Constructing *K*-Connected *M*-Dominating Sets in Wireless Sensor Networks

Yiwei Wu, Feng Wang, My T. Thai and Yingshu Li

Georgia State University
Arizona State University
University of Florida

Outline

- Introduction
- Centralized Algorithm – *CGA*
- Distributed Algorithm – *DDA*
- Simulation Results
- Conclusion

Introduction

- A *Connected Dominating Set (CDS) C* of *G* is a dominating set of *G* which induces a connected subgraph of *G*. The nodes in *C* are called *dominators*, the others are called *dominatees*.
- A *CDS* is the earliest proposed candidate to serve as a virtual backbones in WSNs.

Introduction

- Using this virtual backbone, a sender can send messages to its neighbouring dominator. Then along the *CDS*, the messages are sent to the dominator closest to the receiver. Finally, the messages are delivered to the receiver.
- *k*-Connected *m*-Dominating Set (*kmCDS*) is necessary for fault tolerance and routing flexibility.

Introduction

- k-connected or k-vertex connected
  - A graph is k-connected if and only if it contains k independent paths between any two vertices.
- m-dominating set
  - If each node not in C is dominated by at least m nodes in C, then C is a m-dominating set.
- k-connected m-dominating set

Outline

- Introduction
- Centralized Algorithm – *CGA*
- Distributed Algorithm – *DDA*
- Simulation Results
- Conclusion

Centralized Algorithm CGA

- *CGA* is a greedy algorithm with time complexity of *CGA* is $O(|V|^{3.5}|E|)$.
- The main idea:
  - construct an *m*-dominating set *C*
  - augment this set *C* for *k*-connectivity
  - remove redundant nodes (optional)

Centralized Algorithm CGA

- Notations:
  - N_i: The number of neighbors of a node i.
  - e_i: The energy of a node i.
  - N^c_i : The number of dominator neighbors of node i.
  - ID_i: The ID of node i.
  - C: A k-connected m-dominating set.
- Weight function w(N, e, ID)

Centralized Algorithm CGA

Algorithm 1 CGA$(k, m, G(V, E))$

procedure FINDKMCDS$(k, m, G(V, E))$
    Sort nodes in non-increasing order in G based on their (N_i, e_i, ID_i)
    $C \leftarrow \phi$
    for $i = 1$ to $|V|$ **do**
        if $N_i^c < m$ **then**
            $C \leftarrow C \cup \{v_i\}$
        end if
    end for
    while C is not k connected **do**
        add a dominatee with highest (N_i, e_i, ID_i) into C
    end while
    return C
end procedure

Outline

- Introduction
- Centralized Algorithm – *CGA*
- Distributed Algorithm – *DDA*
- Simulation Results
- Conclusion

Distributed Algorithm - DDA

- *DDA* consists of three phases
  - Phase 1: Use one of the distributed *CDS* algorithms to construct a *CDS C*.
  - Phase 2: Augment *C* to a 1-connected *m*-dominating set by adding *m*-1 *MISs*.
  - Phase 3: Connect set *C* for *k*-connectivity.

Distributed Algorithm – DDA phase 3

- How to make *C* for *k*-connectivity in a distributed way?
- *Lemma 1:* If *G* is a *k*-connected graph, and *G'* is obtained from *G* by adding a new node *v* with at least *k* neighbors in *G*, then *G'* is also a *k*-connected graph.

Distributed Algorithm – DDA phase 3

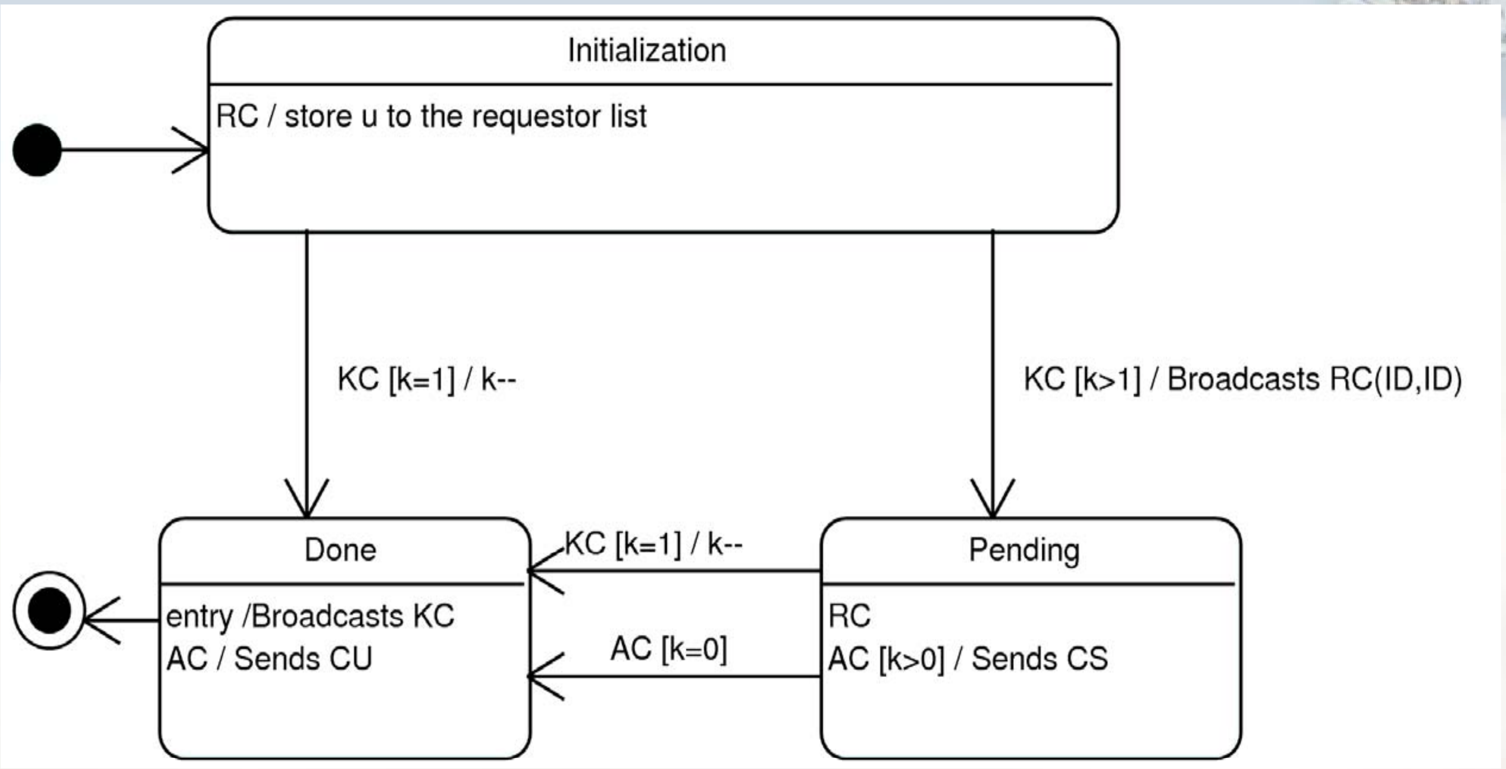
- Main idea:
  - The leader builds a k-connected component.
  - In order to join the k-connected component, one black node adds at most k^2 connectors according to the previous lemma.

Distributed Algorithm – DDA phase 3

- Important messages used in *DDA:*
  - *K-ConnectedComponent (KC) message*
  - *RequireConnector (RC) message*
  - *ACKConnected (AC) message*
  - *ConfirmSuccess (CS) message*
  - *ConfirmUnuse (CU) message*

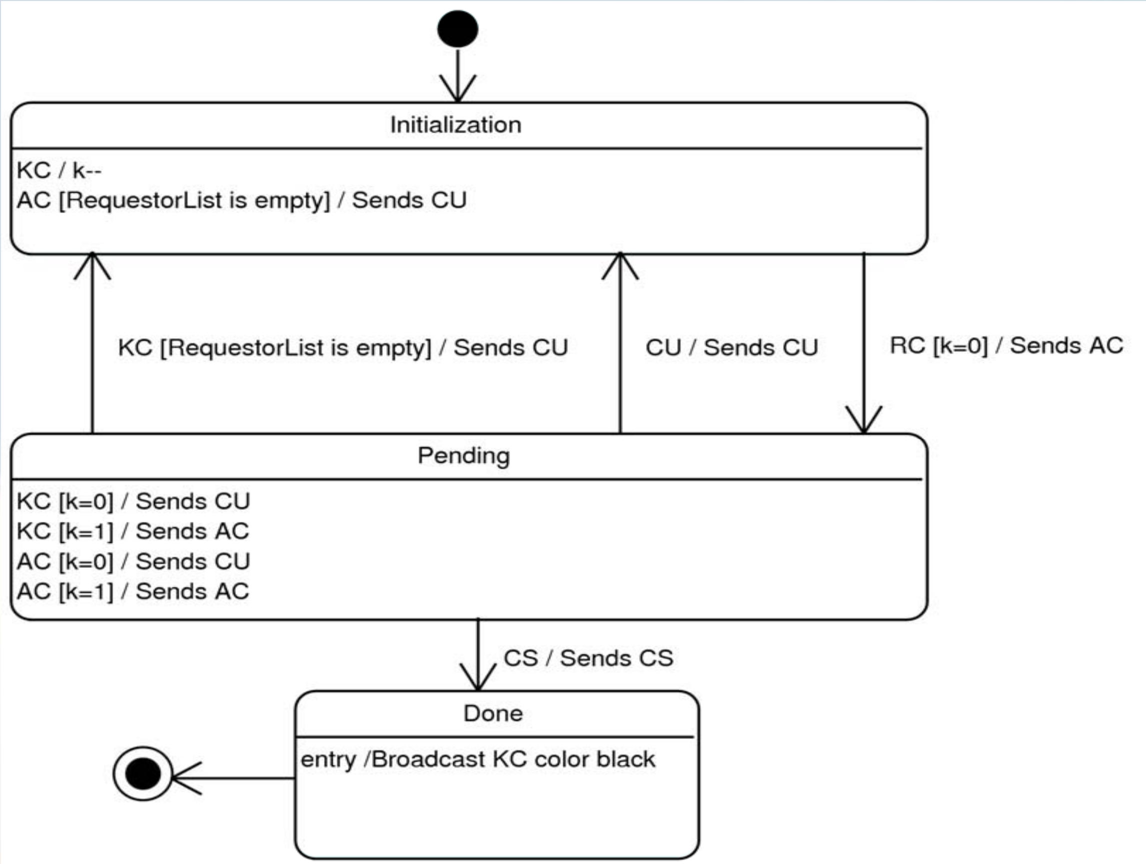
Distributed Algorithm – DDA phase 3

- State transition diagram for black nodes



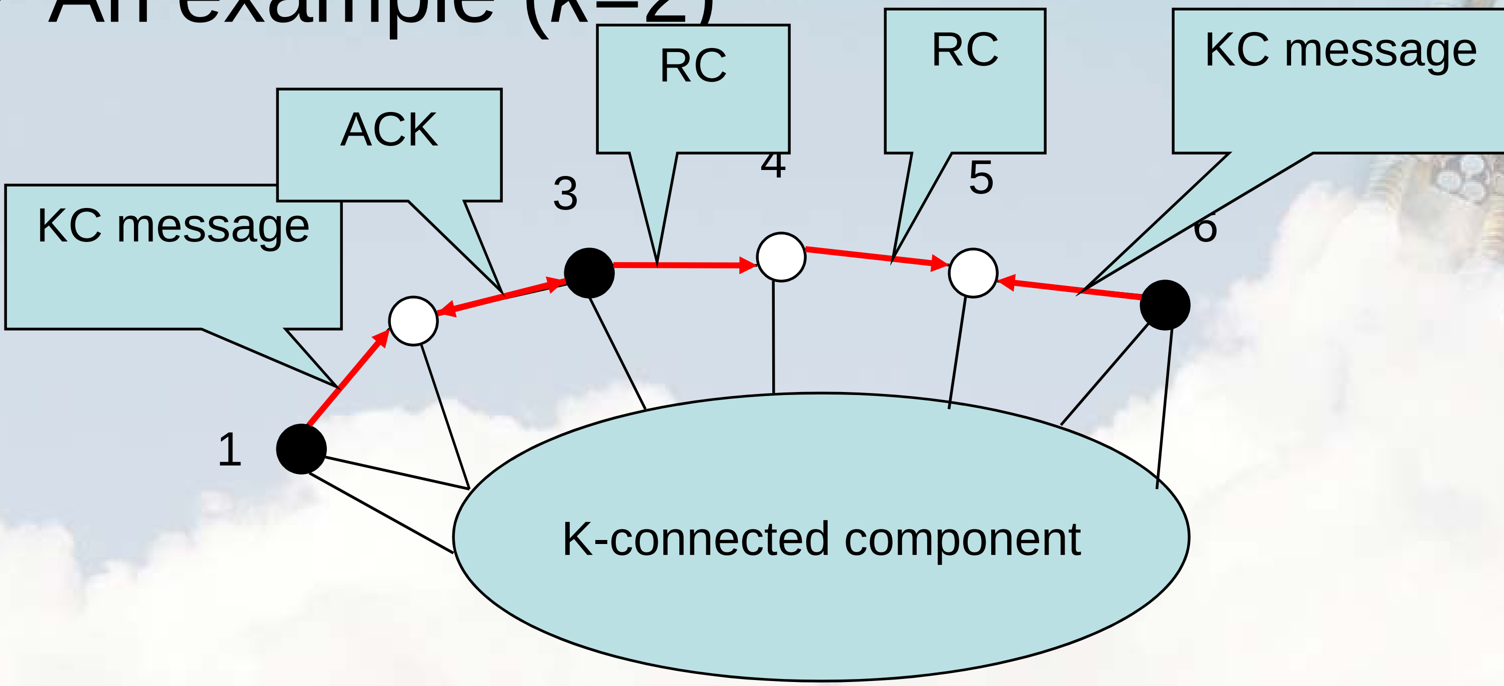
Distributed Algorithm – DDA phase 3

- State transition diagram for white nodes



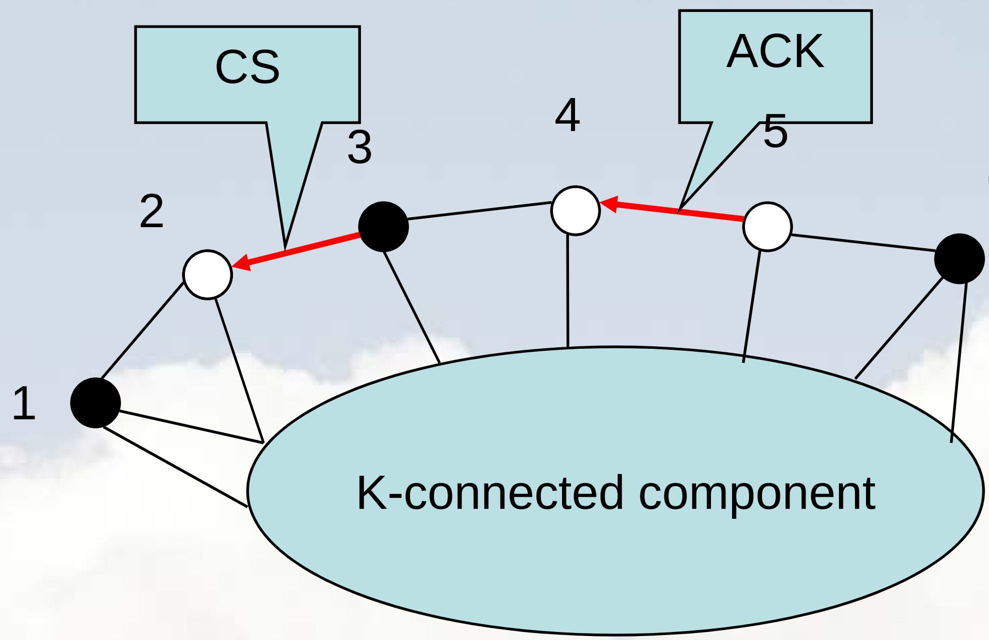
Distributed Algorithm – DDA

- An example (k=2)



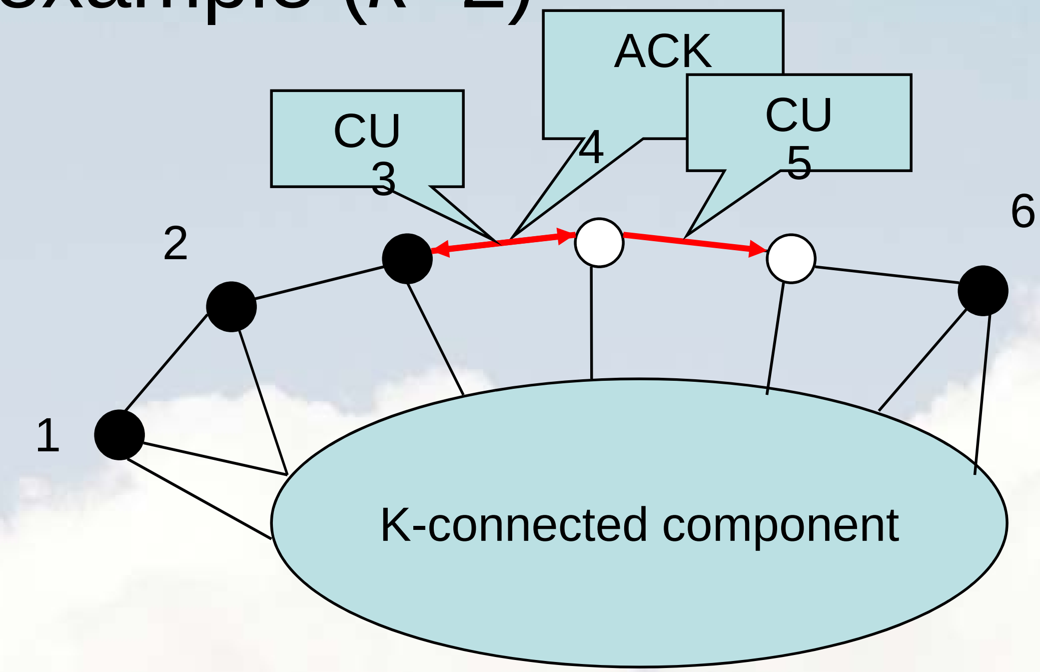
Distributed Algorithm – DDA

- An example (*k*=2)



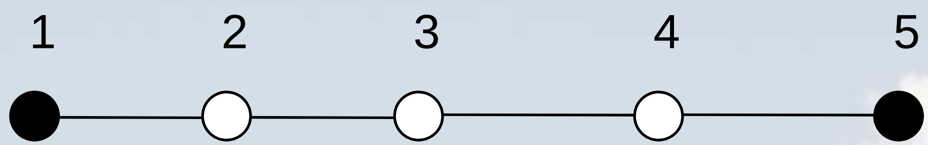
Distributed Algorithm – DDA

- An example (*k*=2)



Distributed Algorithm – DDA

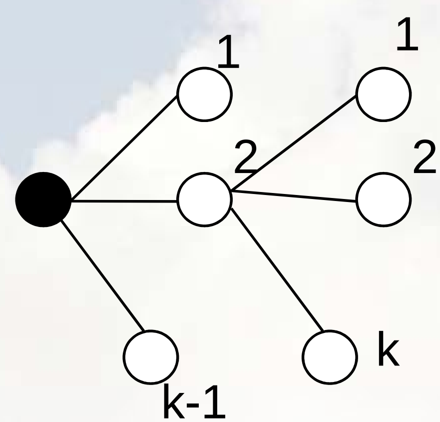
- *Lemma 2*: Every subset of an *MIS* is at most three hops away from its complement.



- *Lemma 3*: Let $G = (V,E)$ be any *UDG* and m be any constant such that $\delta_G \geq m - 1$ where δ_G is the minimum node degree of graph G. Let D^*_m be any optimal m-domination of G and S be any MIS of G. Then $|S| \leq 5m|D^*_m|$.

Distributed Algorithm – DDA

- *Theorem*: If C is a *kmCDS* obtained by *DDA*, then $|C| \le 5m(k^2+1)(m+42)opt$, where *opt* is the size of any optimal *kmCDS* of the network.



- Proof:
  - Phase 1: $43|S|$
  - Phase 2: $(m-1)|S|$
  - Phase 3: k^2 for each black node in phase 1 and 2

Distributed Algorithm – DDA

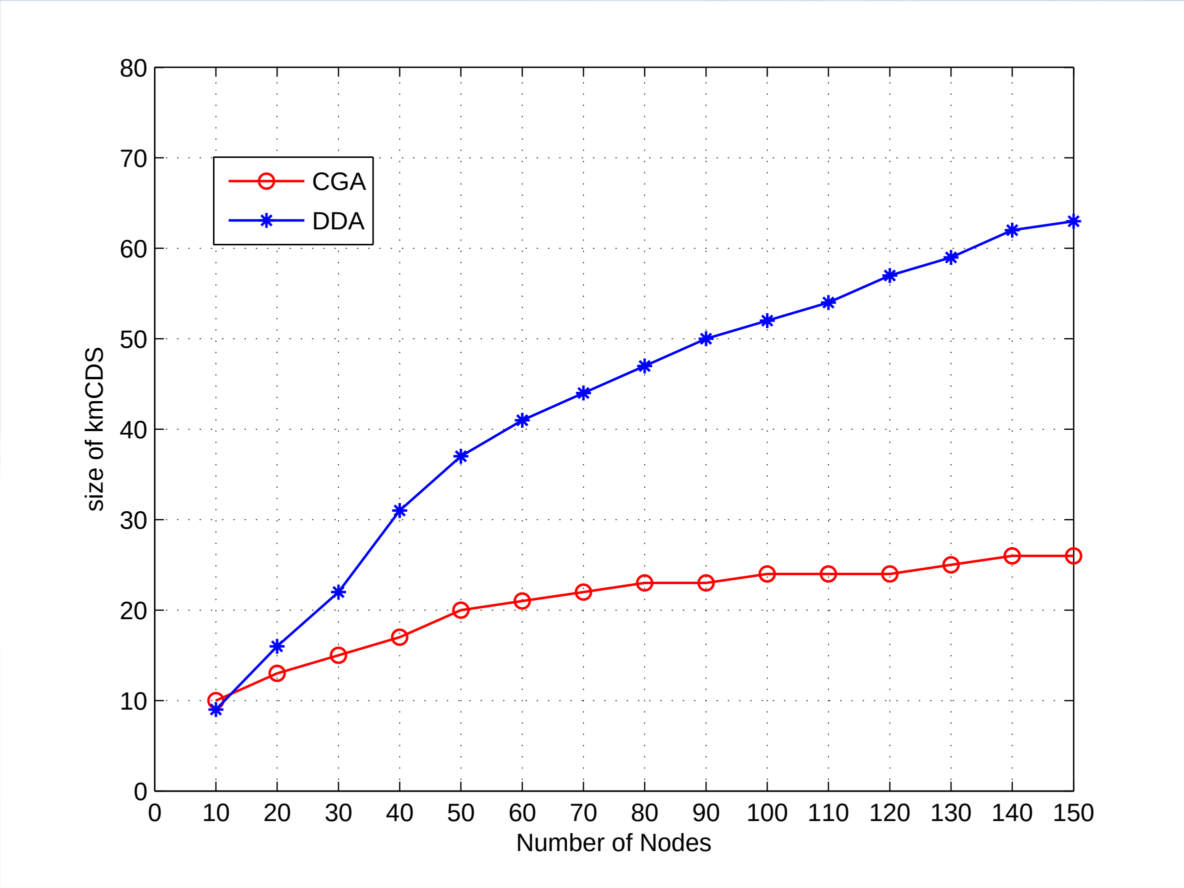
- The message complexity of *DDA* is $O(|V|\Delta^2)$ and time complexity is $O(m\Delta + Diam)$, where Δ is the maximum node degree and *Diam* is the diameter of the network.

Outline

- Introduction
- Centralized Algorithm – *CGA*
- Distributed Algorithm – *DDA*
- Simulation Results
- Conclusion

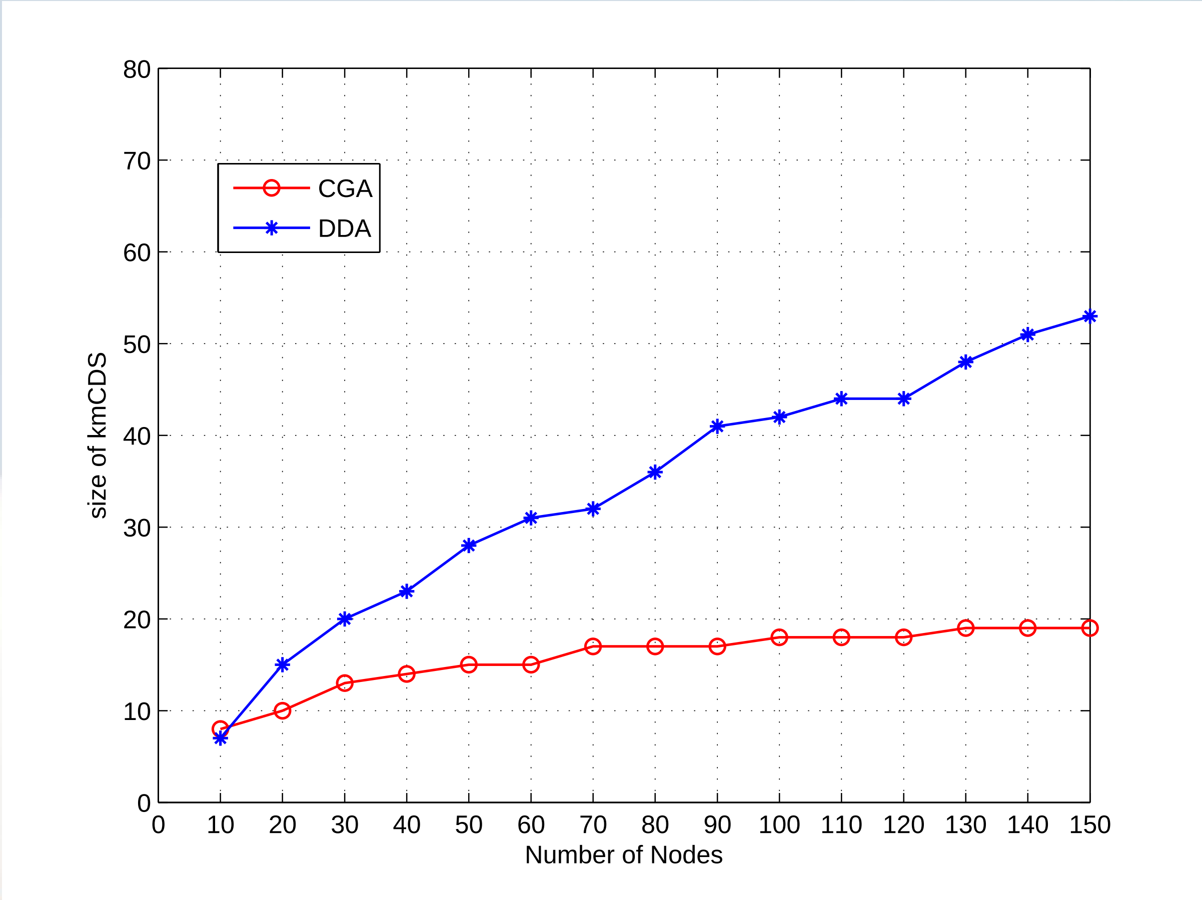
Simulation and Results

- 1000*m* $\times$ 1000*m area;* transmission rage = 250m



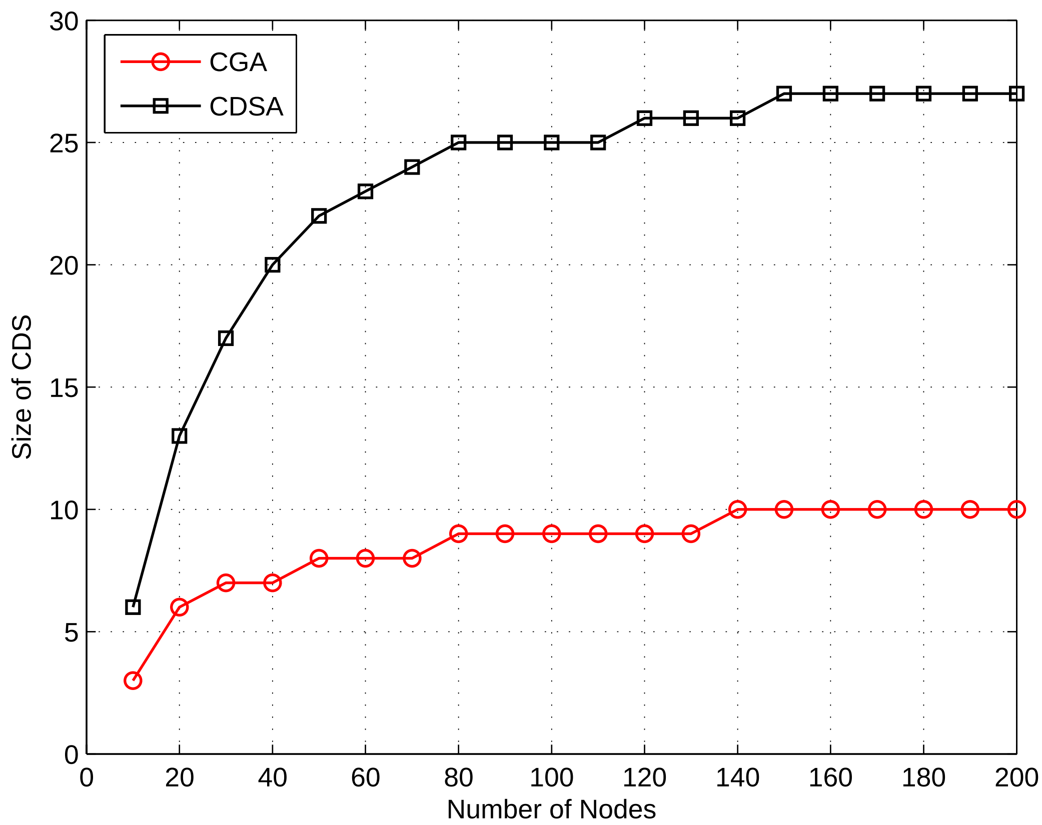
Simulation and Results

- $1000m \times 1000m$, transmission rage = 350m



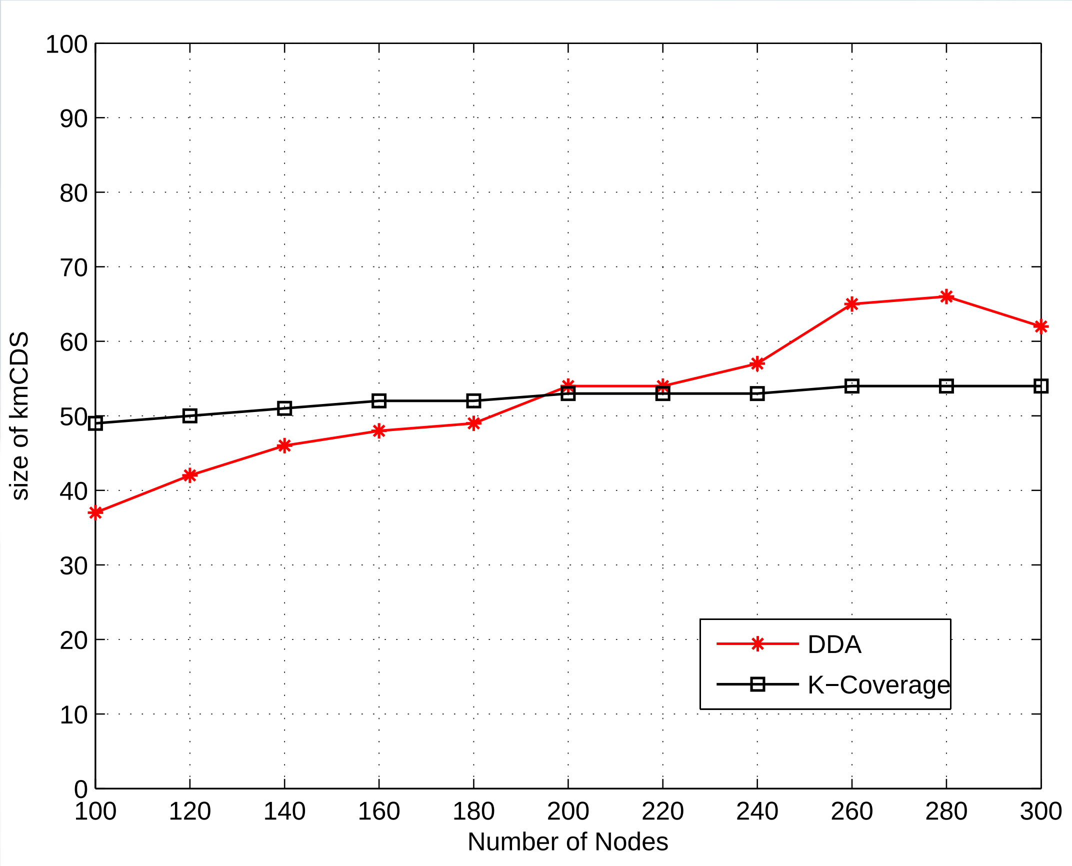
Simulation and Results

- Compare *CGA* with *CDSA* when $k = 2, m = 1$



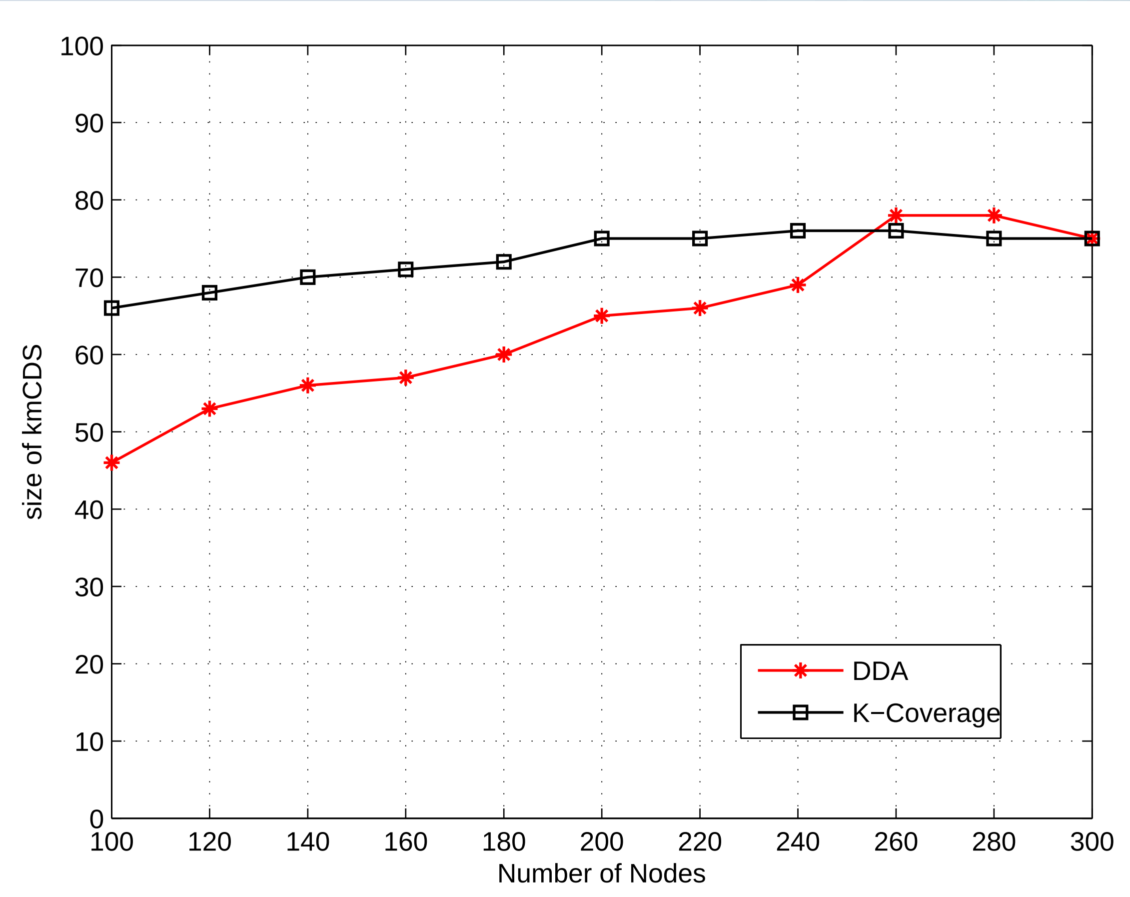
Simulation and Results

- Comparision of *DDA* with *k-Coverage* (*k*=2).



Simulation and Results

- Comparision of *DDA* with *k-Coverage* (*k*=3).



Outline

- Introduction
- Centralized Algorithm – *CGA*
- Distributed Algorithm – *DDA*
- Simulation Results
- Conclusion

Conclusion

- We investigate the problem of constructing a *km*CDS for general *k* and *m*.
- We propose one centralized algorithm *CGA* and one distributed algorithm *DDA*.
- Our algorithms can obtain good results.

Questions?

Thanks!