22nd Annual Conference
Net-Centric
SDN/MPLS/NFV
2020

Distributed Graph Coloring for Cognitive Radio Networks

Juba Saadi, Abishek Mishra, Luciana Arantes, Pierre Sens, Thi-Mai-Trang Nguyen

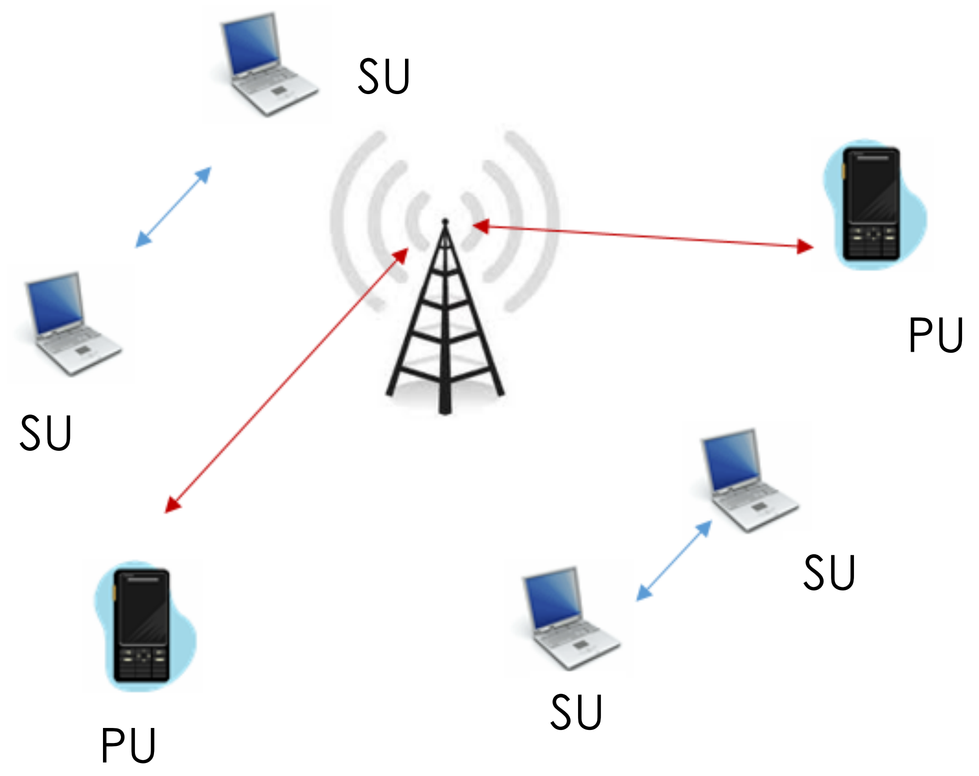
Sorbonne Université
Laboratoire d'Informatique de Paris 6 (LIP6)

Outline

- **Cognitive radio network & graph coloring**
- Propositions
  - Distributed D-Satur
  - Distributed Two-Hop Local Channel Ranking (DTHLCR)
- Performance evaluation
- Conclusion

Cognitive Radio Network

- Primary User : has the priority to use spectrum
- Secondary User : use the spectrum in an opportunistic manner without disturbing PU's activities
- Channel availability in CRN is highly dynamic
- Cognitive Radio AdHoc Network (CRAHN) is an adhoc networks in which nodes are secondary users
- Applications: Wi-Fi – cellular coexistence, D2D, IoT, V2V → increase the spectrum usage efficiency



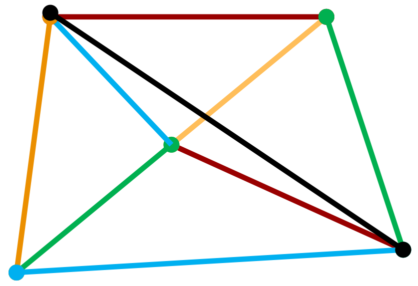
Frequency assignment in CRN

- Frequencies available at different CRN nodes are different depending on the activities of primary users
  - Frequencies used for the communications between nodes may change frequently
  - The coloring quality has a great impact on the interference level in the network
- Frequency assignment in CRN can be done in different manners
  - Use the first common frequency detected by a rendez-vous protocol
  - Use the Common Control Channel (CCC) to exchange the frequency lists and select a common frequency for data exchange
  - Use the channel detected by a rendez-vous protocol to exchange the frequency list then select a frequency for data exchange
- Frequency assignment can be modeled as a graph coloring problem

Graph coloring

- K-coloring (an edge coloring with at most *k* colors)
- The Vizing theorem shows that we can edge-colors any graph G with *d+1* colors where *d* denotes the maximum degree of the graph

d=4



Limitations classical coloring algorithms

- Suppose that we have a sufficient number of colors to color the graph
  - Not the case in practice
- Suppose that all nodes have access to the same set of colors
  - Not the case for CRN
- Neighbors are forbidden to use the same color
  - Possible for wireless networks but with lower performances due to interference
- Most of classical algorithms are for vertex coloring
  - We need edge coloring

Centralized vs. Distributed graph coloring

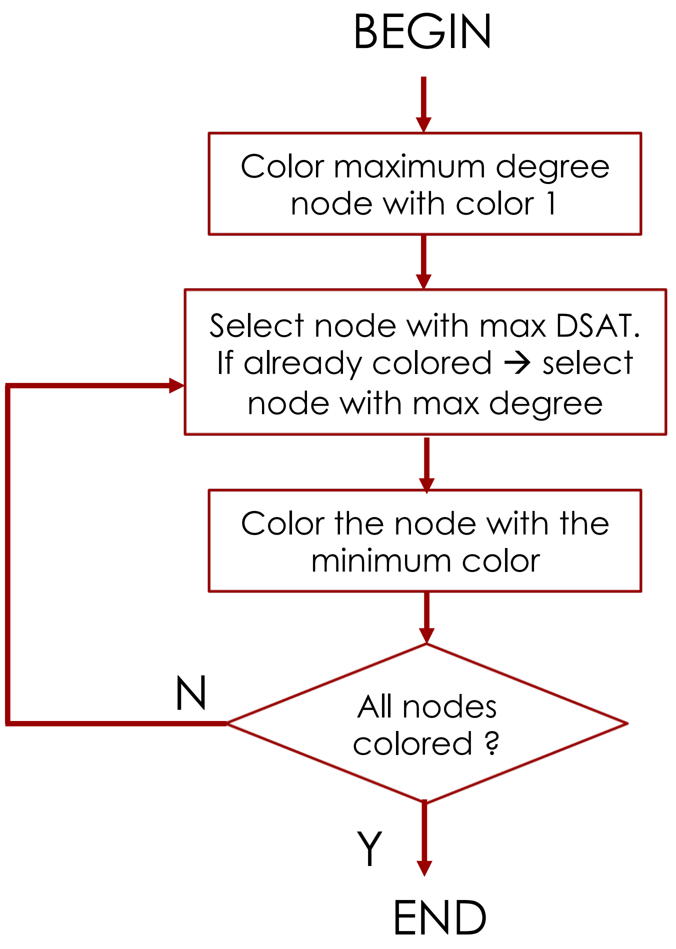
- Centralized graph coloring
  - Need to have a global view of the network and the colors available at each node
  - High cost to collect information on topology and spectrum availability
- Distributed graph coloring
  - Based on local view and local decision at each node
  - Overhead depends on the size of the local view
  - More suitable for large scale wireless sensor network (e.g. IoT networks)
- Our propositions for distributed graph coloring in cognitive radio network
  - Distributed D-Satur
  - Distributed Two-Hop Local Channel-Ranking (DTHLCR)

Outline

- Cognitive radio network
- Graph coloring for channel assignment
- **Propositions**
  - **Distributed D-Satur**
  - **DTHLCR**
- Simulations and results
- Conclusion

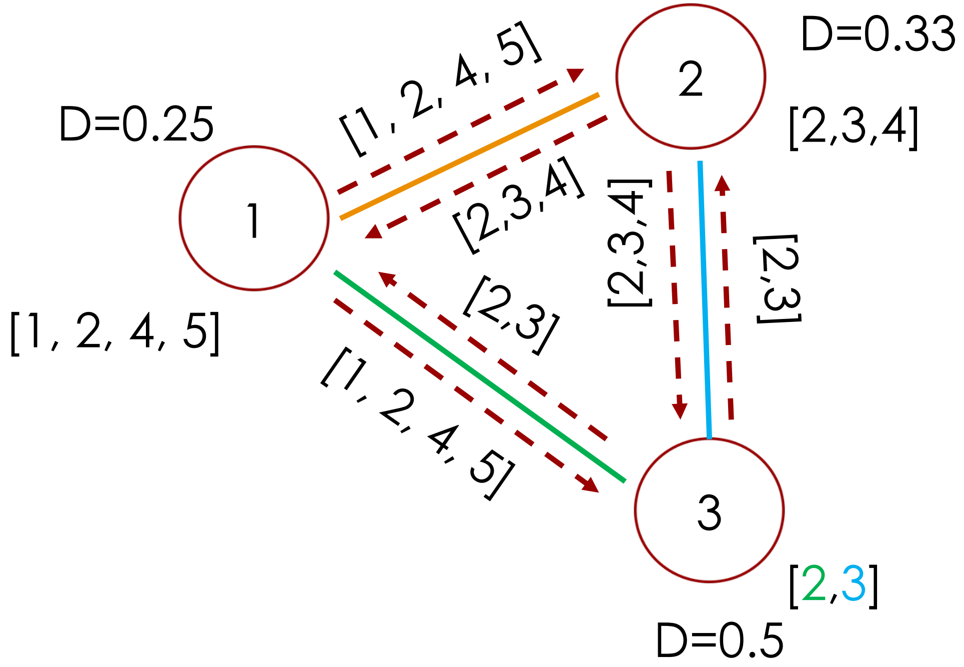
D-Satur

- A centralized algorithm for vertex coloring
- Based on the concept of Degree of Saturation (DSAT)
- Given a graph *G=(V,E)*
- For each vertex v
  - *DSAT(v) = degree(v)* ,
    if none of v's neighbor is colored
  - *DSAT(v)* = number of colors used by v's neighbors, otherwise
- List of colors 1, 2, …., K



Distributed D-Satur

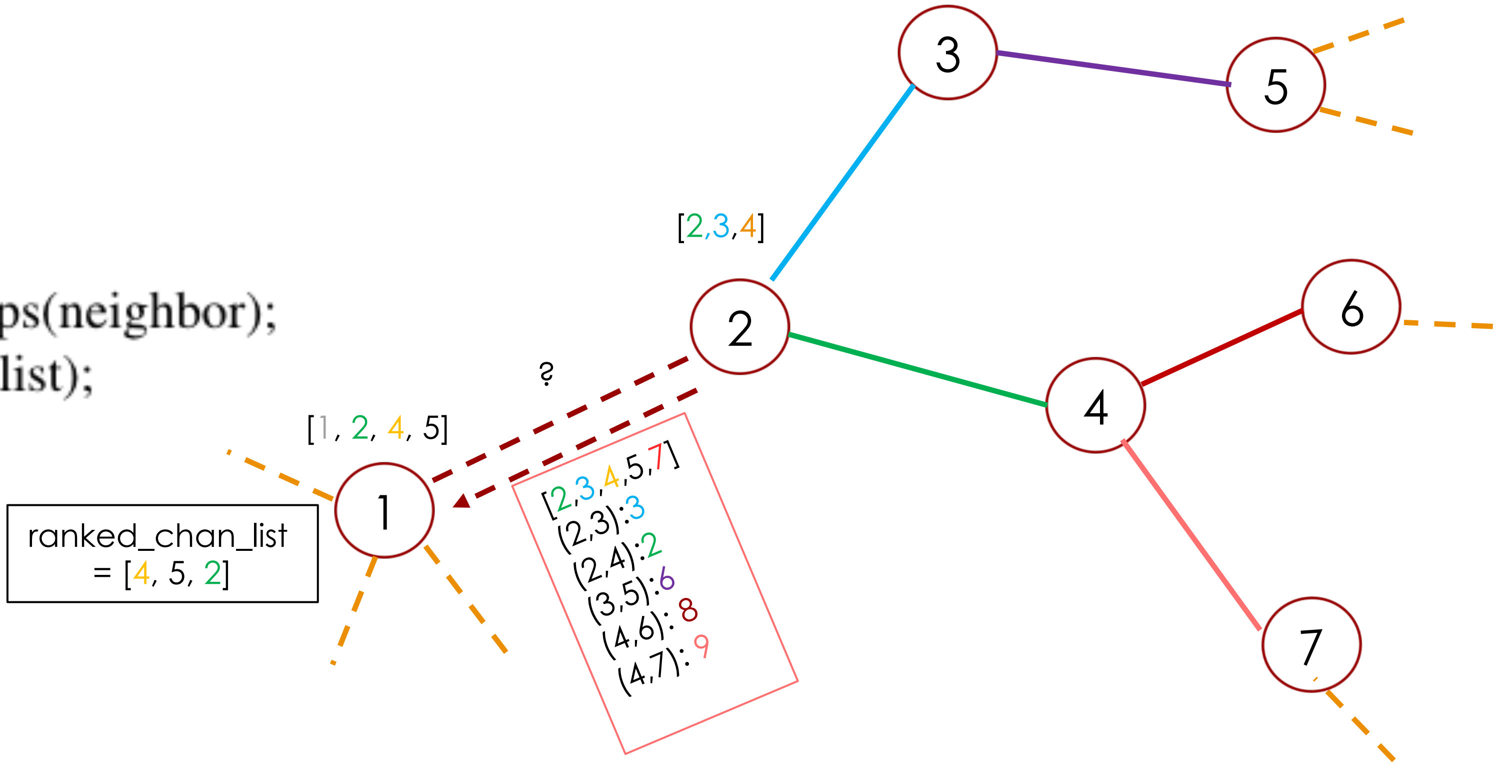
- A variation of centralized D-Satur
- Degree of saturation (D) is defined as the inverse of the number of frequencies available of each node
- Coloring starts with node having higher g-degree of saturation (less choice of channels)
- Node ID is used in case of equal degree of saturation



Node 3 colors their links first

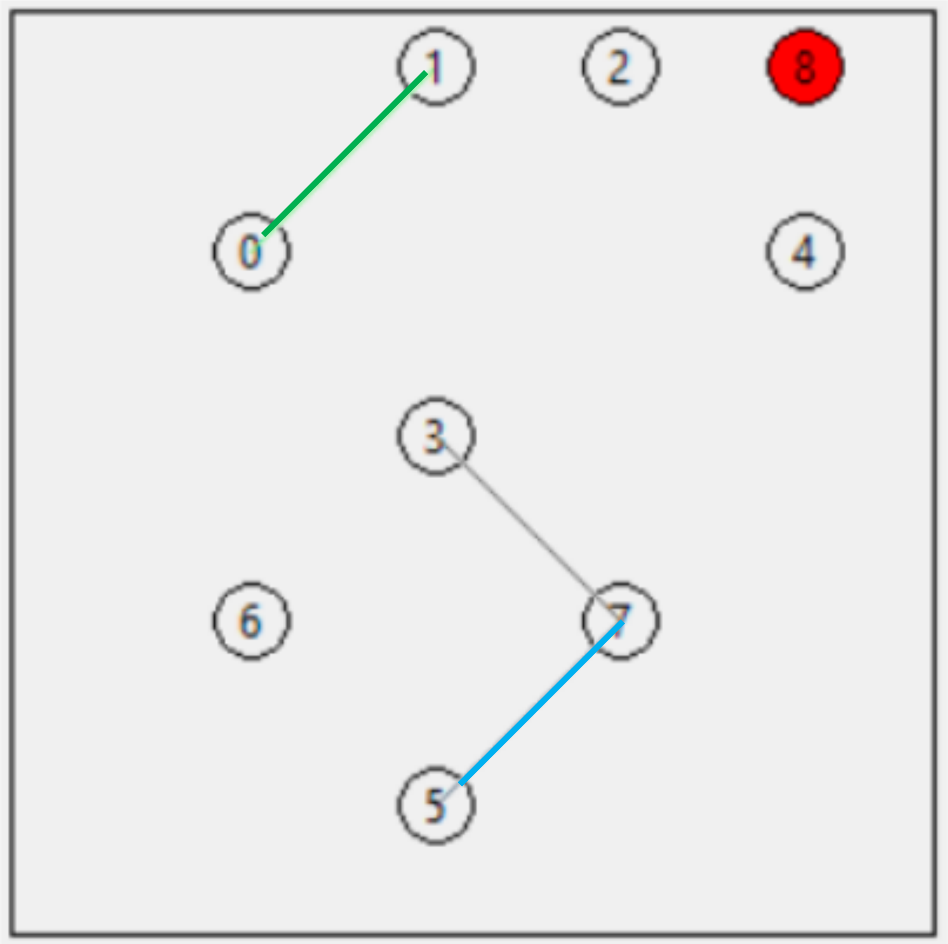
Distributed Two-Hop Local Channel Ranking

```
Result: All edges are colored
to_color = [ ];
forall neighbor in neighbor_nodes do
    if not link_already_colored then
        chan_list = get_common_2_hops(neighbor);
        rank_by_usage_of_color(chan_list);
        color_link(chan_list[0]);
        to_color.append(neighbor);
    end
end
forall to_color do
    start algorithm on node;
end
```



Why two-hop view ?

- Let's consider a CRN with 4 colors available at node 0
  - Blue, red, green, grey
- Node 0 wants to color link (0,3)
- By the local view, node 0 avoids the green color
- By 1-hop neighbor view, node 0 can avoid also using the grey color
- However, it'd be better to avoid using the blue color too !
  - Link (7,5) is colored in blue
- If node 0 knows that one of the neighbor of node 3 is using the blue color, it will try to avoid coloring link (0,3) in blue

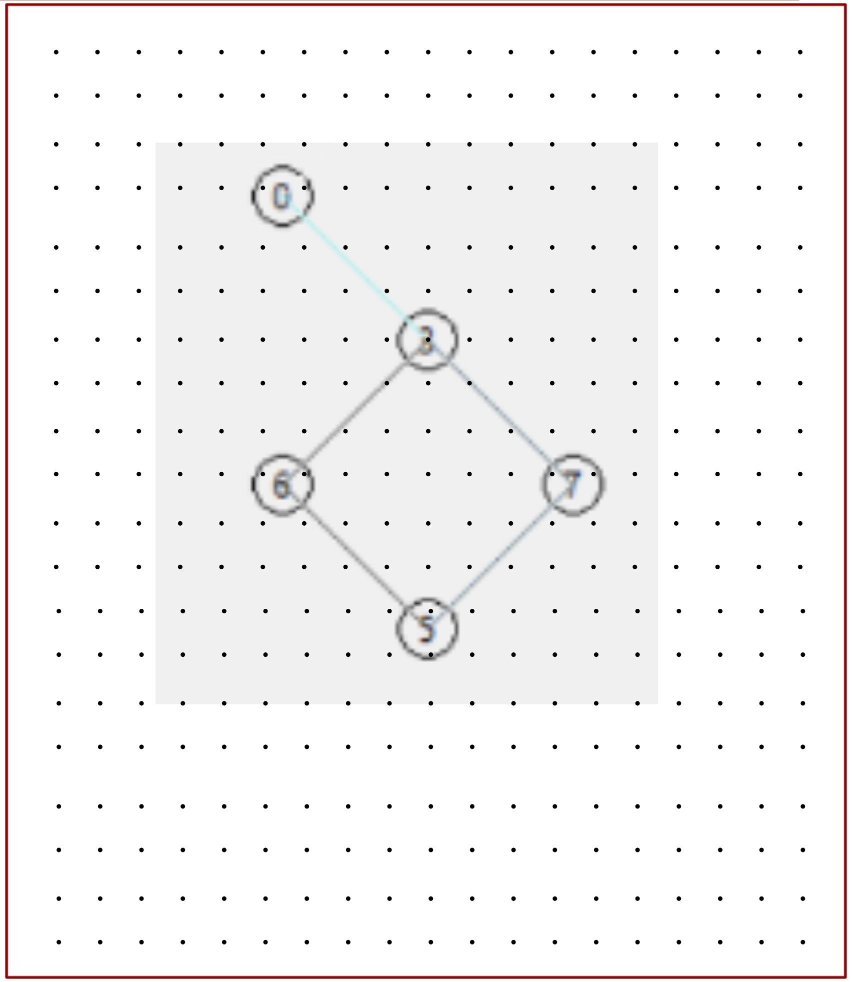


Outline

- Cognitive radio network & graph coloring
- Propositions
  - Distributed D-Satur
  - DTHLCR
- **Performance evaluation**
- Conclusion

About the simulator

- We have developed a simulator written in Python
- Nodes are randomly placed in a 2-dimensional space
- Nodes are randomly put in a 20 x 20 units square area
- A node can be a PU or SU
- The number of available channels at each node can be varied and randomly chosen

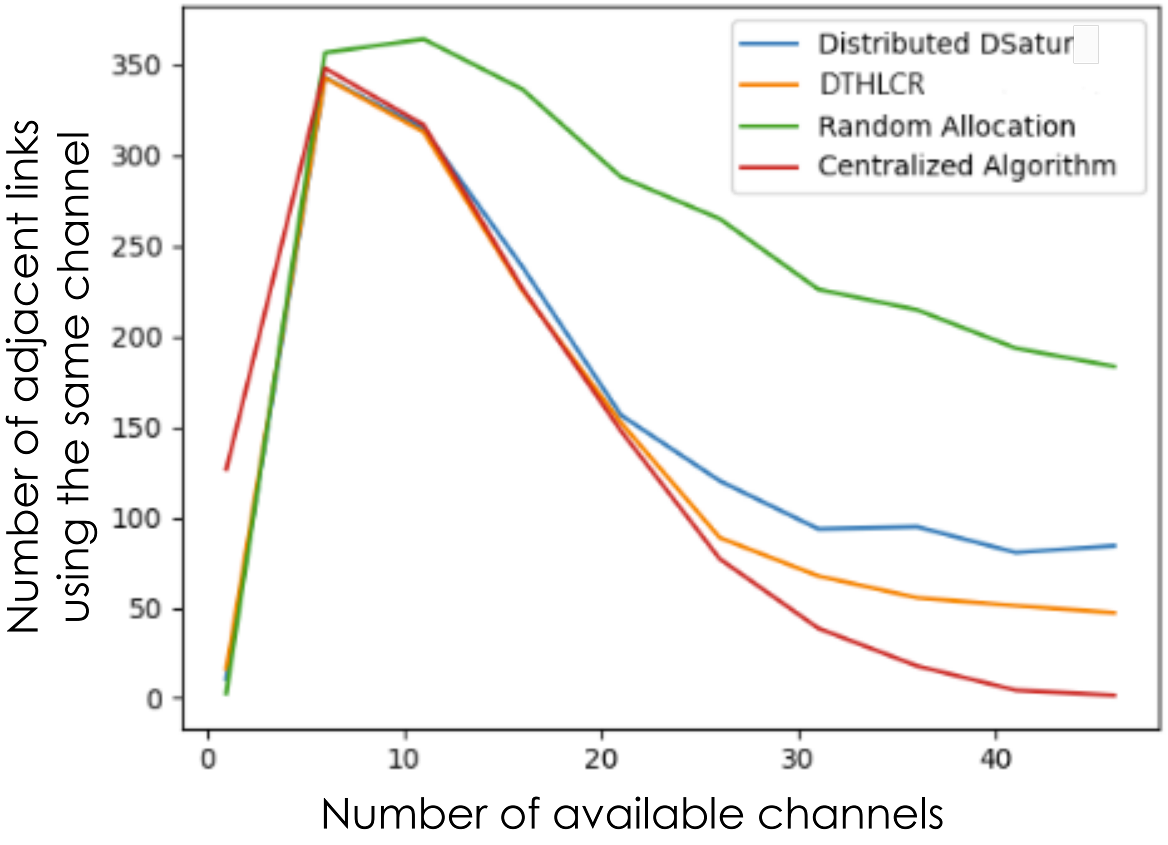


Baseline algorithms

- Random coloring
  - Nodes exchange their sets of available channel
  - One of common available channel is randomly chosen
- Centralized edge coloring
  - Based on the global graph
  - Build a MIS (Maximal Independent Set) for each color
  - For the rest of edges that are still not colored, we are obliged to choose some colors already allocated

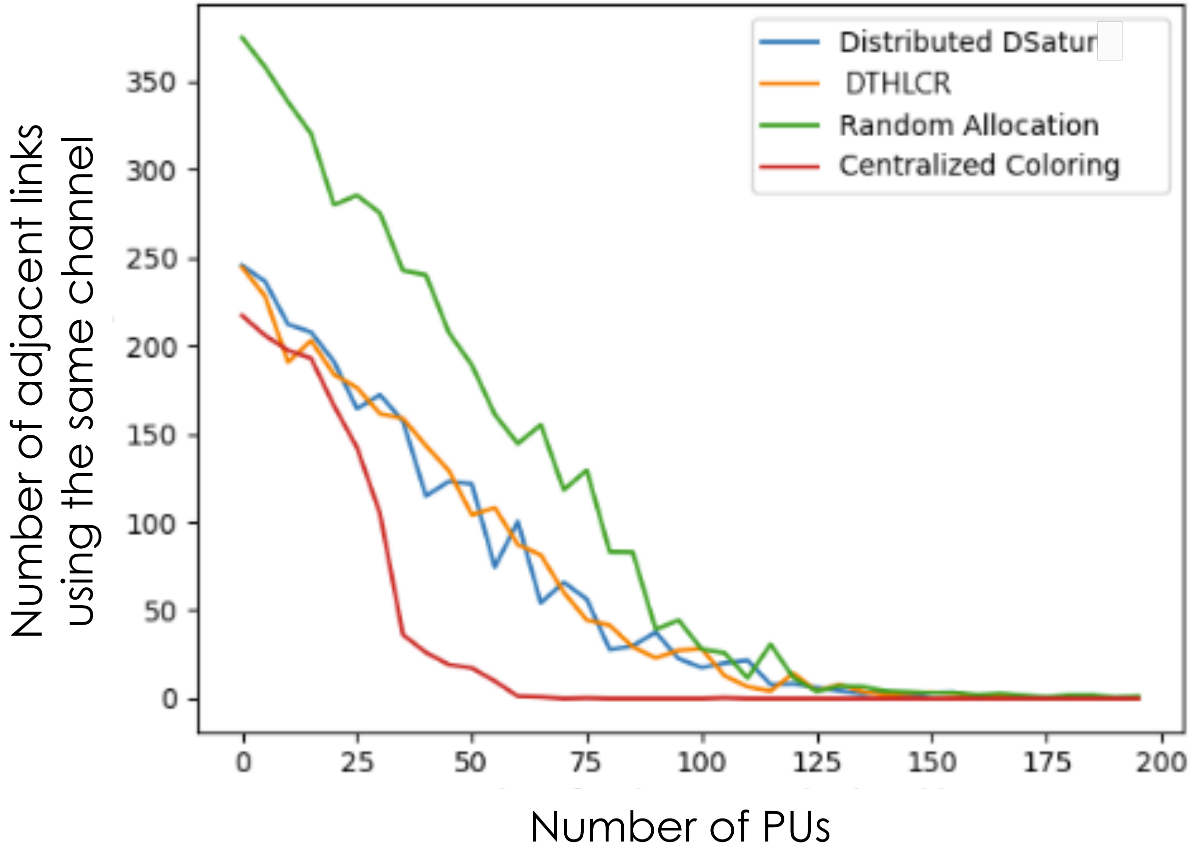
Interference level in function of number of channels

- Simulation configuration
  - Number of available channels varies from 0 to 50
  - Transmission range is 1 unit
  - 250 nodes
- The number of adjacent links using the same channel tends towards a constant more or less near zero
- The centralized coloring algorithm tends to 0 faster
- The DTHLCR algorithm offers a lower interference level compared with the distributed D-Satur algorithm
- Random coloring causes a very high interference level



Interference in function of number of PUs

- Simulation configuration
  - 11 available channels
  - 250 nodes (PUs + SUs)
  - Number of PUs varies from 0 to 200
- DTHLCR and Distributed D-Satur have almost the same performance but the the interference level of Distributed D-Satur has a higher variation
- Centralized coloring offers the best performance
- Random coloring gets the worst interference level



Outline

- Cognitive radio network & graph coloring
- Propositions
  - Distributed D-Satur
  - DTHLCR
- Performance evaluation
- **Conclusion**

Conclusion

- We have proposed two distributed coloring algorithms for channel assignment in cognitive radio networks
- The simulation results show that two-hop local view of link colors together with channel ranking is a good trade-off between protocol overhead and coloring quality
- In future works, we want to evaluate these algorithms in a network simulator to have more realistic network and traffic conditions
- An efficient coloring algorithm is necessary to cope with node mobility

Thank you