

# A clustering algorithm for Large-Scale and Multi-hop Sensor Networks

Natale Guzzo, Traxens SAS, Inria

Nathalie Mitton, Inria

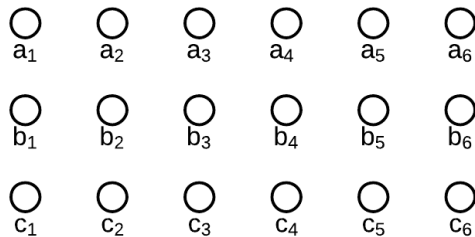
Pascal Daragon, Traxens SAS

Arulnambi Nandagoban, Traxens SAS

# Summary

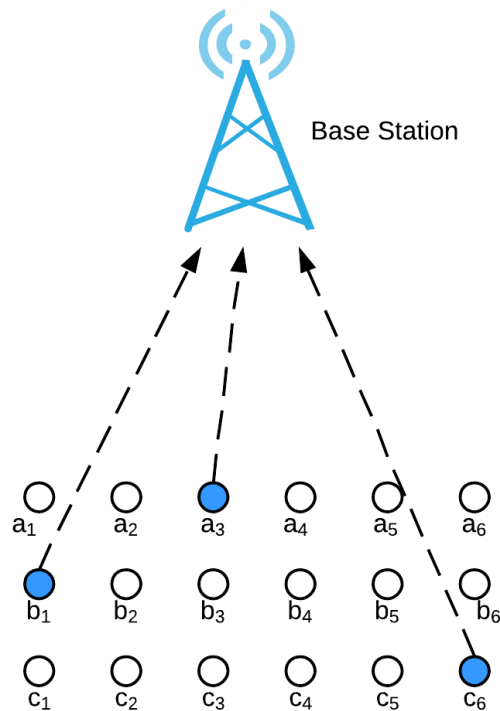
- ▶ Problem Statement
- ▶ CESAR Algorithm
- ▶ HEAD election and resignation
- ▶ Election metric
- ▶ Cluster formation and destruction
- ▶ HEAD competition
- ▶ On-demand scheme
- ▶ Recovering scheme
- ▶ Data aggregation
- ▶ Simulation results
- ▶ Use cases
- ▶ Conclusions

# Problem statement



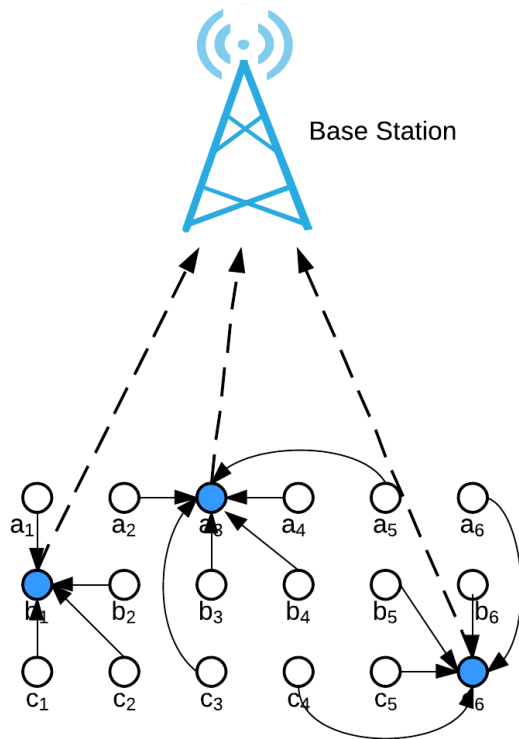
- ▶ The network nodes are equipped with:
  - A long radio range module to communicate with the Base Station (BS-link).
  - a short-range radio module for the communication peer-to-peer between the sensor nodes (P2P-link).

# Problem statement



- ▶ The nodes do not know the position of the BS and they are supposed to communicate directly with it only whether the quality of the BS-link is good enough.
- ▶ As a result, only some of them may be able to connect to the BS.

# Problem statement



- ▶ The nodes unable to connect to the BS use the P2P-link to deliver their data to the nodes able to transmit to the BS.
- ▶ A routing algorithm is needed to build the communication paths between the network nodes.

# CESAR Algorithm

- ▶ The Cluster-based Energy Saving Affiliation Routing protocol (CESAR) is a multi-hop and energy-efficient algorithm designed for low data rate and delay-tolerant applications in large-scale Sensor Networks.
- ▶ CESAR aims to reduce the energy consumption and maximize the robustness of the network by introducing innovative mechanisms for the cluster-head election and the formation of the clusters.

# CESAR Algorithm

- ▶ **Node types:**
  - ▶ **HEAD:** nodes able to communicate with the BS and in charge of clustering the network.
  - ▶ **MEMBER:** nodes which have joined a cluster.
  - ▶ **LOOSE:** nodes which have not joined any cluster.
- ▶ **Routing messages:**
  - ▶ **RANN:** hop-bounded broadcast messages used by HEAD nodes to build their clusters.
  - ▶ **ROFF:** broadcast messages used by HEAD nodes to destroy their clusters.
  - ▶ **DATA:** messages sent in unicast by MEMBER nodes to deliver their data to the cluster-head.

# HEAD election

- ▶ **Main aspects:**
  - ▶ Distributed
  - ▶ Periodic
  - ▶ No synchronization required



# HEAD election

- ▶ Election parameters:
  - ▶ Battery level
  - ▶ BS signal strength
- ▶ A node can become HEAD only whether the battery level and the signal strength on the BS-link are greater than the election thresholds

# HEAD Resignation

- ▶ When a HEAD node no longer meets the HEAD requirements because the battery level or BS signal strength go under the election thresholds, it resigns and becomes LOOSE.

# Election metric

- ▶ At every election round each HEAD node computes its election metric (e-metric) which includes the designed election costs:

$$e\text{-metric} = K_1 * GPRS\ cost + K_2 * Battery\ cost$$

$$Link\ cost = \left\lfloor 10 * \left(1 - \frac{BS\ signal\ strength}{Max\ signal\ strength}\right) \right\rfloor$$

$$Battery\ cost = \left\lfloor 10 * \left(1 - \frac{Remaining\ energy}{Initial\ energy}\right) \right\rfloor$$

# Cluster creation

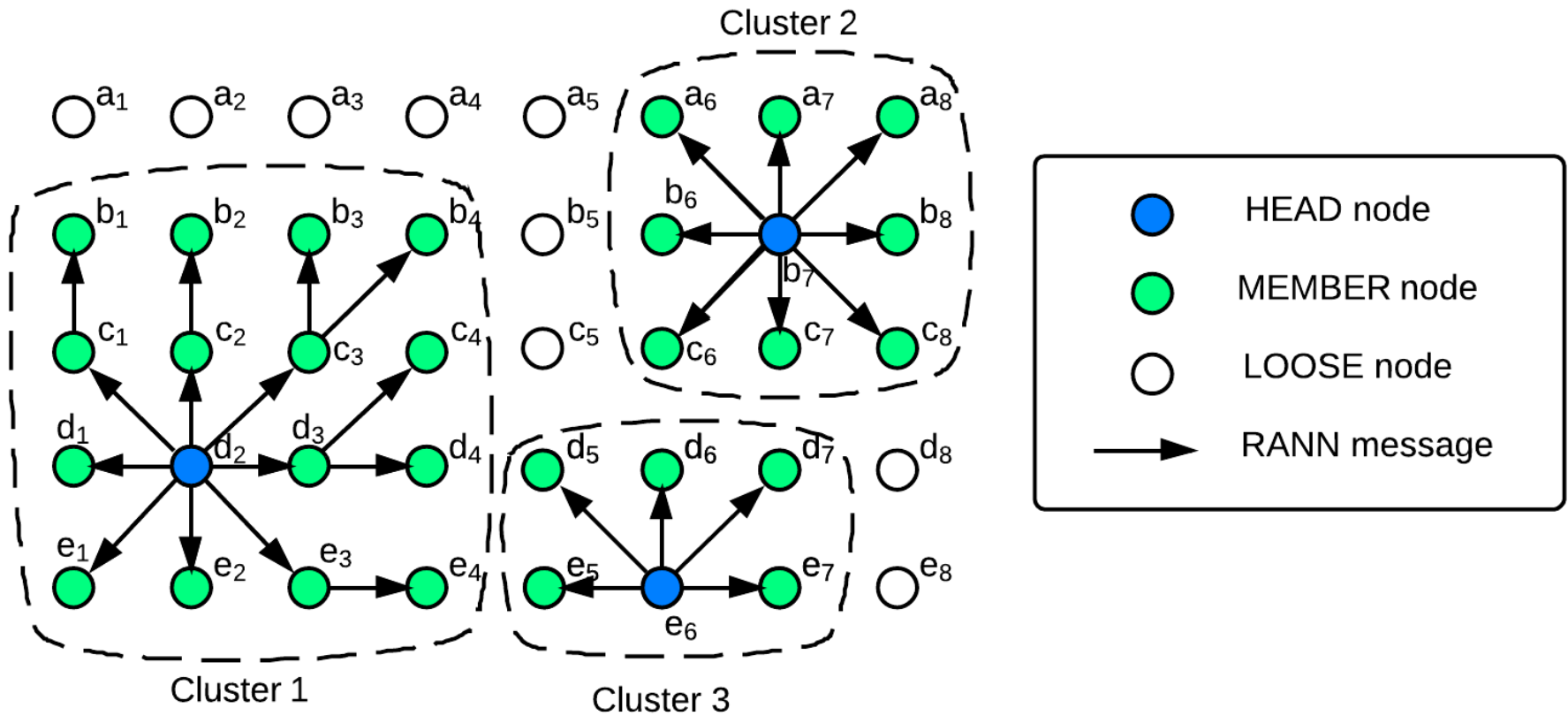
- ▶ Some scenarios consist of large-scale sensor networks in which the topology often changes because of the mobility of the nodes or because of the variations in the radio environment.
- ▶ For such applications, rather than performing the full clustering of the network, CESAR designs a partial clustering approach that forms clusters of limited size

# Cluster creation

- ▶ The clusters are formed by flooding hop-bounded RANN messages containing the sender address and the election parameters of the HEAD nodes.
- ▶ The LOOSE nodes receiving a RANN message are immediately recruited in the respective cluster and become MEMBERS.
- ▶ When a MEMBER node receives the RANN message from another HEAD it compares the e-metrics (including the distance) of the reference HEAD and the announced HEAD to check which cluster is the best to join.

# Clustering creation

## ► Example

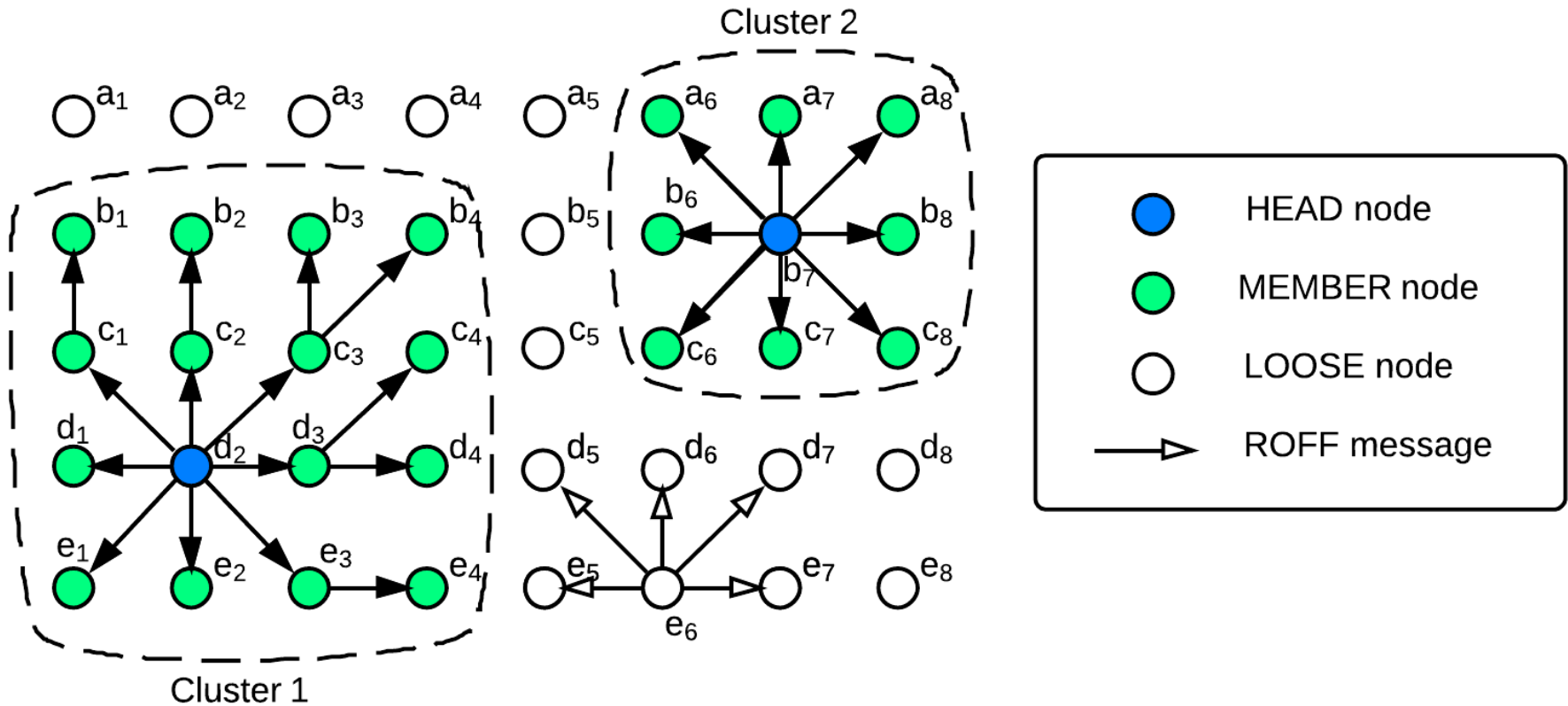


# Cluster destruction

- ▶ When a HEAD node resigns, it destroys its cluster by flooding a ROFF message containing its address. If the nodes receiving such a message find out that the sender is the reference HEAD, they then leave the cluster and become LOOSE nodes

# Cluster destruction

## ► Head $e_6$ resignation



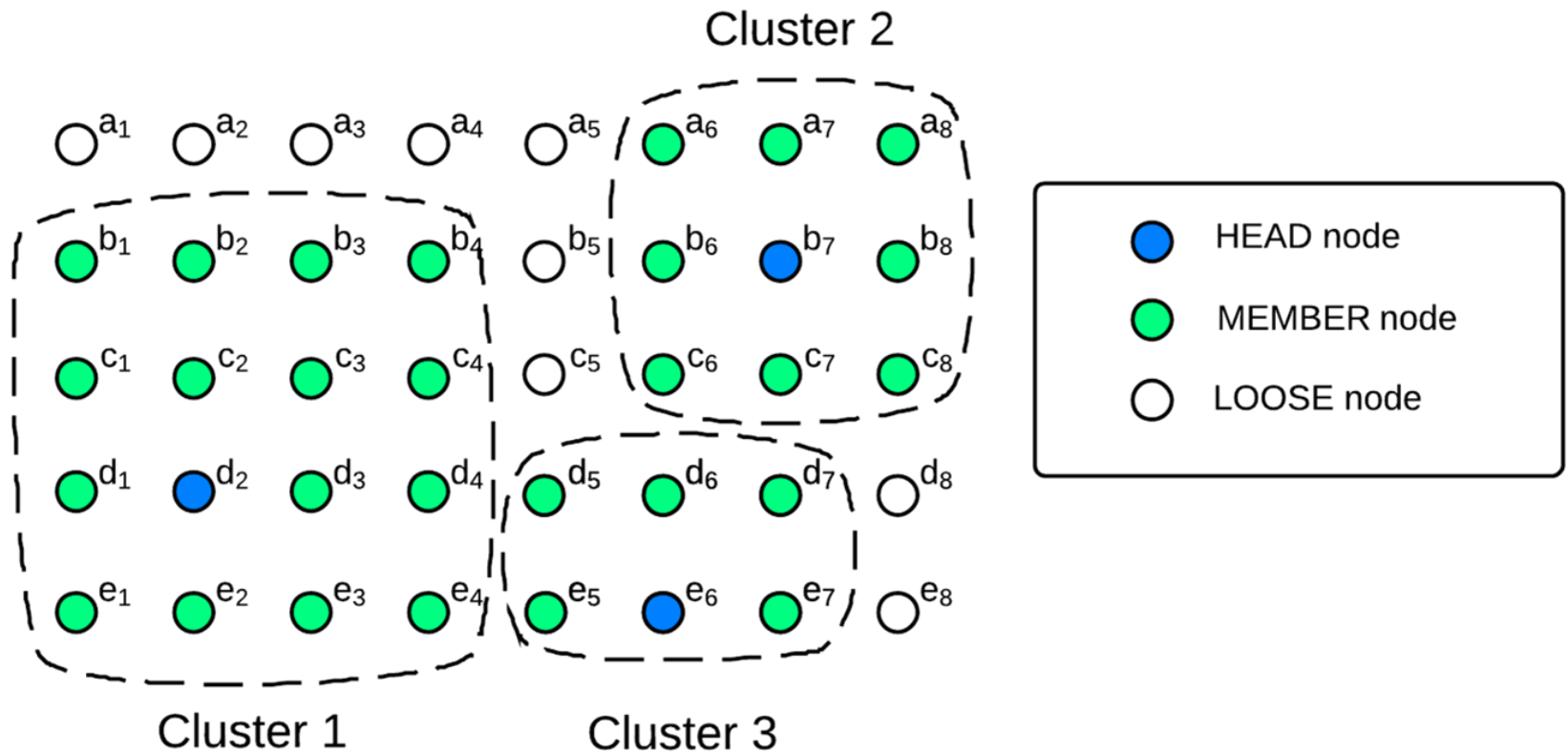


# HEAD competition

- ▶ CESAR aims to reduce the number of transmissions on the BS-link by limiting the number of cluster-heads in the network.
- ▶ If multiple neighboring nodes meet the election requirements, and one of them has the e-metric  $k$  times smaller than those of the other nodes, then it wins the competition and becomes HEAD, while the other competing nodes join its cluster.

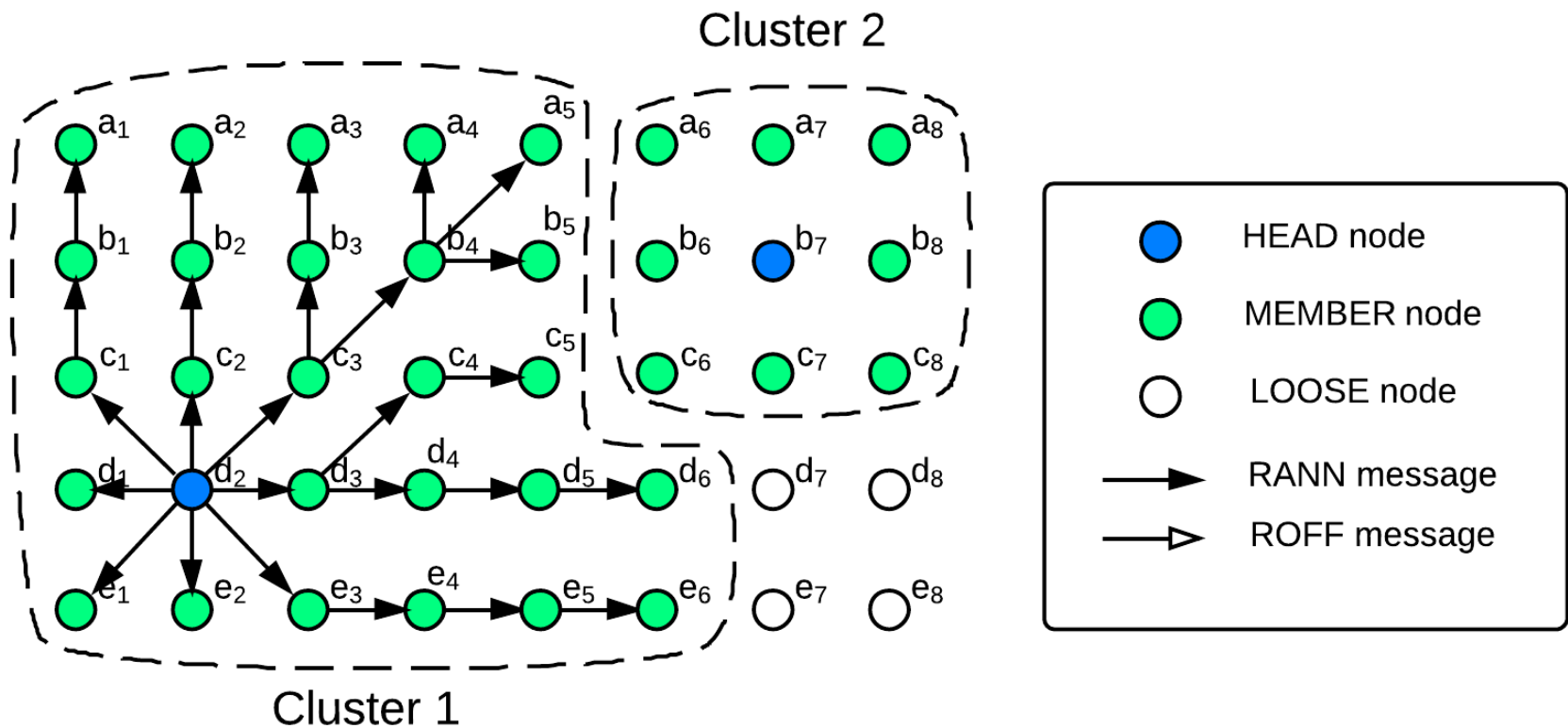
# HEAD competition

## ► Example:



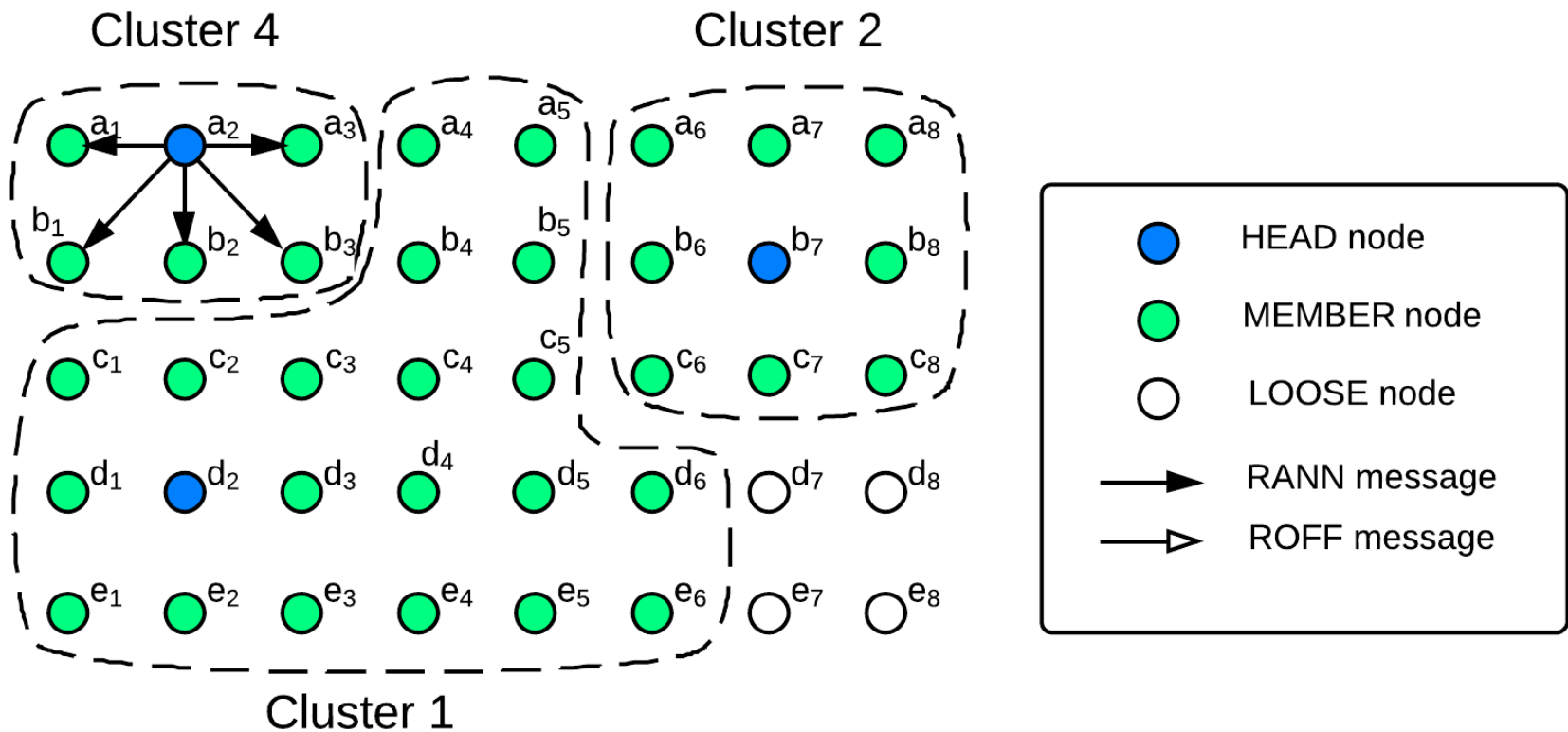
# HEAD competition

- ▶  $k * e\text{-metric}(d_2) < e\text{-metric}(e_6)$



# HEAD competition

- $k * e\text{-metric}(a_2) < e\text{-metric}(d_2)$



# On-demand scheme

- ▶ CESAR defines an on-demand mechanism that allows non-clustered nodes to discover a nearby cluster to which deliver the sensed data.
- ▶ Routing message:
  - ▶ REQR: hop-bounded broadcast messages used to discover nearby clusters.
  - ▶ REP: unicast message used to reply to REQR messages

# On-demand scheme

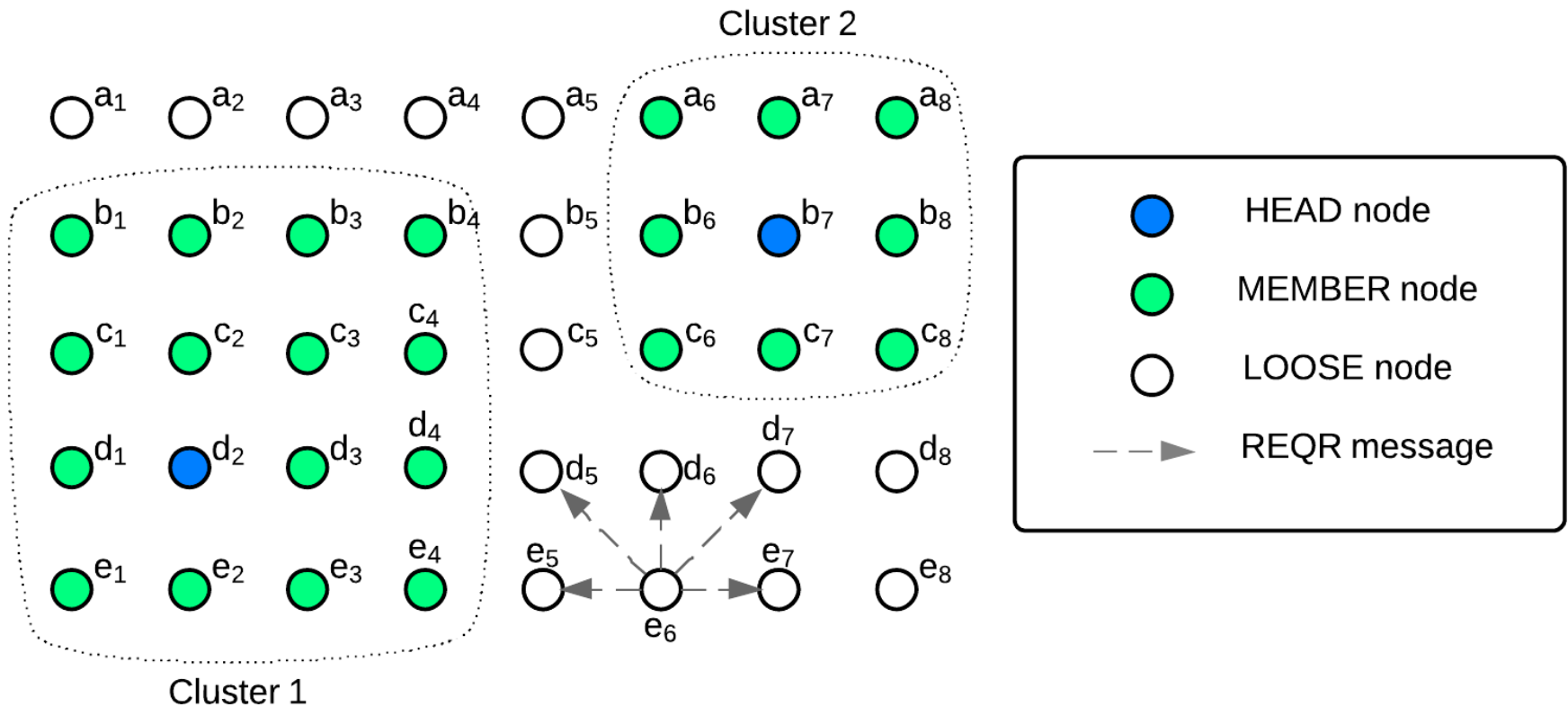
- ▶ The discovering procedure consists of flooding a hop-bounded REQR message in the neighborhood and waiting for a REP message.
- ▶ The range of the REQR is doubled at each attempt until a prefixed limit.
- ▶ When a MEMBER node receives the REQR message, it replies with a REP message containing the network address and the status of its HEAD.

# On-demand scheme

- ▶ If the requesting node receives more than one REP, it chooses to deliver its data to the cluster which HEAD has the lowest e-metric.
- ▶ The nodes receiving a REP message keep the route to the referenced cluster for a limited time after which it will be discarded.

# On-demand scheme

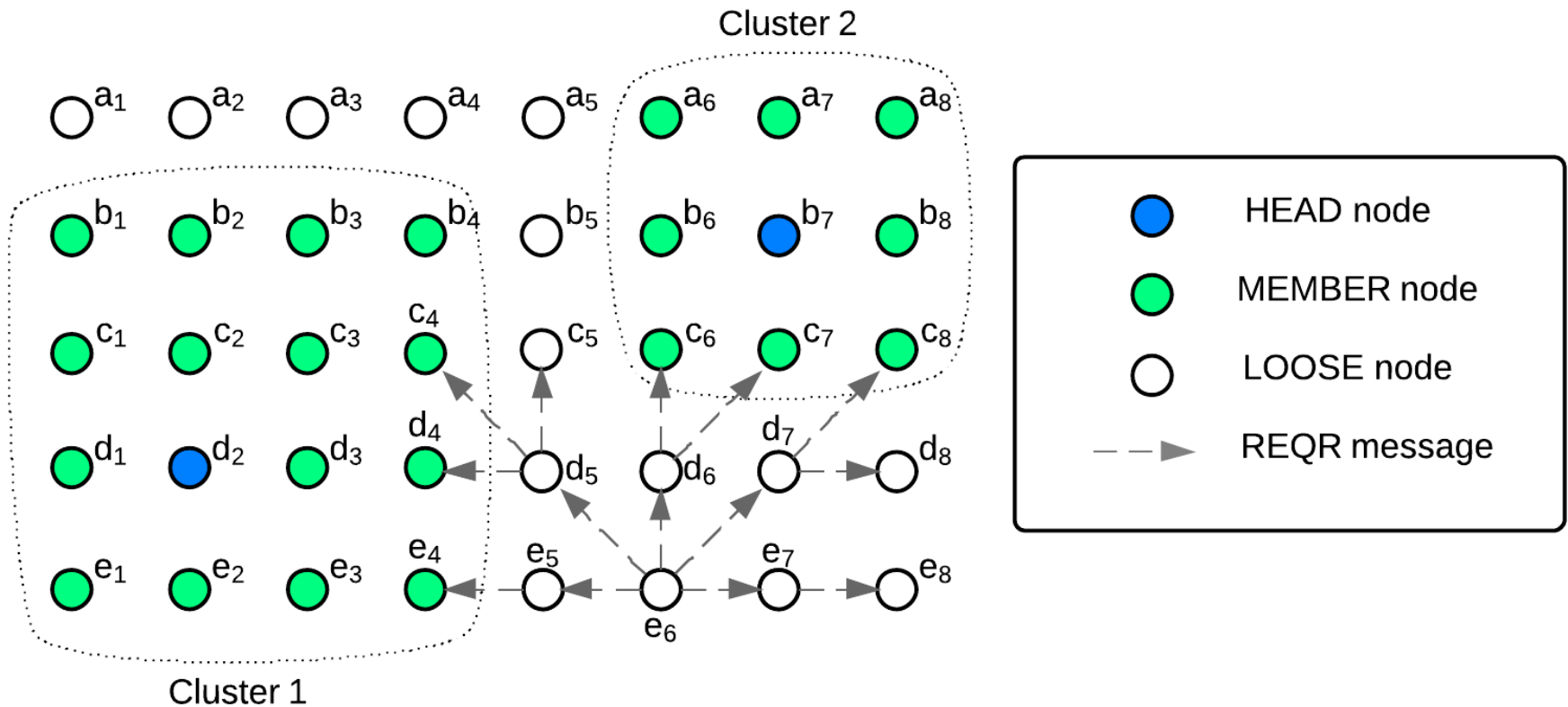
- ▶ Example: The node  $e_6$  has some data to transmit to the BS.





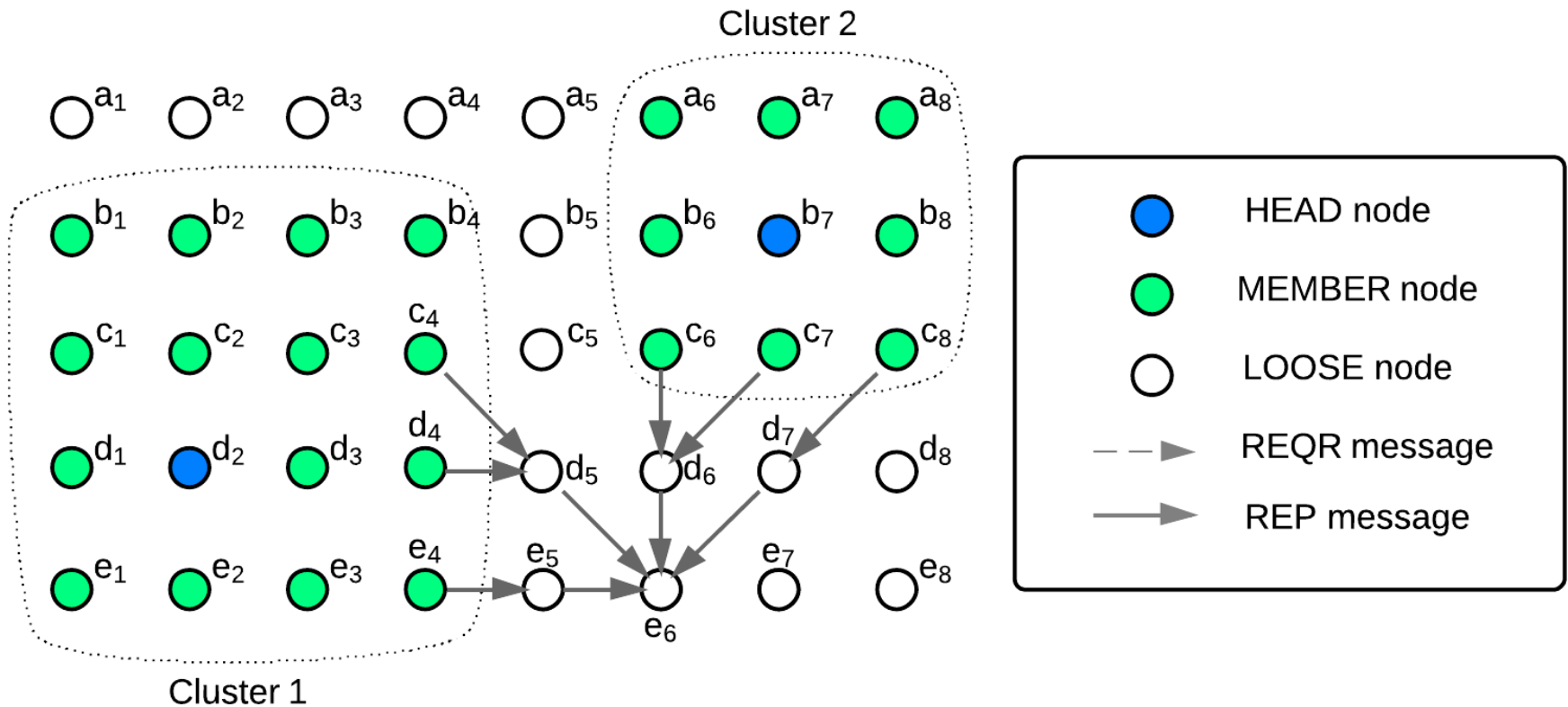
# On-demand scheme

- ▶ Example: The node  $e_6$  has some data to transmit to the BS.



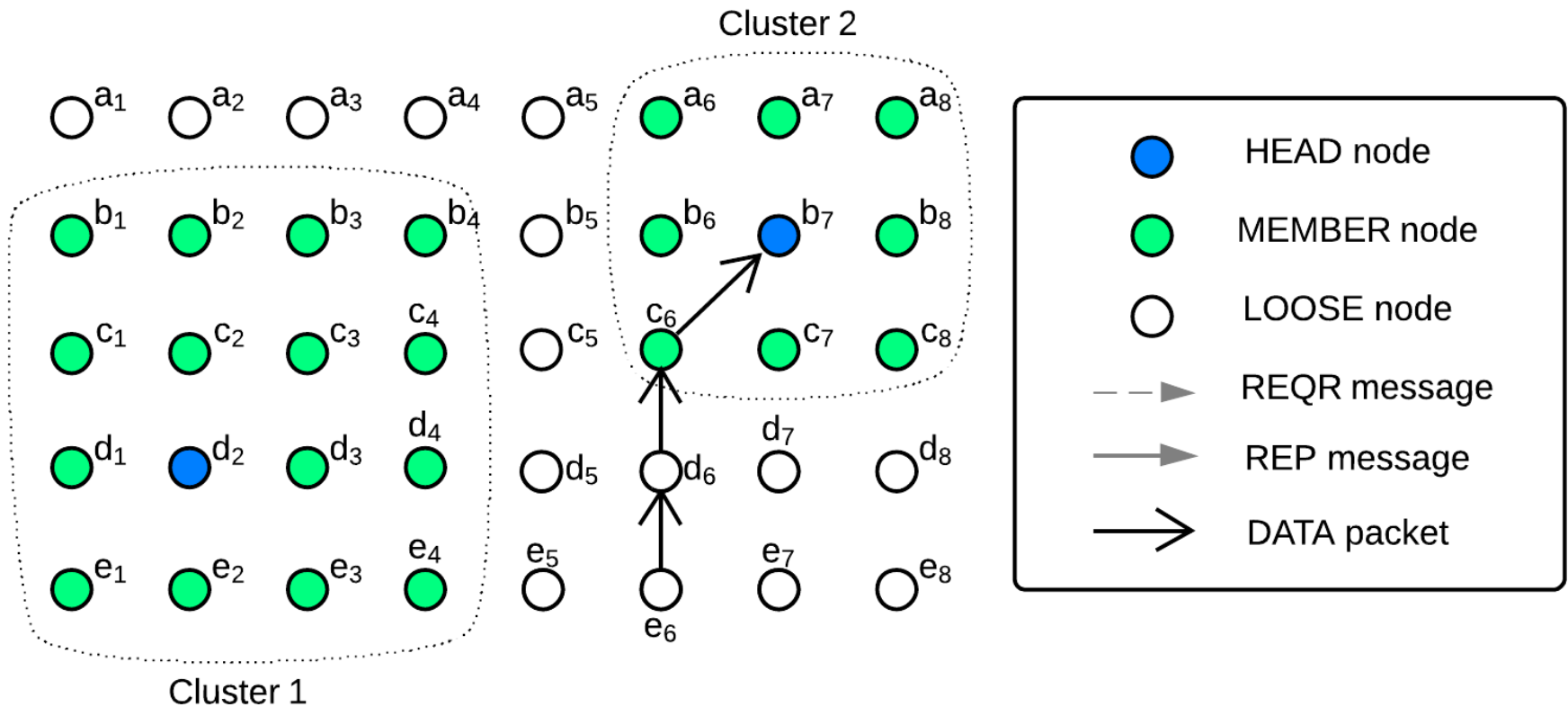
# On-demand scheme

- ▶ Example: The node  $e_6$  has some data to transmit to the BS.



# On-demand scheme

- ▶ Example: The node  $e_6$  has some data to transmit to the BS.



# Recovering scheme

- ▶ The CESAR algorithm uses ACK messages to acknowledge every DATA packets.
- ▶ If no ACK messages are received after the ACK\_timeout, then the sender node considers the packet lost and transmits it again.
- ▶ The ACK\_timeout is doubled at each retransmission in order to prevent congestions in the network.
- ▶ If the transmission fails 3 times, then the considered node becomes LOOSE and restarts the on-demand scheme to find another route to the same HEAD or to another cluster.

# Data aggregation

- ▶ CESAR can employ a data aggregation scheme in order to reduce the amount of data flowing into the network.
- ▶ Three possible aggregation modes:
  - ▶ Only at the HEAD nodes;
  - ▶ At the HEAD nodes and the MEMBER nodes located at the border of the clusters;
  - ▶ At the HEAD nodes and all the MEMBERS of the clusters.

# Simulation settings

- ▶ Simulator: WSNET
- ▶ Simulated algorithms: CESAR, LEACH, HEED
- ▶ Topology: 3D grid 160x160x110 m3

Topology	Nodes	Distance
5x5x4	100	30
6x6x4	144	25
7x7x4	196	21
8x8x4	256	18.5
9x9x4	324	16.5
10x10x4	400	15
11x11x4	484	13.5
12x12x4	576	12.5
13x13x4	676	11.5
14x14x4	784	10.5
15x15x4	900	10

Network topologies

	XBee	XBee PRO
Supply voltage [V]	3.3	
TX current [mA]	50	215
RX current [mA]	45	55

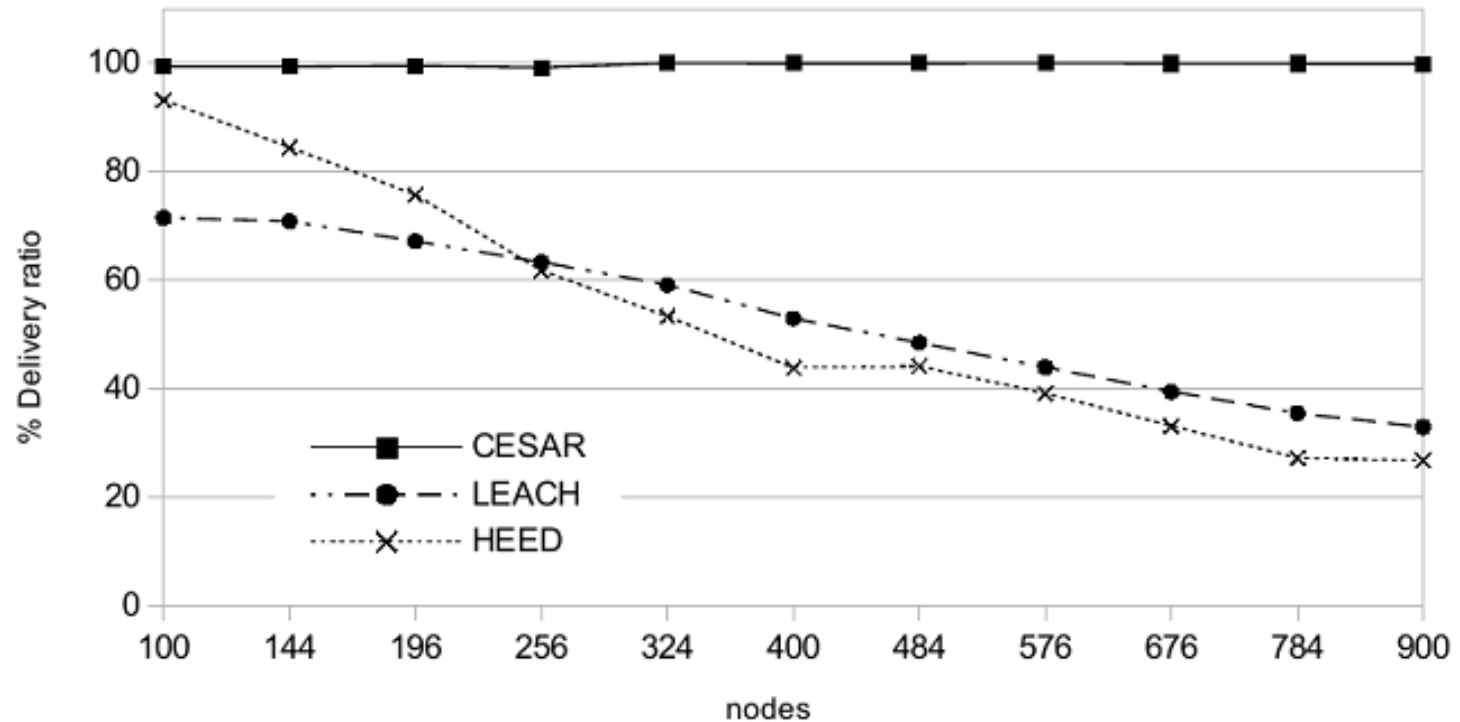
Power requirements (P2P-link)

Supply voltage [V]	4
Operating current [mA]	350
Peak current [A]	2

Power requirements (BS-link)

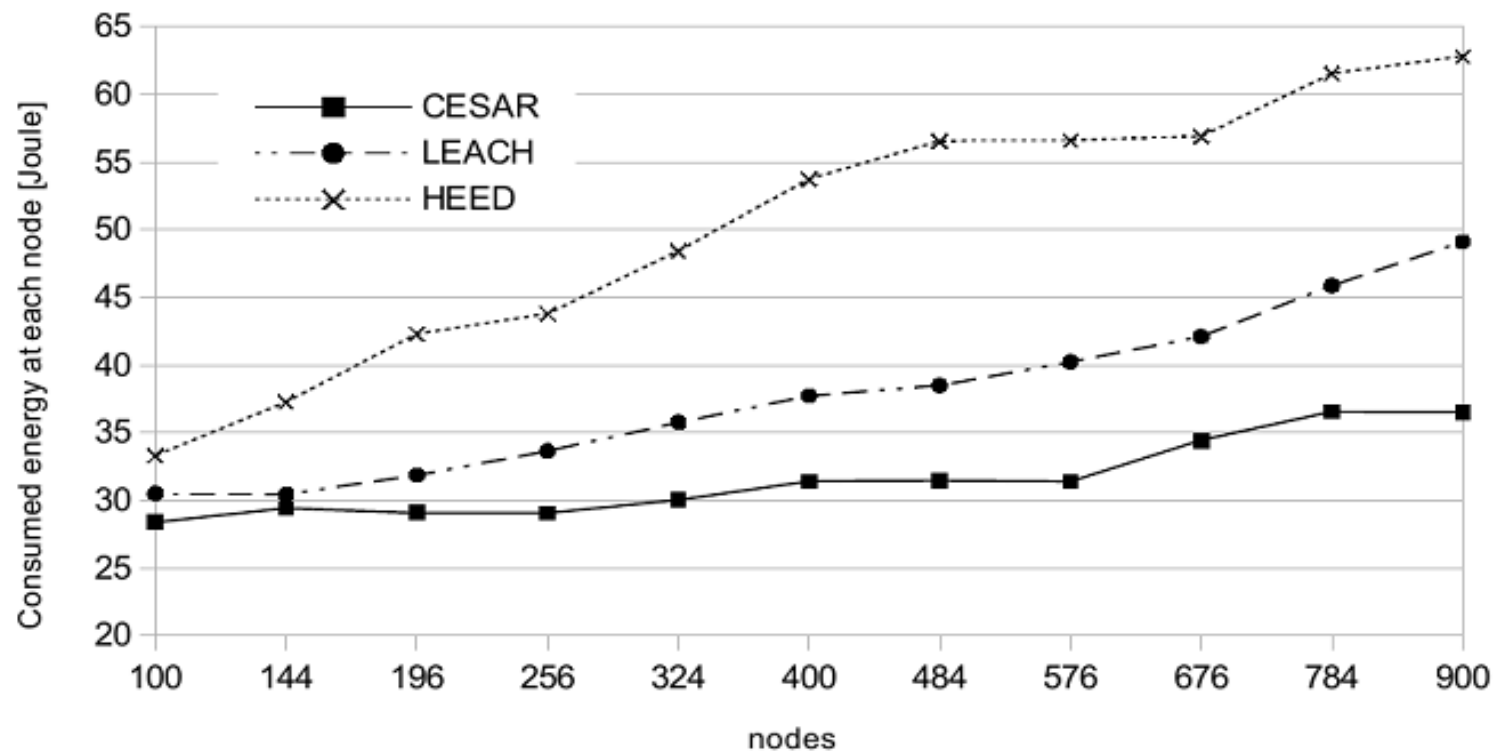
# Simulation results

## ► Data delivery



# Simulation results

## ► Energy consumption





# Use cases

- ▶ **Network model**
  - ▶ Large number of network nodes;
  - ▶ Variable radio environment;
  - ▶ Dynamic topology;
- ▶ A possible example is the monitoring of the goods containers in a terminal where the containers can be added or removed at any time.

# Conclusions

## ▶ Algorithm features:

- ▶ The election scheme elects only the nodes which have the requirements to be HEAD by taking account of the battery level and the quality of the BS-link;
- ▶ An election metric defines the quality of the HEAD nodes by considering the mentioned election parameters;
- ▶ Local competitions between neighboring cluster-heads are performed in order to limit the number of clusters in the network;
- ▶ The size of the clusters is decided according to the election metric of the HEAD nodes;
- ▶ An on-demand scheme is defined for non-clustered nodes;

# Conclusions

- ▶ **Algorithm features:**
  - ▶ CESAR combines the clustering and on-demand schemes in order to improve the robustness of the network and reduce the energy consumption;
  - ▶ The acknowledgement of the packets and a recovering scheme are introduced;
  - ▶ The aggregation of data can be performed in different point of the clusters.



The background of the slide features a complex network diagram. It consists of numerous nodes, some of which are colored blue, green, or white with a black outline. These nodes are interconnected by a series of directed edges, represented by thin gray lines with small black arrowheads indicating the direction of flow. The network is distributed across the slide, with some clusters and many isolated nodes. The text "Thank You" is centered over this network.

Thank You