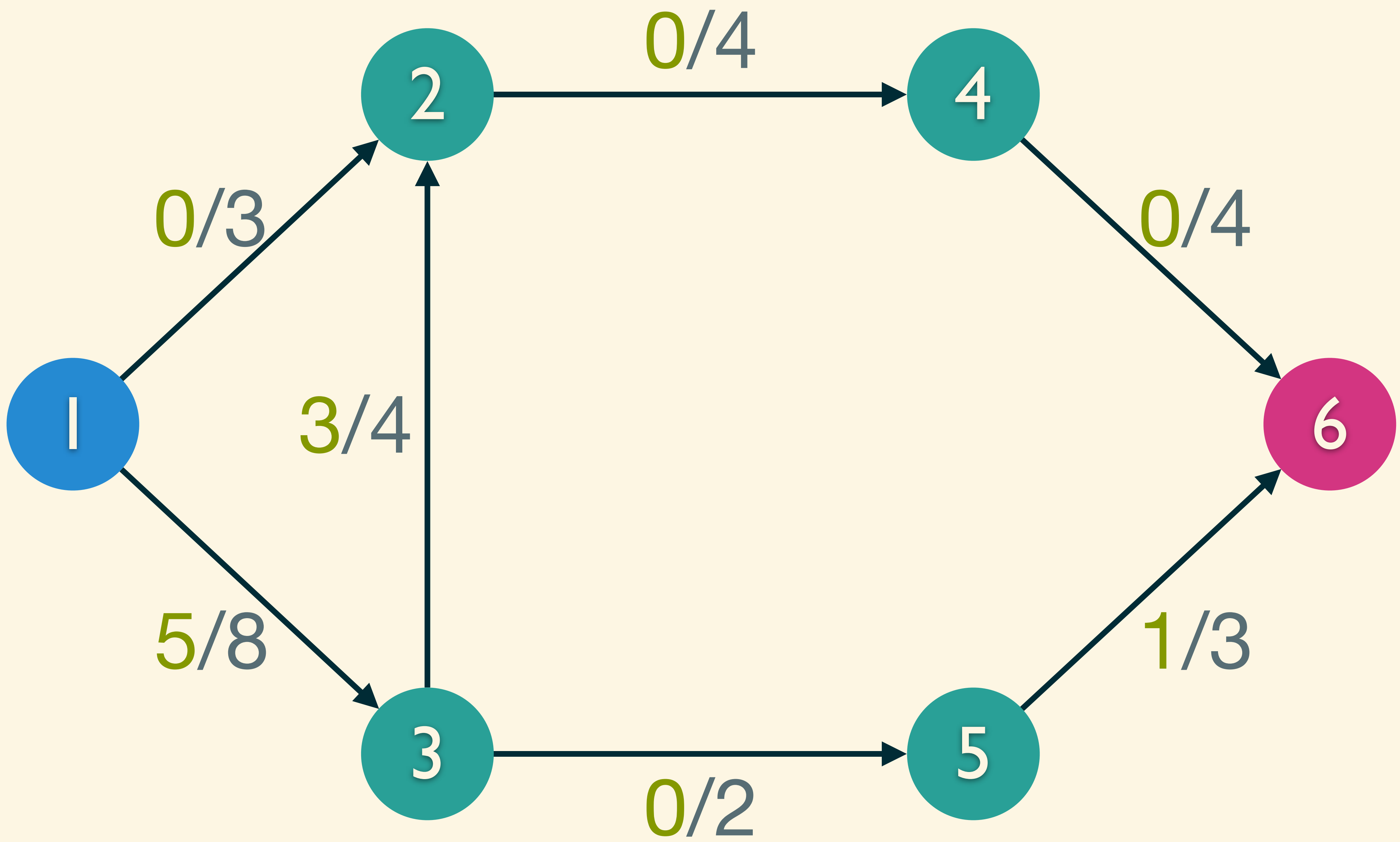


1
2
3
4
5
6
3
8
4
4
2
4
3

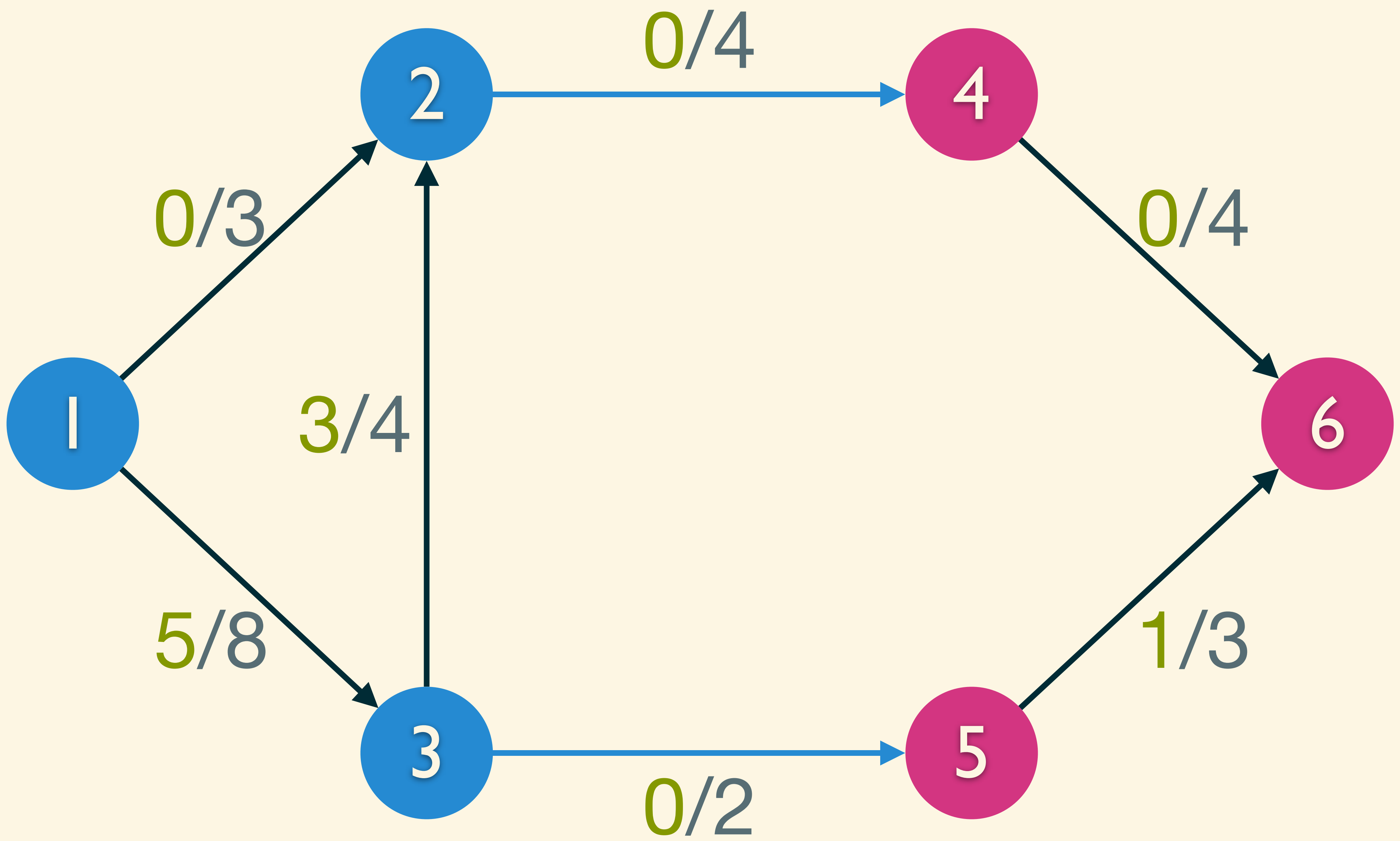
Maximum Flow = 6



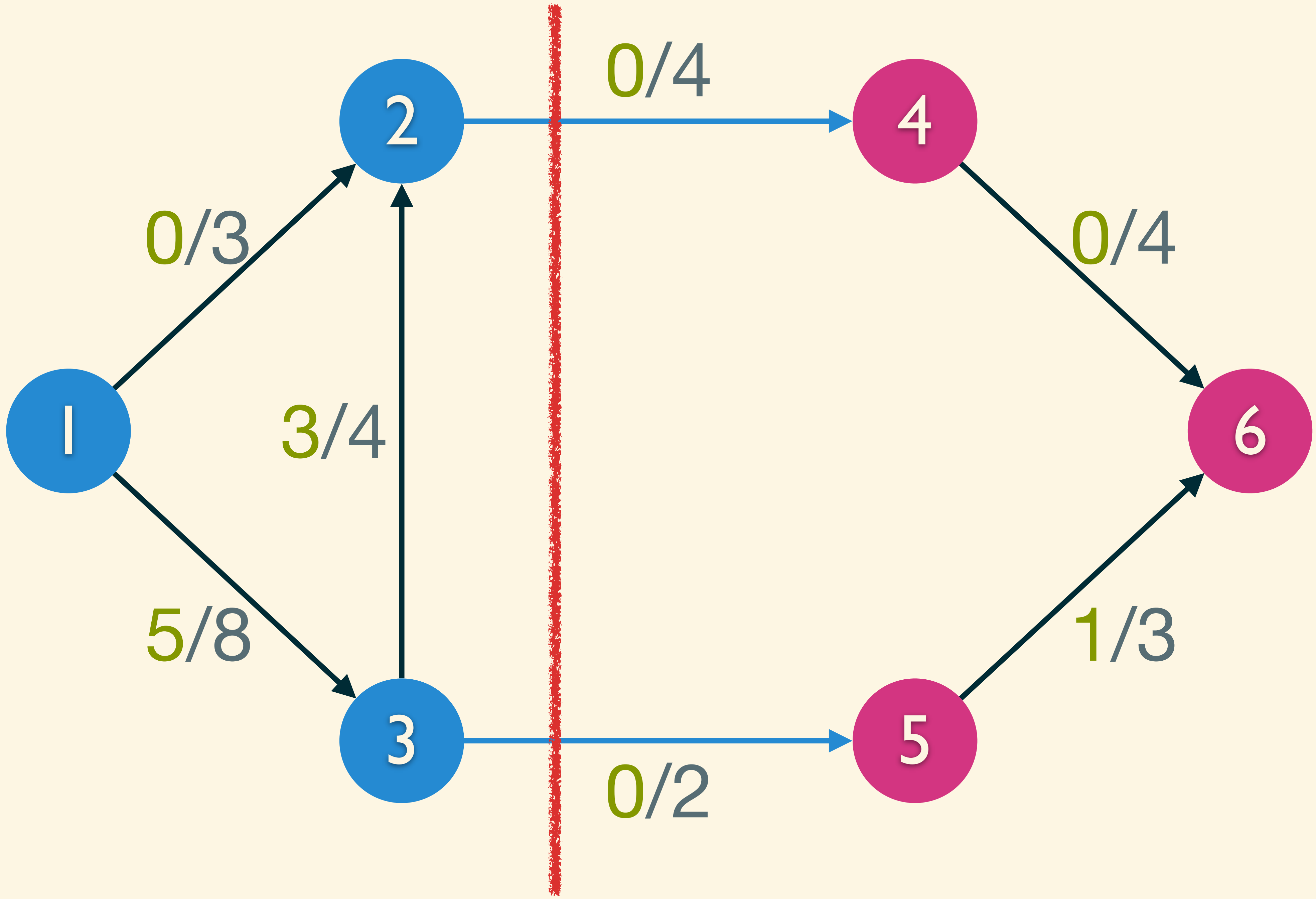
Residual Network



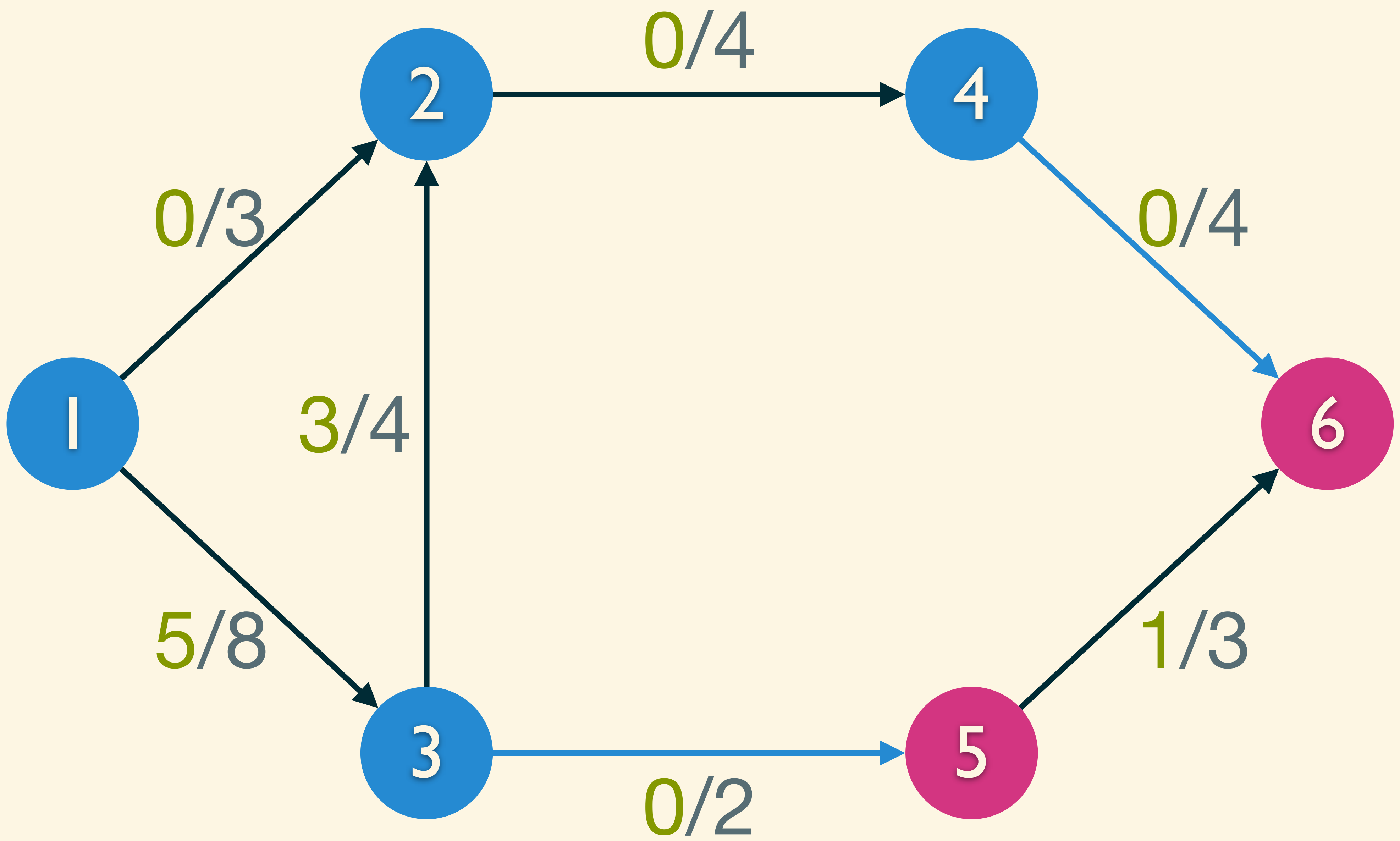
Minimum Cut = 6



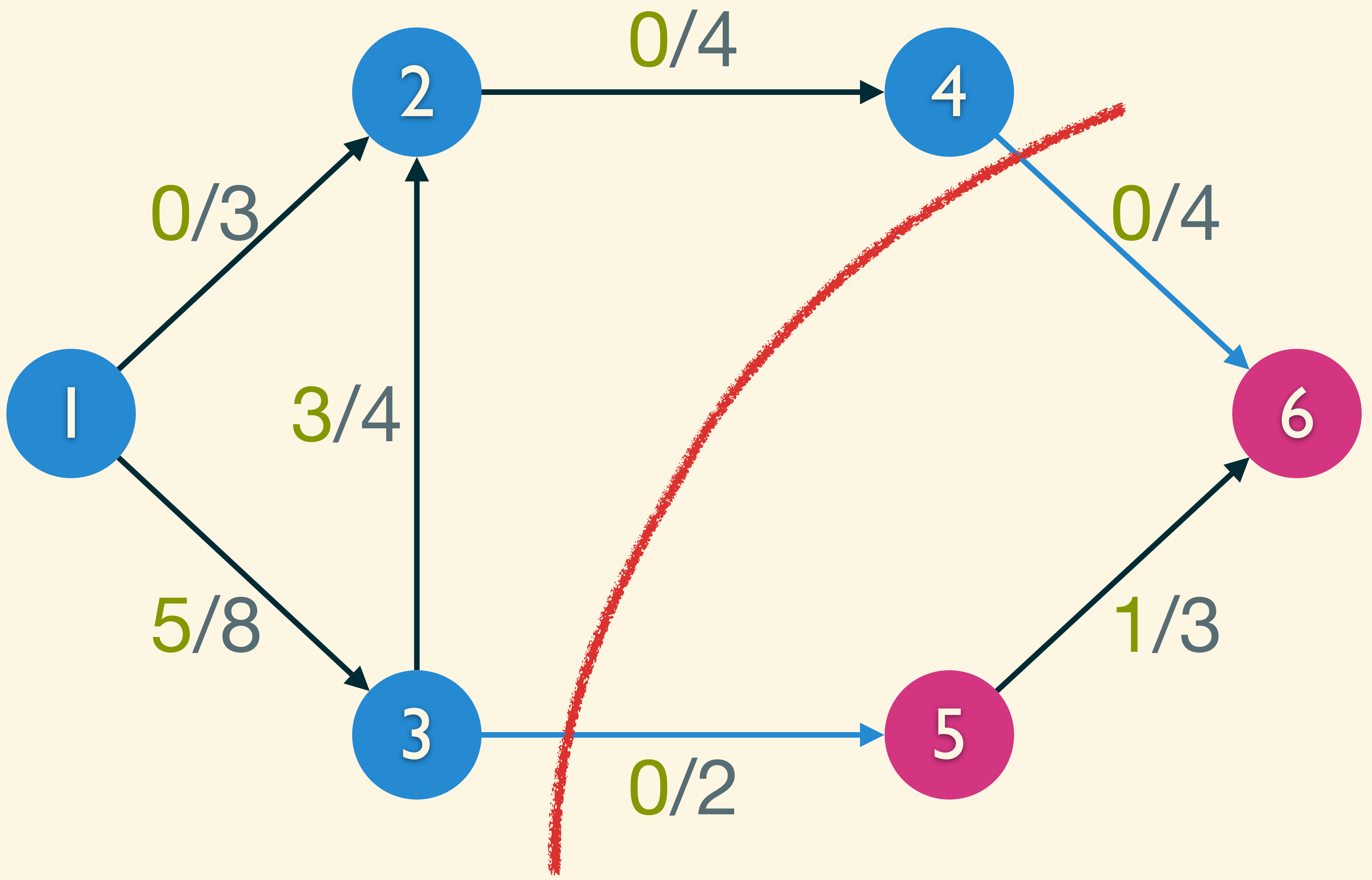
Minimum Cut = 6

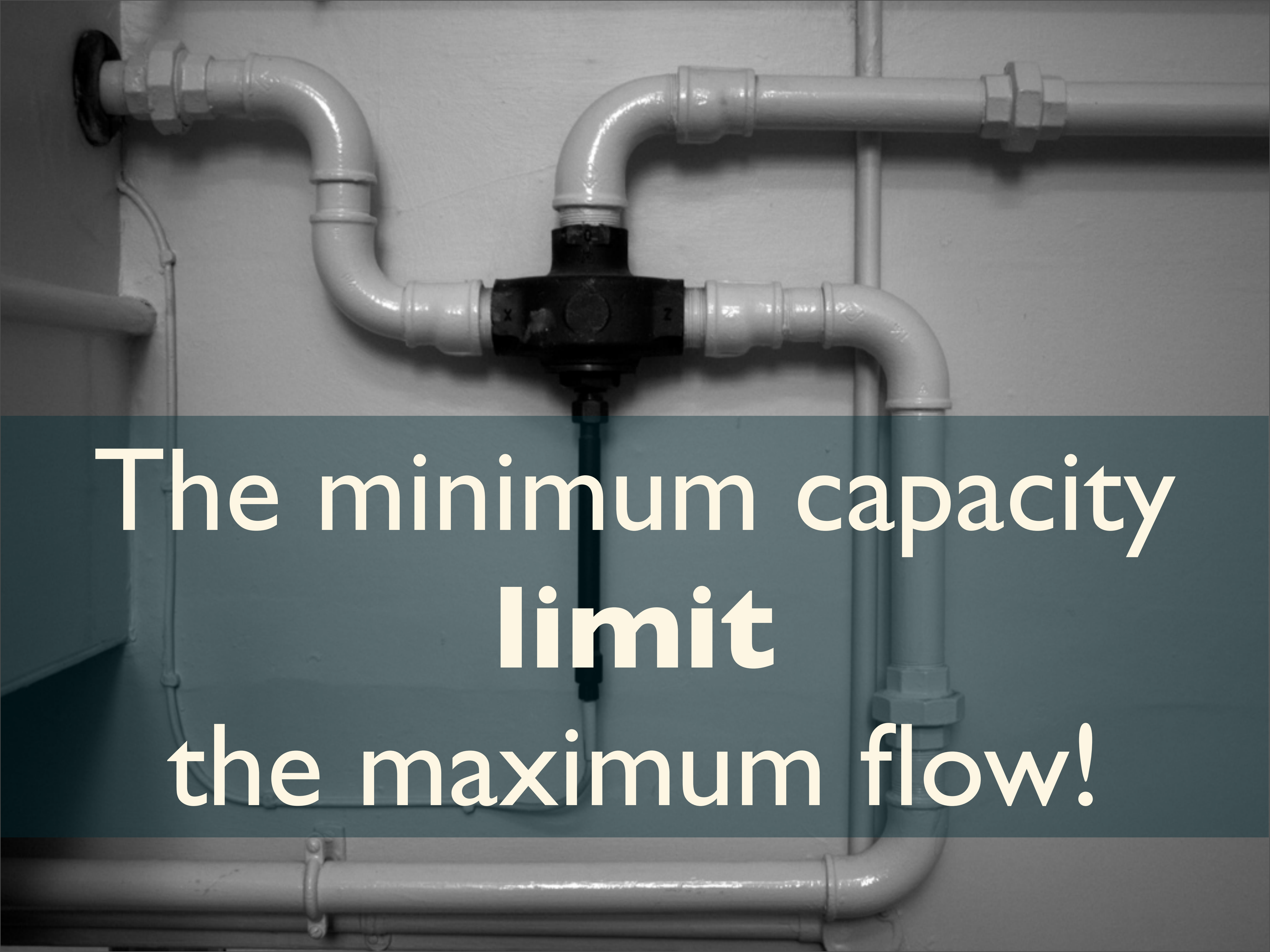


Minimum Cut = 6



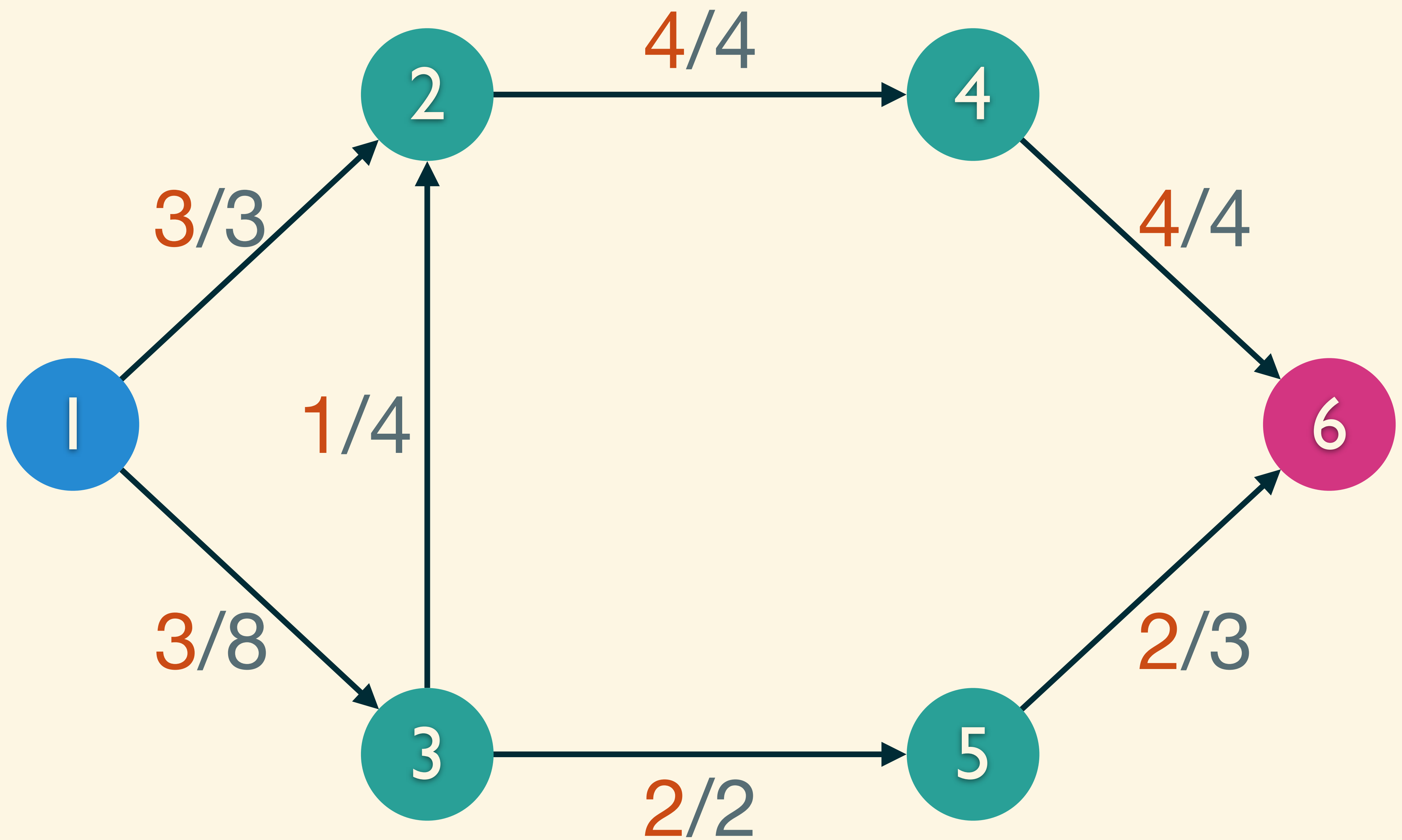
Minimum Cut = 6



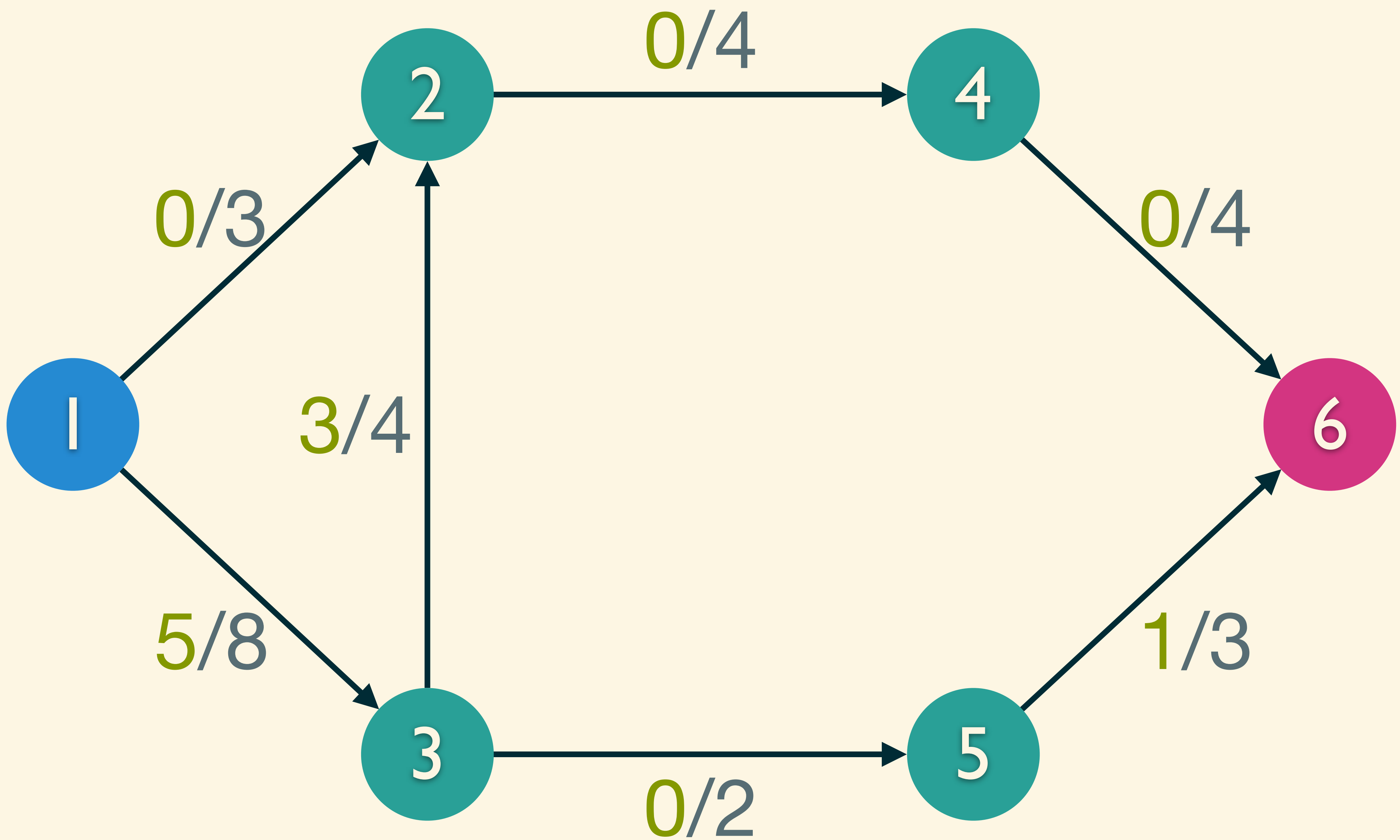

The minimum capacity
limit
the maximum flow!

find a s-t cut

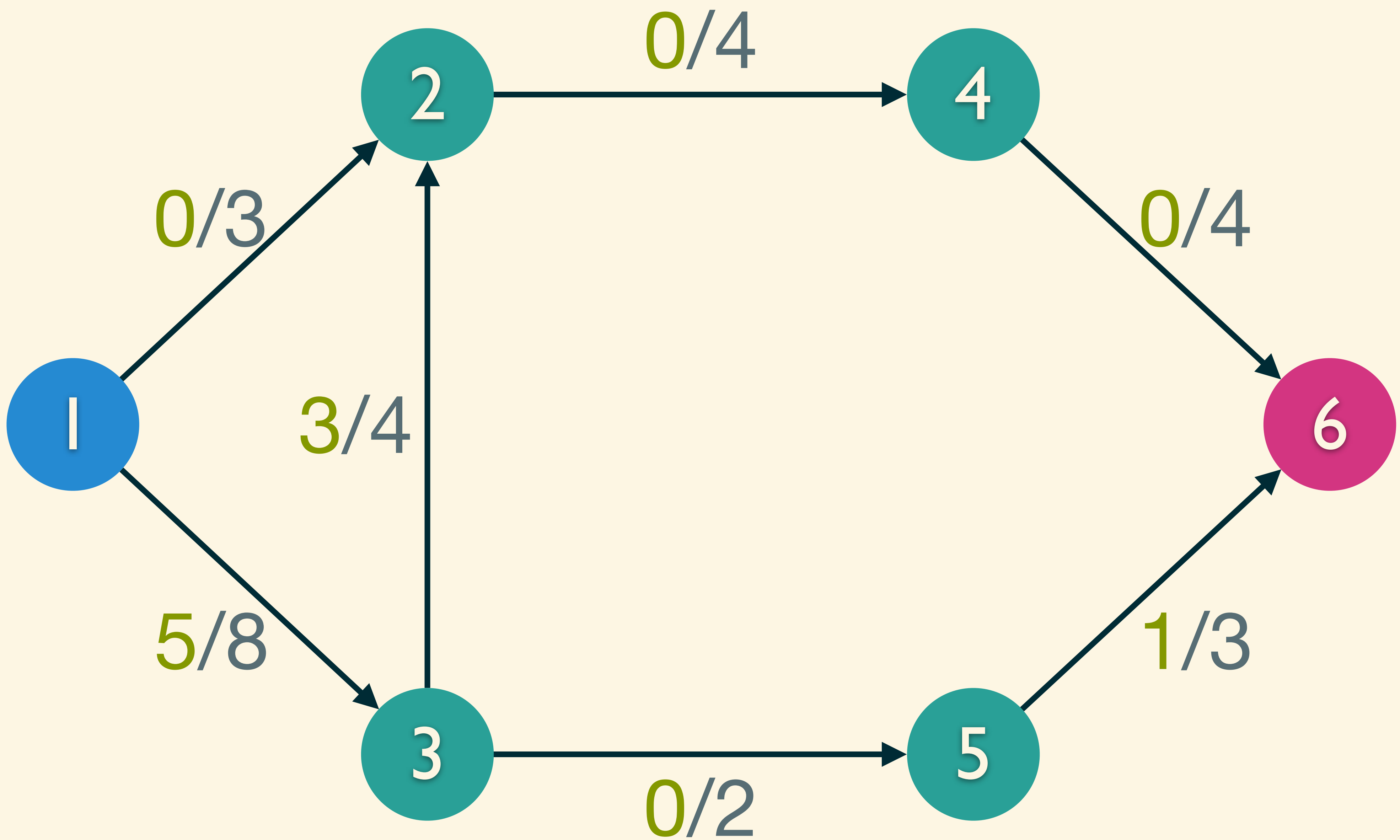
Maximum Flow = 6



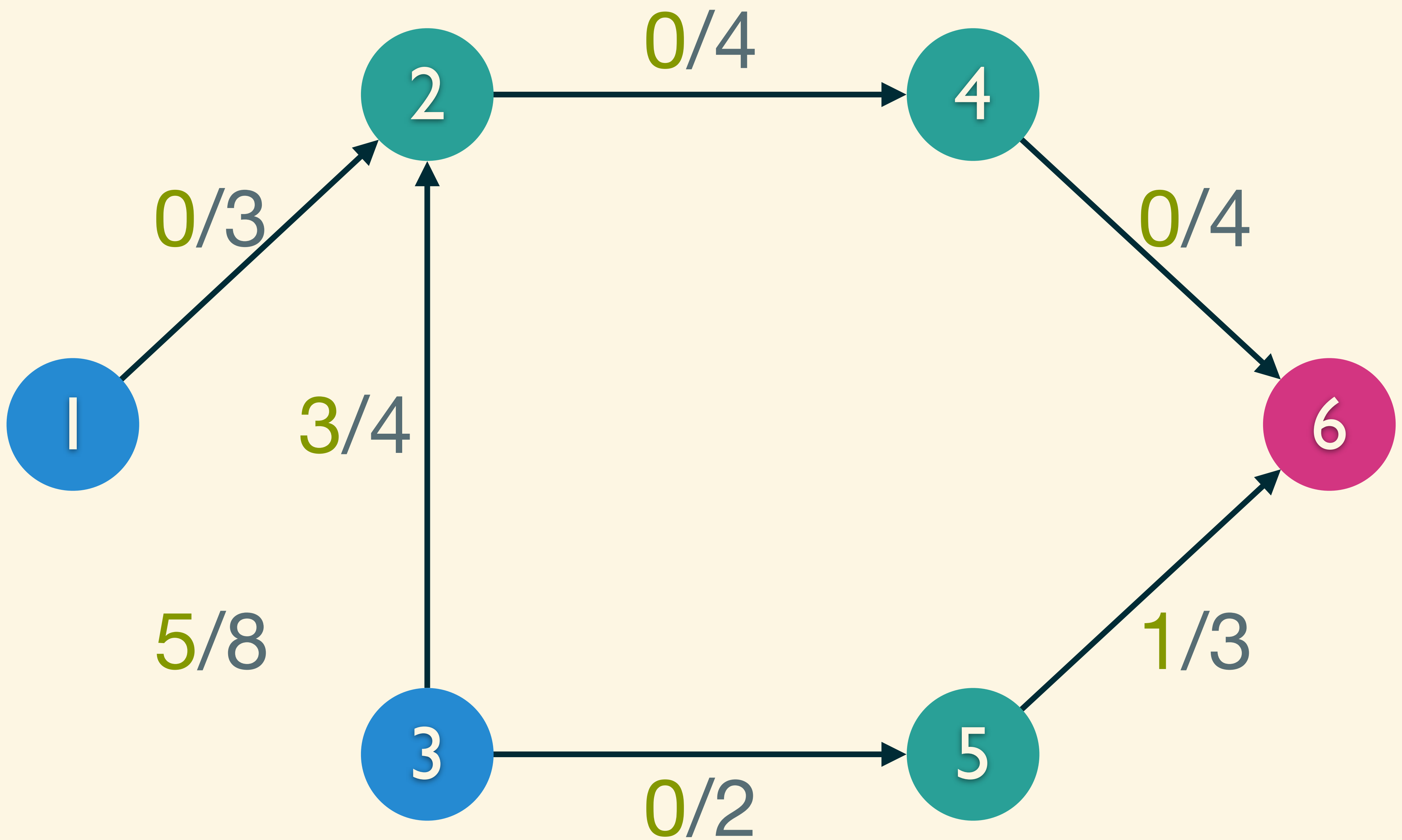
Travel on Residual Network



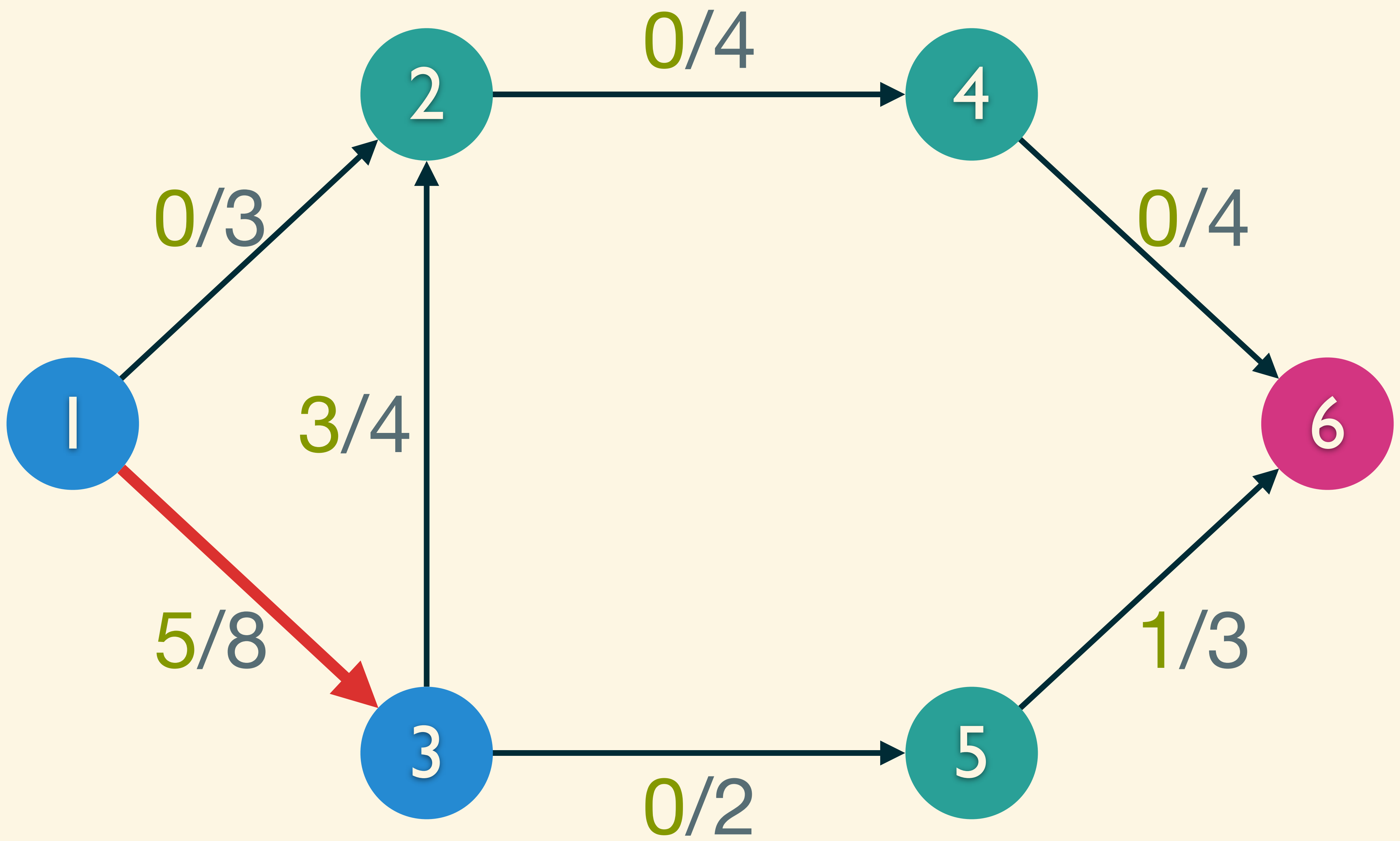
start from source



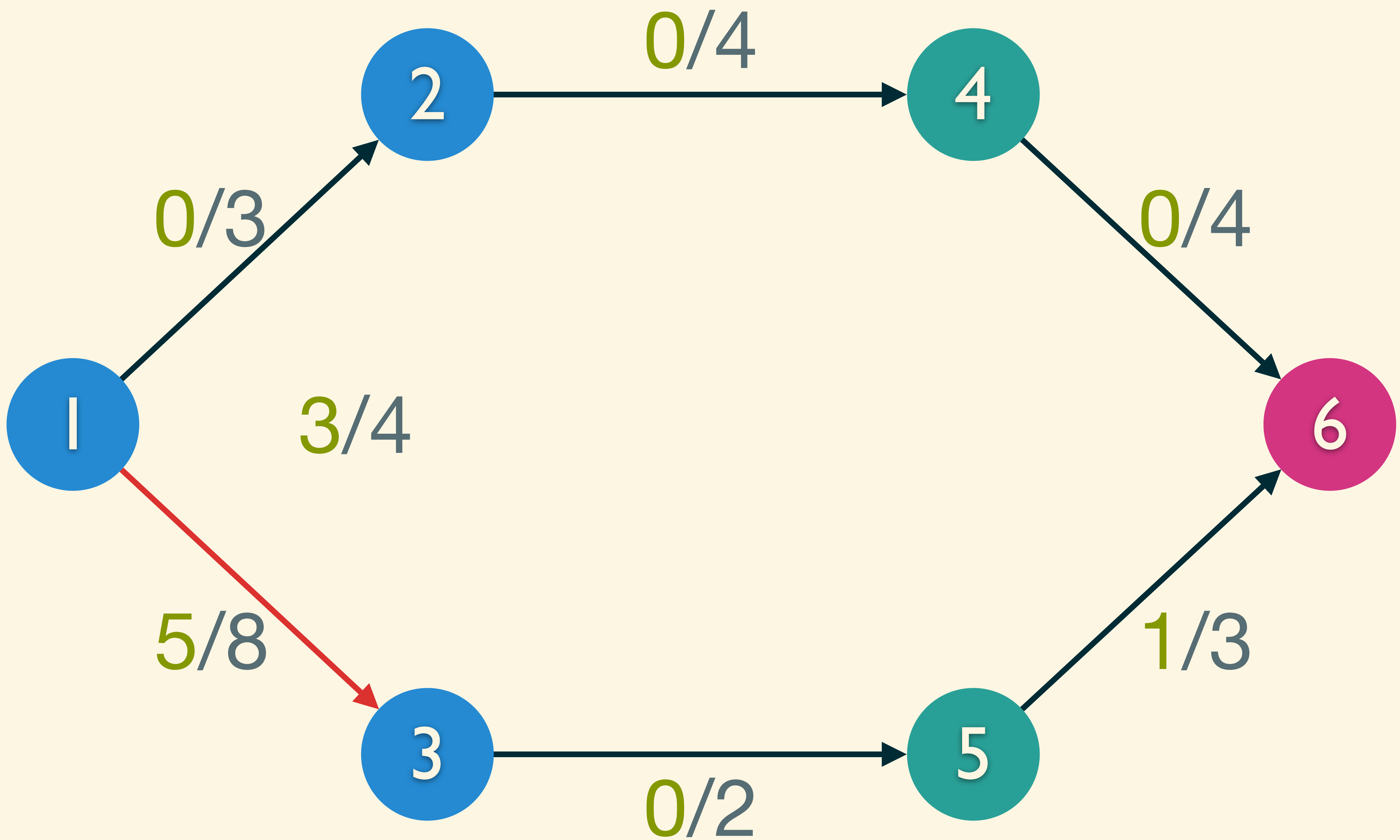
don’t travel through full edge



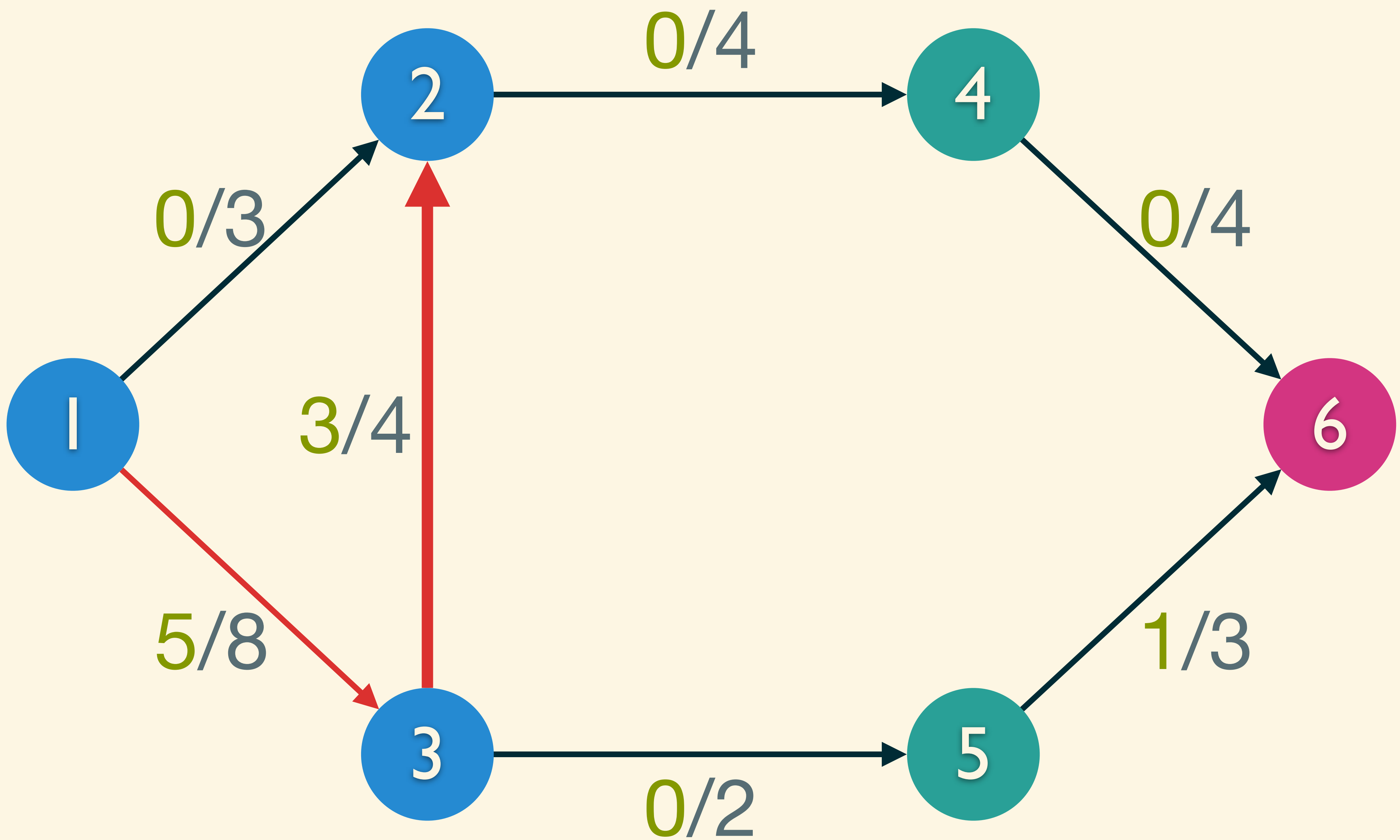
don’t travel through full edge



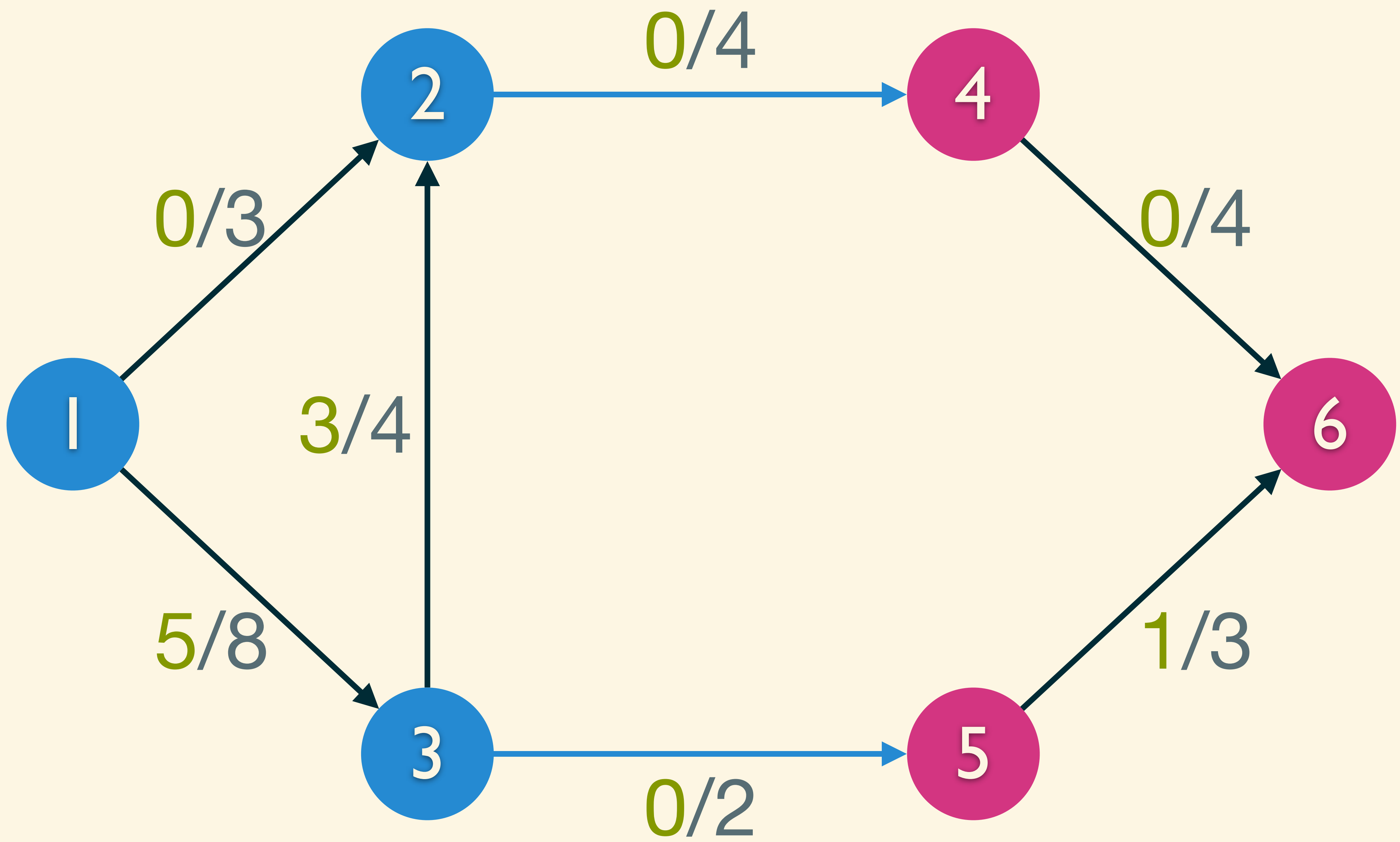
no residual edge



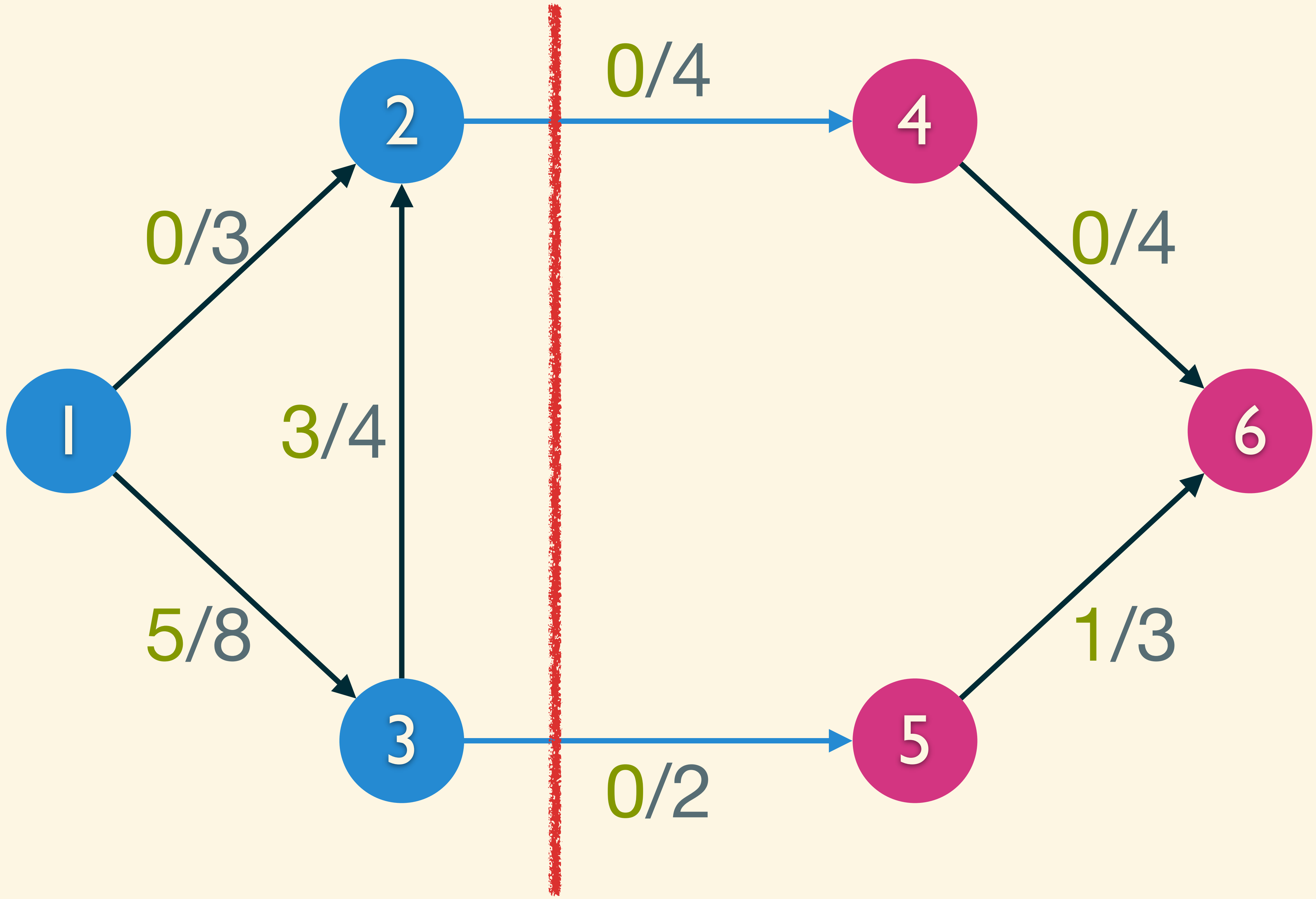
no residual edge



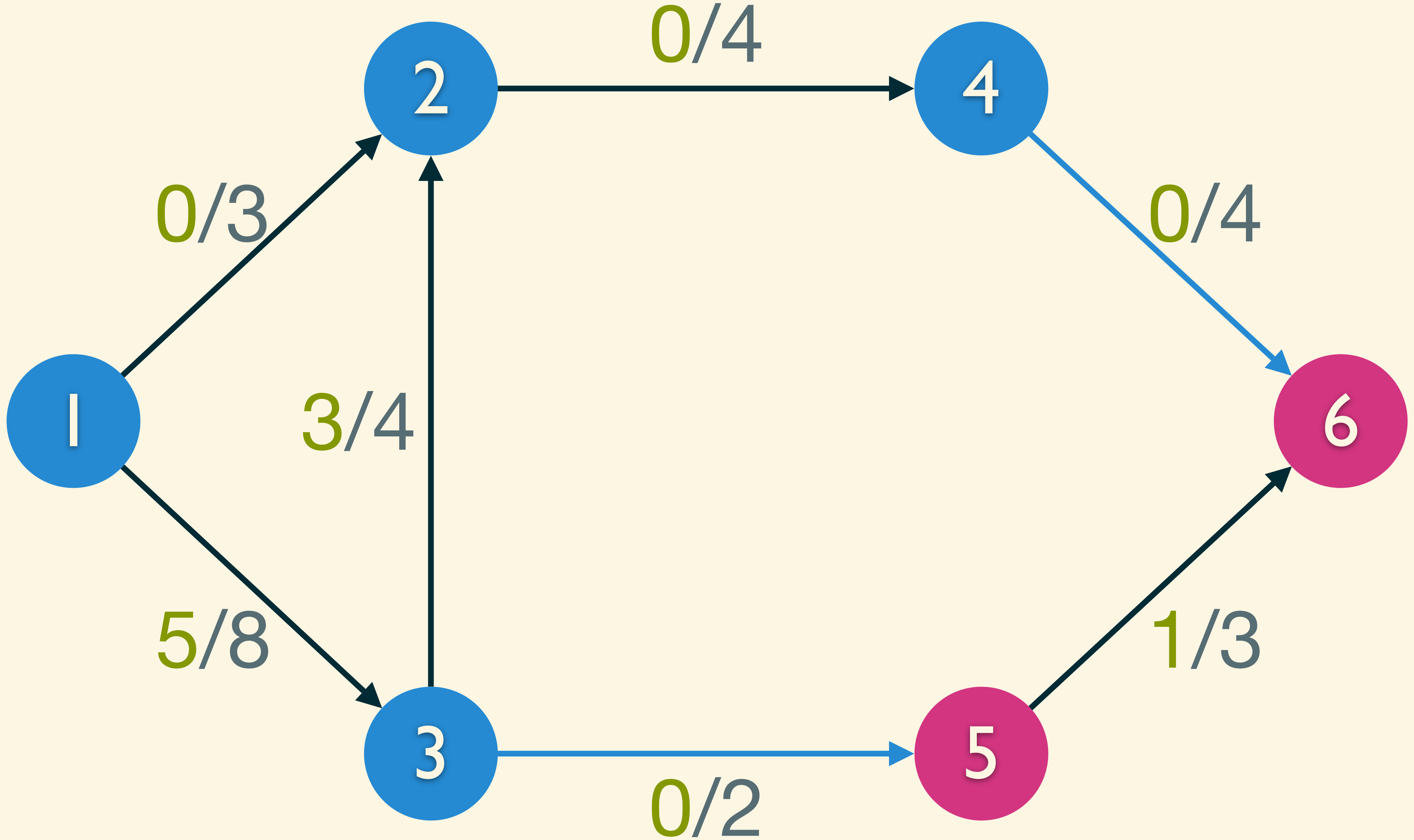
s-t cut



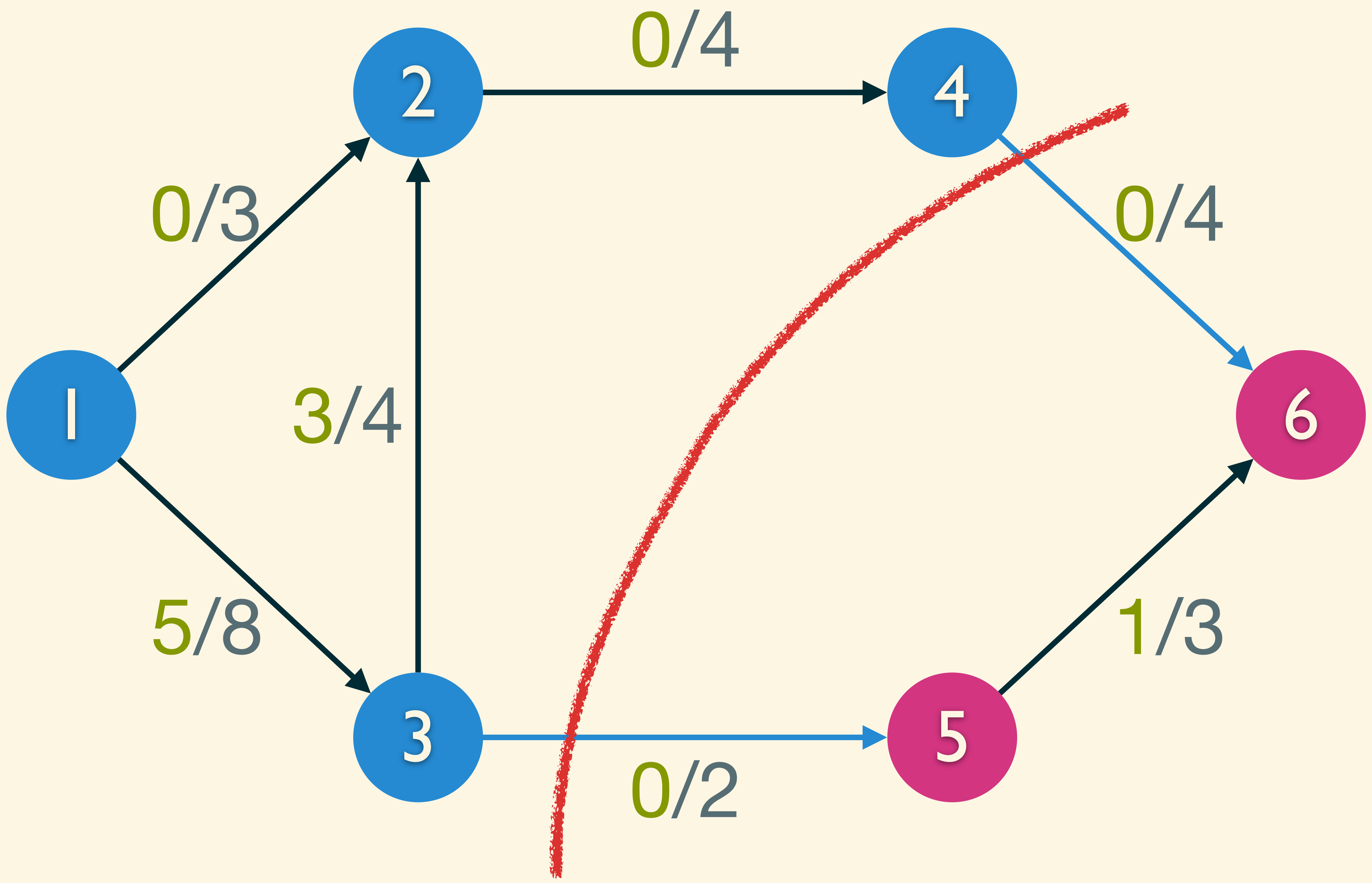
s-t cut



result of starting from **sink**



result of starting from **sink**



Minimum cut is non-unique!

time complexity:
based on max-flow algorithm

Ford-Fulkerson algorithm	$O(EF)$
Edmonds-Karp algorithm	$O(VE^2)$
Dinic algorithm	$O(V^2E)$

Stoer Wagner

only for **undirected graph**

time complexity: $O(\mathbf{N^3})$ or $O(\mathbf{N^2 \log_2 N})$

Practice Now

[UVa 10480 - Sabotage]

Problem List

UVa 10480
UVa 10989
POJ 1815
POJ 2914
POJ 3084
POJ 3308
POJ 3469

Reference

- http://www.flickr.com/photos/dgjones/335788038/
- http://www.flickr.com/photos/njsouthall/3181945005/
- http://www.csie.ntnu.edu.tw/~u91029/Cut.html
- http://en.wikipedia.org/wiki/Cut_(graph_theory)
- http://en.wikipedia.org/wiki/Max-flow_min-cut_theorem
- http://www.cs.princeton.edu/courses/archive/spr04/cos226/lectures/maxflow.4up.pdf
- http://www.cnblogs.com/scau20110726/archive/2012/11/27/2791523.html

Thank You for Your Listening.