Optimal Determination of Source-Destination Connectivity in Random Networks

Sireesha Madabhushi

Signal Processing for Communications (SPC) Lab
Indian Institute of Science (IISc)
Bangalore, India

December 17, 2016

Introduction

Why do we want to know connection/disconnection?

Several applications - priced information query, communication networks, nearest neighbours

Erdos-Renyi random graph, G(n,p)

- All edges between n nodes enabled with an equal probability, p
- Literature has explored asymptotic connectivity
- Optimal policy to find connection/disconnection between source-destination in minimum number of steps
- Sequential testing strategy for a specific realization of the random graph
- Optimal policy does not depend on n or p
- Problem different from that of finding the shortest path

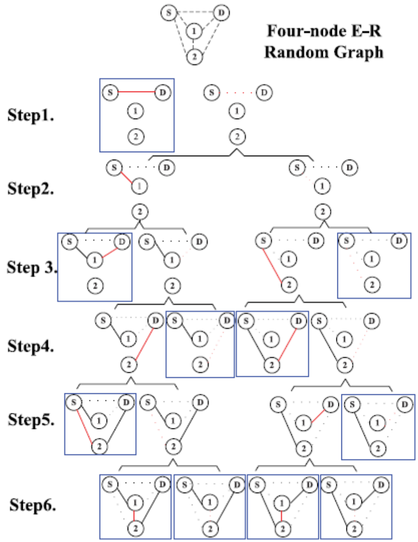

Figure: Policy applied on 4-node ER graph

Notation

- e_t - Edge tested at step t
- G_t - Graph state as known at step t
- $C_{S,t}, C_{D,t}$ and $C_{i,t}$ - Connected component containing the source, destination and the i^{th} component not containing the source and destination respectively
- M_G - Minimum cut for a graph state G with minimum number of potential edges at time t
- Three kinds of edges for this presentation - known edge, known non-edge, potential edge

Alternating policy

Policy $\pi^* = \{\pi_0^*, \pi_1^*, \ldots, \pi_N^*\}$ to check for $S - D$ connectivity in ER graph. Each π_t^* maps G_t to one of the remaining potential edges to be tested at time t.

- *Rule 1*: Test for direct one-hop potential edge between C_S and C_D.
- *Rule 2*: If no edges found from *Rule 1*, get a path list L, with minimum number of potential edges and selected M_G.
  M_G divides the graph into $C \cup C_S$ and C_D or C_S and $C \cup C_D$, where $C = C_1 \cup C_2 \cup \ldots C_r$
- *Rule 3*: Sort $C_1, C_2, \ldots C_r$ and test the edge in L that connects C_S or C_D to component, C_i with largest no.of edges

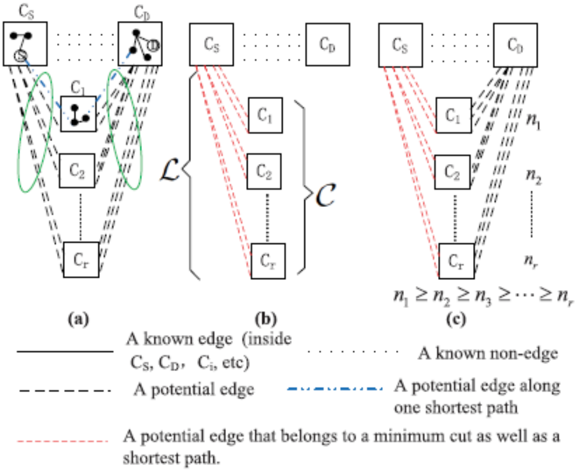

Figure: Illustration of Rules 2 and 3

Claim: The alternating policy π^* is optimal.
Prove that each of the three rules is optimal.

Optimality of Rule 1

Lemma (1)

For any policy $\mathcal{A}$ that tests an edge other than direct potential edge between C_S and C_D, there is a policy $\tilde{\mathcal{A}}$ that tests direct potential edge and incurs a lower cost.

Idea: Stochastic coupling
$U_i(\omega)$: choice made by $\mathcal{A}$ at a step i
$T(\omega)$: termination time of $\mathcal{A}$
$n_d(\omega)$: first time that $\mathcal{A}$ tests d

$$n_d(\omega) = inf\{t \geq 1 : U_t(\omega) = d\}$$

Sequence of edges tested under $\mathcal{A}$ till it terminates are divided into 3 cases.

1. $\{U_1(\omega), U_2(\omega), \ldots, U_T(\omega)\}$ where $U_i(\omega) \neq d, i = 1, \ldots, T(\omega)$. $T(\omega) \leq n_d(\omega) = +\infty$
2. $\{U_1(\omega), U_2(\omega), \ldots, U_{n_d(\omega)-1}(\omega), U_{n_d(\omega)}(\omega) = d\}$. $T(\omega) = n_d(\omega)$
3. $\{U_1(\omega), U_2(\omega), \ldots, U_{n_d(\omega)-1}(\omega), U_{n_d(\omega)}(\omega) = d, U_{n_d(\omega)+1}(\omega), \ldots, U_{T(\omega)}(\omega)\}$

Construct family of policies $\{\mathcal{A}^i\}$ such that

- $\{\mathcal{A}^i\}$ follows $\mathcal{A}$
- If $\mathcal{A}$ has not tested d by step i, then $\{\mathcal{A}^i\}$ does test d by step i
- $\{\mathcal{A}^i\}$ catches up with $\mathcal{A}$ when $\mathcal{A}$ tests d

Sequence of resulting edges tested by $\mathcal{A}^1$ till it terminates has 3 cases

1. If $T^1 = n_d(\omega)$ or $\{T^1 < n_d(\omega), e_d = 1\}$, $\{d\}$
2. If $T^1 < n_d(\omega)$ and $e_d = 0$, $\{d, U_1(\omega), U_2(\omega), \ldots, U_{T^1}(\omega)\}$
3. If $T^1 > n_d(\omega)$ and $e_d = 0$,
   $\{d, U_1(\omega), U_2(\omega), \ldots, U_{n_d(\omega)-1}(\omega), U_{n_d(\omega)}(\omega), U_{n_d(\omega)+1}(\omega), \ldots, U_{T^1}(\omega)\}$

Therefore,

$$\begin{aligned}|T^1| = \mathbb{1}(T^1 = n_d(\omega)) + \mathbb{1}(T^1 < n_d(\omega), e_d = 1) \\ + \mathbb{1}(T^1 < n_d(\omega), e_d = 0)[|T^1|+1] \\ + \mathbb{1}(T^1 > n_d(\omega), e_d = 0)|T^1|\end{aligned}$$

$$
\begin{aligned}
T^1 - T^i = \mathbb{1}(T^1 = n_d(\omega))[1 - T^i] \\
+ \mathbb{1}(T^i < n_d(\omega), e_d = 1)[1 - T^i] \\
+ \mathbb{1}(T^i < n_d(\omega), e_d = 0)1
\end{aligned}
$$

If we have policy $\mathcal{A}^2$, T^2 is defined similarly and,

$$
\begin{aligned}
\mathbb{E}[T^1 - T^2] &= \mathbb{E}[\mathbb{1}(T^2 = 2)(1-2) + \mathbb{1}(T^2 < 2, e_d = 1)x(1-1) \\
&\quad + \mathbb{1}(T^2 < 2, e_d = 0)|T|] \\
&= \mathbb{E}[\mathbb{1}(T^2 = 2)(-1) + \mathbb{1}(T^2 = 1, e_d = 0)] = -p + 0 = -p
\end{aligned}
$$

This implies that $\mathbb{A}^1$ has lower cost than $\mathbb{A}^2$.
Similarly each $\mathbb{A}^i$ has lower cost than $\mathbb{A}^{i+1}$.

Optimality of Rule 2

Lemma (2)

When a policy follows both Rules 1 and 2, all the edges in the minimum cut at any step will be between C_S *and* $\cup_{i=1}^{r} C_i$*, or they will all be between* $\cup_{i=1}^{r} C_i$ *and* C_D*.*

- Suppose that at some step, not all the components other than C_S and C_D lie in the same class
- Number of potential edges will be 3 (not 2), which violates *Rule 2*

Lemma (3)

When there are no direct edges between C_S *and* C_D*, listing all the potential shortest paths and sampling the edges in the minimum set on them will lead to smaller expected cost than sampling any other edge first.*

- Induction on num of steps
- Edges in the minimum cut are all between C_S and $\cup_{i=1}^{r} C_i$
- For step $t+1$, consider policy $\mathcal{A}$ which violates *Rule 2*, tests edge between C_1 and C_D, and then follows $\tilde{\mathcal{A}}$.

Optimality of Rule 3

Lemma (4)

Among all the potential edges that C_1 *and* C_2 *connect to in* M_{G_t} *and* $k_{12} \geq k_{22}$*, it incurs smaller expected cost to first test the ones that* C_1 *is connected to, where* k_{12} *is num of edges from* C_1 *to* C_D *and* k_{22} *is num of edges from* C_2 *to* C_D*.*

- Induction on the num of potential edges
- Rules 1,2 and 3 hold good for 2 potential edges
- Consider policy $\tilde{\mathcal{A}}$ which follows the Rules and $\mathcal{A}$ which violates *Rule 3* by testing an edge connected to C_2, and then follows the optimal policy for graph with k edges

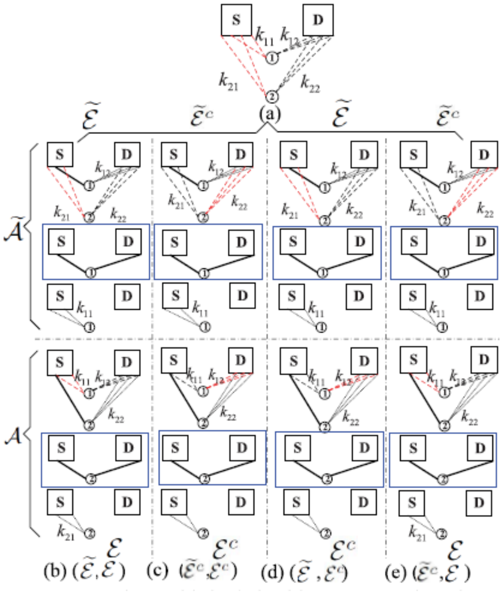

Figure: Illustration of the cases

Computational complexity

Lemma (5)

The optimal policy is implementable with a computational complexity of no more than $30 \log_2 n$ *operations at each step*

Extension to general graphs

- $(1, p)$ random graphs
- $(1, 0, p)$ random graphs
- Series of parallel graphs
- Parallel of series graphs
- Series of parallel of series graphs
- Parallel of series of parallel graphs

References

- Luoyi Fu, Xinbing Wang and P.R.Kumar, "Are We Connected? Optimal Determination of Source-Destination Connectivity in Random Networks", in IEEE/ACM Transactions on Networking , vol.PP, no.99, pp.1-14