Distributed approximation algorithms in unit-disk graphs

Andrzej Czygrinow
Department of Mathematics & Statistics
Arizona State University

Joint work with Michal Hańćkowiak (Adam Mickiewicz University)

Assumptions

- Message-passing model.
- Synchronized.
- Unique identifiers of nodes.

Goal

- Deterministic algorithms.
- Time (the number of rounds) poly-logarithmic in the order of a graph.

Algorithms

- **Distributed Approximations**

  - Given $0 < \epsilon < 1$, algorithm finds in a poly-logarithmic number of rounds a solution of value within

  $$(1 \pm O(\epsilon))OPT.$$

  - Given a positive integer k, algorithm finds in a poly-logarithmic number of rounds a solution of value within

  $$(1 \pm O(1/\ln^k |G|))OPT.$$

- **Unit-disk graphs**

  A graph $G = (V, E)$ is called a unit-disk graph if there is a function $f : V \to R^2$ such that $uv \in E$ if and only if

  $$||f(u) - f(v)||_2 \leq 1.$$

Problems

- The maximum matching problem (MM).
  $M \subseteq E(G)$ is a matching if edges from M have distinct endpoints.

  - *maximal* - maximal with respect to $\subseteq$.
  - *maximum* - of maximum size.

- The minimum connected dominating set problem (MCD).

  - C is a *dominating set* if $N(C) \cup C = V(G)$.
  - C is a *connected dominating set* if in addition $G[C]$ is a connected subgraph of G.

Results

Theorem 1 *Let k be a positive integer. There is a distributed algorithm which finds in a unit-disk graph G a matching M such that*

$$|M| \geq \left(1 - O(1/\log^k |G|)\right) \beta$$

where β is the size of a maximum matching in G. The number of rounds of the algorithm is poly-logarithmic in $|G|$.

Theorem 2 *Let k be a positive integer. There is a distributed algorithm which finds in a connected unit-disk graph G a dominating set D such that $G[D]$ is connected and*

$$|D| \leq \left(1 + O(1/\log^k |G|)\right) \gamma_c$$

where γ_c is the size of the minimum connected dominating set in G. The number of rounds of the algorithm is poly-logarithmic in $|G|$.

Comments

- **Important:** No node has any knowledge about geometrical representation of graph G.

- **Not Important:** More careful results that analyze local complexity can be obtained.

Related Work

- **Starting Point:** T. Nieberg, J. L. Hurink (2005)
  Sequential PTAS for minimum dominating set in UDG.

- **Distributed algorithms:** F. Kuhn, T. Moscibroda, T. Nieberg, R. Wattenhofer (2005)
  Maximal independent set, minimum dominating set, maximum independent set in UDG.

- **Related work for UDG:** Many const-approximations for MCD (different models, often reductions from MIS).

- **Related distributed algorithms:** AC , M. Hanckowiak (2006)
  Minor-closed families.

General strategy of algorithms

(1) Find a (special) clustering of G.

(2) In each cluster find an optimal solution to the problem.

(3) Return union of these solutions.

Clustering

(A) Find an auxiliary partition. This yields a reduced graph $Aux(G)$.

(B) Apply main clustering procedure to $Aux(G)$ and return with partition to G.

Tools

Independent Set:

A subset $I \subseteq V$ such that no edge has both endpoints in I is called an *independent set*.
MIS - maximal with respect to $\subseteq$.

Lemma 3 *Let G be a unit-disk graph and let k be a positive integer. For any independent set I in G and any geometrical representation of G, the number of vertices from I which are contained in a ball in R^2 of radius k is $O(k^2)$.*

Lemma 4 (Kuhn et. al. 2005) *There is a distributed $polylog|G|$-time algorithm which finds in UDG a maximal independent set.*

Tools

Ruling Set:

A *D-ruling set* in G is a subset S of $V(G)$ with two properties:

- For any two distinct vertices s, s' from S, the distance (in G) between s and s' is at least D.

- For any vertex $v \in V(G)$ there is a vertex $s \in S$ such that the distance between s and v is at most $D \log |G|$.

Lemma 5 (Awerbuch et. al. 1985) *There is a deterministic $polylog|G|$-time algorithm which finds a D-ruling set in a graph G for any $D = O(polylog|G|)$.*

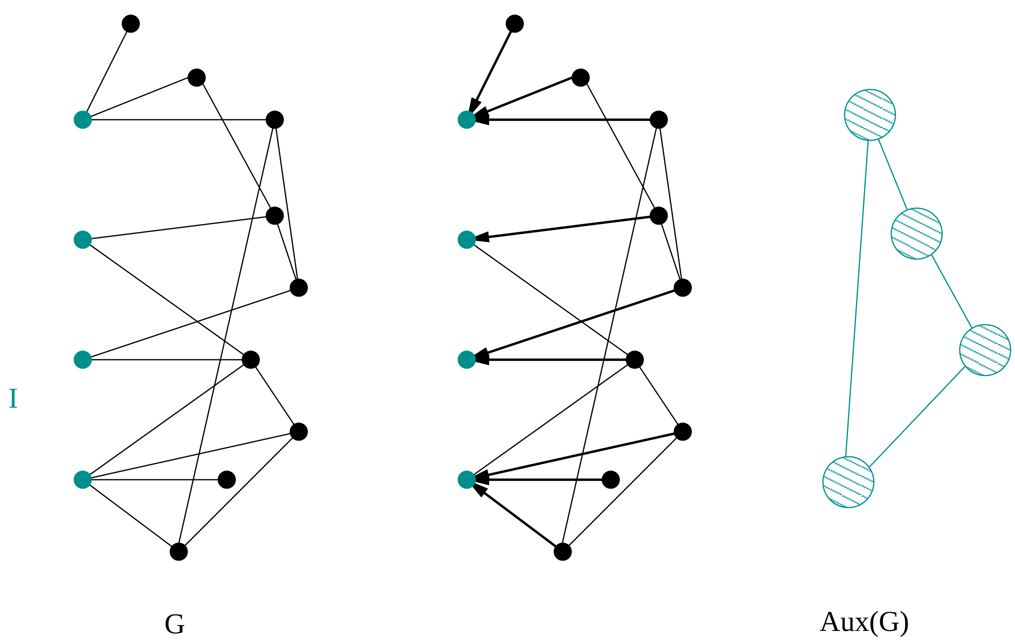
Bounded growth

Definition 1 *A graph H has a $f(k)$-bounded growth if for every vertex v from H and every nonnegative integer k the number of vertices within distance k of v is at most $f(k)$.*

Clustering Phase 1

1. Find MIS I in UDG G.

2. Reduce G to $Aux(G)$. Note that $|Aux(G)| = |I| = O(\gamma(G)) = O(\gamma_c(G))$.



Fact 6 *$Aux(G)$ has quadratic growth ($f(k) = O(k^2)$).*

Clustering Phase 2

1. Find a D-ruling set $S = \{s_1, \ldots, s_k\}$ in $Aux(G)$ ($D = O(1/\epsilon^2)$, ϵ - approximation error).

2. Build subgraphs $N_l(s_i)$ as long as $|N_l(s_i)| > (1 + \epsilon)|N_{l-1}(s_i)|$.

3. Return $N_{l_1}(s_1), \ldots, N_{l_k}(s_k)$ as clusters.

4. Repeat previous steps a $polylog|Aux(G)|$ number of times.

5. Create one-element clusters from what is left.

Analysis

- $|S|$ is relatively large ($|S| \geq |Aux(G)|/(CD^2 \log^2 |Aux(G)|+1)$).

- Radius of each $N_l(s_i)$ is small (radius $\leq l_* = O(1/\epsilon^2)$).

- Number of edges leaving the clusters is small ($C\epsilon \cdot \sum |N_{l_i}(s_i)|$).

- There is a small number one-element clusters.

Lemma 7 *If you set parameters right then diameter of each cluster is* $polylog|Aux(G)|$, *time is* $polylog|Aux(G)|$, *and the number of edges of* $Aux(G)$ *connecting different clusters is* $O(\epsilon|Aux(G)|)$.

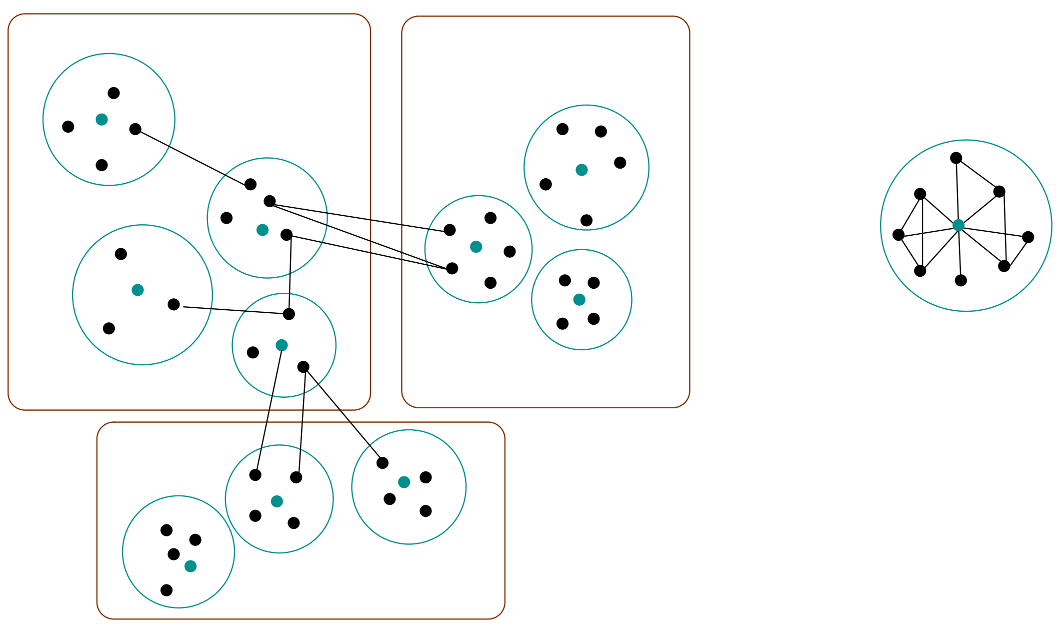
Going back to G

Three types of clusters:

- *Small clusters.* Clusters in graph G which arise from MIS.

- *Auxiliary clusters.* Clusters in $Aux(G)$ obtained by Clustering.

- *Big clusters.* Clusters in G obtained from clusters of $Aux(G)$.



Matchings

1. In each cluster C of G find a maximum matching M_C.

2. Return the union of these matchings.

Why does it work?

- Every small cluster which contains many vertices is a very dense graph - local matching saturates all but at most a constant number of vertices from each small cluster.

- $Aux(G)$ has a constant max degree. The number of small clusters on the border is small with respect to $|Aux(G)|$ and so is small the respect to the size of maximum matching in $|Aux(G)|$ and so small with OPT in G.

- How good is the returned matching M?

$$|M| \geq \sum_{C} |M_C| \geq OPT - O(\epsilon |Aux(G)|)$$

$$= (1 - O(\epsilon))|Aux(G)|.$$

Connected Dominating Set

1. In each cluster C of G find a minimum connected dominating set D_C.

2. For every edge $\{C', C\}$ of $Aux(G)$ join D_C with D'_C by a path of length at most 3 in G.

3. Return the union.

Why does it work?

- Problem: A connected dominating set in G does not need to be connected in $G[C]$.

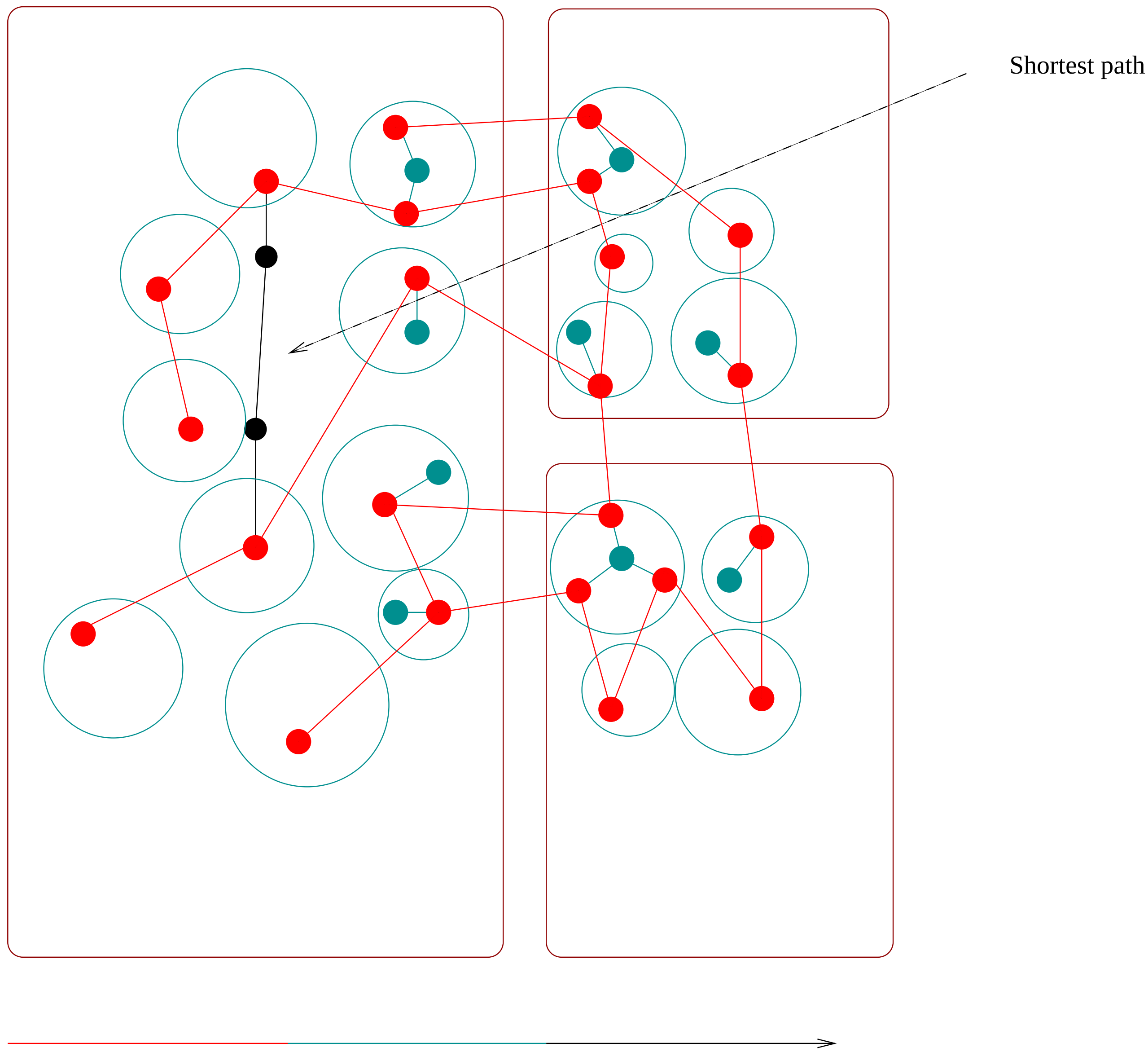
- If it is disconnected then components of $G[C]$ contain border vertices of C.

- The number of components in each $G[C]$ can be large.

- After we add all of the centers of border clusters the number of components is small.

- We can connect these components using a small number of short paths.

- Summary: From an optimal connected dominating set D^* in G we can obtain a new connected dominating set D' such that $|D'| - |D^*| = O(s)$ (s is the number of small border clusters) and each $D' \cap C$ induces a connected subgraph.


Shortest path

Conclusions

Distributed approximations in message passing model for the MM Problem and the MCD Problem in UDG.