Distributed Approximation Algorithm for Minimum Dominating Set in Planar Graphs.

Model of computation

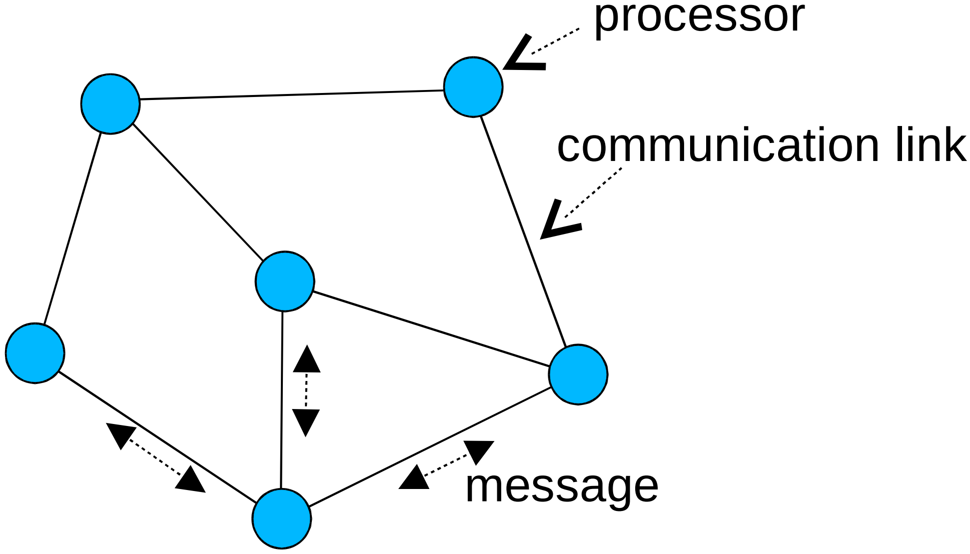
distributed,
synchronous,
message passing
model of computation

Synchronous <=>
computation proceed in rounds

In each *round*, each vertex:

1. send messages to its neighbours
2. receive messages from its neighbours
3. does local computation.

Communication graph:



Model of computation

- Our algorithm computes "something" in the communication graph (there is no other input graph).
- Vertices have different **ID**; **n:=|V(G)|** is known.
- Randomization is not allowed.
- In a ball of radius **t** it is possible to compute "everything" in time **O(t)** (unlimited computational power of a single vertex ...).

Our problem

to find approximation of
Minimum Dominating Set (MDS)
and *Min Connected Dom Set (MCDS)*

- in planar graph
- in distributed/ synchronous model of computation.
- in polylog(|V(G)|) time

Definition of MDS/ MCDS:

1. A set of vertices **D** is a dominating set in graph **G** if each vertex of **G** is in **D** or has a neighbour in **D**.
2. **MDS(G)** is a dominating set of the smallest cardinality.
3. **MCDS(G)** is a dominating set **D** such that **G[D]** is connected, of the smallest cardinality.

Approximation ratio and time

our algorithm computes

almost exact approximation

of MDS/ MCDS problem ...

Approx. ratio → $\frac{|D|}{|D^*|} \le 1 + \frac{1}{\log n}$

n =|V(G)|; G -comm. graph

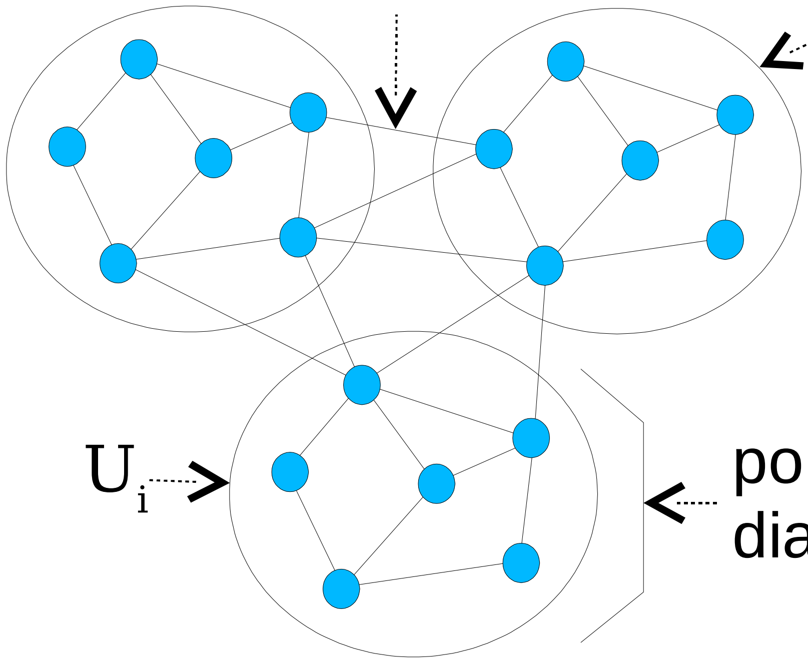
D -output of algorithm

D* =MDS(G); optimal solution

in time $O(\log_2 \log_2 n \, \log^* n \, \log^{6.5} n)$

(time = number of rounds)

Main tool: *clusters*



definition of clusters

1. $\{U_i\}_{i=1..k}$ is a partition of $V(G)$; clusters are connected subgraphs, induced by sets U_i
2. diameter of each cluster is $< \log^d n$
3. no. of connectors is $< \dfrac{1}{\log_2^c n} |E(G)|$

where $d = \dfrac{c \log_2 3}{\log_2 \dfrac{1}{1 - \frac{9}{10}\kappa}} \approx 5.54\,c$

$\kappa = \dfrac{1}{5}$

Main tool: *clusters*

definition of clusters (cont.)

(when edges have weights $\omega(e)$ *)*

$$\omega(C) < \frac{1}{\log_2^c n}\,\omega(E(G))$$

where C *is a set of connectors.*

We know how to compute clusters in polylog time:

$$\log_2 \log_2 n \, \log^* n \, \log_2^{d+1} n$$

What is n?

$$|V(G)| \leq n.$$

Vertex version:

$$|B| < \frac{6}{\log_2^c n}\,|V(G)|$$

where

B – set of border vertices

How to use clusters to approximate MDS ?

1. find clusters $\{U_i\}_{i=1..k}$
2. in each cluster U_i, in parallel, find exact solution D_i
3. return $D_1 \cup \ldots \cup D_k$

It does not work ... why?

approx ratio $\frac{|D|}{|D^*|} \leq \frac{|D^*| + |B|}{|D^*|} = 1 + \frac{|B|}{|D^*|}$

where

D - output of the algorithm

D^* - MDS(G)

B - set of border vertices

from def of clusters: $|B| \leq \frac{O(1)}{\log_2^c n} n$

approx ratio $\frac{|D|}{|D^*|} \leq 1 + \frac{1}{\log_2^c n} \frac{n}{|D^*|}$

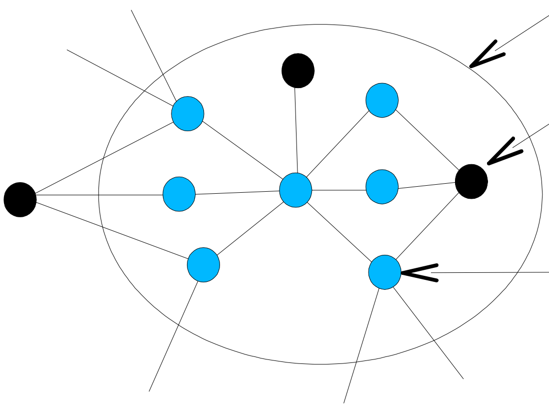
but D^* can be <u>much smaller</u> then n

How to use clusters to approximate MDS ?

explanation of inequality:

$$\frac{|D|}{|D^*|} \leq \frac{|D^*|+|B|}{|D^*|} = 1 + \frac{|B|}{|D^*|}$$

why $|D| \leq |D^*|+|B|$ holds?



because ...

$$\left|MDS(U_i)\right| \leq \left|(U_i \cap D^*) \cup B_i\right|$$

$$|D| = \sum_i \left|MDS(U_i)\right| \leq \sum_i \left|(U_i \cap D^*) \cup B_i\right| = |D^*| + |B|$$

where

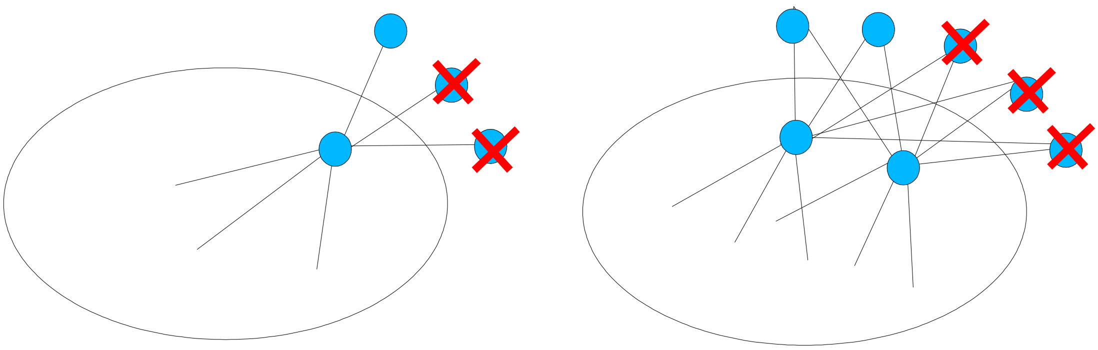
- D - output of the algorithm
- D^* - MDS(G)
- B - set of all border vertices
- B_i - set of border vertices of U_i

To use clusters we need some simple *preprocessing*

an example of simple preprocessing that works for Maximum Matching:

G -> G'

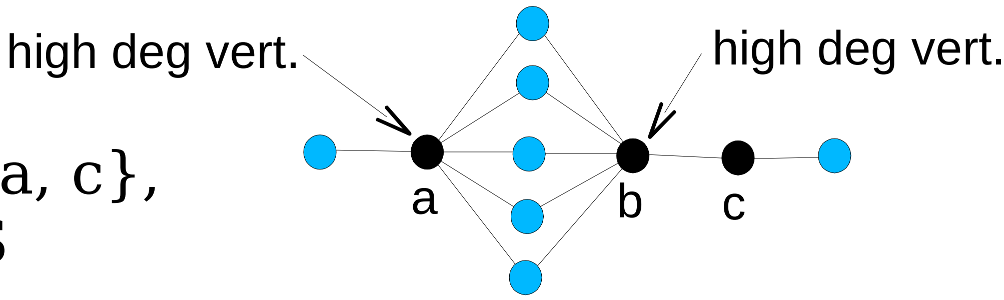
1. |MM(G')| > Ω(|V(G')|)
2. MM(G) = MM (G')



A simple preprocessing for MDS ???

- e.g. "add to the solution all vertices of high degree (at the beginning)"
- *it does not work !!!*

MDS={a, c},
b ∉ MDS



k -gadgets; MDS =2k; output =3k

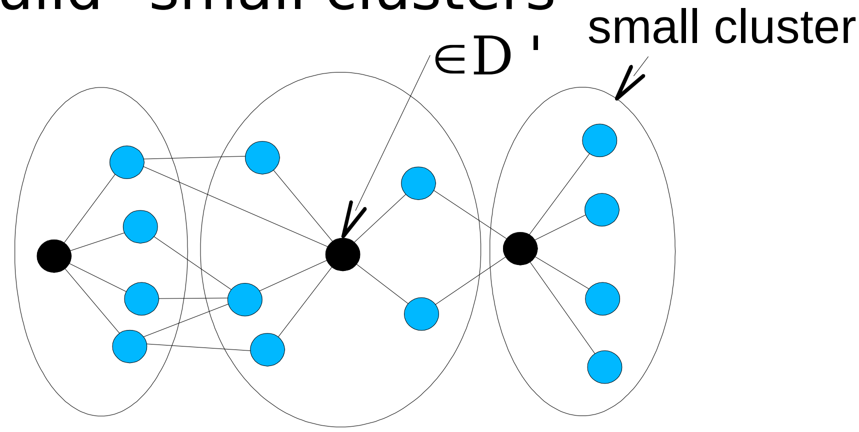
MDS: algorithm

preprocessing

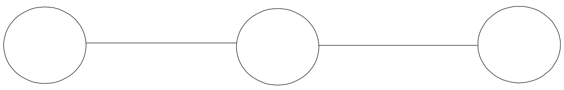
1. compute dom set D' s.t.

$$\frac{|D'|}{|D^*|} \le \log_2 n$$

2. build "small clusters"



3. contract small clusters
   to get graph G'



4. compute clusters in G' with
   $|B'| < O(1)/\log_2^{c+1} n \; |V(G')|$
5. in each <u>big</u> cluster U_i of G,
   in parallel,
   compute exact solution D_i
6. return $D_1 \cup \ldots \cup D_k$

MDS: approximation ratio

$$\frac{|D|}{|D^*|} \overset{[1]}{\leq} 1 + \frac{|B'|}{|D^*|} \overset{[2]}{\leq} 1 + \frac{O(1)}{\log_2^c n}$$

D – output of algorithm

B' – set of all small/ border clusters

D^* - MDS(G)

[2] from def. of clusters:

$$|B'| \leq \frac{O(1)}{\log_2^{c+1} n} |D'| \leq \frac{O(1)}{\log_2^c n} |D^*|$$

D' - logn aprox of MDS in G

[1] why $|D| \leq |D^*| + |B'|$???

$$|MDS(U_i)| \leq |(U_i \cap D^*) \cup C_i|$$

Sum over all clusters

$$\sum_i |MDS(U_i)| \leq \sum_i |(U_i \cap D^*) \cup C_i|$$

$$|D| \leq |D^*| + |B'|$$

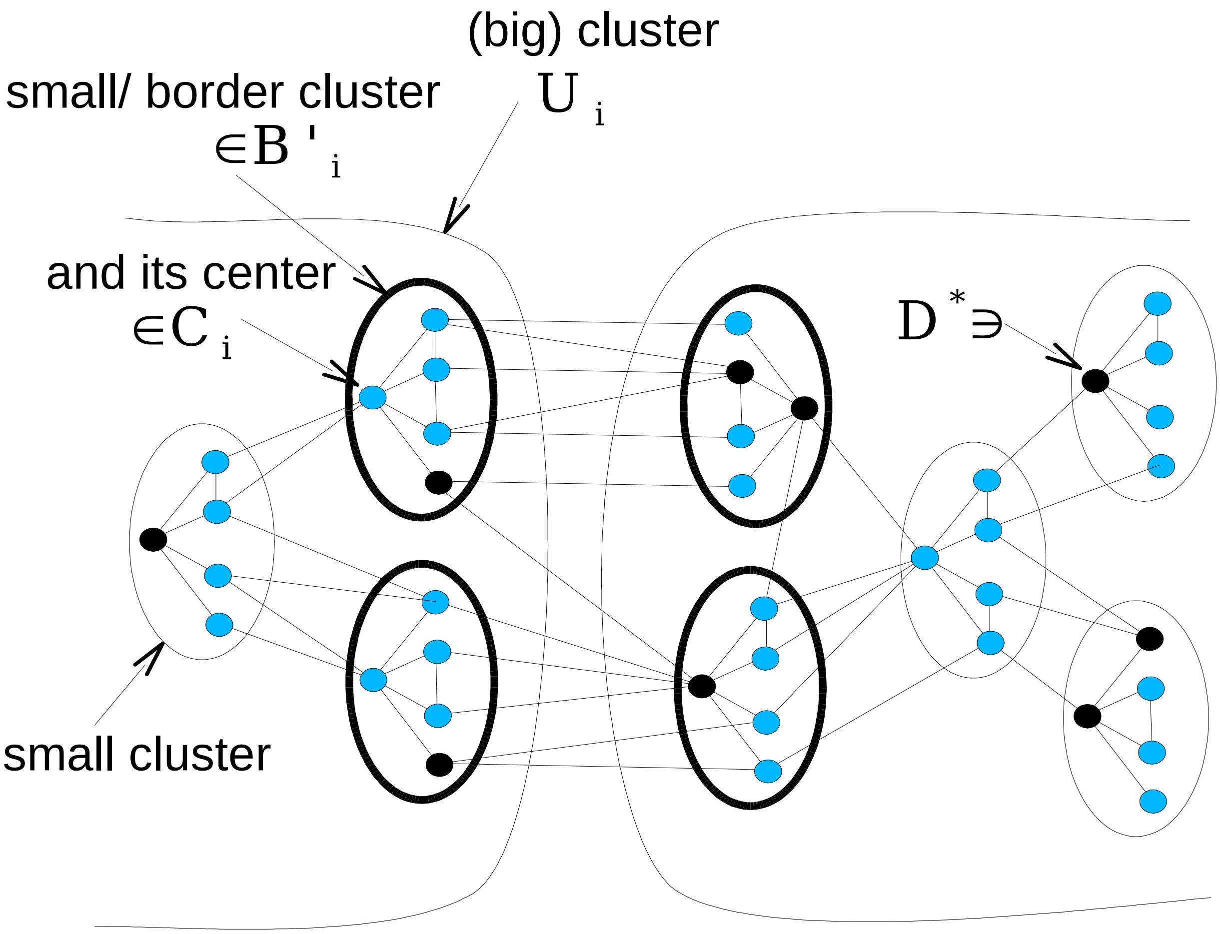
B'_i - set of small/ border clusers of U_i

C_i - set of **centers** of clusters from B'_i

$$|B'_i| = |C_i|$$

MDS: approximation ratio

$(U_i \cap D^*) \cup C_i$ dominates U_i !!!



MCDS: algorithm

1. compute dom set D' s.t.

   $$\frac{|D'|}{|MDS(G)|} \le \log_2 n$$

2. build "small clusters"
3. contract small clusters to get graph G'
4. compute clusters in G' with

   $|B'| < O(1)/\log_2^{c+1} n \; |V(G')|$

5. in each <u>big</u> cluster U_i of G,
   in parallel,
   compute exact solution D_i
   (this time MCDS ...)
6. for every two <u>big</u> clusters U_i and U_j
   that are connected,
   find shortest path P_{ij} connecting D_i
   and D_j
7. return $\bar{D} = \left(\bigcup_i D_i \right) \cup \left(\bigcup_{i,j} V(P_{ij}) \right)$

MCDS: approximation ratio

$$\frac{|\bar{D}|}{|D^*|} \overset{1}{\leq} \frac{|D|}{|D^*|} + \frac{O(1)}{\log_2^c n} \overset{2}{\leq} 1 + \frac{O(|B'|)}{|D^*|} + \frac{O(1)}{\log_2^c n} \overset{3}{\leq} 1 + \frac{O(1)}{\log_2^c n}$$

$\bar{D}$ - output of algorithm;

$$\bar{D} = D \cup \bigcup_{i,j} V(P_{ij})$$

D - sum of MCDS in all clusters

B' - set of all small/ border clusters

D^* - MCDS(G)

[3] again:

$$|B'| \leq \frac{O(1)}{\log_2^{c+1} n} |D'| \leq \frac{O(1)}{\log_2^c n} |MDS(G)| \leq \frac{O(1)}{\log_2^c n} |D^*|$$

[1] $\left| \bigcup_{i,j} V(P_{ij}) \right| \leq O(|B'|) \leq \frac{O(1)}{\log_2^c n} |D^*|$

because paths P_{ij} have length <=3,
and their no. is <=O(|B'|)

MCDS: approximation ratio

[2] why $|D| \leq |D^*| + O(|B'|)$???

1. no. of conn. comp. in $U_i \cap D^*$ may by large
2. no. of conn. comp. in $(U_i \cap D^*) \cup C_i$ is $\leq |B'_i| = |C_i|$, where
   B'_i - set of small/ border clusers of U_i
   C_i - set of **centers** of clusters from B'_i
3. if k is a no. of conn. components in $(U_i \cap D^*) \cup C_i$ then we can connect these comp. adding vertices from $\tilde{C}_i \subset U_i$,
   where $|\tilde{C}_i| \leq O(k)$!!!

therefore:

$(U_i \cap D^*) \cup C_i \cup \tilde{C}_i$ is a CDS in U_i

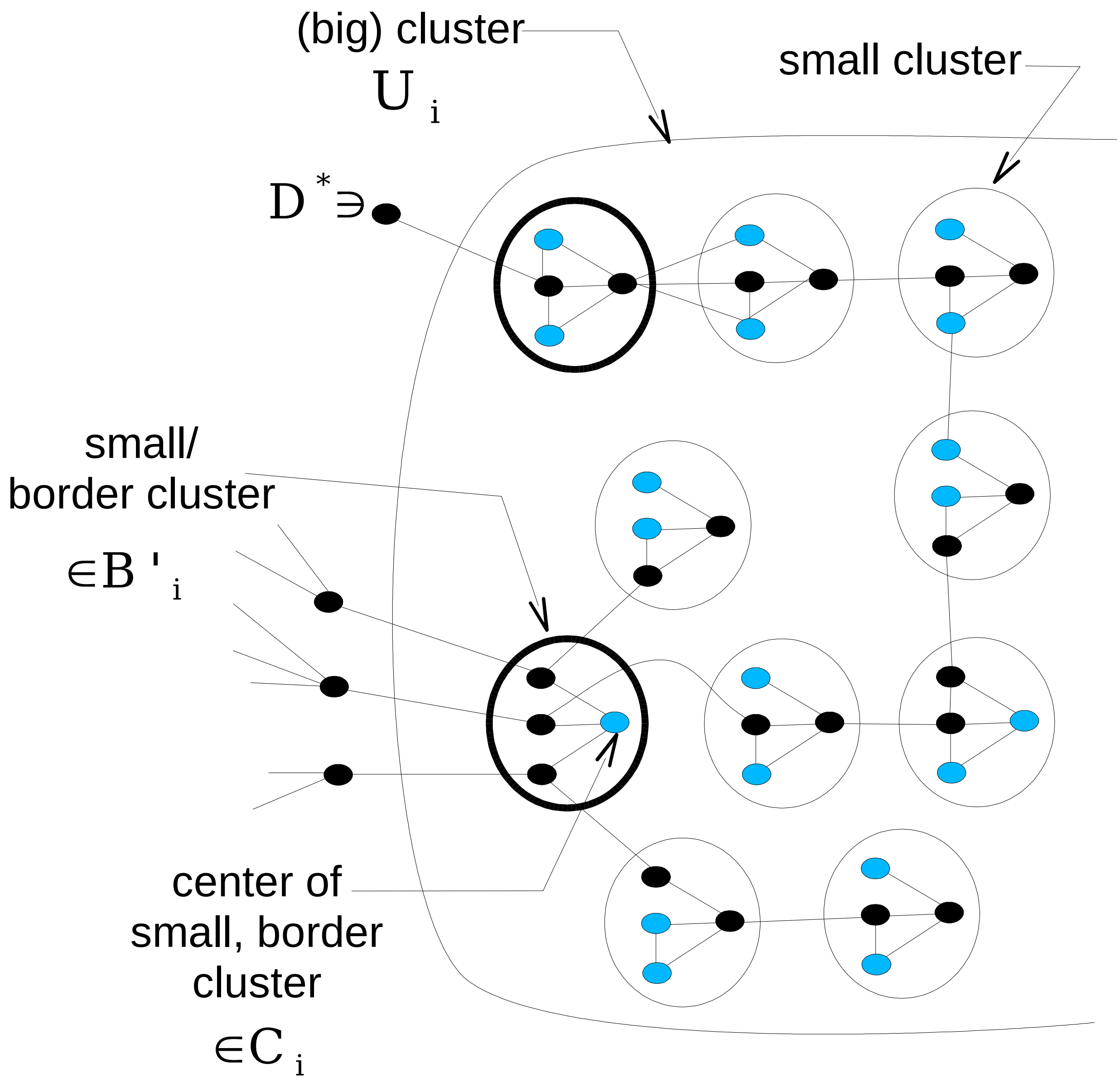
and:

$$\sum_i |MCDS(U_i)| \leq \sum_i |(U_i \cap D^*) \cup C_i \cup \tilde{C}_i|$$

$$|D| \leq |D^*| + O(|B'|)$$

B' - set of all small/ border clusers

MCDS: approximation ratio



log n -aprox of MDS

GreedyDS

1. D:= ø; V_1:=V; V_2:= ø
2. for i:=0 to logn-4 do
   a) B:= { v: $deg_1(v) > n/2^{i+1}$ }
   b) vertices from V_1 dominated by B jump form V_1 to V_2
   c) D:=D ∪ B; delete vertices form B
3. D:= D ∪ V_1; return D

Why this algorithm computes logn approx of MDS in planar G ???

Let D*= MDS(G)
we can prove: $|B^i| < O(|D^*|)$
and sum over all logn iterations ...

Why $\Omega(|B^i|) < |D^*|$?

Generalization to proper minor-closed families of graphs

???

Open problem: MWDS (Weighted Dom Set)

???
???
???