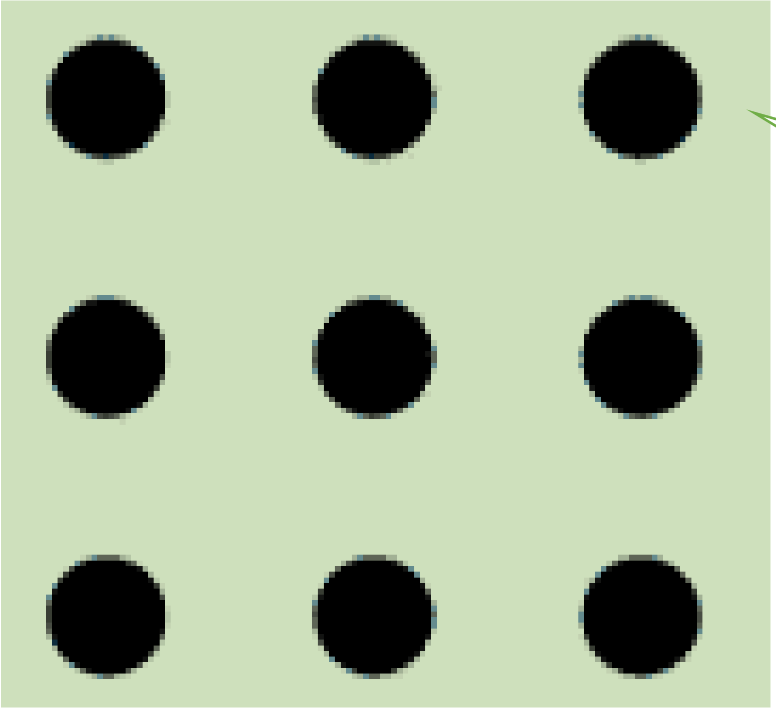
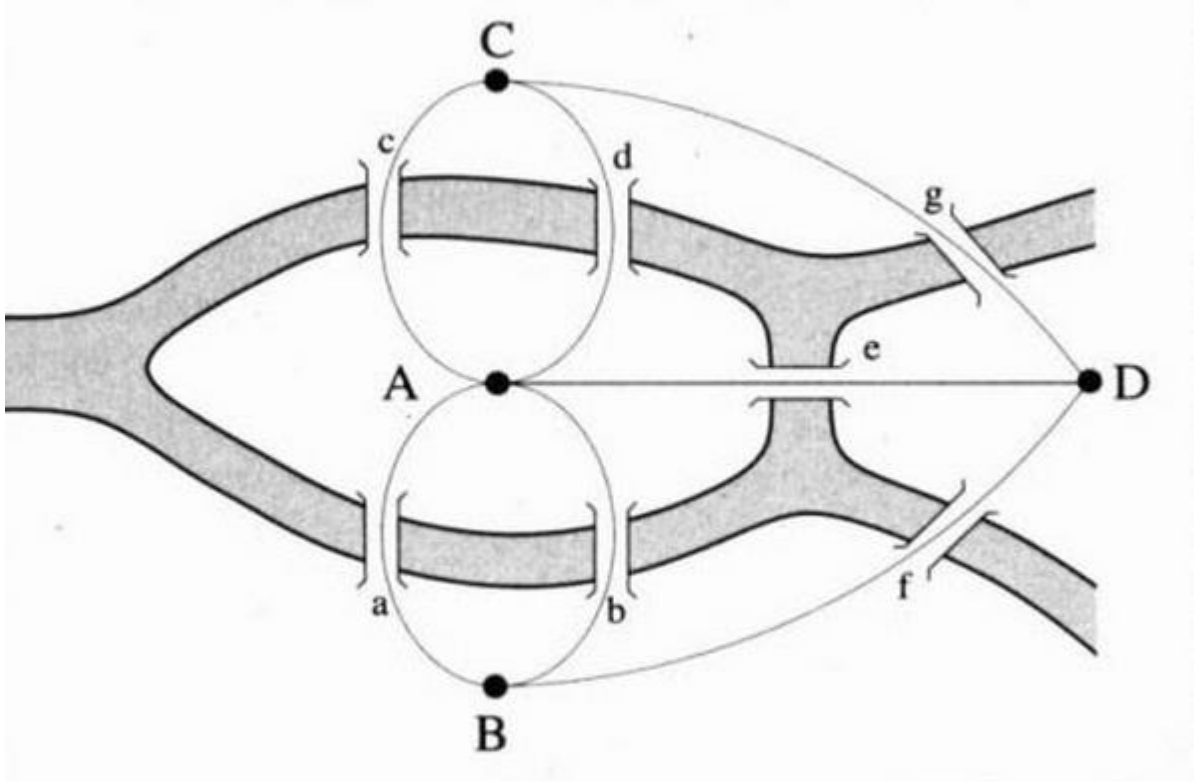
Network Analysis

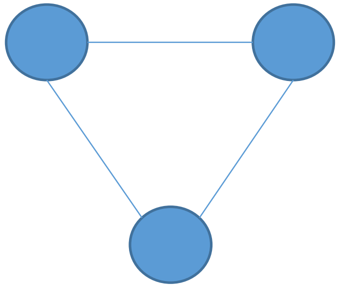


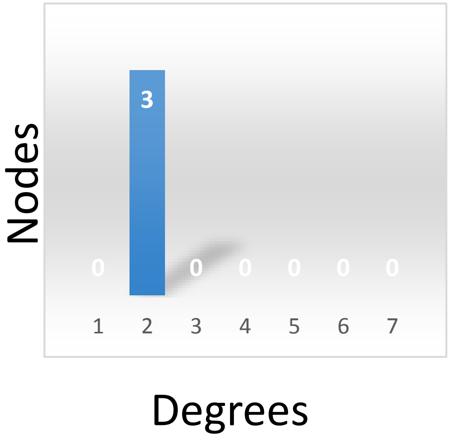
Connect all 9 points with 4 interconnected lines. Each point must be touched once. Last point must be touched twice. Lines are allowed to intersect. Keep the pencil point always on the paper.

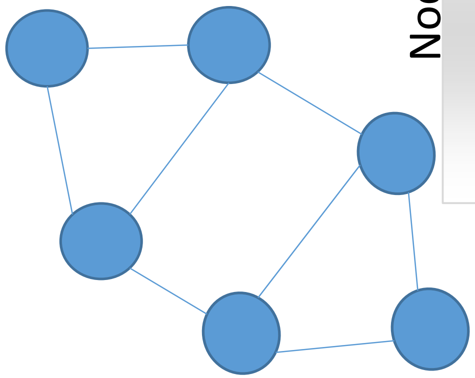
The Bridges of Konigsberg (Euler: Graph Theory)

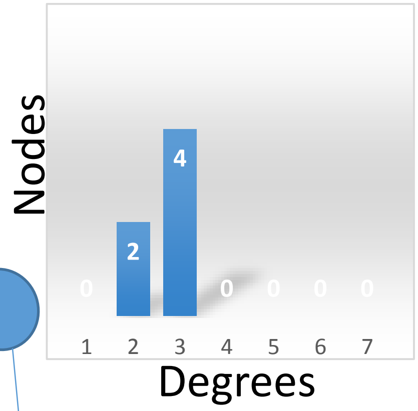


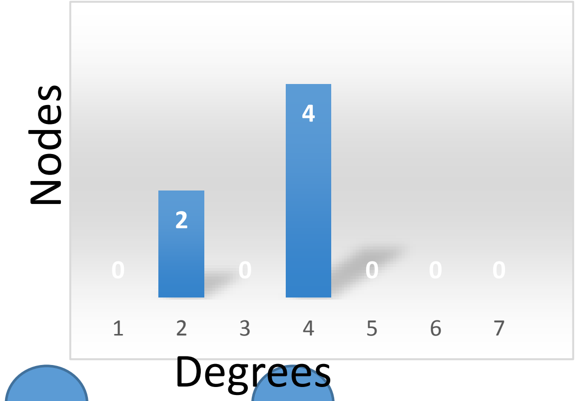
Degree Distribution

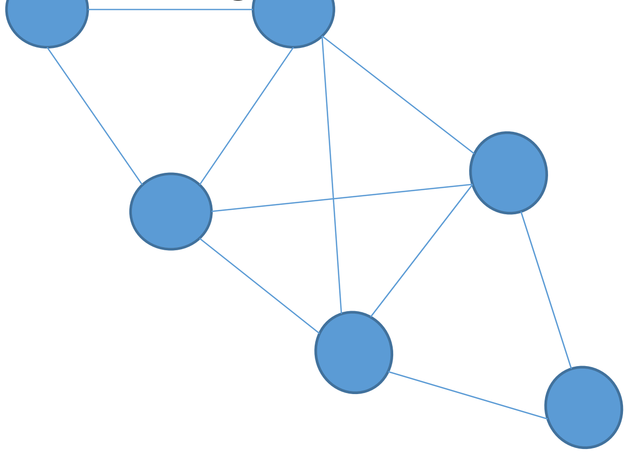


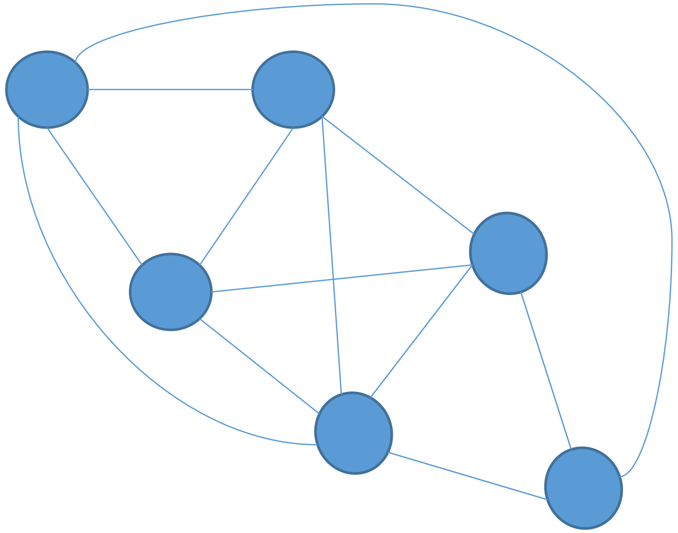


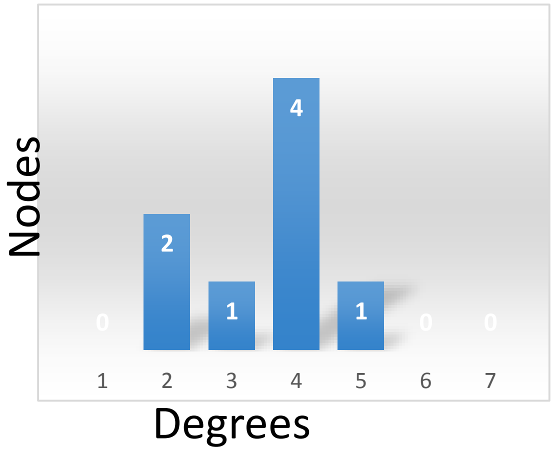




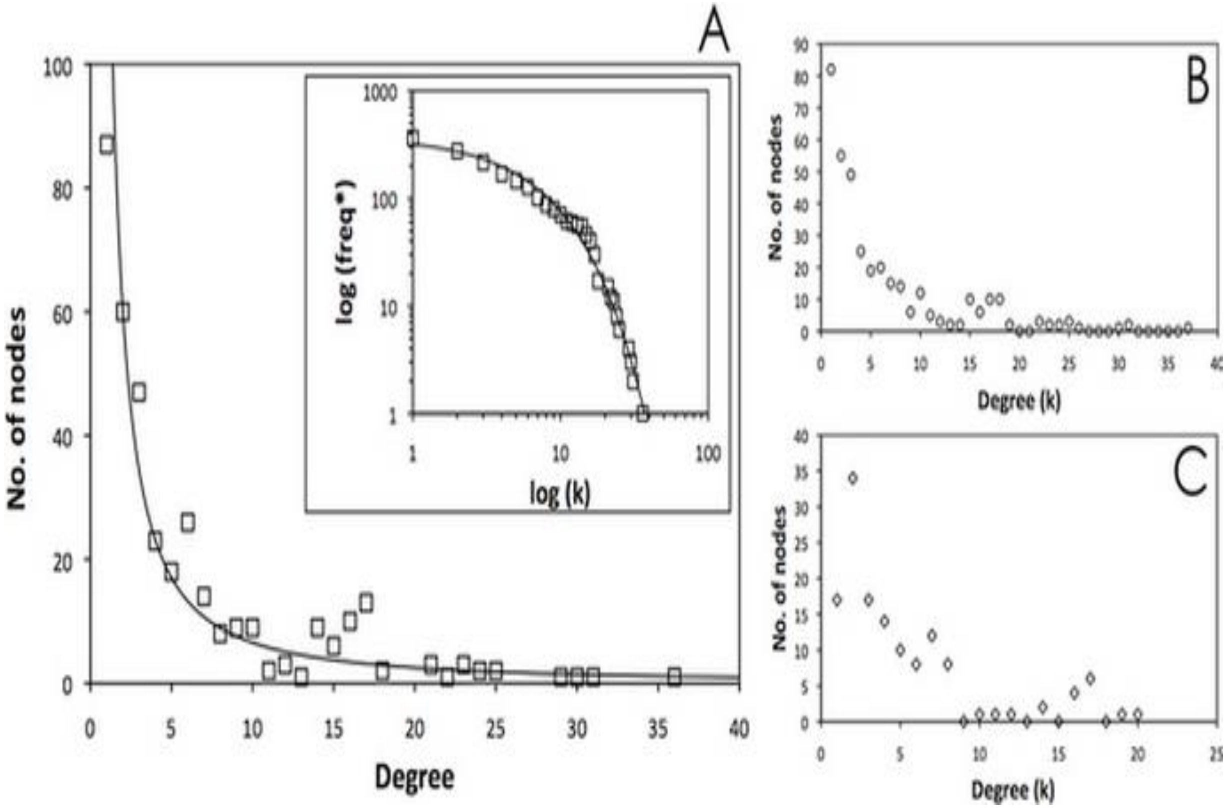


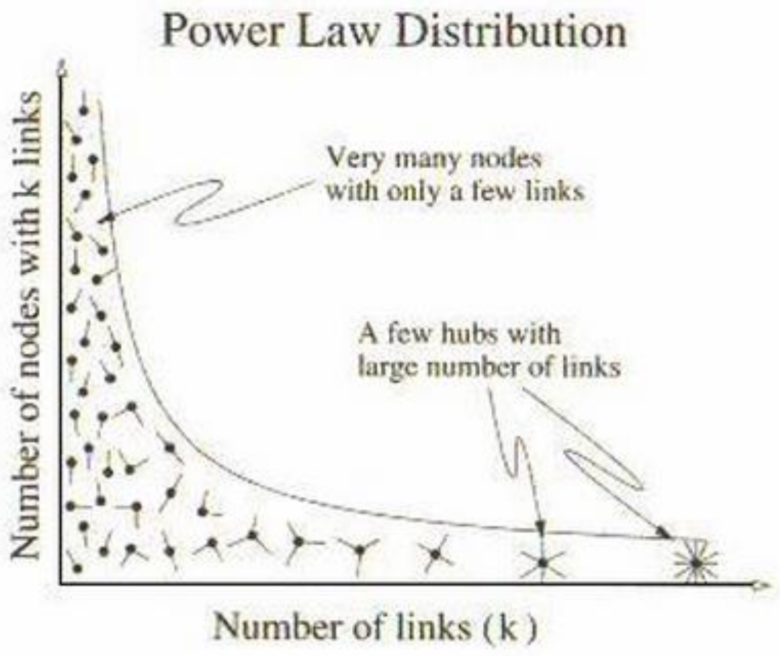






Degree Distribution: **Tens of degrees**

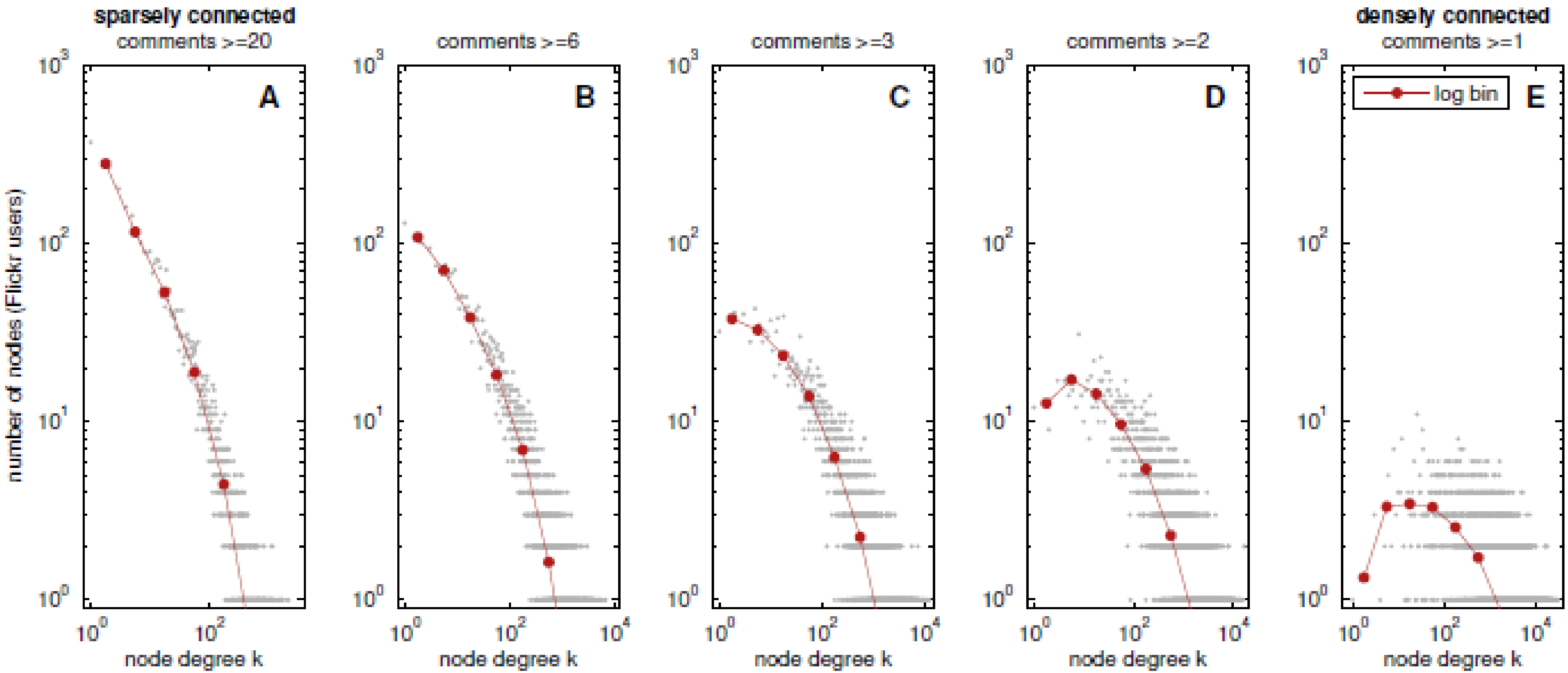




~~T-test~~
~~ANOVA~~
~~Least-Squares Regression~~
~~K-Means~~
~~GMM~~
~~BernoulliNB~~
~~PCA / CCA / SEM~~

(Partially reproduced from here: http://www.nature.com/srep/2011/110811/srep00061/full/srep00061.html#close)

Degree Distribution: **Higher Connectivity**

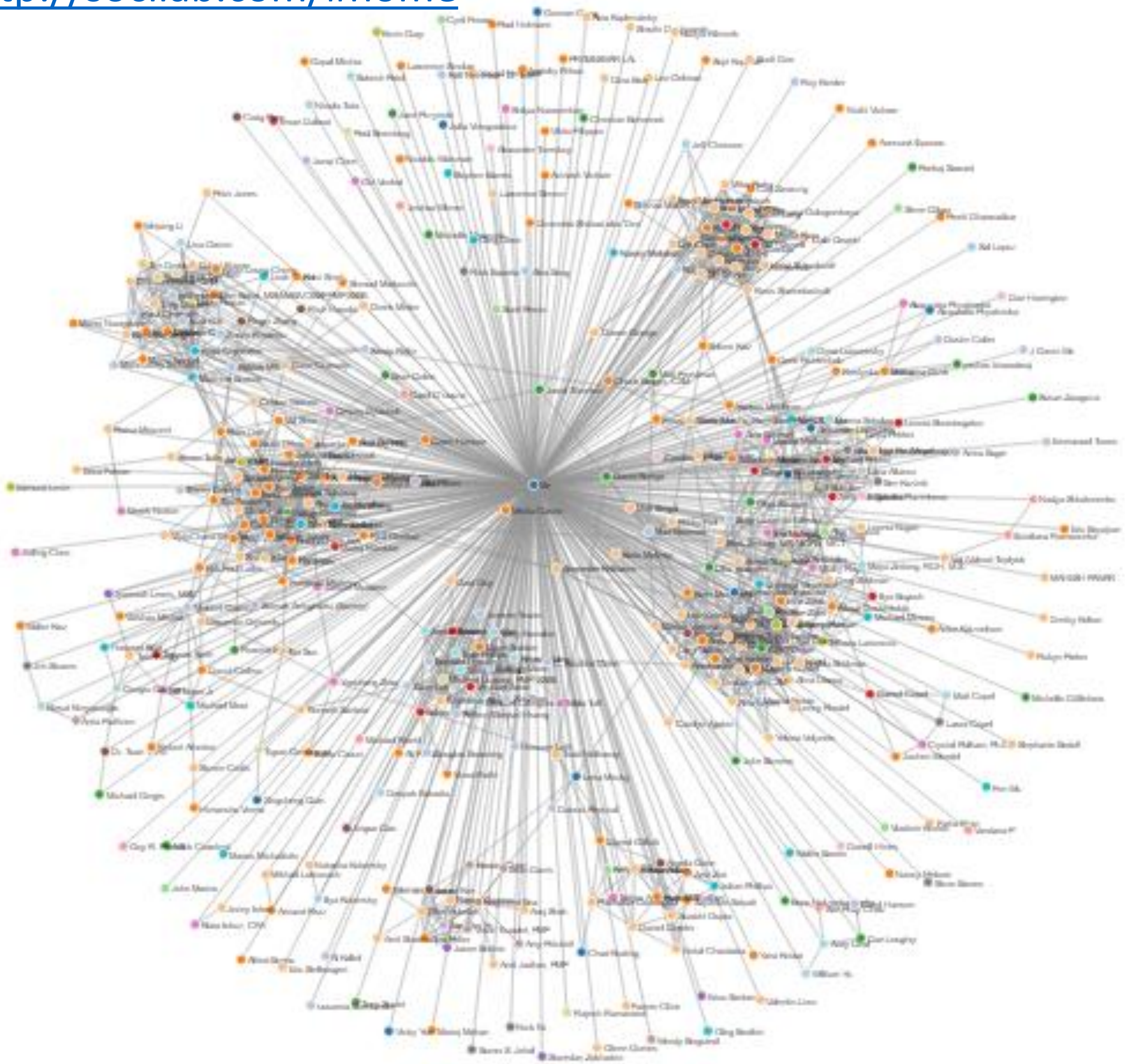


Flickr comment counts: the denser the network, the fewer comments per edge

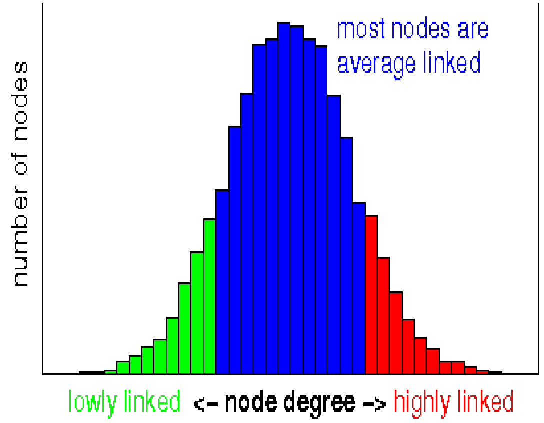
(Study conducted by Matthias Scholz: http://jdmdh.episciences.org/77)

Degree Distribution: **Higher Connectivity: Less Order**

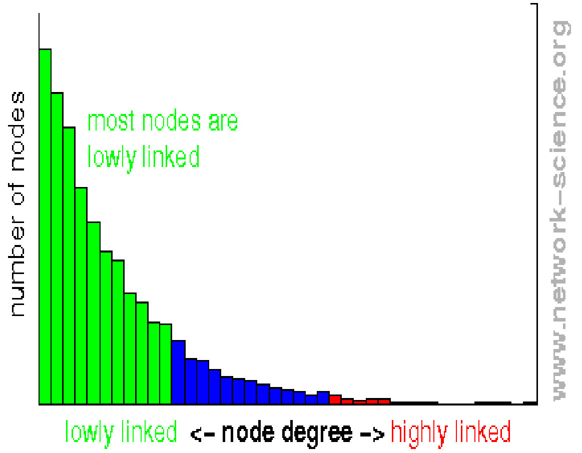
http://socilab.com/#home



Random networks: Poisson distribution



Organized networks: power-law / scale-free



1. Degree Distribution points to network's randomness
2. Be careful comparing networks.
3. Be careful analyzing networks: what works for one, may fail for another.
4. Stay on top of research.

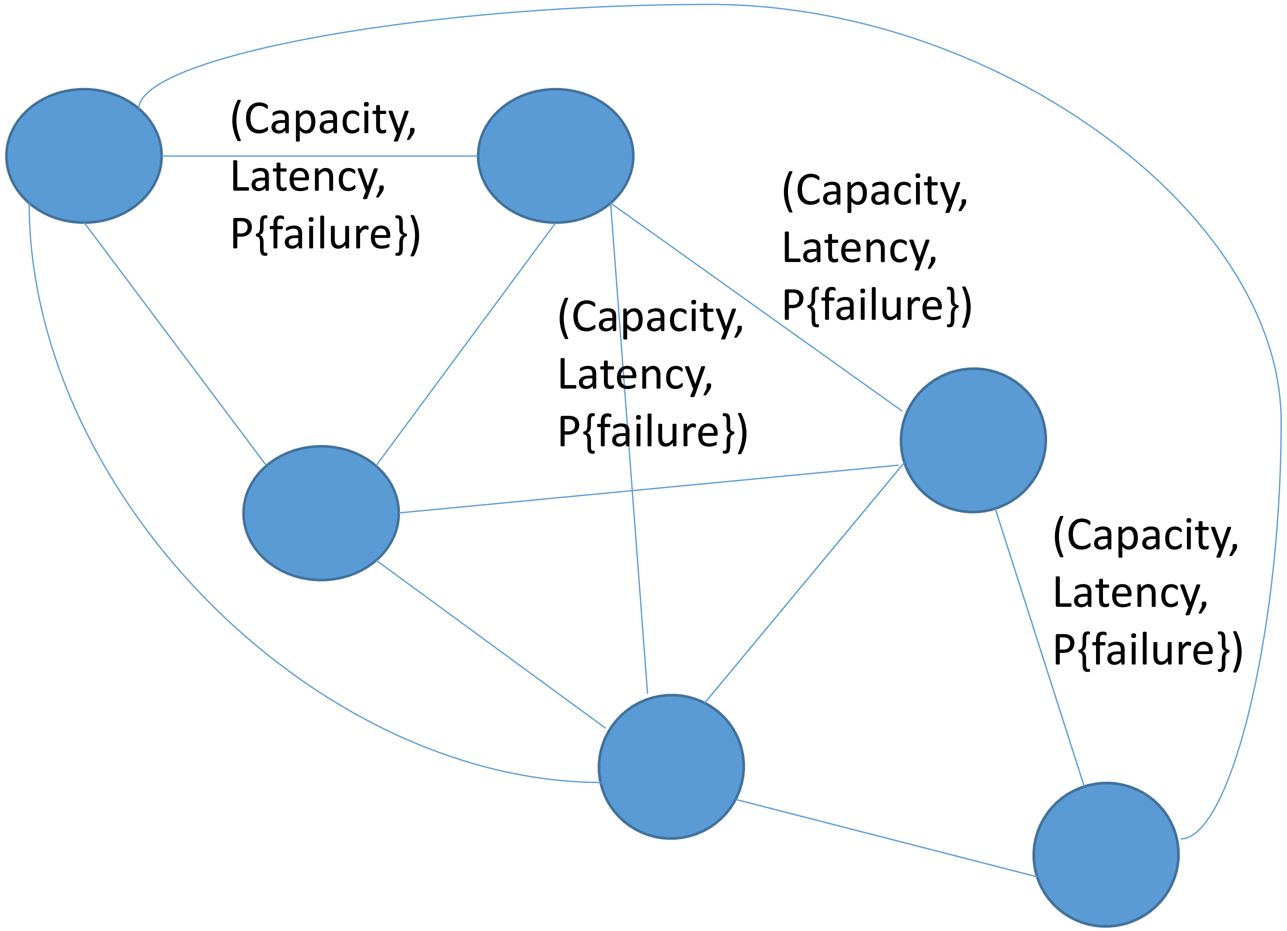
1. Use Non-Parametric Analysis.
2. Use Monte-Carlo Analysis.
3. Avoid Assumptions!

(From http://www.network-science.org/powerlaw_scalefree_node_degree_distribution.html)

Growing Directions in Network Research

- Hematology (angionetworks)
- Physics & Chemistry
- Network Resilience Analysis
  - Centrality & Reliability
  - Centrality & Phase Transition (in physics)
- Network Dynamics
  - Sociology
  - Law Enforcement
  - Computer Science:
    - GitHub / Perforce / SVN / CVS / …
    - Internet
  - Transport and Communication Networks:
    - Aviation / Automotive / Railroad / Waterways
    - Television / Mobile / Fixed-Line Telephone
  - Water Supply Networks:
    - Macro-level: water distribution
    - Micro-level: water permeation through the soil

Applications In the IT world



Link_ID	Capacity	Latency	P{failure}
a12	100	100	1.00E-05
a13	50	65	1.00E-05
a14	500	128	5.00E-06
a23	700	17	1.00E-04
a24	640	250	7.00E-05
a25	480	78	8.00E-06
…	…	…	…
a34	300	360	8.00E-03
a37	80	175	9.00E-07

Goal: Configure and operate the network to maximize QoS for all users

The following 24 slides are from *Alex Gilgur, Brian Eck. "Sources of Traffic Demand Variability and Use of Monte Carlo for Network Capacity Planning" – Performance and Capacity by CMG International Conference (CMG'14) – Atlanta, GA 2014.*

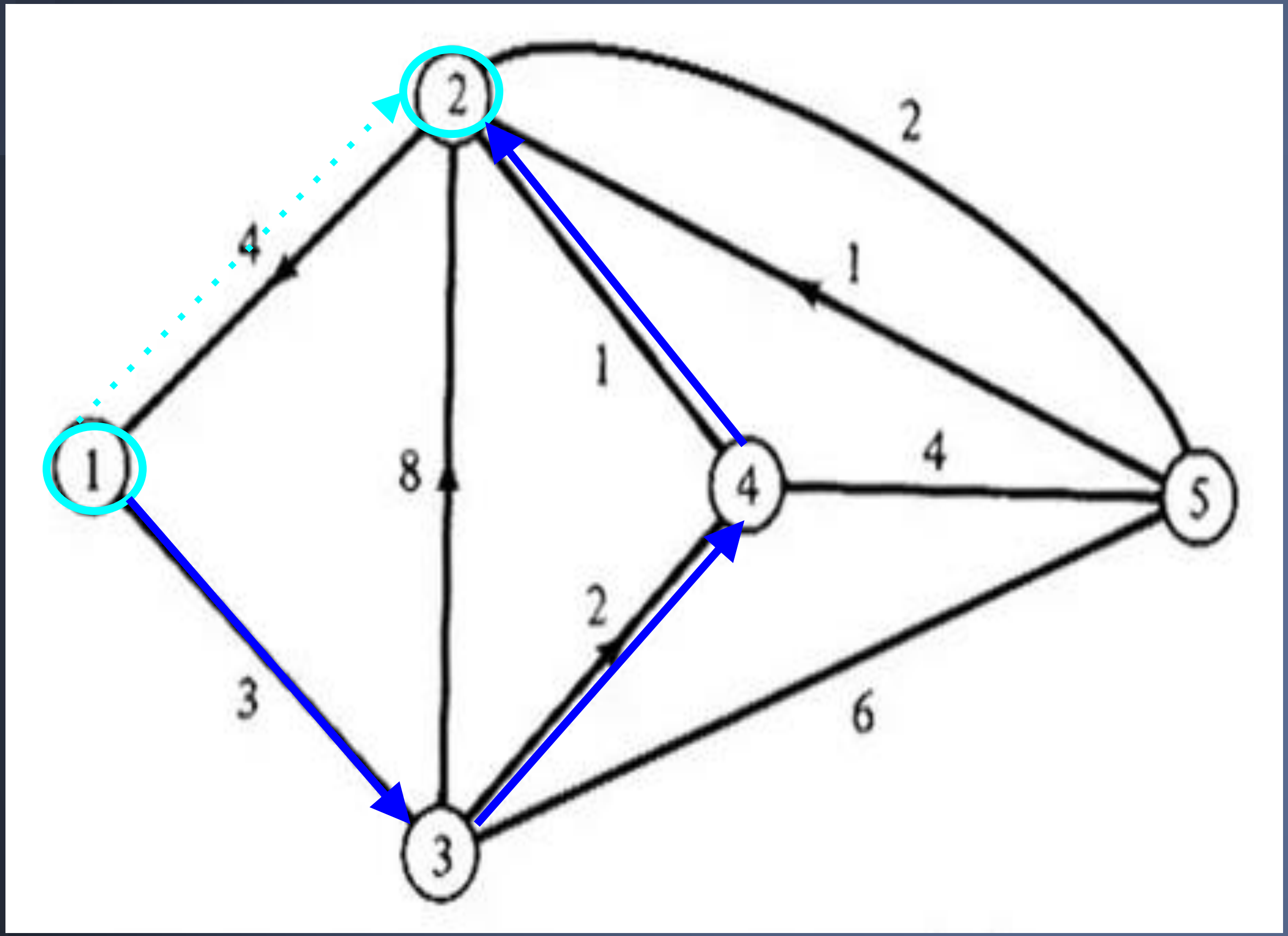
QoS = what's important to user

1. High Probability of:
    a. Delivery
    b. Accuracy

2. Low Latency

1. QoS = "Goodput" = Throughput * Pr{delivery}
2. QoS = 1 / Latency

Routing for Low Latency: SPF: “Travelling Salesman”



= Node 4

= “Latency of this link = 2 units”

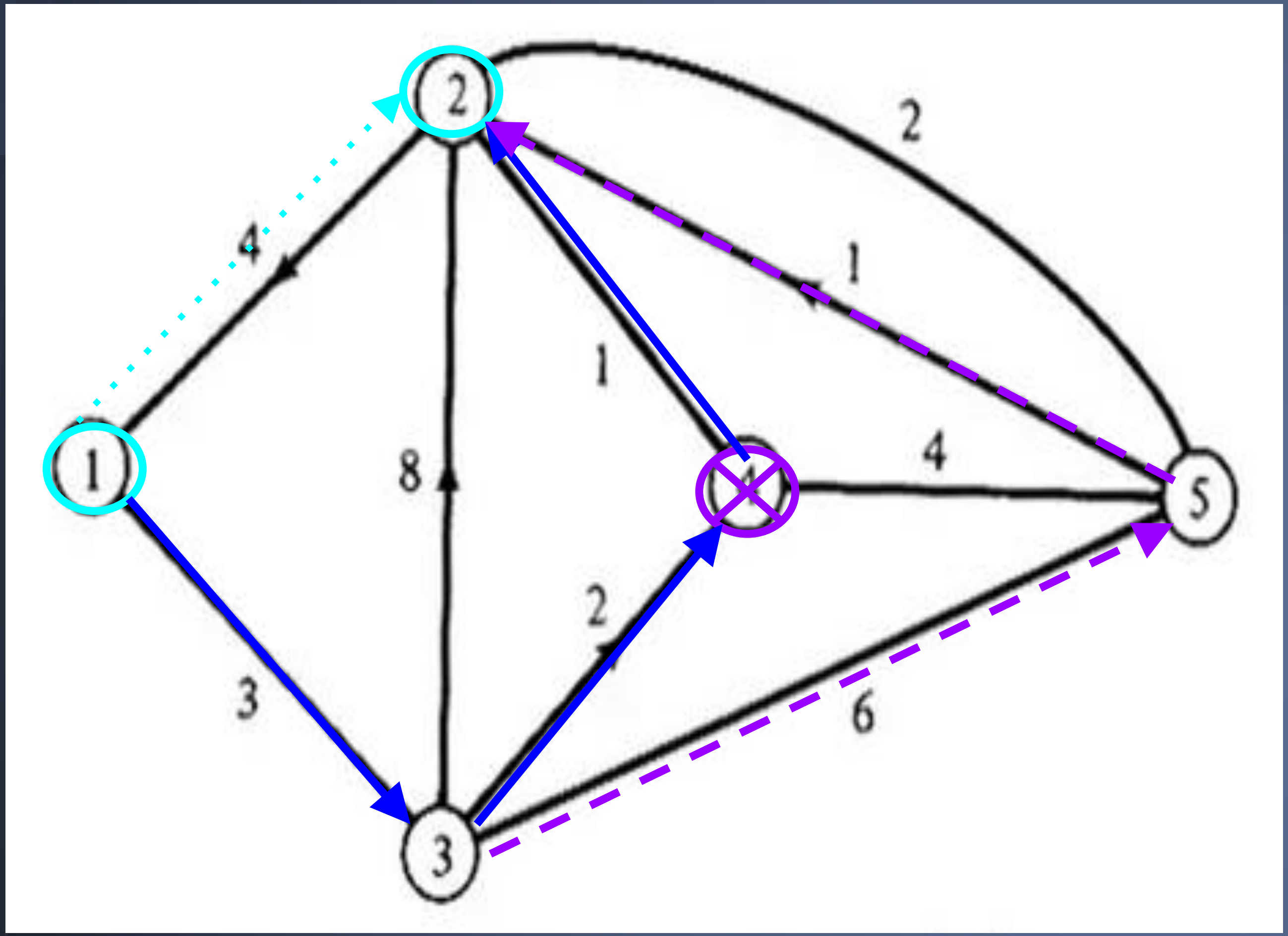
$$Cost_{Path} = \sum_{Link_1}^{Link_n} Cost_{Link} = \sum_{Link_1}^{Link_n} Latency_{Link}$$

Find shortest path from Node 1 to Node 2

Cost = Latency

QoS = 1/Cost = 1/Latency

Routing for Low Latency: SPF: “Travelling Salesman”



= Node 4

= “Latency of this link = 2 units”

$$Cost_{Path} = \sum_{Link_1}^{Link_n} Cost_{Link} = \sum_{Link_1}^{Link_n} Latency_{Link}$$

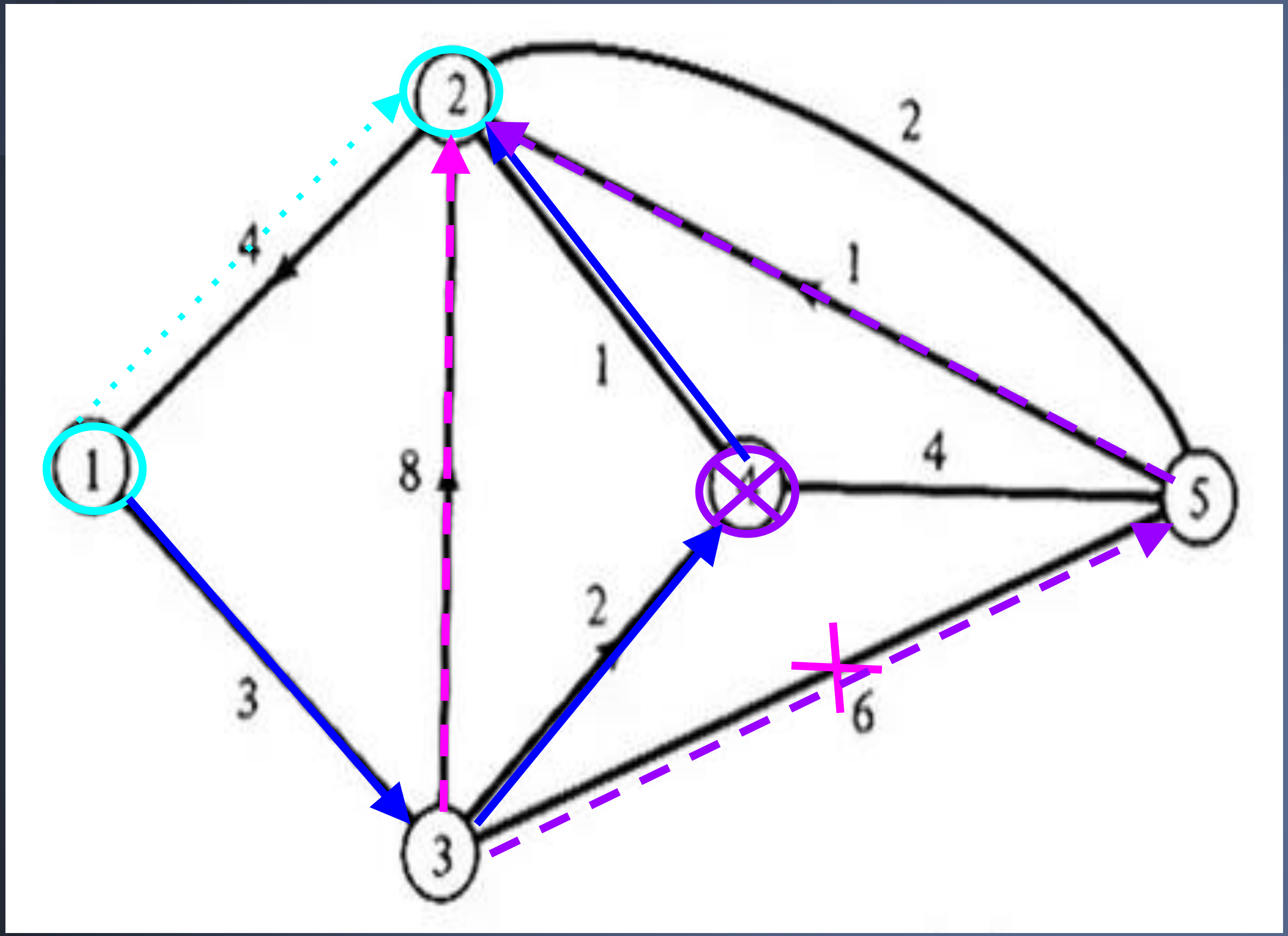
Find shortest path from Node 1 to Node 2

Find shortest path from Node 1 to Node 2 **IF** Node 4 is down

Cost = Latency

QoS = 1/Cost = 1/Latency

Routing for Low Latency: SPF: “Travelling Salesman”



= Node 4

= “Latency of this link = 2 units”

$$Cost_{Path} = \sum_{Link_1}^{Link_n} Cost_{Link} = \sum_{Link_1}^{Link_n} Latency_{Link}$$

Find shortest path from Node 1 to Node 2

Find shortest path from Node 1 to Node 2 **IF** Node 4 is down **...**

Cost = Latency

QoS = 1/Cost = 1/Latency

… **and** Link 3-5 is losing packets

QoS = what's important to user

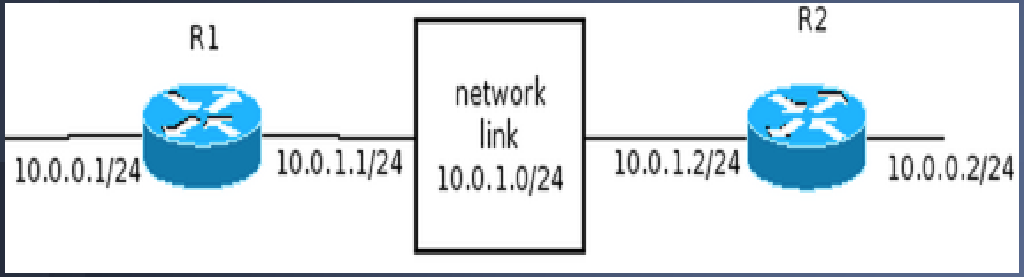
1. High Probability of:
   a. Delivery
   b. Accuracy

2. Low Latency

1. QoS = "Goodput" = Throughput * Pr{delivery}

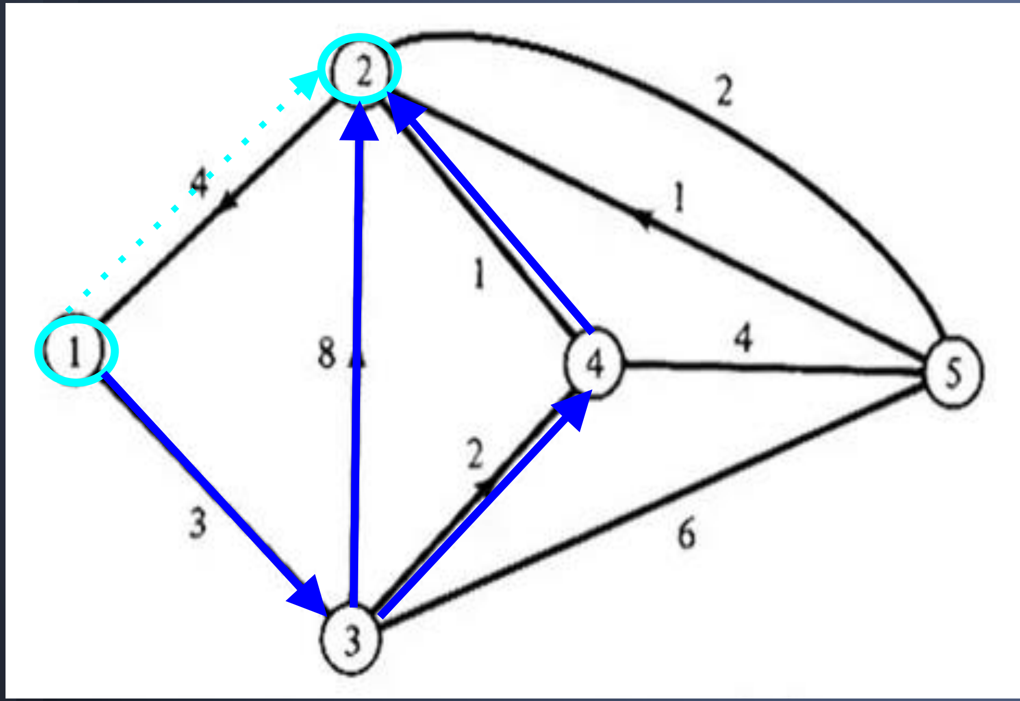
2. QoS = 1 / Latency

Routing for "Goodput": Nonlinear optimization



$$Pr\{delivery_{\ Link}\} = Pr\{NOT\ blocking\ \&\ NOT\ failure\}$$

$$P_D^{Link} = (1 - P_B^{Link}) * (1 - P_F^{Link})$$



$$P_D^{Path} = \prod_{Link_1}^{Link_n} P_D^{Link}$$

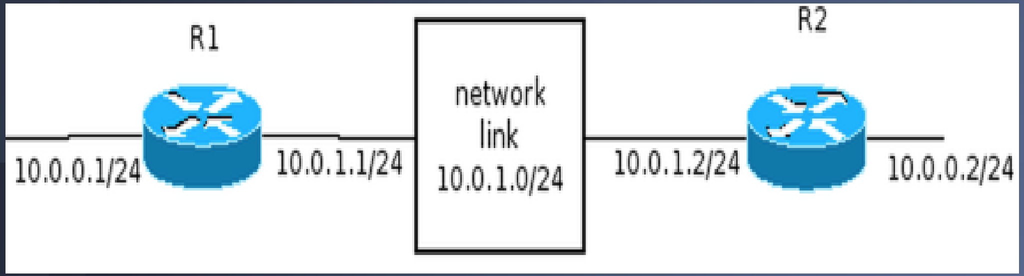
$$Cost_{\ Path} = \frac{1}{X} * \{1 - \prod_{Link_1}^{Link_n} P_D^{Link}\}$$

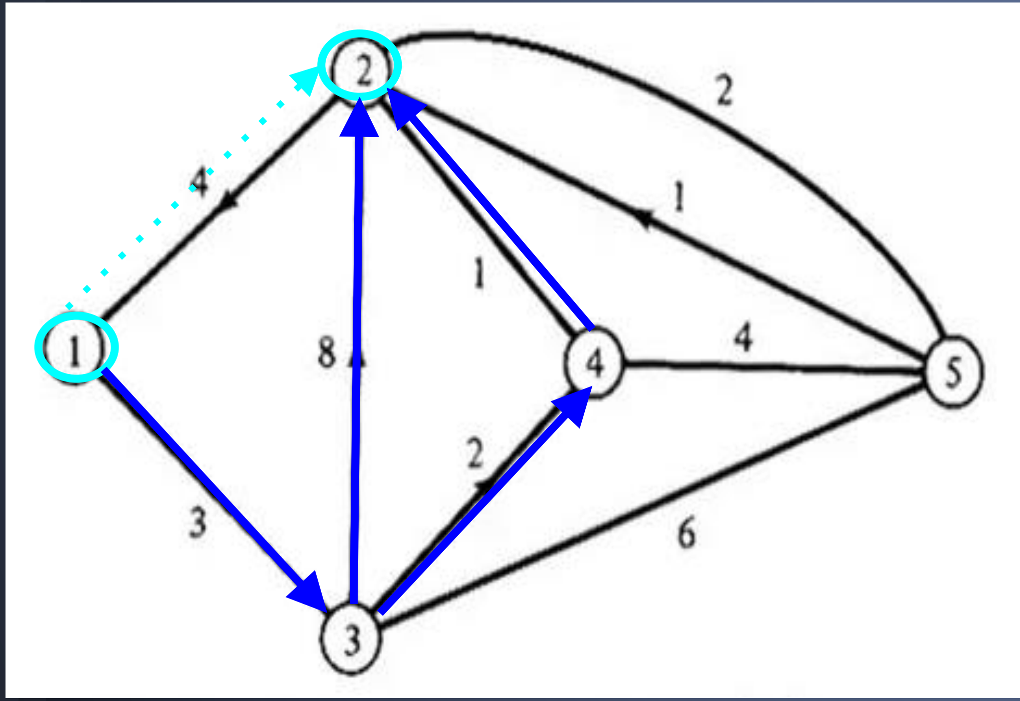
$$Cost_{\ Path} = \frac{1}{X} * \{1 - \prod_{Link_1}^{Link_n} [(1 - P_B^{Link}) * (1 - P_F^{Link})]\}$$

"~~Travelling Salesman~~"

Non-linear optimization

Routing for "Goodput": Can it be simplified?





$$P_D^{Link} = (1 - P_B^{Link}) * (1 - P_F^{Link})$$

Assume:

- No Queueing
  - No Blocking

$$P_B^{Link} = 0$$

$$P_D^{Link} = 1 - P_F^{Link}$$

$$Cost_{Link} = \frac{1}{X} * \{1 - P_D^{Link}\}$$

$$= \frac{1}{X} * P_F^{Link}$$

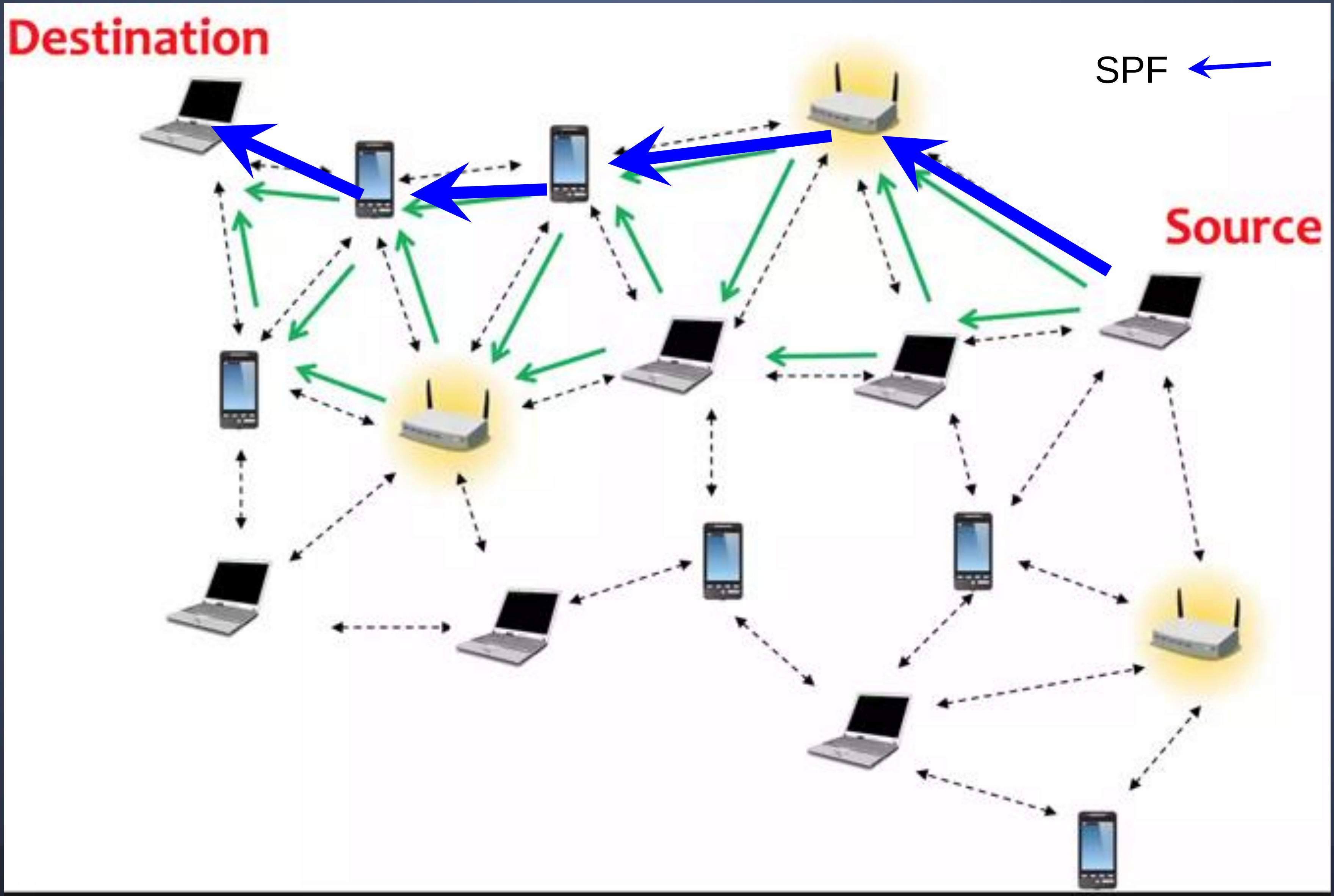
Redefine:

$$ln(Cost) = ln(P_F) - ln(X)$$

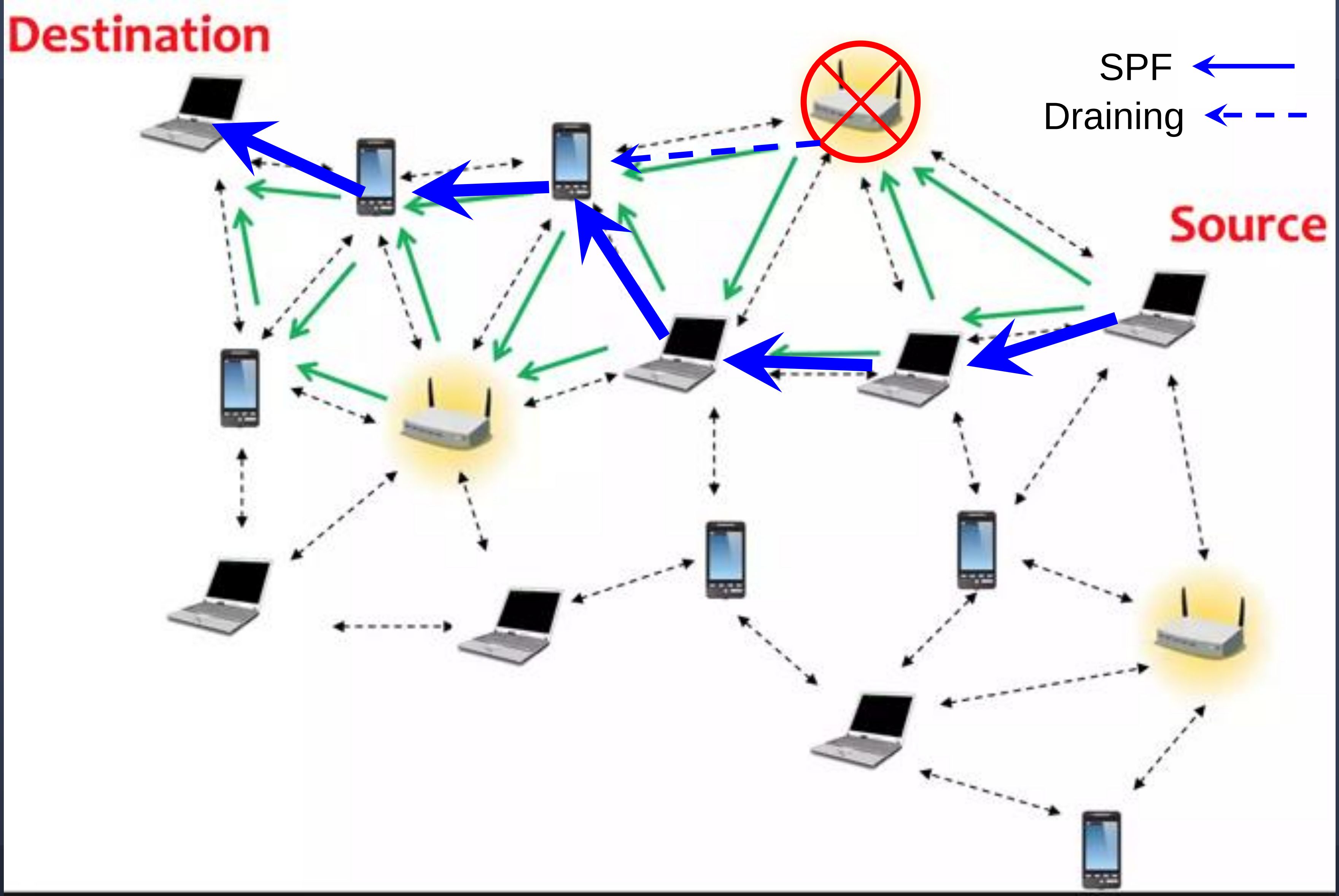
Non-linear optimization

Can be pseudo-linearized

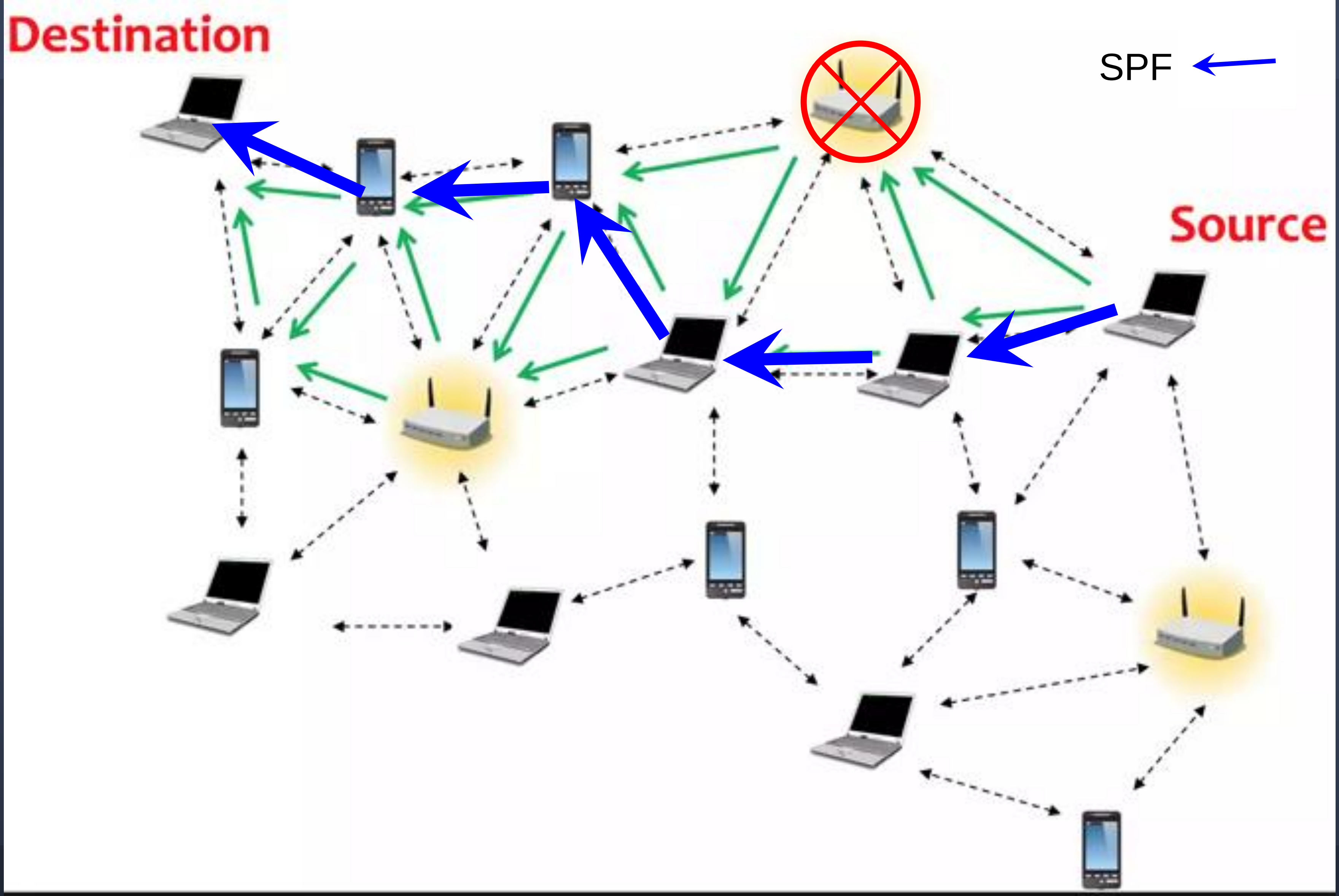
Routing As a Process



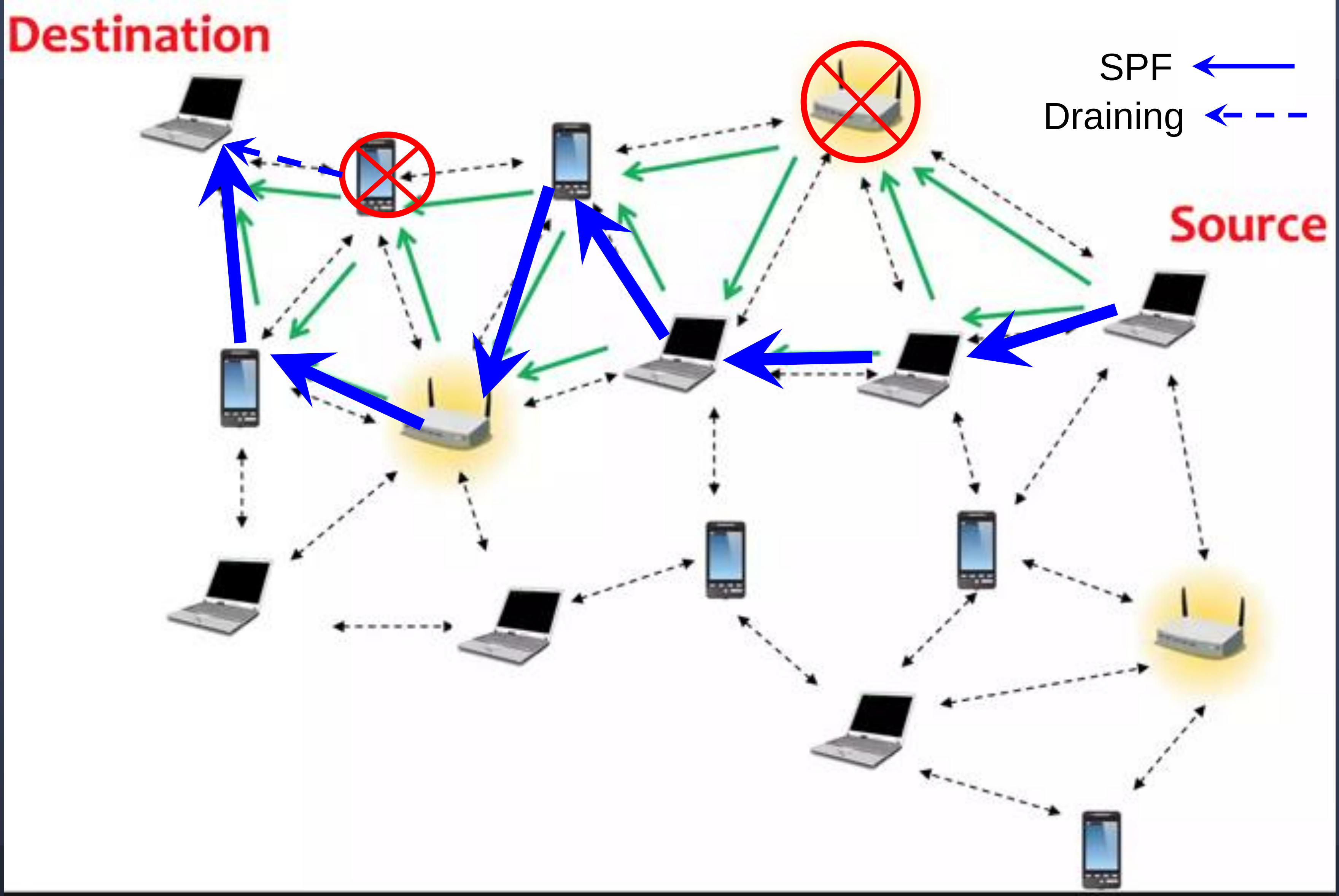
Routing As a Process



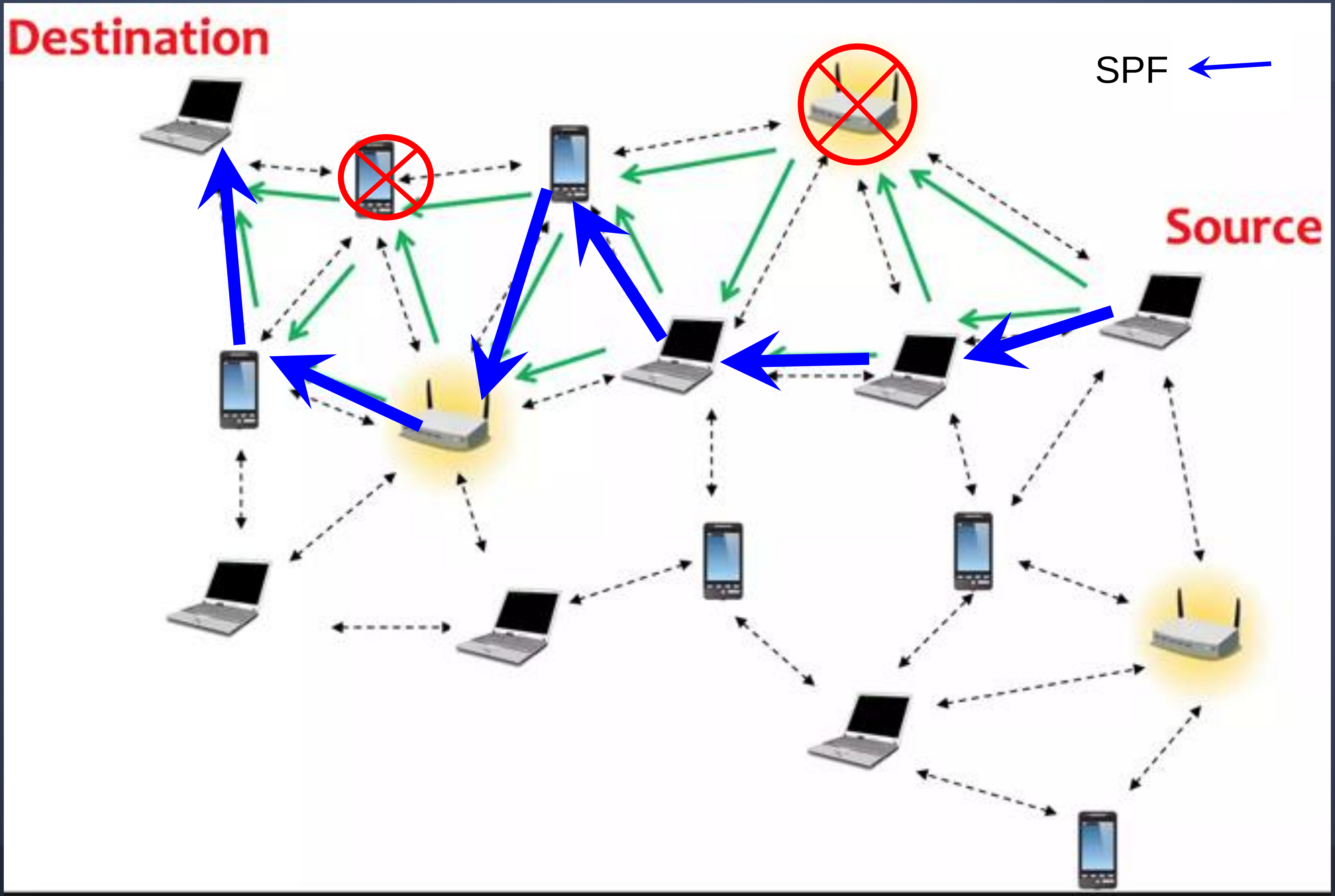
Routing As a Process



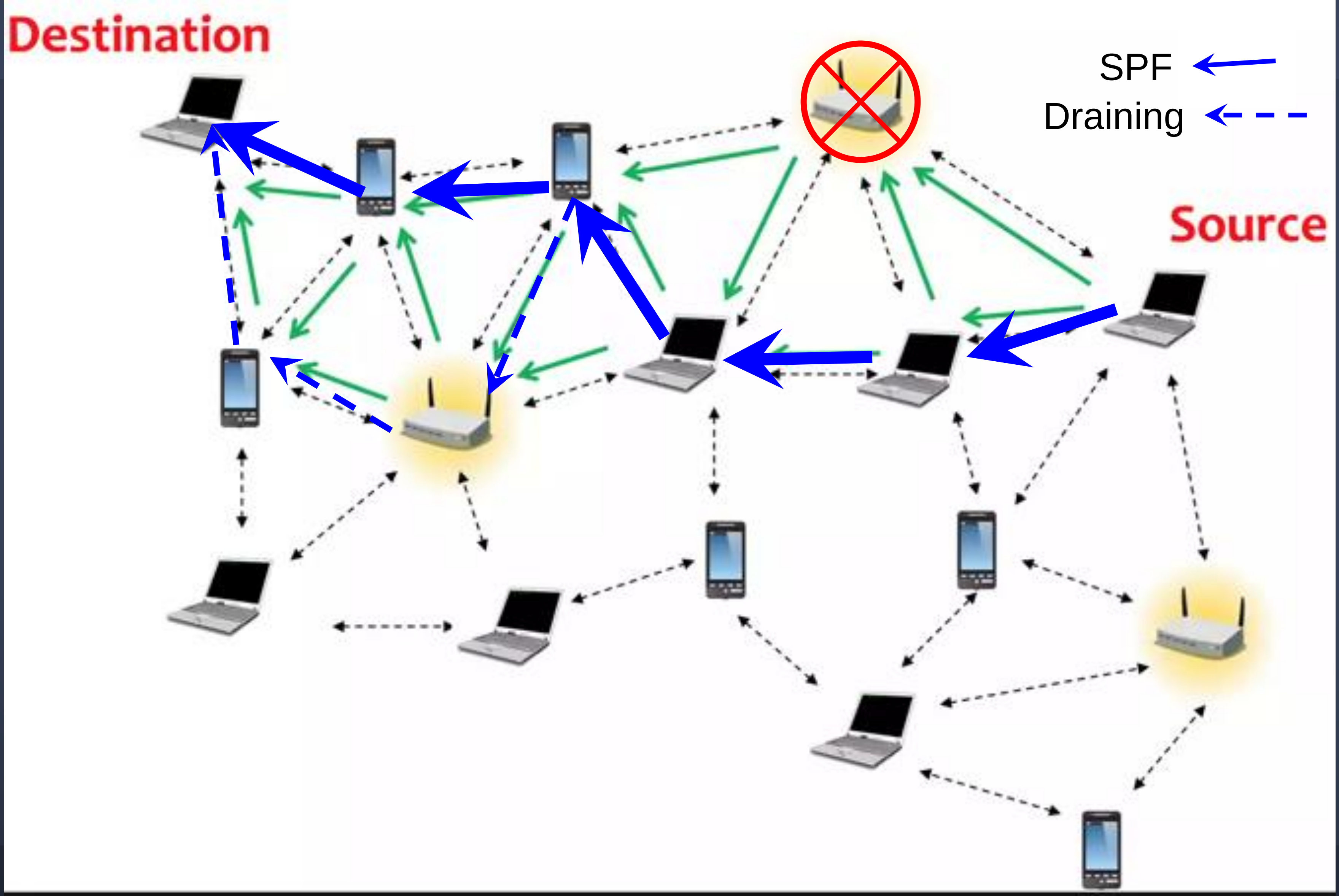
Routing As a Process



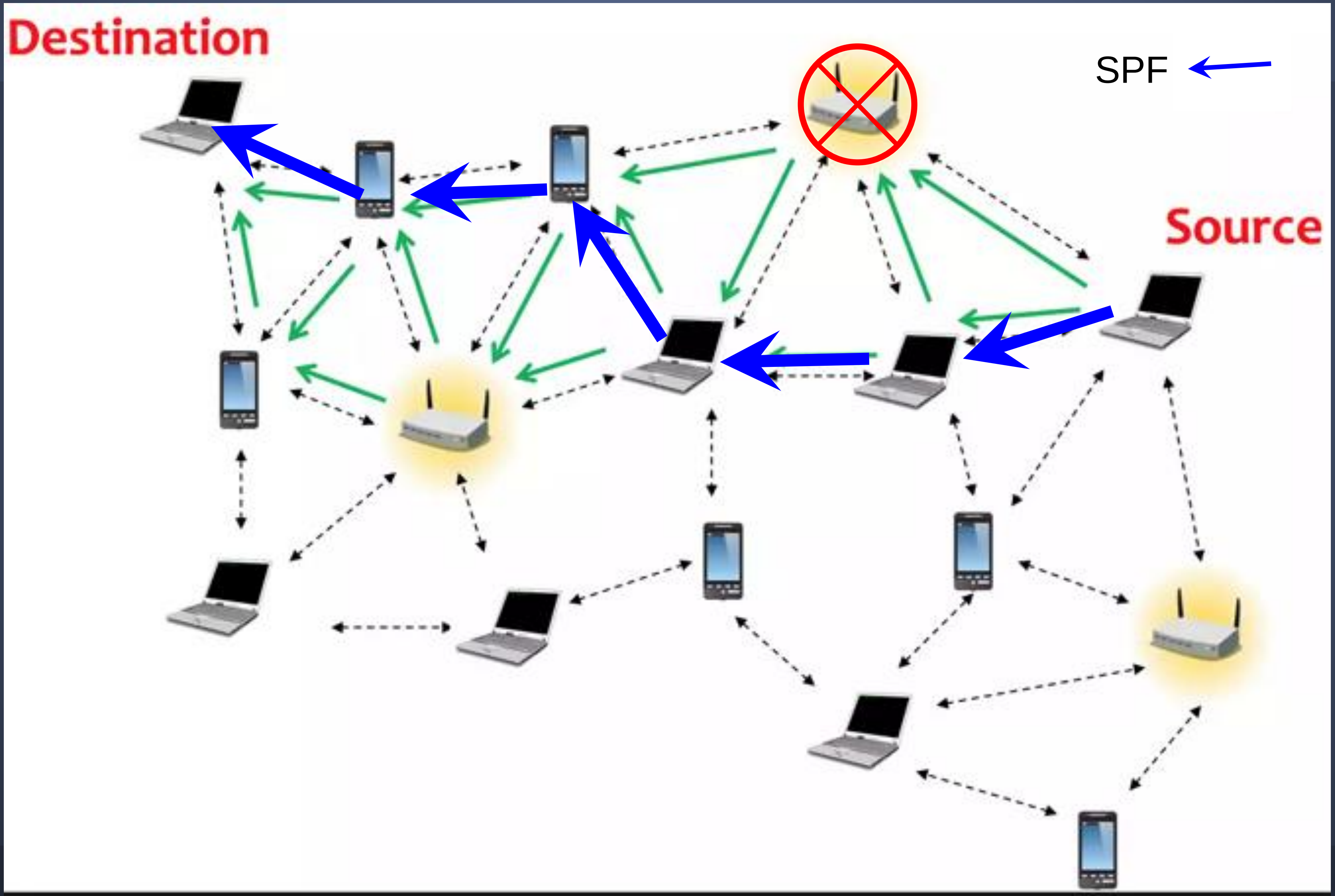
Routing As a Process



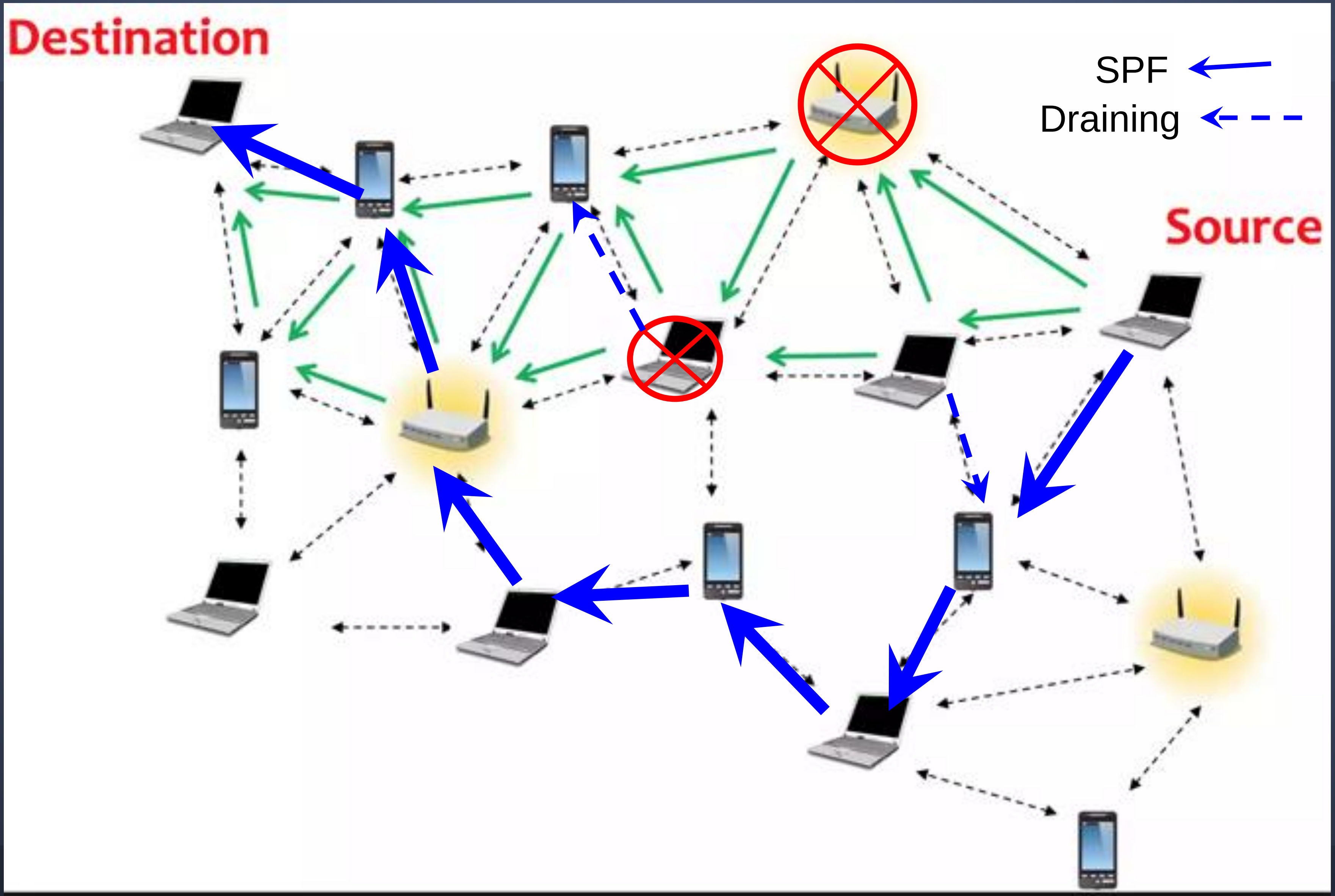
Routing As a Process



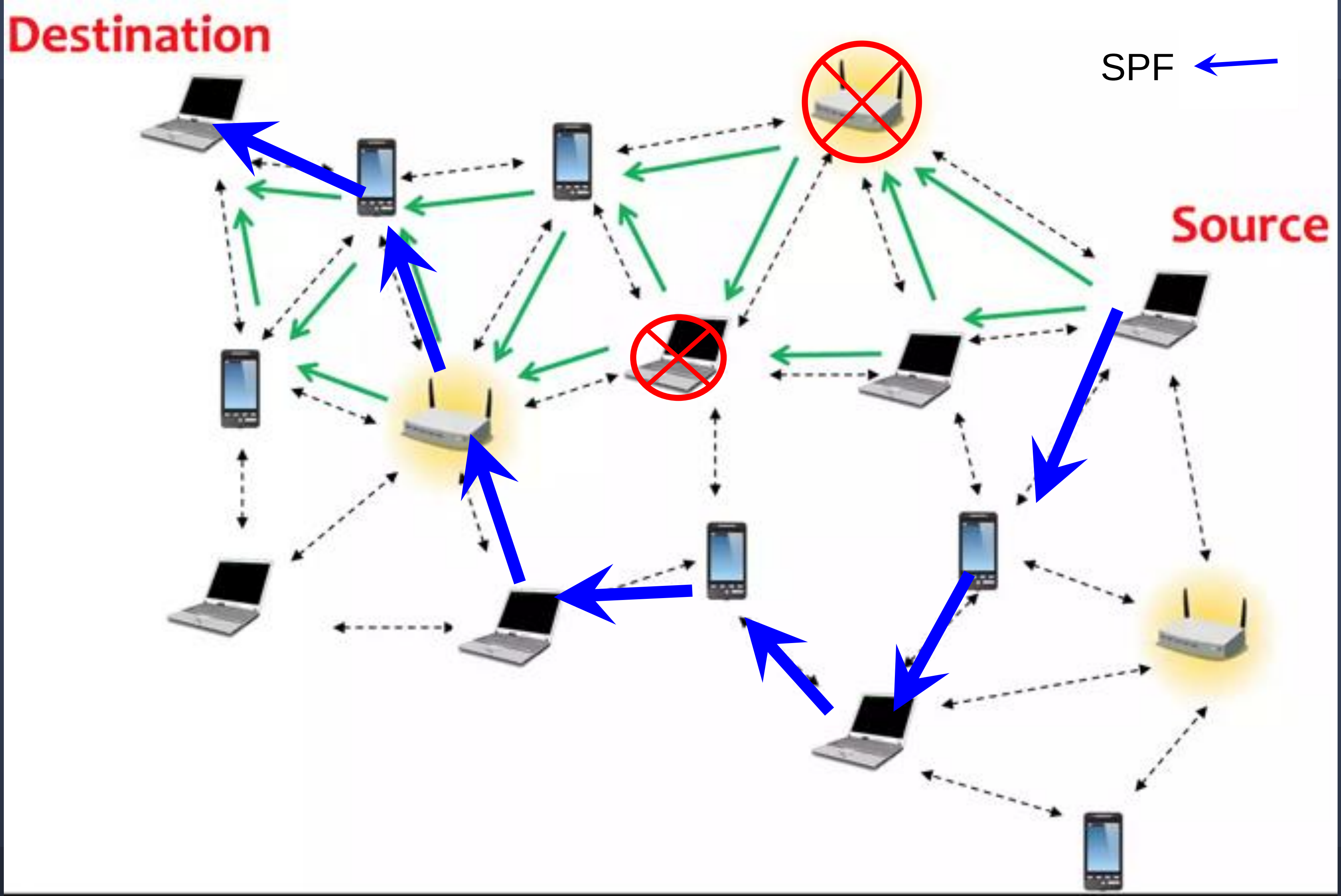
Routing As a Process



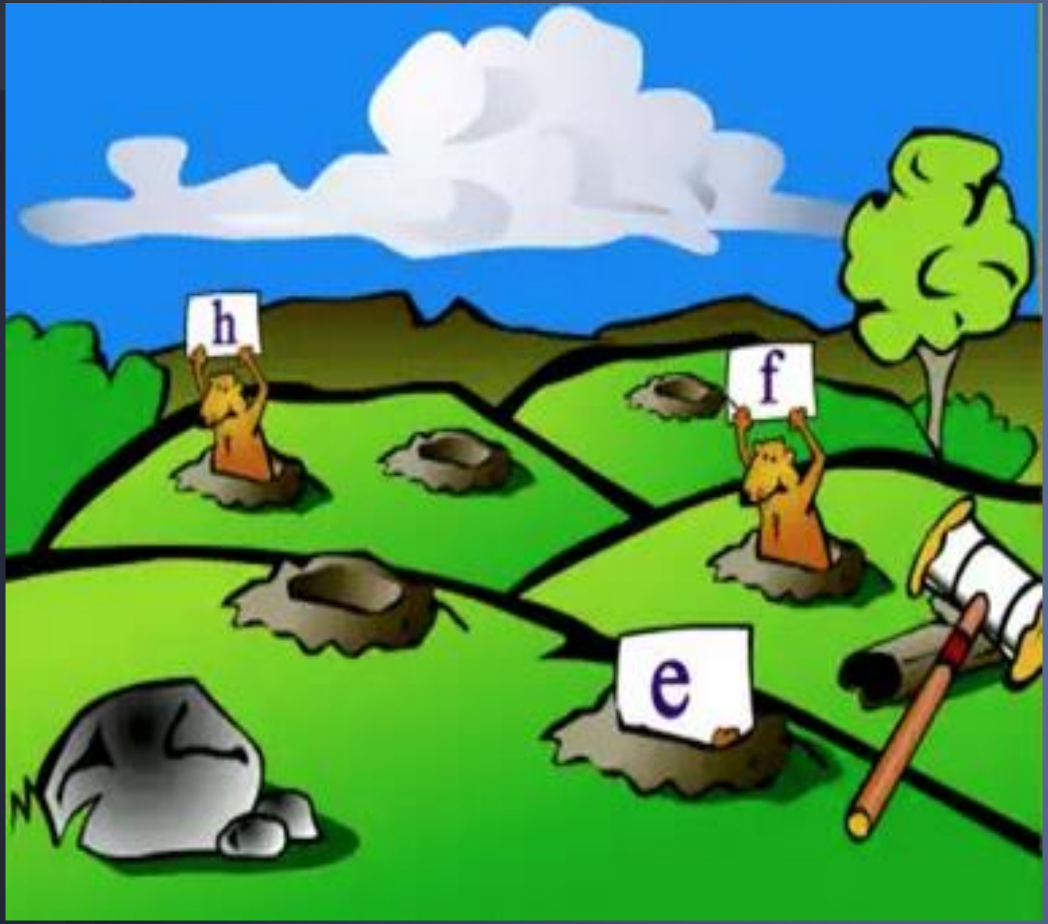
Routing As a Process



Routing As a Process



“Whack-a-Mole!”



Routing is updated all the time via:

- Protocol (e.g., TCP)
- SDN Control

We need to accommodate each Flow’s:

- Primary Paths
- Alternate Paths

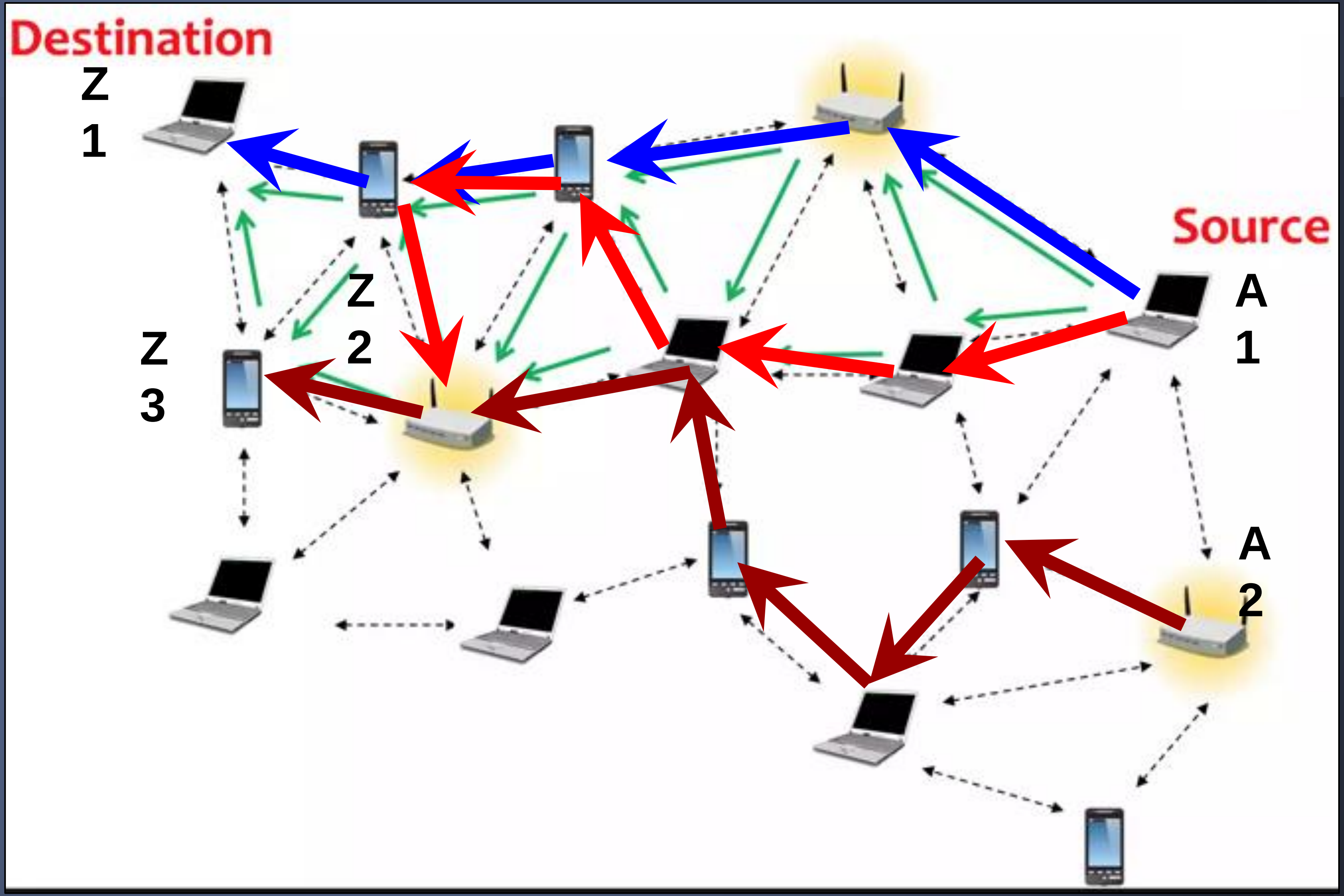
We can forecast Demand

Demand:

- A1 -> Z1 : X11 Gbps
- A1 -> Z2 : X12 Gbps
- A2 -> Z3 : X23 Gbps

Throughput on each Link

Capacity for each Link



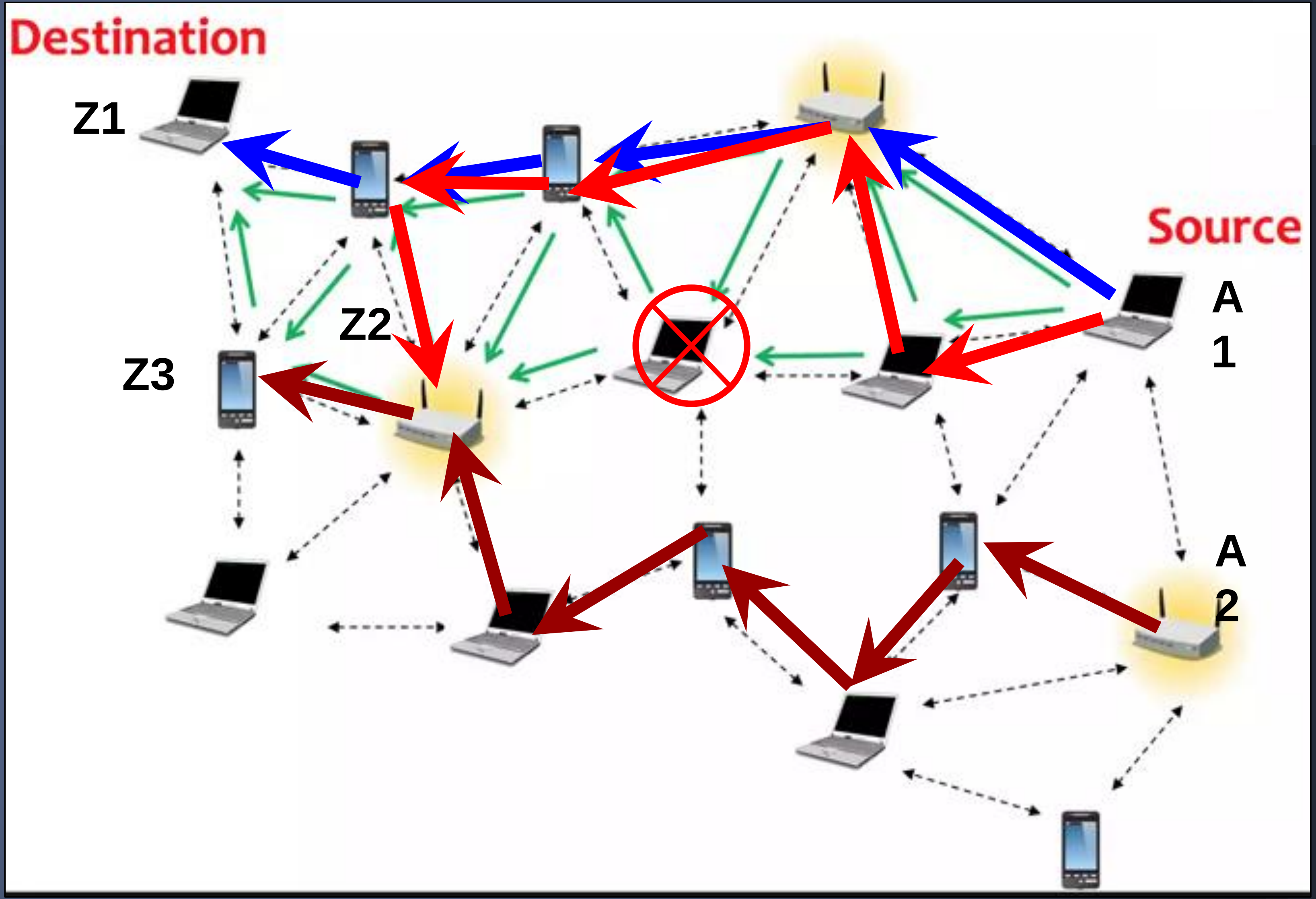
We can forecast Demand

Demand:

- A1 -> Z1 : X11 Gbps
- A1 -> Z2 : X12 Gbps
- A2 -> Z3 : X23 Gbps

Throughput on each Link

Capacity for each Link

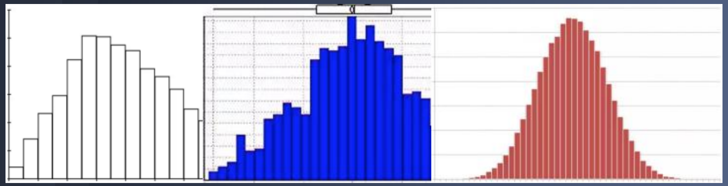


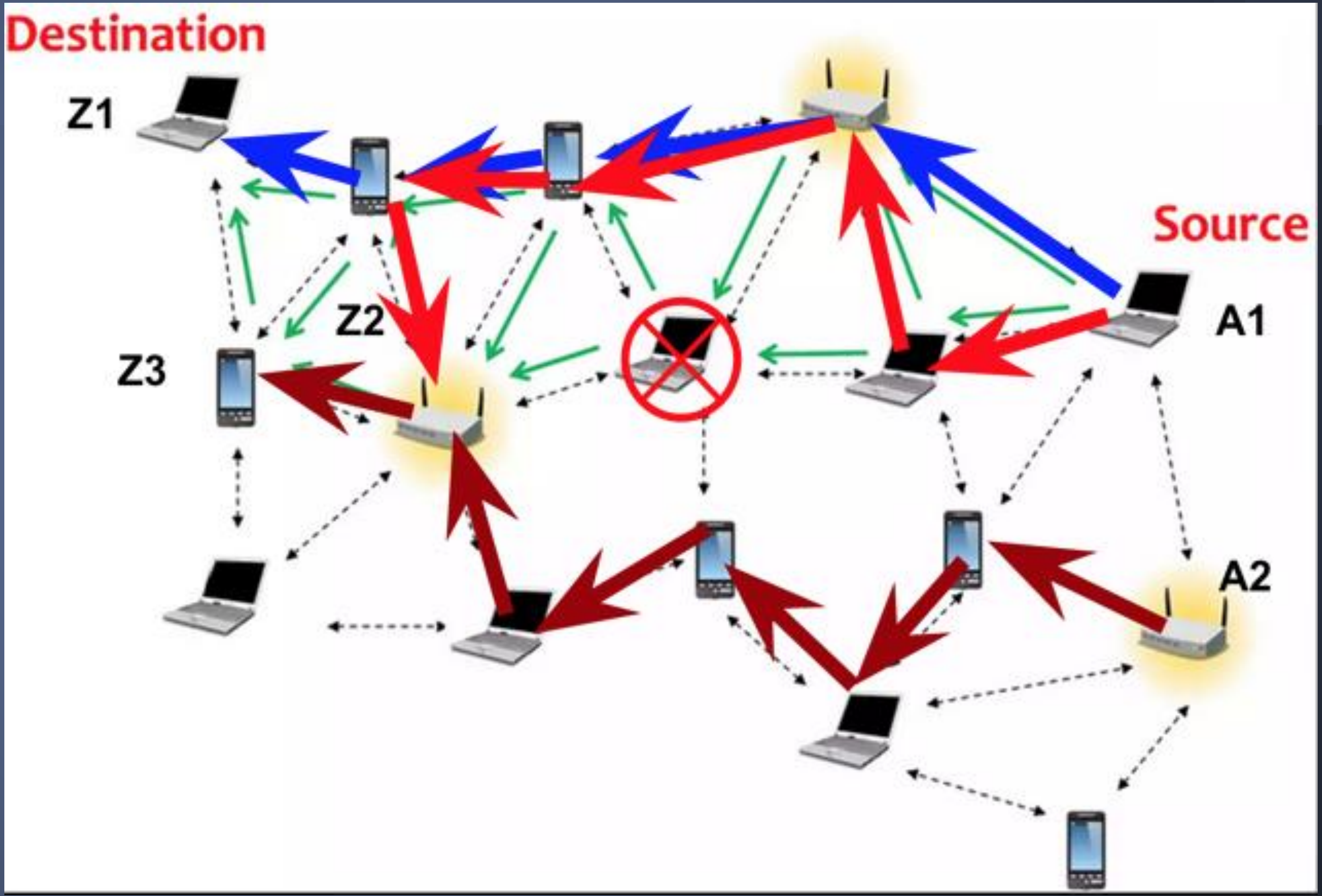
Throughput is combinatorial

Demand is NOT Deterministic

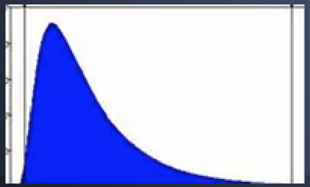
Demand:

- A1 -> Z1 : X11 Gbps
- A1 -> Z2 : X12 Gbps
- A2 -> Z3 : X23 Gbps





Throughput on each Link

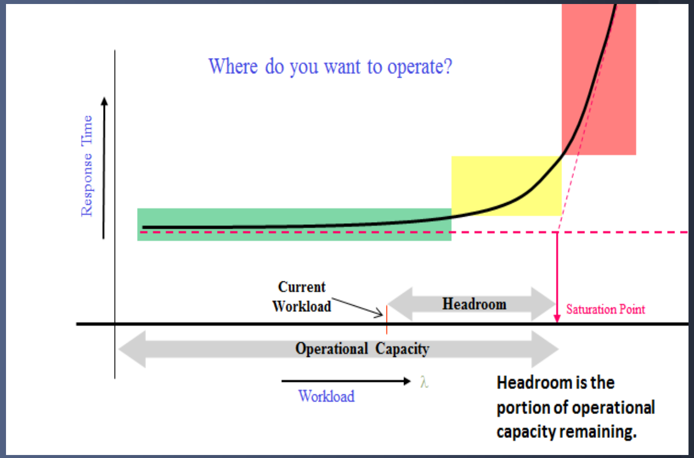


Neither is Throughput

Demand Predictability

- Not all forecasting tools were created equal:
  - Non-Gaussian distributions
  - Non-stationarity
  - Congestion Control

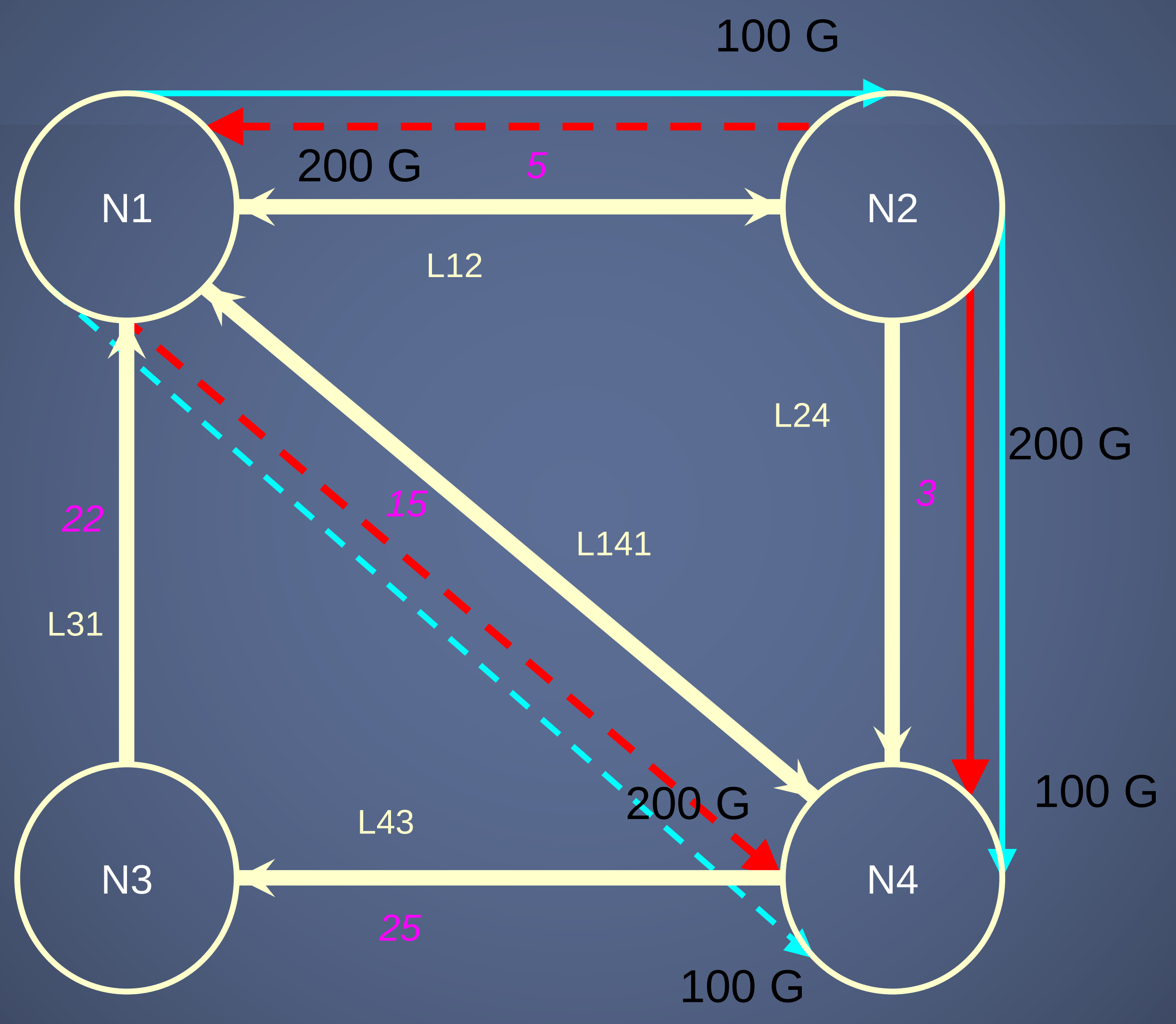
- TSA is not the only way to forecast Demand:
  - Explanatory variables:
    - Timestamp
    - Power
    - CPU
    - Business Metrics



"All models are wrong. Some models are useful" - G.E.P. Box

From Deterministic Demand to Throughput

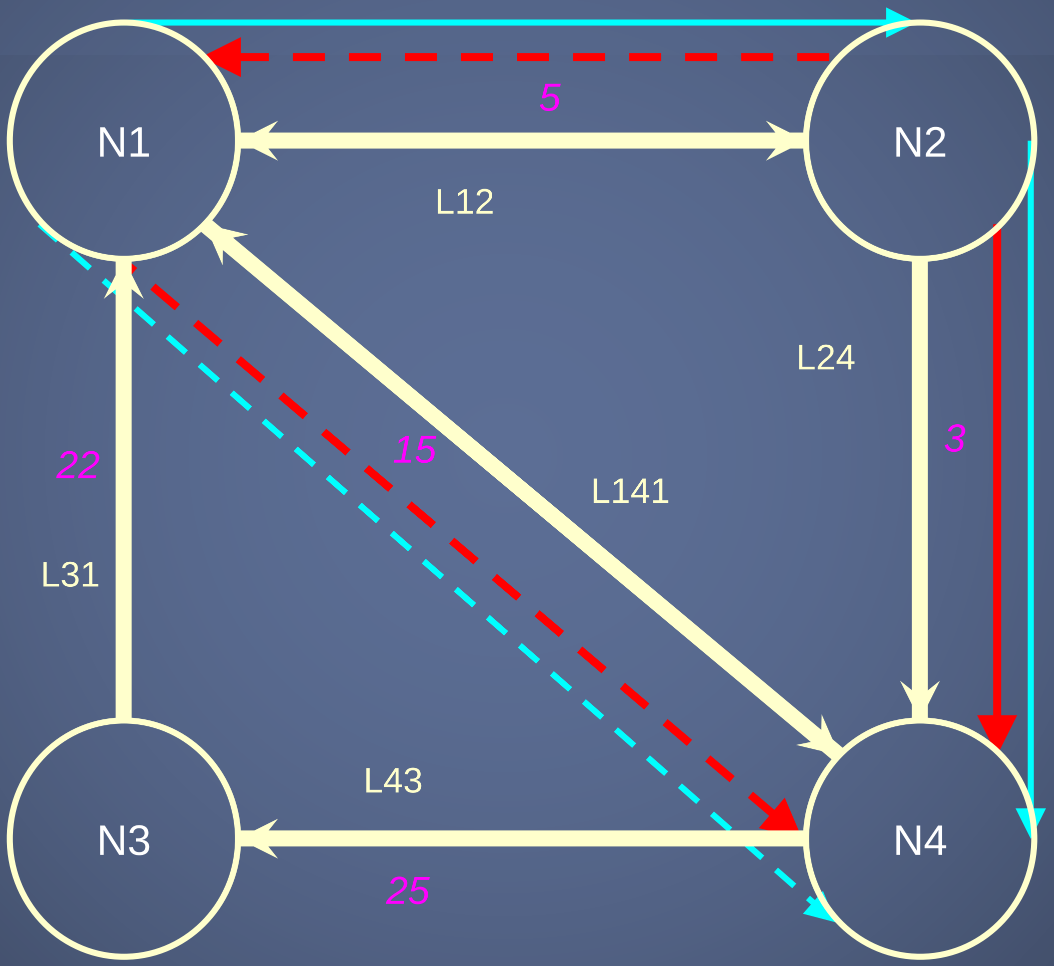
Demand:
N1_N4: 100 G
N2_N4: 200 G



Throughput:
L12 = 100 G
L21 = 200 G
L24 = 300 G
L14 = 300 G
L41 = 0
L43 = 0
L31 = 0

From Gaussian Demand to Throughput:

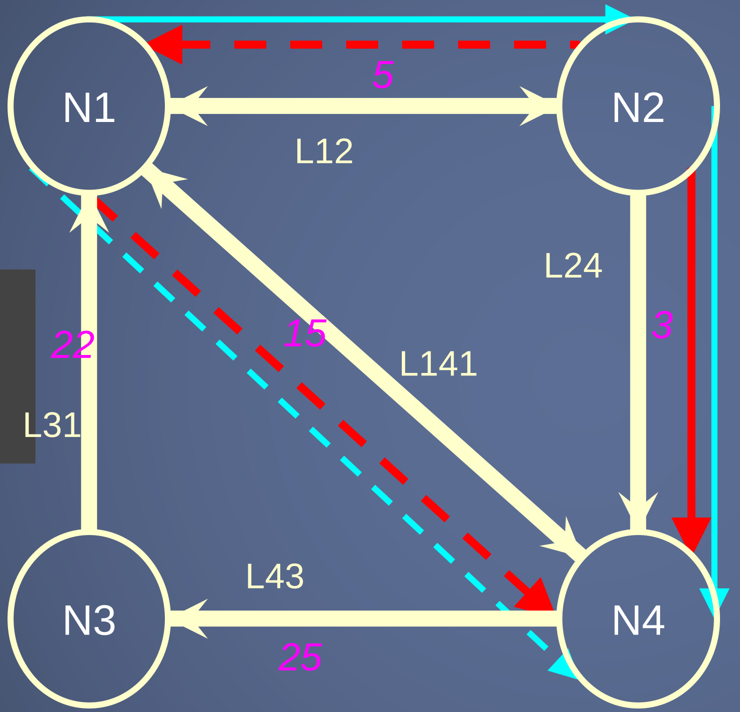
Demand:
N1_N4: N (100, 10) G
N2_N4: N (200, 15) G

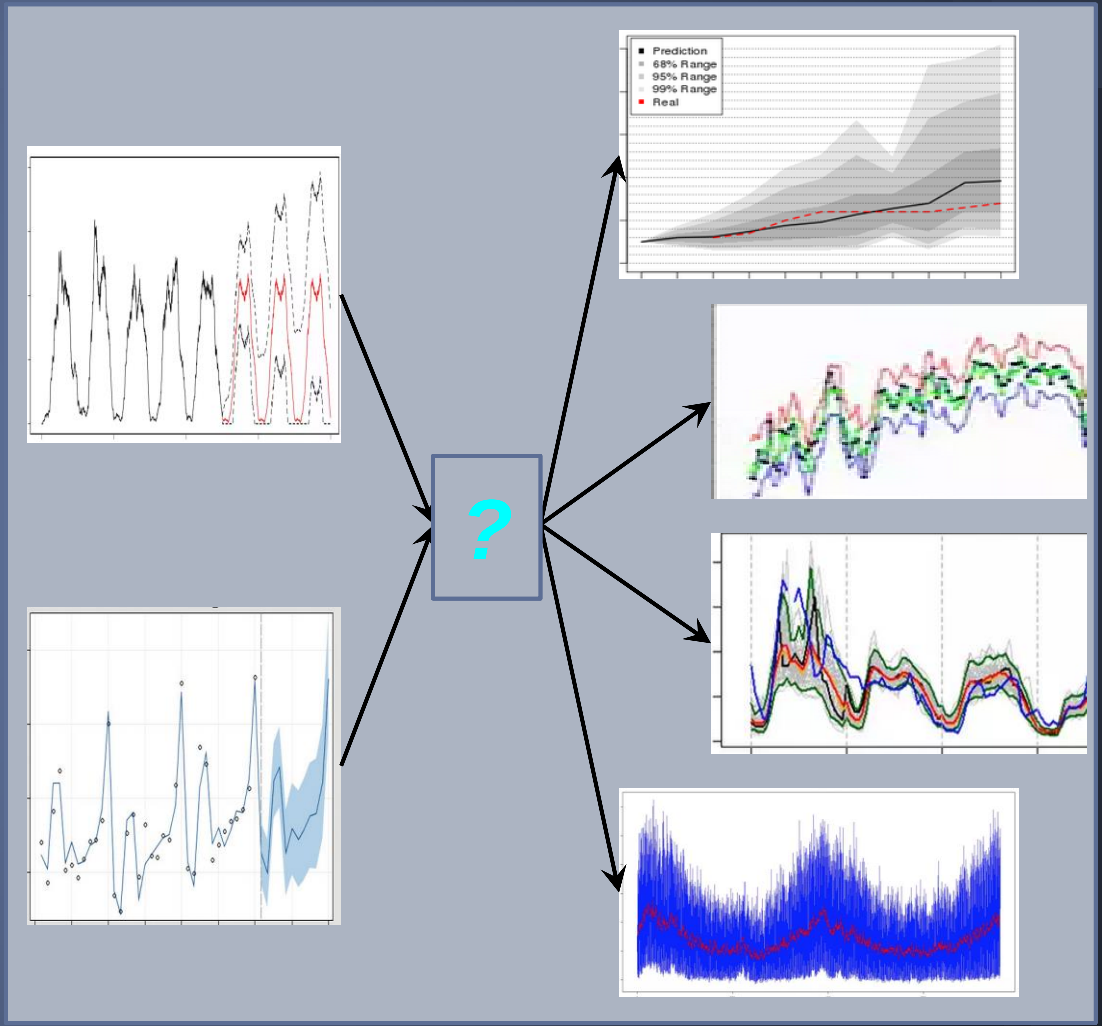


Throughput:
L12 = N (100, 10) G
L21 = N (200, 15) G
L24 = N (300, 18) G
L14 = N (300, 18) G
L41 = 0
L43 = 0
L31 = 0

From Generic Random Demand to Throughput:

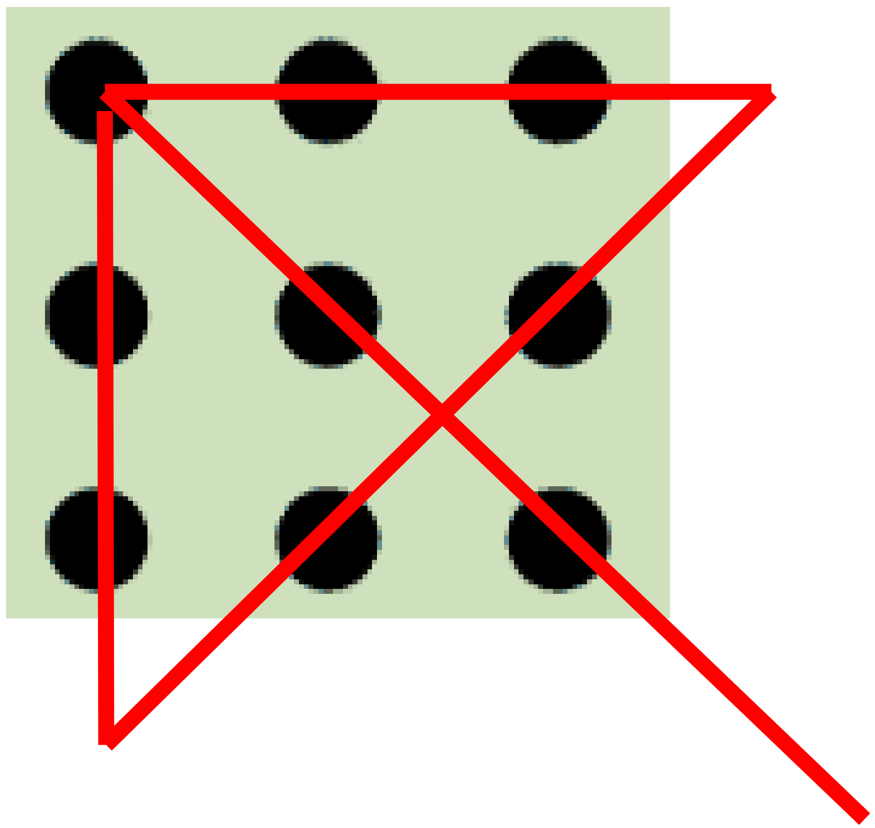
Demand:
N1_N4: G (100, ...) G
N2_N4: G (200, ...) G





Solution: Monte-Carlo

Solution to the problem on Slide 1:



More to Explore: https://www.math.washington.edu/~morrow/336_11/papers/leo.pdf