Optimal Buffer Management Policies for Delay Tolerant Networks

Chadi BARAKAT

INRIA Sophia Antipolis, France
Planète research group

Joint work with Amir Krifa and Thrasyvoulos Spyropoulos

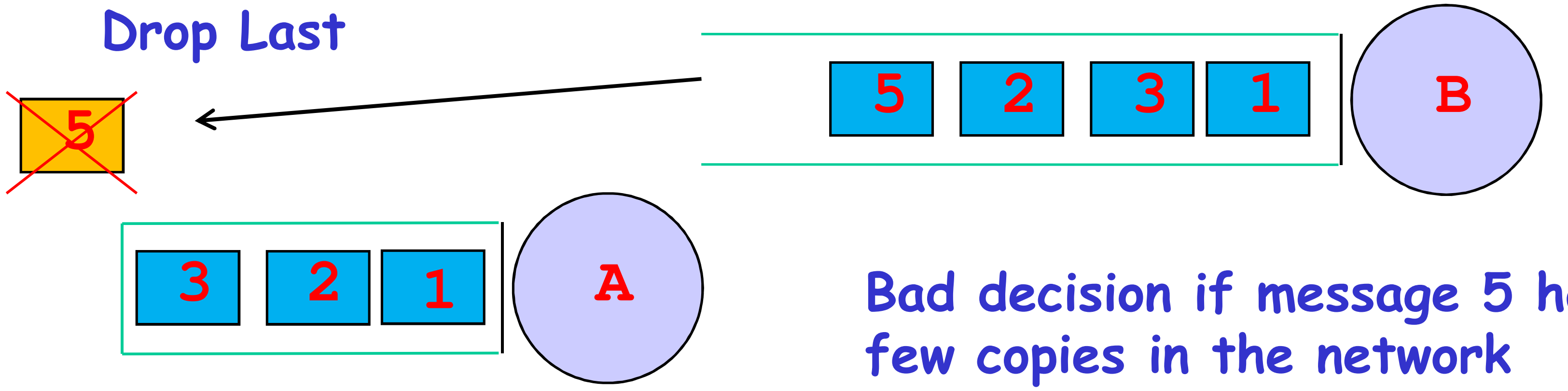
Email: Chadi.Barakat@sophia.inria.fr
WEB: http://www.inria.fr/planete/chadi

The DTN principle

- Rely on nodes' mobility to route through disconnected networks
  - Node: human carrying a laptop or PDA, a bus, a car, a satellite, etc
  - Nodes carry messages of each other while moving
  - Nodes replicate messages to increase the chance of delivery
- **DTN routing:** important DTN block that decides whether to replicate a message or not.
  - An important tradeoff:
    - The more the copies the more the chance
    - But the more the load on the network
  - Different propositions: Epidemic, spray and wait, utility-based, etc

How to Drop Messages in DTNs? Another important block

- Routing can limit the number of copies but is not enough …
  - Routing does not check if resources are available before replication
  - Nodes' buffers can still overflow due to many messages
  - Which message to drop in case of congestion ?
    - Last In ? First In ? Oldest ? Youngest ? Other ?
  - An important problem for which there is no clear solution



Bad decision if message 5 has few copies in the network

Our main contribution

- ❑ An analytical framework to solve the problem in its foundation using the notion of per-message utility
  - The utility can be seen as the marginal loss in some global metric (as the delivery ratio or the delivery delay) when dropping a message
- ❑ The optimal drop policy is then simple: Drop the message with the lowest utility value.
- ❑ State of the art:
  - Basic comparisons as Drop Front better than Drop Tail (Zhang et al.)
  - RAPID (Levine et al.) is relevant to our work but is for message scheduling and makes stronger assumptions.

Methodology

- Consider point-to-point communications
- Suppose first global knowledge about copies
- Take a global routing metric as the delay or the delivery rate
- Estimate its value at the moment of congestion
    - Decisions are taken separately by the nodes
- Find the best message to drop
    - Associate utilities to messages (marginal loss in global performance)
    - Drop the message with the lowest utility value.
- Then try to estimate the global knowledge using global information BUT on old messages …

Notations / Assumptions

- ❑ n = Number of copies of a message in the network
- ❑ m = Number of nodes that have seen a message
  - Used to calculate the probability of message being delivered
- ❑ L = Number of nodes
- ❑ λ = rate at which nodes meet

  Inter-meeting time exponentially distributed
- ❑ TTL = lifetime of messages

Case of delivery rate

- ❑ Global delivery rate at the time of congestion is then:

$$DR = \sum_{all-messages} \Pr(message-delivered)$$

$$\Pr(message-delivered) = \frac{m}{L-1} + \left(1-\frac{m}{L-1}\right)*\left(1-\exp\left(-\lambda nR\right)\right)$$

- ❑ We differentiate DR w.r.t the number of copies to find the best message to drop. It is the one minimizing this value:

$$\left(1-\frac{m}{L-1}\right).\lambda.R.\exp\left(-\lambda.n.R\right)$$

Per-message utility relative to the delivery rate

$$\left(1-\frac{m}{L-1}\right).\lambda.R.\exp\left(-\lambda.n.R\right)$$

This is a function of global information on a message. Models its utility (its importance) w.r.t. the global delivery rate.

We call it per-message utility relative to the delivery rate.

The resulting policy is called GBD (Global knowledge based drop policy)

Global information to be estimated later.

Case of delivery delay

In the same way, one can find the per-message utility relative to the global delivery delay:

$$\frac{1}{n^2\lambda} * \left(1 - \frac{m}{L-1}\right)$$

Again function of global information on a message.

Not easy to relate our utility-based optimal policies to simple ones as Drop Tail and Drop Front

Distributed version: How to calculate n and m ?

- ❑ n = number of copies of a message
  m = number of nodes that have seen the message.
- ❑ One option is to use the value obtained by flooding, but this is not efficient because the info arrives too late.
- ❑ Our solution:
  - Flood global information BUT on old messages.
  - All nodes have the same information after some time.
  - Estimate n and m from what has happened to old messages at the same elapsed time.
  - Of course under some stationarity and ergodicity conditions.

Distributed version (ctd)

- ❑ Suppose m and n follow two random variables M and N.
- ❑ We look for estimators that preserve global network performance:
  - Same average delivery delay and same average delivery rate

Take for example the delivery rate. This translates to:

real delivery rate = estimated delivery rate

$$E\left[\frac{M(T)}{L-1}+\left(1-\frac{M(T)}{L-1}\right)*\left(1-\exp\left(-\lambda N(T)R_i\right)\right)\right]=\frac{\hat{m}(T)}{L-1}+\left(1-\frac{\hat{m}(T)}{L-1}\right)*\left(1-\exp\left(-\lambda\hat{n}(T)R_i\right)\right)$$

Then solve this equation for the estimators of m and n

- Estimator of m is taken equal to its average value.

Distributed version: Message utility expressions

By plugging the estimators in utility expressions, we get:

For the delivery rate: $\lambda R E\left[\left(1-\frac{M}{L-1}\right)*\exp\left(-\lambda R N\right)\right]$

For the delivery delay: $$\frac{E\left[\frac{L-1-M}{N}\right]^2}{\lambda*\left(L-1\right)*\left(L-1-\overline{m}\right)}$$

History Based Drop (HBD)

Expectation calculated by summing over old messages.

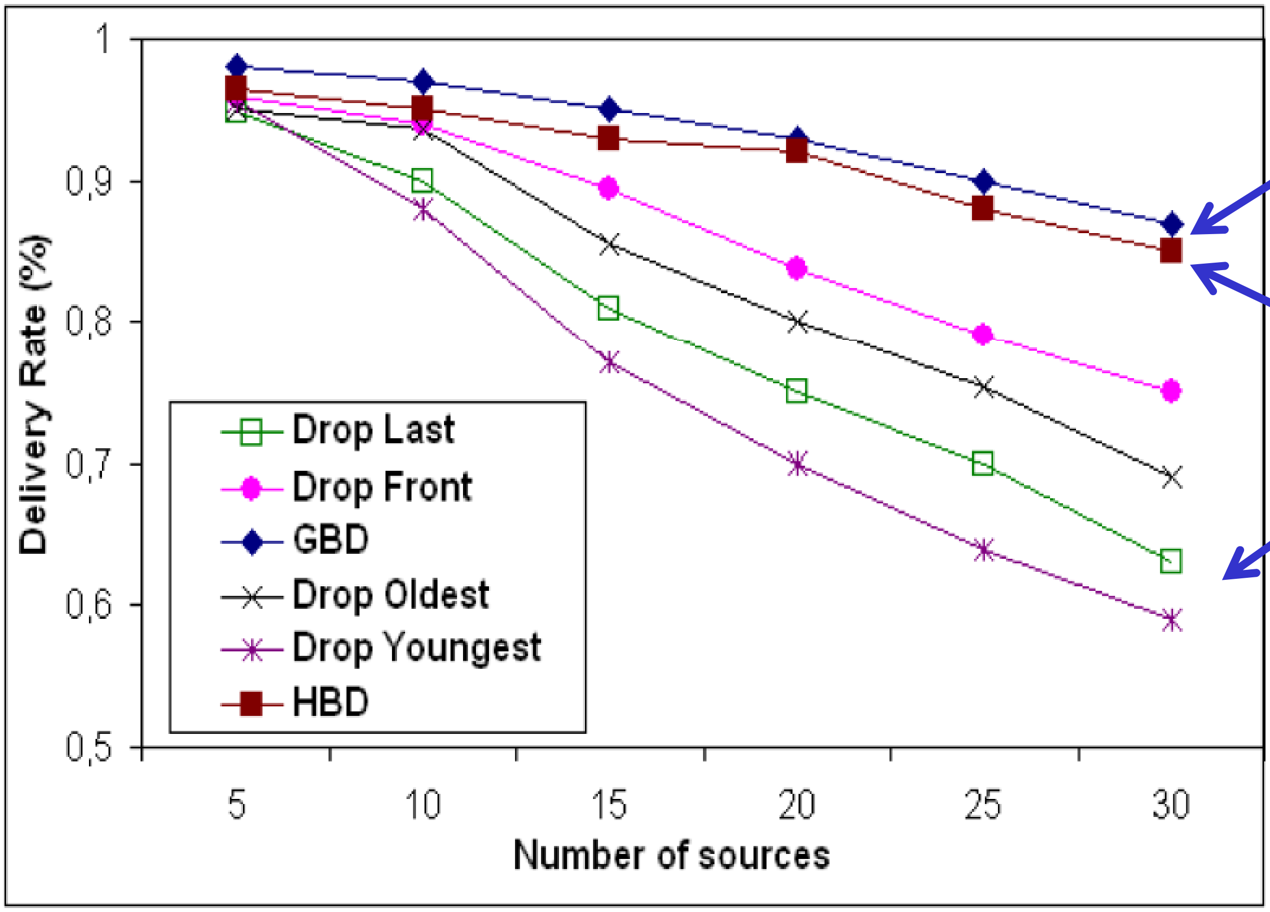
Experimental results: Setup

Mobility model	Random Waypoint	Traces du projet ZebraNet	Traces du projet Cabspotting
Simulation duration (s):	5000	5000	36000
Simulated Surface (m^2):	1000*1000	1500*1500	-
Number of nodes:	30	40	40
Average speed (Km/h) :	6	-	-
TTL (s) :	650	650	7200
Intervalle CBR (s) :	200	200	2100

DTN architecture added to the ns-2 simulator
MAC = 802.11b, range=100m, CBR sources, message size = 1Kbytes
random sources and destinations

Global Delivery Rate

Random Way Point

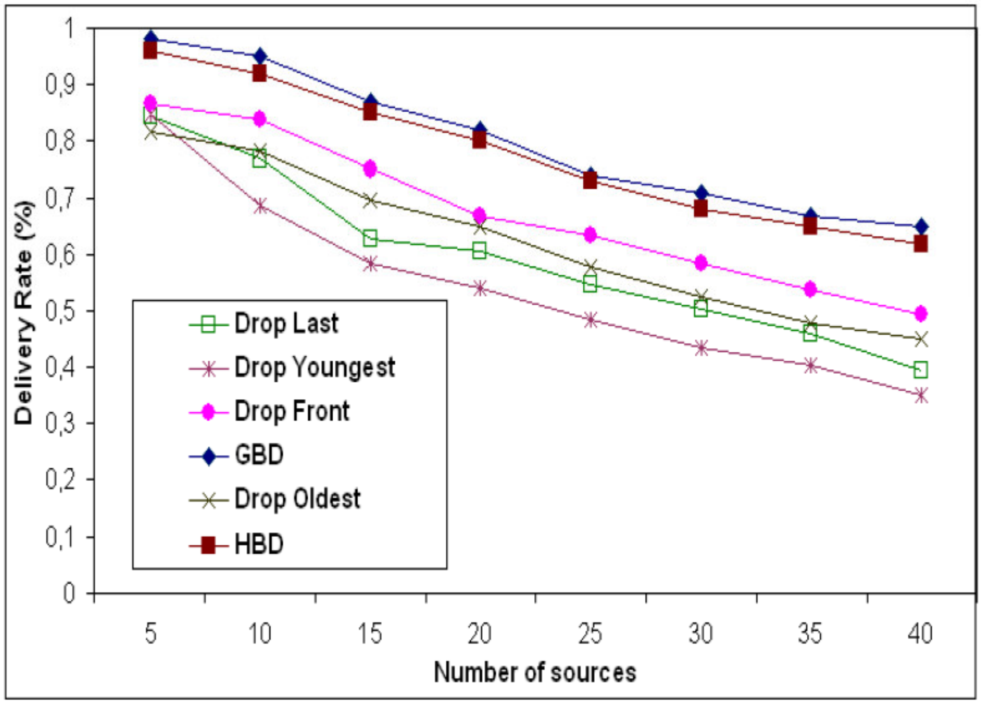


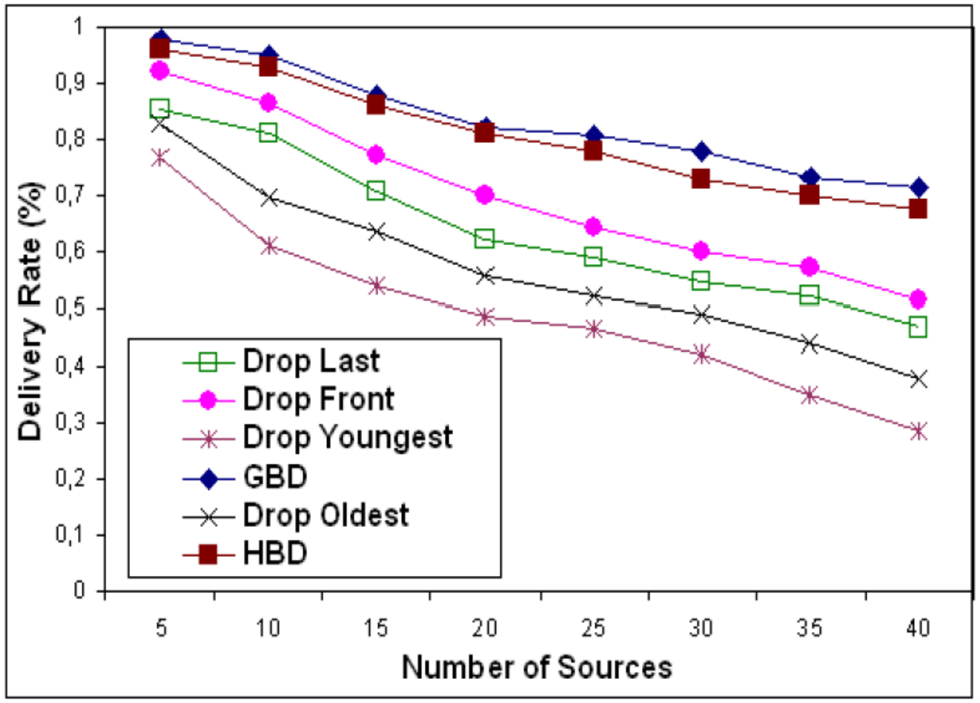
Very close

Almost 50% gain over DropTail

Message Delivery Rate

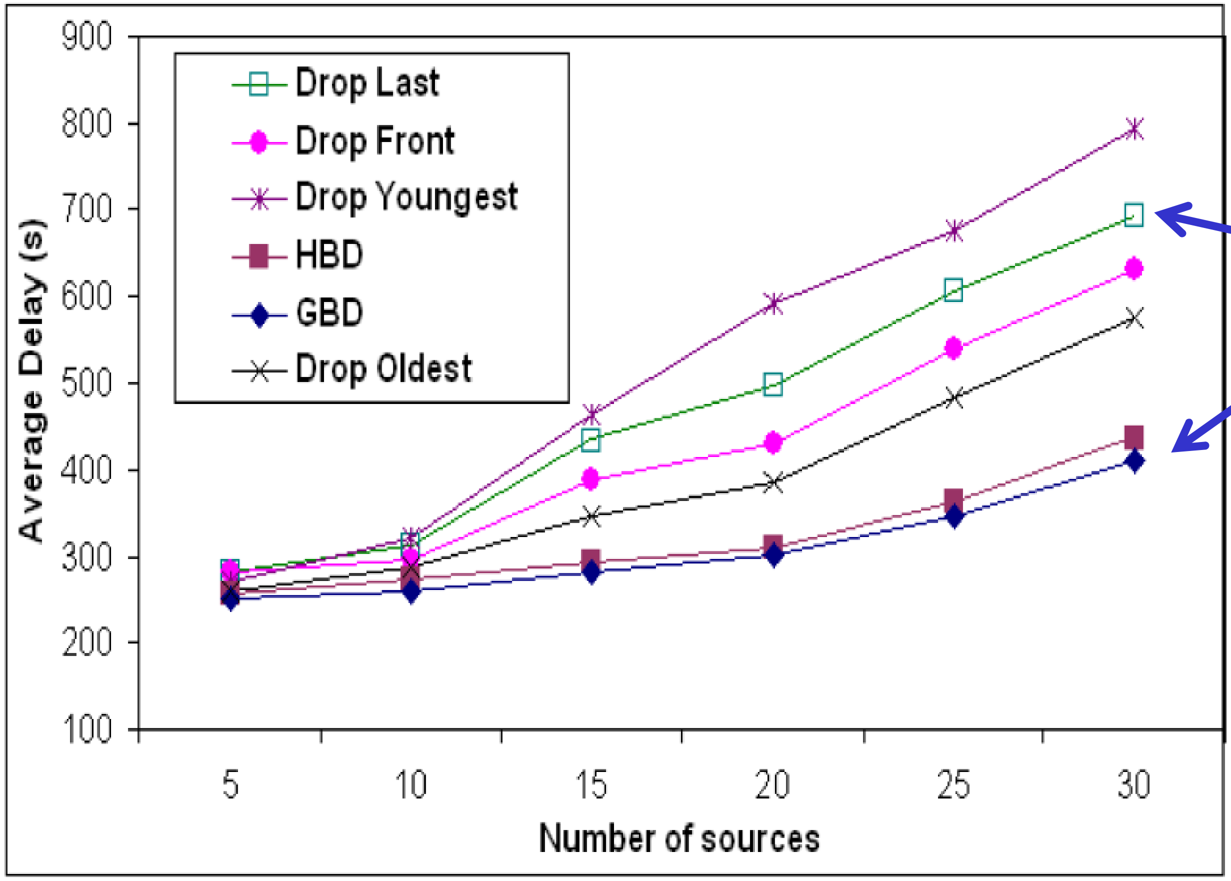
Real Traces





Global Delivery Delay

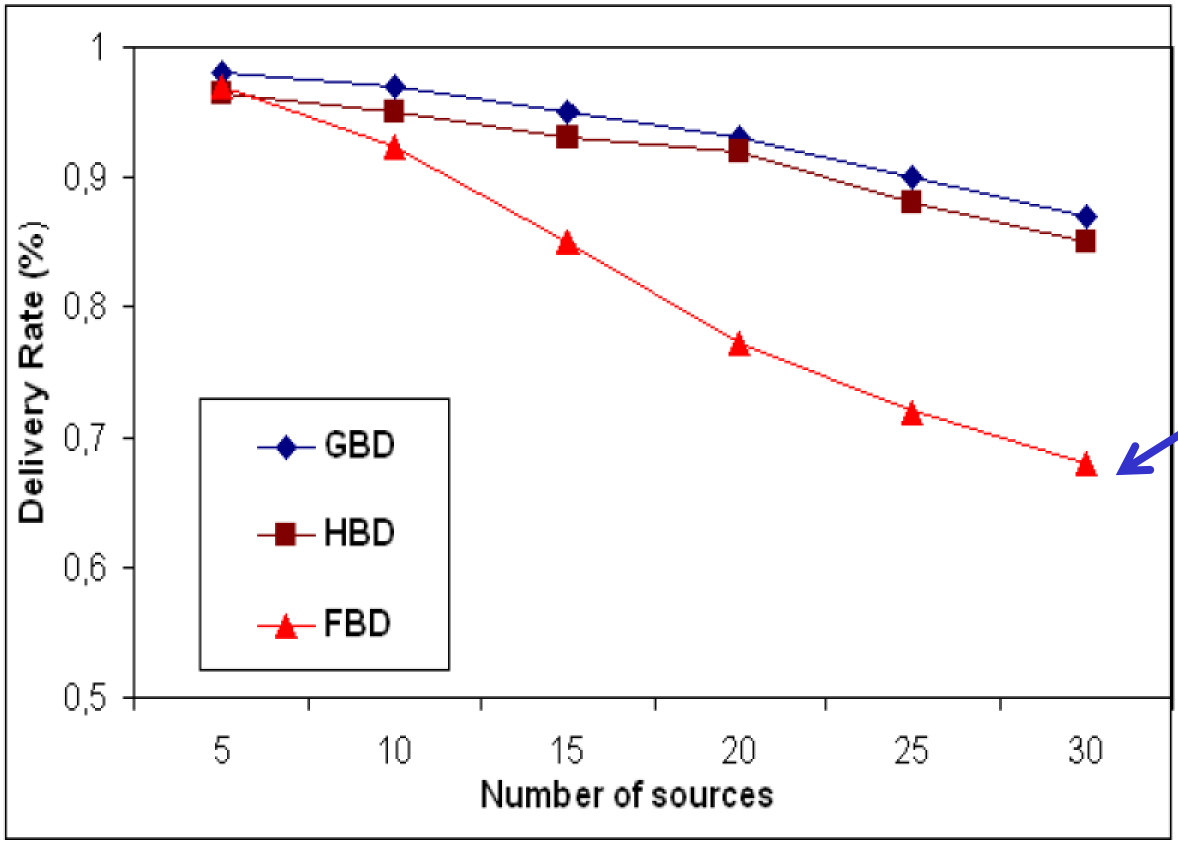
Random Way Point



Again, almost 50% gain

Flooding vs. History

Random Way Point



RAPID-like

Conclusions

- Analytical framework for optimal message drop in DTNs
- Two policies for the cases of delivery rate and delivery delay
- A distributed version that WORK
- Validation with a synthetic mobility model and real traces

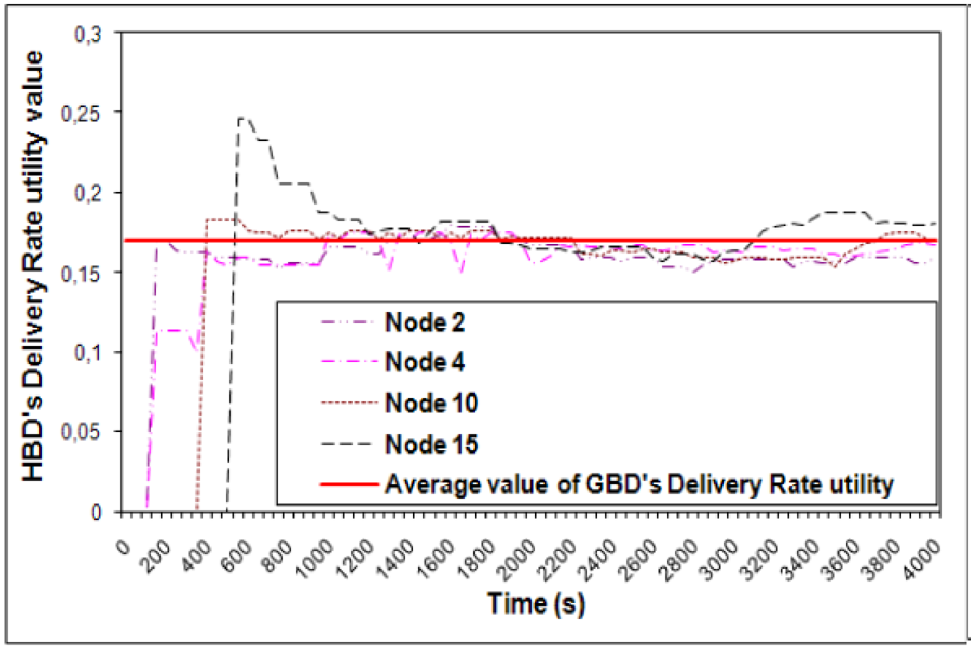
- Many issues still open e.g., find more global metrics and account for factors as the non stationary of the scenario, the variability of the number of nodes etc.

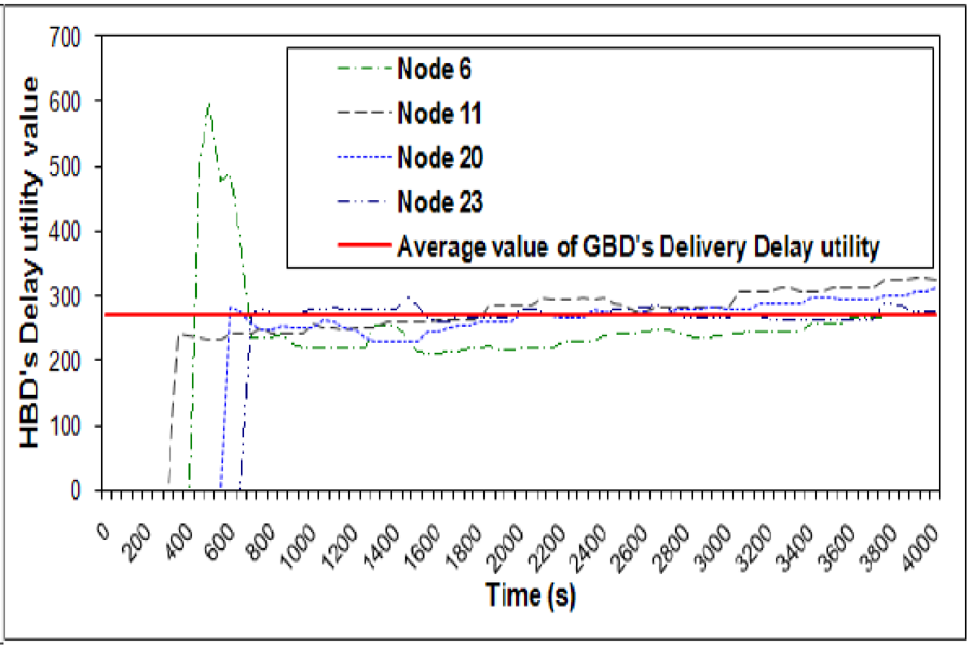
Thanks for your attendance !

http://www.inria.fr/planete/chadi

Utility values convergence

Small messages of 1Kbytes each (Random Way Point) (for elapsed time T = 100s)



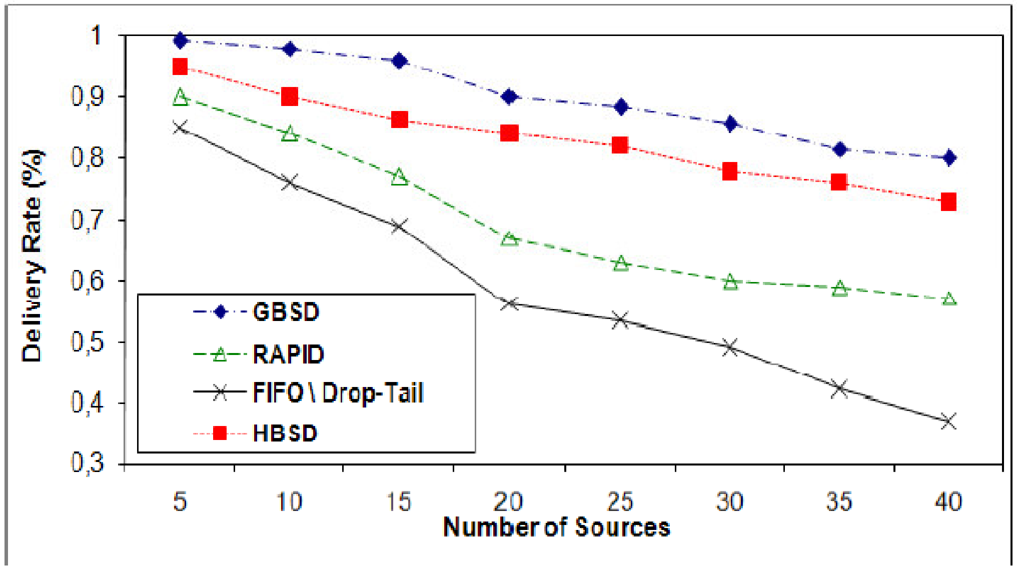


Message Utility for delivery rate

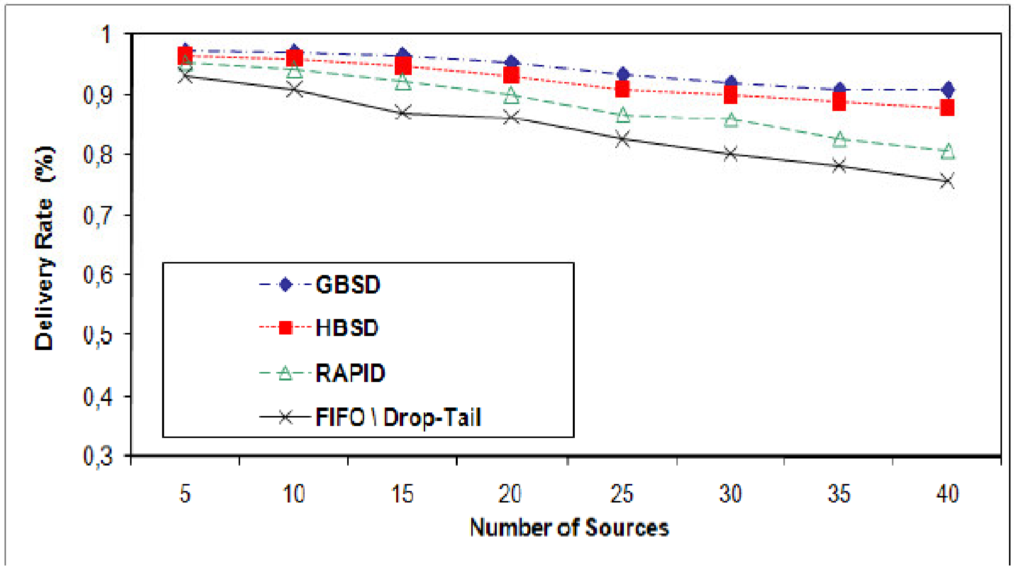
Message Utility for delivery delay

Bandwith limitation: Delivery

Large messages of 85 Kbytes each (Taxi trace)



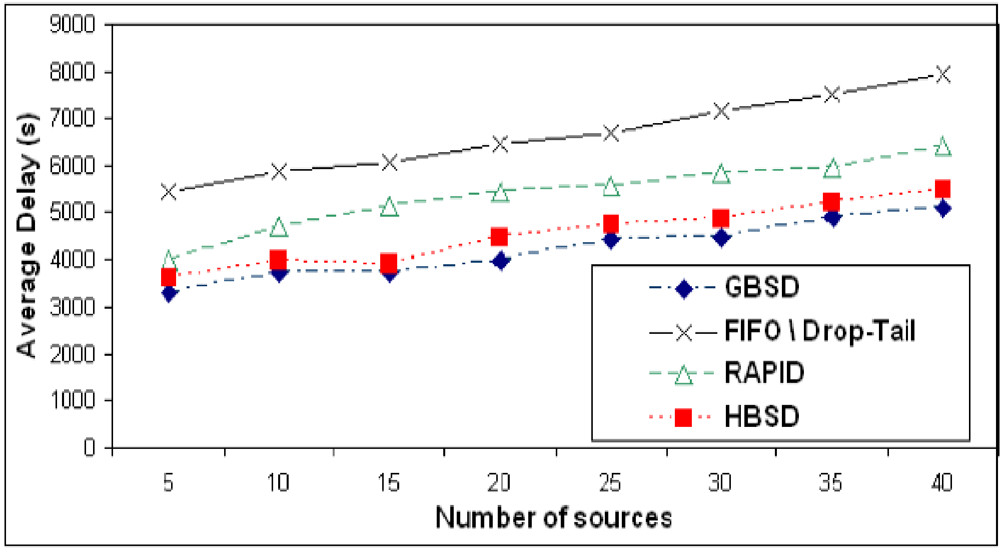
Limited Buffer

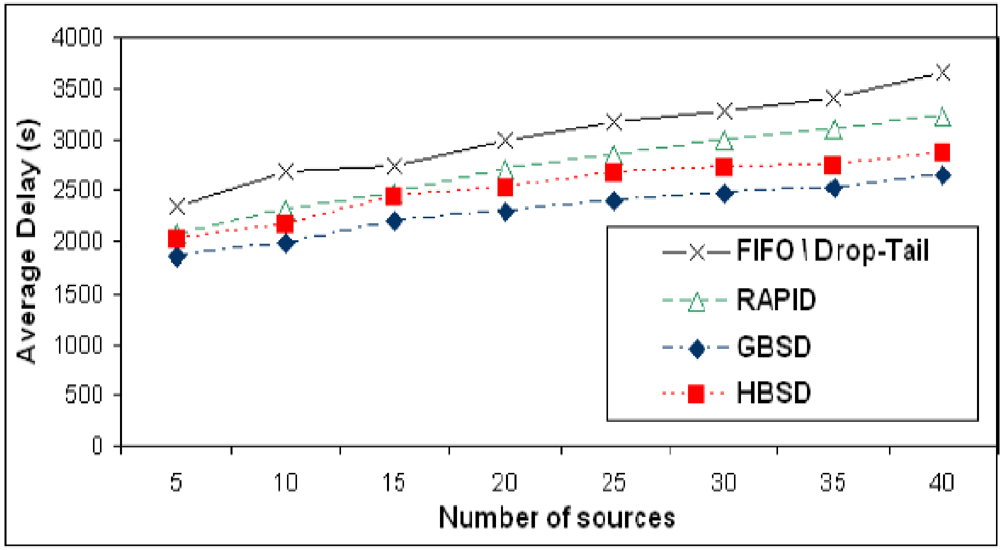


Unlimited Buffer

Bandwith limitation: Delay

Large messages of 85 Kbytes each (Taxi trace)





Limited Buffer **Unlimited Buffer**

For more details on the bandwith limited case:

Amir Krifa, Chadi Barakat, Thrasyvoulos Spyropoulos, "An Optimal Joint Scheduling and Drop Policy for Delay Tolerant Networks", to appear in proceedings of the WoWMoM Workshop on Autonomic and Opportunistic Communications, Newport Beach (CA), June 2008.