EL736 Communications Networks II: Design and Algorithms

Class7: Location and Topological Design

Yong Liu

10/24/2007

Polytechnic University
Celebrating 150 Years in Brooklyn

Outline

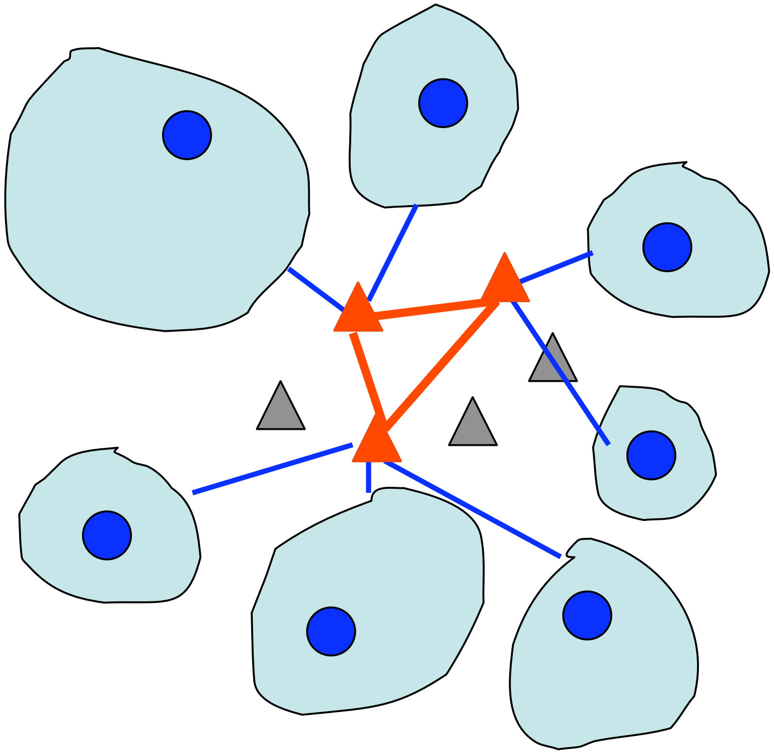
- Topological Design Modeling
  - location
  - connectivity
  - demand/capacity allocation
- Solution
  - heuristic algorithms

Topological Design

- So far deal with link dimensioning
  - links already in place
  - cost increases with link capacity
- Topological Design
  - long-term network planning stage
  - where to place nodes/links
  - installation/opening cost capacity independent
- Two Types of Topological Design
  - pure location: node/link placement to achieve a desirable topology, demand volume not in consideration
  - location & flow: node/link placement+link capacity to realize demands at minimum installation+dimensioning cost

Node Location Problem

- connect N areas through M possible locations
- one access link per area
- maximal connection per location
  - maximum number of ports per router
- cost of opening each location
- cost of connecting each area to each location
- always connect to the cheapest/closest location?



NLD: Formulation

Node Location Design

- **indices**

  $i = 1, 2, ..., N$ areas to be connected

  $j = 1, 2, ..., M$ possible locations for nodes

- **constants**

  ξ_{ij} cost of connecting area i to possible location j

  η_j cost of location j, if opened

  K_j maximum number of areas that can be handled at possible location j

- **variables**

  u_{ij} $= 1$, if area i is connected to location j; 0, otherwise

  r_j $= 1$ if a node is decided to be located at site j; 0, otherwise

- **objective**

  mininimize $\boldsymbol{F} = \sum_i \sum_j \xi_{ij} u_{ij} + \sum_j \eta_j r_j$

- **constraints**

  $\sum_j u_{ij} = 1, \qquad i = 1, 2, ..., N$

  $\sum_i u_{ij} \leq K_j \, r_j, \qquad j = 1, 2, ..., M.$

Add Heuristic

- Local Search Algorithm
- Start with a random location
  - all sources connect to that location
- Iteratively add new locations, one each time
  - to reduce the opening and connection cost
  - choose the candidate with largest reduction
- Stop if cost reduction is not possible

Add Heuristic: the algorithm

ALGORITHM 6.1: Add Heuristic for Node Location Design Problem

Step 0 Select an initial location $\hat{j}$ and assume that all areas are connected to this location. Set $\mathcal{S}_0 = \{\hat{j}\}$. and iteration count to $k = 0$. Compute cost $\boldsymbol{F}^0$ with this configuration. Set $\xi'_i = \xi_{i\hat{j}}, i = 1, 2, ..., N$. Let $\mathcal{M}$ denote the set of locations.

Step 1 Determine $L_j = \{i \,|\, \xi_{ij} - \xi'_i < 0\}$.
Set $I_j = L_j$ if $|L_j| \le K_j$, else $I_j = \underline{L}_j$ if $|L_j| > K_j$
where $\underline{L}_j \subset L_j, |\underline{L}_j| = K_j, \forall \ell \in \underline{L}_j, \forall m \in L_j \backslash \underline{L}_j, \; \xi_{\ell j} - \xi'_\ell \le \xi_{mj} - \xi'_m$.
For ($j \in \mathcal{M} \backslash \mathcal{S}_k$) do

$$\boldsymbol{F}_j^{k+1} = \boldsymbol{F}^k + \sum_{i \in I_j} (\xi_{ij} - \xi'_i) + \eta_j \qquad \text{where } I_j = \{i \,|\, \xi_{ij} - \xi'_i < 0\}.$$

Step 2 Determine a new $\hat{j}$ such that

$$\boldsymbol{F}_{\hat{j}}^{k+1} = \min_{j \in \mathcal{M} \backslash \mathcal{S}_k} \{\boldsymbol{F}_j^{k+1}\} < \boldsymbol{F}^k.$$

If there is no such $\hat{j}$, go to Step 4.

Step 3 Update

$$\mathcal{S}_{k+1} = \mathcal{S}_k \cup \{\hat{j}\} \quad \text{and} \quad \xi'_i = \xi_{i\hat{j}} \quad \text{for } i \in I_{\hat{j}}.$$

Set $\boldsymbol{F}^{k+1} = \boldsymbol{F}_{\hat{j}}^{k+1}$ and $k := k + 1$ and go to Step 1.

Stop 4 No more improvement possible; stop.

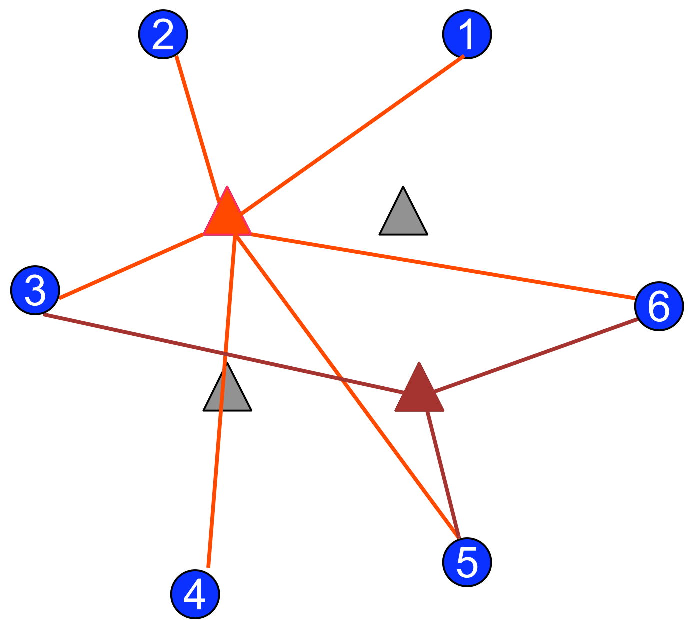
- Complexity: $O(NM^2)$,
  N--number of areas, M--number of locations

Add Heuristic: example

- 6 areas, 4 locations
- $K_j = 3, \forall j \quad \eta_1 = 0, \eta_j = 2, j \geq 2$
- connectivity cost

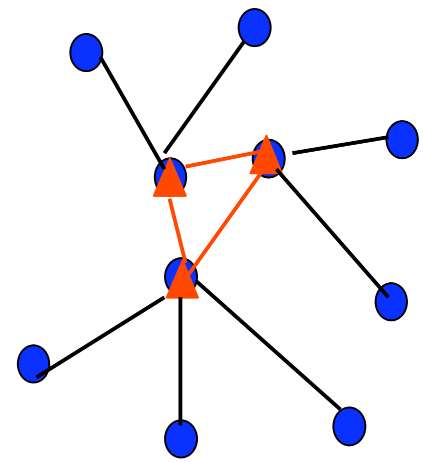
TABLE 6.1 **Cost ξ_{ij} Information for Example 6.1**

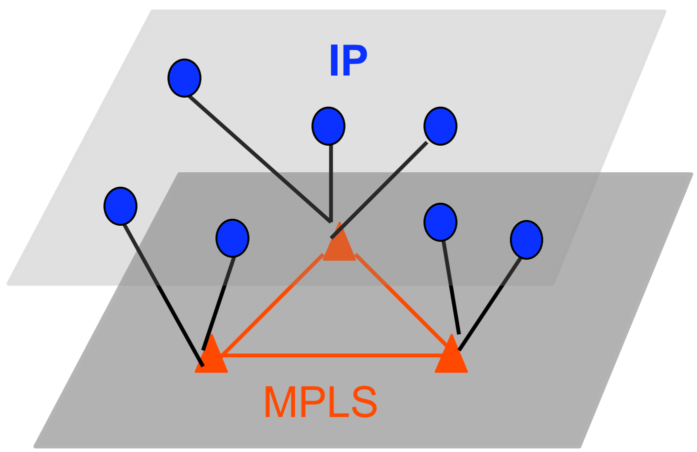
	S1	S2	S3	S4
1	2	1	2	4
2	1	0	1	2
3	4	1	2	2
4	1	2	1	2
5	2	3	2	0
6	4	4	3	2



Joint Node Location and Link Connectivity

- connectivity cost
  - from access nodes to core nodes
  - between core nodes
  - fully connected core
- two models
  - one-level design
    all nodes same type, each site can be possibly a core node site (same node for access and core): IP routers with different capacities
  - two-level design
    access node locations and core node locations are different: IP/MPLS





One-level Design

- N sites, select P core sites, remaining N-P access sites connect to one of P core sites
- core site j handles k_j access sites
- fully connected backbone among core sites
- all sites connected through the core backbone
- symmetric (undirected) connections

One-level Formulation

Node Location and Link-Connectivity

- **indices**

  $i, j = 1, 2, ..., N$ sites to be connected

- **constants**

  P number of switch locations ($P < N$)

  ξ_{ij} cost of connecting site i to site j (assume symmetry, i.e., $\xi_{ij} = \xi_{ji}$)

  η_j cost of location j, if opened

  K_j maximum number of sites that can be handled at location j (if j is selected)

- **variables**

  u_{ij} 1, if site i is connected to site j; 0, otherwise

  r_j 1 if site j is chosen for location of a switch; 0, otherwise

- **objective**

  minimize $\boldsymbol{F} = \sum_{i=1}^{N} \sum_{j=i}^{N} \xi_{ij} u_{ij} + \sum_{j=1}^{N} \eta_j r_j$

- **constraints**

$$\sum_{i=1}^{N} u_{ij} - (P-1) r_j \geq 1, \quad j = 1, 2, ..., N$$

$$\sum_{i=1}^{N} u_{ij} - (K_j + P - 2) r_j \leq 1, \quad j = 1, 2, ..., N$$

$$\sum_{i=1}^{N} \sum_{j=i}^{N} u_{ij} = N + P(P-1)/2$$

$$\sum_{j=1}^{N} r_j = P$$

$$\sum_{i=1}^{N} u_{ii} = P.$$

Two-level Design

- access sites and core sites are different
  - different types of nodes on access/core sites
- N access sites
- P core sites out of M possible locations
- connection cost
  - between an access site and core site
  - between two core sites

Two-level Formulation

Node Location and Link-Connectivity: Two-Level nodes/sites

- **indices**

  $i = 1, 2, ..., N,$ access sites to be connected

  $j, k = 1, 2, ..., M,$ possible location of core nodes

- **constants**

  P number of node locations to be chosen ($P < M$)

  ξ_{ij} cost of connecting access site i to core site j

  η_j cost of core location j, if opened

  ζ_{jk} cost of connecting possible core site j to possible core site k

  K_j maximum number of sites that can be handled at core location j (if j is selected)

- **variables**

  u_{ij} = 1, if access site i is connected to core node site j; 0, otherwise

  r_j = 1 if core node site j is chosen for location of a node device; 0, otherwise

  s_{jk} = 1, if core site site j is connected to core site k; 0, otherwise

- **objective**

  minimize $\boldsymbol{F} = \sum_{i=1}^{N}\sum_{j=1}^{M}\xi_{ij}u_{ij} + \sum_{j=1}^{M}\eta_j r_j + \sum_{j=1}^{M-1}\sum_{k=j+1}^{M}\zeta_{jk}s_{jk}$

- **constraints**

  $\sum_{j=1}^{M} u_{ij} = 1, \quad i = 1, 2, ..., N$

  $\sum_{i=1}^{N} u_{ij} \leq K_j r_j, \quad j = 1, 2, ..., M$

  $\sum_{k=1, k\neq j}^{M} s_{jk} = (P-1) r_j, \quad j = 1, 2, ..., M$

  $\sum_{j=1}^{M-1}\sum_{k=j+1}^{M} s_{jk} = P(P-1)/2$

  $\sum_{k=1}^{M} r_j = P.$

Non-Fully Connected Core

- Non-fully connected in practice
- Avoid disconnecting network
- Delete core link heuristic

ALGORITHM 6.2 **Delete Heuristic to Generate a Less-Than Fully-Connected Topology**

Step 1: Solve D/ONLLC/B (6.2.20)

Step 2: If the minimum number of core network links, L, is specified to be $P(P-1)/2$, then stop; otherwise go to Step 3.

Step 3: Sort all core network links in descending order of their cost values, i.e., appropriate ζ_{jk} corresponding to optimal solution for (6.2.20)

Step 4: While (the number of remaining core-network links $\geq L$) do

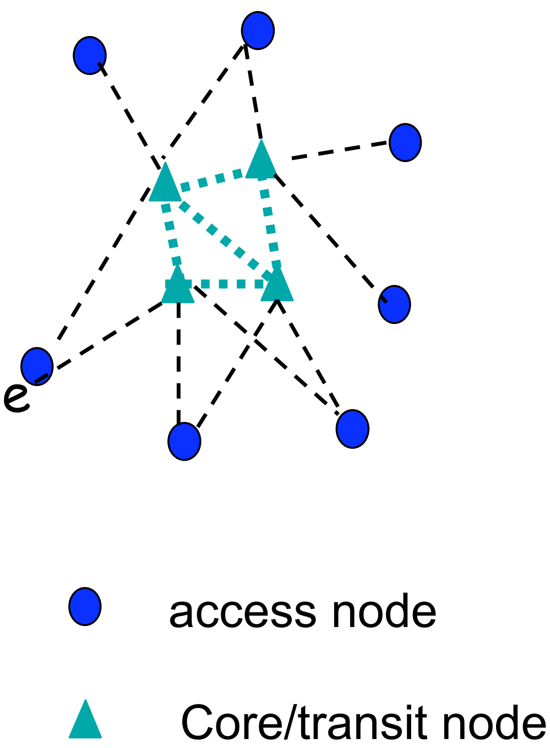
(a) Select the core network link with the highest cost from the current list of links not yet tested

(b) Delete it if the overall core network connectivity is not violated; otherwise, mark it and ignore if the connectivity is violated

endwhile

Comprehensive Topological Design

- traffic demand has big impact on node and link location
- jointly design location/dimension/flow
  - given access/core locations, determine link locations, flow and capacity allocation
  - given access locations only, determine core locations, link locations, flow and capacity allocation
- access nodes represent access networks
- traffic demand between access nodes, to be realized; pure access nodes cannot transit traffic;
- pure core nodes only transit traffic, don't generate traffic
- cost
  - link/node opening
  - Capacity cost



Design with Budget Constraint: optimal network problem

- access/core locations given
- link cost
  - capital cost (installation) must under budget
  - maintenance/operational (capacity dependent) cost to be minimized
- NP-Complete
- enforce path diversity?

- **constants**

B	upper bound for capital budget
h_d	volume of demand d
δ_{edp}	$= 1$ if link e belongs to path j realizing demand d; 0, otherwise
ξ_e	unit maintenance cost on link e
κ_e	fixed cost of installing link e (for capital budget)
C_e	upper bound for the capacity of link e

- **variables**

x_{dp}	non-negative flow realizing demand d allocated to path j (non-negative continuous variable)
y_e	capacity of link e (non-negative integer variable)
u_e	$= 1$ if link e is provided; 0, otherwise (binary variable)

- **objective**

minimize $\boldsymbol{F} = \sum_e \xi_e y_e$

- **constraints**

$$\sum_e \kappa_e u_e \leq B$$
$$\sum_p x_{dp} = h_d, \quad d = 1, 2, ..., D$$
$$\sum_d \sum_p \delta_{edp} x_{dp} \leq y_e, \quad e = 1, 2, ..., E$$
$$y_e \leq C_e u_e, \quad e = 1, 2, ..., E.$$

Optimal Network Problem: extended objective

- what if budget too tight?
- network cost = capital cost + maintenance cost

- **objective**

  minimize $\boldsymbol{F} = \sum_e \xi_e y_e + \sum_e \kappa_e u_e$

- **constraints**

$$\sum_p x_{dp} = h_d, \quad d = 1, 2, ..., D$$

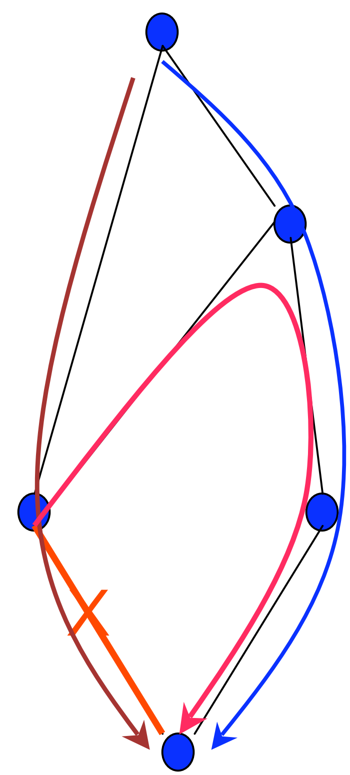
$$\sum_d \sum_p \delta_{edp} x_{dp} \leq y_e, \quad e = 1, 2, ..., E$$

$$y_e \leq C_e u_e, \quad e = 1, 2, ..., E.$$

- NP Complete

Optimal Network Problem: Heuristic Algorithm

- start with all links installed
- iteratively remove links, one each iteration
  - saving from link removal: capital + operational
  - cost from link removal:
    - traffic on the removed link have to be rerouted:
      local rerouting v.s. global rerouting
    - capacity cost incurred on new paths
  - Net gain of remove link e
    - gain(e)=saving(e)-cost(e)
  - Remove the link maximizing gain(e)



Core Nodes and Links Location

- both core nodes and links locations to be determined
- new constraints
  - if a node v not installed, all links attached to v won't be installed
  - bound on core node degree
- Link-path formulation or node-link formulation

Link-Path Formulation

- how to pre-set paths without knowing the locations of core nodes and links?
  - bad set of candidate paths might lead to expensive or infeasible solutions

- **indices**
  - $d = 1, 2, ..., D$ demands
  - $p = 1, 2, ..., P_d$ candidate paths for demand d
  - $e = 1, 2, ..., E$ links
  - $v = 1, 2, ..., V$ transit nodes
- **constants**
  - h_d volume of demand d
  - δ_{edp} $= 1$ if link e belongs to path j realizing demand d, 0 otherwise
  - β_{ev} $= 1$ if link e is incident with node v, 0 otherwise
  - ξ_e marginal cost of link e
  - κ_e fixed cost of installing link e
  - C_e upper bound for the capacity of link e
  - φ_v cost of installing transit node v
  - G_v upper bound for the degree of transit node v
- **variables**
  - x_{dp} flow realizing demand d allocated to path j (non-negative continuous)
  - y_e capacity of link e (non-negative continuous)
  - u_e $= 1$ if link e is provided, 0 otherwise
  - s_v $= 1$ if node v is installed, 0 otherwise

Link-Path Formulation (cont.d)

- **objective**

  minimize $\boldsymbol{F} = \sum_e (\xi_e y_e + \kappa_e u_e) + \sum_v \varphi_v s_v$

- **constraints**

$$\sum_p x_{dp} = h_d, \quad d = 1, 2, ..., D$$

$$\sum_d \sum_p \delta_{edp} x_{dp} \leq y_e, \quad e = 1, 2, ..., E$$
$$y_e \leq C_e u_e, \quad e = 1, 2, ..., E$$
$$\sum_e \beta_{ev} u_e \leq G_v s_v, \quad e = 1, 2, ..., E, \; v = 1, 2, ..., V.$$

Node-Link Formulation

- distinguish access and transit links
  - directed access links: between access and transit nodes
  - directed transit links: between transit nodes
- use source-based link-flow variable

REF: Node-Link Formulation II

- **constants**

  a_{ev} = 1 if link e originates at node v, 0 otherwise

  b_{ev} = 1 if link e terminates in node v, 0 otherwise

  $h_{vv'}$ volume of demand d originating at node v and terminating at node v'

  H_v $= \sum_{v' \neq v} h_{vv'}$ - total demand volume originating in node v

  ξ_e unit cost of link e

- **variables**

  x_{ev} flow realizing *all* demands originating at node v on link e

  y_e capacity of link e

- **objective**

  minimize $\boldsymbol{F} = \sum_e \xi_e y_e$

- **constraints**

$$\sum_e a_{ev} x_{ev} = H_v, \quad v = 1, 2, ..., V$$

$$\sum_e b_{ev'} x_{ev} - \sum_e a_{ev'} x_{ev} = h_{vv'}, \quad v, v' = 1, 2, ..., V, \; v \neq v'$$

$$\sum_v x_{ev} \leq y_e, \quad e = 1, 2, ..., E.$$

Node-Link Formulation

- **indices**

$w = 1, 2, ..., W$	access nodes
$v = 1, 2, ..., V$	transit nodes
$e = 1, 2, ..., E$	links
$f = 1, 2, ..., F$	directed access arcs (between access and transit nodes)
$t = 1, 2, ..., T$	directed transit arcs (between transit nodes)

- **constants**

$h_{ww'}$	volume of demand from access node w to access node w'
H_w	$= \sum_{w'} h_{ww'}$ - total demand outgoing from access node w
β_{ev}	$= 1$ if link e is incident with transit node v
β_{fv}	$= -1$ if access arc f is incoming to transit node v
	$= 1$ if access arc f is outgoing from transit node v
	$= 0$ otherwise
β_{fw}	$= -1$ if access arc f is incoming to access node w
	$= 1$ if access arc f is outgoing from transit node w
	$= 0$ otherwise
β_{tv}	$= -1$ if transit arc t is incoming to transit node v
	$= 1$ if transit arc t is outgoing from transit node v
	$= 0$ otherwise

Node-Link Formulation

- **constants(contd.)**

ω_{ef}	$= 1$ if access arc f is realized on link e, 0 otherwise
ω_{et}	$= 1$ if access arc t is realized on link e, 0 otherwise
ξ_e	marginal cost of link e
κ_e	fixed cost of installing link e
C_e	upper bound for the capacity of link e
β_{ev}	$= 1$ if node v is incident with link e, 0 otherwise
φ_v	cost of installing transit node v
G_v	upper bound for the degree of transit node v

- **variables**

x_{fw}	flow realizing all demands originating at access node w on access arc f
x_{tw}	flow realizing all demands originating at access node w on transit arc t
y_e	capacity of link e
u_e	$= 1$ if link e is provided; 0, otherwise
s_v	$= 1$ if transit node v is installed; 0, otherwise

- **objective**

minimize $\boldsymbol{F} = \sum_e(\xi_e y_e + \kappa_e u_e) + \sum_v \varphi_v s_v$

Node-Link Formulation

- **constraints**

$$\sum_t \omega_{et} \sum_w x_{tw} + \sum_f \omega_{ef} \sum_w x_{fw} \leq y_e, \quad e = 1, 2, ..., E$$

$$\sum_f \beta_{fw} x_{fw} = H_w, \quad w = 1, 2, ..., W$$

$$\sum_f \beta_{fw'} x_{fw} = h_{ww'}, \quad w = 1, 2, ..., W, \; w' = 1, 2, ..., W, w \neq w'$$

$$\sum_t \beta_{tv} x_{tw} + \sum_f \beta_{fv} x_{fw} = 0, \quad v = 1, 2, ..., V, \; w = 1, 2, ..., W$$

$$y_e \leq C_e u_e, \quad e = 1, 2, ..., E$$

$$\sum_e \beta_{ev} u_e \leq G_v s_v, \quad v = 1, 2, ..., V.$$