Topology Control and Clustering

Part 5

CS-6777 Mobile Ad Hoc Networking

Memorial University of Newfoundland

Between PHY and Higher

- Topology Control
  - Adjusting node transmission power or selecting specific neighbors to obtain network structures with certain properties
- Clustering
  - Creating connected substructures of the network for further topology abstraction

Modeling MANETs

- Your approach?

- PHY layer
  - Transmission range and reception
- Higher layers
  - Graph $<V,E>$

PHY Layer Model 1

Physical model

- Given a set of nodes *V*, receiver *r* can successfully receive transmission from sender *s*
  - *Px* – node *x*'s transmission power
  - *d* (*x*, *y*) – distance between *x* and *y*
  - *N* – while noise level
  - α – path loss exponent
  - β – reception threshold

$$\frac{\dfrac{Ps}{d^{\alpha}(s,r)}}{N + \sum_{u \in V, u \neq s} \dfrac{Pu}{d^{\alpha}(u,r)}} \geq \beta$$

PHY Layer Model 2

Protocol model

- Given a set of nodes *V*, receiver *r* can successfully receive transmission from sender *s*
  - *Px* – node *x*'s transmission power
  - *d(x, y)* – distance between *x* and *y*
  - *N* – while noise level
  - α – path loss exponent
  - Δ – models a "guard zone" within the receiver

$$\frac{Ps}{d^{\alpha}(s,r)} \geq (1+\Delta)\frac{Pu}{d^{\alpha}(u,r)} \quad \text{for all } u \in V \text{ and } u \neq s$$

A Higher Layer Model

- d-dimensional space, $d = 1,2,3$
- A MANET is represented by $M_d = (V, P)$, where
  - V is the set of nodes and $|V| = n$
  - $P : N \times T \rightarrow [0,l]^d$ is the placement function, which assigns to every element of V and to any time a set of coordinates in the d–dimensional space
- Node v is said stationary if its physical placement does not vary with time; if all nodes are stationary, then the network is said to be stationary. And thus the function P can be represented simply as $P : N \rightarrow [0,l]^d$

A Higher Layer Model (cont'd)

- A range assignment for network $M_d = (V, P)$ is a function $RA : N \rightarrow (0, r_{max}]$
- Given a network $M_d = (V, P)$ and a range assignment $RA : N \rightarrow (0, r_{max}]$, the transmission graph (AKA communication graph) induced by RA on M_d at time t is defined as the directed graph $G_t = (N, E(t))$, where arc (i, j) exists iff $RA(i) \geq d(P(i,t), P(j,t))$
  - Or simply denoted $G = (N, E)$ when RA is stationary

A Higher Layer Model (cont'd)

- A range assignment *RA* is said to be connecting at time *t* if the induced transmission graph at time *t* is strongly connected
  - Or simply connecting if RA is stationary
- A range assignment *RA* is said to be *r*-homogeneous if all the nodes have the same transmission range *r*
  - Or simply homogeneous if the value of *r* is not important

Topology Control

Goal of topology control is to determine the nodes' transmission ranges in order to obtain some desired properties, e.g.

- Connectivity
- Node degree (max, min, average)
- Link sparseness
- Communication cost (hop number, distance, energy consumption)

Probabilistic Tools

- Classic random graph models of Bollobás
  - $G(n,M)$ – Set of all graphs with n vertices having M edges, in which the graphs have the same probability
  - $G(n,P(\text{edge})=p)$ – Set of all graphs with n vertices in which the edges are chosen independently with probability p
  - Neither works for MANETs – Why?
- More acceptable model – Geometric Random Graphs (GRG)
  - Vertices are placed deterministically or according to some distribution
  - Edges are induced by range assignment

Critical Transmitting Range

- Definition CTR:
  - Suppose n nodes are placed in $R=[0,l]^d$, with $d=1,2,3$; what is the minimum r s.t. the r-homogeneous range assignment is connecting?
- When nodes' placement is known
  - Easy – longest edge of the MST
- When nodes' placement unknown

$$(0, l\sqrt{d}\,]$$

  - Depending on the node distribution assumption

Critical Transmitting Range

- When nodes are uniformly distributed in $[0,1]^2$, the critical transmitting range for connectivity is

$$c\sqrt{\frac{\log n}{n}}$$

w.h.p. for some c>0 [Panchapakensan2001]

- Note: when nodes are distributed uniformly at random and $d > 2$, the longest nearest neighbor link and the longest MST edge have the same length (asymptotically as $n \to \infty$) [Penrose1999], previously known as the giant component phenomenon
  - How do we generalize this to *k*-connectivity?

Non-Homogeneous Topology Control

- Definition – Range Assignment Problem (RA)
  - Let V be the set of nodes in the d-dimensional space (d=1,2,3). Determine the a connecting range assignment RA s.t.

    $$c(RA) = \sum_{v \in V} (RA(v))^{\alpha}$$

    is minimized
- Complexity
  - Solvable in $O(n^4)$ when d=1
  - NP-hard when d=2,3 but can be approximated with a performance ration of 2 [Clementi99, Kirousis00]

Variants of Range Assignment

- Restricted RA – induced graph's diameter bounded by *h*
- Symmetric
  - SRA (Symmetric RA) – Induced graph cannot have unidirectional arcs
  - WSRA (Weakened SRA) – Ignore unidirectional arcs in induced graph when determining connectivity

Energy-Efficient Unicast and Broadcast

- Energy-Efficient Unicast
  - Subgraph with good energy efficiency for transferring messages between all pairs of nodes
- Energy-Efficient Broadcast
  - Subgraph with good energy efficiency for sending messages to all nodes from a given source node

Power Stretch Factor

- Let $G=(V,E)$ be the transmission graph obtained by assigning maximum range to each node and assume it is strongly connected
- Every arc (u,v) of G is weighted the power $d^{\alpha}(u,v)$ needed to transmit a message from u to v
- Given a path in G from node s to t, denoted P, its power cost $pc(P)$ is the sum of the weights of the arcs of P

Power Stretch Factor (Cont'd)

- The path in G connecting s to t consuming the minimum energy is called the minimum-power path, and its power cost is denoted $pc_G(s,t)$
- Let $G'=(V,E')$ be any spanning subgraph of G. The power stretch factor of G' w.r.t G is the maximum ratio of the power cost of minimum-power path in G' to that in G over all node pairs, i.e.:

$$\rho_{G'} = \max_{u,v \in V} \frac{pc_{G'}(u,v)}{pc_G(u,v)}$$

Distance / Hop Stretch Factors

The previous power stretch factor is a generalization of two existing notions:

- Distance stretch factor
  - Ratio of path Euclidean lengths
  - $\alpha=1$
- Hop stretch factor
  - Ratio of path hop lengths
  - $\alpha=0$

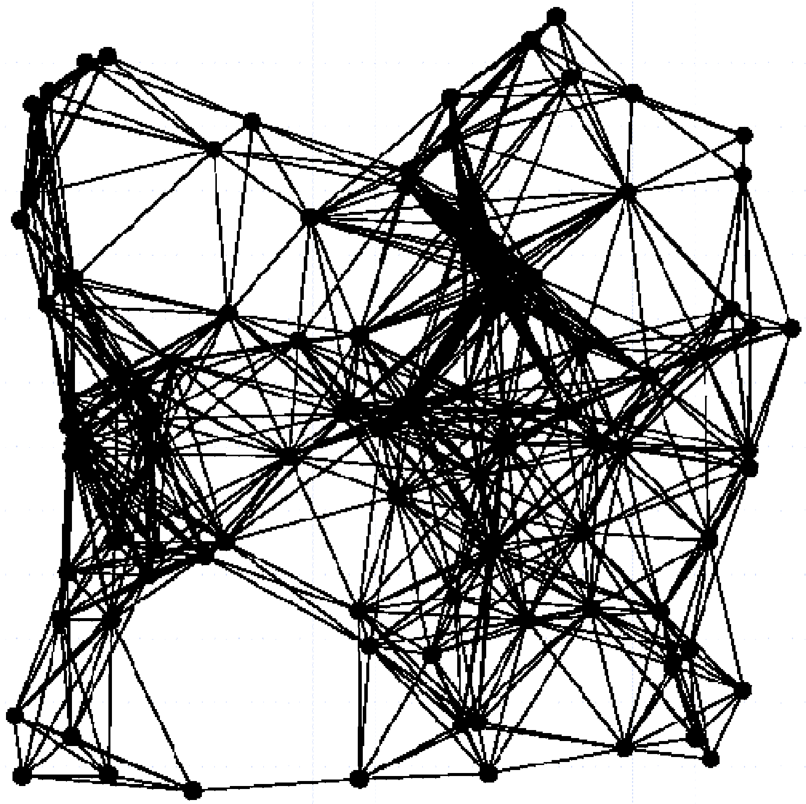
When the power (distance, hop, resp.) stretch factor of *G'* is a constant, then *G'* is called a power (distance, hop, resp.) spanner of *G*

Finding Good Routing Graphs

- A subgraph, called routing graph, should have the following intuitive properties:
  - Constant power stretch factor (power spanner)
  - Linear number of arcs (sparse)
  - Bounded degree (sparse, too)
  - Easy to calculate in a localized distributed way (implementation)
- Most works in the literature focus on finding distance spanners satisfying some of the above
  - We can show that being a distance spanner is stronger than being a power spanner. How?

Unit Disk Graph – UDG

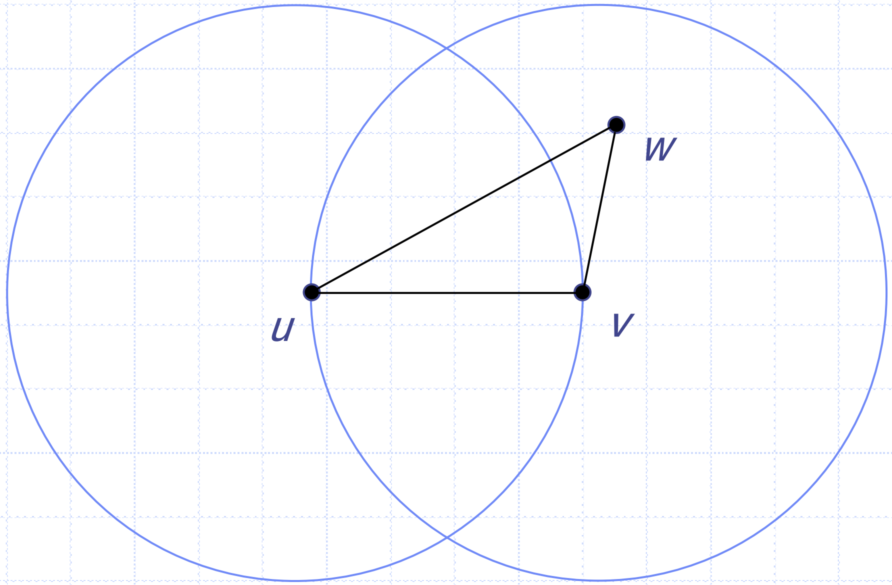
- Let *V* be a set of points in the 2-D Euclidean space
  - The Unit Disk Graph on *V*, denoted *UDG*(*V*), has an edge between nodes *u* and *v* iff $d(u,v) \leq 1$



Relative Neighborhood Graph – RNG

- Let *V* be a set of points in the 2-D Euclidean space
  - The Relative Neighborhood Graph on *V*, denoted *RNG*(*V*), has an edge between nodes *u* and *v* iff there is no node *w* s.t.

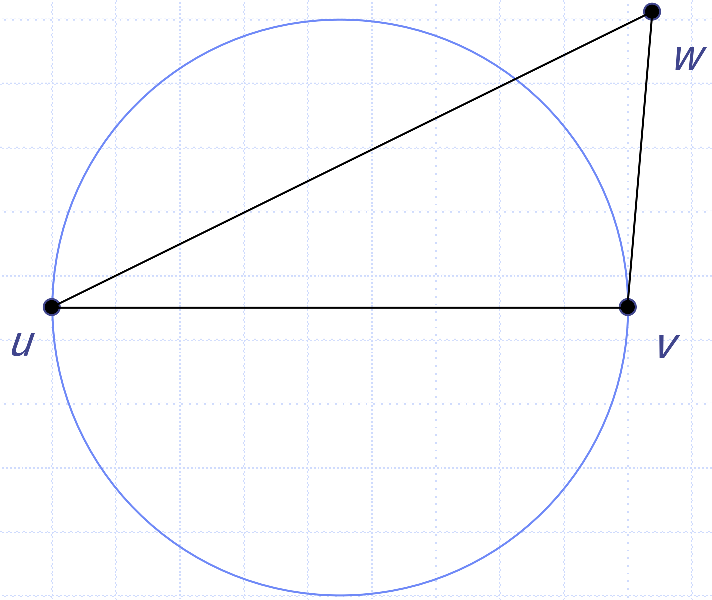
$$\max\{d(u,w), d(v,w)\} \leq d(u,v)$$



Gabriel Graph – GG

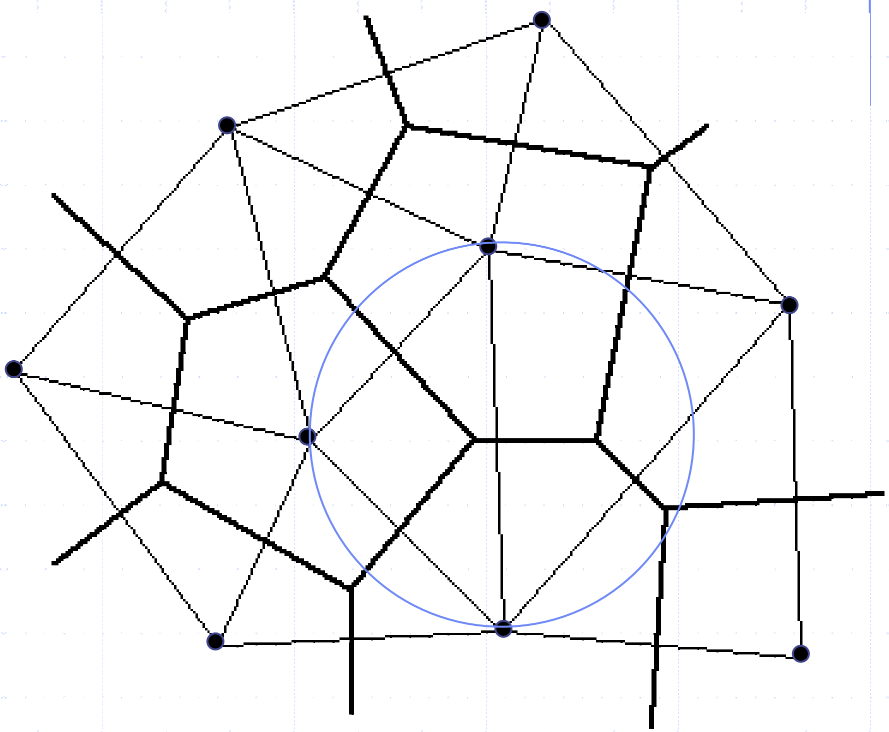
- Let *V* be a set of points in the 2-D Euclidean space
  - The Gabriel Graph on *V*, denoted *GG*(*V*), has an edge between nodes *u* and *v* iff there is no node *w* s.t.

$$d^2(u,w) + d^2(v,w) \leq d^2(u,v)$$



Delaunay Graph – DG

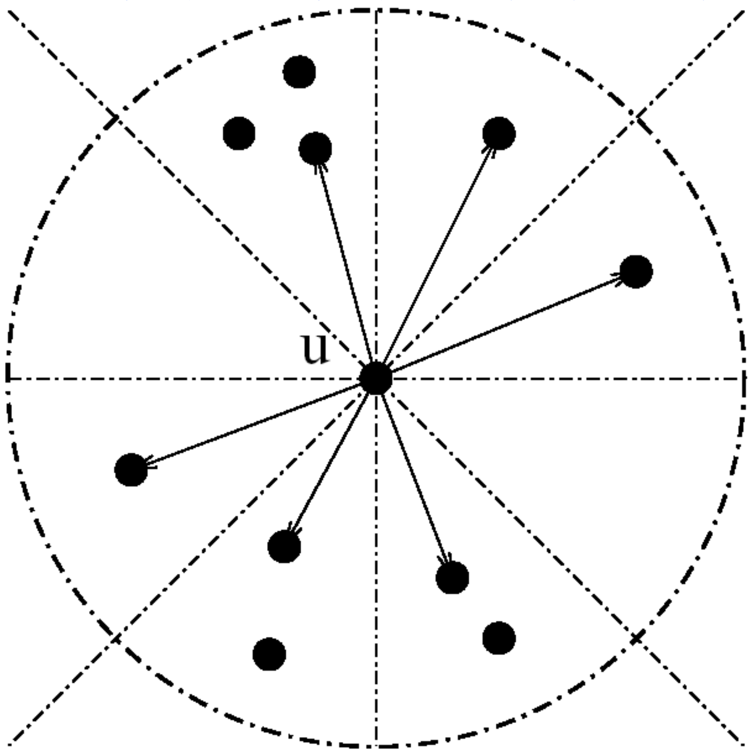
- Let *V* be a set of points in the 2-D Euclidean space
  - The Delaunay Graph on *V*, denoted *DG*(*V*), is the unique triangulation s.t. the circum-circle of every triangle contains no other node
  - Restricted DG (RGD) – all arcs within a maximum length



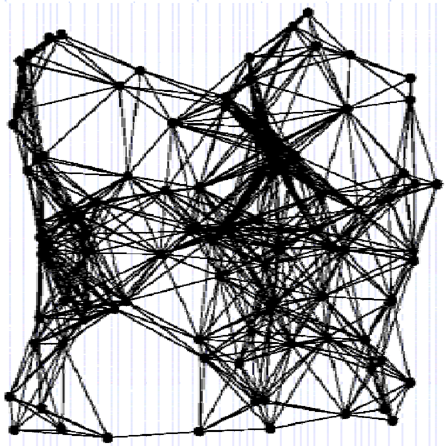
Yao Graph – YG

- Let V be a set of points in the 2-D Euclidean space and given integer $c \geq 6$
  - Yao Graph on V, denoted

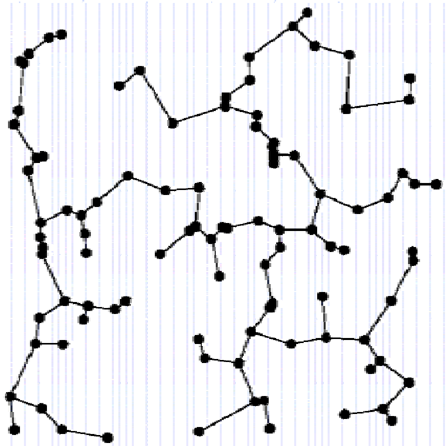
    $YG_c(G)$ – At each node $u \in V$, any c equally separated rays define c equal cone. In each cone, choose the closest node v, if any, and add arc (u,v) to E.
  - Reverse YG
  - Undirected YG



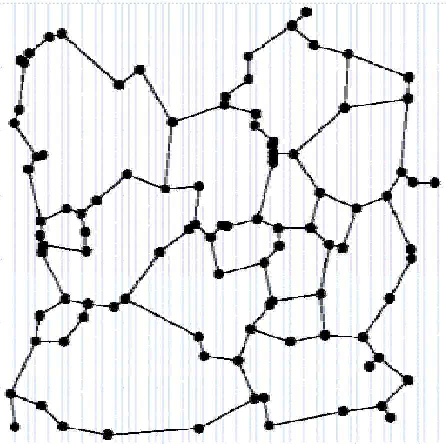
Examples



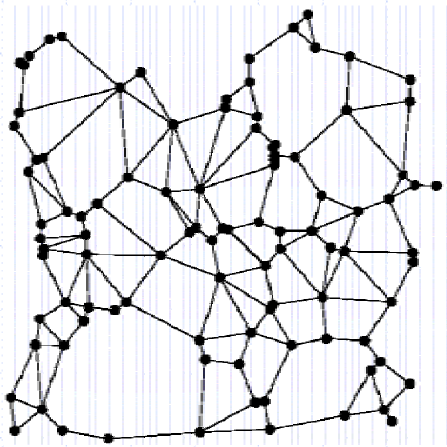
UDG



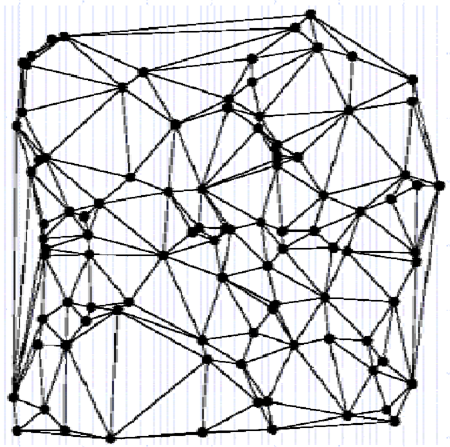
MST



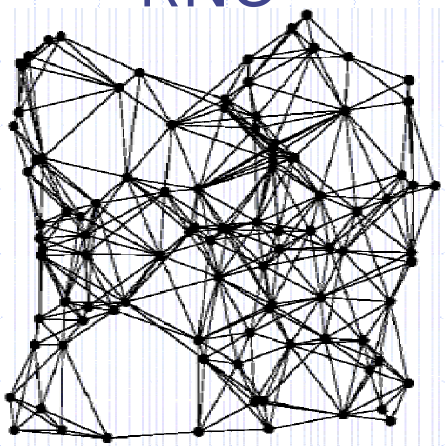
RNG



GG



DG



YG

Notes

- $RNG(V) \subseteq GG(V)$
- $RNG(V) \subseteq YG_c(V)$ for any $c \geq 6$
- $MST(V) \subseteq RNG(V)$ and $MST(V) \subseteq DG(V)$

	Distance	Power	Degree
RNG	$n-1$	$n-1$	$n-1$
GG	$\frac{4\pi\sqrt{2n-4}}{3}$	1	$n-1$
RDG	$\frac{1+\sqrt{5}}{2}\pi$	$\left(\frac{1+\sqrt{5}}{2}\pi\right)^{\alpha}$	$\Theta(n)$
YG_c	$\frac{1}{1-2\sin\frac{\pi}{c}}$	$\frac{1}{1-\left(2\sin\frac{\pi}{c}\right)^{\alpha}}$	$n-1$

Note: all these graphs have linear number of edges

Broadcast Tree

- Let $G=(V,E)$ be the transmission graph obtained by assigning maximum range to each node and assume it is strongly connected
- Any broadcast originating from node r can be represented as a directed spanning tree T rooted at r, where arc $(u,v) \in T$ means node v receives the message from u for the first time
- Power cost of node u in T is defined as

$$pc_T(u) = \max_{(u,v)\in E(T)} d^{\alpha}(u,v)$$

  - Power cost of a leaf node is 0
- Power cost of the broadcast T is defined as

$$pc_T(T) = \sum_{u\in V} pc_T(u)$$

Broadcast Stretch Factor

- Let u be the broadcast source
- The broadcast tree in G rooted at u that consumes the least energy is called minimum power broadcast tree
- Let G' be an arbitrary subgraph of G. The broadcast stretch factor of G' w.r.t. G is the maximum ratio between the minimum power broadcast tree in G' to that in G over all nodes. Formally

$$\beta_{G'} = \max_{s \in V} \frac{P_{G'}(s)}{P_G(s)}$$

where $P_H(s)$ denotes the power cost of the minimum power broadcast tree in graph H rooted at node s

Broadcast Tree – Complexity

- Let *u* be the broadcast source
- Finding the minimum power broadcast tree rooted at *u* is NP-C [Cagali et al. 2002; Liang 2002]
- *MST* has constant broadcast stretch factor
  - How local can an MST algorithm be? [Wan 2002]

In the Real World

- A topology control protocol should
  - Be fully distributed and asynchronous
  - Rely on local information only
  - Generate a connected topology (at least w.h.p.) composed of bi-directional links
  - Use little information
- Many protocols out there; do they have such properties?

Location Discovery

- Where am I? – absolute
- Where are my neighbors? – relative

- Answers include
  - GPS (alt. Galileo, GLONASS, Beidou)
  - Measurement technologies
    - Signal strength
    - Time of flight (ToF)
    - Angle of arrival (AoA)

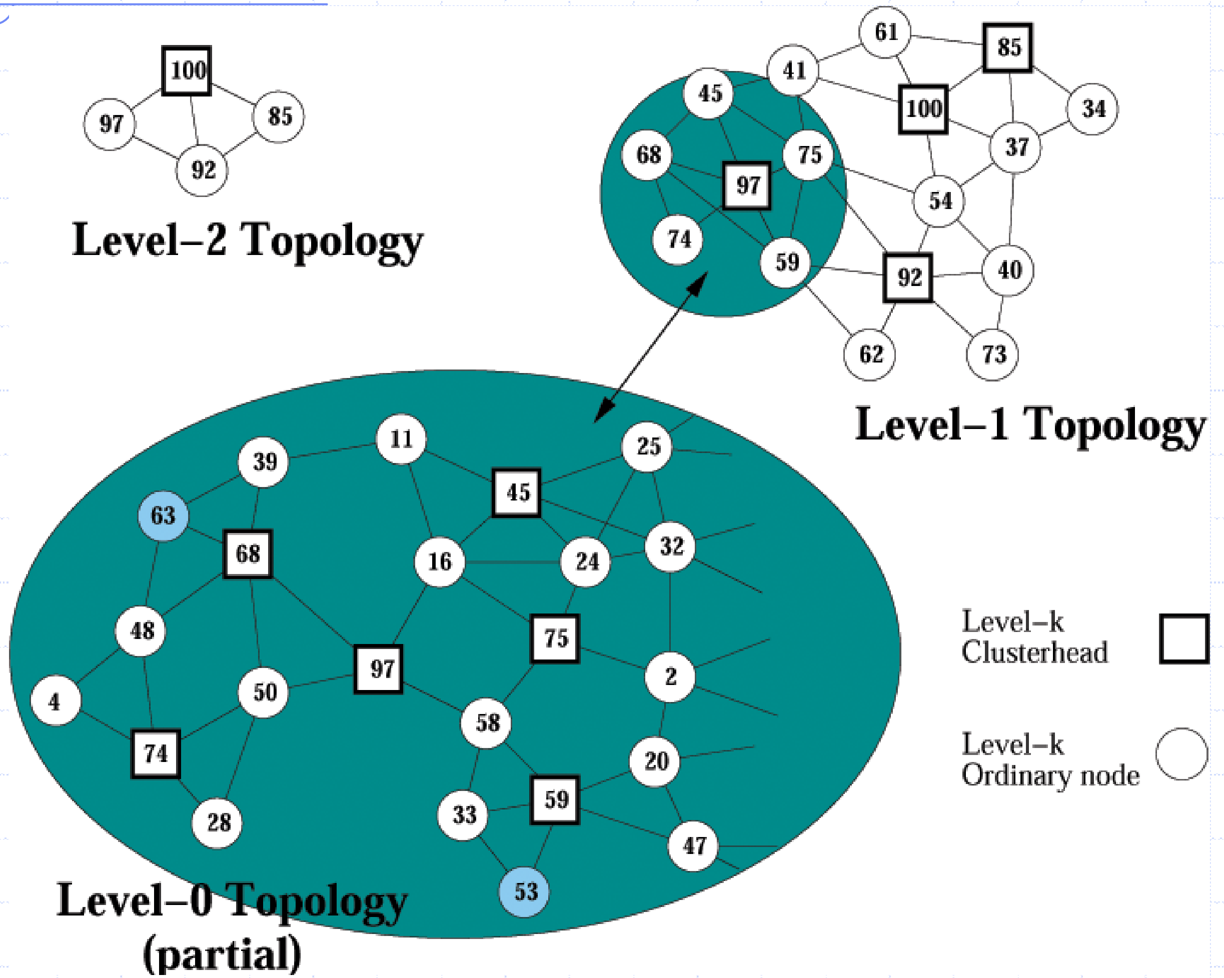
Clustering

- Process of define substructures and abstract topology of the original network
- A cluster is a subset of nodes that are close from one to another
- A cluster represents a “super node” in the abstract topology
- Links among clusters represent “super edges”
- Can be done recursively

Why Clustering

- Retrospect
  - PRNet of DARPA started in 1972
  - ARPANet deployed in 1969
- While Internet now connects billions of computers, why aren't there any MANET of even similar size?
- Scalability is the key
  - Routing efficiency

Hierarchical Routing



Hierarchical address

From: 53(97,59,53)
To: 63(97,68,63)

Clustering, Formally

- Give a graph $G = (V, E)$
- Divide the nodes into a collection of subsets $\{V_1, V_2, V_3, \ldots, V_k\}$, where $V = \bigcup_{i=1}^{k} V_i$ such that V_i induces a connected subgraph of $G = (V, E)$
  - Note that these subgraphs can overlap
- An abstract graph $G' = (V', E')$ is constructed, where each vertex $v'_i \in V'$ corresponds to a subset; there is an edge (v'_i, v'_j) iff there is an edge from some vertex $u_i \in V_i$ to some vertex $u_j \in V_j$

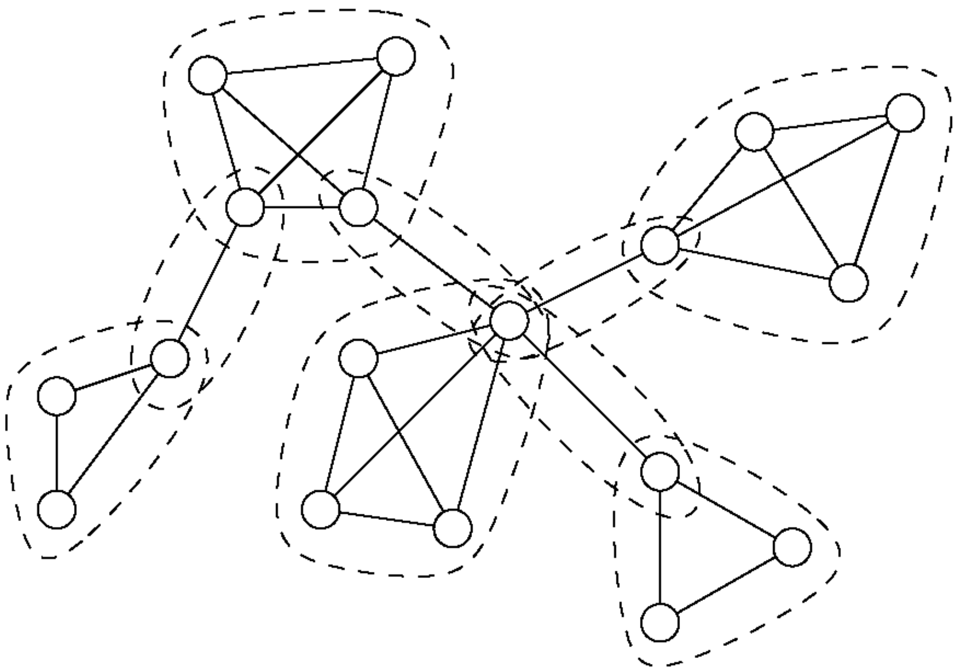
Clustering Methods

- Graph domination based
  - Dominating set
  - Independent dominating set
  - Connected dominating set
  - Weakly-connected dominating set
- Other
  - Clique
  - Tree

Clique Based Clustering

[Krishna et al. 97]

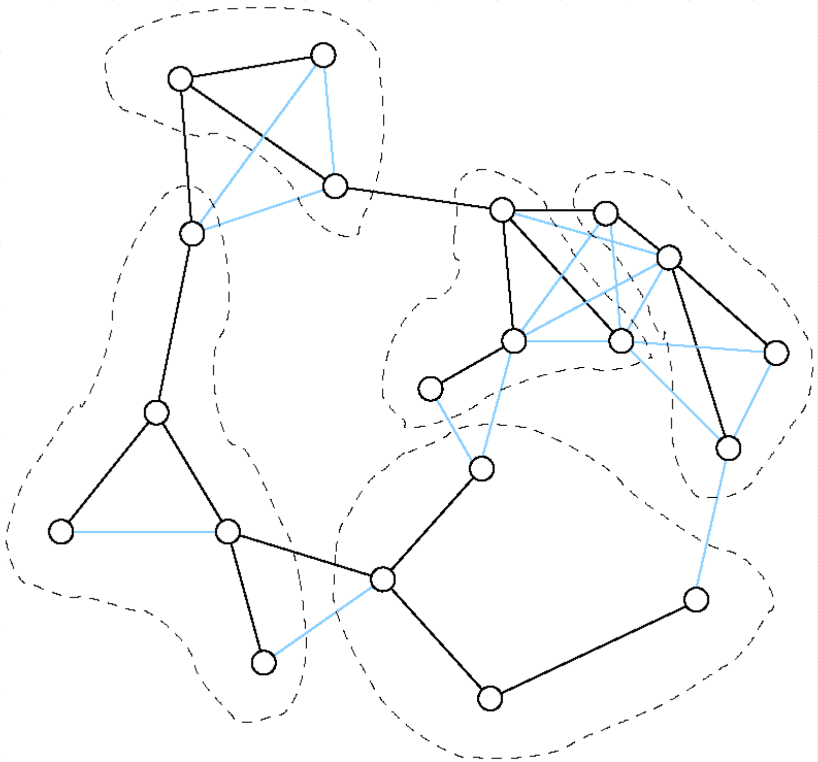
- Clique
  - Complete subgraph



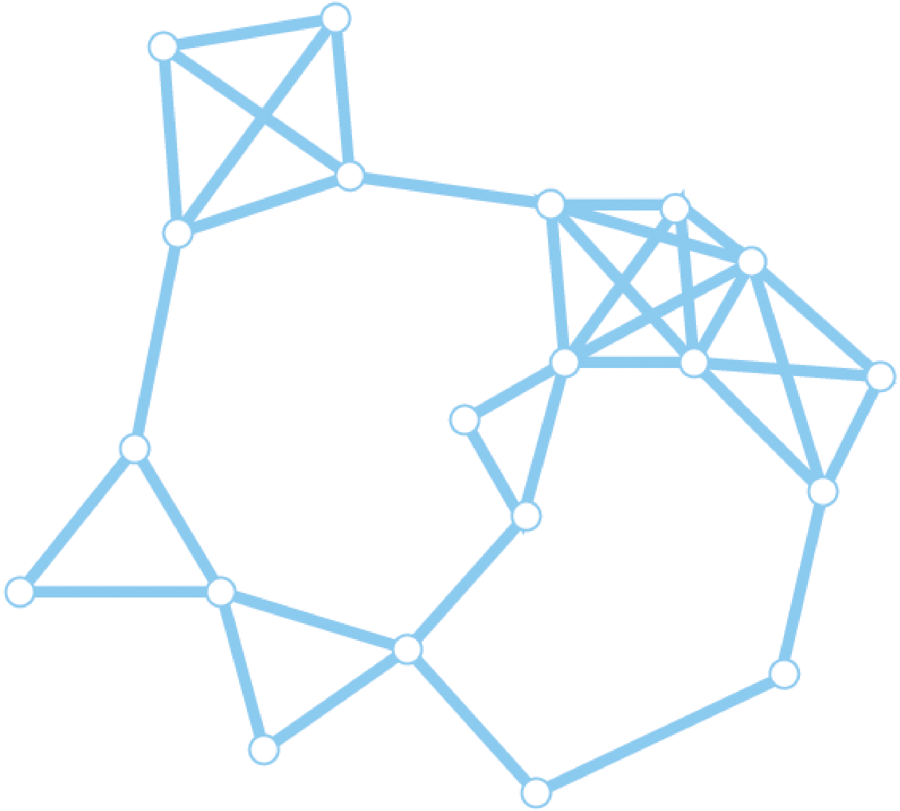
Tree Based Clustering

[Banerjee & Khuller 2000]

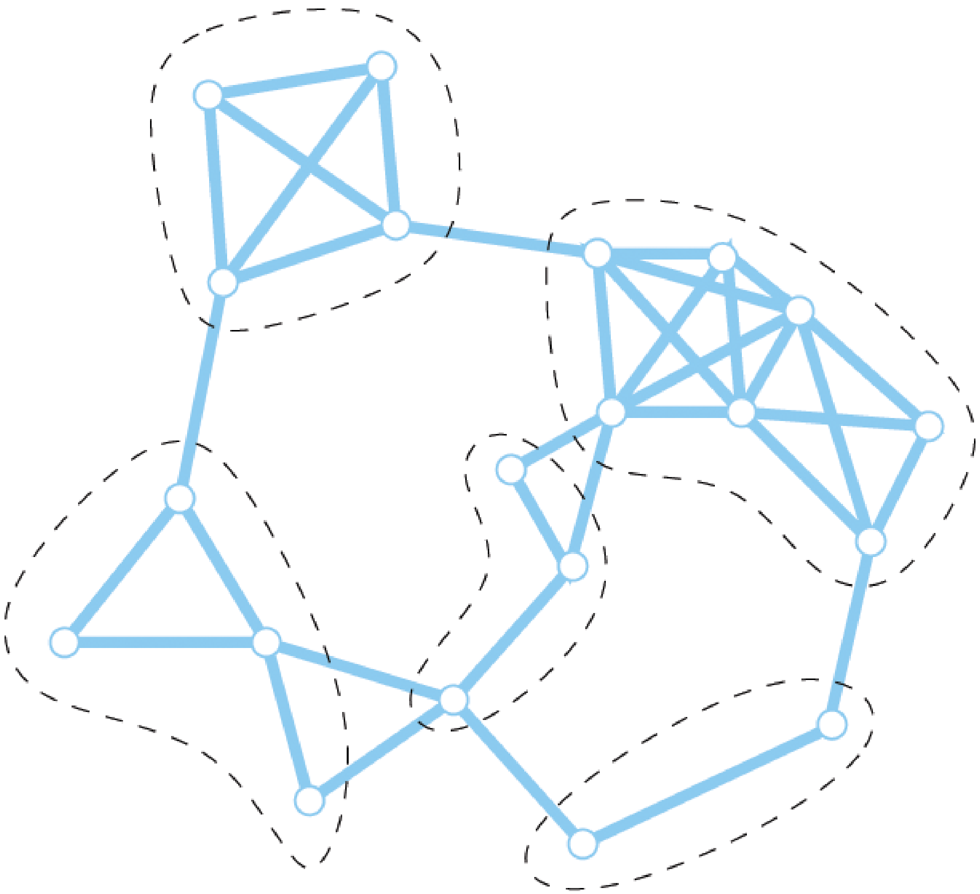
- Spanning tree based
  - Clusters are groups of branches of the tree



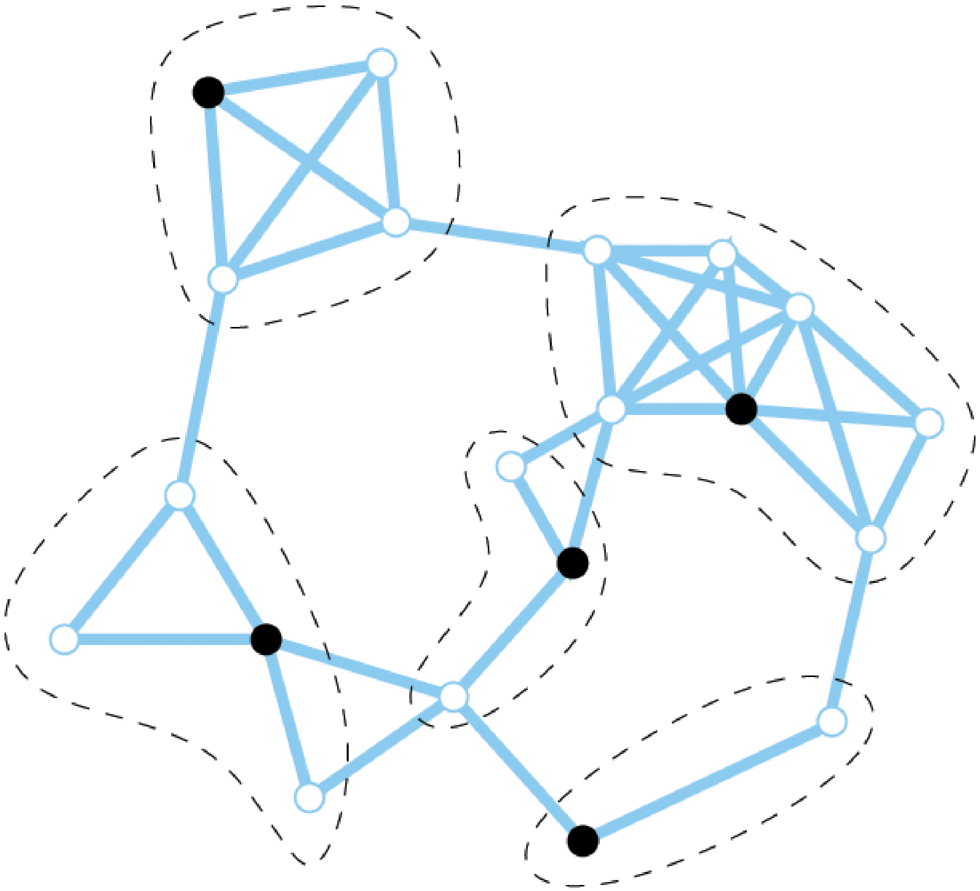
Why Graph Domination



Why Graph Domination



Why Graph Domination

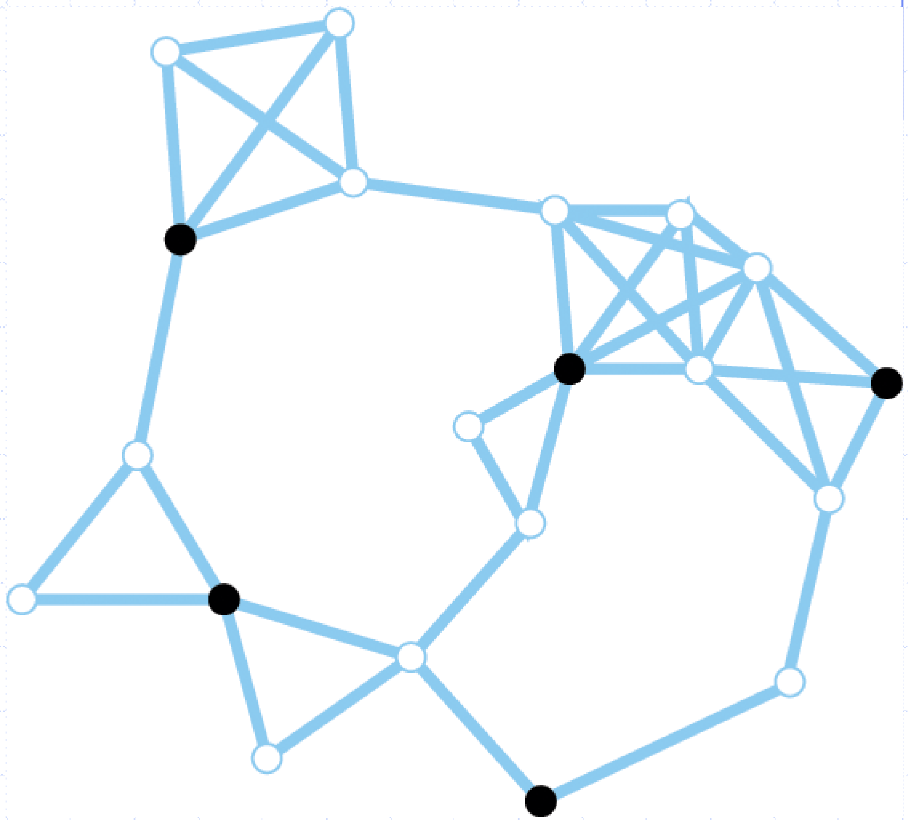


Dominating Set

- Given a graph $G = (V, E)$
- A subset $S \subseteq V$ is called a dominating set if for any vertex $v \notin S$ it has a neighbor in S

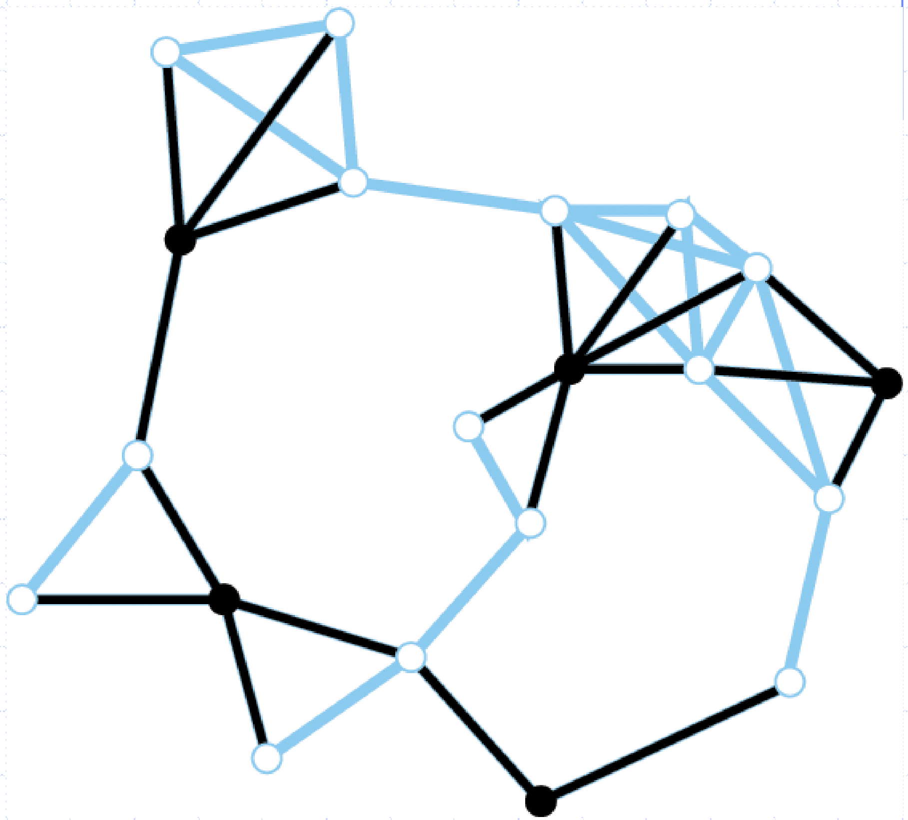
Graph Domination Variants

- Dominating set



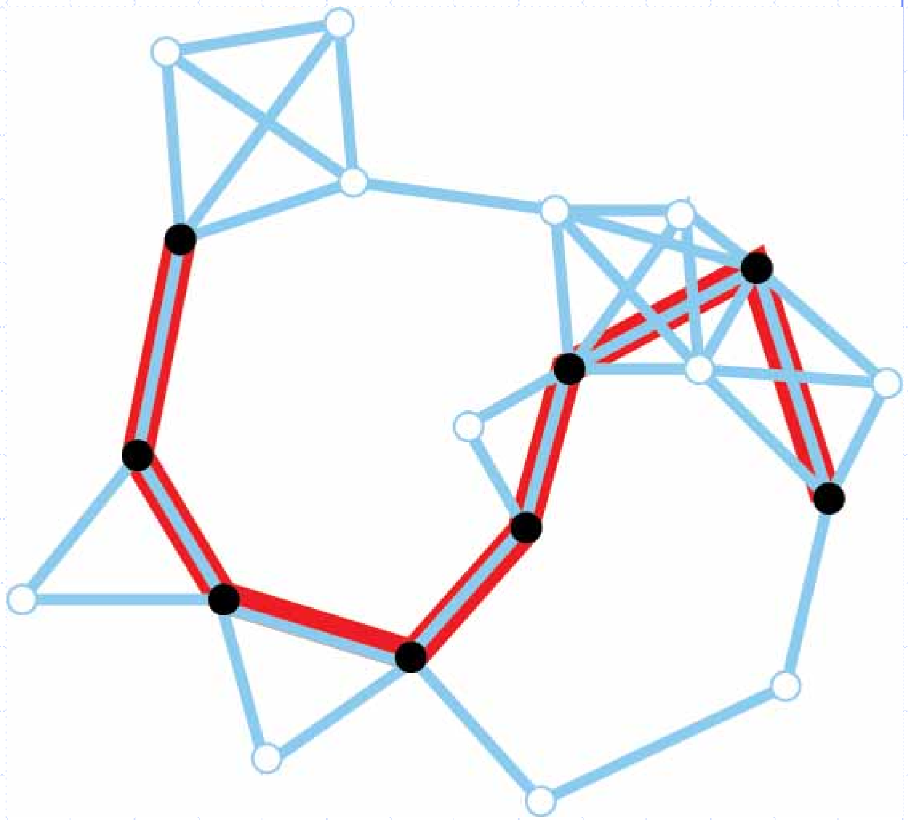
Graph Domination Variants

- Dominating set



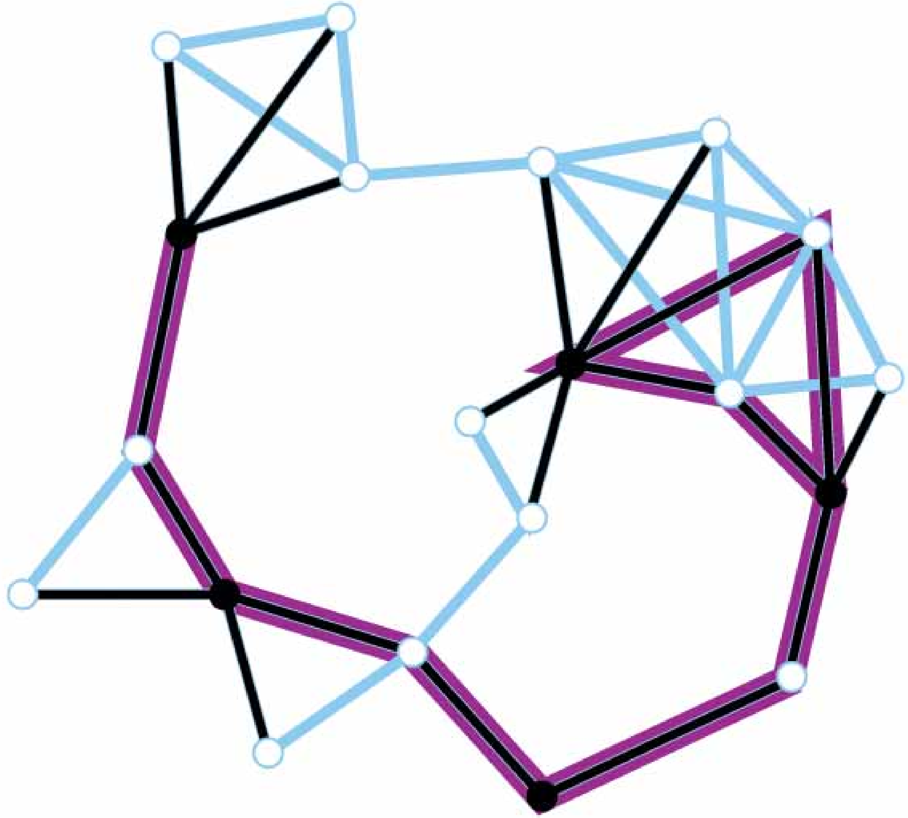
Graph Domination Variants

- Dominating set
- Connected dominating set



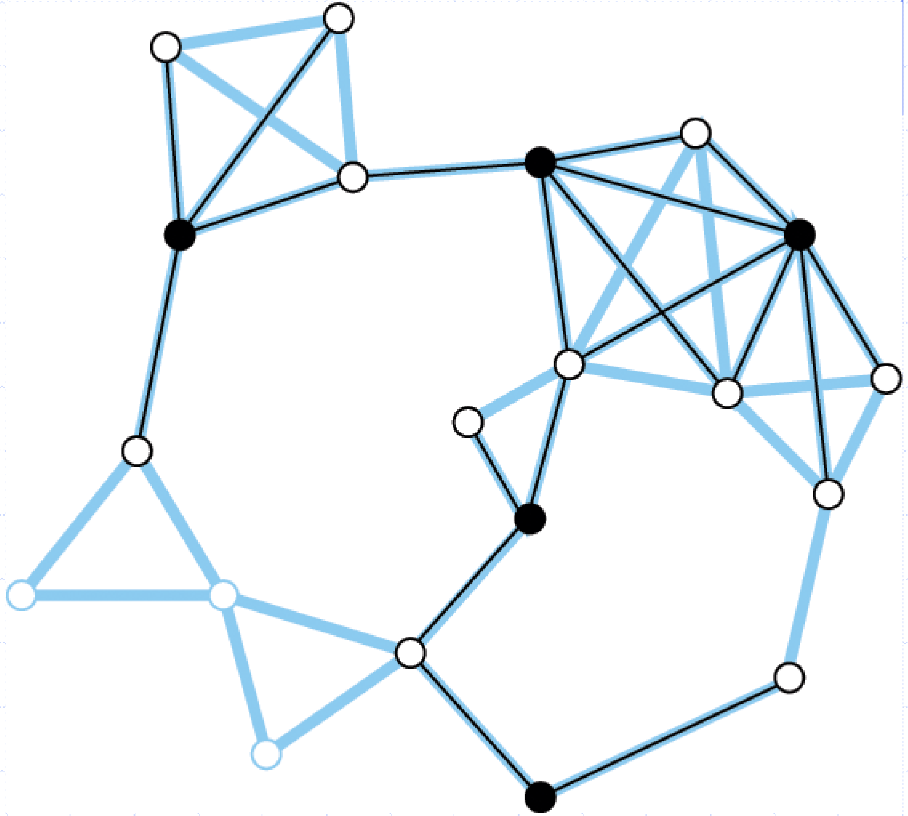
Graph Domination Variants

- Dominating set
- Connected dominating set
- Weakly-connected dominating set



Graph Domination Variants

- Dominating set
- Connected dominating set
- Weakly-connected dominating set
  - Weakly induced subgraph <S>$_w$



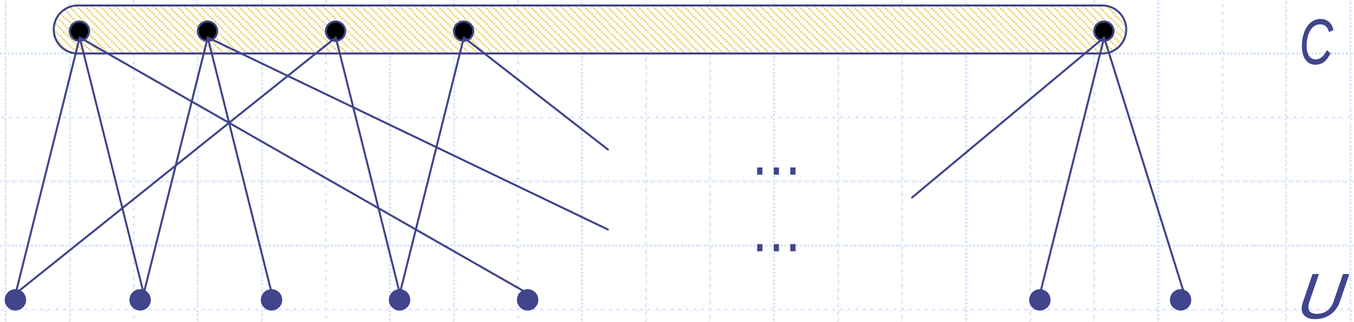
Cost of Simplicity

- Notation
  - Domination number – γ
  - Connected Domination number – γ_c
  - Weakly-Connected Domination number – γ_w
- To determine γ , γ_c and γ_w is NP-complete
- Impossible to approximate with a ratio better than $(1-\varepsilon)\log\Delta$ for any $\varepsilon > 0$ unless

$$NP \subseteq DTIME(n^{\log\log n})$$

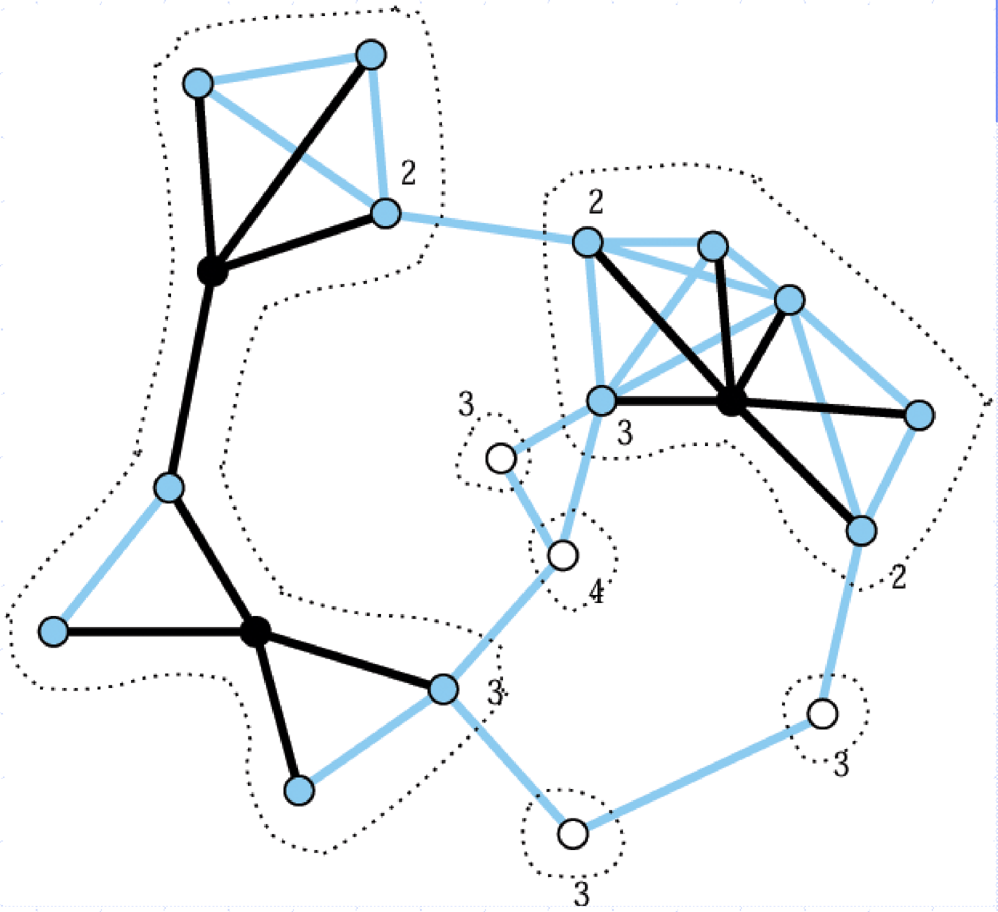
NP-Completeness and Approximability

- Reduction from the SETCOVER problem
  - Instance: Universe $U = \{v_1, v_2, v_3, \ldots, v_n\}$, collection of subsets $C = \{C_1, C_2, C_3, \ldots, C_m\}$ of U
  - Question: Does there exist a subset set of C of size k that covers U?



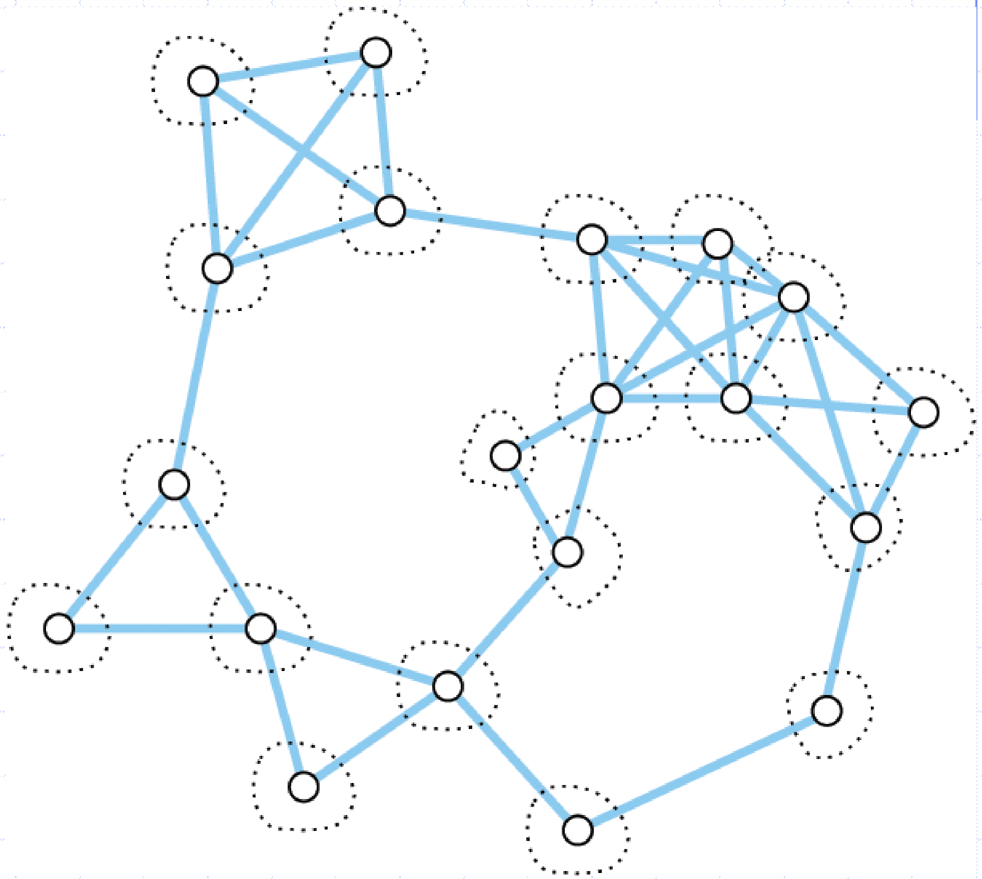
Approximation Algorithm I

- Greedy process of growing multiple pieces
- Notions
  - White piece
  - Black piece
  - Improvement value



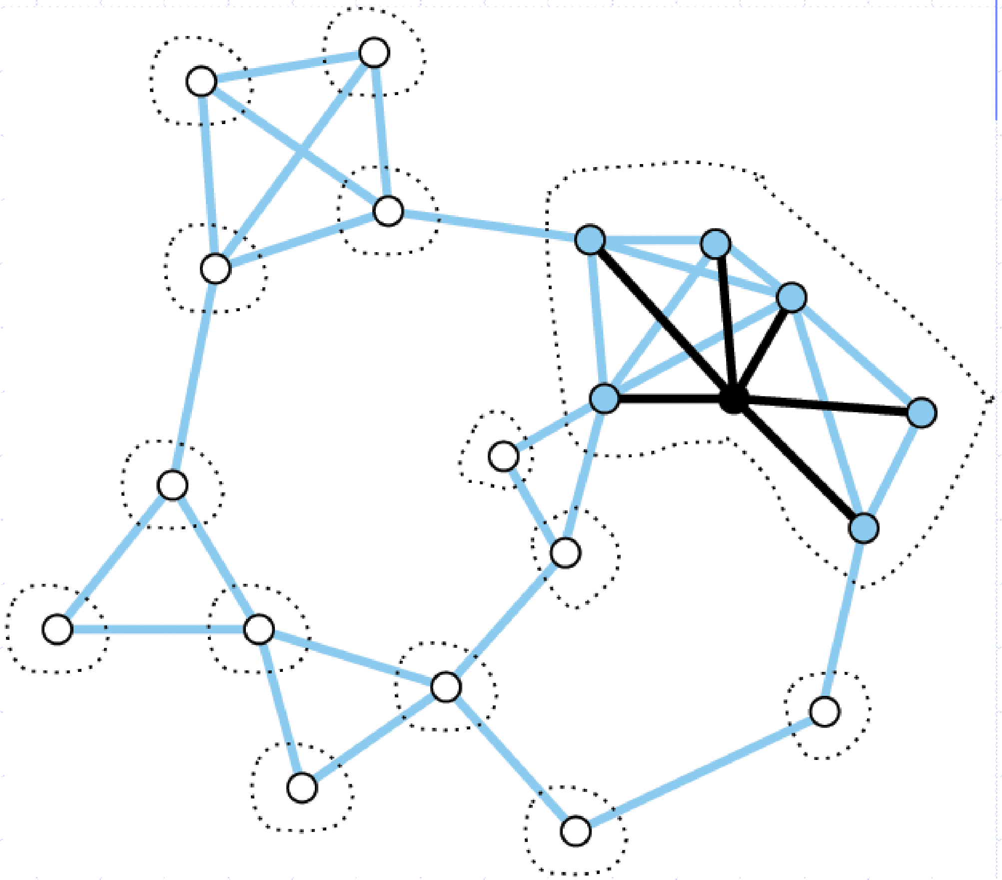
Approximation Algorithm I

- Greedy process of growing multiple pieces
- Notions
  - White piece
  - Black piece
  - Improvement value



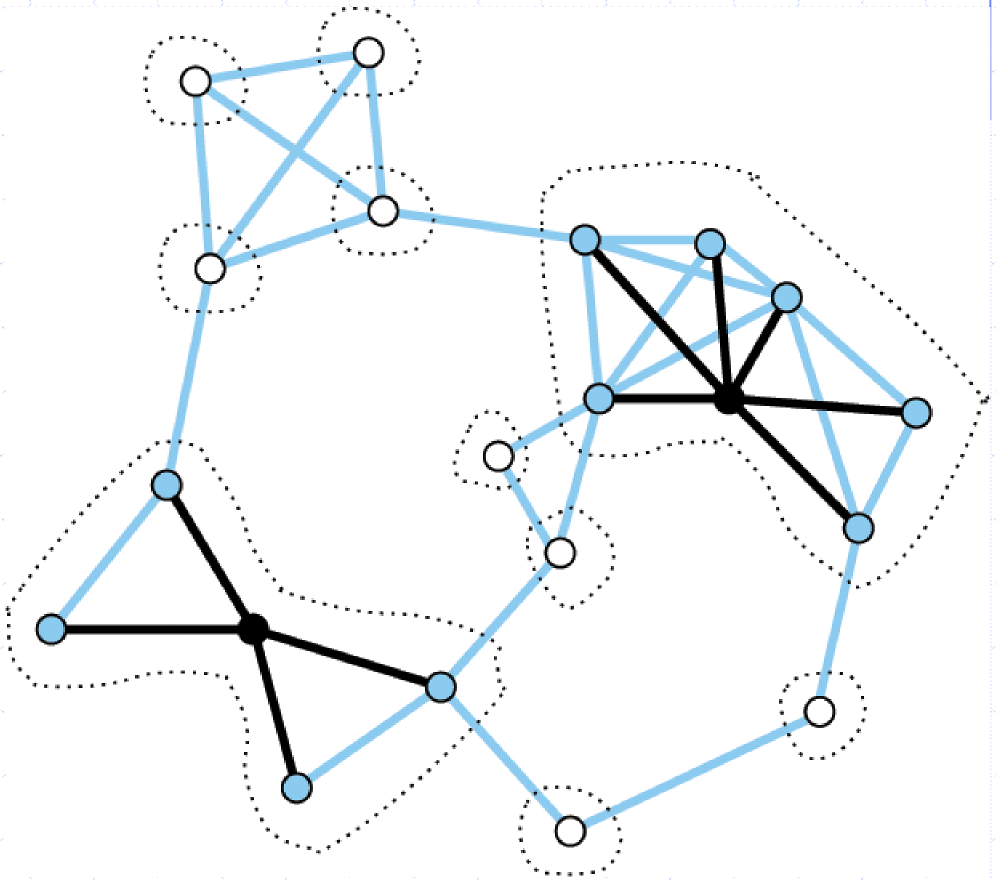
Approximation Algorithm I

- Greedy process of growing multiple pieces
- Notions
  - White piece
  - Black piece
  - Improvement value



Approximation Algorithm I

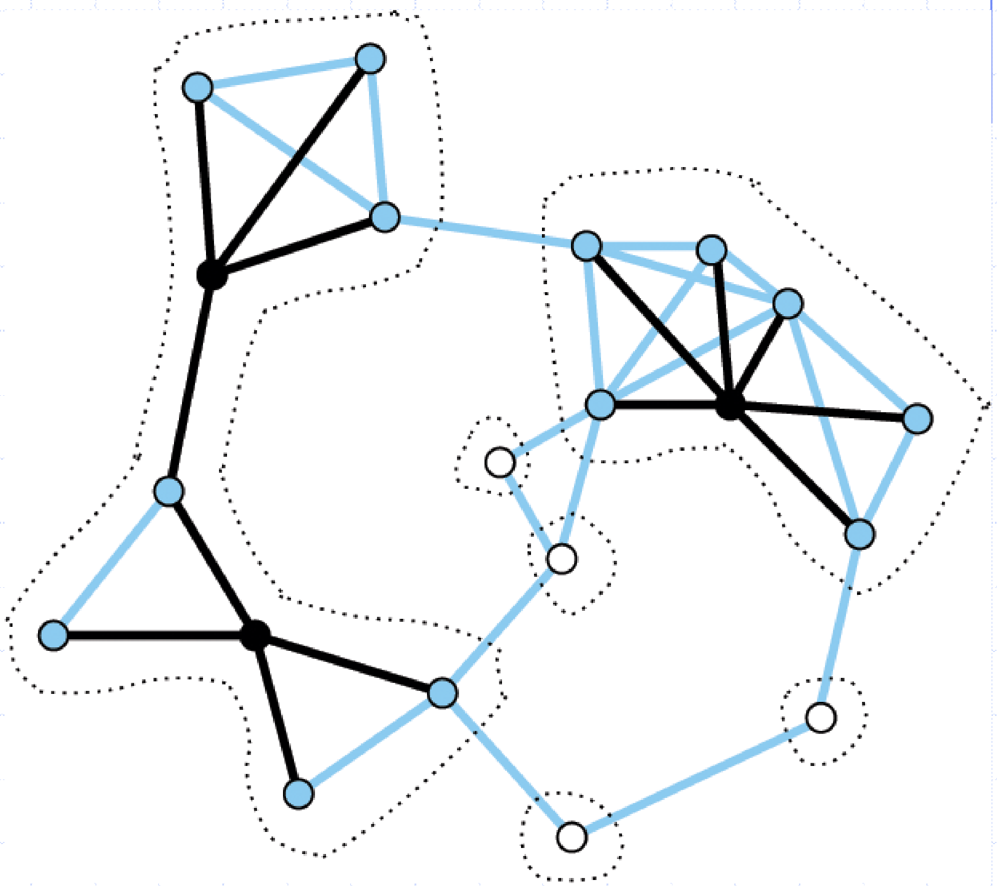
- Greedy process of growing multiple pieces
- Notions
  - White piece
  - Black piece
  - Improvement value



Approximation Algorithm I

- Greedy process of growing multiple pieces

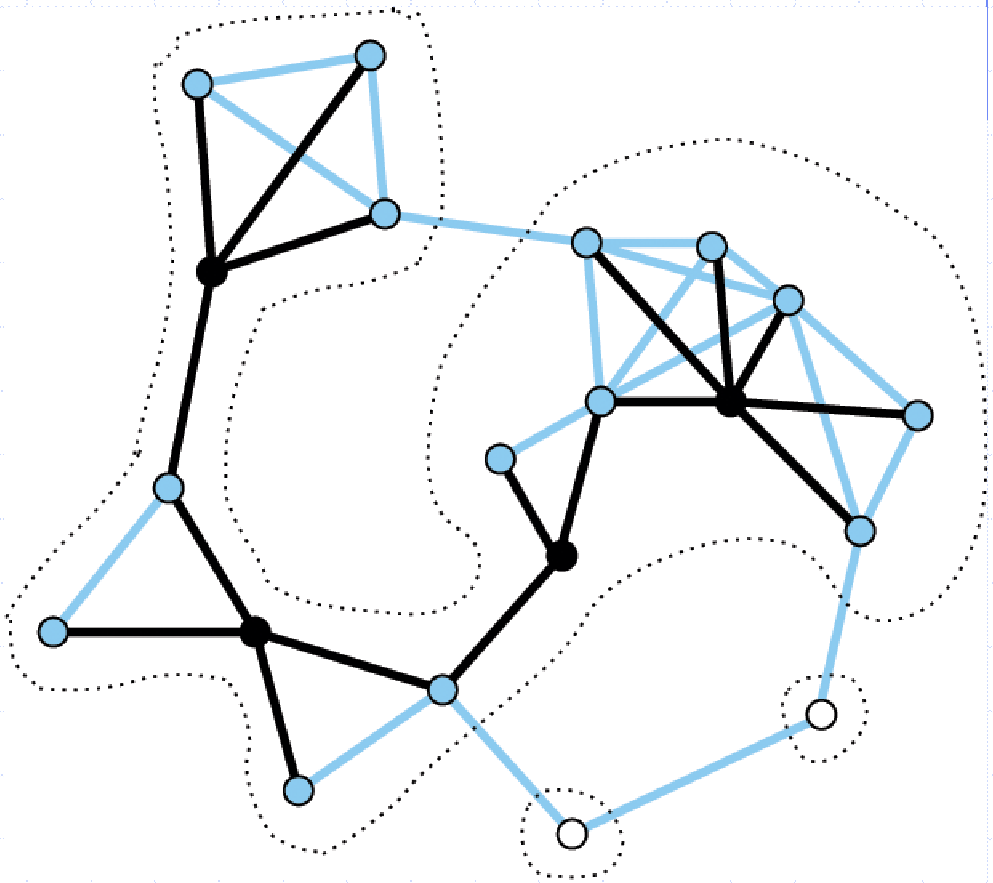
- Notions
  - White piece
  - Black piece
  - Improvement value



Approximation Algorithm I

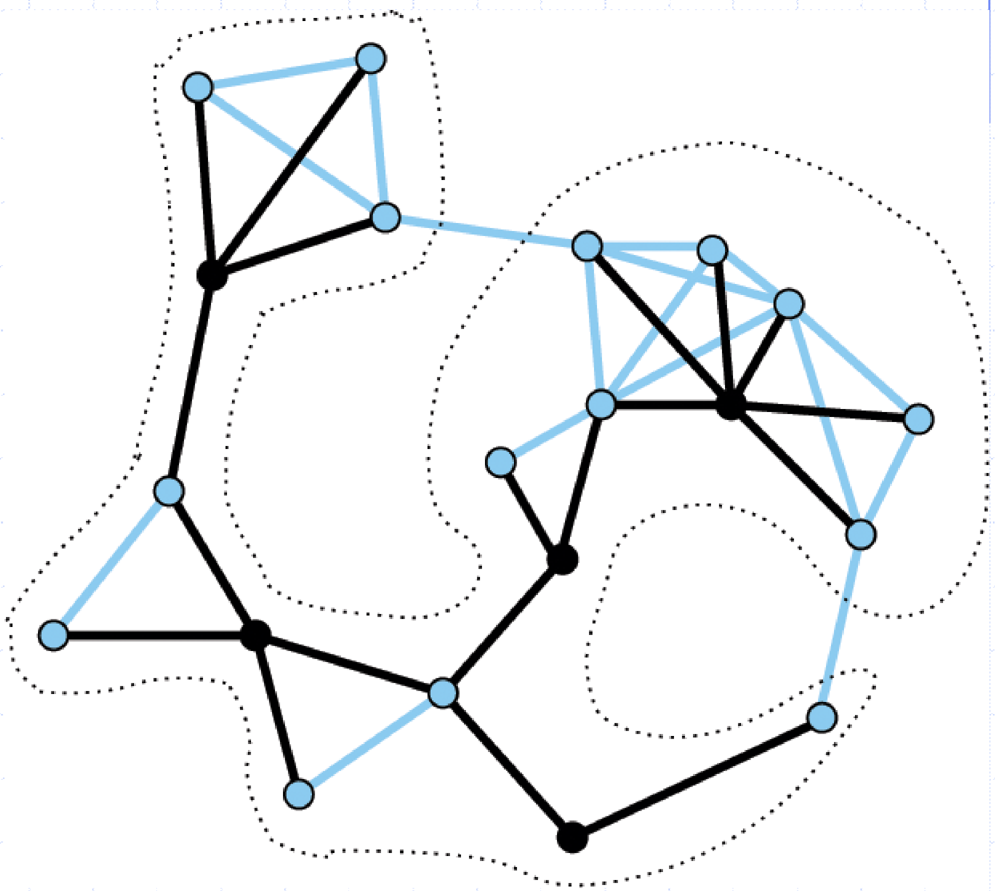
- Greedy process of growing multiple pieces

- Notions
  - White piece
  - Black piece
  - Improvement value



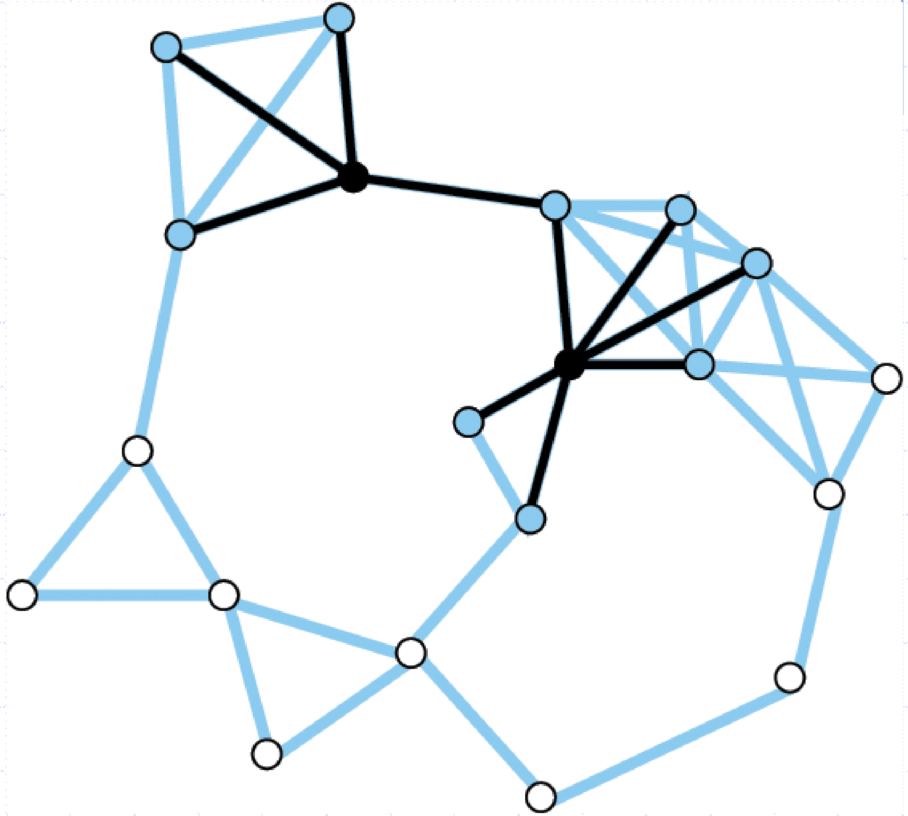
Approximation Algorithm I

- Greedy process of growing multiple pieces
- Notions
  - White piece
  - Black piece
  - Improvement value



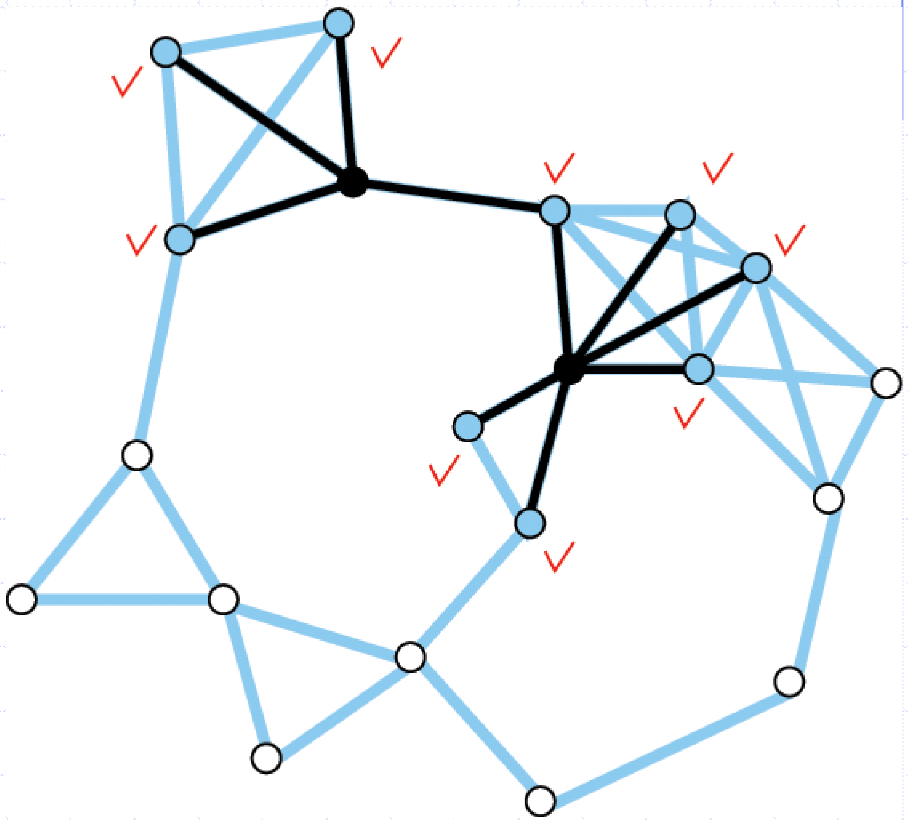
Approximation Algorithm II

- Greedy process of growing single piece
- Notions
  - Candidate vertex



Approximation Algorithm II

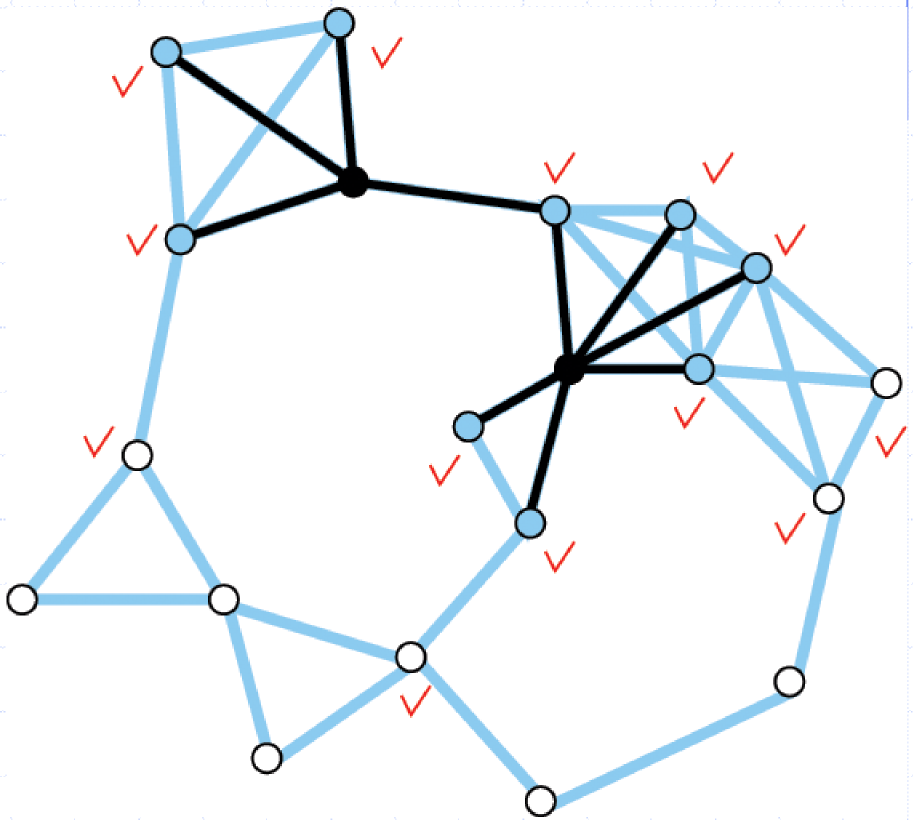
- Greedy process of growing single piece
- Notions
  - Candidate vertex



Approximation Algorithm II

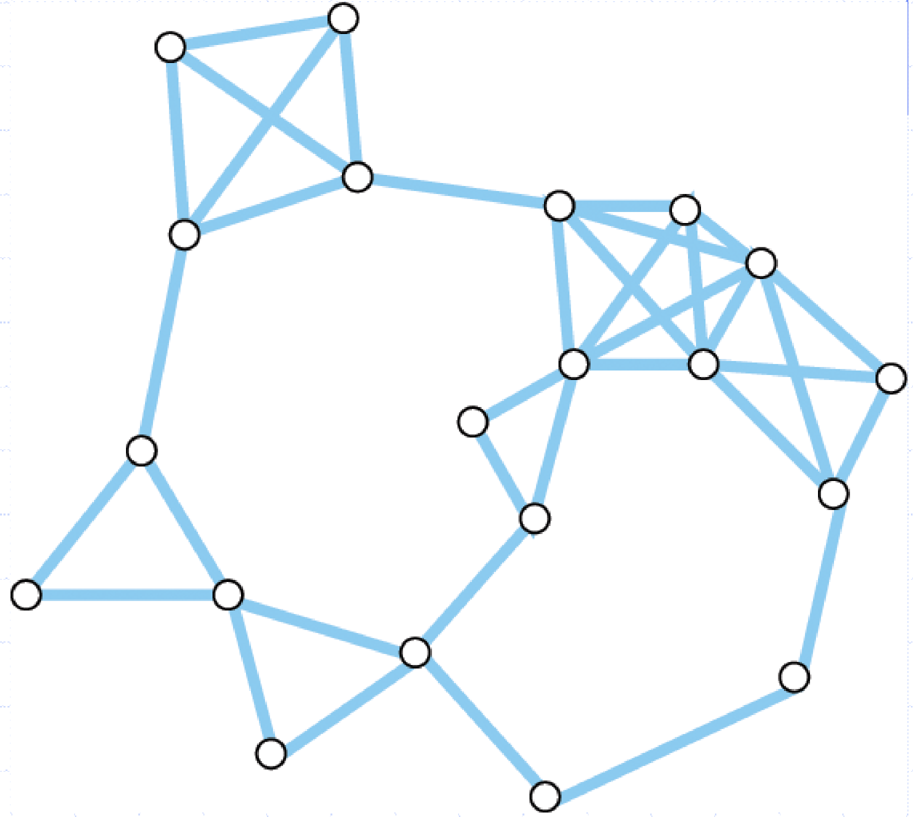
- Greedy process of growing single piece

- Notions
  - Candidate vertex



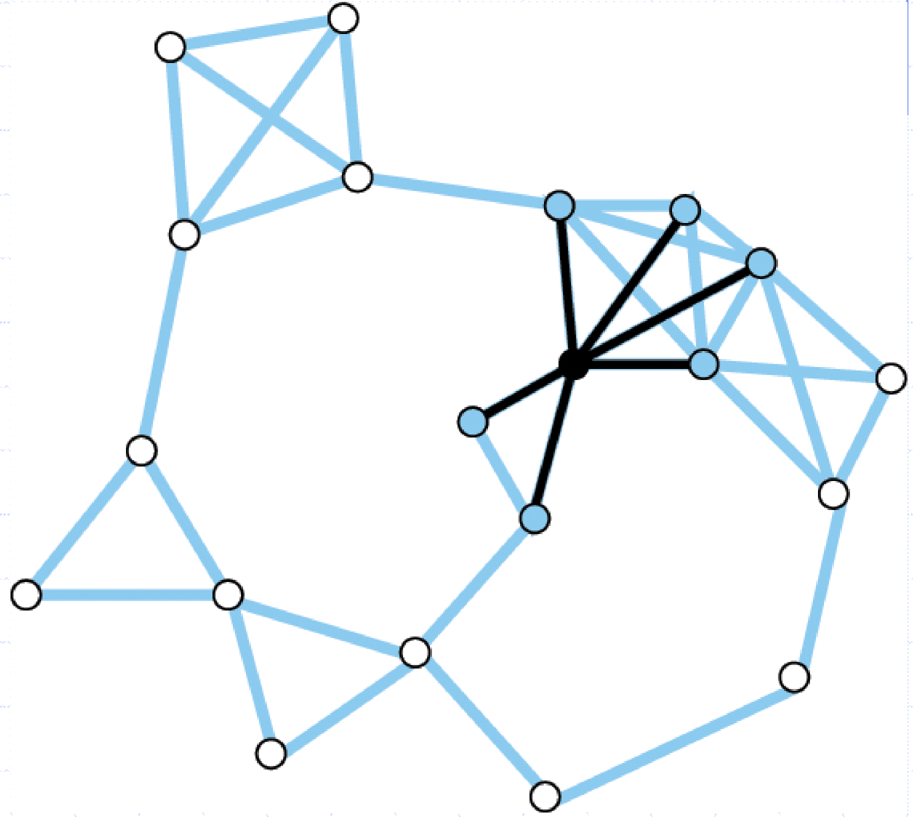
Approximation Algorithm II

- Greedy process of growing single piece
- Notions
  - Candidate vertex



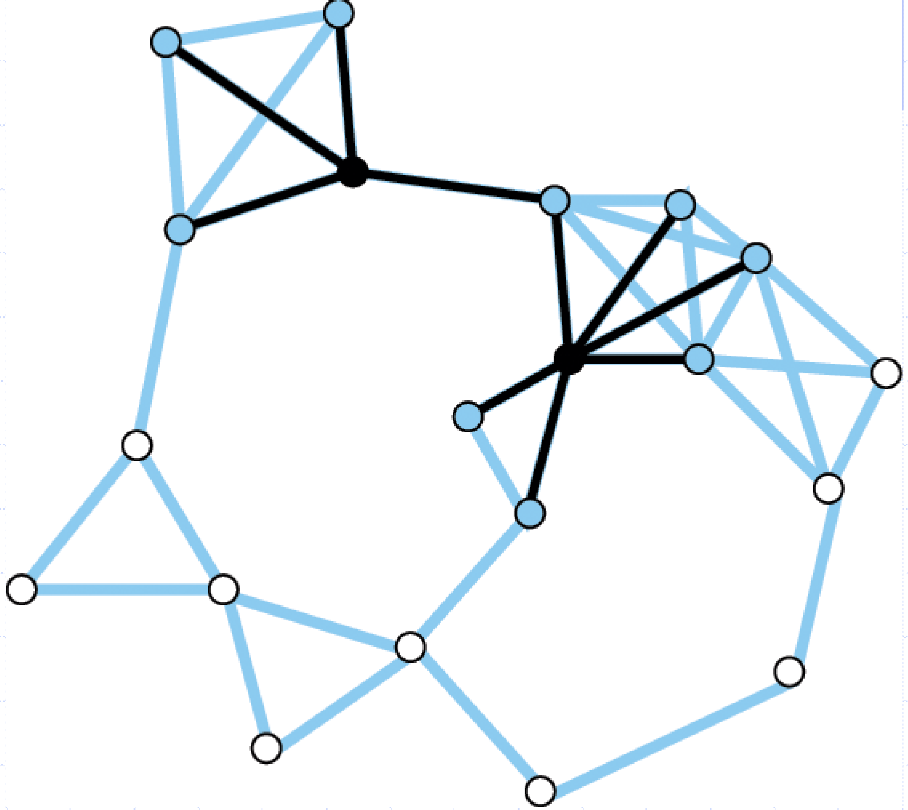
Approximation Algorithm II

- Greedy process of growing single piece
- Notions
  - Candidate vertex



Approximation Algorithm II

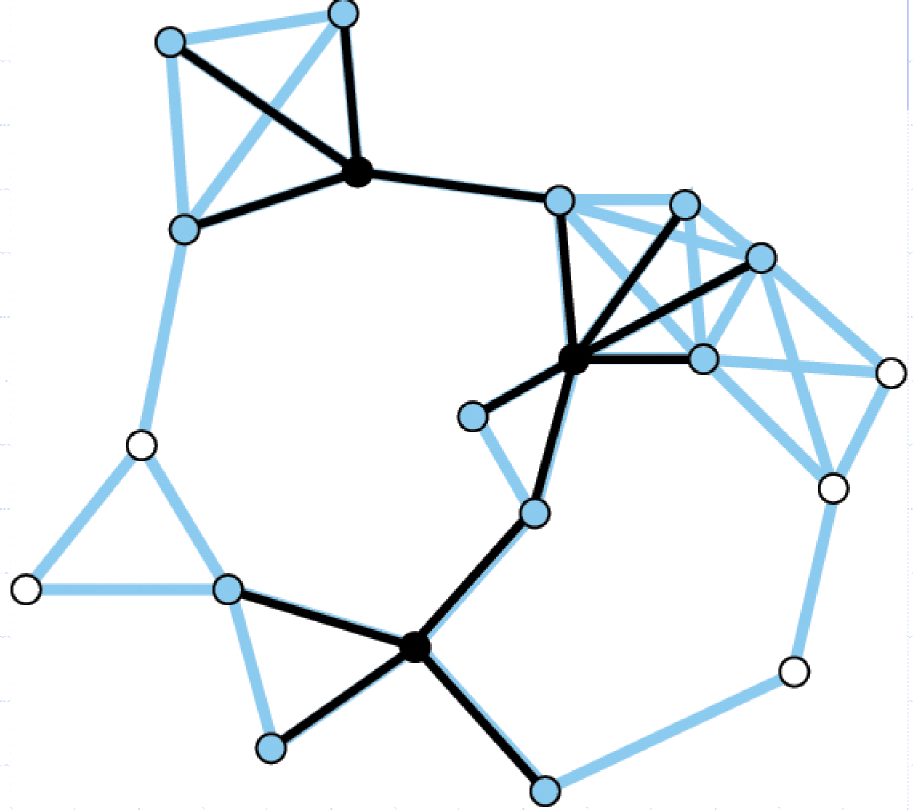
- Greedy process of growing single piece
- Notions
  - Candidate vertex



Approximation Algorithm II

- Greedy process of growing single piece

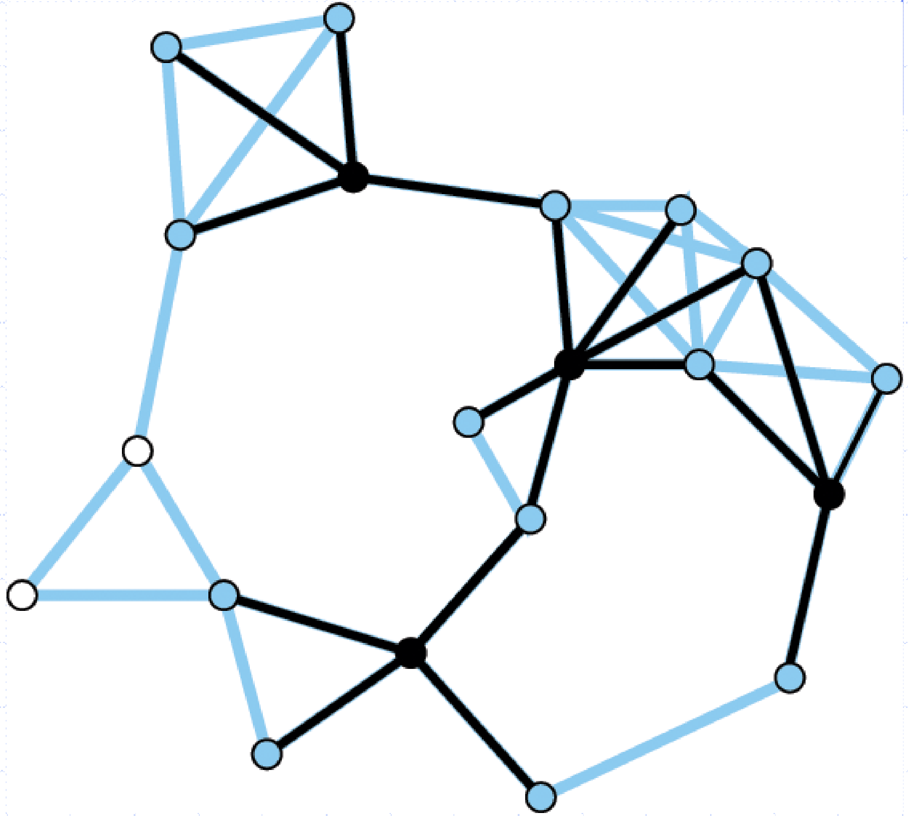
- Notions
  - Candidate vertex



Approximation Algorithm II

- Greedy process of growing single piece

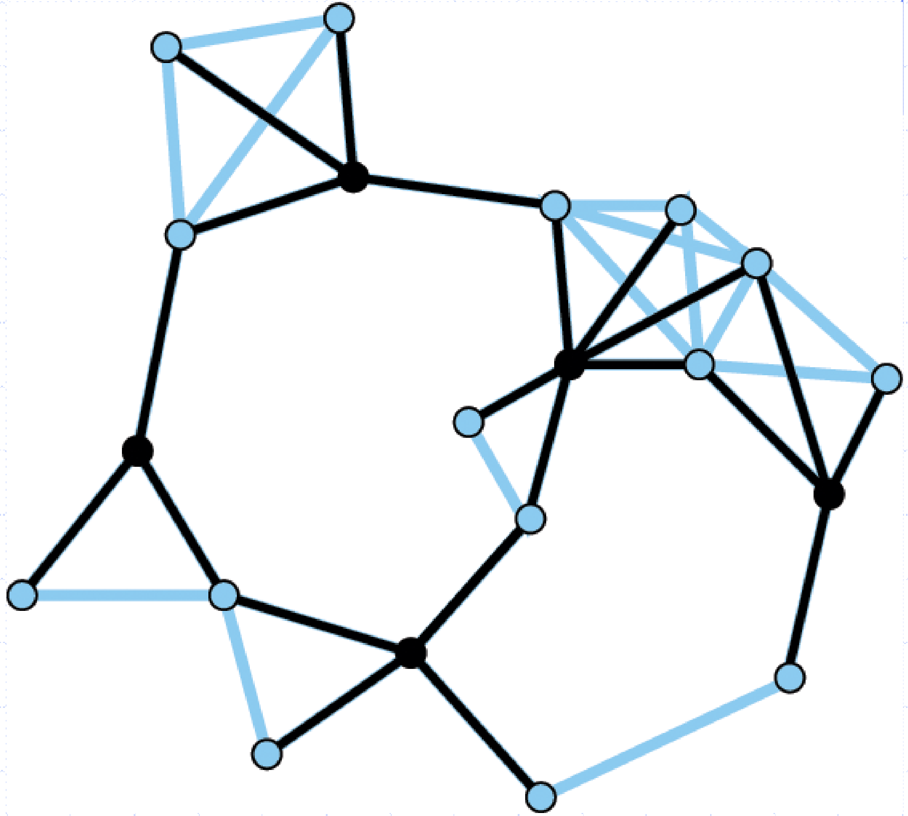
- Notions
  - Candidate vertex



Approximation Algorithm II

- Greedy process of growing single piece

- Notions
  - Candidate vertex

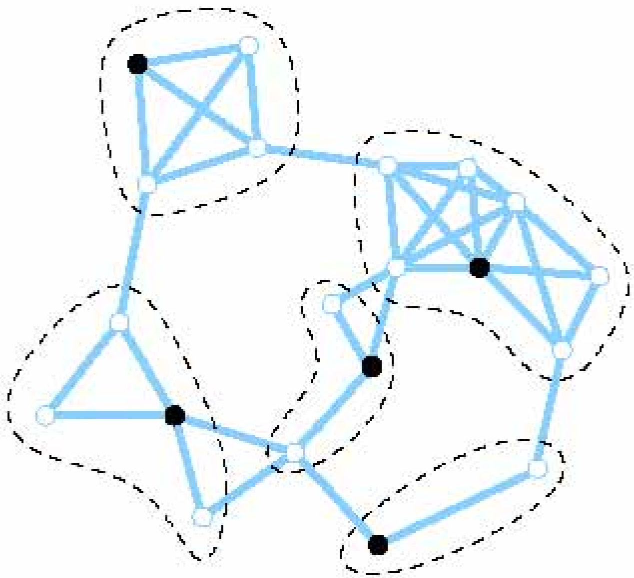


Performance Ratios

- The performance ratio of Algorithm I is $\log\Delta + 1$
- The performance ratio of Algorithm II is $\log\Delta + 2$

Proof Sketch (Algorithm II)

- $OPT_{DS} = \{v_1, v_2, \ldots, v_\gamma\}$ — minimum dominating set
- Partition V into γ disjoint sets P_i $(1 \le i \le \gamma)$, s.t.
  - $v_i \in P_i$
  - $\forall w \in V - OPT_{DS}$, place w into P_i, where v_i is any dominator of w.



Proof Sketch (Cont'd)

Charging analysis

- When a vertex is colored black, charge one unit
- Charge distributed among white vertices colored gray — every white vertex is charged $\frac{1}{x}$ if x white vertices become gray
- Total charge within P_i is at most $\ln \Delta + 2$

Proof Sketch (Cont'd)

- u_j — number of white vertices in P_i after the jth iteration
  - $u_0 = |P_i|$

- Number of vertices of P_i colored after the 1st iteration is $u_0 - u_1$
  - Each such vertex is charged at most $\frac{1}{u_0 - u_1}$

Proof Sketch (Cont'd)

- Once any vertex of P_i is colored in iteration $j-1$, v_i is a candidate
  - In iteration j, coloring v_i black would dominate at least u_{j-1} new vertices
  - Any vertex actually colored black in iteration j dominates at least u_{j-1} new vertices
- Thus, in the iteration j, at most $\frac{u_{j-1}-u_j}{u_{j-1}}$ charge is incurred in P_i
- Until $u_k = 0$

Proof Sketch (Cont'd)

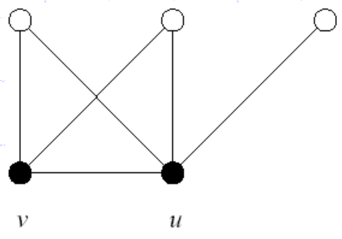
Summing the charges within P_i, we get at most

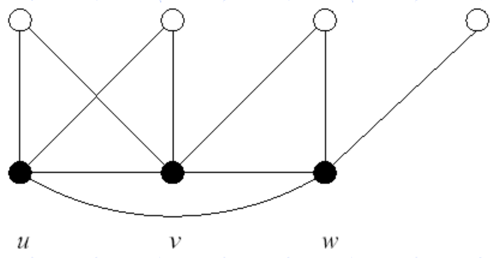
$$\begin{aligned}
& \frac{1}{u_0 - u_1}(u_0 - u_1) + \sum_{j=2}^{k} \frac{1}{u_{j-1}}(u_{j-1} - u_j) \\
= \quad & 1 + \sum_{j=2}^{k} \frac{u_{j-1} - u_j}{u_{j-1}} \\
\leq \quad & 1 + \sum_{j=2}^{k} (H(u_{j-1}) - H(u_j)) \\
= \quad & 1 + (H(u_1) - H(u_k)) \\
\leq \quad & 2 + \ln \Delta
\end{aligned}$$

Local Computation

[Wu&Li99]

- Marking phase – each node marks it self *T* if any two of its neighbors are not adjacent
- Un-marking phase – a *T* vertex *v* changes its mark back to F if either of the following holds
  - $\exists u \in N(v)$ which is marked *T* such that $N[v] \subseteq N[u]$ and $id(v) < id(u)$
  - $\exists u, w \in N(v)$ which are both marked *T* such that $N[v] \subseteq N[u] \cup N[w]$ and $id(v) < \min\{id(u), id(w)\}$

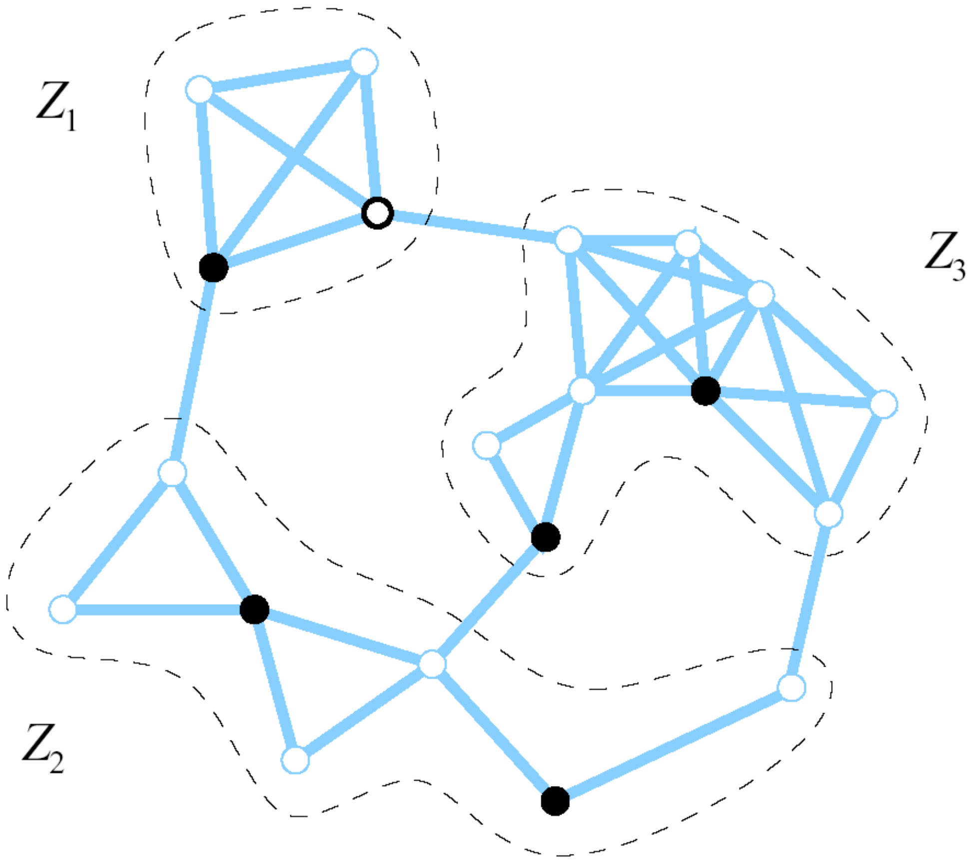




Zonal Computation

[Chen&Liestman03]

- Zone
- Size control parameter *x*
- Intra-zonal
- Inter-zonal



Without Connectivity

- [Liang & Haas 00] Localized distributed approximation algorithm for small dominating sets with performance ratio O(log Δ)
- [Jia & Rajaraman & Suel 01] Randomization of above, with expected performance ratio O(log Δ) and terminates in O(log Δ log| V|) w.h.p.

Connectivity and Approximation Ratio

- [Dubhashi, et al 03] Semi-localized distributed approximation algorithm for small dominating sets with performance ratio O(log Δ)
- Information needed within radius $\lfloor 1+2\log|V|\rfloor$ of each node
- Let S be the small dominating set of $G=<V,E>$ generated by the algorithm of Liang&Haas
- Construct $G'=<S,E'>$ where each element of E' corresponds to a path in G of length ≤ 3
- Remove cycles in $G'=<S,E'>$ of length less than $\lfloor 1+2\log|S|\rfloor$
- It can be shown that G' has at most $2|S|$ edges left
  - Lemma 15.3.2 of [Matousek02]: A simple undirected graph on n vertices of girth g has at most $n^{1+\frac{2}{g-1}}+n$ edges

Clustering – How Expensive?

- John Sucec, Ivan Marsic: Hierarchical Routing Overhead in Mobile Ad Hoc Networks. IEEE Trans. Mob. Comput. 3(1):46-56, 2004