Socially Aware Routing Algorithms for Context Aware Cognitive Radio

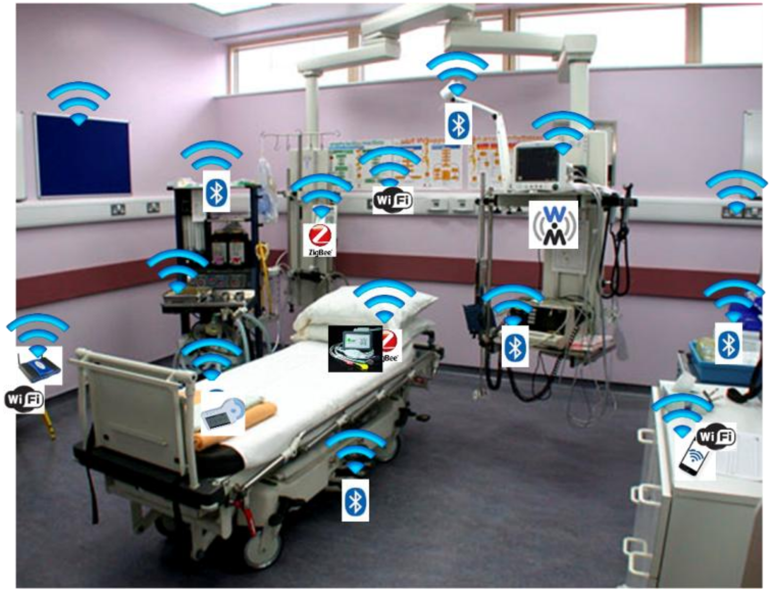
James Neel
james.neel@crtwireless.com
Cognitive Radio Technologies, LLC
March 23, 2015
WInnComm 2015

Messages

- Significant redundancy and inefficiencies in MANET routing
- Should be able to improve existing algorithms by incorporating context
- Adapt algorithms to the situation

What is Context?

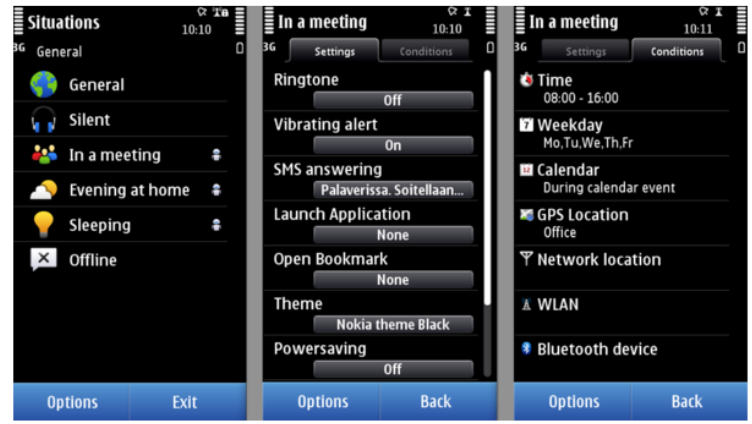
- **Context**:
  - the parts of [communications] not directly communicated that influence its meaning or effect (Modified from dictionary.com)
  - any information that can be used to characterize the situation of an entity (Dey)
- **Situations**
  - external semantic interpretations of context
  - objects having properties and standing in relations to one another
- **Episodes**
  - Situations in time



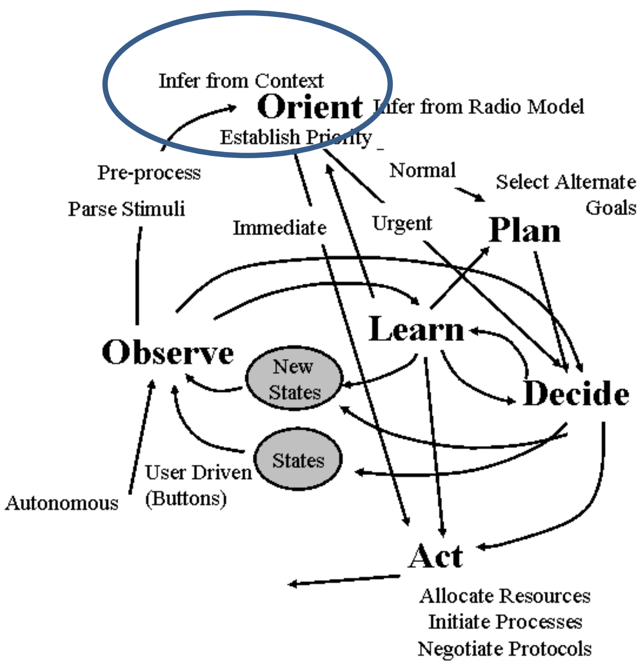
Context Aware Cognitive Radio

- Leverage information beyond normal wireless chipset metrics
- Adapt wireless parameters based on:
  - Social network considerations, mission objectives, shifting user objectives, disaster response policies

Situations from Nokia

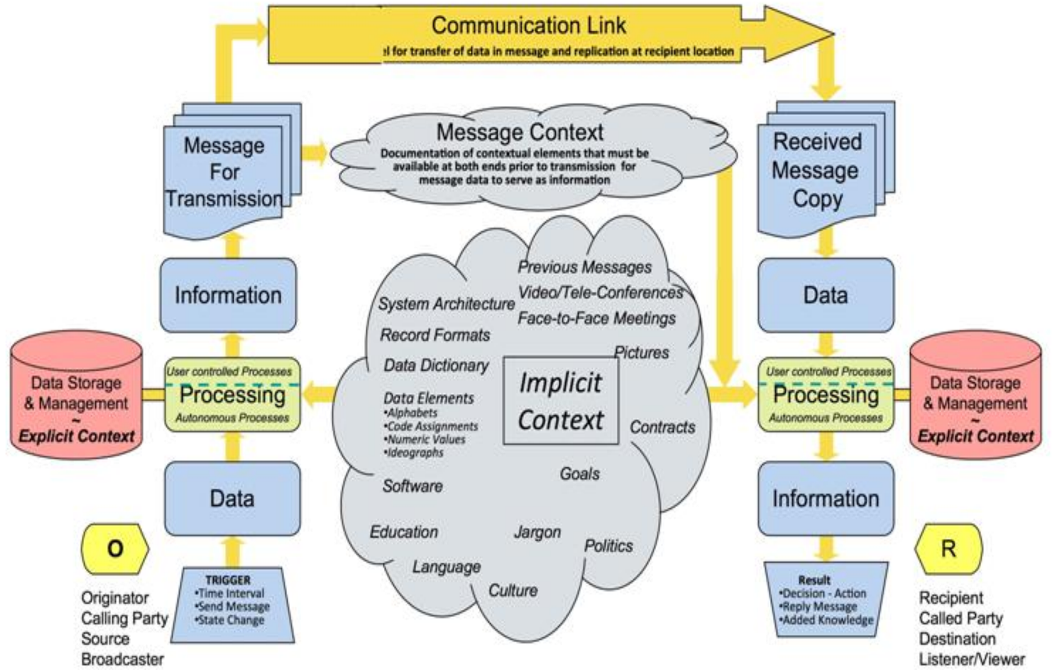


http://www.readwriteweb.com/archives/nokias_new_situations_app_makes_phones_self-aware.php



Context in Communications

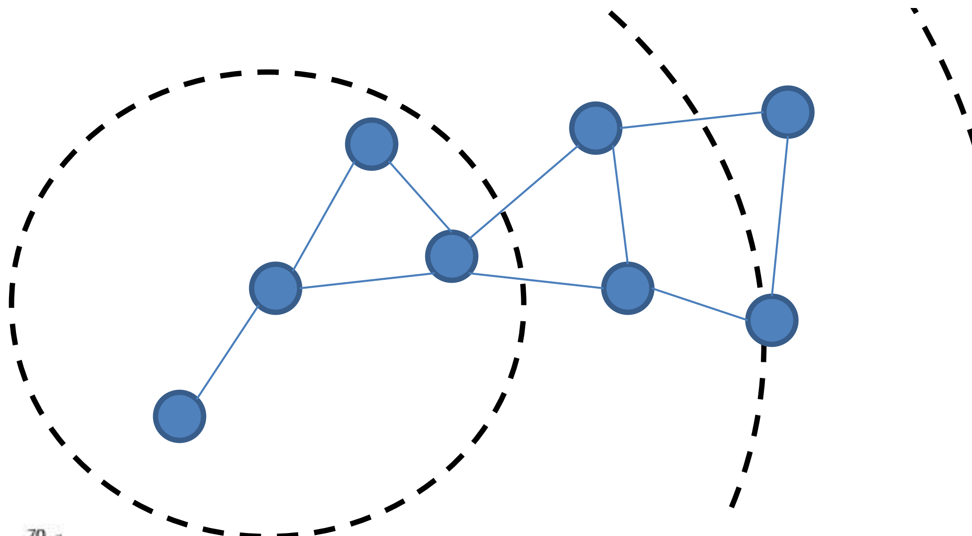
- Related insights from previous work (IPA v1,2):
  - Shared context between sender and receiver is critical to communications
  - Knowing you have a shared context can greatly reduce bandwidth requirements
  - Open problem of how a cognitive radio “knows” and shares its operational context
- Significant overhead in routing as networks scale

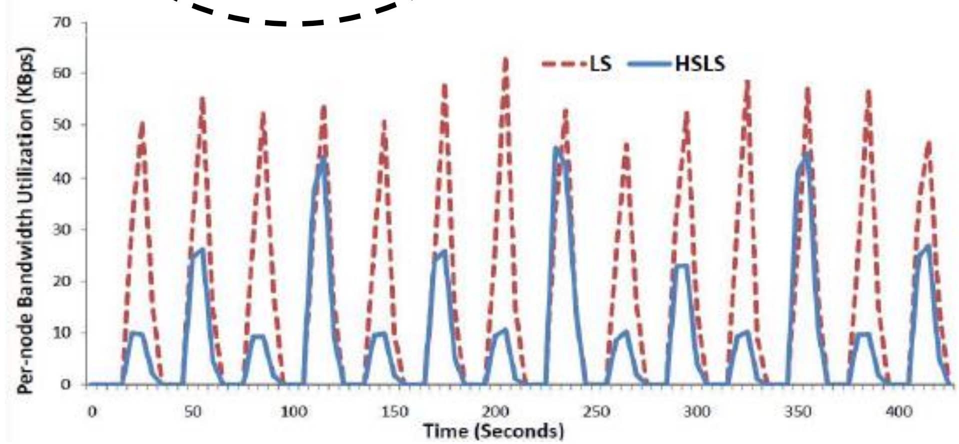


IPA Comm Cycle

Hazy-Sighted Link State Routing

- Basic design
  - Send out link state updates when links fail
  - Send out updates at regular intervals
    - Intervals double when no links break until reach a limit
    - Doubles # of hops
- Other considerations
  - Detect routing loops
  - Unidirectional support
- Scalability ~ $O(N^{1.5})$



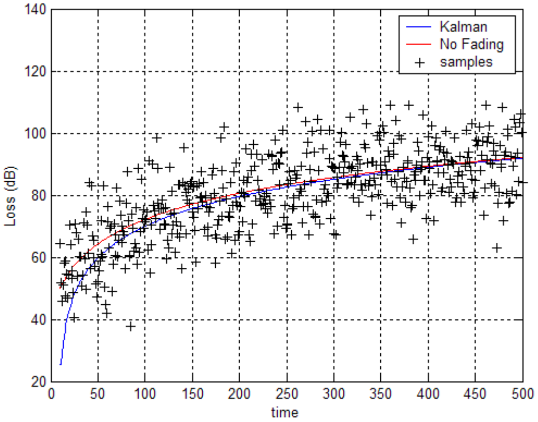


http://www2.research.att.com/~changbl/pubs/icnp09_talk.pdf

C.A. Santivanez, R. Ramanathan, "Hazy Sighted Link State (HSLS) Routing: A Scalable Link State Algorithm," BBN Technical Memorandum No. 1301, August 2001. Available at http://www.ir.bbn.com/documents/techmemos/TM1301.pdf

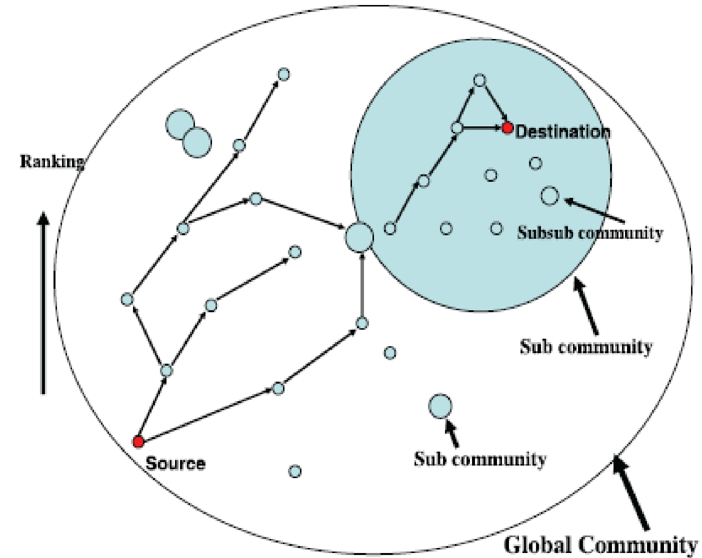
Higher-Order Hazy-Sighted Routing

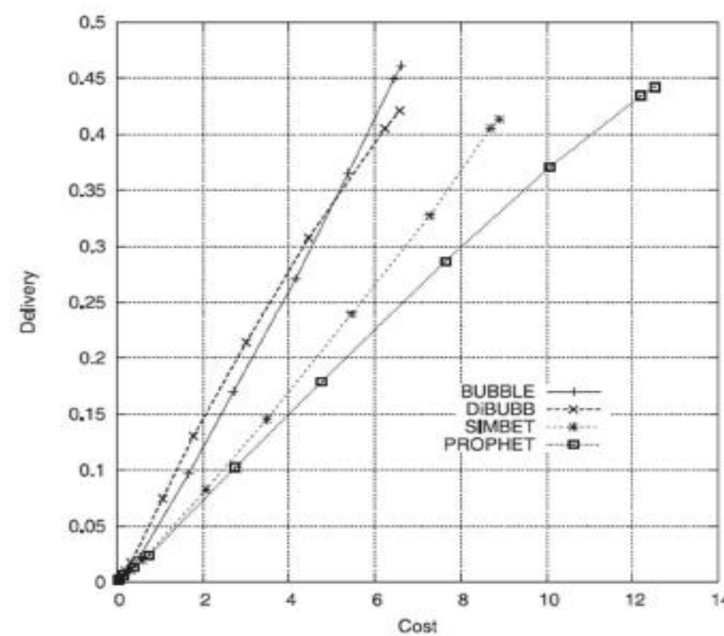
- Compression results were for a network running HSLS
- Predict with contextual information?
  - Wait out expected short interruptions
    - Queue data
  - Soft predictive re-routing when longer interruptions expected
- Prioritize based on social network
  - Combine with DTN techniques
- Build internal learning models and update only when deviates from model
  - Detect when nodes will be moving out of range in near future



Bubble-RAP for Disruption Tolerant Networks

- Assumptions:
  - Radios correspond to people
  - People tend to cluster
  - Some people are more interconnected (Kevin Bacon)
- Forward packet to better connected radios until reaches desired bubble
- Algorithms for learning bubbles and social ranks
- Do better:
  - With known social networks?
  - With known schedules or patterns of movement?

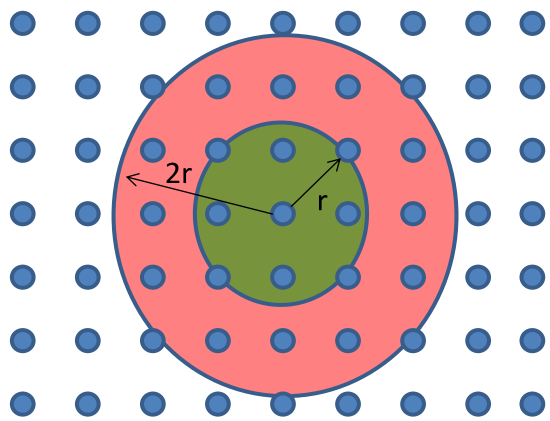




P. Hui, J. Crowcroft, E. Yoneki, "BUBBLE Rap: Social-Based Forwarding in Delay-Tolerant Networks," *IEEE Transactions on Mobile Computing*, vol. 10, , no 11, Nov 2011, pp. 1576-1589.

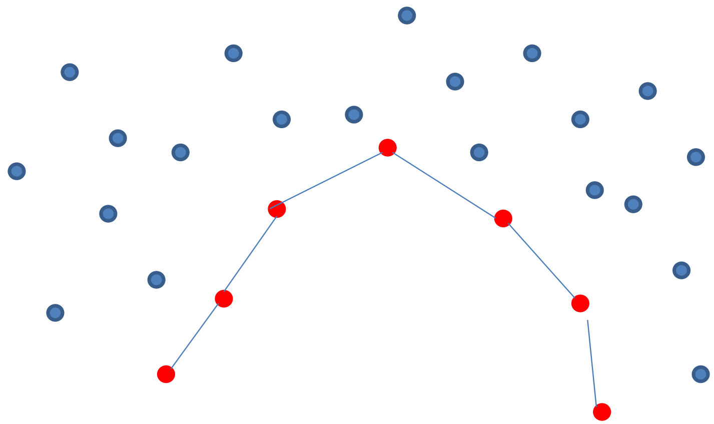
Exploiting Preferred Routes

- Capacity Problem

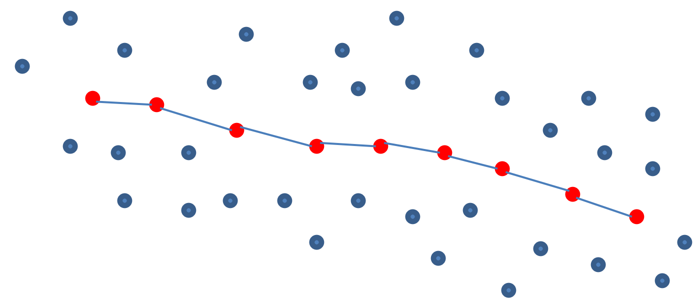


- Preferred Routes / Artery Nodes simplify routes and in some situations can increase capacity
  - Some situations decrease capacity!
- Topology analyses could reveal when would be effective

Artery Nodes Increase Capacity



Artery Nodes Decrease Capacity



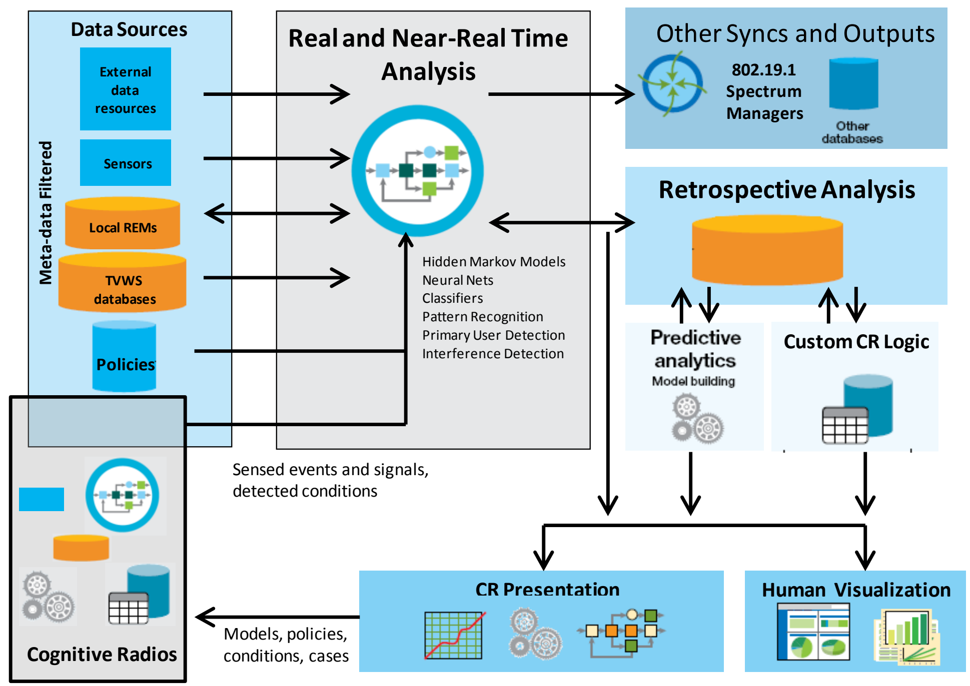
Matching Algorithm and Situation

- Different algorithms work better in different environments
- DTN algorithms for DTN environments
- Vary link metrics based on node mobility
  - Transmission counts vs RTT

TABLE IV

Even with all other parameters held constant, varying the Observations (O), Actions (A), Decision processes (D), Goals (G), or Context (C) can lead to radically different outcomes

			Scenario 1			Scenario 2
Observation	Parameters	O	*Interference at access point from other access points*	Parameters	O	*Interference seen by clients*
		A	Frequency (channel)		A	Frequency (channel)
		D	Lowest interference channel		D	Lowest interference channel
		G	Minimize interference		G	Minimize interference
		C	Tent city		C	Tent city
	Result		Converges to near-optimal frequency reuse pattern [48]	Result		Enters an infinite loop with probability 1 as network scales [51]
Actions	Parameters	O	SINR at cluster head	Parameters	O	SINR at cluster head
		A	*Frequency*		A	*Power*
		D	Maximize goal		D	Maximize goal
		G	Maximize SINR		G	Maximize SINR
		C	Isolated cluster		C	Isolated cluster
	Result		Network tends to converge to low interference states	Result		Network tends to converge to self-jamming states
Decisions	Parameters	O	Collisions	Parameters	O	Collisions
		A	Transmission times		A	Transmission times
		D	*Collaborate on times*		D	*Noncollaboratively choose times*
		G	Maximize collisions		G	Maximize collisions
		C	Isolated cluster		C	Isolated cluster
	Result		Rapid convergence to minimal interference state, adjustable to different user priorities	Result		Slow (if at all) convergence, throughput as low as ALOHA $(1/e)$
Goals	Parameters	O	SINR at cluster head	Parameters	O	SINR at cluster head
		A	Power		A	Power
		D	Maximize goal		D	Maximize goal
		G	*Target SINR*		G	*Maximize SINR*
		C	Isolated cluster		C	Isolated cluster
	Result		If target SINR is feasible, converges to target SINR [78]	Result		Network tends to converge to self-jamming states
Context	Parameters	O	SINR at cluster head	Parameters	O	SINR at cluster head
		A	Power		A	Power
		D	Punish (jam) radios deviating from target SINR		D	Punish (jam) radios deviating from target SINR
		G	Target SINR		G	Target SINR
		C	*Isolated cluster*		C	*Isolated cluster with a jammer*
	Result		Network overcomes defection problems for significant improvement in performance [41]	Result		Network self-jams as it "punishes" the jammer



Takeaways

- Significant redundancy and inefficiencies in MANET routing
- Should be able to improve existing algorithms by incorporating context
- Adapt algorithms to the situation

TABLE IV

EVEN WITH ALL OTHER PARAMETERS HELD CONSTANT, VARYING THE OBSERVATIONS (O), ACTIONS (A), DECISION PROCESSES (D), GOALS (G), OR CONTEXT (C) CAN LEAD TO RADICALLY DIFFERENT OUTCOMES

	Scenario 1			Scenario 2		
Observation	Parameters	O	*Interference at access point from other access points*	Parameters	O	*Interference seen by clients*
		A	Frequency (channel)		A	Frequency (channel)
		D	Lowest interference channel		D	Lowest interference channel
		G	Minimize interference		G	Minimize interference
		C	Tent city		C	Tent city
	Result		Converges to near-optimal frequency reuse pattern [48]	Result		Enters an infinite loop with probability 1 as network scales [51]
Actions	Parameters	O	SINR at cluster head	Parameters	O	SINR at cluster head
		A	*Frequency*		A	*Power*
		D	Maximize goal		D	Maximize goal
		G	Maximize SINR		G	Maximize SINR
		C	Isolated cluster		C	Isolated cluster
	Result		Network tends to converge to low interference states	Result		Network tends to converge to self-jamming states
Decisions	Parameters	O	Collisions	Parameters	O	Collisions
		A	Transmission times		A	Transmission times
		D	*Collaborate on times*		D	*Noncollaboratively choose times*
		G	Maximize collisions		G	Maximize collisions
		C	Isolated cluster		C	Isolated cluster
	Result		Rapid convergence to minimal interference state, adjustable to different user priorities	Result		Slow (if at all) convergence, throughput as low as ALOHA $(1/e)$
Goals	Parameters	O	SINR at cluster head	Parameters	O	SINR at cluster head
		A	Power		A	Power
		D	Maximize goal		D	Maximize goal
		G	*Target SINR*		G	*Maximize SINR*
		C	Isolated cluster		C	Isolated cluster
	Result		If target SINR is feasible, converges to target SINR [78]	Result		Network tends to converge to self-jamming states
Context	Parameters	O	SINR at cluster head	Parameters	O	SINR at cluster head
		A	Power		A	Power
		D	Punish (jam) radios deviating from target SINR		D	Punish (jam) radios deviating from target SINR
		G	Target SINR		G	Target SINR
		C	*Isolated cluster*		C	*Isolated cluster with a jammer*
	Result		Network overcomes defection problems for significant improvement in performance [41]	Result		Network self-jams as it "punishes" the jammer