TCP DCM+ (DYNAMIC CONGESTION MANAGEMENT PROTOCOL) FOR ENHANCING PERFORMANCE IN MOBILE AND WIRELESS NETWORKS

Jerusalem – Palestine

RUSHDI HAMAMREH / DERAR KHADER
1. INTRODUCTION
2. ORIGINS OF TCP DCM+
3. BANDWIDTH ESTIMATION
4. COMPONENTS OF TCP DCM+
5. DCM+ ALGORITHM
6. SIMULATION RESULTS
7. FUTURE WORK
1. INTRODUCTION

• CONGESTION CONTROL:

  CONTROLS THE E2E DATA RATE ON THE LINK THROUGH:

  1- PACKET LOSSES, OR
  2- ARRIVAL OF N DUPLICATE ACKNOWLEDGEMENT PACKETS
     (UNORDERED ARRIVAL OF PACKETS).

• FLOW CONTROL:

  ENSURES THAT A SENDER IS NOT OVERWHELMING THE RECEIVER.
  (IF IT SENDS PACKETS FASTER THAN THE RECEIVER BUFFER CAN HANDLE.)
1. **INTRODUCTION (CONT.)**

- **E2E (END-TO-END):**

1. **ONLY TCP SOURCE AND TCP DESTINATION MANAGE THE CONNECTION.** *(SETTING WINDOW SIZE OF THE TRANSMISSION).*

2. **INTERMEDIATE NODES (Routers) DON'T CARE ABOUT LOST PACKETS OR CONGESTION EVENTS.**

3. **TCP DCM+ STANDS FOR (DYNAMIC CONGESTION CONTROL PROTOCOL FOR MOBILE AND WIRELESS NETWORKS)**

4. **DCM+ IS AN E2E PROTOCOL, THAT IS DESIGNED TO CONTROL THE DATA RATE IN WIRED, WIRELESS AND MANETS.**
1. INTRODUCTION (CONT.)

End to end communication [1]

- TCP has no knowledge of the underlying Internet structure
INTRODUCTION

ORIGINS OF TCP DCM+

BANDWIDTH ESTIMATION

COMPONENTS OF TCP DCM+

TCP DCM+ ALGORITHM

SIMULATION RESULTS

FUTURE WORK
2. ORIGINS OF TCP DCM+

- TCP DCM+ [6] follows the paradigm (AIADD) like TCP WESTWOOD+
- AIADD: Additive Increase/Adaptive Decrease

- **Additive Increase**: Like TCP NewReno behavior in
  Slow-Start (SS) region.

- **Adaptive Decrease**: Unlike TCP NewReno:
  1. No drop to the half of congestion windows (CWND)
  2. Network congestion detected: New CWND depends on the available channel capacity, also known as Slow-Start Threshold (SSThresh).
  3. New SSThresh calculation using a Bandwidth Estimation (BWE) algorithm.
TCP NewReno behavior (AIMD)
Additive Increase / Multiplicative Decrease

Congestion Avoidance (CA):

- **Lost Packet**
  -> retransmission timeout, set cwnd to 1 segment and enters SS.

- **Unordered arrival** of data packets
  -> receiver sends 3 DUPACK packets to sender, set cwnd to new ssthresh and enters SS.

$$S_{thresh_{new}} = \frac{cwnd_{old}}{2}$$
2. ORIGINS OF TCP DCM+ (CONT.)

- At the discrete time ($T_n$):

  \( W_n \): Congestion Window (\textit{cwnd}),

  \( \hat{B}_n \): Estimated BW (\textit{BWE}),

  \( R_n \): round-trip-time (\textit{RTT}) at time \( n \)

  \textbf{New cwnd} is inverse proportional to the measured \textit{RTT}:

  \[
  W_n \propto \frac{1}{R_n}
  \]
2. ORIGINS OF TCP DCM+ (CONT.)

• TCP WELCOME [4]

(WIRELESS ENVIRONMENT, LINK LOSSES, CONGESTION MODELS)

PROPOSED (2009) FOR IMPROVING PERFORMANCE OF NETWORKS WITH DYNAMIC TOPOLOGY.

AIM: CONGESTION CONTROL IN MOBILE ADHOC NETWORKS (MANETS)

• IDEA: \( \frac{RTO_{OLD}}{RTO_{NEW}} = \frac{RTT_{OLD}}{RTT_{NEW}} \) (RTT ↑ ↔ RTO ↑)
2. ORIGINS OF TCP DCM+ (CONT.)

- **DCM**: DYNAMIC CONGESTION MODEL [5]

- **PROPOSED**: MAY 2015 (MSC. THESIS)  
  R. A. HAMAMREH AND M. BAWATNA.

- **AIM**: ADDRESSES THE WEAKNESS OF TCP WELCOME TO USE CONVENTIONAL TCP NEWRENO CONGESTION CONTROL TECHNIQUES IN MANETS.

- **IDEA**: USE THE RESULTS OF ROUTE REQUEST (RREQ) TO DETECT CONGESTION AND DYNAMICALLY SELECT A DIFFERENT PATH, IF AVAILABLE.

- **DCM REQUIRES THE COOPERATION OF L3 AND L4 IN THE OSI MODEL.**
2. ORIGINS OF TCP DCM+ (CONT.)

• DCM EQUATIONS (ALGORITHM):

1- \( \text{CONGESTION\_THRESHOLD} = 1.8 \times \text{RTT}_{\text{MIN}} \)

2- \( \text{RTT} > \text{CONGESTION\_THRESHOLD} \) -->
L3 NOTIFIES L4 TO SELECT ANOTHER PATH WITH LOWER RTT.

3- IF NEW PATH FOUND -> IT IS SET AS THE NEW MAJOR PATH.

4- **ESTIMATE** THE CHANNEL CAPACITY (BW) TO FIND NEW SSTHRESH ACCORDING TO TCP WESTWOOD. ALSO, SET NEW CWND.

5- **EQUATIONS**:

\[
BWE = \frac{\text{ACK\_SIZE}}{\text{ACK\_INTERVAL}}, \quad \text{SSTHRESH}_{\text{NEW}} = \max(2, BWE \times \frac{\text{RTT}_{\text{MIN}}}{{\text{SEGSIZE}}})
\]

\[
\text{RTO}_{\text{NEW}} = \text{RTO}_{\text{OLD}} \times \frac{\text{RTT}_{\text{NEW}}}{\text{RTT}_{\text{OLD}}}, \quad \text{CWND}_{\text{NEW}} = \beta \times \frac{\text{RTT}_{\text{OLD}}}{\text{RTT}_{\text{NEW}}} \times \text{SSTHRESH}_{\text{NEW}}, \quad \text{WHERE} \quad (\beta = 0.8)
\]
5. IF NO PATH IS FOUND WITH LOWER $RTT$:

$$\begin{align*}
\text{CWND}_{\text{NEW}} &= \begin{cases} \\
\text{CWND}_{\text{OLD}} + \frac{\text{RTT}_{\text{OLD}}}{\text{RTT}_{\text{NEW}}} \times 0.9 &; \text{CWND}_{\text{OLD}} < \text{SSTHRESH} \\
\text{CWND}_{\text{OLD}} + \frac{\text{RTT}_{\text{NEW}}}{\text{RTT}_{\text{OLD}}} \times \frac{1}{\text{CWND}_{\text{OLD}}} &; \text{CWND}_{\text{OLD}} \geq \text{SSTHRESH}
\end{cases}
\end{align*}$$
2. ORIGINS OF TCP DCM+ (CONT.) [6]

TCP DCM+ HAS ITS ROOTS IN DIFFERENT APPROACHES, MAINLY TCP WESTWOOD+, WHICH DELIVERS THE TECHNIQUE FOR BANDWIDTH ESTIMATION.
2. **ORIGINS OF TCP DCM+ (CONT.) – SACK OPTION**

- TCP DCM+ PERFORMS BEST WITH SACK OPTION ENABLED FOR LARGE FILES.
2. ORIGINS OF TCP DCM+ (CONT.) – SACK OPTION

- The advantages noticed for mid-sized files are the stability and the minimal number of CWND drops, but the transmission takes longer time.
- Small files are best sent with DCM+ without SACK option enabled.
CONTENT

• INTRODUCTION
• ORIGINS OF TCP DCM+
• BANDWIDTH ESTIMATION
• COMPONENTS OF TCP DCM+
• DCM+ ALGORITHM
• SIMULATION RESULTS
• FUTURE WORK
3. **BANDWIDTH ESTIMATION (BWE)** [19]

- **WESTWOOD:**
  - **BWE** samples calculated **after each ACK**
  - → **OVERESTIMATION** → **WRONG ESTIMATION.**

- **WESTWOOD+:** **BWE** samples calculated **after each RTT.**
  - → **MORE ACCURATE**
  - → **REFLECTS THE AVAILABLE CHANNEL CAPACITY.**
3. BANDWIDTH ESTIMATION (CONT.)

• BWE IN TCP WW+ [7][8]: USING A LOW-PASS FILTER (LP) 1\textsuperscript{ST} ORDER.

ANTI-ALIASING FILTER (AAF) IS USED AS LP FILTER $\rightarrow$ LIMITING SIGNAL BW.

• CALCULATION:

$$\text{SSTHRESH} = \max (2, \frac{\text{BWE} \times \text{RTT}_{\text{MIN}}}{\text{SEG\_SIZE}});$$

• 3 DUPACKS $\rightarrow$ CWND = SSTHRESH (SEGMENTS)

• RETRANSMISSION TIMEOUT (RTO) $\rightarrow$ CWND = 1 (SEGMENT)
3. BANDWIDTH ESTIMATION (CONT.)

• OUTPUT OF AAF ARE $B_i$ SAMPLES

$$B_i = \frac{d_i}{RTT_i}$$

• $B_i$: ANTI-ALIASING BW SAMPLE

• $D_i$: ACKNOWLEDGED DATA

• FILTER EQUATION:

$$\hat{b}_k = \alpha * \hat{b}_{k-1} + (1 - \alpha) * b_k ;$$

$\hat{b}_k$: ESTIMATED BW SAMPLE ; $\alpha$ = FILTER CONSTANT
CONTENT

• INTRODUCTION
• ORIGINS OF TCP DCM+
• BANDWIDTH ESTIMATION
• COMPONENTS OF TCP DCM+
• DCM+ ALGORITHM
• SIMULATION RESULTS
• FUTURE WORK
4. COMPONENTS OF TCP DCM+

1. \[ \text{RATECA} = \frac{\text{RTT}_{\text{OLD}}}{\text{RTT}_{\text{MIN}}} \]

\text{RATECA} : \hspace{1cm} \text{CONGESTION RATE}

\text{RTT}_{\text{MIN}} : \hspace{1cm} \text{MINIMUM TIME NEEDED FOR THE TCP TRANSMISSION WITH EMPTY TCP BUFFER AT THE DESTINATION.}

\[ \text{RTT} = \sum_{i=1}^{4} D_i \]

\[ \text{RTT}_{\text{MIN}} = \sum_{i=1}^{3} \min(D_i) \hspace{1cm} \text{;}(D_4 = D_Q = 0) \]

\(D_1\) : TRANSMISSION DELAY \((D_{TR})\)
\(D_2\) : PROPAGATION DELAY \((D_{PROP})\)
\(D_3\) : PROCESSING DELAY \((D_{PROC})\)
\(D_4\) : QUEUEING DELAY \((D_Q)\)
NOTE: \[ \begin{align*}
rateCA > 1 & \implies \text{TCP buffer decreasing} \\
rateCA < 1 & \implies \text{TCP buffer increasing}
\end{align*} \]

RATECA > 1: **NO CONGESTION** EXPECTED $\implies$ **BURST TRANSMISSION**

$\implies$ **ADVANCING** ( $\implies$ **LARGE BURSTS** )

RATECA < 1: **GRADUAL INCREASE** OF RTT $\implies$ **CONGESTION** EXPECTED

$\implies$ **DANGER** ( $\implies$ **MINI BURSTS** / **CONSTANT TRANSMISSION**)
4. COMPONENTS OF TCP DCM+ (CONT.) – BURST TRANSMISSION
4. COMPONENTS OF TCP DCM+ (CONT.) – DYNAMIC BEHAVIOR OF DCM+ (CWND IS TRACKING SSTHRESH)
4. COMPONENTS OF TCP DCM+ (CONT.)

2- \[ RTO_{\text{NEW}} = RTO_{\text{old}} \times \frac{RTT_{\text{new}}}{RTT_{\text{old}}} = \frac{RTO_{\text{old}}}{\text{rateCA}} \]

- RATECA ↑ → RTO ↓

**CLEAR:** INCREASING OF RATECA MEANS CONGESTIONS ARE NOT EXPECTED,

→ NO NEED TO WAIT LONGER FOR THE PACKETS.

- RATECA ↓ → RTO ↑ (BECAUSE OF DANGER OF CONGESTION → WAIT LONGER)
3- **NEW MEMBER FUNCTION IN C++ FOR THE CONGESTION AVOIDANCE**

- **VOID CONGESTIONAVOIDANCE (TCP_SOCKET, ACKEDSEGMENTS)**
- **WW AND WW+ DOES NOT HAVE THIS MEMBER FUNCTION IN NS3.**
- **THIS FUNCTION OVERRIDES THE BEHAVIOR OF THE SAME FUNCTION EXISTING IN TCP NEWRENO CLASS.**
- **TCP DCM+ BEHAVIOR WITHIN THE MEMBER FUNCTION:**

```cpp
if (number of segments acked > 0)
{
    if (last cwnd < available channel capacity)
        cwnd += 2 * rateca;  // ADVANCING
    else
        cwnd += 2/(rateca*cwnd);  // DANGER
}
```
INTRODUCTION

ORIGINS OF TCP DCM+

BANDWIDTH ESTIMATION

COMPONENTS OF TCP DCM+

TCP DCM+ ALGORITHM

SIMULATION RESULTS

FUTURE WORK
5. TCP DCM+ ALGORITHM

• DCM+ EQUATIONS:

1- \[ \text{RATECA} = \frac{\text{RTT}_{\text{OLD}}}{\text{RTT}_{\text{MIN}}} \]

2- \[ \frac{\text{RTO}_{\text{NEW}}}{\text{RTO}_{\text{OLD}}} = \frac{\text{RTT}_{\text{NEW}}}{\text{RTT}_{\text{OLD}}} \]

3- \[ \text{CWND}_{\text{NEW}} = \begin{cases} \text{CWND}_{\text{OLD}} + 2 \times \text{RATE}_{\text{CA}} ; & \text{CWND < SSTHRESH} \\ \text{CWND}_{\text{OLD}} + \left( \frac{2}{\text{RATE}_{\text{CA}} \times \text{CWND}_{\text{OLD}}} \right) ; & \text{CWND \geq SSTHRESH} \end{cases} \]
5. TCP DCM+ ALGORITHM (CONT.)

Measuring BURST ratio:

\[ \text{rateCA}_{\text{old}} = \frac{\text{RTT}_{\text{old}}}{\text{RTT}_{\text{min}}} \]

\[ \text{rateCA}_{\text{new}} = \frac{\text{RTT}_{\text{new}}}{\text{RTT}_{\text{min}}} \]

Assumption: \( \text{RTT}_{\text{min}} \) remain unchanged.

Space ratio on the channel:

\[ S = \frac{\text{rateCA}_{\text{old}}}{\text{rateCA}_{\text{new}}} = \frac{\text{RTT}_{\text{old}}}{\text{RTT}_{\text{min}}} * \frac{\text{RTT}_{\text{min}}}{\text{RTT}_{\text{new}}} \]

\[ S = \frac{\text{RTT}_{\text{old}}}{\text{RTT}_{\text{new}}} \approx \frac{D_q,\text{old}}{D_q,\text{new}} \propto \frac{\text{old buffer capacity}}{\text{new buffer capacity}} \]
5. TCP DCM+ ALGORITHM [6]

We see the different phases in this algorithm: NewReno, Westwood+ and the time parameters RTT and RTO.
5. **TCP DCM+ ALGORITHM (CONT.)**

All the following phases can appear depending on the following parameters:

- **Error rate**, **data size**, **MTU size** and **TCP buffer size**.

- **Phases of DCM+**
  1. **Initialization** (probing) phase (IP)
  2. **Advancing** phase (AP)
  3. **Near-channel-capacity** phase (NCCP)
  4. **Losses** phase (LP) (**optional**)
  5. **End** phase (EP) (**optional**)

![TCP DCM+ - Window Dynamics](image-url)
CONTENT

• INTRODUCTION
• ORIGINS OF TCP DCM+
• BANDWIDTH ESTIMATION
• COMPONENTS OF TCP DCM+
• DCM+ ALGORITHM
• SIMULATION RESULTS
• FUTURE WORK
6. SIMULATION RESULTS –

TCP DCM+ can also detect losses:
1. spikes in RTT plot OR
2. resetting of RTO values.

![Graph showing simulation results](image)
6. SIMULATION RESULTS - THROUGHPUT

• THROUGHPUT: DCM+ SHOWS THE HIGHEST THROUGHPUT IN THE RANGE (1E-3 TO 5E-2).
6. SIMULATION RESULTS – (NAI) ROBUSTNESS

NAI: NORMALIZED ADVANCING INDEX

NAI SHOWS THE ROBUSTNESS OF THE TRANSMISSION

DCM+ HAS THE HIGHEST ROBUSTNESS IN THE RANGE (1E-5 TO 4E-2)
6. SIMULATION RESULTS – CTT (TRANSMISSION TIME)

CTT: COMPLETE TRANSMISSION TIME

DCM+ SHOWS THE SHORTEST TIME NEEDED FOR THE TCP TRANSMISSION
6. SIMULATION RESULTS – PACKET LOSSES

DCM+ HAS EQUAL OR LESS LOSSES THAN OTHER APPROACHES. TCP BIC SHOWS HIGHEST LOSSES IN THE RANGE (1E-5 TO 1E-3)
6. SIMULATION RESULTS – AVG. E2E DELAY

DCM+ shows the lowest delay, while BIC shows the highest delay.
6. SIMULATION RESULTS – THE SIZE OF RECEIVER BUFFER AFFECTS THE TRANSMISSION TIME, ROBUSTNESS AND THE DROPS OF WINDOW SIZE.
6. SIMULATION RESULTS –
TCP DCM+ TRANSMISSION ROBUSTNESS AS A FUNCTION OF RECEIVER BUFFER

<table>
<thead>
<tr>
<th>Buffer Size (KB)</th>
<th>1/sec</th>
<th>Robustness (NAI)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>0</td>
<td>13.77</td>
</tr>
<tr>
<td>64</td>
<td>50</td>
<td>26.33</td>
</tr>
<tr>
<td>256</td>
<td>150</td>
<td>44.97</td>
</tr>
<tr>
<td>512</td>
<td>200</td>
<td>66.44</td>
</tr>
<tr>
<td>1024</td>
<td>250</td>
<td>97.13</td>
</tr>
<tr>
<td>2048</td>
<td>300</td>
<td>121.54</td>
</tr>
<tr>
<td>4096</td>
<td>350</td>
<td>128.83</td>
</tr>
<tr>
<td>8192</td>
<td>400</td>
<td>122.02</td>
</tr>
<tr>
<td>16384</td>
<td></td>
<td>196.91</td>
</tr>
</tbody>
</table>

Robustness (NAI)
6. SIMULATION RESULTS –

TCP DCM+ WITH TCP BUFFER SIZE = 1MB REACHES ENOUGH LOW CTT.
6. SIMULATION RESULTS –

DCM+ THROUGHPUT AS A FUNCTION OF RECEIVER BUFFER SIZE

![Graph showing throughput as a function of TCP buffer size.](image-url)
6. SIMULATION RESULTS –

DCM+ LOSSES AS A FUNCTION OF RECEIVER BUFFER SIZE

Packet Losses

TCP Buffer Size (KB)
6. SIMULATION RESULTS –

SEGMENT SIZE OPTIMIZATION BEFORE THE TRANSMISSION CAN IMPROVE THE PERFORMANCE OF TCP DCM+
6. SIMULATION RESULTS – MAXIMUM THROUGHPUT OF TCP DCM+ IS GUARANTEED FOR DESTINATION BW ≥ BOTTLENECK BW. (HERE: DESTINATION BW = 1 GBPS).
6. **SIMULATION RESULTS –**

MAXIMUM THROUGHPUT OF TCP DCM+ IS GUARANTEED FOR DESTINATION BW ≥ BOTTLENECK BW. (HERE: BOTTLENECK BW = 100 MBPS).

![Graph showing the relationship between access bandwidth of destination (Mbps) and throughput.](image-url)
• INTRODUCTION
• ORIGINS OF TCP DCM+
• BANDWIDTH ESTIMATION
• COMPONENTS OF TCP DCM+
• DCM+ ALGORITHM
• SIMULATION RESULTS
• FUTURE WORK
REFERENCES


7. TCP WESTWOOD+ CONGESTION CONTROL (HTTPS://C3LAB.POLIBA.IT/INDEX.PHP/WESTWOOD)