IEEE 802.21 MEDIA INDEPENDENT HANDOVER

DCN: 21-07-0268-00-0000

Title: **Performance Evaluation of L3 MIH Transport**

Date Submitted: July 16, 2007

Presented at IEEE 802.21 session #21 in San Francisco, CA

Authors or Source(s):

Richard Rouil, Nada Golmie, David Griffith

Abstract: This presentation shows a performance simulation study of the MIH Transport using UDP, TCP, and the MIH acknowledgement protocol.

IEEE 802.21 presentation release statements

This document has been prepared to assist the IEEE 802.21 Working Group. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein.

The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE's name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE's sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that this contribution may be made public by IEEE 802.21.

The contributor is familiar with IEEE patent policy, as stated in Section 6 of the IEEE-SA Standards Board bylaws <http://standards.ieee.org/guides/bylaws/sect6-7.html#6> and in *Understanding Patent Issues During IEEE Standards Development* http://standards.ieee.org/board/pat/faq.pdf>

Performance Evaluation of L3 Transport Protocols for IEEE 802.21

Richard Rouil, Nada Golmie and David Griffith

National Institute of Standards and Technology

http://www.antd.nist.gov/seamlessandsecure.shtml

21-07-0268-00-0000

Outline

- MIH protocol
  - Overview
  - Transaction ID
  - Acknowledgement
- MIH_NET_SAP
- Message flow for UDP and TCP
- Performance results
  - Network Scenario
  - UDP evaluation
  - TCP evaluation
- Conclusions

MIH protocol

- The IEEE 802.21 draft defines an MIH protocol to carry messages between two remote MIHF entities.
- The messages contain different type of information including:
  - Service management
  - Events
  - Commands (requests and responses)
  - Information service
- The MIH messages can be carried over layer 2 or layer 3+ protocols, depending on the location of the PoS and the technology used.

Transaction ID

- Standard states: “Transaction Identifier (Transaction ID) is an identifier that is used within a message sent by the requesting MIHF and its corresponding response message. This is also required to match each request, response or indication message and its acknowledgement.”
- A transaction state is maintained and it is used to detect duplicate messages.

Ack mechanisms

- Section 8.2.1: MIH messages require reliability for remote communication between peer MIH entities to ensure the receipt of data to the intended destination.
- Acknowledgement can be provided by different means:
  - Use of a reliable transport protocol such as TCP.
  - Use of the MIH protocol acknowledgement operation.

Ack using transport protocol

- The MIHF relies on the transport layer to carry the message to the remote MIHF.
- Reliability requirement is specified in the MIH_NET_SAP primitives.
- The transport may not provide feedback to the MIHF in the event of a successful transmission.

Pros	Cons
-Leverage processing in the MIHF -No timer required for indications and ACK.	• If a packet is eventually lost at the transport layer, the MIHF does not know about it. •Timers for transport protocol may be long •In the case of a request message, the MIHF still needs a timer to wait for a response. This value should be higher than the time required for the transport to send the request (including retransmission).

Ack using MIH protocol

- Used when the transport protocol is not reliable.
- The remote MIHF sends an acknowledgement upon receiving a message.
- When a response is not ready, the destination can ACK the message without payload.

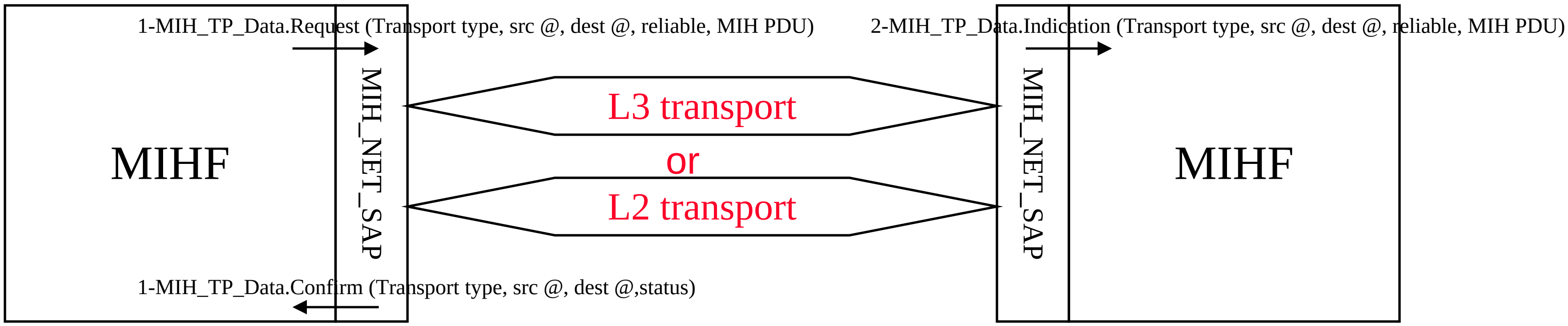
Pros	Cons
-The MIHF is aware of the all the messages exchanged. -Additional control over the handling of failed messages (for example retransmit using a different interface)	• Additional processing in the MIHF • Additional timers to wait for ACK

MIH_NET_SAP

- Definition: “Abstract media-dependent interface of MIHF which provides transport services over the data plane on the local node, supporting the exchange of MIH information and messages with the remote MIHF. For all transport services over L2, the MIH_NET_SAP uses the primitives specified by the MIH_LINK_SAP”.
- The MIH_NET_SAP defines the primitives to interact with the Transport Service Provider
- The Transport Service Provider communicates with the transport protocols such as UDP/TCP and lower layer to carry messages.

MIH_NET_SAP primitives

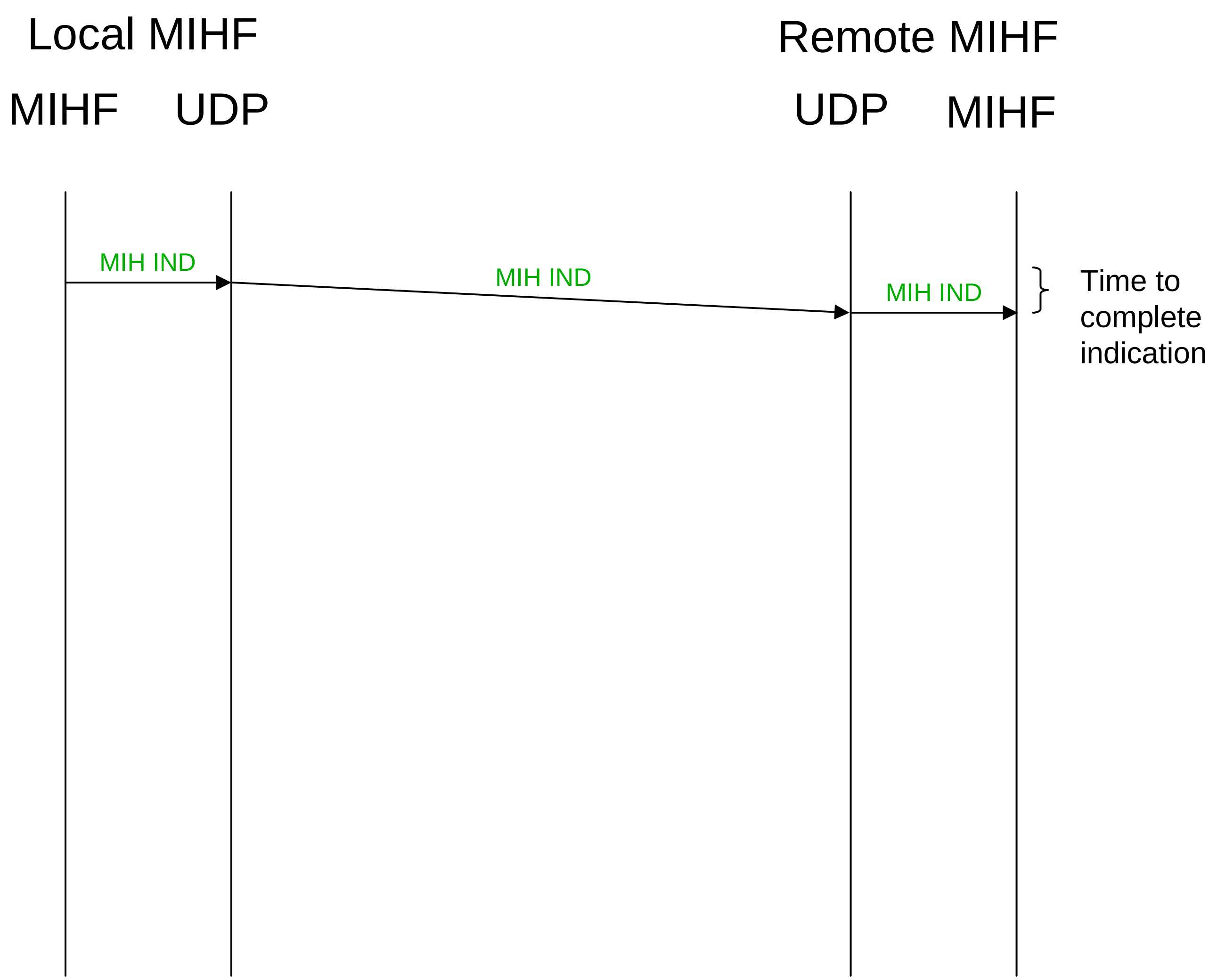
- MIH_NET_SAP defines one function to communicate with a remote node:
  - MIH_TP_Data
    - Request: to send a message
    - Indication: inform a request was received
    - Confirm: confirm a request to send PDU succeeded



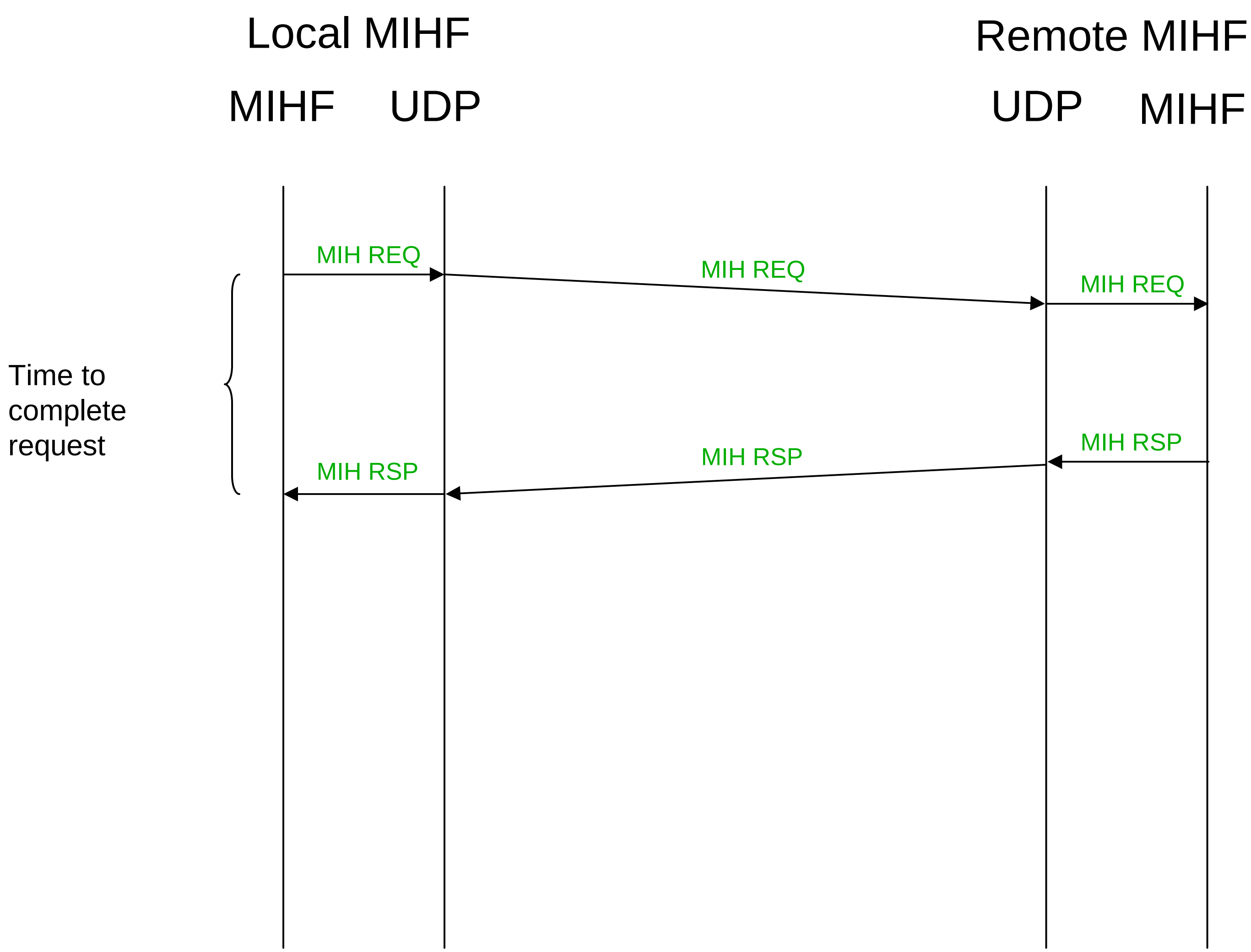
Flow diagrams

- The following slides show the flow diagrams for:
  - UDP
  - TCP
- For indications and requests
- With/without the use of MIH acknowledgement mechanisms

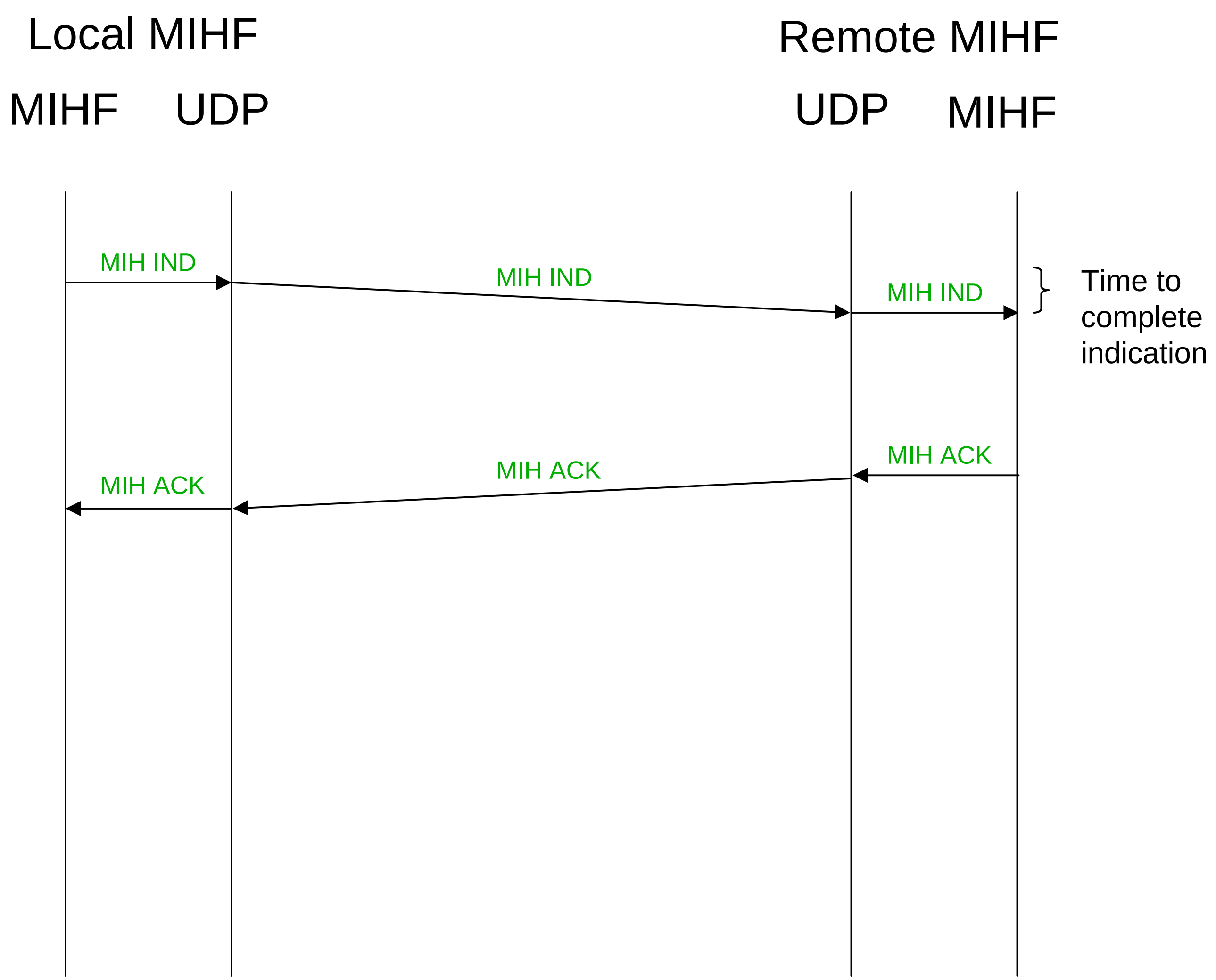
Indication: UDP+No ACK



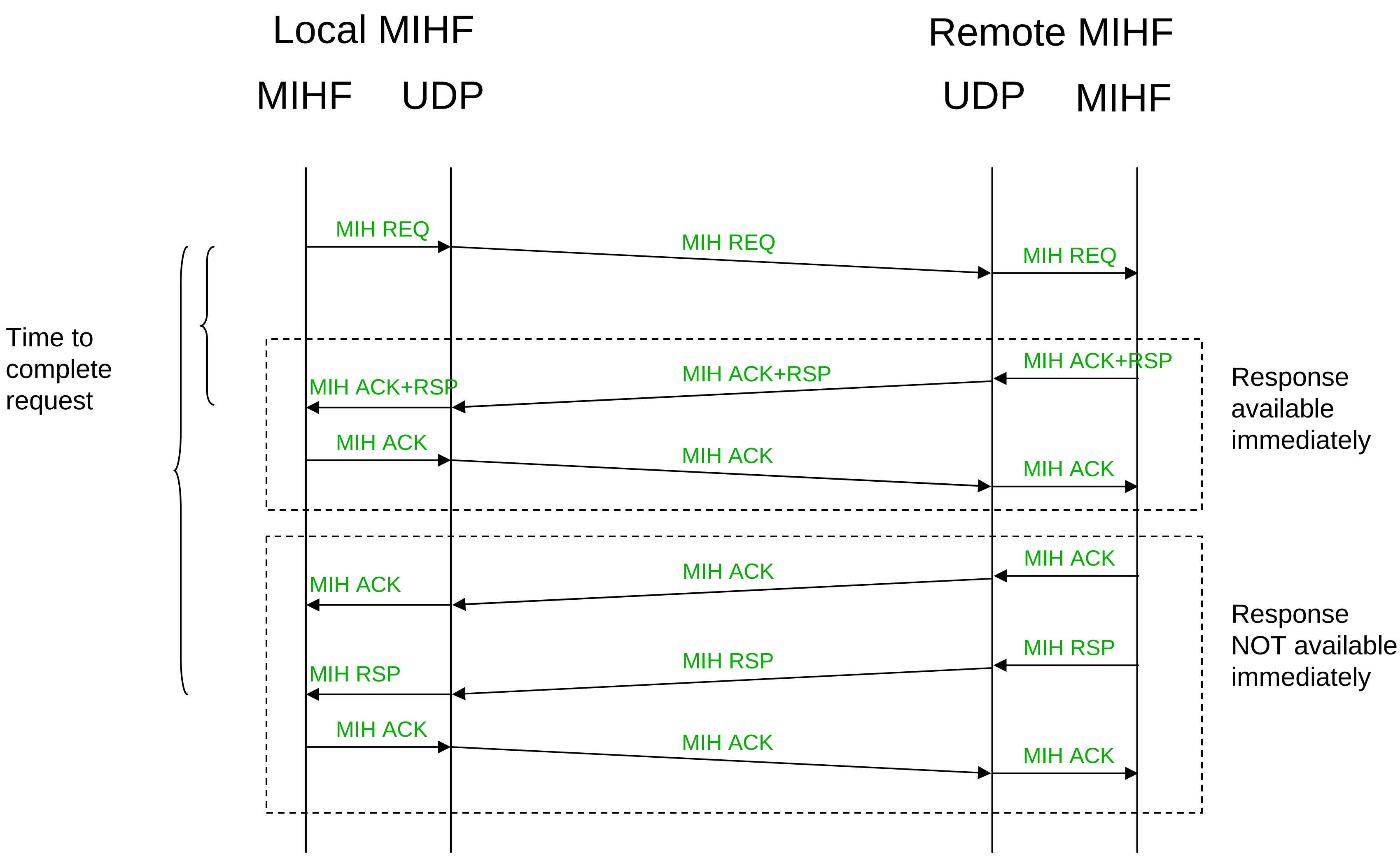
Request: UDP+No ACK



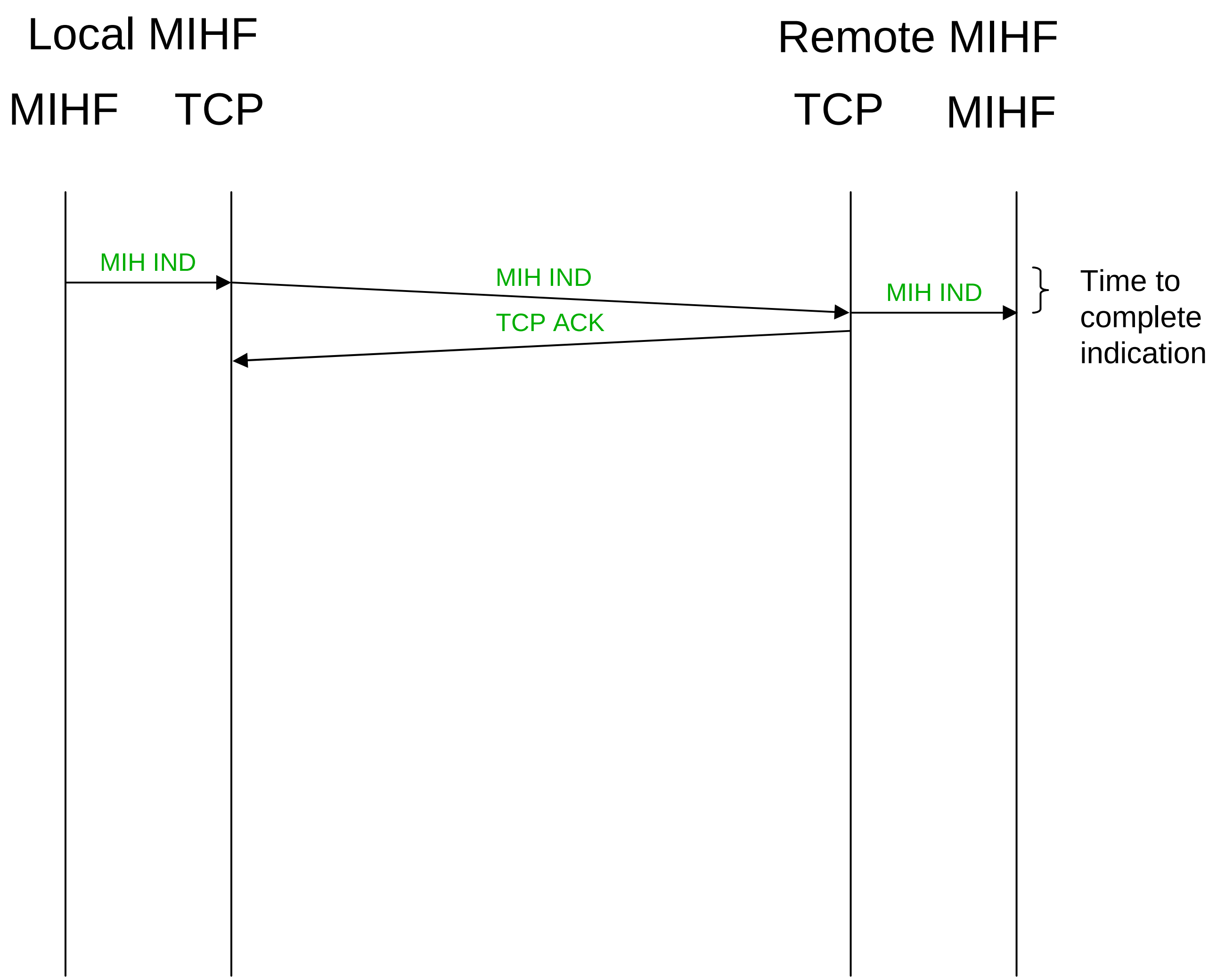
Indication: UDP+ACK



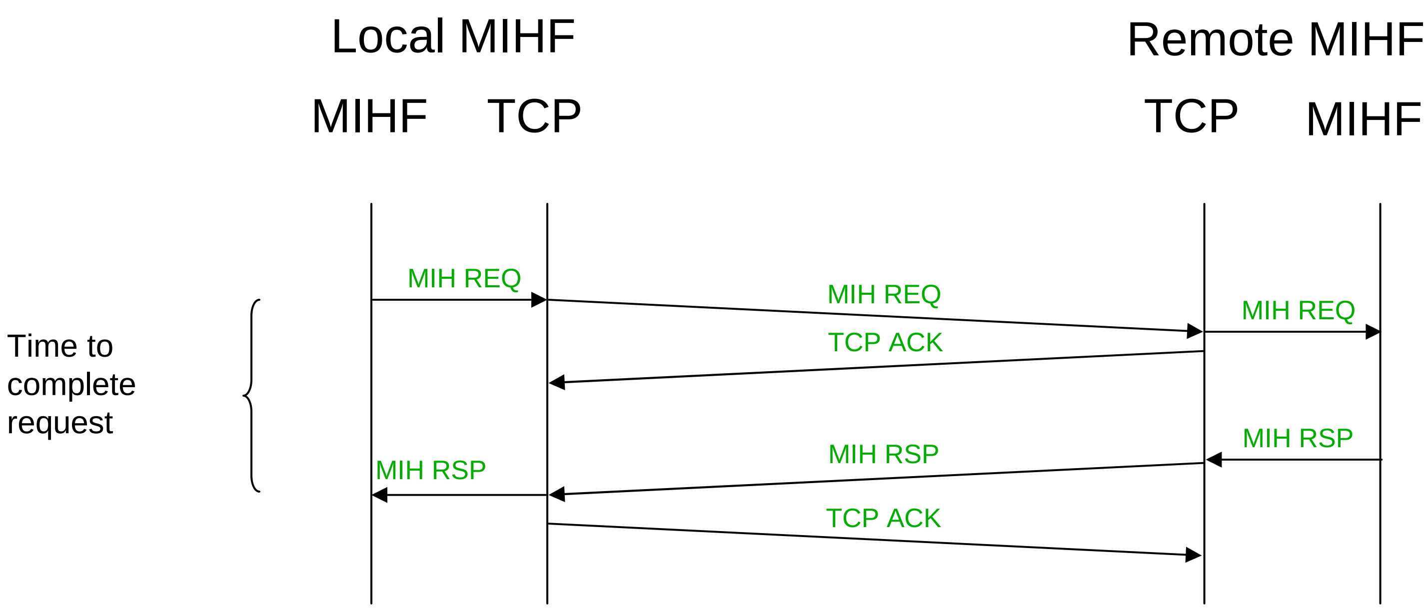
Request: UDP+ACK



Indication: TCP+no ACK

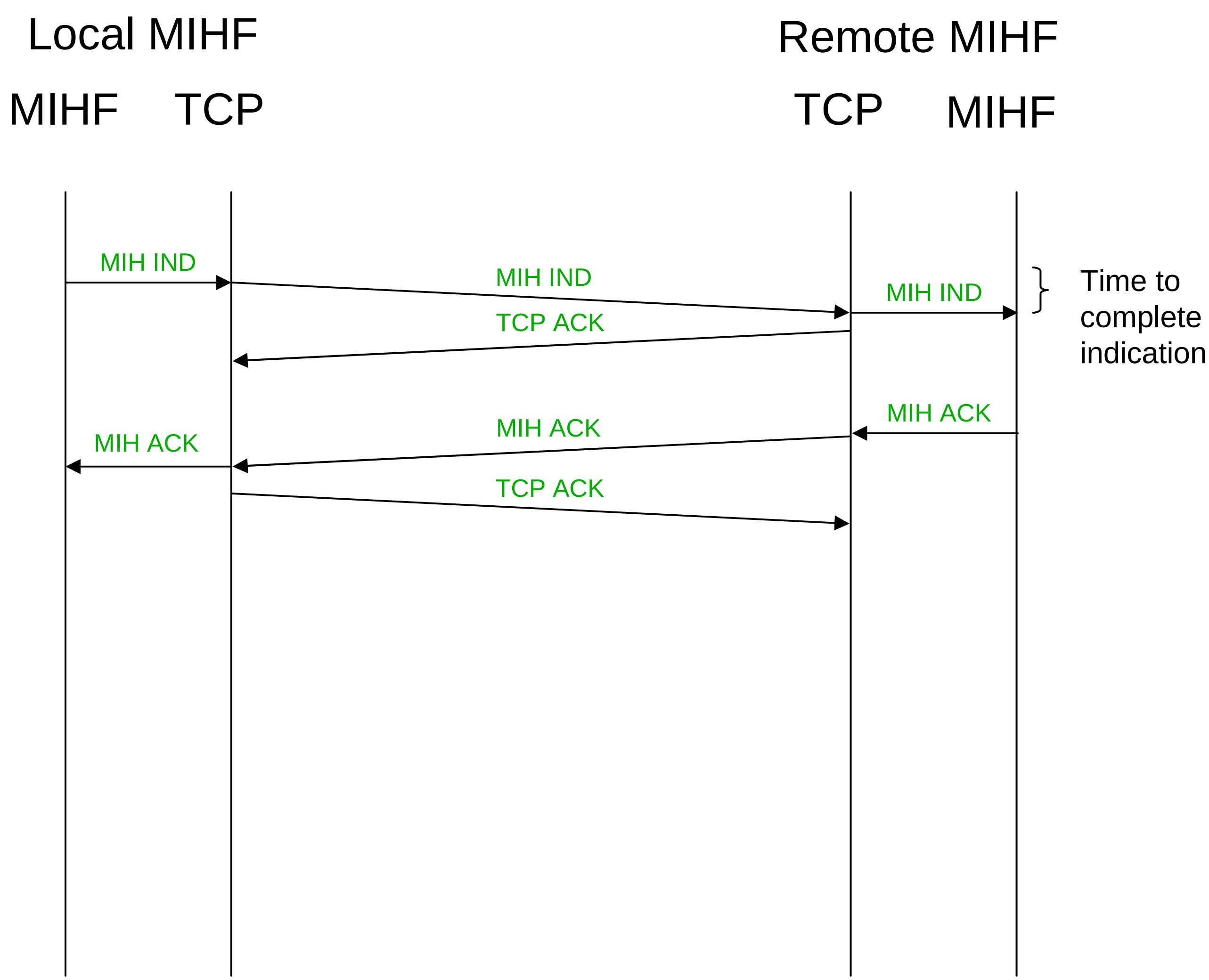


Request: TCP+no ACK

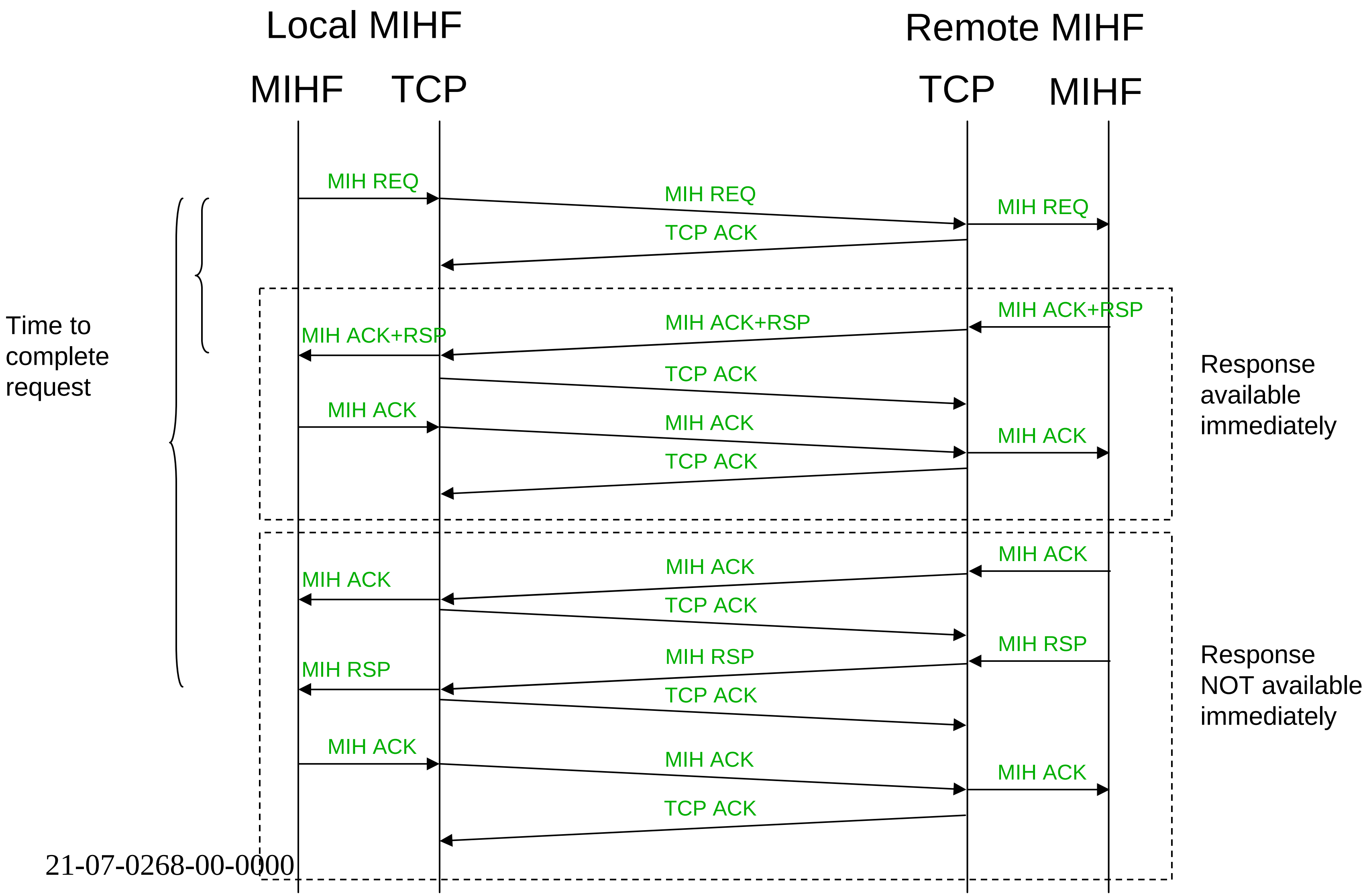


Note: The TCP acknowledgement and the MIH Response may be located in the same TCP segment if TCP delays its acknowledgement

Indication: TCP+ACK



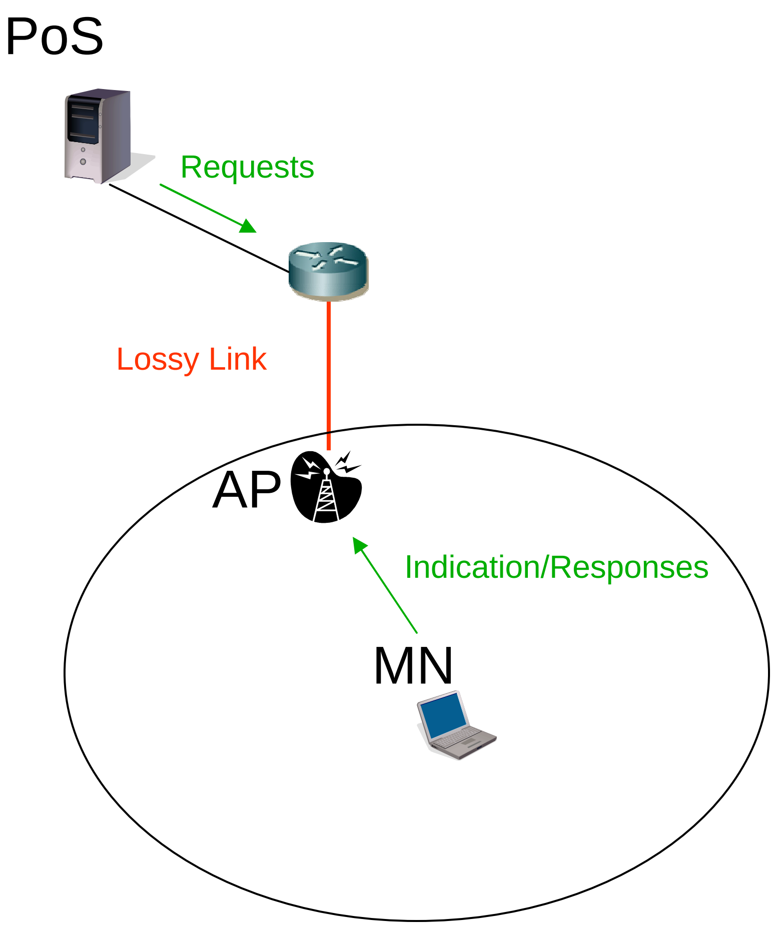
Request: TCP+ACK



Motivation

- What transport layer to use?
  - Choice between UDP versus TCP transport
- What are proper timer values for the MIH ACK?
  - According to the Service (ES/CS/IS)?
  - According to the message type?
  - Maximum number of retransmissions?
- What are the timer values for a response:
  - According to the Service (Service Management/CS)?
  - According to the message type (Scan is longer than probing) parameters)?
- How to handle errors? For example what happens if an ACK is received for the request with no response?
  - Retransmit
  - Abort
  - Use a different interface
- How to set the transaction timer?
  - Short timeout reduces detection of retransmission
  - Long timeout increases memory need

Network Scenario



IEEE 802.11	
Data rate (Mb/s)	11Mb/s
Coverage area – radius (m)	50
Links	
Speed (Mb/s)	100
Delay (s)	0.01
UDP	
Max packet size (byte)	1000
Header size (bytes)	8
TCP	
Maximum segment size (bytes)	1280
Min RTO (s)	0.2
Max retransmission	Unlimited
Queue size	Unlimited
Header size (bytes)	20
IP header	
IPv6 header (bytes)	40
MIH Function	
Transaction timeout (s)	none
Maximum number of retransmission	2
Request processing time (s)	0.2
Simulation configuration	
Duration (s)	6005 with traffic between 5 and 4005
Loss model	None first 5s, variable [0, 50%]
Max RTO for TCP connections (s)	0.2, 0.3, 0.5, 0.75, 1
Number of indications generated (indication/s)	2
Number of requests generated (request/s)	2
MIH Packet size (bytes)	200

Performance Evaluation of the Transport Protocol

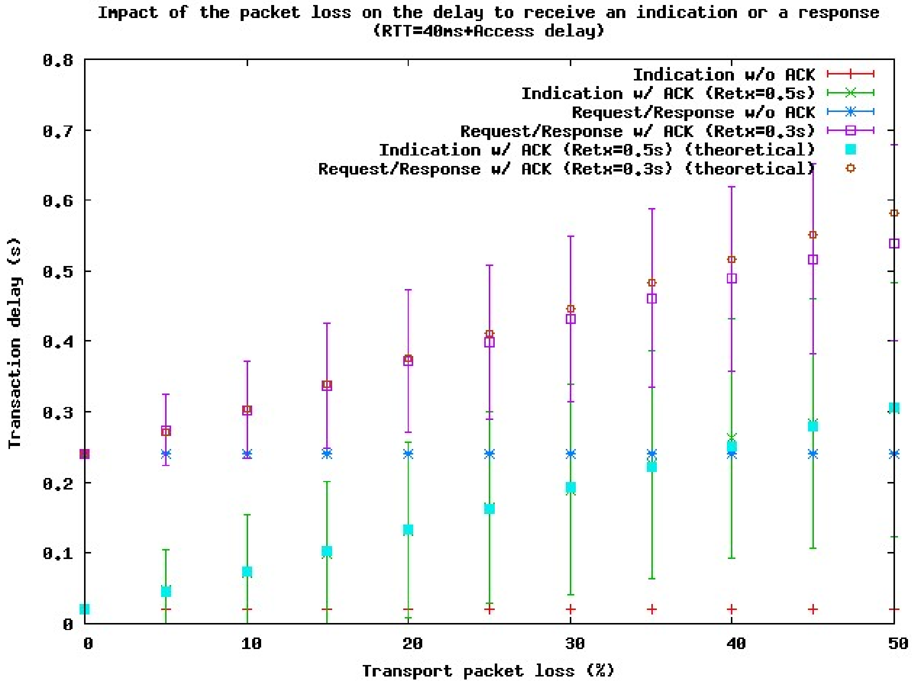
Scenario:
- MN connects to AP
- MN registers with PoS
- Case 1: MN generates indications every 0.5 second
- Case 2: PoS generates requests every 0.5 second

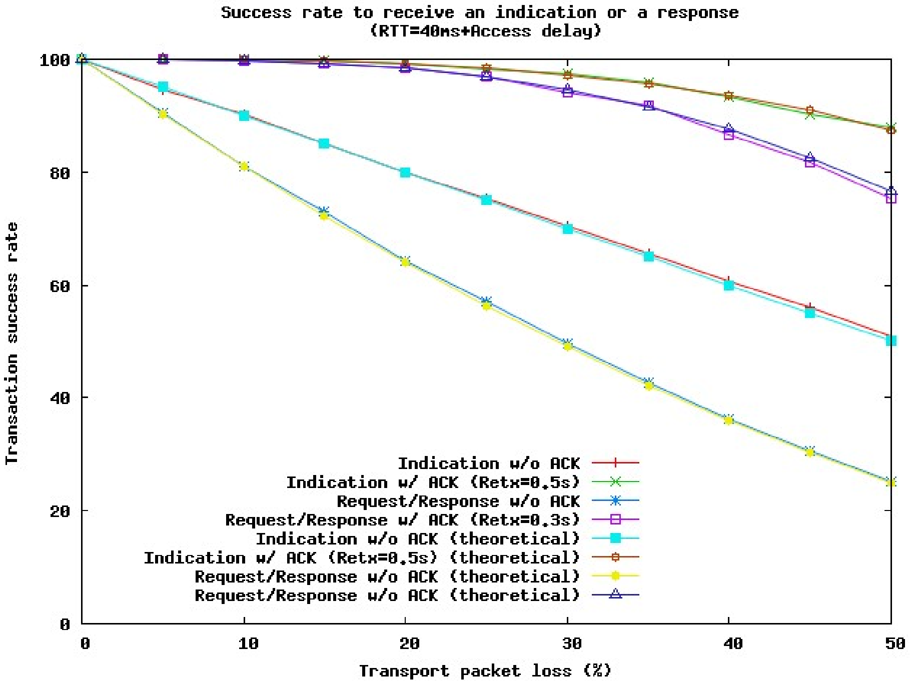
Measurements (average over 4000 seconds of simulation time):
- Transaction success rate (i.e. indication or response received)
- Delay to complete a transaction
- Overhead created by the MIH acknowledgement mechanisms and the transport layer
- Transport throughput

Input parameters:
- Transport layer used (UDP or TCP)
- ACK mechanism at MIH level
- Packet loss in the network [0-50%]
- TCP max retransmission timeout (RTO)
- Timer values for retransmission at MIH level

UDP Performance

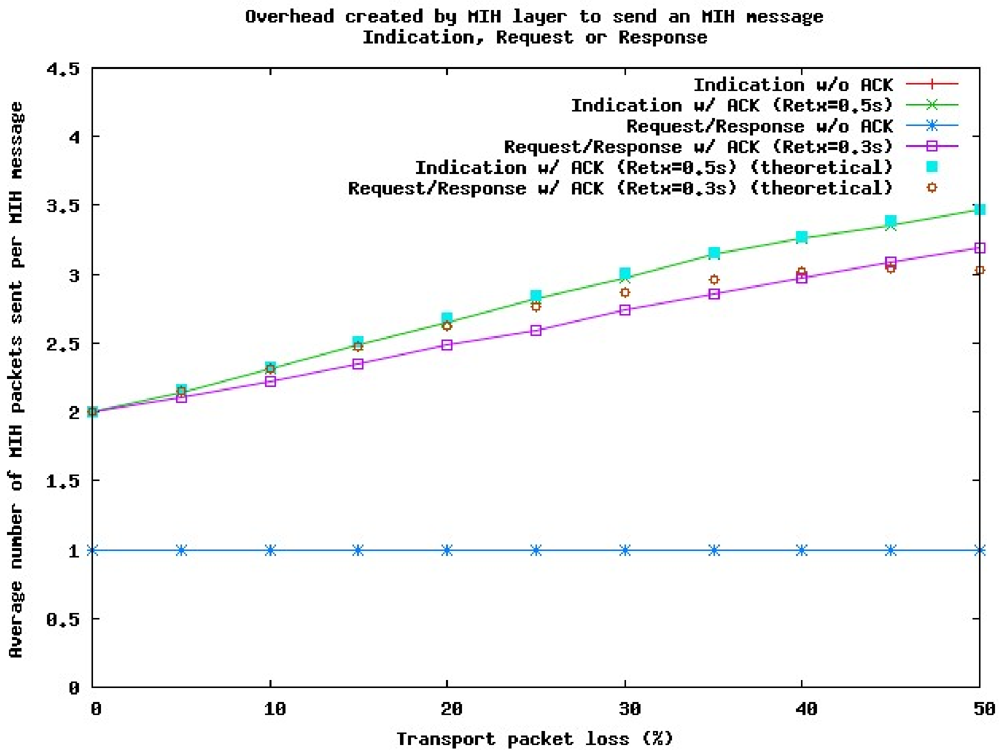
Transaction delay and success rate





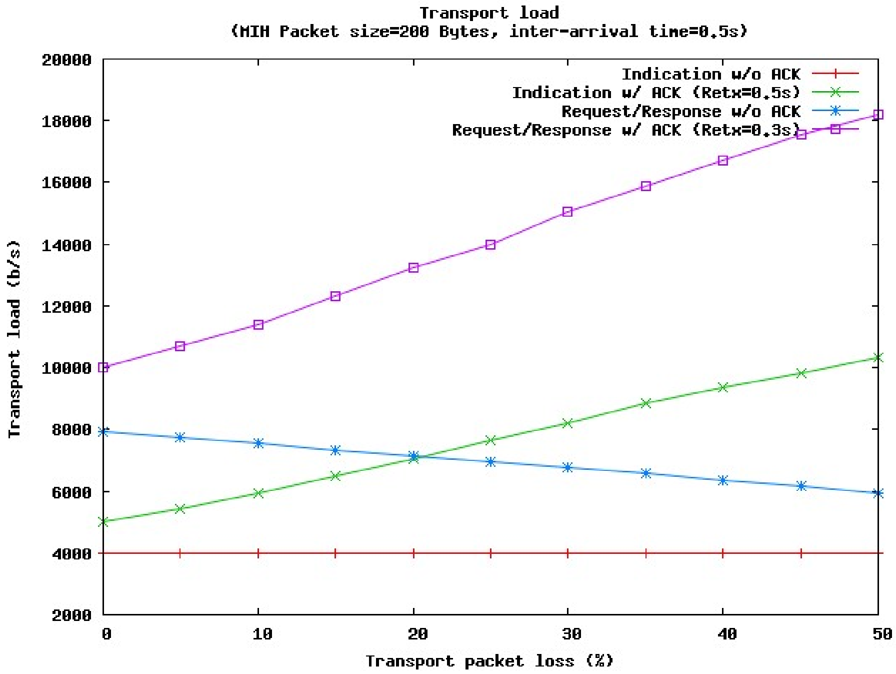
- When MIH ACK is not used, the transmission of the packet must succeed otherwise the transaction fails. Therefore, the delay for the indication transaction is around 20 ms (half the RTT) and 240 ms for the request transaction (RTT+200 ms processing time).
- With MIH ACK, a packet may be retransmitted twice if the acknowledgement is not received, thus increasing the probability to complete a transaction but incurring additional delays proportional to the retransmission timeout.
- The theoretical success rates are as follow (with p=packet loss):
  - For an indication without MIH ACK: Psucc = 1-p
  - For a request without MIH ACK: Psucc = $(1-p)^2$
  - For an indication with MIH ACK: Psucc = $1-p^3$
  - For a request with MIH ACK: Psucc = $(1-p^3)^2$

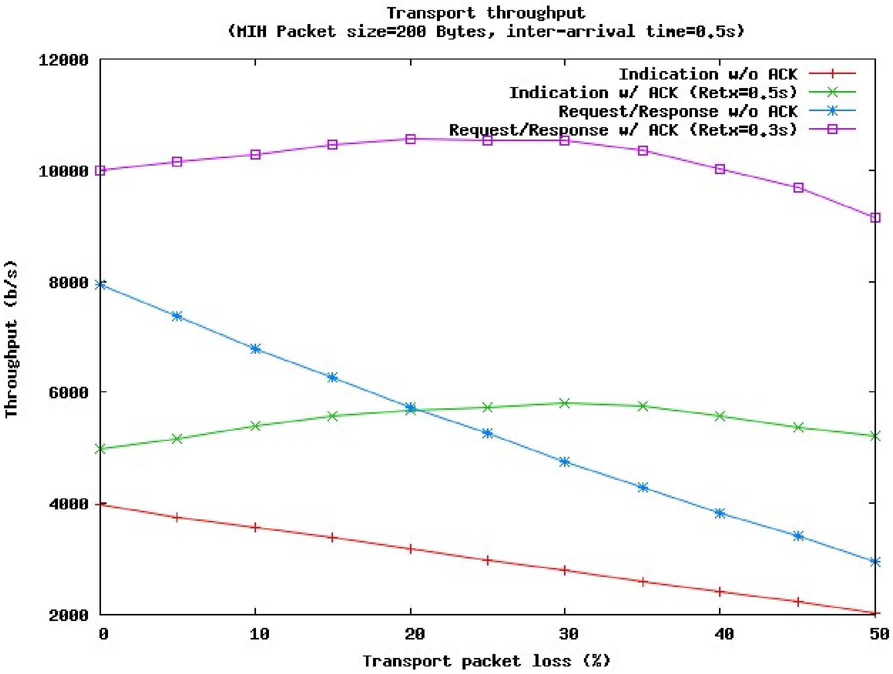
MIH overhead



- The MIH overhead is defined by the number of packets transmitted by the MIHF to the MIH_NET (i.e., transport layer) including retransmission and acknowledgement over the number of MIH messages carrying information (i.e., Indication, Request, or Response).
- When the MIH ACK is used and no packet loss is incurred in the network the overhead is two, since there is an MIH message and an MIH ACK for each MIH message.
- The overhead for requests is lower than for indications, since a response may be ignored by the sender if it arrives late and the MIH ACK is not generated.

UDP load and throughput



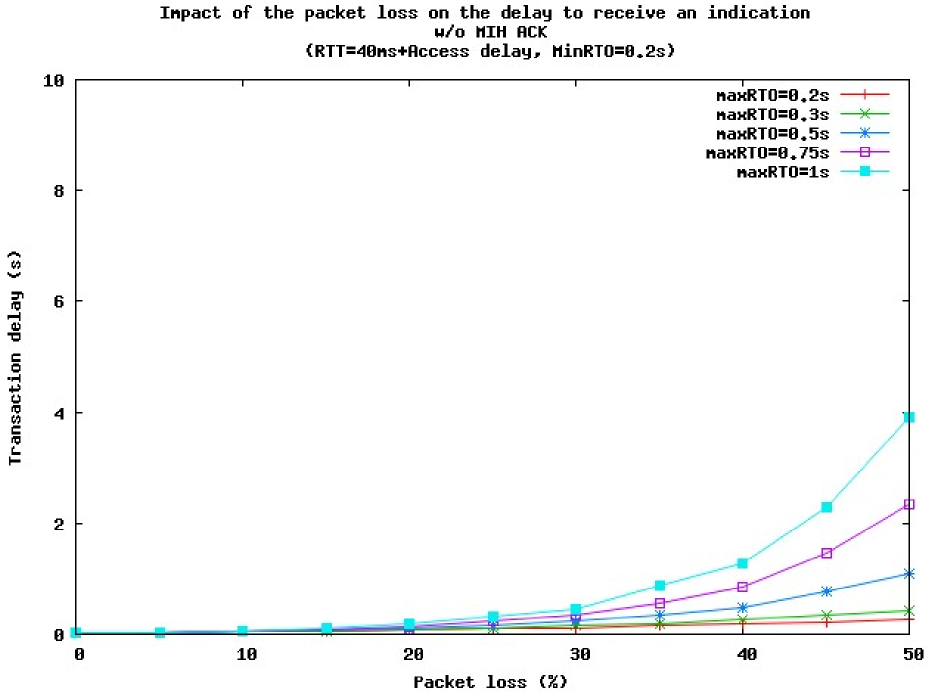


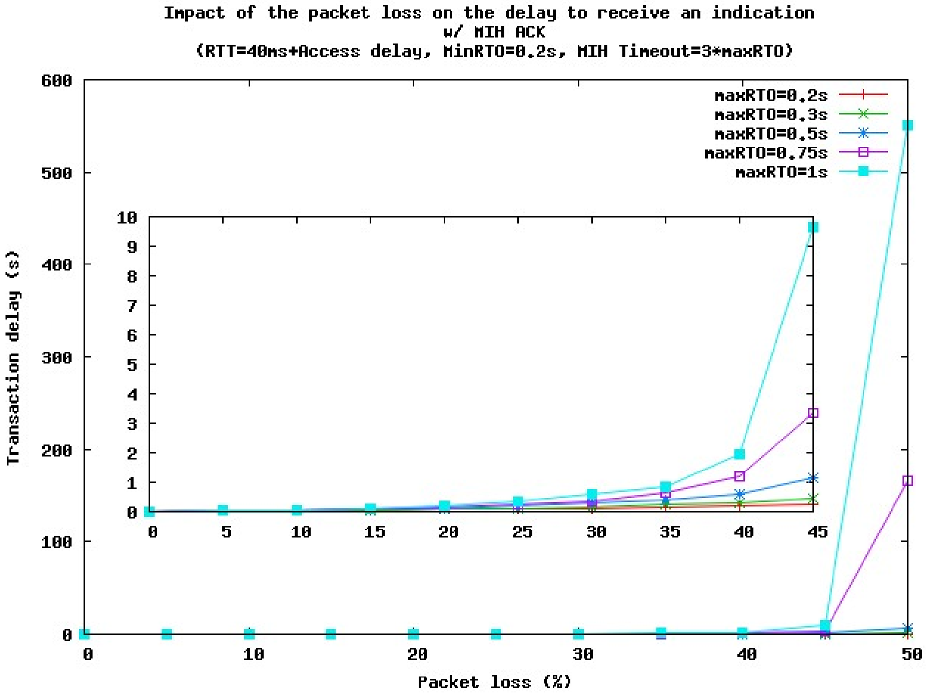
- The graphs show the aggregate traffic generated (load) and received (throughput) by the transport layers, i.e., UDP, between the MN and the PoS.
- When the MIH ACK is used, retransmissions are able to maintain the throughput for low packet losses. The maximum number of retransmissions limits the capabilities of the MIH ACK mechanism.

TCP performance

- When packet loss occurs, TCP retransmits the segments using the Retransmission TimeOut (RTO) value (doubled up to MaxRTO).
- The following results show the impact of the RTO value on the performance of the TCP transport.
- When the MIH ACK is used with TCP, the MIH timeout value is set to 3*MaxRTO in order to let TCP retransmit a lost segment before sending a duplicate MIH message.

Transaction delay for MIH indications

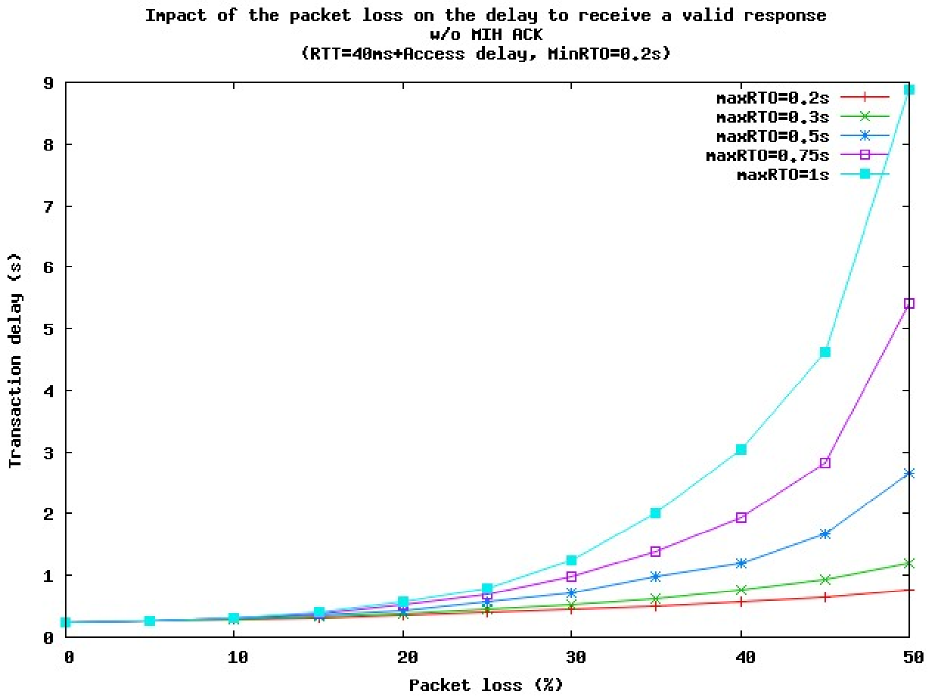


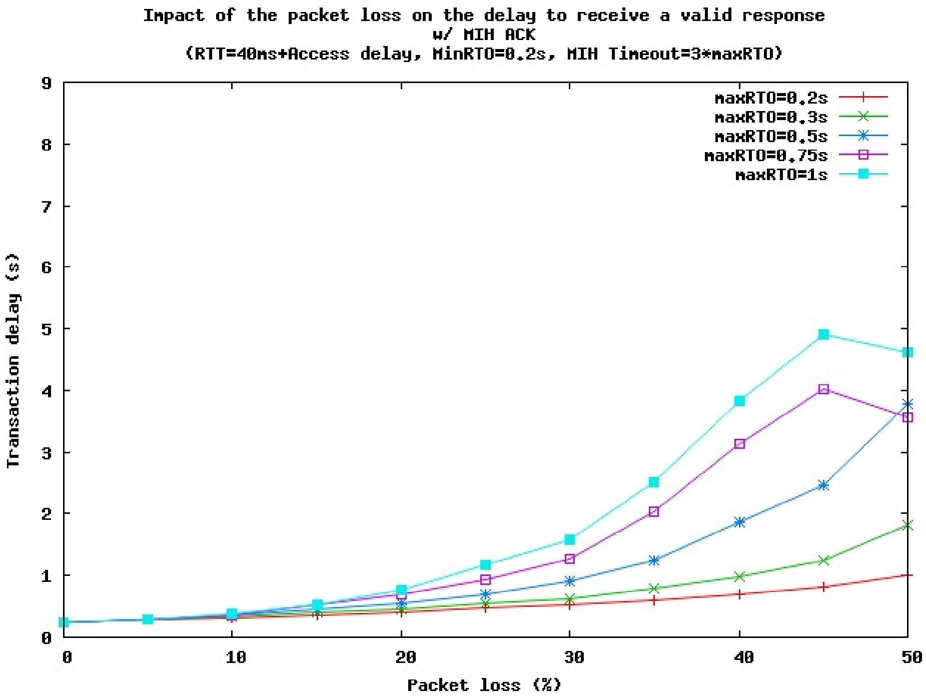


•The delay to perform a transaction increases exponentially with the value of maxRTO.

•When MIH ACK is used and the TCP delays are greater than the MIH retransmission timeout, MIH places duplicate packets in the TCP queue. Since TCP is reliable, these duplicate packets only cause additional delays to transmit useful messages.

Transaction delay for MIH request/response



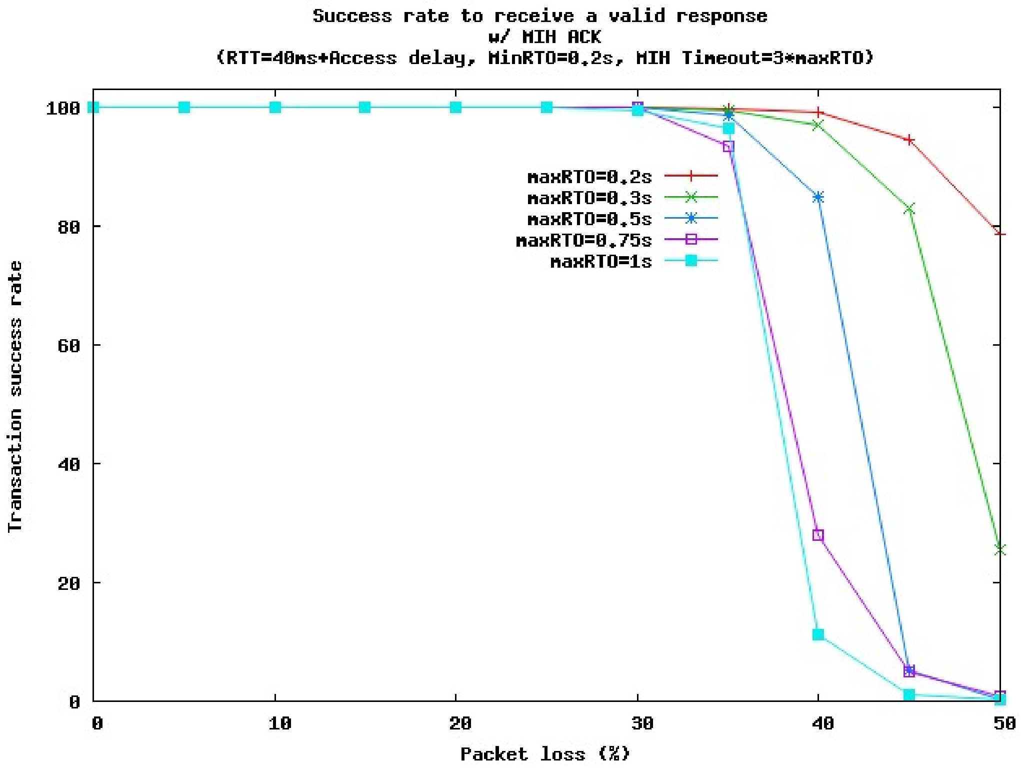


- If no MIH ACK is used, the MIHF sends a request and waits for a response. In our scenario, the responses received are always considered valid although that might not be the case in real implementations.
- When the MIH ACK is used, we observe that the time to receive a response decreases when the packet loss reaches 45% for large values of maxRTO. This is because beyond this packet loss level, the delays are significant and fewer transactions succeed within the ACK timeout interval.

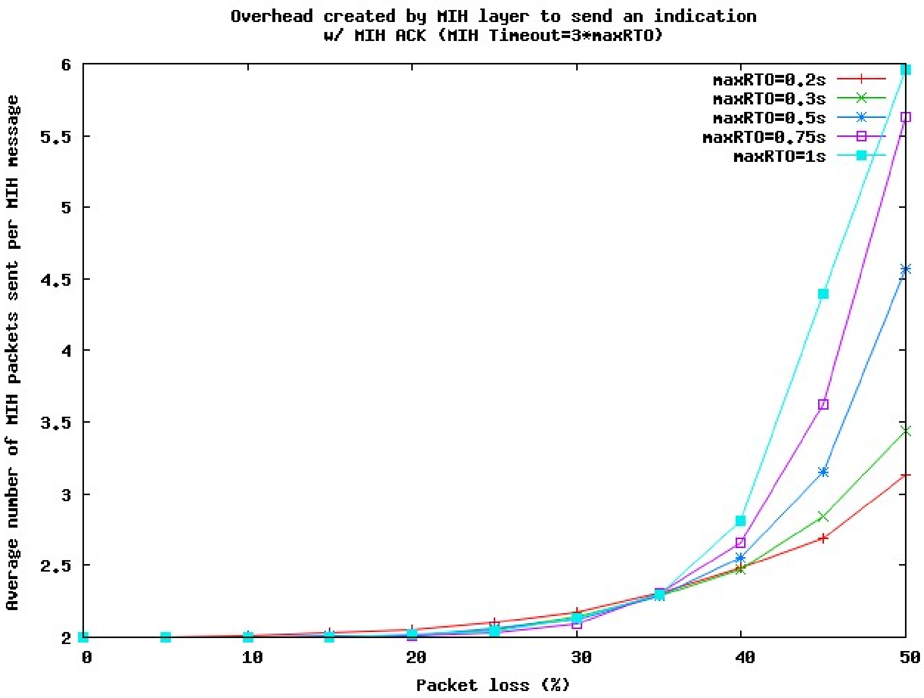
Transaction success rate

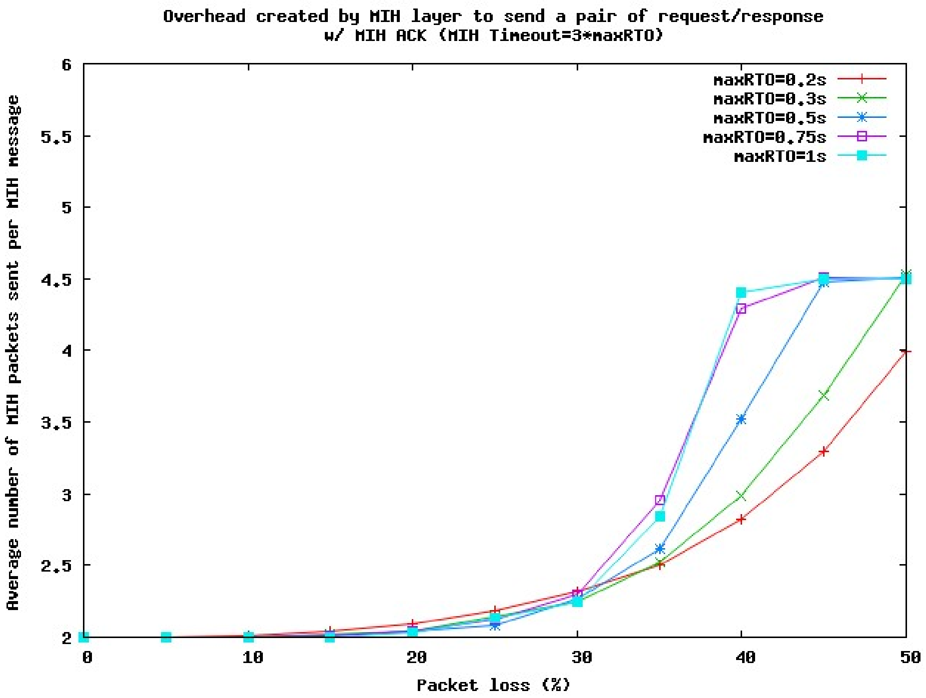
- Since TCP is reliable, the transaction success rate is expected to be 100%.
- This is true as long as the delays to complete the transaction are within the time constraints imposed by the MIH.
- For indications, the receiver processes the indication regardless of when it was sent and therefore the success rate is 100%.
- For requests, the sender expects a response and an ACK (if the MIH acknowledgement is used).

  Late ACKs or responses are discarded.



Overhead generated by the MIH ACK mechanism

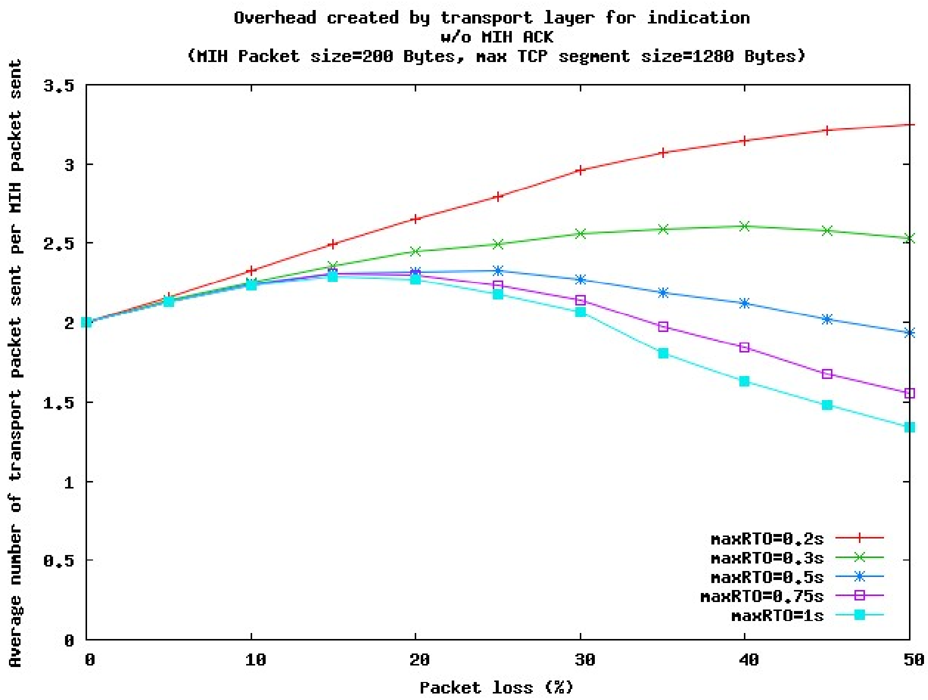


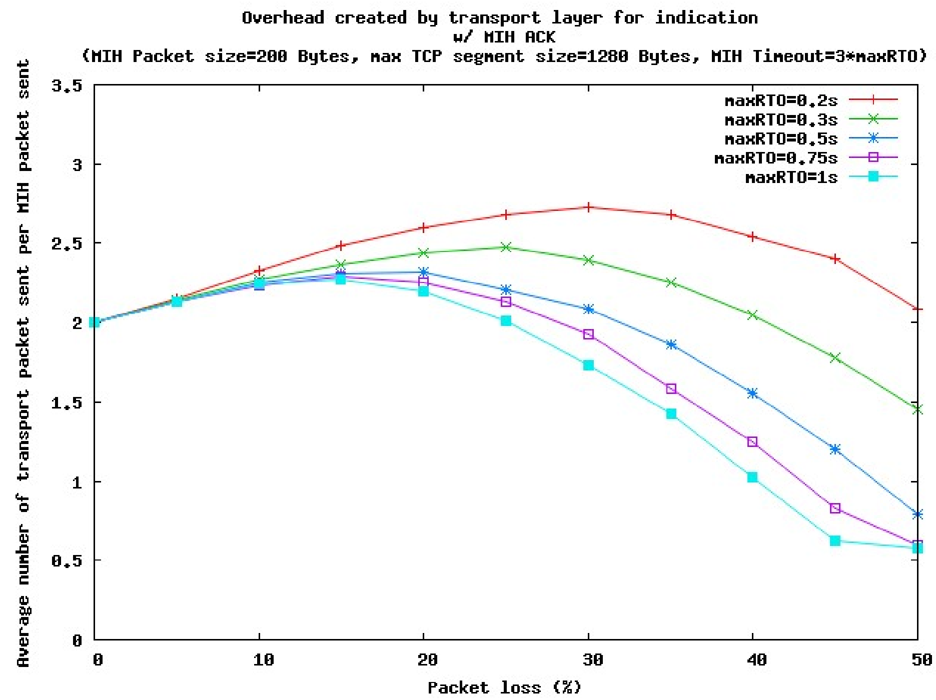


•When there is no packet loss, there is an ACK MIH message sent for each indication/request/response. Therefore the number of MIH packets sent per message is two.

•As packet loss increases, ACK messages are not received and the MIH retransmits up to two times. The maximum number of packets for each MIH indication is 6 (3 Transmissions+3 ACK). For responses, when the TCP delays are too high, the requests arrive late and the generated response will be ignored by the sender. The overhead limit is then 4.5 ((3 transmissions for requests+ 3 ACK + 3 responses) / (1 request+1 response)).

Overhead generated by TCP for MIH indications



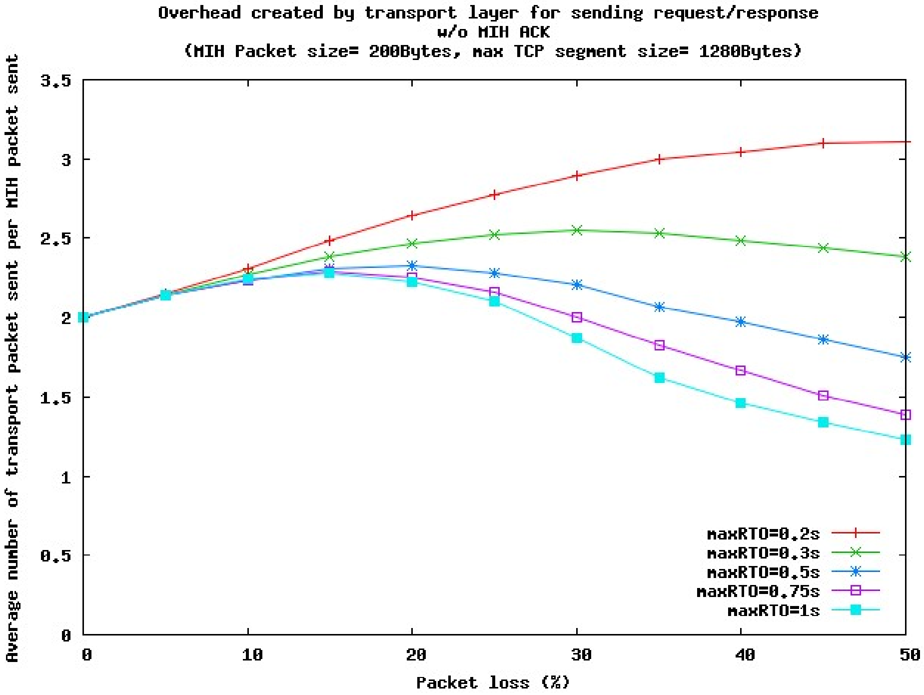


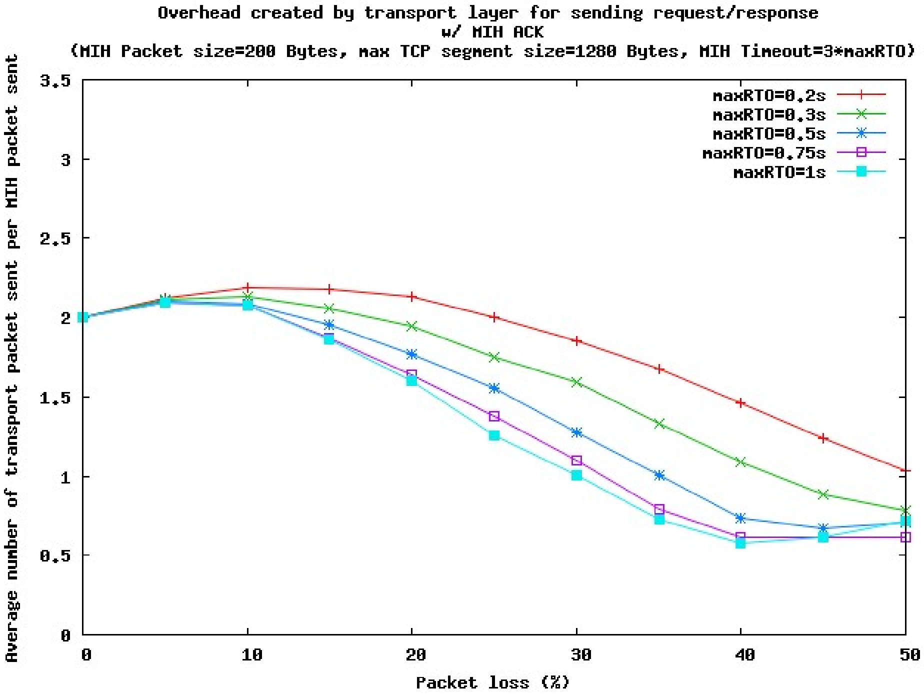
•We observe that as the packet loss increases, the average number of TCP segments sent increases then decreases. This phenomenon can be observed for all values of maxRTO.

•When there is no packet loss, and due to the inter-arrival time of the packets, one MIH packet will be carried in one TCP segment. The TCP ACK is then carried back in another TCP segment creating 2 TCP segments per MIH message.

•Packet loss causes TCP retransmissions. While the packet loss is low, TCP segments will be resent for the same MIH packet but when the packet loss is higher, the TCP queue will grow and TCP will carry multiple MIH packets into one TCP segment, thus reducing the average number of TCP segments per MIH packet.

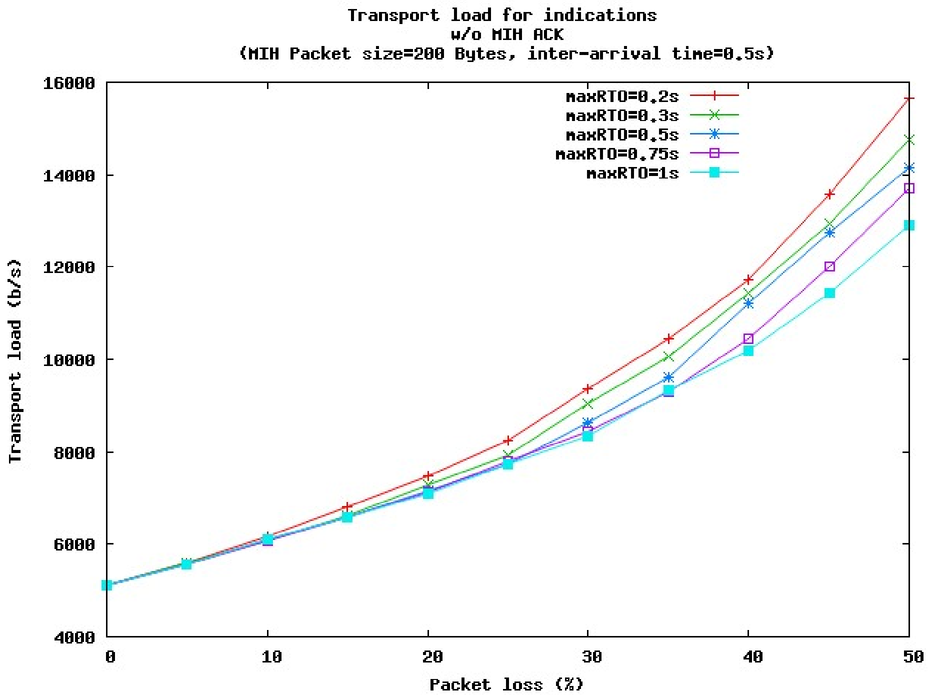
Overhead generated by TCP on MIH requests/responses

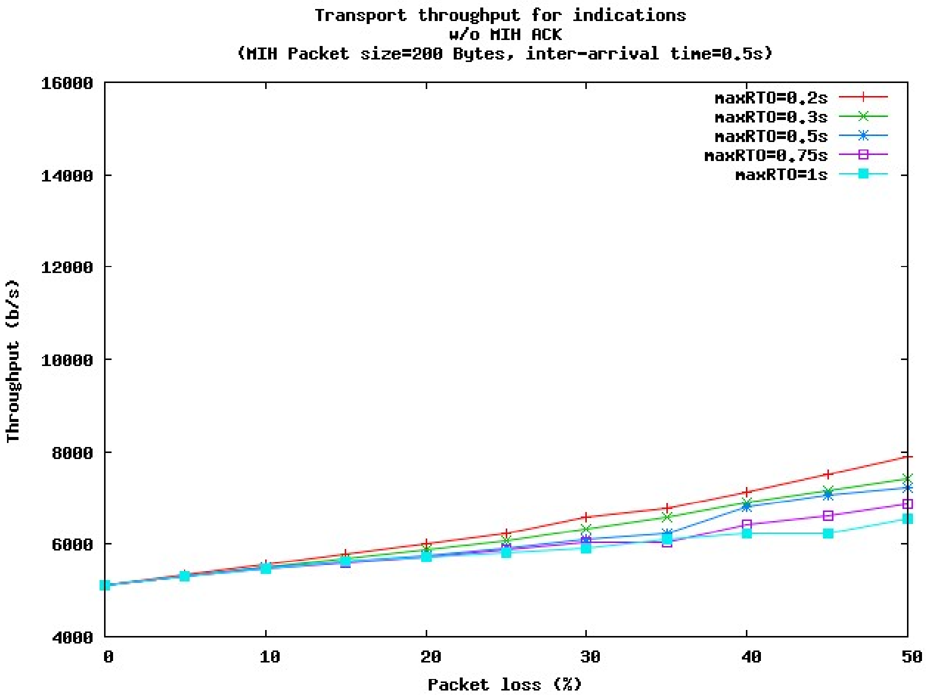




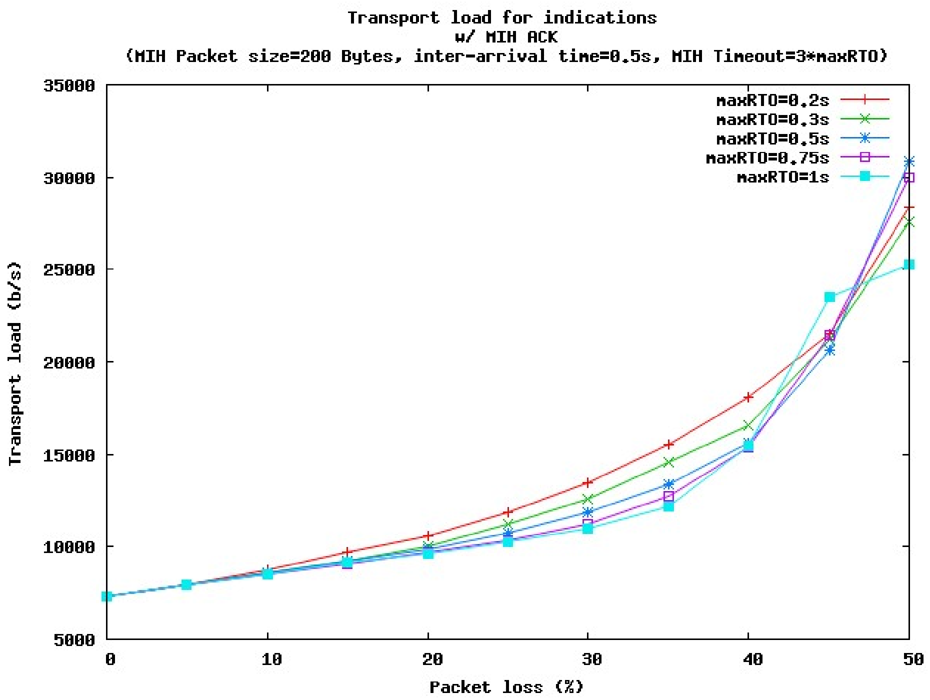
- The observations are similar to the indication but the effects of the TCP segment carrying multiple MIH packet occurs sooner.
- For example, it starts at 10% instead of 20% when MIH acknowledgement is used. This is due to an increase in the amount of data transported.

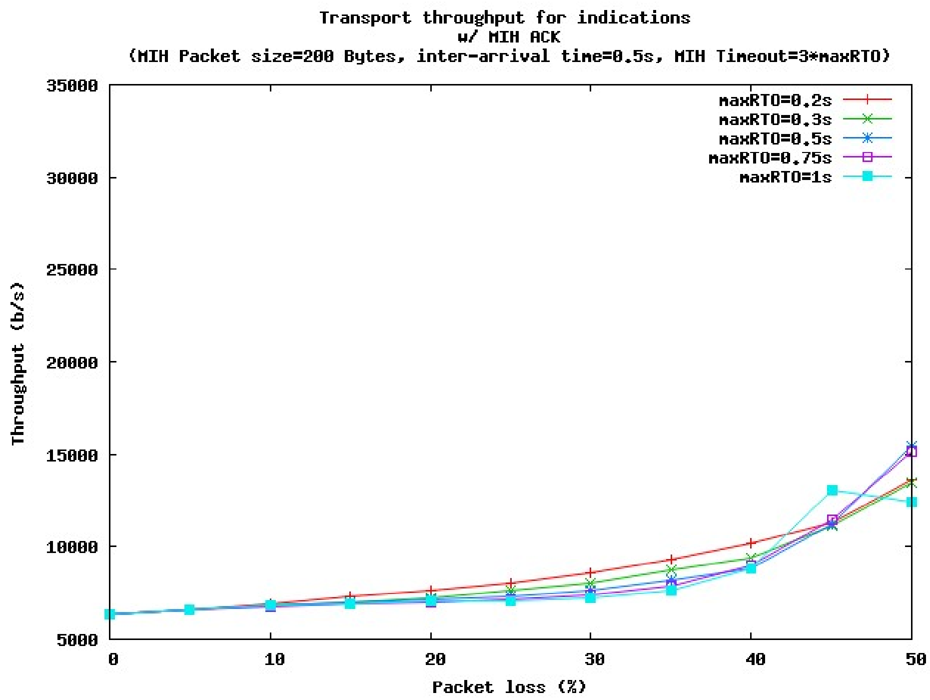
TCP load and throughput for MIH indications without MIH ACK





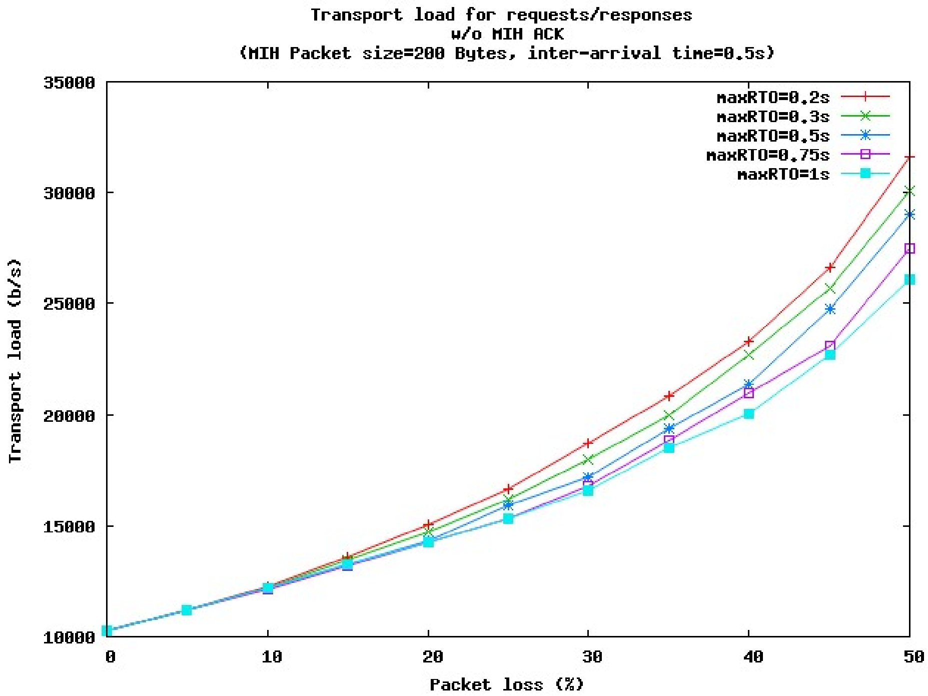
TCP load and throughput for MIH indications with MIH ACK

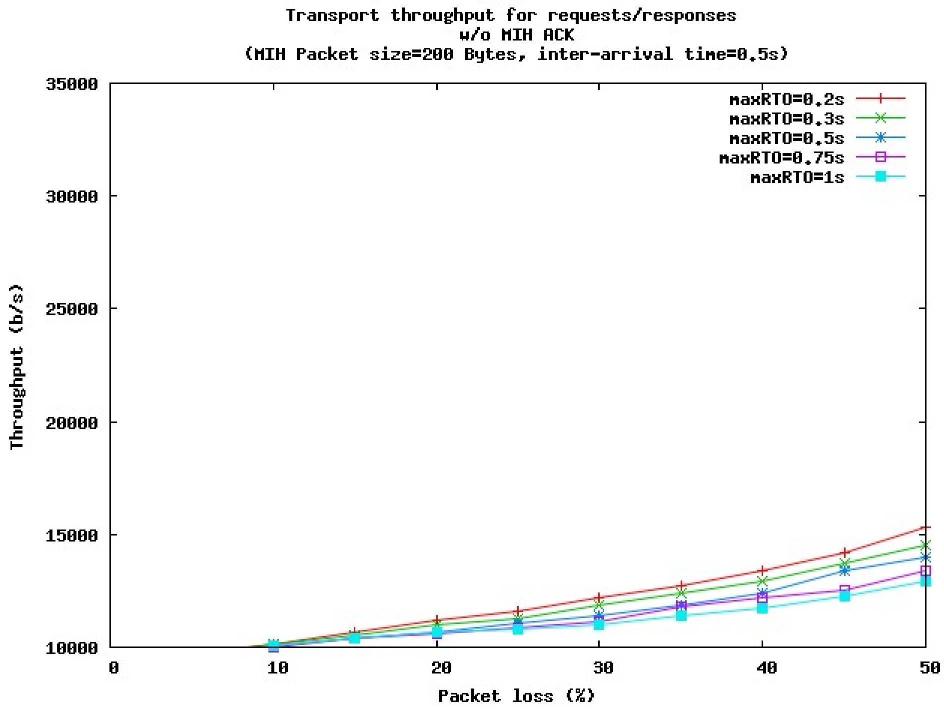




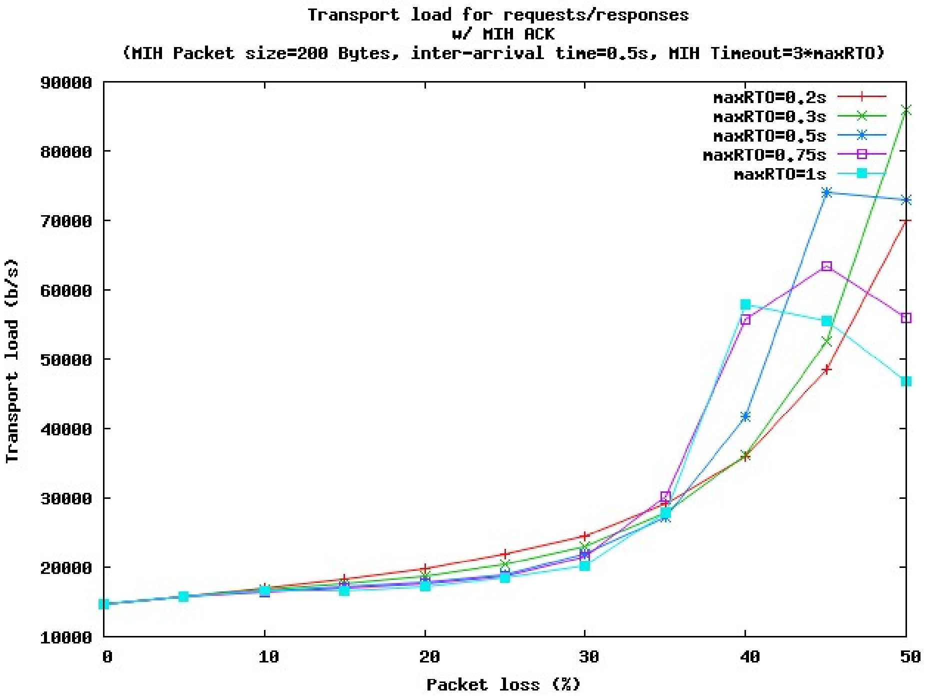
•We observe the load increasing with the packet loss since TCP and MIH retransmit data. Since the number of retransmissions of MIH messages is limited, the load passed to TCP has a maximum value. When reached, TCP will keep on retransmitting and taking more time to send the data thus the load will slow down as shown when maxRTO=1s.

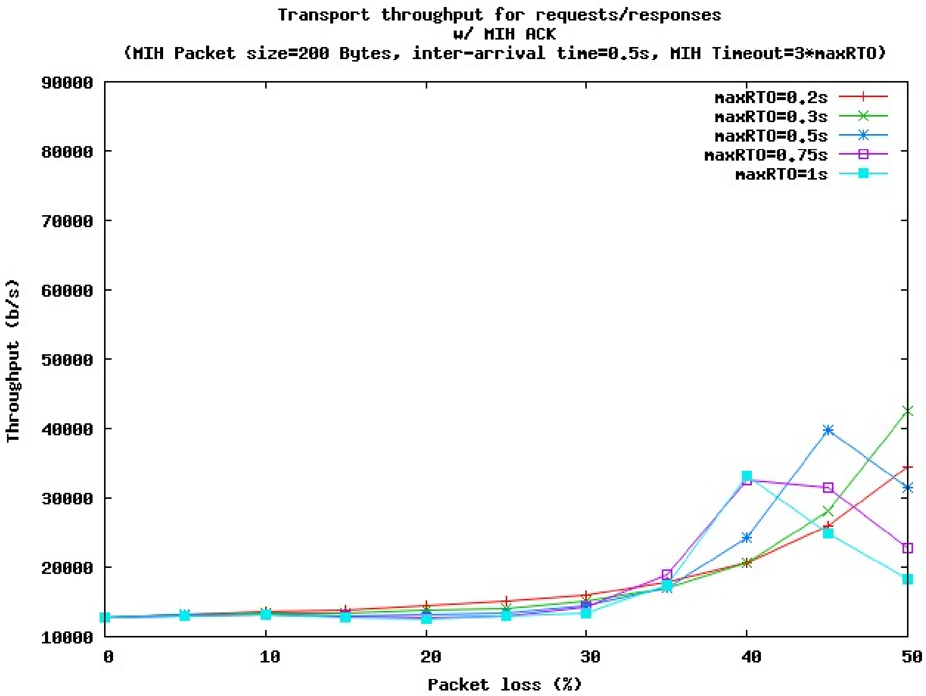
TCP load and throughput for MIH requests without MIH ACK





TCP load and throughput for MIH requests with MIH ACK





- For low packet losses, TCP and MIH retransmitting packets increases the load.
- If the sending MIHF does not receive an ACK for a request, it will consider the transaction failed and will not respond to the responses that are coming late. This happens when the TCP delays are too high due to packet loss. When packet loss is high, this happens more often thus reducing the data sent by the transport layer.