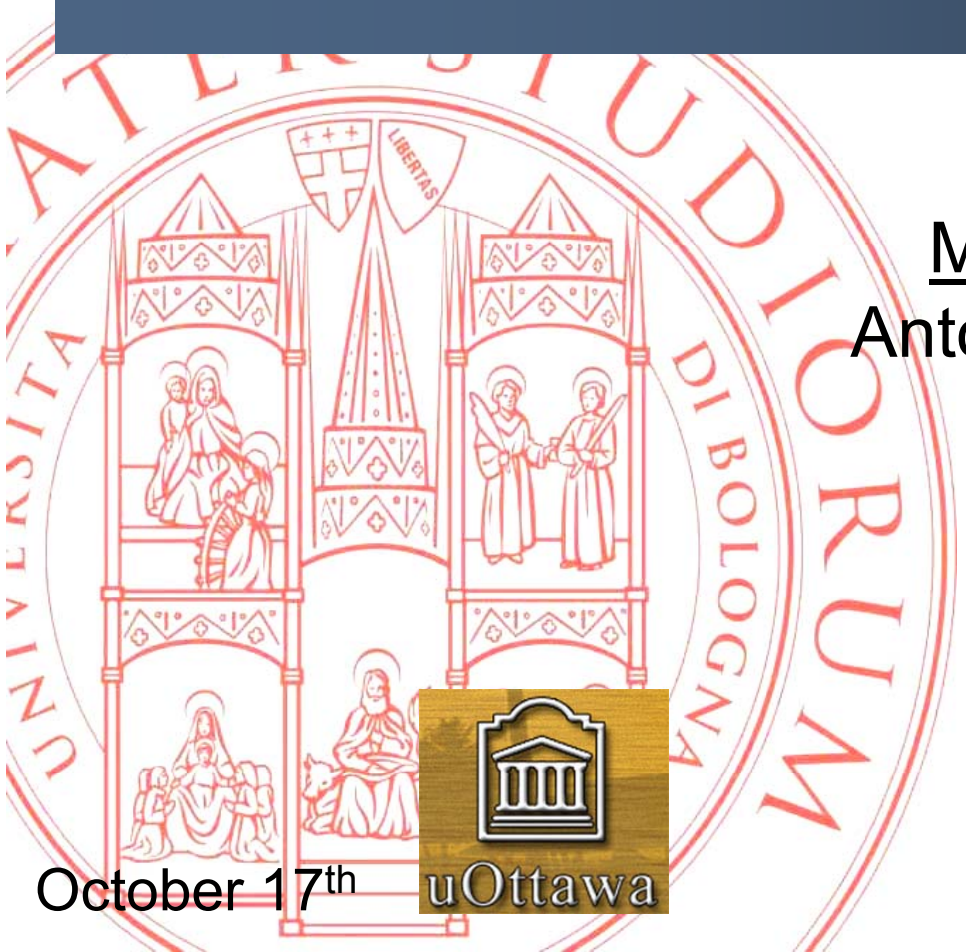
Self-Adaptive and Time-Constrained Data Distribution Paths for Emergency Response Scenarios



Mario Fanelli[1,2], Luca Foschini[2], Antonio Corradi[2], Azzedine Boukerche[1]

[1] PARADISE Research Laboratory, SITE, University of Ottawa, Canada
[2] Dipartimento di Elettronica, Informatica e Sistemistica (DEIS), University of Bologna, Italy

October 17th

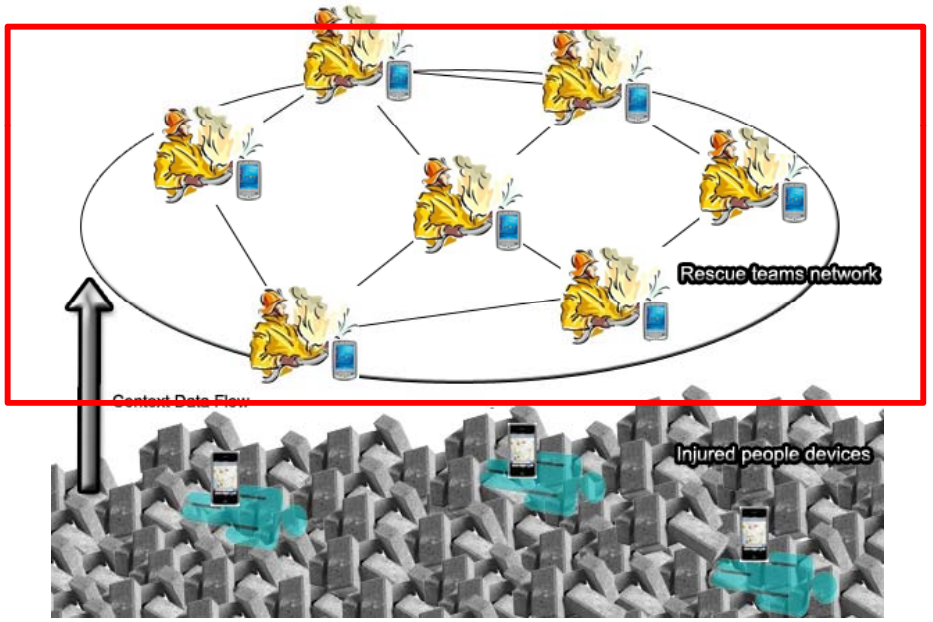
Agenda

1. Emergency Response Scenarios
2. Context Distribution Issues
3. Context Distribution Guidelines
4. Time-Constrained Distribution Process
5. Self-Adaptive Distribution Process
   - Query Distribution Suppression
   - Adaptive Paths Replication
6. Experimental Results
7. Conclusions and Ongoing Work

Emergency Response Scenarios

- **Context-Aware Services in Emergency Response Scenarios**
  - Rescue force devices build a **M**obile **A**d-hoc **NET**work **(MANET)**
  - Injured people devices advertise medical records, vital signs, and position

- **What do we need**
  - Timely data distribution
  - High reliability
  - Large research scopes



Reliable and **E**fficient **C**Ontext-aware data dissemination middle**W**are for **E**mergency **R**esponse **(RECOWER)**

Context Distribution Issues

- **Main Open Issues**
  - **Communication Layer**
    - **Heterogeneous**, **bandwidth-constrained** and **unreliable** links
    - **Congestion-prone communications** due to **high node density** and **context data traffic**

  - **Data Management Layer**
    - **Distributed** and **MANET-based** context repository
    - **Context data** and **routing information** stored on **mobile devices**

Context Distribution Guidelines

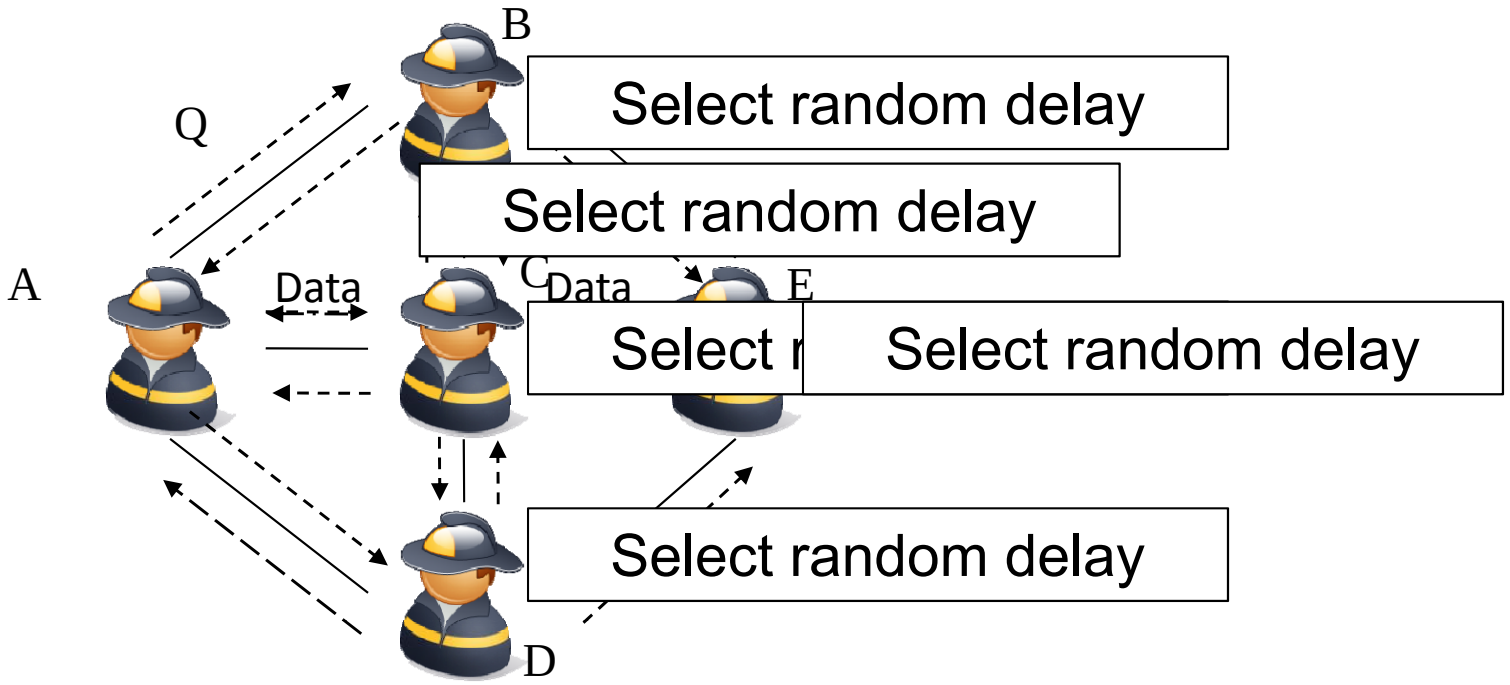
- To enable previous scenarios in real-world systems, the Context Data Distribution Infrastructure (CDDI) has to adopt:
  - **Differentiated quality levels** to manage the routing process and to reduce data distribution overhead
  - **Context-awareness** to self-adapt routing decisions by monitoring available resources, neighbor status, …

→ **Time-Constrained and Self-Adaptive Context Distribution**

- **Time-constrained Data Distribution**
  - Mobile devices require **continuous access** to their own context while **roaming**
  - CDDI has to enforce **data retrieval time** to avoid the delivery of useless or stale data
- **Self-adaptive Data Distribution**
  - CDDI has to balance **quality requests** and **available resources**
  - CDDI has to **automatically take over reconfiguration decisions** to tailor the run-time overhead

Time-Constrained Distribution Process

- **RECOWER context distribution** is based on **queries** and **data**
  - Queries transmitted in **broadcast** to reduce the final overhead
  - Data transmitted in **unicast** to better control data routing back
- Each query has a TTL (service-specified) to limit its propagation
- To avoid wireless channel congestion, each node introduces a random delay less than **data retrieval time/2*TTL**



Self-Adaptive Distribution Process

- **Broadcast-based query distribution**
  - High paths replication and reliability
  - Complete coverage in the physical area
  - High number of distributed queries may lead to path breaks due to memory saturation
  - High number of data distributions may lead to wireless congestion

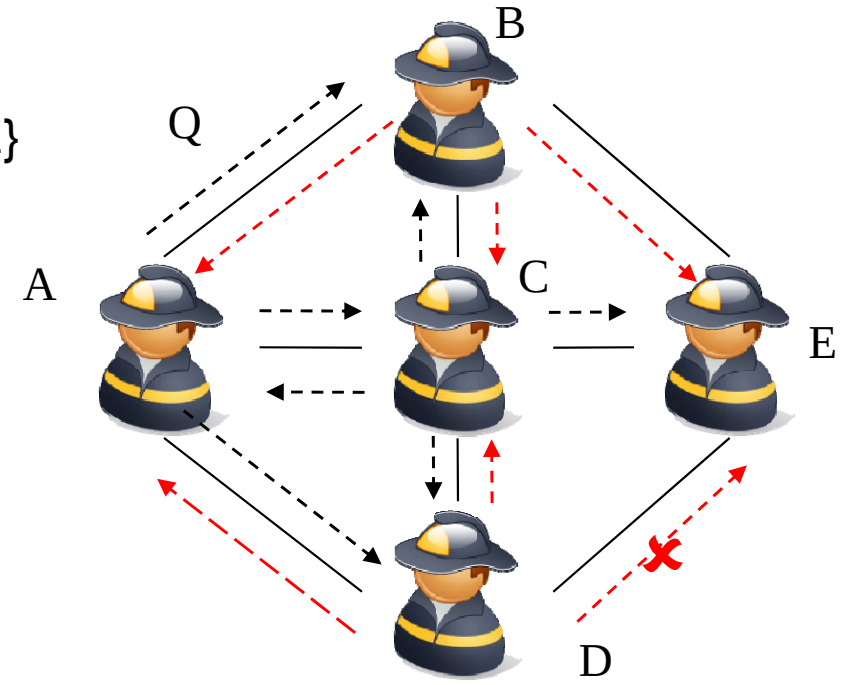
→ Self-adaptive query distribution process

1. **Query distribution suppression:** avoid those query distributions that will hit only nodes that had already received the query
2. **Adaptive paths replication:** modify distribution paths replication at run-time. A broadcast query message is considered only by a subset of current neighbors. These nodes are selected to follow paths with **high repository diversity,** i.e., close nodes do not have many data in common.

Query Distribution Suppression

- Each node knows its own one-hop neighbors (stored in a local *Routing Table (RT)*) by means of periodic mobility beacons
- Each query has an *Already Disseminated Nodes List (ADNL)* parameter to store all the identifiers associated with the nodes that had already received the query

$RT_A = \{B, C, D\}$
$RT_B = \{A, C, E\}$
$RT_C = \{A, B, D, E\}$
$RT_D = \{A, C, E\}$
$RT_E = \{B, C, D\}$

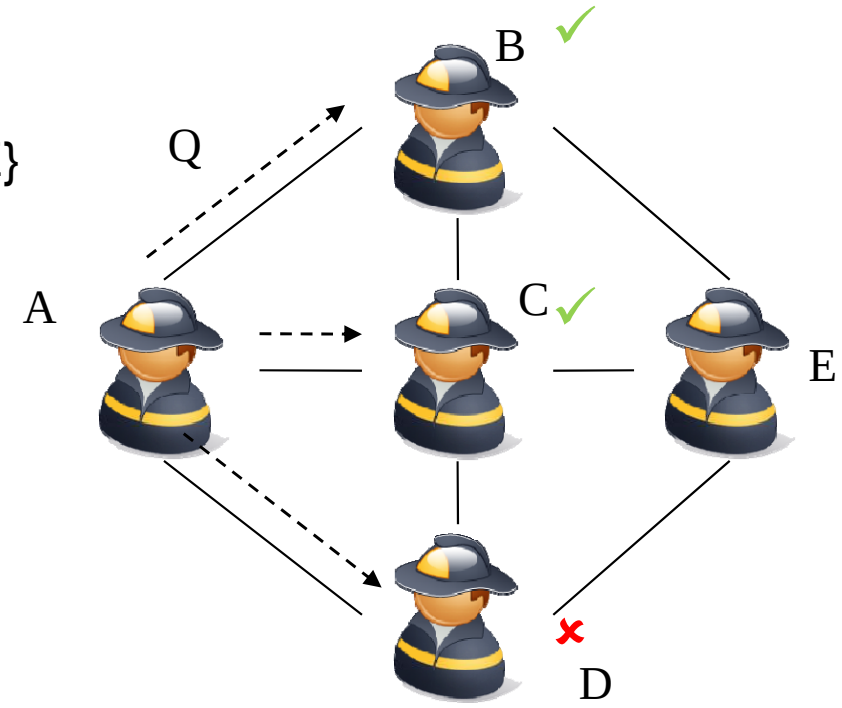


A: $Q_{ADNL} = \{A, B, C, D, E\}$
$\rightarrow Q_{ADNL} = \{\} = \{A, B, C, D, E\}$
B: $Q_{ADNL} = \{A, B, C, D, E\}$
B: $RT_B / Q_{ADNL} = \{A, B\} = \{C, D\}$
C: $Q_{ADNL} = \{A, B, C, D, E\}$
C: $Q_{ADNL} = \{A, B, C, D, E\}$
C: $Q_{ADNL} = \{A, B, C, D, E\}$
D: $Q_{ADNL} = \{A, B, C, D, E\}$
D: $RT_D / Q_{ADNL} = \{A, E\} = \{C, D\}$
$\rightarrow Q_{ADNL} = \{A, B, C, D, E\}$
E: $Q_{ADNL} = \{A, B, C, D, E\}$
E: $Q_{ADNL} = \{A, B, C, D, E\}$

Adaptive Paths Replication

- Sender node selects which neighbors have to process a particular query and inserts them in Q_{ADNL}
- Receiver node checks if its own identifier is in Q_{ADNL}: if yes, process the query; otherwise, discharge it

$RT_A = \{B, C, D\}$
$RT_B = \{A, C, E\}$
$RT_C = \{A, B, D, E\}$
$RT_D = \{A, C, E\}$
$RT_E = \{B, C, D\}$



B: $RT_B \cap Q_{ADNL} = \{B, C, D\}$
$Q_{ADNL} = \{\}$
→ Process and distribute Q
→ $Q_{ADNL} = \{A, B, C\}$

C: $Q_{ADNL} \cap C = \{A, B, C, D\}$
→ Process and distribute Q

D: $Q_{ADNL} \cap D =$
→ Discard Q

Adaptive Paths Replication

- RECOWER adopts an automatic selection process based on both **available memory** and **repository diversity**

- **Management Information**
  - **Local Query Load Factor** is the available memory to store queries
  - **Data Keys List** represents the locally memorized data keys and is useful to calculate repository diversity
  - **Data Repositories Diversity Factor** is the average diversity between local repositories and the ones deployed on one-hop neighbors

- **Neighbors selection**
  - Once considered the average memory load, RECOWER applies a linear function to retrieve the neighborhood cardinality, and selects the neighbors with the higher repositories diversity factors

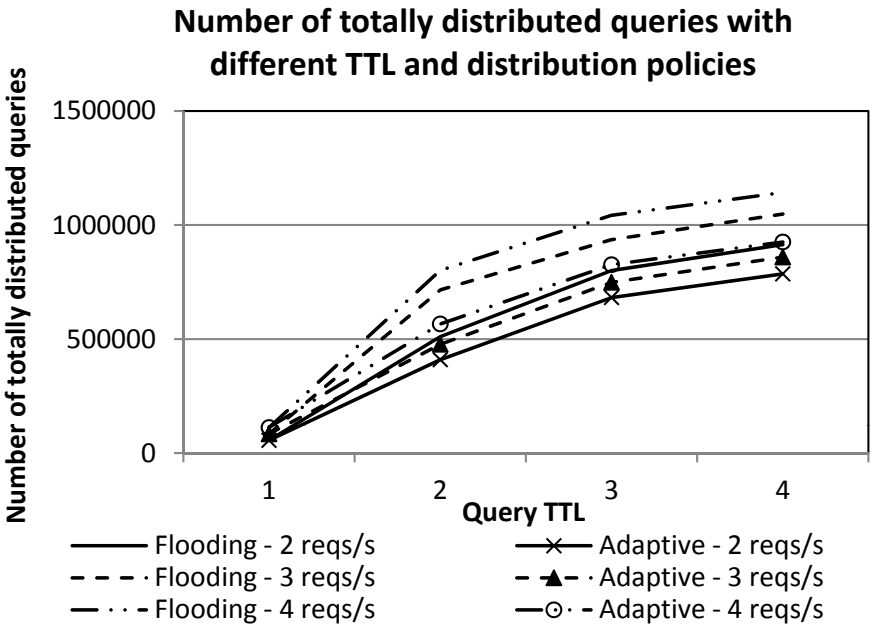
Experimental Results

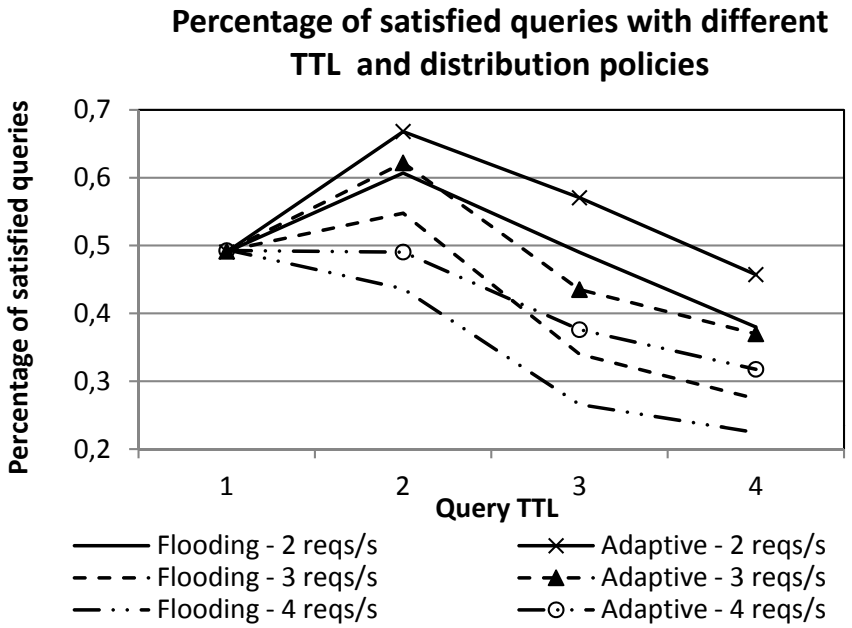
- ns-2.34
  - 50 mobile nodes roaming in an area of 350x350m
  - IEEE 802.11 WiFi model, 100m transmission range
  - Simulation time 600 seconds, 33 runs with different mobility scenario
- Mobility model
  - Random Waypoint with uniform speed in [0.5; 1] m/s and uniform pause time in [0; 10] seconds
  - Each node selects the next waypoint before reaching borders
- RECOWER parameter
  - Mobility beacons emitted every 10 seconds
  - Both query ADNL and data keys list are represented by means of **Bloom Filters** to reduce management overhead

Experimental Results

- Context data production
  - Each node has 10 local context data sources, and can cache at most 30 context data
- Default context query production
  - TTL = 2, data retrieval time of 2 seconds
  - Each node can memorize a maximum number of queries (Q_{MAX}) equal to 70, and periodically requests data choosing among the 500 context data sources by using a uniform distribution

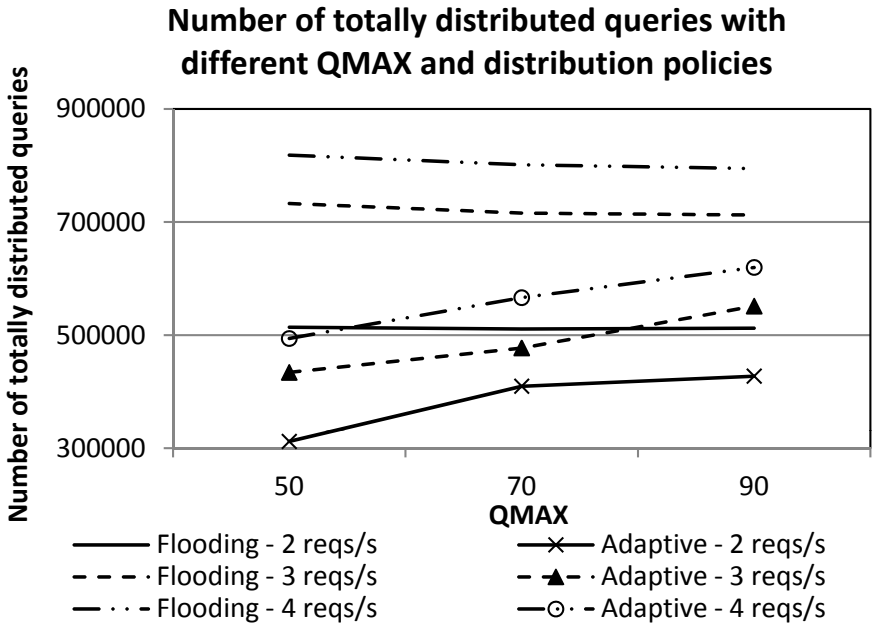
Experimental Results

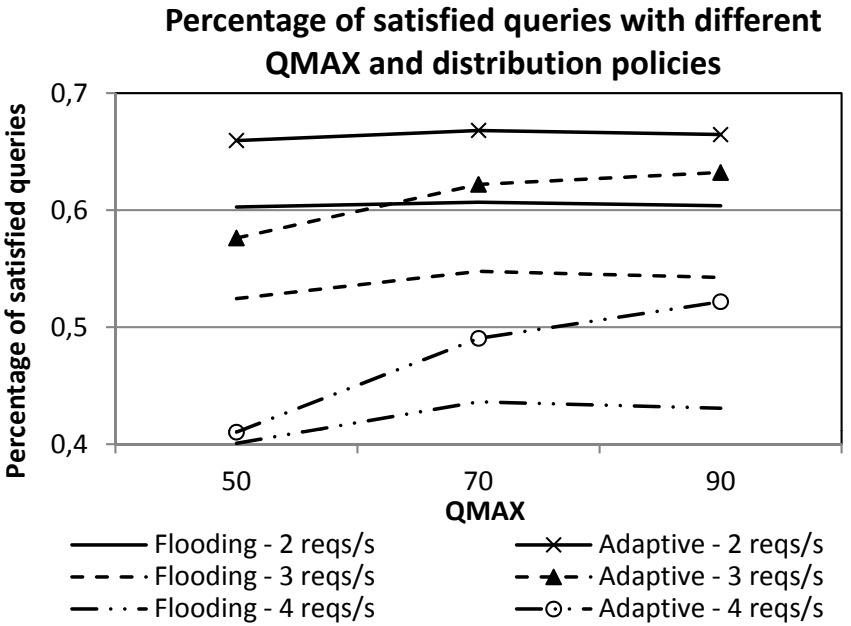




- Adaptive approach always distributes fewer queries
- Differences are more visible for higher request rates and higher TTL since both affect the load perceived by the adaptive solution
- Reduced traffic increases the distribution process reliability

Experimental Results





- Higher Q_{MAX} values increase the number of logical neighbors and distributed queries
- Higher Q_{MAX} values always result in higher reliability since against query replacement
- Adaptive solution reliability is more sensitive to Q_{MAX} since higher values lead to reduced memory load and increased path replication

Conclusions

CONCLUSIONS

- Agreed quality levels are fundamental to correctly manage the context distribution process
- Reduced number of broadcasts positively affects both scalability and reliability
- Our solution reduces research scopes, but fewer collisions and usage of paths with high data repositories diversity make our solution valid

ONGOING WORK

- Graph-based dissemination paths
- Role-based context data memorization and distribution

RECOWER project web site and contacts

- Prototype code and information:
  http://lia.deis.unibo.it/Research/RECOWER

- Contacts: Mario Fanelli (mario.fanelli@unibo.it)

Thanks for your attention!