

# SEVA

Computação Móvel

# GLUE:Um modelo de clusterização utilizando mobilidade

Mestrando: Fernando Augusto Medeiros Silva

Orientador: Ricardo Augusto Rabelo de Oliveira



UFOP

Universidade Federal  
de Ouro Preto



# Organização

- ▶ Problema
  - Contextualização
  - Contribuição
- ▶ Referencial Teórico
- ▶ Relacionados
- ▶ Análise
- ▶ Proposta
- ▶ Resultados
- ▶ Conclusão e Trabalhos Futuros

# Problema

- ▶ Turismo com apoio digital frequente
- ▶ Informação originada e consumida in-loco
- ▶ Suporte exclusivo na nuvem deixando de lado a comunicação direta



# Problema

- ▶ Dispositivos inteligentes com comunicação direta WiFi Direct, Bluetooth
- ▶ Comunicação direta não estruturada
  - Topologia dinâmica – Mobilidade

# Contribuição

- ▶ Estudo de algoritmos de clusterização
- ▶ Implementação de algoritmo para o contexto de mobilidade
- ▶ Simulação dos algoritmos
- ▶ Estudo de viabilidade em equipamento comodity utilizando WiFi Direct
- ▶ Evolução do conhecimento comprovado pelo artigo publicado no MOBIWAC 2013: The 11<sup>th</sup> ACM International Symposium on Mobility Management and Wireless Access

# Organização

- ▶ Problema
  - Contextualização
  - Contribuição
- ▶ Referencial Teórico
- ▶ Relacionados
- ▶ Análise
- ▶ Proposta
- ▶ Resultados
- ▶ Conclusão e Trabalhos Futuros

# Referencial Teórico

- ▶ Sistemas Distribuídos
- ▶ Cluster em MANETs
- ▶ Redes Moveis Ad Hoc
- ▶ WiFi P2P

# Sistemas Distribuídos

- ▶ Coleção de entidades que executam em processadores separados e concorrentemente.
- ▶ Esses sistemas interagem por algum modelo de troca de informações e de forma a resolver um problema em comum, agindo como um só elemento.

# Algoritmos Distribuídos

- ▶ Algoritmos para sistemas distribuídos
- ▶ Caracterizam pela troca de mensagens entre os elementos processados
- ▶ Diferem-se pelos seus atributos
  - Modelo de tempo
    - Síncronos
    - Assíncronos

# Modelos de Tempo

- ▶ Síncrono: assume que as entidades executam computação simultaneamente em resposta a um relógio global.
- ▶ Assíncrono: Executam os passos em ordem arbitrária

# Template Sincrono

---

## Algoritmo 3.1: S\_Template

---

**Input:**  $s = 0, MSG_i(0) = \emptyset$

**2 Action:** if  $n_i \in N_0$  then

3     *Executa processamento;*

4     *Envia uma mensagem em cada aresta do conjunto  $Out_i$  (possivelmente vazio)*

**Input:**  $s > 0, MSG_i(1), \dots, MSG_i(s)$  sendo que  $origin(msg) = c_k \in In_i$  com  $1 \leq k \leq |In_i|$  para  $msg \in \bigcup_{r=1}^s MSG_i(r)$

**6 Action:** begin

7     *Executa processamento;*

8     *Envia uma mensagem em cada aresta do conjunto  $Out_i$  (possivelmente vazio)*

---

# Template Assincrono

---

## Algoritmo 3.2: A\_Template

---

**Input:**  $msg_i = nil$

**2 Action:** if  $n_i \in N_0$  then

3     *Executa processamento;*

4     *Envia uma mensagem em cada aresta do conjunto  $Out_i$  (possivelmente vazio)*

**Input:**  $msg_i$  sendo que  $origin(msg) = c_k \in In_i$  com  $1 \leq k \leq |In_i|$

**6 Action:** if  $B_k$  then

7     begin

8         *Executa processamento;*

9         *Envia uma mensagem em cada aresta do conjunto  $Out_i$  (possivelmente vazio)*

---

# Referencial Teórico

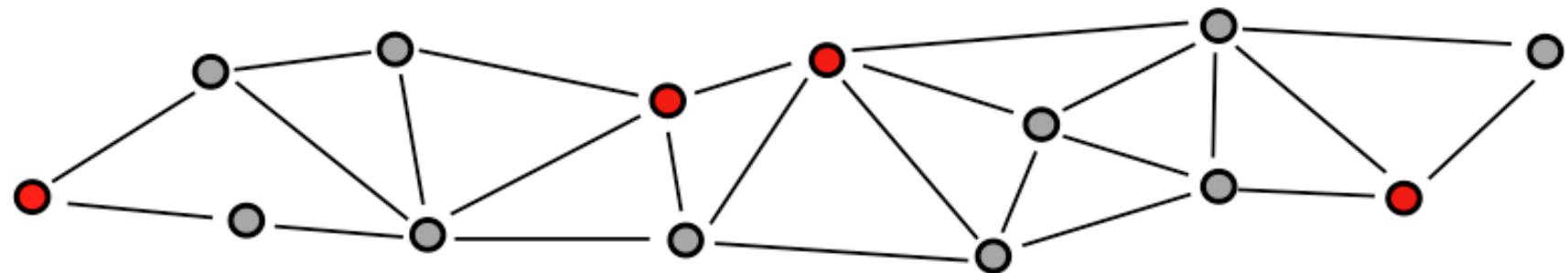
- ▶ Sistemas Distribuídos
- ▶ Clusterização
- ▶ Redes Moveis Ad Hoc
- ▶ WiFi P2P

# Clusterização

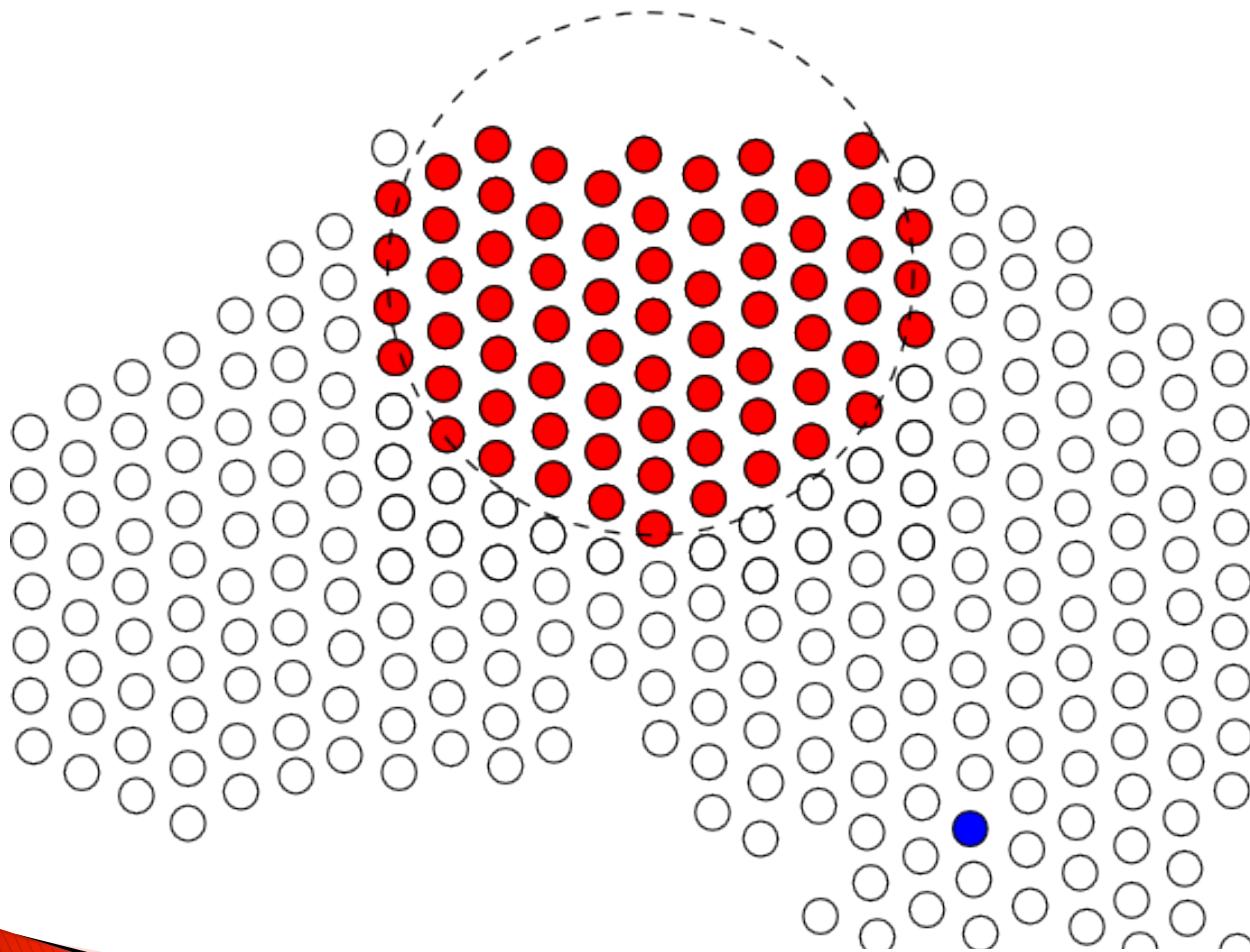
- ▶ Formação de um conjunto dominante mínimo
- ▶ Problema NP Completo

# Cluster

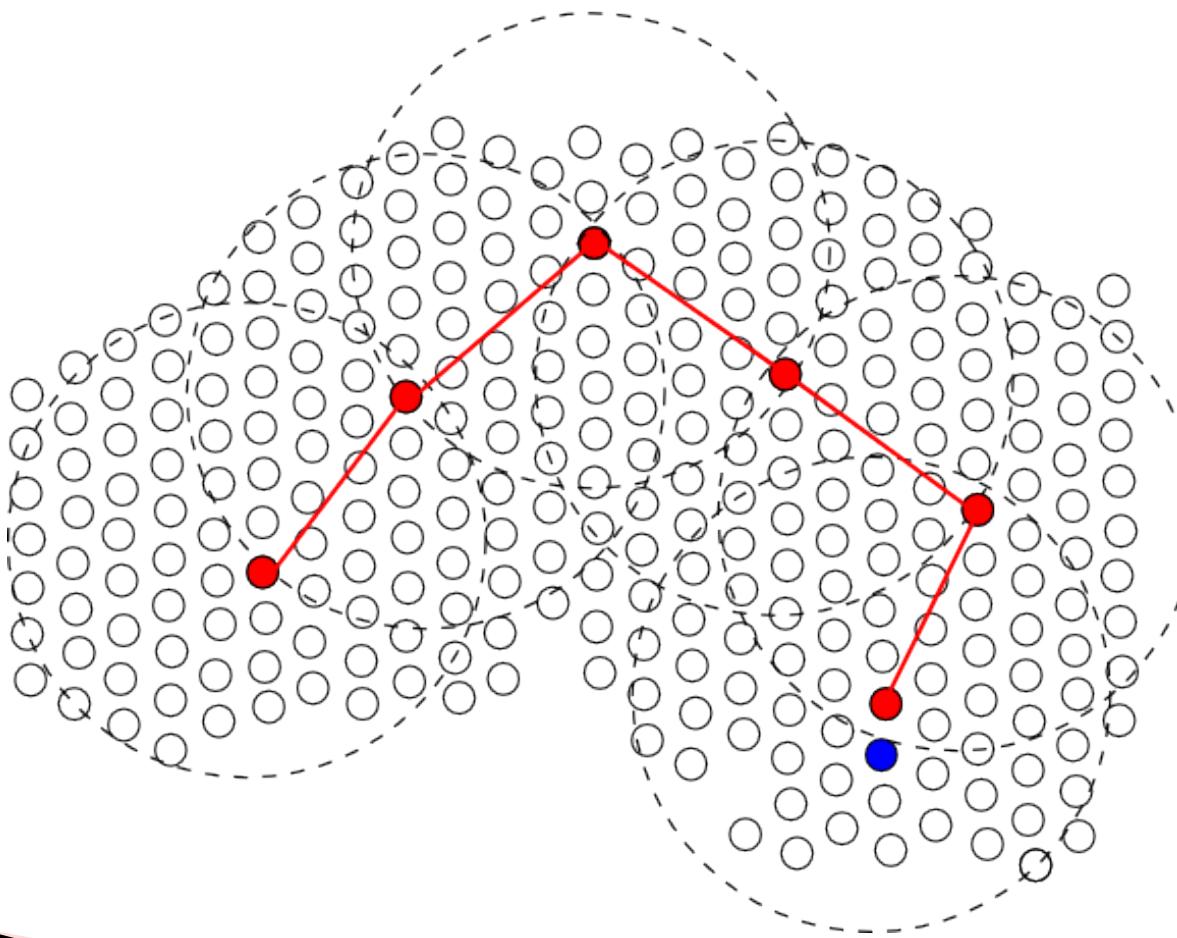
- ▶ Vértices do conjunto dominante são Cluster Heads
- ▶ Roteamento Eficiente



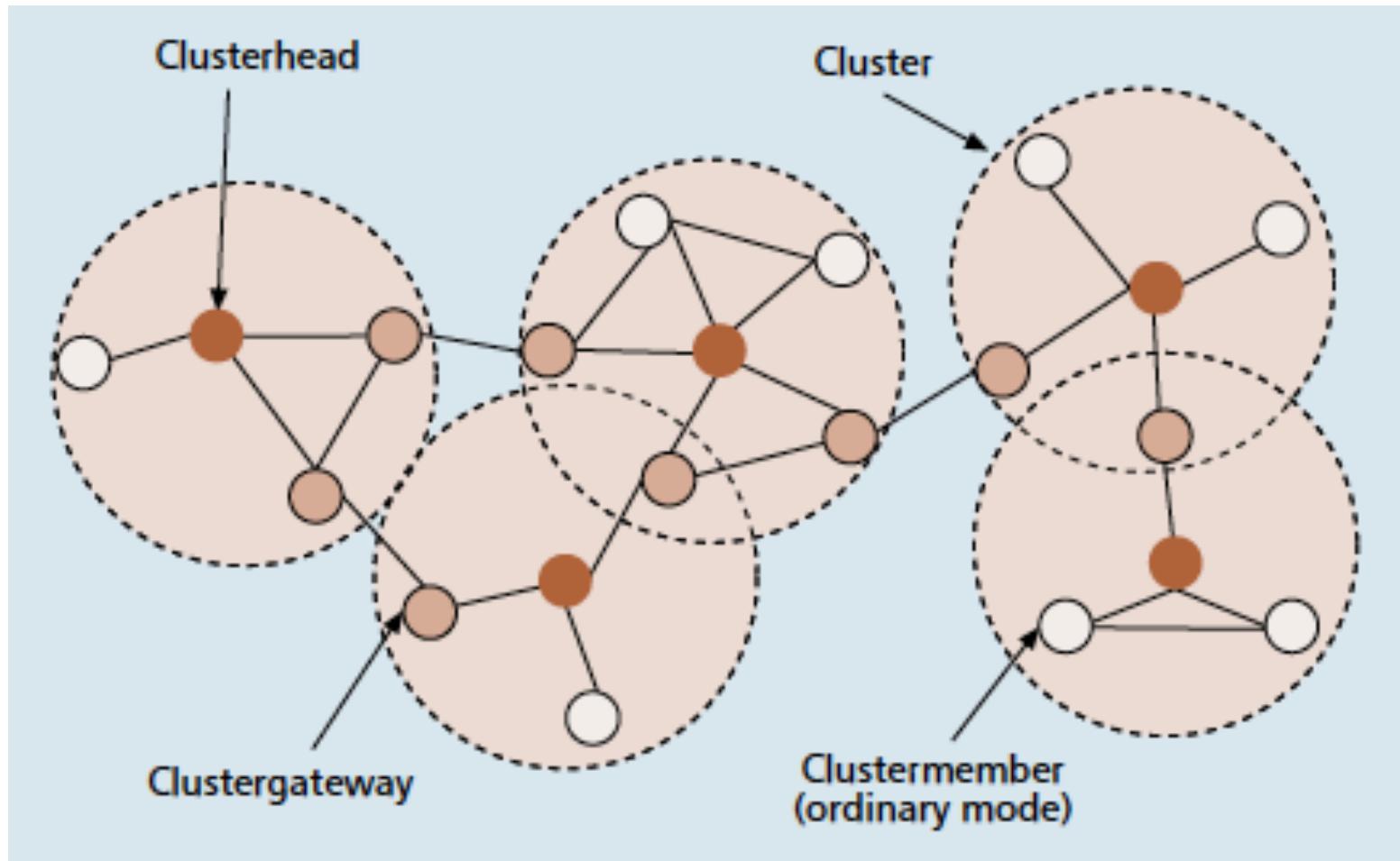
# Flood



# Hierárquico por Cluster



# Estrutura do Cluster



# Referencial Teórico

- ▶ Sistemas Distribuídos
- ▶ Cluster em MANETs
- ▶ **Redes Moveis Ad Hoc**
- ▶ WiFi P2P



# Redes Moveis Ad Hoc

- ▶ Topologia dinâmica
- ▶ Banda limitada
- ▶ Energia limitada
- ▶ Segurança física limitada

# Topologia dinâmica

- ▶ Elementos não conhecidos
- ▶ Elementos alteram suas propriedades
- ▶ Elementos entram e saem da rede

# P2P

- ▶ Lida com problemas parecidos das MANETs
- ▶ Origem em compartilhamento de arquivos
- ▶ Controle descentralizado
- ▶ Auto-Organização
- ▶ Escalabilidade
- ▶ Movimento para Ubicomp

# Referencial Teórico

- ▶ Sistemas Distribuídos
- ▶ Cluster em MANETs
- ▶ Redes Moveis Ad Hoc
- ▶ WiFi P2P

# WiFi P2P

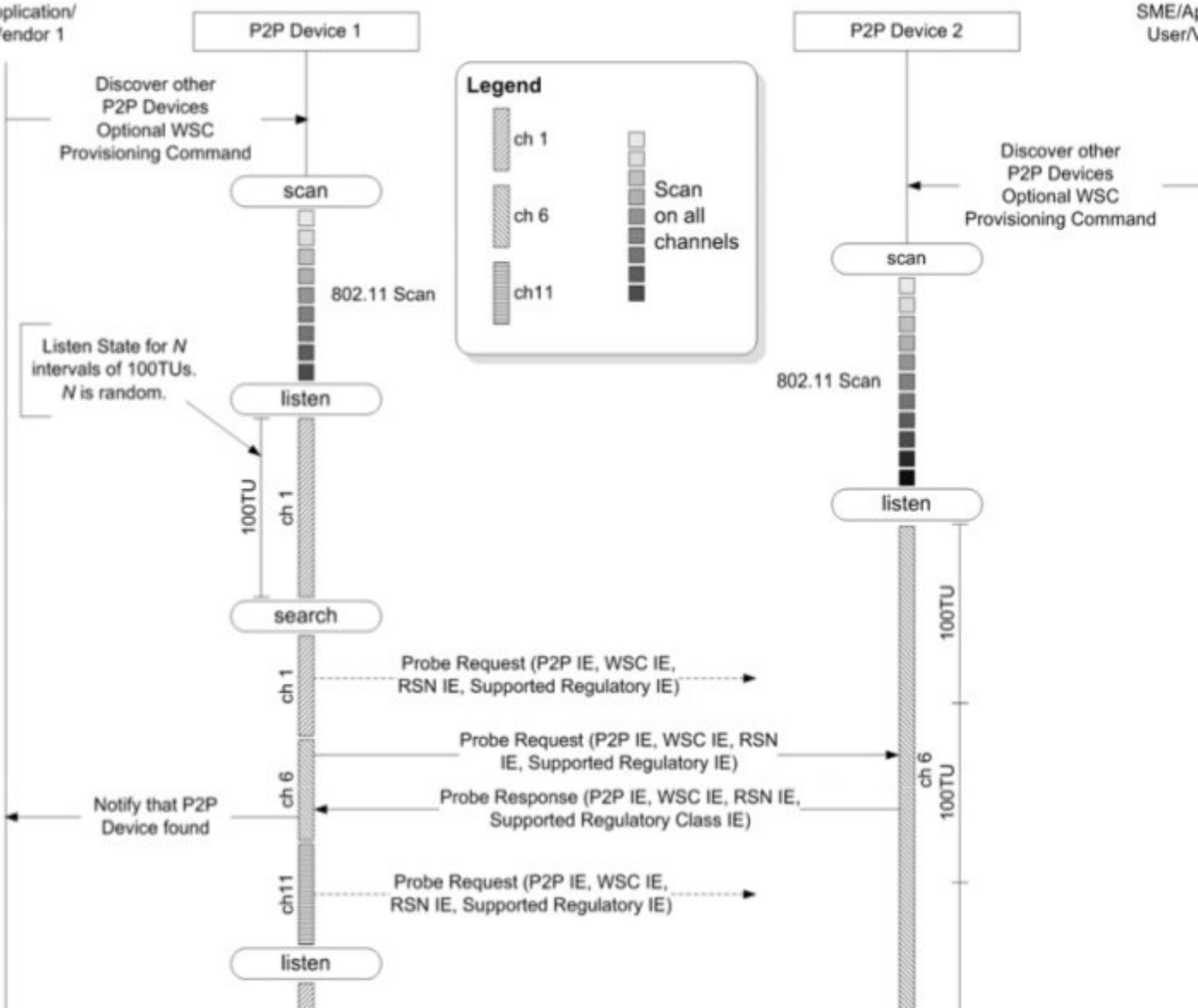
- ▶ Especificação do WiFi Alliance
- ▶ Diferente do modo Ad Hoc do 802.11
- ▶ Utiliza dos fundamentos do modo infraestruturado do 802.11

# WiFi P2P

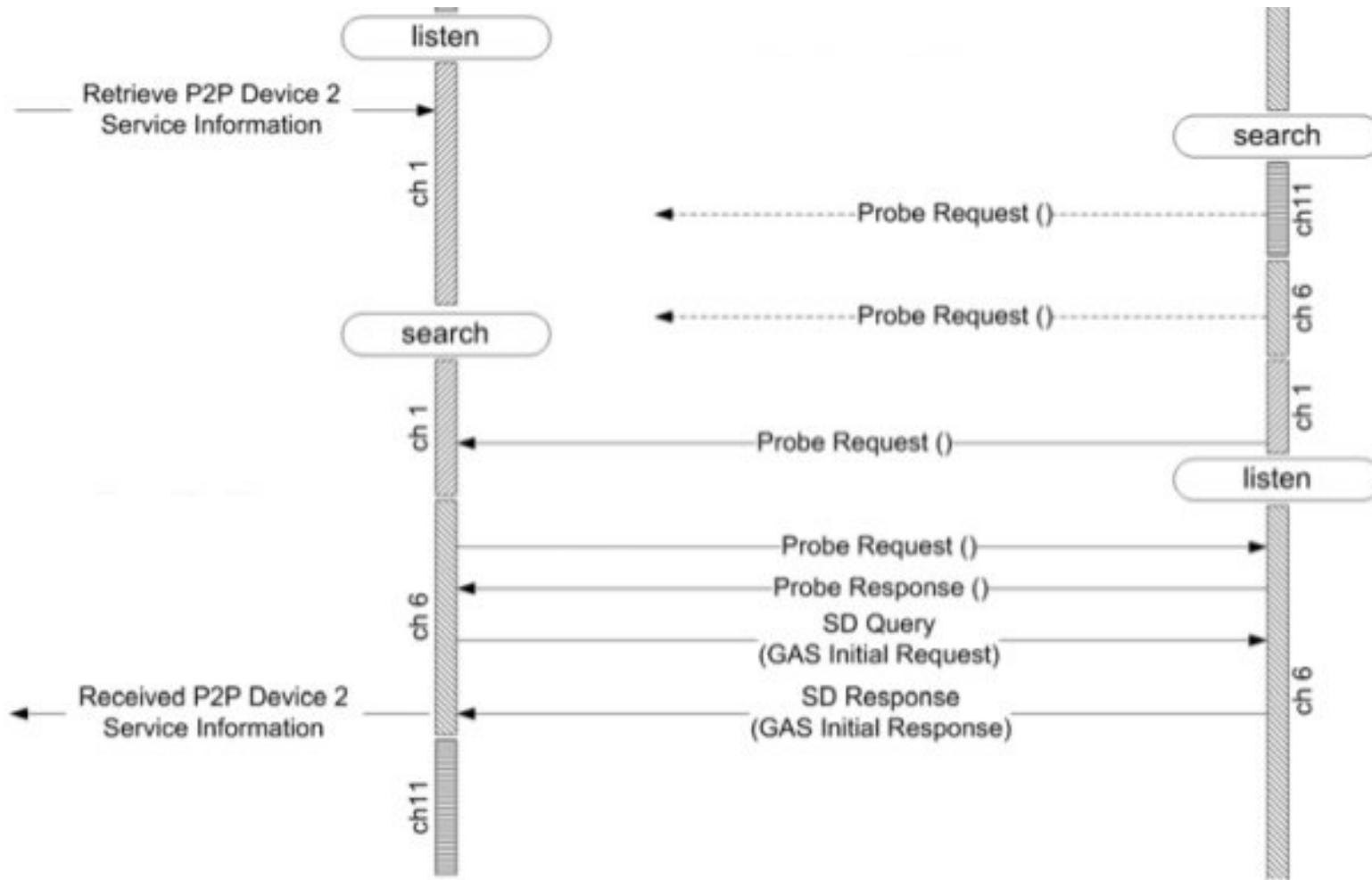
- ▶ Implementado em dispositivos móveis
- ▶ Descoberta de serviços próximos
- ▶ Construída sobre alicerce maduro

# WiFi P2P – Elementos

- ▶ P2P Devices
  - Dispositivos que implementam os papéis Cliente e GO
- ▶ P2P Cliente
  - Implementa o Cliente STA
- ▶ P2P GO
  - Implementa funcionalidade equivalente ao AP



# WiFi P2P – Descoberta de Serviços



# WiFi P2P – Negociação de grupo

Device

P2P D

## GO Negotiation Request

(The P2P IE includes P2P Capability, P2P Device Info, Group Owner Intent, Configuration Timeout, Listen Channel, Extended Listen Timing, Intended P2P Interface Address, Channel List, and Operating Channel attributes. The WSC IE includes Device Password ID attribute)



## GO Negotiation Response

(The P2P IE includes P2P Capability, P2P Device Info, Group Owner Intent, Configuration Timeout, Intended P2P Interface Address, Channel List, and Operating Channel attributes. The WSC IE includes Device Password ID attribute)



## GO Negotiation Confirmation

(The P2P IE includes P2P Capability, Status, Channel List, and Operating Channel attributes)



# Organização

- ▶ Problema
  - Contextualização
  - Contribuição
- ▶ Referencial Teórico
- ▶ Relacionados
- ▶ Análise
- ▶ Proposta
- ▶ Resultados
- ▶ Conclusão e Trabalhos Futuros

# Artigos Relacionados

Li - Utiliza um esquema por ID com uso mínimo de mensagens

LCC - Constroi clusters ao redor do menor ID,

GAME - Utiliza vários parâmetros

MCFA - Utiliza Mobilidade como métrica para formação

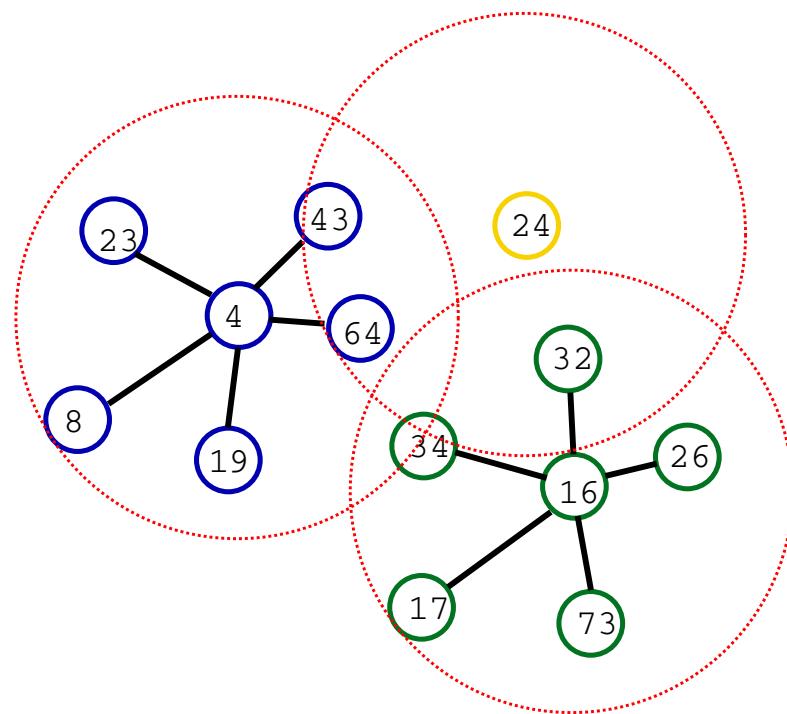
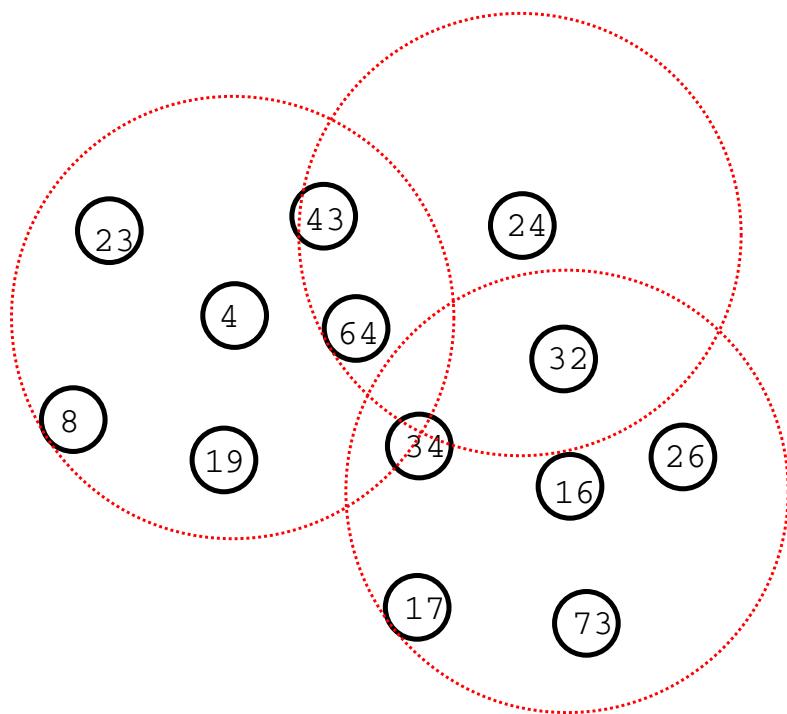
# Artigos Relacionados

- ▶ Li (Li & Dai 2009)
  - Uso do ID como contador decrescente
  - Uma mensagem para inicio de formação por cluster
  - Não lida com a mobilidade
  - Não permite fusão

# Artigos Relacionados

- ▶ LCC de Lin (Lin & Gerla 1997)
  - Nós publicam seu ID
  - Nó que escuta IDs maiores que o seu é o Head
  - Não lida com a mobilidade
  - Permite a fusão de clusters

# LIC/LCC



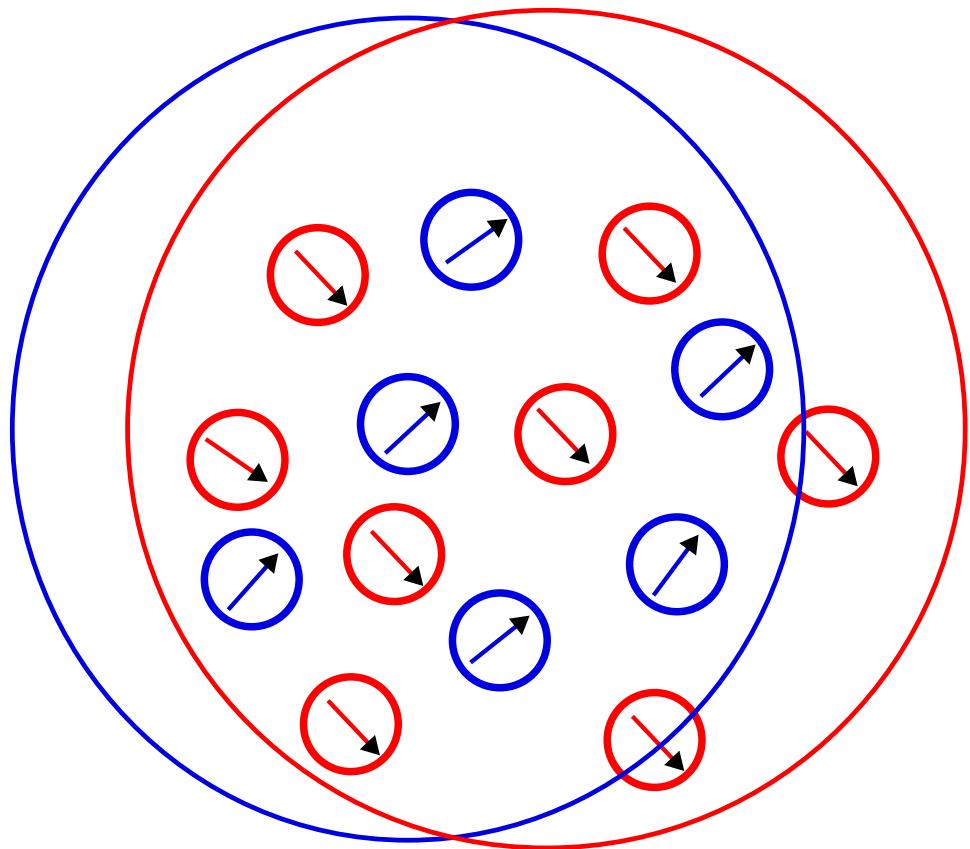
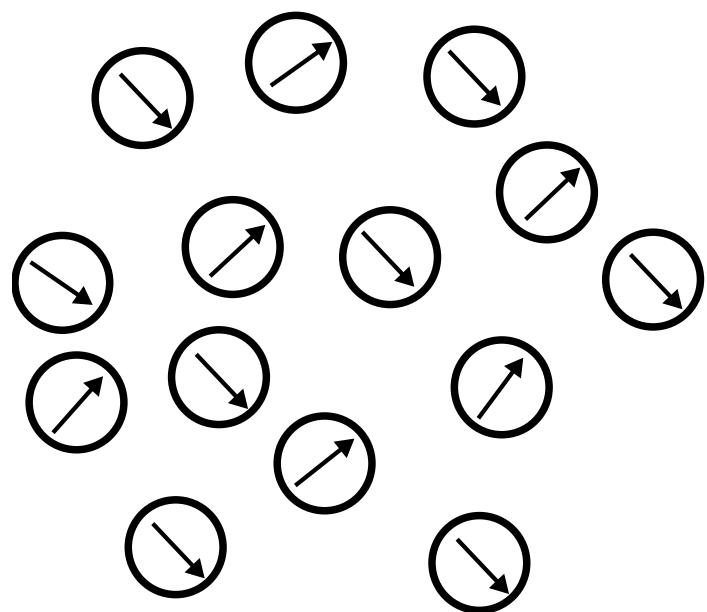
# Artigos Relacionados

- ▶ Método GAME de Sun e Sun (hui Sun, Sun, qing Yang & jiu Xu 2011)
  - Mais recente
  - Utiliza uma métrica combinada com pesos
  - Lida com a mobilidade
  - Permite a fusão

# Artigos Relacionados

- ▶ MCFA de Akbari (Akbari Torkestani & Meybodi 2011)
  - Baseado em mobilidade
  - Utiliza um Automato
  - Utiliza a mobilidade
  - Não permite fusão

# MCFA



# Organização

- ▶ Problema
  - Contextualização
  - Contribuição
- ▶ Referencial Teórico
- ▶ Relacionados
- ▶ Análise dos Algoritmos
- ▶ Proposta
- ▶ Resultados
- ▶ Trabalhos Futuros

# Análise

- ▶ Classificação baseada no modelo de Yu (Yu e Chong 2005)
- ▶ Adicionado parâmetro de possibilidade de fusão de clusters

# Parâmetros

Parâmetro	Descrição
Uso de mensagens de controle específicas	Requer mensagens específicas para a formação do cluster. Não pode ser formado com mensagens de outra natureza como dados.
Efeito Cascata	Indica que reeleger um único líder pode afetar a estrutura de varios outros clusters e alterar a topologia de toda a rede
Assume nos estaticos na formação	Os nós devem ser assumidos como estaticos na fase de formação para que informação precisa de vizinhança seja obtida
Ciclo de computação constante	O número de vezes que o processo de formação deve ser executado para completar.
Fusão de Clusters	Utiliza alguma criterio para fundir dois clusters em um só

# Comparação dos Algoritmos

Algoritmo	LCC	Li	GAME	MCFA	GLUE
Uso de mensagens específicas	Sim	Sim	Sim	Sim	Sim
Assume nós estáticos na formação	Sim	Sim	Não	Não	Não
Efeito Cascata	Sim	Não	Não	Não	Não
Fase de formação de cluster	Estática	Estática	Estática	Recursiva	Recursiva
Permite fundir clusters	Sim	Não	Sim	Não	Sim
Uso de mobilidade na manutenção	Não	Não	Sim	Sim	Sim

# Organização

- ▶ Problema
  - Contextualização
  - Contribuição
- ▶ Referencial Teórico
- ▶ Relacionados
- ▶ Análise dos Algoritmos
- ▶ Proposta
- ▶ Resultados
- ▶ Conclusão e Trabalhos Futuros

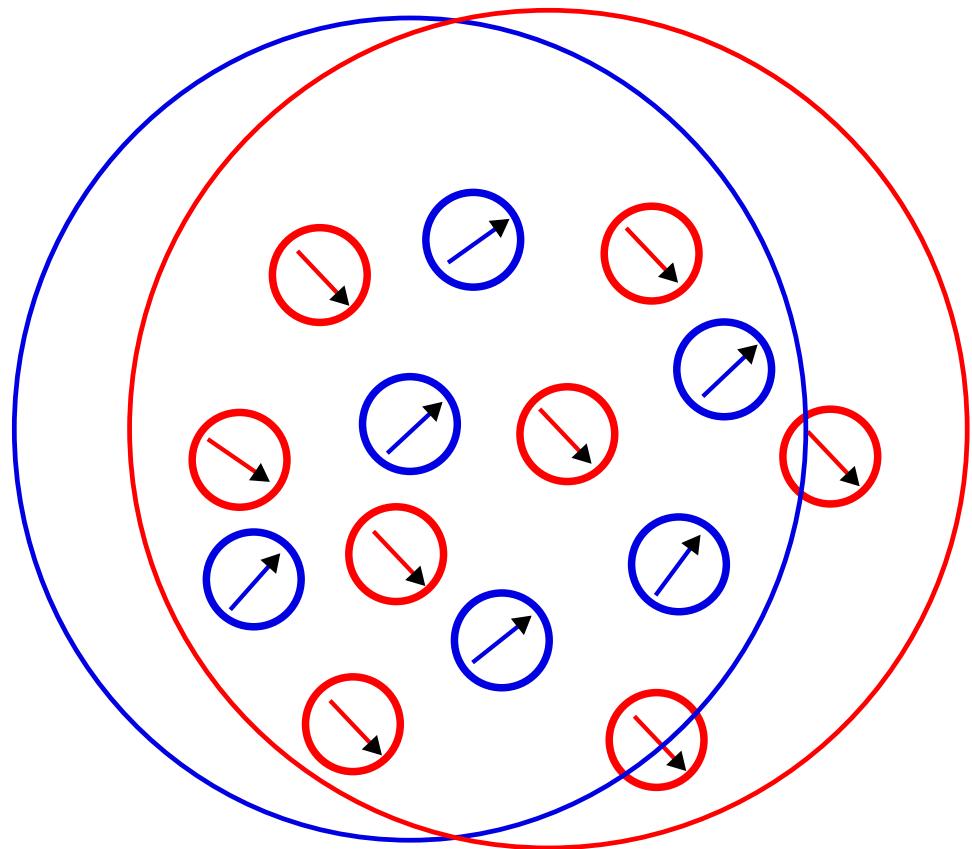
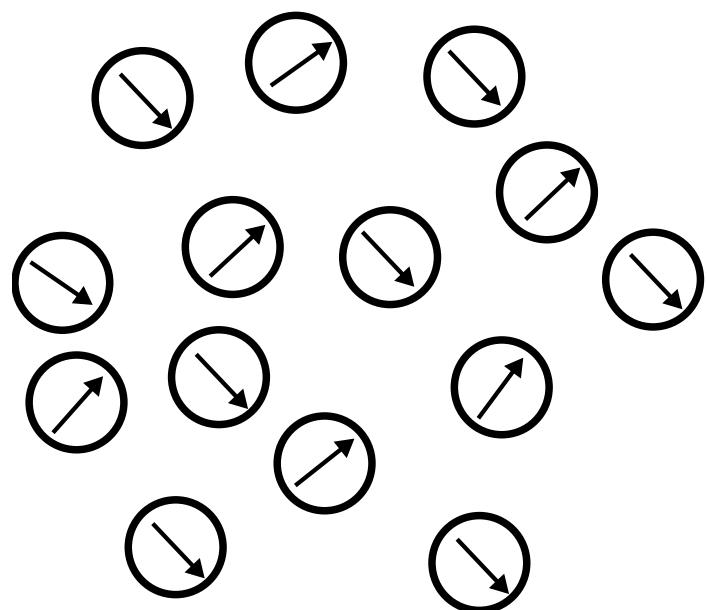
# GLUE

- ▶ Algoritmo Cluster com Líder (Cluster Head)
- ▶ Formação
- ▶ Manutenção

# Mensagens

Mensagem	Uso
MOBILITY UPDATE	Atualizaçāo de Mobilidade e informação de fusão
ASFREQ	Formação. Descoberta
RERM	Pedido de Mobilidade Relativa
CHSEL	Notificação de escolha de líder
JREQ	Pedido de ingresso em cluster
LREQ	Notificação de abandono de cluster

# GLUE – Formação



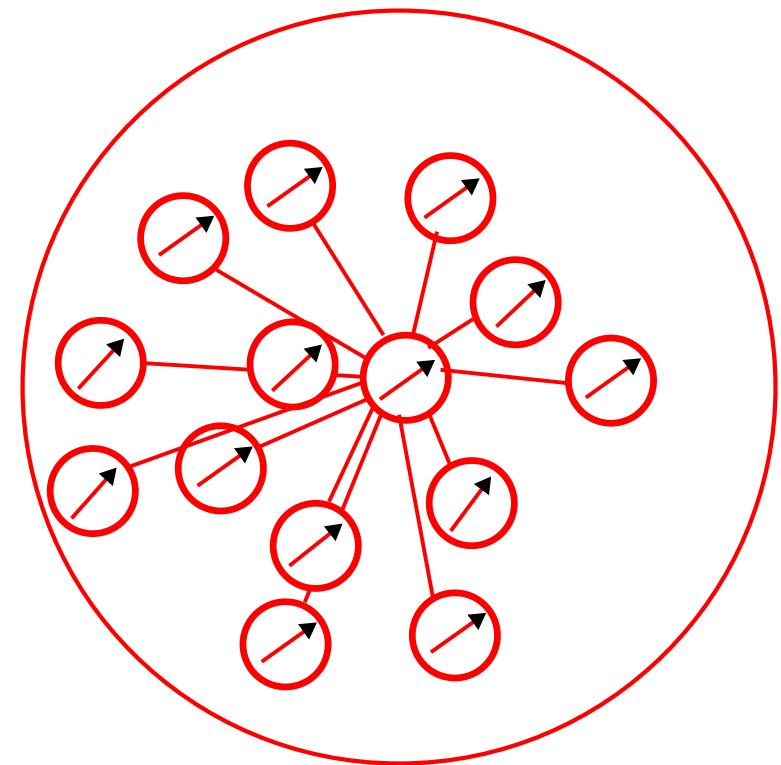
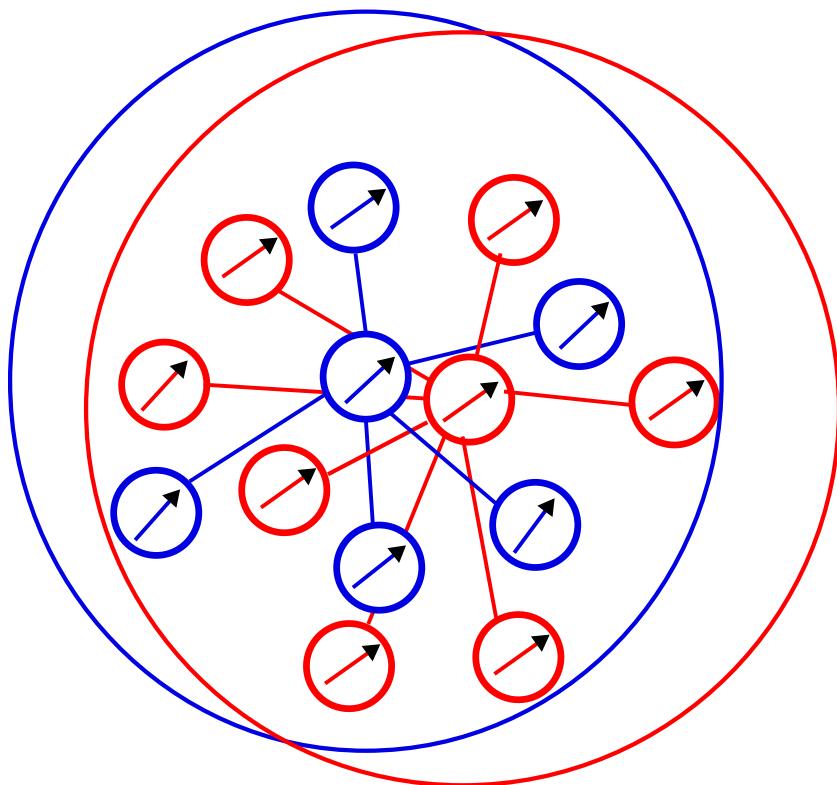
# Formação

- ▶ Formação de ActionSet
  - Mensagens de Mobilidade
  - Formação do ActionSet
  - Cálculo de Mobilidade Agregada
- ▶ Eleição de Líder
  - Requisição de Mobilidade
  - Eleição iterativa utilizando Autômato

# Manutenção

- ▶ Perda de conexão
  - Join
  - Reinício
- ▶ Sobreposição de Mobilidade
  - Fusão do Glue

# GLUE – Fusão



# Fusão

- ▶ Princípios
  - Manutenção pela mobilidade
  - Sobreposição Física e Móvel inicia fusão
  - Sobreposição somente física ou móvel não altera o cluster

# Fusão

- ▶ Nós Cluster Head enviam parâmetros de fusão na mensagem MOBILITY\_UPDATE
  - *ClusterID*
  - Lista de Nós
- ▶ Nós Cluster Head que recebem mensagem de mobilidade de outro Cluster Head verificam se estão sobrepostos por nós e mobilidade

# Verificação de Sobreposição

- ▶ Verificação de Fusão de Heads
  - Velocidade Relativa ( $RM_{(i;j)}$ ) entre os Heads  $< M$  (valor arbitrário)
  - Nós vizinhos são os mesmos  $.N_i = N_j$
  - Verificar se o  $ERM_{(i)}$  maior que o nó originário

# Fusão

- ▶ Se um nó Cluster Head detecta sobreposição e mobilidade relativa maior, ele desiste do papel enviando um LREQ
- ▶ Nós que recebem a mensagem LREQ procuram se afiliar ao cluster mais próximo com menor mobilidade

# Organização

- ▶ Problema
  - Contextualização
  - Contribuição
- ▶ Referencial Teórico
- ▶ Relacionados
- ▶ Análise dos Algoritmos
- ▶ Proposta
- ▶ **Resultados**
- ▶ Trabalhos Futuros

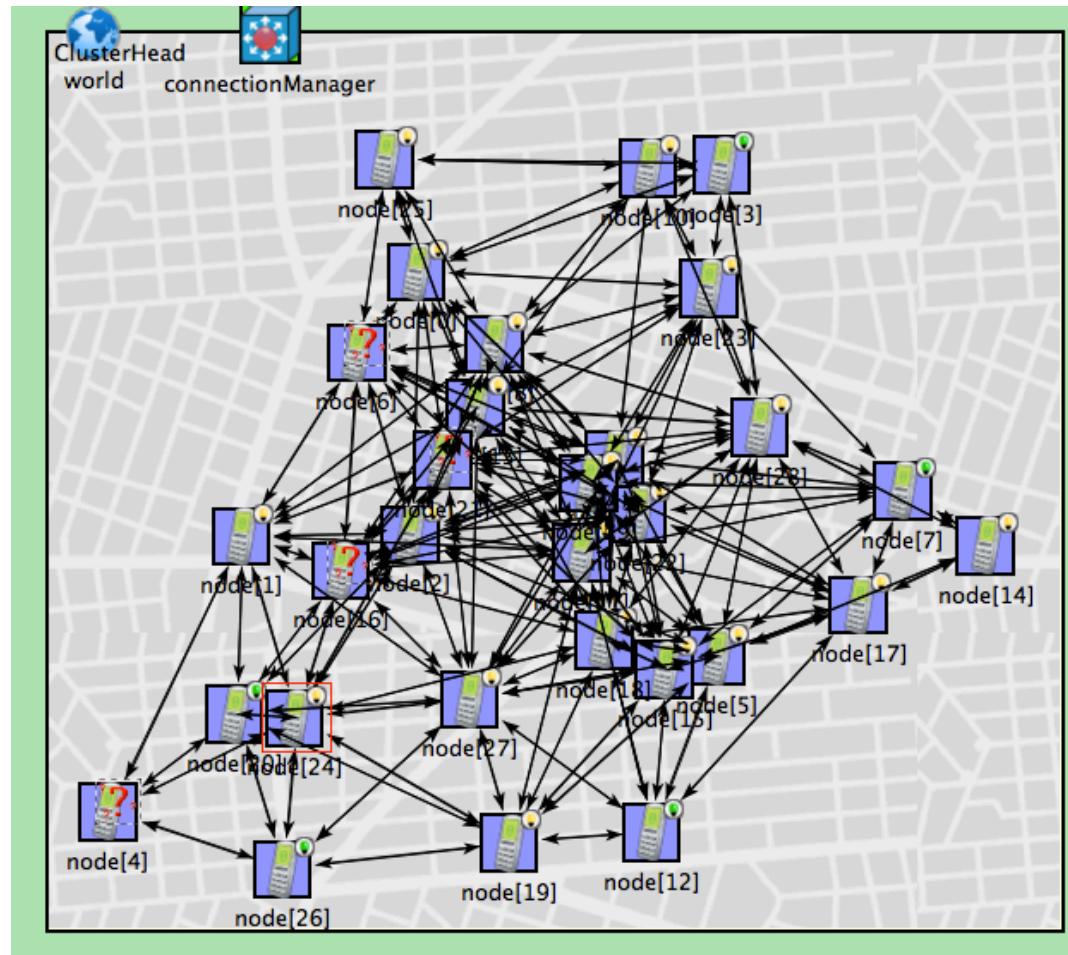
# Resultados

- ▶ Ambiente de Testes
- ▶ Resultados Simulação
- ▶ Teste de Viabilidade

# Ambiente de Testes

- ▶ Omnet++ 4.2
  - Simulador Discreto de eventos
  - Livre para Academia
- ▶ MiXiM
  - Suite de Mobilidade e Redes Sem Fio
- ▶ BonnMotion
  - Gerador de Padrões de Mobilidade

# Visualizador TK



# Padrões de Mobilidade

- ▶ **RWP – Random Waypoint**
  - Movimento aleatório sobre caminhos pré-estabelecidos
- ▶ **RPGM – Reference Point Group Model**
  - Movimento de caminhos seguindo líder, comportamento de cooperação
- ▶ **RWP+RPGM**
  - Dois modelos combinados

# Parâmetros de Simulação

- ▶ Testes com 30, 45, 60, 75, 90 nós.
- ▶ 30 Execuções
- ▶ Ambiente de 500mx500m
- ▶ Velocidade de 0 m/s à 2 m/s
- ▶ Condição de parada para a formação MCFA/ GLUE 0.9
- ▶ Limiar de fusão para o GLUE 0.1

# Resultados

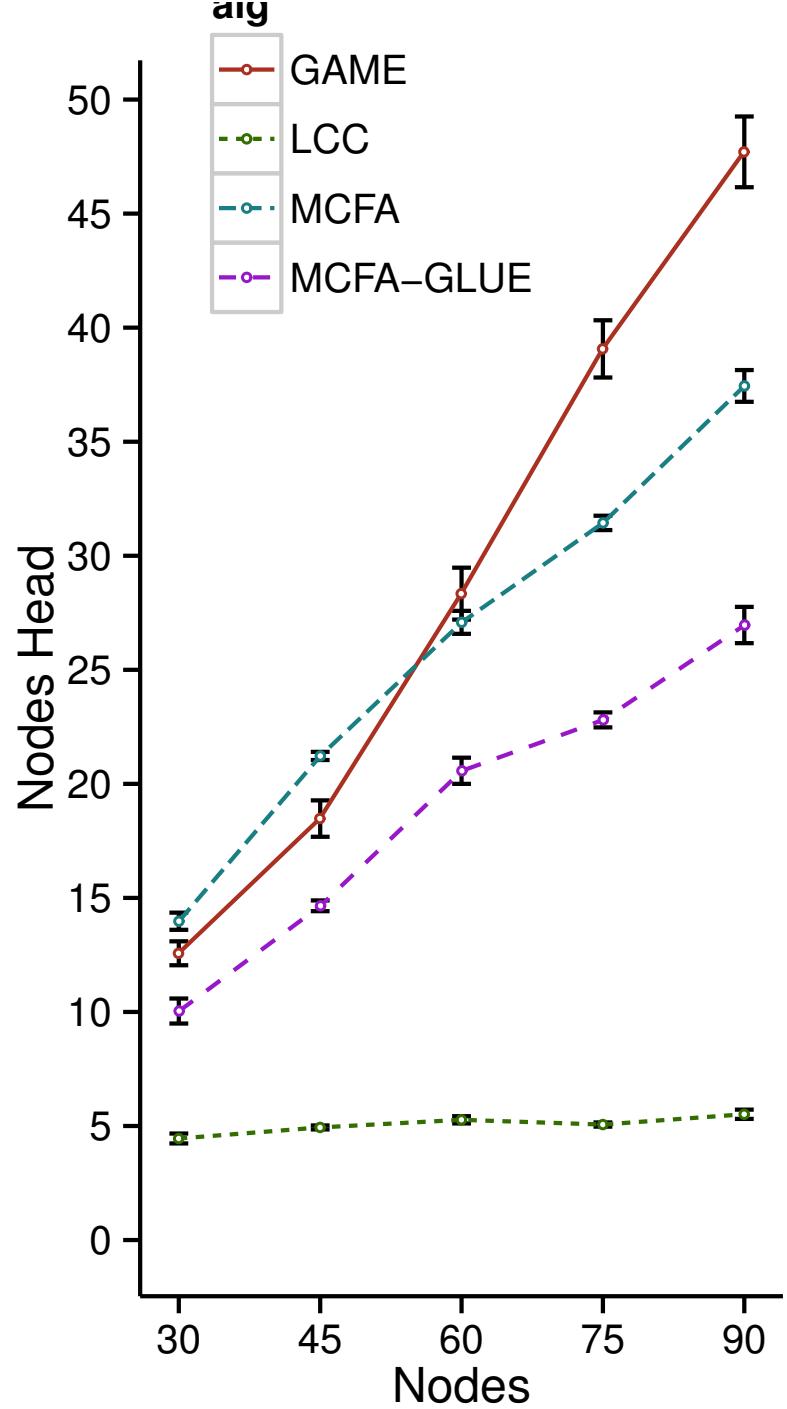
- ▶ Ambiente de Testes
- ▶ Resultados Simulação
- ▶ Teste de Viabilidade

# Resultados

- ▶ Número de clusters Formados
  - Análise do número médio de nós no estado Cluster Head
  - Mostra a eficiência do algoritmo
  - 50 segundos é o tempo de estabilização
- ▶ Desconexões por segundo
  - Estabilidade do cluster
  - Medido nas mudanças de estado para *UNDEFINED*

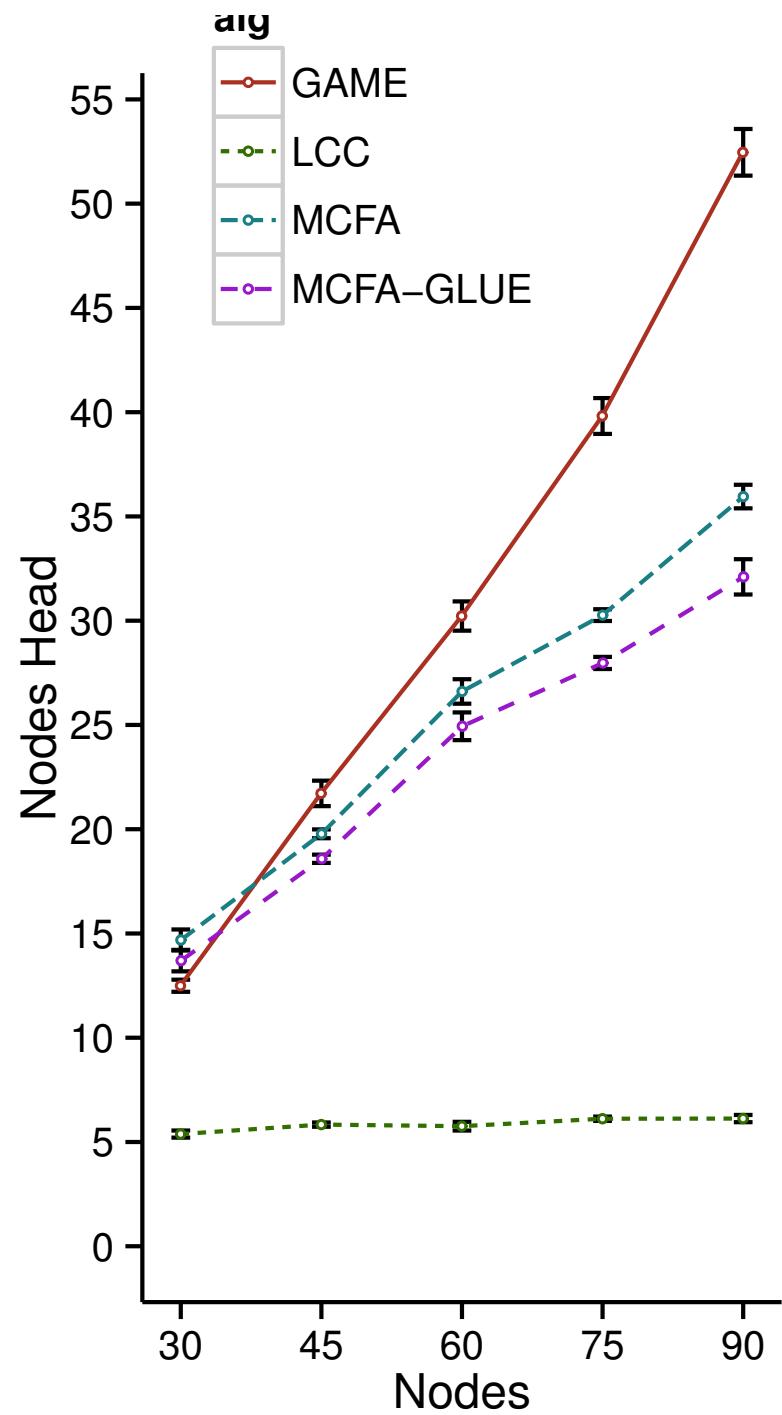
# Clusters Médios

Clusters formados  
utilizando  
Modelo RPGM



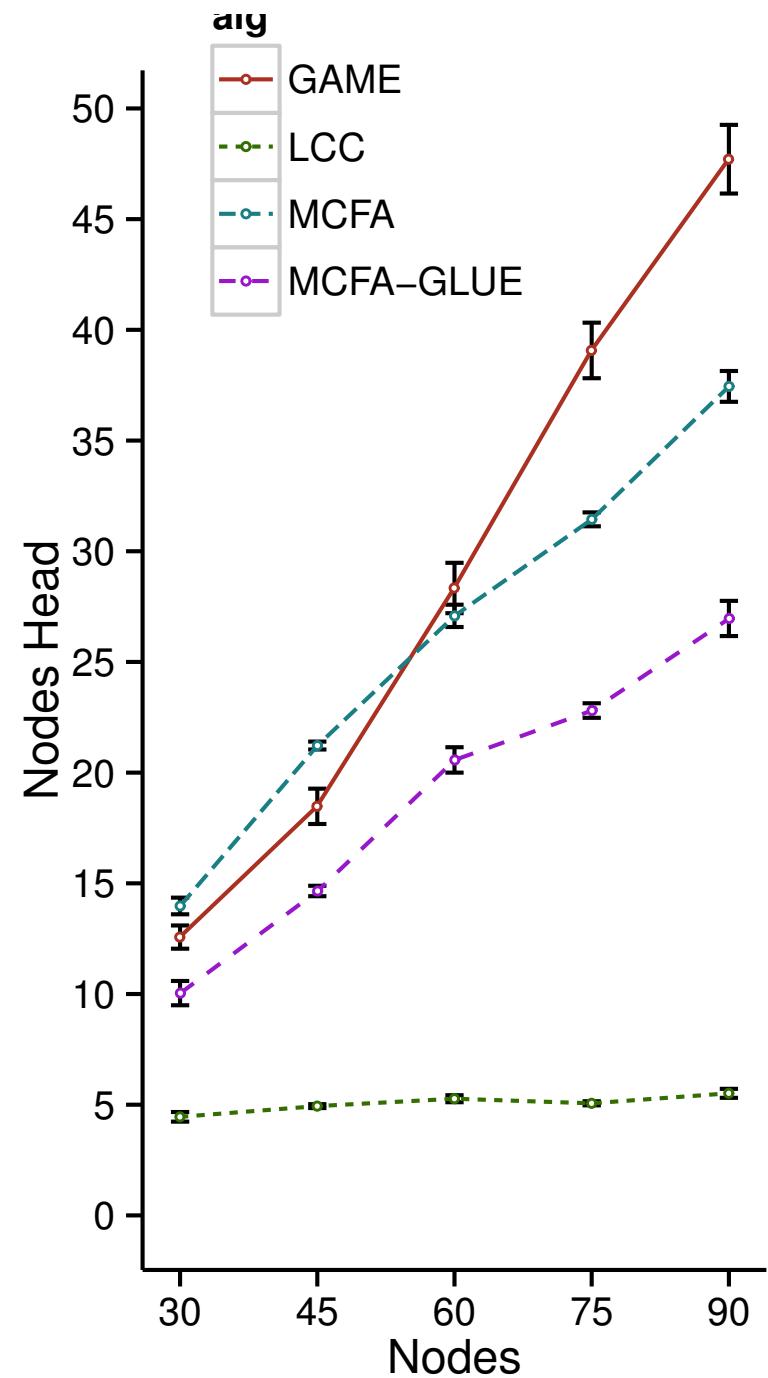
# Clusters Médios

Clusters formados  
utilizando  
Modelo RWP



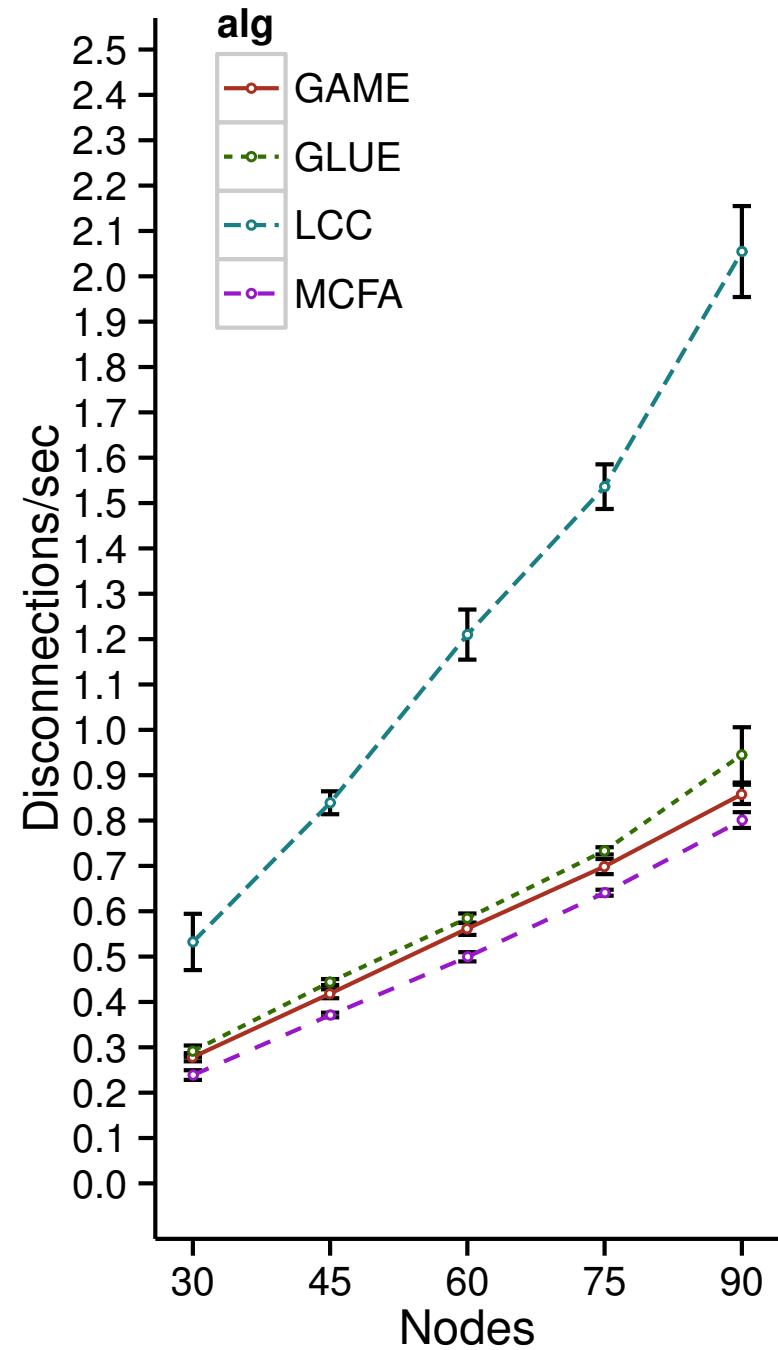
# Clusters Médios

Clusters formados  
utilizando  
RPGM e RWP  
Combinados



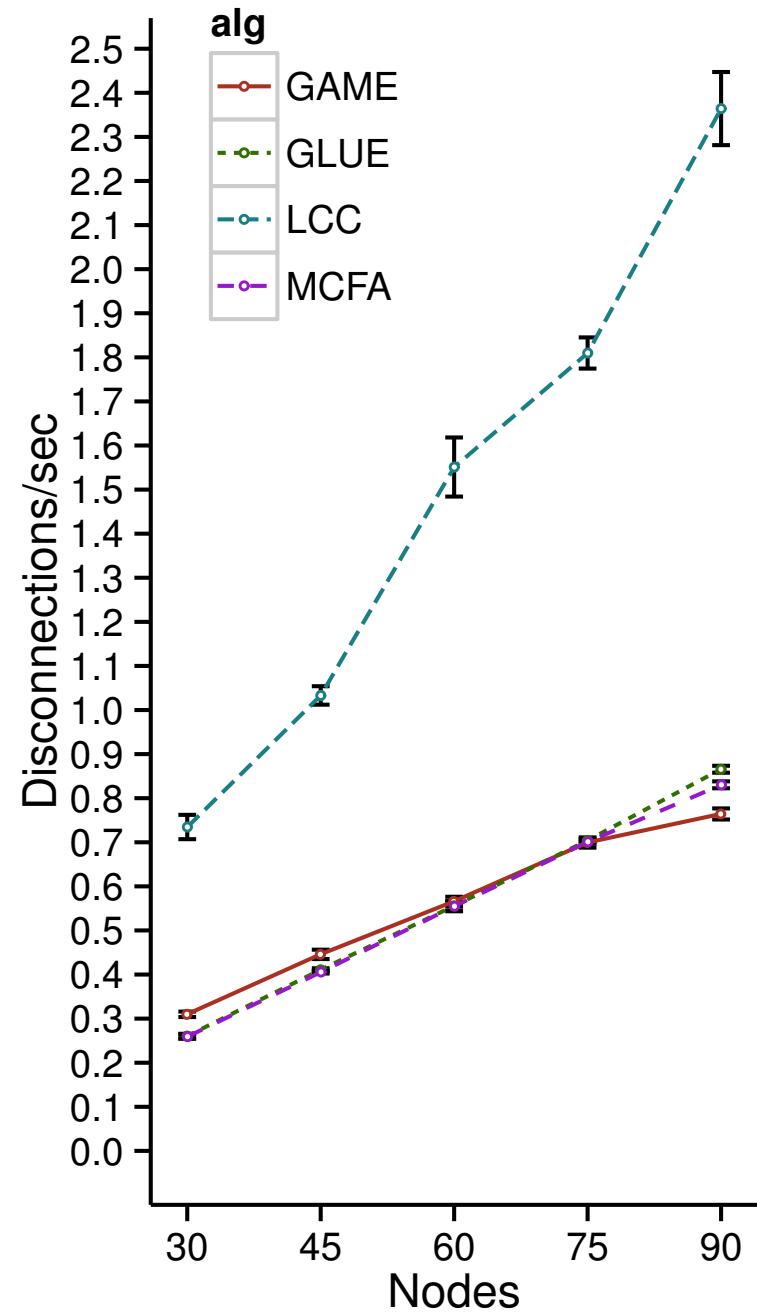
# Desconexões

Desconexões/sec  
utilizando  
Modelo RPBM



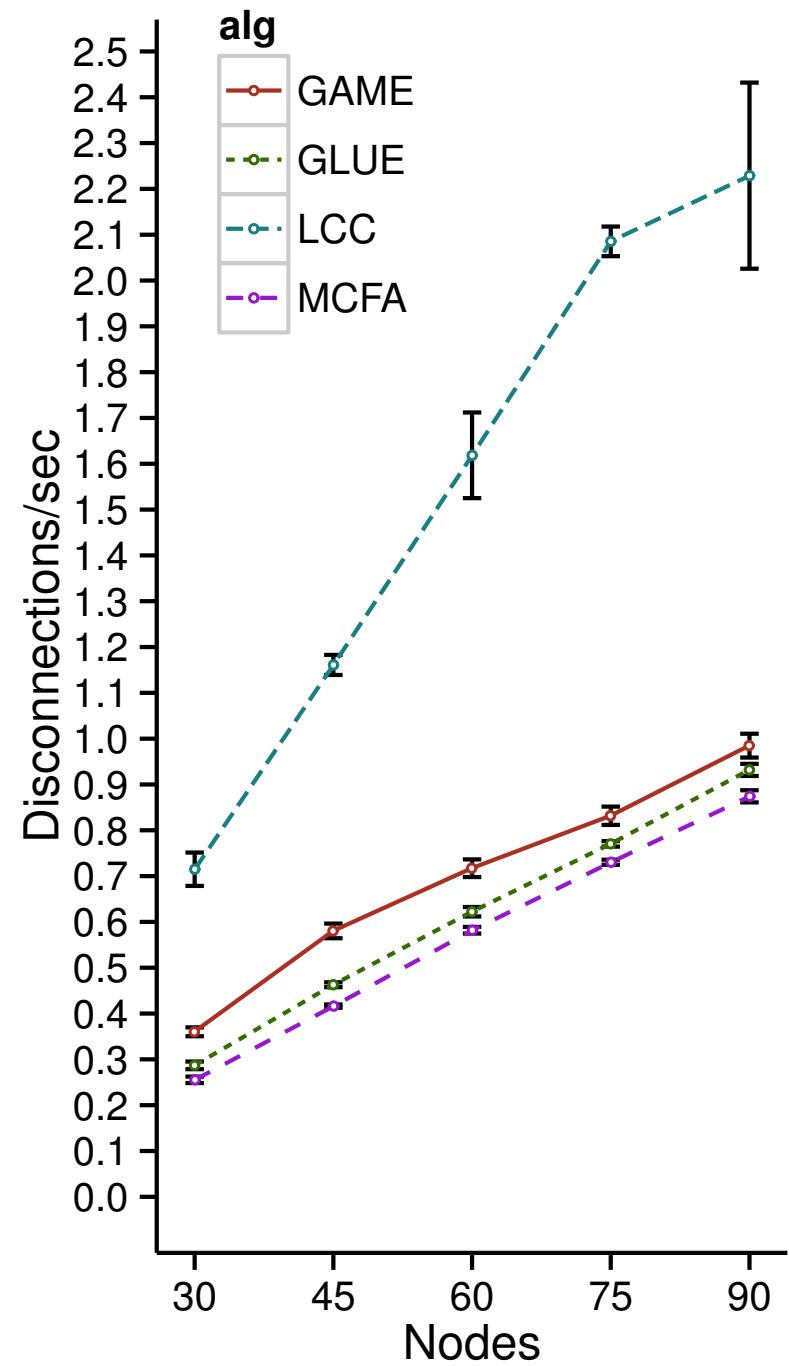
# Desconexões

Desconexões/sec  
utilizando  
Modelo RWP



# Desconexões

Desconexões/s  
utilizando  
RPGM e RWP  
Combinados



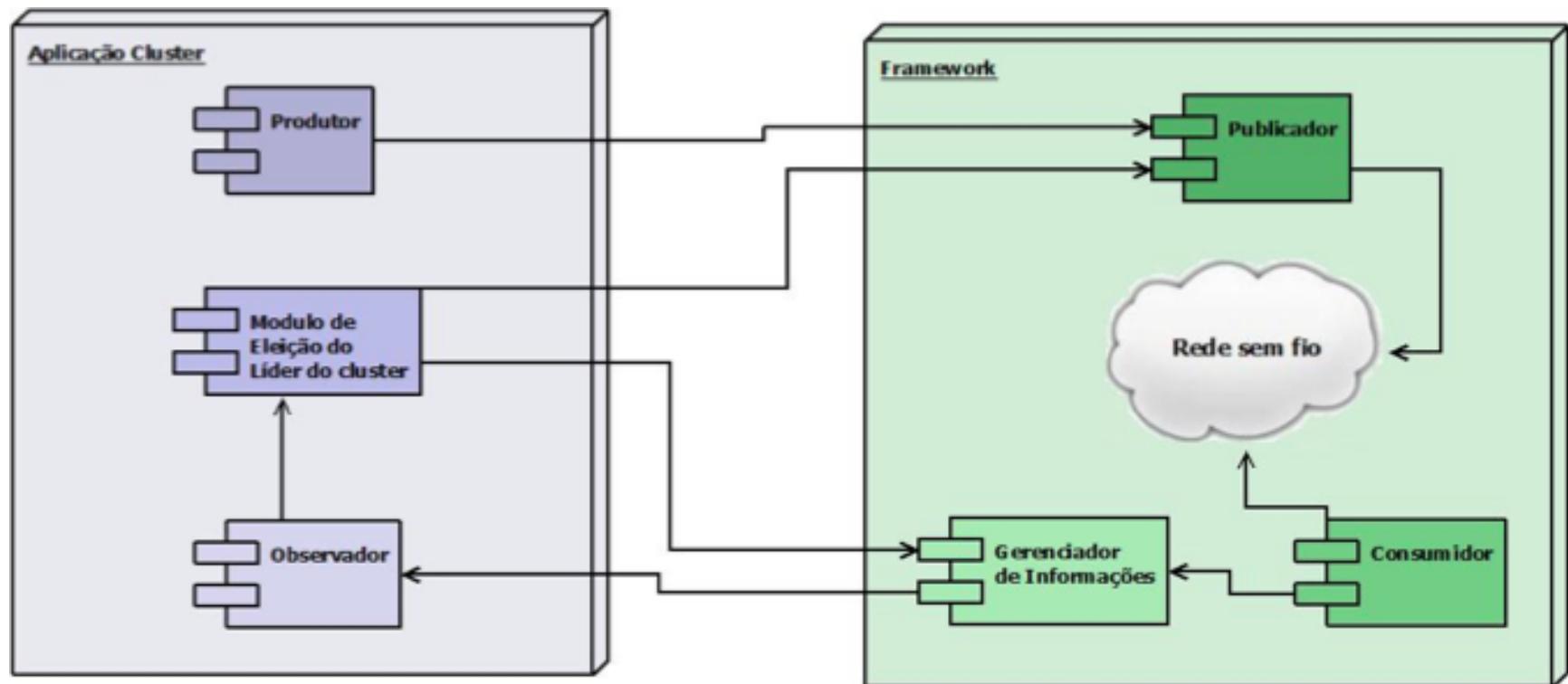
# Resultados

- ▶ Ambiente de Testes
- ▶ Resultados Simulação
- ▶ **Teste de Viabilidade**

# Teste de Viabilidade

- ▶ Implementação em APK Android
- ▶ Hardware Comodity
- ▶ Implementado sobre a notificação de serviços
- ▶ Implementado sobre Framework WiFi Direct iMobilis
- ▶ Testes mostram a viabilidade, ainda é necessário maturidade

# Framework



# Resultados Teste de Viabilidade

Movimentação	Número de Tablets	Número de Clusters
Parado	2	2
Movimentando	2	2
Parado	3	3
Movimentando	3	2
Parado	4	4
Movimentando	4	2

# Organização

- ▶ Problema
  - Contextualização
  - Contribuição
- ▶ Referencial Teórico
- ▶ Relacionados
- ▶ Análise dos Algoritmos
- ▶ Proposta
- ▶ Resultados
- ▶ **Conclusão e Trabalhos Futuros**

# Conclusão

- ▶ O Algoritmo do GLUE permitiu a fusão de clusters utilizando uma sobreposição de mobilidade.
- ▶ O número de clusters formado foi o segundo menor sem perder a estabilidade

# Trabalhos Futuros

- ▶ Aprofundar na implementação do GLUE utilizando o Framework do WiFi Direct
- ▶ Implementar o Glue diretamente na formação do grupo do WiFi Direct