CHAPTER 4
AN OPTIMIZED FUZZY APPROACH FOR
REPRESENTATIVE NODE SELECTION FOR CROSS-
LAYER TCP (CLTCP)

4.1 INTRODUCTION

The Transmission Control Protocol (TCP) is a core Internet Protocol Suite protocol used by many popular Internet applications like File Transfer, e-mail, Web, Secure-Shell, and streaming media applications. High Bit Error Rate (BER) as well as path re-calculation effects on TCP performance and reasons behind the effects are described.

- High BER Effect: Bit errors corrupt packets leading to TCP acknowledgments and data segments being lost. A TCP sender in a given time, retransmits a segment when acknowledgements fail to reach it, backing off exponentially its retransmission timer for subsequent retransmissions and decreasing congestion windows. Repeating errors lead to a small congestion window with the source causing low throughputs (Ahmed et al. 2010).

- Impact of Path Re-calculations: When older paths are not accessible, routing protocols at source try to locate new destination routes. Locating new routes takes time. So, TCP source times out, retransmits packets invoking congestion
control techniques which is undesirable as TCP connection becomes ineffective. If we think of a network where path calculations are frequent, the TCP link will never have a chance to forward at a maximal negotiated rate.

- **Power Scarcity**: As batteries carried in mobile nodes have limited power their life is limited. As a node acts as router and end system, unnecessary TCP segments retransmissions consumes power causing inefficient use of power (EI-Sayed et al. 2005).

- **Multipath routing**: Routing protocols have multiple source-destination routes to reduce frequent route re-computation. But, this results in the arrival of out-of-sequence packets at a receiver generating duplicate ACKs causing the sender to invoke congestion control.

- **Interaction between TCP and MAC protocol**: Interaction between TCP protocol mechanisms and 801.11 MAC protocol result in unexpected serious problems in multi-hop environments. Problems include “link capture effect” instability and one-hop unfairness.

#### 4.1.1 TCP Performance in MANET

Improve TCP performance in MANET, to classify the four groups according to the four issues which are (Hamrioui et al. 2013):

1) TCP fails to distinguish between packet loss caused by route failure, network congestion.

2) Common roads failure.
3) Wireless channel contentions.

4) TCP’s non fairness.

Problems (1) and (2) are leading causes for TCP performance degradation in MANETs.

Two layer based solutions report improved performance than those that have their basis in one layer. But the latter respect the isolation concept to design long term protocols. To choose between solutions one has to understand our priority; whether it is performance optimization or architecture. Performance optimization leads to time savings (short-term). Infrastructure has its basis in long-term considerations. Also, solutions between layers are harder to execute and formulate than those that have their basis in a layer. Solution execution between layers needs two OSI layer alters and their design needs the model be regarded in its complete nature.

Many approaches were used to improve MANETs TCP performance. Most approaches distinguished between network states causing packet loss and initiating action is such cases. Approaches need feedback from a network to transport layer, or else they are end-to-end approaches where a transport layer does the job by itself.

High BER causes packet losses in wireless channels and between two cells handoffs, and in MANETs such losses are due to route breakages and medium contention in addition to radio channel errors. So, though TCP performance in wired networks is good, its performance degrades in wireless networks when non congestion related losses are misinterpreted as congestion thereby wrongly invoking congestion control and avoidance processes. This was proved by analyses as well as simulation (Chen et al. 2005).
Wireless networks are split into one hop wireless networks which include cellular networks and WLANs. The other is a multihop network which includes MANETs. This is due to quickly deteriorating TCP performance in ad hoc networks when compared to cellular networks and WLANs. For understanding TCP behaviour and enhancing its performance across wireless networks in such specific wireless issues, many studies were performed and several models implemented.

Many adhoc network devices share a common resource (medium) and so there is competition for link bandwidths, leading to network overloads. To avoid congestion/network overload a sender adjusts data sending rate and size of the window (Sreenivasa et al. 2012).

Research is being undertaken in congestion control, packet routing, standard TCP protocol modification and designing new routing protocols in MANETs.

Transport layer is responsible for congestion control in the OSI reference model. A combination of and dependability characteristics in TCP, permits congestion control management with no data regarding its status in a network. Techniques have to be adopted for avoiding MANETs congestion collapse as this modifies TCP congestion method. Altered TCP ought to ensure error as well as flow control. Flow control ensures that sources do not flood receivers by forwarding information at a faster rate than destinations process. It ought to ensure dependable end to end data transmissions over MANET. Altered TCP ought to provide full-duplex, dependable as well as test services to application.
4.1.2 Optimized TCP

Basic TCP/IP protocol suite developed today’s Internet. Specifically, TCP was successful due to robust reaction to changing network traffic and ensuring end to end based reliability. This motivated TCP application development, thereby extending the protocol to wireless networks. But this was a challenge to TCP as it was not meant to work in complex environments, where BER level due to physical medium is non negligible. Also high mobility degrades end-to-end performance because TCP lowers its transmission rate on seeing a dropped packet (Khelage & Kolekar 2014).

Many approaches optimized TCP’s wireless network performance which are split into two categories: The first researches, modifies TCP to improve its compatibility to wireless networks, and the second improves TCP performance without modifying it to a TCP state machine though modifying the algorithms in other layers (Mahmoodi et al. 2007).

A TCP performance improvement approach uses some form of Automatic Repeat reQuest (ARQ) mechanism to prevent misinterpreting packet loss by TCP source as due to congestion. TCP over ARQ was studied in the recent pasts. ARQ potentially increases, or causes fluctuations in TCP’s Round Trip Time (RTT). This interferes with TCP timeout. TCP retransmission timer may expire when a lost packet is retransmitted over a wireless link. Such shortcomings improved through adaptation of maximum local link retransmissions which depend on TCP state machine. So, attempts to optimize link-layer retransmissions depend on a TCP perceived end-to-end packet loss rate.

TCP Congestion (TCPC) optimization uses MAC layer feedback to differentiate packet loss due to link errors and network congestion. MAC- TCP layer feedback generates when function Retransmit DATA () or function RetransmitRTS () retry count exceeds thresholds (Chang & Gaydadjiev 2012).
Standard TCP is used here. Function slowdown() in TCP layer does nothing to maintain size of congestion window on receiving MAC layer feedback. Optimization is registered/activated when simulations start. It is suppressed by Global Load Balancing optimization. Optimization using one-hop neighbour’s link information. The aim is to avoid collisions or nodes with lower SNR. Received signal strength and collision degree represent local view. These parameters are translated into network density and interfere with neighbouring nodes.

4.2 Optimized Cross-Layer Design (CLD) for TCP

Achieving computer network communication was due to adopting a layered architecture like the OSI reference model. Its design as well as execution is split into modules designed in a separate manner, implemented and interconnected using a cross-layer concept with five layers called a TCP/IP suit (Kamble & Kharat 2014).

If a network with single-rate devices/fixed channel is considered, the calculate network resource allocation as a utility maximization issue regarding network layer rate limits and link layer schedulability limits. Then the system problem using duality theory is decomposed vertically into congestion control, routing as well as scheduling sub issues which communicate via congestion rates.

A communication system has varied layers like physical, medium access, network, transport and application layers. Every layer supports differing configurations and operation modes that determine that layer’s communication performance. Combining performances of every single layer in a protocol stack determines an application’s overall performance; ie; to improve all layers should be tuned to end-to-end performance configuration parameters.
It is intuitive that separate optimization of a system’s component yields sub-optimal performance regarding to what could, be achieved in theory through all components being jointly optimized. This is called cross-layer Optimization in communication systems: breaking conventional communication protocol stack encapsulation to enable more effective optimization of total communication quality. Cross-layer Optimization due to superior performance is expected from Cognitive Radio Networks (Baldo & Zorzi 2008).

CLD, an adaptive protocol design meets fast growing wireless networking demands. Issue fixed within layers locally and optimization lead to inadequate results. CLD increases performance through exploitation of dependencies and inter layer interaction. Ensuring knowledge of physical/MAC channel conditions to routing, transport as well as application layers was a promising paradigm for wireless systems performance optimizations (Thaseen et al. 2012).

Wireless communication handles real time traffic like voice traffic, audio, video, multimedia, data traffic like web browsing, messaging, gaming and video conferences, and file traffic. Such diverse applications have differing QoS requirements guarantees for different traffic types. So a cross layer protocol interaction increases network efficiency providing better QoS support. Network throughputs are extremely optimized because of information exchange between layers.

Some innovating techniques were developed to improve network performances. When a layered model protocol is designed in an independent fashion, at the conclusion of executing a low level algorithm, destination node data consumption does not influence high level protocols behaviour. Cross-layer concept in an opposite operating mode, adapts protocols to wireless contexts through sharing inter layer information and via total optimization
instead of many optimizations at varied levels. Many significant experiments were performed already (Tiado et al. 2005).

Cross–layer method is utilized by algorithms at many levels as such interactions implementation enhances the performance of global systems. Particular algorithms are those improving TCP throughput. Different models implementing cross–layer interactions include MobileMan or CLASS.

There is need for information to flow among different protocol stack layers to achieve the desired optimization goal. This is called an CLD approach which relies on a network stack’s inter layer interactions; see Table 4.1. Cross layering ensures great performance benefits though it was proved that layered design was a key element for success of the Internet. Layers can share locally available information to improve performance.

(Source: Debbarma et al. 2012)

<table>
<thead>
<tr>
<th>Application layer</th>
<th>Energy management</th>
<th>Group communication &amp; service location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transport layer</td>
<td>Quality of service</td>
<td>Transport layer protocols</td>
</tr>
<tr>
<td>Network layer</td>
<td>Security and cooperation</td>
<td>TCP/IP routing, Addressing, Forwarding</td>
</tr>
<tr>
<td>MAC layer</td>
<td>Mobility management</td>
<td>Framing, Error Detection &amp; Control, Congestion</td>
</tr>
<tr>
<td>Link layer</td>
<td></td>
<td>Antennas, MAC, Bluetooth, Power control, 802.11, Hyper LAN</td>
</tr>
</tbody>
</table>

(Source: Debbarma et al. 2012)

**Figure 4.1 MANET Functions Sharing Between Different Layers through Cross-Layer Design**

Different characteristics of current CLD architecture are listed as stated below:
CLD involves the combinations of layers physical-MAC-network, MAC-network, network-transport only.

It provides an individual solution for power conservation, energy minimization, flow and congestion controls and fault tolerance.

Only local link information from MAC layer is used by congestion avoidance algorithm.

There is high/expensive overhead.

CLD exploit communication patterns with specific behaviour that differs from conventional communication. Another common CLD area is real-time multimedia transmissions demanding efficient adaptation of transmission rate according to link ability and vice versa. Joint optimization of application, physical and data link layers (Khan 2012).

4.3 FUZZY LOGIC FOR TCP

A classical logic set with degree of membership is fuzzy logic designed to represent fuzziness and vagueness mathematically ensuring a basic concept to handle imprecision in measurement and subjective evaluation problems. It is based not on probability but on possibility. Fuzzy logic is not the same as conventional logic systems as it aims to model imprecise modes of reasoning having important roles in human ability to take rational decisions during uncertainty/imprecision. This is dependent on the ability to infer approximate answers to questions based on stored knowledge incomplete or not available totally (Fatima et al. 2012).

Fuzzy logic constitutes fuzzy sets; methods representing approximate reasoning and non statistical uncertainty including operations for making inferences. Uncertainties are fuzzy sets (A_i), expressed in words but
interpreted through membership functions $\mu_A$. A fuzzifier is used at system input converting crisp to fuzzy data. A defuzzifier does this in reverse.

A hierarchical, cascaded Fuzzy Logic Controllers (FLC) structure ensured that fuzzy inference system is managed with simple fuzzy rules. Also multiple metrics were fused into one decision. The receiving node’s first FLC estimates channel quality with 2 metrics (SNR, LOS/ NLOS indication) and a decision is sent through feedback with acknowledgment to the next FLC (transmitting node) to select an appropriate FEC scheme for next transmission. This generic approach tunes radio transmission parameters like back-off, transmission power and data-rate. Mamdani-type inference and centroid as defuzzification methods are used by a full fuzzy system (Singh & Pesch 2011).

4.4 METHODOLOGY

Metaheuristic algorithms are often nature-inspired, and they are now among the most widely used algorithms for optimization. They have many advantages over conventional algorithms. Metaheuristic algorithms are very diverse, including Genetic Algorithms (GA), Differential Evolution (DE), Ant Colony Optimization (ACO), Bee Swarm Algorithms, Bat Algorithm, Particle Swarm Optimization (PSO), Harmony Search (HS), Firefly Algorithm (FA), Cuckoo Search (CS) and others.

Studies show that CS satisfy the global convergence requirements and thus has guaranteed global convergence properties. Also, CS has two search capabilities: local search and global search, controlled by a switching/discovery probability (Yang & Deb 2014). A further advantage of CS is that its global search using L´evy flights or process instead of random
walks. The Lévy flights help explore the search space efficiently. Due to these advantages, the CS is chosen in this work.

This work proposed to improve an optimization for Fuzzy Rule Selection using Cuckoo Search (CS) algorithm by proposing QoS parameter estimation and selection of representative node using fuzzy logic. A detailed discussion of each technique is conferred.

### 4.4.1 Fuzzy Logic Control

Fuzzy logic control system incorporates fuzzy inference engine, fuzzy rules, fuzzifier and de-fuzzifier. Mamdani Method, a common fuzzy inference technique (Negnevitsky 2005) is resorted to and includes fuzzification, rule evaluation, aggregation and de-fuzzification. Defuzzification locates a point wherein a vertical line slices aggregate set chance in two equal masses. Centre of Gravity (COG) is computed over an aggregate output membership function’s points sample with the formula:

\[
COG = \left( \frac{\sum_{i} \mu_{A_i}(x) \cdot x}{\sum_{i} \mu_{A_i}(x)} \right)
\]  

(4.1)

where, \( \mu_{A_i}(x) \) is a set A’s membership function.

Fuzzy logic inference flow from input variables to output variables is identified by system structure. Fuzzifications of input interfaces translate analog inputs to fuzzy values. Fuzzy inference acts with linguistic control rules in rule blocks. Linguistic variables are rule block outputs. Output interfaces defuzzification is translated to analog variables.
4.4.2 Proposed Optimization for Fuzzy Rule Selection using Cuckoo Search (CS)

Cuckoo Search (CS) Fuzzy CLTCP uses fuzzy rule selection for fuzzy rules first localization in input space. Rules in a complete rules set are (Alcala et al. 2007): Bad Rules (conflicting rules/erroneous) degrade performance of the system (rules not in final solution); Irrelevant/redundant rules fail to improve system performance; Complementary Rules complement others by improving lightly system performances; Important Rules not removed ensure reasonable system performance.

Determining rule types in advance is impossible as they are concrete rules configuration dependent and on CS parameters optimal configuration for each rule configuration. So, it is impossible to establish criteria for use in search.

Considering existence of such rules estimates tuning parameters and different rule configurations, following zones in objective space:

- Zone with Bad Rules, have solutions with bad rules. Here Pareto front does not exist is removing rules improves accuracy while solutions are dominated by others.
- Zone with Redundant/Irrelevant Rule, has solutions sans bad rules but which still have redundant/irrelevant rules. Accuracy is the same even when rules are deleted.
- Complementary Rules: includes solutions without bad/redundant rules. When these rules are removed accuracy slightly decreases.
• Zone with Important Rules: has a solution of essential rules. Removing such rules affects accuracy.

Yang and Deb (Yang & Deb 2009) proposed CS a meta-heuristic search algorithm inspired by cuckoo’s reproduction strategy. Cuckoos lay eggs in other birds’ nests of even different species. A host bird on discovering the eggs are not it’s own either destroys the egg or abandons the nest. This lead to evolution of cuckoo eggs which mimic local host birds eggs (Yang & Deb 2010 and Walton et al. 2011). The three idealized rules include:

• A cuckoo lays one egg, representing a solution coordinates set, and dumps it in a random nest;
• A fraction of nests with best eggs/solutions are carried over to next generation;
• Number of nests is fixed with a probability that a host will discover an alien egg. When this happens, the host either discards the eggs or the nest, and builds a new nest elsewhere.

4.5 RESULTS AND DISCUSSION

This work proposes CS Fuzzy CLTCP to increase throughput with reduced delay. Figure 4.2 and 4.3 shows the resulting graph for throughput and end to end delay. Throughputs refer to the rate of successful packet deliveries over transmission channels. The information can be transmitted over physical or logical links or pass through specific network nodes.

Throughputs are assessed in bits per second (bit/s or bps), as well as in data packets per second or data packets per time slot. End-to-end delays refer to time taken for packets to be forwarded from sources to destinations across a network.
Figure 4.2 Throughput in bits/sec

It is observed from Figure 4.2 that the throughput is improved for proposed technique. By taking average values proposed CS Fuzzy CLTCP achieves better accuracy by 12.92% than TCP, by 4.68% than CLTCP and by 2.41% than Fuzzy CLTCP.

Figure 4.3 End to End Delay in Seconds
It is observed from Figure 4.3 that the end to end delay is lowered for proposed techniques. By taking average values proposed CS Fuzzy CLTCP lowers end to end delay by 42.58% than TCP, by 33.71% than CLTCP and by 1.29% than Fuzzy CLTCP.

4.6 CONCLUSION

An optimized fuzzy rule selection uses Cuckoo Search (CS) to select representative node dynamically. Performance metrics like throughput and end to end delay are evaluated. The simulation results demonstrate the effectiveness of the proposed CS Fuzzy CLTCP. It is observed that on average values, the new CS Fuzzy CLTCP achieves 12.92% accuracy than TCP, 4.68% than CLTCP and 2.41% than Fuzzy CLTCP. By taking average values, the new CS Fuzzy CLTCP lowers end to end delay by 42.58% than TCP, 33.71% than CLTCP and 1.29% than Fuzzy CLTCP.