A Greedy Approximation for Minimum Connected Dominating Sets in Wireless Ad Hoc Networks

Presented By Yingshu Li
yili@cs.umn.edu

10-3-2003

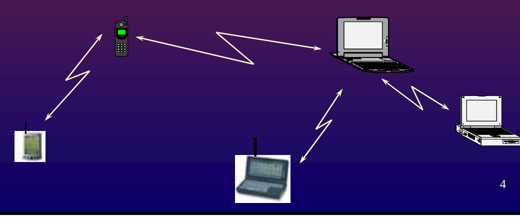
Outline

- Introduction
- Problem Definition and Preliminaries
- A Greedy Approximation Algorithm
- Evaluation of the Algorithm
- Conclusion and future work

Introduction

Introduction --- Wireless Ad Hoc networks

A wireless ad hoc network consists of a collection of mobile hosts dynamically forming a temporary network without the use of any existing network infrastructure.



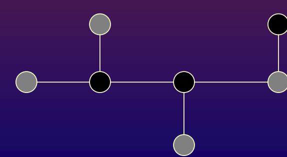
Introduction --- Wireless Ad Hoc networks

- Instantly deployable and re-configurable
- Node mobility
- Shared, scarce wireless channel
- Limited battery power
- Multihop routing

5

Introduction --- Dominating set

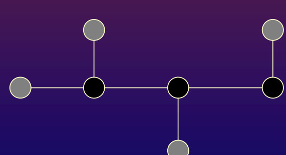
A dominating set of a graph is a subset of all the nodes such that each node is either in the dominating set or adjacent to some node in the dominating set.



6

Introduction --- Connected Dominating set

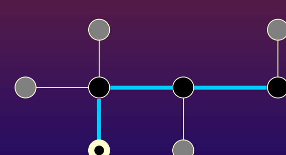
A connected dominating set of a graph is a subset of the nodes such that it forms a dominating set in the graph and the subgraph induced is connected.



7

Introduction --- Applications

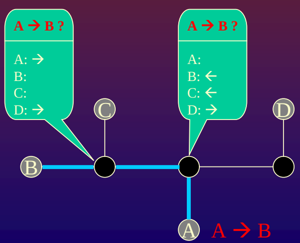
Doing broadcast



8

Introduction --- Applications

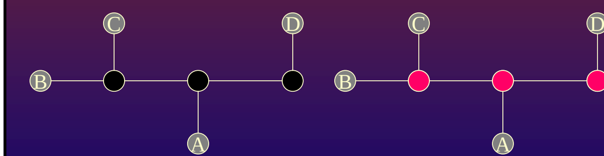
Efficient unicast routing



9

Introduction --- Applications

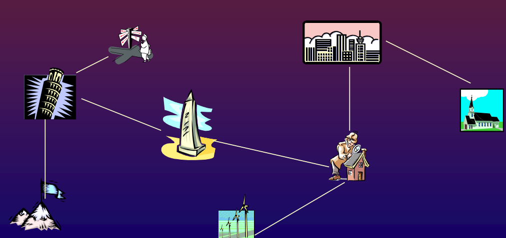
Virtual backbone



10

Introduction --- Applications

Traveling tourist problem



11

Introduction --- Current solutions

- Centralized algorithms

  Best performance ratio --- $3 + ln\delta$

  δ: the maximum vertex degree

- Distributed algorithms

  Best performance ratio --- 7

12

Problem Definition and Preliminaries

Problem Definition and Preliminaries

Minimum connected dominating set

Given a graph $G = (V, E)$, find the smallest subset C of vertices such that C induces a connected subgraph and each vertex in $V - C$ is adjacent to at least one vertex in C.

14

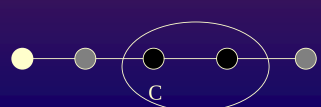
Problem Definition and Preliminaries

Given a graph G and a subset C of vertices in G, all vertices in G can be divided into three classes.

Black vertices: Vertices belong to C

Grey vertices: Vertices are not in C but adjacent to C

White vertices: Vertices are not in C and not adjacent to C.



15

Problem Definition and Preliminaries

Black component: a connect component in the subgraph induced by all black vertices

f = Number of white vertices + Number of black components

$f = 1$

16

Problem Definition and Preliminaries

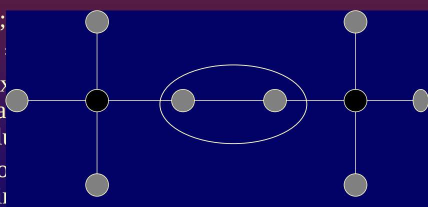
Two-stage Greedy Algorithm by Guha and Khuller

Stage 1:

Set *w*:=1;

while (*w*

if there ex   g it in
black a   l reduce
the valu



then choo   luced in
a maxi

else set *w*:=0; }

17

Problem Definition and Preliminaries

Stage 2:

Recursively connect pairs of black components by coloring a chain of gray vertices black, until there is one connected black component.

Performance ratio = 3 + $ln\delta$

δ: the maximum vertex degree

18

A Greedy Approximation

A Greedy Approximation

Improvements:

Modified function *f*

One-stage algorithm

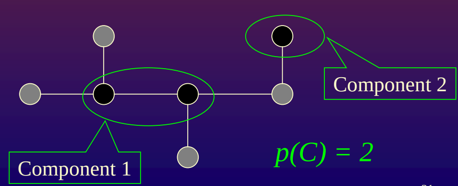
Performance ratio = 2 + $ln\delta$

20

A Greedy Approximation

For each vertex subset C,

$p(C)$: the number of connected components of the subgraph induced by all black vertices in C.



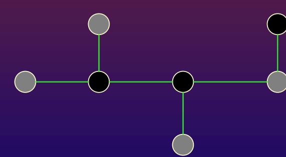
$p(C) = 2$

21

A Greedy Approximation

For each vertex subset C,

$D(C)$: the set of all edges incident to vertices in C

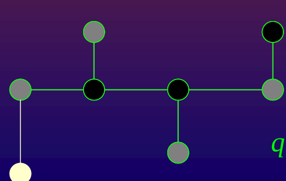


22

A Greedy Approximation

For each vertex subset C,

$q(C)$: the number of connected components of the subgraph with vertex set V and edge set $D(C)$, denoted by $(V, D(C))$



$q(C) = 1$

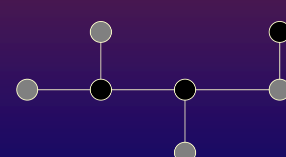
23

A Greedy Approximation

Revised function $f(C) = p(C) + q(C)$

$f(C) \geq f(C \cup \{A\})$, A is any subset of V

$f(C) \geq 2$



$f(C) = 3$

24

A Greedy Approximation

Stage 1:

Set w:=1;

while (w =1) {

if there exits a white or gray vertex such that coloring it in black and its adjacent white vertices in gray would reduce the value of the function f

then choose such a vertex to make the value of f reduced in a maximum amount

else set w:=0; }

25

Evaluation of the Algorithm

Evaluation of the Algorithm

Lemma 1 *Suppose G is a connected graph with at least three vertices. Then, C is a connected dominating set if and only if* $f(C \cup \{x\}) = f(C)$ *for every* $x \in V$.

27

Evaluation of the Algorithm

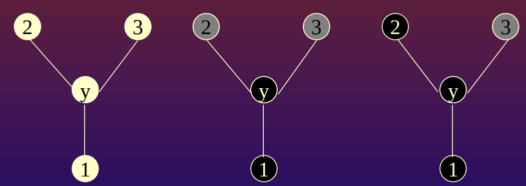
For any $C \subseteq V$, $y \in V$ and $y \notin C$, denote

$$\Delta_y q(C) = q(C) - q(C \cup \{y\})$$

Lemma 2 *If* $A \subset B$, *then* $\Delta_y q(A) \geq \Delta_y q(B)$.

28

Evaluation of the Algorithm



A = {1}
B = {1, 2}

G_A = (V, D(A))
$\Delta_y q$(A) = 3-1=2

G_B = (V, D(B))
$\Delta_y q$(B) = 2-1=1

Evaluation of the Algorithm

C^* : a minimum connected dominating set for G

a_i : the value of f when i vertices have been colored black in Greedy Algorithm, $a_0 = n$

Lemma 3 *For i = 1, 2,…, |C|.*

$$a_i \leq a_{i-1} - \frac{a_{i-1} - 2}{|C^*|} + 1$$

Evaluation of the Algorithm

Theorem 1 *The Greedy algorithm potential function f produces an approximation solution for minimum connected dominating set with performance ratio 2 + lnδ where δ is the maximum vertex-degree in input graph.*

$$|C_G| \leq |C^*|(2 + \ln \delta)$$

Conclusion and future work

- A greedy approximation for minimum connected dominating sets with performance ratio 2 + *lnδ.*
- Hosts may have different transmission range
- Distributed algorithms in mobile environment

Proof of Lemma 1

C is a connected dominating set $\Rightarrow f(C \cup \{A\}) = f(C)$

If C is a connected dominating set, then $f(C) = 2$, which reaches the minimum value. Therefore, $f(C \cup \{A\}) = f(C)$ for every $x \in V$.

Proof of Lemma 1

$f(C \cup \{A\}) = f(C) \Rightarrow C$ is a connected dominating set

a. C cannot be an empty set

Suppose $C = \varnothing$.

G is a connected graph with ≥ 3 vertices $\Rightarrow$ there must exist a vertex x with degree ≥ 2 and for such a vertex x, $f(C \cup \{x\}) < f(C) \Rightarrow$ Contradiction.

Proof of Lemma 1

b. Consider a connected component of the subgraph induced by C.

For every gray vertex y adjacent to C,

- y is adjacent to a black vertex not in C: $p(C \cup \{y\}) < p(C)$ and $q(C \cup \{y\}) < q(C)$
- y is adjacent to a white vertex or a gray vertex not adjacent to C: $p(C \cup \{y\}) = p(C)$ and $q(C \cup \{y\}) < q(C)$

Proof of Lemma 1

- Therefore, every gray vertex adjacent to C cannot be adjacent to any vertex neither in C nor adjacent to C.
- It follows that every vertex of G must belong to C or adjacent to C. That is, C is a connected dominating set.

Proof of Lemma 3

Let $x_1, x_2, \ldots x_{|C|}$ be elements of C_G in the ordering of their appearance in the Greedy Algorithm. Denote $C_i = \{x_1, x_2, \ldots, x_i\}$. Then

$$a_i = f(C_i) = a_{i-1} - \Delta_{x_i} f(C_{i-1})$$

Where

$$\Delta_{x_i} f(C_{i-1}) = \max_y \Delta_y f(C_{i-1}).$$

37

Proof of Lemma 3

$C_i = \{x_1, x_2, \ldots, x_i\}$, $C_j^* = \{y_1, y_2, \ldots, y_j\}$.

Since C^* is a connected dominating set, we can always arrange elements of C^* in an ordering $y_1, y_2, \ldots, y_{|C^*|}$ such that y_1 is adjacent to a vertex in C_{i-1} and for $j \geq 2$, y_j is adjacent to a vertex in $\{y_1, y_2, \ldots, y_{j-1}\}$. Denote

$$\Delta_{C^*} f(C_{i-1}) = \sum_{j=1}^{|C^*|} \Delta_{y_j} f\left(C_{i-1} \bigcup C^*_{j-1}\right)$$

38

Proof of Lemma 3

Note that

$$\Delta_{y_j} p\left(C_{i-1} \bigcup C^*_{j-1}\right) \leq \Delta_{y_j} p(C_{i-1}) + 1$$

Moreover, by Lemma 2,

$$\Delta_{y_j} q\left(C_{i-1} \bigcup C^*_{j-1}\right) \leq \Delta_{y_j} q(C_{i-1})$$

Therefore,

$$\Delta_{y_j} f\left(C_{i-1} \bigcup C^*_{j-1}\right) \leq \Delta_{y_j} f(C_{i-1}) + 1$$

39

Proof of Lemma 3

It follows that

$$\begin{aligned} a_{i-1} - 2 &= f(C_{i-1}) - f\left(C_{i-1} \bigcup C^*\right) \\ &= \Delta_{C^*} f(C_{i-1}) \\ &\leq \sum_{j=1}^{|C^*|} \left(\Delta_{y_j} f(C_{i-1}) + 1\right) \end{aligned}$$

40

Proof of Lemma 3

There exists $y_j \in C^*$ such that

$$\Delta_{y_j} f(C_{i-1}) + 1 \ge \frac{a_{i-1} - 2}{|C^*|}$$

Hence,

$$a_{i-1} - a_i = \Delta_{x_i} f(C_{i-1}) \ge \frac{a_{i-1} - 2}{|C^*|} - 1$$

It implies that

$$a_i \le a_{i-1} - \frac{a_{i-1} - 2}{|C^*|} + 1$$

41

Proof of Theorem 1

By Lemma 3,

$$\begin{aligned} a_i - 2 &\le (a_{i-1} - 2)\left(1 - \frac{1}{|C^*|}\right) + 1 \\ &\le (a_0 - 2)\left(1 - \frac{1}{|C^*|}\right)^i + \sum_{k=0}^{i-1}\left(1 - \frac{1}{|C^*|}\right)^k \\ &= (a_0 - 2)\left(1 - \frac{1}{|C^*|}\right)^i + |C^*|\left(1 - \left(1 - \frac{1}{|C^*|}\right)^i\right) \\ &= (n - 2 - |C^*|)\left(1 - \frac{1}{|C^*|}\right)^i + |C^*| \end{aligned}$$

42

Proof of Theorem 1

$a_{i-1} \ge a_i + 1 \Rightarrow a_{|C_G|-1} \ge a_{|C_G|} + 1$

$\Rightarrow a_{|C_G|-2|C^*|} \ge a_{|C_G|} + 2|C^*|$

$\Rightarrow a_{|C_G|-2|C^*|} \ge 2 + 2|C^*|$

Set $i = |C_G| - 2|C^*|$, *then* $2|C^*| \le a_i - 2$, *thus*

$$2|C^*| \le (n - 2 - |C^*|)\left(1 - \frac{1}{|C^*|}\right)^i + |C^*|$$

43

Proof of Theorem 1

Since

$$\left(1 - \frac{1}{|C^*|}\right)^i \le e^{-\frac{i}{|C^*|}}$$

We obtain

$$i \le |C^*| \ln \frac{n - 2 - |C^*|}{|C^*|}$$

44

Proof of Theorem 1

Note that each vertex can dominate at most $\delta+1$ vertices. Hence, $n/|C^*| \le \delta+1$. Therefore,

$$|C_G| = i + 2|C^*| \le |C^*|(2 + \ln \delta)$$

45