ONRECT: Scheduling Algorithm for Opportunistic Networks

Saima Ali and <u>Andreas Willig</u>

University of Canterbury

UC UNIVERSITY OF CANTERBURY Te Whare Wānanga o Waitaha CHRISTCHURCH NEW ZEALAND

Lille, July 4, 2016

NRG Network Research Group

Motivation

- Opportunistic or delay-tolerant networks (DTN) operate on the store-carry-and-forward principle to deal with only intermittent connectivity
- Main performance metrics: E2E delay, delivery rate, overhead
- Much attention on routing in DTN, less on message scheduling:
  - Suppose a node has several messages to forward and currently has several neighbors as potential next hops
  - **Q**: In which sequence should messages be sent to their next hop nodes?
- We hypothesize that including the duration of the (remaining) contact time into message scheduling will have significant performance impact

Contributions

- We design a message scheduling scheme (ONRECT) which is based both on contact time information and the "quality" of a neighbor
  - ONRECT: Opportunistic Networks Routing with rEmaining Contact Time
  - Quality = distance of neighbors final destination to message final destination
- We compare ONRECT against the ORWAR scheme from the literature, which also utilizes contact time information (but not neighbor quality)
- We investigate both schemes under perfect and approximate knowledge of the contact time

System Model

- Playground:
  - 3500 x 4500 m2
  - Street layout resembling parts of Helsinki
- 1500 nodes
  - 500 stationary nodes, randomly placed, serving as message destinations (e.g. representing access points), but can also serve as relays and generate messages
  - 500 mobile nodes with average speeds from 10 – 50 km/h (representing cars)
  - 500 mobile nodes with average speeds from 10 – 40 km/h (representing trams)
  - Nodes in the last two groups can generate messages but do not serve as destination
  - Nodes have unlimited buffer capacity
- Mobility model: Shortest-path map-based movement model
  - Node Positions are restricted to roads
  - At each turn a node picks a new random speed from speed interval
- Unit disk model as channel model

System Model (2)

- Message format:

src id	dest id	msg id	ttl	µ	size	data

Src id	=	Source Address	Ttl	=	Time to live
Dest id	=	Destination address	µ	=	Utility value
Msg id	=	Unique Id of Message	Size	=	Size of the Msg

  - Identifiers encode geographical locations

- Messages can be very large, are not fragmented

- Nodes use IEEE 802.11g PHY with 2 Mbps rate, but no MAC
  - No MAC: we want to focus on message scheduling under idealized circumstances
  - Only 2 Mbps: we wanted to make sure that scheduling is non-trivial

- All nodes have one interface and use same transmit power

Baseline: ORWAR Scheme

- ORWAR uses binary Spray-and-Wait (SnW):
  - A source node "sprays" L-1 message replicas to relay nodes
  - In plain SnW the source hands over a message to L-1 different relays, which keep the message and either deliver to destination or drop after TTL expiry
  - Binary SnW:
    - Source replicates message to first relay, gives it an allowance of L/2 and keeps an own allowance of L/2
    - Any node having the message and an allowance > 1 replicates the message to a new relay node and halves the allowance
    - Any node having the message and an allowance of 1 keeps the message until delivered or TTL expires

- ORWAR differentiates messages according to user-defined utility, higher utility messages can have a larger initial allowance L

- Messages are stored in a buffer according to priority and utility-per-bit ratio

Baseline: ORWAR Scheme (2)

- ORWAR uses the following estimate of remaining contact time between two nodes:

$$t_{cw} = \frac{2 \cdot \min\{r_1, r_2\} \cdot \cos\alpha}{\|\boldsymbol{v}\|}$$

  - Where r_1 and r_2 are the transmission radii of both nodes, **v** is the difference of the velocity vectors of both nodes, and α is the angle between nodes during the contact

- ORWAR prioritizes messages with higher utility-per-bit ratio and a size that is small enough to fit within the remaining contact time

- ORWAR also employs a vaccination mechanism

ONRECT Scheme

- ONRECT combines routing and scheduling
- It shares some characteristics with ORWAR:
  - It uses binary spray-and-wait
  - It uses a vaccination mechanism
  - Users assign a utility value to prioritize messages in the sending queue and furthermore a utility-per-bit is calculated
- ONRECT uses:
  - An underlying protocol / algorithm which maintains a table with the current neighborhood, including their current speed and direction
  - Knowledge / estimates of remaining contact time for all neighbors

ONRECT Scheme – Scheduling

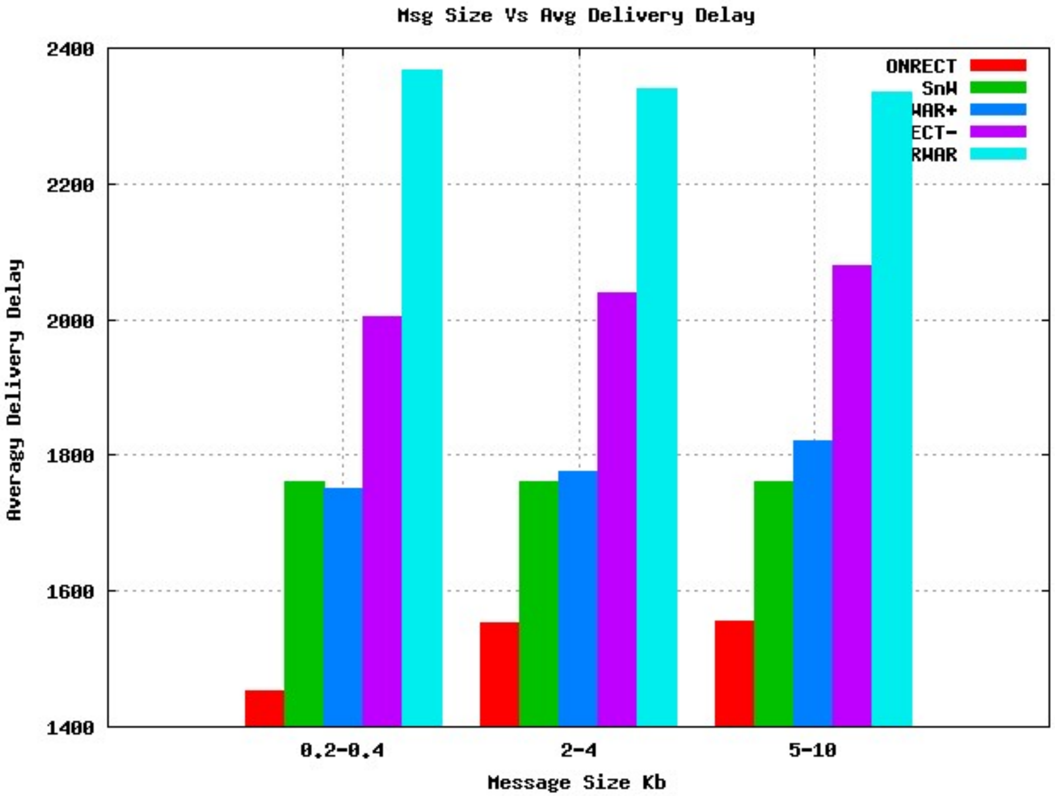
- Scheduling Algorithm takes into account:
  - Remaining contact time
  - The best neighbor to forward message to
    - Only neighbors with destination in some range of message destination eligible
    - Among these, the neighbor getting closest to the message destination is picked
  - Position of message in queue

- Algorithm (running upon each change of neighborhood):
  - If a message can be delivered to final destination, deliver and drop it
  - Otherwise, traverse queue starting from the highest utility-per-bit ratio:
    - Find best neighbor for message
    - If it can be transferred within contact time, then do so, otherwise skip it

Performance Analysis

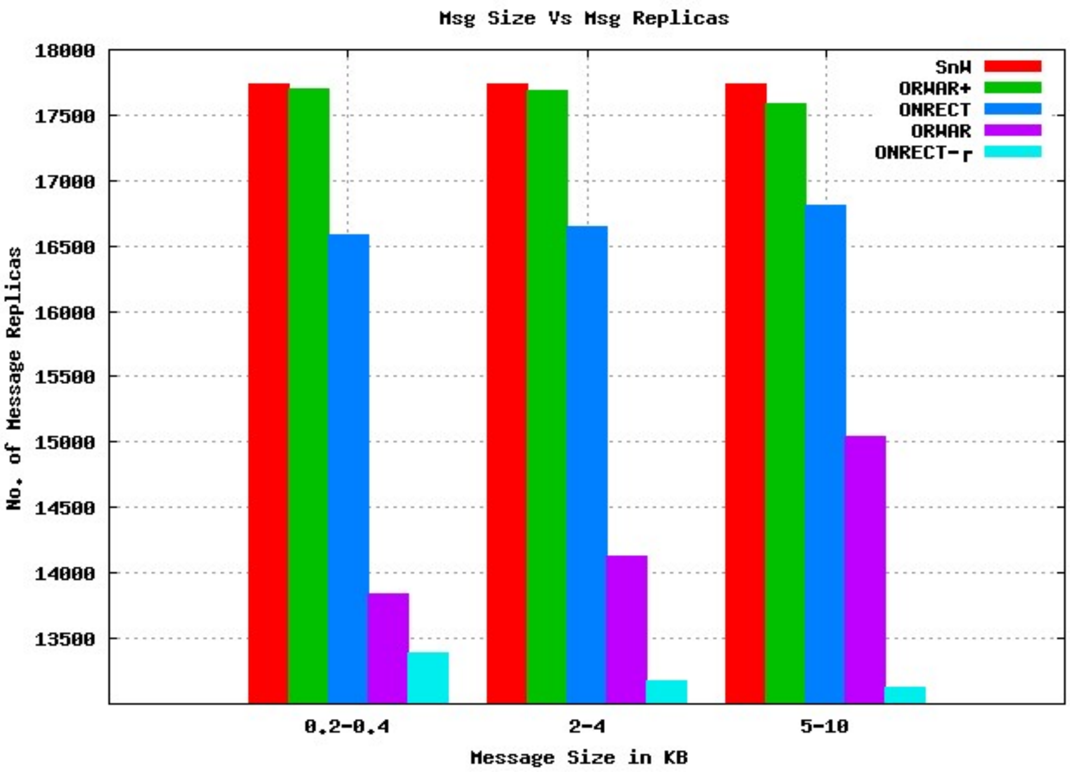
- Performance has been assessed by simulation (ONE simulator)
- Schemes considered:
  - Binary spray-and-wait
  - "ORWAR" = ORWAR algorithm with estimated contact time
  - "ONRECT" = ONRECT with ground truth for contact time
  - "ONRECT-" = ONRECT with ORWAR's estimate of contact time
  - "ORWAR+" = ORWAR with ground truth for contact time
- We have varied message size distribution:
  - Small: drawn randomly from 200 – 400 B
  - Medium: drawn randomly from 1 – 2 kB
  - Large: drawn randomly from 5 – 10 kB
- Message generation interval: 1 – 10 sec (for entire network)

Performance Analysis – Average Delay



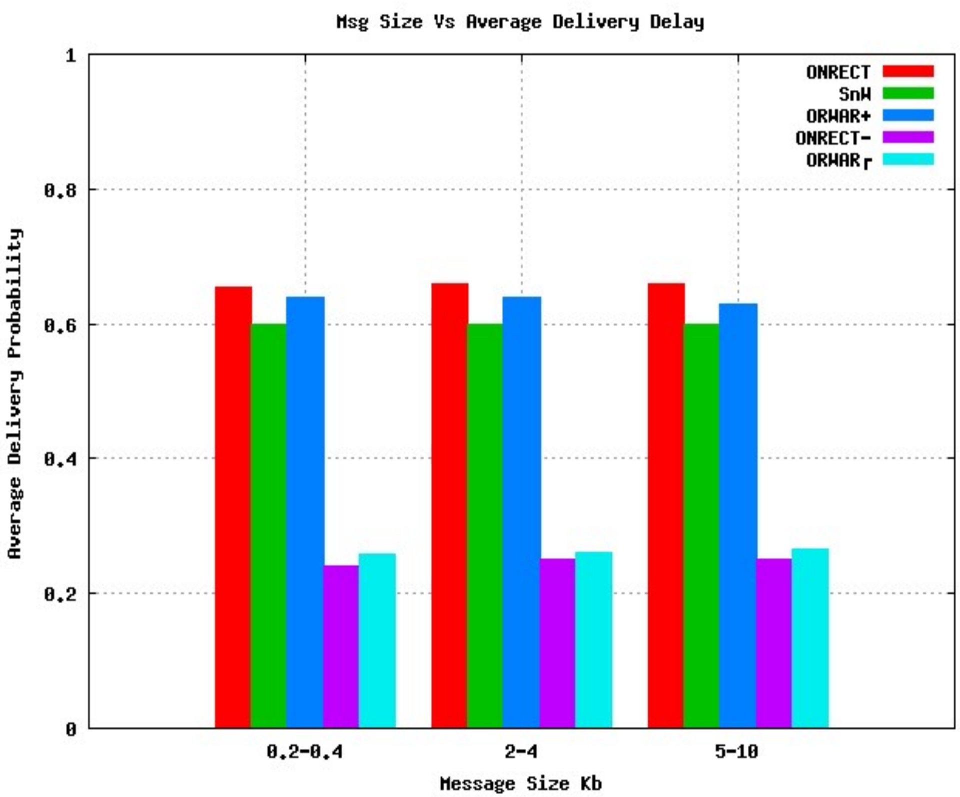
- Delay: measured between generation and first reception by final destination
- ONRECT is much better capable of exploring (accurate) contact time information than ORWAR

Performance Analysis – Overhead



- Overhead is measured in total number of message replications for all messages generated during simulation time
- ORWAR+ and ONRECT have much larger numbers: the more accurate contact time information allows to transmit more messages
- We suspect that ORWARs estimate is too low on average

Performance Analysis – Delivery Ratio



- Delivery Ratio = Fraction of all messages reaching final destination within given upper bound of 1000 s
- Having accurate contact time information has substantial impact
- This motivates need to find better estimators

Conclusions

- Comparing ONRECT and ORWAR:
  - ONRECT has shorter delay than ORWAR
  - But both have about the same delivery probability for the same level of contact time accuracy
- Our results motivate the inclusion of contact time information into message scheduling algorithms
- They also motivate the need to find better estimators for the remaining contact time

THANK YOU !

Lille, July 4, 2016