International Journal of Engineering & Technology, 7 (3.27) (2018) 248-251

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

SOBM - Server Occupancy Based Migration by Optimizing Link and Storage Capacity in Check Points

Sreeleja N. Unnithan[1*], S. Bhavani[2]

[1]Research Scholar, Karpagam Academy of Higher Education, Coimbatore.
[2]Professor and Head, Department of ECE, Karpagam Academy of Higher Education, Coimbatore.

Abstract

Checkpoint (CP) routing requires a procedure for heavy load and dynamic routing packets over a cluster network, by monitoring network load. High traffic CP network does intense transactions by multiple applications and shares the information among the nodes in the network and delivers the same to the server. The Proposed work assigns time slots and each slot checks the path to send or migrate the data to manage the load and data accumulation between servers and if there arises any data loss it recovers from the previous node buffers. Data rate and delay is checked by each link and adjust the data flow to various servers. Based on time, network state, incoming packet rates, server load, link-based migration is applied. It increases network packets receiving rates, and minimizes the delay in checkpoints.

Keywords: *Checkpoint, Data Migration, Data loss, Load, Link Capacity, Receiving Rate.*

1. Introduction

Our aim is to create optimized checkpoint secure network of high performance. It minimizes the communication issues of inter node and intra node by grouping the nodes and controlling the hop counts from source to destination. In network, nodes may join or leave the communication at any time. Data administration in such a big environment is highly significant. When transmission load increases imbalance takes place in a network, or if node moved out of range can create big loss in network transmissions.

In natural world communications, network devices are apt to be heterogeneous. On the platform for checkpoint communications these devices can be connected for mutual processing. Contest for transmission resources does not lead to assured good output.

2. Literature Review

The qualities like element topology, impediments in vitality asset, stockpiling gadget and correspondence channel debilitate the examination group to grow more secure framework to keep the client from information adversity and dependability[10]. To recover from saved checkpoint which is saved in another cluster head, the checkpoint needs to be transferred to the cluster head in which the failed host will recover[6]. A packet from this optimal commodity is transmitted, the responsibility of forwarding the packet to its destination is shifted to the receiver node that maximizes the differential backlog[1]. To support backbone architecture, the Cluster Heads should be a part of the backbone and the fewer the number of backbone nodes the better. Fewer nodes in the backbone can reduce the quality of messages exchanged by backbone nodes[5].

The cloud will then monitor the connectivity of the nodes to notify them when both node states change from disconnected to active[9]. An attacker may attempt to make a route by itself to appear longer by adding virtual nodes to the route[4]. A new computing environment is created in which some hosts are mobile computers connected by wireless communication networks and some are stationary computers connected to a fixed network[8]. In addition, network coding requires each node to maintain more queues and our routing solution at least reduces the number of queues to be maintained for routing purposes, thus partially mitigating the problem[3]. The node transmit a message to another node which is out of range, the cooperation of other nodes in the network is required; it is represented by multi-hop communication[7]. We have more-over implemented this mechanism in TinyOS and experimentally verified its performance over a 25 node sensor network testbed, demonstrating that it provides good network performance in the presence of maliciously compromised nodes, while inducing minimal computation overhead[2].

3. Implementations of Resources Handling in Checkpoints

Executions can be transferred to high power devices to look for quicker computation or richer quality processing services. In this, a process can transfer from a device to a server with higher processing speed and dynamic memory to resolve computation centric issues. A technique to checkpoint an execution in the heterogeneous network could also be helpful for error tolerance. Checkpoint allows the process of the transmission to start again from a saved execution process level rather than its initial stage. So the execution can be renewed at another device in a scattered environment, this controls the error tolerance which can be attained by either restarting the execution at the existing device after the error is resolved or restarting at a new device. All network transmissions of checkpoint network are managed and handled without infrastructural holdup and vulnerable to security becomes an intrinsic weakness. Nodes located within communication range which accesses the shareable channel can be infected by attackers easily. Secure connection between nodes needs to be incorporated into the devices. To establish secure connection node should be able to find nearby nodes initially with identical bandwidth, location energy, and queue size to construct the neighbor list.

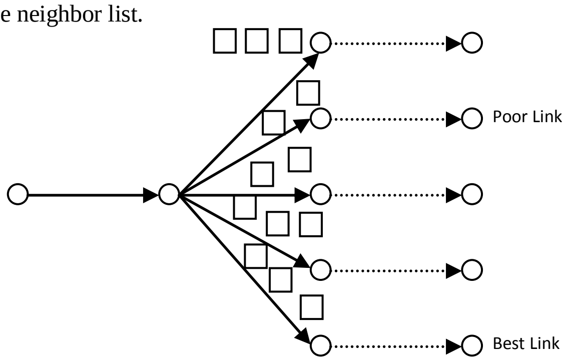

Fig. 1: Buffer accumulation state

Implementation

Checkpoint routing submits a protocol for changeable traffic over a cluster network due to overloading ascent. The high volume data handling server based network, where big data streams appear to an environment must be transferred to the appropriate server. The protocol works under cluster-based environment with security in allocated time, and in each slot it looks for the available data and its duration to route data in selected group head (GH) directions . GH sending rates can be computed to know the balance of GH. Features of the implementations are in the strong time, measuring network can achieve high bits per second based on the z data arrival rates and channel limit.

In CP network group communication-based channel usage with dynamic data arrivals and different GH selection as main GH and secondary GH is considered.

4. Group-Based Communication

Consider a group based network with a set of nodes. The devices work in time t based allocation TA and t∈(t0,.. tn). On each TA, routing and data transmission TA judgments are made in an attempt to transport all packets to its decided server. Nodes update new data at every TA period. Each server node holds inner memories that keep information as per its restriction. Let $M^{(I)}_{node}(t)$ be the presently available data in the server, also said as the memory accumulation. The memory accumulations from one TA to the consecutive TA computed as,

$$M^{(I)}_{node}(t)<=\max K \quad (1)$$

$$K= (M^{(I)}_{node}(t) – TxRate) + NewD_{node}(t) \quad (2)$$

Where $NewD_{node}(t)$is the dynamic data arrival to the node at each t, and TxRate is the sending rate assigned to the channel (GM, GH) during t. When the TxRate(t) is higher than the channel band width and the bandwidth is more than the capacity of the channel between (GH, GM) at a particular t computation of max[] task is needed else no computation required. In such case there may not be sufficient data in GM. To check the dissimilarity between GH and GM, $\sum^{S}_{x=1} {}_{E}^{(d)}$ (t) may be surplus data than the real appearances on t and this is determined based on the incoming extra data. As stated by this, presume that $M^{(I)}$ (t) = 0 for all TA in t and all information ∈GM, thus no GM memory holds data. Vibrant data arrivals updates as New $D_{node}(t)$ which is kept as matrix of receiving data and this matrix is parallel shared (x,y,z) over TA.

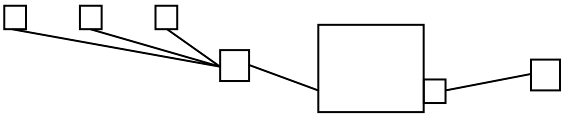

Fig. 2: Packet flow to server

The CP might have periodical-unreliable channels and/or communication state. Let T stand for the network state on time t, which keeps the fine points related to the GM to GH sending rates that is supported by the present time. The Central server (CS)monitors T(t) at the commencement of each time t, and selects a matrix (TxS (t)) of sending speed based on T

$$(TxS(t)€ Tcs \quad (3)$$

Where, Tcs is the network supporting state along with orthogonal frequency division multiplexing OFDM links and public channel nosiness with known manageable sending speeds without sending faults. Each time t, the CS selects new TxS, and finds routing limitations RL:

$$(\Sigma RL < TxS(t) \quad (4)$$

Once these conclusions are complete, the queue updates start.

In T over the time slots with likelihood LH = TxS[T(t) = LH].The CP network server S capability CB is based on the incoming rate ($\lambda S^{(CB)}$) monitored by T(t) and at each time t select sending rates ($RL^{(cB)}$) and path dynamics TxS(t) as per limitations of all buffers. CP GH and S(server) complete buffers demands based on the received packets rate (PR) that are identical tothe entire data transport to its target. It is frequently helpful to suppose that (server capability)$\lambda S^{(CB)}$ is center to GH, so that $\lambda S^{(CB)}$ computed among GH and GM.

$$PR = \lambda S(\text{CB}) + RL \quad (5)$$

As a practical necessity, the sending rates TxS(t) need to be monitored among the CP network devices ‘heaviness

$$GH\ \delta\boldsymbol{\Sigma}\ BA(t) \quad (6)$$

In CP, network virtually migrate the transmission and routing packets as per the queue and storage capacity and BA. In this stage acquisitively (greedily) minimize an area on the float specifically to describe B(t) = ($B_n^{(CB)}$ (t)) as the table of present buffer accumulation BA. Compute total buffer accumulation with CB since $B_{CB}^{(CB)}$ (t) = 0 for all nodes and time t.

The changing float f(t) is defined:

$$f(t) = PR(H(t+1) – hBA) \quad (7)$$

Find the maximum of buffer free servers to migrate data at time t as (max[BA]) and choose under load servers at time t

$$\boldsymbol{\Sigma}\lambda S^{(CB)}PR + \boldsymbol{\Sigma}TxRate \quad (8)$$

The CP network handles network state and server buffer state at each time depending on network load adjusting TxRate to control the load limit.

We can make the most of the expectation through increasing the load in a server as per the virtual transactions and load. That is, need to monitor network state and TxRate(t)) to increase

$$TxRate\ [S1 –S2] \quad (9)$$

Where S is the server 1 and 2, and compute server load by

$$\boldsymbol{\Sigma}BA1 –BA2 \quad (10)$$

It gives BA state at each server which is called the load among servers. To mange load checks the TxRate to select the migration data to servers. Already network uses clustering path to reach the destination servers. Paths are updated by each device dynamically as primary and secondary cluster heads, and dynamically can choose depending on network parameters. Network delay may come because of the network load and link processing.

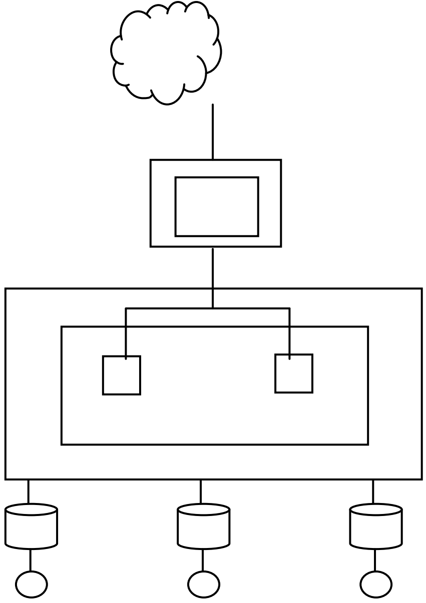

Fig. 3: Transmission medium

5. Simulation Analysis

The SOBM is implemented using the tool NS2.Network is constructed with 50 nodes. Topology size is considered as 500 x 500 square meter with the packet size of 512 bytes. CBR traffic source is used. The following parameters is analyzed by four protocols. The results are highly reliant on state of network. The simulation time and the data transmission time depend on the intervals assigned. Simulation preparation, timings is first 15 seconds then the data will start.

6. Results and Discussion

Here we present the stable cluster network in checkpoints. Because, the network monitors the state changing and the TxRate periodically to select decisions to transfer the data to the server. These conclusions provide dynamic output. Also the incoming rates $\lambda S^{(CB)} \in \Lambda$ incomings.

The packet Delivery Ratio shows the best outcome in SOBM than the available protocols. Because of optimized server migration selection, Analysis of multiple parameters like data rate, network state, and incoming packets SOBM shows the best ratio of receiving.

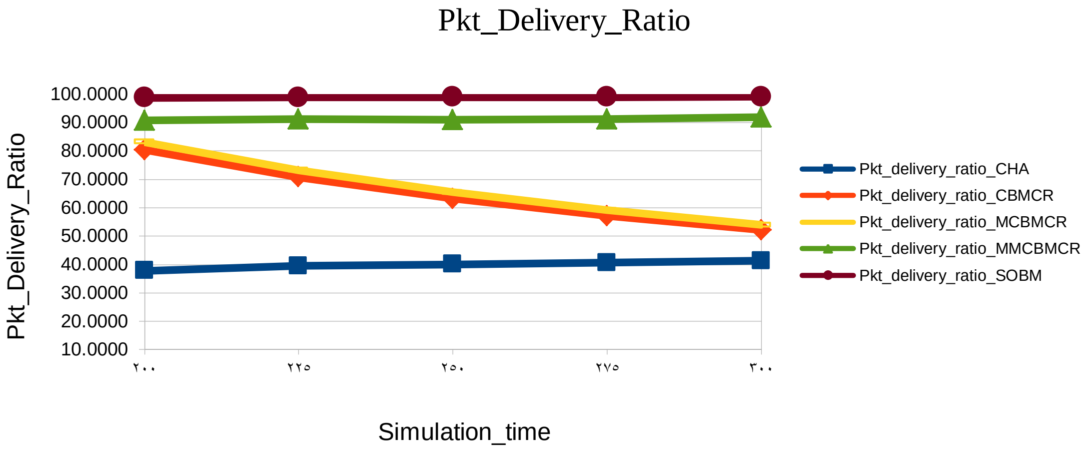

Fig. 4: Packet Delivery Ratio based on Simulation Time

Received packets in bits per seconds computed as throughput. If protocol shows high throughput means best outcome resulting in performance. SBOM shows the high throughput as result because of multiple analyses. Storage capacity monitoring handles the node filling state and according to that data migration controls the packet loss [12].

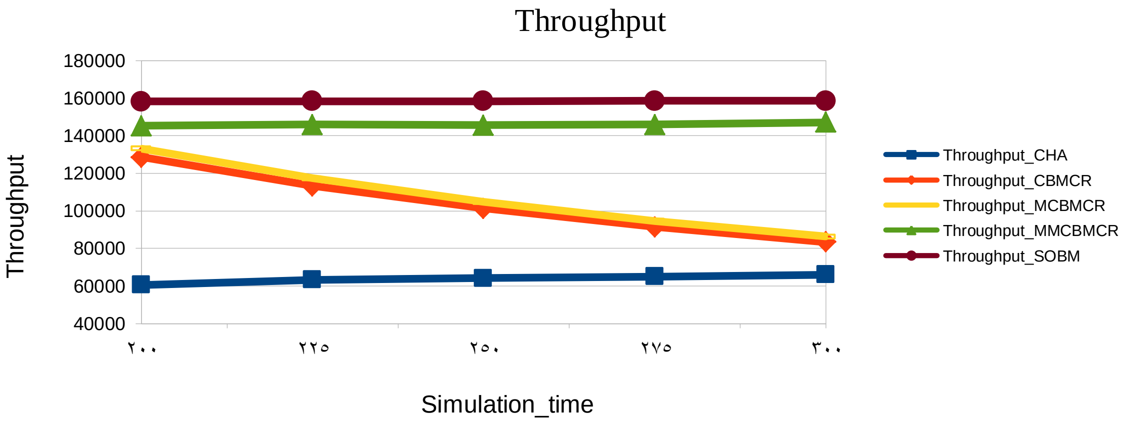

Fig. 5: Throughput based on Simulation Time

Delay computed as node processing time, buffering time, transmission time. All together is taken as final delay [11]. Delay minimized in protocol shows best path selection and data transmissions processed. SBOM shows the minimum delay than the other available protocols.

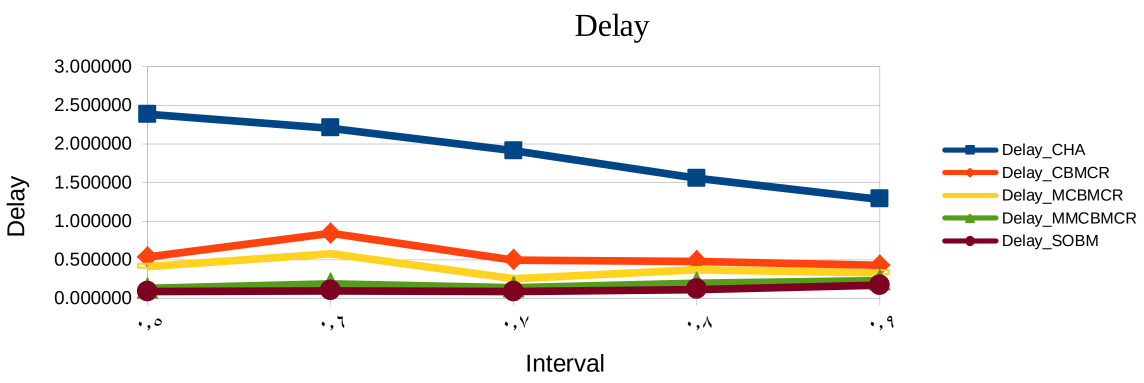

Fig. 6: Delay Vs Interval

7. Conclusion

SOBM has been exposed to work in combination of transmission rates and network state. It allots the time slots and each slot checks the path to send or migrate the data to manage the load and data accumulation between servers and the data losses recovered from the previous node buffers. Data rate and delay checked by each link and adjusted the data flow to various servers. Based on time, network state, incoming packet rates, server load, a capacity of link-based migration was applied. It increased network packets receiving rates, and minimized the delay in checkpoints. This conclusion meanly minimize a loss and delay in huge server network.

References

[1] Neely MJ & Urgaonkar R, “Optimal backpressure routing for wireless networks with multi-receiver diversity”, *Ad Hoc Networks*, Vol.7, No.5, (2009), pp.862-881.

[2] Venkataraman R, Moeller S, Krishnamachari B & Rao TR, “Trust based backpressure routing in wireless sensor networks”, *International Journal of Sensor Networks*, Vol.17, No.1, (2015), pp.27-39.

[3] Bui XL, Athanasopoulou E, Ji T, Srikant R & Stolyar A, “Backpressure-based packet-by-packet adaptive routing in communication networks”, NJ, USA, (2012).

[4] Chayal D, “Assessment of Security in Mobile Ad-Hoc Networks (MANET)”, *Journal of Global Research in Computer Science*, Vol.2, No.6, (2011), pp.137-139.

[5] Tuli R & Kumar P, “Minimum process coordinated Check pointing scheme for ad hoc Networks”, *International Journal on AdHoc Networking Systems (IJANS)* Vol.1, No.2, (2011), pp.51-63.

[6] Biswas S, Neogy S & Dey P, “Mobility based check pointing and trust based recovery in MANET”, *International Journal of Wireless & Mobile Networks*, Vol.4, No.4, (2012), pp.53-69.

[7] Karamjeet S & Chakshu G, “Using MD5 AND RSA Algorithm Improve Security in MANETs Systems”, *International Journal of Advances in Science and Technology (IJAST)*, Vol.2, No.2, (2014).

[8] Patial S & Thakur J, “Check pointing and Rollback Recovery Algorithms for Fault Tolerance in MANETs: A Review”, *International Journal of Advanced Networking and Applications*, Vol.6, No.3, (2014), pp.2308-2313.

[9] Alshareef HN & Grigoras D, “Robust cloud management of MANET checkpoint sessions”, *Concurrency and Computation: Practice and Experience*, Vol.29, No.2, (2017), pp.66-73.

[10] Narayana MV, Narsimha G & Sarma SSVN, “SHA-ZHLS: Security Enhancement in MANETs using SHA Algorithm”, *International Journal of Advanced Research in Computer and Communication Engineering*, Vol.5, No.2, (2016), pp.59-65.

[11] G Ainabekova, Z Bayanbayeva, B Joldasbekova, A Zhaksylykov (2018). The author in esthetic activity and the functional text (on the basis of V. Mikhaylov’s narrative (“The chronicle of the great jute”). Opción, Año 33. 63-80.

[12] Z Yesembayeva (2018). Determination of the pedagogical conditions for forming the readiness of future primary school teachers, Opción, Año 33. 475-499