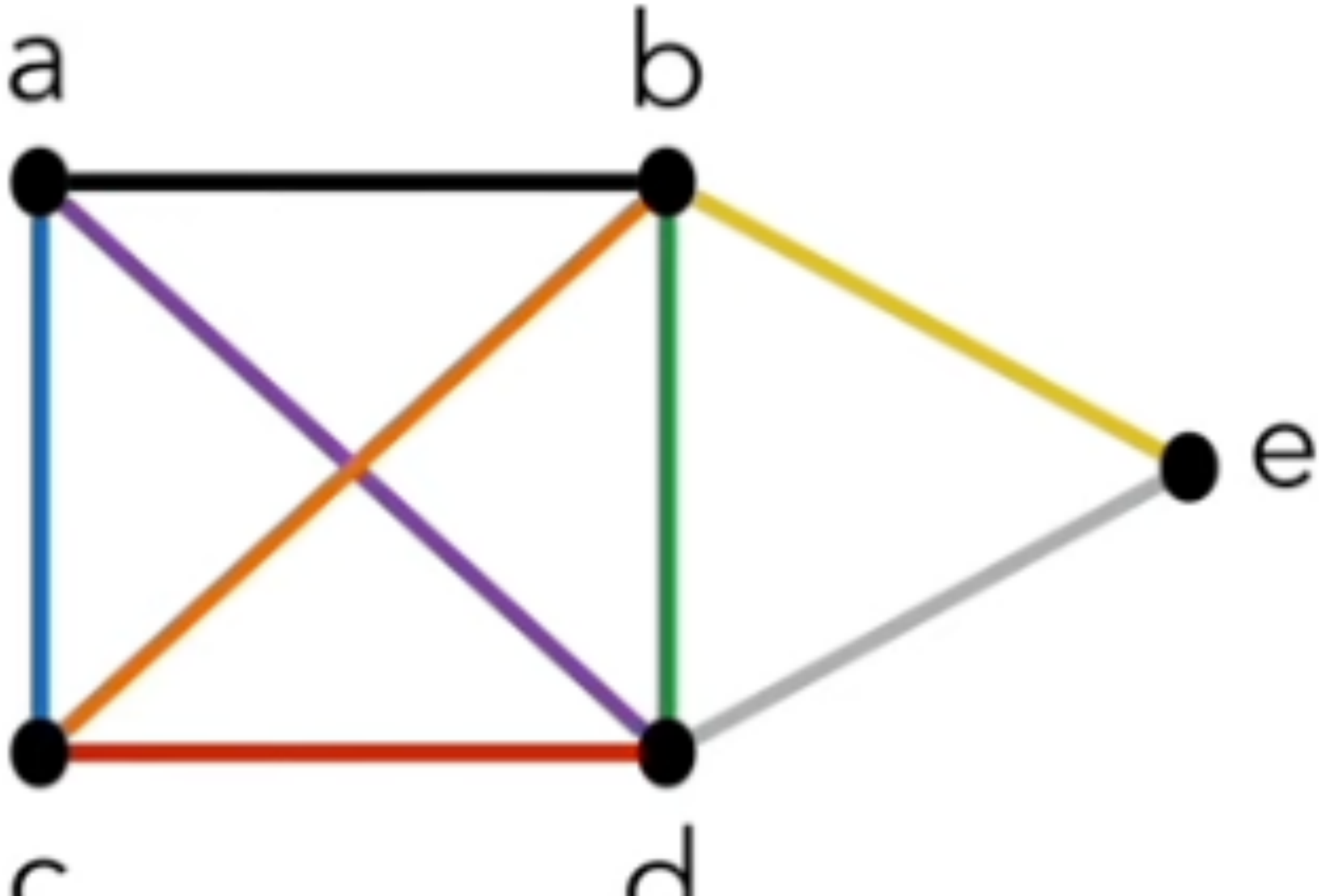
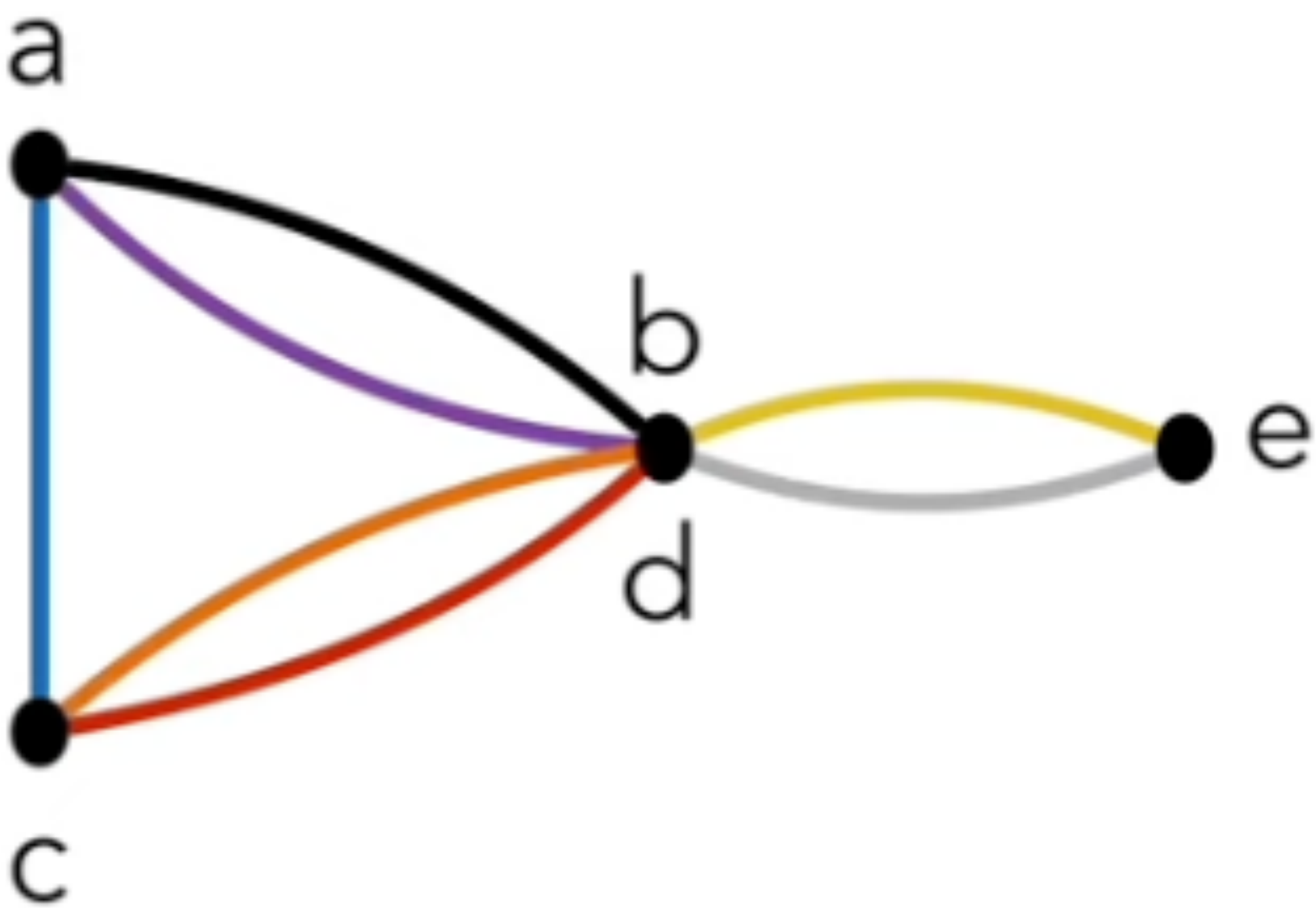
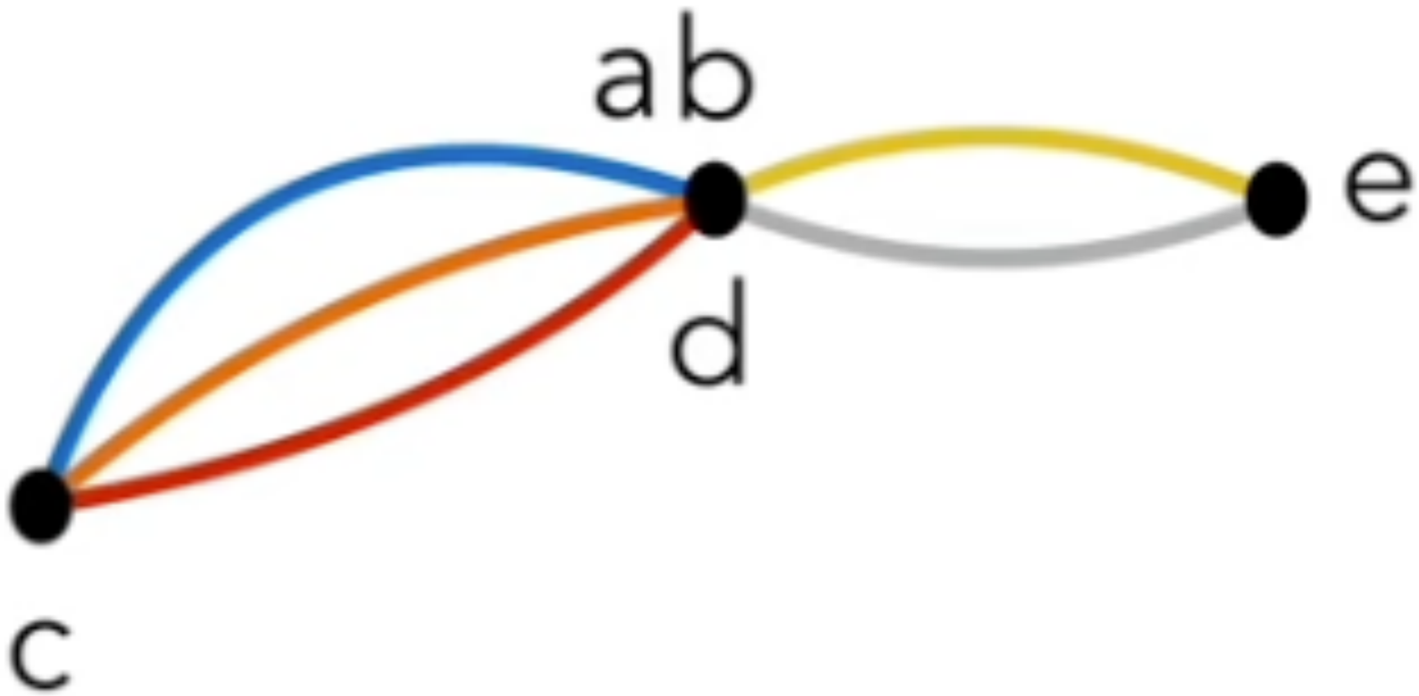
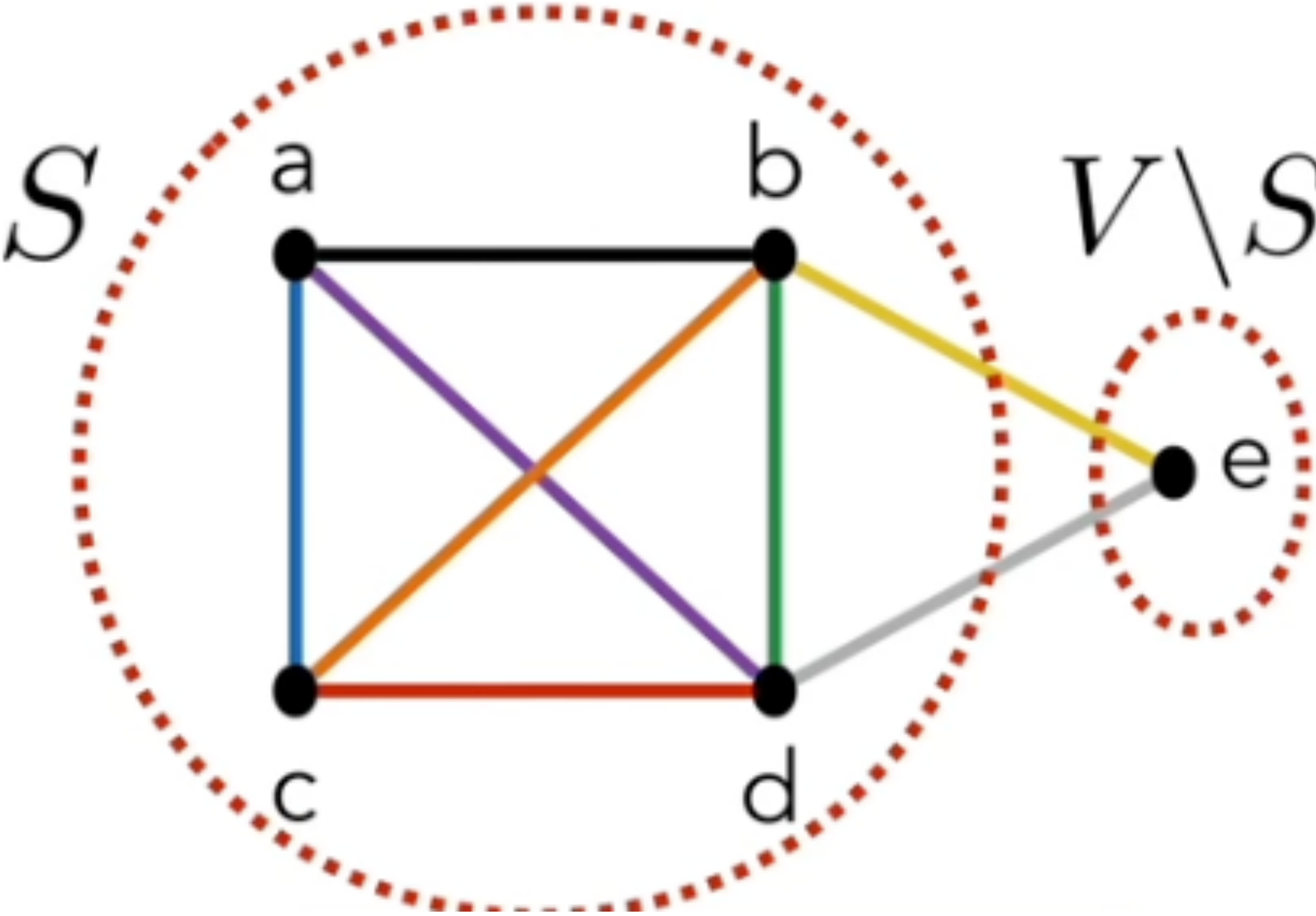
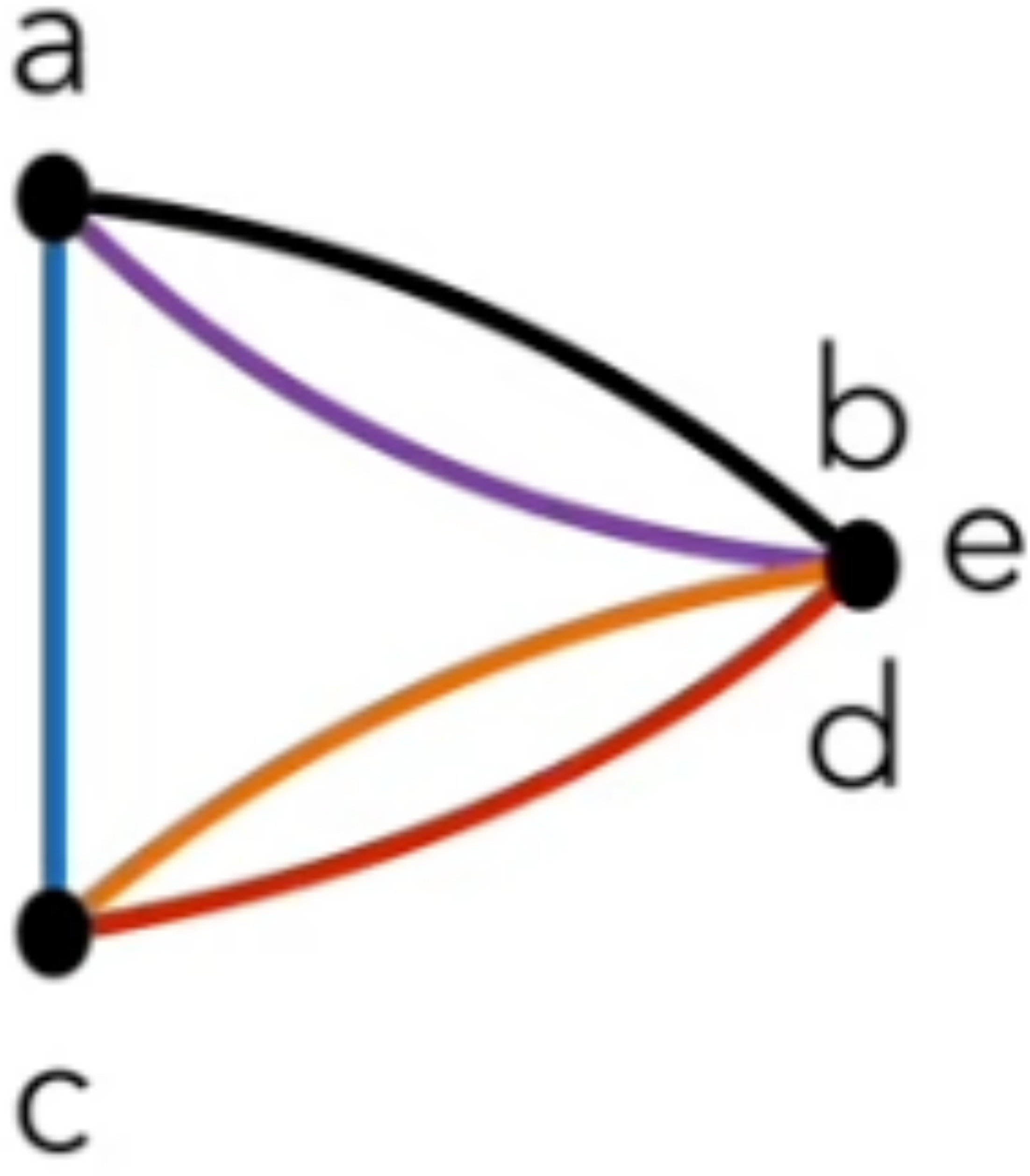

a
b
e
c
d

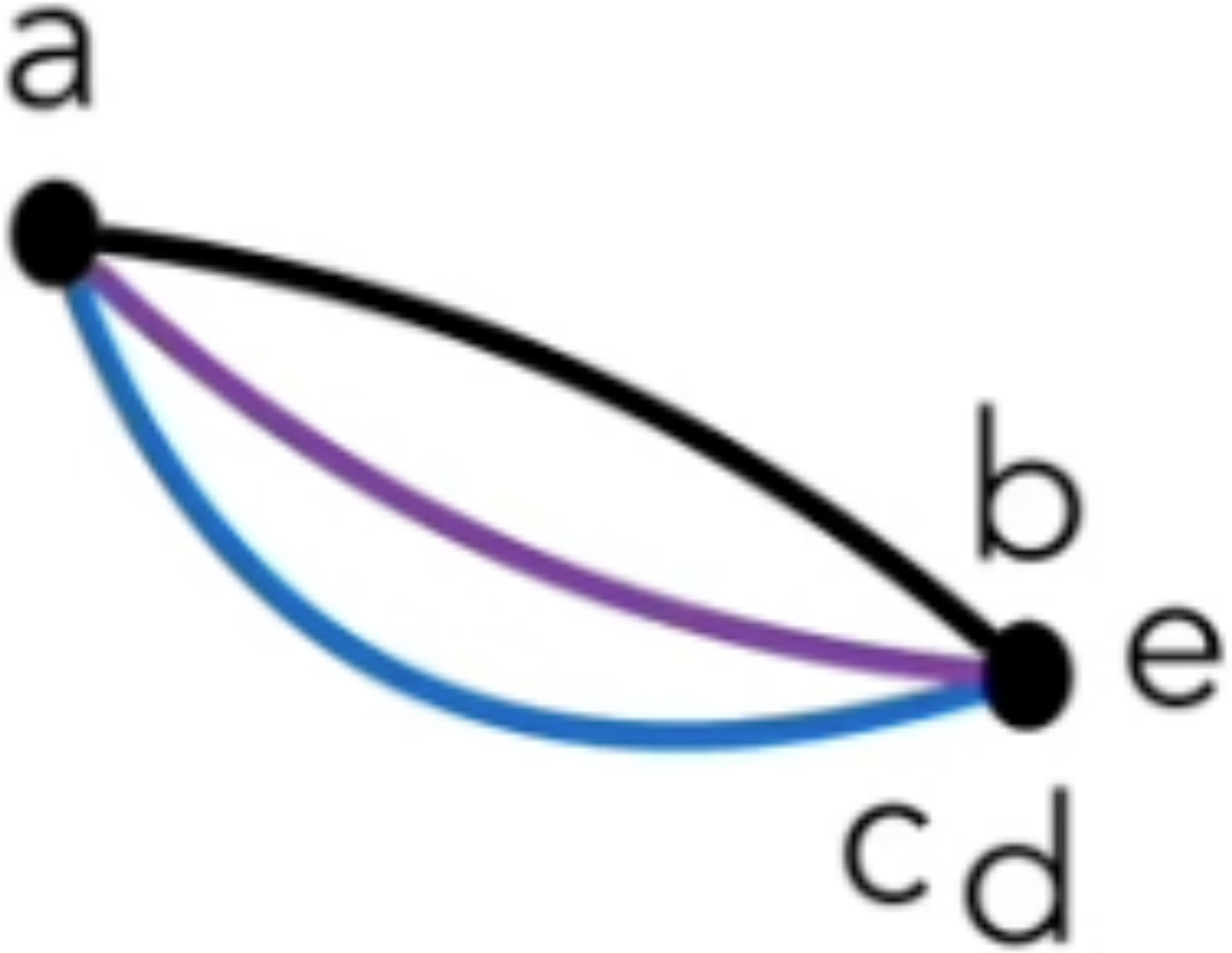

a
b
e
d
c

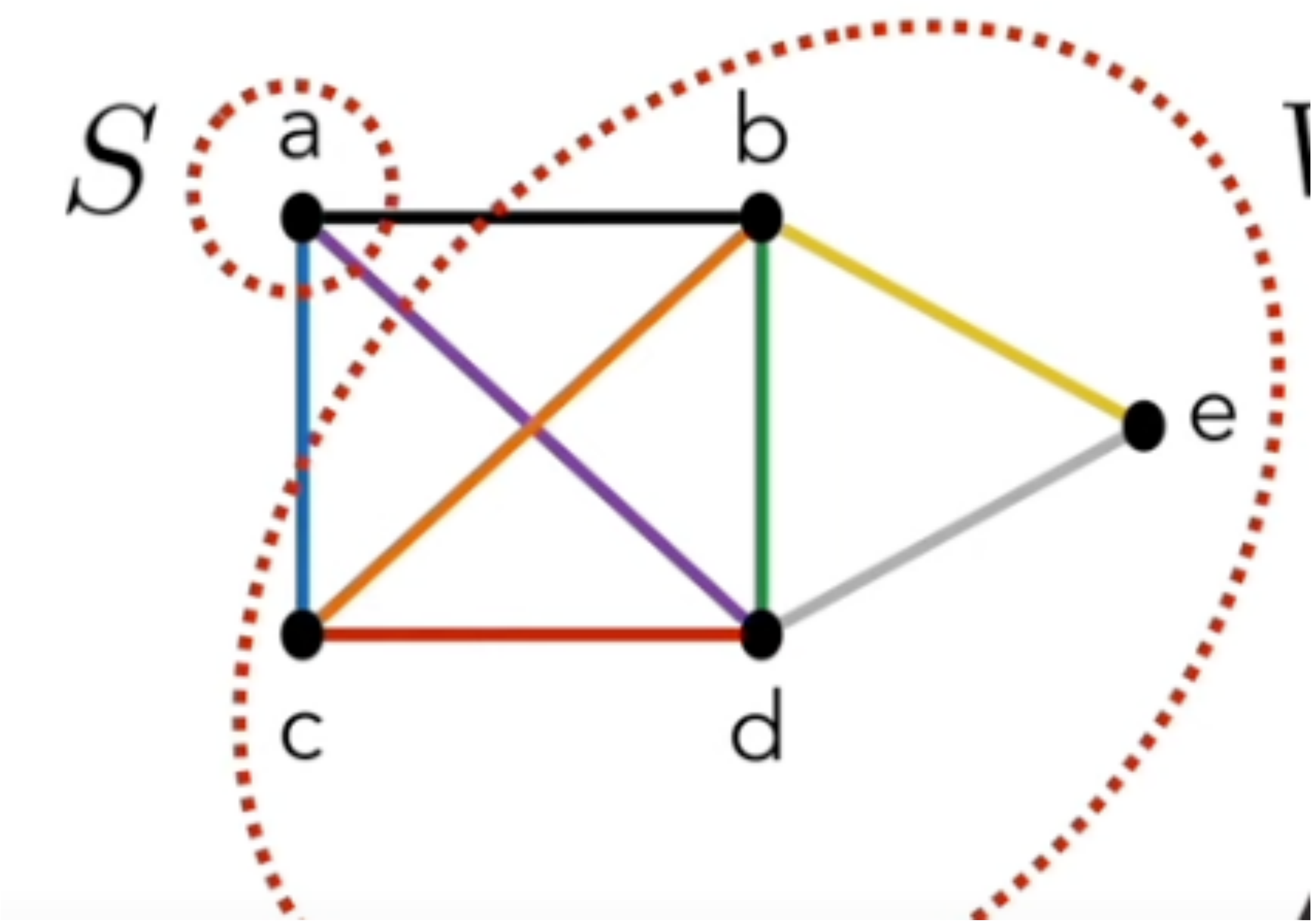

ab
d
e
c



S
a
b
$V \setminus S$
e
c
d


a
b
e
d
c


a
b
e
c
d


S
a
b
c
d
e

Contraction Algorithm: # Iterations

$$G = G_0 \xrightarrow{\text{contract}} G_1 \xrightarrow{\text{contract}} G_2 \xrightarrow{\text{contract}} \cdots \xrightarrow{\text{contract}} G_{n-2}$$

# vertices:	n	$n-1$	$n-2$	$\cdots$	2

$\boxed{n-2 \text{ iterations}}$

Contraction Algorithm: # Iterations

$$G = G_0 \xrightarrow[\text{iteration } 0]{\text{contract}} G_1 \xrightarrow[\text{iteration } 1]{\text{contract}} G_2 \xrightarrow[\text{iteration } 2]{\text{contract}} \cdots \xrightarrow[\text{iteration } n-3]{\text{contract}} G_{n-2}$$

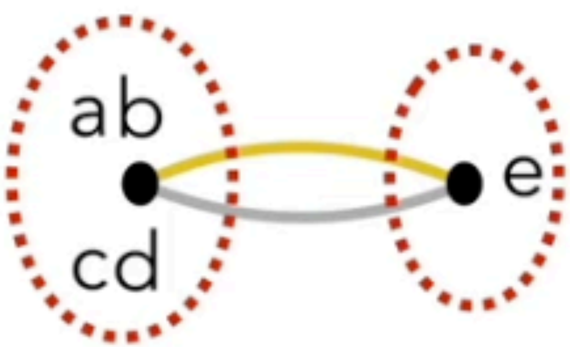
 # vertices: n $\quad n-1 \quad n-2 \quad \cdots \quad 2$

$$\boxed{n-2 \text{ iterations}}$$

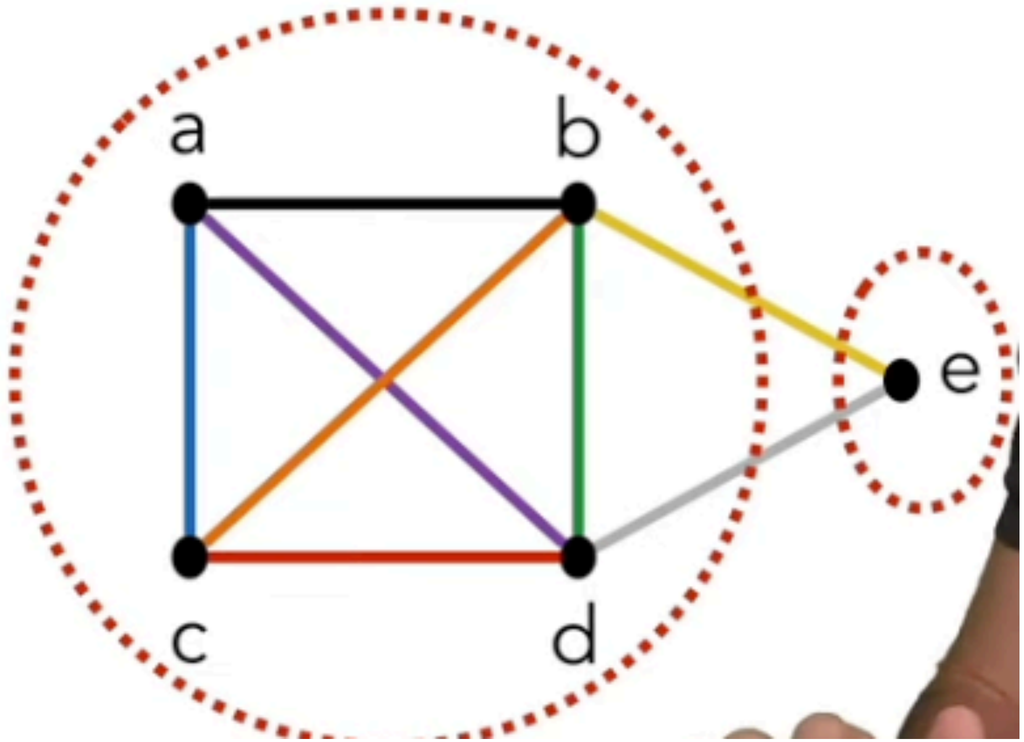
Observation:

For any i: A cut in G_i of size k corresponds exactly to a cut in G of size k.

G_{n-2}



G



Contraction Algorithm: Success probability

Theorem: Let $G = (V, E)$ be a graph with n vertices.
Then $\Pr[\text{contraction alg. outputs a min cut}] \geq 1/n^2$.

Should we be impressed?

- The algorithm runs in polynomial time.

- There are exponentially many cuts. ($\sim 2^n$)