Disclaimer

- The material provided in this document is not my original work and is a summary of some one else's work(s).
- A simple Google search of the title of the document will direct you to the original source of the material.
- I do not guarantee the accuracy, completeness, timeliness, validity, non-omission, merchantability or fitness of the contents of this document for any particular purpose.
-

Pushing The Limits of CAN

Scheduling Frames With Offsets Provides A Major Performance Boost

Presented By

Pushing The Limits of CAN

Scheduling Frames With Offsets Provides A Major Performance Boost

Presented By
Najeeb
2013 – 4 – 18

Outline

- Introduction
- Offset Assignment Algorithm
- Worst Case Response Times Results
- Offsets for Higher Network Loads
- Conclusion

Introduction

Introduction

- The data traffic in automotive networks is rapidly growing

Introduction

- The data traffic in automotive networks is rapidly growing
- In CAN, Worst Case Response Times increase drastically with the load

Introduction

- The data traffic in automotive networks is rapidly growing
- In CAN, Worst Case Response Times increase drastically with the load
- In order to remain as a prominent automotive network, some enhancements in CAN are required to satisfy the future high bandwidth requirements.

Introduction Contd…

Introduction Contd…

- WCRT for a frame corresponds to the scenario where all higher priority CAN messages are released synchronously

Introduction Contd…

- WCRT for a frame corresponds to the scenario where all higher priority CAN messages are released synchronously
- This situation can be avoided by scheduling stream of messages with offsets

Introduction Contd…

- WCRT for a frame corresponds to the scenario where all higher priority CAN messages are released synchronously
- This situation can be avoided by scheduling stream of messages with offsets
- The first instance of a stream of periodic frames is released with a delay, called the offset, relative to the first time at which the station is ready to transmit

Introduction Contd…

- WCRT for a frame corresponds to the scenario where all higher priority CAN messages are released synchronously
- This situation can be avoided by scheduling stream of messages with offsets
- The first instance of a stream of periodic frames is released with a delay, called the offset, relative to the first time at which the station is ready to transmit
- The challenge is to set the offsets in such a way as to minimize the WCRT

Offset Assignment Algorithm

Offset Assignment Algorithm

- The underlying idea of the algorithm is to distribute the workload as uniformly as possible over time

Offset Assignment Algorithm

- The underlying idea of the algorithm is to distribute the workload as uniformly as possible over time
- Avoiding synchronous releases leading to traffic peaks and thus to large frame response times

Offset Assignment Algorithm

- The underlying idea of the algorithm is to distribute the workload as uniformly as possible over time
- Avoiding synchronous releases leading to traffic peaks and thus to large frame response times
- Tries to schedule the transmissions as far apart as possible

Offset Assignment Algorithm

- The underlying idea of the algorithm is to distribute the workload as uniformly as possible over time
- Avoiding synchronous releases leading to traffic peaks and thus to large frame response times
- Tries to schedule the transmissions as far apart as possible
- The algorithm is executed on each station independently

Design Hypotheses

Design Hypotheses

- There are only a few distinct values for the periods (e.g. 5 to 10)

Design Hypotheses

- There are only a few distinct values for the periods (e.g. 5 to 10)
- The time is discrete with a certain granularity g

Design Hypotheses

- There are only a few distinct values for the periods (e.g. 5 to 10)
- The time is discrete with a certain granularity g
- A time instant that is a multiple of g is called a Possible Release Time i^{th} PRT occurs at time (i-1)*g

Notations

Notations

- On station *i*, the *k*th stream of frames, denoted by f^i_k, is characterized by the tuple $(C^i_k, D^i_k, T^i_k, O^i_k)$

Notations

- On station *i*, the *k*th stream of frames, denoted by f^i_k, is characterized by the tuple (C^i_k, D^i_k, T^i_k, O^i_k)
  - C^i_k = Worst Case Transmission Time

Notations

- On station i, the kth stream of frames, denoted by f^i_k, is characterized by the tuple $(C^i_k, D^i_k, T^i_k, O^i_k)$
  - C^i_k = Worst Case Transmission Time
  - D^i_k = Relative Deadline

Notations

- On station *i*, the *k*th stream of frames, denoted by f^i_k, is characterized by the tuple (C^i_k, D^i_k, T^i_k, O^i_k)
  - C^i_k = Worst Case Transmission Time
  - D^i_k = Relative Deadline
  - T^i_k = Inter-Arrival Time Period

Notations

- On station *i*, the *k*th stream of frames, denoted by f^i_k, is characterized by the tuple (C^i_k, D^i_k, T^i_k, O^i_k)
  - C^i_k = Worst Case Transmission Time
  - D^i_k = Relative Deadline
  - T^i_k = Inter-Arrival Time Period
  - O^i_k = Offset: The duration between the first instant at which the station is operational and the transmission of the first frame of stream f^i_k

Analysis Duration

Analysis Duration

- For each stream f_k the offset is chosen in the interval $[0,T_k]$ so generally analysis is performed on the interval $[0,T_{max}]$ where $T_{max} = max\{T_k\}$

Analysis Duration

- For each stream f_k the offset is chosen in the interval $[0,T_k]$ so generally analysis is performed on the interval $[0,T_{max}]$ where $T_{max} = max\{T_k\}$
- The release times of the frames in the interval $[0,T_{max}]$ are stored in an array R having T_{max}/g elements where the *i*th element R[*i*] is the set of frames released at PRT *i* (i.e. at time $(i - 1) * g$)

Analysis Duration

- For each stream f_k the offset is chosen in the interval $[0,T_k]$ so generally analysis is performed on the interval $[0,T_{max}]$ where $T_{max} = \max\{T_k\}$
- The release times of the frames in the interval $[0,T_{max}]$ are stored in an array R having T_{max}/g elements where the *i*th element R[*i*] is the set of frames released at PRT *i* (i.e. at time $(i - 1) * g$)
- Lets look at an example

Analysis Duration

$f_1 = (T_1 = 10, O_1 = 4)$, $f_2 = (20, 8)$ and $f_3 = (20, 18)$ $(T_{\max} = 20)$ with a granularity $g = 2$.

time	0	2	4	6	8	10	12	14	16	18
possible release time i	1	2	3	4	5	6	7	8	9	10
$R[i]$ (frames released)			$f_{1,1}$		$f_{2,1}$			$f_{1,2}$		$f_{3,1}$

Adjacent PRT

Adjacent PRT

- For a stream f_k and a time granularity g, the Possible Release Times *i* and *i'* are adjacent iff:

Adjacent PRT

- For a stream f_k and a time granularity g, the Possible Release Times *i* and *i'* are adjacent iff:

$$\left|\left(i \mod \frac{T_k}{g}\right) - \left(i' \mod \frac{T_k}{g}\right)\right| = 1$$

Adjacent PRT

- For a stream f_k and a time granularity g, the Possible Release Times *i* and *i'* are adjacent iff:

$$\left| \left(i \mod \frac{T_k}{g} \right) - \left(i' \mod \frac{T_k}{g} \right) \right| = 1$$

- Consider an example stream f_1 having $T_1 = 10$ and g = 2

Adjacent PRT

- For a stream f_k and a time granularity g, the Possible Release Times *i* and *i'* are adjacent iff:

$$\left| \left(i \mod \frac{T_k}{g} \right) - \left(i' \mod \frac{T_k}{g} \right) \right| = 1$$

- Consider an example stream f_1 having $T_1 = 10$ and g = 2

time	0	2	4	6	8
possible release time i	1	2	3	4	5
possible release times adjacent to i	{5,2}	{1,3}	{2,4}	{3,5}	{4,1}

Interval

Interval

- For a stream f_k, an interval is an ordered set of possible release times where the *i*th and $(i + 1)$th elements are adjacent.

Interval

- For a stream f_k , an interval is an ordered set of possible release times where the *i*th and $(i + 1)$th elements are adjacent.
- Example of an interval for f_1 is {4,5,1,2}

Interval

- For a stream f_k , an interval is an ordered set of possible release times where the *i*th and (*i* + 1)th elements are adjacent.
- Example of an interval for f_1 is {4,5,1,2}
- Load of PRT *i* is the number of releases scheduled for transmission at *i* i.e. clock time (*i*-1) * g

Interval

- For a stream f_k , an interval is an ordered set of possible release times where the *i*th and (*i* + 1)th elements are adjacent.
- Example of an interval for f_1 is {4,5,1,2}
- Load of PRT *i* is the number of releases scheduled for transmission at *i* i.e. clock time (*i*-1) * g
- We consider only the intervals made of PRT with the same Load

Interval

- For a stream f_k , an interval is an ordered set of possible release times where the *i*th and (*i* + 1)th elements are adjacent.
- Example of an interval for f_1 is {4,5,1,2}
- Load of PRT *i* is the number of releases scheduled for transmission at *i* i.e. clock time (*i*-1) * g
- We consider only the intervals made of PRT with the same Load
- The Least Loaded Intervals only comprise possible release times having a load equal to l_k ; the smallest Load in the interval [0 T_k]

Interval

time	0	2	4	6	8	10	12	14	16	18
possible release time i	1	2	3	4	5	6	7	8	9	10
$R[i]$ (frames released)			$f_{1,1}$		$f_{2,1}$			$f_{1,2}$		$f_{3,1}$

- Load of PRT *i* is the number of releases scheduled for transmission at *i* i.e. clock time (*i*-1) * g
- We consider only the intervals made of PRT with the same Load
- The Least Loaded Intervals only comprise possible release times having a load equal to l_k ; the smallest Load in the interval [0 T_k]

Description of The Algorithm

Description of The Algorithm

- Message Streams are sorted by increasing value of their period

Description of The Algorithm

- Message Streams are sorted by increasing value of their period
- Let us consider that the stream under analysis is f_k

Description of The Algorithm

- Message Streams are sorted by increasing value of their period
- Let us consider that the stream under analysis is f_k
  A. Look for l_k in the interval $[0, T_k]$

Description of The Algorithm

- Message Streams are sorted by increasing value of their period
- Let us consider that the stream under analysis is f_k
  A. Look for l_k in the interval $[0, T_k]$
  B. Look for one of the Longest Least Loaded Intervals in $[0, T_k]$, B_k = First PRT , E_k = Last PRT

Description of The Algorithm

- Message Streams are sorted by increasing value of their period
- Let us consider that the stream under analysis is f_k
  A. Look for l_k in the interval $[0, T_k]$
  B. Look for one of the Longest Least Loaded Intervals in $[0, T_k]$, B_k = First PRT , E_k = Last PRT
  C. Set the offset O_k in the middle of the selected interval, the corresponding possible release time is r_k

Description of The Algorithm

- Message Streams are sorted by increasing value of their period
- Let us consider that the stream under analysis is f_k
  A. Look for l_k in the interval $[0, T_k]$
  B. Look for one of the Longest Least Loaded Intervals in $[0, T_k]$, B_k = First PRT , E_k = Last PRT
  C. Set the offset O_k in the middle of the selected interval, the corresponding possible release time is r_k
  D. Update the release array R to store the frames of f_k released in the interval $[0, T_{max}]$

$$\forall i \in \mathbb{N} \text{ and } r_k + i \cdot \frac{T_k}{g} \leq \frac{T_{\max}}{g}$$

$$\text{do } R\left[r_k + i \cdot \frac{T_k}{g}\right] = R\left[r_k + i \cdot \frac{T_k}{g}\right] \cup f_{k,i+1}$$

Example

$f_1 = (T_1 = 10, O_1 = 4)$, $f_2 = (20, 8)$, $f_3 = (20, 18)$ and a time granularity equal to 2. First the algorithm decides the offset for f_1: $l_1 = 0$ (step $1.(a)$), $B_1 = 1$ and $E_1 = 5$ (step $1.(b)$), thus $r_1 = 3$ (step $1.(c)$), which means that the offset of the stream is 4. Then array R is updated: $R[3] = \{f_{1,1}\}$ and $R[8] = \{f_{1,2}\}$ (step $1.(d)$). For stream f_2: $l_2 = 0$, the selected interval is $\{4, 5, 6, 7\}$ thus $B_2 = 4$, $E_2 = 7$ and $r_2 = 5$ with $R[5] = \{f_{2,1}\}$. For stream f_3, $l_3 = 0$, the selected interval is $\{9, 10, 1, 2\}$ thus $B_3 = 9$, $E_3 = 2$ and $r_3 = 10$ with $R[10] = \{f_{3,1}\}$.

Example

$f_1 = (T_1 = 10, O_1 = 4)$, $f_2 = (20, 8)$ and $f_3 = (20, 18)$ $(T_{\max} = 20)$ with a granularity $g = 2$.

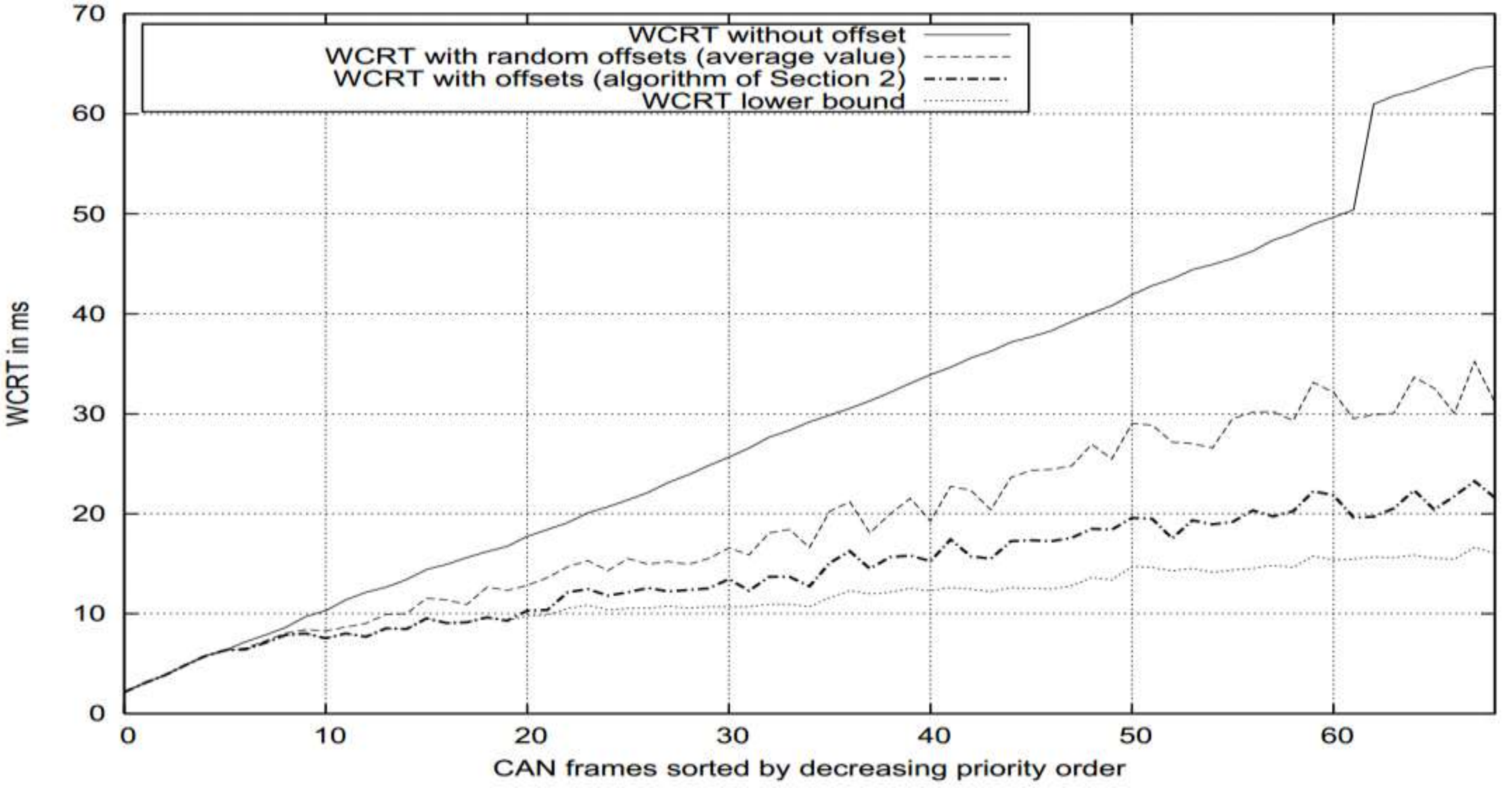
time	0	2	4	6	8	10	12	14	16	18
possible release time i	1	2	3	4	5	6	7	8	9	10
$R[i]$ (frames released)			$f_{1,1}$		$f_{2,1}$			$f_{1,2}$		$f_{3,1}$

Worst Case Response Times Results

Worst Case Response Times Results

- Offsets results in the reduction of the WCRT for low priority messages

Worst Case Response Times Results



Worst Case Response Times Results

- Offsets results in the reduction of the WCRT for low priority messages

Worst Case Response Times Results

- Offsets results in the reduction of the WCRT for low priority messages
- For the lowest priority frame of this example, the WCRT with offsets is decreased by 43.2 ms (from 64.8 to 21.6)

WCRT Reduction Ratio (No Concentration)

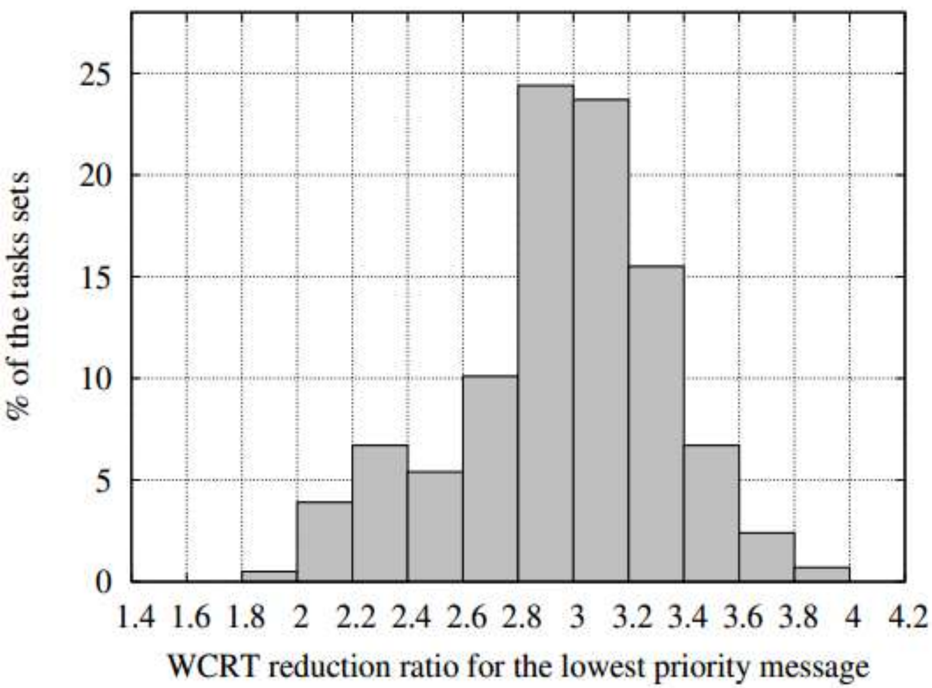
WCRT Reduction Ratio (No Concentration)

- The performance of Offset assignments over 1000 random sets of messages was evaluated

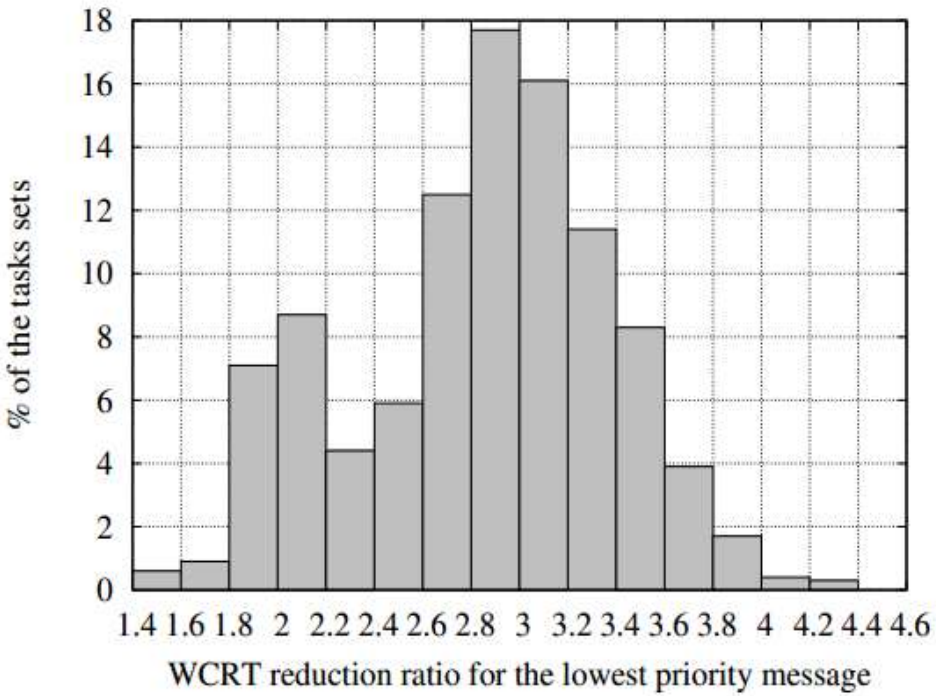
WCRT Reduction Ratio (No Concentration)

- The performance of Offset assignments over 1000 random sets of messages was evaluated
- The performance metric is the ratio of WCRT reduction when using offsets with the algorithm described

WCRT Reduction Ratio
(No Concentration)

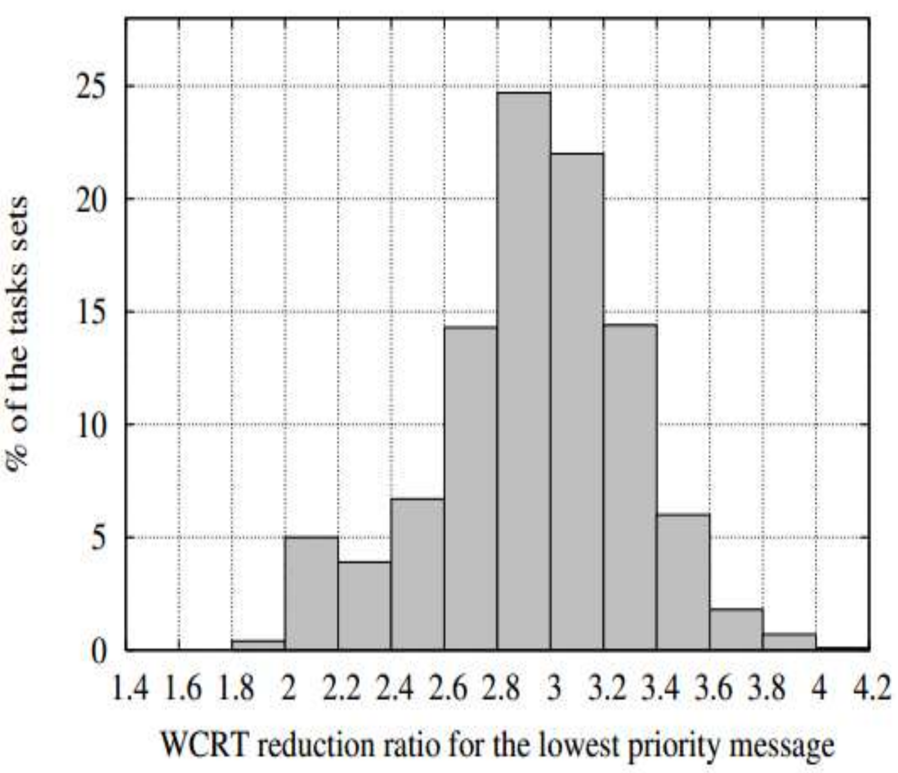


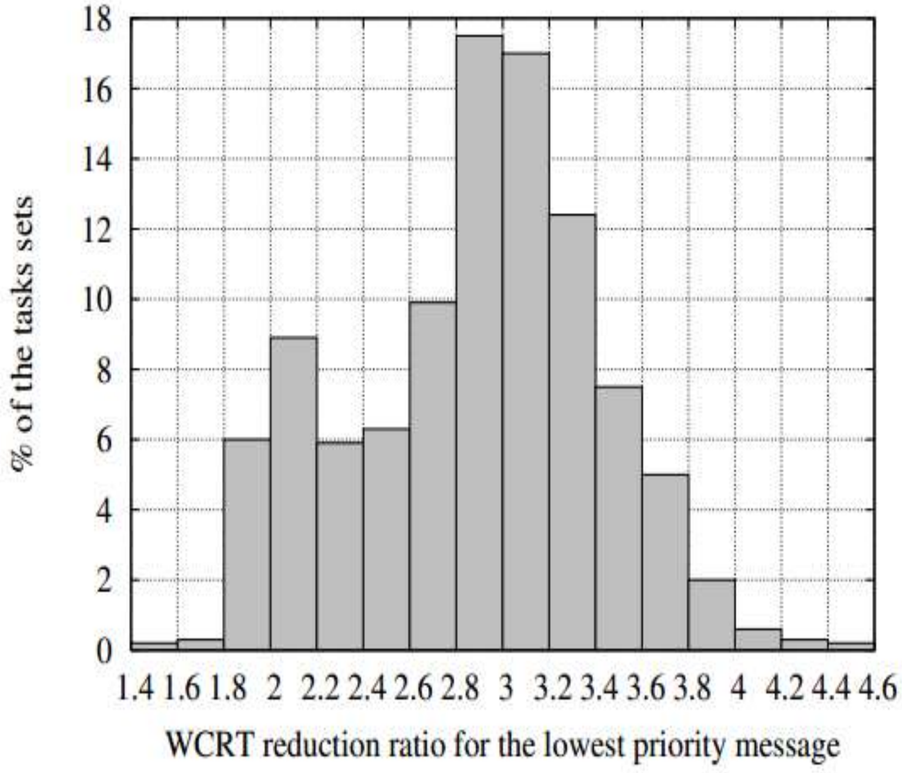
Body Network



Chassis Network

WCRT Reduction Ratio (30% Concentration)



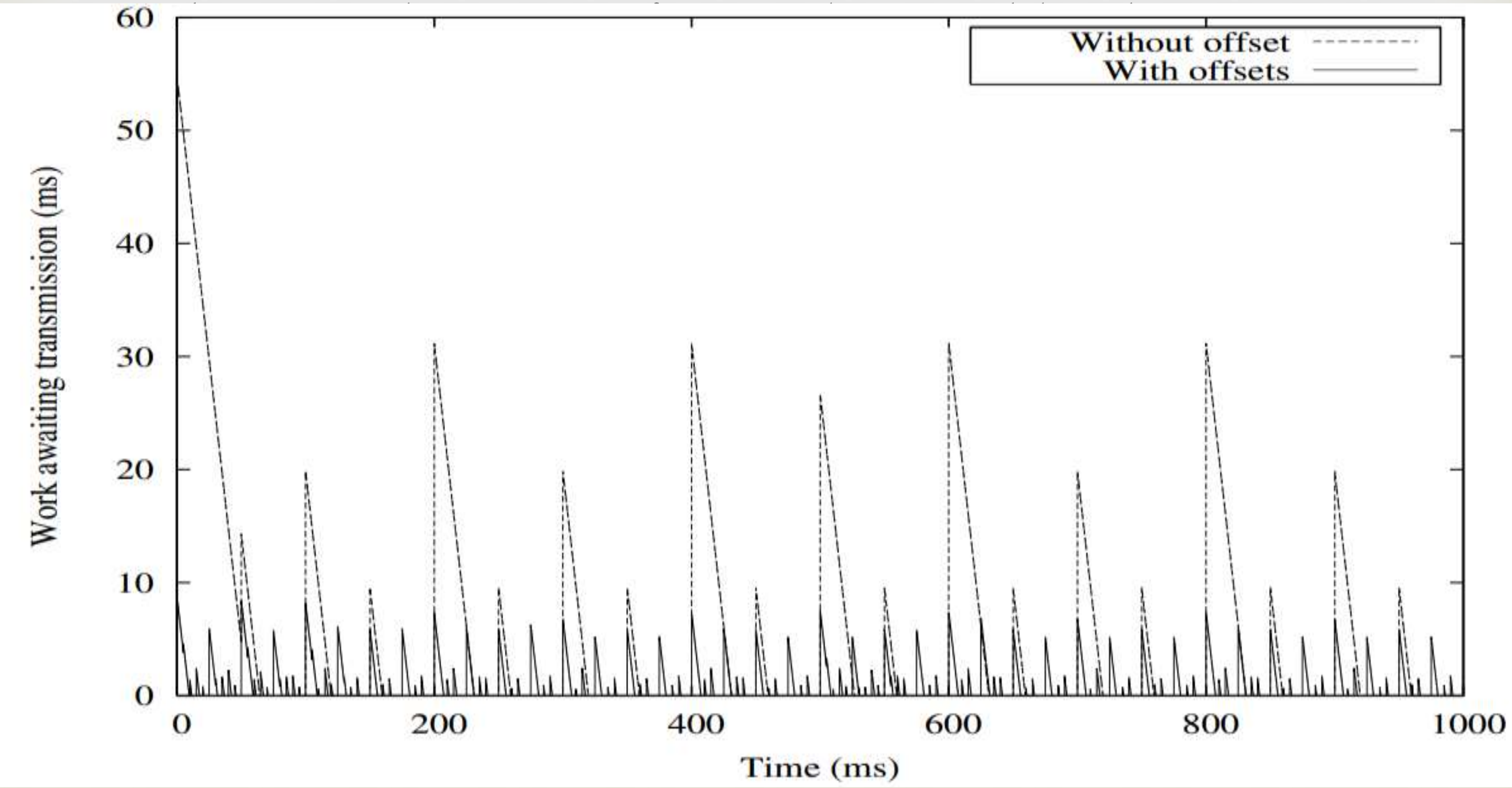


Network Load Distribution

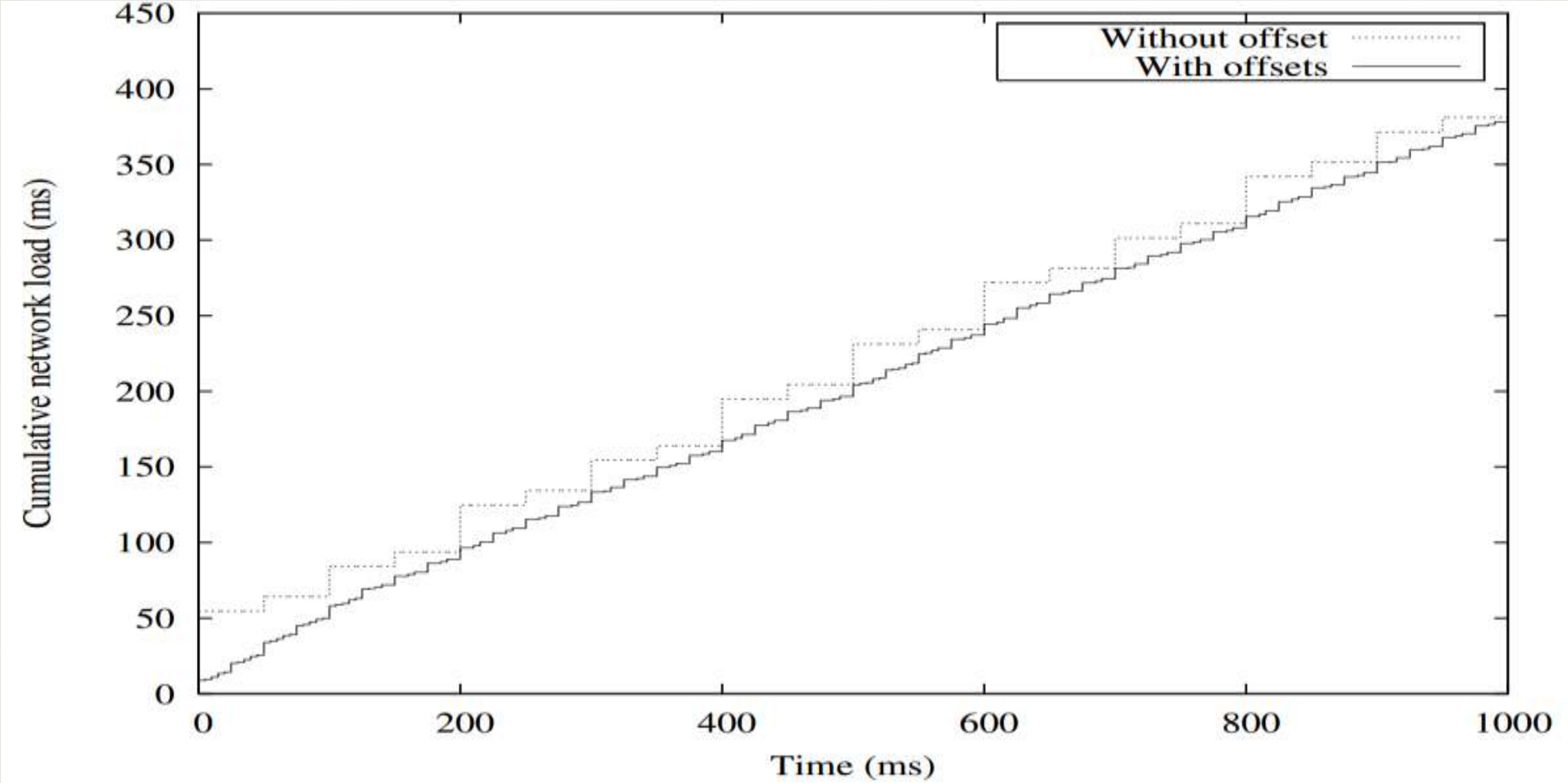
Network Load Distribution

- The evolution of total workload awaiting transmission is measured during one second with and without offsets

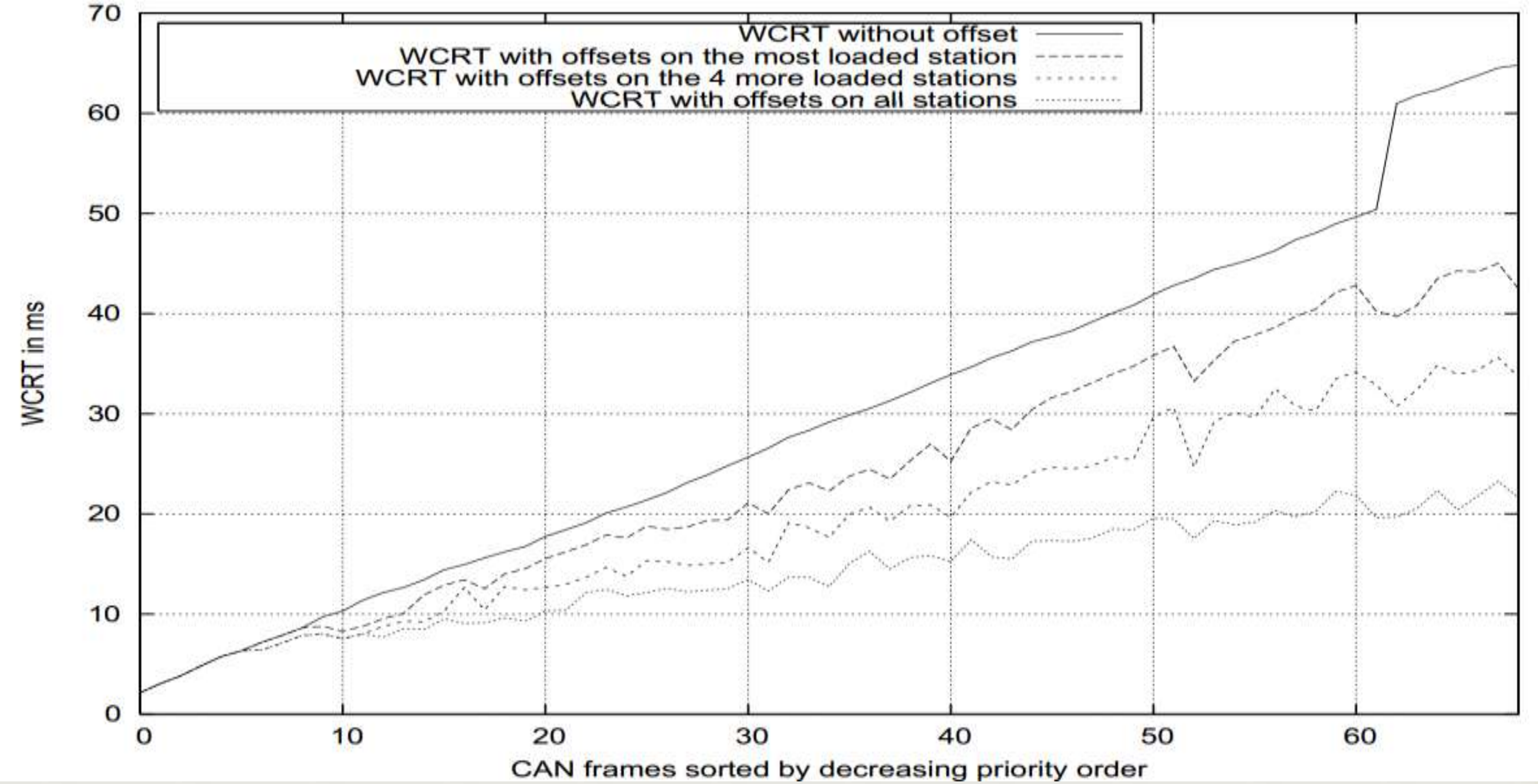
Network Load Distribution



Cumulative Network Load



Partial Offset Usage

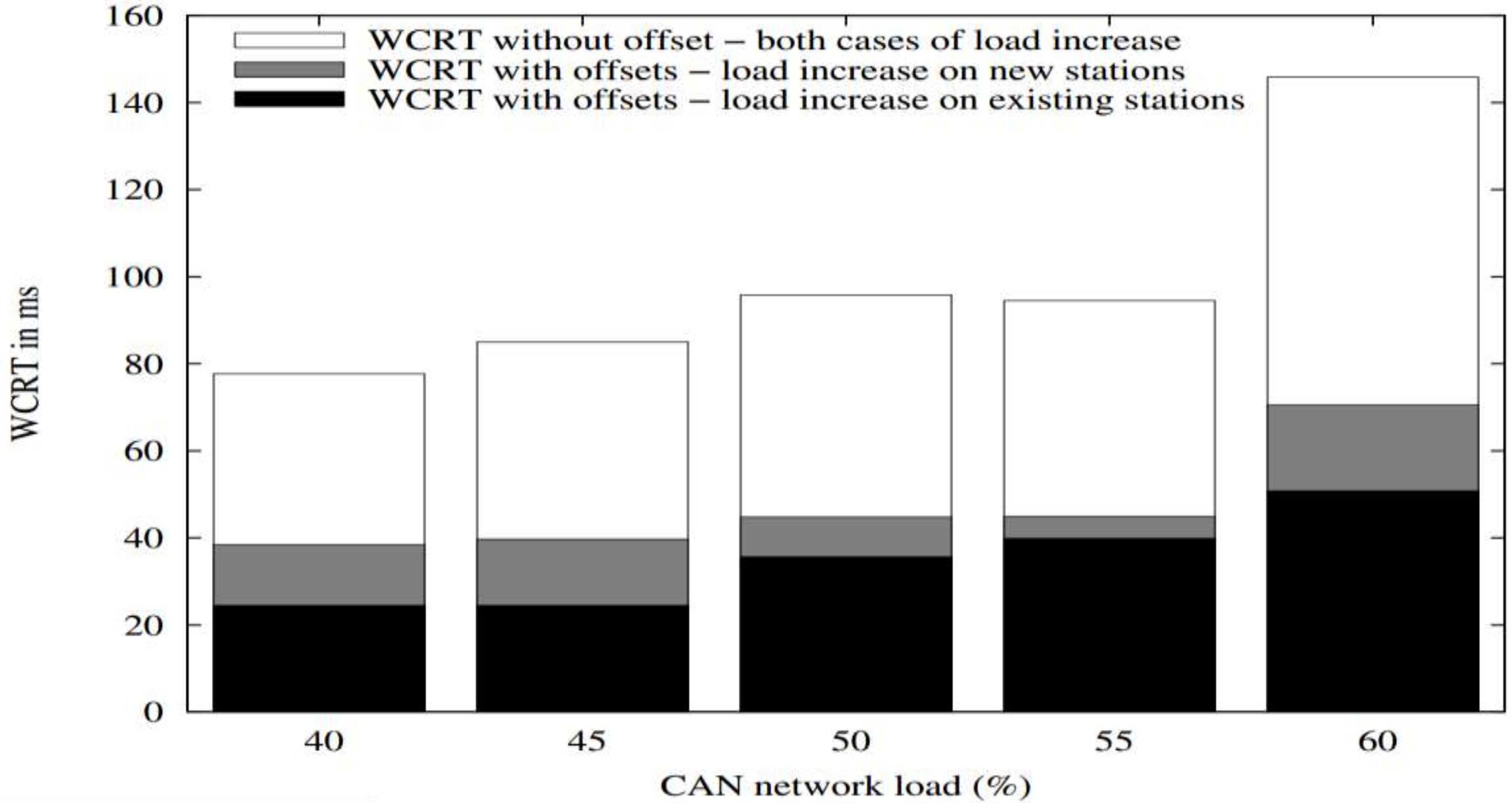


Offsets and Load

Offsets and Load

- Load can be increased either adding more stations or messages

Offsets and Load



Conclusion

- A low-complexity algorithm for deciding offsets, which has good performances for typical automotive networks
- Using offsets is a robust technique that might actually provide a solution in the short term to deal with the increasing network load
- Offsets, which impose constraints on the frame release dates, can be seen as a trade-off between event-triggered communications and time-triggered communications