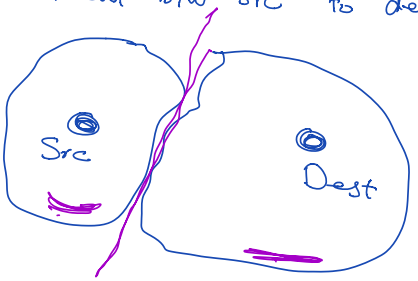
* We measure the quality of a monte-Carlo Alg. with Prob. of generating the Correct output

```
while check
{  output = MCAlg(...)
   check = verify(output)
}
```

---

A Monte Carlo Randomized Alg. for Min-Cut Problem

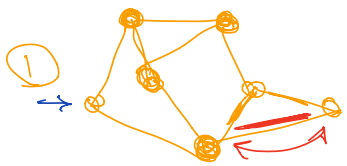
Min-Cut b/w Src to Dest

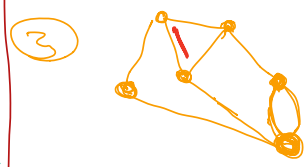


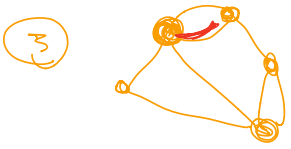

Min-Cut: we want to find the min. no. of edges that their removal disConnects the graph

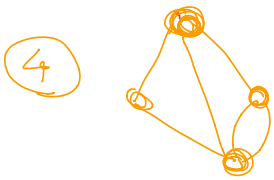
Deterministic Alg.

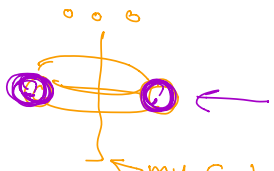
```
minCut=∞
for i = 1 to n-1
    for j=i+1 to n
        Cut = minCut(i,j)
        if (Cut < minCut)
        {   minCut = Cut
```




Rand. Alg.

```
while |v| > 2
    (u,v) = Select an edge uniformly
            at random
    update the graph by replacing
    (u,v) w/t a Supernode
```

the edges b/w the Survived pair of nodes is the Cut Set

* what is the Success prob.
what is the Prob. that this Alg. finds a Cut of Size at most $k$

* we assume min cut Size $= k$

Consider an optimal Cut

* The min # edges

$nk/2$ : The degree of each node is $k$

what is the Prob. of selecting one edge from the Cut Set

$P(\mathcal{E}_i)$ : the prob. that at iteration $i$, none of the min-cut edges is selected

$$P(\mathcal{E}_1) = 1 - \frac{k}{nk/2} = 1 - 2/n$$

* after every iteration the # nodes gets reduced by 1

| # nodes | iteration |
|---|---|
| $n$ | 1 |
| $n-1$ | 2 |
| ... | ... |
| $n-i+1$ | $i$ |

* the Size of opt. Cut does not reduce during the iterations

what is the min # edges at Step $i$

$$(n-i+1)\,k/2$$

$$P(\mathcal{E}_i \mid \mathcal{E}_1 \cdots \mathcal{E}_{i-1}) = 1 - \frac{k}{(n-i+1)k/2}$$

$$= 1 - \frac{2}{n-i+1}$$

$$P(\text{Success}) = P(\mathcal{E}_1)P(\mathcal{E}_2|\mathcal{E}_1)\cdots P(\mathcal{E}_{n-2} \mid \mathcal{E}_1 \cdots \mathcal{E}_{n-3})$$

$$= \prod_{i=1}^{n-2} P(\mathcal{E}_i \mid \bigwedge_{j=1}^{i-1} \mathcal{E}_j)$$

$$= \prod_{i=1}^{n-2} \left(1 - \frac{2}{n-i+1}\right)$$

$$= \frac{2}{n(n-1)} > \boxed{\frac{2}{n^2}}$$

---

Run the algorithm $\frac{n^2}{2}$

$$P(\text{failure}) = \left(1 - 2/n^2\right)^{n^2/2}$$

$\rightarrow \boxed{< 1/e}$ ✓

3 SAT

- $v_1, v_2, \ldots, v_n$
- $C_1, C_2, \ldots, C_m$

$\stackrel{?}{\exists}$ assignment to variables such that all clauses are Satisfied?

MaxSAT:

Maximize the # of clauses that get Satisfied

Alg. Randomly assign {True, false} w/t Prob 0.5 to Variables

$P(C_i = \text{True}) = 7/8$

$\hookrightarrow v_{i_1} \vee v_{i_2} \vee v_{i_3}$

0 0 0 } $2^3 = 8$

1 1 1 }

$v_1 \vee \bar{v}_2 \vee v_3$

$0 \vee 1 \vee 0 = \text{false}$

$E[\text{Satisfied clauses}] = \underbrace{7/8}_{T}\, m$ ✓

Assume I have 7 clauses (i.e., $m=7$)

There always Exists an assignment that Satisfies all clauses?

$E[T] = 7/8 \times 7 = 6\ldots$